SPACE ENVIRONMENT TECHNOLOGIES

SET TR 2008-001

MgII c/w Ratio Data Processing for SET Operations

S. Dave Bouwer

July 2008

SET Technical Report SET_TR2008-001

MgII c/w Ratio Data Processing for SET Operations

S Dave Bouwer, Space Environment Technologies

July 12, 2008

MgII c/w Ratio Data Processing for SET Operations

12 Jul 2008, Dave Bouwer, SET

Table of Contents

Introduction

The new MgII processing software is designed to be a comprehensive collection of Java and IDL software that retrieves and processes MgII data from remote web/ftp sites, storing the data in a mySQL database for subsequent data processing and retrieval, and creating the legacy file `~/operations_folder/time_series_data/NOAAMgII.dat` for S2K ops to use. One of the key design considerations is that there will be a number of MgII values for a particular day, with a number of sources, quality, versions, and so forth. The database is designed to store all this information for each record so simple queries can target specific MgII values with certain attributes. For example, one could select all non-missing MgII values for 2007 from NOAA-16 with the sql command: "`select year doy from MgII.mg2 where mg2 > 0 and year >= 2007 order by time_key asc`". Unlike traditional programming methods, multiple files do not have to be maintained, the data is retrieved nearly instantly, can easily target specific data records, and allows traceability of changes. The database uses a primary `time_key` in a YYYYMMDDHHMMSS format. Every time a data record changes (i.e., MgII value, source, level, version, processing programs), the SS in the `time_key` is incremented by one, uniquely storing a new record. The software will ignore attempts to store a duplicate record, but permits multiple records for the same day to be stored. Subsequent classes retrieve data from the database, processing multiple records for a day to select the "Best of 4" MgII value for the day.

This document details the software and data for all future reference.

Inputs

LASP

- `ftp laspftp.colorado.edu, sorce_solstice_mg2_index.txt`
- Satellite is SORCE/SOLSTICE.
- Cadence is daily.
- Latency is generally several days, missing days are rare, if present at all.
- Quality is good, but its daily values are "noisy", unlike the smooth day-to-day variations of NOAA16,17 and GOME. Its long-term calibration is considered stable.

SWPC NOAA

- `http://www.swpc.noaa.gov/ftpdir/sbuv/NOAAMgII_Last_30_Days.dat` (NOAA-16, not used after 1 Jun 08, but it is still stored for possible later use)

- `ftp://helios.swpc.noaa.gov/pub/sbuv/NOAAMgIIUnix.dat` (NOAA-17, begins 09 Jun 06)

NESDIS

- ftp: `dds.nesdis.noaa.gov`
- Level 0, last daily scan: `PRD.OZONE.1B.N17.DYYdoy.Shhmm.Ehhmm.Bxxxxx`
- Level 2: `GOME_M02_MGII_YYYYMMDD`

Outputs

- `NOAAMgII.dat`: A concantenation of monotonically-increasing daily MgII values in a text file table, related dates, and a single index number identifying the satellite source (4 for GOME, 5 for Solstice, 6 for NOAA-16, 7 for SWPC NOAA17, 8 for NOAA/NESDIS NOAA17). A file of daily values may contain the current days (determined from computer clock) predicted value using the IDL ts_forecast function.
    - `/spacewx/dataIO/mg2/NOAA_MgII_78_current.txt` (copied to:)
        - `/spacewx/s2kops/main/operations_folder/time_series_data/NOAAMgII.dat`
- Files above look like:

```
:Data_List: /spacewx/dataIO/mg2/NOAA_MgII_78_current.txt version v3.1
:Created: 2008-07-12 17:06:56
:Updated_by: Concantenate_NOAA_Mg2files
:Number_of_Data: 11097
# Metadata:
# Data Source: TIROS/NOAA9 1, NOAA11 2: NOAA9 alternate algorithm 3: SOLSTICE 4:
GOME 5: NOAA16 6: NOAA17 7: NESD17 8: NOAA9 9: missing 0
# Year Month Day Date   MgII   Source
#-----------------------------------
 1978 11 07 11/07/1978  0.2772   1
 1978 11 08 11/08/1978  0.2770   1
```

- Other "overwritten" files in /spacewx/dataIO/mg2/ are:
  - `NSD17_Mg2_data.txt` NOAA-17 from NESDIS, since 06 Mar 08
  - `NOAAMgIIUnix.dat` The NOAA17 file from SWPC
  - `NOAAMgII_Last_30_Days.dat.txt` N16 from SWPC (deprecated)
  - `sorce_solstice_mg2_index.txt` Downloaded from LASP
  - `mg2_est.dat.txt` The best MgII values from 1 Jun 08 to today
  - `Mg2_adj_predictor_plt.jpg` The plot of `mg2_est.dat.txt`
  - `GOM_Mg2_data.txt` The past 81 days of GOME
  - `SOL_Mg2_data.txt` The past 81-days of Solstice
  - `NSD17_Mg2_data.txt` The past 81-days of N17 from NESDIS
  - `N17_Mg2_data.txt` The past 81-days NOAA-17 from SWPC
  - `N16_Mg2_data.txt` The past 81-days of N16
  - `Mg2_allSats_plt.jpg` The plot of all input data for past 81-days
  - `/Users/s2kops/Sites/data/`
    - `mg2_atmos.dat.txt` Plot of all atmos. MgII output data since 1978
    - `NOAAMgII.dat.txt` Plot of all sol. MgII output data since 1978
  - `/Users/s2kops/Sites/current_files/`
    - `Mg2_adj_pred_atmos_plt.jpg`
    - `Mg2_adj_predictor_plt.jpg`
    - `Mg2_allSats_plt.jpg`
    - `Plot_NOAAMgII_dat_plt.jpg`
    - `cur_nesdis_N17_mg2.jpg`
- `/spacewx/dataIO/mg2/Atmos_MgII_78_current.dat.txt`, to be verified and to replace : `/spacewx/s2kops/main/operations_folder/time_series_data/NOAAMgII.dat`

- File above look like:

```
:Data_List: /spacewx/dataIO/mg2/Atmos_MgII_78_current.txt version v3.1
:Created: 2008-07-12 17:06:57
:Updated_by: Concantenate_Atmo_Mg2files.java
:Number_of_Data: 10841
:Missing_value_flag: -999.0
# Metadata:
# Data Source: 1TIROSx : SWPC composite, NOAA TIROS and NOAA 9
# Data Source: 2NOAA11 : SWPC composite, NOAA 11
# Data Source: 3NOAA9x : SWPC composite, NOAA 9  (Alternate Algorithm)
# Data Source: 4SOLSTa : SWPC composite, UARS SOLSTICE
# Data Source: 5GOMEax : SWPC composite, GOME
# Data Source: 6NOAA16 : SWPC composite, NOAA 16
# Data Source: 8NOAA18 : SWPC composite, NOAA 18
# Data Source: 9NOAA9b : SWPC unassigned
# Data Source: ADLnd17 : DeLands NOAA-17 (Jun 2008)
# Data Source: 7NOAA17 : SWPC NOAA 17    Best-Of-4 Atmosphere version
# Data Source: ASOLSTB : Solstice,       Best-Of-4 Atmosphere version
# Data Source: ANESD17 : NESDIS NOAA 17, Best-Of-4 Atmosphere version
# Data Source: AGOMEbx : NESDIS GOME     Best-Of-4 Atmosphere version
# Data Source: ASET40x : From DCA v4_0 from SETops, (Jun 2008)
# Data Source: Absentx : No satellite ID available
# Data Source: 0missng : No MgII data available
#File verified as monotonically increasing
#YYYY MM DD JD            MgII     Source
#---- -- -- ------------- -------- -------
 1978 11 07 2443820.00000 0.277200 1TIROSx
 1978 11 08 2443821.00000 0.277000 1TIROSx
```

MgII Operations

The daily operations of MgII involve a cron job (Mg2_DB_main.java) that is run twice daily (e.g., 6 and 18 UT). The list of the sequence of steps and related data files, and the flowchart below illustrates the major components.

```
1  start Mg2_DB_main.main() to execute all relevant classes and external IDL
programs

2  get and store Solstice data
(/spacewx/dataIO/mg2/sorce_solstice_mg2_index.txt)
      2.1 Execute FTP_getSolstice_mg2dbase.java
      2.2 Call Mg2_DB_main.read_store_SOLSTICE_file()
         2.2.1 Instantiate ReadStore_Mg2file.java
            2.2.1.1 call ReadStore_Mg2file.read_SOLSTICE_Mg2file()
            2.2.1.2 call ReadStore_Mg2file.store_records()

3  get and store GOME data
(/spacewx/dataIO/mg2/nesdis_gome/GOME_M02_MGII_YYYYMMDD)
      3.1 Execute FTP_get_NesdisGome.java
      3.2 Call Mg2_DB_main.read_store_GOME_file()
         3.2.1 Instantiate ReadStore_Mg2file.java
            3.2.1.1 call ReadStore_Mg2file.read_GOME_Mg2file()
            3.2.1.2 call ReadStore_Mg2file.store_records()

4  get, process (using the refactored Matt DeLand algorithm), and store most
recent daily NESDIS NOAA-17 data
(/spacewx/dataIO/mg2/nesdis_n17/PRD.OZONE.1B.N17.DYYdoy.Shhmm.Ehhmm.Bxxxxx)
      4.1 execute FTP_get_NesdisN17.java
      4.2 execute Update_refl_coeffs.java
      4.3 execute nesdis_mg2ops_noaa17.pro (creates orbital
/spacewx/s2kops/mg2dbase/mg2ops_data_dock/*.unf files)
      4.4 execute N17_mg2_daily.pro (creates a
/spacewx/s2kops/mg2dbase/mg2ops_data_dock/N17_Mg2_daily_vals.txt file)
      4.5 execute N17_mg2_daily_plot.pro (creates a
/spacewx/s2kops/mg2dbase/mg2ops_data_dock/cur_nesdis_N17_mg2.jpg)
      4.6 call Mg2_DB_main.read_store_NESDIS_N17()
         4.6.1 Instantiate ReadStore_Mg2file.java
            4.6.1.1 call ReadStore_Mg2file.read_NESDIS_N17_Mg2file()
            4.6.1.2 call ReadStore_Mg2file.store_records()

5 get and store SWPC data
      5.1 call Mg2_DB_main.get_SWPC_Mg2files()
         5.1.1 Instantiate Get_NOAA_Mg2files.java
         5.1.2 call Get_NOAA_Mg2files.run_urlGets()
      5.2 call
Mg2_DB_main.read_store_SWPC_Mg2files(/spacewx/dataIO/mg2/NOAAMgII_Last_30_Days.
dat.txt), NOAA-16
            5.2.1 call ReadStore_Mg2file.read_NESDIS_N17_Mg2file()
            5.2.2 call ReadStore_Mg2file.store_records()
      5.2 call
Mg2_DB_main.read_store_SWPC_Mg2files(/spacewx/dataIO/mg2/NOAAMgIIUnix.dat),
NOAA-17
            5.2.1 call ReadStore_Mg2file.read_NESDIS_N17_Mg2file()
            5.2.2 call ReadStore_Mg2file.store_records()
```

```
6  Execute
Adj_mg2_bestOf4("20000101","/spacewx/dataIO/mg2/","mg2_2000_to_current.txt")
      6.1 Instantiate Adj_Mg2_Sol(NESD17,NOA17,GOME)
         6.1.1 call LinearRegression()

7  Execute Report_Sat_mg2.java (creates
/spacewx/dataIO/mg2/NOAAMgII_Adjusted.dat.txt);

8  Execute Mg2_adj_predictor.pro (creates /spacewx/dataio/mg2/mg2_est.dat.txt,
/spacewx/dataio/mg2/Mg2_adj_predictor_plt.jpg)

9  Execute Plot_Mg2_allSats.pro (creates
/spacewx/dataio/mg2/Mg2_allSats_plt.jpg)

10 Execute Concantenate_NOAA_Mg2files.java
      10.1 read_inpFile() (/spacewx/dataIO/mg2/NOAAMgII.1978-99dat.txt +
mg2_2000_to_current.txt -> NOAA_MgII_78_current.txt)
      10.2 adj_buff3data() (/spacewx/dataIO/mg2/mg2_est.dat.txt ->
/spacewx/dataIO/mg2/NOAA_MgII_78_current.txt)

11 Execute CopyTextFile(/spacewx/dataIO/mg2/NOAA_MgII_78_current.txt to:
/spacewx/s2kops/main/operations_folder/time_series_data/NOAAMgII.dat)

12 Execute FileDeliver (copies
/spacewx/s2kops/mg2dbase/mg2ops_data_dock/cur_nesdis_N17_mg2.jpg to:
/Users/s2kops/Sites/data/cur_nesdis_N17_mg2.jpg)

13 Execute manage_Mg2_files(cleans:
/spacewx/dataIO/mg2/nesdis_gome/,/spacewx/dataIO/mg2/nesdis_n17/,/spacewx/s2kop
s/mg2dbase/mg2ops_data_dock/)
```

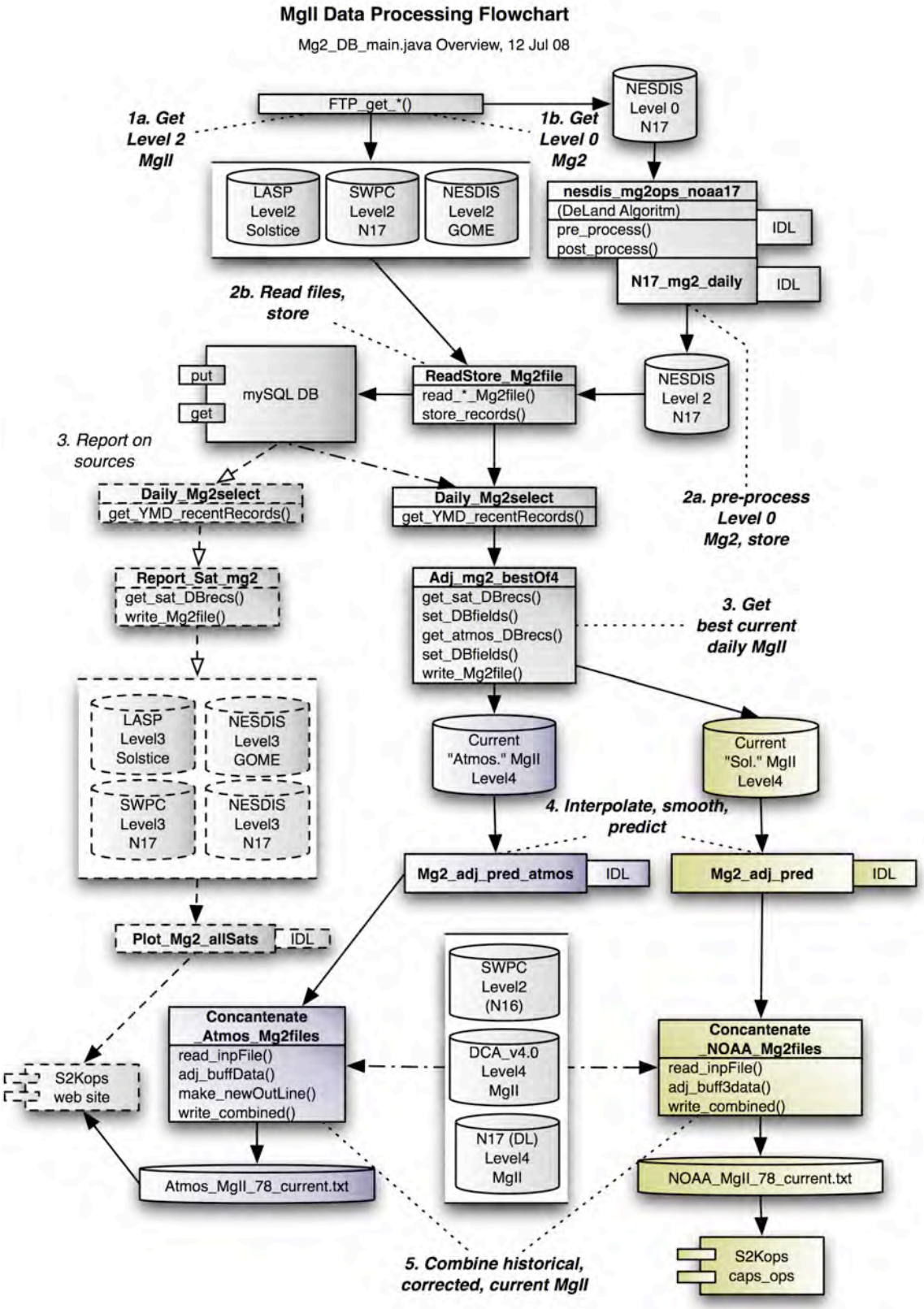


Figure 1. Flowchart of the processing of the MgII data from all sources.

Directory Structure

`/spacewx/s2kops/mg2dbase` Where all the code is kept.
`……./coeffs` Important files for NESDIS N17 processing.
`/spacewx/dataIO/mg2` Where current and plot IO data are kept.
`……./nesdis_gome` Where NESDIS GOME files are downloaded.
`……./nesidis_n17` Where NESDIS N17 data files are downloaded.
`/Users/s2kops/Sites/` For access to `http://spacewx.net/~s2kops/`
`……./nowcast` Assorted current data and plots.
`……./data (NOAAMgII.dat.txt)`
`……./current_files` Assorted current data and plots.

The "Best of 4" Algorithm

One persistent problem in the past was the unreliability of retrieving the MgII data from SWPC. Data would come in with bad values, days late, etc. After several attempts at "patching" the code, it became clear that a more robust solution was needed. After considerable effort, The LASP Solstice MgII data was added (which also has its problems with latency and noisy day-to-day values), code was written to retrieve the NOAA17/SBUV raw telemetry data from NESDIS (and Matt Deland's algorithm was re-written for daily operations), and the GOME MgII data at NESDIS was added. A "Best of 4" algorithm was written to go down decision tree to use one of the satellites, depending on what MgII value was available for the day (See flowchart below). The algorithm is designed to always fall back to either an existing MgII value, for any time, or use a predictive algorithm to always produce a current daily value. It does not depend on any file being up-to-date, retrieving data from the database instead. Any two satellites can be off-line for a requested day, although there needs to be at least 5 days (preferably a lot more!) from Solstice in the past 81 days to perform a linear regressing between another satellite source, if the "Sol" version of the algorithm is used.

Note that the algorithm has methods applied that, prior to performing a linear regression (requires numbercruncher/mathutils/ java lib), it removes outliers from arrays by rejecting points more than +-2 standard deviations from the mean of the past 27 days. It then interpolates missing data. Next it casts a prediction from the tail end of the data vector until the current day, if the data is not current to the day.

For the "Sol" version (current the operational one), the NOAA/NESIS NOAA-17 MgII values are used for all days from 1 Jun 08, and are calibrated (via a linear regression) to the past 81-days of Solstice data. Other data sources are used in the event the NESDIS/NOAA-17 is not available.

For the "Atmosphere" version (will eventually be the operational one), the NOAA/NESIS NOAA-17 MgII values are used for all days from 1 Jun 08, and are calibrated (via a linear regression) to coefficients derived from a linear regression with the MgII data created by Kent from the DCA v4.0 file (see `plt_mg2_spcl.pro`) . Other data sources are used in the event the NESDIS/NOAA-17 is not available.

A few notes on NOAA SBUV satellite data:
The NOAA-16 values became increasingly unreliable in late 2007, going off line from about January 24, 2008, until it came back online April 4, 2008. But from private communication with Matt DeLand, it appears the angle to the SBUV diffuser plate on NOAA-16 is getting so narrow that good MgII values are hard to reliably get, and the problem will get worse. That, and the fact the NOAA-16 instrument is getting quite old resulted in the decision to stop using the NOAA-16 data for S2K operations after 1 Sep 07, and rely on the NOAA-17 data (although orbital dynamics will result in it having the same diffuser plate narrow angle eventually as well). NOAA-16 is still stored in the database however, for possible future use. The NOAA-18 instrument failed in May 2008, so it looks like that instrument will not be available in the future. Obviously, it will be important to stay current on all SBUV satellite calibrations.

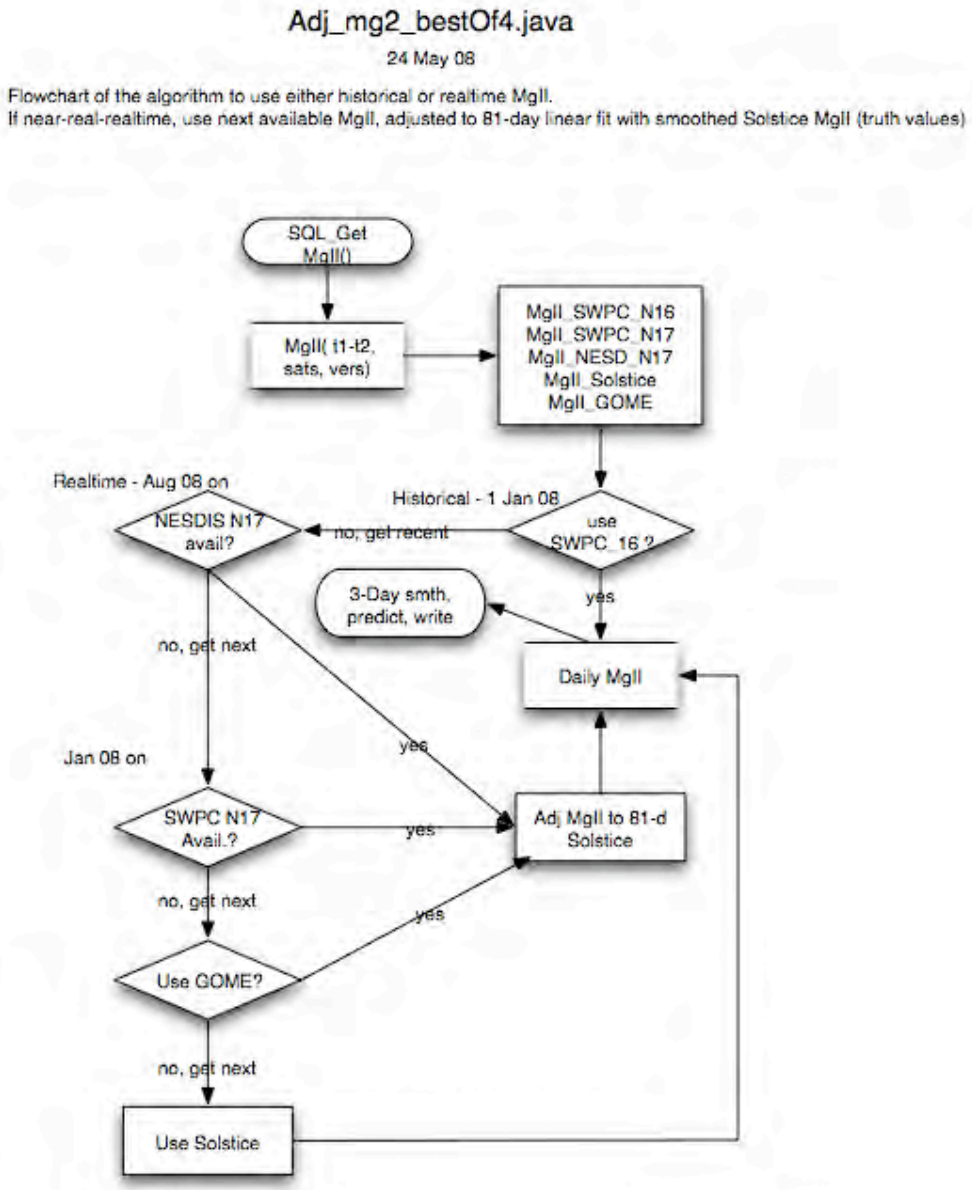


Figure 2. The "Best of 4" algorithm for the sol version. The atmosphere version uses coefficients from the NOAA-17 to the DCA v4.0 of MgII.

Data Schema

The mySQL MgII.mg2 database is designed to maintain a variety of data sources, processing levels, and other fields in a record. It is expected that Mg2 values will change over time, as corrections are made, new "rules" are applied (such as which satellite to use on a particular day). The data is stored in the MgII database, under the mg2 table. The entire database can be deleted and re-created from comprehensive data files; this takes about an hour because of the large volume of data, and very large data sets may exceed memory capacity so they need to be broken into smaller files. To re-create a database, the following statement must be used (this also describes the variables and types for accessing the data records and fields in the mySQL database):

```
CREATE TABLE mg2( time_key char(14) primary key,
    mg2 float default -999.0 not null,
    lya float default -999.0,
    jd double default 0,
    year int default 0,
    doy int default 0,
    quality int default 0,
    level int default 0,
    current char(1) default "f",
    inputfile char(80) default "na",
    institute char(8) default "na",
    source char(8) default "na",
    version char(8) default "na",
    rules  char(80) default "na",
    created char(19) default "0000000000000",
    modified char(19) default "0000000000000",
    preprocessor char(40) default "na",
    processor char(40) default "na",
    postprocessor char(40) default "na",
    accessed char(19) default "0000000000000",
    accessor char(40) default "na",
    todo char(40) default "na",
    outputfile char(80) default "na",
    note char(40) default "na",
    valid char(1) default "f",
    verif char(1) default "f" );
```

At the time of this writing, some fields are unused in the code or just have placeholders, and their definition and use expanded: quality, level, current, inputfile, todo, outputfile, quality, note, verify, etc. The version field needs to be updated in Config_mg2.java when there is a significant change that affects the MgII values. Also maintain the `/spacewx/s2kops/mg2dbase/Mg2_Rules.txt` file. This field is not used in the code, but it will increasingly prove valuable in verifying the MgII values. At this writing, it contains:

```
# Rules for MgII records stored in the rules field in the
MgII/mg2 mySQL database
:New_SQL_Record_Rule: 10_Nov_07 V2.0 D_Bouwer
#   New Record generated in every time any of the following
changes;
#   Mg2_value, data_level, data_source, data_vers, prepro_prog,
process_prog, postpro_prog.
:BestOf4_Rule: 30 May 08 V2.0.b D_Bouwer
#   Class Config_mg2->N16_lastValJD(1Jan08)    = 2454466.0, After
this, NOAA-16 is assumed bad, GOME preferred to SWPC NOAA-17
#NESDIS-17 preferred to SWPC NOAA-17.
#
Prior to this, all GOME data considered equivalent to NOAA-16
#   Class Config_mg2->GOM_lastValJD(12Mar01)   = 2451981.0  Prior
to this, all GOME data considered equivalent to NOAA-16
# After 1 Jan 2008, NESD17 used first, NOAA17 next, GOME next,
all calibrated to Solstice
```

Java Classes

Java is the main programming language used, but with crucial components done in IDL. All the java classes, IDL programs, and working files lie under the `/spacewx/s2kops/mg2dbase/` directory. All data IO files (with the exception of the final `/spacewx/s2kops/main/operations_folder/time_series_data/NOAAMgII.dat` for use by the S2Kops software) reside under the `/spacewx/dataIO/mg2/` directory.

Note that all Java classes have an associated javadoc HTML page for documentation. Refer to each classes javadoc page for reference (use your browser to load the file: `file:///spacewx/s2kops/mg2dbase/index.html`). Also note that for many of the Java classes to operate, the SQL J-connector library for connecting to the database must be in the compile/runtime classpath (`/lib/mysql-connector-java/`). And, to use an ftp library to connect to the remote ftp sites, `/lib/sftp/lib/sftp.jar` must be in the classpath.

Nearly all fixed parameters used by the Java code are specified in the Java classes `Config_mg2.java`. All directory paths, fixed numeric values, etc. are set in this class, making it easy to change parameters without changing a lot of code. However, the IDL programs have hard-wired paths (consistent with the Java classes) that will need to be edited if there are changes.

A critical Java class is `MgII_record.java`. As soon as a daily MgII value is defined – from any source – an instance of a MgII_record is created. It contains a large number of attributes (the Julian Day, source, institute, date, etc) unique to each record, in addition with methods to validate a MgII value, convert it to Ly-a, etc. There can be a MgII_record instance for a day with missing-value flags. These MgII_record instances are stored in Vectors, passed between classes, stored and retrieved in the database, etc. In a way, it is the "blood-cell" of all the paths the data travels in the processing.

Other important classes:

- **`Adj_mg2_bestOf4`** Gets a vector of Mg2 records from the MgII/mg2 database. Can be used for historical or realtime applications. For each day, it first uses the NOAA/NESDIS N17 values if available, then SOLSTICE, then the SPWC NOAA17, and lastly the GOME, then adjust the current value from a 81-day linear regressing with Solstice.
- **`Adj_Mg2_to_Sol`** Gets a vector of Mg2 records from the MgII/mg2 database for two sets of Mg2 indices, one of which is Solstice, the other "source" (NOAA-17, GOME, etc). Performs a linear regression that creates Mg2 estimates from Solstice, providing the two coefficients for adjusting a source to an equivalent Solstice value. It adjusts for any missing values between the two satellites.
- **`Daily_Mg2select`** Retrieve a daily Mg2 record for the day. Because multiple mg2 records may exist for the same day, this class provides several methods to

retrieve a record on the basis of mg2val non-missing, the most recent mg2 value, or a mg2 value from a specific satellite.

- **`FTP_get_NesdisGome`** Downloads GOME MgII files from NOAA/NESDIS.
- **`FTP_get_NesdisN17`** Downloads NOAA-17 MgII files from NOAA/NESDIS.
- **`FTP_getSolstice_mg2dbase`** Downloads SORCE/Solstice files from LASP.
- **`Get_NOAA_Mg2files`** Downloads two (NOAA-16, NOAA-17) MgII files from SWPC.
- **`Mg2M10Conversion_v2_0_a`** Converts the NOAA SEC Mg2 data to a M10 value (corrects for instrument degradation). Get the coefficients from the `/spacewx/s2kops/main/operations_folder/OPScommon/s2k_Mg10_convert_vx_yz.txt` file, and reads the file `/spacewx/s2kops/main/operations_folder/OPScommon/SD_M10.txt` for a 365-day average.
- **`ReadStore_Mg2file`** This class reads a SWPC/NOAA, NESDIS, Solstice, or GOME mg2 file, using a specific format (see for example, Config_mg2.remoteSWPCsbuv30), and inserts any new records found in the file into the Mg2/mg2 mySQL database. It calls `Update_Mg2DBrec`, and if it finds a record at the same Day, it checks to see if the Mg2_value, quality_flag, data_level, data_source, data_vers, prepro_prog, process_prog, postpro_prog changes, and adds one second to YYYYMMDDHHMMSS prior to storing.
- **`Read_Mg2DB`** Gets a vector of Mg2 records (NOAA16, NOAA17, or SOSTICE) from the MgII/mg2 database from a start and stop YYYYMMDD. /lib/mysql-connector-java-5.0.7/mysql-connector-java-5.0.7-bin.jar must be in the classpath.
- **`Report_Sat_mg2`** Version 2.0a. Gets a vector of Mg2 records from the MgII/mg2 database. Report on the values from each satellite so an IDL program can read and plot values.
- **`Select_Mg2rec`** Reads a single MgII record out of the MgII/mg2 database, based on its time_key, instantiating a Mg2_record.

IDL Programs

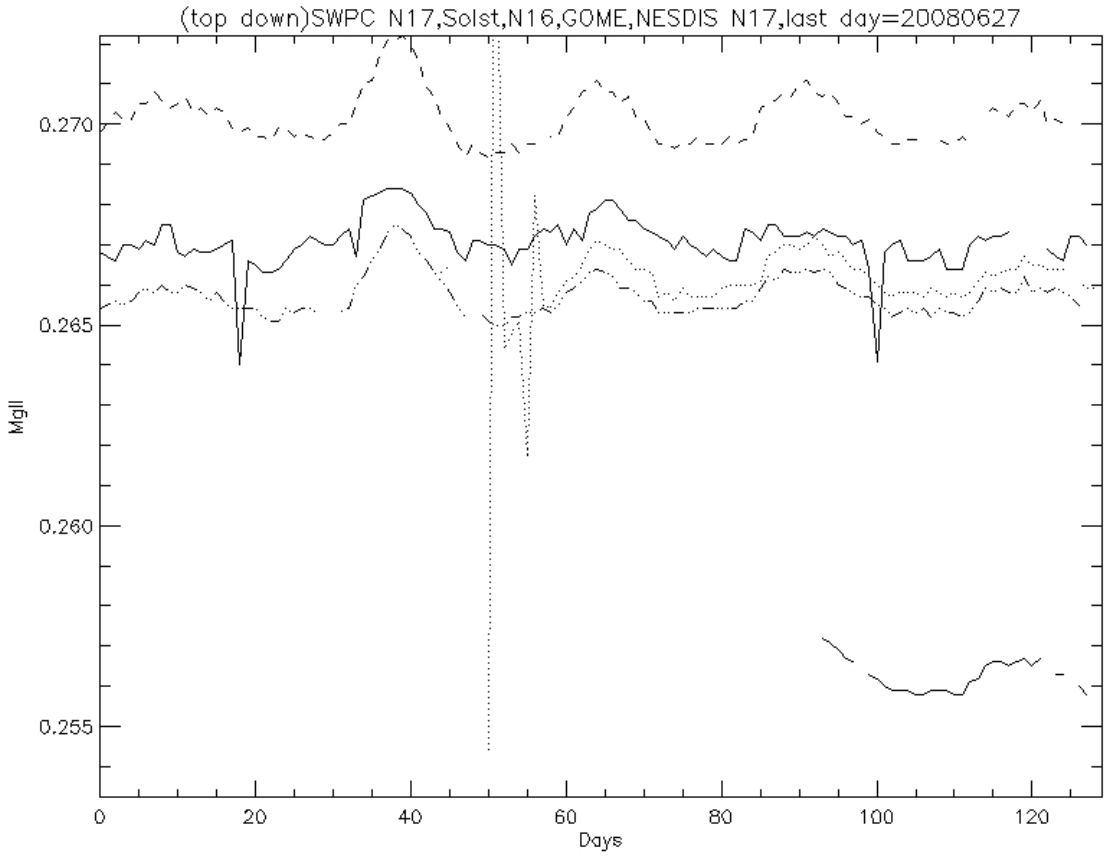
The following IDL Programs (and their associated support programs) are used as parts of operations:

- **`N17_mg2_daily_plot.pro`** Plots the output of `N17_mg2_daily.pro`
- **`N17_mg2_daily.pro`** Calculates the daily MgII value, using the output from `nesdis_mg2ops_noaa17.pro`
- **`nesdis_mg2ops_noaa17.pro`** Unpacks the binary telemetry file from NESDIS.
- **`Mg2_adj_predictor.pro`** Post-processing output from **`Adj_mg2_bestOf4.java`**, producing final file for S2K ops.
- **`Mg2_adj_pred_atmos.pro`** Post-processing output from **`Adj_mg2_bestOf4.java`**, producing an improved version of MgII output data
- **`Plot_Mg2_allSats.pro`** For visual verification, plots the input from all the satellites.

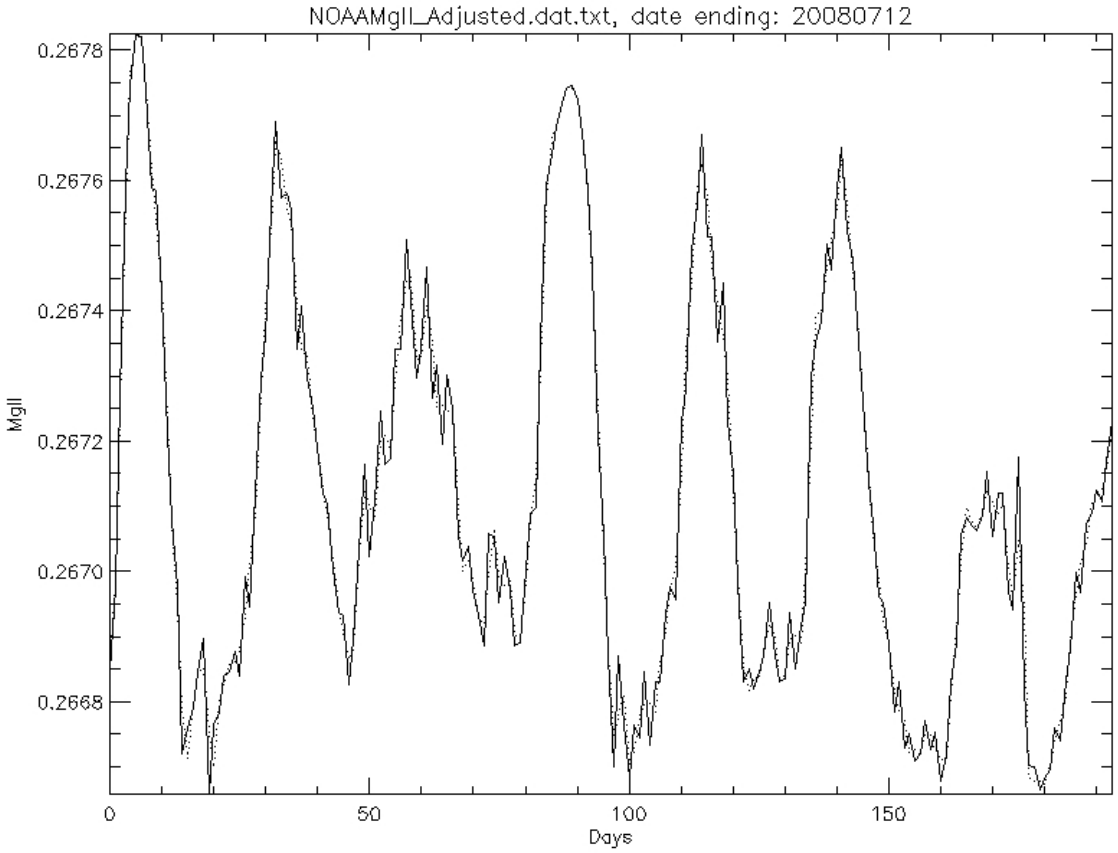
- **`plot_mg2_spcl.pro,`** Used to inter-compare various MgII data sources, and produces the linear regression coefficients between NOAA-17 data and the adjusted MgII from DCA v4.0 comparisons.

Note: The two main programs from Matt DeLand, `nesdis_mg2ops_noaa17.pro (originally DISCRETE_SOLAR_NOAA17.PRO), N17_mg2_daily.pro (originally TIME_NOAA17_MGII.PRO)`, were re-written from their original form to be used for daily operations on a new platform and configuration. The source code (it ain't pretty!) maintains the majority of the original syntax, and in-line comments

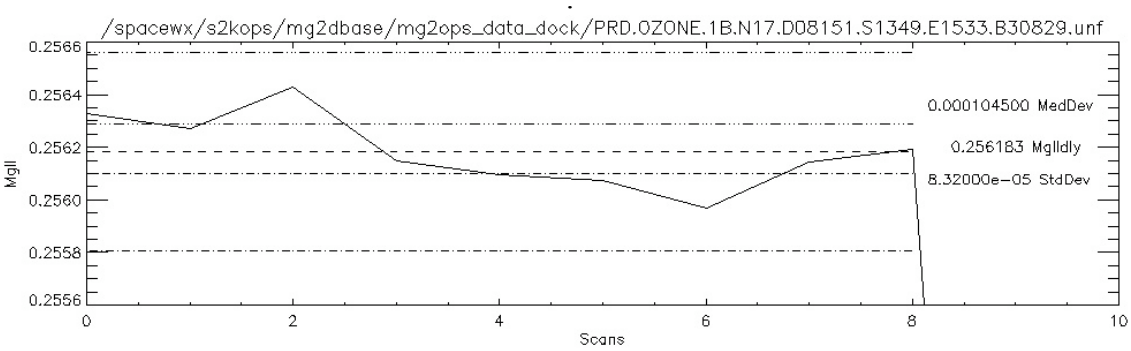
Sample Figures



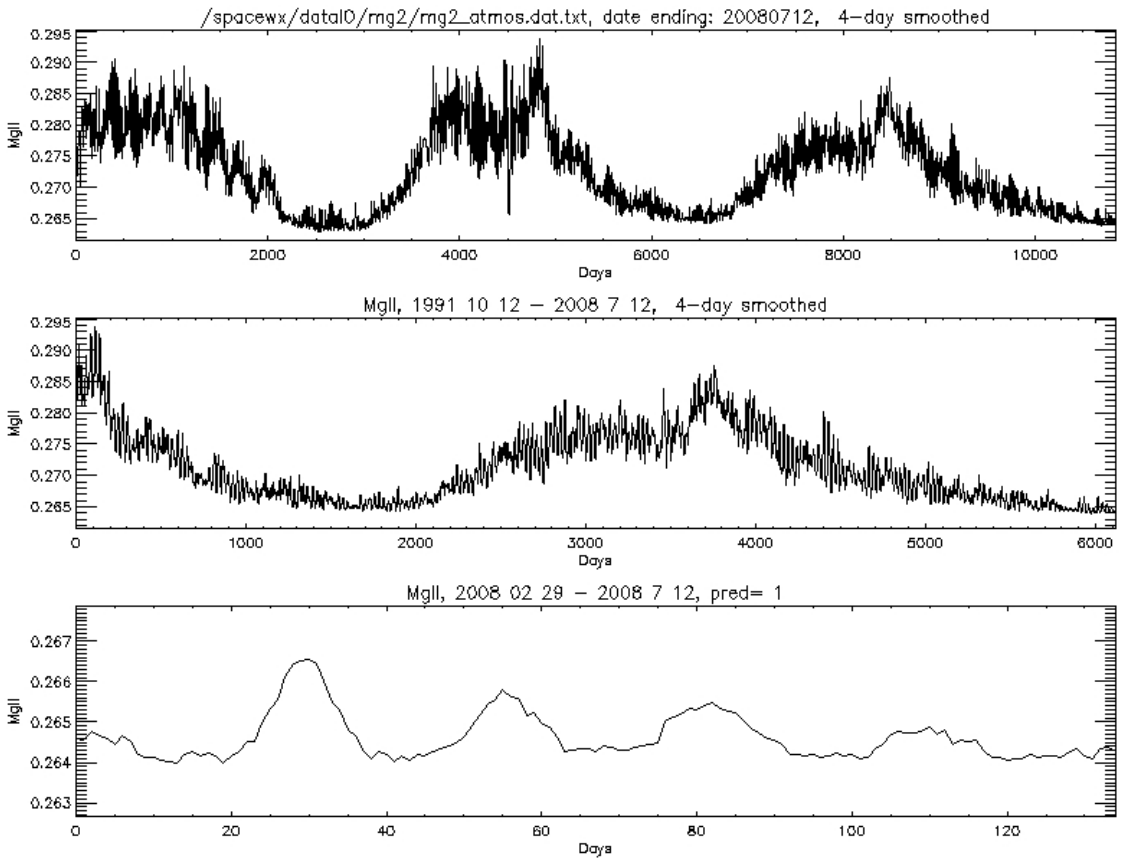
Data Fig 1. The MgII input data (Level 2), prior to processing.



Data Fig 2. The past 135 days of MgII (Level 4) after the Adj_Mg2_best_of4 algorithm (dotted line is smoothed values, not used in output files)



Data Fig 3. The NESDIS/NOAA-17 MgII data for 8 scans, the average over 8 scans is the daily MgII used after outlier is removed.



Data Fig 4. MgII (Level 4) for several time periods, from the "Atmosphere" version (see Addendum).

Related Documents and emails:

Discrete Solar Processing Flow Chart *(from DeLand)*

- Get ancillary data files.
- Get 1B data file.
- Identify discrete scans for processing loop.
    - Extract flags and identify Mg II solar scans using memory segment.
    - Determine electronic offset, PMT temperature, extract solar angles.
    - Identify type of scan based on initial grating position, then select wavelength scale and calibration constants appropriately.
    - Loop through all samples in each scan.
        - Extract grating position, raw counts, gain range, then select best values.
        - Subtract offset to get net counts.
        - Apply non-linearity correction (typically < 1%).
        - Interpolate major frame boundary angles to current sample, then calculate goniometric correction.
        - Calculate and apply interrange ratio to produce equivalent Range 3 counts. Simple process for Range 3 anode data (*normal*), more complex for Range 3 cathode data.
        - Calculate diffuser degradation correction.
        - Calculate correction for PMT temperature (difference from 20°C). Typically < 1%.
        - Apply all corrections with radiometric calibration constants to get irradiance for current sample.
    - End of loop through current scan.
    - Write lots of information to binary file for each scan, subset of information to print file.
- End of loop through all scans.

From: "Matthew DeLand" <matthew_deland@ssaihq.com>
Date: November 7, 2007 2:21:11 PM GMT
To: "S Dave Bouwer" <dbouwer@spacenvironment.net>
Cc: "W. Kent Tobiska" <ktobiska@spacenvironment.net>
Subject: RE: NOAA-17,18 processing

Hi Dave,

We (SSAI) do a lot of work with the SBUV/2 1b data for instrument characterization and solar irradiance processing in addition to the ozone processing, so we probably have a more extensive procedure than what Rodney and Larry use. Short answer: Yes, I'm happy to collaborate with you, and I'm willing to share my IDL processing code. Let me lay out my thoughts, and then you can decide how you want to proceed.

1. All of my solar irradiance software was written in IDL 10-15 years ago. There are various revisions needed for each new SBUV/2 instrument, but the basic structure hasn't changed. This does mean that some of the syntax looks a little outdated for an up-to-date programmer, but everything does run in IDL 6.3.

2. When I do my normal processing on monthly 1B data files, I usually strip out any solar irradiance data records first because they're a small portion of the overall file. This step was more important years ago, when reading 65 MB input files was time-comsuming, but it's not such a big deal with modern computers. If you're going to be working with daily or near-real time data files that are probably 2 MB or less, I doubt that you would need this step.

3. My main processing step is to convert the 1B data from raw counts into calibrated irradiances,

applying all appropriate corrections: electronic offset (similar to dark current), non-linearity, goniometry, gain range conversion, PMT temperature sensitivity, diffuser degradation, and finally radiometric calibration. This algorithm does not include a correction for overall throughput change, but that isn't an issue for Mg II index calculations. All solar irradiance measurements in the file are processed automatically, whether they're made at ozone or Mg II wavelengths. Everything stays in binary form at this point, because there's enough supporting information for each scan to make ASCII files unwieldy. There are separate versions of the code for NOAA-17 and NOAA-18 because of difference in instrument parameters, but the basic algorithm is exactly the same.

4. I then run a separate piece of code to calculate Mg II index values from the irradiances for each scan, compute daily average values, and write out a time series if so desired (binary or ASCII). This function could probably be merged with the irradiance processing in step #3 if you wanted to have a single piece of code for operational use. Our Mg II algorithm is very simple; it's just the (3 core)/(2 wing + 2 wing) formula that Don Heath and Barry Schlesinger originally developed.

5. I suspect that any stand-alone documentation I have for this work is years out of date. My code is fairly well-commented for internal documentation, but at the moment that includes accumulated stuff in some places that you wouldn't need. I would feel better if I took a little time to clean up those programs before sending them out to you.

6. As an example of my code and a starting point for you, I've attached a program (dump_raw_1b_discrete.pro) that's designed to read raw 1B data and dump information from each scan into an ASCII file. My understanding from Donna McNamara is that the near-real time data sets use the same record format as the monthly data sets, so in principle this code should be ready to use. The on-line NOAA KLM User's Guide (http://www2.ncdc.noaa.gov/docs/klm/html/c9/sec97-1.htm) also discusses the 1B data set in detail.

Hopefully this will give you a decent start. Please let me know what else you need or want. Thanks!

Matt

Addendum – Provisional Operational Calibrations

Based on linear regressions between the data from the SET database MgII, the MgII used in the S2Kops code (DCA vers 4.0), and NOAA-17 from Matt DeLand, two data sets are now created:

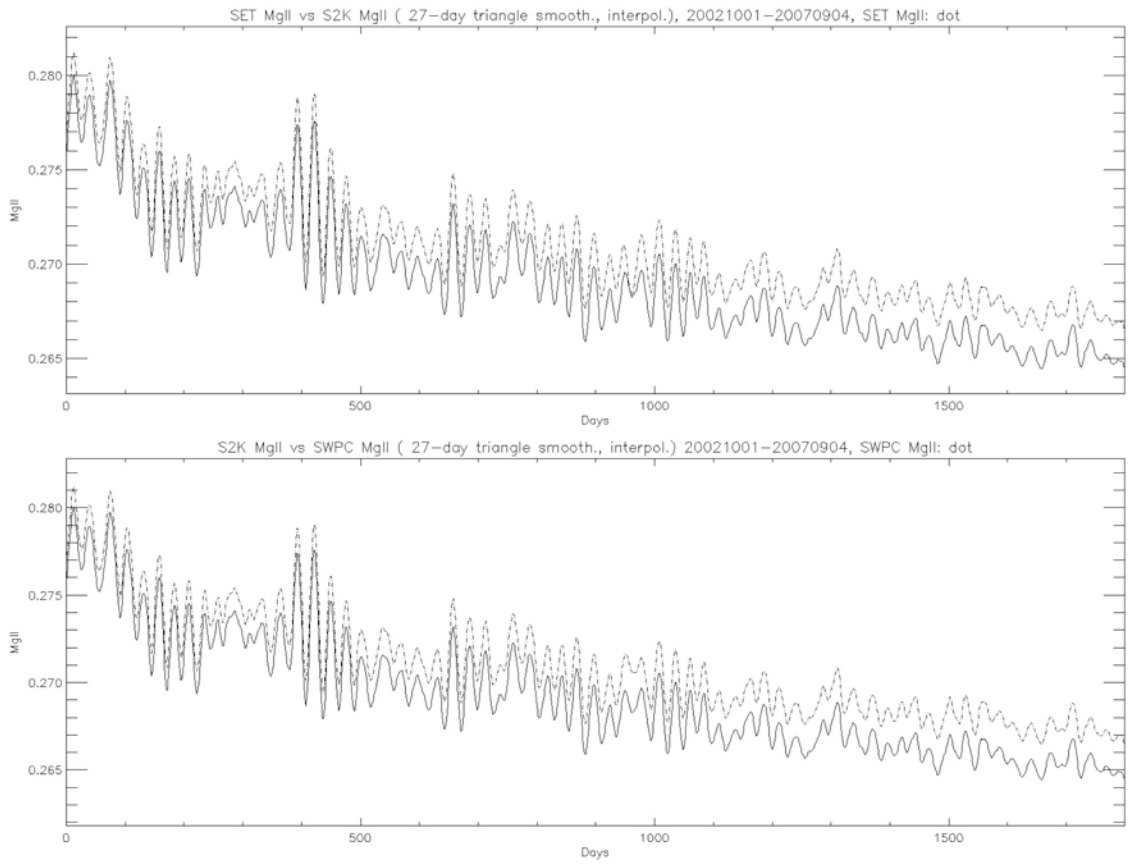
- MgII "Atmosphere" – NOAA-17 adjusted to the historical S2Kops MgII data. This is what will be used operationally at some point.
- MgII "Solstice" – NOAA-17 adjusted to the Solstice 81-day average as described above.

This section summarizes the method (see plot_mg2_spcl.pro) used to derive the MgII Atmosphere version. The pertinent data files used in this study (program `plot_mg2_spcl.pro`) are (all are located in `/spacewx/dataIO/mg2/`):

```
mg2N17file    mgii_n17_200805.dat.txt              from DeLand N17
mg2SWPCfile   MgII_SWPC_Hist_Adj.dat.txt           from Viereck
mg2SETfile    mg2_01_Oct_02-02_Mar_08.txt          from SET database
mg2S2Kfile    MgII_Tobiska_91-08.dat.txt           from reset_get_mgii.pro
pltFile       SWPChist_vs_MDn17.jpg                plot of datasets
outN17adjFile N17_adj2-S2Kops_1Oct02-31May08.txt   NOAA-17 adj to S2Kops
```

Between all the datasets, the concurrent dates were in the linear regressions are 1 Oct 02 to 31 May 08, in which both the "S2K" version (From Kent Tobiska, based on DCA 4.0 long-term corrections). On verification is in the comparison of the SET database MgII, the NOAAMgII.dat, and the S2Kops version: The figure below shows the SET database and the NOAAMgII.dat versions are consistent between themselves and S2Kops (there are daily variations not shown because the SET version uses some data not available in the NOAAMgII.dat version). 27-day smoothing is used to make the figures easier to read.

There is very one important point. The long-term MgII data available at the SWPC website (NOAAMgII.dat) has missing days and repeated days, so while it appears monotonically increasing, it is not, and care must be taken to account for this when reading the data



Addendum Fig 1. Inter-comparison of NOAAMgII.dat, SET database, and S2Kops MgII.

The result of the linear regression between the S2Kops MgII and NOAA-17 MgII (DeLand) are shown below (a 27-day smoothed comparison was done in the top panel to more clearly see the differences, shown in the top panel in the figure below). The correlation is extremely close (> 99.99%). To adjust the (unsmoothed) NOAA-17 values to S2Kops the resultant coefficients, using daily values (bottom panel), are:

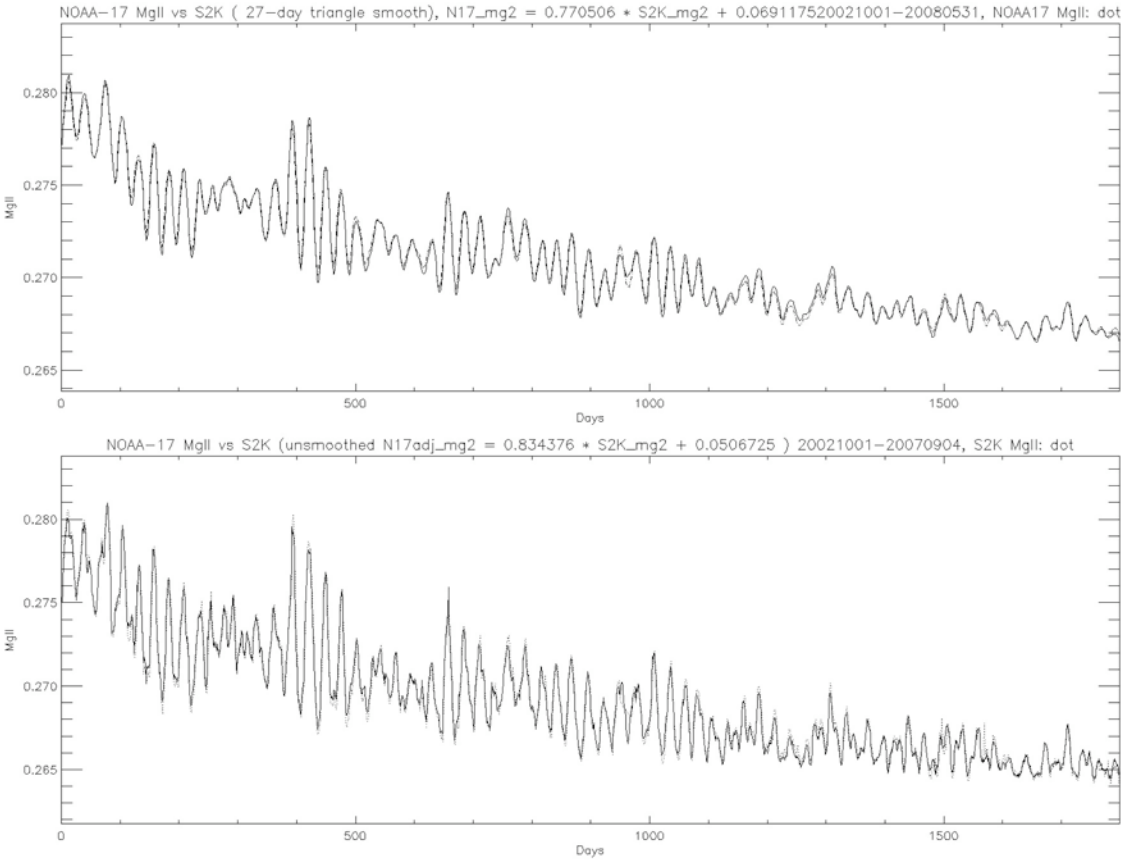
Mg2_N17_adj_to_S2Kops = 0. 850108* Mg2_N17_meas + 0. 0464865

These are the coefficients used to adjust the "Atmosphere version" of NOAA-17 data from NOAA/NESDIS processed daily by `adj_mg2_bestOf4.java` and `Mg2_adj_pred_atmos.pro`

There is one minor difference between the version of DeLand's code and that used in `N17_mg2_daily.pro`: the new version rejects outliers (more than two standard deviations from the mean) before re-calculating the mean of the 3-8 SBUV scans for the day. It was not clear this was done in DeLand's code, but the difference seems to be very small (<0.002%). However, this should be verified at a later date when there are sufficient data to perform the comparison, and to ensure there are no differences due to instrumental (e.g., long-term drift) or algorithmic changes.

The comparison of DeLand's historical MgII and the new NESDIS MgII at this time is as follows:

DeLand MgII				SET MgII
2008	145	2336	0.25713	0.257125
2008	146	2337	0.25694	0.257125
2008	147	2338	0.25669	0.256684
2008	148	2339	0.25656	0.256557
2008	149	2340	0.25638	0.256331
2008	150	2341	0.25632	0.256316
2008	151	2342	0.25619	0.256183
2008	152	2343	0.25593	0.255976



Addendum Fig 2. Comparison of NOAA-17 (DeLand MgII) to S2Kops MgII

Notes on data sources used in S2Kops/mg2dbase

```
The operational SET-Atmosphere MgII data comes from 3 data sources:
The 3 sources used are stitched together from the following (orig) data files:

(1) MgII_mg2, NOAAMgII.1978-99dat.txt, 07-Nov-78 to 31-Dec-99, 2443820 to
2451544, Historical SWPC composite (from database)

(2) DCA_v4_KT, MgII_Tobiska_91-08.dat.txt, 12-Oct-91 to 15-Jun-08, 2448542 to
2448542 S2Kops, adj to F10 long-term (from Kent)

(3) MgII_DL_N17, MgII_Deland_02-08.dat.txt, 01-Oct-02 to 31-May-08, 2452549 to
2454618, NOAA-17 (from Matt Deland)
```

(4) MgII_mg2, MgII.mg2 database, 24-May-08 to current, 2454611 to current, NESDIS NOAA17 created daily

Notes:

(a) NOAA16 based data goes bad 01-Sep-07 2454345 (244), so Deland's data needs to fill in from here till NESD17 is avail.

(b) NESD17 database data begins 24-May-08 2454611 (145), which is the current operational set (other sats used when missing)

(c) The Mg2_DB_Main.java program creates the file:
/spacewx/dataIO/mg2/NOAA_MgII_78_current.txt
and copies it to
/spacewx/s2kops/main/operations_folder/time_series_data/NOAAMgII.dat

Final MgII data delivered to S2Kops/operations_folder/NOAAMgII.dat comes from the data sources stitched together from:

**
 * Start Stop DataSource JD_start JD_stop Description
 * --- --- --- --- --- ---
 * 07-Nov-78 11-Oct-91 MgII_mg2 2443820 2448541 Stored in file MgII_1978-07.dat.txt
 * 12-Oct-91 31-Aug-07 DCA_v4_KT 2448542 2454344 Stored in file MgII_1978-07.dat.txt
 * 01-Sep-07 31-May-08 MgII_DL_N17 2454345 2454618 Stored in file MgII_2007-08.dat.txt after adjustment to DCA_v4_KT ->2
 * 01-Jun-08 current MgII_mg2 2454619 current Retrieved from MgII.mg2 DB, adjust, written to file MgII_DB_cur.dat.txt ->2**

1-> The Adj_mgII_bestOf4.java algorithm reads the database for recent data, substituting SWPC-NOAAA17, GOME, or Solstice values when the NESD17 data is unavailable after they are adjusted to the past 128-days of NESD17

2-> The NESD17 data is adjusted to DCA_v4_KT values via regression:
Mg2_N17_adj_to_DCA_v4_KT = 0.824165*Mg2_N17_meas+0.0500744