NCAR/EOL
Earth Observing Laboratory

White Paper

Community Software Tools for the Radars, Lidars and Profilers of the NSF Lower Atmosphere Observing Facilities

Request for community comment

Mike Dixon,Wen-Chau Lee
Mike Daniels, Charlie Martin,
Steve Cohn, Bill Brown

2012-09-18

Executive summary

Atmospheric researchers make extensive use of scanning and profiling remote sensors, including microwave radars, wind profilers, lidars and sodars. Extracting the full value from measurements with these sophisticated instruments depends on having good software tools, and on shared analysis among the community of users. Engineering good software is expensive - there are savings when software and data formats are shared rather than developed locally by each user.

NCAR has a history of software and community format development for scanning radars, and it is appropriate for a national center to provide an efficiency of this type. This proposal describes the creation of core software and an open-source software exchange, facilitated by NCAR, but with software development and maintenance occurring throughout the user community. This exchange would function well beyond the scanning radar community to benefit users of wind profilers, some lidars, and possibly other remote sensors. Such a shared approach could be used for all software used by NSF's LAOF community. Furthermore, converters for common commercial wind profiler formats could greatly broaden the use of non-LAOF wind profiler datasets.

The proposal builds on 25 years of software and standards created by NCAR and UNIDATA. This software has been a foundation for the weather radar research community, but it has become clear in recent years that NCAR cannot expand or even continue to maintain the existing suite of software in a rapidly changing technological environment. The new model we propose is an NCAR-Community hybrid in which NCAR develops data formats and core software, and enables and encourages the user community to participate by developing application modules and algorithms. These modules would extend the core software to meet specialized data quality control, scientific analysis, display, and data integration needs of the developer, and ideally would be useful to others in the community. This hybrid approach is analogous to the successful approach used by the Weather Research and Forecasting model (WRF) in which NCAR designs and develops data formats and core software, while the user community develops modules written in efficient high-level languages, such as MatLab® , IDL®, NCAR Command Language (NCL), and Python.

We estimate that building on past work to develop core software and web hosting will require 2.5 software engineers for 3 years, as well as science and project management work, for a total of 9 person-years spread over 3 years. To put this in perspective, the prior radar work upon which the system will be based represents more than 80 person-years of effort. Through the proposed work we can provide the research community with access to high quality software in a form that is flexible and maintainable over time. The support extends beyond weather radar and lidar research software to also facilitate wind profiler and other remote sensor research. Without a new approach we are concerned that even the existing capabilities for the community will continue to degrade and soon be lost in today's world of quickly evolving software and instrument development.

Community Software Tools for the Radars, Wind Profilers, and Lidars of the NSF Lower Atmosphere Observing Facilities

1 Introduction

1.1 History – the legacy software

Software for the atmospheric radar and wind profiler research communities evolved from different beginnings. For the past 25 years NCAR's Earth Observing Laboratory (EOL), its former identity the Atmospheric Technology Division (ATD), and the Mesoscale and Microscale Meteorology Division (MMM) have assumed the role of developing, maintaining and supporting a suite of radar analysis software ranging from the data exchange format (UF and DORADE[1]), data format translation (e.g., TRANSLATOR, FORAY), data quality control (soloii), interpolation and synthesis (reorder, sprint, and CEDRIC), and data integration and visualization (Zebra and soloii). This software suite was widely distributed and used by the US and international radar community.

Software to make full use of EOL's deployable radar wind profilers has been less available. Basic data display routines are available from wind profiler manufacturers, and EOL has informally shared high-level manipulation, display, and analysis code developed by scientists for our internal use. Most often wind profiler routines to analyze field project data have been written locally by researchers or graduate students at each institution. The lack of flexible, reusable code has been a barrier to full use of wind profiler data where local researchers do not have the time or skills for such development.

The legacy software infrastructure, built on mainframe computing concepts of 30 years ago, can no longer fulfill the needs of researchers using powerful desktop workstations to develop analysis routines to address specific scientific problems. The completion of the US national NEXRAD radar network, the rapidly increasing number of research mobile Doppler and polarimetric radars, and the recent predominance of the internet for data delivery and visualization have increased the community need for modern software tools tailored to their requirements.

The demand from the community for NCAR to support and improve the existing radar software remains strong, and requests for us to share wind profiler software are made after nearly every ISS deployment. EOL scientists and software engineers routinely receive inquiries from the community users regarding the usage, interpretation and debugging of the existing software, and requests for added functionality. Although some incremental upgrades to radar and wind profiler packages have been provided over the years, it has become difficult to maintain and improve their functionality because of the rapid evolution of operating systems, programming languages, and the internet. A major revision of and extension to these tools is needed if EOL is to continue to provide modern, high-quality software capabilities to the community.

1.2 Consulting the users

During recent years, EOL staff members have met with the radar and wind profiler research communities through site visits and informal discussions during scientific conferences. For example, discussions have taken place at AMS Annual Meetings, technology-specific meetings such as the AMS Radar Conferences and various profiling symposia, and topical conferences such as the AMS Conferences on Mountain Meteorology. In addition, user surveys were performed in 2005[2] and 2009[3], and in 2010 the EOL/CDS and EOL/RSF Facility Managers visited several universities and identified common radar software needs[4]. The first town hall meeting specifically to discuss radar software was held at the 2005 34th Radar Conference where the 30+ participants expressed their concerns on NCAR's ability to maintain and support the legacy software that is critical to the national and international radar community. Wind profiler needs were discussed at the 7th International Symposium on Tropospheric Profiling in 2006 and the 35th Radar Conference in 2007 (Cairns, Australia). *The response from the community shows that there is a significant and urgent need for an updated suite of tools for the gathering, analysis, display and dissemination of radar, wind profiler, lidar, and other remote sensor data.* The consensus is that EOL should play a central role in this development effort.

1.3 A new approach

We propose a new approach to meet the needs of radar, wind profiler, and lidar data users over the long term, with possible extension to other remote sensors. Given the current resource constraints, we are proposing an open source concept that encourages community participation in algorithm and display development. Under this proposal, EOL will establish a core framework within which to develop, maintain and upgrade commonly-used functionality, including data exchange/archive formats, format conversion, data handling and integration, algorithms and displays. The community, including NCAR scientists, will be able to use this framework for common processing tasks and to develop modules to meet more specialized needs using efficient high level languages such as Matlab®, IDL®, NCAR Command Language (NCL), and Python. These modules will be shared with the full community in a manner similar to that adopted by the Weather Research and Forecasting model (WRF) community.

EOL will establish a code repository to promote sharing of both the core software and the community-developed modules. EOL will assume the responsibility for maintenance and enhancement of the core software, while the developers of the community modules will take the responsibility for their maintenance. Where possible, the software will be engineered for both real-time operations (data acquisition) and post-operations analysis and research. This will minimize the cost of development, keep the software as simple as possible, and reduce training requirements.

1.4 Building on prior work

Work at EOL, MMM and RAL (NCAR's Research Applications Laboratory), along with UNIDATA, has provided a sound software base from which to develop this suite of radar tools. A large fraction of the required software for the core functionality has already been developed, and a capable software system can be created for a relatively modest incremental effort. It is estimated that over 80 person-years have been expended on this previous work at NCAR from various funding sources (see section 2.7). More modestly, internal tools for wind profiler data quality checks, display, and analysis have been created over the years. These wind profiler tools are already informally shared with the community and can be a strong basis for hardened core capabilities.

Significant work remains to update these radar and wind profiler components and to integrate them into an open software system. This proposal outlines the goals of the new system, the capabilities of the existing software, and the work required to convert the existing components into an integrated system.

7.5 person-years are required for the software engineering work and 1.5 person-years for science and project management support. This represents around 10% of the work that NSF and other agencies have already invested.

1.5 Facilities supported by this work

The software system proposed here will support many of the NSF Lower Atmosphere Observing Facilities (LAOF), as well as work by researchers using other platforms, such as NWS NEXRAD and the radars and profilers deployed by DOE/ARM, NOAA, NASA, and by individual universities.

The following NSF LAOF facilities will be supported by work carried out under this proposal:

- Ground-based and airborne radars: SPOL and Ka-band, CSU-CHILL, DOW5, DOW6 and DOW 7, ELDORA, Wyoming Cloud Radar, HIAPER Cloud Radar (HCR).
- Wind profilers: 915-MHz Integrated Sounding System deployable wind profiler network, 449-MHz modular wind profiler network under development.
- Ground-based and airborne lidars: HSRL – high spectral resolution lidar, WCR – Wyoming cloud lidar, NCAR-Montana State Univ. water vapor DIAL.

1.6 Goals

The primary goal of the proposed work is to develop a core suite of software tools that enable researchers, students, professors, engineers and system operators to efficiently and easily manipulate and visualize radar and wind profiler data, and related data sets (e.g. lidar, sodar), as applicable.

Our intent is that common tools would be available for real-time **operations** during field campaigns and other observing activities, for post-analysis and scientific **research** after the experiment, and for **education** in the laboratory or classroom.

To achieve this, we need to achieve the following specific goals:

1. Develop data archive and exchange formats. Specifically, design a data storage sub-system using self-describing, portable (i.e. computer platform-independent) well-understood formats, with an emphasis on NetCDF.
2. Provide the following core services: data format translation and sharing, data quality control, data manipulation algorithms, data integration and visualization.
3. Foster community development of analysis tools in high level languages such as MatLab®, IDL®, NCAR Command Language and Python.
4. Enhance data accessibility by providing tools that are accessible via the web, and providing an integration layer between data and web-enabled software.

2 Implementation Plan

This section describes characteristics, details, and rationale of the proposed work.

2.1 Data formats

2.1.1 Introduction

If possible, data formats will be based on the UNIDATA NetCDF format. When NetCDF on its own is not suitable, stand-alone XML, or XML stored within a NetCDF file, will be used where possible.

Binary-only formats will be limited to those situations in which no other solution is workable (for example, high-rate time-series data). For such formats, source code libraries with well-documented interfaces will be implemented and made available for translation to and from these binary formats.

2.1.2 Self-describing formats

One of the goals is to support data sets using open, self-describing, industry-standard formats.

The requirements of the selected data set formats are:

- Portable (i.e., computer-platform independent)
- maintainable
- self-describing
- easy to handle (read/write/understand)
- documented
- standardized

The radial data exchange format of choice, the NetCDF-based CfRadial[5] format, has already been developed by NCAR in consultation with the user community.

2.1.3 NetCDF conventions

NetCDF is a flexible format, and can be used to represent the same data in many different ways. Therefore it is necessary to follow agreed-upon 'conventions'.

If suitable, the following existing NetCDF conventions will be followed:

- CF (Climate and Forecasting) conventions[6] for Cartesian gridded data (referred to in this proposal as CfGrid: http://cf-pcmdi.llnl.gov/documents/cf-conventions/1.6/cf-conventions.html
- CfRadial convention for radial radar, lidar and profiler data: http://www.ral.ucar.edu/projects/titan/docs/radial_formats/#cfradial

- CF conventions for other data, as applicable. For example, UNIDATA has been working on a CF convention for point and profile data, which will be used as appropriate: www.unidata.ucar.edu/staff/caron/presentations/Point%20Data.pptx

2.1.4 XML formats

Some data types are not easily handled in NetCDF. An example of this is the tree hierarchy in the storm tracks identified by TITAN. In this and similar cases, XML will be used to provide a suitable representation of the data, as opposed to a strict NetCDF representation.

The aim, however, is to make use of NetCDF wherever possible.

2.1.5 Binary formats

For some data types, such as high-data-rate raw time series (I/Q) data, it may be necessary to use a binary representation for real-time data acquisition.

In these cases, translation applications will be provided as part of the code suite to convert the binary data to NetCDF for analysis purposes.

2.1.6 BUFR formats

The WMO BUFR format is widely used to code observations for bundling into meteorological datasets. For example, wind profiler observations would be coded into BUFR format for ingesting by the GTS and forecast models.

http://www.wmo.int/pages/prog/www/WDM/Guides/Guide-binary-1A.html

2.2 The core suite and community suite

The proposed suite of tools will comprise two groups of applications:

- **core** applications – those developed and maintained by NCAR, to provide the base level functionality of the suite.
- **community** applications – those developed by members of the user community (including NCAR), to provide specialized functionality which is beyond the scope of the core suite.

The community will be encouraged to share these applications amongst others in the user community under an open software framework. The mechanisms to accomplish this will include:

- **informal sharing**: users develop applications and are encouraged to share these between peers. NCAR will not be involved.
- **formal sharing**: users who have developed an application with wide appeal may submit it to peer review for quality assurance. If such an application proves to be of high quality and has wide interest, it may be hosted on an NCAR web site, to be available for download by the community. The author institution will maintain and support the application and provide enhancements and bug fixes as they become available.

- **inclusion in the core suite**: it is possible that some community-developed applications would be formally adopted by NCAR and included in the core suite. In such a case NCAR would take responsibility for ongoing maintenance and possible enhancement.

2.2.1 Core suite diagram

The core system will be developed at NCAR. Figure 1 below shows the basic services of the core suite.

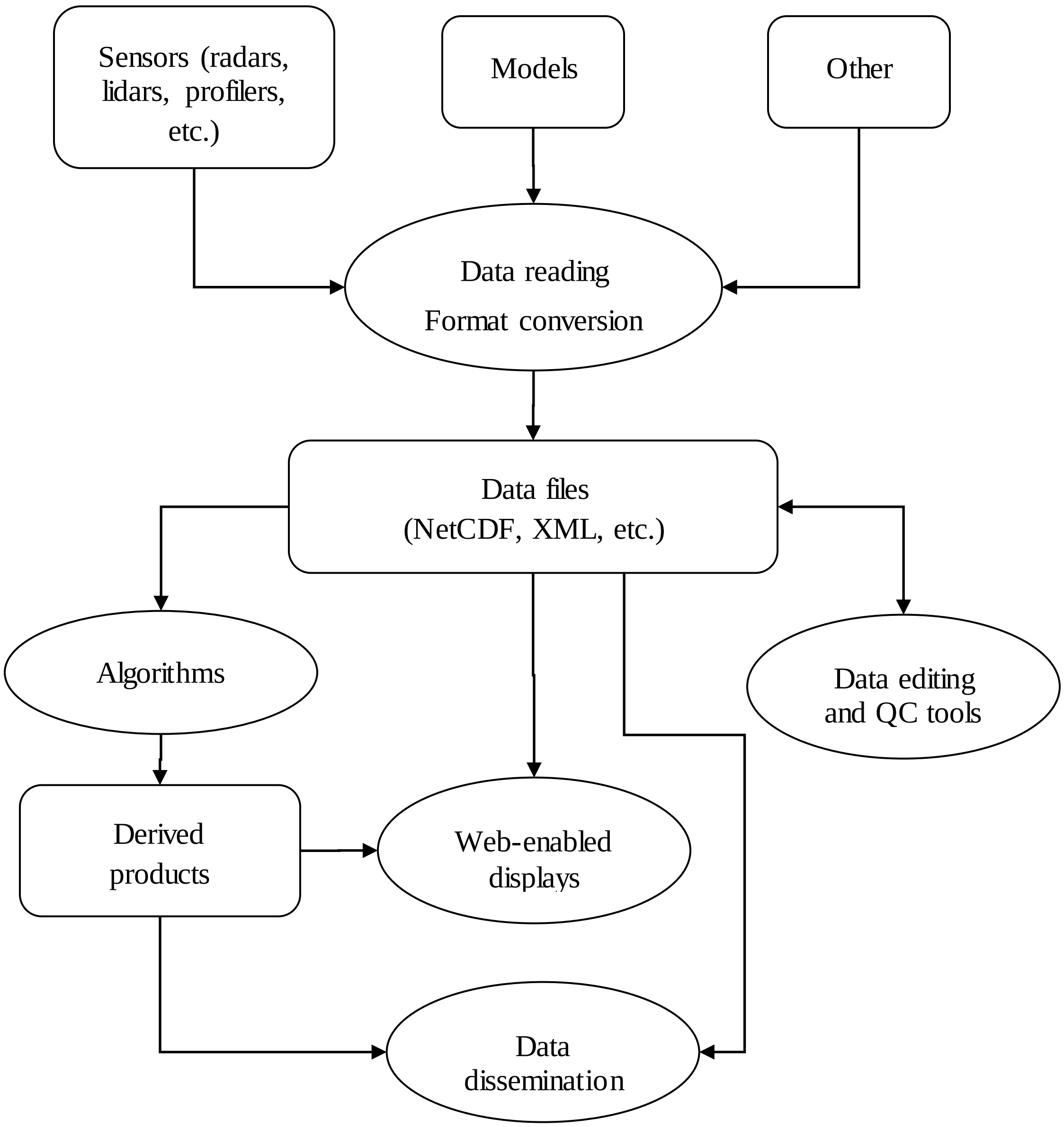

Figure 1: data flow scope for the core suite

2.2.2 Community suite diagram

The community suite of tools will be more limited in scope than the core suite, with an emphasis on data processing, algorithms and displays. Figure 2 below shows the likely scope of the community suite.

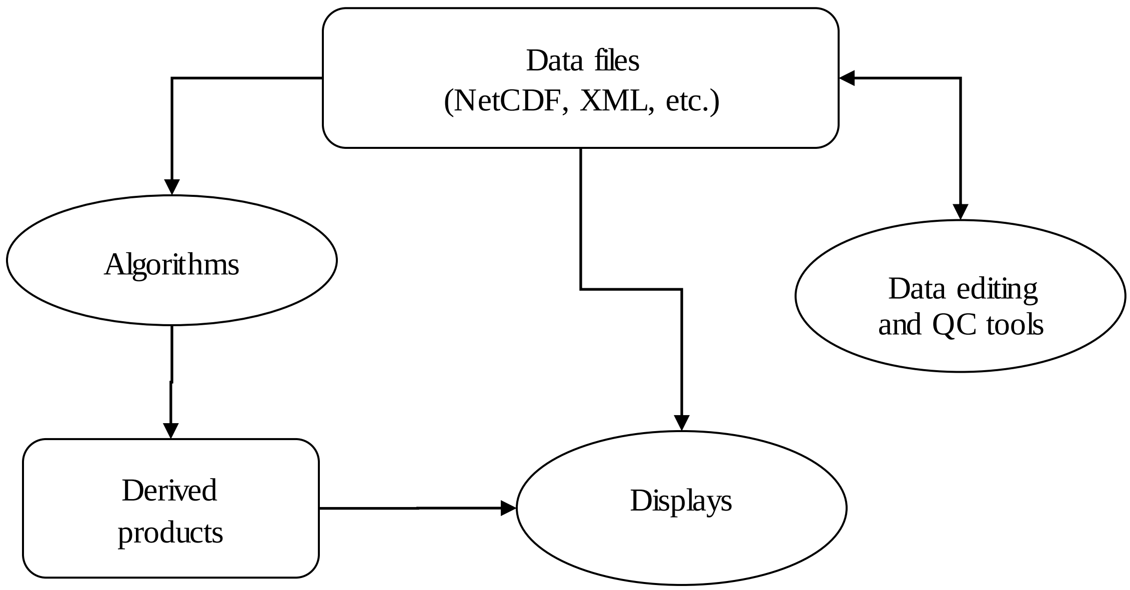

Figure 2: data flow scope for the community suite

2.3 System overview

Figure 3 shows the data flow overview for the end-to-end system. This is a high-level diagram, in which many of the details have been omitted.

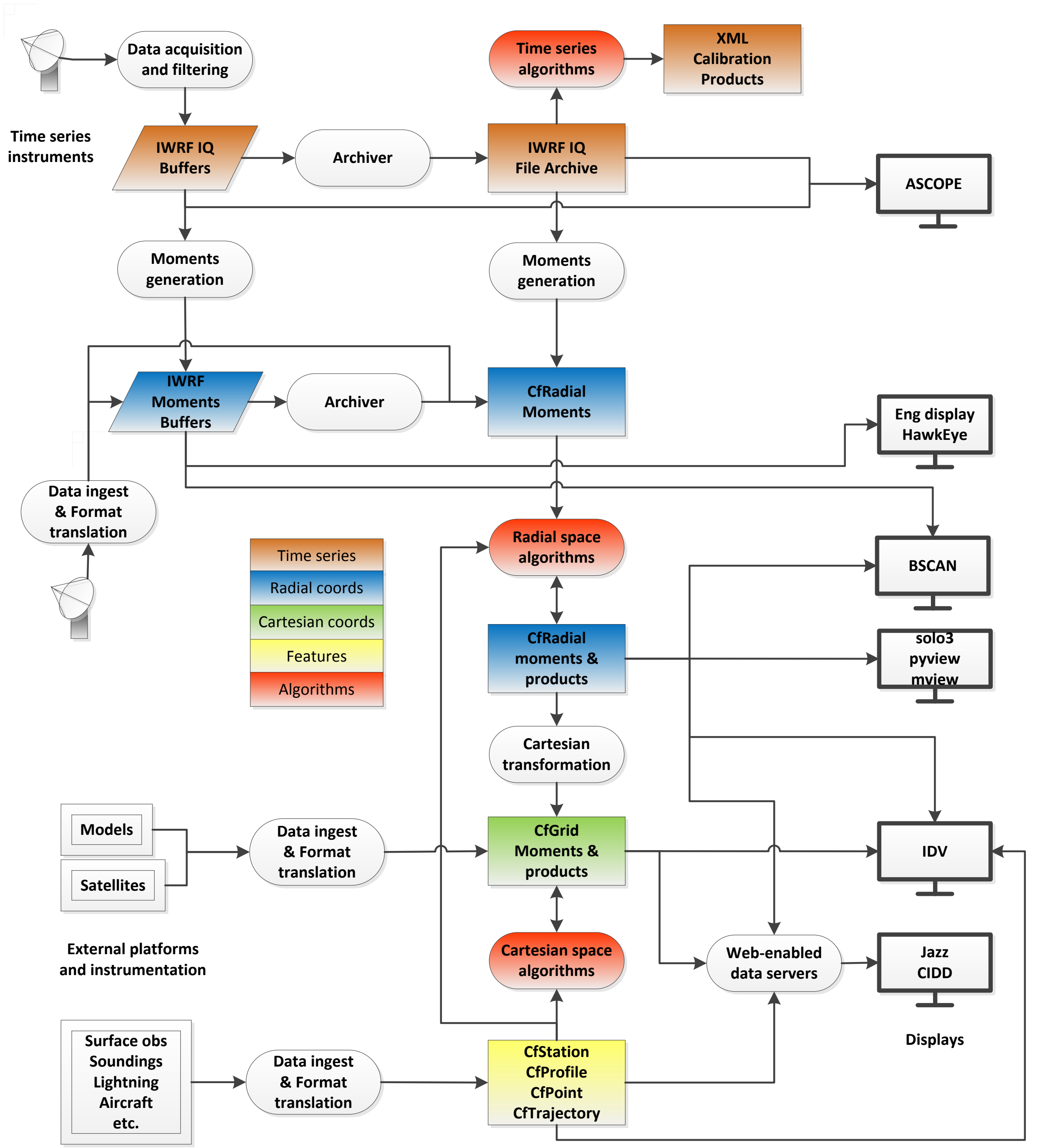

Figure 3: data flow overview for end-to-end system

On the left are the primary data sources, such as radars, lidars and other instruments. The data is read in, filtering and data quality procedures are applied appropriately, and the data is then stored using standardized formats.

Each data set belongs to one of four main types:

I/Q time series (brown)

3-D radial coordinate space (blue)

Cartesian coordinate space (green)

Point and other feature data (yellow)

The primary scientific components of the system are the ‘algorithms’, shown in red on the diagram. It is the algorithms that will allow researchers to apply scientific methods to the data, and they will be the focus of much of the collaborative work on this project. (NOTE - by ‘algorithm’ we mean an application which reads one or more data sets, performs some form of analysis, and writes the results to a derived data set.)

On the right hand side of the diagram are the various displays associated with the system. These are vital, since they allow scientists to efficiently visualize the data and the results of the algorithms.

At the top of the diagram is the processing related to time series data, shown in brown. This is the most raw, and verbose, representation of the data. As a practical matter this section only applies to certain instruments, such as SPOL-Ka, for which the raw time series data is available. Processing applied at this stage yields the so-called radar or lidar ‘moments’ – the fields most commonly used by meteorologists.

For many instruments the time series data is not available, and instead we read the moments or data of a similar nature.

In the center of the diagram, in blue, are data sets in radial (polar) coordinates – the raw coordinate system for the instruments. Some algorithms are specific to data sets in this coordinate space.

Lower down, in green, are the data sets in Cartesian space. Data sets in radial coordinates are transformed into Cartesian space using either interpolation or ‘nearest neighbor’ methods. Some algorithms are specific to Cartesian data sets.

Cartesian coordinates are useful for integrating radar and lidar data with data from other sources, such as satellite sensors and models. Algorithms and displays can integrate data only if a common coordinate system is used.

At the bottom of the diagram we show how we plan to handle reading in external data sets such as surface observations, soundings, aircraft data and lighting. These data sets, shown in yellow, generally involve point measurements at a specific time and location, some of which are grouped into profiles (soundings) or trajectories (flight data). This data may also be used by the algorithms. For example, sounding data is used to provide the temperature profile for the particle type (PID) algorithm. In addition data of this type can be integrated with the gridded data on the displays such as IDV, CIDD and Jazz.

2.4 Algorithms and displays

2.4.1 Core suite algorithms

The algorithms that will be included in the initial core suite are listed in Table 1. We show the completion percentage to be 100% if the algorithm is already in a working state. This does not imply that the algorithm will not be modified or enhanced in the future.

Progressive enhancements are typical of the iterative nature of scientific software development. It is likely that all of these algorithms will be enhanced as the community makes use of them, bugs become apparent and requests for enhancements are received.

The algorithms listed in this section do not yet include any specifically designed for lidars. These will be added as appropriate.

Table 1. Algorithms will be included in the initial core suite and their status.

Algorithm	**Estimated percent complete**
NIMA wind profiler moment and winds estimation	100
ICRA intermittent clutter removal	80
Boundary layer depth profiler reflectivity tracing	25
Profiler winds bird filtering	25
Profiler spaced-antenna full correlation analysis	60
Weber-Wuertz profiler Doppler spectra processing	25
Profiler precipitation / clear air identification	25
Wavelet profiler IQ time series processing	25
Gridding / interpolation from radial to Cartesian coordinates	90
Merging multiple radars into single Cartesian grid	90
Clutter detection (NP, AP) in time series	100
Clutter filtering (NP,AP) in time series	100
Moments computations from time series (single polarization and dual polarization, staggered PRT)	100
AP detection and removal using moments data	100
Multiple-radar Doppler analysis (e.g. dual-Doppler synthesis)	25
Storm tracking – convective (TITAN)	100
Storm analysis and climatology – convective (TITAN)	100
Echo tracking from correlation (ctrec)	100

Algorithm	**Estimated percent complete**
Echo tracking using optical flow	50
VIL – vertically integrated liquid	100
PID – particle identification from dual polarization variables	100
VAD – velocity azimuth display (enhanced, includes an estimate of divergence)	100
Refractivity (estimation of RH field from clutter)	80
Precipitation rate from dual polarization variables	100
Precipitation accumulation over time	100
Velocity de-aliasing (James and Houze algorithm)	100
Bright band detection and mitigation in Cartesian data	100
Azimuthal and radial shear from radial velocity	100
VDRAS (Variational Doppler Radar Assimilation System)	100
Convective / stratiform partitioning	100
Power and reflectivity calibration	100
ZDR calibration from vertical pointing scanning	100
ZDR calibration from cross-polar power, using clutter echoes	90
Sun-based calibration	80

Note that the list above is somewhat NCAR-centric this stage, since it based on NCAR prior work. Other institutions may have their own versions of these algorithms, and these will be added to the core if appropriate.

2.4.2 Examples illustrating the use of the algorithms

The following examples will help the reader to visualize how the algorithms and other applications can be chained together to perform desired analyses.

2.4.2.1 Dual Doppler example

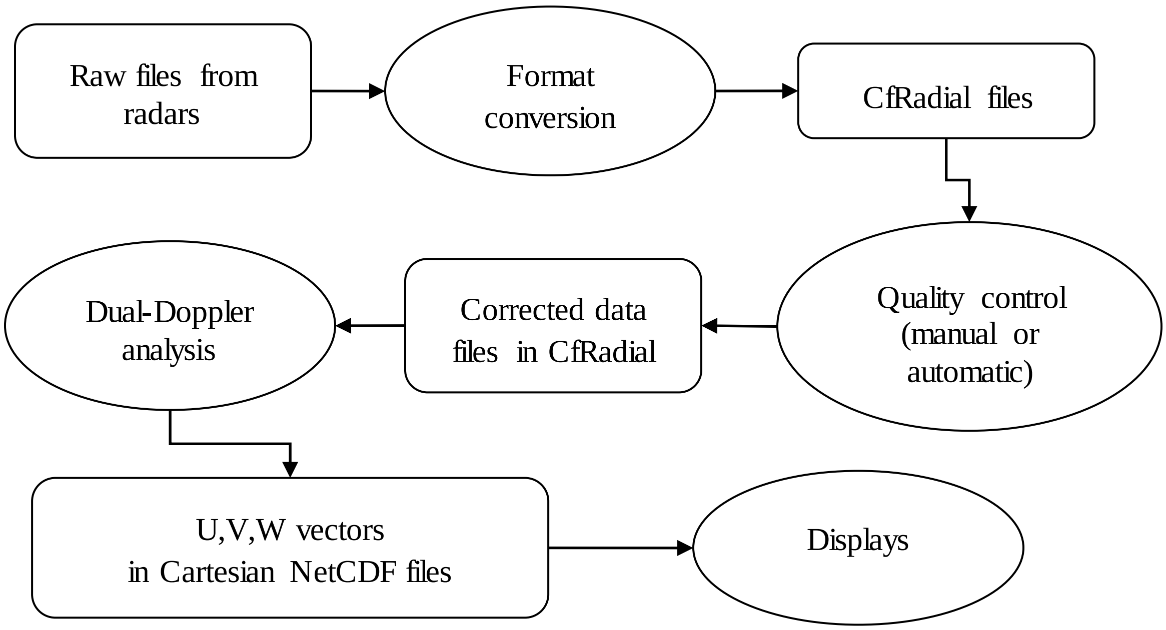

Figure 4: example data flow for dual Doppler processing

2.4.2.2 Storm tracking example

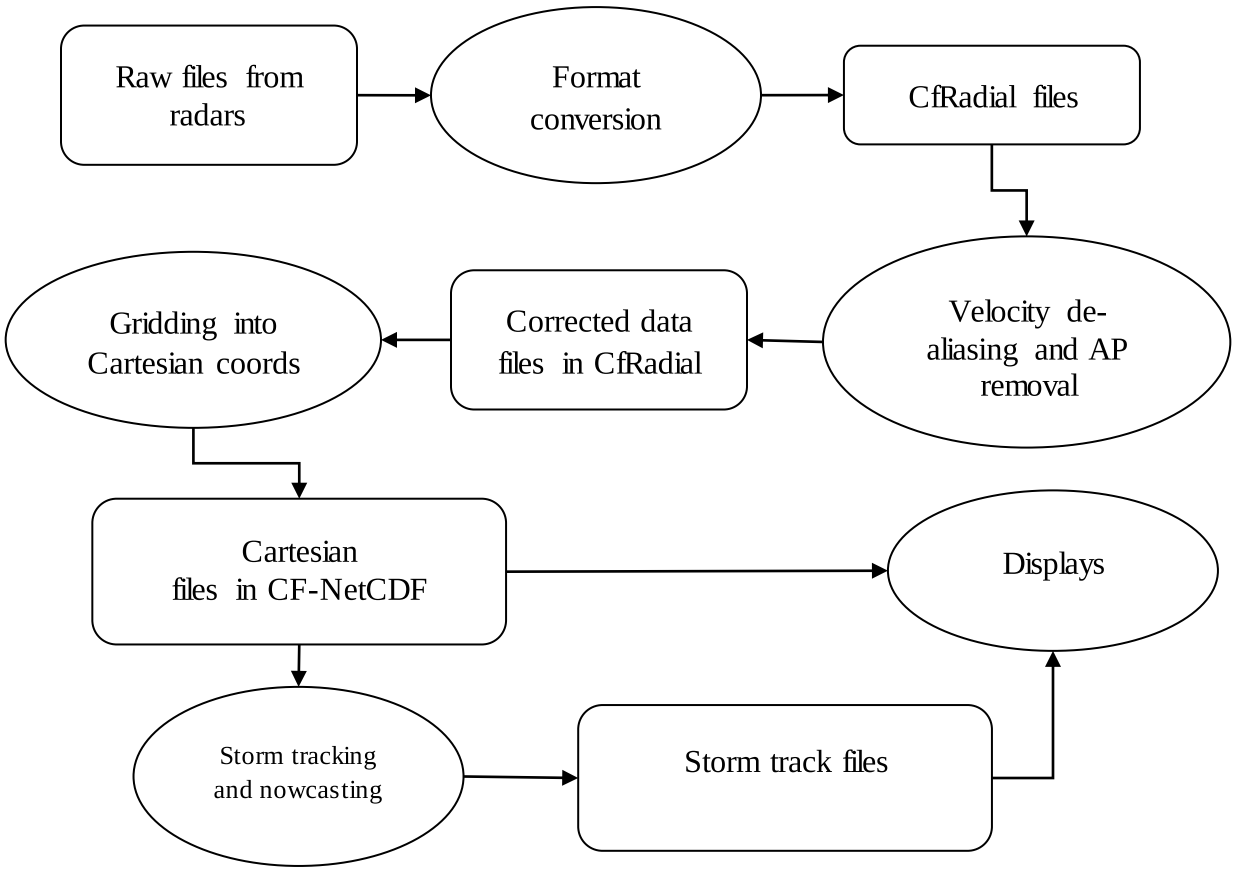

Figure 5: example data flow for convective storm tracking

2.4.3 Core suite displays

The following displays (Table 2) will be included in the core suite of applications:

Table 2. Displays will be included in the core suite.

Name	**Description**	**Work to be performed**	**Estimated percent complete**
solo3	Viewer and editor for radial data	Upgrade to C++. Bug fixes. Handle CfRadial. Improve maintainability.	65
Jazz	Java-based web-enabled viewer for integrating radar and other data sets into a single display	Major enhancements	70

Name	Description	Work to be performed	Estimated percent complete
CIDD	Legacy C++ version of Jazz	Bug fixes	100
VChill	Java-based viewer for radial data	Provide support for VChill by developing a server for CfRadial data	25
IDV	UNIDATA Integrated Data Viewer	Provide support for CfRadial and profiler obs in IDV	80
ProfilerDisplay	Wind profiler spectra and correlation display	Extend interoperability	80
Xprof	IDL wind profiler plotting package	Upgrade and improve maintainability	50
mview	Radial data viewer in MatLab	Design and development	10
pyview	Radial data viewer in Python	Design and development	0

Work on the displays will be a significant part of the total for this project.

2.4.4 Possible additional displays

The following displays, or similar applications, may be added if appropriate (Table 3):

Table 3. Optional displays for the core suite.

Name	Description	Work to be performed	Estimated percent complete
VAPOR	NCAR model viewer	Provide support for radar data in VAPOR	0
WorldWind	NASA display integrating scientific data with 3D earth-referenced display	Provide support for radar data in WorldWind	0
Google Earth	3-D viewer for earth-referenced data	Provide support for Google Earth	0
Webprof	Web interface for xprof wind profiler display	Design and development	0
Dataserver	Serves up profiler data files on the web for users	Design and development	0

2.4.5 Community suite algorithms and displays

It is envisaged that the community will produce algorithms and displays, tailored to their specific needs, using high-level languages, such as:

- MatLab ® or Octave
- IDL ®
- Python
- R
- NCAR Command Language (NCL)

2.5 Core infrastructure

In order for the core suite to be fully functional, a number of infrastructure services are required.

These include:

- reading input data from sensors and other sources
- writing output data
- translating data into supported formats, primarily NetCDF
- moving data between hosts
- providing data to the algorithms
- serving data to the displays
- disseminating data to users

As with the displays, work on the core infrastructure will be a significant fraction of the total for the project.

2.5.1 Reading / writing and translating radial data

The core suite will provide the ability to read and write data in the following community formats, and to translate between them:

- CfRadial
- DORADE format files
- UF format files
- NEXRAD msg1 and msg31 format files
- OPERA HDF5 format files
- FORAY (NCAR's predecessor to CfRadial) (read-only support)
- Wind Profiler formats (NPN and ARM)
- simple ASCII exchange format (yet to be defined)

The core will also be able to read the following commercial formats:

- Vaisala ® Sigmet ® RAW volumes
- Gematronik ® raw XML volumes
- Vaisala / Radian Wind Profiler formats
- other commercial formats as applicable

Some commercial vendors are already providing their data in CfRadial format.

2.5.2 Platform types supported

The following radar/lidar/profiler sensor types will be supported:

- fixed
- mobile
- ship-borne
- airborne
- single polarization
- dual polarization
- profiler
- others to be added as required

2.5.3 Serving data to users, displays and algorithms

The NCAR base software, on which the core will be built, already includes applications for serving data from files to users, algorithms and displays. However, since this was originally built to support the MDV format, some changes will be required to handle NetCDF data in a seamless manner.

Some of this work has already been completed by projects funded by the FAA.

2.5.4 Languages and operating systems

The core suite infrastructure will be predominantly based on 3rd generation languages, specifically:

- C++
- FORTRAN
- Java

However, over time, core modules will include code in higher level languages such as MatLab® and Python.

The core will support the LINUX and Apple OSX® operating systems, both of which are variants of UNIX.

Decisions concerning the design of the core suite will be made with maximum portability in mind. It is a goal that some of the core will be able to run on Windows ® under the Cygwin or MinGW frameworks, but it is not yet known how practical that will be.

2.5.5 Support for external data sets

Support will be provided in the core for handling the following types of non-radar data:

Gridded data:

- Models: WRF NetCDF, GRIB, CF-NetCDF, MM5
- Satellite: McIdas AREA files, McIdas NetCDF, Eumetsat HRIT

Projection support will be provided for all relevant projections specified in the NetCDF CF convention.

Point data:

- Surface observations (land, ship-borne)
- METAR/SIGMET/AIRMET
- Lightning
- Soundings (CLASS, TTAA/TTBB, WMO)
- Profile observations (wind profiler, sodar, vertically pointing radar/lidar)
- Aircraft tracks and associated airborne measurements

3 Statement of work

The following outline lists the main tasks to be carried out under this proposal.

3.1 Core infrastructure

- Enhance data translation capabilities.
- Enhance support for radial data in CfRadial format.
- Enhance support for Cartesian data in CF-NetCDF format.
- Review available CF-NetCDF conventions for point data time series. Develop new conventions for this type of data, if required.
- Enhance data server applications for NetCDF support.
- Enhance data quality control tools and display (e.g., soloii).
- Enhance gridding and dual-Doppler analysis software.
- Enhance data integration tools and functionality (e.g., CIDD, Jazz).
- Integrate support for profile data (eg, wind profilers, lidar)
- Develop portable wind profiler algorithm implementations.

- Review formats and provide support for profile data ingest into model (including BUFR)
- Develop community code repository.

3.2 Displays

- Modify soloii to natively handle CfRadial.
- Complete Jazz development.
- Improve visualization of point data (surface observations), soundings, flight tracks etc.
- Develop CfRadial-based server for VChill, or similar display.
- Improve CfRadial and profiler/lidar support in IDV.
- Develop support for radar data in VAPOR if appropriate.
- Complete xprof wind profiler display tool, adding integration with multiple data sources
- Investigate other web-enabled display technology as appropriate.

3.3 Community tools

- Organize workshops for the user community for update of the core suite and a forum to exchange community-developed algorithms.
- Provide training on the core suite.
- Develop documentation on the core suite.
- Provide outreach to the community.

3.4 Required resources

The following are the estimated resources required for this project:

Development:

- Software engineering: 2.5 senior persons x 3 years
- Science support: 0.3 persons x 3 years
- Management: 0.2 persons x 3 years

Long term support:

- software engineering: 0.5 persons per year
- science: 0.2 persons per year

3.5 Building on prior work

The success of this proposal depends on a large body of software that has been completed previously, funded by a variety of sources. The intention of the proposed work is to ensure that this prior work is made available in a manner which is useful to the research community, and that it can be maintained, supported and updated as needed. ***Without this extra effort it is likely that access to the previous work will be lost.***

The software engineering funding requested for this proposal is 7.5 person-years. This will build on the estimated 87 person-years of effort which have already been funded (see table below). The compelling point is that for every person-year expended on this work now, over 10 person-years of prior work will be made available to the research community.

Table 4 lists the estimated effort which has been funded to date:

Table 4. Software list including the developing organization, funding source and estimated total resource involved in development and upgrade. The years in parenthesis indicate the first available date of the software.

Item	**Organization**	**Project**	**Funding source**	**Estimated person-years**
Sprint, Cedric (1983)	MMM	Gridding, Dual Doppler	NSF base	10
Reorder (1986)	EOL (ATD)	Gridding for airborne radar	NSF base	3
soloii, Dorade (1991)	EOL (ATD)		NSF base	5.5
CfRadial (2010)	EOL	SPOL-Ka	NSF base	0.5
Particle identification (1999)	EOL	SPOL-Ka	NSF base	1
Algorithms (1998)	RAL	AWRP	FAA	14
Algorithms (2002)	EOL	NEXRAD enhancement	NWS	3
Data system infrastructure (1999)	RAL	AWRP AOAWS	FAA Govt of Taiwan	12
Data display and integration (CIDD - 1998, Jazz - 2011)	RAL	AWRP, AOAWS	FAA, Govt. of Taiwan	13
VDRAS multi-Doppler wind field estimation (1998)	RAL	Convective weather	FAA	4
Automated Airborne Doppler	EOL	ARRA + base	NSF	3

Item	Organization	Project	Funding source	Estimated person-years
radar data QC and navigation correction (2011)				
NIMA wind profile estimation (Cornman et al 1998)	RAL	Airport safety	FAA	10
ICRA clutter removal (Merritt 1995)	NOAA	NPN	NOAA	3
Wind Profiler algorithms and display	EOL	ISS	NSF base	5
Total:				**87**

4 Glossary

AIRMET: Airmen's Meteorological Information

AOAWS: Advanced Operational Aviation Weather System, funded by the Civil Aeronautics Administration of Taiwan

ARRA: American Recovery and Reinvestment Act

ATD: Atmospheric Technology Division at NCAR – previous name for EOL

ARM: Atmospheric Radiation Measurement program at the Department of Energy

AWRP: The FAA Aviation Weather Research Program

BUFR : WMO Binary Universal Form for the Representation of meteorological data

CfRadial: NetCDF format for radial radar and lidar data

CF-NetCDF: Climate and Forecasting conventions for NetCDF

CIDD: Cartesian Integrated Data Display

CLASS: Cross-chain Loran Atmospheric Sounding System

DORADE: Doppler Radar Data Exchange format

EOL: Earth Observing Laboratory at NCAR

FAA: Federal Aviation Administration

GRIB: Gridded Binary format – WMO format for compressed gridded data

IDV: UNIDATA Integrated Data Viewer

HDF5: NASA Hierarchical Data Format, upon which NetCDF4 is based.

HRIT: High Rate Information Transmission, for satellite data, Eumetsat

ICRA : Intermittent Clutter Reduction Algorithm for wind profiler data

Jazz: not an acronym – Java display to replace CIDD

MDV: Meteorological Data Volume format for Cartesian data, developed at RAL

METAR: aerodrome routine meteorological report

MMM: Mesoscale and Microscale Meteorology Division at NCAR

NCAR: National Center for Atmospheric Research

NetCDF: Network Common Data Format

NEXRAD: NWS Next generation Radar (WSR-88D)

NIMA : NCAR Improved Moments Algorithm for wind profiler data

NPN: NOAA National wind Profiler Network

NSF: National Science Foundation

NWS: National Weather Service

OPERA: Operational Programme for the Exchange of weather RAdar information

RAL: Research Applications Laboratory at NCAR

SIGMET: Significant Meteorological Information

soloii: radial coordinate display for analysis and data quality editing

TITAN: Thunderstorm Identification, Tracking, Analysis and Nowcasting

TTAA/TTBB: WMO formats for soundings

UF: Universal Format for radial radar data

UNIDATA: diverse community of over 160 institutions vested in the common goal of sharing data, and tools to access and visualize that data

VAPOR: Visualization and Analysis Platform for Ocean, Atmosphere, and Solar Researchers

VChill: Java-based viewer for radial data, developed by CSU/CHILL

VDRAS: Variational Doppler Radar Assimilation System

WRF: Weather Research and Forecasting model

WMO: World Meteorological Organization

XML: Extensible Markup Language

Zebra: data system and display software developed by EOL in the 1990s

5 References / links

[1] DORADE: http://www.ral.ucar.edu/projects/titan/docs/radial_formats/ - dorade

[2] For 2005 community survey responses, see http://www.eol.ucar.edu/cds/2005_retreat/community_full_responses.html

[3] For responses to the 2009 UCAR Community Survey as they pertain to the Earth Observing Lab, see http://www.eol.ucar.edu/cds/2009_UCAR_survey/2009_UCAR_Community_Survey_for_EOL.htm

[4] April 2010 University of Washington Site Visit Report, http://www.eol.ucar.edu/cds/2010_Site_Visits/UofWash_April2010_full.pdf

June 2010 Nolan Atkins/Lyndon State Interview Report, http://www.eol.ucar.edu/cds/2010_Site_Visits/Lyndon_State_Nolan_Atkins_June_7th_2010.pdf

August 2010 North Carolina State Site Visit Report, http://www.eol.ucar.edu/cds/2010_Site_Visits/NC_State_August2010.pdf

October 2010 Colorado State University Site Visit Report, http://www.eol.ucar.edu/cds/2010_Site_Visits/CSU_10_04_11.pdf

[5] CfRadial: http://www.ral.ucar.edu/projects/titan/docs/radial_formats/cfradial.html

[6] NetCDF CF: http://cf-pcmdi.llnl.gov/

Other References

IWRF time series: http://www.ral.ucar.edu/projects/titan/docs/radial_formats/#iwrf

Lee, W., P. Dodge, F. D. Marks Jr. and P. Hildebrand, 1994: Mapping of Airborne Doppler Radar Data. *J. Atmos. Oceanic Technol.*, 11, 572 – 578.

James, C.N. and R.A.Houze. Jr., 2001: A Real-Time Four-Dimensional Doppler Dealiasing Scheme. J. Atmos. Oceanic Technol., 18,1674-1683.

Jazz examples: http://www.ral.ucar.edu/projects/jazz/

Lim, Eunha, Juanzhen Sun, 2010: A Velocity Dealiasing Technique Using Rapidly Updated Analysis from a Four-Dimensional Variational Doppler Radar Data Assimilation System. J. Atmos. Oceanic Technol.., 27, 1140–1152.

Cornman, L.B., Goodrich, R.K., Morse, C.S., and Ecklund, W.L., 1998: "A Fuzzy Logic Method for Improved Moment Estimation From Doppler Spectra", J. Atmospheric and Oceanic Technology, 15, p 1287-1305.

Merritt, D. A., 1995: A statistical averaging method for wind profiler Doppler spectra. J. Atmos. Oceanic Technol., 12 (5), 985–995.

Weber, B. L., and D. B. Wuertz, 1990: Comparison of rawinsonde and wind profiler radar measurements. J. Atmos. Oceanic Technol., 7, 158-175.