AIRNOW INTERNATIONAL (AIRNOW-I) DRAFT SPECIFICATIONS DOCUMENT

By:
Timothy S. Dye
Alan C. Chan
Eric A. Gray
Judd E. Reed
Daniel A. Pryden
Sonoma Technology, Inc.
Petaluma, CA

John White
Phil Dickerson
Scott Jackson
U.S. Environmental Protection Agency
Research Triangle Park, NC

Version 1
STI-905403.06-3282

April 8, 2008

This PDF document contains blank pages to accommodate two-sided printing.

TABLE OF CONTENTS

Section **Page**

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

Term	Definition
Aggregate	Process of averaging shorter-term data values into longer-term values (e.g., averaging 60 1-minute values into an hourly averaged value)
AIRNow DMS	AIRNow Data Management System (DMS) that processes, quality controls, and validates data
AIRNow-I	Collection of software components that process, quality control, and distribute data; also generate maps and files
AIRNow Info Service	Information service that allows distribution of data and map products
AIRNow International	The overall program to develop the AIRNow-I software and build a community of practice on the methods, challenges, and approaches for collecting, processing, and communicating air quality data and forecasts
AIRNow International Partners	Organizations that are collaborating and exchanging data using the AIRNow-I system
AIRNow Mapper	GIS mapping system that automatically creates maps showing air quality data
API	Air Pollution Index
AQI	Air Quality Index
AQS	Air Quality System
ArcGIS Engine with Spatial Analyst Extension	A developer product for creating custom GIS desktop applications. ArcGIS Engine provides application programming interfaces (APIs) for COM, .NET, Java, and C++.
ArcMap	A component of ESRI's ArcGIS Geographic Information System (GIS). It is developed as client software specifically for the Microsoft Windows environment to enable mapping and presentation of ArcGIS data.
Arc objects	A collection of software components with GIS functionality and programmable interfaces. Customization is performed using a COM-compliant programming language such as C#.
ArcSDE	Technology that is an integrated part of ArcGIS Desktop and ArcGIS Server. It acts as the database access engine to spatial data, its associated attributes, and metadata stored within a relational database management system (RDBMS).

Term	Definition
Chain-of-Custody	Documentation showing the custody, control, transfer, analysis, and disposition of physical and electronic data and information.
DAL	Data access layer (DAL), part of a computer program that provides simplified access to data stored in a relational database
Data Products	Data files and web service that provide data from AIRNow-I
DMC	Data Management Center
EPA	U.S. Environmental Protection Agency
ESRI	Environmental Systems Research Institute (GIS software company)
GEOSS	Global Earth Observation System of Systems (www.epa.gov/geoss)
GIS	Geographic information system
Map Products	Maps produced by AIRNow Mapper
Map template	Base maps created with ArcMap and stored as *.MXD files.
OGC	Open GIS Consortium
Parameter	Air quality or meteorological variables (e.g., ozone, carbon monoxide)
RSS	Really Simple Syndication
SEMC	Shanghai Environmental Monitoring Center
SEPB	Shanghai Environmental Protection Bureau
SQL	Structured Query Language (SQL), a database computer language designed for the retrieval and management of data in relational database management systems (RDBMS), database schema creation and modification, and database object access control management.
WCS	Web Coverage Service (WCS) that provides an interface allowing requests for geographical coverage across the web using platform-independent calls.
WFS	Web Feature Service (WFS) that allows a client to retrieve and update geospatial data encoded in Geography Markup Language (GML) from multiple Web Feature Services. The specification defines interfaces for data access and manipulation operations on geographic features, using HTTP as the distributed computing platform.
Wiki	Web technology that promotes collaborative sharing and updating of information

Term	Definition
WMS	Web Map Service (WMS) that produces maps of spatially referenced data dynamically from geographic information. WMS-produced maps are generally rendered in a pictorial format such as .png, .gif, or .jpg, or occasionally as vector-based graphical elements in Scalable Vector Graphics (SVG) or Web Computer Graphics Metafile (WebCGM) formats.

1. INTRODUCTION

1.1 PURPOSE

The AIRNow program (www.airnow.gov) processes, maps, and communicates air quality conditions and forecasts to the public. This program has been operating in the United States since 1997 and has undergone many changes, additions, and improvements to make it a very successful program. Recently, several countries (China, Brazil, Poland, South Korea, and South Africa) have expressed interest in an AIRNow-like system. However, the current U.S. system is a customized collection of software programs and databases that are not easily transferable.

To overcome this issue, the U.S. Environmental Protection Agency (EPA) plans to develop software to improve the U.S. AIRNow program and to provide an international version of the AIRNow software that includes data processing, quality control (QC), system diagnostic monitoring, mapping, and distribution. This software package, called AIRNow-I, will provide the basic features of the AIRNow program based on AIRNow's technology and will be fully integrated with a geographic information system (GIS) to provide flexible mapping tools applicable worldwide. It will run on personal computers and will be made available to countries that are interested in distributing data to the public and decision makers and in exchanging data with other organizations.

The AIRNow-I software package is part of the larger AIRNow International Program. The goal of the program is to develop a community of organizations that collect, process, exchange, and communicate air quality observations and forecasts. Collaboration among AIRNow International Partners is important to help improve the Partners' understanding of air quality and keep the software systems updated to meet current and future needs. In addition, collaboration helps all organizations and countries learn from each other about methods, challenges, and approaches in collecting, processing, analyzing, and communicating air quality observations and forecasts.

This specifications document describes the AIRNow International Program as well as the features and functions of the AIRNow-I system. Potential international partners can review the development plans described in this document and provide insights, suggestions, and recommendations so that the AIRNow-I system can meet as many needs as possible. The AIRNow-I system is planned for release in fall 2009. Please provide any comments to John White at white.johne@epa.gov and Tim Dye at dye.tim@epa.gov.

1.2 BENEFITS

The AIRNow-I software package will offer several benefits:

- Enable automated processing and QC of air quality data and generation of air quality information products (for example, maps and charts) with minimal human oversight
- Provide readily available air quality information to decision makers, especially during special and international events

- Facilitate decision makers' understanding of air quality and related health issues
- Allow decision makers to promptly respond to air quality-related events, for example, dust storms and forest fires, and take appropriate measures
- Encourage better maintenance of air quality instruments, which will improve quality assurance and data quality for other air quality management activities (e.g., forecast model verification)
- Provide a tool to address air quality monitoring and forecasting issues and challenges
- Promote regional data exchange among adjacent cities, provinces, and countries through explicit data exchange protocols and guidelines.
- Improve understanding of transboundary air quality issues, promoting awareness and cooperation among bordering air quality agencies

1.3 AIRNOW-I MODULES

AIRNow-I will consist of a suite of modules (software programs and schedulers) centered around an AIRNow Database. The AIRNow Database is a relational database that will store data, site information, metadata, program settings, and status and event log information. The AIRNow International Program, including the AIRNow-I modules, is illustrated in **Figure 1-1**. The modules are described as follows:

1. AIRNow Data Management System (DMS). This module will perform the following functions:

- Ingest air quality and meteorological data (real-time and historical)
- Ingest data from other AIRNow-I systems (in the future)
- Quality control data (real-time and historical)
- Create average and aggregates of the data
- Automatically process data (will allow manual processing)
- Configure and compute different Air Quality Indices (API, PSI, AQI, etc.)
- Enable addition, deletion, and modification of data, parameters, sites, agencies, and users
- Monitor system performance and outputs
- Create data reports

2. AIRNow Mapper. This module will perform the following functions using ESRI GIS functions and programs:

- Extract data points
- Perform interpolation and contouring
- Blend different maps
- Generate maps, animate maps, and generate data files

3. AIRNow Info Service. This module will perform the following functions:

- Create mechanism to share and distribute data
- Use data standards for data ingest, data distribution, and sharing
- Conform to interoperability standards and conventions being developed under the Group on Earth Observations, World Meteorological Organization, Open Geospatial Consortium
- Provide flexibility to add new data outputs
- Provide ability to control data distribution and sharing

4. AIRNow Administration. This module will be a user interface that allows users to control the features and functions of AIRNow-I. The user interface will allow multi-language capabilities, which are described in Section 2.3. The Administration module will allow the following controls:

- Configure the AIRNow DMS module
- Configure the AIRNow Mapper module
- Configure and control how data are processed and distributed with the AIRNow Info Service module
- Monitor the flow and processing of data, event logs, and data products.

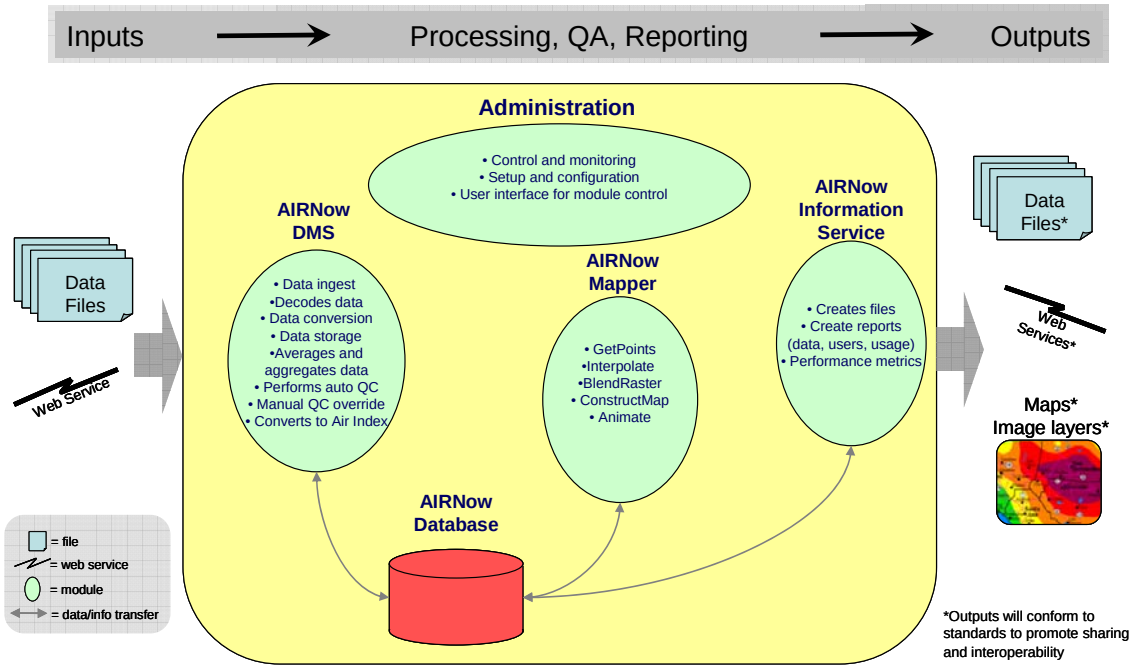

Figure 1-1. Conceptual and functional design of the AIRNow International program including a schematic of AIRNow-I modules and data flow.

1.4 PRODUCTS

The products of the AIRNow-I system will include the following types of data and information:

- Maps showing air quality data (see **Figure 1-2**) in the form of color-coded contoured maps, animated contoured maps, color-coded point maps, and site maps.
- Quality-controlled data with chain-of-custody information suitable for submitting to EPA's Air Quality System (AQS) and other data systems.
- Data files and services that allow easy exchange of information and that conform to international data standards. These formats are outlined in Section 2.4.
- Various types of data reports and graphs (some examples are shown in **Figure 1-3**).

The AIRNow International Program web site (www.AIRNowInternational.org) will provide the following educational information:

- Manuals for AIRNow-I that include detailed installation and configuration instructions and examples
- Web tutorials that show the features and functions of AIRNow-I
- A portal for sharing information about AIRNow-I and developing a community of practice to share experience about data processing, mapping, and public communication

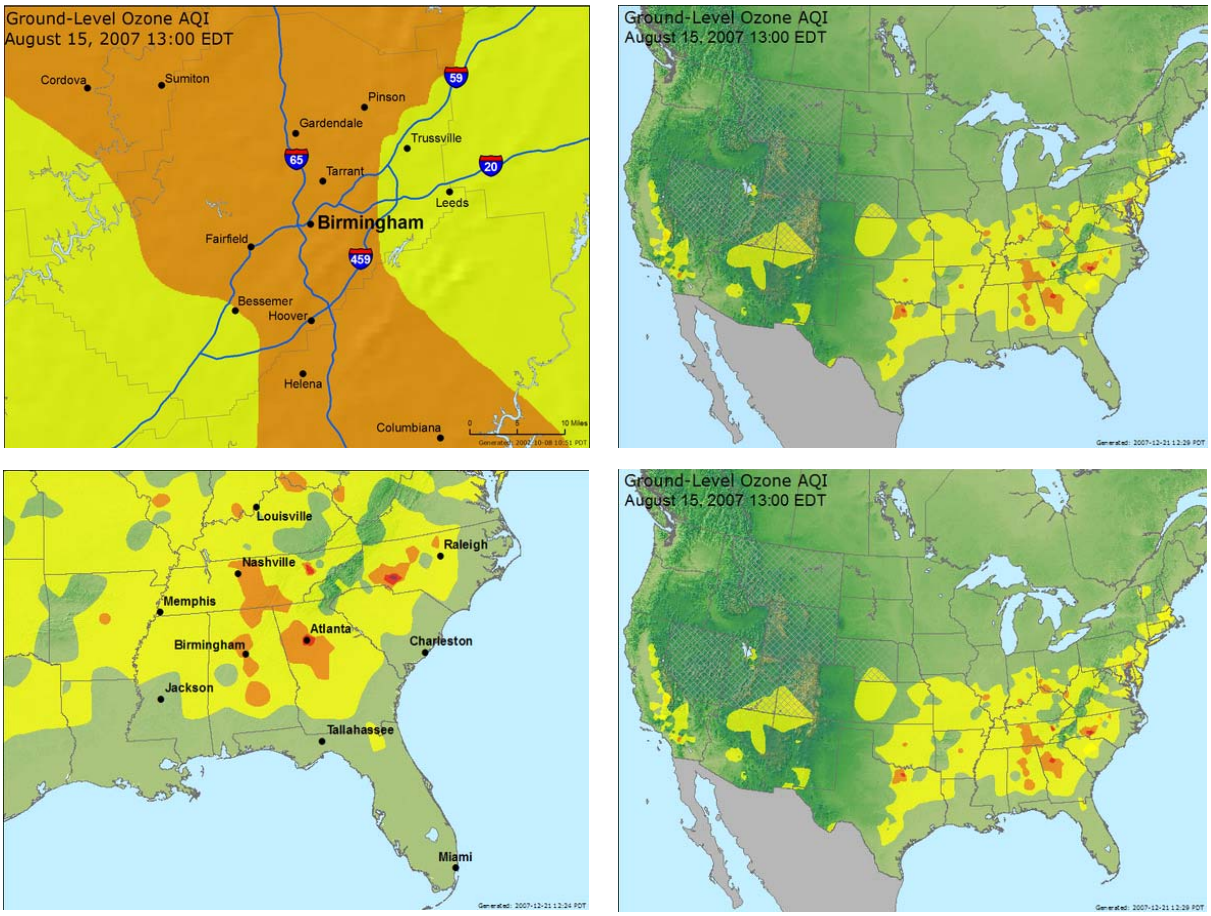

Figure 1-2. Example maps created with AIRNow Mapper.

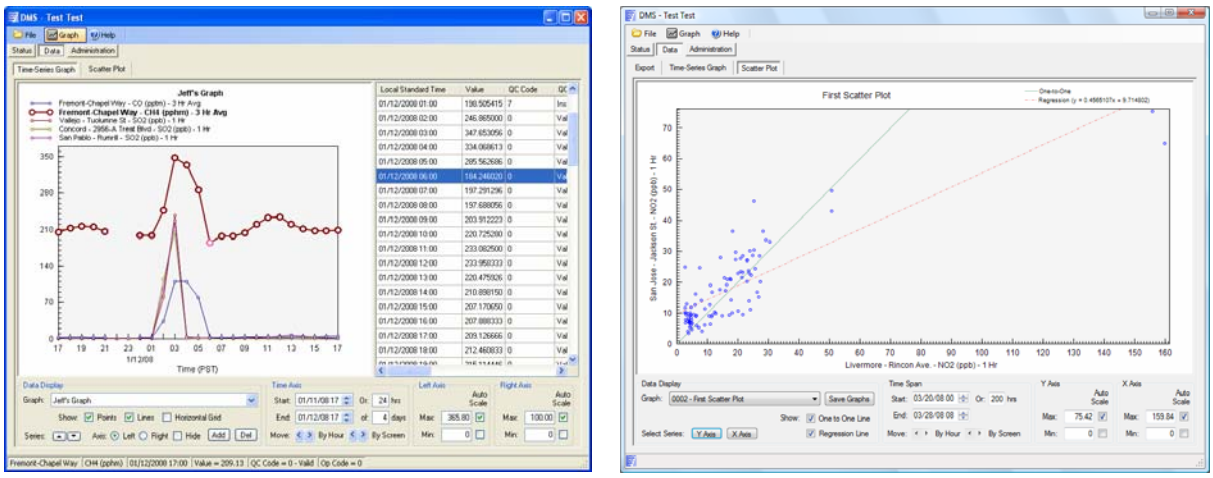

Figure 1-3. Example graphs generated from AIRNow DMS.

1.5 ASSUMPTIONS

We made a number of assumptions in developing this specifications document for AIRNow-I. In this section, AIRNow International Partners refers to the city, local, state, federal, and international organizations that will operate AIRNow-I within their country or region. Our list of assumptions includes the following:

- Interested organizations must agree to the terms and conditions outlined in the AIRNow International Partnership Agreement in Section 8.
- Part of the requirement for obtaining and using the AIRNow-I system is that the participating Partners distribute their data to other organizations/agencies (i.e., Partners) involved in the AIRNow International community. It is the responsibility of the Partners to determine how much and to what extent the data and information will be distributed.
- A continuous monitoring network of several air quality and/or meteorological monitors is needed for AIRNow-I. It is the responsibility of the Partners to have a functional system to routinely (hourly or more frequently) transfer data from monitoring sites to a central facility at the Partner organization (where AIRNow-I will be located). AIRNow-I does not include software to poll or transfer data from a monitoring site to a central location at the Partner organization.
- The commercial hardware and software listed in Sections 2.1 and 2.2 are required to run the AIRNow-I system. It is the responsibility of the Partners to purchase, install, and maintain this hardware and any associated software licenses needed to operate the AIRNow-I system.

2. SYSTEM DESIGN

2.1 SYSTEM SOFTWARE

The software system will largely be based on Microsoft .Net framework with a Microsoft SQL-Server database. In determining the types of software and operating systems to use for AIRNow-I, we received input from many organizations and tried to balance using software that is widely available, yet modestly priced. The software requirements for each of the AIRNow-I modules are shown in **Table 2-1**.

Table 2-1. Software requirements for each AIRNow-I module.

AIRNow-I Module	Software Requirements
AIRNow Database	SQL Server 2005 Work Group, Standard, or enterprise editions. Microsoft Windows 2000 Server with Service Pack (SP) 4 or later; or Windows Server 2003 Standard Edition, Enterprise Edition, or Datacenter Edition with SP 1 or later; or Windows Small Business Server 2003 with SP 1 or later.
Administrative User Interface	.NET Framework 2.0 Microsoft Windows 2000, 2003, XP, or Vista
AIRNow DMS	.NET Framework 2.0 Microsoft Windows 2000, 2003, XP, or Vista
AIRNow Mapper	.NET Framework 2.0 Microsoft Windows 2000, 2003, XP, or Vista ESRI ArcGIS Engine with Spatial Analyst Extension
AIRNow Info Service	.NET Framework 2.0 Microsoft Windows 2000, 2003, XP, or Vista

2.2 HARDWARE

The computer configuration in **Figure 2-1** shows that AIRNow-I can run on a single computer for small networks but the recommended configuration is three computers. The minimum hardware requirements for AIRNow-I are listed in **Table 2-2**.

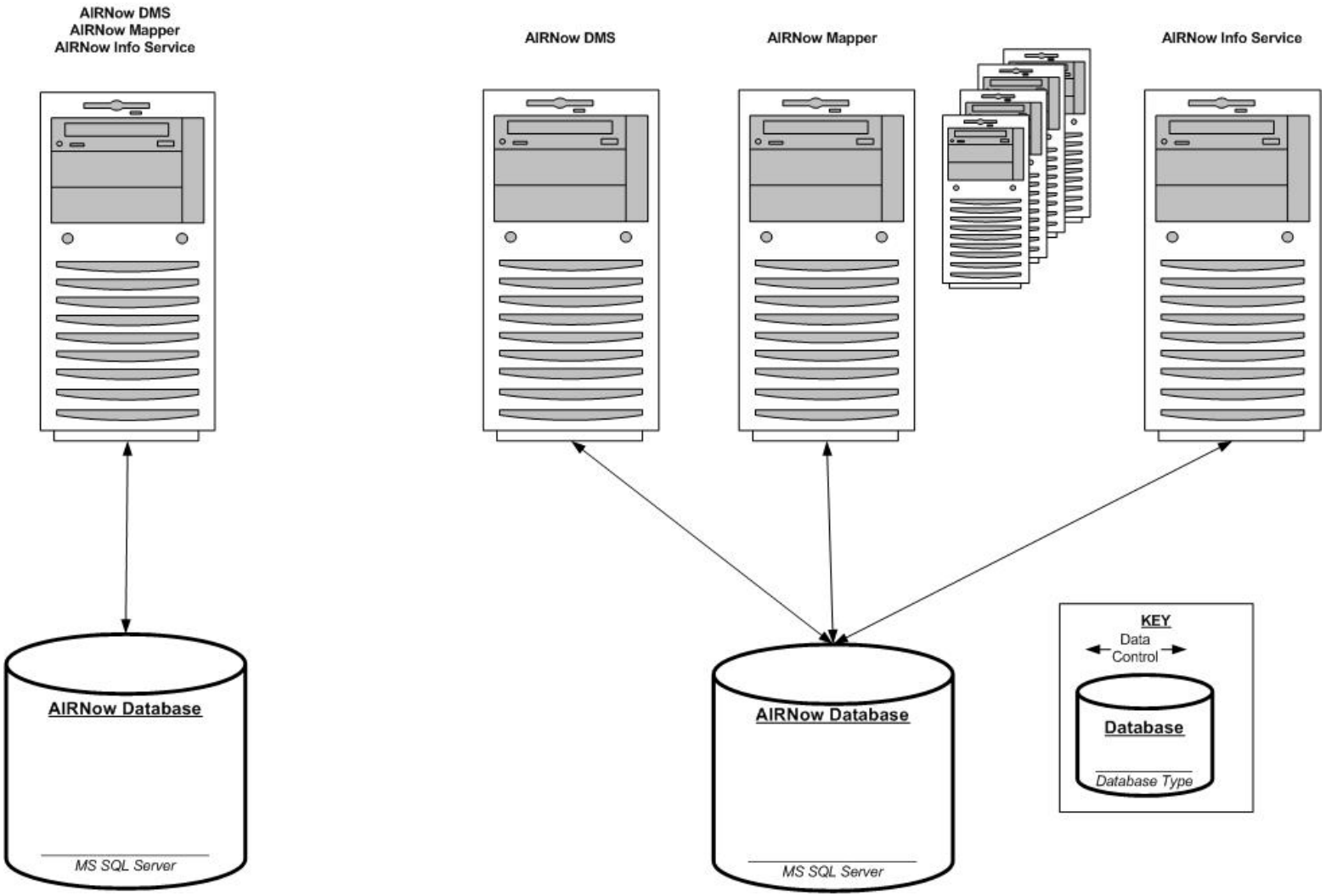

Figure 2-1. AIRNow-I computer configuration.

Table 2-2. List of the hardware functions and requirements for AIRNow-I.

Hardware Component	Function(s)	Specifications and Minimum Requirements
Database Server	Hosts AIRNow SQL Server 2005 for AIRNow Databases, DMS, and Mapper	32-bit: 600-megahertz (MHz) Pentium III-compatible or faster processor; 1-gigahertz (GHz) or faster processor 64-bit: 1-GHz AMD Opteron, AMD Athlon 64, Intel Xeon with Intel EM64T support, Intel Pentium IV with EM64T support processor; 1-GHz (recommended)
Mapper Server	Produces grids, maps, animations	32-bit: 600-megahertz (MHz) Pentium III-compatible or faster processor; 2-GHz or faster processor recommended 64-bit: 1-GHz AMD Opteron, AMD Athlon 64, Intel Xeon with Intel EM64T support, Intel Pentium IV with EM64T support; 2-GHz or faster processor recommended processor
Work Stations	AIRNow DMS, Mapper, Info Service, Administrative User Interfaces	Any Windows-compatible PC running Microsoft Windows 2000, 2003, XP, or Vista, and .NET Framework 2.0

2.3 LOCALIZATION APPROACH (MULTI-LANGUAGE)

AIRNow-I will be designed to support many languages. The language used in both the generated data products (e.g., maps) and the system's administrative user interfaces can be set by the administrator. The system will initially support English, Chinese, Portuguese, and Spanish. Support for additional languages can be added with minimal effort in future releases of AIRNow-I.

Configuring AIRNow-I for languages other than English requires that an administrator perform these one-time tasks for each language:

1. Create a language translation file. The language translation file is an ASCII file that contains text labels (i.e., labels in the user interface) and their corresponding translated text. Each record in the file will contain the label/translation pairs for every text label in the AIRNow-I system. Translated text for each language will be stored in a separate language translation file.

2. Compile the language translation file (with the Microsoft tool "resgen", <http://msdn2.microsoft.com/en-us/library/ccec7sz1.aspx>) into a binary language resource file that can be loaded by AIRNow-I at run time.

When a user starts the AIRNow-I user interface, the interface will check the system's configuration and select the appropriate language resource file. This resource file will supply all of the language-specific data for use in the user interface. This approach allows a single program to support multiple languages.

The AIRNow Database can store text, such as place names or measurement dimensions, in any language. However, data values will be stored as numerical values. The actual table and field names in the database (see Section 3.5) cannot be translated.

The text on maps produced by AIRNow Mapper is derived from format strings stored in pre-defined map templates. Map templates are created using ArcMap (a program by ESRI). A map template can include any character supported by ArcMap and the ArcSDE. When the map is produced, the format string is modified to contain specific information such as date, time, and parameter.

AIRNow-I includes user interfaces written in C# or VB.Net that use the Microsoft .net framework and that are developed using Microsoft Visual Studio. We will use Visual Studio's internationalization and localization tools to provide multi-language capability. Even with the aid of these tools, internationalization remains challenging. Where practical, we will use symbols and metaphor images instead of text to reduce the translation effort, increase the usability in any language, and provide familiar landmarks for those using the system in multiple languages.

2.4 DATA SHARING/DISTRIBUTION STANDARDS AND APPROACHES

Data will initially be distributed in the following file formats and web services by the AIRNow Info Service module:

- Comma Separated Value (CSV) – for data, sites, parameters, etc.
- NetCDF with Climate Forecast (CF) conventions (www.unidata.ucar.edu/software/libcf/)
- ESRI files (in ESRI Raster and Shapefile formats)
- Open Geospatial Consortium (OGC) web services (Web Mapping Service, Web Feature Service, Web Coverage Service)
- Google Earth (KMZ) files

2.5 SOURCE CODE DISTRIBUTION APPROACH

The approach to building and maintaining AIRNow-I is to gradually move toward an open source code distribution system for AIRNow International Partners (see Section 8). There are several reasons for making an open system: to provide transparency so organizations

understand how their data are processed and distributed; to help organizations better maintain and improve on the AIRNow-I software; and to help foster a community of practice for advancing all organizations' knowledge of air quality data, processing, mapping, and public communication.

We propose a two-phased approach for source code distribution.

- Phase I. From the launch of AIRNow-I and for roughly two years thereafter, we propose to make the source code for the AIRNow Info Service and the AIRNow Database (tables and stored procedures) available to AIRNow International Partners. By accessing the Database source code agencies will be able to examine how their data are stored and handled. Open access to AIRNow Info Service source code will help create transparency for the module that provides data to outside organizations. This openness will likely result in quicker adoption of AIRNow-I. In addition, we anticipate that AIRNow Partner organizations (initially SEPB) will develop applications to output different types of files using the data transformation library (described in Section 3.4.2).
- Phase II. After the initial launch of AIRNow-I in Shanghai and after installation in other countries, we propose making all AIRNow-I source code available for AIRNow International Partners. The source code would include the AIRNow DMS, Mapper, and Administration modules as well as the items listed in Phase I.

3. SOFTWARE PRODUCT DESCRIPTION

3.1 ADMINISTRATION USER INTERFACE

3.1.1 Overview

The AIRNow-I User Interface allows administration of all of the AIRNow-I modules (DMS, Mapper, Info Service) and administration of the AIRNow Database. The features and functions are described in this section.

3.1.2 System Design and Functions

A schematic of the control and data flow of the AIRNow-I User Interface is shown in **Figure 3-1**. Through the user interface, users will be able to configure and control each module and monitor the processes of each module. As discussed in Section 3.3, the user interface will be built to handle multiple languages. This section describes the user interface for each module.

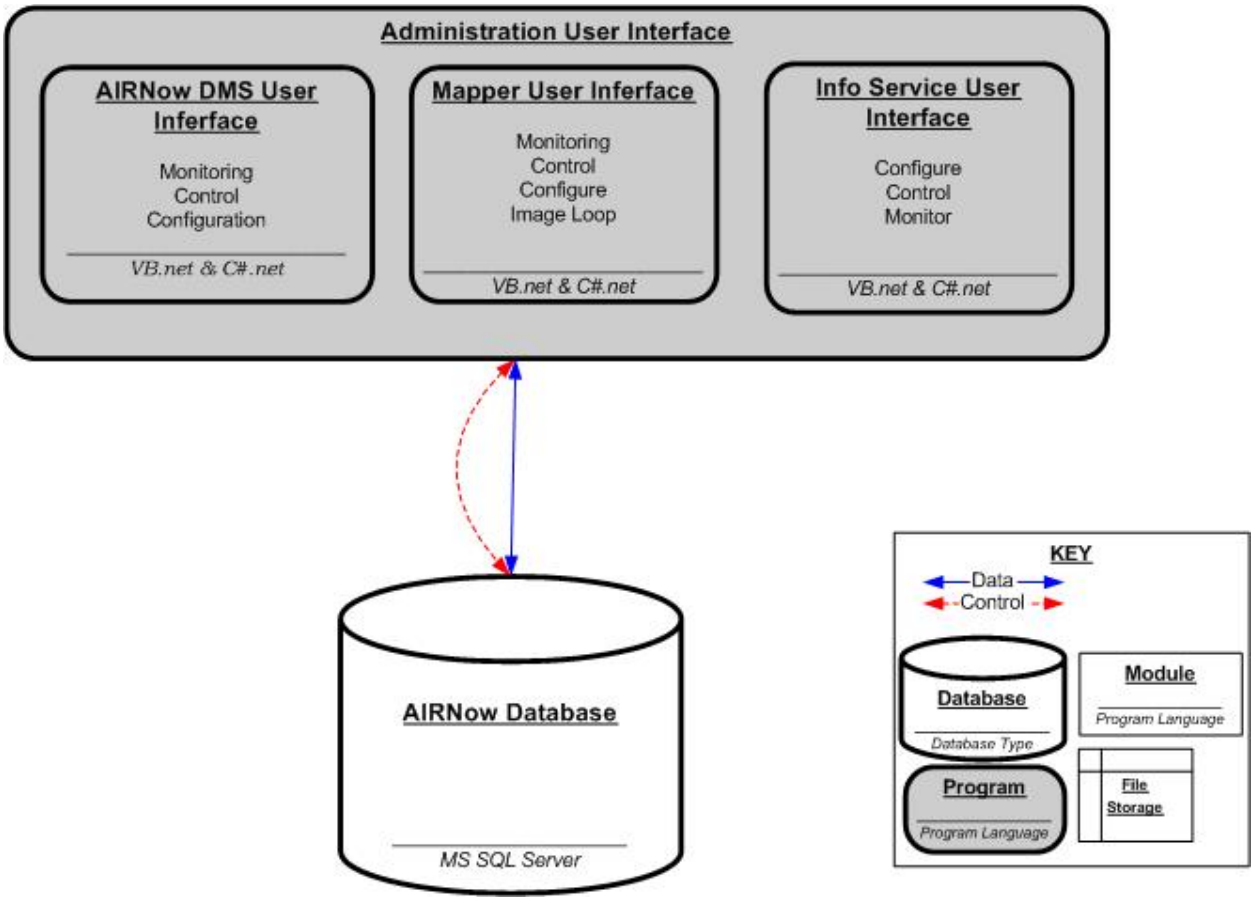

Figure 3-1. Control and data flow of the AIRNow-I user interface.

AIRNow DMS Module User Interface

Function: Allows users to configure all DMS settings, configure QC processing, edit and QC data, and monitor all processes.

Data Inputs: Settings are made via the user interface.

Data Outputs: User settings and configuration will be stored in the AIRNow Database.

Operation: Through the AIRNow DMS interface users will be able to perform the following functions:

- Control data input and ingest
- Edit user information
- Edit site information
- Edit instrument information
- Edit parameter information
- Edit site QC information
- Configure QC processing and rules
- Display time series, graph settings, and data information
- View logs (Site, Instruments, QC Status, Events)
- Review QC status

Code: VB.net or C#.net using Visual Studio 2008

AIRNow Mapper Module User Interface

Function: Allows users to configure mapping production to create new maps, edit map configurations, schedule map production, monitor mapping progress, etc.

Data Inputs: Settings are made via the user interface.

Data Outputs: User settings and configuration will be stored in the AIRNow Database.

Operation: Through the AIRNow Mapper interface, users will be able to perform the following functions:

- Configure individual map production using a scheduler (Dispatcher)
- Set up file output and path options
- Manually create a map
- Create new grids using the Blend Raster module to merge multiple grids into a single grid
- Monitor map production and status
- Monitor and adjust the priority settings in Dispatcher for generating maps

Code: VB.net or C#.net using Visual Studio 2008

AIRNow Info Service Module User Interface

Function: Allows users to configure the Info Service to produce new data files, authenticate new users, schedule production tasks, etc.

Data Inputs: Settings are made via the user interface.

Data Outputs: User settings and configuration will be stored in the AIRNow Database.

Operation: Through the Info Service interface, users will be able to perform the following functions:

- Authenticate user accounts, edit user preferences, monitor user activities, and delete user accounts
- Create data file products by associating parameters, sites, and data formats, and schedule the production of these data files
- Monitor data file production and web service activities (i.e., data requests made via the web server)
- Monitor and adjust the priority settings in Dispatcher

Code: VB.net or C#.net using Visual Studio 2008

3.2 AIRNOW DATA MANAGEMENT SYSTEM (DMS)

3.2.1 Overview

The AIRNow DMS is the core for data processing in the AIRNow-I system. It enables QC and management of air quality and meteorological data (**Figure 3-2**). This section describes the general operation of the AIRNow DMS, including hourly, daily, and specific tasks. Details of the core functions, data ingest, and system design are discussed in following sections.

The DMS will operate automatically once it is set up and configured. However, system operators have the ability to intervene manually. For example, data files are automatically processed and quality controlled with manual intervention such as graphical data review and interactive QC override. In addition, operators can modify monitoring site information, add parameters, and configure different air quality indices (AQI, API, etc.).

The system will be capable of decoding input files and ingesting hourly and sub-hourly data 24 hours a day. It will perform multiple tasks on either an event-driven or scheduled basis. For example, when data files are detected, the system will immediately process data, perform automated QC against several pre-defined criteria, and convert data into appropriate units and aggregate them. Furthermore, several daily tasks, including Air Index calculations, data report generations, data archiving into the AIRNow Database, will be performed. **Table 3-1** lists the major features of AIRNow DMS.

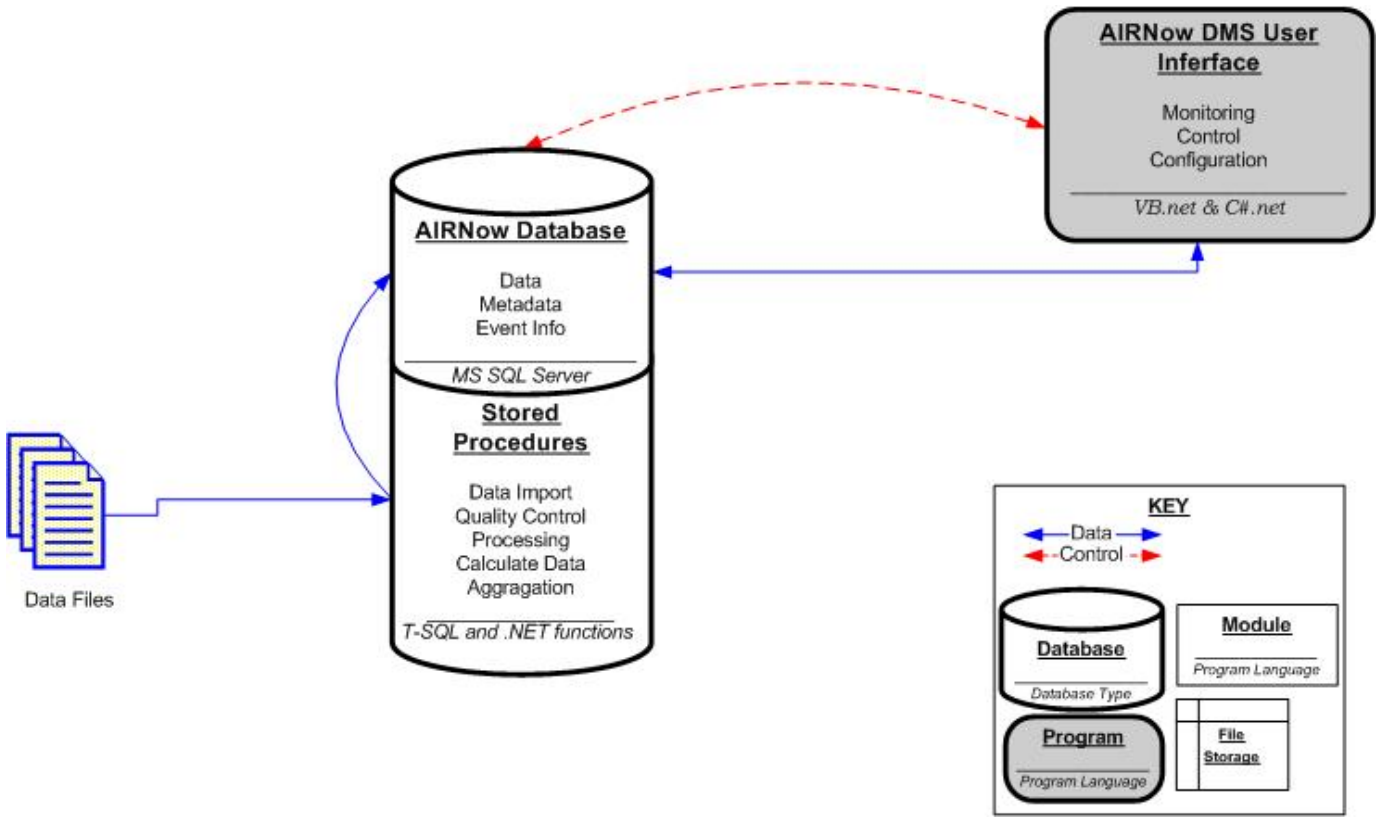

Figure 3-2. AIRNow DMS overview.

Table 3-1. Major processes and functions of AIRNow DMS.

Process	Functions
Decode data	Parse data from data files and web service
Ingest data	Process data and write to the AIRNow Database
Convert data	Determine whether data are delivered in pre-specified units; convert data based on conversion table in database.
Perform QC	Check data against site, parameter, and hour-specific QC criteria stored in the database; write QC flags; modify data if necessary.
Manual QC	Allow flagged data to be reviewed by DMS operators; make changes to/overwrite automated QC codes and data based on manual QC actions.
Calculate data	Read quality controlled data; aggregate data and calculate averages; convert averages into various pre-defined air quality indices.
Generate reports	Produce user-specified reports with quality-controlled data. This process can also be invoked for archived data.
Archive and store data	Move older, processed data from the operational data table in the AIRNow Database to a separate, archived data table.

3.2.2 System Design and Functions

Specifications of the core functions of the AIRNow DMS module are described in this section.

Data Processing

Function: Ingest numeric, raw data collected at monitoring sites. Real-time and historical data of any duration and frequency will be collected.

Data Inputs: CSV files and additional file formats (to be determined)

Data Outputs: Data will be written to the AIRNow Database through a SQL Server job.

Operation: The DMS processes imported and processed data using a scheduled SQL Server job consisting of a series of stored procedures. Data will be converted to common units in the database.

Code: MS SQL

Automated Quality Control

Function: Automatically QC data according to configurable criteria based on hour, site, and parameter.

Data Inputs: Data and QC parameters from the AIRNow Database.

Data Outputs: Data will be written to the AIRNow Database through a SQL Server job.

Operation: Apply multiple QC checks to site-parameter values from monitoring sites. Automatic QC checks are discussed in Section 3.2.3. Assign data QC code information based on QC check test results.

Code: MS T-SQL

Manual Quality Control and Data Editing

Function: Allow operators to visually review and edit data with a graphical user interface.

Data Inputs: Data values flagged by automated QC.

Data Outputs: Data will be written to the AIRNow Database through a SQL Server job.

Operation: Log data values failing automated QC checks in the database and delegate values to Manual QC Processing. DMS users review and edit data. Automatically log records with electronic chain of custody documenting changes made to data. Store fully reviewed data in the database.

Code: MS SQL

Data Averaging and Air Index Calculations

Function: Automatically average data in user-defined time intervals and convert averaged data into region-specific air indices.

Data Inputs: Data will be stored in the AIRNow Database.

Data Outputs: Data will be written to the AIRNow Database through a SQL server job.

Operation: Users define averaging time period and customize air index calculations through the user interface. Stored procedures in SQL aggregate data into different periods (8-hr, 24-hr average) and convert data values into air indices on a scheduled basis.

Code: MS SQL

3.2.3 Automatic Quality Control Checks

AIRNow DMS can run QC checks on both instrument diagnostics and air quality measurements. Each QC check identifies a specific test type, the input information required for the test, and the action DMS is to take if the test fails. QC check types are Minimum, Maximum, Zero, Sticking, Rate-of-Change, Range, and Buddy Checks.

Each QC check is site- and parameter-specific and has three possible outcomes:

1. If the QC check cannot be conducted because insufficient site or parameter values are available or the values have QC Codes indicating invalid data, no action is taken.
2. If the QC check passes (is true), no action is taken.
3. If the QC check condition fails (is false), the DMS will apply QC codes and/or result codes to the target data according to the QC check definitions. An entry is placed into the Chain-of-Custody Log indicating
    a. QC check date/time
    b. Target site-parameter, instrument, and data record (date/time, value)
    c. Test site-parameter, instrument, and data record(s) (date/time, value)
    d. QC test type and test values
    e. Target site-parameter data record adjusted value (if any)
    f. QC code action (if any)
    g. Result code action (if any)

The DMS can perform both automated and manual QC checks. Automated QC checks are performed after data ingest and are described below. Manual QC is performed by DMS users.

1. Automated QC checks are performed according to a hierarchical structure. Each level determines the order in which the tests are performed. All tests at a level must be performed before the tests at the next higher level can be started. The levels are
    - Self Check – QC checks performed on site- and parameter-specific data
    - Instrument Check – related data from the same site and instrument but a different parameter
    - Site Check – related data from the same site but a different instrument
    - Offsite Check – related data from different sites and parameters. Data are considered “related” if they have the same time stamps or consecutive prior time stamps along a

designated delta value, e.g., last three hours or minutes of data (hours or minutes is known by the nature of the parameter).

3. There is no limit to the number of checks that may be applied to an ingested data value.
4. When a particular check fails, no further automated QC will be performed for that ingested data value and a log record of the error is produced in the database with all relevant information required for a manual review of the failure. The QC check shows the QC Code applied to the data value, identifying it as having failed automated QC checks.
5. Additional processes may occur based on a user-assigned Result Code for a failed test.
6. Some checks may not be completed because the required data have not yet been ingested or have already been invalidated by a prior QC check. QC checks will be re-attempted with each new file ingest. When a designated end-time is reached, all remaining tests are run a final time before the data value is delegated to manual QC processing.

3.3 AIRNOW MAPPER

3.3.1 Overview

AIRNow Mapper is a module of AIRNow-I that can automatically produce maps of air quality data using GIS software. In addition, AIRNow Mapper will also allow merging of data grids to create complex grids that show air quality conditions in a comprehensive manner. **Figure 3-3** shows a schematic of the major features, data flow, and control of AIRNow Mapper. Details about the features and functions are described in Section 3.3.2. The software and hardware requirements to support AIRNow Mapper are listed in Sections 2.1 and 2.2.

Important design considerations for AIRNow Mapper include the following:

- Use existing commercial GIS software to perform complex interpolations and mapping instead of re-inventing it.
- Create a system that allows scheduling and automatic production of maps, yet also allows manual creation of maps.
- Develop software that enables users to blend (i.e., merge) different data sets (ozone, $PM_{2.5}$, PM_{10}, etc.) to create a single data set for mapping. This is important when trying to account for data sets with sparse monitoring networks and for incorporating satellite data into AIRNow.

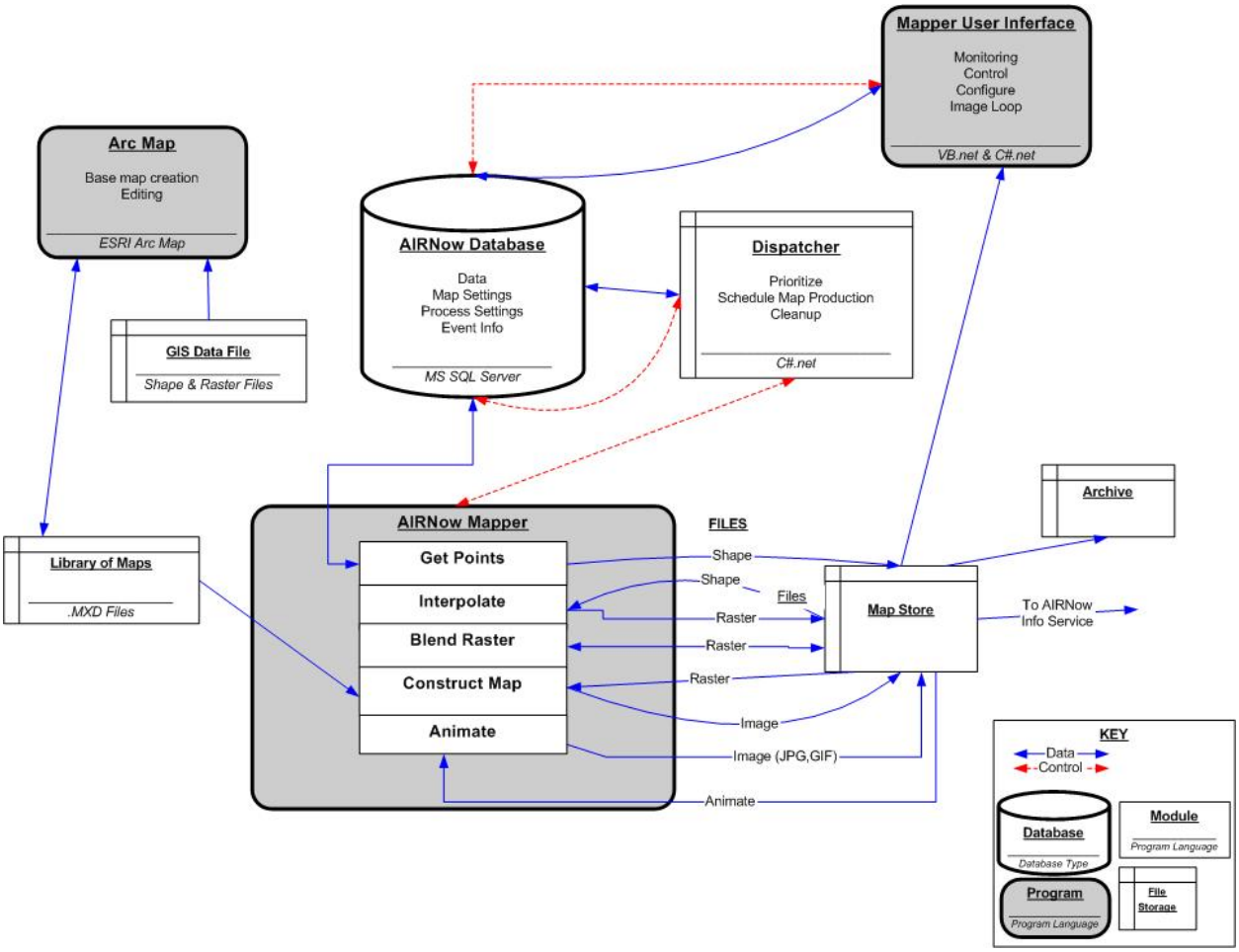

Figure 3-3. A schematic of the major features, data flow, and control of AIRNow Mapper.

3.3.2 System Design and Functions

AIRNow Mapper requires that setup tasks be performed manually; then the program runs final map creation tasks automatically. For each new map, a user must manually create a base map with ArcMap using GIS data (terrain, rivers and lakes, roads, etc.) and then store the map file on the AIRNow Mapper computer. Through the user interface, users can schedule automatic map production or generate maps manually.

AIRNow Mapper automatically generates maps using the Dispatcher module to

1. Continuously query the AIRNow Database for tasks to start. Once Dispatcher finds a task, it will begin building a list of tasks to make a map product.
2. Prioritize the cue of tasks and map products to produce
3. Launch AIRNow Mapper sub-modules (Get Points, Interpolate, Blend Raster, Create Map, and Animate) to generate the map
4. Update the AIRNow Database with event status (e.g., pending, completed, etc.)

The seven primary functions and features of AIRNow Mapper are described in the rest of this section.

Map Store

Function:	Archives maps in a large, fast, arrayed disk file system. It stores maps and map component products.
Data Inputs:	Mapper deposits map product files into the Map Store. It also deletes obsolete files as scheduled.
Data Outputs:	Mapper will retrieve files from the Map Store as need to create maps. The Information Service will also retrieve files from the Map Store and deliver them, or reformatted versions of them, to external data clients.
Operation:	An array of disk drives is used to maximize storage capacity and access speed. The file system is organized with folders and subfolders for separate data product types. Data files are stored within these folders. Each file's name includes the extent, units, and time of its data. File extensions are used to distinguish file format types: • Shapefiles (set of four files with extensions .shx, .shp, .prj, and .db) • Grids (.img) • Maps in several standard image file formats (.jpg, .png, .gif) • Animations (.gif)
Code:	(N.A.)

Dispatcher

Function:	A version of this program is constantly running on the Mapping server(s). Dispatcher receives schedule information from the database and manages a list of map products to build. Using the list of products, it determines which tasks to preformed next and executes the Mapper sub-modules.
Data Inputs:	Dispatcher relies on map production schedules and related data that are queried from the database. The status of scheduled work is also received for the database.
Data Outputs:	None
Operation:	The Mapping server(s) are configured to run Dispatcher continuously. If it terminates, a new copy is automatically started.
Code:	C# using Microsoft .net Framework

Get Points Sub-Module

Function:	Queries the database for air quality data and translates data into the format needed by the 'Interpolate' sub-module
Data Inputs:	Air quality data values and site location information from the database

Data Outputs:	ESRI Shapefile format data sets
Operation:	This program is started by Dispatcher and is controlled with command-line parameters
Code:	C# using Microsoft .net and ESRI ArcObjects frameworks

Interpolate Sub-Module

Function:	Uses ESRI ArcObjects to interpolate sample points into raster map layers. It also outputs a map layer that records the quality of fit between the input points and the resulting raster.
Data Inputs:	ESRI Shapefiles generated by the Get Points sub-module.
Data Outputs:	A raster map layer. Optionally, an additional map layer, depicting the interpolation's error or standard deviation.
Operation:	This program is started by Dispatcher and is controlled with command-line parameters.
Code:	C# using Microsoft .net and ESRI ArcObjects frameworks

Blend Raster Sub-Module

Function:	Combines map layers (grids or rasters) to produce new raster files. It performs basic image manipulations such as scaling, thresholding, addition, Boolean combination of images, etc. For example, to create an AQI raster, this sub-module finds the maximum value at each location in Ozone and PM rasters and combines it into an AQI raster. This sub-module can also generate map layers for animation frames at 20-minute intervals by interpolation from hourly grids.
Data Inputs:	Raster map layers from the Map Store
Data Outputs:	Raster map layers placed in the Map Store
Operation:	This program is started by Dispatcher and is controlled with command-line parameters.
Code:	C# using Microsoft .net and ESRI ArcObjects frameworks

Construct Map Sub-Module

Function:	Uses ESRI ArcObjects "Exporter" components to generate image files in standard formats. It uses the manually generated map templates stored as ESRI map documents (.mxd files) and combines the raster layers with the map template.
Data Inputs:	An .mxd format map template and numerous base map layers are loaded from the Map Store. Additional layers that are generated by other Mapper components are also loaded and inserted into the map product. Additional data, such as titles and labels, are received from the Dispatcher.

Data Outputs:	A single image file. This file may be a .jpg, .gif, .tif, or .png formatted file.
Operation:	This program is started by Dispatcher, which derives its parameters from production schedules in the AIRNow Database.
Code:	C# using Microsoft .net and ESRI ArcObjects frameworks

Animate Sub-Module

Function:	Create animated maps from individual map files
Data Inputs:	Map images in .gif format
Data Outputs:	Animated images in .gif format
Operation:	This program is started by Dispatcher, which derives its parameters from production schedules in the AIRNow Database. In addition, the Animate sub-module uses the Convert program from the ImageMagick suite.

3.4 AIRNOW INFO SERVICE

3.4.1 Overview

The AIRNow Info Service is a component of AIRNow International that creates a data gateway from the AIRNow Database and products (maps and data) to other users and systems. **Figure 3-4** shows a schematic of the major features, data flow, and control of the AIRNow Info Service. Details about the features and functions are described in Section 3.4.2. The data formats and web services supported by AIRNow International are listed in Section 3.5.

Important design considerations for AIRNow Info Service include the following:

- Develop and distribute it with an open source code to allow Partner organizations to create new software that can output data in different formats. In addition, the open source code approach for this module will provide a level of transparency so that agencies will be able to see how their data are being distributed.
- Control of who accesses and uses data will be provided by the AIRNow Info Service. This will enable users to approve, monitor, and control access to data and products generated by AIRNow Info Service.
- Produce data files and support web services. Several data standards are emerging for distributing and sharing air quality information. AIRNow Info Service will also produce some basic data files and web services as described in Section 3.5. These data files and services will conform to the standards being developed as part of Global Earth Observing System of Systems (GEOSS).

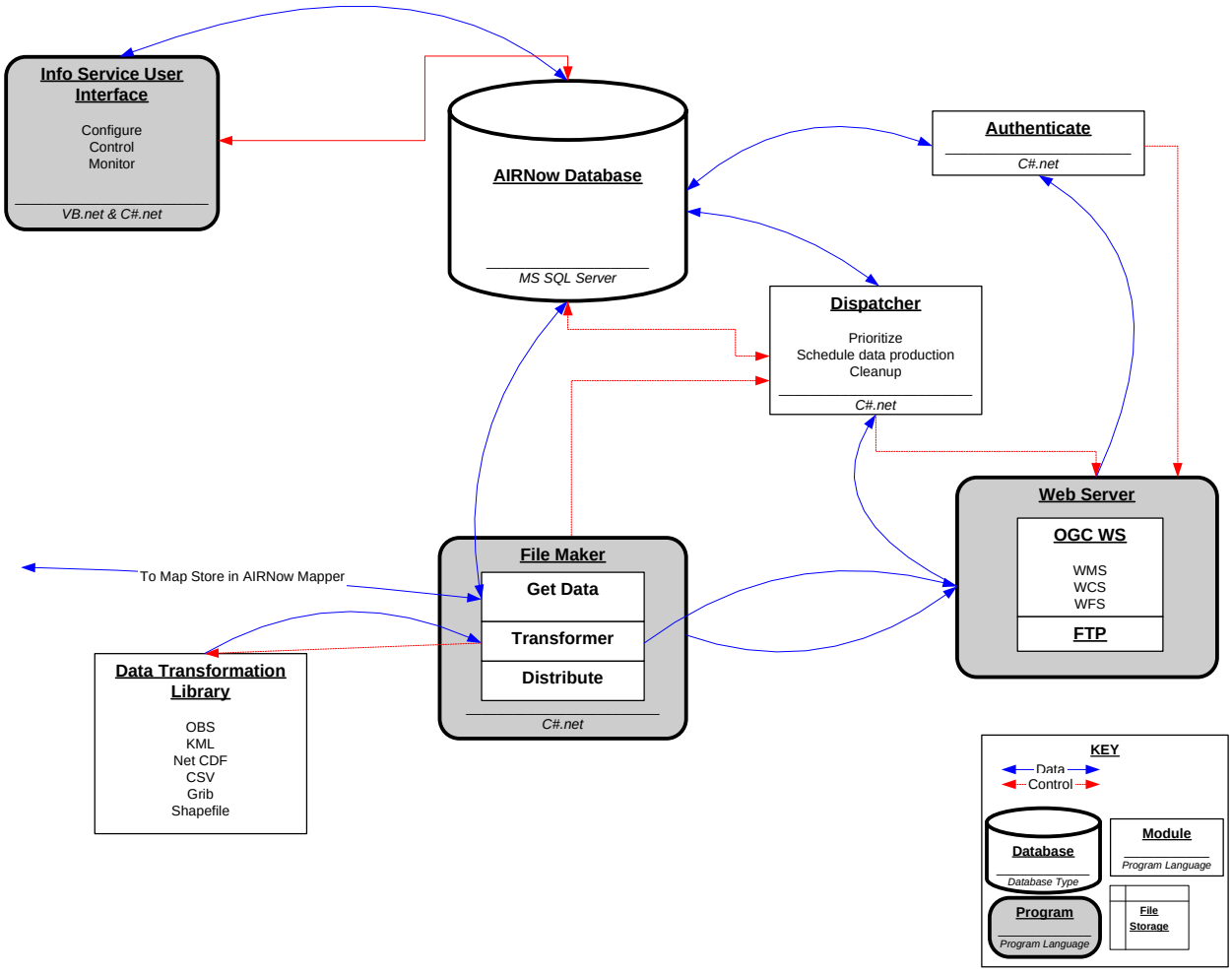

Figure 3-4. A schematic of the major features, data flow, and control of AIRNow Info Service.

3.4.2 System Design and Functions

Figure 3-4 also shows the major components of AIRNow Info Service.

Dispatcher

Function: Cue, prioritize, and sequence the events to produce and distribute data and prioritize requests from web services

Data Inputs: Receive information stored in the AIRNow Database about activities and events to schedule

Data Outputs: Event information (problems, completed tasks, etc.) will be placed in the event table of the AIRNow Database

Operation: Dispatcher will run and monitor data continuously. It will receive scheduling tasks from the AIRNow Database and start the appropriate tasks based on the system clock and schedule settings in the AIRNow Database. In addition, when web service requests are made, the Dispatcher will prioritize these requests and launch the appropriate tasks needed to meet the requests.

Code: VB.net or C#.net using Visual Studio 2008

File Maker

Function: Query data, produce files (and data payloads for web services), and distribute data

Data Inputs: Receive instructions from Dispatcher

Data Outputs: Files in specified formats. Supported formats are listed in Section 2.4.

Operation: Users will configure and control File Maker via the user interface. File Maker will receive instructions from Dispatcher on which data to query, which file formats to produce, and where to distribute the data. Users can generate and dispatch many different types of files (as discussed in Section 3.5). File Maker will use a series of small data transformation programs to produce the different types of data files. These data transformation programs will request data via a SQL query from the AIRNow Database and then transform the data into a specified format. A template will be provided so that agencies running AIRNow-I can develop their own data transformation programs to output data in their custom formats. For web services, File Maker will be used to generate the data payload that is distributed via the web server. A web service request will be sent to Dispatcher and Dispatcher will start File Maker to retrieve the data and create the data payload for the web service.

Another important function of File Maker is the ability to distribute data via File Transfer Protocol (FTP) to a web service, other system, or locations on the Internet. Users will control where to distribute the data via the user interface. Simple settings will allow for an FTP or local drive address and user name and password settings.

Code: VB.net or C#.net using Visual Studio 2008

Authenticate

Function: Will validate users who want to access data from the FTP site or the web server

Data Inputs: Receive request from the web server

Data Outputs: Provide an approval or denial of the data request

Operation: This system will leverage the AIRNow Gateway system (www.AIRNowGateway.org) currently running at the AIRNow Data Management Center. AIRNow Gateway handles authentication. Users can request an account for AIRNow-I and local administrators will be able to approve or deny requests. Once approved, users would be able to access data files and web services.

Code: VB.net or C#.net using Visual Studio 2008

Web Server

Function: The web server will provide a network portal for external users and systems.

Data Inputs: Receive requests from external clients for login. Receive approval or denial of access from Authenticate. Receive requests for data products from external clients. Retrieve requested air quality data from the AIRNow Database. Get map product files from disk.

Data Outputs: Principal outputs are air quality data and map products delivered over the web to external data clients. Other outputs are user login data passed to Authenticate and data preparation requests sent to the Dispatcher via the AIRNow Database.

Operation: Web services are a set of programs that are connected to the web via specific sockets. They monitor these sockets for messages from complimentary client programs running elsewhere on the network. When a message is received, the service either acts on it directly or translates and forwards the message to other programs within AIRNow Info Service. The University of Minnesota "MapServer", Geoserver, or a similar product will be used. These are open source programs that support WCS, WMS, and WFS services.

Code: Much of this code will be adapted from other open source projects. The FTP server will be a customized version of the "Apache FTP Server". AIRNow-specific customizations will be written in java. AIRNow customization of MapServer will be written in C#.

Data Transformation Library

Function: The data transformation library will contain a series of small programs that will convert data into a specified format.

Data Inputs: Data from the AIRNow Database accessed via a Data Access Layer (DAL)

Data Outputs: Data files specified in Section 2.4.

Operation: The library will contain the individual data transformation programs that convert data to specific file formats. Several data transformation programs will be written and provided as part of AIRNow Info Service to produce the data file formats specified in Section 3.5. Since these

programs will be open source, programmers at Partner organizations will be able to modify existing programs to create new formats.

Once created, a data transformation program will need to be registered within the AIRNow Database via the user interface. Once registered, Dispatcher will launch the data transformation program. Each program will then use the DAL to extract data, metadata, or other information from the AIRNow Database. The program will then transform the data into the specified format and write the data file.

Code: Will vary based on the type of file being created

3.5 AIRNOW DATABASE SPECIFICATIONS

3.5.1 Database and Approach

The AIRNow Database is organized into four major components:

1. Data and metadata tables with data and supporting metadata including sites, parameters, instruments, and QC codes
2. Logging and monitoring tables that store data change information, system events, and user comments
3. Mapping service tables that contain user configuration information and scheduled tasks for AIRNow Mapper and its sub-modules
4. Security and user management tables with system administrator's settings, user groups, and specific data access, editing, and reporting rights

The AIRNow Database makes extensive use of SQL for processing and accessing data. All data updates and queries are handled using hundreds of SQL views, T-SQL stored procedures, and T-SQL and .NET functions maintained within the database. By taking this architectural approach to database access we can assure a high level of data integrity for the multiple Web and stand-alone applications that will use the database. We also provide a Data Access Layer (DAL) written in C#.NET that any .NET application can use to simplify DMS data handling. The DAL simplifies connecting to and calling the numerous DMS stored procedures and functions available.

The AIRNow Database supports defining instruments for measuring any number of data parameters. QC tests can be created and associated with a particular parameter measured by a particular instrument at a particular site. QC tests are automatically run against ingested data. After automatic QC checks are performed, the user can manually review and QC data. Changes to data QC codes and/or values during review are logged and linked to a user. The DMS also supports historical ingest of data into the database via AQS format files. Data ingested in this way are not associated with an instrument but only with a site. **Figure 3-5** is a diagram of the DMS conceptual data model. **Figure 3-6** is the logical data model of key tables in the DMS system.

Table 3-2 defines the fields in each of the major database tables. We anticipate that additional fields will be needed to make this database more universal for the international community.

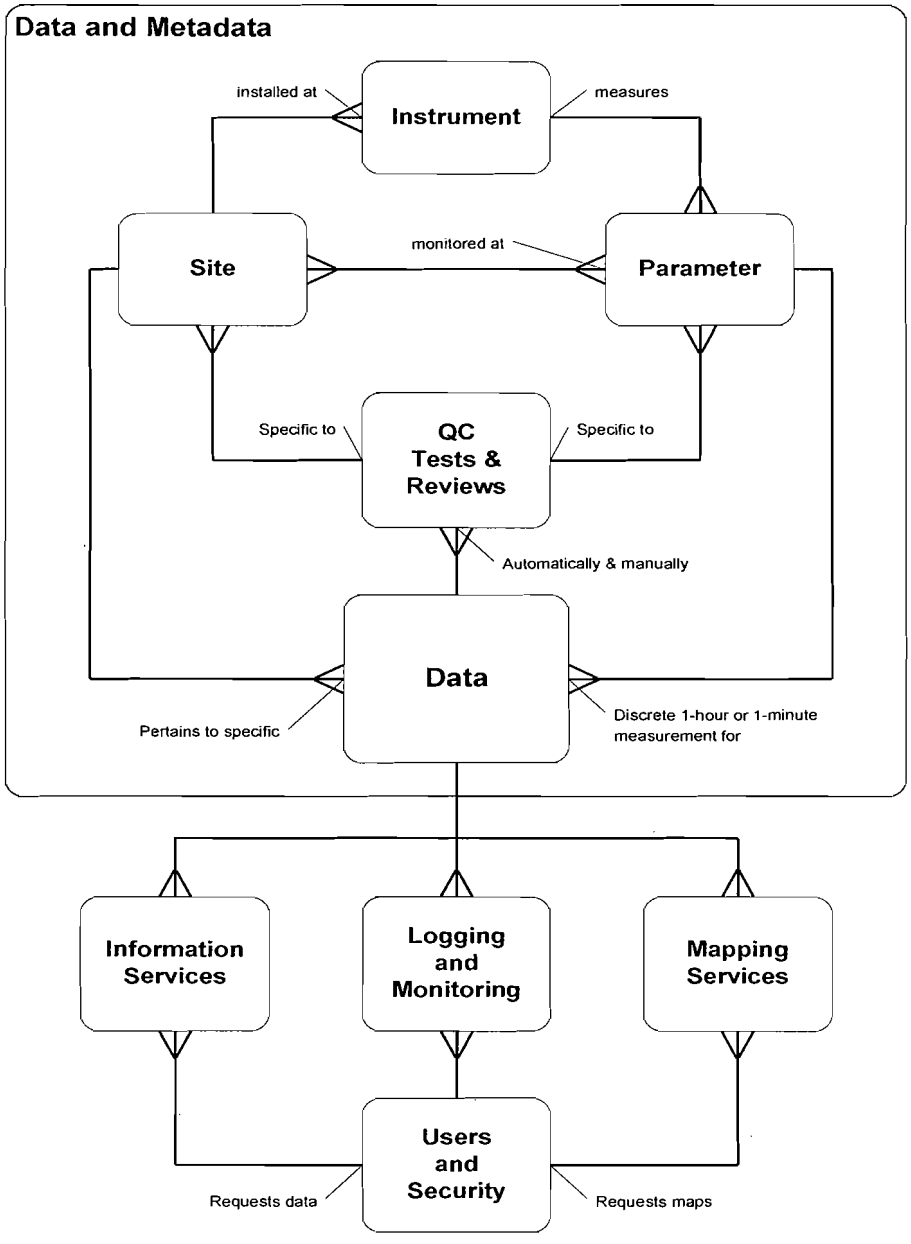

Figure 3-5. AIRNow DMS conceptual data model.

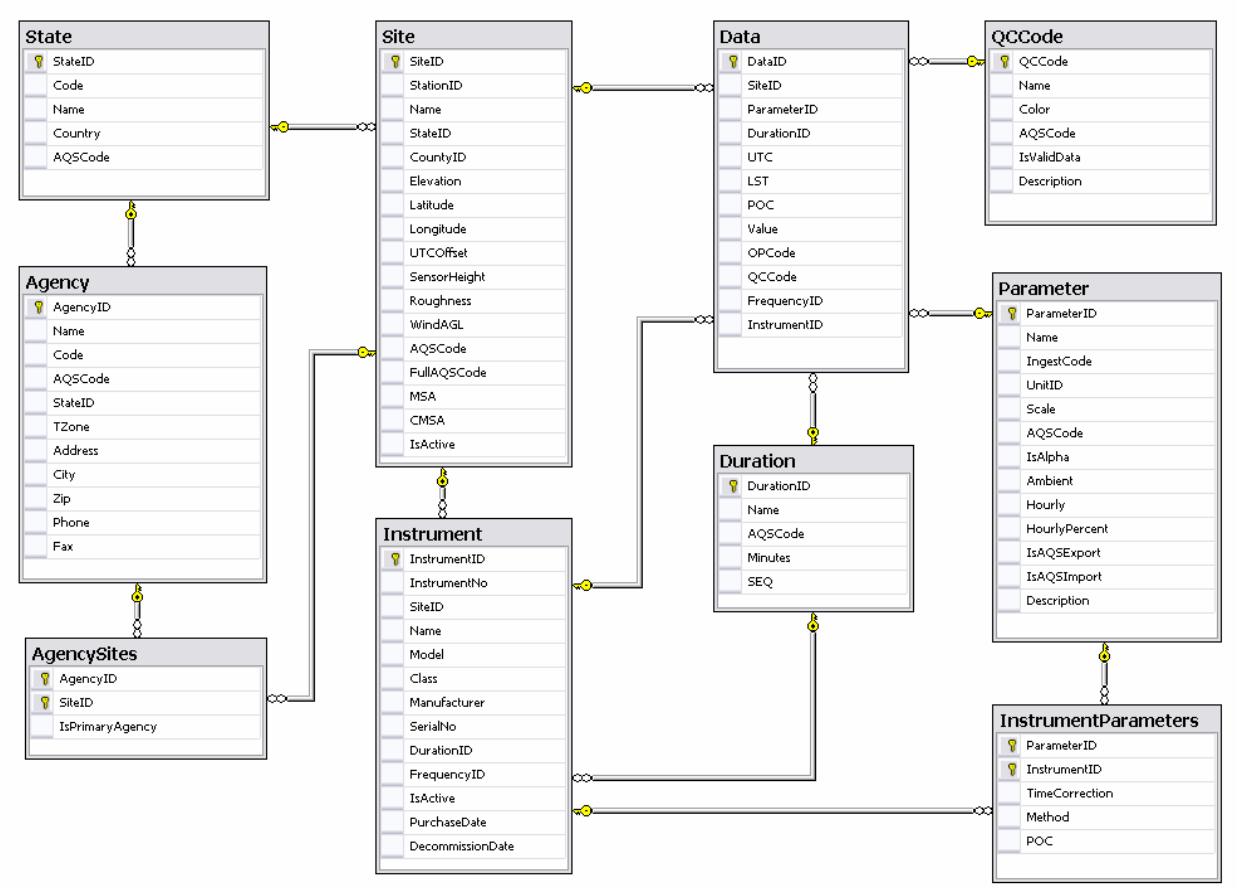

Figure 3-6. Draft AIRNow DMS logical data model.

Table 3-2. Major AIRNow Database tables and their associated fields. We anticipate that additional fields will be needed to make this database universal for the international community.
Abbreviations: PK=Primary Key, UK=Unique Key (numbered if compound), FK= Foreign Key.

Agency Table: Agency associated with monitoring sites

Field Name	Type	Size	Description	Note
AgencyID	smallint	2	Surrogate primary key	PK
Name	varchar	100	Full unique agency name	UK
Code	varchar	3	Alternative unique identifier	UK
AQSCode	varchar	5	AQS code or “Organization” in XML specification	UK
StateID	smallint	2	Surrogate foreign key	FK
TZone	char	3	Time zone	
Address	varchar	100	Mailing street address of contact	
City	varchar	50	City for mailing address	
Zip	varchar	10	ZIP Code for mailing address	
Phone	varchar	50	Office phone number for primary contact	
Fax	varchar	50	Office fax number for primary contact	

Data Table: Primary data table storing data

Field Name	Type	Size	Description	Constraints
DataID	int	4	Surrogate primary key	PK
SiteID	smallint	2	Surrogate foreign key	FK, UK1
ParameterID	smallint	2	Surrogate foreign key	FK, UK2
DurationID	smallint	2	Surrogate foreign key	FK, UK3
UTC	datetime	4	Coordinated universal time	UK4
LST	datetime	4	Local standard time	UK4
POC	tinyint	1	Parameter Occurrence Code	UK5
Value	numeric	(10,5)	Measurement value	
OPCode	tinyint	2	Natural foreign key – OP codes	FK
QCCode	tinyint	2	Natural foreign key – QC Codes	FK
FrequencyID	smallint	2	Surrogate foreign key – Frequency table with AQS frequency codes	FK
InstrumentID	smallint	2	Surrogate foreign key	FK

Duration Table: Data averaging periods

Field Name	Type	Size	Description	Note
DurationID	tinyint	1	Primary surrogate key	PK
Name	varchar	50	Duration name like “1 minute”, “1 hour”, and “24 Hour”	UK
AQS Code	char	3	AQS 3 character code if available	
Minutes	int	3	Duration or averaging period in minutes	

Instrument Table: Site instrument information

Field Name	Type	Size	Description	Note
InstrumentID	smallint	2	Primary surrogate key	PK
Name	varchar	50	Instrument name	
InstrumentNo	varchar	50	Unique instrument alphanumeric identifier	UK
SiteID	smallint	2	Foreign surrogate key	
Model	varchar	25	Instrument model name	
Class	varchar	25	Instrument classification	
Manufacture	varchar	25	Manufacturer	
SerialNo	varchar	25	Serial number	
FrequencyID	tinyint	1	Foreign surrogate key	
DurationID	tinyint	1	Foreign surrogate key	
Purchase date	datetime	4		
Decommission date	datetime	4		
IsActive	bit	1	Is the instrument active.	

Instrument Parameters Table: Parameters monitored by an instrument

Field Name	Type	Size	Description	Note
InstrumentID	smallint	2	Primary foreign surrogate key	PK,FK
ParameterID	smallint	2	Primary foreign surrogate key	PK,FK
Time Correction	smallint	2	Minutes to be added or subtracted from the ingested data’s time	
Method	char	3	AQS 3 character method code – needed for export	
POC	char	2	Parameter Occurrence Code	

Parameter Table: Air quality and meteorological metadata

Field Name	Type	Size	Description	Note
ParameterID	smallint	2	Primary surrogate key	PK
Name	varchar	50	Full unique parameter name used in UI.	UK
IngestCode	varchar	50	Parameter code used in ingest file for identification	UK
UnitID	smallint	2	Parameter units foreign surrogate key	
Scale	tinyint	1	Number of digits to the right of the decimal point for purposes of display and output	
AQSCode	char	5	The AQS code or "SubstanceIdentifier" in XML specification	UK
IsAlpha	bit	1	Boolean; is data "Value" field alphanumeric?	
Ambient	bit	1	Boolean; is data an ambient parameter value?	
Hourly	bit	1	Boolean; will 1-min data be aggregated to 1-hr?	
Hourly Percent	int	4	1-hr aggregation data completion percentage	
IsAQSExport	bit	1	Boolean; can data be exported to AQS?	
IsAQSImport	bit	1	Boolean; parameter to be associated with AQS imports	

QCLogData Table: Chain-of-custody table for historically recording changes to data QC codes, OP codes, or value. Linked to this table but not shown on the diagram is the **QCLog** table that also records user and user comments concerning changes.

Field Name	Type	Size	Description	Note
QCLogID	smallint	2	Primary foreign surrogate key	PK
DataID	bigint	2	Foreign surrogate key	FK
OldValue OldQCCode	numeric tinyint	10,5 1	Value prior to Data table value change QC code prior to Data table QC code change	
OldOPCode	tinyint	1	OP code prior to Data table OP code change	

Site Table: Monitoring site metadata

Field Name	Type	Size	Description	Note
Name	varchar	50	Full unique site name	UK
StationID	varchar	10	Alternative unique site identifier	UK
Name	varchar	50	Full Name of site	
StateID	smallint	2	Surrogate foreign key	FK
CountyID	smallint	2	Surrogate foreign key	FK
AQS Code	char	4	AQS code or "FacilitySiteIdentifier" in XML specification	
Elevation	int	4	Elevation of monitoring site in meters	
Latitude	numeric	(8,6)	Latitude of site in decimal degrees	
Longitude	numeric	(9,6)	Longitude of site in decimal degrees	
UTC offset	tinyint	1	Time zone offset of monitoring site from GMT	
AQSCode	char	4		
MSA	char	4		
FullAQSCode	char	9	Full 9 char ASC code for easier searches.	
IsActive	bit	1	Is site an active site	

State Table: State or province metadata

Field Name	Type	Size	Description	Note
StateID	smallint	2	Primary surrogate key	PK
Code	varchar	10	State or province abbreviation	UK
Name	varchar	50	State or province name	UK
Country	varchar	50	County name	
AQS Code	char	2	AQS 2 character code if available	

QCCode Table: Data QC codes

Field Name	Type	Size	Description	Note
QCCode	tinyint	1	Primary natural key	PK
Name	varchar	50	Name of QC code to appear in UI	UK
Description	varchar	512	Full description of code	UK
Color	varchar	20	Controls display color for code in UI	
IsValidData	bit	1	Code indicates valid or invalid data	
AQS Code	char	1	AQS 1 character code if available	

4. SUPPORT AND MAINTENANCE

Initially, the support and maintenance for AIRNow-I will come from the U.S. EPA and its AIRNow program. As described in the next section, additional support and maintenance for the system is the responsibility of the Partner organization.

5. MANUALS AND INSTRUCTIONAL WEB SITE

AIRNow-I manuals and a web site are needed for the AIRNow International Program to help Partner agencies understand how to use the system, to help create a community of practice that will encourage collaboration, and to facilitate a sustainable system. Since the AIRNow-I system will evolve rapidly, we propose all manuals and documentation be kept on-line in a digital format at the AIRNow International web site. We will use new technologies such as Wikipedia (Wiki) and Really Simple Syndication (RSS) to promote collaboration and sharing of knowledge and information.

The web site (www.AIRNowInternational.org) will have an open content management system (e.g., Wiki) and contain the following information:

- Home page – overview of AIRNow International and contact information, map of Partner organizations.
- Software – overview of the software and hardware/software requirements for running the AIRNow-I system.
- Community – a discussion and collaboration area to exchange information, questions, problems, issues, etc. With organizations working in different time zones, schedules, and languages, this community discussion process help us all communicate more efficiently.
- Resources – list of web links, instructional videos, and manuals. Manuals will be written in English. Our development partner at the SEPB will translate them to Chinese. Manuals in both English and Chinese (and eventually other languages) will be available on the web site.

We propose to quickly create the web site so it will be a communication portal between the U.S. EPA and SEPB during the development of AIRNow-I.

6. TESTING PLAN AND APPROACH

AIRNow-I has many software modules and programs that need to be tested. As part of the development of AIRNow-I, the team will use the following testing approaches:

- System and module testing – test each module, routine, subroutine, etc., when it is being constructed.
- Performance testing – evaluate the overall end-to-end performance of AIRNow-I.
- Security testing – evaluate the security features of the AIRNow Database, the Authenticate module in AIRNow Info Services, and other security features.
- Automated testing – test the ability of AIRNow-I to automatically process data, map data, and distribute data products.
- Stress and volume testing – challenge the AIRNow-I system by increasing the processing loads and increasing the output (maps and data files) to help determine the maximum capabilities of the system.
- Recovery testing – determine the ability of the system to recover and restart from errors, problems, or power failures.
- Documentation testing – examine the documentation and ensure that all system features are properly described in the documentation. In addition, ensure that all steps or procedures in the documentation work properly with the system.
- Beta testing – test the AIRNow-I system in an end-to-end configuration at another location. This testing will initially be conducted with our development partner at the SEPB.
- User acceptance testing – test the final AIRNow-I system in an operational environment. Final acceptance testing will be conducted with our development partners at the SEPB.

In addition, we will use several tools to help with the testing, debugging, and maintenance of the software. These include the following:

- A test plan containing the procedures for testing the AIRNow-I software.
- Regular testing and bug review meetings built into the detailed project schedule.
- FogBugz (web-based bug tracking software) for documenting, tracking, and reviewing the testing and fixing of bugs. A link on the project web site will allow Partners to review the status of testing and bugs.

The schedule for developing, testing, and launching the AIRNow-I system is shown in **Table 6-1**.

Table 6-1. Schedule for developing, testing, and launching the AIRNow-I system.

Task	Organization	Due Date
Design and Specification Plan		
Meet to discuss initial approach and understand needs (completed)	EPA, SEMC	May 2007
Determine target operating system, GIS, and database systems (completed)	EPA, SEMC	Nov 2007
Create initial detailed features, functions, and requirements list (completed)	STI	Nov 2007
Determine special features for Shanghai (completed)	SEMC	Dec 2007
Determine local requirements for Shanghai (completed)	SEMC	Dec 2007
Determine special features for other countries (completed)	EPA	Dec 2007
Create design specification document. It will cover DMS, Mapper, Info Service, and Administration. It will also include reports, system requirements, user interface diagrams, data input standards, data output standards, etc. (initial draft document completed, i.e., this document)	EPA	Apr 2008
Review initial design specification document	SEPB	Apr 2008
Review design specification document	EPA	Apr 2008
Conduct design review meeting	SEPB, EPA	May 2008
Create final specifications document	EPA	May 2008
AIRNow Data Management System		
Develop DMS	EPA	June 2008
Test additions and modifications to DMS	EPA, SEPB	Jul 2008
Document the data processing system	EPA	Jul 2008
Integrate current SEMC system with AIRNow-I	SEPB	Nov 2008
AIRNow Mapper		
Create modules	EPA	July 2008
Create mapping control interface	EPA	July 2008
Create mapping monitoring system	EPA	July 2008
Review and test	EPA, SEPB	Nov 2008
Address issues/problems	EPA	Dec 2008
Integrate data mapping system into AIRNow-I	EPA	Dec 2008
AIRNow Info Service		
Create Info Service	EPA	Aug 2008
Improve methods for distributing data	EPA	Aug 2008
Create methods to ingest data into AIRNow-I	SEPB	Aug 2008
Create methods to use data from AIRNow-I	SEPB	Aug 2008
Review and test service	EPA, SEPB	Nov 2008
Address issues/problems	EPA	Dec 2008
Integrate data distribution system into AIRNow-I	EPA	Dec 2008

Table 6-1. Schedule for developing, testing, and launching the AIRNow-I system.

Task	Organization	Due Date
Administration System		
Program user interface	EPA	Dec 2008
Create performance monitoring system	EPA	Dec 2008
Review and test data distribution component	EPA, SEPB	Dec 2008
Address issues/problems	EPA	Jan 2009
Integrate administration system into AIRNow-I	EPA	Jan 2009
Adding Multi-Language Capability		
Determine how to create a multi-language program	EPA, SEPB	Apr 2008
Set up procedures/protocols for multi-language programming	EPA	June 2008
Integrate multi-language capability into program	EPA, SEPB	July 2008
Program Chinese version (interface, etc.)	SEPB	Nov 2008
Testing		
Test entire system	EPA, SEPB	Feb 2009
Review and address issues/problems	EPA, SEPB	Mar 2009
Second-level test of entire system	EPA, SEPB	Mar 2009
Conduct final acceptance test	EPA, SEPB	Apr 2009
Manuals/Tutorials/Instructional Web Site		
Write manuals/tutorials	EPA	Feb 2009
Write Chinese version	SEPB	Feb 2009
Design instructional web site	EPA	Mar 2009
Set up instructional web site	EPA	Apr 2009
Test and deploy instructional web site	EPA	May 2009
Launch		
Install in Shanghai	EPA, SEPB	Apr 2009
Configure	SEPB, EPA	Apr 2009
Test and modify	SEPB, EPA	Apr 2009
Launch	EPA, SEPB	Apr 2009

7. PARTNERSHIP AGREEMENT (DRAFT)

A major goal of the AIRNow International Program is to develop a community of organizations that collect, process, exchange, and communicate air quality observations and forecasts. As part of this community, a Partner organization has certain responsibilities, which are outlined in this Partnership Agreement. This agreement covers the general terms and conditions for participating in the AIRNow International community, sharing data with the community, and receiving and using the AIRNow-I software.

Community: Participation in the broader user community is a requirement of AIRNow International. Collaboration among the AIRNow International Partners is important to help improve the Partners' understanding of air quality and keep the software systems updated to meet current and future needs. In addition, collaboration helps all organizations and countries learn from each other about methods, challenges, and approaches in collecting, processing, analyzing, and communicating air quality observations and forecasts.

Partners must participate in the user community by

- Contributing to a Wiki on the AIRNow International web site (www.AIRNowInternational.org) to collaboratively create, edit, link, and organize the content on the website
- Responding to periodic email inquiries from other Partners
- Participating in a conference related to AIRNow International (e.g., National Air Quality Conference)
- Providing periodic (two times per year) updates on activities and plans that relate to AIRNow International
- Providing and updating contact information so that Partners can communicate with each other

Air Quality Observations (Data): Partners are required to share air quality observations, air quality forecasts, and other related air quality information. The following guidelines should be followed by the Partner organizations that both provide and use these data.

Software: The AIRNow-I software (consisting of DMS, Mapper, and Info Service, and the AIRNow Database) is available to non-profit and government organizations at no cost provided that the organizations abide by the terms and conditions outlined in this Partnership Agreement. To use the software, an agency must agree to the following:

- Software may only be used by the Partner organizations that agree to all the terms and conditions in this Partnership Agreement.
- A Partner organization can modify the open source modules of AIRNow-I (AIRNow Info Service and the AIRNow Database). Any modifications to the software must be distributed via the AIRNow International web site (www.AirnowInternational.org).

- A Partner organization may not distribute AIRNow-I software to anyone. All distributions should come from the AIRNow hub to avoid propagation of outdated code. However, a Partner organization can encourage other organizations to become part of the community.
- A Partner organization cannot sell the AIRNow-I software or any of its modules or the software derived or modified from the AIRNow-I software or any of its modules.

Hardware and Commercial Software: As outlined in Sections 2.1 and 2.2 of the AIRNow International specification document, hardware and commercial software are needed to operate the AIRNow-I software system. Each organization is responsible for the cost of:

- Acquiring and installing the recommended hardware and commercial software for running AIRNow-I
- Maintaining current licenses for commercial software
- Maintaining and upgrading the hardware and commercial software

Other Items

- This Agreement does not include any professional or consulting services in connection with AIRNow International for training, assistance, and installation or product support.
- EPA does not warrant AIRNow-I against defects or claims of infringement caused by (1) modifications made to AIRNow-I or any portion of it; (2) Partner organizations failure to use any new or corrected versions of AIRNow-I; or (3) any derivative work that may contain all or part of AIRNow-I software.
- EPA does not warrant that the Software will be error-free or that its operation will be uninterrupted. The Partner organization acknowledges that it is solely responsible for the results obtained from using the Software.