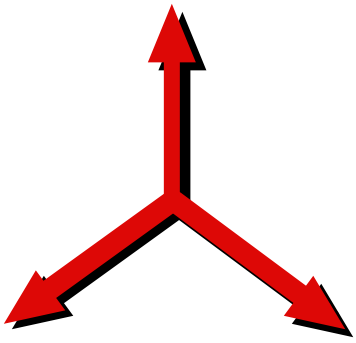
# EXPANDED VERY LARGE ARRAY



## Operational Interface

## Software Requirements Specification

Project Document: **A24101N0001**
Revision: 2.0

**April 4, 2002**

Rich Moeser

NATIONAL RADIO ASTRONOMY OBSERVATORY
P.O. Box 0, Socorro, New Mexico 87801

Operated by Associated Universities, Inc.
Under Contract with the National Science Foundation

# Revision History

| Date | Version | Description | Author |
|---|---|---|---|
| 09-Nov-2001 | 1.0 | Initial draft | Rich Moeser |
| 04-Apr-2001 | 2.0 | Incorporated feedback from Barry Clark, Peggy Perley, Rick Perley, Kevin Ryan and Bill Sahr. Added sections to conform to IEEE standard. Modified title page and headers and footers to conform to EVLA document standard for SRS documents. | Rich Moeser |

# Table of Contents

# 1 Introduction

## 1.1 Purpose

The primary goal of this document is to provide a complete and accurate list of requirements for the EVLA Operational Interface.[1] Upon completion, the document will act as a binding contract between developers and users and will provide a common point of reference for system expectations.

The primary audience of this document includes, but is not limited to, project leaders, the designers and developers of the system and the end user. The document may also be of interest to EVLA project scientists and engineers or as a reference for individuals involved in similar projects with similar requirements.

Each requirement within the document is defined by a description and a priority. The description specifies a function or characteristic the software must exhibit and must have a single interpretation. The priorities specify the importance of the requirement and are defined as follows:

- **0** (*essential*): This is the highest priority. Requirements with this priority must be designed and built into the software.

- **1** (*desirable*): Requirements with this priority may or may not be included in the software. Decisions of inclusion will be based on time and budget constraints.

- **2** (*if possible*): This is the lowest priority. As with priority 1 requirements, there are no guarantees that priority 2 requirements will be included in the software and the decisions of inclusion will be based on time and budget constraints.

[1] The term "interface" refers to the interface between the user and the computer, in this case a graphical user interface, and should not be confused with an application programming interface (API) used by programmers.

## 1.2 Scope

The Operational Interface is one of many components that constitute the EVLA Monitor and Control System.[2] The primary responsibility of the Operational Interface is to provide a suite of software tools that allow the array operators, engineers, technicians, scientists, software developers, and other authorized users to interact with the array in a safe and reliable manner.

The Operational Interface will provide the following functions:

- The ability to monitor the status and overall health of the array.
- The ability to control certain aspects of an antenna, subarray or the entire array.
- The ability to view and manage the operator/system log.
- The ability to view and manage users' system access and privileges.
- The ability to view and manage observing schedules.[3]

This document will define only those requirements that must be fulfilled by the Operational Interface.

## 1.3 Definitions, Acronyms and Abbreviations

### 1.3.1 Definitions

**Array** – A collection of antennas. At times it will be used to indicate a subarray, but will most often refer to the VLA, NMA or VLBA.
**Observer** – An individual granted time on the array to conduct a scientific investigation or test.
**Operator** – An individual authorized to issue commands to the array. In most instances, the term operator will refer to the array operator, an NRAO employee whose duty it is oversee the success and safety of an observation.

[2]An overview of the EVLA Monitor and Control System can be obtained by reading the *System Requirements Specification* and the *EVLA Software Architecture and Design* documents.

[3] This is largely an e2e issue. An application programming interface to the scheduling routines will be made available by e2e project developers.

**Operator station** – The primary work area of an array operator, consisting of computers, monitors and communications equipment.
**Scan** – An atomic element of an observation. It defines the source being observed, the duration and the hardware setup.
**Administrator** – An individual with unrestricted access to all aspects of the system.
**Display (or Screen)** – A GUI component contained within a frame. A display may be static or non-static and may or may not accept input from the user.
**Frame (or Window)** – A top level GUI component that contains other GUI components, such as menu bars, tool bars, etc.

### 1.3.2 Acronyms

**AOC** – Array Operations Center
**API** – Application programming interface
**e2e** – End-to-end project
**GUI** – Graphical user interface
**M&C** – monitor and control
**NMA** – New Mexico Array
**VLA** – Very Large Array
**EVLA** – The VLA Expansion Project
**VLBA** – Very Long Baseline Array
**LST** – Local Sidereal Time
**UTC** –Universal Time Coordinated
**RA** – Right ascension
**RPC** – Remote procedure call
**WVR** – Water Vapor Radiometer
**SyRS** – Refers to the *System Requirements Specification* document.
**SRS** – Refers to the *Software Requirements Specification* document.

### 1.3.3 Abbreviations

**Az** – azimuth
**Dec** - declination
**El** – elevation

## 1.4 References

1) ANSI/IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications
2) ANSI/IEEE Std 1233-1996, IEEE Guide for Developing System Requirements Specifications
3) EVLA Memo No. 15, Scientific Requirements for the EVLA Real-Time System, http://www.aoc.nrao.edu/evla/memolist.shtml
4) EVLA Project Book, http://www.aoc.nrao.edu/evla/pbook.shtml
5) EVLA System Requirements Specification (SyRS), http://www.aoc.nrao.edu/evla/pbook.shtml
6) EVLA Software Architecture and Design, http://www.aoc.nrao.edu/evla/techdocs/computer/workdocs/index.shtml
7) The Very Large Array Observing Log (J. Nieri, February 1994)
8) NRAO Computing Security Policy, http://www.aoc.nrao.edu/evla/techdocs/computer/workdocs/index.shtml

## 1.5 Overview

The remainder of this document contains a more detailed description of the Operational Interface as well as the requirements necessary to design and build it.

Section 2 provides an overall description of the Operational Interface. It defines the functions the Operational Interface will provide, its role in the larger EVLA Monitor and Control System and a description of its users.

Section 3 details the requirements of the Operational Interface and is the core of this document. It contains both the functional requirements, those that define the functions the system must perform, and non-functional requirements, those that describe the manner in which the functions will be performed.

The format of the document closely follows that outlined in the IEEE Std 830-1998 document, IEEE Recommended Practice for Software Requirements Specifications.

# 2 Overall Description

## *2.1 Product Perspective*

The EVLA Monitor and Control System will be designed and implemented as a distributed system. Computers in the system may be of different types and operating systems and will communicate with one another over a network. Currently, only a conceptual diagram exists for the system (See Figure 1) and should be viewed as such. Referring to the diagram, it's likely the Operational Interface software will divide its responsibilities between the Operations Server and the client stations located throughout the diagram.

### 2.1.1 Client Station

A client station is defined as a user's computer that allows interaction with the Operational Interface. Its primary responsibility is to run the Operational Interface and will do this either as a standalone program that executes on the client station or through a browser. The type and operating system of each client station will likely vary, as will the location. Client stations can be located at the VLA Control Building, the AOC, or any Web-accessible location. (See Appendix A, Prototypical operator's workstation.)

### 2.1.2 Operations Server

The Operational Interface will communicate directly with the operations server. The server will have the responsibility of transmitting data acquired from the core M&C real-time system to the outlying client stations. It will also receive commands issued from authorized client stations and forward those commands to the core real-time system.

### 2.1.3 Client/Server Communication

The communications protocol between the client and the server are unknown at this time. It will likely be an RPC-like communications middleware specification, for example, CORBA, RMI, XML-RPC, SOAP, etc.

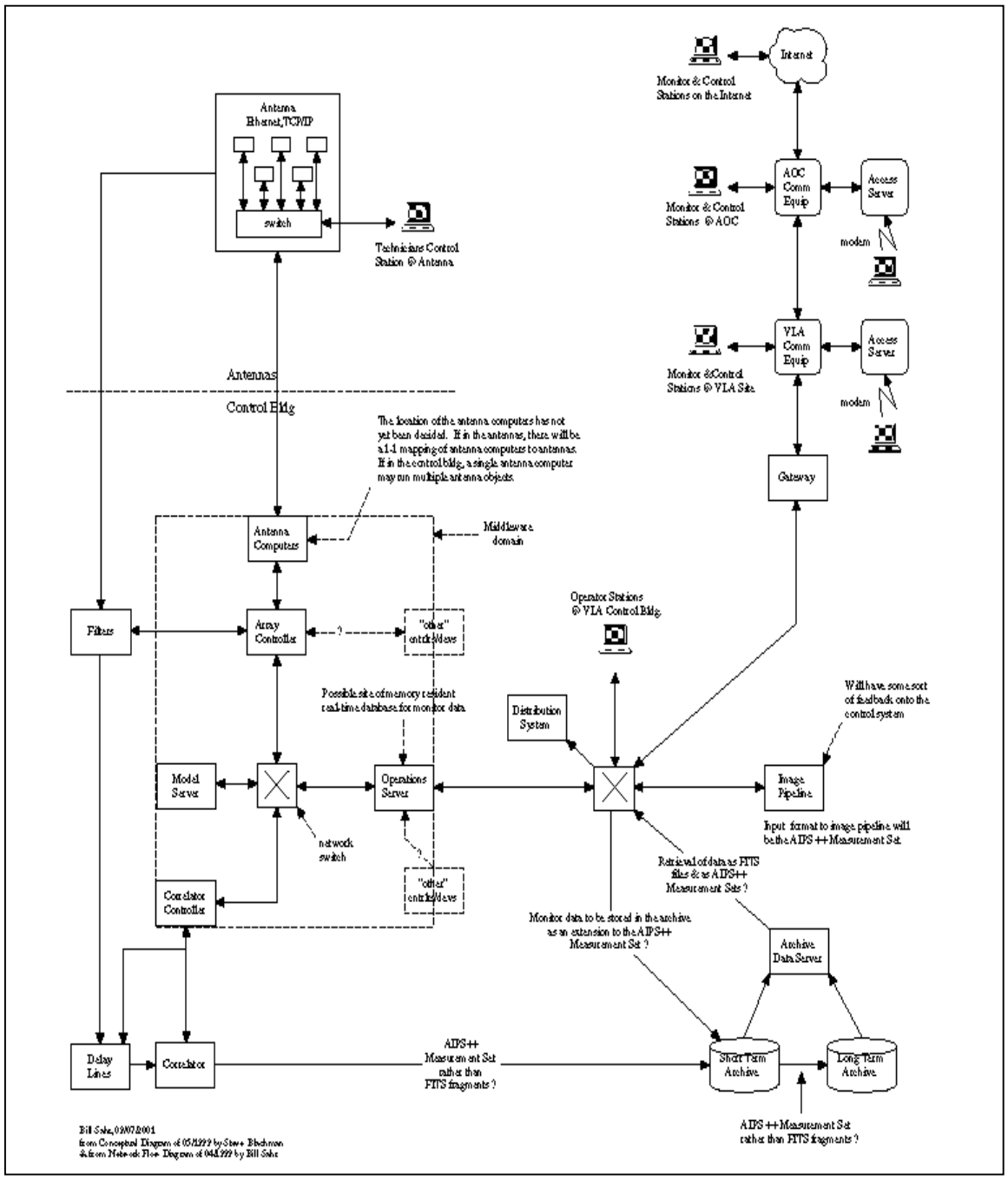
### 2.1.4 Display Framework

A display framework serves two purposes: 1) it forces structure on a display, and 2) it provides a mechanism for integrating the display into the Operational Interface.

Each display will be required to follow a predefined set of rules, thus creating a high-level of common behavior inherent in each display. Once this common behavior has been achieved, the framework can manage each display through this high-level interface.

After a display has been created, the integration of the display into the framework should be a simple process. Two approaches should be offered. The first would allow the user to manually import a display into the Operational Interface. This approach would assist the development and testing of displays and would be used most often by the display developers. The second approach would allow the display to be added to a package of displays that would be recognized at runtime by the Operational Interface. This would be the standard method of integrating displays into the Operational Interface during deployment.

## Figure 1: EVLA M&C System Conceptual Diagram



## 2.2 Product Functions

### 2.2.1 Array Monitoring

The Operational Interface will supply the array operators and other users with high-level and low-level monitoring abilities. High-level displays will provide information on the overall health of the array whereas the low-level displays will give detailed information on specific components within the system. The displays will be composed of textual and graphical components and will use color and audible alerts to inform the user of unexpected events and conditions. The number and exact types of displays are unknown at this time, although, it is expected that displays will be created to provide information on – to name only a few – current weather conditions and the status and health of individual antennas, antenna subsystems, and subarrays.

### 2.2.2 Array Control

Several displays within the Operational Interface will allow authorized users to control all or parts of the array. Control functionality will be built into the displays using GUI components (sliders, buttons, combo boxes, etc.) that accept keyboard or mouse input from the user. The number and exact types of control displays to be created is unknown at this time.

### 2.2.3 Logging

The Operational Interface will provide a tool that enables authorized users to create and send messages to a message log. This will replace the functionality currently provided by the observing log that is generated by the array operators using Microsoft Excel.

### 2.2.4 Operator/Observer Chat

The Operational Interface will provide a simple messaging (or chat) tool that allows operators and observers to communicate during an observation. This will allow corrections or changes, suggested by the observer, to be made during the observation in an attempt to increase the quality of the data being collected.

### 2.2.5 Scheduling

The majority of the scheduling functionality will be provided by the e2e project. However, the Operational Interface requires access to this scheduling information and will need an API to the e2e project scheduling routines.

## *2.3 User Characteristics*

### 2.3.1 NRAO Staff

#### 2.3.1.1 Array Operators

The array operators are responsible for the overall success and safety of all observations and will be the primary users of the Operational Interface. They require both monitor and control capabilities of the array and perform their duties from either the VLA Control Building or the AOC. The majority of requirements contained in this document directly reflect his/her needs and duties.

#### 2.3.1.2 Engineers

Engineers are responsible for the design, development and testing of the mechanical and electrical components within the system. They require the ability to inspect/control individual system components both remotely and at the antenna during working and non-working hours.

#### 2.3.1.3 Technicians

Technicians are responsible for the day-to-day monitoring and maintenance of the mechanical and electrical components within the system and are usually the first to be notified in the event of a non-working or malfunctioning component. As with the engineers, technicians require the ability to inspect/control individual system components both remotely and at the antenna during working and non-working hours.

#### 2.3.1.4 Scientists

Scientists are granted time on the array to conduct scientific investigations or tests. Their primary interest lies in the scientific data obtained by the instrument. They require remote access to both monitor data and visibility data to assess the progress to help make decisions during an observation. Scientists may or may not require control capabilities.

#### 2.3.1.5 Programmers

Programmers are responsible for creating the software that drives the system. They must have access (with control capabilities) to the system, both locally and remotely, for testing and troubleshooting during working and non-working hours.

### 2.3.2 Non-NRAO Staff

#### 2.3.2.1 Scientists

See Section 2.3.1.4 (NRAO) Scientists. The non-NRAO scientists will not have any control capabilities.

#### 2.3.2.2 Web User

Web users are those individuals that are part of the general public. They will have the ability to monitor the system and will not have any control capabilities.

## *2.4 Constraints*

### 2.4.1 Criticality of the Application

The Operational Interface is one of many critical components of the EVLA M&C System. Without it, the operators will not have the ability to monitor or control the array, thus jeopardizing the success of the observation.

### 2.4.2 Portability

The Operational Interface must be highly portable, as it will run on several different client machines each running disparate operating systems.

### 2.4.3 Safety and Security Considerations

The Operational Interface must be safe and secure. It should never put equipment or personnel at risk and never allow unauthorized users access to control functions of the array.

### 2.4.4 Interfaces to Other Applications

The Operational Interface must work in conjunction with the software developed by the e2e project. In particular, an interface to the e2e scheduling routines must be provided.

### 2.4.5 Higher Order Language Requirements

The Operational Interface must use Glish as the standard scripting language.

## *2.5 Assumptions*

None.

## *2.6 Future Considerations (Apportioning of Requirements)*

This subsection describes non-critical features that may be added to the Operational Interface at a later date. The Operational Interface should be designed and built with the “hooks” in place for the future addition of these features.

### 2.6.1 Simulator

A simulator would allow for the simulation of real-time situations. This would be highly beneficial for the training of array operators and for testing software.

### 2.6.2 Display Builder

A display builder would allow developers and end-users to design and create their own displays. The displays can be added to the standard displays package or imported into the Operational Interface.

# 3 Specific Requirements

## *3.1 External Interface Requirements*

### 3.1.1 User Interfaces

The Operational Interface may have to cooperate or work in tandem with GUIs developed by the e2e group.

### 3.1.2 Hardware Interfaces

The Operational Interface will not directly interface with hardware, as it will be separated from the hardware by many layers of software.

### 3.1.3 Software Interfaces

- **Monitor and Control System Interface**

  This is the primary interface between the Monitor and Control System and the Operational Interface. It is through this interface that data and commands will be exchanged between the two systems.

- **e2e Interface**

  An e2e interface to the scheduling software will be required.

- **Maintenance Database Interface**

  Information from the maintenance database (currently MainSaver™) will be required. The Operational Interface will have read-only access to the database as MainSaver licensing restricts any modifications to the database from outside the MainSaver environment.

### 3.1.4 Communications Interfaces

No specific communications interface requirements have been addressed at this time.

## *3.2 Functional Requirements*

### 3.2.1 Remote Observing

The primary goal of the Operational Interface software is to provide the array operators with the tools to safely, effectively and reliably monitor and control the array. In meeting this goal the system designers must also build into the system a secondary goal that allows access to the system independent of location.

There are several reasons for remote observing:

1) Observers can monitor the progress of their observing program and make or request changes during their observation to increase the quality of data.
2) Hardware and software engineers will have the ability to access the system from remote locations during working and non-working hours to do first-order problem solving.
3) In the future, operators may be stationed at the AOC in Socorro.

The ability to remotely observe does not imply that the software must be Web-based and that it run within the confines of, or is launched from, a browser. It only suggests that the software be accessible via the internet and that it provide the same functionality (or subset thereof), remotely, as the user interface software used by the array operator.

At times, it may be necessary for the array operator to have some form of communications link with a remote observer other than telephone and/or email. A simple solution would be to create a “chat” program that allows the operator and observer to exchange messages in near real-time.

**Table 1: Remote Observing Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00001 | 0 | The software shall be accessible via the internet (downloadable, running within a browser, or launched from a browser, etc). |
| 00002 | 0 | A request for data or commands submitted by the operator or from an operator's station shall take precedence over requests and submissions by non-operators. |
| 00003 | 1 | A remote user with authorized privileges shall have the ability to communicate with the operator using a simple messaging system (chat tool). |

### 3.2.2 Installation and Upgrades

An installation and upgrade mechanism must exist so that users have access to the latest version of the software. The user's role and level of involvement in the installation and upgrade process must be determined early in the design phase as it will affect both the design of the product and the technologies used.

Several options exist for distributing the software, each requiring various levels of user involvement. A Web-based solution may provide the user with the latest software features without the user having to install anything, whereas, a more active user role would be assumed if the user has to navigate to a web site, download the software bundle and install it. If the user will be required to partake in any aspect of the installation process, the software should be designed such that the process of upgrading is automated as much as possible and it should guide the user through the entire installation/upgrade of the product.

**Table 2: Installation and Upgrade Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00004 | 0 | The system shall inform the user that a new version is available. |

| | | |
|---|---|---|
| 00005 | 0 | The system shall inform the user whether the new version is required or recommended. |
| 00006 | 0 | During the installation process, the system shall allow the user to cancel or delay the installation/update. |
| 00007 | 2 | The system shall automatically download and install the new software. |

### 3.2.3 System Login

All users will be required to log on to the system. Before a user can interact with the system he/she must first supply his/her system identification information (e.g., username and password) to the system. Using this information, the system will authenticate the user and either grant or deny the user access to the system.

**Table 3: System Login Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00008 | 0 | Each user must login to the system. |
| 00009 | 0 | Each user must have a unique identifier and password to gain access to the system. |
| 00010 | 0 | The system shall not echo password characters. |
| 00011 | 0 | The user shall have the ability to discontinue or cancel a login attempt at any time during the login process. |
| 00012 | 0 | After (TBD) unsuccessful login attempts the system shall (TBD). |
| 00013 | 0 | The system shall log, at a minimum, the following login and end of session information to a persistent store (db or file): |

| | | |
|---|---|---|
| | | 1) the time the session was established and ended<br>2) the user identifier<br>3) the status of the login attempt (login only)<br>4) the duration of the session (end of session only)<br>5) the cause of session termination (user, administrator, error, etc.) |

### 3.2.4 Warning and Error Messages

Warning and error conditions are meant to alert users of problems in the system that require immediate attention or less serious conditions that, if left unacknowledged, could have the potential to develop into more serious problems. Conditions that warrant warnings or errors include, but are not limited to, loss of communication with a subsystem, device errors, out of range monitor points, invalid device commands, network failures and software errors. The messages should accurately describe the problem and its consequences and provide access to the source (or cause) of the problem as well as a list of possible solutions to the problem.

The system should make every attempt to limit warning and error conditions to only those relevant to the current system activity. All others must be suppressed or ignored. It should not be the responsibility of the array operator to visually weed through a list of faults to determine the actual problem.

During the design process, the system designers will want to investigate fault tree analyzers developed elsewhere, e.g., BIMA and GBT.

**Table 4: Warning and Error Message Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00014 | 0 | The system shall allow messages to be color-coded based on the type and severity level of the message. |
| 00015 | 0 | The system shall provide audible alerts for fault conditions based on the type and severity level of the message. |
| 00016 | 0 | The user shall have the ability to suppress fault conditions |

| | | |
|---|---|---|
| | | for a specified period of time (minutes, hours, days, etc.) or event (source change). |
| 00017 | 0 | The user shall have the ability to view the fault conditions for the following groups:<br>1) a single antenna<br>2) all antennas<br>3) a subarray |
| 00018 | 0 | At a minimum, the system shall provide access to the following fault condition properties:<br>1) the date/time of the message<br>2) origin of the message (computer, subsystem, etc.)<br>3) the message type (WARNING or ERROR, etc.)<br>4) the message code<br>5) a terse description of the problem<br>6) a severity level<br>7) a detailed description of the problem<br>8) a detailed description of possible or likely consequences<br>9) a detailed description of possible or likely causes<br>10) a detailed description of corrective actions<br>11) a link to relevant or helpful documents |
| 00019 | 0 | The user shall have the ability to redirect warning or error messages to:<br>1) the operator (or system) log<br>2) a printer |
| 00020 | 0 | The user shall have the ability to retrieve and sort messages by time, message code, type, source (subsystem) and severity level. |
| 00021 | 2 | The user shall have the ability to create and enter a known solution to a problem and submit the solution to a persistent store for future reference. |

### 3.2.5 Monitor Points

A monitor point can be defined as an autonomous piece of data. Each monitor point should have a unique ID, description, value, units, etc. A user should have access to all the information pertaining to a monitor point at his/her disposal.

**Table 5: Monitor Point Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00022 | 0 | The user shall be able to plot any monitor point over a specified time range. |
| 00023 | 0 | The user shall be able to list a monitor point's values over a specified time range. |
| 00024 | 0 | The user shall be able to view the following monitor point properties:<br>1) date/time stamp<br>2) current value<br>3) raw units<br>4) units conversion factor (e.g., 1 volt = 10 degrees celcius)<br>5) engineering units<br>6) description (terse and detailed)<br>7) type (raw, derived, etc.)<br>8) data type (binary, decimal, hexadecimal, octal, logical, floating point, or integer) |

### 3.2.6 General Data Requirements

The monitor and control displays will allow users to visualize the current state of the system at varying levels of detail as well as provide the required control capabilities. The number and details of the M&C displays to be generated is unknown at this time, however, the data required by the displays can be addressed.

### 3.2.6.1 Weather/Environmental Data

**Table 6: General Weather Data Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00025 | 0 | The following weather data shall be available:<br>1) wind direction<br>2) wind speed<br>3) peak wind speed and time (with the ability to reset)<br>4) current temperature<br>5) dew point<br>6) barometric pressure |
| 00026 | 0 | The user shall have the ability to plot the history of any of the above weather data. |
| 00027 | 0 | The user shall have the ability to submit current weather data to the log. |

**Table 7: Atmospheric Phase Interferometer Data Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00028 | 0 | The user shall have the ability to display real-time atmospheric phase interferometer data. |
| 00029 | 0 | The user shall have the ability to display archival atmospheric phase interferometer data for a user-specified time range. |

**Table 8: Ionospheric Data Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00030 | 0 | The user shall have the ability to display real-time ionospheric data. |
| 00031 | 0 | The user shall have the ability to display archival ionospheric data for a user-specified time range. |

**Table 9: RFI Data Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00032 | 0 | The following information shall be available:<br>1) duty cycle<br>2) average signal received power<br>3) peak signal received power<br>4) origin<br>5) a matrix database that shows the presence of RFI at different thresholds greater than ITU thresholds and in 3dB steps<br>6) provide overlay to identify RFI in each band |

### 3.2.6.2 Antenna Data

**Table 10: Antenna Data Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00033 | 0 | The following information shall be available for each antenna:<br>1) antenna number<br>2) antenna address<br>3) station or pad ID<br>4) subarray affiliation<br>5) azimuth and elevation |

| | | |
|---|---|---|
| | | 6) current source<br>7) RA and Dec of current source<br>8) pointing errors<br>9) cable wrap<br>10) weather data (anemometers 1 & 2)<br>11) stop time of observation<br>12) affiliated project<br>13) observation title (observation file name)<br>14) slew rate (azimuth and elevation) |
| 00034 | 0 | Authorized users shall have the following antenna control capabilities:<br>1) park<br>2) stow<br>3) point<br>4) "avoid snow"<br>5) "tip for snow" |

### 3.2.6.3 Subarray Data

**Table 11: Subarray Displays Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00035 | 0 | The following data shall be available for each subarray:<br>1) affiliated project<br>2) current source<br>3) current position<br>4) antenna(s)<br>5) mode(s) |
| 00036 | 0 | Authorized users shall have the following subarray control capabilities:<br>1) park<br>2) stow<br>3) point<br>4) "avoid snow"<br>5) "tip for snow" |

| | | |
|---|---|---|

## 3.2.7 Scheduling[4]

The Operational Interface must provide the operators with the ability to manipulate individual scans (or scheduling blocks) prior to submitting them to the core M&C system for execution.

**Table 12: Scheduling Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00037 | 0 | The system shall expect scan stop times to be in VLA LST. (See Section 3.2.3 of the Scientific Requirements document.) |
| 00038 | 0 | The user shall have the ability to view the individual scans of a project. |
| 00039 | 0 | The user shall have the ability to reorder the scans. |
| 00040 | 0 | The user shall have the ability to set the end time for a specific source or scan (overriding the end time in the observation script file). |
| 00041 | 0 | At a minimum, the user shall have access to the following scan properties:<br>1) end time (in VLA LST)<br>2) source<br>3) position<br>4) frequency<br>5) equipment configuration<br>6) subarray affiliation<br>7) mode(s)<br>8) quality of calibrator |

[4] The day-to-day scheduling, either fixed or dynamic, is the responsibility of the e2e project. For a detailed description of scheduling see the e2e Project Book.

| | | |
|---|---|---|
| 00042 | 0 | The user shall have the ability to extend on a source or scan:<br>1) until a specified time<br>2) for a specified duration |

## 3.2.8 Log Facility (operator or system log)

The current VLA operator log contains information concerning an observer's project and is intended to alert the observer of conditions or events that occurred during the project that may have affected the collection or quality of the data. The operator logs are project based, meaning that a new log is created and exists for each project rather than a continuous entry of, possibly unrelated, time-stamped events. At the completion of a project the observer is e-mailed a copy of the log.

The logging facility should be designed as a distributed tool that can be easily embedded into other applications. There has been discussion that logging tools with similar functionality are needed at other sites for other instruments. The logging facility should thus be designed and developed with this in mind so that the tool is as general purpose as possible with preferential treatment given to the EVLA.

### 3.2.8.1 General Requirements

**Table 13: General Logging Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00043 | 0 | The system shall save log entries to a persistent store. |
| 00044 | 0 | The user shall have the ability to query the persistent store (by time range, message, type, program, etc.). |
| 00045 | 0 | The logging system must be distributed and platform independent. |

### 3.2.8.2 Log Entries

A log entry is any user or system generated message that will be submitted to the operator/system log. Authorized users should have the ability to create and submit log entries as well as modify and delete them.

| Req. ID | Priority | Description |
|---|---|---|
| 00046 | 0 | An authorized user shall have the ability to create a log entry. |
| 00047 | 0 | The system shall automatically provide the event timestamp at the moment the entry is created. |
| 00048 | 0 | An authorized user shall have the ability to select the type of entry (operator note, operator to operator message, antenna visit, observers note, weather, etc.). |
| 00049 | 0 | An authorized user shall have the ability to edit a log entry.[5] |
| 00050 | 0 | An authorized user shall have the ability to delete a log entry. [6] |
| 00051 | 0 | An authorized user shall have the ability to save or submit a log entry. |
| 00052 | 0 | The system shall automatically timestamp the entry on submission. |

[5] A new copy of the original entry could be made and edited and the original could be marked as "void".

[6] The entry could just be marked as "void" rather than deleting it.

### 3.2.8.3 Reminders

A reminder is simply a mechanism that allows the user to generate a message and have that message displayed at a specified time.[7] This feature does not necessarily have to be a part the logging system and could easily be promoted to an overall system feature.

**Table 14: Reminder Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00053 | 1 | The user shall have the ability to create a reminder. |
| 00054 | 1 | The user shall have the ability to specify all properties of the reminder (the trigger time, the message to display, etc.) |
| 00055 | 1 | The user shall have the ability to edit a reminder. |
| 00056 | 1 | The user shall have the ability to delete a reminder. |
| 00057 | 1 | The user shall have the ability to view all scheduled reminders. |
| 00058 | 1 | The user shall have the ability to specify the frequency of the reminder (once, every hour, daily, monthly, etc.) |
| 00059 | 1 | The system shall automatically delete all triggered and acknowledged nonrecurring reminders. |
| 00060 | 1 | The user shall have the ability to enter the reminder trigger time in any of the time formats:<br>1) UTC<br>2) VLA LST<br>3) local time |

[7] Most operating systems are equipped with a tool that provides similar functionality. The advantage of having it implemented in the software is to provide the user with a tool that has the same behavior regardless of the operating system.

| | | |
|---|---|---|
| 00061 | 2 | The user shall have the ability to link a reminder to an event (program change, source change, parameter value). |
| 00062 | 2 | The user shall have the ability to specify the method of notification: popup message (default), beeping, flashing, email. |
| 00063 | 2 | The user shall have the ability to postpone a reminder. |

### 3.2.8.4 Scheduled Tasks

A scheduled task is similar in functionality to the reminders, however, rather than displaying a message, a scheduled task would instead launch a script (or executable) at a specified time. This feature could be used to enter weather data on an hourly basis into the operator/system log.[8]

**Table 15: Scheduled Task Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00064 | 2 | The user shall have the ability to create a scheduled task. |
| 00065 | 2 | The user shall have the ability to edit a scheduled task. |
| 00066 | 2 | The user shall have the ability to delete a scheduled task. |
| 00067 | 2 | The user shall have the ability to postpone a scheduled task. |
| 00068 | 2 | The user shall have the ability to view upcoming scheduled tasks. |

[8] See previous footnote.

| | | |
|---|---|---|

### 3.2.8.5 Macros

The ability to record and save a macro would allow the user to create predefined entries that could be linked to user interface buttons. The current VLA operator log uses Excel macros to provide this functionality.

**Table 16: Macro Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00069 | 0 | The user shall have the ability to record a macro. |
| 00070 | 0 | The user shall have the ability to save a macro. |
| 00071 | 0 | The user shall have the ability to name a macro. |
| 00072 | 0 | The user shall have the ability to delete a macro. |
| 00073 | 0 | The user shall have the ability to run a macro. |
| 00074 | 0 | The user shall have the ability to link a macro to a user interface component (button or menu component) or keyboard shortcut. |

### 3.2.8.6 Maintenance database

This feature would allow the user limited access to the maintenance database (MainSaver).[9] Simple queries could be performed so that users could locate maintenance database entries for known problems that are potentially causing or

[9] Due to MainSaver licensing restrictions, the database cannot be updated through non-MainSaver applications. This means that the database can only be queried.

having some impact on an observation. When the user finds the database entry he or she should be able to link that entry to a log entry.

**Table 17: Maintenance Database Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00075 | 0 | The system shall have the ability to search the maintenance database. |
| 00076 | 0 | The system shall have the ability to link maintenance database entries to log entries. |

### 3.2.8.7 Reporting

The user should have the ability to create a number of reports from the data contained in the log. A standard package of reports should be available to the user and possibly in the future a report-building feature would be added to the system.

**Table 18: Reporting**

| Req. ID | Priority | Description |
|---|---|---|
| 00077 | 0 | The user shall have the ability to create a report for a specified project.[10] |

## 3.2.9 User Management

This subsection defines the requirements needed to control and manage the users of the system as well as a user's access to the system. The use of the term administrator below refers to the array operator.

[10] The result of such a report would be equivalent to the current observing log.

**Table 19: User Management Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00078 | 0 | The administrator shall have the ability to create and add a new user to the system. |
| 00079 | 0 | The administrator shall have the ability to remove a user from the system. |
| 00080 | 0 | The administrator shall have the ability to edit a user's system access properties. |
| 00081 | 0 | The administrator shall have the ability to block all commands sent to the array by all users or selectively by user. (All users will remain logged into the system and will not be allowed to issue commands to any part of the array until the command block is released. The administrator will still have the ability to issue commands to the array.) |
| 00082 | 0 | The administrator shall have the ability to block all access to the system for all users or selectively by user. (All blocked users with active sessions will automatically be logged off.) |

### 3.2.10 Online Help

A help facility should be incorporated into the design of the software system from the start and it should allow the array operator to find the information he/she seeks quickly and accurately. At a minimum, it should include a table of contents, an index, full-text search and context sensitive help.

**Table 20: Online Help Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00083 | 0 | The system shall provide an online help facility with the following features:<br>1) table of contents |

| | | |
|---|---|---|
| | | 2) index<br>3) full-text search |
| 00084 | 1 | The online help facility shall provide context sensitive help. |

### 3.2.11 Documentation

**Table 21: Documentation Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00085 | 0 | The system shall provide the following online documentation for all tools associated with the system:<br>1) user manual |

### 3.2.12 Printing

Users of the system will need the ability to capture and print screens as well as the ability to create and print reports.

**Table 22: Printing Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00086 | 0 | The user shall have the ability to capture and print any display. |

### 3.2.13 Plotting

A general-purpose plotting component will be needed. Several of the standard M&C displays will likely contain plots of monitor points relevant to that display and a user must have the ability to plot any monitor points on the fly.

**Table 23: Plotting Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00087 | 0 | The user shall be able to generate the following plot types:<br>1) scatter plot<br>2) histogram<br>3) line plot |
| 00088 | 0 | The user shall be able to view the plot dynamically (real-time) or statically (offline). |
| 00089 | 0 | The user shall have the ability to give the plot a title. |
| 00090 | 0 | The user shall have the ability to define the plot axis labels in the following manner:<br>1) user defined character string<br>2) import predefined labels from a monitor point definition database |
| 00091 | 0 | The user shall have the ability to specify the number of major and minor tick marks for a plot. |
| 00092 | 0 | The user shall have the ability to view multiple monitor points on a single axis. |
| 00093 | 0 | The user shall have the ability to specify whether the major and minor tick marks are:<br>1) linear<br>2) logarithmic |
| 00094 | 0 | The user shall have the ability to auto-scale plots. |
| 00095 | 0 | The user shall have the ability to specify the scale of the plot for all axes. |

| | | |
|---|---|---|
| 00096 | 0 | The user shall be able to plot any monitor point in the system. |
| 00097 | 0 | The user shall be able to plot the same monitor point for all antennas or selected antennas. |
| 00098 | 0 | The user shall have the ability to infinitely zoom in and out of the plot. |
| 00099 | 0 | The user shall be able to select various point styles:<br>1) none<br>2) dots<br>3) points<br>4) shapes (triangles, squares, circles, etc.) |
| 00100 | 0 | The user shall be able to show error bars. |
| 00101 | 0 | The user shall have the ability to view the plot's legend. |
| 00102 | 0 | The user shall be able to print a hardcopy of the plot. |
| 00103 | 0 | The system shall identify the following characteristics of a monitor point over a specified time range:<br>1) minimum value<br>2) maximum value<br>3) average<br>4) RMS |
| 00104 | 0 | The user shall have the ability to set maximum and minimum limit markers. |

## *3.3 Performance Requirements*

The following requirements describe the performance expected from the Operational Interface.

**Table 24: Performance Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00105 | 0 | The system displays shall be capable of, at a minimum, a (TBD) data refresh rate over a low-bandwidth connection. |
| 00106 | 0 | The operator stations and technician stations must be capable of much greater refresh rates, not less than (TBD). |
| 00107 | 0 | The system shall support a minimum of 30 concurrent user sessions.[11] |

## *3.4 Logical Database Requirements*

There are no known database requirements at this time.

## *3.5 Design Constraints*

There are no known design constraints at this time.

## *3.6 Software System Attributes (Non-functional Requirements)*

### 3.6.1 Reliability

The following requirements describe the expected reliability of the Operational Interface.

**Table 25: Reliability Requirements**

| Req. ID | Priority | Description |
|---|---|---|

[11] This value is generated by doubling the maximum number of expected concurrent users, which is 15.

| | | |
|---|---|---|
| 00108 | 0 | The Operational Interface shall have a Mean Time Between Failures (MTBF) of no less than 7 days.[12] |

### 3.6.2 Availability

The following requirements indicate the expected availability of the Operational Interface.

**Table 26: Availability Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00109 | 0 | The system shall be available 99.5%[13] of the time. |

### 3.6.3 Security

The software system needs a robust security mechanism in place so that unauthorized users are not allowed access to parts of the system that may compromise the success of an observation, cause damage to an antenna or jeopardize the safety of personnel in or around an antenna.

This mechanism must be built into the system from the start. An attempt to add security to the system after the fact would likely prove to be a costly and difficult task.

All users of the system must be uniquely identified. This could be done by using a username and associated password scheme that would authenticate and authorize the user access to the system and, if applicable, grant the user access to restricted or controlled parts of the system. If a user cannot be identified, he/she will be given "anonymous" access with read-only capabilities. In order to monitor all past access to the system, all attempts to access the system must be logged.

[12] A Mean Time To Repair (MTTR) is not supplied, as it is operationally defined and highly variable.

[13] This works out to the system being unavailable a total of 48 hours over a year's time.

Users will be separated into two groups, those we trust and those we don't. Trusted users will have privileged access to the system, namely control capabilities, whereas the non-trusted users will only have monitoring capabilities.

**Table 27: Security Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00110 | 0 | All users of the system shall login using some form of unique identification (e.g., username and password) |
| 00111 | 0 | A default user identifier shall exist that allows anonymous, read-only access to the system (e.g., "guest") |
| 00112 | 0 | All login attempts shall be done so in a secure manner (e.g., encrypted passwords) |
| 00113 | 0 | A system administrator shall have unrestricted access to all aspects of the system. |
| 00114 | 0 | Each user shall either be trusted or not trusted. |

### 3.6.4 Maintainability

The following table includes requirements that increase the maintainability of the Operational Interface software.

**Table 28: Maintainability Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00115 | 0 | All source code and development related documents shall be controlled under a version control system (e.g., CVS, RCS or SCCS). |
| 00116 | 0 | All source code shall adhere to an agreed upon and well-defined set of coding standards for each development |

| | | |
|---|---|---|
| | | language used. |
| 00117 | 0 | A standard naming convention for classes, variables and packages shall be agreed upon and adhered to. |

### 3.6.5 Portability

#### 3.6.5.1 Hardware

The manufacturer and type of workstations to be used by the array operators and other authorized users to monitor and control the array are unknown. In the interest of minimizing the cost of these computers and without unnecessarily limiting the available hardware options - now and in the future - the software must be reasonably platform independent so it will run on a variety of machines with minimal impact to the software. This would allow a change in the underlying hardware without requiring a significant rewrite of the software. It would only require changes to a configuration file or to a small layer of the software system.

**Table 29: Hardware Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00118 | 0 | The system shall be compatible with commodity PCs. |
| 00119 | 0 | The system shall be compatible with Sun Microsystems workstations. |
| 00120 | 2 | The system shall be compatible with Macintosh computers. |

#### 3.6.5.2 Operating Systems

Since the software must run on several popular hardware platforms and the goal is to achieve a reasonable level of platform independence, the software must also be capable of running on a number of operating systems supporting that hardware. At a minimum, the software should run on Linux, Solaris and Windows operating systems, all of which are supported by NRAO's staff of system administrators. As

with the hardware, a change in operating systems should have minimal impact on the software system.

**Table 30: Operating System Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00121 | 0 | The system shall be compatible with the Linux Operating System (Red Hat 7.0 or Greater). |
| 00122 | 0 | The system shall be compatible with the Solaris Operating Environment (Solaris 8 or greater). |
| 00123 | 0 | The system shall be compatible with the Microsoft Windows Operating System (NT, 2000, XP or greater). |
| 00124 | 2 | The system shall be compatible with Macintosh (Mac OS X or greater) |

### 3.6.6 Usability

The system's user interface should be intuitive, easy to use and provide an overall positive user experience. It should do what the user expects it to do, inform the user of its current state, and when something goes wrong it should explain the problem in a meaningful context that is understandable by the user and offer guidance toward correcting the problem. It should also conform to a specified set of user interface guidelines to foster consistency between different tools within the application.

An intuitive and easy to use application will undoubtedly decrease the amount of time it currently takes to train an operator. The operator should only be concerned with what they need to do with the tool to perform the task at hand rather than focusing on how to use the tool.

**Table 31: General Usability Requirements**

| Req. ID | Priority | Description |
|---|---|---|
| 00125 | 0 | The system shall adhere to a set of user interface design guidelines. |
| 00126 | 0 | The system shall allow the user to display multiple windows simultaneously. |
| 00127 | 0 | Each display shall reflect the level of access and the privileges of the user (security aware). |
| 00128 | 0 | If the user requests to close a window or exit the system after making uncommitted changes to a screen, the system shall prompt the user to commit or cancel those changes. |
| 00129 | 0 | The system shall use (TBD) as the default time format. |
| 00130 | 0 | Time shall always be available and displayed in the following formats:<br>1) Coordinated Universal Time (UTC)<br>2) VLA local sidereal time (VLA LST)<br>3) Local time |
| 00131 | 0 | The user shall have the ability to copy (or cut) text to the system "clipboard" and paste the text to other text-accepting components. |
| 00132 | 1 | Where applicable, the user shall have the ability to select preferences (color, font, etc.) |
| 00133 | 2 | In situations that are applicable, the system shall allow for "Undo" and "Redo". [14] |
| 00134 | 2 | The system shall be internationalized.[15] |

[14] The requirement for undo/redo may or may not be applicable in a real-time control context, but it would be appropriate for the logging and chat tools.

# Appendix A

## Prototypical Operator's Workstation

The operator workstation is the computing system designated to run the software used by the array operators to monitor and control the array. The workstation(s) will be located in the control room of the VLA Control Building and possibly the AOC in Socorro.

The number of workstations and/or terminals required is undecided at this time and is somewhat dependent on the working style and personality of the operator. However, at a very minimum, the operator should have two workstations, with at least one of them having multiple monitor capability. One workstation would be used for array monitor and control tasks and the other would be used for logging, email and other non-M&C activities. Also, in the event that one of the systems becomes unusable the working system can be used as a backup system to monitor and control the array.

The specifications provided below are to be used as a guide in selecting such a system and does not target any particular manufacturer or system model.

- 2GHz processor
- 1 GB RAM
- 80 – 100 GB hard drive
- CD-RW drive
- DVD-ROM drive
- Keyboard and mouse (perhaps wireless)
- 19" display (perhaps digital flat panel)

[15] Internationalized applications are easy to tailor to the languages of users in other countries. It also, has a desirable side affect of enforcing good coding practices by separating locale specific information from the source code. For example human readable labels are contained in files that are loaded at runtime. A change in a label does not require a recompile of the source code.