SOFTWARE REQUIREMENTS SPECIFICATION

EVLA Correlator Backend

Project Document: A25251N0000

Revision 1.0

Tom Morgan, March 5, 2002

National Radio Astronomy Observatory

Array Operations Center

P.O. Box O

Socorro, NM 87801-0387

Revision History

Date	Version	Description	Author
05-Mar-2002	1.0	Initial draft	Tom Morgan

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 Correlator Backend System. 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.

The requirements contained in this document are numbered based on the section/subsection in which they appear. Each has a description and has been assigned a priority (0 = *essential*, 1 = *desirable* and 2 = *if possible*).

Note: Text found between “<” and “>” indicates questions or comments to myself and/or readers. And In most cases, the phrase “The user” can be replaced with “An authorized user”.

1.2 Scope

The Correlator Backend System is the primary component of the real-time data processing pipeline resting between the Correlator and the End-to-End System. Its primary responsibility is to perform basic data assembly, formatting and processing services and to support the need for real-time inspection of the Astronomical data stream.

Some of the functions the Correlator Backend System must provide are as follows:
- The ability to receive data from the correlator in real-time.
- The ability to assemble time-series from the Correlator lag output.
- The ability to perform Fourier Transforms of the assembled time series.
- The ability to perform additional processes upon user request.
- The ability to deliver suitably formatted results to the End-to-End System.

This document will define only those requirements that must be fulfilled by the Correlator Backend System.

1.3 Definitions, Acronyms, and Abbreviations

1.3.1 Definitions

Administrator – An individual with unrestricted access to all aspects of the system.
Auxiliary Data – All other (non-astronomical) data.
Data – Astronomical observation data.
Metadata – All data about the astronomical data.

Non-real-time – Offline operations with data input from some external storage device or generated internal (e.g., for testing).
Processor – A physical computation device (hardware).
Process – A data processing procedure (software).
Real-time – Online operations with active astronomical data streaming from the Correlator.

1.3.2 Acronyms

AOC –Array Operations Center
M/C – monitor and control
EVLA – The VLA Expansion Project
SyRS – Refers to the *System Requirements* document.
SRS – Refers to the *Software Requirements Specification* document.

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
4) EVLA Project Book
5) EVLA System Requirements (SyRS)
6) EVLA Architecture and Design
7) The Very Large Array Observing Log (J. Nieri, February 1994)
8) Refined EVLA WIDAR Correlator Architecture, NRC-EVLA Memo# 014, Brent Carlson, Oct. 2, 2001.
9) EVLA Correlator Monitor and Control System,. Test Software, and Backend Software Requirements and Design Concepts, NRC-EVLA Memo # 015, Brent Carlson, Jan. 23, 2002.

1.5 Overview

The remainder of this document contains a more detailed description of the Correlator Backend System as well as the requirements necessary to design and build the system. Section 2 provides a general description of the Correlator Backend System. Section 3 details the requirements of the product and is the core of this document.

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

2 Overall Description

2.1 Product Perspective

The EVLA Correlator Backend System will be designed and implemented as a real-time data processing system. Computers in the system will all be exactly the same and operating systems and applications running on them will communicate with one another and the Monitor and Control System over a network. Data input to the system from the Correlator and output from it to the End-to-End System will be over very high speed networks. Currently, only a conceptual diagram exists for the system and should be viewed as such (see Figure 1)

2.2 Product Functionality

2.2.1 Data Input

Incoming Correlator lag data will be received, assembled into time series, and evaluated for completeness and correctness.

2.2.2 Data Processing

The assembled time series will be Fourier Transformed and other user selectable time and/or frequency domain processes will be applied

2.2.3 Data Output

The processed time series and their auxiliary data and parameters will be formatted and transferred to the End-to-End System.

2.2.4 Monitoring

The Correlator Backend System will conduct a number of self-monitoring activities on application and system software as well as hardware systems to detect system failure and out of spec conditions.

2.2.5 Recovery

The ability to recover from failure and out of spec performance conditions will be built into the system.

2.2.6 Control

The system will provide control and auxiliary parameters to the input, output, processing, monitor and recovery functions and receive status and performance data from them. It will also communicate with the external Monitor and Control System.

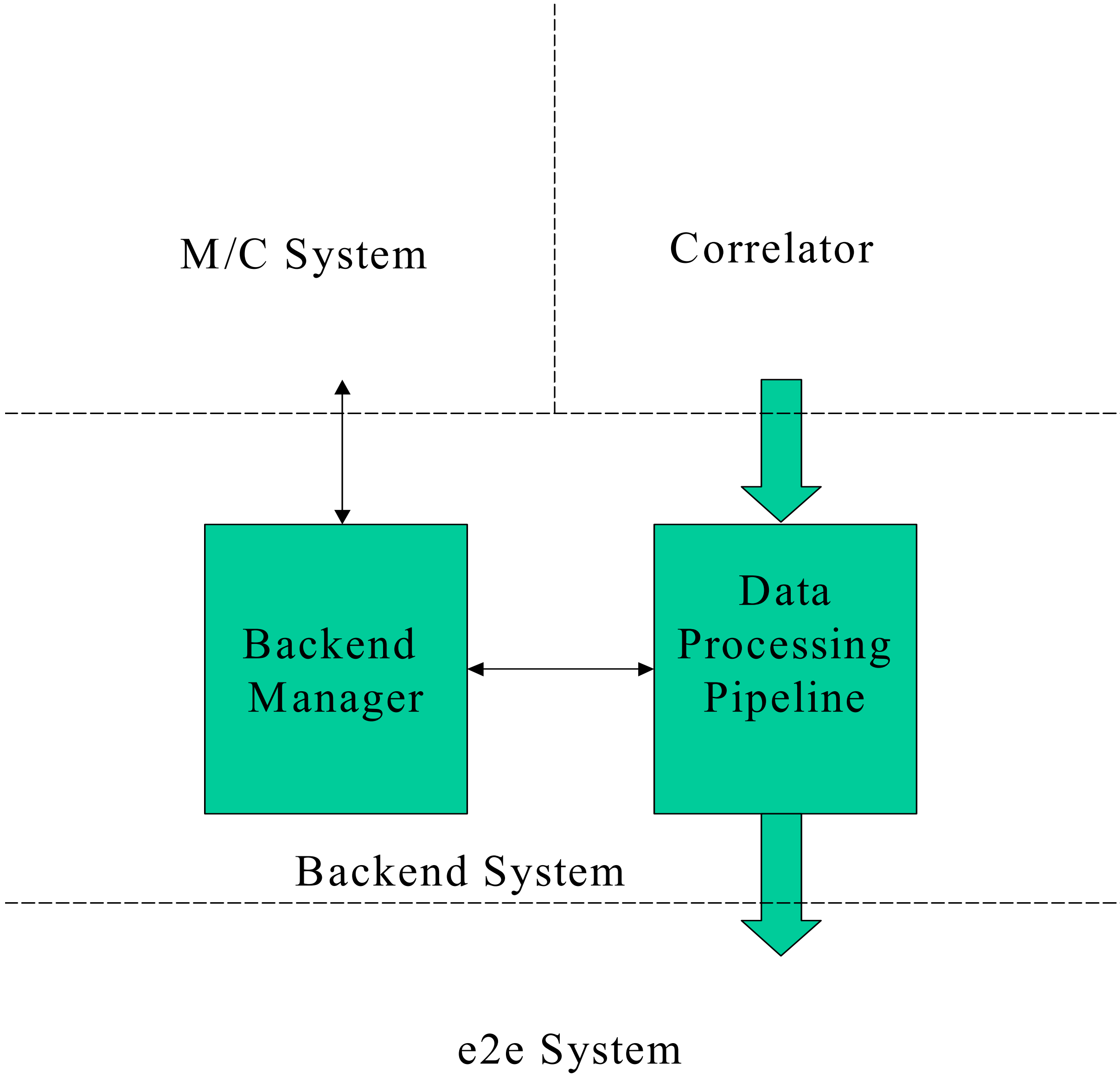

Figure 1: Correlator Backend System Main Functional Components Diagram

2.3 User characteristics

All use of the Correlator Backend System will be indirect via the Monitor and Control System.

2.3.1 Array Operator

The primary contact with array operations will be via status and error messages channeled through the Monitor and Control System.

2.3.2 Engineers and Technicians

The ability of the Backend System to achieve and maintain real-time processing will be vitally dependent upon reliable operation and rapid diagnosis and repair of faults in the hardware and software systems. These individuals will be responsible for performing corrective and preventive maintenance along with periodic performance tests and upgrades. Engineers and technicians will need tools to inspect individual devices from remote locations.

2.3.3 Astronomer/Scientist

These individuals are primarily interested in the science that is obtained from the instrument. Their main interaction will be to select and provide parameters for any additional data processing beyond the Fourier transforms.

2.3.4 Software Developer

These individuals are responsible for developing the software and will interact with the system to ensure that it is functioning properly. The software developer requires remote access to the system so that troubleshooting can be accomplished away from the EVLA and during non-working hours.

2.3.5 Web User

A few authorized individuals may be allowed access to parts of the system that are usually considered restricted.

2.4 Constraints

2.4.1 Criticality of the Application

The Correlator Backend System is a critical component in the Astronomical data path. If it is unavailable, incoming astronomical data will be lost.

2.4.2 Computer Hardware Limitations

The ultimate throughput capability of the real-time data processing pipeline of the Backend System will be constrained by the computational performance limits of available computer hardware and the practical ability to configure and maintain large numbers of processors.

2.4.3 Communications Limitations

The ability to realize and maintain real-time operations is critically dependent upon the performance levels of available network systems.

2.4.4 Reliability

The ability to maintain real-time operations over realistic extended periods of time is dependent on the mean time to failure of the hardware and software components of the computing and communications systems.

2.5 Assumptions

2.5.1 Incoming Data Stream

It is assumed that the Correlator will deliver suitably formatted network data packets to the input network of the Backend System.

2.5.2 Auxiliary Data

It is assumed that all auxiliary data needed for processing operations are provided directly by the correlator via the incoming data packets or by the Monitor and Control System.

2.5.3 Outgoing Data Stream

It is assumed that the End-to-End System will be capable of accepting output data rates and volumes generated by the Backend System.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 Correlator to Backend Interface

The primary astronomical data interface between the two systems. Correlator lag data in the form of LTA frames will flow across it from the Correlator Baseline Boards to the Backend data processing computers.

3.1.2 Backend to End-to-End Interface

The primary astronomical data interface between the two systems. Backend produced complex spectra formatted as AIPS++ Measurement Sets or components thereof will flow across it. The output data sets will include information from external sources obtained via the Monitor and Control System as well as auxiliary and meta- data generated during processing.

3.1.3 Backend to/from Monitor and Control Interface

The primary auxiliary data and command/control data interface between the Backend System, the other components of the overall EVLA System and the outside world

3.2 Functional Requirements

3.2.1 Information and data flows

Req. ID	Priority	Description
3.2.1.1	0	External Auxiliary Data – request, obtain and acknowledge receipt of all necessary parameters and system state and configuration data for the Correlator and End-to-End Systems.
3.2.1.2	0	External Input Astronomical Data – obtain and acknowledge receipt of data from the Correlator.
3.2.1.3	0	External Output Astronomical Data – deliver and receive acknowledgement of receipt of data to the End-to-End System.
3.2.1.4	0	Error Reports – Deliver (without necessarily being probed) and receive acknowledgement of receipt of internal Backend System error conditions as they occur. These include, but are not limited to, hardware failures, software interrupts and failures, I/O and processing performance deficiencies, and resultant swap-in of new resources events.
3.2.1.5	0	Run State Reports – The Backend System must be probeable from outside. It must be able to accept and deliver upon requests made to determine its run state.

3.2.2 Process Descriptions

Req. ID	Priority	Description
3.2.2.1	0	Data Receive – receive incoming data packets from the Correlator to Backend network interface.
3.2.2.2	0	Verify Receive – verify that incoming data from the Correlator has been received successfully. This includes checking for receive errors and determining that all expected data was received. Accumulate error statistics and compare against tolerances. Report all out of tolerance conditions.
3.2.2.3		Input Data Manager – store input data records in a memory buffer and track buffer locations of all input data until data processing is complete. Report any errors from the receive operation.
3.2.2.4	0	Processing Manager – handle incoming correlator mode changes, user optional processing sequence and/or parameter changes, and other external inputs that affect the data processing pipeline. Update internal parameter tables and synchronize data processing pipeline with new operational conditions.
3.2.2.5	0	Time Series Assembly – assemble the received input data into continuous time series.
3.2.2.6	0	Data Integrity Verification – verify that time series data is correctly ordered and contains valid data values along its entire extent. Flag and repair invalid data values. Accumulate data integrity statistic and compare against tolerances. Report all out of tolerance conditions.
3.2.2.7	1	Time Domain Processing – apply user selected time domain processes. These processes should be constructed to be chainable (output of any time domain process can be piped to input of any other, including replica of self and Fourier Transform) and repeatable in the chain.
3.2.2.8	0	Fourier Transform Processing – power-of-two complex-to-complex Fast Fourier Transform with retention of all output positive and negative frequencies. This process must be able to accept as input the output of any of the time domain processes or the data integrity process.
3.2.2.9	1	Frequency Domain Processing - apply user selected frequency domain processes. These processes should be constructed to be chainable (output of Fourier Transform and any frequency domain process can be piped to input of any frequency domain process including replica of self) and repeatable in the chain.
3.2.2.10	0	Integration – summing of spectra coming out of the Fourier Transform or in the last frequency domain process.
3.2.2.11	0	Output Formatting – combine the finished spectra with auxiliary data to form AIPS++ Measurement Set entities.
3.2.2.12	0	Output Data Manager– store formatted output data records in a memory buffer with backup disk buffering. Track the location of data ready for transmission to the End-to-End System until data receipt has been acknowledged. Report any errors that occur.
3.2.2.13	0	Data Send – send output data to the End-to-End System.
3.2.2.14	0	Send Verify – verify that all sent data was successfully received. Report all errors.
3.2.2.15	0	Monitor I/O Performance – determine input from Correlator and output to End-to-End data transfer rates. Accumulate data transfer statistics and compare against tolerances. Report all out of tolerance conditions.
3.2.2.16	0	Monitor Compute Performance – determine overall data processing rate. Compare against tolerance and report all out of tolerance conditions.
3.2.2.17	0	Monitor Compute Errors – trap and repair inf’s, NaN’s , underflows,

		overflows and other computation errors. Accumulate computation error statistics and compare against tolerances. Report all out of tolerance conditions.
3.2.2.18	0	Monitor Processes – periodically or upon request check PID's and assure that all started task are alive and running. Report missing, stopped, defunct and other damaged processes.
3.2.2.19	0	Monitor Processors – periodically or upon request check Backend physical processors and assure that all needed processors are alive and responding. Report all crashed, stropped, or unresponsive processors.
3.2.2.20	0	Monitor Networks – periodically or upon request check all Backend internal networks and assure that all communication connects are intact and functioning. Report all non-functioning components.
3.2.2.21	0	Start Process – initiate a processing task on an available Backend processor.
3.2.2.22	0	Stop Process – kill a processing task running on a Backend processor regardless of current state.
3.2.2.23	0	Reboot Processor – reboot a Backend physical processor regardless of current state.
3.2.2.24	0	Reboot network – reboot some or all of the Backend System internal network connections regardless of current state.
3.2.2.25	0	Control Processors – convert processor failure reports into processor reboot requests. Covert M&C System data on correlator mode and user processing requests into an estimate of the number of processors needed to accomplish real-time activities. Acquire needed processing resources. Report all reboot, reboot failures and resource shortfalls to M&C System.
3.2.2.26	0	Control Internal Networks - convert internal Backend network failure reports into network reboot requests. Report all reboots and reboot failures to M&C System.
3.2.2.27	0	Control Processes – convert process failure reports into process stop and start requests. Covert M&C System data on correlator mode and user processing requests into an estimate of the number of processes needed to accomplish real-time activities. Generate necessary start or stop processing task requests. Report all process stops, start failures, and resource shortfalls to M&C System.
3.2.2.28	0	Control Processing – Report all external data I/O (receives from Correlator, sends to End-to-End) errors and performance out of tolerance conditions to the M&C System. Report all processing rate, and processing error rate out of tolerance conditions to the M&C System.

3.2.3 Data Construct Specifications

Req. ID	Priority	Description
3.2.3.1	0	Input Data Queue – a buffer of one dimensional arrays of time series data plus time series sequence number, length and other parameters needed to assemble the time series in proper order and perform all requested processes. Data entry status queue to track each record in the buffer.
3.2.3.2	0	Output Data Queue – a memory buffer plus backup disk storage of all output AIPS++ Measurement Set entities. Data entry status queue to track each record in the buffer.
3.2.3.3	0	Processing Parameters – names, position(s) in sequence, and adjustable parameters for all fixed and user selectable processing pipeline

		applications.
3.2.3.4	0	Processing flags – a table of flags needed to identify various internal conditions relating to error response and processing state.
3.2.3.5	0	Metadata – All internally generated data about the processed time series including invalid data repairs, processes applied, coordinates, etc.
3.2.3.6	0	Error Reports – error number (translatable into textural error message), error source, error rates (as applicable), and time stamp.
3.2.3.7	0	Failure Reports – internal system component (e.g., disk drive, processors, processes, and networks) failure number (translatable into textural error message), failure source, and time stamp.

3.3 Performance Requirements

3.3.1 General

3.3.1.1	0	Data Integrity – the Backend System shall maintain input data fidelity and dynamic range across all processing, manipulation and I/O functions.
3.3.1.2	0	Error Handling – the system must be capable of flagging and marking corrupted data segments and proceeding without interruption or effect on other data. This includes, but is not limited to, partial data, zero data, underflows, overflows, infinities, and NaN's whether obtained on input or arising during processing.

3.3.2 Hardware

Req. ID	Priority	Description
3.3.2.1	0	Input – capable of accepting an aggregate data input stream from the Correlator of a minimum of 1.6 Gbytes/sec within acceptable loss tolerance specifications (TBD). This must be done simultaneously with the output stream, but not necessarily over the same interconnects.
3.3.2.2	0	Output – capable of delivering an output data stream to the End-to-End System of a minimum of 25 Mbytes/sec (including resends and simultaneous transfer of data stored due to End-to-End connection outage) without loss. This must be done simultaneously with the output stream, but not necessarily over the same interconnects.
3.3.2.3	0	CPU – processor capability (combination of numbers of processors and individual processor speed) sufficient to accomplish all processing tasks while avoiding loss or delay on the input and output data streams, plus a standby reserve (amount TBD) to meet reliability needs and handle outage recovery demands.
3.3.2.4	0	Memory – sufficient (amount TBD) memory with sufficient (rate TBD) access speeds to accomplish all processing tasks while avoiding loss or delay on the input and output data streams.
3.3.2.5	0	Excess Storage – sufficient storage (memory and/or disk) with sufficient access speeds to meet short duration Correlator bursting demands (level TBD) plus a standby reserve (amount TBD) to meet reliability needs and handle outage recovery demands.

3.3.3 Software

3.3.3.1	0	Applications – all math/science application software must take optimal advantage of all language, compiler, and system computational features and resources to reduce run times to the minimum practical level.
3.3.3.2	0	Management – all management software functions must take optimal advantage of all language, compiler and system features and resources to reduce overheads to the minimum practical level.
3.3.3.2	0	I/O – all input and output, and storage and retrieval operations must take optimal advantage of all system resources to reduce overhead and latency to the minimal practical level.
3.3.3.4	0	Processing – all data processing functions shall be chainable (outputs pipeable to inputs) and repeatable in the processing pipeline in cases where this makes computational sense.

3.4 Reliability/Availability

3.4.1	0	Auto-correction – the Backend System must be self-monitoring and capable of detecting, reporting on and automatically taking action to remedy or lessen the impact of the following abnormal conditions: processor hardware failure, operating system hangs or crashes, computational performance below minimum specifications, computational error rates above maximum specification, internal communications failures, and external communications disruptions.
3.4.2	0	Software – the software part of the system shall be able to perform without restart due to internal failure for a minimum of thirty days (generally the current period between array maintenance slots).
3.4.3	0	Hardware – the hardware part of the system shall be able to perform indefinitely without complete loss of service, except in the event of total failure of primary and backup power.
3.4.4	0	Correlator mode changes – the system must be capable of responding in a loss-less manner to I/O and processing changes arising from Correlator mode changes.
3.4.5	0	Loss of End-to-End – the system shall continue to operate in a loss-less manner in the event of a temporary (time duration TBD) loss of availability of the End-to-End System.
3.4.6	0	Loss of Correlator – the system shall be able to complete processing of all onboard data, deliver the results to the End-to-End System and maintain availability for immediate resumption of operations once Correlator access is restored.
3.4.7	0	Operations Activities – the system shall continue operations, although not necessarily at full capacity, on all unaffected resources during partial shutdowns for maintenance, repair and/or upgrade.
3.4.8	0	Loss of M&C – the system shall continue to operate during the absence of the M&C System until the first encounter of unavailable critical auxiliary data. The system shall be able to cache a predetermined amount (TBD) of correlator data after the first encounter of unavailable critical data (see 3.4.6 above) and complete all requested operations on cached data once the unavailable critical data is obtained.
3.4.9	0	Standby Mode – the system shall be able to sit at idle and resume operations with minimal (amount TBD) delay.

3.4.10	0	Test Mode – the system shall be able to handle non-real-time operations in a transparent fashion (i.e., as if real-time). Note: non-real-time refers to input data from a source other than the Correlator (defined as real-time).
3.4.11	0	Replaceability – maximal practical use of available “hot-swappable” devices and components shall be made.

3.5 Serviceability

3.5.1	0	Hardware Accessibility – all system processing and interconnect hardware must be readily accessible for maintenance, repair, replacement and/or reconfiguration.
3.5.2	0	Software Accessibility – all systems and application source code shall be available to or on the systems that execute it.
3.5.3	0	Debugging – all software application modules must be debuggable.
3.5.4	0	Processes – all software processes must be killable, restartable, debuggable and testable without affecting normal operations.

3.6 Maintainability

3.6.1	0	Hardware – complete and comprehensible hardware systems specifications and configuration information should be readily available.
3.6.2	0	Software Coding Practices– software system and application code shall be well documented and written in a generally familiar language or languages (preferably not more than two). Software shall be written in a style that is easily readable and using practices that allow for minimal confusion.
3.6.3	0	Software tools – software tools and pre-built applications that do not have source code available must come with a complete diagnostic package and customer support.
3.6.4	0	Operating Systems – operating system software shall either have source code available or come with sufficient diagnostics and customer support.

3.7 Scalability

3.7.1	0	Hardware – I/O, communications, and processing hardware must be easily expandable, reconfigureable, augmentable and replaceable to meet increasing data transfer and processing demands imposed by EVLA science, Correlator changes, and availability of new hardware.
3.7.2	0	Transparency – 3.7.1, above, must be accomplished in manner that is transparent to processing, communications and I/O software functions with the possible exception of recompilation of executables.
3.7.3	0	Seamlessness – 3.7.1, above, must be accomplished in a manner that is seamless, in that it does not affect hardware modules or software functionality that it meets at interfaces.
3.7.4	0	Performance – the Backend system must ultimately be scaleable to an extent that it will be capable of handling up to two Gbytes per second per Correlator output channel in real-time.

3.8 Security

The Backend System needs a robust security mechanism in place so that unauthorized users are not allowed access. Authorized users are expected to be restricted to software and hardware development, testing, maintenance and operations personnel.

All users of the Backend System must be uniquely identified. This could be done via 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, they will not be given access. In order to monitor all past access to the system, all attempts to access the system should be logged.

Users' needs and expectations from the system will be different. Systems operations should be given unrestricted access to all aspects of the system and should have the authority to grant and revoke privileges on a per-user basis. Development, testing and maintenance personnel, on the other hand, require access to some parts of the system, but not all, indicating that an access level is needed that allows privileges to be granted on a per-user and what-do-you-need-to-do basis.

Req. ID	Priority	Description
3.8.1	0	All users of the system shall login using some form of unique identification (e.g., username and password)
3.8.2	0	All login attempts shall be done in a secure manner (e.g., encrypted passwords)
3.8.3	0	A system administrator shall have unrestricted access to all aspects of the system.
3.8.4	0	Each user shall have a set of system access properties that defines the user's privileges within the system (e.g., the subsystems a user may control or system tools the user may access).
3.8.5	0	The administrator shall have the ability to create and add a new user to the system.
3.8.6	0	The administrator shall have the ability to remove a user from the system.
3.8.7	0	The administrator shall have the ability to edit a user's system access properties.
3.8.8	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.)

3.9 Installation and Upgrades

3.10 Documentation

.