EVLA Monitor and Control Software

Status as of Q2 2005

Version 1.0.0

file:
/users/bsahr/EVLA/architecture_design/MandC_design/MandC_NearTerm_StatusI.doc

Revision History

Revision	Date	Author(s)	Description of Changes
1.0.0	June 20, 2005	Bill Sahr	Original Version

Table of Contents

1 Introduction 3
2 Phases I of the EVLA M&C Transition 3
2.1 Phase I, EVLA M&C Transition 3
2.2 Interim Antsol/Telcal 3
3 Phase I, EVLA M&C Transition 4
3.1 Objective: Use of EVLA Antennas in VLA Observing 4
3.1.1 Relevant Milestones & Target Dates: Use of EVLA Antennas in VLA Observing 4
3.1.2 Timeline: Use of EVLA Antennas in VLA Observing 4
3.1.3 Task Breakdown: Use of EVLA Antennas in VLA Observing 4
4 WIDAR Correlator 17
4.1 Objective: WIDAR Board-Level Tests 17
4.1.1 Relevant Milestones & Target Dates: WIDAR Board-Level Tests 17
4.1.2 Discussion & Timeline: WIDAR Board-Level Tests 18
4.1.3 Task Breakdown: WIDAR Board-Level Tests 18

1 Introduction

The document entitled EVLA Monitor and Control Near-Term Software Development Plan, Version 1.1.0 (available on the Computing Working Documents web page at http://www.aoc.nrao.edu/evla/techdocs/computer/workdocs/index.shtml, as document # 41) presents a software development plan for 1) the transition from the VLA Control System to an EVLA Control System, and 2) support of the WIDAR correlator prototype board tests. This document is a report on the status of the software outlined in that development plan as of the end of Q2 2005.

2 Phases I of the EVLA M&C Transition

The transition from the VLA Control System to the EVLA Control System has been divided into three phases. The work that was to be done in Phase I is outlined below.

2.1 Phase I, EVLA M&C Transition

Phase I of the EVLA M&C Transition concentrates on

- The control of EVLA antennas via the interim Observation Executor
- Initial work on the final version of the Observation Executor
- Synchronization between the VLA and EVLA Control Systems
- Flagging of EVLA antennas
- EVLA Control System access to Antsol solutions, pointing offsets and phasing information (Interim Antsol/Telcal), with reference pointing implemented for EVLA antennas
- Job submission for EVLA Antennas
- Initial work on the control of VLA antennas by the EVLA Control System
- Development of the software needed for EVLA antenna Checkout, Startup and Other Tests that is required even when the Modcomps are still present.
- Bringing the new VLA Correlator Controller to operational status

2.2 Interim Antsol/Telcal

The plan for the interim Antsol/Telcal is to move the Antsol functionality from the Modcomps to a computing platform (igloo) that is now a staging/spooling area for the VLA archive records. Antsol can be adapted to use this real-time stream of archive records as its input. The program that solves for pointing offsets can be similarly adapted. Igloo, the computer that receives the archive record data stream is a Linux-based system that can be made accessible to both the VLA and EVLA Control Systems. Relocating Antsol and pointing offsets calculation to igloo allows the Modcomps and the VLA Control system to be retired without perturbing the EVLA Control System's access to the antenna gains and pointing offsets. The EVLA Control System can operate indefinitely in this configuration, and it probably foreshadows the configuration that will be used for the EVLA Telcal.

3 Phase I, EVLA M&C Transition

3.1 Objective: Use of EVLA Antennas in VLA Observing

The primary objective of Phase I is to complete the software needed to enable the use of EVLA antennas in VLA scientific observations, while laying some of the groundwork needed to prepare for retirement of the Modcomps and the VLA Control system.

3.1.1 Relevant Milestones & Target Dates: Use of EVLA Antennas in VLA Observing

	Target Date as of Mar 25, 2005	Target Date as of Jun 08, 2005
D30x ICD (ver E) ready for software	Apr 14, 2005	Jun 27, 2005
D30x ver E formatter tested	May 05, 2005	
D30x ver E MIB software ready	May 20, 2005	Jul 18, 2005
M301 hardware ready for software	May 27, 2005	Jul 01, 2005
M301 ready to install on antennas	Jun 08, 2005	Jul 15, 2005
Antenna 14, ready to participate in scientific observations	Jun 23, 2005	late July (?)
Antenna 16 ready to participate in scientific observations	Jul 01, 2005	requires review
Antenna 13 ready to participate in scientific observations	Jul 08, 2005	requires review
Antenna 18 ready	Aug 22, 2005	requires review

3.1.2 Timeline: Use of EVLA Antennas in VLA Observing

Q2 2005 (Apr – Jun 2005)

3.1.3 Task Breakdown: Use of EVLA Antennas in VLA Observing

1. Observation Executor, interim version (Barry Clark)
    a. The main focus for the interim version of the Observation Executor for Phase I is the control of EVLA antennas
    b. Scan synchronization mechanism between the VLA Control System and the EVLA Control System
    Status (as of 3/21/05): Done
    c. Support for more of the capabilities specifiable via a VLA Observe file, including:
        i. Reference pointing (requires access to pointing offsets)
        Status (as of 6/8/05): Reference pointing scans are supported, but pointing offsets are not yet available and will not be available any sooner than the end of Aug 2005. The full implementation of a reference pointing capability in the EVLA software is not expected until sometime in September 2005.
        ii. Phased array operation (requires access to phasing information)
        Status (as of 6/8/2005): Phased array operation is not supported and phasing information is not yet available. Full support of autophasing may not be in place until the end of Q3 2005 or mid Q4 2005. The priority currently assigned to autophasing places it

behind the implementation of 1) the round-trip phase correction and 2) reference pointing.

d. Integral obs2script (automatic conversion of (J)Observe files to control scripts)
   Status (as of 6/8/05): There is a problem with large/long scripts. In an attempt to solve this problem, jython was configured to evaluate scripts on a line-by-line basis. This change has led to other problems including problems with loops, and problems within a loop with comments that are not indented. There are problems with scripts that give scan times entirely in durations and do not include a special comment line giving the start date/time of the 1st scan.
e. All devices and subsystems configured by Executor. Device browser not needed to configure EVLA devices for an observation.
   Status (as of 6/8/2005): Neither the deformatter nor the current version of the formatter boards are configured via the Observation Executor. The deformatter board does now present a MIB interface in addition to the original TCL-based interface, but slot ID is not dependable for the deformatter board so the TCL-based interface is used for configuration. The current formatter boards do not implement slot-ID. The Observation Executor will not attempt to configure the current generation of formatter boards. That capability will be deferred until the new version of the formatter boards are in place.
f. Improved feedback to operators on status of script execution
   Status (as of 6/8/05): Logging routines have been added to the Executor that will allow improved feedback to be presented to users via the Observation Executor GUI interface.
g. Add DeviceListener thread to monitor antenna subsystems (see Barry Clark's posting to evla-sw-discuss of 1/5/2002, entitled "Design of the current EVLA real-time system)
   Status (as of 6/8/05): No progress on this item and its currently assigned priority places it behind initial work to control VLA antennas and phased array operation.
h. Update scan parameters supported (equivalent to a skip n 0)
   Status (as of 6/8/05): A set hardware method has been added to the interim Observation Executor that can be used to update parameters at an antenna. However, the Observation Executor interface &/or other operator tools have not yet been extended to provide the ability to invoke this method. There is an additional issue of updating the parameters database and then pulling those updated parameters into a script. Currently, to get updated parameters from the database into a script, one must update the database, stop the script and then restart the script.
i. Script stop supported
   Status (as of 6/8/2005): There are still problems with this function. Basically, a script stop is not honored until there is a source change. For the case of stopping a script during a long scan, the result is very unsatisfactory. A method, that works, for removing antennas from the

control of a script does exist. Currently, the recommended procedure is to remove all antennas from a script's control and then attempt to stop the script.

j. Scan skip & scan extend supported, or equivalent functionality
    i. Skips and extends are difficult to implement. They do not map in a natural fashion to the structure of the Executor.
    Status (as of 6/8/2005): No progress. There is debate on the best approach to take on this item. Some feel that skips and extends must be supported, even if doing so means a rework of the basic framework or structure of the Executor. Others feel that the correct approach is to wait and see if skips and extends are really needed, or if different ways of working that eliminate the need for skips and extends may emerge.

k. Begin work needed to support configuration and control of VLA antennas
    i. Barry Clark has begun work on a loifsetup construct for VLA antennas
    ii. Methods for communicating with VLA antennas (see also the CMP)

    Overall Status (as of 6/8/05): No additional work on a VLA loifsetup in the Observation Executor has been done. A MIB interface utilizing IP aliasing to allow addressing of individual antennas has been implemented on the CMP, allowing communication with VLA antennas to take substantially the same form as communication with EVLA antennas. The hooks for a HTTP interface to the VLA antennas have also been put in place on the CMP, but the actual interface has not yet been implemented. At least one test of the ability to command the ACU of a VLA antenna has been conducted and the test was successful.

l. Begin work needed to support configuration and update of VLA correlator (new VLA Correlator Controller required for testing)
    Status (as of 6/8/2005): No work done is this area. It sits below several other items in Barry's work queue.

m. Items identified for the interim Observation Executor that were not listed in version 1.1.0 of the Near-Term Software Development Plan
    i. Implementation of the round trip phase (RTP) correction
        1. There are design and implementation issues here. Should it be done as a monitor point or by some other route? This question should be settled now even if the RTP correction is not needed until we are ready to send TSYS to the correlator.
        2. This issue will have an impact on MIB software
        3. Barry's initial thinking is that one way of distributing RTP info is via a separate multicast group and that this multicast group should, in some sense, be a part of the ostream.

    Status (as of 6/8/2005): An initial implementation of the RTP correction may be in place by mid Q3 2005. Barry's current priorities are 1) RTP, 2) reference pointing, and 3) autophasing.

2. Observation Executor, final version (Observation Executor team: Butler, Benson, Cai, Moeser, Waters)
    a. Requirements
        i. Note: For testing, the final version of the Executor may need to support scan synchronization with the VLA Control System
        Status (as of 6/10/2005): It is expected that requirements will be circulated for review and comment by 6/30/2005. The team is currently nearing completion of a first draft of the requirements for review and revision by the team prior to distribution of the requirements to team consultants and other individuals.
    b. Detailed Design, including
        i. Specification of interfaces to
            1. Antenna Monitor and Control (AMCS)
            2. Correlator Monitor and Control (CMCS)
            3. Observation Scheduler
            4. Interim Antsol/Telcal if needed
            5. EVLA Telcal
            6. EVLA DCAF

        Status (as of 6/10/2005): Not yet begun. For Phase II, this item should be rephrased to include both the creation of the basic objects and the specification of the interfaces.
    c. Determination of degree of reuse/refactoring of interim Observation Executor code
    Status (as of 6/10/2005): Not yet begun.
3. Antenna field testing (Clark, Sowinski)
    a. Significant amounts of time required from senior personnel also working on EVLA M&C software development
    Status (as of 6/10/2005): Ongoing
4. Flagging for EVLA antennas (Sowinski, Cai)
    a. Development of flags in EVLA Control System from MIB-generated alerts. (Implies flagging for EVLA antennas.)
    b. Transfer of flagging information to VLA Control System
    c. Merger of EVLA flagging with VLA flagging in the VLA Control System
    d. VLA Archive records annotated with EVLA and VLA flagging information by the VLA Control System (the Modcomps)

    Status (as of 3/21/05): Tests have demonstrated that the full path from an alert generated in a MIB to annotation of VLA Archive records written by the VLA Control System is functional. MIB alerts need additional work. The development of flags at a level higher than the MIBs needs work. More testing is needed.

    Status Update (as of 6/10/2005): The framework that was put in place and demonstrated in March 2005 has not been used on a regular basis, and there has been little or no evaluation of the quality of the flags or additional flags that may be needed.
5. Interim Antsol/Telcal (antenna gains, pointing offsets, phasing information, focus, delays) (Sowinski, Cai)

a. 6/10/2005 – While is has always been implicitly understood that the first step for the Interim Antsol/Telcal was to make the complex gains of the antennas, as produced by Antsol, available to the EVLA Monitor and Control System, the SW Development Plan failed to call out that item explicitly. This item has been added to address that omission.
   Status (as of 6/10/2005): A version of Antsol, running on igloo, that works from disk-based records was created & tested. This version was validated as producing correct results. The disk-based version of Antsol was then modified to distribute the Antsol results using an XML message format that is multicast. Building on this work, a version of Antsol, again running on igloo, that works from the real-time data stream was created. This version was also validated as producing correct results. The version of Antsol that works from the real-time data stream must now be modified to include distribution of the results in an XML format via multicast. Ken Sowinski will make these modifications. It is expected that these modifications will be completed sometime in late June or late July 2005.
b. EVLA access to pointing offsets and phasing information is required in this phase of software development.
   Status (as of 6/10/2005): Pointing offsets. A proof-of-concept pointing program that runs on igloo has been created. It produces the correct results. This program must be converted to a production quality implementation and modified to include multicast of the results in an XML format. Chunai Cai is currently working on both the XML format for the results and the routine(s) needed for the multicast. She is also working on client software to receive, parse and display the results. It is expected that the work on pointing offsets will be well in hand if not actually completed by the end of August 2005.
   Phasing information: Getting phasing information to the EVLA Monitor and Control System will follow the same path as the work being done for Antsol. Little or no work has been done in this area as of June 2005.
c. Begin work on focus determinations and the determination of delays by the EVLA Control System
   i. For EVLA antennas
      Status (as of 6/10/2005): The revision of the Phase II task list should break out focus and delays as two separate items. Focus determinations for EVLA antennas are still being made on an adhoc basis and now is not the time to elevate the priority of this matter. There are much more pressing issues. The determination of delays will continue to work properly until the Modcomps are retired. Both items will be moved into Phase II of the SW Development plan.

6. EVLA Telcal (Sowinski, Cai)
   a. For Phase I, the focus will be on Interim Antsol/Telcal. EVLA Telcal will receive little formal attention during this period. However, its needs and design will be kept in mind while work proceeds on the Interim Antsol/Telcal.

Status (as of 6/10/2005): No formal progress on this item was expected or scheduled for Phase I, and no work has been done in this area except to give the matter some thought will working on the Interim Antsol/Telcal.

7. VLA Data Capture and Format (VLA DCAF) (Sowinski, Cai)
    a. VLA archive records still created (formatted) by the VLA Control System (Boss, DUMP)
    b. VLA archive records still written by the VLA Control System (Boss)
    c. Develop a detailed design for migration of VLA DCAF (old DUMP) functionality from the VLA Control System, including task breakdowns and a schedule with milestones
    d. Document the design, task breakdown, and schedule
    e. Begin development
8. VLA Archive (Sowinski, Cai, Benson)
    a. Review the current VLA archive format for decisions and changes that must be made in order to write VLA format archive records in a Modcomp independent format.
        i. Develop a task breakdown for the changes to be made
        ii. Set milestones and schedule
        iii. Begin development
        iv. Document the decisions, task breakdown, and schedule

    Status (as of 3/21/05): Ken Sowinski has written and distributed an email (posted to evla-sw-discuss on 18 Feb 2005 08:43:22, Subject: the transition era VLA/eVLA archive record) that discusses the changes required in the VLA archive format to create a Modcomp independent version that can be written by the EVLA Monitor and Control System.
    b. Coordinate with other groups (AIPS, AIPS++ (?), others?) on changes in post-processing software needed to support changes in VLA archive format records

VLA DCAF & VLA Archive, Status (as of 6/8/2005): A decision has been made that as with the Interim Antsol/Telcal, the VLA DCAF will reside on igloo, and the VLA archive record will be written from igloo. The work needed to formulate and write VLA archive records in a Modcomp independent format is still in the thinking stages. Some discussions concerning an upcoming format change in the VLA archive records have been held with Eric Greisen (AIPS) and Joe Mcmullin (AIPS++). The new VLA archive format must be specified fairly carefully. Most importantly, the differences between the old and new formats must be described precisely and in detail. The conceptual groundwork has been laid for migration of the writing of VLA archive records from the Modcomps. Implementation of this change will not begin until Phase II of the Transition Plan. Getting this task well in hand will be a major goal for Phase II and will continue into Phase III. The new goal/milestone is to have made significant progress on an implementation of the VLA DCAF and the VLA Archive by the end of September 2005 (end of Q3 2005).

9. Operator tools (Moeser, Butler, Sowinski)
    a. Written instructions and help screens
        i. Written cookbook style instructions needed for common tasks

ii. Continue development of help screens
Status for items i & ii (as of 6/8/2005): Ongoing. Several training sessions have been held with the VLA Operators. The VLA Operators are now submitting jobs that use EVLA antennas either alone or in combination with VLA antennas via the Executor (job submission) interface.

b. Job submission
   i. GUI interface to interim Observation Executor
   Status: Functional. Further development expected.

c. Ability to monitor observation progress and status
   i. Specification of what is needed
      1. Status of control script execution for EVLA antennas
      2. Data quality displays that include EVLA antennas
      Status: The F and D10 displays are based on output from the VLA Correlator, and therefore provide feedback on EVLA as well as VLA antennas.

Job Submission, Observation Progress: Status (as of 6/8/2005):
Considerable progress has been made on the Observation Executor Interface. While the current Executor interface is web-based, a new set of requirements for the Executor Interface that are substantially more complex and include the need for higher refresh rates for some of the information has resulted in the development of a Java application to replace or serve as an alternative to the web-based interface. The Java application will include all of the functionality of the web-based interface plus

- considerably enhanced capabilities for finding/loading the control script to be run
- a formal, defined directory structure specifying where control scripts may reside
- a queue of control scripts to be run (sequentially) with the ability to manipulate the queue
- an array/subarray screen that present information such as time (multiple formats), the source name, ra, dec, az, and el of the current source, pointing errors, frequency information, and subreflector rotation and focus positions
- a "console" used to display informational messages coming from the actual Executor

It is expected that this new Executor Interface will be ready for field-testing by the end of Q2 2005 (end of June 2005). Substantial portions of the above listed functionality have already been demonstrated.
No work has been done on developing data quality displays for the EVLA. None is planned for the near future. The data quality displays require access to Antsol results, which is an item that is not yet complete, and the current Modcomp-based displays support EVLA as well as VLA antennas.

d. Ability to monitor antenna status, EVLA antennas
   i. EVLA hosted Checker function for EVLA antennas

Status (as of 3/21/05): Two Checker screens have been demonstrated. One captures current alert-on messages. The other is a history screen that provides a log of alert-on messages for which a matching alert-off message has been found. Work may be needed on prioritization or other indicators of severity, color-coding, etc.

Status update as of 6/8/2005: Apart from an update to the Checker software for compatibility with changes to the message format and some additional safeguards against abnormal termination due to unexpected message formats, no additional development of this software has taken place. Checker will undoubtedly see further development during Phases II & III of the Transition Plan. As the requirements for Checker develop, it is possible that a Java application may be developed as either a replacement for or as an alternative to the current web-based approach.

ii. Operator Screens
    1. Framework for general approach
    2. Specification of screens to be developed
    3. Prototypes for some significant subset of the screens
        a. Ability to monitor and control MIB-connected antenna devices (EVLA antennas)

Status (as of 6/8/2005): Rich Moeser is working with the operators and technicians to help develop their requirements. Samples of what is possible and can be done in the context of web-based interfaces and Java applications have been developed and given to the VLA Operators. A prototype, web-based antenna screen exists. Prototype, web-based screens for an ACU display and a L301-1 display exist. A general framework for Operator Screens has not yet been developed. The fundamental issue of the best method for getting data to Operator Screens has not been resolved.

e. Specification and documentation of the general approach to interfaces for non MIB-connected processes
    i. Executor, Checker, Flagger, others?
    ii. At least prototype GUIs for these processes

Status (as of 3/21/05): A general approach to these interfaces has been developed. It is web-based. GUIs for the Executor, Checker, and Flagger have been demonstrated. Documentation of the approach has been written, but is now somewhat obsolete and needs to be updated. The documentation has not yet been made publicly available. Much more field-testing of this approach to the interfaces is needed to be certain of its viability.

Status Update (as of 6/8/2005): A general approach to these interfaces has been developed and is in place. This approach is two-fold. For web-based interfaces it consists of the use of a Java servlet container such as Jetty or

Tomcat plus the development of a servlet and an interface class. For both web-based interfaces and Java applications, several Java classes supporting a variety of communication methods have been developed. These classes include the evla-client class and the module-client class. Tests and benchmarks of different approaches to the issue of XML parsing were developed. The two approaches under consideration are 1) the use of CASTOR to convert XML into Java structures that can be queried, and 2) the use of the document object model (DOM) with XPATH queries.

10. Device Browser (Moeser)
    a. Ability to access all attributes of all monitor and control points for MIB-connected devices (I.e., EVLA antennas)
    b. Ability to set all writeable attributes of MIB-connected devices (I.e., EVLA antennas)
    c. Annotation of plot scales
    d. Ability to sample monitor points at high sample rates and save data to local computer

    Status (as of 6/8/2005): Work on the device browser is scheduled to resume on 6/20/2005. It is likely that items a. & b. will be completed by 6/30/2005. It is possible that item c will also be completed by that date. Item d, as a generic capability of the device browser, may prove to be unnecessary. The tendency has been to develop special purpose, MIB-based data dump routines tailored to the needs of specific modules. As items a. through c. are developed, a review of any further capabilities needed in the device browser should be conducted.
11. Monitor data archive (Moeser, Benson)
    a. Monitor data accessible using fully qualified, fully descriptive device names that accurately reference the data source
    b. Ability to search for & list multiple monitor points with one query
    c. Ability to search and plot multiple monitor points with one query (?)
    d. 6/14/2005 An item omitted from the Phase I task list for the Monitor data archive is the development of the monitor data archiving task (Monarch) and its interface. Monarch's function is to catch all monitor data archive multicast messages, unpack and buffer the messages, and to then forward the buffer contents to the Oracle database that is used as the monitor data archive. Monarch has been in place and functioning since the early stages of Phase I of the transition plan. It has proven to be robust and reliable. A very useful, web-based interface to Monarch also exists.
12. MIB software (Whiteis, Ben Frej)
    a. Module software as hardware becomes available
        i. T304/T305

        Status (as of 6/8/2005): Done, in the sense that the software is fully up-to-date with the current version of the hardware, with the exception of the implementation of Synch Detection.
        ii. L302 mods

        Status (as of 6/8/2005): Done
        iii. D30x

Status (as of 6/8/2005): Awaiting hardware. Now due on 6/27/2005.

iv. M301
Status (as of 6/8/2005): Awaiting hardware.

v. F320
Status (as of 6/8/2005): Awaiting hardware. Now due on 6/15/2005.

vi. F317
Status (as of 6/8/2005): Awaiting hardware. No due date available. As of 6/13/2005, the F317 is to be delayed by as much as 6 months.

b. Get *.*.* disallowed as a MIB command
   i. Implementation
   ii. Update documentation (Service Port ICD)
   Status (as of 6/8/2005): The implementation is complete. The Service Port ICD has not been updated.

c. Addition of alert status in response to get commands for monitor point values
   i. Implementation
   Status (as of 3/21/05): Implemented in release 0.20 of the MIB framework software (3/2005). Currently undergoing testing.
   ii. Update documentation (Service Port ICD)
   Status Update (as of 6/8/2005): Implementation complete. The Service port ICD has not been updated.

d. Addition of alert status to data port multicasts of monitor point values
   i. Implementation
   Status (as of 3/21/05): Implemented in release 0.20 of the MIB framework software (3/2005). Currently undergoing testing.
   ii. Update documentation (Data Port ICD)
   Status Update (as of 6/8/2005): Implementation complete. The Data Port ICD has not been updated.

e. "In Use" flag
   i. Initial implementation
   ii. Update documentation (Framework Software)
   Status (as of 6/8/2005): Not yet started. Will move into Phase II.

f. Upgrade to latest version of Nucleus OS
   i. Fix TCP_Retransmit bug
   ii. Make transition to latest version of Nucleus OS
   iii. Update old images for MIBs already in the field
   Status (as of 6/8/2005): A patch has been made to fix the TCP_Retransmit bug, and the latest version of the Nucleus OS is now the basis for all new MIB software development. Updates of the older images already in the field is happening as opportunities present themselves, but a formal, concerted effort to update all old images has not been undertaken.

g. Decision on upgrade to latest version of toolset

i. Begin exploring a GNU-based toolchain ?
Status (as of 6/8/2005): A decision has been made to explore the possibility of going directly to a GNU-based toolchain, skipping the upgrade to the latest version of the Altium TASKING toolset now used for MIB software development. GNU tools are being explored, but the level of effort is low.

13. Control and Monitor Processor (CMP) (Ben Frej)
    a. VLA monitor data available in EVLA format
    b. VLA monitor data to EVLA monitor data archive
    c. Implementation of a command path to VLA antennas
        i. Low-level implementation within the CMP
        ii. Public interface to command path for use by external processes (such as the Observation Executor)

    Overall Status (as of 6/8/2005): Done. VLA monitor data is now available in EVLA format and is being archived. One operational test of the full command path to VLA antennas has been made. The test consisted of an attempt to send a VLA antenna to a commanded position. The test was successful. The CMP should achieve full operational status early in Q3 2005.
14. New VLA Correlator Controller (Sowinski, Rowen)
    a. Fully operational by the end of Q2 2005, controlled by Modcomp hosted software
    Status (as of 6/8/2005): The goal of attaining operational status, under control of the Modcomps by the end of Q2 2005 was not met, in large part due to delays in hardware development. The new target date is the end of Q3 2005, i.e., by the end of September 2005. Hardware development may continue to be a factor.
15. Software support for Hardware Quality Assurance
    a. GUI interface to interim Observation Executor
    Status (as of 3/21/05): GUI is functional. Further development expected.
    b. Appropriate test scripts available
    Status: Done
    c. User training
    Status: Done
16. Software support for Antenna Checkout/Acceptance, Startup After Maintenance Day, Other Tests (See "VLA Antenna Checkout for Operations, EVLA Notes/Questions", by Pat Van Buskirk, 1/12/2005, and an email from Ken Sowinski, 15 Jan 2005 16:42:15, Subject: Transition Software list) (Sowinski, others – TBD)
    a. Items needed while VLA Control System (Modcomps) are still present
        i. Verify network connectivity to antenna
        ii. Round Trip Phase. Ability to verify that the antenna is synchronized to the 10 sec tick.
        iii. Ability to verify:

1. Control of feed heaters (initially by the F320 module, ultimately by the M302 utility module). Feed heaters to be available sometime in March 2005.
2. Data Sets. Proper operation of backend filters (dataset 5)
3. Focus/Rotation. Successful setup at all bands.
4. LO. Scripts needed for checkout of L301 & L302 settings at all bands
5. Cryo. F317 & F320. Verify that cryo temp < 20K, vacuum < 1u. Use device browser, perhaps with a special screen?
6. Front End (F317 & F320) default settings for all bands.

iv. 600 MHZ round trip phase. EVLA is optical. What needs to be checked and how will we do it?

v. Delays. Verify delays found for all IFs at all bands. No change with Modcomps and VLA Correlator, but Phase I is the right time to begin thinking about what we will do when the Modcomps are retired.

vi. Pointing1. Set first tilt terms & A7. Requires an interface to the parameters database. No other changes until WIDAR correlator.
Status (as of 6/8/2005): Chunai Cai has been assigned the task of developing a special purpose interface to be used for updating the pointing coefficients in the parameters database. It is expected that this task will be completed sometime during Phase II of the Transition Plan.

vii. Focus, recommissioned antennas only. Verify focus found for all bands. New design and software needed.

viii. L Band, recommissioned antennas only. Check crossed-hand polarizations. Available via D10 display or data analysis via AIPS. Should be changed to check all receivers.

ix. Baselines, install after valid baseline run/analysis. Requires an interface to the parameters database.
Status (as of 6/8/2005): Chunai Cai has been assigned the task of understanding the output of the AIPS locate routine, parsing that output, and creating a means by which the baseline parameters will be updated in the parameters database.

x. Pointing2, install updated coefficients after valid pointing run/analysis. Requires both the creation of an input file for the PEEK analysis program and an interface to the parameters database.
Status (as of 6/8/2005): Chunai Cai is working on the creation of the input file for PEEK and has been assigned the task of developing a special purpose interface to be used for updating the pointing coefficients in the parameters database.

xi. Tipping. New software needed for EVLA antennas.

xii. Systest. Needs requirements and definition. Total Power and Synch Detector voltages must be monitored.

17. Software needed to support Scientific Commissioning

a. It is not clear that any software beyond that already needed to integrate EVLA antennas into VLA observing will be needed.

18. Other (Sahr)
    a. Update and revise EVLA M&C Software Development Plan
       Status (as of 6/8/2005): Version 1.1.0 of the Near-Term Software Development Plan in combination with this status review will serve as the basis for the next version of the software development plan. Work on the next version of the software development plan will begin in July 2005.
    b. Create an EVLA M&C Design document
       Status (as of 6/8/2005): Begun, but seriously behind schedule.
19. New items identified over the course of software development for Phase I of the Transition Plan.
    a. There is a need to develop a system for software updates, i.e., a software release system that will present stable, known versions of the software to the operators while still permitting software engineers and expert observers to test and work with software versions under development. This release system should include a comprehensive review of the software development and execution environment. For example, the launch script for the interim Observation Executor probably still references a Java environment and application JAR files that must be loaded from the AOC. Several different Java environments are available on mchost at the site, and it is possible to instruct Maven to deploy JAR files directly to mchost. If these elements of a solution are not satisfactory then more attention must be paid to the plans to place a mirror of some significant portion of the AOC file system at the site (end of Aug 2005 ?).
    b. Pruning the monitor database archive
        i. Barry Clark has prepared a program to prune the monitor data archive, but there are some organizational issues that must be addressed. As of 6/15/2005 the pruning program is running as a cron job.
    c. Maven. Intervention as required. Maven provides a build environment and web deployment of various software metrics. The EVLA M&C software performs nightly builds. Errors encountered during the build require intervention by a software engineer. Fuller deployment of Maven features also requires work & time. Work is needed to configure Maven to work correctly with Java 1.5. Rich Moeser handles Maven issues.
    d. Support, as needed, for the LabView interface to MIBs. This interface allows engineers and technicians to develop LabView screens that can receive monitor data from and send commands to MIB-based devices.
20. Issues raised by the software developers
    a. Mchost is experiencing a fault condition that manifests itself as an inability to correctly execute Java. When this fault condition is active, mchost seems to view one or more JAR files as corrupted, when, in fact, they are not. James Robnett has suggested that this condition may be occurring because the JAR files are being changed while they are in use. Barry Clark feels his unsuccessful attempts to use JAR itself to reference

the files while this fault condition is manifest contradicts this possible explanation.

b. Will Oracle be up to the task of archiving monitor data as the number of antennas increases? If it begins to lose data will it alert us to his fact or be silent about the data loss?

4 WIDAR Correlator

4.1 Objective: WIDAR Board-Level Tests

4.1.1 Relevant Milestones & Target Dates: WIDAR Board-Level Tests

Gbit Transmission Test	Target Date as of 3/25/2005	Target Date as as of 6/8/2005
Gbit Test Plan document available (A formal test plan will not be written)	Not Applicable	
Hardware testbed ready	Not Applicable	
Bruce Rowen at Penticton for 2 weeks (depends on parts acquisition problem)	mid-May 2005	done
Software installed on testbed and ready for use	mid-May 2005	done
Finished assemblies available for testing	awaiting parts	wk of 6/20/05
Start of tests	mid-May 2005	end of 6/2005
Duration of tests	approx 1 week	
End of tests	end of May 2005	early Jul 2005
Timecode Generator Board (TCGB) Tests		
Timecode Generator Board test plan available	Apr 04, 2005	
Hardware testbed ready (dependent upon parts deliveries)	mid-May 2005	
Software installed on testbed & ready for use	mid-May 2005	done
Finished assemblies available for testing	mid-May 2005	wk of 6/20/05
Start of tests	mid-May 2005	end of 6/2005
Duration of tests	a few weeks	
End of tests	end of May 2005 or mid-Jun 2005	mid or late Jul 2005
Station Board Prototype Tests		
Station Board Test Plan document available	Apr 6, 2005	
1st prototype board available for testing	Aug 31, 2005	Oct 14,2005
Testbed ready, hardware & software	Aug 31, 2005	
Start of tests	Sep 01, 2005	
Duration of tests	6 to 9 months	
End of test	Feb 2006 to May 2006	

Baseline Board Prototype Tests		
Baseline Board Test Plan document available	Mar 17, 2005	
Prototype CBE ready, hardware & software (Tom Morgan at Penticton)	mid-May 2005	
1st prototype board available for testing	Aug 31, 2005	Oct 14,2005
Testbed ready, hardware & software	Aug 31, 2005	
Start of tests	Sep 01, 2005	
Duration of test	6 to 9 months	
End of tests	Feb 2006 to May 2006	

4.1.2 Discussion & Timeline: WIDAR Board-Level Tests

For all of the tests, the test software consists of, essentially, three components. First, one or more module access handlers (MAHs) are required. An MAH consists of a driver for a module plus a layer that sits above the driver that decodes XML packaged parameters for communication to the driver and encodes into XML quantities coming from the driver. Most of the boards consist of multiple modules, so multiple MAHs are needed to fully test a board. The second component is the means by which the MAH and the test applications will communicate. The third software component is the test applications. Test applications consist of test application logic plus GUIs.

The number of GUI screens needed is large and some of the screens are quite complex. The Gbit Transmission test requires only a minimal hardware testbed and a Raw Register read/write GUI. The Timecode Generator Board test plan calls for only 3 GUIs. The Station Board test plan calls for over 30 GUIs, however many of them are quite similar to one another. The Baseline Board test plan specifies approximately 12 GUIs. The highest level of complexity per screen seems to lie with the Baseline Board GUIs.

4.1.3 Task Breakdown: WIDAR Board-Level Tests

1. CMIB <-> test application/GUI communications software
    a. Due: Apr 29, 2005 Developers: Bruce Rowen & Kevin Ryan
       Status (as of 6/8/2005): Done, on time
2. Specification of GUI framework technology
    a. Due: Apr 29, 2005 Developer: Kevin Ryan
       Status (as of 6/8/2005): Done, on time
3. Sample common look & feel for GUIs
    a. Due: May 6, 2005 Developer: Kevin Ryan
       Status (as of 6/8/2005): Done, on time
4. Gbit Transmission test software
    a. Software to load personality file into FPGA
        1. Due: late Jun 2005 Developer: Bruce Rowen
           Status (as of 6/8/2005): Done. Need hardware to test.
    b. Raw Register Read/Write GUI
        1. Due: late Jun 2005 Developer: Kevin Ryan
           Status as of (6/8/2005): Done, to the degree that it can be tested in the absence of hardware.

5. Timecode Generator Board test software
    a. MAH
        1. Due: late Jun 2005 Developer: Bruce Rowen
           Status (as of 6/8/2005): Done. Need hardware to test.
    b. GUIs. Since only 3 GUIs have been specified for the Timecode Generator Board tests, they will be listed here.
        1. Board Top-Level GUI
            a. Due: late Jun 2005 Developer: K. Ryan
               Status (as of 6/16/2005): This screen is responsible for the loading of the personality file into the FPGA. It has been agreed that his function will be handled from a command line interface (CLI) until the MAH can be enhanced to support this functionality in a GUI. The CLI-based capability is in place and ready.
        2. Timecode Generator FPGA GUI
            a. Due: late Jun 2005 Developer: K. Ryan
               Status (as of 6/16/2005): Done, tested to the degree possible in the absence of the actual hardware.
        3. Raw Register read/write GUI
            a. Due: late Jun 2005 Developer: K. Ryan
               Status (as of 6/14/2005): Will use the same Raw Register Read/Write GUI as the Gbit Transmission Test.
6. Station Board Prototype Test Software
    a. Input MAH
       Due: Oct 15, 2005 Developer: Rowen
       Status (as of 6/8/2005): Nearly complete
    b. Delay MAH
       Due: Oct 15, 2005 Developer: Rowen
       Status (as of 6/8/2005): Nearly complete
    c. Autocorrelation MAH
       Due: Oct 15, 2005 Developer: Rowen
       Status (as of 6/8/2005): Nearly complete
    d. FIR Filter MAH
       Due: Oct 15, 2005 Developer: Rowen
       Status (as of Apr 8, 2005): ~ 90% complete
       Status Update (as of 6/8/2005): Reformulated, with numerous changes. New version is ~ 50% complete.
    e. Output MAH
       Due: Oct 15, 2005 Developer: Rowen
       Status (as of Apr 8, 2005): not yet started, awaiting RFS document
       Status (as of 6/8/2005): RFS now available, MAH just started
    f. Timing MAH
       Due: Oct 15, 2005 Developer: Rowen
       Status (as of Apr 8, 2005): ~ 90% complete
       Status Update (as of 6/8/2005): Reformulated, with numerous changes. New version has just been started.

g. VSI MAH
   Due: Oct 15, 2005 Developer: Rowen
   Status (as of Apr 8, 2005): not yet started, awaiting RFS document
   Status Update (as of 6/8/2005): This item is the VLBA interface. As such it is really not funded for Phase I of the EVLA. The only plan for Phase I of the EVLA is to leave a blank footprint on the Station Board. It is unlikely that there will be any hardware or software development of this item during Phase I of the EVLA.

h. GUIs. Some minimal prioritization of the GUIs has been specified. Of the 30+ screens specified in the TVP, those needed soonest will be:
   1. The CRC GUIs. The CRC GUIs are a group of 11 screens that display accumulated CRC errors for the inputs and outputs of the Input Chip, the Delay modules, the Autocorrelation Chip, the Filter Banks, and the inputs of the Output and Timing Chips.
   2. A GUI or GUIs to display internal FPGA errors
   3. A GUI to display Input Chip state count histograms
   4. A GUI to display Filter Chip state count histograms

   Due: Oct 15, 2005 Developers: K. Ryan (Socorro), Sonja Vrcic (Penticton), plus a third individual at Penticton
   Status (as of 6/8/2005): Awaiting updated GUI screen definitions due from Dave Fort. The plan is to handoff a framework for GUI screens to software engineers at Penticton at the end of the WIDAR PDR (7/11/2005 – 7/13/2005). The development of these screens will then be undertaken at Penticton while K. Ryan works on the Baseline Board GUIs in Socorro.

7. Baseline Board Prototype Test Software
   a. Correlator Chip MAH
      Due: Oct 15, 2005 Developer: Rowen
      Status (as of Dec 13, 2004): > 90% complete
      Status Update (as of 6/8/2005): Some reformulation & changes. Now approximately 98% complete.
   b. Recirculation Controller MAH
      Due Date: Oct 15, 2004 Developer: Rowen
      Status (as of Dec 13, 2004): > 80% complete
      Status Update (as of 6/8/2005): Some reformulation & changes. Now approximately 98% complete.
   c. Ethernet Transmitter MAH
      Due Date: Oct 15, 2005 Developer: Rowen
      Status (as of Dec 13, 2004): > 65% compete
      Status Update (as of 6/8/2005): Some reformulation & changes. Now approximately 98% complete.
   d. LTA Controller MAH
      Due Date: Oct 15, 2005 Developer: Rowen
      Status (as of Dec 13, 2004): > 80% complete
      Status Update (as of 6/8/2005): Some reformulation & changes. Now approximately 98% complete.
   e. MCB Interface & Clock Selector MAH

Due Date: Oct 15, 2005 Developer: Rowen
Status: Likely that no software will be required.

f. GUIs. As with the Station Board, the GUIs will not be specified in full in this document. Some minimal prioritization has been specified. In order of need:
    1. The Recirculation Controller GUI, in full
       Due: Oct 15, 2005 Developer: K. Ryan
       Status (as of 6/14/2005): Approximately 60 % complete
    2. The Correlator Chip GUI
       Due: Oct 15, 2005 Developer: K. Ryan
       Status (as of 6/14/2005): started
    3. For the LTA, a Raw Register display with read/write capability would suffice.
       Due: Oct 15, 2005 Developer: K. Ryan
       Status (as of 6/14/2005): Will use the same Raw Register Read/Write GUI as the Gbit Transmission Test.
    4. Approximately 9 additional GUIs, described in the Baseline Board Test & Verification Plan
       Due: Oct 15, 2005 Developer: K. Ryan
       Status (as of 6/14/2005): Not yet started.
g. Additional items not called out in the Near-Term Software Development Plan
    1. CMIB server at DRAO. Basically, this item creates a means by which CMIBs on test boards can download the CMIB OS and other files and applications.
       Status (as of 6/8/2005): Done, tested. All seems OK.
    2. CMIB server at Socorro.
       Status (as of 6/8/2005): A crude, but serviceable version has existed for many months. A refined version with enhanced capabilities is needed. The sysadmin group at Socorro is aware of the need, but further discussion as to configuration and capabilities is needed. Further action on this item has been deferred until sometime after the WIDAR PDR (7/11-7/13/2005).
    3. PCMC board tests. The PCMC (PC/104 Monitor/Control Mezzanine Card) board is an interface board that allows the PC/104 Plus CMIB to communicate with modules on the WIDAR Correlator boards.
       Status (as of 6/8/2005): A PCMC board was available during Bruce Rowen's latest trip to Penticton. It was possible to test the ability of the CMIB and PCMC to communicate, and to read the interface to the monitor/control bus (MCB) that is to be present on the correlator boards. All of these tests were successful. Additional hardware, not yet available, is needed to test read/write access to modules on the correlator boards.

4. Access to CMIBs in Penticton, B.C. from Socorro. NM
   Status (as of 6/15/2005): Done.
5. Correlator Backend (CBE) software installation at Penticton to support testing of prototype correlator boards. (T. Morgan)
   a. Installation of "correlator simulator" software to produce simulated correlator frames. To be installed on Penticton server.
      Status (as of 6/8/2005): Done
   b. Installation of CBE input task to receive lag frames from prototype baseline board. To be installed on Penticton server.
      Status (as of 6/8/2005): Done
   c. Installation of CBE data processing task on Penticton server.
      Status (as of 6/8/2005): Done
   d. Installation of command line interface, control task, and monitor task on Tom Morgan's laptop.
      Status (as of 6/8/2005): Done
   e. Tests of the ability of sever & laptop installed tasks to intercommunicate with one another as needed.
      Status (as of 6/8/2005): Done