E U R O P E A N   S O U T H E R N   O B S E R V A T O R Y

Organisation Européenne pour des Recherches Astronomiques dans l'Hémisphère Austral

Europäische Organisation für astronomische Forschung in der südlichen Hemisphäre

VERY LARGE TELESCOPE

VLT

Astronomical Site Monitor

Control Software

Functional Specification

Doc.No. VLT-SPE-ESO-17441-1175

Issue 1.0

Date 25.03.1997

Prepared S.Sandrock ....
Name Date Signature

Approved M.Sarazin, K.Wirenstrand ....
Name Date Signature

Released G.Raffi ....
Name Date Signature

Change Record

Issue/Rev.	Date	Section/Page affected	Reason/Initiation/Document/Remarks
1.0/prep.1	24.07.1996		Draft for review
1.0/prep.2	25.11.1996		Included changes after review.
		1	Added points concerning archive/INS.
		2.2	Extended descr. which data are provided; historical data only by VLT archive, not ASM!
		8.1.3, 8.1.4	Updated enclosure part for serial line.
		2.3	Revised and added diagrams.
		7	Added sections to cover *all* HW units.
		7.1	**Enclosure requirements entirely changed.**
		7.2	Merged section about encoder and amplifier.
		3.1	Dropped history data and LS-compatible file format, introduced FITS to DMD archive instead.
		3.2	Added *asmarc* for DMD archive interface.
		3.3.1	Added asmControl process.
		3.3.2	Added common monitor process.
		3.3.5	Added *VERSION*, *TEST*, *BREAK*, *KILL*.
		3.4	Revised enclosure specification.
		3.5	**Telescope control revised for TCS modules.**
		3.6	Extended guide-star-list functionality.
		3.8	DIMM control updated.
		3.9	Meteo updated.
		3.16	**Added on-line archive interface.**
		4	**Interface chapter largely revised.**
		6	Planning included.
1.0/prep.3	17.12.1996	All	Term ‘particle sensor’ changed to ‘dust monitor’.
		2.3.3	DIMM calculations split on LCU and WS.
		2.3.6	Extended DIMM diagram (quality-control data).
		3.2	Merged *asmws* and *asmlcu* to *asmi* module.
		3.5	Replaced *limit*-swtiches by *init*-switches.
		6.4	Portable X-terminal shall be availabe for test.
1.0	25.03.1997	2.1	New introductory section.
		3.3.1	Updated state diagram of *asmControl* process.
		3.5	Added figure with coordinate definitions.
		3.8	Updated table of DIMM provided data.
		7	Moved hw requirements chapter, unchanged.
		8.1	Moved hardware overview to appendix.

TABLE OF CONTENTS

1 INTRODUCTION

The **Astronomical Site Monitor (ASM)** of the VLT works independently from the unit telescopes. It provides TCS and other external packages with information about the current meteorological data, outside air, seeing, cloud coverage etc. It also produces the data to be archived together with scientific and other data sets in the VLT on-line archive.

The hardware of the ASM consists of the Differential Image Motion Monitor (**DIMM**, a 35-cm telescope with enclosure, technical CCD detector), and several external sensor facilities like a meteorological station.

1.1 Purpose

The present document contains the functional specification, the attributes, and the design constraints of the workstation and LCU control software included in the ASM subsystem. It shall provide the necessary information for the detailed design and implementation.

The ASM functionality described here shall fulfill the requirements listed in the TCS Functional Specification [1] and those in the DIMM Upgrade Plan [2].

Intended readers are those designing and implementing the ASM control SW, as well as developers of SW that will access or use ASM data, i.e. TCS, VLT on-line archive, and instrumentation.

1.2 Scope

The TCS Functional Specification [1] contains the basic requirements for the ASM from TCS-side. The DIMM Upgrade Plan [2] contains the requirements concerning external interfaces in general, and for the DIMM instrument in particular. Knowledge of both documents is recommended.

While the term *ASM* means *all* facilities, the *DIMM* includes only the telescope with connected CCD-equipment, i.e. not the meteo-station or other external sensor equipment. In this document the term *DIMM* is normally not used to denote a group of facilities.

This document describes the functionality foreseen to control the ASM facilities, record data in an on-line database, and execute information requests for current data from TCS etc.

This includes:

- real-time control of ASM-telescope, sensors, and instruments
- scheduling and coordination of instruments in order to query and calculate the required data
- on-line database recording of the required data
- public on-line database interface to TCS etc.
- interface to VLT on-line archive
- user-interfaces for automated and manual operation of the ASM

This does ***not*** include:

- definition of a long-term database for the recorded data
- display and statistical evaluation of the recorded scientific data
- public database interfaces (i.e. through ftp or WWW) to the recorded data or statistics
- transfer routines to the VLT archive

  These points will be provided by DMD and INS [13], respectively handled by DMD based on the archived data. [14]
- specification of additional data-extensions, e.g. for the planned data modelling or prediction unit.

This will be described in separate document(s) TBD.

1.3 Applicable Documents

The following documents, of the exact issue shown, form a part of this document to the extent specified herein. In the event of conflict between the documents referenced herein and the contents of this document, the contents of this document shall be considered as a superseding requirement.

[1] Telescope Control System Functional Specification,	VLT-SPE-ESO-11720-0001, 1.1, 02/03/95
[2] DIMM Upgrade Plan,	VLT-SPE-ESO-17410-1174, 2.0, 19/12/96
[3] LCU Software Specification,	VLT-SPE-ESO-17210-0002, 2.2, 26/10/93
[4] VLT Software Requirements Specification,	VLT-SPE-ESO-10000-0011, 2.0, 30/09/92
[5] VLT Software Management Plan,	VLT-PLA-ESO-00000-0006, 2.0, 21/05/92
[6] VLT Software Programming Standards,	VLT-PRO-ESO-10000-0228, 1.0, 10/03/93

1.4 Reference Documents

The following documents are referenced in this document.

[7] Seeing Monitor User's Manual, ADES SA, 22 mars 1988	
[8] LCU Common Software User Manual,	VLT-MAN-SBI-17210-0001, 3.3, 30/10/96
[9] CCS User Manual,	VLT-MAN-ESO-17210-0619, 1.6, 31/10/96
[10]CCS On Line Database Loader User Manual,	VLT-MAN-ESO-17210-0707, 1.4, 30/06/96
[11]CCD Detectors Control SW Functional Spec.,	VLT-SPE-ESO-17240-0227, 1.0, 08/04/93
[12]Technical CCD Systems User Manual,	VLT-MAN-DJO-11700-0001, 1.0, 06/05/96
[13]ICD between INS and VLT Archive System,	VLT-ICD-ESO-17240-19400 ***(in preparation)***
[14]VLT On-line Archive Facility User Requirements,	VLT-SPE-ESO-19400-1162 ***(in preparation)***
[15]ASM Hardware Design Description	VLT-TRE-ESO-17430-1271 ***(in preparation)***
[16]TCS Axis Control Module Design Description,	VLT-SPE-ESO-17130-1210 (in preparation)
[17]TCS Tracking Design Description,	VLT-SPE-ESO-17230-0755, 1.0
[18]DIMM Retractable Enclosure Functional Specification	VLT-SPE-ESO-17410-1253, 1.0, 21.11.1996

1.5 Abbreviations And Acronyms

The following abbreviations and acronyms are used in this document:

ACE	Array Control Electronics (for CCD Cameras)
ADC	Analog to Digital Converter
ASM	Astronomical Site Monitor
CCD	Charge Coupled Device
CCS	Central Control Software
CDT	Command Definition Table
CI	Command Interpreter
CIT	Command Interpreter Table (for LCU CI)
CPU	Central Processing Unit
DAC	Digital to Analog Converter

DB	Database
DIMM	Differential Image Motion Monitor
DIN	Deutsches Institut für Normung
DMD	Data Management Division
FTP	File Transfer Protocol
GUI	Graphical User Interface
HW	Hardware
ICD	Interface Control Document
INS	Instrument Common Software
I/F	Interface
I/O	Input/Output
ISO	International Organization for Standardization
LAN	Local Area Network
LCC	LCU Common Software
LCU	Local Control Unit
N/A	Not Applicable
RTAP	Real Time Application Platform
SW	Software
TBC	To Be Confirmed/Clarified
TBD	To Be Defined
TCCD	Technical CCD
TCS	Telescope Control Software
TIM	Time Interface Module
TRS	Time Reference System
VLT	Very Large Telescope
VME	Versa Module Eurocard (Bus-System)
WS	Workstation
WWW	World Wide Web

1.6 Stylistic Conventions

The following styles are used:

bold
: in the text, for commands, filenames, prefixes/suffixes as they have to be typed.

italic
: in the text, for parts that have to be substituted with the real content before typing.

`teletype`
: for examples.

`<name>`
: in the examples, for parts that have to be substituted with the real content before typing.

bold and *italic* are also used to highlight words.

1.7 Terminology

assert - a signal is set to its active state

negate - a signal is cleared to its inactive state

axes coordinates - position in an horizontal coordinate system denoted by angles suitable for the telescope mounting

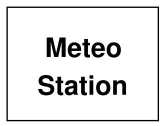
target coordinates - position of an object in an equatorial coordinate system

1.8 Diagram Conventions

Control/data flow diagrams show the relations of a system with the environment, not in terms of physical location, but in an abstract form. The following symbolic notation is used for the elements of a diagram:

Software Data Process:

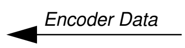
External Unit (HW or SW):



Database or File Unit:

Control Flow:

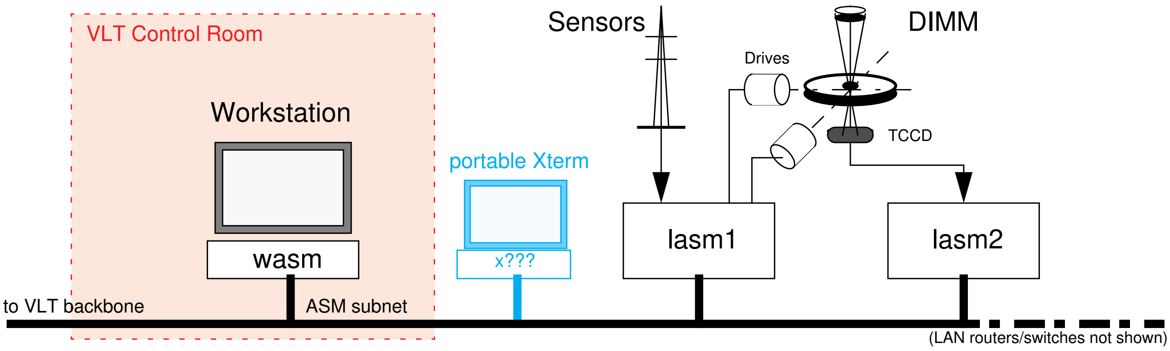
Data Flow:



2 OVERVIEW

2.1 ASM Facilities

The Astronomical Site Monitor (ASM) provides information about weather conditions and other environment parameters. Particularly TCS functions need data provided by this system, as input for on-line calculations and for information to the users. The ASM will also provide additional data which might be useful for other packages and an operator. All data provided by the ASM are stored in an RTAP on-line database, from which they are available via CCS database interface.



The ASM is a distributed system consisting of:

- A workstation. All user requests for data shall be directed to this unit.
- The DIMM with a telescope and CCD detector for seeing measurement.
- Several external sensor units, incl. meteorological station.
- Two LCUs (possibly more) controlling the DIMM and external sensor units.
- It shall be possible to connect a portable X-terminal close to the LCUs for test and maintenance.

2.2 ASM Provided Data

The site monitor data are based on cyclical reading of a variety of sensors and measuring equipment and calculation of derived data. This includes:

- Seeing, scintillation, cloud coverage
- Meteorological data (outside temperature, humidity, pressure, wind conditions)
- All-sky image

The ASM database will contain and provide the latest value of all measured variables, statistical data for some variables and also some calculated variables. The statistical data (*"short-term history"*) are averages and RMS values over fixed time periods (not more than one hour). There will not be any statistical calculations over time periods defined by external events, like e.g. 'Exposure start/stop'. If an external package wants such calculations, it must itself read on events, do the scanning in between and do the calculations, as appropriate.

The data produced by the ASM will be saved periodically (typically every minute) into the VLT on-line archive, tagged with FITS keywords, by means of a dedicated client process [13]. The ASM keeps historical data only for internal purposes, e.g. for DIMM quality-control data and the planned prediction unit. Publicly available from the ASM database are only the ***current*** data and statistical values, while for historical data - e.g. all seeing measurements for the actual night - one has to query the VLT on-line archive [14]. (An Rtap-based history-table access may be added also for public use, but is not planned.)

Note that the ASM is a stand-alone system and ***not assigned to any unit telescope***. Data which are specific to UT 1-4 - e.g. dome temperature in UT 1 - are not available from the ASM.

2.3 Software Overview

2.3.1 Software Standards

Standard CCS-LCC mechanisms (message system, on-line database, scan system, and alarms) shall be used for implementation of the ASM SW. Portability to VxWorks Tornado has to be considered.

The Rtap facilities for history data recording, command scheduling, and plotting + trending shall be used for processing of the data.

Short-FITS format files shall be the interface platform to the VLT on-line archive.

2.3.2 On-Line Database

The LCU on-line database contains attributes for the provided ASM data, and for all signals connected to the LCU. Beside that, all control and status information that is not related to any signal shall be represented by an attribute in the database. Most of the attributes in the LCU database will be scanned into the ASM WS database, and some will have alarms attached to them.

For the database design, an object oriented approach shall be used. All attributes belonging to an object shall be located under the same branch. The database shall include separate branches for each ASM unit (Meteo, DIMM, etc.). The source format shall be compatible with the CCS database loader (dbl) [10].

Database points and attributes needed for Rtap historian and plotting facilities may have to be created and configured interactively, since no class support by CCS is currently available. It will therefore not be possible to generate the database entirely from dbl sources.

Rtap security mechanisms shall be used to inhibit unauthorized writing to the database from outside.

2.3.3 Software Distribution

On the WS the following functionality shall be implemented:

- All user interfaces, both graphical and command oriented
- Storage of history data, public data-files, and possibly image data
- Control process for DIMM, inclusive auto-guiding part
- DIMM seeing calculation routines which do not require image data
- Interface to on-line archive
- ASM target-star list
- Engineering and maintenance tools

On the LCU(s) the following functionality shall be implemented:

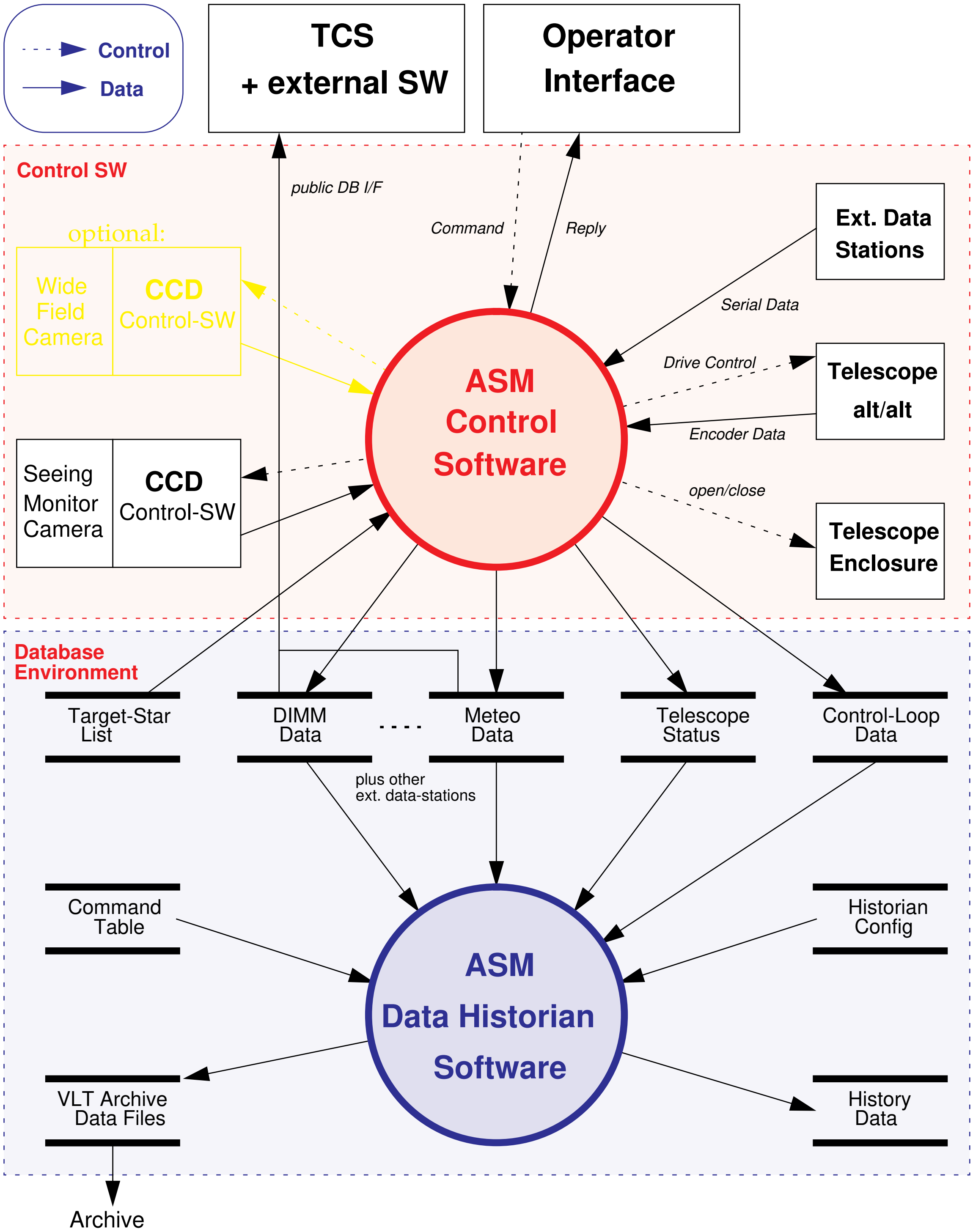
- All drivers and interface processes to external HW
- Real-time part of CCD SW
- Telescope control with presetting and tracking
- DIMM seeing calculation routines which require image data

The SW distribution will be more detailed in chapter 3.

2.3.4 Top-level Software Diagram

The diagram shows the relations between the ASM software and its environment:

Fig. 1 Top-level Software Diagram



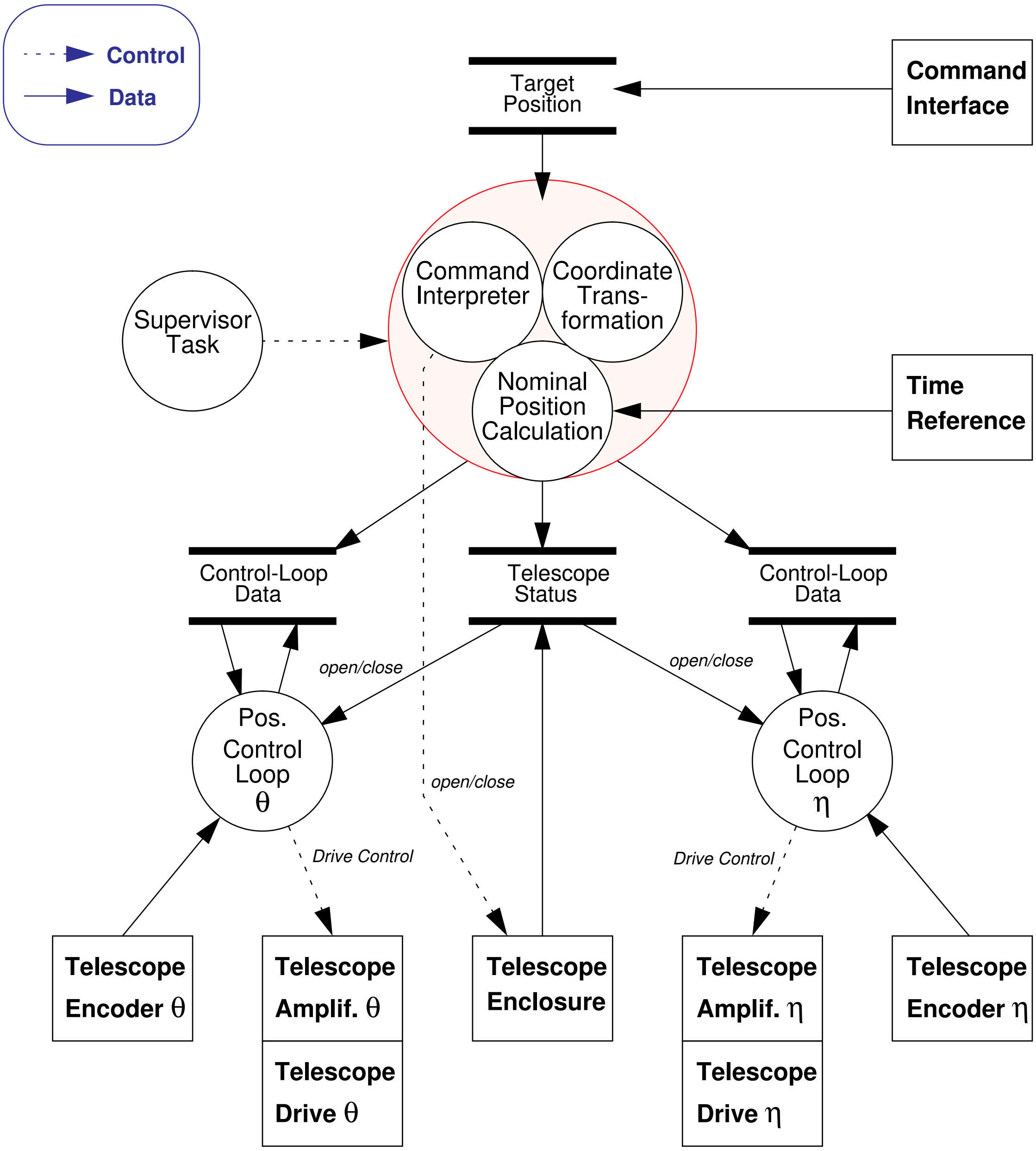
The *ASM Control Software* block represents the functionality to control external HW and SW, and to provide the on-line data. It consists of real-time parts as well as WS parts.

The *ASM Data Historian Software* includes functionality to produce data for the VLT on-line archive and internal history data for second-level processing of the on-line data. It is mainly located in the WS environment.

2.3.5 Diagram of ASM-Telescope Motion Control

This functionality requires real-time performance in all parts. The diagram shows the objects in the ASM-Telescope Motion Control software:

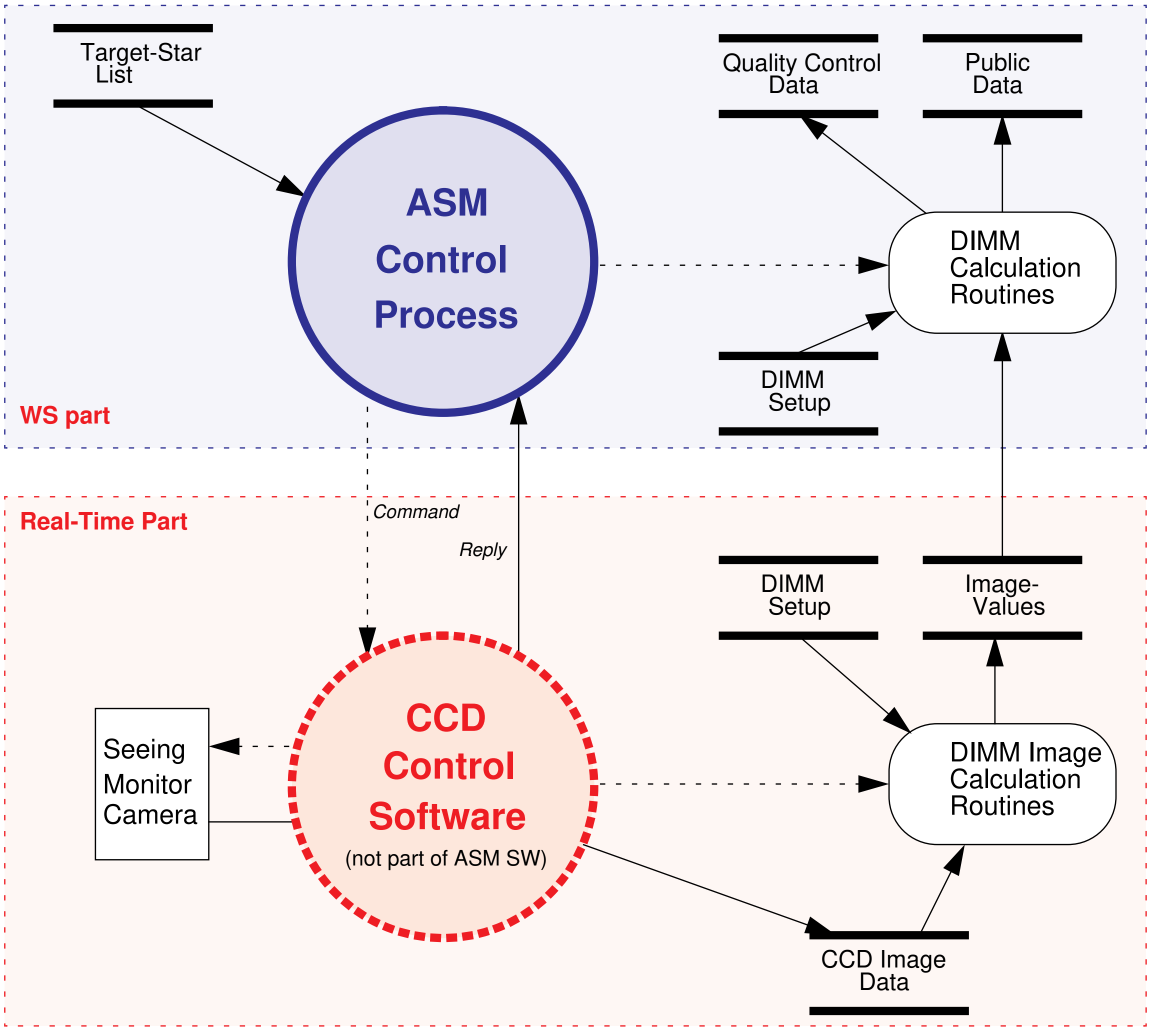
Fig. 2 Diagram of ASM-Telescope Motion Control Software



The existing TCS modules shall be re-used when possible.

2.3.6 Diagram of DIMM Software

Fig. 3 DIMM Calculation Software Diagram



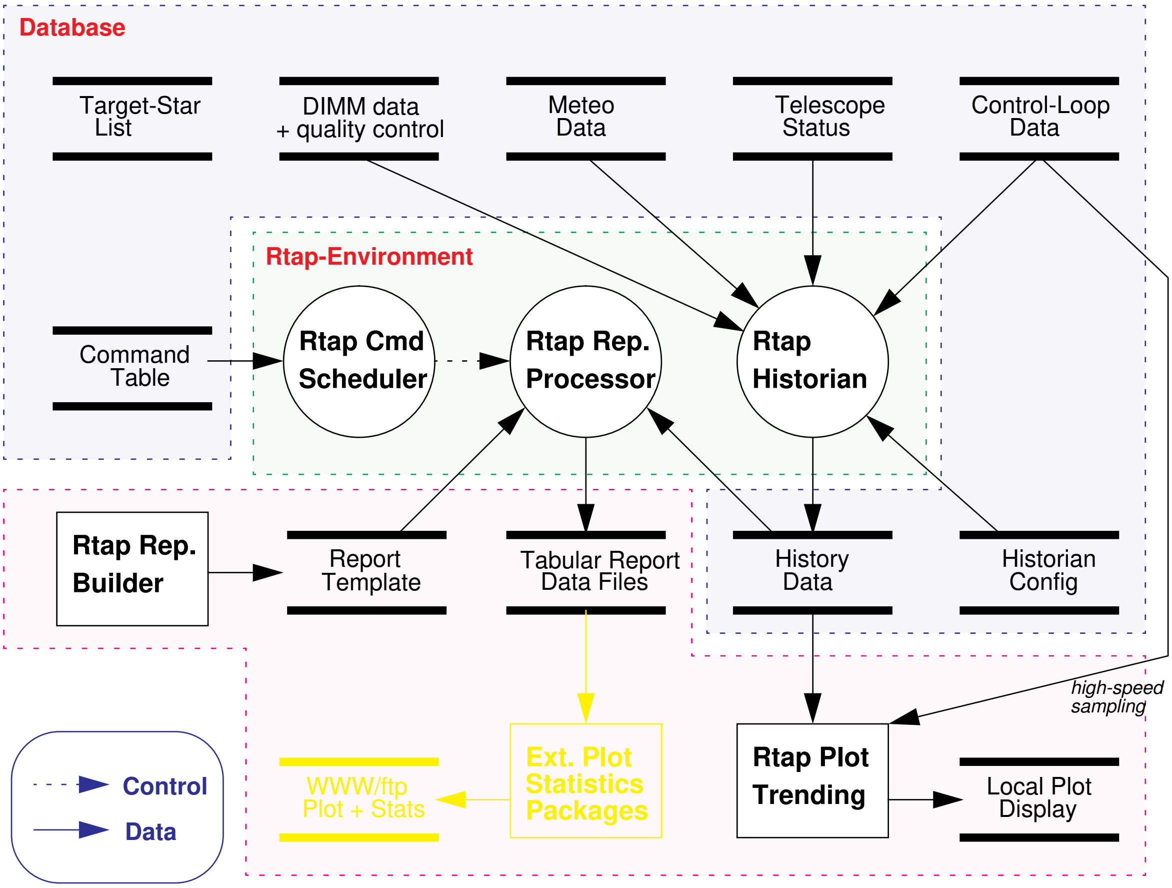
This part requests exposures of the seeing monitor camera from the CCD software, performs the needed calculations on the queried images, and provides the results in the on-line database. The ASM control process coordinates this.

Calculation which require the queried image data shall be done on the LCU to avoid image-transfer to the WS. The rest of the calculation shall be done on the WS.

The quality-control data include only the instant values. The history recording for ASM-internal purpose is done separately, see section 2.3.7.

2.3.7 Diagram of internal History Data Processing

Fig. 4 Diagram of Database Processing



This block performs the second-level processing of on-line data for ASM-internal purposes:

- history data recording
- generation of plots and periodical reports

Data-files for diagnostic purposes can also be generated directly by the real-time part of the ASM control software, provided that the sensor facilities provide the required format one-to-one and no post-processing is necessary.

The DIMM-data shall also include the quality-control part. There must be a history recording of this data, accessible for external SW that performs the quality checks.

3 SYSTEM FUNCTIONS AND FACILITIES

3.1 ASM Global Specifications

Following points apply to the ASM SW as a whole:

1. Fully **automated operation** shall be supported, i.e. upon reboot of WS and LCU(s) the operation shall start without the need of any manual intervention (except possibly a simple *start* command), provided that there are no problems in the system environment.
2. The only **entry point** to the ASM facilities from external users and software shall be the ASM WS. It is not allowed to access the connected LCU(s) directly from outside.
3. The **provided ASM data** shall be updated in the on-line database periodically in the respective intervals stated in the module descriptions (section 3.2 ff.).
4. **Redundancy** for ASM data items shall be supported by calculating the average value of the available sensors measuring the same physical parameter (e.g. wind-speed in 10/20/30 m height), leaving out the invalid data. It is not foreseen to have totally independent sets of sensor units.
5. It shall be supported to **read the current values** from the on-line database. This shall be possible from any VLT WS environment via CCS message system (database interface).
6. **Public data** shall be made readable for everyone in the on-line database, while reading of private attributes or writing from outside shall be inhibited.
7. **History data is not provided** publicly by the ASM; it will be the task of the VLT on-line archive to build a long-term database, that can be queried for history data.
8. The **duration of history data** recorded in the on-line database shall include at least the last week. A wrap-around mode shall be used, so that the oldest data is overwritten after that duration. The history data shall only be for *internal* use.
9. The ASM SW shall not directly **influence VLT operations** and telescope functions by its own initiative. It shall only provide the data for external SW that may or may not take actions depending on the ASM data.
10. The ASM data shall be stored periodically (typically every minute) into the VLT on-line archive in form of FITS files. The ASM SW will not provide a user access interface to those files.
11. It shall be possible to add further **'Sensor Extensions'**, similar to the foreseen meteo-station, seismic monitor or infrared emissivity sensor. They shall consist of a VLT-standard module [6] with a provided database branch (source in dbl format [10]) and application functions for LCU that receive and process the provided data from serial RS232 line. Sensor extensions should not require WS facilities further to a scanned database copy.
12. It shall be possible to add further **'Data Extensions'**, as the planned data prediction unit. They shall consist of a VLT-standard module [6] with a provided database branch (source in dbl format [10]) and an application program for WS that calculates the provided data from already existing database attributes based on physical models. Data extensions should not require LCU facilities. Included are the possibility of **'Link Extensions'** that evaluate data from non-local sensors e.g. via ftp/WWW, as required in [2].
13. **Backup tools** to save the ASM data on (possibly removable) media are not in the scope of the ASM SW. This is in contrast to [2]. The VLT on-line archive already provides such feature.
14. Every ASM-specific GUI panel should be designed to be usable from the foreseen portable X-terminal, i.e. it should not be larger than the provided screen size (worst case 640x480 pixels).
15. The standard CCS tools shall be available for **engineering** purposes, i.e. sampling for control loop tuning, command interface to move telescope etc.

3.2 ASM Module Subdivision

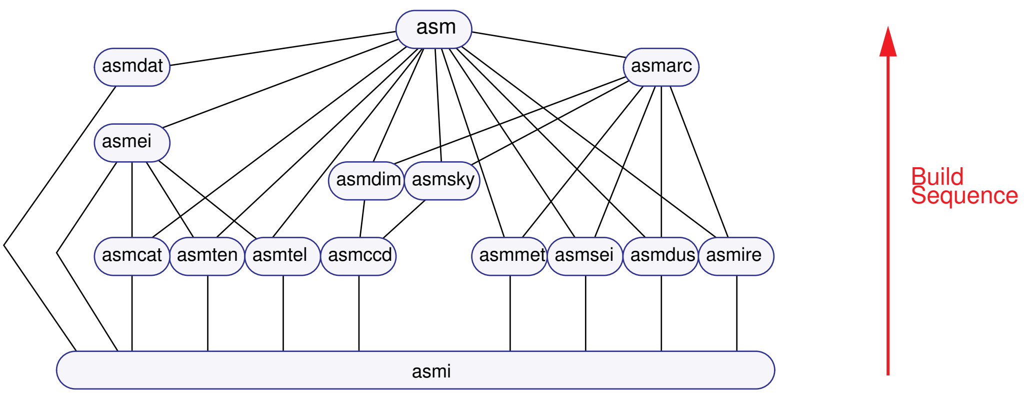
The whole ASM system is subdivided into the following parts, that will be separately evaluated in the coming sections. The table contains also the distribution on modules, and whether the functionality shall be implemented on LCU or workstation:

Table 1: ASM module overview

Purpose	Device ID [a]	Module ID	on WS	on LCU	Design Priority[b]	Complexity [c]	Ref.-Section[d]
Common Functionality (top level)	Control Monitor	asm	X	X	high	high	3.3
Common Functionality (bottom level)	N/A	asmi	X	X	high	low	3.3
ASM-Telescope Enclosure	Enclosure	asmten		X	medium	low	3.4
ASM-Telescope Motion Control	Telescope	asmtel		X	high	high	3.5
Target-Star List	Targets	asmcat	X		high	low	3.6
CCD Control	(CCD name)	asmccd	X		high	high	3.7
DIMM Control (incl. cloud monitor line-of-sight))	Dimm	asmdim	X	X	high	high	3.8
Meteo-Station	Meteo	asmmet		X	medium	medium	3.9
All-Sky Monitor	Sky	asmsky	X		low	?	3.10
Seismic Monitor	Seismic	asmsei		X	low	?	3.11
Dust Monitor	Dust	asmdus		X	low	?	3.12
Infrared Emissivity Sensor	Infrared	asmire		X	low	?	3.13
Engineering and Maintenance Tools	N/A	asmei	X		low	medium	3.14
Internal History Data Handling	N/A	asmdat	X		low	medium	3.15
VLT on-line archive interface	N/A	asmarc	X		medium	medium	3.16

a. *Device ID* is used as name of the software-device, and its root-point of the database branch.
b. *Priority* is assigned according to the urgency of implementation at the present time.
c. *Complexity* is an estimation of how difficult the implementation will be.
d. Cross reference to the section containing the detailed device description.

Fig. 5 Hierarchy of modules



The hierarchy of modules is given by the diagram. The interconnection between modules is in one direction only, in order to avoid circular dependencies. Modules connected by a line may make use of functionality contained in the respective lower level module, and all farther below. The build-sequence must be from bottom to top level.

3.3 Common Functionality

These modules shall provide facilities to lower level modules, as well as contain facilities to bind lower level functionality to a common unit:

- Library functions that are needed by more than one of the higher level modules. Which exactly these are, shall be decided in detail during the design.
- Procedures that bind functionality of the lower level modules to a unit, e.g. to link LCU libraries to one binary module.

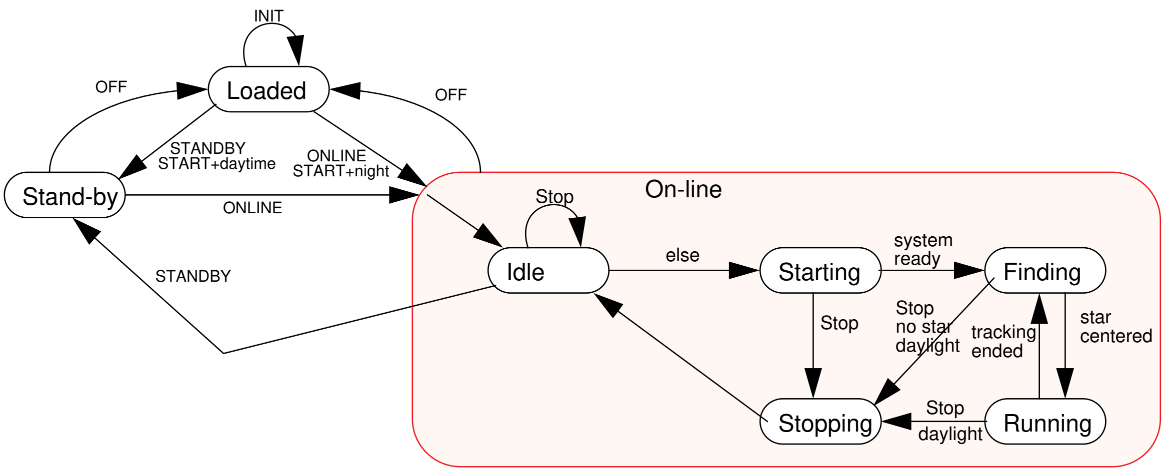
Parts for WS and LCU shall be included. To avoid circular dependencies, two modules are introduced:

1. **asm**: Top-level module to bind LCU and WS facilities from lower level modules
2. **asmi**: Bottom-level module for LCU and WS common functionality

3.3.1 WS Control Process

The WS control process shall be named *asmControl*. Its purpose is to perform the automatic operation, the start-up/shutdown sequences, and the state-switching On-line/Stand-by for connected LCUs according to the time of day, and meteorological conditions, e.g. too high wind speed.

Fig. 6 asmControl state diagram



3.3.1.1 Start-Up Initialization

An *INIT* command given to the *asmControl* process shall result in:

1. Send *lccWarmStart* command to *lccServer* of all connected LCU environments.
2. Check that specified target-state *On-Line/Stand-By* has been reached.

3.3.1.2 Automatic Operation

During day-time *Stand-By* state, and during night-time the *On-line* state shall be selected. A ***START*** command shall enter the automatic operation and select *On-line/Stand-by* state. The following sequence shall take place:

1. *Idle*:
   Check sun position and meteo conditions. If sun below horizon and meteo OK then continue with next step, otherwise wait.
2. *Starting*:
   Send *lccOnline* command to LCUs.

3. *Finding*:
   Determine a suitable target star by using the facilities of the target-star list.
4. Telescope control: forward target-star position to telescope motion control in order to preset to target position and continue with tracking for a certain duration.
5. *Centering*:
   determine star offsets and forward them to telescope motion control in order to adjust the target-position. Repeat until target star is centered.
6. *Running*:
   Call query routines and update on-line database with calculated data.
7. Auto-guiding: determine star offsets (as provided by DIMM) and forward them to telescope motion control in order to adjust the target-position.
8. Check sun position and meteo conditions. If not OK then go to step 11.
9. If tracking time has expired, go to step 3. in order to select new target star.
10. Continue with step 3. after *update interval* has expired.
11. *Stopping*:
    Send *lccStandby* command to LCUs and go to step 1.

3.3.1.3 Manual Operation

When an explicit *ONLINE* or *STANDBY* command is given to the *asmControl* process, the automatic operation shall be interrupted and the corresponding target state shall be selected. A *START* command shall then resume the automatic operation.

3.3.1.4 Shut-Down

An *OFF* command given to the *asmControl* process shall result in:

1. Send *lccShutdown* command to *lccServer* of all connected LCU environments.
2. Check that target-state *Loaded* has been reached.

3.3.2 Common LCU Monitor Process

A common monitor process shall exist that takes care of asynchronous events that require state-switching. This makes it unnecessary to implement dedicated background tasks for each software device. The monitor process shall be configured by a file. This can contain several *directives*, with each takes an *item-name* to be monitored:

SIGNAL <signal-name> - monitor signal values; result is the signal value

ATTR <attr-name> - monitor attribute values; result is the attribute value

TASK <task-name> - monitor tasks; result is 1 for alive and 0 for dead

CMD <command-name> - monitor with commands; result is the error-number, i.e. 0 for OK; parameters cannot be given to the command

The condition under which the item shall be monitored and the possible action is set with fields:

condition - operator (*equal* or *not equal*) and the expected result

device - associated software-device

monitorState - state in which to monitor the item

fallbackState - state to switch to when the condition is not met

fallbackAction - server process and command to send when the condition is not met

Each directive shall then be interpreted in the following way: Verify the given *item* according to the *directive* if the given *device* is currently in *monitorState*; if *condition* is not met then execute the *fallbackAction* and/or set the device-state to *fallbackState*.

This approach provides a flexible way to handle asynchronous events in any of the sub-module within the common ASM module, whereby code duplication can be greatly reduced, and the number of background tasks minimized.

3.3.3 Common LCU Command Interpreter

The ASM LCU(s) will have several LCC devices, of which most are relatively simple and have only few commands. It is therefore reasonable to foresee only one common CI to handle all devices [1], although the functionality is spread over different modules. To accomplish this, the common functionality module shall implement this single CI with all standard LCC commands, while the lower level modules only provide a library with the command handler functions that are specific to one device.

For standard commands, LCC passes the name of the LCC-device as parameter to the standard command handler. Depending on that the device-specific handlers shall be called. This shall be named '<*Module-ID*><*Command-Name*>', where command-name is capitalized (first letter upper case). If such function exists then it is executed during the common standard command handler, otherwise there is no action for this specific device.

3.3.4 Common CDT/CIT

The CDT and CIT for the LCU server process shall be included in this module, although the specific command handler function is implemented in the sub-level module.

Specialized CDT and CIT for the TCS tracking and axis control servers shall be provided and installed by the corresponding telescope-control sub-module, see section 3.5.

3.3.5 Standard States and Commands

All processes and devices shall support the following list of standard states:

Table 2: Common Standard States

State	Description
Off	device not available (same for all devices)
Loaded	undefined (same for all devices)
Stand-by	during day-time (details depend on specific device)
On-line	during night-time (details depend on specific device)

The implications for each specific device are stated in the coming sections.

The list of state transition commands to be supported are given in the following table:

Table 3: Common Standard Transition Commands

Command	Permitted in State:	Next State	Actions
INIT	Loaded	Loaded	initialize HW, set encoder positions, (re-)spawn background tasks
STANDBY	Loaded, On-line	Stand-by	bring HW to 'sleeping' position
ONLINE	Loaded, Stand-by	On-line	bring to fully operational state

1. In [8] it is recommended to have one CI per device, but nothing prevents to deviate from this.

Command	Permitted in State:	Next State	Actions
OFF	Stand-by, On-line	Loaded	reset HW, kill background tasks
EXIT	all	Off	calls OFF implicitly
STOP	all	(depends on device)	abort any HW movement

BREAK and *KILL* shall be implemented as STOP and EXIT, respectively, but with by-passing the internal message queue. It applies to all software-devices controlled by the server process.

In addition to the state transition commands, the following standard LCC commands shall be supported. Some commands, however, do not trigger any action and only return OK.

Table 4: Common Standard Auxiliary Commands

Command	Description
SIMULAT	start simulation of the hardware device controlled by the software device (hardware simulation)
STOPSIM	stop simulation of the software device; stop simulation shall put the software device in non-initialized state (Loaded)
SELFTST	perform a self-test of the software device. Self-test shall be performed on the lowest level only checking software devices, but not moving any hardware
STATUS	get device status as plain text.
VERBOSE	put the software device in verbose mode (per device, not per process)
VERSION	return version ID of loaded SW: archiving and compilation date/time
TEST	perform test-sequence with device

The detailed actions for each specific device are stated in the coming sections.

3.3.6 Bottom-Level Part

This module for WS and LCU common functionality shall contain features that are used by more than one module. The exact contents shall be decided during implementation.

3.3.6.1 Database

This shall include dbl classes that are needed by more than one module:

- Serial device database classes: communication parameters
- LCC digital and analog signals
- Generic module database branches

3.3.6.2 Headers

- Error handling macros
- Reply handling macros

3.3.6.3 Common Function Library

The library shall include functions that are needed by more than one module:

- Serial device with database handling: ComOpen, ComClose, ComReceive
  Note: the TCS/hb module contains similar routines that could be adopted!
- Verbose reply functions

3.4 ASM-Telescope Enclosure

3.4.1 Properties

The enclosure can be operated in two modes, selectable by an operator-switch. It shall be stored as enclosure substate in the database, additionally to a *Error* indication:

- ***Local***: Enclosure position is fully under local operator control.
- ***Automatic***: Enclosure position is under software-control.
- ***Error***: unrecoverable hardware error detected

The enclosure is regarded as fully independent from the telescope.

3.4.2 Interface

In all cases the maximum range is protected by *limit-switches*, that disable movement in the same direction by means of HW. In automatic mode, a *daylight-sensor* forces the enclosure - regardless of software commands - to the closed position; it can be overruled when *local* mode is selected.

The HW is controlled over a serial line by the LCU. The properties are as specified in section 7.1.

3.4.3 Functions

Control Commands to be implemented: [1]

- **tenOpen**: Request to open enclosure
- **tenClose**: Request to close enclosure

All commands shall send a final reply when the movement is completed, taking a maximum timeout of 60 seconds into account. A command shall be effectively ignored when the corresponding limit is already reached.

- **tenIsOpen**: checks if enclosure is open
- **tenIsClose**: checks if enclosure is closed

Error cases for commands are:

- During movement there is no change in position reported within one second
- Timeout exceeded without reaching limit position
- *Daylight* status asserted when *open* is requested
- *Local* status asserted

Alarms to be implemented (with scope/severity):

- *Local* substate asserted (Local/Warning)
- *Error* substate asserted (Local/Fatal)

Information to be logged (additionally to unrecoverable errors):

- Any change of the enclosure state or substate
- When the enclosure reaches the open/close limit
- When the daylight sensor becomes asserted

---

1. Command names are symbolic. They shall be defined during design, and have the module name as prefix.

3.4.4 Standard States and Transitions

The enclosure shall correspond to the LCC-device ***Enclosure***.

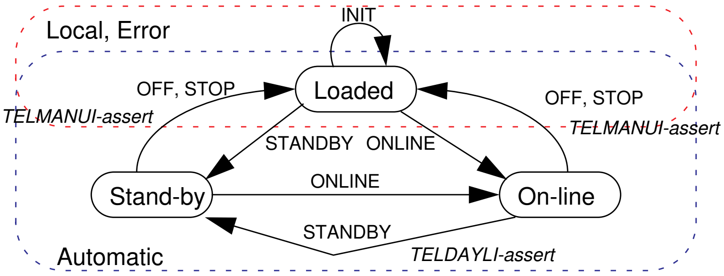
Table 5: 'Enclosure' LCC standard states

State	Implication
Off	Undefined
Loaded	Undefined
Stand-by	Enclosure closed
On-line	Enclosure opened

Table 6: 'Enclosure' LCC standard commands

Command	Permitted in State:	Next State	Actions
tenINIT	Loaded	Loaded	none
tenSTANDBY	Loaded, On-line	Stand-by	identical to **tenClose**
tenONLINE	Loaded, Stand-by	On-line	identical to **tenOpen**
tenOFF	Stand-by, On-line	Loaded	none
tenEXIT	Loaded	Off	none
tenSTOP	all	Loaded	abort movement of enclosure
tenSELFTST	all	same	check that serial line is working, check for Error substate
tenISOPEN	all	same	identical to **tenIsOpen**
tenISCLOSE	all	same	identical to **tenIsClose**

Fig. 7 'enclosure' state diagram



3.5 ASM-Telescope Motion Control

3.5.1 Properties

The current telescope substate must be stored in the database. Defined telescope substates are:

- ***Local***: telescope is not under SW control, but entirely controlled by HW (handset)
- ***Idle***: telescope is under SW control, but no movement
- ***Presetting***: telescope is moving to the target position
- ***Tracking***: telescope is following target position
- ***Error***: unrecoverable hardware error detected

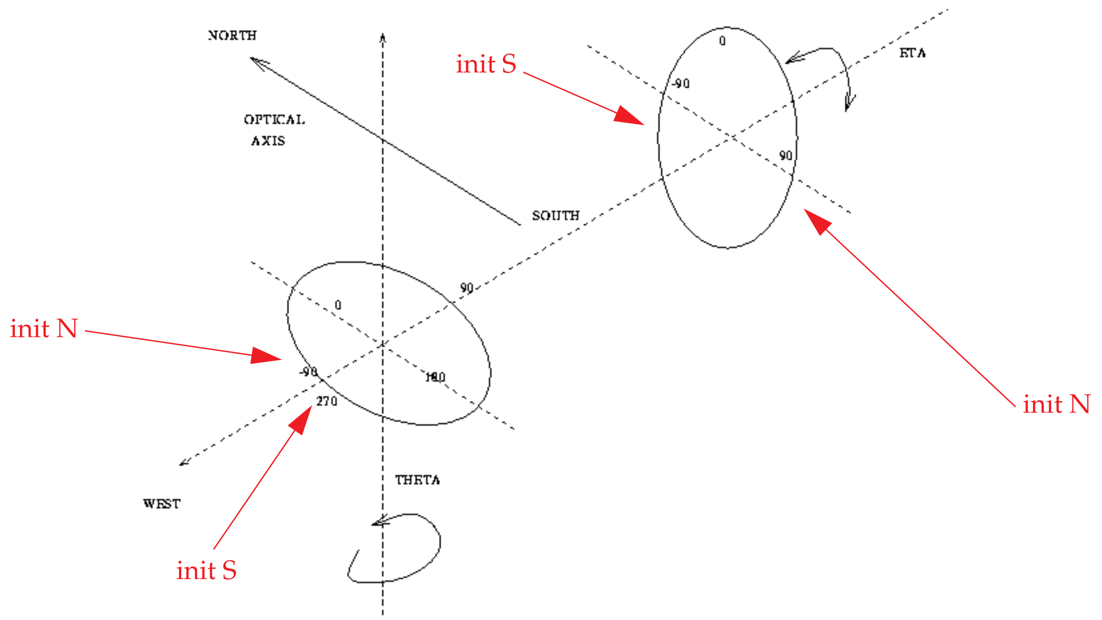
Static database values to be defined (they are not foreseen for interactive input):

- Location of telescope (latitude [*degree/arcmin/arcsec*] and longitude [*degree/arcmin/arcsec*])
- Software limits of axes coordinates:
  south tracking zone ($-100^\circ < \eta < 0^\circ$, $95^\circ < \theta < 265^\circ$) and
  north tracking zone ($0^\circ < \eta < 100^\circ$, $-85^\circ < \theta < 85^\circ$), see [2] and [7].

In the present DIMM SW, the two axes are never moved at the same time. This would however be possible. The two axes can be moved independently as long as the tracking zone is not exceeded. During design it shall be investigated whether SW-limits can be applied dynamically depending on the tracking zone.

Encoder initialization points shall be foreseen in each half-sky before the ends of the operational range. HW interlocks shall inhibit to exceed the tracking zones.

Fig. 8 Telescope coordinate definitions and initialization points



The telescope is regarded as fully independent from the enclosure.

3.5.2 Interface

In all cases the maximum range shall be protected by *collision-detectors* (TBD, part of HW design), that disable movement by means of HW. Additionally a *tilt-switch* is implemented that disables the power of the drives in case of SW malfunction. The tilt-switch can only be reset by operator intervention, no status signal for the SW is foreseen.

Status signals/values provided by HW:

- 2 encoder init-switches, asserted when the respective axis is at the init-point in either half-sky (same signal applies to both ends), as digital inputs.
- 2 interlock signals
- 2 encoder signals, sin/cos type incremental with reference-mark
- 2 tacho signals, to be handled within HW, no SW intervention foreseen.

Control signals/values to be driven by SW: [1]

- ***TELILENA(1/2)O***: Interlock enable, as digital outputs.
- ***TELVREF(1/2)O***: Velocity reference to amplifier for each axis, as analog outputs.

3.5.3 Functions

The **TCS modules *axis*** [16] and ***trk*** [17] shall be used to control the telescope. The entry-point to presetting and tracking of the telescope is then the ***asmtrkServer*** process. For test purposes the two *axisServer*s (instanciated as ***etaServer*** and ***thetaServer***) can also accept presetting commands. The *asmtel* module shall then additionally provide the 'glue-logic' for the two TCS modules:

- encoder interface functions
- interface functions to amplifier and other hardware
- coordinate conversion from alt/az (used by *trk*) to eta/theta (used by *axis*)

The TCS modules will typically handle the following commands:

- **axisServer/PRESET**: preset to specified target position. This command completes when the position is reached.
- **asmtrkServer/OBJSTAR**: preset to specified target position (*Presetting* substate, with velocity control), and when it is reached continue immediately with tracking for the specified duration (*Tracking* substate, with position control). This command completes after *Tracking* substate is entered.
- **asmtrkServer/OFFSAD**: give an auto-guiding offset step.

Additional control commands that are not handled by the TCS modules and which have to be implemented within the asmtel module are: [2]

- **telFlip**: move the telescope to one of the special positions [7]:
    - *north origin position*: optical axis points to north ($\eta = 0°$, $\theta = 0°$).
    - *south origin position*: optical axis points to south ($\eta = 90°$, $\theta = 180°$).
    - *dismount position*: starting from north origin position, the optical axis is flipped upwards to point vertically ($\eta = 90°$, $\theta = 0°$).

  This shall also do a 'flip-over' sequence which is not supported by the TCS modules. When position is reached *Idle* substate is entered, and the command is completed. Only allowed in *Idle* substate.
- **telSelftst**: perform self-test without movement. Allowed in any state. Items to be checked:
    - *Error* substate asserted?
    - two limit-switches at the same time asserted?
    - control-loop tasks alive?
    - encoder I/F in consistent state?
    - if velocity reference <> 0: is motor current zero? (only if supported by amplifier)

1. Signal names are symbolic. The exact names are specified in [15].
2. Command names are symbolic. They shall be defined during design, and have the module name as prefix.

The 'glue-logic' for *axis* and *trk* will consist of:

- **telTrkSendRefs** - convert *alt/az* coordinates provided by *trk* to *eta/theta* before passing them to *axis*.
- **telEncInit** - perform initialization sequence of the encoder. This includes moving to an init-switch, since there is no absolute reference foreseen.
- **telEncRead** - read the encoder position, incl. scaling and offsetting
- **telHwWriteSpeed** - write velocity reference to the amplifier.
- **telHwCheckInterlock** - check the external interlock signals.

The encoder functions shall use direct ikon driver calls.

Command Input Parameters:

- Target positions are specified in equatorial coordinate format
  - right ascension α [*hour/min/sec*]
  - declination δ [*degree/arcmin/arcsec*]

  (as they are usually listed in star catalogs) and internally converted to axes coordinates (η/θ system) as appropriate for the telescope mounting, with use of the time-reference system. Conversion formulas are given in [7].
- Relative position offsets are specified in *arc-seconds* and fractions of it.
- Special home-positions are marked with: *north*, *south*, *dismount*.
- Times are specified in *seconds*.

Error cases for commands are:

- The specified target position cannot be realized (i.e. outside SW-limits, below horizon etc.).
- During movement a limit-switch is asserted.
- During movement the tracking zone [7] is exceeded. An automatic 'flip-over' to the other tracking zone is not foreseen.
- The specified relative position is too large, i.e. its absolute value is greater than a maximum value specified in the database (approximately 100 arcsec ***TBD***).
- A relative offset is commanded when not in *Tracking* substate.
- Hardware malfunction detected, e.g.:
  - Two opposite limit-switches at the same time asserted
  - Drive/encoder does not react on velocity reference (possibly the tilt-switch is tripped)
  - Interlock signal asserted

Alarms to be implemented (with scope/severity):

- One of the limit-switches is asserted (Local/Warning)
- Telescope is in *Local* substate (Local/Warning)
- Telescope is in *Error* substate (Local/Fatal)

Information to be logged (additionally to unrecoverable errors):

- Any change of the telescope state or substate
- When a new target position is commanded (with positions)
- When a new target position is reached
- When a special home position is commanded

Additional remarks about the telescope motion control SW:

- It does not know about the target-star catalog, it uses only numerical positions.
- It does not incorporate an auto-guiding functionality, only the offset command is provided.
- During presetting it must be possible to abort the movement.
- The actual position must be constantly updated in the database, also in *Local* and *Idle* substate.
- There is no connection with the enclosure, i.e. the telescope can in principle be moved even when the enclosure is closed.

3.5.4 States and Transitions

The ASM-telescope shall correspond to the LCC-device ***Telescope***.

Table 7: 'telescope' device states

State/Substate	Implication
Off	Undefined
Loaded Loaded/Error.	Undefined Undefined, telescope defective and not operational
Stand-by	Telescope movement stopped (at any position).
On-line/Idle On-line/Presetting On-line/Tracking On-line/Local	Telescope ready without movement Telescope presets to target star Telescope follows target star Telescope under handset control **In general the TCS definitions shall be used!**

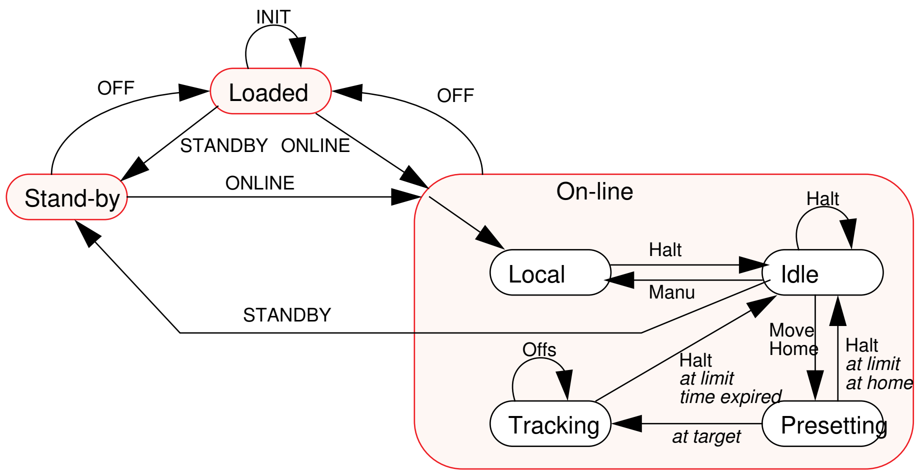
Table 8: 'Telescope' device-specific commands

Command	Permitted in State/Substate:	Next State	Actions
telHome	On-line/Idle	On-line/Presetting..Idle	see text

Table 9: 'Telescope' LCC standard commands

Command	Permitted in State/Substate:	Next State	Actions
telSELFTST	all	same	call **telSelf**

Fig. 9 'telescope' state diagram



Note: the state-diagram shall only be a suggestion. The definitions of the TCS modules will in any case take precedence.

3.5.5 Velocity Control-Loop

This loop shall be implemented in hardware (see [15]), no software intervention is foreseen. It should be incorporated into the power amplifier. **The amplifier must provide and handle the handset connection without any SW- support!**

3.5.6 Position Control-Loop

This loop shall be implemented in software.

Requirements:

- controller type: PI
- tracking accuracy: < 1 CCD-pixel nominal, < 20 arcsec max. ***TBC***
- presetting accuracy: 5 arcmin max.
- servo loop period: 5 ms

SW considerations:

- For each axis a separate task is foreseen, as instances of the same function.
- Each task works on different data sets.
- The real-time priority of these tasks must be high (VxWorks ca. 1 to 20).
- It is a background task and has no command interface.
- Nothing shall be logged by these tasks (for performance reasons).

Required control-Loop values in the database consist of:

- W: nominal position
- X: actual position
- Y: velocity reference
- Ep: position error
- Kp: Proportional Gain
- Ki: Integral Gain
- max_velocity
- max_acceleration
- max_jerk

For the values W, X, Y, and Ep it is necessary to calculate average and RMS values over 1 second on the real-time system and store them into the database. This is needed since the minimum scan-period is 1 second, and it shall be possible to scan and record the control-loop values in an Rtap database for maintenance purpose. The suffixes '*_avg*' and '*_rms*' shall be used for the attribute names of these values. These values are, however, not needed for normal operation, and are therefore optional.

3.5.7 Movement Characteristics

The HW design shall provide scaling factors and limits that are applicable to amplifier and drives.

A switchable motion controller shall be used, so that the control-loop algorithm depends on the magnitude of the position error. The values of the switching points shall be adjustable in the database.

Tuning of the axis control-loops shall be done by modifying the parameters in the database and updating the values by an *UPDATE* command to the *axisServers*.

3.6 Target-Star List

3.6.1 Properties

The ASM target-star list contains a (small) number of star parameters that are used as target positions for the telescope. It is not intended to use existing star catalogs in the VLT environment, since the implementation of a corresponding interface is considered as more difficult than an independent solution.

A database table shall be used, which contains for each star the fields:

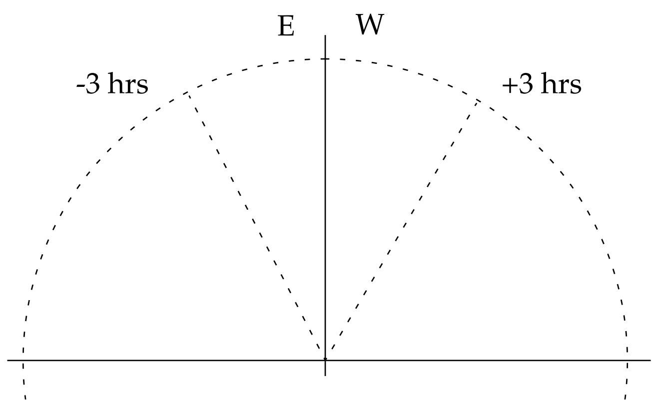
- *Name*
- *Position* in equatorial coordinate format:
    - right ascension α [*hour/min/sec*]
    - declination δ [*degree/arcmin/arcsec*]
- *Magnitude*
- *Flux* value that is updated at run-time with the highest value ever measured. This is used for the line-of-sight cloud-monitor calculation.

Max. number of stars to be stored: 30

The table can contain stars of both half-skies, but due to the mechanical restriction of the telescope mounting (see 3.5), an additional database flag shall be provided that restricts the selection to stars of either the southern or northern half-sky, and ignores the others.

Global coefficients shall be foreseen in the database to allow a weighted selection of a star based on three criteria:

a. minimize the number of changes of a target-star ⇒ select star with *HA* closest to -3 hrs
b. minimize airmass ⇒ select star closest to zenith, i.e. *HA* closest to zero
c. maximize brightness ⇒ select star with highest (most negative) magnitude *M*



3.6.2 Functions

A function shall be provided that determines the brightest star with negative hour-angle at the present time [2]. It shall also provide the maximum tracking time-span until the star exceeds 3 hrs.

With the database coefficients, the star selection is then based on:

$$z = -a * HA - b * |HA| - c * M \text{ with } -3 \text{ hrs} \leq HA \leq +3 \text{ hrs}$$

and the star with highest z is selected.

3.7 CCD Control

3.7.1 Properties

Standard CCD SW shall be used [11]. Further to that no special application SW for CCD control is foreseen.

The CCD chip is of frame transfer type. This kind of chip can provide an accuracy of exposure time of the order of 1-2 ms without shutter. This is suitable for the seeing monitor application.

The Real Time Display (RTD) facility shall be used on the ASM WS to view the CCD image. No other monitor facility is foreseen, particularly no hardware screen, so that the RTD is mandatory for adjustment procedures.

3.7.2 Interface

Status signals provided by HW: none (only CCD internals)

Control signals to be driven by SW: none (only CCD internals)

3.7.3 Functions

The entry point for all CCD commands is the ASM WS. The DIMM control SW uses the CCD SW for image readout and auto-guiding.

The purpose of this module is:

- Act as interface to the standard CCD SW
- Provide CCD setup files suitable for the DIMM application
- Provide tools for CCD maintenance ***(TBC if something further to the CCD SW is needed)***

Note: The dedicated ASM-module to interface to the CCD-SW (*asmccd*) can be omitted if all needed facilities are already provided. Only if more than one ASM-module needs a functionality that is not included in the CCD-SW, *asmccd* shall be realized.

Open CCD questions:

1. ***can the coherence monitor mode [2] be implemented (with shifting of lines after each of successive 1-2 ms exposures)?***

3.8 DIMM Control

3.8.1 Properties

The DIMM control shall be regarded as an instrument consisting of the seeing-monitor CCD camera. It queries images and calculates the data to be provided. It has a real-time part to do the image analysis, and a WS part for data transfer and calculation.

The DIMM operation is dependent on the following resources contained in other modules, which are coordinated by the *asmControl* process:

- Target-star list
- Telescope movement
- CCD control software

3.8.2 Interface

A signal interface is not required.

3.8.3 Provided Data

The DIMM SW shall provide the following data [2], which shall be stored in the on-line database in a table-attribute similar to the structure below:

Table 10: DIMM Provided Data

Tag [a]	Parameter	Unit	Update Interval	Statistics Interval [b]	Provided Fields
Seeing	Seeing (at Wavelength)	arcsec	60 s	3600 s *TBC*	**value** *optional*: rms, average, minimum, maximum, deviation
Wavelength	Wavelength	m	const.	N/A	value
Airmass	Airmass		variable	N/A	value
TimespanDimm	Time span seeing estimate	s	variable	N/A	value
Scintillation	Scintillation Index	% at 0.5 μm	60 s	N/A	value
Extinction	Atmospheric Extinction	Magn/airm	variable	N/A	value
ExtinctionRMS	Atmospheric Extinction (RMS)	Magn/airm	variable	N/A	value
PosAlpha	RA line-of-sight	HHMMSS.ttt	variable	N/A	value
PosDelta	Dec line-of-sight	DDMMSS.ttt	variable	N/A	value
TimespanSky	Time span sky monitor	s	variable	N/A	value
	Future extensions:				
WaveVelocity	Average Wavefront Velocity	m/s			
WaveDirection	Average Wavefront Direction	degree (mod. 180)			
WaveDispVelocity	Wavefront Dispersion Velocity	m/s			

a. To be used for 'addressing by content'.
b. Statistical values are calculated over this interval.

The calculation of the data is described in [2]. Only the calculation of a subset of those data are foreseen in the first release of the ASM software, while the remaining, e.g. *WaveVelocity*, *WaveDirection*, and *WaveDispVelocity*, are planned to be added later.

Invalid values - indicating a defective sensor or calculation failure - are indicated by a special value (if possible 'not-a-number' NaN) in the database.

The present values (*value* tag) are stored together with a time-stamp. Additionally some statistical values shall optionally be calculated for each delivered data item. They shall be stored in the database under the corresponding field tags *rms*, *average*, *minimum*, *maximum*, *deviation*.

3.8.4 Data Processing

The DIMM control SW is requiring the devices for telescope motion control, enclosure control, target-star list, and CCD control.

No operation is foreseen during day-time in Stand-by state.

During night-time, after the DIMM device has entered the On-line state, the following sequence shall take place (details in [2]):

1. CCD Control: query image (full frame exposure).
2. *Centering*: determine star offsets and make them available in the database for telescope motion control in order to adjust the target-position.
3. *Running*: CCD Control: query series of images, 1..10 ms (configurable) exposure time each
4. Calculate seeing and other data values from image series.
5. Save quality control data in history record.
6. Update on-line database with calculated data.
7. *Auto-guiding*: determine star offsets and make them available in the database for telescope motion control in order to adjust the target-position.
8. Writing of the data to a FITS file for archiving.

The image analysis calculations (i.e. center of gravity, maximum, total illumination, FWHM) shall be done on the LCU, by the CCD-SW itself, or by a function that is indirectly called by the CCD image sampling. The transfer of the calculated data from the LCU database to the WS shall be done only once per sequence to minimize LAN traffic.

The detailed distribution of tasks between the DIMM control and the top-level control process has to be specified during design. The side-effects are not fully clear yet, due to the interdependencies with the other software-devices.

The data-table is located in the WS database, from which all table elements are foreseen for public access. *Addressing by content* shall be used, using the stated tags. Additionally there shall be an history recording of internal quality control data, which are saved after each sequence.

3.8.5 Functions

Presumable list of classes to be implemented: [1]

- **dimDEVICE**: CCD-SW interfacing, calculation and storage of data
- **dimCAMERA**: CCD detector control SW interfacing (setup, exposures)
- **dimCONFIG**: handling of DIMM configuration and setup values in database
- **dimSEQUENCE**: handling of one exposure sequence
- **dimDATA**: handling of public provided data

---

1. All names are symbolic. They shall be defined during design, and have the module name as prefix.

- **dimAG**: auto-guiding, error-vector calculation

Alarms to be implemented (with scope/severity):

- One of the data values is invalid (Local/Severe)
- No data available (Local/Fatal)

Information to be logged (additionally to unrecoverable errors):

- Any change of the state
- Reason of rejected frames or sequences, together with the value of the involved parameter

3.8.6 Standard States and Transitions

Table 11: DIMM standard states

State/Substate	Implication
Off	Undefined
Loaded	Undefined
Stand-by	Telescope and enclosure set to Stand-by state. Controlled hardware is detached, no calculation and updating of data
On-line/Idle On-line/Starting On-line/Finding On-line/Running On-line/Stopping	Telescope and enclosure set to On-line state. Ready to control hardware and calculating data dto., preparing enclosure and telescope dto., selecting target-star and centering it dto., data query and calculation in progress dto., detaching enclosure and telescope

Table 12: DIMM standard commands

Command	Permitted in State/Substate:	Next State	Actions
dimINIT	Loaded	Loaded	read database
dimSTANDBY	Loaded, On-line/Idle	Stand-by	stop periodic image query and calculations
dimONLINE	Loaded, Stand-by	On-line	start periodic image query and calculations
dimOFF	Stand-by, On-line/Idle	Loaded	none
dimEXIT	Loaded	Off	none
dimSTOP	all	same	none

3.9 Meteo-Station

3.9.1 Interface

The unit is regarded as a black box. All data is transferred via serial RS232 line with the parameters:

- 9600 Baud, 8 bit, no parity, one stop

The name of the serial device and communication parameters shall be stored in the database.

The sample data format is described in section 8.2.2.

Status signals provided by HW: none

Control signals to be driven by SW: none

3.9.2 Provided Data

The meteo-station provides the following data [2], which shall be stored in the on-line database in a table-attribute similar to the structure below:

Table 13: 'meteo' Provided Data

Tag [a]	Parameter	Unit	Update Interval	Statistics Interval [b]	Provided Fields
Temp (*) Temp1 Temp2 Temp3	Temperature - at ground - at 2 m height - at 30 m height	Celsius	60 s	1200 s	**value**, average, minimum, maximum, deviation
DewTemp	Dew Temperature	Celsius	60 s	1200 s	dto.
Humidity	Relative Humidity	%	60 s	1200 s	dto.
Pressure Pressure0	Pressure in 2 m height Normalized Pressure (at sea level)	mbar	60 s	1200 s	dto.
WindSpeed (*) WindSpeed1 WindSpeed2 WindSpeed3	Wind Speed - at 10 m height - at 20 m height - at 30 m height	m/s	60 s	1200 s	dto.
WindDir (*) WindDir1 WindDir2 WindDir3	Wind Direction - at 10 m height - at 20 m height - at 30 m height	degree (N=0, E=90)	60 s	1200 s	dto.

a. To be used for 'addressing by content'.
b. Statistical values are calculated over this interval.

(*) The items marked with (*) are provided as calculated averages of respective items 1/2/3, due to the required redundancy [2]. If one sensor fails, then external SW can rely on the calculated value, or use one of the other specific values. The ASM SW only indicates invalid values, it does not calculate a value for the failed parameter from other values.

Note: there is only one humidity sensor, so that the required redundancy [2] for this parameter cannot be implemented!

Invalid values - indicating a defective sensor - are indicated by ***???*** in the data-stream and by a special value (if possible 'not-a-number' NaN) in the database. ***TBC***

The present values (*value* tag) are stored together with a time-stamp. Additionally the meteo-station provides also statistical values of each delivered data item. They shall be stored in the database under the corresponding field tags *average*, *minimum*, *maximum*, *deviation*.

3.9.3 Data Processing

Once every minute a new telegram is received from the meteo-station, then the following steps shall take place:

1. Parse the telegram and store the individual instant values into the database (not the statistics).
2. Optional: Append the telegram 1-to-1 to a file, which name is defined in the database, normally *meteo.log*. The file shall be located in the LCU's home directory.
3. If the minute reported in the telegram is 00, 20, or 40, then store also the statistical values into the database. They shall also be logged.
4. In a dedicated attribute it is signalled that new data are available.

The reception and processing of data shall be done by a background task on the LCU. [1]

By a dedicated WS-background-process the table contents shall then be transported from LCU to the WS database, from which all table elements are foreseen for public access. *Addressing by content* shall be used, using the stated tags. The WS-process shall also generate a file for the on-line archive and initiate its transfer, see section 3.16.

There shall be an Rtap history table where the *average*, *minimum*, *maximum* values are sampled once every *statistics interval*.

The date and time reported in the telegram by the meteo-station shall be checked, and if it differs for more than one minute from the LCU time, then an alarm shall be raised. Apart from that the reported time is not taken into account. There is also no possibility to set the time from the LCU; this must be done by an operator.

3.9.4 Functions

Control Commands to be implemented: [2]

- **metInit**: kill background task if already running, spawn background task from scratch. Open file-descriptor for serial line, allocate resources.
- **metKill**: kill background task, release allocated resources.

Error cases for commands are:

- Background task cannot be spawned resp. killed.

Alarms to be implemented (with scope/severity):

- One of the data value is invalid, i.e. defective sensor (Local/Severe)
- No data received within timeout (Local/Fatal), shall put the device into state *Loaded*
- Date/time reported by meteo-station is incorrect (difference > 1 min) and requires adjustment by operator (Local/Warning)

Information to be logged (additionally to unrecoverable errors):

- Any change of the state
- Selected statistical values, to be specified during design/implementation

---

1. Remark: the existing routines of the La Silla DIMM2 can be re-used to a certain extend.
2. Command names are symbolic. They shall be defined during design, and have the module name as prefix.

3.9.5 Standard States and Transitions

The enclosure shall correspond to the LCC-device ***Meteo***.

Table 14: 'Meteo' LCC standard states

State	Implication
Off	Undefined
Loaded	Undefined
Stand-by	Receiving, processing, and updating data (same as On-line)
On-line	Receiving, processing, and updating data

Table 15: 'Meteo' LCC standard commands

Command	Permitted in State:	Next State	Actions
metINIT	Loaded	Loaded	call **metInit**.
metSTANDBY	Loaded, On-line	Stand-by	none
metONLINE	Loaded, Stand-by	On-line	none
metOFF	Stand-by, On-line	Loaded	call **metKill**
metEXIT	Loaded	Off	none
metSTOP	all	Loaded	none

3.10 All-Sky Monitor

The all-sky monitor shall provide at any time the image of the whole sky. This facility will not be provided during the first phase of the project.

Note that the calculation of line-of-sight cloud cover shall be included in the DIMM control SW.

3.11 Seismic Monitor

This unit will be added in a second phase, after the basic modules have been completed.

3.12 Dust Monitor

This unit will be added in a second phase, after the basic modules have been completed.

3.13 Infrared Emissivity Sensor

This unit will be added in a second phase, after the basic modules have been completed.

3.14 Engineering and Maintenance Tools

This unit will be added in a second phase, after the basic modules have been completed.

3.15 Internal History Data Handling

This unit will be added in a second phase, after the basic modules have been completed.

3.16 VLT On-line Archive Connection

3.16.1 Generation of Data Sets

The ASM application running on LCU or WS shall signal that new data are available in its on-line database. Depending on the specific sub-module, this can occur with different update periods.

A sub-module specific background process will recognize this event, possibly copy the data from LCU to WS, and then produce a short-FITS compliant file from the data to be archived (which may be a subset of the whole data available). File naming conventions are as in section 3.16.3, generated FITS keywords as in section 3.16.4. The file generation procedure shall make use of the INS/*oslxServer*. The actual archiving procedure is outlined in the next section.

3.16.2 Archiving Procedure

To send a generated file to the archive, the following steps shall be done:

1. The VLT On-line Archive Client (VOLAC) process is instructed to archive that file. This is done by writing the filename into a dedicated *newData* attribute in the Rtap database, to which VOLAC has attached an event. More than one *newData* attribute in the same Rtap database will be necessary to handle different update periods correctly.
2. The VOLAC takes ownership of the file and takes care of sending it to the VLT archive. The ASM application will in any case assume that the file will actually be archived. No error handling after the *newData* event trigger is foreseen.
3. The VOLAC returns ownership to ASM after the file has been completely transferred. The ASM operator may now delete the file, or use it for other purposes. The ASM application will not automatically move or delete the file.

Libraries for FITS file handling (oslx) and the VOLAC will be provided by INS. It is assumed that the VOLAC can handle any type of file, so that the original INS version can be adopted, and no ASM-specific extension or adaption of its code is necessary.

3.16.3 File Naming Convention

All ASM data files are automatically generated as PAF (parameter files) in short FITS format. They shall be named according to the scheme:

asm*xxx-yyyymmdd-HHMMSS*.paf

where:

- asm*xxx*: indication of data source, equal to module name, i.e. *asmmet*, *asmdim*, etc.
- *yyyy*: year in decimal 4-digit notation
- *mm*: month in decimal 2-digit notation [01-12]
- *dd*: day in decimal 2-digit notation [01-31]
- *HH*: hour in decimal 2-digit notation [00-23]
- *MM*: minute in decimal 2-digit notation [00-59]
- *SS*: second in decimal 2-digit notation [00-59]

3.16.4 ASM Archive FITS keywords

All FITS keywords have still to be defined with DMD.

Not all data values that are planned will actually be delivered in the first phase. The format must support extension of keywords for new values when they become available.

4 SYSTEM INTERFACES

This chapter contains the interfaces to the data provided by the ASM, and to its control.

4.1 Public Software Interfaces

The only interface provided by ASM to external user and software packages is the workstation on-line database with its public attributes. Command interfaces are only intended for ASM-internal use, including GUI panels for operation and maintenance of the ASM.

4.1.1 Public On-Line Database Interface

Any VLT software can retrieve ASM data from the ASM on-line database. Only accesses to attributes which are specified in chapter 3 are allowed (public attributes). Accesses to other attributes shall be rejected by means of Rtap security mechanisms.

Only the parameters at present time can be queried, no interface to historical data is foreseen (but a corresponding extension can be considered).

Access is done with the standard CCS or sequencer database functions, directed to the environment *wasm*. The detailed attribute names and access mechanisms shall be defined in the ASM software design description.

TCS may provide a library with access functions to the ASM, as requested in [1]. These will make use of the database interface. The implementation of such a library is completely under control of TCS, and not in the scope of ASM.

4.1.2 VLT On-Line Archive Interface

The ASM software shall generate files in short-FITS (i.e. 'PAF') format, which contain keywords that have to be agreed with DMD. It will use INS facilities to send the files to the archive. Details are given in section 3.16.

ASM itself will not support public access to these files, the ingested data may however be queried from the archive directly.

4.1.3 Interface to TCS

According to [1], some TCS functions need data from the ASM for on-line use (refraction calculations, enclosure devices setting, safety etc.). The following parameters are needed with the approximate reading periods:

- air temperature (outside the telescope enclosure) (period: 5 min)
- air pressure (period: 5 min)
- air humidity (period: 5 min)
- wind speed (outside the telescope enclosure) (period: 1 min)
- wind direction (outside the telescope enclosure) (period: 1 min)
- atmospheric seeing (outside the telescope enclosure) (period: 5 min)
- cloud coverage (in different directions) (period: 20 min)

TBC: cloud coverage in different directions is not available; there is only line-of sight!

TCS shall query these values from the ASM database as described above. No special TCS interface is foreseen.

4.2 Internal Software Interfaces

4.2.1 Command Interface

For operation of the ASM commands shall be directed, typically by an operator through GUI panels, to the *asmControl* process on the workstation.

The LCU processes will receive commands from the *asmControl* process.

4.2.2 Protocol Interfaces

The format of serial line protocols is described in chapter 3 for the respective sensor type:

- Enclosure
- Meteo-station (see also section 8.2.2)
- Seismic monitor
- Dust monitor
- Infrared emissivity sensor

4.2.3 Configuration Interfaces

- Signal Configuration File

  A configuration file for all signals will be provided by the HW designer, in order to have as much as possible a centralized definition of signals. It shall consist of one line per signal, which have the following format for the various signal-types:

  - Digital signal: sigName bitNum bitCount simValue
  - Analog cabled signal: sigName sigNum cvtFactor lower upper gain simValue

  In 'sigName' the direction is indicated by the suffix 'I' or 'O'; for active-low digital signals, a '*' is appended. The remaining parameters of the cabled signals correspond to the order in the LCC signal configuration call [8].

4.3 Graphical User Interfaces

GUI panels to be foreseen for normal operation:

Automatic Operation Panel

CCD software user interface

Real-time display

GUI panels and applications for internal history data recording:

Rtap plot facility

Rtap historian with statistics

GUI panels for commisioning and maintenance:

DIMM configuration

Manual Target Star Selection

Meteo Status Panel

For special purposes the CCS/LCC Engineering Interface can be used to access the LCU.

5 SYSTEM ATTRIBUTES

5.1 Rules and Standards

5.1.1 Naming rules

1. In general all the rules of the programming standards [6] shall apply.
2. **Installed files** shall have the module name as prefix.
3. **Public functions** shall have the module name as prefix. The remainder shall be capitalized. The object shall come before the verb, e.g. *StatusGet* instead of *GetStatus*.
4. **Commands** are given for each device in chapter 3. They are symbolic names, which are only used for identification purpose, and shall be more detailed during the design phase. The function that implements the command handler shall in any case have the module name as prefix, and the remainder shall correspond to the capitalized command name.
5. **Database classes** shall have the module name as prefix. The remainder shall be capitalized.
6. **Signal names** shall be all upper case. The module name should be used as prefix, except that *ASM* is omitted. The postfix shall be 'I' for input, 'O' for output, '*' for active-low.
7. Nothing shall be installed by modules under *$VLTDATA*.

5.1.2 Rules for WS Modules

1. Applications shall be based on the CCS event-handler (evh) module.
2. The main WS control process shall be named *asmControl*.

5.1.3 Rules for LCU Modules

1. There shall not be one CI per SW-device, unlike recommended in [8]. Since there will be only few commands for each device, one common CDT and CI shall be used whenever possible, and implemented by the *asm* module.
2. Each module shall provide its functions as a library. The *asm* module shall then link the provided libraries to one single binary module for loading.
   Note: this is unusual compared to existing VLT SW.

5.1.4 Rules for CCS/LCC Software Devices

The following rules shall be implemented in connection with LCC software devices:

1. SW-devices have a ***state*** and may have a ***substate*** if needed.
2. Standard LCC commands influence the *state* of the SW-device, but not its *substate*.
3. Standard LCC commands call SW-device-specific command handlers to perform the requested action(s).
4. SW-device-specific commands influence the ***substate***, and may set the ***state*** to a lower value.
5. SW-device-specific commands do not call Standard LCC command handlers.
6. Events may set the *state* to a lower value, and influence the *substate*.
7. *Stand-alone* state is handled identically to *On-line*.
8. The CI shall accept the standard state-transition commands as defined in [8]. Some commands, however, do not trigger any action and only return OK.

5.1.5 Rules for Database

1. All configuration and run-time data except from auxiliary data shall be kept in the database.
2. Point names shall be capitalized (first letter upper case), attribute names shall have lower case.

3. All points and attributes of a module shall be grouped in a separate branch.
4. The name of the root-point shall be as stated in the table in section 3.2 (*Device ID*).
5. The alias of the root-point of the branch shall have the name of the module.
6. Point aliases shall in general be formed by adding the prefix *asm* to the point name.
7. The root-point shall contain numerical attributes for *state*, *substate*, and *verbose*, even if not used.
8. Some of the LCU attributes are reported through the scan system to the ASM workstation, where they are used to generate in particular alarm notification messages.
9. Since scanning into tables is not possible, but tables are very convenient to use (e.g. *addressing by content*), a periodic update shall be scheduled in case of low frequency updates ($\geq$ 1 min).
10. An object oriented approach shall be used for the LCU on-line database. All attributes belonging to an object shall be located under the same branch.
11. The source format shall be compatible with the CCS database loader (dbl) [10].

5.2 Availability

To the possible extend, the ASM SW shall implement self-recovery procedures to solve anomalous situations. Occurrences of anomalous situations and failures are recorded via the CCS logging system to be available for further analysis.

From the experiences with the VLT common SW so far, mainly LAN network problems can be expected. The MTBF shall not be lower than one month.

5.3 Maintainability

The maintainability of the ASM control software is enhanced by the following factors:

1. Use of VLT-standard HW components for LCUs, as far as possible.
2. Use of standard software (Rtap, CCS, LCC, drivers), exclusively.
3. Use of the database to store configuration parameters, reducing the needs for changes in the software.
4. Use of a signal configuration file that is derived from a central definition, to avoid - as far as possible - hard-coding of signal names and configuration in the LCU module itself.

5.4 Safety

The SW shall switch off the HW immediately when any interlock signal is asserted. Recovery from an interlock condition can only be done by operator intervention.

Protection against damage of equipment shall be implemented in HW, e.g. by means of:

- limit-switches, overheat-protection
- current limiting circuits
- over-voltage-protection
- galvanic isolation of external I/F lines
- shielding of wires on both ends
- electrostatic discharge protection at inputs.

5.5 Security

The CCS checking mechanisms (booking and command checks) are used to verify that received requests are authorized and valid.

A backup of the following file-systems has to be done regularly:

- All environment directories under *$VLTDATA*.
- All data files to be archived.

The backup must be performed daily, weekly and monthly.

The ASM WS database can normally not be regenerated from archived modules. An 'unload branch' operation has to be done regularly, in order to be able to restore the DB in case of corrupted data.

5.6 Adaptability and Enhancement Potential

The extension of the ASM SW with additional data SW-units is specified in section 3.1.

The adaption of the existing ASMs (DIMM2) to the same design as specified here can be considered.

5.7 Training

Since the ASM software is normally not accessed through TCS, there will be the need for special training. A course for operators and maintenance personnel based on the user-manual has to be considered, after the complete HW and SW is available.

5.8 Documentation

This is the foreseen set of documents for the ASM SW:

1. Functional Specification: this document
2. Design Description: detailed description of implementation
3. Test Plan: procedure of on-site verification tests
4. Test Report: verification test results
5. User Manual: for users of the ASM provided public data
6. Maintenance Manual: to describe operation, maintenance and configuration of the ASM

6 DEVELOPMENT AND TEST FACTORS

6.1 Design and Development Considerations

The design of the ASM software is closely related to those of the hardware of the controlled equipment. Since HW and SW development goes in parallel, the design and the implementation of the ASM software shall localize hardware dependencies in restricted areas where it can be modified in case of hardware changes.

Implementation shall, whenever it is possible, make use of standard common software (LCC and CCS). It shall also follow what is defined in the programming standards document [6].

6.2 Planning

The following phases are defined for the first release, incl. tentative schedule:

1. Analysis, resulting in the present document, and therefore terminated.
2. Oct. 1996 - Dec. 1996: Design of the modules indicated in section 3.2 with priority high or medium. All modules with low priority will not be realized during phase 1. Final clarification for interfaces with TCS, INS, and DMD.
3. Dec. 1996 - Feb. 1997: Implementation of the modules.
4. Jan. 1997: Verification of pointing accuracy of drive mechanics with special test-setup in Garching.
5. Mar. 1997 - June 1997: Test setup, integration and test of the modules in Garching.
6. June/July 1997: Installation of the HW, integration and test of DIMM-related modules in La Silla.
7. July 1997 - Sept. 1997: Comparison with present La Silla DIMM.
8. Oct. 1997: Transportation to Paranal, installation

For the integration period at La Silla only the telescope with CCD equipment will be available, but **not** the enclosure (which will be directly installed on Paranal), meteo-station, etc.

6.3 Development Requirements

1. LCU equipped as described in section 8.1.4
2. A development workstation with LAN interface to connect the LCU
3. X11 terminal(s)
4. LCC software for the LCU
5. CCS software for the WS (including one Rtap license)
6. CCD detector control SW
7. TCS software

6.4 Test requirements

Additionally to the development requirements:

1. WS with CCS/LCC/CCD/TCS SW installed, incl. Rtap licenses
2. Two LCUs in total, for telescope control and CCD.
3. One set of CCD equipment
4. One DIMM equivalent motor setup (amplifier, motor, encoder, gearbox)
5. The portable X-terminal for test of user interfaces (particularly concerning screen-size)

7 HARDWARE REQUIREMENTS

This chapter contains requirements from SW side. They shall be taken into account for the ASM hardware units that have still to be designed or modified, respectively that will have to be purchased.

7.1 Telescope Enclosure

The DIMM enclosure [18] has to be designed for the ASM. It consists of two halves (north/south) that can be opened independently. After a first approach using digital I/O for control from the LCU, the concept has been changed to use a serial line, since this would give advantages with respect to long cabling.

The requirements from SW-side are now:

- RS232 standard connection between LCU and enclosure controller:
  - 9600 baud, 8 bits, 1 stop, no parity (recommended, is configurable on LCU-side)
  - hardware handshake (dto.)

  Details about cabling shall be included into the hardware design [15].
- At the limits the movement shall stop by itself without need of any serial line command.
- Only 'human-readable' ASCII characters shall be exchanged on the serial line. It shall be possible to operate the enclosure also with a standard text terminal.
- The serial line protocol shall be as simple as possible.
- Commands to be supported via serial line are:
  - **Move** halves to given position
  - **Stop** movement at current position
  - **Get** current positions and whole status
  - Get version ID (optional)

  White space characters, i.e. space, tab, and line-feed, shall be handled as separators, and carriage-return as command/reply terminators.

  The detailed serial protocol shall be included into the ASM software design description.
- Exchanged position values shall be in integer degrees.
- Exchanged status values shall be 0 (negated) or 1 (asserted).
- All serial commands, whether legal or not, shall be acknowledged by a reply within 0.5 seconds. Further replies shall **not** be sent, e.g. when the target position is reached, since this is evident by the get-status command.
- Nothing further to these replies shall be sent by the enclosure controller.
- A **daylight sensor** shall exist that closes the enclosure automatically without being initiated via serial line. This is foreseen as hardware protection in case of stalled software or serial line failure. A corresponding status indication shall be available via serial line (as additional information, i.e. the LCU SW could work without it).
- A **Local-Mode switch** shall exist to select between automatic and handset control. In local-mode the enclosure can only be moved by an operator directly with push-buttons. Serial-line *Move* and *Stop* commands shall be rejected. A local-mode status indication shall be provided via serial line.

7.2 Telescope Structure

Resolvers are not supported, since they would require special I/F boards that are not foreseen anywhere in the VLT environment. The existing resolver will therefore have to be replaced by optical encoders. The VLT-standard solution is the **IK320** encoder I/F board. This provides high (possibly for the mechanical structure even too high) resolution with sin/cos-type incremental or absolute encoders. This board shall be used.

At the location of the present ‘fast’ resolver, i.e. between the reduction stages, an optical encoder shall be mounted and used for telescope positioning. An absolute reference is not foreseen, so that the encoder will have to be initialized by moving to the limit-switches (detailed sequence TBD).

Details about encoder type and accuracy are specified in [15].

An external drive amplifier with analog speed reference and built-in velocity control shall be used. Only the position control loop shall be implemented in SW. The velocity loop shall automatically be closed at power-up. No sensitivity switching shall be used. No brake control is foreseen.

7.3 Seeing Monitor

Standard TCCD equipment [12] shall be used. The foreseen pixel sizes for VLT-standard TCCDs are 386 x 290 or 578 x 578, each with 22 µm pixels. The small version is to be preferred for the seeing monitor CCD. No separate shutter is included.

7.4 All-Sky Monitor

Standard TCCD equipment [12] with a fish-eye lens shall be used.

7.5 Seismic Monitor

The unit is regarded as an independent black box.

All data shall be transferred exclusively via serial RS232 line with the parameters:

- 9600 Baud, 8 bit, no parity, one stop ***Preferably the same as the meteo-station. TBC***

The data shall be transferred periodically (not faster than every minute) as a block in ASCII format (plain text). The block shall optionally consist of several lines separated by <CR><LF>. The block shall have a special terminating character, preferably ‘=’. Section 8.2.2 shows the sample format used by the existing meteo-station, which shall be used as guideline.

7.6 Dust Monitor

Same requirements as section 7.5.

7.7 Infrared Emissivity Sensor

Same requirements as section 7.5.

8 APPENDIX

8.1 Hardware Overview

8.1.1 Hardware Location at Paranal

A separate control room for the ASM is foreseen, which is located near the telescope tower and sensor facilities. LCU(s) will be located either in this room or directly at the tower, while the ASM WS is placed in the main VLT control room. The distance between LCUs and sensor equipment must not be larger than 20 m, particularly encoder cabling is limited to this length.

8.1.2 Existing Hardware of the La Silla DIMM

The hardware components listed below are already existing, and shall - if possible - be re-used for the new design or be purchased in same version for the VLT-ASM. Their detailed description is in [2] and [7].

- Telescope (made by AMOS):

  Mechanics and optics: alt/alt mounting (eta/theta axis), 36 cm M1, 5 cm M2.

  Drives: motors with transmissions, type AXEM F12M2 from GEC-Alsthom/Parvex;
  rated: I_n = 11.7 A, U_n = 24 V, P_n = 190 W.

  Angle sensors: two resolvers per axis, one coupled to the main axis (*'slow resolver'*), one between the two stages of reduction (*'fast resolver'*), both to be replaced.
- Meteorological Station:

  black-box with corresponding sensors, data transfer via serial RS232 line.

8.1.3 Control Hardware Upgrade for Paranal

This includes exchange of existing HW with improved or more maintainable VLT-standard parts, and addition of new equipment.

1. One HP-UX 10 Workstation, equipped with:
   a. Monitor with at least 20-inch, 1200x1024 resolution
   b. hard-disk space to store programs and generated data-files
   c. LAN interface to LCUs used for the ASM, TCS software, archive, and external access
   d. Backup media (e.g. DAT drive), unless backup is done via LAN
2. One portable X-terminal (e.g. laptop) for maintenance operations directly at the hardware:
   a. LCD screen with at least 800x600 pixels
   b. X-server software
   c. LAN interface
3. Local Control Unit(s)
   a. general VMEbus racks with CPU boards etc.
   b. I/F unit(s) to telescope encoders, drives
   c. I/F unit to telescope enclosure
   d. I/F units to CCD cameras
   e. serial I/F to meteo station and new sensor boxes
   f. LAN I/F to ASM workstation
   g. Time interface(s) to TRS
   h. power supply
4. Telescope Enclosure

   a. serial RS232 control line
   b. local/automatic switch with corresponding status signal
   c. additionally one daylight sensor status signal, which enforces in automatic mode HW-wise the closing of the enclosure
5. Additional telescope parts
   a. new telescope encoders on main axis for position loop
   b. new drive amplifiers and tachos for velocity loop (to be attached to motor axis)
6. Instrumentation and sensor parts
   a. Instrument for the Differential Image Motion Monitor (DIMM):
      Consisting of one (or two) cooled TCCD camera(s):
      seeing monitor camera with a 6x6 arcmin field,
      wide field camera with a 10x10 degree field (only necessary for star-finder purposes, i.e. redundant if pointing accuracy is better than the field of the seeing monitor camera)
   b. All-sky monitor instrumentation ***TBD***
   c. Seismic Monitor: black-box with one serial RS232 interface
   d. Dust Monitor: black-box with one serial RS232 interface
   e. Infrared Emissivity Sensor: black-box with one serial RS232 interface

8.1.4 Projected Hardware Setup for the VLT-ASM

Chapter 7 presents detailed requirements for the hardware units to be designed, as seen from the software side.

For the control LCUs, hardware shall be chosen exclusively from the VLT-standard equipment:

- MVME167 CPU board from Motorola
  http://www.mot.com/GSS/MCG/products/boards/vme/mvme167/mvme167.html
- PPC CPU board from Motorola ***TBD***
  http://www.mot.com/GSS/MCG/products/boards/powerpc/MVME1600.html
- Digital I/O board AVME948X from ACROMAG (64 I/O bits)
- Analog I/O board VMIVME-3111 from VMIC (2 DAC, 16 ADC channels)
  http://www.vmic.com/catalog/section_headings/analog-input-boards/vme-3111.html
- Encoder I/F board IK320 from Heidenhain (two counter channels)
- Serial I/F board ISER8 from ESD (eight serial RS232/V.24 channels)
- ESO Time Interface Module board
- External amplifier with built-in analog velocity control
- Optical encoders from Heidenhain

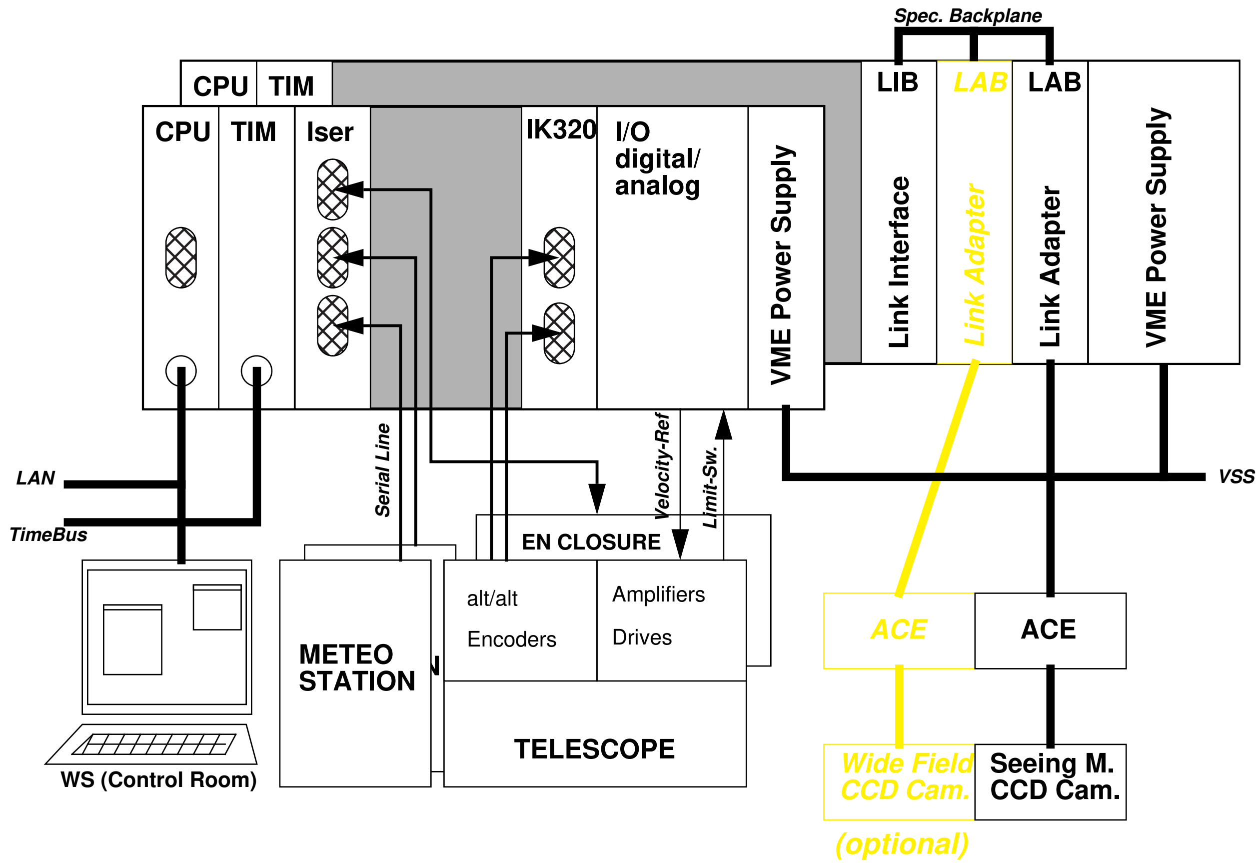
In lack of an absolute reference, the encoder will have to be initialized by moving to a limit-switch.

Detailed hardware specifications are contained in [15].

The most probable HW setup is listed here as example. Depending on the requirements of the mechanics it may change accordingly.

TBC: Can the pointing accuracy be high enough to omit the wide field CCD camera?
Although not entirely clear yet, it is assumed here that it can be omitted; however, as "emergency solution" it remains in this specification.

Fig. 10 Projected Hardware Setup



LCU for telescope control and data units:

- CPU board (sufficiently performant for two servo loops)
- TIM board
- Serial I/F board for enclosure and sensor stations (up to 8 channels)
- IK320 encoder I/F board (2 channels)
- Analog I/O board (amplifier velocity reference)
- Digital I/O board (limit-switches etc.)
- VMEbus and motor power supply

LCU for seeing monitor camera:

- CPU board with sufficient memory to serve one CCD (approximately 16 MB)
- TIM board
- one Link I/F board
- one Link Adapter boards
- ACE/CCD power supply
- split 32/16-bit VMEbus backplane (32-bit for CCD boards)

8.2 File Formats

The following sections show the file formats that are currently generated by the La Silla DIMM software in examples.

8.2.1 dimm.last

```
13 Jun 1996#07:48:42:  Current seeing=0.00 arcsec at 0.5 mu (Airmass 1.18)<CR><LF>
```

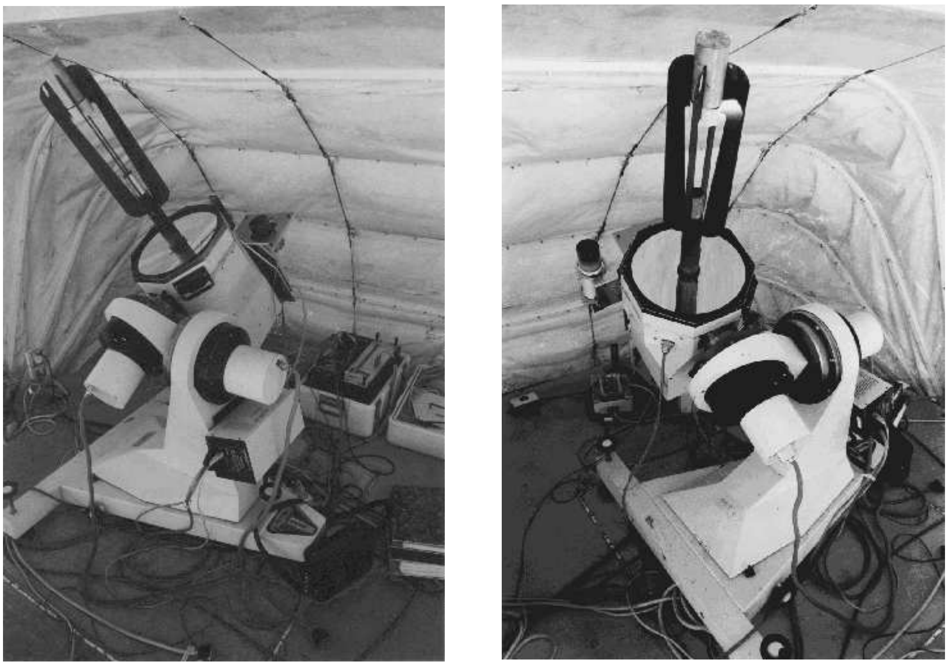
8.2.2 meteo.last

```
<CR><LF>
1996-06-13  08:12 LASILLA1<CR><LF>
       INTV   INST    AVE    MAX    MIN    DEV<CR><LF>
T 1 C   20M    8.1    7.8    8.0    7.6    0.1<CR><LF>
T 2 C   20M   11.3   11.4   11.5   11.3    0.0<CR><LF>
T 3 C   20M    8.3    8.4    8.7    8.0    0.2<CR><LF>
RH  %   20M     23     23     23     23      0<CR><LF>
TD  C   20M  -11.6  -12.0  -11.8  -12.1    0.1<CR><LF>
P   MB  20M  770.2  770.4  770.6  770.2    0.1<CR><LF>
QNH MB  20M 1024.9 1025.1 1025.4 1024.9    0.1<CR><LF>
WS1 M/S 20M   15.7   16.4   19.5   13.5    1.0<CR><LF>
WS2 M/S 20M   16.5   16.2   19.7   12.6    1.2<CR><LF>
WS3 M/S 20M   17.1   16.8   20.4   13.3    1.2<CR><LF>
WD1 DEG 20M     28     32     45     17      4<CR><LF>
WD2 DEG 20M     23     27     39     17      4<CR><LF>
WD3 DEG 20M     23     31     45     17      4<CR><LF>
=<CR><LF>
```

8.2.3 sky.last

```
13 Jun 1996^08:13:55Abs Ave= 9.990  Rms= 9.990 AM= 1.0876RA:22h8 ` Dec:-46D58<CR><LF>
```

Fig. 11 The DIMM telescope at La Silla



___oOo___