LBT-ADOPT
TECHNICAL REPORT

Doc.No : Soft-002
Version : 1.2
Date : October 2005

Integration of the AdOpt Software into TCS

Luca Fini, Alfio Puglisi

INAF - Osservatorio di Arcetri
http://lbtwww.arcetri.astro.it/adopt/

ABSTRACT

In the following memo we describe our view on some aspects related to the integration and interoperability of the Adaptive optics software within the LBT TCS software.

Contents

1 Introduction **2**
1.1 Framework . . . 2
1.2 Software Components . . . 3
1.3 Who, When, Where . . . 3

2 The AO Supervisor **4**
2.1 Operating states . . . 4
2.1.1 Standalone state . . . 4
2.1.2 Engineering state . . . 4
2.1.3 Observation state . . . 5
2.2 Human interaction . . . 5
2.3 Startup procedure . . . 6
2.4 Environment . . . 6

3 AOS Functionalities **7**
3.1 Querying TCS for information . . . 7
3.2 Information to be provided to TCS . . . 7
3.3 Commands to be issued to TCS . . . 8
3.4 Commands accepted from TCS . . . 8

4 AOS Implementation Details **9**
4.1 Communication between AO Supervisor and AOS . . . 9
4.2 Binocular Operation . . . 9
4.3 AOS Startup and Housekeeping Functions . . . 9

Glossary of terms and acronyms

AO System. The hardware and software components of the LBT first light Adaptive Optics System. Includes the Wavefront Sensor, the Adaptive Secondary Mirror, the AO Computer and some auxiliary devices (such as networking hardware).

AO-CI. The AO Software Command Interpreter: a component of AO-SW which can execute scripts.

AO Computer. The computer (or farm of computers) running the AO-SW.

AO Console. The operator console of the AO Computer.

AO-SS. the script executed to startup the AO-SW.

AO-SW. The software dedicated to the managements of the Adaptive Optics System. Its main component is the Supervisor.

AOS. A part (subsystem) of TCS dedicated to interaction with the AO-SW.

AO Supervisor. The software system which manages all the components of the AO System

TCS. Telescope Control System. The software dedicated to the management of the LBT telescope.

TCS Computer. The Computer (or farm of computers) running the TCS.

1 Introduction

1.1 Framework

Many aspects of the proposed solutions for the integration of the Adaptive Optics software (AO-SW, in the following) within the TCS are based on a few key points which have driven the design of the software.

- The AO-SW should be able to run as a standalone system with no need of support from any other software package (i.e.: the TCS, instrument related software and the like).
- The AO-SW, except for the interface to the TCS quoted below, should be able to compile with no need of external libraries (e.g.: the LBT common software, instrument related software libraries and the like).
- The AO-SW will run on a dedicated server. It will not participate to the TCS dynamic hardware allocation mechanism. Hardware allocation (e.g.: the server on which the AO-SW runs on), anyway, will not be completely static (i.e.: coded into software), but will be indicated by some (possibly, a single) configuration parameter. Changing server configuration will require the modification of relevant parameters and restart of the AO-SW.
- The AO-SW will have a dedicated console (display/keyboard/mouse) for engineering operations (see below). The AO console will be separated from other telescope consoles and will not be actively used during observations (see also user interface discussion below).
- Interaction between AO-SW and the TCS is provided by a TCS Subsystem: the Adaptive Optics Subsystem (AOS). This is a standard subsystem running in the TCS environment, integrated in the TCS and developed according to standard TCS rules.

  The AOS will communicate with the AO-SW via the AO-SW IPC mechanisms, available through the `msglib`/`rtdblib` libraries which are part of the AO-SW.

  All dependencies on LBT Common SW are concentrated in this software subsystem. The AO-SW is designed so that it can operate without the support of the AOS (except for TCS related requests).
- The AOS is made up of two symmetric sides named, accordingly to TCS naming rules, AOSL and AOSR. For the first light system only the AOSL will be considered.

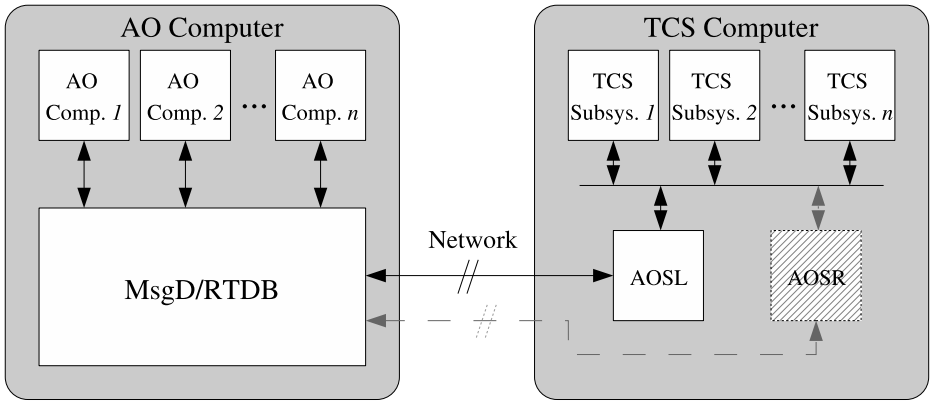

Figure 1: Interaction between the TCS and the AO Software

1.2 Software Components

For a better understanding of the interaction between the AO System and the rest of the TCS, we will refer to figure 1.

The AOS is the subsystem of the TCS dedicated to interaction with the AO-SW. All operational requests from the TCS to the AO Subsystems and from the AO-SW to the Telescope will be managed by this subsystem.

From the O.S. point of view it is a process running on the TCS Computer which is integrated with the rest of the TCS and communicates with the AO-SW through the network.

In the following pages we will describe in deeper detail many functional aspects of the AOS and its interaction with the AO-SW.

1.3 Who, When, Where

In this paragraph we try a first assessment of responsibilities (who's doing what), times, and logistics. The main purpose of this paragraph is to clarify all the required steps needed to develop, test and install at the telescope the AO-SW and mainly the AOS and to give a time estimate of various steps.

1. AOS design and implementation is responsibility of the Arcetri software team in strict coordination with the TCS group. A replica of the source code tree of TCS will be maintained at Arcetri and AOS related modules will be developed there.

2. During the development phase two integration tests to be performed in Tucson will be needed. The first one is scheduled for mid November 2005, and will be devoted to identify and solve software issues, such as: correct use of API's, correct use of TCS provided software facilities, interaction between the two software teams, problems resulting from O.S. environment, and the like. This will be performed coding a bare bone version of the AOS which can address the basic aspect of interfacing the AO-SW with the TCS.

3. After this test a prototype AOS will be developed in Arcetri. The prototype AOS will include all functionalities required for the operation of the first light LBT AO System.

4. A final integration test to be scheduled in 3rd qt. 2006 will be performed in order to verify the correct operation of the AOS prototype from a software point of view. This test will be performed against the TCS version running at the telescope and will exercise as extensively as possible all functionalities not involving the AO System hardware.

5. During the end of the first Solar Tower test for the AO System, a working version of the TCS will be installed in Arcetri in order to check as much as possible TCS related functions against the actual working AO hardware.

6. The AOS will then be finalized to version 0.9 and will be ready for delivery, completed with related documentation.

7. The final step will be performed during commissioning of the AO System at the telescope: global tests of interaction between TCS and the AO System are, in fact, only possible when the AO hardware is in place.

2 The AO Supervisor

The overall architecture of the AO Software and of its two components, the Supervisor and the Real-Time Software, have been described elsewhere [2, 3].

Here we will concentrate on some aspects of interest in relation to the interaction between the AO Supervisor and the TCS.

2.1 Operating states

The AO Supervisor will essentially provide Three operating states: STANDALONE, ENGINEERING and OBSERVATION.

2.1.1 Standalone state

In STANDALONE state the AO Supervisor system will not accept any interaction from the TCS except a status request. This state is provided in order to allow basic tests on the AO System, when we don't want interference from outside, and the TCS is operating on its own. The AOS must not attempt to perform any other query or request to the AO Supervisor; any query or request will be, anyway, simply ignored by AO Supervisor.

2.1.2 Engineering state

The ENGINEERING state is used for non-observation tasks (calibration, maintenance, etc.) which need support from TCS for operating any telescope device or subsystem. When in ENGINEERING state the AO Supervisor will be under the control of the engineering user interfaces running on the AO console.

The AO Supervisor in ENGINEERING state may send commands to the AOS for requesting services (pointing, tracking, hexapode adjusting and the like). It will also possibly send log information to be registered in the TCS central logging facility.

The AOS will provide the requested services (after proper safety checks) and will return proper status information.

While in this state the AO Supervisor will obey to status requests and emergency commands issued by the AOS[1].

2.1.3 Observation state

The OBSERVATION state is entered in order to allow the TCS to control the AO system when an observation has to be performed. When in OBSERVATION state the AO Supervisor operates under the control of commands issued by the TCS through the AOS. Commands will be executed by the AO Supervisor (after proper security checks) and corresponding status information will be sent back.

While in OBSERVATION state the AO Supervisor may asynchronously send a limited set of requests to the AOS, e.g.: mode offload, emergency status notification, and the like. AOS must properly manage such requests.

The AO Supervisor will also periodically send to the TCS log information to be stored in the TCS central logging facility.

Note that while in OBSERVATION state the AO Supervisor may be set in a number of different operating modes (e.g.: diffraction limited mode, ready to close loop, closed loop suspended, etc.) by proper commands issued by the TCS. A description of all available operating modes is outside the scope of this memo and is dealt with elsewhere.

2.2 Human interaction

The AO Supervisor will interact with an "operator" (which may be either a maintenance engineer or a telescope operator, depending on system state) through proper user interfaces.

When in ENGINEERING state the operator will have available the engineering interface (ENG-IF), i.e.: a set of GUI's displayed on the AO Supervisor console. This interface will be only bused in ENGINEERING state to perform non-observation tasks, and will operate as "master"; i.e.: it will provide full access to AO System functions and will possibly send requests to the TCS. The ENG-IF will also be available in OBSERVATION state; in this latter case it will operate as "slave", i.e.: it will be essentially non-operational and will possibly show status and diagnostic information not intended for the astronomer or the telescope operator[2]. It will also provide something like an "emergency button" which will stop the OBSERVATION state and return to ENGINEERING state (this function will possibly be password protected to avoid unauthorized use), this will be done after suitable handshaking with the AOS.

When in OBSERVATION state, the AOS will provide an Adaptive Optics operator interface, completely integrated within the TCS GUI system [1] to allow the telescope operator, or the astronomer, to select adaptive optics modes, provide required parameters, and possibly to show a few status indicators (current operating mode, image quality, and the like).

[1] When doing non-observation operations, an "emergency shutdown" request from the TCS (see section 3.3) will not be automatically executed, but will provide a suitable prompt on the AO console.

[2] This function will allow an "adaptive optics" engineer to monitor the AO System operating and performances during actual observation in order to provide information useful to identify problems and possible optimization paths during the commissioning phase.

2.3 Startup procedure

The AO Supervisor, in its final version at the telescope, will be started as the final step of the AO Computer bootstrap procedure and must became operational with no need of human intervention[3].

The AO Supervisor startup sequence will include the following steps:

1. The Command Interpreter (AO-CI) is started
2. The AO-CI starts the minimum number of AO Supervisor components needed at startup.
3. The AO-CI executes a startup script (AO-SS). More than one startup script will be available in order to select different startup sequences (e.g.: to be used during commissioning or for maintenance tasks); a default startup procedure will be automatically executed when the system is in normal operating configuration. As an example we briefly describe below the default startup procedure.

When activated at startup the default AO Supervisor startup script will perform the following steps:

a) The AO-SS starts the software components needed for minimal operation (typically a few device controllers).

b) The AO-SS operates the power switches of hardware components needed for minimal operation.

c) The AO-SS performs software security checks (network, system, etc.)

d) The AO-SS sends commands to device controllers to perform hardware initialization.

e) The AO-SS sends commands to device controllers to perform hardware security checks.

f) The AO-SS starts the engineering interface in slave mode and waits for connection from the AOS (see section 4.3). At this point an operator may stop the automatic procedure and force ENGINEERING state by means of the ENG-IF on the AO Computer console.

All subsequent operations will be controlled either by the ENG-IF or by the AOS.

Alternate startup procedures will be provided for special purposes (e.g.: when we want that the system after startup goes to STANDALONE state).

2.4 Environment

Although the identification of the final running environment is not a critical issue[4], we are aware that at some point in the development we must fix such details as Linux kernel version, Linux distribution, etc. which will be used on the AO Computer at the telescope. This will be, most likely, the same as on the rest of the TCS CPU's for obvious reasons[5].

[3] This simply reflects the fact that the entire AO System must come up to a running condition with no need of human intervention.

[4] One of the design criteria has been the adoption of sound coding standards and devote some effort to design the system to be as much independent as possible on the O.S. version and details.

[5] This is desirable for system management and maintenance purposes, but it is not mandatory: AO Supervisor and TCS are actually interacting by means of a communication protocol: if needed the two system could run on completely different OS environments.

3 AOS Functionalities

In this section we will detail in an informal way the functionalities which must be implemented in the AOS.

In the selection of supported functions we started from a preliminary assumption: the need to avoid duplication of any functionality which is provided by the TCS itself. Consequently all commands to operate the telescope which must be issued by a human operator are assumed to be available at the TCS operator console; AO Supervisor must be able, instead, to issue commands which are needed to implement automatic procedures.

As an example AO Supervisor my need a command to request the TCS to offset telescope pointing in order to implement mapping procedures, but the AO engineering staff, when needed, will use the TCS operator console to select a proper astronomical object and point the telescope on it.

Most of the functions implemented in the AOS will be needed when in ENGINEERING state, while only a small subset will be operational when the AO Supervisor is in OBSERVATION state.

3.1 Querying TCS for information

The following pieces of information will be periodically updated within the AO Supervisor variable data base, so that any AO Supervisor component will find them available with no need to poll the AOS. They will be refreshed when needed or at a convenient rate by AOS.

- **Telescope Status**. This generic piece of information must specify the general status of the telescope. One of the statuses must be: **AdOptEngineering**, indicating that the AO Subsystem can proceed with engineering operations.
- **Telescope Pointing Status**. This says whether the telescope is stopped or is tracking, and provides pointing coordinates, both in mechanical and in sky coordinates.
- **Instrument Status**. This indicates which instrument is on-line and the status of the instrument.
- **Derotator Status**. Current status of derotator, including mechanical position.
- **Hexapode Status**. Current status of hexapode, including mechanical position (a suitable reference system for the hexapode must be established).
- **Environment Data**. Temperature, humidity, air pressure in the dome, wind speed and direction, etc.

3.2 Information to be provided to TCS

- **AO status**. Current status of AO System. Possible values are: NOT READY, STANDALONE, ENGINEERING, OBSERVATION, and the like. The AOS will periodically poll this value so that it will also act as a watchdog mechanism.
- **Observation state mode**. When in OBSERVATION state this item provides further details about operating mode or status of individual devices. E.g.: **Flat Mirror**, WFS Ready, **Failure**, etc.
- **Image Quality**. When in OBSERVATION state some, yet to be defined, image quality indicator will be provided to the observer's console.

3.3 Commands to be issued to TCS

During operation the AO Supervisor may need to issue a number of operational commands to the TCS.

- **Move Hexapode**. A request to adjust the position of secondary mirror.
- **Rotate Derotator**. A request to adjust the position of the derotator.
- **Pointing offset**. A request to offset current telescope pointing.
- **Emergency Shutdown**. This indicates that the AO Supervisor has detected some dangerous condition and is autonomously deciding to shutdown.
- **Offload Modes**. A request to TCS to deal with low-order modes adjustment.
- **Log Item**. Store a log record into TCS main history log.

3.4 Commands accepted from TCS

The following commands will be accepted by the AO Supervisor, subject to proper preconditions, as detailed in the command descriptions.

- **Emergency Shutdown**. This is a request from TCS to shutdown everything; shutdown must be done as fast as compatible with safety rules. After an emergency shutdown the system is considered in critical status and it might be required personnel intervention to restore it to a safe status. Precondition: none (its an emergency!).
- **Prepare**. This is a request from TCS to set up everything for closing the loop. TCS will provide all the necessary pieces of information needed to acquire the reference star and to set up loop parameters. The request will provide further specifications for selecting specific operating modes. Precondition: the startup procedure has been completed properly and the system is operational.
- **Start**. Start observation. Depending on current operating mode it may require to close the adaptive loop. Precondition: the set up procedure has been completed properly and the system is ready.
- **Stop**. The observation has finished. The AO System must next be ready to be prepared for another observation or to receive another Start command. Depending on current operating mode the Stop command may require to open the adaptive optics loop. Precondition: the loop is currently closed. The AO System should go back to a ready status.
- **Set Tip-Tilt**. Set mirror Tip-Tilt; this is an example of command which can be used to allow some level of control of the adaptive mirror from some external subsystem (such as an instrument). Precondition: the AO System is in "externally controlled tip-tilt" status.
- **Reset**. Go back to a rest status. Precondition: the AO System is in a suitable status (E.g.: not in Error or the like). The status is such that the hardware is safe, and the system is accepting commands from the TCS to know what to do next.
- **Shutdown**. Stop operations and go back to a safe status. Precondition: the AO System is in any non critical status. After shutdown the AO System orderly shuts down all hardware and becomes ready for poweroff (or for a warm restart from the AO Console).

4 AOS Implementation Details

In this section we will discuss a few details related to AOS implementation and some functional aspects which are not directly related to the AO System operations.

4.1 Communication between AO Supervisor and AOS

The AOS communicates with the AO Supervisor via a message based low level protocol which is the basic IPC mechanism of the AO Supervisor itself. The support for this protocol is provided by two libraries: `msglib`, which supports the basic message based protocol; and `rtdblib` which implements an API to the RTDB (the Real-Time Database), a software facility to manage a set of global variables.

From a system point of view the AOS will open a network connection to the AO Supervisor main process (named: `MsgD/RTDB`, see figure 1) which supports message exchange and the RTDB.

4.2 Binocular Operation

Although the first light system will only provide a single Adaptive Optics system, the AOS architecture must take into account the final binocular mode.

This is accomplished in the same way as other TCS subsystems, by running a left and a right instances of the AOS, i.e.: AOSL and AOSR (see figure 1). On the AO Supervisor side, although most of the components will be also duplicated, there is a single `MsgD/RTDB` process which coordinates left and right sides of the whole Adaptive Optics System.

4.3 AOS Startup and Housekeeping Functions

At startup the AOS will perform the following operations:

- Startup status is: "Waiting for AO Supervisor"
- Wait for `MsgD/RTDB` to come up trying to open a connection. An indefinite wait in this status must not affect operations of TCS in any way (except for Adaptive Optics related requests). Status of AOS will not change until the `MsgD/RTDB` accepts connection.
- Open a connection to `MsgD/RTDB`. Status changes to "Connected".
- Query for AO Supervisor status. Initial AO Supervisor status will be "Standalone", the AOS must wait for AO Supervisor to go to either ENGINEERING or OBSERVATION states before doing anything else.
- Activate AOS function according to AO Supervisor status.

References

[1] M. D. De La Peña, "LBT Telescope Control System GUI Guidelines". *LBT Observatory*, LBT CAN 481s010a.

[2] L. Fini, A. Puglisi, and A. Riccardi, "LBT-adopt control software," in *Advanced software, control, and communication systems for astronomy. Edited by L. Hilton and G. Raffi*, vol. 5496 of Proc. SPIE, pp. 528-537, 2004.

[3] R. Biasi, M. Andrighettoni, D. Pescoller, "LBT AO Real Time Software". Presentation at *AO Progress Report Meeting. February 2005*. See:
`http://lbtwww.arcetri.astro.it/adopt/lbtao/review-feb-2005/`

[4] T. Axelrod, "LBT Software System Definition". *LBT Observatory*, LBT CAN 481s001.