



General remarks

AO arbitrator FSM has been simplified by removing the on/off control of the AdSec and WFS subsystems. It is now expected that the two subsystems can be turned on and off independently and it is the responsibility of the T/O to ensure that the subsystems required for observation are properly set up (only the AdSec for seeing limited operation, both the AdSec and the WFS for AO observations).

- Each of the AO Arbitrator states defines a specific substate of the underlying AdSec and Wfs arbitrators, except for the first **Ready** state where the system is waiting for the preset command before checking the lower-level arbitrators' status.
- Most commands are implemented sending a corresponding command to one or both of the underlying arbitrators, and checking the return status

General use case of each command

1. Validate parameters
2. Send commands to AdSec and WFS arbitrators to do the work. Handle coordination between the two if needed.
3. If previous step is successful, report SUCCESS state
4. If an AdSec or WFS command fails with RETRY, rollback the other subsystem (if applicable), and report RETRY state
 1. If the rollback fails, go to step 5 using the rollbacked subsystem as failed subsystem.
 2. If the rollback gives RETRY, issue a Stop to both subsystems, handle any failures or retries with **RecoverFailure**.
5. If an AdSec or WFS command fails with FAILURE, send a Stop to the other subsystem (if applicable), and execute a

RecoverFailure on the failed subsystem.

6. If **RecoverFailure** successful, go back into **Ready** status, and notify TCS of error
7. If **RecoverFailure** is unsuccessful, go to **Unrecoverable Error** state.

Arbitrator states

AOArb state	AdSec state	WFS state
Ready	any	any
<u>ReadyToAcquire</u>	AOSet	AOPrepared
<u>RefAcquired</u>	AOSet	AOSet
<u>LoopClosed</u>	AORunning	<u>LoopClosed</u>
<u>LoopSuspended</u>	AOPaused	<u>LoopPaused</u>
Unrecoverable fault	undefined	undefined

In **Ready** state, in order to accept any command the AdSec state must be AOSet. In addition, to accept a **PresetAO** command the WFS state must be **Operating** or **AOPrepared**. This is done by the T/O turning on the WFS independently of this state machine.

Ready

System is waiting for a PresetFlat or PresetAO command. Upon receiving the command, the following checks are done:

PresetFlat: AdSec is in AOSet state

PresetAO: AdSec is in AOSet state and WFS is in Operating or AOPrepared state.

Thus AdSec and WFS states are set by the T/O using commands independent from this state machine.

ReadyToAcquire

AdSec: mirror set in a flat position.

WFS: configured for a specific AO guide star magnitude and position. Ready to acquire AO guide star as soon as telescope is tracking on the scientific object. AO guide star must be present on AO acquisition camera field of view, which has a 15 arcsec diameter. From this state on, pupil derotator is tracking telescope derotation.

RefAcquired

AdSec: mirror set in a flat position and configured for closed loop based on measures done during AO guide star acquisition.

WFS: configured for closed loop based on measures done during AO guide star acquisition. Wavefront signals calculated but not yet sent to mirror.

LoopClosed

AdSec: mirror in closed loop, receiving signals from WFS and moving to compensate atmospheric aberration.

WFS: wfs in closed loop. Wavefront signals sent to mirror.

LoopSuspended

AdSec: mirror stopped in last position before the suspension

WFS: Wavefront signals calculated but not sent to mirror.

Unrecoverable fault

An error occurred and system was unable to recover automatically from it. Operator intervention required.

Command use cases

PresetFlat

- AdSec: send command **LoadShape**
 - Error: try **RecoverFailure** on AdSec. If successful, go to **Ready** state.
- WFS: no action

ApplyZernikes

- AdSec: send command **ApplyModes**
 - Error: send **RecoverFailure** on AdSec. If successful, go to **Ready** state.
- WFS: no action

PresetAO

- AdSec: check that state is **AOSet**.
- WFS: send command **PrepareAcquireRef**.
 - Outcome RETRY: remain in current state (**Ready**)
 - Outcome ERROR: issue **RecoverFailure**. If successful, go to **Ready**.

CheckRefAO

- AdSec: no action
- WFS: send command **CheckRef**
 - Outcome RETRY: remain in current state (either **ReadyToAcquire** or **RefAcquired**)
 - Outcome ERROR: issue **RecoverFailure**. If successful, go to **Ready**.

AcquireRefAO

1. Send command **AcquireRef** to WFS
 - Outcome RETRY: remain in current state (**ReadyToAcquire**)
 - Outcome ERROR: issue **RecoverFailure**. If successful, go to **Ready**.
2. Send commands **SetRecMat** and **SetGain** to AdSec
 - Outcome RETRY: rollback WFS with **PrepareAcquireRef** with original parameters. If successful, remain in current state (**ReadyToAcquire**)
 - Outcome ERROR: send **Stop** to WFS. Send **RecoverFailure** to AdSec. If successful, go to **Ready**.

RefineAO

ModifyAO

StartAO

1. Send command **StartAO** to AdSec
 - Outcome RETRY: remain in current state (**RefAcquired**)
 - Outcome ERROR: send **Stop** to WFS. Send **RecoverFailure** to AdSec. If successful, go to **Ready**.
2. Send commands **StartAO** to WFS
 - Outcome RETRY: rollback AdSec with **Stop**, **SetRecMat** and **SetGain** restoring previous parameters.
 - Outcome ERROR: send **Stop** to AdSec. Send **RecoverFailure** to WFS. If successful, go to **Ready**.

CorrectModes

- Send command **CorrectModes** to WFS
 - Outcome RETRY: remain in current state (**LoopClosed**)
 - Outcome ERROR: send **Stop** to AdSec. Send **RecoverFailure** to WFS. If successful, go to **Ready**.
- AdSec: no action

OffsetXY

- Send command **OffsetXY** to WFS
 - Outcome RETRY: remain in current state (**LoopClosed**)
 - Outcome ERROR: send **Stop** to AdSec. Send **RecoverFailure** to WFS. If successful, go to **Ready**.
- AdSec: no action

OffsetZ

- Send command **OffsetZ** to WFS
 - Outcome RETRY: remain in current state (**LoopClosed**)
 - Outcome ERROR: send **Stop** to AdSec. Send **RecoverFailure** to WFS. If successful, go to **Ready**.
- AdSec: no action

Pause

1. Send command **PauseLoop** to WFS
 - Outcome RETRY: remain in current state (**LoopClosed**)
 - Outcome ERROR: send **StopAo** to AdSec. Issue **RecoverFailure** to WFS. If successful, go to **Ready**.
2. Send commands **PauseAo** to AdSec
 - Outcome RETRY: rollback Wfs with **ResumeLoop**. If successful, remain in current state (**LoopClosed**)
 - Outcome ERROR: send **Stop** to WFS. Send **RecoverFailure** to AdSec. If successful, go to **Ready**.

Resume

1. Send command **ResumeAo** to AdSec
 - Outcome RETRY: remain in current state (**LoopPaused**)
 - Outcome ERROR: send **StopLoop** to WFS. Issue **RecoverFailure** to AdSec. If successful, go to **Ready**.
2. Send commands **ResumeLoop** to WFS
 - Outcome RETRY: rollback AdSec with **PauseAo**. If successful, remain in current state (**LoopPaused**)
 - Outcome ERROR: send **Stop** to AdSec. Send **RecoverFailure** to WFS. If successful, go to **Ready**.

Stop

- If in **LoopClosed** or **LoopSuspended** state:
 1. Send command **StopLoop** to WFS
 - Outcome RETRY: remain in current state (**LoopClosed** or **LoopPaused**)
 - Outcome ERROR: succede in caso di WFS o BCU irraggiungibile. non si può stoppare l'adsec perché le slopes continuano ad arrivare. Rip per sicurezza?
 2. Send command **StopAO** to AdSec
 - Outcome RETRY: force a Failure on the AdSec and follow the ERROR outcome.
 - Outcome ERROR: send **Stop** to WFS. Send **RecoverFailure** to AdSec. If successful, go to **Ready**
- If in **Ready** state:
 - no action

- If in **ReadyToAcquire** or **RefAcquired state**:
 - WFS: send **Stop** command to WFS
 - Outcome RETRY: force a failure on the WFS and follow the ERROR outcome.
 - Outcome ERROR: send **RecoverFailure** to WFS.
 - AdSec: no action

Asynchronous errors

If an Alert from a lower-level arbitrator is received indicating a Failure of that Arbitrator (e.g. a ripped shell), the following actions will be taken:

1. Send a Stop command to the *other* arbitrator(s) if they are in any state excluding **PowerOff** or **Ready**.
2. Send a **RecoverFailure** command to the failed arbitrator
 - Outcome SUCCESS: go to **Ready** state.
 - Outcome RETRY/ERROR: go to **Unrecoverable Error** state because we cannot recover.

Attachments (2)

[Attach files](#)

[Show options](#)



[AOArbFsmCore_new.fsm](#) (42.67K)

Version 1 uploaded by Alfio Puglisi on 12 Jul 2010

- 14:14

fsm file



[AOArbFsmCore_new2.png](#) (60.69K)

Version 1 uploaded by Alfio Puglisi on 09 Jul 2010

- 16:24

[Select all](#)

[Edit](#) | [Attach](#) | [Print version](#) | [History: r7 < r6 < r5 < r4](#) | [Backlinks](#) | [View wiki text](#) | [Edit wiki text](#) | [More topic actions](#)

Topic revision: r7 - 16 Jun 2011, LucaFini

Copyright © by the contributing authors. All material on this collaboration platform is the property of the contributing authors. Ideas, requests, problems regarding Foswiki? Send feedback

