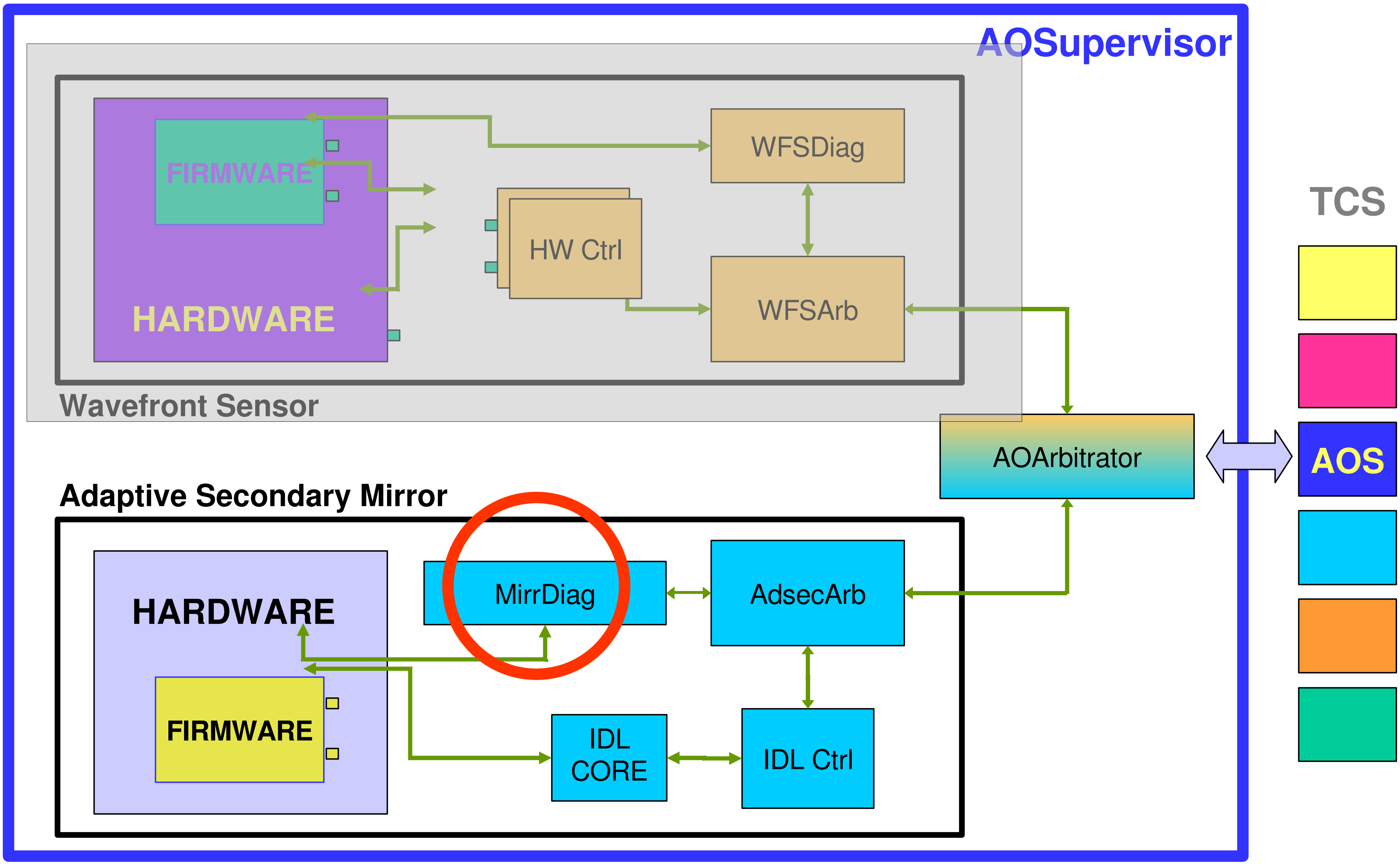
LBT Adaptive Secondary Diagnostic Software

L. Busoni

M.Xompero, D. Zanotti, A. Riccardi, A. Puglisi, L. Fini

Where?


AOSupervisor
FIRMWARE
HARDWARE
HW Ctrl
WFSDiag
WFSArb
Wavefront Sensor
TCS
AOArbitrator
AOS
Adaptive Secondary Mirror
HARDWARE
FIRMWARE
MirrDiag
AdsecArb
IDL CORE
IDL Ctrl

Target

- We want to keep under control all those parameters that can suggest the onset of a safety risk both for the electronics devices and for the glass shell.

- We want to promptly react when a dangerous situation is detected.

- We can use the diagnostic parameters provided by the control loop electronics to work out an on-line analysis of the quality of the AO loop.

Where do risks come from?

- Bad commands sent by the wave-front sensor.
- Dust grains between the thin shell and the reference plate can induce stresses above a critical threshold.
- Hardware malfunctioning (e.g. of capacitive sensors or DACs) can result in the application of arbitrary high forces to the shell.

Risks do not come from:

Not from turbulence: during “normal” functioning, shell shape corrections are not big enough to break the shell.

Diagnostic sensors and raw data

Slow

- Health status of electronics: boards temperatures, power supply currents and voltages, coils currents, etc...
- Periodically written on the on board memory and downloaded to the Supervisor.

RealTime

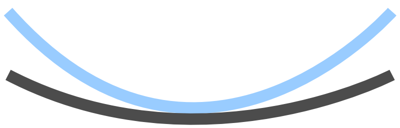
- Control loop parameters: distances measured by capacitive sensors, forces applied by coils, wave-front modal amplitudes measured by WFS, etc...
- Written at full rate (up to 1kHz) on the on-board SDRAM memory. Downloaded to the Supervisor through Gbit ethernet.

What can we monitor?

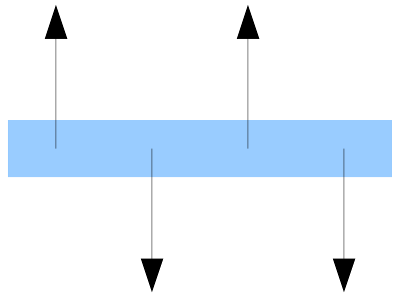
Obviously applied forces to prevent saturation of current drivers and shell position to prevent falling of the shell (!!) and other direct quantities.

Shell stress: from Finite Element Analysis and actuator positions we compute the stress pattern in the shell.

Correction residual: difference between the shape required by the wave-front sensor and the measured figure.



Forces distribution on the shell to prevent a checkerboard pattern of the applied forces



More coming soon .....

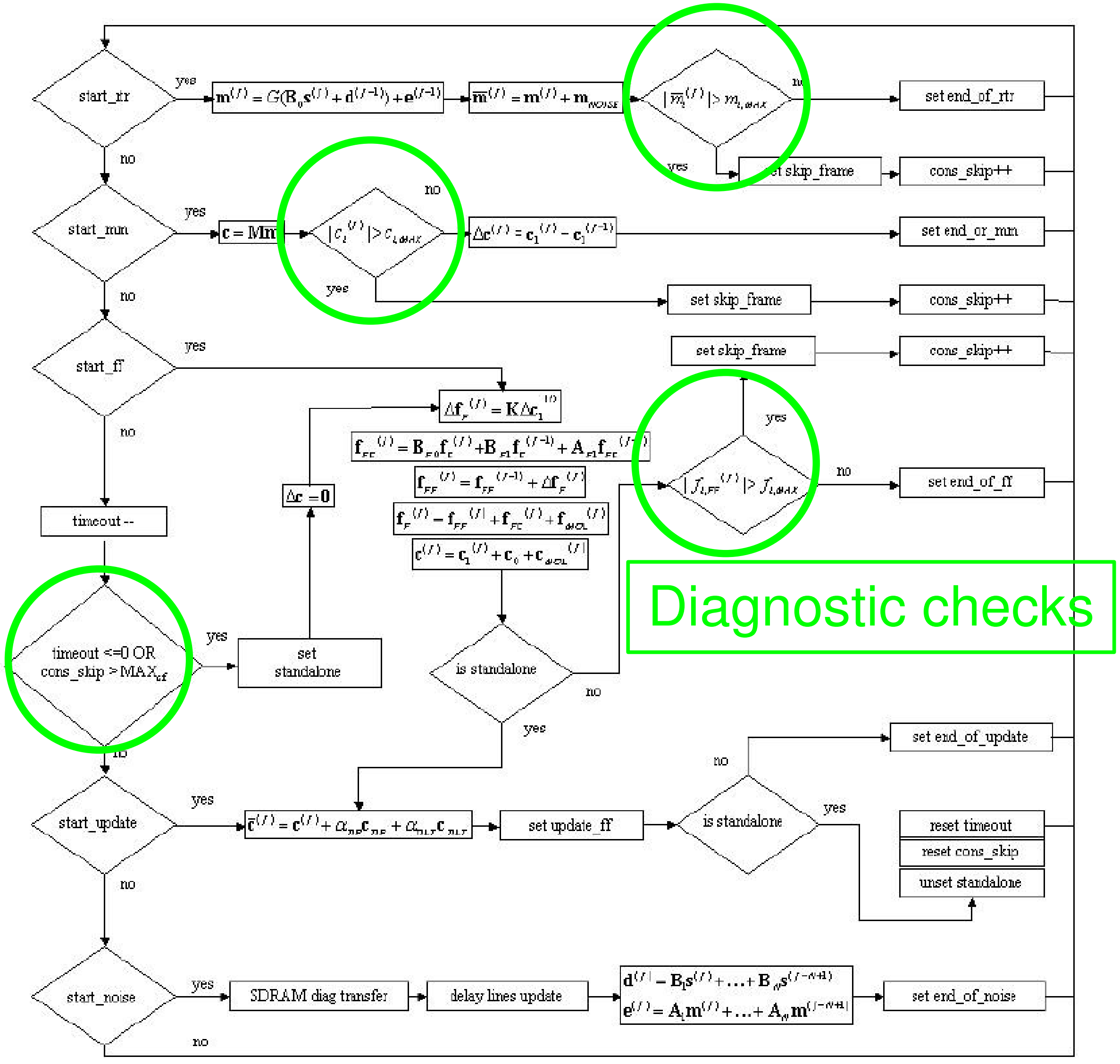
On board real-time diagnostic

A real-time diagnostic is implemented in the DSP code used for the control loop.
We can discard those frames that would result in too much high forces or stresses.

Pro: real-time. The “bad” frame is discarded without being applied.

Con: Limited memory and power on DSP. No temporal history.

DSP code is developed by Microgate

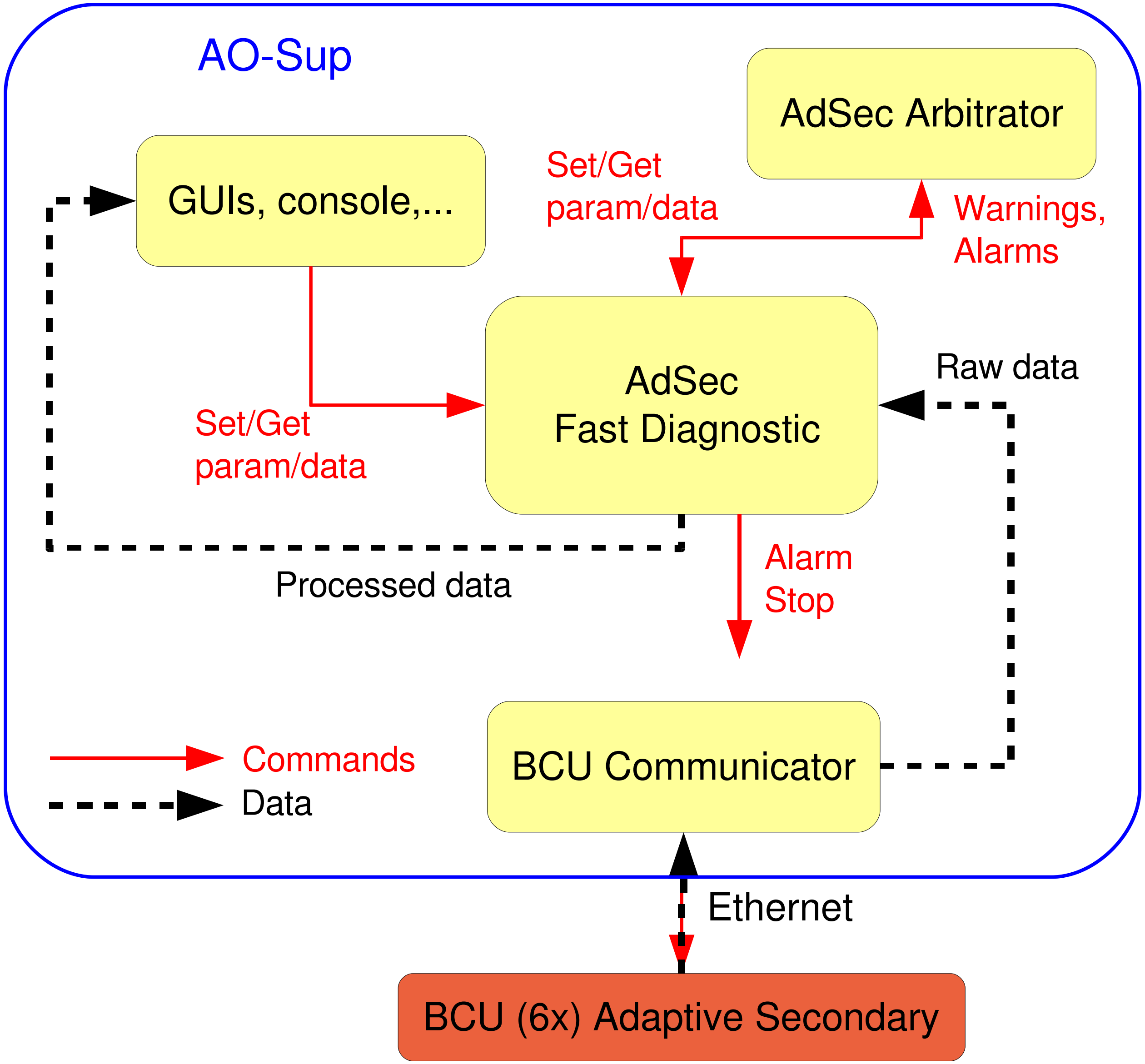


On-line analysis of diagnostic data

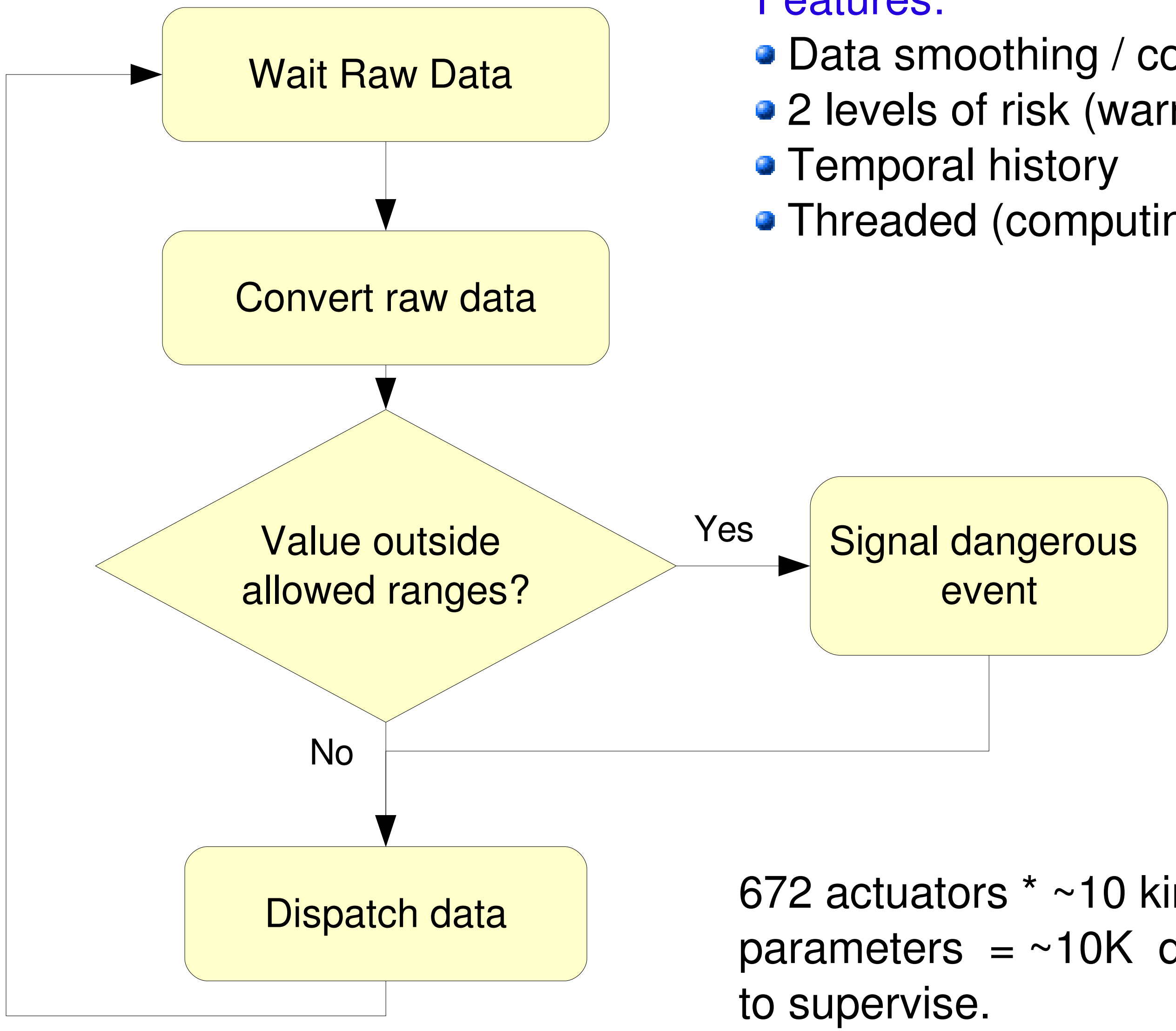
DSP code can't trap every dangerous situations before applying commands.

Need a “fast” software that analyzes a posteriori the diagnostic data, detects the onset of pathological conditions and reacts.

The delay of such an analysis must be < ~ 10ms.



Diagnostic Application (DiagnApp)



Features:

- Data smoothing / consecutive allowed faults
- 2 levels of risk (warning / alarm)
- Temporal history
- Threaded (computing shell stress can be long)

Change behavior at run-time:

- enable/disable a variable
- modify warn/alarm ranges
- modify smoothing filters
- configure time-history
- etc...

672 actuators * ~10 kinds of diagnostic parameters = ~10K diagnostic variables to supervise.

Derived from DiagnApp

- *fastdiagn* : monitor of the secondary shell
- *housekeeper* : monitor of the health status of electronics
- *AOdiagnostic* : monitor of the quality of optical loop