

The ELT Control System: recent developments

G. Chiozzi, L. Andolfato, J. Argomedo, N. Benes,
C. Diaz Cano, A. Hoffstadt Urrutia, M. Kiekebusch,
N. Kornweibel, U. Lampater, F. Pellegrin, M. Schilling,
B. Sedghi, H. Sommer, M. Suarez Valles

European Southern Observatory



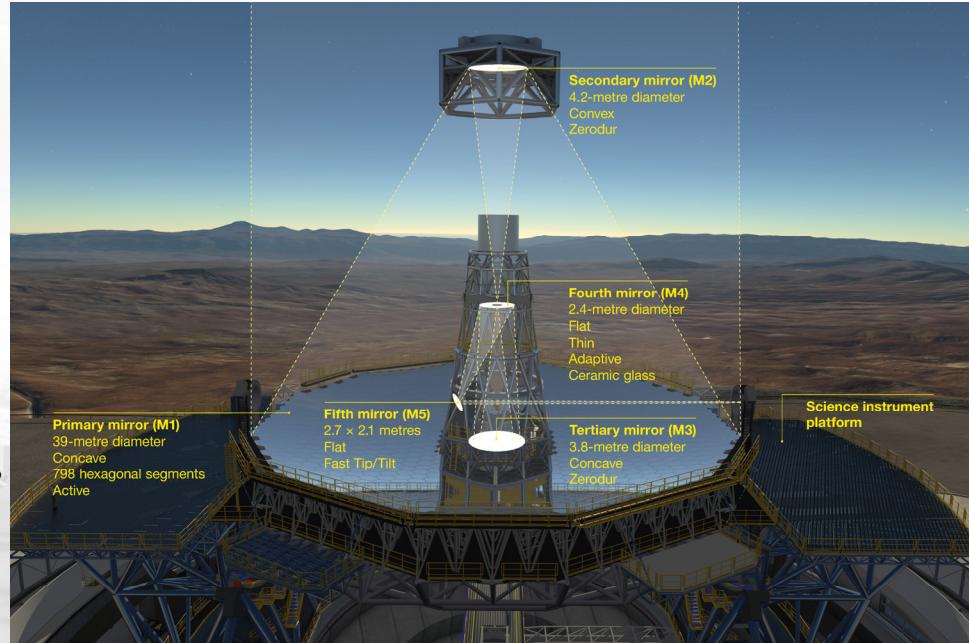


Outline

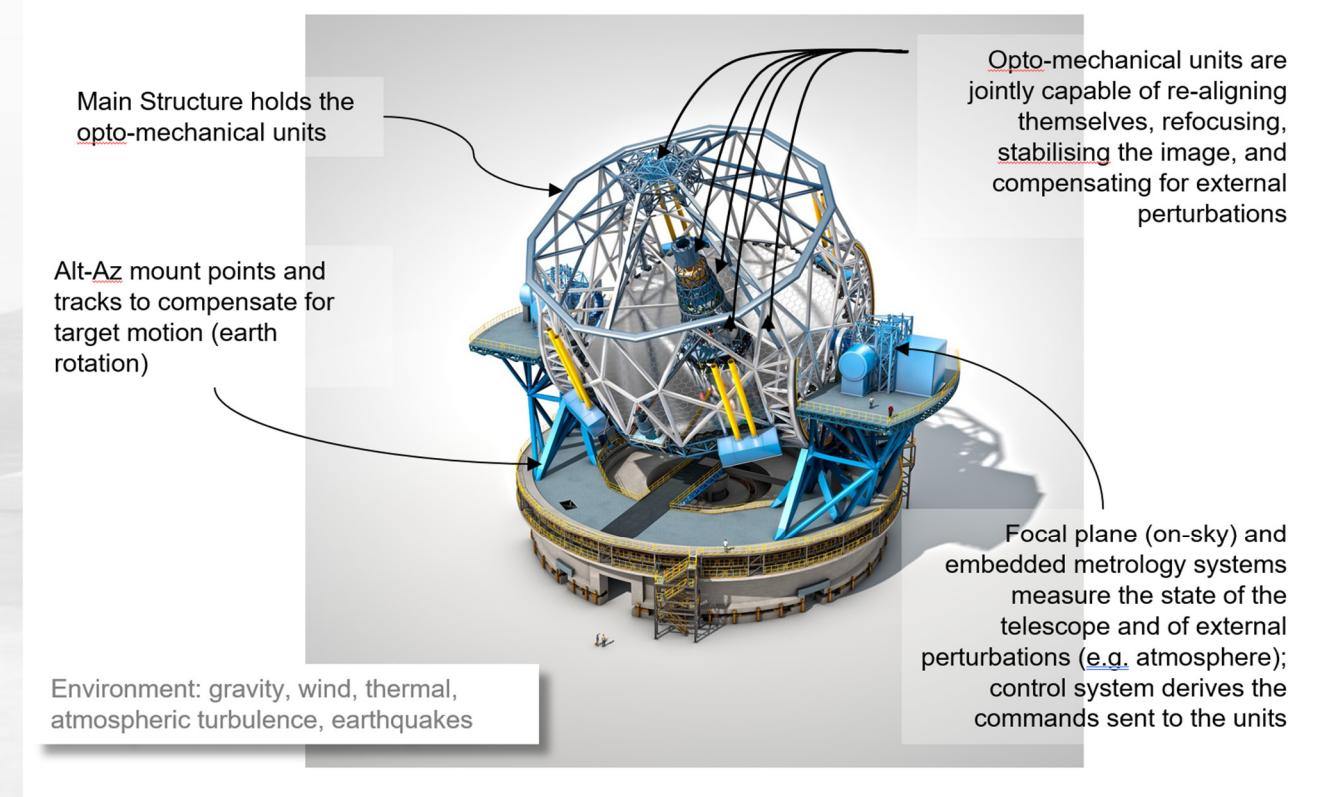
- The ELT
 - The ELT Control Strategy
 - Control System Architecture
 - Local Control Systems
 - Central Control System
 - Conclusion
-
- More on paper and references:
 - RTC and Adaptive Optics, The Minuscule ELT, Common Software Infrastructure

The ELT

- Large segmented telescope
- 5-mirror scheme
- 10000 tons,
25000 sensors,
15000 actuators
- M1 (39M):
800 1.4M segments
- M4 (2.5M):
5300 actuators
- 6 instruments
- Many distributed control loops: from 0.01Hz to kHz.
- Distributed control requiring synchronization to ms.



The ELT Telescope Structure



Control Strategy

■ Goal:

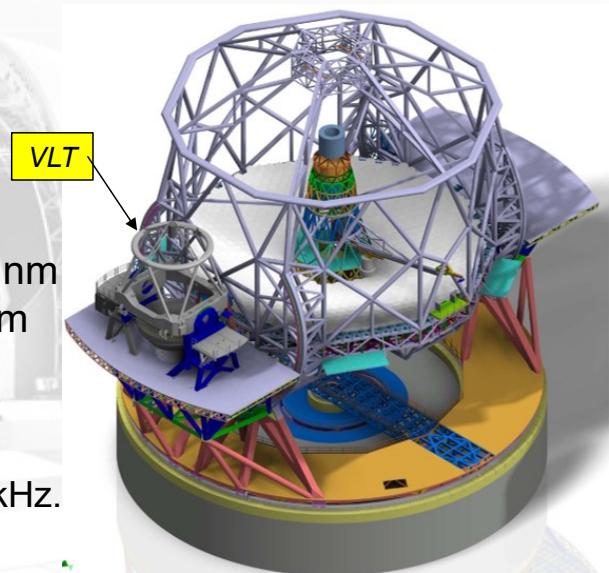
- diffraction limitable beam at each of the ELT Nasmyth foci

■ Challenge:

- keep wavefront within error $\sim 10\text{s}$ of nm with perturbations in the range of mm

■ Control

- Deformable M4:
 - on-sky loop closed at rates up to 1 kHz.
 - limited stroke (100um)
- Feed forward control during blind phases
 - brings telescope within the acquisition range of on-sky sensors
- Feedback loops based on telescope internal metrology
 - M1 Figure Loop keep deformations within the capture range of M4
- Background stroke management



ELT CS architecture drivers

Architecture drivers:

- Subsystems contracted to industry.
- Instruments developed by consortia.
- De-coupling of subsystems
- Flexible at AIV/Commissioning
- Obsolescence management

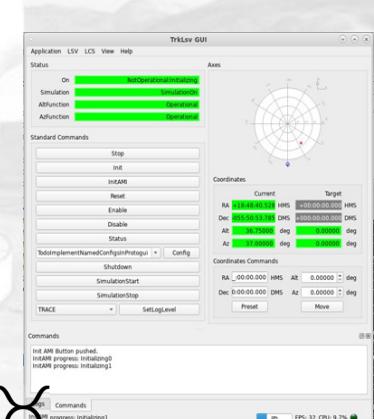


Foundation's status, June 2021



Some ELT standards

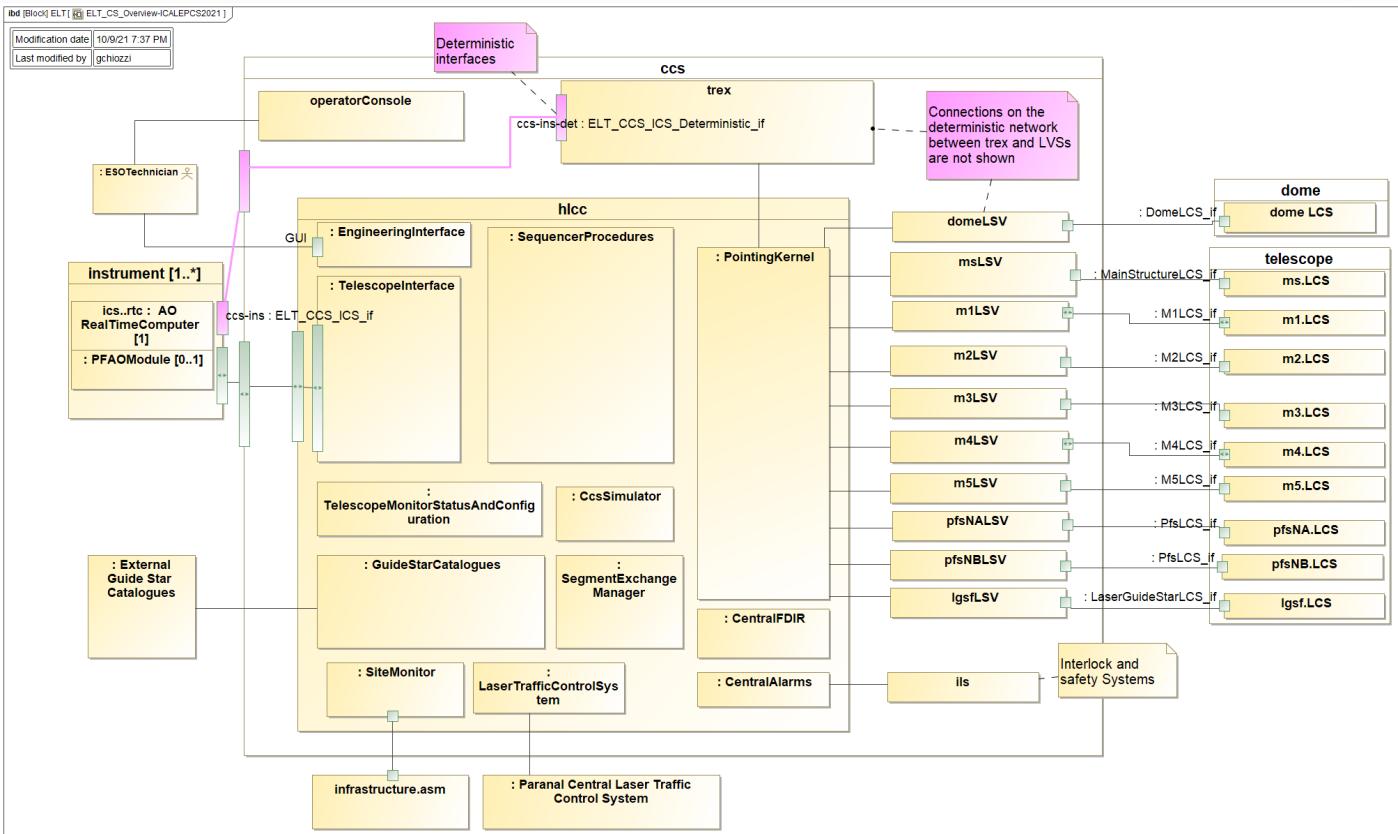
- Communication: Ethernet, EtherCAT, PROFINET, PROFISAFE, UDP/TCP.
- Middleware: OPC/UA, DDS, ZeroMQ.
- Time synchronization: PTP and NTP.
- Runtime platforms:
 - Linux CentOS/Linux RT for WS applications.
 - Beckhoff PLCs (TwinCAT), SIMATIC S7 for LCS software.
- Safety: SIMATIC Safety Advanced, TwinSAFE.
- Languages:
 - C++, Java and Python,
 - Structured Text and Function Block Diagrams for PLC code,
 - MATLAB/Simulink, LabVIEW-G.
- Data serialization: Google Protocol Buffer
- State Machines: SCXML.
- GUIs: **TAURUS**, Qt (C++ and Python) for operators



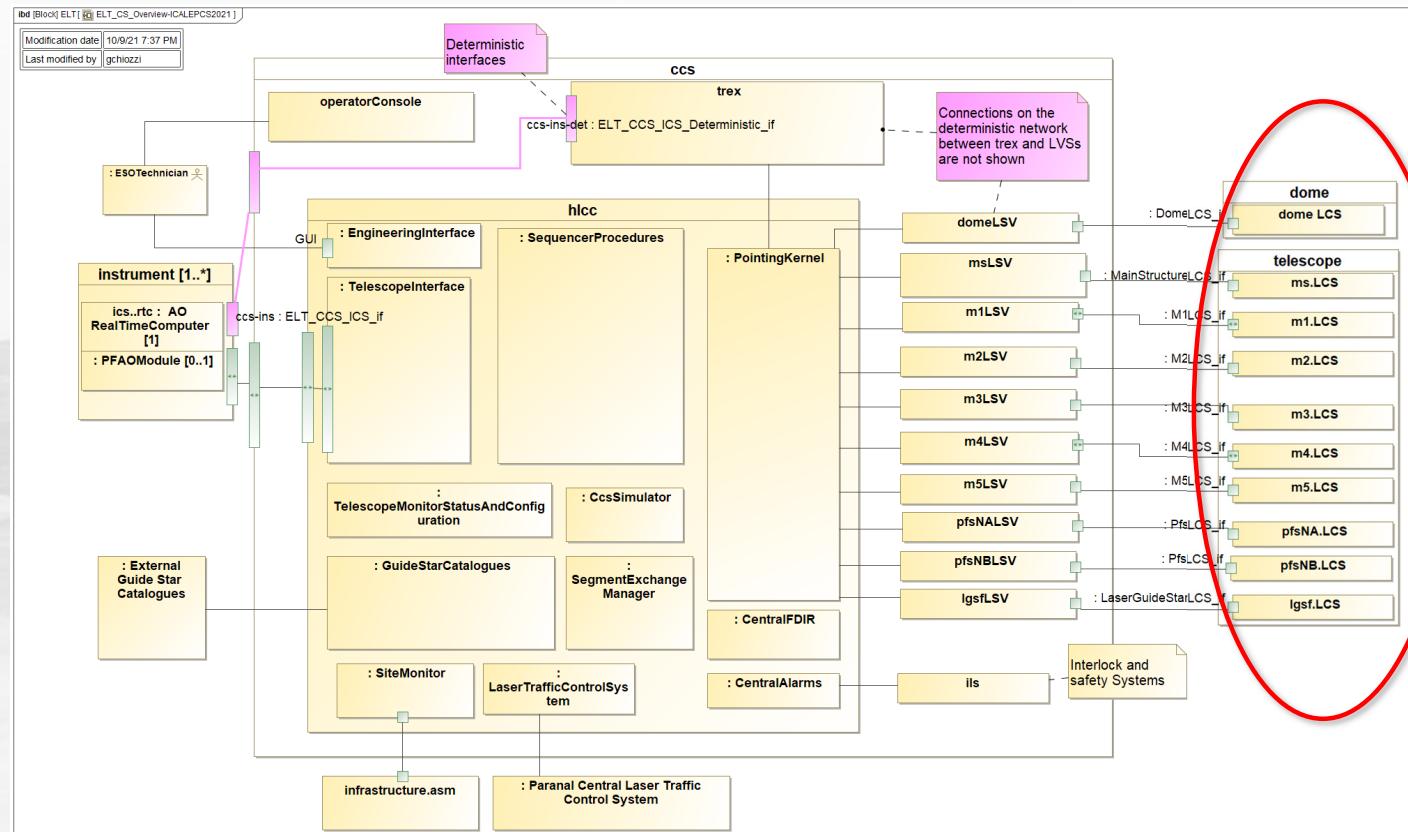
LabVIEW or touch panel HMI for Engineering UIs and hardware control panels.



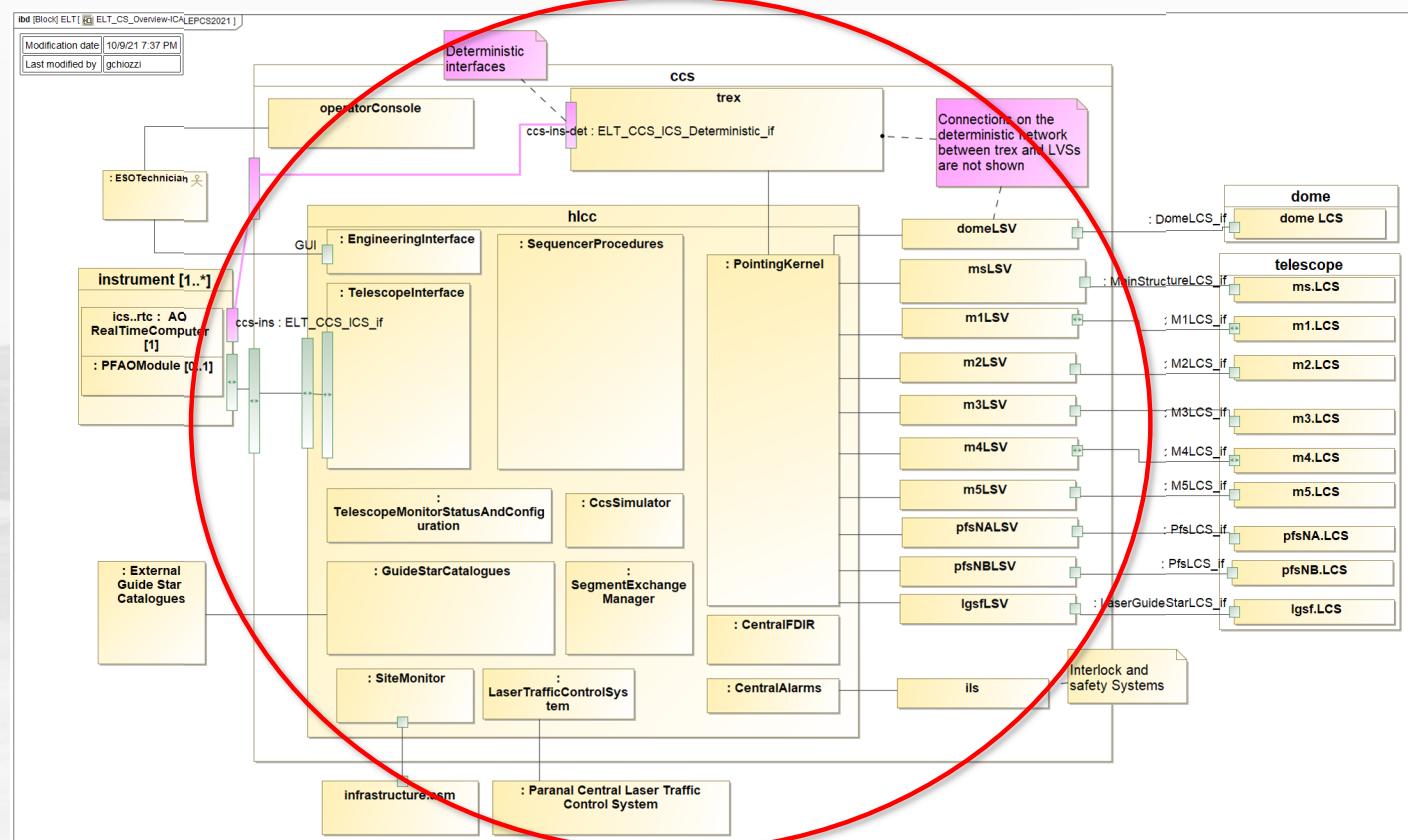
Control System Overview



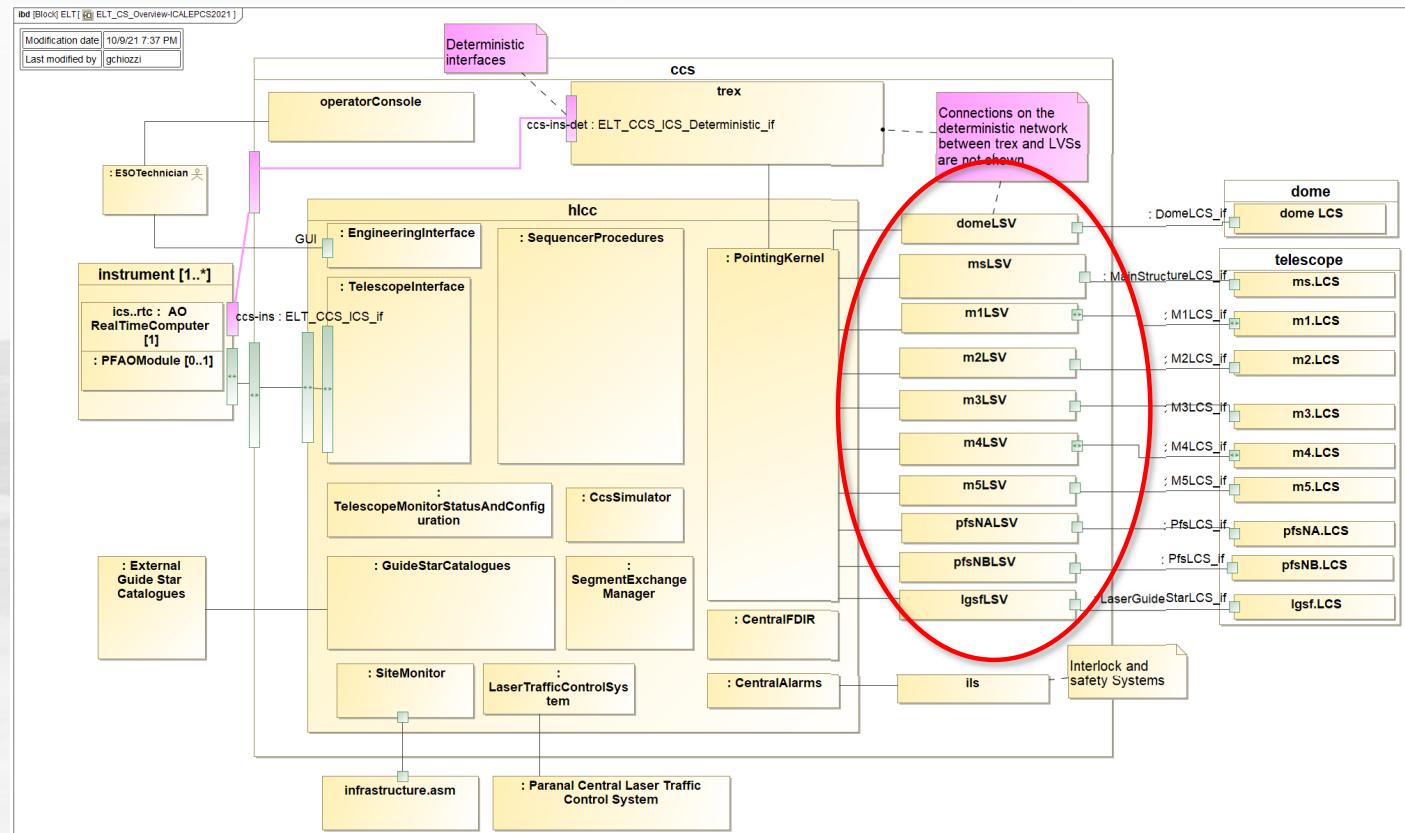
Local Control Systems



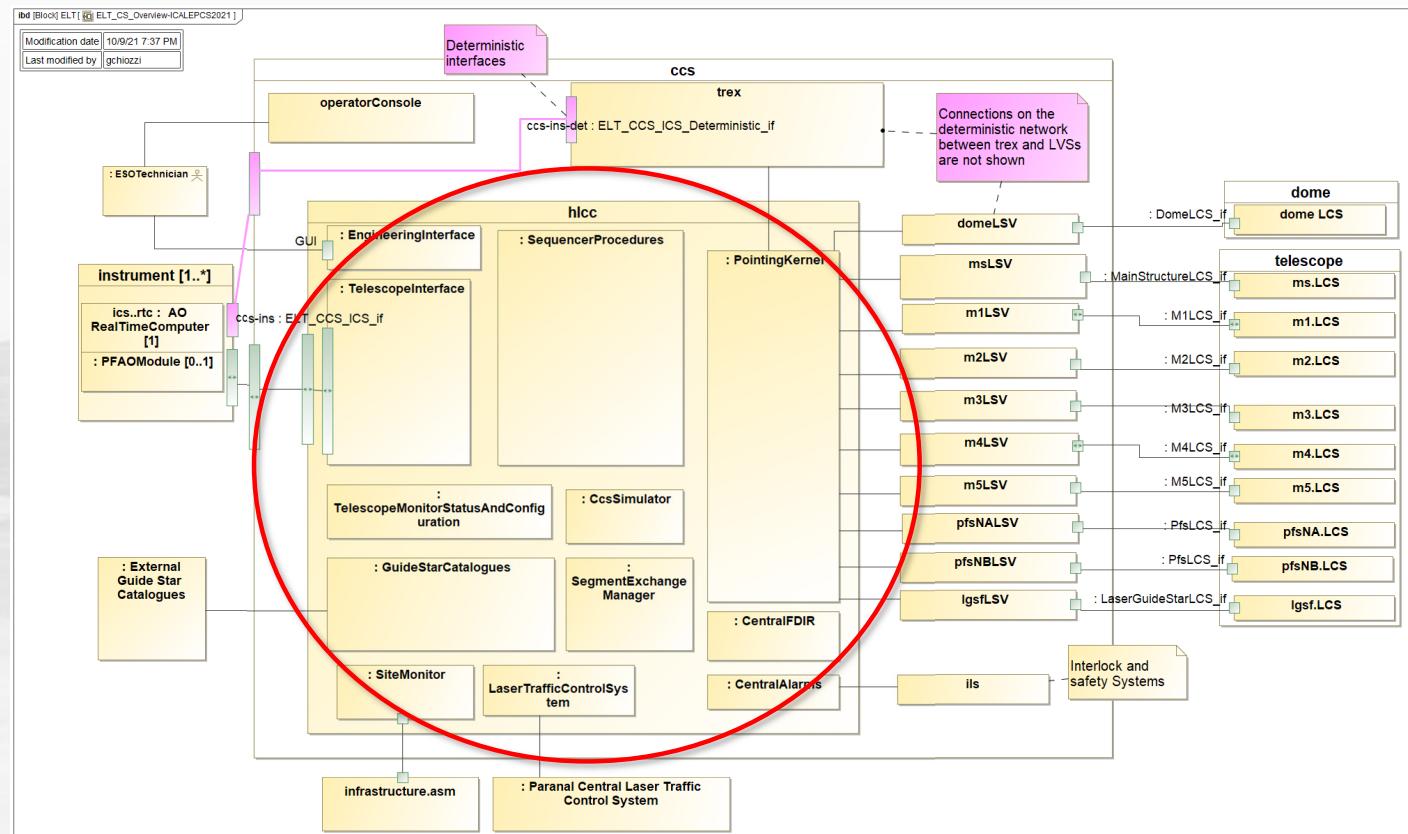
Central Control System



Local Supervisors

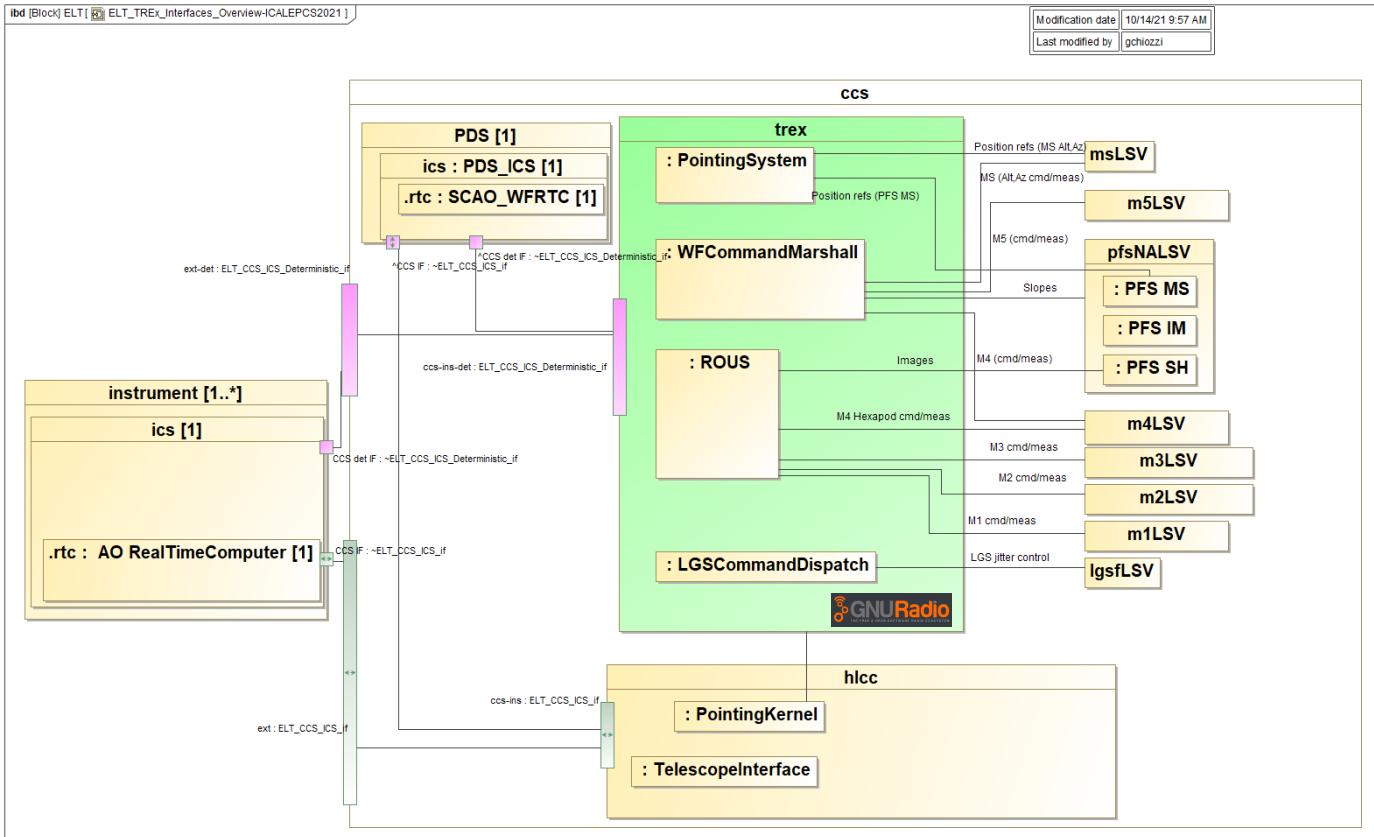


High Level Coordination and Control





TREx and real-time control



Conclusions





Conclusions

- Reqs, architecture, design carried on in the past years.
- Most subsystems (LCSs) contracted out and in production.
- Technical infrastructure and prototypes have been developed.
- We are now moving to serial development of system components (HLCC, LSVs, RTC, AO).
- Requirements for TREx real time components are being collected and design will follow.
- Validation test benches (MELT) are operational



Conclusions

- Reqs, architecture, design carried on in the past years.
- Most subsystems (LCSs) contracted out and in production.
- Technical infrastructure and prototypes have been developed.
- We are now moving to serial development of system components (HLCC, LSVs, RTC, AO).
- Requirements for TREx real time components are being collected and design will follow.
- Validation test benches (MELT) are operational
- Scientific First Light is foreseen for end 2027