



Gianluca Chiozzi is Senior SW Engineer in the Control SW and Engineering Department at ESO.

He is now responsible for the control software of the Astronomical Site Monitor upgrade and is involved in the development of the E-ELT control software.

From 2007 until 2013 he has been the Head of the Control and Instrumentation Software Department.

Before that and since 2000 he was responsible for the ALMA Common Software architecture and development, with a team distributed in various sites in Europe and North America. ACS is the software infrastructure for the ALMA project and is used also by other projects.


During his first years at ESO he has been heavily involved in the design and implementation of the VLT Common Software and Telescope Control Software. Before ESO he has been employed at the IBM Technical and Scientific Research Center in Milan, working on image recognition systems and on operator's user interfaces for utility management systems.



Content

- ASM: what and why
- Upgrade goals
- System overview
- Control system
- Data processing
- Database
- Web display
- Conclusions





What is and why an ASM?


- A set of instruments to provide:
 - Weather and environmental parameters: wind, humidity, temperature, seeing... and now atmospheric profile...


Used by:

- Astronomers and operators
 - Decide schedule of observations
 - Optimize observation parameters
- Telescope/instrument control system
 - Adjust internal parameters (track corrections, temperature control...)
- Scientists
 - Data reduction of scientific data
 - Analysis of weather historical trends
 - Weather forecast

SPIE 2016 – AS16-AS110-91 – ESO ASM Upgrade

3





The weather and environmental parameters provided by the Astronomical Site Monitors at all ESO observatories are indispensable during the whole lifecycle of observations, from the scheduling to the final analysis of the data.

The improvement in the science image quality brought by the new VLT AOF (under commissioning) will be strongly dependent on the vertical distribution of the atmospheric turbulence.

At the moment the system is not providing an official prediction for the seeing, but personnel at the observatory use the available data to make their own short term predictions regarding the seeing/wind/transparency to schedule the observations. Official predictions will be made available in the future.

Paranal Upgrade goals - 1

■ Provide real time information on the vertical distribution of turbulence and other parameters required for planning and optimization of AO observations:

Old ASM dataset

```

nrtso_paramet
start_date datetime 'ykm'
interval scintill
temp1 numerics(2)
temp2 numerics(2)
temp3 numerics(2)
temp4 numerics(2)
temp5a1 numerics(4,2)
temp5a2 numerics(4,2)
rhgr11 string
rhgr12 string
jprmo numerics(2)
wind_gh1 scintill
wind_gh2 scintill
wind_speed1 numerics(4,2)
wind_speed2 numerics(4,2)
wind_speed3 numerics(4,2)
wind_speed4 numerics(4,2)
wind_speed5 numerics(4,2)
dust11 ee
dust12 ee
dust21 ee
dust22 ee

```

seeing_paramet

```

start_date datetime 'ykm'
interval scintill
seeing1 numerics(4,2)
seeing2 numerics(4,2)
seeing3 numerics(4,2)
seeing4 numerics(4,2)
seeing5 numerics(4,2)
seeing6 numerics(4,2)
seeing7 numerics(4,2)
seeing8 numerics(4,2)
seeing9 numerics(4,2)
seeing10 numerics(4,2)
seeing11 numerics(4,2)
seeing12 numerics(4,2)
seeing13 numerics(4,2)
seeing14 numerics(4,2)
seeing15 numerics(4,2)
seeing16 numerics(4,2)
seeing17 numerics(4,2)
seeing18 numerics(4,2)
seeing19 numerics(4,2)
seeing20 numerics(4,2)
seeing21 numerics(4,2)
seeing22 numerics(4,2)
seeing23 numerics(4,2)
seeing24 numerics(4,2)
seeing25 numerics(4,2)
seeing26 numerics(4,2)
seeing27 numerics(4,2)
seeing28 numerics(4,2)
seeing29 numerics(4,2)
seeing30 numerics(4,2)
seeing31 numerics(4,2)
seeing32 numerics(4,2)
seeing33 numerics(4,2)
seeing34 numerics(4,2)
seeing35 numerics(4,2)
seeing36 numerics(4,2)
seeing37 numerics(4,2)
seeing38 numerics(4,2)
seeing39 numerics(4,2)
seeing40 numerics(4,2)
seeing41 numerics(4,2)
seeing42 numerics(4,2)
seeing43 numerics(4,2)
seeing44 numerics(4,2)
seeing45 numerics(4,2)
seeing46 numerics(4,2)
seeing47 numerics(4,2)
seeing48 numerics(4,2)
seeing49 numerics(4,2)
seeing50 numerics(4,2)
seeing51 numerics(4,2)
seeing52 numerics(4,2)
seeing53 numerics(4,2)
seeing54 numerics(4,2)
seeing55 numerics(4,2)
seeing56 numerics(4,2)
seeing57 numerics(4,2)
seeing58 numerics(4,2)
seeing59 numerics(4,2)
seeing60 numerics(4,2)
seeing61 numerics(4,2)
seeing62 numerics(4,2)
seeing63 numerics(4,2)
seeing64 numerics(4,2)
seeing65 numerics(4,2)
seeing66 numerics(4,2)
seeing67 numerics(4,2)
seeing68 numerics(4,2)
seeing69 numerics(4,2)
seeing70 numerics(4,2)
seeing71 numerics(4,2)
seeing72 numerics(4,2)
seeing73 numerics(4,2)
seeing74 numerics(4,2)
seeing75 numerics(4,2)
seeing76 numerics(4,2)
seeing77 numerics(4,2)
seeing78 numerics(4,2)
seeing79 numerics(4,2)
seeing80 numerics(4,2)
seeing81 numerics(4,2)
seeing82 numerics(4,2)
seeing83 numerics(4,2)
seeing84 numerics(4,2)
seeing85 numerics(4,2)
seeing86 numerics(4,2)
seeing87 numerics(4,2)
seeing88 numerics(4,2)
seeing89 numerics(4,2)
seeing90 numerics(4,2)
seeing91 numerics(4,2)
seeing92 numerics(4,2)
seeing93 numerics(4,2)
seeing94 numerics(4,2)
seeing95 numerics(4,2)
seeing96 numerics(4,2)
seeing97 numerics(4,2)
seeing98 numerics(4,2)
seeing99 numerics(4,2)
seeing100 numerics(4,2)

```

New ASM dataset

name	unit	comment
start_date	datetime	'ykm'
interval	scintill	
temp1	numerics(2)	
temp2	numerics(2)	
temp3	numerics(2)	
temp4	numerics(2)	
temp5a1	numerics(4,2)	
temp5a2	numerics(4,2)	
rhgr11	string	
rhgr12	string	
jprmo	numerics(2)	
wind_gh1	scintill	
wind_gh2	scintill	
wind_speed1	numerics(4,2)	
wind_speed2	numerics(4,2)	
wind_speed3	numerics(4,2)	
wind_speed4	numerics(4,2)	
wind_speed5	numerics(4,2)	
wind_speed6	numerics(4,2)	
wind_speed7	numerics(4,2)	
wind_speed8	numerics(4,2)	
wind_speed9	numerics(4,2)	
wind_speed10	numerics(4,2)	
wind_speed11	numerics(4,2)	
wind_speed12	numerics(4,2)	
wind_speed13	numerics(4,2)	
wind_speed14	numerics(4,2)	
wind_speed15	numerics(4,2)	
wind_speed16	numerics(4,2)	
wind_speed17	numerics(4,2)	
wind_speed18	numerics(4,2)	
wind_speed19	numerics(4,2)	
wind_speed20	numerics(4,2)	
wind_speed21	numerics(4,2)	
wind_speed22	numerics(4,2)	
wind_speed23	numerics(4,2)	
wind_speed24	numerics(4,2)	
wind_speed25	numerics(4,2)	
wind_speed26	numerics(4,2)	
wind_speed27	numerics(4,2)	
wind_speed28	numerics(4,2)	
wind_speed29	numerics(4,2)	
wind_speed30	numerics(4,2)	
wind_speed31	numerics(4,2)	
wind_speed32	numerics(4,2)	
wind_speed33	numerics(4,2)	
wind_speed34	numerics(4,2)	
wind_speed35	numerics(4,2)	
wind_speed36	numerics(4,2)	
wind_speed37	numerics(4,2)	
wind_speed38	numerics(4,2)	
wind_speed39	numerics(4,2)	
wind_speed40	numerics(4,2)	
wind_speed41	numerics(4,2)	
wind_speed42	numerics(4,2)	
wind_speed43	numerics(4,2)	
wind_speed44	numerics(4,2)	
wind_speed45	numerics(4,2)	
wind_speed46	numerics(4,2)	
wind_speed47	numerics(4,2)	
wind_speed48	numerics(4,2)	
wind_speed49	numerics(4,2)	
wind_speed50	numerics(4,2)	
wind_speed51	numerics(4,2)	
wind_speed52	numerics(4,2)	
wind_speed53	numerics(4,2)	
wind_speed54	numerics(4,2)	
wind_speed55	numerics(4,2)	
wind_speed56	numerics(4,2)	
wind_speed57	numerics(4,2)	
wind_speed58	numerics(4,2)	
wind_speed59	numerics(4,2)	
wind_speed60	numerics(4,2)	
wind_speed61	numerics(4,2)	
wind_speed62	numerics(4,2)	
wind_speed63	numerics(4,2)	
wind_speed64	numerics(4,2)	
wind_speed65	numerics(4,2)	
wind_speed66	numerics(4,2)	
wind_speed67	numerics(4,2)	
wind_speed68	numerics(4,2)	
wind_speed69	numerics(4,2)	
wind_speed70	numerics(4,2)	
wind_speed71	numerics(4,2)	
wind_speed72	numerics(4,2)	
wind_speed73	numerics(4,2)	
wind_speed74	numerics(4,2)	
wind_speed75	numerics(4,2)	
wind_speed76	numerics(4,2)	
wind_speed77	numerics(4,2)	
wind_speed78	numerics(4,2)	
wind_speed79	numerics(4,2)	
wind_speed80	numerics(4,2)	
wind_speed81	numerics(4,2)	
wind_speed82	numerics(4,2)	
wind_speed83	numerics(4,2)	
wind_speed84	numerics(4,2)	
wind_speed85	numerics(4,2)	
wind_speed86	numerics(4,2)	
wind_speed87	numerics(4,2)	
wind_speed88	numerics(4,2)	
wind_speed89	numerics(4,2)	
wind_speed90	numerics(4,2)	
wind_speed91	numerics(4,2)	
wind_speed92	numerics(4,2)	
wind_speed93	numerics(4,2)	
wind_speed94	numerics(4,2)	
wind_speed95	numerics(4,2)	
wind_speed96	numerics(4,2)	
wind_speed97	numerics(4,2)	
wind_speed98	numerics(4,2)	
wind_speed99	numerics(4,2)	
wind_speed100	numerics(4,2)	

- Integrate new instruments:
 - MASS
 - SLODAR
 - PWV
- Make available a much wider dataset

New features:

- Multi Aperture Scintillation Sensor (MASS)^[4] to measure the vertical distribution of turbulence in the high atmosphere and its characteristic velocity by analyzing the scintillation of bright stars;
- SLOpe Detection And Ranging (SLODAR)^[5] telescope, for measuring the altitude and intensity of turbulent layers in the low atmosphere by means of a triangulation method, in which the turbulence profile is recovered from observations of bright binary stars using a Shack-Hartmann wavefront sensor.
- Integrate Paranal Water Vapour Monitor:
 - Measure of the water vapour content of the atmosphere with high precision and time resolution, allowing the execution of infrared observations in periods of low precipitable water vapour.
 - Measure of infrared sky brightness
 - ... and other data.
- About x5 more data items with respect to the old system

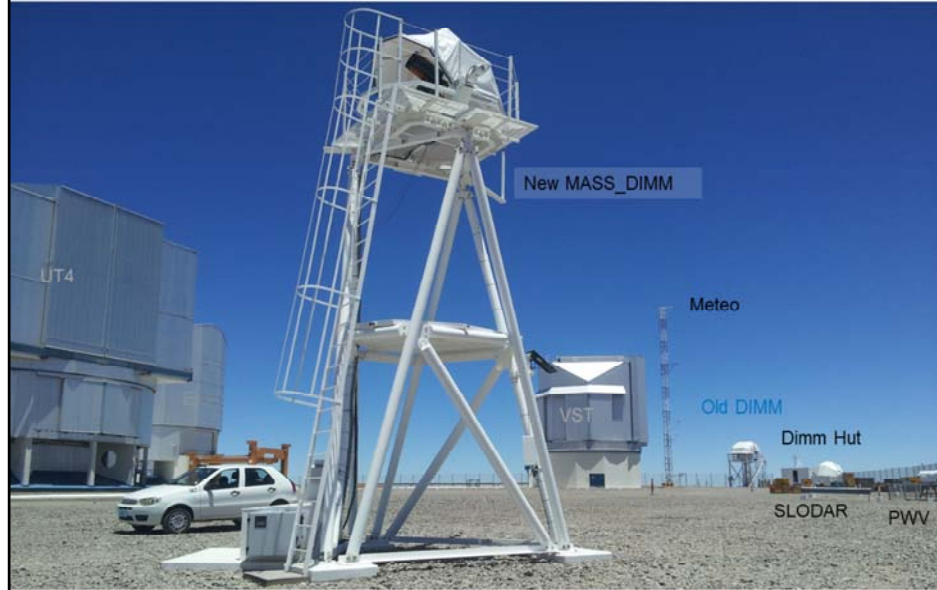
Paranal ASM Upgrade goals - 2

- Overcome technical limitations:
 - Limited number of displays in control room
 - Limited access to data outside the control room
 - No more than 24h data in control room
 - Manage HW and SW obsolescence
 - Original system in operation since the 90s
 - HP-RTAP SCADA development dead
 - HP-UX workstation (necessary for RTAP) not on market
 - Several boards not produced any more
 - ESO TCCD to be replaced
 - More maintainable and expandable system:
 - New devices/sensors to come
 - Integrate off-the-shelf components
 - Update to mainstream technologies and languages
- ... at the same time...
- Keep backward compatibility in external interfaces
 - Look and feel the same in the control room
 - Robust fully autonomous operation





ASM upgrade instruments



Instruments distributed in convenient positions on the platform.

MASS-DIMM and SLODAR instruments on identical commercial mounts (Astelco NTM500) and enclosures (Halfmann).

MASS-DIMM developed jointly at Sternberg Institute (Moscow, Russia) and Cerro Tololo Observatory.

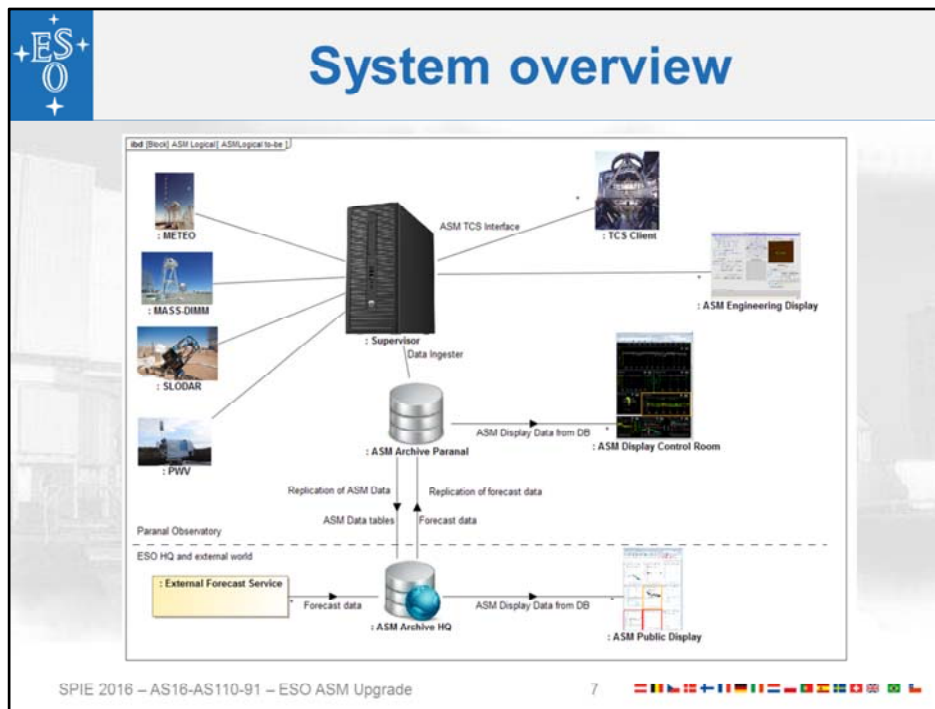
More than 30 devices are in use around the world.

SLODAR developed at the center for Advanced Instrumentation (CAI, Durham University, UK)

SLODAR systems, based on small telescopes (typically 50cm aperture), have been employed for characterization of the optical turbulence at Paranal, ORM (La Palma), Mauna Kea and SAAO observatories.

MASS-DIMM and SLODAR subsystems are operational during night time, as they rely on the observation of stars

Meteo tower and water vapour radiometer are in operation 24/7.

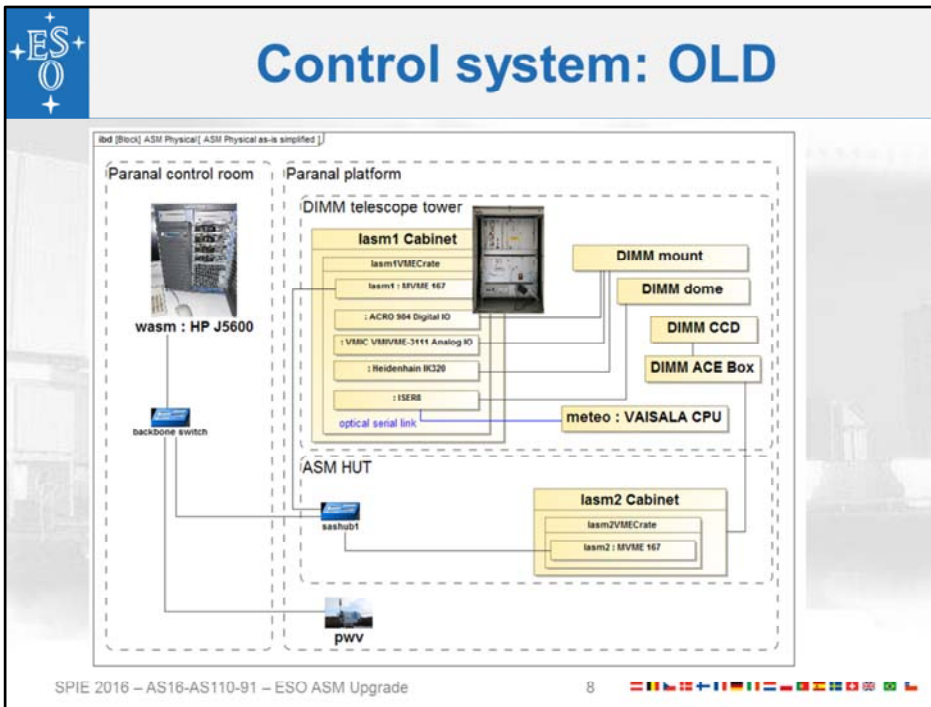


OLD control system was responsible both for

- collecting data
 - providing display of the weather conditions in the control room.
- Once a day, during day-time, part of the data archived in a relational database for offline usage.
- Data available in the control room was limited to the last 12 or 24 hours.

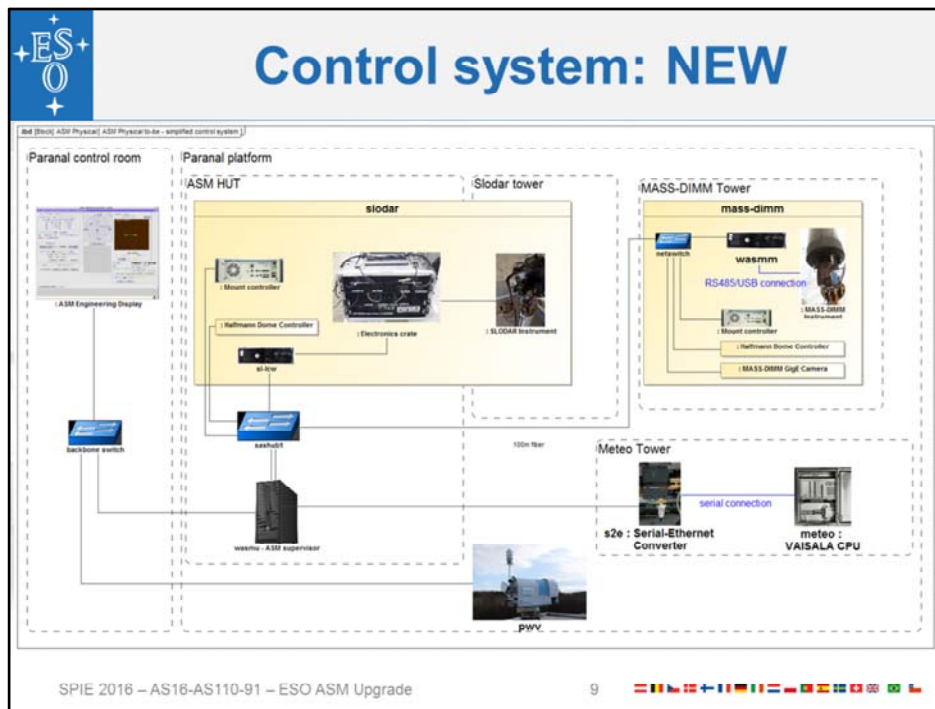
The NEW system decouples data display, retrieval and analysis from data collection:

- The control system
 - collects the data
 - sends it to a relational database.
 - provides real time data access to telescopes and instruments, using backward compatible interfaces.
- Any application, including the display in the control room, gets the data from the relational database.
- Data replication allows access both in Paranal and in near real time also at ESO HQ in Garching with appropriate quality of service characteristics.
- We integrate into the database forecast data provided by external services (European Center for Medium Range Forecasts' (**ECMWF**))



The OLD control system was fully built on the classical VLT architecture:

- devices connected to IO boards on VME-based Local Control Units (LCUs), with the code primarily developed in ANSI C;
- DIMM using a VLT Technical CCD camera, developed in-house by ESO;
- the mount of the telescope was controlled by the same tracking software as used on the VLT Unit Telescopes, modified to support the specific hardware;
- an HP Unix workstation (obsolete since years) was running the supervisory software, written primarily in C++

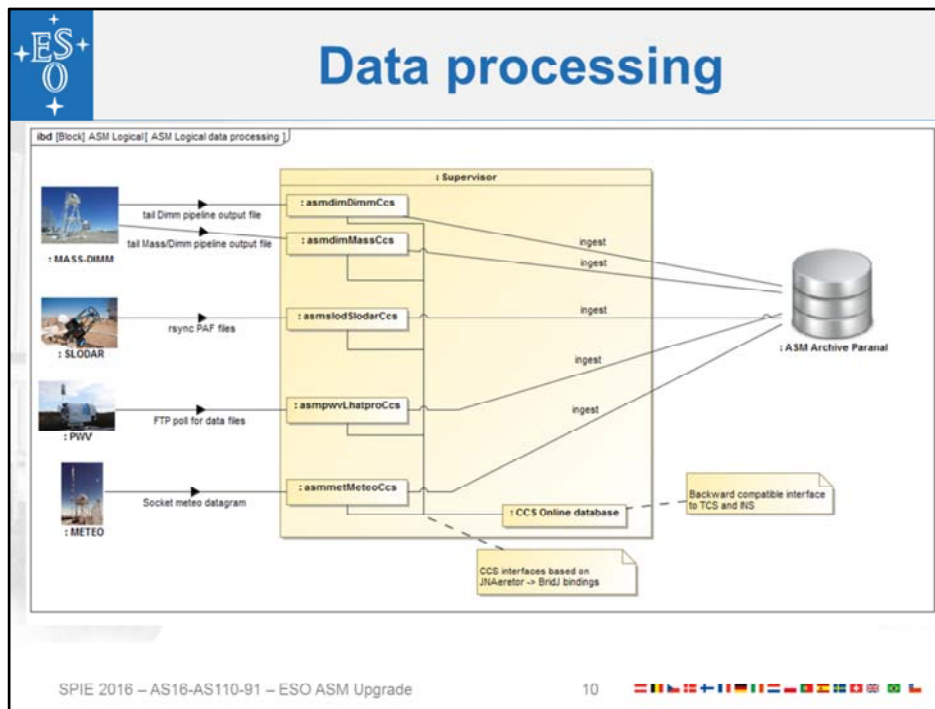


NEW control system:

- Latest VLT SW, with Java and Python replacing C/C++ (with exception for C++ external interfaces)
- Integrate off-the-shelves devices and their control systems (MASS-DIMM and SLODAR).

Do not re-implement what available:

- Synergies with other users
- We own the code
- Drawback: MASS uses TCL
- No LCUs (no devices connected to IO boards, no strict real time control requirements)
- MASS-DIMM and SLODAR have Linux workstations for the control
- Central supervisory Linux workstation to coordinate the operation of all instruments:
- Meteo station is now connected through a serial-to-Ethernet adapter:
 - No cable length limitations
 - Simple socket interface
- Mounts are driven by the vendor commercial controller (linux based)
- TCCD replaced by GigE Prosilica commercial camera
- MASS still requires a USB connection for a RS485-to-USB converter



SPIE 2016 – AS16-AS110-91 – ESO ASM Upgrade

10



The ASM subsystems deliver their proprietary data at different levels of complexity:

- SLODAR produces ready-to-use PAF data files in the black-box vendor's software;
- METEO publishes custom ASCII datagrams that we read from a socket connection;
- LHATPRO publishes custom data files for us to poll from an FTP server;
- MASS-DIMM uses for each detector a two-staged pipeline: data supervisors follow data as it gets appended to files, mix output, and identify new data blocks in the continuous streams of new data lines.

Independent Data Supervisors:

- Collect, reformat and aggregate device data
- Perform data scaling to different units
- Compute derived data items
- Ingest into the archive in Paranal
- Write in the CCS Online database for TCS/INS usage (backward compatible)

Attention placed on robust data flow:

- Any stage can be restarted independently, or will wait if started too early.
- No data can ever be lost, thanks to buffering in the file system or the online database.
- Re-processing of old data after restart is recognized and suppressed.
- Data files get deleted only after the ingestion in the relational database has been verified independently

All data flow applications are implemented in Java because of better support


- Multithreading
- database connection pooling (c3p0)
- Parsing .. and many other tasks of data flow.

Wrapped the C API of VLT common software using generated Java bindings:

jnaerator using the BridJ Java-C binding target.

The JNA binding is more widely used, but BridJ appeared clearer in the mapping of complex pointer constructs.


We also use BridJ to wrap the slalib astronomical routines for Java.

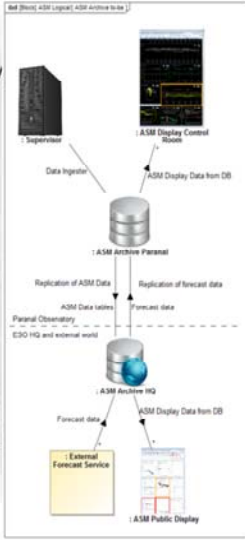


ASM database

- Based on existing ESO dataflow infrastructure:
 - SAP ASE relational database and SAP replication server.
 - Architecture allows fast and reliable access from the observatory
 - Replication makes data available immediately also at HQ
 - Data rate is about 2.5 MB per day.
- Ingestion and display based on configuration tables.
 - Easy to add new parameters or instruments
- For large historical data samples down-sampled data is stored into hourly, daily and weekly averages.
- Data can be retrieved using standardized query forms:


<http://archive.eso.org/cms/eso-data/ambient-conditions/paranal-ambient-query-forms.html>





SPIE 2016 – AS16-AS110-91 – ESO ASM Upgrade

11



The database has been created in the existing ESO operational data flow system:

- local database server in the observatories
- central database server at ESO Headquarters.
- synchronization of the databases by means of database replication technology

The distributed architecture is a major step forward, since:

- It provides fast and reliable access to the data from the observatory,
- It protects from connectivity outages from headquarters.
- Allows real time access from the database in headquarters of the full history of the observatories
- The database in the observatories can be kept small, therefore improving the recovery time of the local database if needed.

The process that ingests data uses configuration tables in the database itself to identify the data destination table and column.

In this way, any new parameter can be easily added by introducing a new database table or database column and adding the proper configuration in the database.

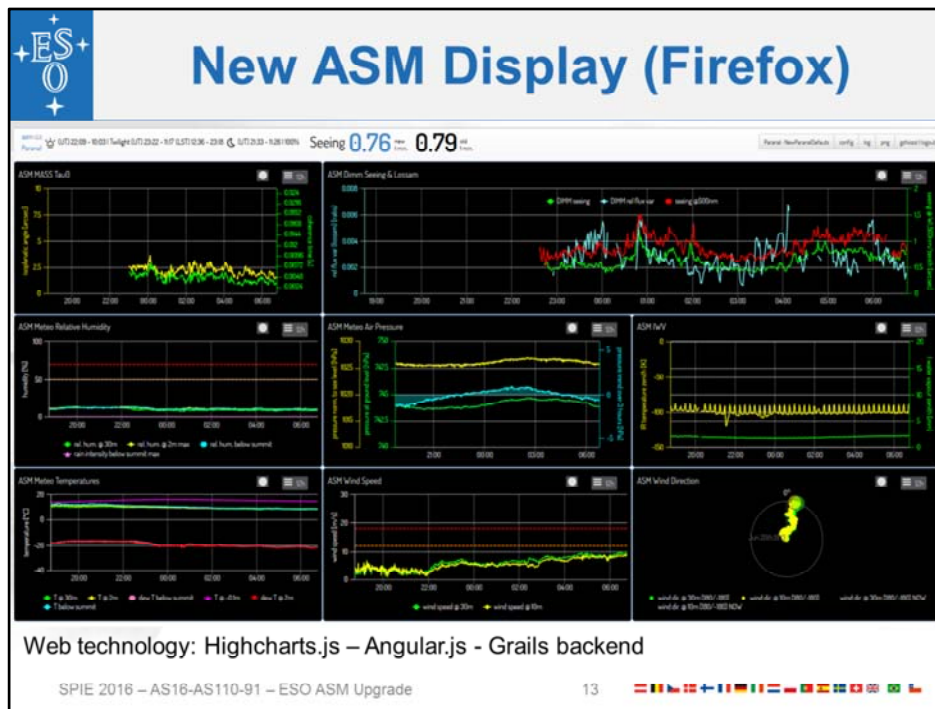
These configuration tables are used as well by the web display to decide which parameters can be plotted in the tool.

Therefore, making a new instrument or parameter available for plotting does not require any code change.

Forecast data retrieved from external services are inserted at ESO headquarters. They can be used in the ASM web display. Such data are replicated from headquarters to Paranal to be available at the observatory as well.

In order to allow the display of large historical time ranges, database tables are down-sampled into hourly, daily or weekly averages, populated periodically by an external process.

The ASM data is accessible also directly using standardized query forms, common to all ESO observatories.



And this is the new ASM Display as seen in the Paranal control room.

Our second driver was to make the live updating plots easily accessible also from stakeholders worldwide.

Consequently we decided for a web-based solution using modern browsers (Chrome, Firefox, Safari) as visualization platform driven by JavaScript. This allows to offload the CPU needed for generating the graphs to the web browsers running on the clients, while the data is retrieved from the database and not any more from the control workstation.

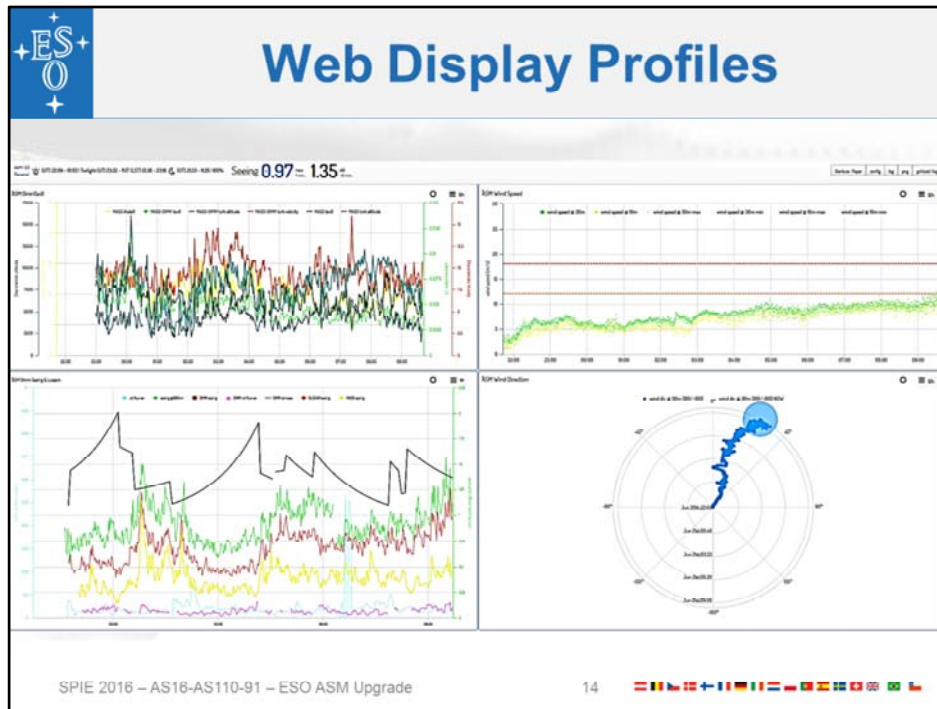
In this way we do not have any more the constraints of the old system on the number of consoles that are allowed to display weather data. Moreover, if very complex graphs, with a lot of historical data, are requested by users, only the client web browser will struggle and the rest of the system will not be affected.

Visualization requirements were demanding. We initially considered using the flexible JavaScript library d3.js.

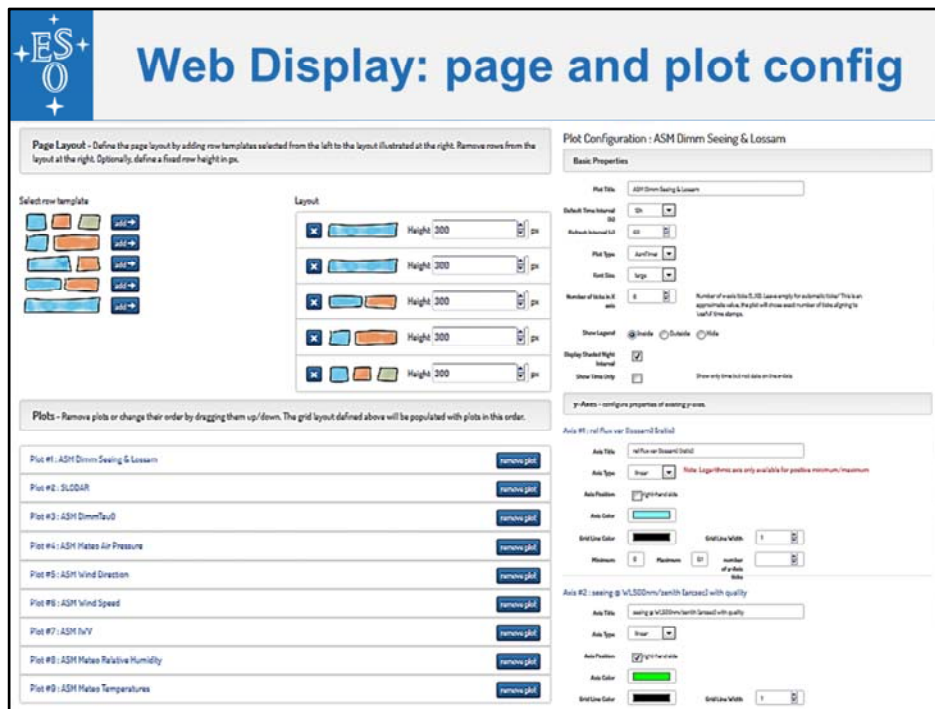
After prototyping we realized that it was too low-level and we were spending too much time reinventing the wheel.

We decided to go for highcharts.js and its companion library highstock.js.

We found these libraries to be extremely feature-rich, well-documented and supported, highly customizable and to have a large user base.



A personal profile with an active warning indication: a wind speed threshold was crossed



- Configuration:
 - Profiles
 - Page
 - Plot

Offering a rich and dynamic configuration user interface in a web browser - similar to a desktop GUI application - is not trivial.

Over the last couple of years, browser programming in JavaScript has undergone a major paradigm change:

- Before dynamic browser behavior was primarily using asynchronous server calls (AJAX) triggering callbacks that would directly modify the browser's DOM tree. This often led to hard to the *callback hell*, a codebase that was difficult to understand and maintain and a tight coupling of data and view.
- More recently, browser programming moved away from this *imperative* programming paradigm to a *declarative* one using a client-side model-view-controller pattern. While the controller fetches and maintains the model data, the view *declares* its dynamic dependencies on model changes. The DOM is never directly modified.

For the web display, we chose **Google's angular.js**, one of the most widespread frameworks supporting this declarative paradigm

ES O

Web Display: API for Data Retrieval


REQUEST
 http://www.eso.org/asm/api?
 from=2016-06-16T21:00:00.000Z&
 to=2016-06-17T12:00:00.000Z&
 fields=meteo_paranal-rhum1,meteo_paranal-rhum2

RESPONSE in JSON format

```
{
  meteo_paranal-rhum1: [
    [1466110801000, 11], // timestamp, value
    [1466110861000, 14],
    ...
  ],
  meteo_paranal-rhum2: [
    [1466110801000, 16],
    [1466110861000, 17],
    ...
  ]
}
```



Separation of concerns: data access and visualization


- Some browser-based user interface libraries use a server-centric approach
- Complete separation between data access and visualization was very important since various stakeholders have increasing demand to access live or historic data programmatically.
- As a general strategic trend, we strive for web-based solutions, exposing services through RESTful APIs to be consumed by independent GUIs as well as by programmatic clients.
- For ASM, a simple REST API consumes an HTTP GET call with *from ..to* parameters to specify the time interval for which data points are requested and a *fields* parameter that lists the required physical measurement values.
- Results are returned in the lightweight JSON format that is easily consumable by JavaScript clients
- In order to prevent excessive DB queries, different restrictions apply depending on whether the query is made anonymously or by an authenticated user
- We implemented the back-end of the ASM web display using the Grails framework
- The exchanged information is minimal, performance is very good
- Additional APIs allow storage and retrieval of configuration profiles and other actions



Conclusions

- Commissioning completed in March 2016:
 - Plenty of new features
 - More maintainable and extendable
 - We could keep the systems running in parallel for more than 1y
- AOF will start soon to use extensively the new features.
 - This will rise new requirements and provide feedback
- Display:
 - [highcharts.js](#) highly recommended
 - Declarative client-side with [angular.js](#) has been essential
- La Silla upgrade and Armazones installation being planned

SPIE 2016 – AS16-AS110-91 – ESO ASM Upgrade
17


Despite additional devices and much wider set of parameters, the switch to a modern hardware architecture and programming languages has allowed us to implement a robust, maintainable and extensible system.

Running the new system in parallel to the old one for more than one year has been extremely valuable.

This has allowed us to compare the operational behavior of the two systems and tune our new software.

An interesting example is the discovery and investigation of a Ghost manipulating data.

On the control system side we have already received requests for improvements and for adding further data.

The next big testbed will be when the new AOF, being integrated, will use the atmospheric profile data from the ASM.

As far as the web display is concerned, using highcharts.js for visualization is highly recommended. The usage of the "declarative" client-side web framework angular.js was essential for building a desktop-like dynamic user experience and is also highly recommended.

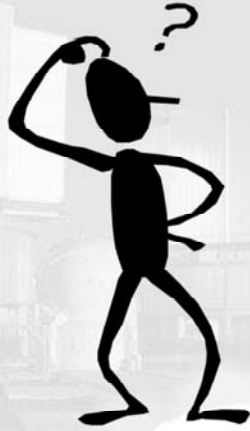
The upgrade of the La Silla ASM and the installation of a new system at the E-ELT site are now being planned.

Already now La Silla data, coming from the old ASM, has been integrated in the new ASM Database and Display applications.

This makes it already possible to compare the weather conditions at the two sites.



Questions (& Answers)



<http://www.eso.org/asm>



SPIE 2016 – AS16-AS110-91 – ESO ASM Upgrade

18

