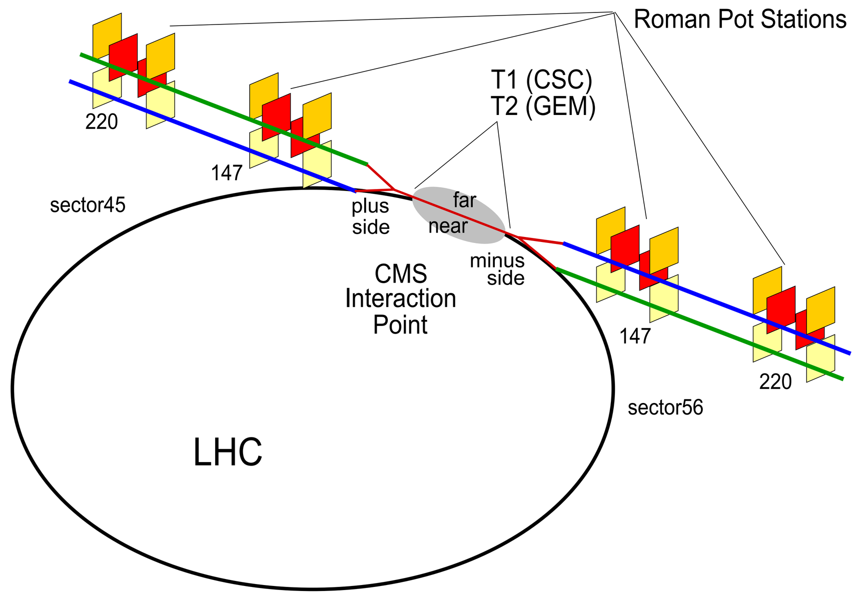
-Big.Brother- The TOTEM Detector Control System

F. Lucas Rodriguez*, I. Atanassov, P. Palazzi, F. Ravotti
CERN

BIG.BROTHER
TOTEM DCS - Detector Control System

Introduction

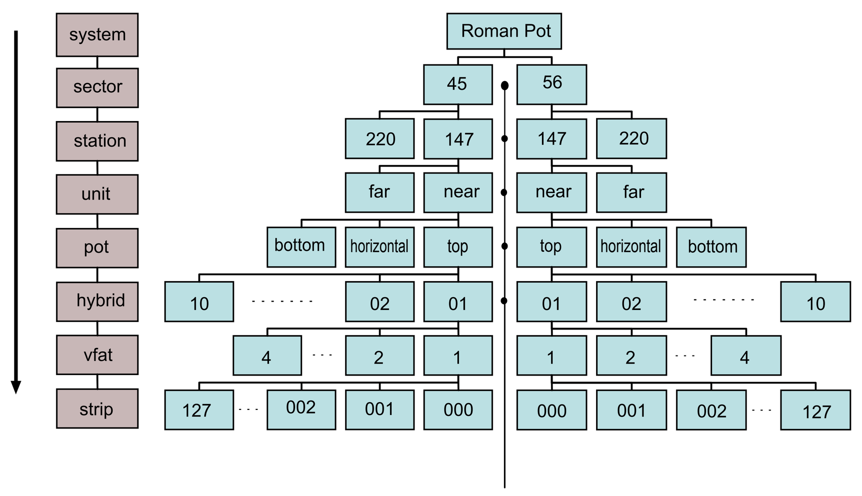
- The TOTEM (Total crOss secTion, Elastic scattering and diffraction dissociation Measurements) experiment at CERN will measure the size of the proton and also monitor accurately the LHC's luminosity. To do this TOTEM must be able to detect particles produced very close to the LHC beams.
- TOTEM consists of "Roman Pot Stations" (RP), "Cathode Strip Chambers" (CSC) Telescope 1 (T1) and "Gas Electron Multipliers" (GEM) Telescope 2 (T2). The T1 and T2 detectors are located on each side of the CMS interaction point in the very forward region, but still within the CMS cavern. Two Roman Pot stations are located on each side of the interaction point at 220 m and 147 m inside the LHC tunnel. Each Roman Pot station consists of two groups of three Roman Pots separated by a few meters.



- Such kind of experiment have a learning phase that will produce elaborated requirements for the Control System. In a first approach in needed to establish all the inputs and outputs of the controlled plant (the experiment) and the relation among them. At a later step it will be evaluated under what exact circumstances actions have to take place.

Product Breakdown Structure of the detector

- A Product Breakdown Structure (PBS) is a hierarchical decomposition. It is structured using nested levels that conforms the main system.
- As an example TOTEM Roman Pot System develops on a hierarchical structure of eight levels which go from the whole roman pot system ("system") to its ultimate granularity ("strip").
- The mirror symmetry with respect to the CMS interaction point has been used to define the names at the different levels. Each one of the sides follow the LHC sectors naming scheme ("sector 45" and "sector 56"). The distance form the central point (CMS) identifies the stations and the units ("station 147", "station 220" and "unit near", "unit far"). The pot name is derived form its position with respect to the beam axis ("pot top", "pot horizontal",...).

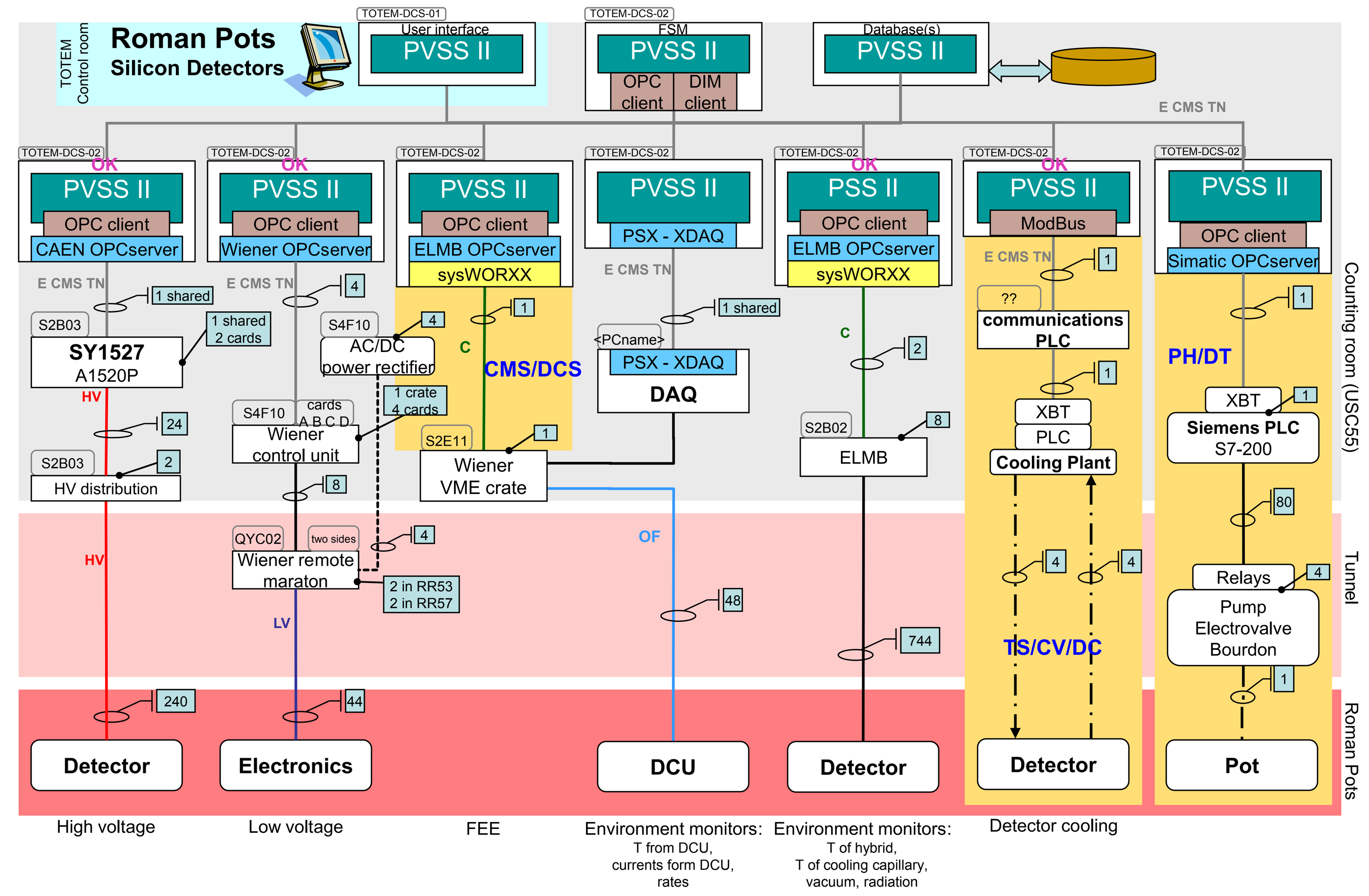


Naming scheme of the detector

- A clear naming scheme is of vital importance in this kind of systems where many almost autonomous subsystems integrate among them. If this scheme does not exist each group uses different conventions, names or ordering. Resulting in systems that cannot be interfaced directly and they become very difficult to develop and maintain.
- The naming of each piece of equipment of the Roman Pot detector is built by concatenating the naming tag of its hierarchy in the PBS.
- For example, the 4th VFAT (that is one of the electronics chips) in the 2nd Hybrid of the *top* Pot in the *far* Unit of the Station at *147*m of the sector *45* is named as *rp_45_147_fr_tp_02_004*.
- It is possible to build a Backus-Naur Form (BNF) grammar for the nomenclature. After this it is easy to validate the names used in the software developments, and define algorithms that only applies to specific PBS items.

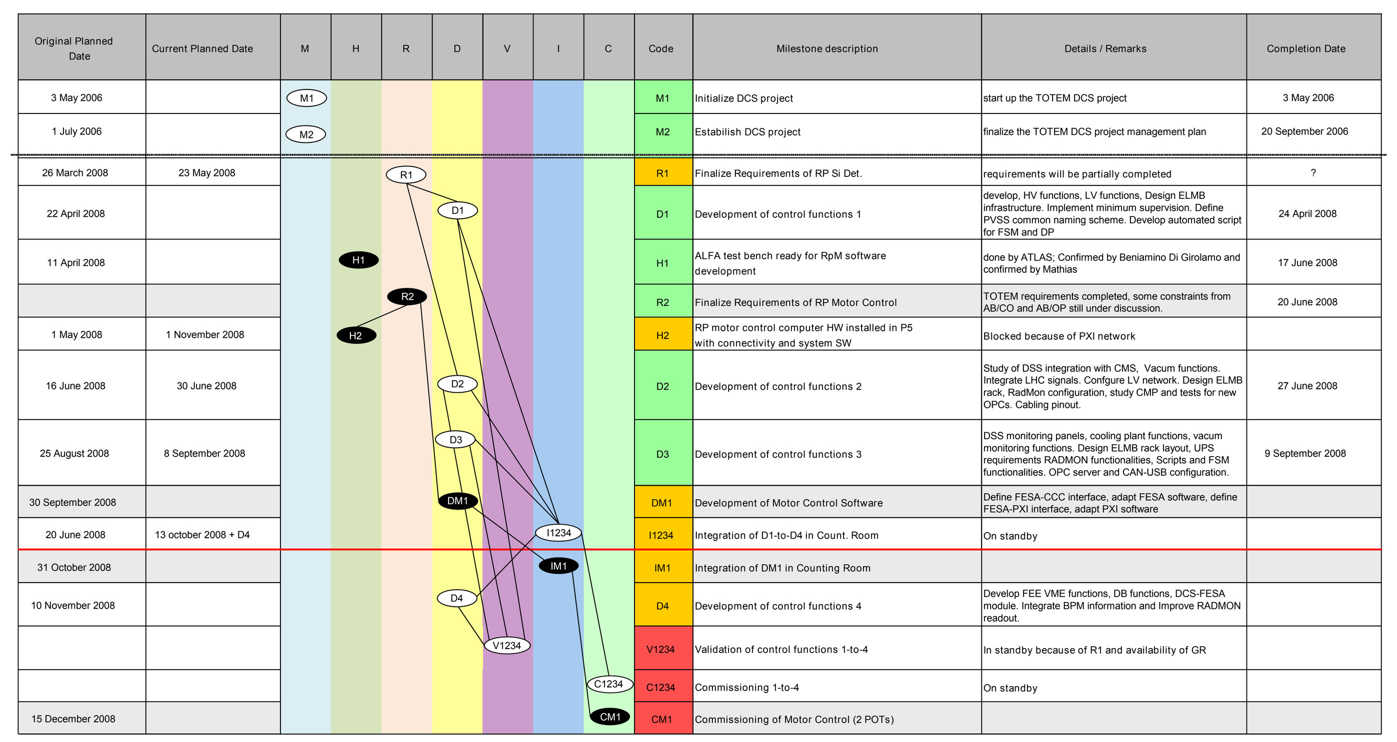
Product Breakdown Structure of the DCS

- In the same way that exists an PBS for the detector exists another one for the DCS itself. It is based decomposing the system by functionality: High Voltage (Hv), Low Voltage (Lv), Environmental sensors, Frond end electronics, Cooling plant,...
- In top of that there is the PVSS software, and the behaviour formalization using Finite State Machines (FSM), monitoring and executing the relevant actions. This software is structured around the concept of "datapoints". The value of the sensors is stored inside datapoints and the commands to the actuators are sent by writing new values into the datapoints.
- Such decomposition is represented as graphical diagrams based on the ALICE DCS.



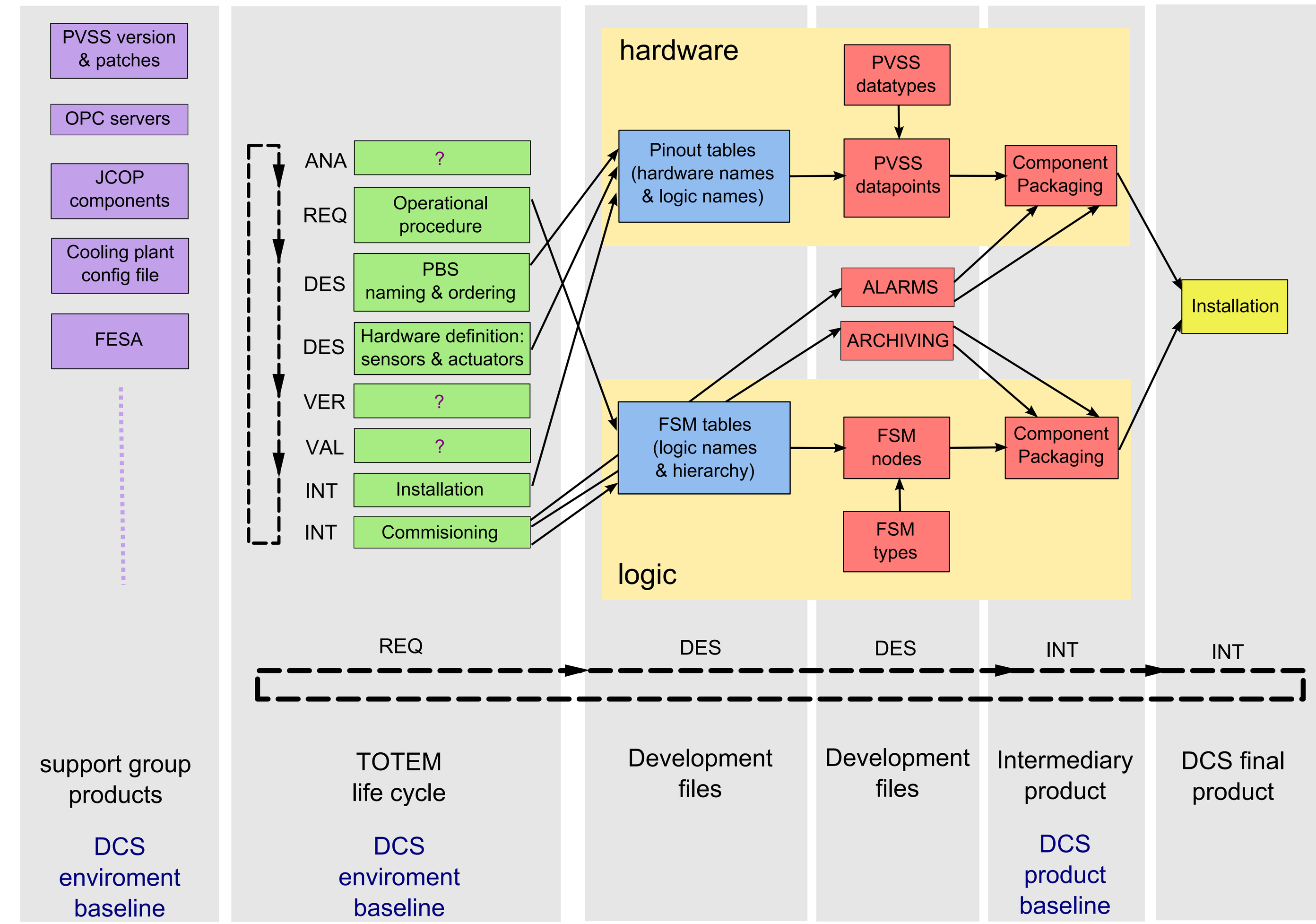
Planning

- The DCS project uses Goal Directed Project Management (GDPM) as planning methodology. This methodology proposes a set of tools and principles for planning, organizing, leading and controlling projects. The method originated from PSO (People, System, Organization) projects in the IT domain. The method encourages a team oriented approach towards planning and controlling projects.
- We establish 6 different kind of DCS activities:
  - Project **M**anagement
  - **H**ardware
  - **R**equirements elicitation
  - **D**evelopment and unit testing
  - **I**ntegration
  - **C**ommissioning
- For each activity a list of milestones are represented in the form of bubbles. Linked milestones mean that the following one cannot be achieved before the previous has been completed; in other words, the finalization of the first one is necessary in order to complete the following linked milestone. Each milestone is decomposed in a detailed Activity Plan consisting of several Work Packages (WP). Each WP corresponds to the single piece of control that has to be developed, tested and implemented and commissioned to guarantee the operation of the TOTEM experiment.



Life Cycle

- Each new iteration of the development cycle can have a huge impact on the initial requirements. And how the purpose of the software is basic research, the previous experience is very limited. Is is needed to provide correct releases as fast as possible to match the new cabling or fix operational logic. Each software release helps validating the initial requirements and assumptions, and clarifies the next development cycles.



- There are four different kinds of blocks:
  - **Green blocks**
    Requirements and the physical construction of the detector. Naming scheme, pinout tables, commissioning results (after development iterations), …
  - **Blue blocks**
    Engineering formalization of all the requirements in a way that can be processed automatically. However, they do not attempt to be a 100% formalization of the requirements. The order of magnitude for hardware control functions or sensors (PVSS datapoints) can be near 4000 *items*, and the number of FSM nodes can be around 2500 *items*. Generate such a huge amount of items inside a PVSS project in a manual, or semiautomatic way is not good enough. The tedious JCOP procedure of manual generation of all those items can lead to human errors. Also this intermediary representation allows the physicist or any other provider of requirements to validate our development in a very early stage.
  - **Red blocks**
    PVSS developments; datapoints, datapoint types, FSM types, scripts, panels,… Some of them are internal to TOTEM, but others are sent to CMS as packages for integration.

Configuration Management

- Configuration management is applied in all the steps of the DCS, but defining two major types of baselines:
  - **DCS environment baseline**
    The one of pieces the DCS depend on (such as PVSS version, OPC servers, JCOP components,…).
  - **DCS product baseline**
    The DCS development process output; the CMS-compatible components for integration.
- All the code, requirements, documentation and even the webpage itself are stored in a Subversion repository, so the traceability of the changes is assured.

Conclusion

- The control system must have a development methodology flexible enough to provide a new release of the system a few days after new requirements have been defined. It must be also a well defined procedure, so the changes in the code can be traced back, and automatized as much as possible to avoid human mistakes.
- The work presented in this article describes the global structure of the project. Also a tool to estimate the response time of some interlocks has been implemented. It uses an Information Theory approach.

References

- G. Anelli, G. Antchev, P. Aspell *et al.*, "The TOTEM experiment at the CERN Large Hadron Collider," JINST, 2008.
- F. Lucas Rodríguez, "Design, Development and Verification of the Detector Control System for the TOTEM experiment at the CERN LHC," Ph.D. dissertation, Universidad de Sevilla, May 2009.