The SPS-2001 Software Project

Pierre Charrue, Bernard Denis,
Michel Jonker, Marc Vanden Eynden,
CERN, Geneva, Switzerland

Abstract

In December 1996, it has been decided to set up a project [1] to reengineer the SPS application software to allow efficient operation and to cover the specific needs of the SPS as LHC injector. This presentation reviews the reasons why this project was launched and its scope and objectives. The project plan will present the OO methods and the phase decomposition to reach the first important deadline in June 2001.The first results of the analysis phase and their implications on the application software, on the controls infrastructure and on the equipment groups equipment will be presented.

1. MANDATE OF THE SPS-2001 PROJECT

The application software used today to control the SPS is the result of 20 years of evolution of the accelerator and computer technology. Many application have been designed for and by equipment specialists. Although parts of the application software are much appreciated by the operators, today software does not allow an efficient exploitation of the SPS machine. Since the SPS will probably run for another 20 years, it has been decided to set up a project to do the re-engineering of the SPS application software to allow more efficient operation and to cover the need for the LHC era, that is the usage of the SPS as LHC injector.

1.1 Purpose.

The purpose of the project is to provide an application software suite to operate the SPS and its transfer lines in the LHC era that is allowing LHC filling, fixed target operation and MD's. An important benefit of this project is to provide an homogenous software suite to the control room.

1.2 Scope.

The mandate of the SPS-2001 project consists of:

1. the definition of the strategy to operate SPS in 2001 onwards,
2. the definition of the strategy to implement the new software,
3. the production of all the software that will be used by the operation crew to operate SPS in 2001 onwards.

In order to provide an homogenous solution to the control room, the SPS-2001 project will:

1. specify the interface to all equipment that must be accessed for the operation of SPS,
2. define the standard services that these equipment should provide,
3. define a standard Man Machine Interface,
4. specify general purpose services and their interfaces needed to build the SPS-2001 software; e.g. Data Management Service, Alarm Service, etc. ,
5. define a certification principle for (1), (2) and (3).

The SPS-2001 project will establish a communication path with the equipment groups and other entities that might be affected by the project. In addition, it will communicate SPS-2001 specific needs and requirements to other relevant projects such as the PS/SL Controls Convergence Project and its sub-projects (i.e. Java API, Middleware, Timing Convergence, etc.)

The SPS-2001 project is not responsible for:

1. the specialist software needed by equipment groups,
2. the definitions and implementation of the controls system infrastructure.

1.3 Constraints and Interfaces.

According to the number of services required by the SPS-2001 project (i.e. Alarms, DBMS services,…) it is mandatory to clarify, once the requirements are defined, which of these services will be part of the scope of the SPS-2001 project and which will be handled by external projects or initiatives. Responsibilities, communication procedures and interfaces shall be defined for each of these services.

1.4 Objectives.

The SPS-2001 project will provide:

1. the strategy to operate SPS in 2001 onwards,
2. the strategy to implement the new software,
3. the SPS-2001 certification document,
4. guidelines to produce equipment services as well as application software,
5. a first working version of the application suite in June 2001,
6. a complete final version in July 2003.

The SPS-2001 project will organize periodic information meetings with equipment groups and other entities to present the project and its last developments and to ensure a good synchronization between the SPS-2001 project and the other initiatives taken in the scope of the SLI project.

2 THE LHC REQUIREMENTS

The LHC machine has specific requirements on the SPS operation that are driving the SPS-2001 project. These are Fast and flexible cycle change, high machine availability and new operational concepts.

The most obvious one is what is called the Fast and Flexible cycle change. Changing from one cycle or supercycle to another is possible today in the SPS but this takes a non negligible amount of time. The future SPS software should provide some facilities that will allow the change of cycle in a much more flexible and rapid way. This will allow a high machine availability and switching from LHC filling to fixed target physics will be done smoothly and rapidly. But this implies that equipment should not anymore include assumption about the super cycle structure and also that they need to be able to manage resident cycles and that new functions has to be implemented in cycle dependant equipment (such as load, unload, activate, …).

Switching from one mode of operation to another requires a new equipment surveillance to ensure that equipment are in the desired state for future mode changes. Subscription mechanisms would give the future software efficiency and flexibility.

3. RELATION WITH OTHER PROJECTS

The SLI project [2] is providing all LHC requirements that should be taken into account by the SPS-2001 project. One member of the SPS-2001 project is also member of the SLI project and the SLI project leader is considered as one of the sponsor of the SPS-2001 project.

The SPS-2001 is responsible of the SPS operational software but it relies heavily on other controls projects that are defining the Controls Infrastructure of the coming years. These projects are organized in the so-called PS/SL convergence project [3].

3.1 The PS/SL convergence project

In December 1996, a memorandum [4] from the CERN director of Accelerators expressed the view of the Accelerator Board on the present status and medium-term future of the Controls Groups in the Accelerator Sector. The convergence of the PS and SL systems was relaunched in 1997 and triggered by a directive from the PS and SL Divisions [5] to investigate and propose concrete actions. An Audit on the PS and SL control systems has been made by an outside controls expert (C.W.Watson from TJNAF). The result of this audit opened the way towards a PS and SL convergence project aimed at providing a common controls infrastructure, including support and maintenance, to both PS and SL divisions.

Several sub-projects are now running and the SPS-2001 is relying on the following:

3.2 The JAVA API project

This project is responsible for defining a standard accelerator device model and then to build a Java Application Interface (API) deploying this model. A first version of the model is now available [6] as well as a first JAVA implementation actually used for some AD applications [7]

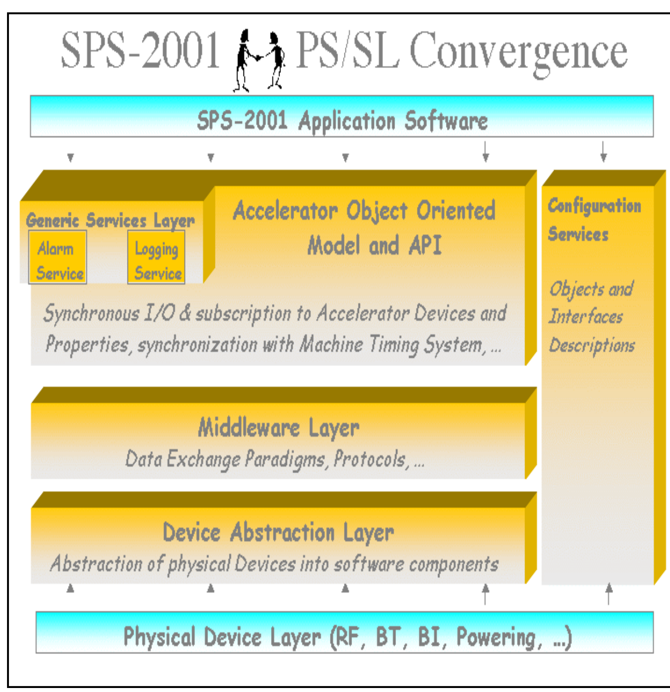
3.3 The middleware project

This project is responsible for providing a Software Communication Architecture and services allowing inter-objects communication and supporting, as first priority, the CERN Java API Technical Specification, including the Standard Accelerator Device Model [8].

3.4 The timing project

This project is responsible for building a reciprocal understanding of the two timing systems, of maximizing the sharing of the analysis, design and implementation of timing hardware, firmware and software components and of providing the new central timing facility required to operate the CERN accelerator complex in the LHC era [9].

3.5 Graphical representation



4. WHAT WILL CHANGE

4.1 New Sequence Concept

4.1.1 Beam-process

A beam-process is a specific manipulation of the beam within the context of an accelerator cycle. Examples of beam-processes are transfer, injection, capture/bunch, debunch, cooling, acceleration, deceleration, store, extraction, dump, etc. Beam-processes are usually limited in space (transfer zone, ring) and/or time (injection, acceleration, extraction). However, this does not exclude that beam-processes may overlap both in time and space within a given cycle. In terms of accelerator control it may have particular interest to differentiate the beam-processes in a cycle when there exist independent control parameter. (e.g. transfer line settings, extraction bumps). A beam-process is defined by its characteristics. The major ones are the location (i.e. the parts of the machine involved), the time-extend and the set of associated beam control parameters. Furthermore, one can usually define explicit and independent performance indicators to beam-processes. (e.g. transmission, emittance blow-up).

4.1.2 Cycle

A Cycle is a predefined set of beam-processes to transport (and transform) beam from source to destination. These beam-processes can have a predefined fixed length or they can have an undetermined length (accumulation, storage). The number of beam-processes in an accelerator cycle does not have to be predefined either; an example is multiple injection for beam accumulation. The evolution of one beam-process to the next in a Cycle can be predetermined, or under control of an external process (control-sequencer or operator). The parameters of a Cycle can be prescribed or adaptable (i.e. in final energy, extraction intensities). In the case of the present SPS control system, the cycle parameters are prescribed, and the evolution of the beam-processes is predetermined. The only exception, where the beam-process evolution is not predetermined, is in case the SPS is put into coast.

4.1.3 Cycle vs. beam-process

Each beam-process has a set of associated equipment to control the beam-process. However, several beam-processes can share the same equipment even at the same time during a cycle. At present, in the SPS, the equipment functions that are loaded in the hardware and that control the beam, are defined on a cycle basis rather than on the basis of individual beam-processes. In fact, the concept of beam-processes has not a firmly defined foundation in the present SPS control software. In the future, virtual equipment control channels (i.e. channels acting on the same equipment but controlled by independent functions and timing events), may make it possible that certain beam-processes could be implemented with their own associated equipment functions.

4.1.4 Sequence

A sequence is a predefined group of cycles. An accelerator can be multi-cycling with several cycles executed in a sequence. The sequence of cycles can be a fixed repetitive pattern (super cycle) or a more flexible sequence of cycles adapting to the status of the users and equipment. To speed up sequence switching, several sequences may be resident in the equipment, and executable under control of the timing system. The beam scheduling system will respond appropriately to changing interlock and request conditions in order to select the sequence that makes best use of the available resources. One should further note that a given cycle could be present in more than one sequence.

For the case where the evolution of beam-processes is predetermined, it is useful to make the following refinement in the definitions of sequences:

Predefined Sequence

The predefined sequences are the sequences that are known to the control system as legitimate combinations of cycles.

Resident Sequence

A sequence is resident if all constituting cycles and cycle linkage parameters are resident. A cycle is resident if all the required equipment settings are resident in the hardware and the cycle is ready for execution.

Active Sequence

The active sequence is the sequence that is currently active in the machine.

4.2 Simple view of the new operating strategy

1.Before each period (TBD) physicists and OP crew define the physics program of the next period (TBD). This physics program will allow Users to issue Beam Requests.

2.All the corresponding sequences that should be used to handle this physics program are selected from the sequence database and the sequence settings are made resident in the SPS equipment.

3.Then, depending on the user and operators requests and the possible machine state, the active sequence is chosen within the resident sequences depending on the priorities of the client requests and operators veto. NOTE: This choice of the active sequence can be made via human intervention or automatic.

4.The central timing system is now directing the scene by "playing" the active sequence in loop until another sequence is chosen.

5.NOTE: The equipment themselves are not aware of which sequence is running. They have been loaded with the settings corresponding to the possible Cycles that could be run. The local timing system will warn them that the Cycle x will be played soon and then send all the timing events/telegrams for the running Cycle

6.During operation, things may change either in the user or operators requests or in the current machine state (for example due to interlocks). Therefore, the active sequence may well not be relevant anymore and then a new active sequence will be selected as explained in 3 above.

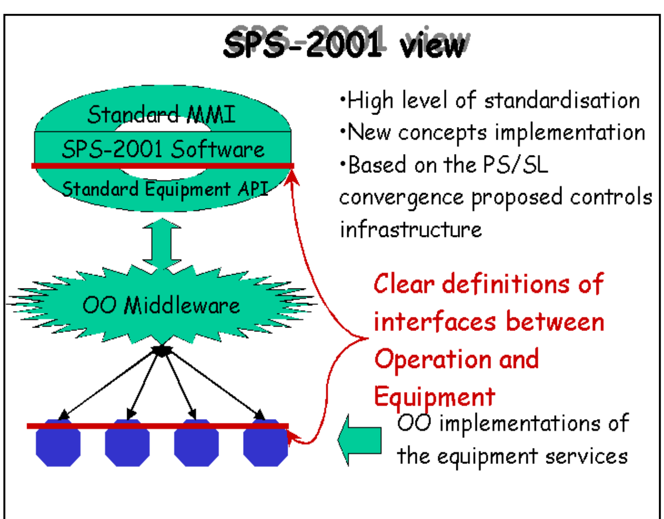
4.3 SPS-2001 Equipment Interface

The SPS-2001 project proposes to define a standard equipment interface that will be deployed on top of the proposed architecture given by the middleware project. This standard interface will be defined in conjunction with the equipment groups and the operation crew and will be the base of the SPS-2001 application software. It will define services that equipment can provide and also services that operation would like to see provided by the equipment. These services will be classified into groups of services, such as identification, global status information, state management, cycle management, Cycle parameters, cycle functions, measurement parameters or information subscription.

The definition and implementation of this standard equipment interface is one of the next important task of the SPS-2001 project. It is expected to have a first version of the definition by June 1999 with some prototyping activities until December 1999 and the final implementation for June 2001.

4.4 Standard Man Machine Interface

Another piece of standardization that the SPS-2001 project would like to perform is the definition of a standard Man Machine Interface. This MMI will offer to the operation crew a standard environment and therefore an homogeneous view of the SPS operation. The definition and the implementation of this standard MMI will be carried out in the next months in close collaboration with the Operational teams.



5. THE APPLICATION SOFTWARE

The application software represents all what is between the MMI and the equipment interface and represents the biggest part of the SPS-2001 project. An Object Oriented Analysis and Design phase is currently under way [10] to define and then implement these applications.

The purpose of the Analysis phase is to provide a complete, consistent, readable, and reviewable **description** of the software needed to operate the SPS accelerator in the LHC era.

The objectives of the Analysis exercise are to describe the functions the future SPS software will perform and to state what kind of performance and resources are required.

The scope of the analysis will cover all behavioural aspects of the future software, not its form. Issues related to software component design and implementation will not be pursued during this phase.

The actions required to reach the objectives of the SPS 2001 analysis phase will be organized around two main ideas, the identification of the functions the future system has to provide and the analysis of the behaviour of the system for each function.

4.4.1 Identifying System Functions

This activity will focus on the identification - from the requirements point of view - of the functions the future system has to provide. We consider functions as the expression of the need rather than the expression of solutions. The aim of expressing the need through its functions is to find a better way to breakdown the concept and to clarify the perception of the problem. Once identified, the functions are then clustered according to hierarchies.

4.4.2 Behaviour Analysis

This key activity of the analysis phase will concentrate on running practical scenarios for each function of the system. These scenarios will help to find the entities the future software will be composed of. Next, the relations between these entities and the services they provide will be studied in greater detail. Initiators of these scenarios (ie. operators, external triggers) and contributors (ie. external service providers) will also be identified during this behavioural analysis phase. This identification will help, amongst other things, to specify interfaces with external hardware or software system required to run the SPS as LHC injector.

Analysis will be successfully complete when scenarios for all fundamental (central to the application's purpose) system behaviours will have been developed and validated by the experts in accelerator operations and people responsible for the external systems that will interface with the future SPS software.

6. CONCLUSIONS

The SPS-2001 software project is an ambitious project to handle the new LHC requirements for the SPS as LHC injector and also to provide a coherent application software suite to operate the SPS. This task could not be made possible without an attentive analysis of the needs. But neither would it be possible without a good separation of responsibilities between all the parties involved. The definition of standard equipment access as well as standard Man Machine Interface will ease the main task of writing the application software.

Finally, this project must be seen as a divisional effort in the SL division and the collaboration of all equipment groups as well as the operation crew is essential.

REFERENCES

[1] The SPS-2001 project - Web site http://venice.cern.ch/~slaps/hot/sps_lhc/intro.html

[2] The SLI project - Web site http://nicewww.cern.ch/sl/sli/Sli.html

[3] The PS/SL convergence project - Web site http://hpslweb.cern.ch/pssl

[4] Mandate for COM K.Hubner AC/KH/dl 96-83

[5] Collaboration of PS and SL Controls Groups K.H. Kissler, D.J. Simon CERN Memorandum PS/DI/Memo 97-37 - SL/DI/Memo 97-23

[6] Java Controls API Technical Specification - PS/CO/Note 98-16 - P. Charrue, J. Cuperus, I. Deloose, F. Di Maio, K. Kostro, M. Vanden Eynden, W. Watson - 11 August 1998

[7] The Java API - Web site http://hpslweb/pssl/projects/javapi/javapi.html

[8] The middleware project - Web site http://hpslweb/pssl/projects/middleware/middleware.html

[9] The timing project - Web site - http://hpslweb/pssl/projects/timing/timing.html

[10] The Analysis Phase Plan of the SPS-2001 project - http://venice/~slaps/hot/sps_lhc/docs/docs.hml