

PS/CO/MIN 87-009  
26 February 1987

Project: NAPS  
Domain: SMACC  
Category: MINUTES  
Status: FINAL

*MEETING ON LANGUAGE SUPPORT FOR THE SMACC*

Cl.H.Sicard

Abstract

Minutes of a meeting on Feb 24th, 1987.

Present: L.Casalegno, G.Cuisinier, F.di Maio, A.Gagnaire, B.Kuiper  
N.de Metz-Noblat, F.Perriollat, R.Cailliau, A.Daneels, C.H.Sicard.

Purpose: Discuss the short and medium term strategies for language support on the Smacc, based on current experience.

Geneva, Switzerland

## 1 Objectives

This meeting attempts to define a minimal number of production chains, with their associated language, which could be supported by PS-CO for the different categories of programmers involved in PS Controls.

The Smacc application programs fall broadly in 2 categories:

-Equipment-modules(Property-codes, e.g. routines called with structured parameters, doing mainly arithmetic or table handling.

-Real-time tasks, communicating in addition with RMS and output devices (terminal or video displays).

The emphasis is on production chains, i.e. a set of tools to edit, compile, link and test Smacc code, because:

- 1) There has been a certain proliferation of these chains, used by one or a few persons, and adapted to a limited number of tasks. Although they have been implemented at the time for good reasons, the associated procedures are not documented, the total time for their maintenance is not negligible.
- 2) The efficiency of a production chain has an effect on the total program productivity.
- 3) These chains often involve other development computers than the PRDEV, for which a coherent support strategy should be defined.
- 4) for users outside CO, the language itself is less a 'knowledge barrier' than the intricate procedures implying several machines and user-interfaces.

## 2 The currently used production chains

They are ordered according to the computer on which resides the compiler.

### 2.1 PRDEV

This machine contains the final part of the production chain Assembler, linker, pusher, Create-Image.

Its advantages are: same user interface as the FECs (used for on-line tests), fastest file transfer to FECs.

Its disadvantages are: practically no support tools provided by ND.

- 1) PLANC; Used to produce Datagram system in Smacc and Fec with same code- generates code for 68000,ND100,ND500. ND supported, produces CUF (via converter pgm). Not compatible with Standard Cern, interface routines needed for each procedure. No 48-bit format for 68000.
- 2) P+ Vers.B; Used for POW Eq-module in Smacc, plus several other waiting. Not complete yet (no string handling, i/o) Future support not to be expected after Robert's departure to DD.
- 3) RT-FORTRAN 77;

(H.von der Schmitt, Univ.Heidelberg). Generates assembler for 68000. No 48-bit floats.

## 2.2 PRIAM/UNIX

Advantages: good off-line test facilities and support tools. All languages provided follow the Cern Standard. Disadvantages: file transfer via Cernet, heavy load, different user interface.

- 1) C; Compiler made by outside company(ACE) for CERN, generates assembler. Used for the GM and PPM-tasks in Smacc. Small problems: External references are truncated to 7 chars, and preceded by an underline character;
- 2) PASCAL; Siemens compiler, modified by H.von Eicken. Generates directly CUF. (speed advantage).Has been tested for Procos.
- 3) Fortran77; Not the same library as Pascal.
- 4) Modula2; The least well-known language, its support may decline in case of reduction of DD resources. Used at SPS (Nodal system,...)

## 2.3 IBM/VM-CMS

Not used currently, but as it is the most powerful machine, with continuing support, it is worth examining it as a candidate, although the tools and user interface it provides are not the best. Remote job entry could be run on it from the Prdev. It provides now a part of the Priam tools (Pascal, Modula2, Assembler +CUFOM processors).

## 2.4 VAX8800/VMS

This is a public machine, offering more power than Priam. The DEC program environment is advanced (language-sensitive editor, good debugging tools). No current tools to cross-compile for Smacc.

## 2.5 McINTOSH/Hard Disc

Being a single-user machine, it offers speed advantages. Its tools are adequate, but not as powerful as Unix or Vax/VMS. A Unix-like shell (MPW) allows to compile 'C' or Pascal code for the Smacc, with a certain number of restrictions:

- Modules cannot be more than 32K
- 'C' Stack format is compatible, but lacks standard error-return and Routine address as defined in Cern Standard-call convention.
- Pascal stack convention is different (McIntosh convention), thus not usable.
- Code generated must be reconverted to CUF (special program written).

### 3 The different categories of users

one can identify three main categories of users:

- 1) Stand-alone Smacc programmers. In this case, CO can only give suggestions and counseling, the Priam standard chains with Monica answer to these needs. For large projects, the MacIntosh/Hard disc chain can also be used.
- 2) Non-CO programmers for Smaccs in PS Controls.  
These users have an experience of a programming language, usually Pascal for young people, or Fortran for a decreasing part.  
The main difficulty for them is to be acquainted to the different environments (Editing, compiling, file transfer, tests), and for this reason it is best to propose the most homogeneous solutions.
- 3) CO programmers. In this case, the speed and availability of the production chain is more important, but everyone does not wish to learn too many different environments.

### 4 Selection of languages

Due to the limited support we can provide, it is proposed to restrict to the following list of languages:

- a) Pascal; This should be the normal language for application development in the future. No compliant row facility exists.
- b) C; For system work, it should replace assembler which had been used in the first Smacc developments. It has been used for the whole PPM task, including ISRs.
- c) Fortran; Not usable for Procos coding, due to lack of structured data handling. F.P. insists on the support of this language. It is agreed that it is not the highest priority, but it can be done if the additional work in porting libraries is not too big.
- d) ADA is not to be forgotten, as it may provide a much better environment than other languages.

### 5 Future production chains to be supported

F.P. does not want to import effort on Prdev (to migrate production chains currently on Priam). One solution is then to use the Priam for the first part of the production (compil+prelink), but it implies more resources on this machine. It would be important to improve the Priam speed, and to have reliable networking to it. (Cernet is sometimes down for half a day or more) Remote job submission from PRDEV would be an advantage. Some pressure on DD must be done by PS hierarchy if we go in this direction. If it is felt that Priam is declining, one could ask DD to provide the same service on IBM.

The McIntosh chain is used by F.Di Maio and is a good candidate for 'closed' projects (IK-Box, TSU-like). As the Mac is used more and more for user interaction, this must not be forgotten.

6 The problem of 48-bit floating-point format

It is clear that 48-bit format will not be supported by DD in the future. If we want to stick to the policy of using as much DD support as possible, we must tackle this problem as soon as possible. A meeting will be announced in the next weeks on this subject.

7 Some figures for production times

A comparison of the time needed to compile and link the same task has been made: 10mn on Priam and 1mn30 on MacIntosh. Another test with native C compilers on Priam and PRDEV gave similar compilation times between the two machines.

Distribution: List 2a