

**Abstract of the Proceedings at the SEAS APL Working Committee's Meeting  
Grenoble, Feb. 15th, 16th/71**

Mr. Yves Leborgne, after giving a history of APL at the Research Centre, LaGaude, listed the following 10 current applications at that centre:

1. Financial Analysis
2. Budget Systems
3. Simulation of Hardware
4. Software Design – here “poor and simple” APL aids fast conversion to Assembler
5. Project Planning – 1050 card input recommended
6. Text Applications
7. Review Distribution
8. Documentation
9. Personnel File System – again 1050 card input
10. Process Control

It was stated that programs in APL are so easy to write that it is often easier to make new programs than to modify old ones.

At the IBM Centre in Copenhagen an information retrieval system has been developed in APL.

Mr. Abrams told how to get the backspace character in the XM6 product for use with the new high resolution typeball (No. 1167114)

```
 )COPY 314159 TEST KK  
BS+KK[16]
```

Now BS is the backspace character. Mr. Abrams also mentioned that backspace and overstrike both enter into the width count for an output line.

Mr. Rainer Kogan, IBM World Trade said that sometimes he does not get information about a new product until the time that it is announced in the US, resulting in later availability in World Trade. He also explained the difference between an “alpha announcement” when product tests still remain to be done and an “availability announcement” which gives a date for the shipping of a copy to the 6 distribution companies in World Trade. Further time is necessary for checking and duplication at the distribution company.

The meeting divided into 3 working groups. The first discussed installation problems which mostly concerned the XM1 product. It was the general impression that work concerning XM1 on several installations had been given up due to bad documentation and the announcement of XM6.

A second group considering application and teaching concluded that APL reduced the need for application packages – which are normally easy to develop when the need arises. They also suggested that any APL packages produced should be designed so

that they could be further developed by users. It was felt that basic training in EDP should be reshaped with more emphasis on information handling and less on flow-charting and the processing of scalars. The group also decided to circulate material through the chairman or secretary who would distribute it directly rather than start a European Quote-Quad.

Adin Falkoff of the IBM Scientific Centre, Philadelphia gave a talk on the use of APL in teaching and system design. He emphasized the inspiring effect APL produces in giving students an easy way of working with real problems. APL could be used as a substitute for Course-Writer – the advantage being that a student could be given control. He can then experiment with some functions analogously to performing a laboratory experiment.

MINIPERT designed in Assembler took 6 man years, in FORTRAN – 2 man years, in APL – 6 man months. APL tends to keep thinking simple in system design. Generality is important – APL is not committed to specific hardware or software, to a special logical system or specific data-types.

Mr. Falkoff stated that only a small part of OS had been modelled in APL. The experimental execute function implemented at the Philadelphia Scientific Centre does not execute system commands. Mr. Falkoff could not tell if or when IBM would implement an execute function with or without an ability to execute system commands.

Appendices to the report are:

- I List of Participants.
- II Article – “Selection of a Medium for Program Exchange”. This article is reproduced below.
- III APL notes for Principles of Transistors course from M.I.T.
- IV An algorithm using the backspace character and the number 1167114 typeball for plotting fine curves. This will be reproduced in the algorithms section.
- V A listing of the part numbers of the APL typeballs and the language BCD typeballs used with APL in various countries.

Anyone wishing for more information on this meeting should contact:

Niels Gellert, NEUCC,  
Tech. University of Denmark, Building 305,  
2800 Lyngby, Denmark.

#### **Selection of a medium for program exchange (Appendix II to SEAS APL Proceedings)**

Ed. Hendricks

During the SEAS-APL Working Committee meeting at NEUCC in October 1970, I was given the task of investigating the problem of “selection of a medium for program exchange”. Although I am not certain what might be involved in a proper investigation, I have given the topic some thought and I now offer a rather obvious suggestion and some comments.

I propose that all APL systems provide a mechanism for reading and writing source program in ordinary EBCDIC card image records. Of course, the actual physical medium could be cards, tape, disks, or most anything else. The actual procedures for the processing of the symbolic program records would involve some new system commands, no doubt, and details would necessarily vary between unlike processors. (I presume that a variety of APL language processors will begin to appear shortly). The advantages of card image source as a medium are obvious – program exchange between unlike APL