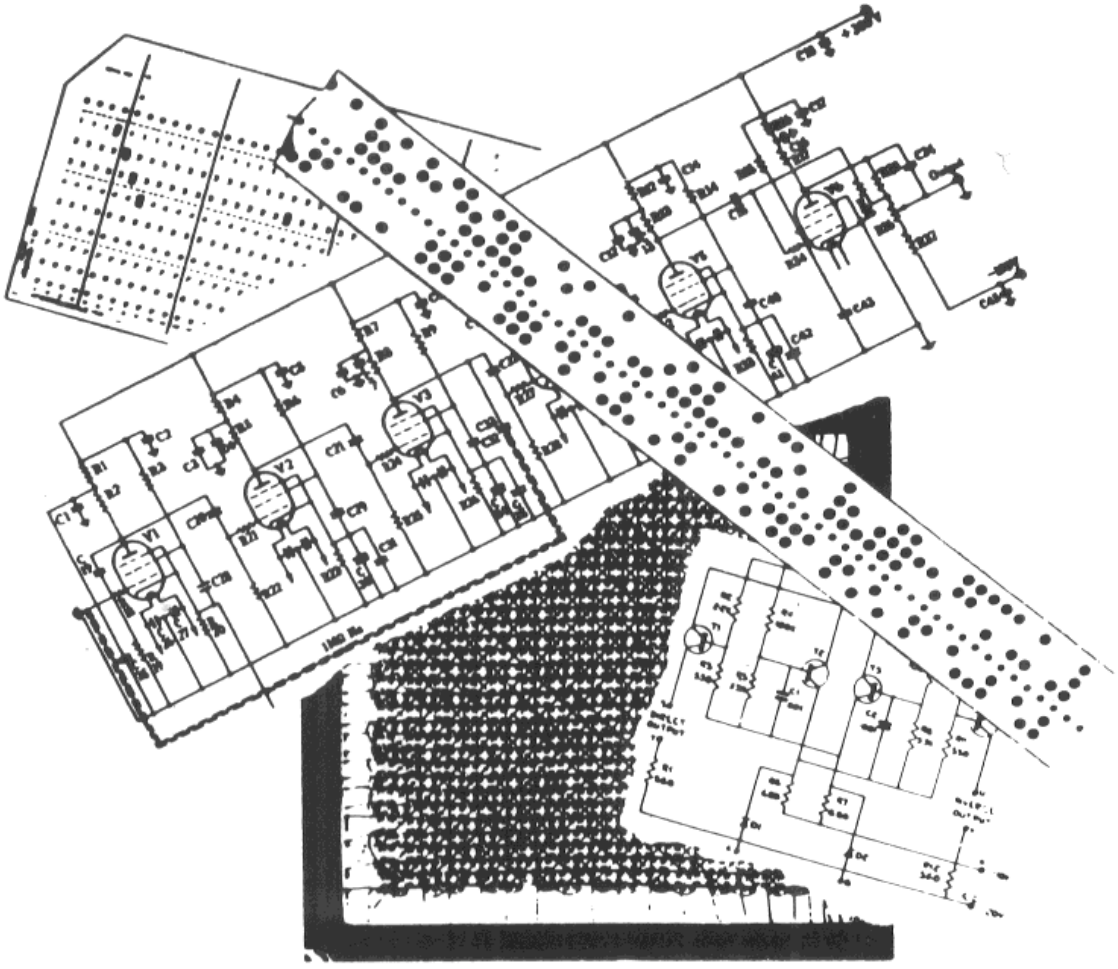
Issue Number 34 **Spring 2005**

Computer

RESURRECTION

The Bulletin of the Computer Conservation Society



science museum

BCS THE BRITISH COMPUTER SOCIETY

THE MUSEUM OF SCIENCE & INDUSTRY IN MANCHESTER

Computer Conservation Society

Aims and objectives

The Computer Conservation Society (CCS) is a co-operative venture between the British Computer Society, the Science Museum of London and the Museum of Science and Industry in Manchester.

The CCS was constituted in September 1989 as a Specialist Group of the British Computer Society (BCS). It is thus covered by the Royal Charter and charitable status of the BCS.

The aims of the CCS are to

◊ Promote the conservation of historic computers and to identify existing computers which may need to be archived in the future

◊ Develop awareness of the importance of historic computers

◊ Encourage research on historic computers and their impact on society

Membership is open to anyone interested in computer conservation and the history of computing.

The CCS is funded and supported by voluntary subscriptions from members, a grant from the BCS, fees from corporate membership, donations, and by the free use of Science Museum facilities. Some charges may be made for publications and attendance at seminars and conferences.

There are a number of active Working Parties on specific computer restorations and early computer technologies and software. Younger people are especially encouraged to take part in order to achieve skills transfer.

Resurrection

The Bulletin of the Computer Conservation Society

ISSN 0958-7403

Number 34

Spring 2005

Contents

News Round-Up **2**

Society Activity **3**

CCS Web Site Information **6**

Early Insurance Broking Applications **7**
Ernest Morris

Appeal for Help **14**
Hamish Carmichael

Pericles - A Pioneering Online Service **15**
Hamish Carmichael

Trojan Horses and National Champions **22**
Paul Gannon

Letters to the Editor **29**

Forthcoming Events **31**

News Round-Up

The Annual General Meeting of the Society will take place this year on 12 May.

101010101

Readers planning to visit Bletchley Park should note that admission prices have gone up. Tickets are now £10 for adults, £8 for children and concessions on most days. These prices are raised by £4 on days when there are additional attractions.

101010101

Jeffrey Darlington has retired from The National Archives, and consequently also from his positions as TNA's representative on the CCS Committee. His replacement is David Glover.

101010101

CCS member Andrew Wylie has published information about early British transistorised computer cards in his possession on the Web at <ourworld.compuserve.com/homepages/Andrew_Wylie/computer.htm>. He welcomes comments from fellow members.

101010101

Another member, Dik Leatherdale, has been writing an emulator for Atlas, the first computer he worked on. Although not complete, the emulator is demonstrable, and readers are invited to see it at Dik's Web site at <www.leatherdale.net>.

101010101

David Wheeler, the British software pioneer who played a key role in the development of Edsac, died in December 2004.

101010101

Ken Iverson, an IBM Fellow who invented the APL programming language, died in October 2004 aged 83.

101010101

Peter Twinn, a mathematician who became a leading member of the Bletchley Park codebreaking team during World War 2, died in October 2004 aged 88.

101010101

Society Activity

Mil-Dap Working Party

Brian M Russell

With the imminent departure of staff from the West Gorton site, we have had to remove the Mil-Dap and all other equipment from the site. As we have no other premises in the Manchester area for long-term work, we have reluctantly decided to close down the working party.

By the end of 2004, progress on the Mil-Dap had been minimal for some two and a half years. Our terms of reference state that if progress comes to a standstill, then control of the activity would be returned to the CCS committee. In the three years that the working party was active, we had succeeded in bringing together a complete set of parts for the Mil-Dap and its host Perq. The equipment has been physically cleaned up, tested for electrical safety and preserved in such a manner as to prevent further loss. It has not proved reasonably practicable to restore the machine to working order.

With the approval of the N-W Committee, the Mil-Dap hardware has been donated to The Science Museum by invoking the 'safety-net' arrangements that were put in place in 1999. This is in accordance with our duty to prevent further loss and to provide access to the machine for interested parties. The machine was collected from West Gorton by The Science Museum on 20 October 2004, after which it went to their transit stores at Blyth House. Eventually, it will probably go to their outstation at Wroughton. Access can be arranged –contact me for details.

There remain in the possession of the working party two further machines, a Perq-1 and a Perq-2. The Perq-1, donated by David Hunt of the lately closed CPP, is thought to contain Dap software. There is still some hope of archiving the software and forwarding a copy to The Science Museum. The Perq-2, donated by Jon Gogan, is merely a useful tool. These machines were taken to The Museum of Science and Industry in Manchester for a short time in November/December last year, when we made some progress in understanding what must be done to recover the software. The two machines have now been moved to Chris Burton's place at Llansilin for a last attempt at recovering the software. In a few months time the machines will be given away. John Wild has asked for the Perq-2. The destination for the Perq-1 has yet to be agreed.

I would like to record my thanks to Bob Whittaker and other colleagues who helped with the Mil-Dap, to Jenny Wetton and Jim Gledhill at the Museum of Science and Industry in Manchester and to Chris Burton. Whatever the results of the software recovery exercise, once both machines have been given away the working party will be formally wound up.

Contact Brian Russell at <brian.russell@iclway.co.uk>.

Small-Scale Experimental Machine

Chris Burton

Weekly demonstrations continue with an enthusiastic band of volunteers talking to the public about the machine and its history. Plans are afoot to move the whole installation to a new site on a lower floor of the 1830 Warehouse building, probably in early April.

Contact Chris Burton at <c.p.b@envex.demon.co.uk>.

Pegasus Working Party

Len Hewitt & Peter Holland

Pegasus has continued to work with very few problems and, with few exceptions, has been run every two weeks.

The Creed paper tape equipment has been restored to life by the services of two Creed engineers, Henry Lovell and Derek Hewson. However, there remain problems with the tape editing sets, which need further work. In one case spare parts are required which we have been unable to find.

We are very grateful to the donors of the Pegasus manuals and other literature which we have been given over the past few months. Anyone thinking of getting rid of similar items is asked to remember us first, please.

Contact Peter Holland at <peterholland@care4free.net>.

Bombe Rebuild Project

John Harper

It has been almost a year since my last report in *Resurrection* Number 33 so there is a great deal to report but I will try to keep things short.

First I will report on the items mentioned in my last report. All 36 Letchworth Enigmas are now fully fitted and tested. All external covers are complete, finished in black crackle and fitted. In fact if we close the rear gate and door the machine looks complete.

We have solved the supply problems of Special Wirewound Non Inductive Resistors. With help, we managed to track down the full number required, recovered from obsolete telephone exchanges. Most of these are of the incorrect value and we will have to rewind them. This however is now lined up having now obtained after much searching the correct gauge of insulated resistance wire.

The lubrication system is now complete and installed. The amount of pipework visible when the back of the machine is open is considerable. With something over 500 joints making up the feed to over 160 places around the machine, this has been a major task for the team members involved.

We have just got most of the running circuit working correctly, but not yet including the Stop and Cancel system. This is an arrangement of commutators and relays that allow the machine to stop on later cycles at the same point as the ‘Stop’ occurred. We have recovered relays but they do not have the correct contact configuration. The new contacts need tungsten tips and we are currently obtaining professional help with the remanufacture of these. Hopefully we will solve this problem in the next couple of months.

Drum manufacture continues at a steady pace, with 12 sets of five about to be completed. Many parts to make up the full set of 200 are also complete and ready for assembly when other parts are completed.

Around one third of the Cross and Connect plugs, needed to plug up a menu economically, are very close to completion. With about two thirds of the menu cables and one of the three reflector plugboards complete, we are only short of sensing relays to be able to start testing the machine with a genuine menu. We only need one third of the machine working to find Enigma settings. The rest can come later as parts are completed.

The sense relay remanufacture presents no supply problems. It is just a matter of making and assembling parts now to hand. We are tentatively looking at this autumn for the first batch becoming available, but please nobody try to hold us to this!

In one sense we will have 'cheated' in getting one third of the machine working because we will have bypassed a group of relays that route the stop letter to the indicator unit. These will still have to be made before all three banks can operate at the same time. These groups of seven relays provide 78 change over contacts and therefore constitute a considerable amount of effort. Just to give you some idea we need over 20 metres of spring steel for the spring leaves!

Our Web site can be accessed at *www.jharper.demon.co.uk/bombe1.htm*.

CCS Web Site Information

The Society has its own World Wide Web (WWW) site: it is located at <www.bcs.org.uk/sg/ccs>. This is in addition to the FTP site at <ftp.cs.man.ac.uk/pub/CCS-Archive> (please note that this latter URL is case-sensitive). Readers wishing to access past issues of *Resurrection* should use this FTP site, which thanks to sterling work by Chris Burton over the past year now contains a copy of every past issue. The Web site includes information about the SSEM project as well. Readers can download files, including simulators for historic machines.

Early Insurance Broking Applications

Ernest Morris

In my last address as Chairman of CCS in May 2003 I reminded the Society that unless we, the early practitioners in computing, recorded our experiences of the early days they would be lost to history and no-one would know the extent of the challenges faced at the time in making the technology work nor, more importantly, our achievements in doing so. This article is an attempt to record two of the applications in which I was involved in the late fifties and early sixties, and to note some of the characteristics of computing applications of the day using first and second generation computers.

From September 1954 to September 1958 I was employed by British Tabulating Machine Co Ltd (BTM) as a Technical Serviceman (a position that would now be called a Systems Engineer) which, after a year's training in the company's punched card tabulating equipment, saw me assigned to the company's City of London branch. The purpose of the position was to help customers plan for, design and install applications following the sale of punched card (pc) equipment by the company's salesmen. It seemed usually, indeed almost always, to be a case of squeezing a quart into a pint pot, because however capable the pc technology, the demands of customer applications seemed to exceed them. Nevertheless, punched card equipment contributed valuably to business efficiency.

In 1956 1 was inducted into the mysteries of computers via the 1201, and became involved with some early installations. One of them, CT Bowring & Co (Insurance) Ltd, persuaded me to leave BTM (at the time it announced a merger with Powers Samas) to become the 'Leader of the Computer Study Team' (a title later changed to EDP and O&M Manager). My new employer was one of the largest Lloyd's insurance brokers in the London market, a part of the C T Bowring (CTB) group which had shipping and general trading subsidiaries as well as several insurance-related activities[1].

As an insurance broker CTB was a significant customer of BTM, using its pc equipment of tabulators and calculators for several applications, mainly in accounting. In 1958 it had on order a 1201 computer for the application which I shall describe.

[1] The insurance broking activity is now part of Marsh & McClennan, an American company.

A brief description of the background to the application, for readers who are not familiar with the world of insurance broking, was included in a special supplement to *The Times* of 7 January 1966 (by which time CTB's computing story had moved on to another phase – but that is another story) in the following words:

> "Insurance broking is a very complex and highly technical business, the development of which in this country is primarily due to the unique underwriting institution known as Lloyd's. Essentially the insurance broker's function is to arrange the underwriting of risks for those who seek cover. In fact, the broker performs many of the services associated with the business for both client and underwriter… The broker must have the technical knowledge and resources to ascertain all the relevant factors and present the proposition in terms which enable the underwriter to quote a premium… Having placed the risk at Lloyd's, the broker attends to drawing up the policy, handling the accounting and paying the underwriters… Since the Lloyd's broker is not permitted to advertise or canvas for clients, continuity and growth of his business depends on his reputation for integrity, competence and efficiency."

Broking Application

While the complexity of this activity included a variety of accounting and statistical elements, only some of which used pc equipment in the fifties, the main purpose for which the computer was ordered was the preparation of the documents which recorded the successful placing of the risk at Lloyd's, this forming the basis of all subsequent accounting records and documents.

The start of the risk-placing operation was the completion at Lloyd's of the 'underwriting slip'. Initially the slip was typed in a broking department with details of client and risk, for clients worldwide and especially in the United States. The risks included ships and their cargoes, aeroplanes, buildings and many others, and often came to millions of dollars or pounds sterling. The slip would be manually annotated by each underwriter with the identifying number of the syndicate, reference, percentage of the risk accepted, and premium. A typical line written by an underwriter would have been 'syndicate number 904, reference X4824X, 12% of the total amount at risk accepted'. Note the free-form reference, which varied widely between syndicates.

Usually, up to 100 of these slips would be completed daily. Although this sounds insubstantial, each slip could have a number of entries (perhaps over 100) depending on the size of the risk, which for

marine risks could be many millions of dollars. The sequence in which underwriters wrote on the slip was not in syndicate number order, partly because some syndicates were recognised to be ‘leaders’ in a particular category and partly because the broker visited in turn those who were available for discussion at the time. The slip usually finished with over 100% of the risk underwritten.

In processing the slip, the broker’s accounts department had first to convert the ‘written’ line (12% in the example above) per syndicate to a percentage which accumulated to exactly 100% overall, and which was incorporated into the ‘signing slip’ for subsequent confirmation at Lloyd’s. This arithmetic calculation was carried out in CTB in the fifties by a comptometer section, adding to each line the amended percentage.

These calculations would then be entered into punched cards, containing CTB and client references (20 alpha characters, 10 client characters, the total sum assured and premium due), and per underwriting syndicate (on separate cards) a 3-digit syndicate number, a 10-character syndicate reference, a 5-digit written line and 3-digit currency code. With these cards the pc tabulators could produce ‘signing slips’ in the same sequence as the originals (although there was no recognisable sequencing number possible in the limited space of the punched card).

The main purpose of using a computer was to remove the necessity for the comptometer operation. However, equipment capacity limitations still applied. These comprised partly the 80-column limitation of the punched card which was the input medium for the 1201 as for the earlier equipment, and partly the capabilities of the computer itself.

BTM 1201 configuration

The main input/output device for the 1201 was the Type 589/0 machine, a variant of the 900 series tabulator, with card input and printer output. In addition there was a Type 582/4 Gang Punch for punched card output.

The main storage medium of the 1201 was a magnetic drum capable of holding 1024 40-bit words (there were just four words of immediate access storage). The drum held programs as well as data, and the conventional processing and control devices permitted arithmetic calculations and data comparisons and transfers to and from the drum.

Programs were written in 3-address mnemonic language and converted to binary (mentally or manually), punched into 80-column cards and input via the tabulator. The only packaged software was a ‘boot-up’ couple of instructions to start the machine off. Subsequent

operations were controlled by program plus the occasional manual use of appropriate switches.

Although it was a big improvement in capability over tabulating equipment, the 1201 computer still struggled to deal with replacing a clerical/comptometer-based operation. The problem was the indeterminate number of cards, each representing a written line, all of which had to be stored before reducing the accepted percentage to 100% and calculating the premium due to each syndicate.

Fortunately, in 1958-59 the 1201 was upgraded to a 1202 with 4096 words of drum storage, and this helped – so an upgraded order was placed (more expensively, of course).

With this extra capacity, the broking application became possible, but it was still a challenge. As eventually operated, the system comprised the following actions:

- load into the tabulator a hopper-full of slips/worth of punched cards (maybe 1000 cards in all, depending on the number of slips);
- input these and accumulate per slip the written lines;
- calculate these amounts and reduce them to part of 100% per slip;
- transfer the hopper-full of cards into the punch and output into them the newly calculated lines;
- transfer the cards into the tabulator, and read in the details including the newly calculated percentages;
- calculate the premium due to each percentage;
- print a ‘signing slip’ with the new ‘signed’ lines showing the original information and premium due;
- punch the premium due per line into the relevant card.

The printed documents could then be sent to Lloyd’s for processing, and the punched cards processed further in accounting systems for clients and underwriters.

This system worked well for some five years from mid-1960 (except for the low mean time between failure of valves, which was often measured in hours).

The big operational worry was that as there was no sequence number on the punched cards, they had to be kept strictly in the same order

throughout processing, and the fear was that a tray of some 2000 cards might be dropped. The cards would then have to be put back in order manually by reference to original paper documents. In practice, I seem to remember this calamity happening only once or possibly twice, so we were never really affected.

Other applications

In addition to the broking application, the 1202 successfully ran some accounting and statistical programs. The design and programming team of two made an enormous contribution in all this. My own direct effort was focussed on design and debugging. I was no good at programming but was great at debugging – flicking binary switches to find out what was going wrong in program testing, usually overnight!

We also transferred from the BTM 542 calculator and tabulator the monthly gross-to-net payroll program which I had installed in the late fifties. This was not complicated because the allowances and deductions were few, but I had to work out our own method of calculating PAYE (as I had for the 542), as the Inland Revenue provided only printed tax tables at the time.

The application of which I have fond and proud memories, however, is one that was ground-breaking in the early sixties. The group had a subsidiary in Canada – Bowring Brothers in Newfoundland, whither Benjamin Bowring had emigrated in 1816.

The subsidiary had several boutique stores in airports in mainland Canada and a department store in St John's, Newfoundland. A visit by the MD, Derrick Bowring, to our installation in 1960 set in train a successful use of the 1202 and its replacement computer for several years.

This was a sales analysis and stock control application, with weekly analyses of best and worst sellers in the various parts of the store. It doesn't sound impressive by today's standards, especially as the stock control function never seemed to be up-to-date and accurate. However, the impressive part to me is that we successfully bridged the 3000 mile gap between England and Canada to produce meaningful information for local profit generation.

Cash registers in several departments in the St John's store produced paper tape from Kimball tags of all the transactions, recording product code, quantity sold and price. These tapes were air-lifted weekly by British Airways to London, converted into punched cards and input to the 1202. The printed sales analyses were then mailed back to Newfoundland before the week was out.

I don't know of any comparable application service of its time. It was enthusiastically received by the store's management for its insight into sales patterns. It worked well for several years and became the model for an eventual local computer system.

Postscript

After three years, it was decided in early 1963 to replace the 1202 with a third generation computer which incorporated magnetic tapes and thereby enabled a far more comprehensive application which integrated the original broking application with follow-on accounting applications. This led to the installation of a Honeywell 400 in March 1965… but that, as I said at the beginning, is another story!

I started this article with a reference to the desirability of recognising the challenges which were faced and overcome in the early days of computing. The most evident of these in the above account of applications at CTB was in the capacity and reliability of first and second generation machines.

Despite the enormous increase in capacity and flexibility over punched card equipment it was a stretching challenge to accommodate applications. Even when third generation computers came along, this was still a challenge. For example, at CTB much of the work done in broking departments to prepare documentation for clients and underwriters describing the risks continued to be done in that way, using typewriters and in some cases Friden Flexowriters. Using these, after the third-generation computer was installed in 1965, data punched into paper tape could be input directly to the computer.

A similar challenge existed in software, which was not known by that name at the time. The days of 'application packages' were in the future – even 'library sub-routines' for organising and administering data were unknown and therefore had to be created from scratch for individual applications. As I mentioned, we had to write everything except the initial couple of instructions to start up the computer.

In those early days, naturally, there was no prior experience of computerisation, and that was another challenge. Programming was learned in a short course, usually (certainly in the case of CTB) by people already employed in the company. 'Systems analysis' was based on the established 'O&M process' – this was undoubtedly a good thing because of its disciplined approach to documenting existing methods and deriving new ones.

As a result of taking people from within the company to program the new methods, we did have the advantage that the work was, in the main, done by people who were familiar with the 'old' system. So although there was the gap in understanding of 'new technology' generally within the company, and (as later generations of programmers and analysts continued to experience) to some extent a failure to appreciate the merits of the new breed of worker, we had a good basis on which to try to get to grips with new techniques.

At CTB, a challenge we did not have was lack of top management support. The company was mainly directed by members of the Bowring family who firmly supported the computerisation processes, even though they were not familiar with its detail. The challenge of accepting responsibility for targets of time and cost existed then, of course, as it still does, but the support of the top management made our task much more bearable.

This became apparent particularly in the case of the application for Bowring Brothers in Newfoundland, because although the stock control element rapidly fell foul of the difficulties in ensuring accurate, up-to-date input, which meant that computer stock analyses were suspect (ie often plain wrong), the value of the sales analyses to the local management was highly appreciated in terms of its contribution to profit.

Editor's note: Contact Ernest Morris at Ernest.Morris@btinternet.com.

Appeal for help

Hamish Carmichael

Mayreau is a small island in the Grenadines, toward the southern end of the Windward Islands in the Caribbean. Population about 400. Nominally administered from St Vincent, but for most purposes it has to get by without outside help. It is a delightful place – friendly people, immaculately turned out schoolchildren, very steep road up through the village, and wonderful view from the top.

A friend of Peter Hall, Sir Bryan Thwaites, has personally sponsored the school there for a number of years. When he landed there during his annual sailing holiday last year, he was distressed to find that Hurricane Ivan had devastated the place and, in particular, had destroyed all the school's laptop computers. So he's looking for any secondhand laptops that may be available. I've promised to help by notifying the CCS membership in case anybody could help.

Can anyone who can spare a laptop contact Sir Bryan at <bryan.thwaites@btinternet.com>. A very good cause.

Editorial contact details

Readers wishing to contact the Editor may do so by email to <wk@nenticknap.fsnet.co.uk>.

Pericles – A Pioneering Online Service

Hamish Carmichael

The author spent the 1960s and 1970s in Corporate Systems, the department within ICL that looked after internal data processing, the applications that ran the fundamental administrative processes of the company. Here he describes the Pericles pioneering online service developed by his department around 1970, and the manifold obstacles encountered along the way.

Corporate Systems originated with several separate departments: Putney looked after sales, order processing, personnel, current planning and corporate reporting; Letchworth handled accounts; Stevenage, Kidsgrove and West Gorton dealt with their respective forms of manufacturing; Sydenham did all the spares stockholding. In the early days there was considerable rivalry, verging on hostility, between the various teams, and 'not invented here' was a guiding principle.

Dave Dace had the unenviable task of forming a single organisation with a common set of objectives out of these very disparate units. He succeeded remarkably well in establishing a community of friends, by emphasising the common nature of the problems we all faced, and by minimising the feeling that anyone was trying to take over anyone else. Mike Forrest later took this further, by rationalising our processing into several large data centres, by getting all our services running under George 3, and by improving the professionalism with which our services were run.

All the principal applications operated in batch processing mode, and there was little in common between them. Database management software had yet to be invented, and all programs were written in bespoke code, some of it in Cobol, but any parts that were performance critical in PLAN, the assembly-level language of the 1900 Series. Development times were slow; as an illustration, it was said that implementing any new spares stockholding policy typically took longer than the incumbency of the director who had called for the change.

There was never enough machine time available for testing new programs. Management had to be bludgeoned into sanctioning expenditure on machines even to run operational systems. Extra machine capacity for testing was seen as pushing it too far. In one crucial respect

we differed from all other users of ICL equipment: we did not have ICL's sales force and consultants arguing on our behalf for the resources we needed, or presenting an independent view of the business case for the expenditure we were recommending. In this respect top management were running in blinkers. I still think that overall they did a remarkably good job.

Testing a new program was therefore a case of assembling a sheet of printed instructions to the remote operators, the pack of punched cards representing the program, the parameters, and the data, and leaving or sending it to be run overnight. Too often one found the next morning no more than a program dump with the evidence of a single trivial mis-coding, instead of the expected stack of beautifully formatted output. There was an inevitable 24 hour wait between successive test runs.

This was compounded for us in Putney by the fact that we never had any dedicated computer facilities on site, and of course in those days there was no question of communicating with remote equipment. So, apart from a brief spell when we had the use of a computer room in the bomb-sight factory at Croydon, jobs for compilation of new programs or testing of programs under development had to be assembled toward the end of the afternoon, in order to catch the Letchworth van, and many were the panics in which people tried to delay Ted for a few more vital minutes before he left. Worse still was the disappointment if the job came back next morning un-run, because there hadn't been enough processing time available overnight.

We tried to assign priorities to the jobs going in the van, and it often seemed to us as though the remote operators took a perverse delight in disrupting our carefully reasoned sequencing. From 40 miles away we couldn't see that there hadn't been enough tape decks free overnight for any of the A jobs, but that they could clear off a run of the C jobs because all they needed was a corner of core store and some spooled printout.

Far too high a proportion of our so-called development resource, in programmers especially, was swallowed up in generating report programs to satisfy management requests. There was no formal RPG to simplify the task for us, so we specified our own, which used a formidable pile of parameters – punched in cards, of course – to specify record content, record selection rules, line formats, totalling, and all the rest.

This might have been a godsend, but unfortunately the task of writing it was assigned to Dataskil, the company's in-house consultancy and software house, and what they produced was the worst load of

unprofessional junk that could be imagined. I know, because I spent days debugging the horrid garbage.

One particular horror has remained with me as an example of the sloppiest possible coding: in PLAN it was legitimate to branch from one program segment to a labelled entry point in another segment; but only a demented ninny would branch from one segment to the instruction *one instruction after* the labelled entry point in another, without any comment or documentation whatsoever at either end of the jump.

The updating processes for virtually all the main data files culminated in complete file print-outs, often on multi-part paper with interleaved carbons, so that those with data responsibility for the files could satisfy many enquiries at the level of individual records by flipping through the latest print-out to the relevant page.

In accordance with all the rules of good systems analysis, when we were setting out on a new phase of computerisation, we used to try to find out, by prolonged and subtle questioning, what information the users would be likely to want to derive from their computerised files. And, surprise, surprise, we always got it wrong; they always ended up by asking for something different. In our naïve youth we never realised the extent to which managers were themselves driven by outside influences – what MacMillan called “Events, dear boy, events.” So one rule has remained with me over 40 years : all important management questions are the ones that are being asked for the first time.

One solution was Cafs, but that was at least a dozen years into the future.

The other was the arrival of the first video terminals, and the first basic software that accompanied them. Soon users were clamouring to have online access to their data, and we rushed to meet their requirements. So primitive online systems (and some of them were *very* primitive) were developed by every team, and each one served its own set of dedicated terminals. But could a Letchworth online user look at a Stevenage file? Of course not. Or even could a Putney user of Putney system *A* use his terminal to look at Putney system *B*? Not likely; he’d have to go to another terminal of the right flavour. There were no standards, and no sort of compatibility.

From this there emerged among us in Putney the realisation that what we should be aiming to provide was a single coherent online service, rather than a number of independent online applications. From this seed there grew the magnificent software which acquired the name

Pericles. As I recall it, the programmers who deserve the chief credit for its development were Brian Gosling, Richard Beverley and Jerome Tucker. If there should be other names in the list, I apologise. I can't either recall the sequence in which facilities were implemented, but after nearly 40 years that may not matter too much. Among those facilities I can recall these:

Every user had a unique username, and associated with it a list of the applications which he or she was allowed to access. I don't think we bothered with passwords; the world was still too innocent. Within an application, a user had a profile listing the permitted functions. The most frequent use of this was to distinguish users who had update privileges from those restricted to enquiry-only access.

A common format was enforced for certain common functions, which were usually denoted by a command of the form **nn*, where *nn* was any two characters. Thus sign-off was always **SO*, which meant signing off from the application and detaching from the whole service. By contrast **LT* meant logout from the application but leave the terminal attached to the service. **??* called up a primitive help screen. Another * command meant 'what functions am I allowed to execute within the current application ?'; the response was trimmed to accord with the user's access profile. **CA nnnn* meant change to application *nnnn*, which executed a logout from one application and a login to the other without all the palaver of disconnecting from the system and re-establishing a link to it.

We quickly encountered the risk of clashing updates, so Pericles acquired a record-locking mechanism. A record could be opened either 'for enquiry' or 'for update'. In the latter case, before it was opened the key of the requested record would be checked against the lock table. If it was found, the user would be told 'record not available – please try later'. If the check proved clear, the record would be opened, its key stored in the lock table, and the transaction could proceed. If a transaction failed with a 'record-not-available' condition, the system was subtle enough to unlock any records which had been locked in an earlier phase of the transaction, as a way of preventing deadlock conditions.

Obviously there was transaction logging, so that after any system failure files could be recovered from clean copies plus the logged transaction details. But this was taken a stage further, because certain transactions could perform multiple file updates, and there was a risk that an inopportune system failure in the middle of a transaction could leave the files in an inconsistent and irrecoverable state. All updated records

were therefore written, not to their original file, but to a delayed update file, and were only written back to their home file when the whole transaction was completed.

These mechanisms worked not only for simple transactions where all the work was completed within a single message-pair, but also for more complex multi-stage transactions, where continuity had to be preserved through a succession of related message-pairs.

(An unexpected hazard arose from the habit of referring to the delayed update files as "DUF" files. A harassed operator in the middle of the night, faced with a disc-space crisis, interpreted "DUF" as meaning useless, deleted the files, and neatly solved his problem. In the morning, for some reason, the online service wouldn't open!)

There were also commands which the operator or service controller could issue to control the service as a whole, or applications within it, or to communicate with users. For example, if the overnight processing of a file had not finished by the time the service was due to open, the operators could open the affected application in read-only mode with a copy of yesterday's file. Users could be warned to log out at the appropriate time, so that the application could be closed, the updated file be brought online, and the application re-opened. For such functions the bottom line of the screen could be used as a message line.

The system maintained a variety of performance statistics, without which it would have been much more difficult to tune it to provide an acceptable level of service.

It was also possible to incorporate some standard applications. Among these the most widely used was Scratchpad (not, I fear, an original name) which allowed each user a number of screens – one record to a screen – on which they could create, store and retrieve their own data. There was a way of reviewing print-outs online, subject to the difficulty of mapping the 120-character width of a line printer to the 80-character width of a screen. And the late Arnie Shaw of West Gorton developed I-Spy, an application which allowed a user to initiate a serial search of a specified file, with simple Boolean combinations of search conditions and simple formatting of hit records; this, however, was such an antisocial component of an online service that its use was generally very restricted.

Because the underlying communication software remained the standard ICT/ICL product version, very little change was needed to the modules of application code to allow them to run within a Pericles

environment instead of talking directly to the comms software. Soon we had Pericles services running at the Letchworth and Stevenage data centres and also at Kidsgrove and West Gorton.

In an ideal world, we should at this stage have ensured that Pericles was adopted as company standard software, and offered to customers for them all to have the same sort of advantages that we had experienced. But unfortunately we were too far ahead of the game: the developers of standard software did not understand the need for service-level facilities. Besides, we were only users, mere application-monkeys; what could we understand about the elevated subtleties of the world of real software? And if that did not settle the question, they could always argue – and they did – that because we were within ICL our problems were not the same as those of real users. A very few enterprising customers did enter into private contracts to take Pericles from us, but we were not set up to be a support organisation, so this fizzled out. On the whole an opportunity was lost.

Other downsides? Well, a multi-application Pericles service was a very large object. Obviously multi-threading gave it a degree of internal parallelism, and the construction of an application as a succession of 'beads' along a thread made the structure sufficiently modular to be manageable. But as far as the operating system knew, and as far as the compilation and consolidation processes were concerned, a Pericles service was a single program, and had to be created as a single whole.

This made it difficult to incorporate changes as quickly as they were required. New or changed facilities had to be rigorously tested in a development Pericles service, and could only be incorporated in the live service at a scheduled release date. If anything then went wrong, the whole service would have to revert to the previous state. The resulting discipline was probably good for us, though our users did not always approve.

We got as far as speculating: surely it ought to be possible to compile a corrected version of a defective module, append it to the existing object program file, and patch the relevant entry in the overlay table so that it refers to the revised version. But implementing anything of the sort would really have been treading on the toes of our software colleagues, and they made sure that we never found out how to attempt it.

And then along came the 2900, and we had to consider all over again whether a son-of-Pericles would be needed in a VME environment. But that is another story.

Why Pericles? Well, Dave Dace and I were walking one lunch time along the Upper Richmond Road, and he suggested that the new service ought to have a proper product name, something serious and respectable. "What about a Greek god or hero?" "I think most of them have been used already… but we could perhaps try Greek statesmen… such as Pericles[2]." And then it simultaneously hit us that the name had "ICL" smack in the middle, which made it ideal. Another 10 yards and we had completed the acronym : PERsonalised ICL Enquiry Service, and so it became. One of my school books contained a picture of the portrait bust by Cresilas of Pericles wearing the helmet of Athene, goddess of wisdom, and Bob Manning, the company artist in ICL House, turned that into a splendid logo, which completed everything.

Happy days !

Hamish Carmichael can be contacted at hamishc@globalnet.co.uk.

North West Group contact details

Chairman Tom Hinchliffe: Tel: 01663 765040.
Email: tom.h@dial.pipex.com
Secretary William Gunn Tel: 01663 764997.
Email: william.gunn@ntlworld.com
Science & Industry Museum representative Jenny Wetton, Museum of Science & Industry, Liverpool Road, Castlefield, Manchester M3 4JP. Tel: 0161 832 2244. Email: j.wetton@msim.org.uk

[2] A statesman of Athens in the 5th century BC, leader of the Athenian democracy during its glory years, which culminated in the Peloponnesian War against Sparta. At the end of its first year, he gave a memorable funeral oration at a service for the war dead, recorded in the first book of the*Histories* of Thucydides. A remarkable politician, general and orator, he died in 429 BC.

Trojan Horses and National Champions

Paul Gannon

The failure of Governments to provide adequate support to their emerging computer industries was not solely a UK phenomenon. Governments supported their nascent IT industries heavily in Germany and France as well. The author describes how European computer manufacture went through three phases – the emergence of individual companies, the enforced fusion of these companies into individual national champions, and the current period of pan-European cooperation.

I am going to concentrate on events in Germany and France, partly to keep the subject manageable, but also because these two countries are useful for comparisons with British government policy in support of the computer industry. Philips and Olivetti will enter the story later, fully-formed and ready to implode. In fact by 1955, computers had been designed and built in nine west European countries and three East European ones as well as in Britain: Belgium, Denmark, France, Italy, The Netherlands, Norway, Sweden, Switzerland, West Germany, the USSR, East Germany and Czechoslovakia.

I will start with a couple of comments about developments in wartime Germany. I will briefly mention two characters. The more important was Konrad Zuse, who built a series of electro-mechanical computing machines before and during the war, called Z1 to Z4. The Z3, which survived the war, came closest to the later concept of the computer, but was electro-mechanical. Zuse also designed and constructed some electro-mechanical control machines for guiding flying bombs.

The other character was Helmut Schreyer, who proposed an electronic machine with 1500 valves (about the same as Colossus I). Permission was refused to build this machine and Schreyer had to be satisfied with a small-scale experimental machine with 150 valves. This machine was either lost, destroyed or captured by the Red Army.

Zuse and Schreyer, for security reasons, knew nothing of each other's work. Zuse was a fair-weather Nazi and Schreyer a committed one – he fled to South America at the end of the war – but this did not

help either of them get support for serious development of automatic computing machines.

Immediately after the war Germany was in a state of widespread physical destruction. At places such as Nuremburg, the first job facing the workers at Siemens factories was to clean up bricks from the ruins in order to rebuild the factories before production could restart. Zuse was more concerned, he tells us in his autobiography, with where his next meal was coming from than with building computers. In any event, the Allied occupying powers imposed a ban on developing commercial computers.

In 1947 currency reform was introduced and the Marshall Plan began to aid recovery, leading to the *Wirtschaftswunder* – the economic miracle of the 1950s. By 1955 four scientific computers had been developed at three German research institutes and one at Zuse's company, Zuse KG. In the same year, the ban on developing commercial computers was lifted and four companies entered the market: Zuse KG, Siemens, SEL, and Telefunken.

Zuse KG's Z5 was the first full computer to be released to the market in 1955, followed by the Siemens 2002 in 1957, and computers from SEL in 1959 and Telefunken in 1961.

The result of this late entry into the market was that IBM was easily able to gain dominance in Germany. Additionally, there was very little government support for German computer companies in this period due to the *laissez faire* policies of the government.

In the early 1960s the *Deutsche Forschungs Gemeinschaft* – German Research Council – ordered four computers for use in German universities, one from each of the four German companies. Zuse later recalled that this small assistance was critical to his company's survival at this time.

But it was too little too late. By 1964 the industry was in crisis. SEL sold its computer operations to Siemens. Zuse KG was sold to Brown Boveri and then to Siemens, making the latter the major German computer company. Telefunken allied with AEG, creating Telefunken-AEG, to ensure at least one other German company.

Siemens, despite its deep pockets, needed help as it could not develop all the technologies it needed to keep up in the industry, so it signed a technical agreement with RCA of the USA. RCA was seen as a technology supplier and strategic competitor to IBM which continued to extend its share of the German market.

By 1965, thus, the German industry was in deep crisis.

Turning to France, there was also considerable war damage to be faced, which meant that there were no computers built in France until the early 1950s. The very first French computer was produced by a company called SEA, known as Cuba – *calculateur universel binaire à l'armament* – and was used for military purposes. Some other small French companies entered the market.

The main company was Bull. It was originally a paper company. It had been annoyed that IBM held the monopoly for supplying punched cards to leasers of its tabulating machines. To circumvent this, Bull purchased some Norwegian patents and entered the punched card machine industry.

It released two electronic calculators, the Gamma 51 and 52, in 1951 and 1957 respectively. The Gamma 52 did very well, selling over 1000 machines, and was especially popular in the French financial services industry. But Bull delayed releasing its first computer, the Gamma ET, until 1958. Despite obsolete delay line technology, it sold well, again recording high sales in the financial services industry.

This meant that France was the only country where IBM had been held at bay. "Euphoria reigned" according to one French economic historian, especially as it had happened without government support.

However, Bull was lured into developing a supercomputer, the Gamma 60. The project was too ambitious for the company which lacked the resources to develop an operating system and software. The company suffered massive losses and only 17 machines were sold. IBM started to make inroads into the financial services sector with its small 1401 computer, while Bull was distracted by the development of what was a status symbol.

Bull fell deeper into financial crisis in 1963 as losses on the Gamma 60 soared. The government refused financial help, but also initially refused to allow Bull to sell its computer operations to an American company. But in 1964 the government realised that it had no choice and allowed GE of the USA to take over Bull. As one observer put it, "The only French company capable of taking on IBM was no longer French".

IBM treated Europe as a single market and benefited from the economic structure created by the Treaty of Rome, whereas European companies continued to treat Europe as a series of separate markets and failed to gain the benefits.

In that same year, 1964, IBM released its Series 360, transforming early leadership of the global computer industry into an overwhelmingly dominant position for the next two decades.

We now enter the era of the National Champions.

Companies had to make two strategic decisions. First, whether to compete in a niche not dominated by IBM, or to take it head on in competition for the mainframe market. Second, whether to work alone or with partners.

By 1964 SEA and CAE had 7% of the French market and Siemens just 5% of the German market. This led to growing government concern in both countries. According to an influential writer of the times, Jean-Jacques Servan-Schreiber, "No area of industry can ever be independent so long as we rely on them for computers. If there is a battle for the future, it is the battle of the computer".

The French government changed policy and launched *Plan Calcul* to foster a French industry. The plan stated that "for France it is an absolute necessity to have in a short time a national computer company if the country wants to be master of its own destiny". SEA and CAE were merged into CII (*Compagnie Internationale pour l'Informatique*), the national champion.

At this stage GE-Bull was the 'enemy', although it had 20% of the French market.

Substantial support was given to CII. In 1968 it had its first computer, the Iris 50, based on US technology, and very shortly afterwards the Iris 80, using independent technology, but the two were incompatible, so there was no transfer from one system to the other. In effect the company was run by civil servants and administrators, with its objective not to make a profit but to design its own hardware for a French computer. After three years, market share had risen from 7% to 7.5%. The creation of CII had also ended talks between ICL and CAE. The national champion policy took priority over inter-European developments. Britain and Germany followed suit.

Germany in 1967 launched the EDP programme with funds of DM387 million, three-quarters of which went to Siemens and to Telefunken-AEG. This was a deliberate effort to create a German computer industry. The Ministry of Research was created, and it is worth noting that a significant part of the funds was given to training – 12% of the total.

Siemens received $120 million to design its own computer technology to complement RCA's technology, and allow it to move into wider areas of computing – automation control and so on – with the long-term aim of removing the dependence on US technology. There was also a "Buy German" policy. Altogether this programme was much more effective than the French programme. Siemens' market share went up from 5% to 13% between 1965 and 1969.

But some people were realising that the national champion policy was self-defeating. It was hurting European companies, but not American ones. In *Computer Weekly* in August 1971, Chris Hipwell wrote an article under the headline "National Policies are not enough".

So there was a growing realisation that no single European country could support an entire computer industry on its own, and Servan-Schreiber for example talked about a European technology community.

1969 was also the year in which Charles de Gaulle stepped down. This was seen as providing potential for a new era of co-operation and expansion of the EEC, eventually leading to the accession of Britain. So the first attempt at a 'European solution' was made in the late 1960s, and all the main European companies, ICL, CII, Siemens, Telefunken-AEG, Olivetti and Phillips were invited to join in on a common European computer design. Every single one of the companies rejected the idea. That ended the first attempt to create a common European policy.

There was an economic crisis at that time in the US. General Electric and RCA left the industry. General Electric's computer operations went to Honeywell and so Honeywell-Bull became a new French-American company. The government in France began to take a different attitude towards Bull, seeing it less as an enemy and more of a French company, but it had a dilemma over what to do with the ailing CII.

Similarly at the time ICL was talking to Burroughs, so by 1971 conditions were again ready for a European approach. ICL also talked with CII and with Nixdorf: neither of these talks came to anything. The other grouping was CII, Siemens, and later on Philips, under the name Unidata. Operations started in 1973 and formally the company lasted until 1975, but in reality it was stillborn.

1973 was the year of the flotation of the dollar, the oil shock, and the resumption of non-cooperative policies. There was also the inability of any of the three partners in Unidata to sacrifice their own technology for a common solution. Three corporate headquarters, three R&D centres, and unanimity needed on the board in matters major and minor.

The French government now saw Bull as a French national champion. CII. was taken over by Honeywell-Bull, and the French government acquired 51% of the company's French subsidiary. Siemens took over Telefunken-AEG. ICL was left on its own. So we had the three major European national champions created in the early to mid 1970s, and it was another 10 years before there were any further attempts on a European level.

Siemens went to Fujitsu for technology, but Honeywell-Bull and ICL remained firmly wedded to the idea of a nationally designed mainframe computer. According to one account about the French (this is definitely not true of ICL): "No-one ever took a look at what the rest of the world had to offer. There was hardly any reading of American technical reviews, and nobody attended professional conferences. It all had to come out of French organisations."

Siemens and Honeywell-Bull grew rapidly in the 1970s. At this stage they moved beyond ICL, which had previously been much larger than the continental companies. But both companies needed large amounts of government funds to keep up with the exploding technology, and again it was realised that the national champion policy was self-defeating, as it only applied in its own country.

In the 1980s, as we move into growing acceptance of the idea of government support, in Japan there was the VLSI project and the Fifth Generation Programme, all of which had significant effects on changing people's attitudes within Europe to government support. There was a new round of national support in France, called *Mission Filiere Electronique*. Honeywell-Bull was finally nationalised and became Bull. In Germany the *Informationstechnik* programme was worth DM3 billion over five years. In the UK we saw Alvey – £350 million for R&D and £100 million for training – none of which, to use Mrs Thatcher's phrase, was "new money"; it all came from other parts of the DTI or the Education budget.

But significantly at this time, some of the companies themselves saw the need for co-operation across borders, and started to back the idea of the European Community's research and development projects. At the end of 1982 they agreed in principle to support European-wide projects. The UK was against this development, but did not veto it. The first pilot projects were funded to the total of 11.5 million ECUs, a small amount, but in February 1984 the first framework programme was finally agreed and the sixth, or it might even be the seventh, framework programme is going on at the moment.

In 1984 Esprit was launched with 750 million ECUs and Race (Research in Advanced Communications in Europe) with 21 million ECUs. Esprit was the first programme with significant amounts of money for cross-border projects. It is important to remember that this was the time of the development of the European single market, the start of the telecommunications liberalisation process within the European community, and a welter of other economic modernisation measures within Europe. And, of course, this was the time when Jacques Delors was the Commission President and he drove a lot of this forward.

In 1987 the second framework programme was agreed. That was 1.5 billion ECUs for Esprit, twice the level of the first framework, while Race went up from 21 million ECUs to 462 million. There was a new programme – IT Applications – which won 105 million, and which spawned a whole series of subsequent programmes whose names may be familiar – Drive, AIM, Delta, Tedis, Impact, and so on. But despite the opportunities presented by the EC R&D programmes, the national champions failed to overcome their national horizons.

The irony is that by the time that European co-operation had got into motion the whole situation had changed. IBM's mainframe technology had been circumvented by newer technologies and not by better cheaper mainframes. So the European mainframe companies were left isolated and collapsing like dominoes in 1990. But in any event, the European national champions remained national in outlook.

The conclusion is not that government support is inevitably wrong. There are plenty of individual examples where it has been useful. And of course there is one supreme example where government support in many forms was very successful and that was in the United States. The failure of the European companies, and I think this applies to the UK as well, was for combined reasons such as the wrong strategic technical vision and government support that was too little, too late, and wrongly focussed, not to overlook a narrow national outlook in an ineluctably international technology.

Editor's note: This article in based on the talk given by the author to the CCS seminar on "The Background to Government Support" at the Science Museum on 20 January 2005.

Paul Gannon's book "Trojan Horses and National Champions: A History of the European Computing and Telecommunications Industry" was published in 1997 at RRP £30 (ISBN 0953028402). (Copies are available from the author at £12.99, including P&P: send cheques to 174 Leighton Road, London NW5 2RE).

Letters to the Editor

Dear Editor,

I am writing about the comment on page 2 in *Resurrection* issue 33 that "This year sees the Golden Jubilee of the Fortran programming language, which was first released in November 1954".

As Chairman of the BCS Fortran Specialist Group as well as a member of the CCS I am always ready to take any opportunity to raise the profile of Fortran, especially as a revised version of the ISO standard for the language, Fortan 2003, is due to be released. However I think that Golden Jubilee celebrations are a little premature.

I made enquiries with our Vice-Chairman and Archivist, David Muxworthy, who was also a founder member of the FSG in 1970, and he has provided me with the following information.

Design started in 1954, the specification was also published in 1954, the first manual appeared in 1956 and the first software was issued in 1957, with Fortran II software in 1958. IBM seems to regard 1957 as the official date as the company produced a film *The Origins of Fortran* in 1982, the 25th anniversary.

In Jean Sammet's invaluable *Programming Languages: History and Fundamentals* (Prentice Hall, 1969), she says that the earliest record she could find was *Preliminary Report: Specification for the IBM FORmula TRANslation System, FORTRAN* published as an IBM Applied Science Division report on 10 November 1954. This looks like the origin of the Computer Conservation Society's report. It is not in the title, but the design was explicitly for the IBM 704.

She goes on to say that the programmer's reference manual *The FORTRAN Automatic Coding System for the IBM 704 EDPM*, IBM doc 32-7026, was issued in October 1956 and a primer, *Programmer's Primer for the FORTRAN Automatic Coding System for the IBM 704*, IBM doc 32-0306-1, appeared in 1957.

Software for the IBM 704 was released "early in 1957", with Fortran II appearing for the 704 in June 1958 and later in the year for the 709 and 650. Fortran II introduced subroutines and the common block, amongst other things.

Other information from the book is as follows. The first Fortran system on a non-IBM machine was Altac, an extended Fortran II, on the Philco 2000 in 1960. The first system called Fortran on a non-IBM

machine was Fortran I on a Univac SS80 in 1961. By 1961 IBM had eight different compilers on different systems, and already language variations were causing such problems that IBM issued a manual contrasting the facilities in the eight compilers. A paper in *Datamation* (August 1964, pp 25-29) noted that 43 compilers existed *in toto*. The first meeting of what was to become the American Fortran standards committee J3 was held in May 1962.

Jean Sammet also notes that there was much opposition to Fortran at first on the grounds that the code generated by compilers would be inefficient relative to hand-coded assembler. I think compiling techniques must have developed very quickly. I worked on a Univac 1107 in the mid-1960s and the emitted code was generally very clever indeed – often better than one would normally have written in assembler.

My own much more limited knowledge also confirms David's conclusions. It is chiefly based on an article published in the *New York Times* on 13 June 2001 entitled *Pioneers of the 'Fortran' Programming Language*. It described how a team of 10 under Jim Backus at IBM began work in 1954 with a six month timescale to simplify programming and open up computing to more people. The timescale slipped to three years (some things don't change!) and Fortran was finally presented to the computing world in February 1957 at the Western Joint Computer Conference in Los Angeles. One of the points made in the first demonstrations was that Fortran programs could run as quickly as ones written in hand-coded assembler carrying out the same task.

So in human terms we feel that November 2004 was the Golden Jubilee of the conception of Fortran and February 2007 will be the Golden Jubilee of its birth.

We would be happy to discuss this further with other CCS colleagues. As one of our Committee members recently pointed out, there is now a T-shirt slogan saying *Fortran – A Language with a Past and a Future*, so we wish to build on the historic foundations of Fortran and to show that it is a language for the 21st century. For more information on the current and future features of Fortran see the FSG web site at *www.fortran.bcs.org*.

Yours sincerely,

Peter Crouch
Chairman, BCS Fortran Specialist Group
18 June 2004

Forthcoming Events

Every Tuesday at 1200 and 1400 Demonstrations of the replica Small-Scale Experimental Machine at Manchester Museum of Science and Industry.

Weekday afternoons and every weekend Guided tours and exhibitions at Bletchley Park, price £10.00, or £8.00 for children and concessions (prices are raised on some weekends when there are additional attractions). Exhibition of wartime code-breaking equipment and procedures, including the replica Colossus, plus 60 minute tours of the wartime buildings.

12 May 2005 Computer Conservation Society AGM

Starts at 1400 in Fellows' Library of Science Museum

12 May 2005 "Computing Before Computers"

London seminar immediately following AGM, to be chaired by Doron Swade, with overseas speakers as well as UK experts including Martin Campbell-Kelly, Mary Croarken, Pamela Vass and David Hogan.

Details are subject to change. Members wishing to attend any meeting are advised to check in the Diary section of the BCS Web site, or in the Events Diary columns of *Computing* and *Computer Weekly*, where accurate final details will be published nearer the time. London meetings take place in the Director's Suite of the Science Museum, starting at 1430. North West Group meetings take place in the Conference room at the Manchester Museum of Science and Industry, starting usually at 1730; tea is served from 1700.

Queries about London meetings should be addressed to Hamish Carmichael, and about Manchester meetings to William Gunn on 01663 764997 or at <william.gunn@ntlworld.com>.

Committee of the Society

Chairman **Dr Roger Johnson FBCS**, 9 Stanhope Way, Riverhead, Sevenoaks, Kent TN13 2DZ. Tel: 020 7631 6709. Email: r.johnson@bcs.org.uk
Vice-Chairman **Tony Sale Hon FBCS**, 15 Northampton Road, Bromham, Beds MK43 8QB. Tel: 01234 822788. Email: tsale@qufaro.demon.co.uk
Secretary **Hamish Carmichael FBCS**, 63 Collingwood Avenue, Tolworth, Surbiton, Surrey KT5 9PU. Tel: 020 8337 3176. Email: hamishc@globalnet.co.uk
Treasurer **Dan Hayton**, 31 The High Street, Farnborough Village, Orpington, Kent BR6 7BQ. Tel: 01689 852186. Email: Daniel@newcomen.demon.co.uk
Science Museum representative **Dr Xerxes Mazda**, Manager of Collections Access, Science Museum, Exhibition Road, London SW7 2DD. Tel: 020 7942 1100. Email: x.mazda@nmsi.ac.uk
Museum of Science & Industry in Manchester representative **Jenny Wetton**, Museum of Science & Industry, Liverpool Road, Castlefield, Manchester M3 4JP. Tel: 0161 832 2244. Email: j.wetton@msim.org.uk
National Archives representative **David Glover**, Digital Preservation Department, National Archives, Kew, Richmond, Surrey TW9 4DU. Tel: 020 8392 5330. Email: david.glover@nationalarchives.gov.uk
Computer Museum at Bletchley Park representative **Michelle Moore**, The Computer Museum, Bletchley Park, Bletchley MK3 6EB. Tel: 07748 981391.
Email: michellejmoore@hotmail.com
Chairman, Elliott 803 Working Party **John Sinclair**, 9 Plummers Lane, Haynes, Bedford MK45 3PL. Tel: 01234 381 403. Email: john@eurocom-solutions.co.uk
Chairman, Elliott 401 Working Party **Arthur Rowles**, 10 The Vineyard, Bouldnor, Yarmouth, Isle of Wight PO41 0XE Tel: 01983 761399.
Email: rowles01@globalnet.co.uk
Chairman, Pegasus Working Party **Len Hewitt MBCS**, 5 Birch Grove, Kingswood, Surrey KT20 6QU. Tel: 01737 832355. Email: leonard.hewitt@ntlworld.com
Chairman, DEC Working Party **Kevin Murrell**, 25 Comet Close, Ash Vale, Aldershot, Hants GU12 5SG. Tel: 01252 683503. Email: kevin@ps8.co.uk
Chairman, S100 bus Working Party **Robin Shirley**, 41 Guildford Park Avenue, Guildford, Surrey GU2 5NL. Tel: 01483 565220. Email: r.shirley@surrey.ac.uk
Chairman, Bombe Rebuild Project **John Harper CEng MIEE MBCS**, 7 Cedar Avenue, Ickleford, Hitchin, Herts SG5 3XU. Tel: 01462 451970.
Email: bombe@jharper.demon.co.uk
Chairman, Software Conservation Working Party **Dr Dave Holdsworth CEng Hon FBCS**, University Computing Service, University of Leeds, Leeds LS2 9JT. Email: ecldh@leeds.ac.uk
Digital Archivist & Chairman, Our Computer Heritage Working Party **Professor Simon Lavington FBCS FIEE CEng**, Lemon Tree Cottage, High Street, Sproughton, Suffolk IP8 3AH: Tel: 01473 748478. Email:lavis@essex.ac.uk

Committee of the Society (contd)

Chairman, North West Group **Tom Hinchliffe**, 44 Park Road, Disley, Cheshire SK12 2LX. Tel: 01663 765040. Email: tom.h@dial.pipex.com.
Editor, Resurrection **Nicholas Enticknap**, 4 Thornton Court, Grand Drive, Raynes Park SW20 9HJ. Tel: 020 8540 5952. Email: wk@nenticknap.fsnet.co.uk
Dr David Anderson, The Woozle, 1 Oatlands Road, Boorley Green, Hants SO32 2DE. Tel: 0239284 6668. Email: cdpa@btinternet.com
Peter Barnes FBCS, 10 The Broadway, Gustard Wood, Herts AL4 8LN. Tel: 01438 832906. Email: barnes@peterbarnes.freeserve.co.uk
Chris Burton CEng FIEE FBCS, Wern Ddu Fach, Llansilin, Oswestry, Shropshire SY10 9BN. Tel: 01691 791274. Email: cpb@envex.demon.co.uk
Dr Martin Campbell-Kelly, Department of Computer Science, University of Warwick, Coventry CV4 7AL. Tel: 01203 523196. Email: mck@dcs.warwick.ac.uk
George Davis CEng Hon FBCS, 4 Digby Place, Croydon CR0 5QR. Tel: 020 8681 7784. Email: georgedavis@bcs.org.uk
Harold Gearing Hon FBCS, 14 Craft Way, Steeple Morden, Royston, Herts SG8 0PF. Tel: 01763 852567.
Eric Jukes, 153 Kenilworth Crescent, Enfield, Middlesex EN1 3RG. Tel: 020 8366 6162.
Ernest Morris FBCS, 16 Copperkins Lane, Amersham, Bucks HP6 5QF. Tel: 01494 727600. Email: Ernest.Morris@btinternet.com
Dr Doron Swade CEng FBCS, 54 Park Road, Kingston-upon-Thames, Surrey KT2 6AU. Tel: 020 8392 0072. Email: doron.swade@blueyonder.co.uk

Point of Contact

Readers who have general queries to put to the Society should address them to the Secretary: contact details are given on the page opposite.

Members who move house should notify Hamish Carmichael of their new address to ensure that they continue to receive copies of *Resurrection*. Those who are also members of the BCS should note that the CCS membership is different from the BCS list and so needs to be maintained separately.

Resurrection is the bulletin of the Computer Conservation Society. Copies of the current issue are available from the Secretary for £5.00 each.

Editor -- Nicholas Enticknap Typesetting -- Nicholas Enticknap
Cover design -- Tony Sale
Printed by the British Computer Society

© Computer Conservation Society