My Life on Atlas 1 and 2

***Author**: Brian Chapman*

***Abstract**: Brian Chapman worked for Ferranti for six years from spring 1960, initially on Atlas 1, first in Newman Street and then at AERE Harwell. Later he worked on Atlas 2 at AWRE Aldermaston and on Titan at Cambridge.*

I joined Ferranti in spring 1960 specifically to work on Atlas, following earlier computing experience. While at Cambridge University I had developed an interest in computers and worked one vacation for Elliott Brothers on their 402. After graduating in 1958 I spent 2 years with de Havilland Propellers Ltd, writing programs for the Ferranti Pegasus In preparation for test firings of the Blue Streak rocket later to be the first stage of the European Space rocket. This work included trajectory calculations and the analysis of telemetry results, and I periodically used the Pegasus at Ferranti in Portland Place, where I learnt about the Atlas project.

Consequently I joined Ferranti Limited and was initially based in Newman Street where I gained knowledge of Atlas hardware and software. I was working alongside the Supervisor team under David Howarth. I was one of about 4 people working on other design aspects of Atlas, and there were also two sales persons. I was able to contribute to the project by coding several arithmetic extracodes. I was working alongside Stan Gill and Brian Hardisty and also knew many people in the Orion development team including Sandy Fraser and Conway Berners-Lee.

The Atomic Energy Research Establishment at Harwell had become a major user of computing power and was anxious to get the best of British for themselves and also to provide additional facilities for universities. When their negotiations with Ferranti began to mature in mid 1961 I was asked to undertake a sales support role at Harwell and to assist in the coding of their planned Fortran compiler. The compiler team was located within the AERE Theoretical Physics Division, which occupied offices on the southern side of Hangar 8 which was left over from previous use of the site as an RAF airfield, and contained a small reactor. I was the only Ferranti person on site for much of the time. I found that I was suddenly immersed in a punched card environment, which I quickly grew to like despite previously having only ever used paper tape.

The Fortran compiler project has been described in detail elsewhere, notably by Dr. Ian Pyle and Alan Curtis, and is notable for using Fortran in its implementation. The compiler was split into 3 passes, each being a chain in the Fortran sense. The first pass (Ian Pyle and Barbara Stokoe) performed syntax analysis. The second pass designed and coded by myself, generated Atlas code from an intermediate form produced during the first analysis pass. The third pass (Bart Fossey and Paul Bryant) produced relocatable binary output on punched cards in the BAS format as defined by Ian Pyle and Alan Curtis.

In my work I made extensive use of my knowledge of the Atlas instruction set that I had gained earlier, ensuring that the object code generated was as efficient as possible. The transformation of numerical expressions from a reverse Polish notation required careful thought. I also had to consider the effects of the Atlas octal floating point numbers, which were, and probably still are, unique. This also affected the choice of rounding operations. An efficient structure for subroutine calls also had to be determined and encoded. Each subroutine saved as many B-lines as it used so that its use was invisible to calling routines. The number required was closely related to the depth of Fortran DO loop nests.

An important early design decision for the Atlas Fortran compiler had been that it should itself be written in Fortran. This meant that AERE staff on the project, numbering about 7 covering all aspects of the compiler and supporting software, would be coding in a language they knew and understood. All design work as it progressed was coded in Fortran, which I had to quickly master. This also meant that clean compilation and debugging of the compiler could take place well before Atlas itself was available. Compilation and testing began using Fortran on an IBM 7090 at Aldermaston, but when that moved to AERE Risley our work transferred there.

Testing in those days was very different to today – there was no on-line access. The compiler team made use of existing AERE processes which included daily flights between Bournemouth (for Winfrith), Abingdon (for Harwell) and Manchester (for Risley). Each day a stack of Fortran source cards for a selection of subroutines would be taken to an IBM 1401 in a nearby hut (which also housed a Ferranti Mercury) and transformed to magnetic tape. The tape or tapes with all Harwell work would be taken to Risley the next day, processed, and result tapes flown back that night. The following day the 1401 printed out the results and also punched cards for successfully compiled subprograms. This output would be available to us mid morning, when punching, coding or other errors could be corrected and jobs resubmitted. There was thus a two day turnaround on all work. Generally we each worked on at least two areas simultaneously so that something was achieved every day. As deadlines approached we would often fly to Risley for a day and maybe get 3 extra tests done.

Finally when all the code had been written and individual subroutines compiled without errors, we had a complete set of 7090 BSS binary cards which enabled the complete Atlas compiler to be run on the 7090. After initial testing on the 7090 was completed satisfactorily the compiler was then gradually given itself to chew over and the final total output comprised a set of BAS binary cards containing Atlas code. These cards, about 3000 in all, were complemented with a few core machine code subprograms.

It is a Fortran tradition that very large complex applications are made manageable by splitting every program into independently compiled subroutines, which are only linked together immediately prior to execution. Since the standard Atlas operating environment had no such provision this required the writing of a complete run-time environment for Fortran programs. This was called Hartran (written by Bart Fossey) and comprised calls of the Fortran compiler, the ASP symbolic assembler (Bob McLatchie) and the BAS link loader (Arthur Trice from Ferranti). ASP was used for the relatively few but necessary run-time library routines (mainly input/output). The BAS loader linked all compiled or assembled subroutines in BAS and defined the storage environment for the actual execution. This included mapping large Fortran arrays into the very flexible Atlas addressing system dynamically at load time. The Atlas Supervisor job control language required the instruction COMPILER HARTRAN which led to some nomenclature issues and confusion as Hartran was really a subsystem and not just a compiler.

After all the early testing on the IBM 7090 with its grand finale of compiling itself we had had a compiler which was proven only in its own logic, which although very extensive in size was limited other ways. No output had ever been run on Atlas. Also the Hartran system and BAS loader were untested, and that had to be addressed first. When the compiler was finally running on Atlas, the full scope of the Fortran language still required proving. This involved significant work as the compiler itself used mostly integer arithmetic and more normal Fortran application code involving floating point variables was completely untested.

I took the complete system to Manchester as soon as the first production Atlas became available for systems programmers during 1964. The first thing I had to do was to check out the card reader and punch, neither of which had been used by the University. Reading a deck of 3000 cards was far more demanding than typical engineer tests! All testing on Atlas at the time was by dedicated individual time slots. After some single day visits I spent about 3 weeks flying up on a Monday morning, working through that night when Atlas time was readily available, flying back the next day, the back again Thursday morning, working that night and returning home Friday afternoon. As progress was slower than hoped, very late in 1964 I spent 6 weeks in

Manchester to expedite progress. My testing culminated with the installation of the complete Hartran with Fortran system on the Manchester Atlas.

Because of the requirement to service universities as well as AERE the Atlas Computer Laboratory was built at Chilton to house the Harwell Atlas, adjacent to but outside the AERE Harwell security fence to facilitate open access. The director was Jack Howlett. When Atlas was physically installed in April 1964 almost all AERE staff on the project and myself relocated to the new building. When testing of the hardware and Supervisor installation was complete, the full Hartran and Fortran system was loaded and checked out, which did not take long after the previous work in Manchester. A general computing service began in autumn 1964.

Once the complete Hartran and Fortran system was fully proven at the Atlas Computer Laboratory I delivered it to the London University site, where they installed it on their Atlas.

By now AWRE Aldermaston had negotiated the purchase of an Atlas 2. I transferred there late in 1964 to provide Atlas 2 sales support and with the express aim of installing the Hartran environment, into which AWRE would insert their own Fortran compiler (S3) with identical source language to that used on their existing IBM 7030 Stretch computer (S2).

My final involvement with Atlas 2 was taking the Aldermaston Hartran system, including the S3 Fortran compiler, and installing it on the Cambridge Titan. Later Hartran would also be used there in conjunction with a Fortran compiler (T3) written by John Larmouth. The second Cambridge Atlas 2 for the Computer Aided Design Centre was installed after my time and I do not know which Fortran compliers were installed.

Looking back, was the use of Fortran a success? Yes in the sense that a useful system was produced which was used for many years. Was it efficient? Not really, as Paul Bryant showed when he recoded the ASP assembler in ASP instead of the original Fortran. Also the use of Fortran with its numerical bias prevented the code generation in our compiler from using of any of table searching and similar extracodes used effectively in other compilers. Should another language have been used? There wasn't the current choice of high level systems programming languages and the Compiler-compiler was rejected because of its demand that only a complete program could be compiled which was alien to Harwell's exceedingly large Fortran applications.

I left ICL, which by then had absorbed the Ferranti Computer Department, at the end of 1966.

Brian Chapman 11th August 2014

References

Curtis, A R & Pyle I C: “A proposed target language for compilers on Atlas”, Computer Journal vol 5(2) pp 100-106 (1962).
http://www.chilton-computing.org.uk/acl/applications/hartran/p002.htm

Fossey E B & Stokoe B: “Fortran on Atlas: The Hartran System” Atlas Computer Laboratory (1966),
http://www.chilton-computing.org.uk/acl/applications/hartran/p001.htm

Pyle I C: “Memories of Atlas Fortran” (Nov 2012)
http://curation.cs.manchester.ac.uk/atlas/docs/Atlas-Fortran-finalbb.pdf