Considerations about the migration to Alpha of the SAX Local Centres and SDC/SAC.

L.Chiappetti -IFCTR
Issue 1.0 - Wed, 27 Oct 1993
DAWG-REP.9.0/93

1.   Introduction

I have written this short note as an aid for the planning of both the possible changes in the ASI-procured hardware at the Italian SAX Local Centres (nearly immediate replacement of the VAX8250 with an Alpha running OpenVMS/AXP, future later replacement of the MIPS DECstations running Ultrix with Alpha stations running OSF/1) as well as the definition of the hardware for the Scientific Data Centre SAC component.

I will keep this note separated from my personal conclusions on the topic, which will be presented at the DAWG meeting on November 3-4, or circulated via e-mail, possibly in restricted form to the involved parties only (ie. Local Centre responsibles).

This note is articulated as follows : section 2 lists the Digital documents examined , section 3 discusses the technical implications for the XAS software (this section should be of interest for the entire DAWG), section 4 lists all possible problems, and assigns them a "severity" grade.

As a short conclusion, the migration from VAX to AXP VMS seems quite light, users won't notice it, and system managers will have some moderately heavy work. The main problems may be those of migrating to a 64-bit Alpha architecture (common to the OSF migration and discussed in section 3), the absence of Decnet Phase V (required for Italy in 1994), and the support of external software for which there is no source.

The migration from Ultrix to OSF is relatively light, there are no major changes, but a large number of annoying little changes which may affect both users and particularly system managers. The major problems are again those concerned with the migration to a 64-bit Alpha architecture (common to the VMS/AXP migration and discussed in section 3), and the support of external software (which in Unix is particularly numerous), due both to the change to ANSI C and possibly to the lack of sources.

The author gratefully acknowledges a preview reading by and some inputs from D.Dal Fiume, M.Trifoglio and M.C.Maccarone.

2.   Reference Documents

These DEC documents, in the form made available by M.Trifoglio, were used as starting references.

[1]   Ultrix to DEC OSF/1 Migration Guide (AA-PS3EA-TE, March 1993)
[2]   Migrating to an Open VMS AXP System : Planning for Migration
         (AA-PV62A-TE, May 1993)
[3]   A comparison of System Management on Open VMS AXP and Open VMS VAX (AA-PV71A-TE, May 1993)
[4]   Migrating to an Open VMS AXP System : Recompiling and Relinking Applications (AA-PV63A-TE, May 1993)

3.   Impacts on XAS

The impact of a migration to an Alpha system (both with OSF/1 or OpenVMS/AXP) for programmers are due to the fact that Alphas use 64-bit addressing. This involves :

- Filesystems and individual disk files may be very large (in excess of 2 Gbyte). This may be an advantage in some situations, but is actually totally irrelevant to us.

- The system memory page size is larger (this might be more relevant on VMS). This however is likely to be irrelevant to us. At least no XAS routines make explicit use of this information. I ignore whether this has impacts on memory allocation (i.e. C routine `calloc`) but I presume this to be unlikely.

♠ **Addresses in programs are to be accessed as 64-bit entities.** In particular in C <u>pointers are 64-bit</u> entities. Any C program doing pointer arithmetics shall take note of this, and not equate pointers with smaller entities. There is a <u>problem in the definition of `int` and `long`</u> (which were the same under all previous 32-bit architectures; now OSF/1 C defines an `int` as 32 bits and a `long` as 64 bits, VMS/AXP defines both `int` and `long` as 32 bits : **this migh be an annoying difference!**) The only XAS routines which may be affected are the VOS routines `Z_ALLOC` and `Z_DEALLOC` (actually, their C back ends `zc_alloc` and `zc_free`). **However these are widely used**, in particular are essential in all XAS file (header) handling routines. Most likely the change will affect the calling sequence of (I presume that the address argument must be declared as `INTEGER IADDR(2)` or `REAL*8 IADDR`, ie. a 64-bit entity) and therefore **need to be propagated in all higher level routines** (including the definition of HCOMMON) and main programs. It has to be noted that a known bug exists in `zc_free` (see `XAS10.BUGLIST` entry #24). It has to be noted also that Fortran 90 will provide a standard mechanism for dynamic memory allocation, so any redesign of the two VOS routines should possibly be such to be forward-compatible with f90 with a minimum of changes.
This change is well confined to a small part of the code, but unfortunately propagates to all XAS. Consultancy of an expert C programmer might be appreciated.

♠ As a consequence of the 64-bit architecture there are two additional problems. One concerns the fact that **operations on byte and 16-bit integers are no longer "atomic"** (roughly speaking, the system does not handle them in a single machine instruction, but moves a 64-bit quadword and masks it). It is unclear whether longword (32-bit) operations suffer a similar penalty.
The second problem concerns **data alignment** (8-, 16- and (?) 32-bit items which are not aligned on 64-bit boundaries may be handled, but in a much slower way).
It has to be noted that similar problems are possibly common on other RISC architectures (e.g. Ultrix MIPS, Sun SPARC), with the only difference that the "natural unit" is 32 and not 64 bits.
XAS makes wide use of `EQUIVALENCE` statements and `COMMON` blocks, and handles some 16-bit integers (in file input/output, though sparingly, ie. only when required, like in telemetry data). Byte quantities are always handled via `EQUIVALENCE` to `CHARACTER`, which should be a safe mechanism. I presume XAS programs are sound (e.g. `COMMON` blocks are constructed in an orderly manner; also whenever the DEC and Sun compilers gave alignment warnings, I've generally tried to fix the code), but I am **not able to exclude** *a priori* **that some adverse effects are possible.**

♠ Another important point concerns the system's native **internal representation of floating point values (with particular regard to double precision).** This is of concern <u>for VMS/AXP only</u>, since OSF/1 on Alpha will use exactly the same (IEEE with little endian order) floating point representation used by Ultrix. On VMS/AXP instead one has different choices for VMS/VAX compatibility. The following table summarizes the various possibilities available, and **boldface** indicates the default combination.

[3]	C.6	2	the default editor, also for mail, is EVE not EDT (for me is fine, but most users will like EDT, way to go back exists)	U			
[3]	C.8	?	Do not understand point on C Run Time Library [to be verified]		P		
[4]	2.1	0	differences in memory allocation (do not affect calloc, ie. us)		P		
[4]	A.2.1	3	DEC C is ANSI C, impact on external programs (see OSF/1 above)		P		E
[4]	A.4.1.1	1	Fortran logical units no longer limited to 0-99 (but is better for us to continue reserving only those numbers as in DAWG-APD.6.91]		P		
[4]	A.4.1.4	1	SAVEd dummy arrays (Fortran) do not work (do we use any ?)		P		
[4]	A.4.2.2	2	Fortran optimization options are different from Vax		P		
[4]	A.4.4 +rumours	1	there exist routines to convert between different floating point representations (may be used in VOS) Also it should always be possible to use OPEN with CONVERT option to read foreign file (this is already on Ultrix)		P		

4.3. External package compatibility

I enquired about the compatibility of Alpha platforms with popular packages like IRAF and MIDAS (as well as on the support status of Ultrix). At the moment MIDAS is not supported on OSF/1, and IRAF is not supported on either operating systems on Alpha. Both teams are planning to work on it and it is likely that the quoted systems will be supported at the end of 1994. Of course the progress of the work depends on the availability of such machines either in house or in selected test sites.

Concerning Ultrix support, both IRAF and MIDAS intend to continue support to it, with IRAF definitely committed to support what they consider "a popular platform", and MIDAS also committed, but depending on finding test sites.

I have also enquired with Bill Pence about FITSIO, his reply is reported in section 3 above. In addition I have retrieved the VAX and AXP dependent parts only of FITSIO 3.405 and made a quick check. They in fact support conversion from/to F+G format to/from IEEE via byteswap and scaling by 4. Change to direct support of IEEE should be very simple : part of the code in 4 system-dependent routines has to be commented out. This however means creating a new version of the library under our own support.

I have no information on other packages, since we do not use them here at IFCTR.