CONFIGURING THE COMPUTER SYSTEM.

WHAT TO GET.

In the early days of computing, it was not unknown for companies to invest large sums of money in highly fashionable computers (not infrequently on the advice of unscrupulous agents), only to find that they were no more efficient than the manual systems they were intended to replace, and cost a great deal more to run – to the extent that some firms went bankrupt. Of course, we know much better now, don't we ? No – though the phenomenon has changed in scale. Lots of schools, apparently believing that computers were just as magical for education as they were presumably supposed to be for payroll, bought quite unsuitable computers (not infrequently on the advice of unscrupulous agents), and found that they were – oddly enough – quite unsuitable.

The moral is that it's no good buying a computer system without knowing reasonably precisely what you intend to do with it, and making sure that the system you acquire will do the job. For a solitary personal machine, that isn't too hard : decide what software you want to run, and get a machine which will run it, and it's nobody's fault but yours if you didn't foresee the various extensions, updates, new versions, and so on which immediately render it obsolete. But if you are trying to establish a computer service for a large organisation, your task is a lot harder.

The most pressing need in designing or extending a computer service is for information. Here are some of the things you need to know :

- What the system will do. This will tell you something about the software you will require, and – if there are any special requirements – might help to define the hardware too.
- How much the system will do. This gives you some idea of how much of everything you will need.
- Who is going to use the system. You will have to provide appropriate access for all of them.
- How they are going to use it. Do they all need access to a central database ? Is the work mainly word processing, or engineering design, or learning to write Pascal programmes ?
- How they are grouped together. If they can work independently, separate microcomputers might be advantageous; if not, you might need to install a network to link the microcomputers, or perhaps a mainframe with terminals would be better.
- Where they are. This will give you a basis on which to work out the communications requirements.

- and so on. It is not our purpose to tell you how to go about configuring your computer system, but to underline the large amount of diverse information you need before you can begin.

The sheer bulk of the information which must be handled to configure a system makes configuration an exceedingly difficult task. Even if you can decide what sort of work you want your system to do, you must then work out what sort of hardware is needed to perform the work at a satisfactory speed, then you must work out where to put the hardware, and what sort of boxes it will inhabit, and what sort of cables you need to connect the boxes together, and all manner of fiddly detail which must be right or the system won't work. In practice, you might determine the main requirements and leave the rest to the company which supplies the equipment, but it must all be done somehow.

Digital Equipment Corporation (DEC) achieved a sort of fame by developing a very elaborate expert system to help them in the system configuration activities. It began as a research project called R1, but was later inscrutably renamed XCON[MAN1]. It cost them many millions of dollars, and evolved over many years – but DEC asserted that it saved them around US$40 000 000 per annum in costs incurred by getting things wrong. (We have memories of computer installations – not DEC – being delayed for weeks because certain cables had not been ordered, could not be found, or had to be flown in from Brazil.) To illustrate the complexity and mind-numbing triviality of the decisions which must be made, here is a single rule from XCON. It has something to do with deciding whether a particular disc can be put into a cabinet. (The rule as presented here has been translated into an approximation to English; the real one is written in typical computer gibberish, using a language developed from the once-popular language for rule-based systems, OPS5.)

Rule Name: R1a-unmounted-ubx-options

	IF –	*Comments*
C1	The current step in the configuration process involves mounting options in containers;	*This condition is used to distinguish the group of rules that can potentially activate.*
C2	and the system being configured is not a vax11/780, vax11/782, vax11/785, vax8650, or vax8600;	*This rule is not applicable to those types of hardware systems.*
C3	and there is no unconfigured disk which sits on the idc bus;	*This condition insures that the rule will not activate before all disks on an idc bus have been configured ("Disk" actually means "disk drive".)*
C4	and there is an unconfigured rl02-type disk which needs to mount inside a cabinet and whose pre-assigned controller sits on a unibus and it is the first disk assigned;	*This identifies the properties required of an appropriate disk to be configured by this rule.*
C5	and there is no unconfigured rl02-type disk assigned to a controller that is placed closer to the cpu than the controller assigned to the aforementioned disk;	*Disks assigned to controllers that are closest to the cpu need to be placed first, in case there is insufficient capacity for all of them.*
C6	and there is a requirement to cable the disk to be configured to a controller;	*Part of the activity of this rule is to determine that cabling. There may be several possibilities.*
C7	and there has been no connection made between the disk to be configured and anything else;	*This indicates that some of the activities that this rule will perform have not yet occurred.*
C8	and there is a controller to which the disk to be configured has been pre-assigned and which sits on a unibus;	*This identifies the appropriate controller.*
C9	and there is a requirement to cable the controller to a disk whose type and quantity of cable match one of the possibilities specified for the disk;	*This identifies the type and quantity of cable needed for this particular disk/controller combination.*
C10	and there has been no connection created yet to this controller from any disk;	*Another indication that the activities to be performed by this rule have not occurred.*

C11	and there are no unused disk spaces in any unibus cabinet;	*This insures that any spaces appropriate for disks in this type of cabinet will be filled before the rule can activate.*
C12	and there is a description for the capacity of a disk cabinet, whose name is not “h9643”;	*This identifies a special type of cabinet that can only contain disks. An "h9643" is one variation of a cabinet to which the rule does not apply.*
C13	and there is an unconfigured disk cabinet;	*This identifies an appropriate cabinet in which the disk to be configured will be placed by the rule.*
C14	and the top space available for disk placement is unused;	*This identifies a location in the aforementioned cabinet where the disk to be configured can be placed. It needs to be on the top because of the removable medium.*
	THEN	
A1	mark the disk configured;	
A2	and update the top space in the cabinet to be used;	*The location needs to be marked so that nothing else will be placed there.*
A3	and create a connection relationship between the disk and its controller, fully specifying the identifying information for the disk, controller, cabinet, and the type and quantity of the cable to be used for the connection;	*This establishes the connection between the disk and its controller. Other rules will determine the length and choose the exact cable(s).*
A4	and create a containing relationship between the disk and the cabinet, specifying the identifying information for the disk and cabinet as well as the location of the placement;	*This establishes the placement of the disk in the cabinet.*
A5	and create labels for the output diagram showing the disk within the cabinet for both the skyline view of the cabinet layout and the detailed view of the particular cabinet.	*This insures that the output diagram will display this information correctly.*

By September, 1988, XCON had grown to contain 10129 rules, which had led to further research work on managing large expert systems.

In this course, we are – fortunately – not much concerned with putting discs in cabinets, but we are certainly concerned with many of the other decision which have to be taken at some point in the system configuration. Here are some examples; most of them refer to topics we have already discussed at more or less length, and some remain to be dealt with. They are in no particular order, and obviously interact strongly, but they give some idea of the sorts of information you need if you're planning a large system.

- **How much memory ?** That depends on what sort of work we're doing now, and what we expect to be doing in a couple of years. How many people will be using the system ? How much of their memory requirements can be satisfied by virtual memory ?

- **How many processors ?** How many operations per second will we need at peak times ? Do we explicitly require parallel processing, or could a single processor manage ?

- **What sorts of processor ?** Do we want a 64-bit address space ? Do we want hardware segmentation, or protection ? Do we require microprogramming ? Will we need direct memory access ?

- **What devices ?** What are present and projected needs for input and output ? Do we need special devices ? How much disc (or other storage) space do we need ? What about backup ?

- **What organisational requirements ?** Are there any special characteristics of the organisation which might require special facilities ? Is special security an issue ? Do priorities and scheduling reflect the organisation structure ? Do some parts of the organisation require special provision for storage, archiving, etc. ?

- **What networking structure ?** What volume of network traffic do we expect ? What are the peak demands ?

- **What provision for accounting ?** Who is going to pay for the systems ? How is usage to be measured ? What sorts of usage are to be measured ?

- **....... and much, much more,** as the advertisements say.

Clearly, an operating system can't do a lot to provide all this information which is needed before it is even bought; but it can perhaps be constructed to provide the information you'll need next time. Quite generally, any information about system workload and performance can be valuable. It can tell you what your present system is doing, and figures from a comparable installation elsewhere can tell you something about the behaviour you might expect from a proposed new system.

Large shared systems have been keeping more or less of this information for a long time now, but an interesting consequence of the widespread shift to microcomputers in recent years has been that the information has been lost : microcomputer operating systems just don't keep the same sorts of record. Now that a microcomputer's disc space and processor power is more than adequate for most people's needs, there's no particular reason why they shouldn't, but they don't. Without hard information, managers might be reduced to relying on their employees' reports to gauge the need for new or extended machinery. Does "I can't run this new version of the software without the latest processor" really mean just that ? – or does it mean "the new software has been set up to grab 16 megabytes of memory when it really only needs 1 megabyte" ? – or does it mean "the video game that my children use at the weekend doesn't run fast enough on this old processor" ?

This is just one of several problems which attend the use of computers primarily designed as children's toys for serious industrial and commercial work. (The other spectacular failure is in security.) We shall not discuss this any further (because there's nothing there to discuss), but bear in mind that not all the data used in planning is as hard and reliable as we'd like it to be.

There are several stages in implementing a decision about the system configuration. It is obviously necessary to acquire any equipment you need and don't have, and to have it installed and tested. So far as this course is concerned, these matters are of only limited concern, and we hope that someone else will do it for us. After that, we become interested in at least two ways : we have to ensure that matters within the computer itself proceed as required, and we have to ensure that the computer can communicate effectively with its devices. Internally, the task is largely a matter of setting up all the sorts of table we have discussed in the course, and getting the right initial information therein. Externally, the devices must be driven by some sort of software, which must be integrated with the system; so we need appropriate interrupt handlers and device drivers, and we must be able to introduce references to the device software into the device table.

REFERENCES.

MAN1 : V.E. Barker, D.E. O'Connor : "Expert systems for configuration at Digital : Xcon and beyond", *Comm.ACM* **32**, 298 (1989).

QUESTIONS.

Consider how the list of "things you need to know" fits in with the list of "decisions which have to be taken". Which items of the first list are needed to determine items of the second list ? What other items should be included in the first list to answer the questions raised by the second list ? What other items should be included in the second list ?

What sort of usage records would you like to keep on a microcomputer in an ordinary commercial environment ? How could you do it ?
