ISSN 0961-9348


The British Computer Society

Copy to Chris Hook
Bharat Thakker

# JOURNAL

SPRING 1991

Volume 2, Number 1

## MEMBERS' MEETINGS FOR 1990/91

| | | | | |
|---|---|---|---|---|
| 9 Apr 1991 | 4.00pm for 4.30pm | **IBM's DB2 RELATIONAL DATABASE** | An IBM speaker | KPMG Training Centre |
| 15 May 1991 | 9.00am | **ANNUAL CONFERENCE**<br>**Building Successful Business Systems** | | London International Press Centre |
| | | **User and Audit Responsibilities** | Joe Hegarty<br>*Surrey County Council* | |
| | | **Systems Controls** | Peter Morriss<br>*KPMG Peat Marwick McLintock* | |
| | | **Project Management and Control** | Stuart Senior<br>*Coopers Lybrand Deloitte* | |
| | | **Going Live and After Implementation** | Chris Carr<br>*National Audit Office* | |
| 15 May 1991 | 4.30pm | **ANNUAL GENERAL MEETING**<br>*Admission free to members* | | International London Press Centre |

Meetings are free to members, with the exception of the Discussion Groups, the joint meeting with the IIA and the Annual Conference, for which charges are made.

The meeting scheduled for the 12th February 1991 was cancelled due to the inclement weather and bad travel conditions. The subject was "Auditing the MVS OS". It is now expected that this meeting will be re-arranged and included in the 1991/92 programme.

EDITORIAL PANEL

Deborah Ashton
*British Airways*
*081 562 3663*

John Bevan
*Consultant*
*0992 582439*

Virginia Bryant
*City University*
*071 253 4399*

Malcolm Lindsey
*Argos Distributors Ltd*
*0908 690333*

Rob Melville
*City University*
*071 920 0111*

John Nye
*British Aerospace*
*0707 262345*

Bryan Roche
*Inland Revenue*
*0952 294521*

Fred Thomas
*0371 875457*

Philip Weights

Brian Wallis
*City of Westminster*
*071 798 2320*

# EDITORIAL

## "SAME BROOM, DIFFERENT HANDLE..."

**It is always a very pleasant feeling when starting a new venture to know that the difficult tasks have already been done. It is even better when a position which usually requires exhortations for articles to be produced, actually inherited carries several high quality pieces which provide a breathing space at least for this issue! This fortunate position has only a limited shelf life though, and over the next quarter the membership will be prodded for contributions as much as before; perhaps not with Ginny ' s charmingly effective style, but it is hoped with equally good results.**

**The journal has developed into a very useful publication in its short life, and I do not intend to make any radical changes to its character. Rather, I would like to maintain its niche as a specialist journal, produced by and for the benefit of its membership. This does not imply that every article submitted for publication will be accepted intact, nor that a team of referees will scrutinise submissions anonymously. But to continue as a journal which provides for the needs of a very specialised and knowledgeable community, we need to establish a balance between the quality of the technical content and readability. So the editorial team welcomes contributions, but retains the right to the blue pencil. Please accompany contributions with a brief biography and a photograph which does not come from a booth!**

**ROB MELVILLE**

# CHAIRMAN'S CORNER

JOHN MITCHELL

First, an apology. We had to cancel the February meeting (Auditing the MVS Operating System) at very short notice due to the adverse weather conditions (the wrong sort of snow according to British Rail) and the further problem that our speaker was in Edinburgh. I hope that none of you had a wasted journey, as we had left word at the venue and assumed that even potential Scotts of the Artic would check before setting forth into the blizzard. We are trying to re-schedule the talk for early in our next season, so make a provisional note in your diaries for the first Tuesday in October.

Some of you may have noticed that the BCS has recently given the security procedures for the 1991 national census a clean bill of health. You may also be a bit miffed at not being asked to take part, but the blame for you not being selected for this honour does not rest at the door of your committee. I wrote to the BCS some two years ago, drawing their attention to the wealth of skills and experience residing in our membership and received an acknowledgement (not of the skills, but of the letter!) from them.

Now comes the bit that could be straight from Yes Minister. You see, I never heard anything else from them, so when I saw the report in **Computing** that the "audit" had already taken place I contacted the BCS and was told that they couldn't tell me who had actually done the work as it was covered by the Official Secrets Act! I was also told that some members of the group may well have been involved, but even if they were, they wouldn't be able to tell me about it without fear of ending up in the Tower. I then asked how the team had been selected and was told that a top level committee within the BCS had selected people with appropriate skills. Oh well, it's not necessarily what you know, I suppose, but who knows you!

Well enough about that. Let me now digress to Windows 3, over 2.5 million copies of which have been snapped up by eager people anxious to break the DOS 640k barrier and multi-task on their 386 based hardware. I was one of those. Well it may be the best thing since sliced bread to some people, but both I and several other auditors that I have discussed the matter with, have found that the Graphical User Interface (GUI) is not as intuitive as one would expect, the file manager (the thing that allows you to see what files you have on your disks) is just about the worst thing that I have ever dealt with, and the multi-tasking ability of non-windows application is a variable event. That aside, I can see the future here. All those kids playing their arcade games on their Amiga machines will be right at home when they move into gainful employment and find Windows 3 sitting on their desks!

Now a plug for both our annual conference and our AGM. This year they are being held on the 15th May at the London Press Centre. The conference is a chargeable event and well worth the money, but the AGM is even better value as it's free! So why not come along to both? As part of a cunning plan we have scheduled the AGM to follow the conference, in the hope that those of you who attend the conference will stay on for a few minutes for the AGM. If for some strange reason you are not coming to the conference you can still turn up for the AGM; official notice of which you will find elsewhere in this journal. The AGM is **your** chance, to have **your** say, to **your** committee, so please come along and support it.

Finally, once again your dynamic committee has had a few changes. Rob Melville from City University has become Editor of the journal, Alison Webb has volunteered to help with our monthly programme and John Pringle (from the Department of Energy) will assist in organising our 1992 conference. Who ever said auditors never plan ahead? A big welcome to all of them.

THE

# ANNUAL GENERAL MEETING

OF THE

# COMPUTER AUDIT SPECIALIST GROUP

OF

# THE BRITISH COMPUTER SOCIETY

Will be held on **WEDNESDAY 15TH MAY 1991**

at the **INTERNATIONAL LONDON PRESS CENTRE** 76 Shoe Lane, London,EC4A 3JB (entrance in Printer Street) at **4.30 pm**

AGENDA

1. **Approval of the minutes of the AGM held on 15th May 1990**
2. **Chairman's Report**
3. **Treasurer's Report**
4. **Election of Officers**
5. **Election of Committee**
6. **Election of Auditor**
7. **Plans for 1991/1992**
8. **Any Other Business**

**The meeting will follow the close of the Annual Conference. There is no charge for attendance at the AGM which is open to all CASG members irrespective of whether they attend the conference.**

# NOMINATIONS FOR THE MANAGEMENT COMMITTEE

As is usual at this time I am asking for nominations for the Group's Management Committee.

We tend to hold about 10 committee meetings a year at a London location. The meetings start at 5.00pm and we try to finish them by 7.00pm. Each committee member is usually allocated a specific task, but we have no hard and fast rules on the matter and if you wish to join the commitee 'without portfolio' as it were, just to get a feel of what goes on, then please do so.

The committee is definitely not 'cliquey' and we genuinely welcome new people, new ideas and lots of enthusiasm.

If you fancy one of the posts that is already 'filled' (including mine) just put yourself forward and the AGM can vote on it. No one on the committee will be put out by such a display of interest ! A blank nomination form is printed below.

If you would like to discuss any of the posts on the committee then please contact either myself (0707 54040), Ragu Iyer (071 236 8000), or any committee member that you already know (their telephone numbers are given on the inside back page of this Journal.

Remember this is your Group and you should use this opportunity to have your say.

**John Mitchell**

---

# NOMINATION FORM

**FOR THE MANAGEMENT COMMITTEE**

**OF THE COMPUTER AUDIT SPECIALIST GROUP OF**

**THE BRITISH COMPUTER SOCIETY**

**Position:**

**Nominee:**

**Proposer:**

**Seconder:**

**Signature of Nominee agreeing to serve on the Committee .....**

**Dae.....**

# COMPUTER AUDIT SPECIALIST GROUP PUBLICATIONS

During the past year, your Executive Conmittee has - reviewed its book list. and elimiated those items which have become out of date and no longer applicable. The current publications are available on sale to all, but C.A.S.G. Members can obtain them at specially discounted prices, as indicated below.

There are also a few remaining copies of past Conference Papers (1984-86), which are still rele- vant and will be made available to C.A.S.G. members for as long as stocks last.

The publications available directly from the C.A.S.G fall into three categories, and can be ordered using the order form enclosed with this journal.

The first category is a series of four books in the "**BUYING SOFTWARE**" Series, which are designed as a general help to all persons evaluating, buying or auditing software packages. These books highlight the accounting and control features to look for and are written hy specialists (including contributions from H.M.Customs and Excise) for the benefit of those who need to buy, use, specify, design, program or audit that type of software. They are written to help get the requirements right first time and to avoid costly mistakes. Each book has a full section on the required controls. In each the Introduction outlines present problems with packaged software, including the lack of communication between designer and buyer, and explains how to use the guide to deal with such problems. Also, relevant terms are defined in a glossary. These books are intended primarily for users of mini- and micro-computer systems, but will also interest users of larger main- frame systems. The four books in the series are:

* **BUYING STOCK CONTROL SOFTWARE** (43pp.) which covers system requirements such as Stock Coding and quantifying, stock locations and product grouping, Stock Valuations, Stock Level Optimisation, Bill of Materials Processing and Product Replacement, Data Input and Filing, and required output.

* **BUYING PAYROLL SOFTWARE** (39 pp.) covering positive and negative payroll systems, types of payroll calculations, statutory and other deductions, types of output for payments, analyses, costing, tax returns, treatment of holidays, turnround documents, B.A.C.S., Data requirements, records to be maintained.

* **BUYING FINANCIAL ACCOUNTING SOFTWARE** (39pp.) covering the types of financial data to be handled, formats for holding data, data access, Nominal Ledgers, Journal Entries, Legal Require- ments, Reports, Management Facilities and a Checklist.

* **BUYING SOFTWARE - SALES ORDER PROCESSING** (29 pp.) dealing with tracing records through from order or sales invoice or delivery, identifying transactions and customers, provision of adequate information for all purposes, pricing, meeting delivery dates and requirements and linking into other accounting systems, and providing an adequate range of management facilities.

Each book in this category is available for £4.50 plus 50p postage each (The price to C.A.S.G. members is £3.50 plus 5Op postage each).

The second category is the book entitled **CONTROL AND AUDIT Of MINICOMPUTER SYSTEMS** (1983 48 pp. plus Appendices), which was written from contributions by Working Parties of the C.A.S.G, and contains much practical advice which is still relevant, concerning key points of control and some Customs and Excise requirements. This book costs £11.95 plus 5Op postage (The price to C.A.S.G. members is £8.95 plus 5Op postage)

The third category consists of three past Conference Report sets relating to conferences run by the Auditing by Computer Specialist Group (the earlier title of this Group) during 1984-86. These contain much which is still relevant. and are being made available to members for as long as stocks last. These are:

* **1984 - COMPUTER FRAUD - ADDRESSING THE REALITIES**

* **1985 - EFT/POS - TODAY'S CHALLENGE**

* **1986 - MANAGEMENT AND CONTROL OF ACCESS SYSTEMS**

There will be no charge for these to members of the C.A.S.G. , but 5Op each will be charged for postage.

Recently the C.A.S.G. has co-operated with two other BCS Specialist Groups to produce a new version of **CONTROL AND AUDIT OF DATABASE SYSTEMS.** This is now being printed and should be available in the Autumn, but will not be sold as part of the Group Publications Service. More information on this book and its purchase will be made available in this Journal nearer the date of publication.

# SOUND BYTES

*This is a new column, where sound practical advice on computer audit matters can be raised. We have launched it with Brian Wallis (Westminster Council) who writes on PCs. Future contributions could be on any practical topic.*

## HOW MUCH IS THAT MICRO IN THE WINDOW?

The control over the installation of micro-computers is a subject which requires some thought and an audit view on the overall policy operating within the company should be developed. There is a need to review the procedures which govern purchase of both hardware and software for user departments.

Problems may arise when the main thrust of energy for IT purchases is directed at installing the machines and having them 'up and running' in the shortest possible time.

The area which can be overlooked by relevant personnel is that of purchase and installation of the related operational software. For example, DOS 'start-up' software and any wordprocessing packages purchased as part of the original user requirement.

Therefore your attention is drawn to reviewing the internal procedures which ensure that all software received is indeed a valid version of the package and that relevant documentation has been received, bearing in mind the costs of system manuals and software.

When the number of micro - computers currently installed within your organization is considered, it soon becomes aparent that expenditure can reach large figures. Confirmation that monies paid to your computer supplier have been accounted for is an essential part of IT management.

So now you feel comfortable that all your equipment has been properly accounted for, have you checked your maintenance contract for support of the computers?

If not, you could find yourself in difficulty when trying to replace a particular piece of equipment should an operational problem develop, or a later version of software be required.

These are some areas of concern regarding the overall purchase of IT equipment which can be quickly addressed, if time and energy is afforded to raise the issues with management.

TIME SPENT NOW IS MONEY SAVED FOR THE FUTURE

# CRAMM - AN INTERNAL AUDITOR'S VIEW

JOHN BEVAN

*This is the text of a presentation given at the CASG meeting on 16th January 1991.*

**Introduction**

CRAMM is a risk analysis based method for designing an IT system's security. Although it is not an audit technique, it is of special interest to internal auditors, as:

- a practical application of risk analysis theory,

- a source of checklists useful to computer auditors,
- providing valuations for IT systems which could be used, for example, to assist in disaster planning,

- it supplies a list of supposedly installed IT security measures for audit compliance testing,

- a substitute for, or cross-check on, computer audit activity.

It is necessary to describe CRAMM in some detail so that its relevance to internal auditors can be appreciated fully.

CRAMM is the CCTA Risk Analysis and Management Method, and CCTA the UK Government's Central Computer & Communications Agency. CRAMM is applicable to both government and other IT systems. It is the UK Government's preferred method, and is supported by PC software. It is particularly comprehensive in its scope, and can be applied to existing IT systems or to those in development.

CRAMM's three stages

After initial planning to draw system boundaries, to identify the people whom the "reviewer" will interview, and to set a project timetable, there are normally three stages in the typical review. Each stage should be completed and its findings agreed with management before the next stage is started. If this is not done, and the work of one stage is subject to significant revision, then time will be wasted reworking the following stages.

Stage 1

- Identify and enumerate physical and data assets

- Value all assets: - using the nine standard impact types - elaborating the worst case scenarios triggered by the identified impacts - applying standard valuation guidelines to these cases - entering the results into the CRAMM PC software - Link data assets to associated physical assets

Stage 2

- Measure asset threats and vulnerabilities - completing the 32 pairs of questionnaires provided - in the assumed absence of security countermeasures - entering the results into the CRAMM PC software - Security requirements computed by CRAMM software

Stage 3

- Large countermeasure library scanned by CRAMM software - Automatic selection of countermeasures as: - applicable - recommended on security grounds, i.e. not excessive - Reviewer records countermeasures already in place - Reviewer considers and recommends further countermeasures - Management chooses new countermeasures, and perhaps discards some installed ones - Countermeasure status entered into PC - List the in place, agreed for future implementation, and dis- carded countermeasures

**CRAMM's nine standard impacts**

Anything that can adversely effect an IT system will fall into one of the nine standard impacts:

- Physical destruction of equipment

- Destruction of the data

- Data unavailable for short, medium, or long periods (3 impacts)

- Disclosure to the organisation's staff or to outsiders (2)

- Modification of data, either accidentally or deliberately (2)

The reviewer talks to those managers and staff who understand the IT system and its use, and asks them to imagine the worst possible situations that could arise, should any examples of these impacts succeed. This allows the collaborative development of "worst case scenarios", based upon which the valuations are made.

**Eight valuation guidelines**

Some impacts will have clear monetary consequences; others will not. In order to handle both consistently, all valuations are placed on a points scale from one to ten, with ten being the most serious. The valuation guidelines and their extreme values are:

- Political embarrassment, scoring 1 to 10. Loss of customer confidence and similar terms may be substituted for central government concerns here.

- Personal safety, scoring 2 to 10.

- Breach of personal privacy, scoring 1 to 3.

- Failure to meet legal obligations, scoring 3 to 10.

- Breach of commercial confidentiality, scoring 1 to 5.

- Financial loss, scoring 1 to 10.

- Disruption to activities, scoring 1 to 10.

- Replacement costs, scoring 1 to 9.

In some cases certain valuation guidelines will not apply. Each guideline is carefully defined, to enable its consistent application. The results are fed into the PC, for use by the CRAMM software in later stages. As at the end of the other stages, a meeting is held to agree the findings before proceeding further. The software prints reports of the findings to date. In many cases management are surprised at first by the high value correctly placed on IT system data.

When valuations are generally low, a shortened review is done.

**Threat and vulnerability measurement**

The reviewer places all assets into groups, with each asset in a group being considered as subject to the same or similar impacts, threats, etc. For each group of assets CRAMM prints out thirty two pairs of questionnaires to measure threats and vulnerabili- ties. In consultation with management and staff the reviewer completes the questionnaires and enters them into the PC.

The very comprehensive questionnaires cover:

- fire, water damage, natural disaster,

- staff shortage, human error, - wilful damage, theft,

- system infiltration, misuse of resources,

- system, hardware, power, and environmental failure.

CRAMM aims to design the most appropriate set of countermeasures. It follows that some existing countermeasures may not be needed or are to be replaced with new ones. It is therefore necessary for the questionnaires to be answered as if no countermeasures were in place. The reviewer will however note what the installed countermeasures are, for use in stage 3. Next the PC computes security requirements, and prints associated reports.

A management review meeting is held at the end of this stage.

**Countermeasure selection - automatic and manual**

CRAMM possesses a very large and comprehensive library of counter- measures, each one of which is coded in terms of the impact(s) addressed, order of magnitude cost, whether it reduces a threat or just detects its occurrence, etc. When printed with brief descriptions the countermeasures cover more than a hundred pages.

Countermeasures are selected automatically only if applicable, e.g. fire precautions to address fire risks. Similarly the most powerful security measures are only selected when the computed security requirement indicates a real need for them. However after automatic selection overlapping or alternative countermeasures usually remain, many of which are later removed manually.

The reviewer informs the CRAMM software what countermeasures are in place, some of which the software may then suggest are unnecessary. It is then up to the reviewer, acting with management, to select the most appropriate and acceptable countermeasures. Whilst CRAMM provides some assistance, this is very much a manual exercise, requiring the reviewer to exercise significant computer security expertise and familiarity with CRAMM reports. This is the least automated part of the CRAMM review, when the reviewer must use considerable judgment and discretion.

At this stage a voluminous report may be printed by the CRAMM software. It should be summarised by the reviewer for presentation to the final management meetings. On conclusion of the review, a complete list of countermeasures can be

printed, with each one marked with its status: installed, to be implemented, etc. This list can act as the starting point for compliance auditing. The CRAMM methodology excludes compliance testing.

**Evaluation of the method**

The major benefits to be realised from performing a CRAMM review are:

- A very comprehensive review is carried out, covering a broader range of security issues that is normal in a computer audit.

- The review is a three stage process of persuasion: What is the IT system worth? From what is it at risk? And finally, what countermeasures should be taken? This approach may be more persuasive and effective than computer audit methods.

- The review method is standardised, backed by the UK Government. Consequently a high risk assessment, if reached, is not the reviewer's subjective judgment. This also helps persuade.

CRAMM also has a few limitations and potential problem areas:

- It excludes any compliance testing, so no assessment is made of the effectiveness of computer security. Well designed but inoperative security measures have little value.

- It excludes a few very business or operations specific controls that internal auditors usually look for, such as those for income maximisation or purchasing system fraud prevention.

- The reviewer needs to be experienced both in computer security and in using CRAMM. If these skills are lacking, then it is difficult to scope the review, plan the project timetable, arrange management meetings for stage completion, gain full cooperation, group assets, and choose amongst the suggested countermeasures. If these difficulties are experienced, then a review will take much longer than the five weeks or so required for a "typical" IT system.

- CRAMM reporting has two major areas of weakness, for which an experienced reviewer is able to compensate. Countermeasure reporting is relatively inflexible, and not always as helpful as it might be. The stage end reports are very voluminous, and are only reduced in size with a major effort.

- The software is now an essential part of the methodology. If a consultancy firm performs the review, then consultancy fee rates apply. However for in-house use, UK Government Departments must pay a once-off £500 sterling for the software licence, and most other organisations a figure of around £7,500. To this should be added the annual software maintenance costs (about 10%), and the costs of sending staff on a three or four day training course.

**The continuing development of CRAMM**

A new release of CRAMM (version 2) is scheduled for distribution to existing users in early 1991. It should include a variety of improvements to the current version (1.4), most notably in relating to the Structured Systems Analysis and Design Method for the development of new computer systems. SSADM is widely used in the UK by government and elsewhere. However it will not be necessary to use SSADM in order to use CRAMM. Additions to the countermeasure library, and changes to the software's menu structure are also expected. Further enhancement to CRAMM (e.g. in generating improved management reports, or in exhibiting more features of an expert system) may follow later. Major changes to the basic methodology are not anticipated.

**The internal auditor's view**

This final section summarises how CRAMM may interest internal auditors, under the same headings as used in the introduction.

The appendix describes the basics of conventional risk analysis theory. CRAMM's valuation guidelines and inner workings overcome some of the problems of conventional risk analysis. It calculates the "risk" as a security requirement. There is also a special report that suggests what countermeasures may be the most cost effective, but without a full explanation. CRAMM countermeasure selection is in practice largely manual, and does not include comparison of countermeasure costs with risk reduction, although a reviewer may do this additional analysis.

Given access to the CRAMM manual and certain reports, a computer auditor can refer to many useful and comprehensive checklists.

It is possible to apply the CRAMM valuation methods to various IT systems to set relative priorities for disaster planning use.

On completion of a review, a list of installed countermeasures can be supplied for internal audit compliance testing.

If an internal audit department has little computer audit expertise, then a CRAMM review of the organisation's major IT systems may be a very acceptable substitute for an internal audit review of the systems' control and security, and can provide the basis for the usually more straightforward compliance testing. If on the other hand there is a substantial computer audit function within the department, then (after excluding compliance testing and some other types of review) the results of past computer audit work should be broadly similar to the results emerging from a CRAMM review. In this way the latter may be used as a quality check on the former! Finding few if any differences would provide assurance. However if there were major differences this might cause concern for the adequacy of computer audit priori- ties, coverage, or computer control and security standards.

**APPENDIX. Theoretical risk analysis**

Conventional risk analysis theory uses the following formula:

The expected Annual Loss Exposure (ALE) =

Probability of a threat occurring in a year

x Probability of the threat succeeding

x £ Value of assets at risk

Assume that (as a one time banker) I keep £100,000 cash savings hidden in my mattress at home, that the chance of burglary being attempted is 10% per annum, and that a burglar's chance of success is 50%. My ALE is then £5,000 pa ( .10 x .50 x £100,000).

Consider the different ways of using this formula. I could invest in anti-burglary measures, to the extent that I estimate that the probability of a burglary succeeding is reduced from 50% to 25%. ALE would be reduced to £2,500 pa. If the annualised costs of the security measures are less than this figure, then in the long term I should save by installing these security measures. I might choose between different security measures in this way. Alternatively I could reduce the other probability in the formula, by keeping the cash where a burglary is less likely, or I could reduce the value of the assets at risk. I might even put my savings in a bank!

I may be prepared to pay a premium, of not very much more than the estimated ALE, for insurance against burglary risks.

Some problems associated with using this model are that: - probabilities are difficult to estimate, with few helpful statistics being available for guidance, - not all assets are easy to value accurately, - the impact of changed security is difficult to predict exactly, - often we cannot consider just one threat and security measure at a time, as there are some complex inter - relationships between them. For example, an insurance policy or a domestic alarm can provide protection against fire, theft, and even flood risks.

In order to resolve these problems, rankings (e.g. high, medium, or low), non-monetary valuation guidelines, and some clever statistical formulae can be used. CRAMM employs several of these interesting techniques, albeit unseen in some cases.

# END USER SYSTEM DEVELOPMENT

**Ragnall Craighead FCA**
**Investor Relations Manager, British Gas Plc**

*This is the text of a presentation given to one of our evening meetings last season by Ragnall Craighead of British Gas. His final paragraph echoes a point often made by both our existing and previous chairman. Namely that we must be careful not to slavishly follow convention by applying incorrect techniques to the job in hand simply because of what the text books tell us.*

**Objectives**

My objective this evening will be to describe what End User computer means in terms of my own experience of it and to discuss the methods of implementation of End User computing which I have used. As I am sure your interest will lie in the control and audit implications of End User computing I will move on to suggest some problem areas which may arise and discuss their implications for both developer and auditor.

**What is End User Computing**

Like many commonly used computing terms, the definition of End User computing could provide the subject for a lengthy philosophical debate. However satisfying that might be it would be of little practical value and in any event I would not claim that the End User computing developments which I have managed fairly reflect any such definition. Rather I would prefer to define End User computing for the purposes of this lecture by describing a number of the technical features which it possesses.

In my experience, End User computing has had distinctive features in terms of hardware, software, people, applications and development methodology and I will review each of these areas in turn. First of all I would like to introduce the subject by explaining why I adopted End User computing in the first place.

**Why End User Computing**

In 1980 I was running a small department of about ten programmer/analysts and a five supporting staff devoted to the development and operation of financial systems. These systems were almost entirely written in Cobol by professional programmers under the direction of systems analysts reporting to members of my team. So the total number employed on these systems was many times larger than my department alone. The systems so developed covered basic areas of financial accounting such as purchase ledger, stores and general ledger, although pay-roll and sales lay outside our remit.

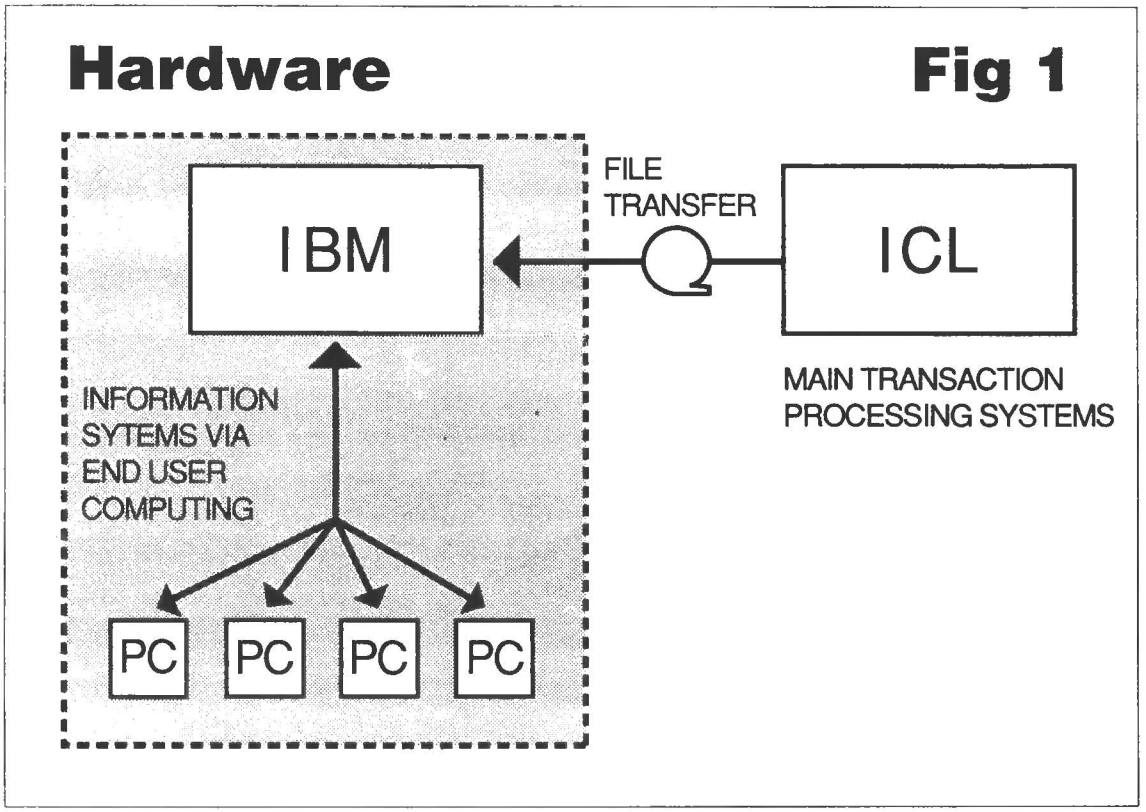
Following a period of rapid development in the 1970's the department was at that time faced with a rapidly expanding workload of maintenance and minor system enhancement which was substantially reducing our ability to develop new systems. At the same time it seemed that with each successive system development, development costs and times scales were escalating at a very rapid rate while the scope available for cost savings was steadily reduced by the savings made with previous systems. The result of this was to make replacement system development difficult to justify in economic terms and there was a perceptible and growing level of customer dissatisfaction with the benefits of computing as a whole.

Looking at this problem it appeared to me that simply trying to work the staff harder or to make the existing development methods more efficient was unlikely to achieve the leap in productivity required to overcome these adverse trends and restore our favour in the eyes of our users. Over the next three years I gradually concluded that conventional systems development was a business out of which we would be wise to get if we could only find a better way of doing things. Fortunately, in the period from 1982 to 1985 a number of hardware and software tools began to become available which offered the alternative modus vivendi that the department needed. I should emphasise that we would not lay claim to having planned all the moves that followed from the beginning; so much as responded to a problem, adopted new tools as they became available and been willing to deploy them in a fashion which was appropriate. End User computing as I describe it tonight, was only foreseen with 20/20 hindsight.

As a result of these changes by 1989 the department had reduced to about five programmer/analysts making very limited use of external software development resources. Virtually all system development for some time had taken place using End User software and a vast majority of department's time was spend supporting development of systems rather than maintenance. The systems completed and under development were by then achieving the desired level of functionality in the eyes of the users at costs and within time scales which they considered to be highly acceptable and would have viewed as impossible a few years before.

**End User Computing - Hardware**

The hardware environment is that of a large utility of with 80,000 employees and an annual turnover of £7 billion. In the Headquarters in which my team worked, the mainframe set up consisted of an Amdhal 5860 operating VM-SP and MVS, a dual ICL 3980 with the VME/B operating system and a very substantial number of micro-computers, most of which would fall into the Intel 8086 and 80386 chip categories operating IBM compatible DOS (Fig 1). The usage of this configuration by finance has generally been to run the basic transaction accounting systems such as General Ledger and purchases on the ICL and to transfer data files to the IBM for Mainframe End User computing. The IBM then acts as a distribution and collection point to PC's which may act as local processors or dumb terminals as occasion demands.




**End User Computing - Software**

The software environment for End User computing in our operation was based around the System W product set for all mainframe End User computing and substantial portions of PC based processing. This particular product is available with a substantial proportion of its most significant features on micro computers. A feature which has been used for training, data collection and distribution. The most significant feature of this particular package is that it is built round two processing engines rather than one. It does possess very substantial database capabilities of a relational type comparable with the dedicated database products on the market for the purposes of the End User. However, more important to System W is the other major engine which is a multidimensional spreadsheet capable of operating not just as a two dimensional spreadsheet with paging as is available in the latest versions of the better known PC spreadsheets as LOTUS 3.1 and SUPERCALC 5 but of processing in a truly multidimensional manner. The package contains some powerful modelling facilities which reflect its genesis and a variety of data acquisition and presentation modules.

We also used PARADOX for dedicated stand-alone database applications on personal computers and SUPERCALC 5 for PC spreadsheet work.

**End user Computing - People**

The people involved in End User computing do not split simply into professionals and End Users. Rather there is a spectrum of technical skill from the technical support specialist with operating system skills through to naive End Users with very limited computing knowledge (Fig 2). Within this spectrum each level of expertise perceives the next level down as being the "End User" population. In fact, what is happening is that expertise is cascaded downwards from the highly expert to the inexpert with increasing numbers at each layer. As you can see from the visual, at the top of this pyramid one has the technical support group of one or two people within my team, possibly receiving external support themselves from dedicated professionals within software and hardware suppliers or specialists sections of the professional DP department. The technical support group have, as their users, the analyst programmers who are each responsible for a number of application systems and End User departments. These analyst programmers, who tend to be recruited from both accounting and data processing backgrounds, will look after a number of leading users in End User departments. These leading users - often called "local heroes", generally represent the most computer skilled member of their department, possibly working at a junior level in their local hierarchy but to whom their department will look for immediate support. I would observe in passing that there is quite frequently job movement between the "local hero" in the user department and the analyst programmer in the End User support group.

The "local heroes" in turn provide support for many more accounting End Users who range in skill levels from those who are capable of initiating and building new End User systems (a minority) to simply End Users operating the systems for the purposes of their job.

Finally, I should draw your attention to the fact that these finance "End Users" may have their End Users in turn, both in terms of paper output and access to systems. This is because the product that they provide - financial numbers - is frequently today delivered in an added value form by being provided as part of an End User system and reports to other departments.

**People - Key Factors**

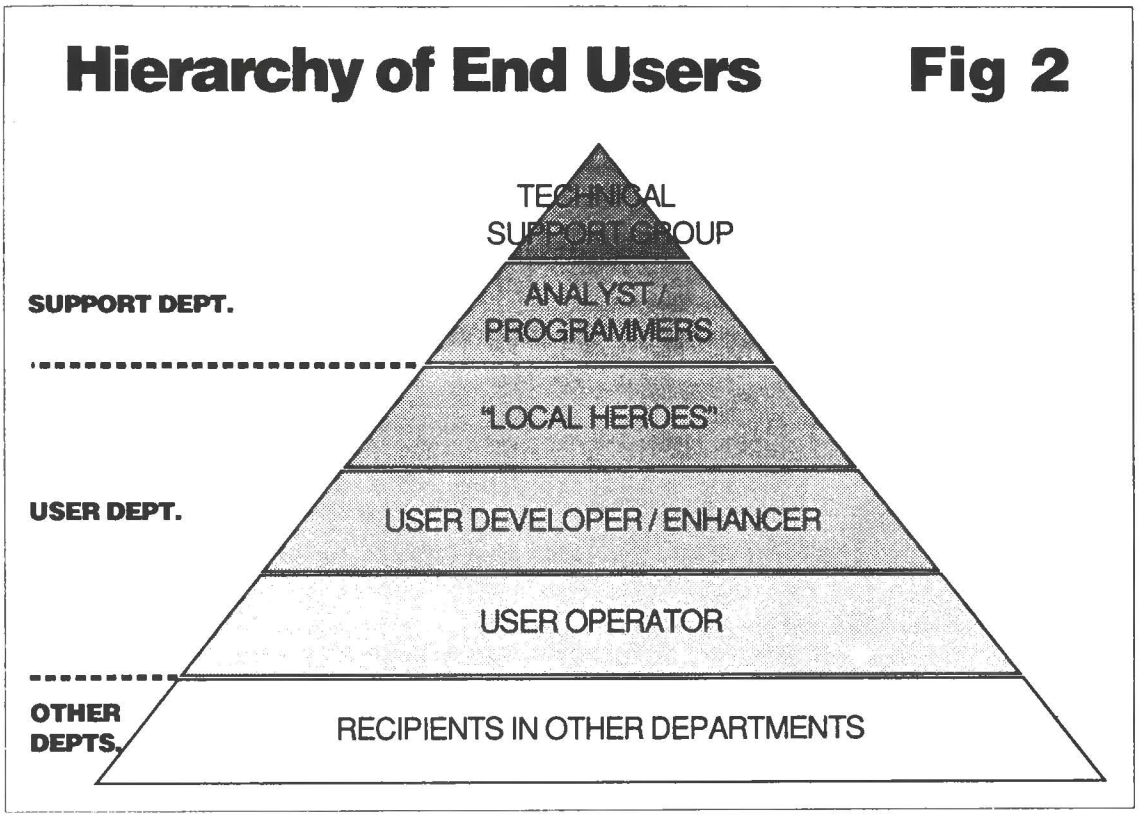
From our limited experience I would suggest a number of key factors in the successful development of the personnel aspects of End User computing. It should be emphasised that a successful policy in managing people is a critical component of a successful End User strategy without which it will surely fail.

The first factor is that it is essential to delegate development as far as possible. By this I mean that delegation should be continuously promoted downwards through the pyramid to the lowest possible level and that one should be continually achieving new standards of delegation in getting more and more done at user levels hitherto considered impossible for the projects in hand. Clearly there is a balance to be struck in this respect but if you fail to constantly probe the acceptable boundaries, the policy will stultify. It is important to remember that software is continuously making possible levels of delegation which were previously uneconomic.

The second key factor is that the whole point of the hierarchy I have described is that problems should be solved as locally as possible. At each level any problem experienced by a person should be solved within the person's own work place by his peer group 99% of the time and only 1% should be pushed up to the next level. In this way very few problems rise to the top of the pyramid saturating its support capabilities and demanding expensive specialist expertise. The corollary benefit to this is that if a problem can be solved within the person's work place it will be solved many many times more quickly than if it has to be dealt with by successively more remote support groups. Any of you who have tried to make use of help-line facilities will know what I mean.

The third key factor is that you must have commitment to continuous training,. It is not good enough just to set up initial training and then to sit back. Training must keep on coming, (a) to extend the skill base at the bottom of the pyramid and (b) to drive people further and further up the skills pyramid. If you do not do this your supply of capable people will dry up and the impetus of End User computing will fade out. A further point you should bear in mind is that facilities which were viewed as being technical a couple of years ago, must become common-place now if we are to carry on driving the solution to problems down through the user levels.

The fourth key factor to bear in mind is that impetus is related to duration of the policy. It takes a lot of effort and a long time to get an End User computing set-up going, but with good commitment and support it can feed on itself, increasing its strength as End Users grow in confidence and ability and as the network of users fills in. Successful End Users move to new jobs taking their skills with them and applying them in new areas while your whole population improves in its computing skills. Thus it is important to develop such a policy over a period of time and to allow it to have sufficient consistency that it can develop some real power.




The fifth point I would emphasise would be our consistent policy of using spare user time as far as possible. Only in the most exceptional cases have we ever allowed End Users to reduce their commitment to their day to day work in order to develop End User computing. Training, development and operational start-up have all been carried out on substantial mainstream projects without the need for additional manpower to be employed on the users routine jobs. This use of spare time frees us from heavy accountability for time spent in developing End User computing and the consequent need to rush systems development in order to improve discounted cash flow calculations.

The sixth point I would make is the cultural one of the importance of letting go. It is very difficult, but the support department have got to accept and assist in transforming themselves from system developers into consultants. Technical people often find this very difficult and it has to be done, not just once, but continuously, because the users will constantly be growing in skill, and software will constantly be de-skilling activities which were previously the professional's preserve.

The last factor is that users should be responsible for initiating projects. Our experience is that the closer to the point of use the application idea comes from the more profitable it will be. Computer professionals and support groups should always be aware that while they may and do initiate projects, they will always come second to the ideas initiated by the ultimate users because of the users enthusiasm and superior knowledge of the subject matter. User initiated projects not only tend to give better paybacks, but quite frequently are deceptive in that their technical difficulty is often relatively low.

**Development Methodology**

I think it is worth talking for a little while about the general methodology which we have used for developing End User computing systems. This is because while it in no way resembles the complex and sophisticated methodologies used so widely within computing and with such varying results, we have nevertheless developed it over a number of years and it seems to work for us.

The first feature of our methodology is the extensive use of fast prototyping. That is the building of small scale working systems. This is an area I will cover in later slides.

The second feature of our development methodology is that it is iterative. By this I mean that there is a repeated cycle of development which comes back to

the user and asks him if what we have developed is what he thought he was asking for when he told us what he wanted. We don't do this once but many times.

The third feature of our methodology is that all facilities are released in phases. The system itself evolves from prototypes into a fully fledged system by a series of improvements to its capabilities each of which is a distinct incremental release.

The fourth feature of our development methodology is that each version of the system we develop - the phases I referred to before, is released not to everybody simultaneously but selectively so that we develop facilities in phases and we release them to users in phases.

The fifth feature of our development methodology is that we force ourselves to restrict specification to an absolute minimum level. This is the trade off for fast prototyping, phased development and phased release.

The sixth characteristic is that documentation is developed after the development process is completed.

In summary, what we try to do is to achieve small safe slices of development so that there are no nasty surprises in the development process.

The methodology does have some tricks to help make it work as well. These I would list as being:

1. Involve the users as the main development workforce in accordance with the principles of maximum delegation to End Users.

2. Use user's spare time.

3. In order to make the use of user's spare time workable, it is imperative to avoid peaky workloads. By this I mean avoiding the classical engineering approach to projects with its massive peak of manpower during the development. This has been commented on by a number of writers as being one of the key causes of project failure even when applied to the sort of projects where it makes most sense. That is to say the ones where computer system development bears the closest analogy to large engineering projects and where the conviction that "if we only planned out everything in sufficient detail before we started", holds the greatest sway. With End User system development it is better to treat the users spare time as a continuous trickle of resource and match it to the work in a series of small phases.

4. The phases must be of deliverables. That is to say the user of the application must be able to work the output of each phase or he won't get any feed back and it wasn't really a phase.

5. It is wise precaution to concentrate on the simple features of major systems first. In our experience avoiding tackling the the complex features of large systems has a number of benefits. The simple features provide a good area for gaining experience with the software tools being used and when a number of simple features have been dealt with, they often provide a far clearer picture of what remains to be done and eliminate ambiguity on the interfaces. This sometimes occurs to such an extent that when towards the end of the project we come to tackle what was expected to be the difficult feature it simply isn't very difficult any more.

**Fast Prototyping**

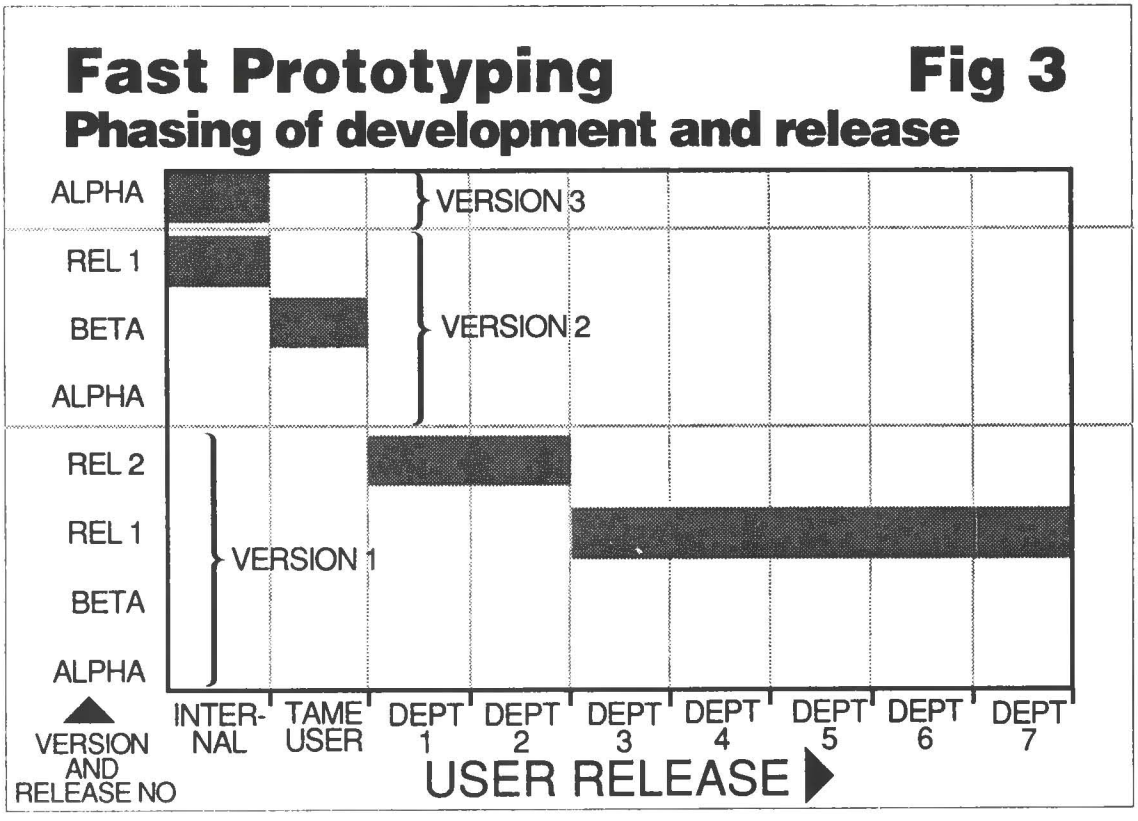
I think I should cover in a little more detail what we mean by fast prototyping and how it works. First of all, we classify prototypes into a number of different classes. These reflect the degree of development of the prototype and its usability. The system is not rigid and the nomenclature I shall show you is perhaps artificially precise. Nevertheless I hope that it will give you an idea of the way we go about it (Fig 3).

The very first prototype to be built is called an Alpha test prototype. This usually has the following characteristics, it is built by a pure technician, it is quite unusable by the End User and its aim is to prove the fundamental technology which is proposed to be used for the job. As a piece of software, it is entirely disposable and there is no prospect of it ever being released to an End User. Instead, it is used for internal evaluation purposes, perhaps to work through with the End Users and discuss how we are going to tackle the job and to suggest benefit areas. Alpha test prototypes are usually undertaken at the very beginning of new projects when we are relatively unsure of the achievability of system features and the best way of going about them. I should emphasise that they are not produced for every development or anything like it. Only a small minority of developments have problems of a sufficiently technical nature to require Alpha test prototypes.

The Beta test prototype is probably still developed within the computer support unit by one of the system analysts/programmers or one of the best End Users in the operating department. A Beta test prototype should be usable by an End User and it should be productionised to some extent, that is to say the code should be capable of being developed and maintained and should be reasonably stable. Quite often some of the code from a Beta test system will survive recognisably in subsequent live releases. A Beta test system

will typically be used on live data instead of a carefully prepared half dozen test items, and it should work reasonably well on them. At the very best, a Beta test version may be released to one very experienced End User for testing in the live environment.

The next version I would call Release 1 changing the nomenclature to reflect the fact that whereas Alpha and Beta prototypes are generally not intended to be used by End Users, subsequent versions entitled "Release 1, 2, 3" should be. Release 1 should be developed by the system developer in the End User department in conjunction with the analyst programmer in the support unit. It should be reliable and usable by End Users with code that is of good quality, stable and well structured, suitable for maintenance and amendment. It should be sufficiently reliable that it can be used for demonstration purposes to the outside public and would probably be released to a limited number of users in a user department in a progressive manner.




The next release being Release 2 would, generally speaking, be Release 1 cleaned up with some small enhancements and suitable to release to a wider audience.

These classes of prototype would operate within versions, that is to say, there would be a version 1 of the system which ran through from Alpha prototype to Release 2 and this would be followed by version 2 of the system which would be developed in the same way. Sometimes, if the technical developments of a subsequent version were of a relatively straight-forward nature, it might not be necessary to produce Alpha and Beta prototypes if there were no fundamental technical developments to worry about. In all of this work it is imperative that the closest and most direct contact is maintained between the levels of the hierarchy. The End Users must advise and criticise the technical experts Alpha test prototype while the advice of technical support may be vital in avoiding technical pitfalls.

The effect of the this approach to prototyping and version release is to minimise system errors in the user community far more effectively than can be done by testing in water-tight compartments and then releasing on an unsuspecting user population in one hit. No matter how good your development, errors are always going to creep through. With the method we use what we have sacrificed in terms of the weight of system development and formal testing, we have gained by restricting the exposure

at any one time to a very small increase in facilities which is released in a very incremental way to users. Thus the chance of a major error affecting a large population of users is, in practice, very low indeed. If I may draw an analogy, the method we use is the method that you use to drive a car. You don't sit down and plan out your journey on a map in enormous detail in advance, because you are aware of the limitations of your ability to measure and forecast, with any accuracy, the precise route and conditions. Instead, what you do is that you drive along, looking at the road, and continuously using the feed-back from it to correct your course which you have planned from a map only in outline terms. It may sound counter-intuitive not to go in for the most elaborate paper planning but in fact continuous feed-back systems are used successfully by humans in many spheres of activity with higher reliability than is commonly encountered by the elaborate pre- planning systems. The use of the analogy of computer system development to engineering projects which has pervaded software development from the first, has, perhaps, been in some ways false and misleading one. Not least because the response to every failure has been to apply the engineering approach even more rigorously thus exacerbating the problem.

**System Specification**

I earlier alluded to our methodology using minimal levels of specification. Good quality specification is important but it is our belief that the quality and usefulness of a specification is inversely proportional to its thickness. As a rough rule of thumb we would, at the inception of a project, insist that the outline specification should not occupy more than 10 man days or be more than 20 pages. Within this scope it should contain the definition of the key modules of the proposed solution and the rationale of the approach to dealing with the problem. This specification is then subjected to a professional review either by a leading and trusted member of the user support team or even, on some occasions, calling in external consultants experienced in the particular software tools. The professional review is normally limited to a maximum of 5 working days. Incidentally, the outline specification will quite often be prepared after an Alpha test prototype has been developed "playing" with the technology which is proposed to use.

**System Documentation**

System documentation is carried out retrospectively when the system has become stable. It consists of file specifications, brief but useful descriptions of of the processing modules and activities, and the necessary instructions for system, operation and support. In this way, the knowledge developed in the building of the system is retained for subsequent work in a form which is sufficiently compact that it will actually get read and which reflects the system as built rather than proposed.

**Applications**

The general approach I have outlined so far in this presentation has been used for quite a wide range of applications. These have covered not only personal computing applications and management information applications, but in many cases have replaced what would otherwise have been major mainframe system developments of a conventional nature. The following is a partial list of systems developed in this way over the last few years:-

1. Consolidation and reporting of national management accounting systems.
2. Budget preparation.
3. Financial consolidation and reporting.
4. General ledger report writer.
5. Executive information systems.
6. On-line retrieval "drill-down" of general ledger and other systems.

**General Ledger Report Writer**

To take one of these applications as an example, I would like to describe how we replaced our general ledger report writer with an End User written system. In the late 1980's, we found ourselves in the position of running an in-house general ledger with a dedicated in-house written report writer, both of which were over 10 years old. The report writer was very restricted in terms of flexibility and produced reports with a very poor standard of presentation. An advanced facility at the time of its design, the report writer now compared badly with the sort of facilities that you are used to in computer audit packages and which are provided in the report writers of general ledger packages available on the market today.

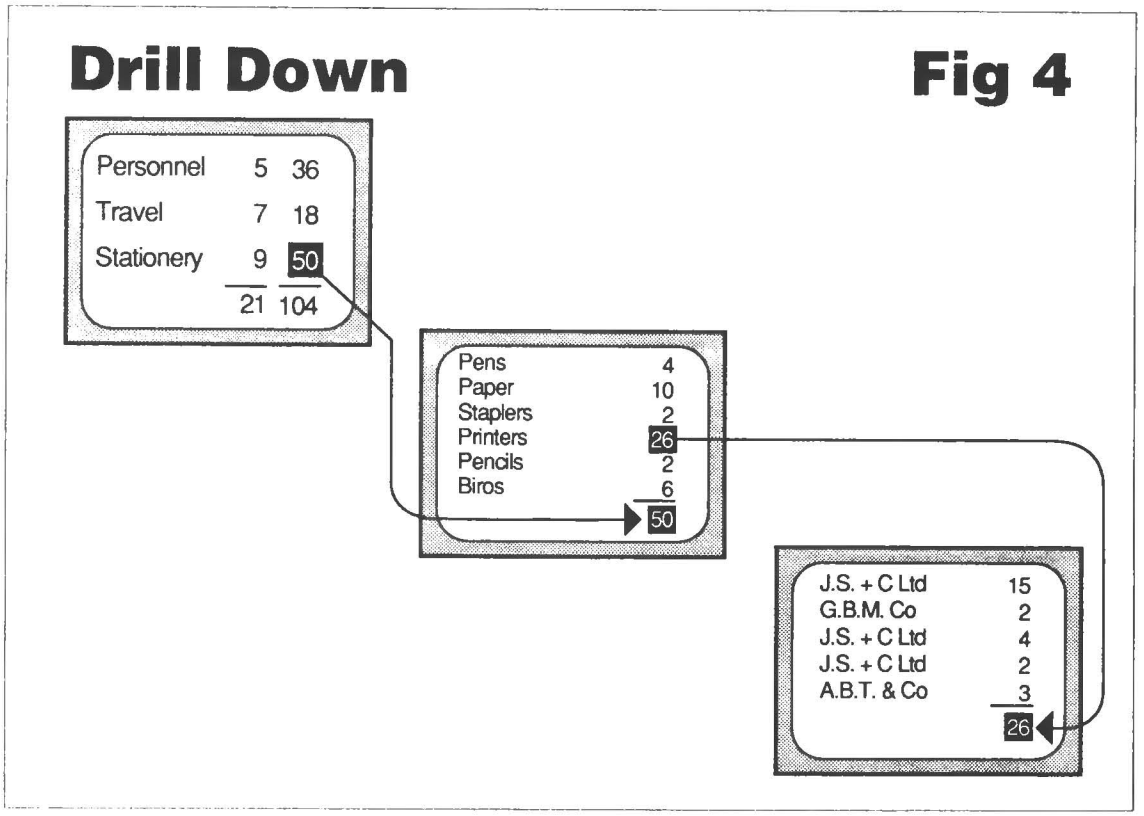
The package operated as a parameter driven Cobol programme. Such was the ease of use of this system that we found it took 500 pages of source code parameters to produce the standard management accounting report for all of the reporting units in a single division.

In the mid 1980's, consideration had been given to providing a new in-house developed report writer for

a proposed replacement system. After several years' work using structured design methodologies, the proposed replacement system would have been written in conventional database software at a cost and delivery date which were both too great already and rising exponentially at the time the project was cancelled. Fortunately, in the Management Accounting Department responsible for maintaining and using the old package, one of the accountants realised that there might be a short-cut alternative using existing End User computing facilities. The work required to link the End User written budget modelling system to the general ledger had already been carried out, and this involved data-mapping between the two systems so that finalised budgets could be carried into the general ledger system for routine accounting purposes and actual expenditures could be transferred to the budgeting system to form the basis for future modelling activities. The accountants realised that this link would work equally well for transferring data from the general ledger into the System W modelling package which could then be used as a powerful report writer. A quick prototype programme was written to prove the feasibility of this approach, which showed that the existing reports could be replaced at a cost of approximately half a page of source code per division, although the programmes as finally run now require 2 pages of source code per division, because of the inclusion of vast improvements to the presentation and printing of the reports. Within a short period of time, this approach was adopted for the management accountant's general ledger reporting for all divisions, with a considerable number of incidental improvements in terms of turn round of amendments and flexibility to cope with change.

**On-line Retrieval System**

Another application which came into use at the end of the 1980's, was the facility to provide on-line access to the general ledger down to the level of individual transactions. It had been accepted, since the computerisation of general ledger accounts, that the data volumes involved with our general ledger were such that the retention of transactions in detailed form on the computer was not possible for more than a very limited period of time and with very limited access facilities. In the late 1980's, the transformation in the costs of storage resulted in this no longer being the case. It had always been a dream by the accountants that they would be able to avoid the lengthy processes of cross referencing and digging through files of printouts in order to get behind the summary ledger totals with which they were provided. In pursuit of this dream, the support group commissioned a very much alpha system prototype with the objective of providing the simplest possible access to such facilities.



The general definition of the requirement was that a manager should be able to look at his summary reports on a screen, move the cursor onto any figure which he wishes to know more about, and hit the return button (Fig 4). The system would then retrieve the make-up figures which had led to that total, working through intermediary layers of detail where reallocations took place, down to the individual transaction at the bottom of the pile. At each stage on the way, the manager when presented with a further break out of a figure he was chasing, would be able to put the cursor onto that figure which most interested him and by hitting return drill down to the next layer.

I must confess it was much to our surprise that we discovered that not only was this possible using our main frame End User computing software, but that it was possible at a cost in terms of resource usage and response time, which equated to the best that the professional computer department could offer, and at a development cost and flexibility which were only a fraction of the professionals'.

As currently operating, the system is menu driven by users with individually personalised security controls over access on a need to know basis. The system provides the information to read on screen, to be dropped down to a local laser printer or as an extract file for manipulation by other forms of software on the mainframe or Pc. The typical retrieval time of 6 seconds would be unacceptable for a transaction processing system, but given the complexity and power of what it offers is distinctly preferable to the two days which such enquiries would have taken before.

One of the features of the system has been that the relational databases facilities have been used to join source data from different systems, so that, for example, expenditure details from the general ledger are provided not with a source code of supplier, but with with suppliers name which is drawn from the separate purchase ledger system.

**Problem Areas**

From an auditor's point of view, there are clearly a number of potential problem areas in the style of computing, which I have outlined in this talk. It seems to me that auditors are likely to be concerned about the reliability of systems built in this way and the security that they offer over the data contained within them. Staff turnover may also be a cause for concern, due to the localised nature of system development and lack of uniform development procedures. Another concern which can be raised, is how such an approach fits in with the concept of "data as a corporate resource". Then again, the auditor will be concerned to know that the systems developed in this way will be as easily auditable as those developed by conventional methodologies. With the development and operation of systems by the same person, it is necessarily the case that some of the division of duties that auditors have become used to with professionally developed system will be lost. Meanwhile, other auditors may raise concerns about machine efficiency and the implications for the relationship with the professional IT department.

With regard to reliability, I can only observe that having worked with this style of End User computing for 9 years, while at the same time managing developments and operations of conventional computer systems, the worst experiences that I have had have been with the conventionally built systems. In general, reliability problems have been caused by poor communication between separate departments and cultures against which all the recommended paraphernalia such as tightly written specifications and regular liaison meetings with formal signing-off procedures have turned out to be no defence.

With regard to security, I would suggest that you should always examine an End User computing system by comparing it with its paper equivalent. It is almost always the case that security levels in End User written systems would be considered quite adequate for paper based operations. I would ask what levels of security we should be operating for the desks and files in our offices which are much more easily accessed and only controlled with much more difficulty. Where necessary, we have in fact introduced considerable security measures including systems that have separate access modes including read only and run only.

Turning now to the potential problems of staff turnover, I would counter this by again emphasising the importance of personnel development policies in making End User computing work. Provided management ensure that the skills pyramid is consistently renewed at the base and that experience is broadened by the use of common software over a wide range of activities, problems of staff turnover will not be material. It is of course important to ensure that software is used for a reasonable duration, so that skill levels can rise and the base be broadened while at the same time the software is used on a wide range of activities again broadening the base. Finally, I would restate the importance of continuing training aggressively throughout the staff base.

Far from End User computing conflicting with the concept of data as a corporate resource, it has generally been found in practice that these tools offer far easier and better access than the professional systems. For example, the general ledger report writer and on-line access in System W on an IBM was

provided by Finance to Engineering department. The Engineers, who were using MAPPER on a UNIVAC, were delighted to find that access to System W files was considerably easier than it had been to the Cobol generated files of its predecessor.

With regard to auditability, I would concur that End User computing must cause problems to some of the classical computer audit approaches. I would suggest that the appropriate response would be less of a systems audit emphasis and more concentration on balance sheet and transactions. Auditors must recognise that most of the control features in conventional systems were installed, not for their benefit, but in response to the many processing problems found in the early days of computing. Auditors merely wrote the features up in their textbooks. To force anachronistic practices upon today's technology in the name of control would be like putting new wine in old bottles. It isn't the right solution and you won't succeed in imposing it. The advent of new approaches to computing will require matching developments in audit technique and I look to the audit community for a thoughtful and original response.

# PEOPLE

## ALISON WEBB BA ACA

Alison is an independent computer audit consultant, based in Cambridge. She divides her time between auditing large IBM mainframe systems, and giving general advice and assistance to firms with business computer needs.

She trained as an accountant with Pannell Kerr Forster in Derby before joining the computer audit department of KPMG Peat Marwick Mclintock in London. There, she specialised in the audit and security of large systems, including audit interrogation, system and application software reviews and installation reviews.

In 1989, she joined Peters Elworthy and Moore, an independent firm of Chartered Accountants based in Cambridge. There, she managed a general audit department (and tried to remember what she'd forgotten about company taxation!), but her special responsibility was to develop computer audit in the firm. She introduced in-house file-interrogation techniques, and worksheets and questionaires based on simple risk assessment to evaluate security and controls in small and medium sized computer installations and systems, as well as guidelines for PC security.

1990, computers triumphed over corporation tax, and Alison left general practice to concentrate on computer consultancy.

She is responsible with John Bevan for organising members' meetings.

## IAN LONGBON

Ian read mathematics at the University of Manchester Institute of Science and Technology and then joined Grant Thornton in London. Having a Maths degree he seemed the obvious candidate to evaluate the new IBM PC and some software called Lotus 123. This lead to his interest in the use of computers in a business environment.

After qualifying as a Chartered Accountant Ian left Grant Thornton to join the Computer Audit Department of Peat Marwick. During his three years there he performed a variety of review work, wrote and ran many interrogations for audit teams, and taught on a number of courses.

He is now Internal Audit Manager with Chartered WestLb, responsible for the audit function within the bank.

Ian is a member of the CASG Management Committee responsible for the organisation of the group's annual conference.

# COURSES AND OTHER DATES OF INTEREST

*This list has been prepared from material collected by several members of the editorial panel in the belief that some of these items may be of interest to CASG members. No responsibility is accepted for the correctness of items. Further details should be sought from the event organisers whose details are given at the end of the list. Listing is free. If you have details of an event that may be of interest to other members please send details to; A.J.Thomas, 3 Kings Court, The Maltings, Great Dunmow, Essex CM6 1UX*

| TITLE | SPEAKER\LEADER | DATE | LOCATION | ORGANISER |
|---|---|---|---|---|
| | | 1991 | | |
| S.S.A.D.M. for Users - Version 4 | | 8-9 April | Manchester | N.C.C. |
| Audit and Security of LANS and Micro to Mainframe Links | | 8-10 April | Amsterdam | Elsevier |
| Filetab (IBM/UNISYS) | | 8-12 April | Birmingham | N.C.C. |
| Risk, Materiality and Audit Planning | Mike Thexton | 9 April | London | CPE |
| Introduction to Computer Auditing | System Security Ltd | 9-11 April | London | IIA |
| Advanced Computer Audit Workshop | System Security Ltd | 16-19 April | Dublin | IIA |
| Risk Management | | 17 April | Nottingham | CIPFA |
| Audit Conference | | 17-19 April | Harrogate | CIPFA |
| The Law and Computer Security | Gerald Vinten City University | 18 April | London | EDPAA |
| Internal Audit in Banks | Peter L. George | 21-26 April | London | MDC |
| Auditing in an IT Environment (I) | | 24-26 April | London | CAET |
| Fraud & Corruption within an Organisation | Michael Levi Cardiff University | 26 April (am) | Birmingham | IIA-Midlands |
| Fraud & the Auditor's Responsibility with regard to PACE | Tony Cavaciuti Gwent Constabulary | 26 April (pm) | Birmingham | IIA-Midlands |
| Computer Audit Workshop for DP Professionals | System Security Ltd | 29 Apr- 3 May | York | IIA |
| Computer Audit Workshop for Accountants and Auditors | System Security Ltd | 29 Apr - 3 May | York | IIA |
| Measuring and Managing IT Benefits | | 30 Apr - 1 May | Henley | Henley M.C. |
| Challenges for Audit Management | | 1 May | London | CIPFA |

| | | | | |
|---|---|---|---|---|
| The Audit of Human Resources | | 2 May | London | CAET |
| Using the Micro to Enhance Audit | | 3 May | Coventry | CIPFA |
| Development Seminar for Heads Internal Audit of | L.G.Westwood | 8-9 May | Bath | IIA |
| Computer Virus Protection | Grenville Mills | 9 May (pm) | London | CHARTAC |
| Recent Developments | Georges Selim City University | 9-10 May | London | MDC |
| Security and Audit of SNA Networks (Incl. ACF/VTAM & NCP) | | 13-15 May | Brussels | Elsevier |
| Managerial Auditing | Gerald Vinten City University | 13-17 May | London | MDC |
| Computer Application Audits - A Methodology | System Security Ltd | 14-15 May | London | IIA |
| IFIP (7th Conference) Implementing Effective Information Security in the 1990's | | 15-17 May | Brighton | Elsevier |
| Creating Confidence in Information Processing | | 15-17 May | Brighton | Elsevier |
| The Audit of Building Societies | | 16 May | London | CAET |
| Computer Security | System Security Ltd | 16-17 May | London | IIA |
| Common Sense Computer Security | Martin Smith | 16-17 May | Manchester | COMPSEC |
| SNA Workshop | | 16-17 May | Brussels | Elsevier |
| The Audit of Management Information Systems | Paul Collier Exeter University | 17 May (pm) | Birmingham | IIA-Midlands |
| Introduction to Internal Audit | Georges Selim | 20-24 May | London | MDC |
| Auditing Microcomputers and Office Based Systems | System Security Ltd | 21-22 May | London | IIA |
| Managing Desk Centred and End User Computing | | 22-23 May | Henley | Henley M.C. |
| IBM CAATT Software | Roy Bradford IBM, Paris | 23 May | London | EDPAA |
| Auditing On-Line Datacommunication Systems | System Security Ltd | 23-24 May | London | IIA |
| Auditing in an IT Environment (II) | | 30-31 May | London | CAET |
| Investigation of Computer Abuse (Reserved for Police Officers) | Mark Tantum | 3-5 June | Oxford | Elsevier |

| | | | | |
|---|---|---|---|---|
| Audit, Control and Security of CICS | | 3-7 June | Amsterdam | Elsevier |
| Filetab Language Level Programming | | 4 June | Birmingham | NCC |
| Winning Competitive Advantage with Computer Networks | | 4-5 June | Henley | Henley M.C. |
| Auditability and Security of RACF | | 4-5 June | London | Elsevier |
| Executive Entry into Internal Auditing | Andrew Chambers City University | 4-6 June | London | MDC |
| S.S.A.D.M. for Users - Version 3 | | 10-11 June | London | NCC |
| Electronic Trading Conference | | 11 June | London | E.D.P.A.A. |
| Risk Analysis Techniques for Internal Audit Planning | Andrew Chambers | 11-12 June | London | MDC |
| Personal Coaching in Interviewing Skills for Auditors | Terence Bates Bryan Platt | 13-14 June | London | MDC |
| Introduction to DEC's VAX/VMS O.S. | | 17-18 June | London | Elsevier |
| Audit and Control of Computer Systems | | 17-21 June | London | NCC |
| Computer Audit Workshop for DP Professionals | System Security Ltd | 17-21 June | London | IIA |
| Computer Audit Workshop for Computer Auditors | System Security Ltd | 17-21 June | London | IIA |
| Advanced Audit, Control and Security of DEC's VAX/VMS | | 19-21 June | London | Elsevier |
| Advanced Computer Audit Workshop | System Security Ltd | 24-28 June | London | IIA |
| Management Information Skills | | 25-26 June | London | CIFPA |
| Internal Audit in Banks | Peter L. George | 7-12 July | London | MDC |
| A Modern Approach to Systems Auditing | Keith Wade | 8-12 July | London | MDC |
| Internal Auditing - Manager's Course | L.G.Westwood | 15-17 July | Edinburgh | IIA |

Many of the above items are courses or conferences charged at economic rates, but some are available at more modest charges. Fuller details may be obtained by contacting the organisers at the addresses listed below;

CAET Courses in "Operational Auditing" by the Management Centre Europe (ref. 1296-17), Telephone (Brussels) 32 2 516.19.11 ext 934. Fax 32 2 513.71.08.

CHARTAC Courses and Conferences, ICAEW, 40 Bernard Street, London WC1N 1LD
Tel; 071 833 3291

CIPFA Courses by Courses and Conferences Unit, The Chartered Institute of Finance and Accountancy, 3, Robert Street, London, WC2N 6BH.
Tel; 071 895 8823 (For Scotland 031 220 4316)

COMPSEC Courses organised by Elsevier Seminars, Mayfield House, Banbury Road, Oxford, OX2 7DH. Tel: 0865 512242

CPE Courses run by CPE courses Ltd. Aldine House, Aldine Place, 142 Uxbridge Road, London W12 8AW Tel; 081 749 7467

EDPAA Meetings organised by the London Chapter of the E.D.P.Auditors Association. Enquiries to Stephen Bones of Neville Russell on 071 377 1000

Elsevier Enquiries to Kay Russell, Elsevier Seminars, 256 Banbury Oxford, OX2 7DH.
Tel: 0865 512242

Henley M.C. Henley Information Technology Series Course organised by Henley,The Management College, Sharon Crabtree, Course Administrator, Greenlands, Henley on Thames, Oxon, RG9 3AU. Tel: 0491 571454

IIA - Midlands District Society. Arrangements for individual meetings will be circulated to all Midlands District members in advance of each meeting. Members from other District Societies are welcome to attend but should inform the Secretary not later than 2 weeks before the meeting. (Secretary; R.O. Welton 0242 236111)

IIA(UK) Courses organised by the Institute of Internal Auditors (U.K.), Course Administrator, 13, Abbeville Mews, 88 Clapham Park Road, London SW4 7BX.
Tel: 071 498 0101

MDC is the Management Development Centre, City University Business School, Frobisher Crescent, Barbican, London EC2Y 8HB
Tel: 071 920 0111 ex. 2278 and 2359 or 071 374 0041 (direct line)

NCC National Computing Centre, Course Administrator, NCC, Oxford Road, Manchester, Ml 7ED.
Tel: 061 228 6333.

# ABSTRACTS

**Read any good articles lately ?**

**If you have seen something that might be of interest to other CASG members please send details to "CASG Abstracts" c/o Rob Melville, Centre for Internal Auditing, City University Business School, Frobisher Crescent, Barbican Centre, London EC2Y 8HB.**

**INTERNAL AUDITING** The Computers Section of the Internal Auditing Journal for January, February and March includes;

- Microcomputer Viruses (Part Two)

- Introducing Lap Tops

- The Auditor on your Lap

- Micro Computer Clinic

- Introduction to DOS

- AS400 Software Security

- Data Downloading (Feb)

*(Reported by Malcolm Lindsey)*

# MANAGEMENT COMMITTEE

| | | | |
|---|---|---|---|
| Chairman | John Mitchell | Little Heath Services | (0707) 54040 |
| Secretary | Ragu Iyer | KPMG Peat Marwick McLintock | (071) 236 8000 |
| Treasurer | Fred Thomas | | (0371) 875457 |
| Publications | John Hession | Hertfordshire County Council | (0992) 555323 |
| Members' Meetings | John Bevan | | (0992) 582439 |
| Annual Conference | Ian Longbon | CWB Limited | (071) 220 8495 |
| Discussion Groups | Chris Birt | Ernst & Young | (071) 928 2000 |
| Marketing & PR | Harry Branchdale | British Amercan Tobacco | (071) 222 1222 |
| Membership Secretary | Peter Martin | E D & F Man Ltd | (071) 626 8788 |
| Editors, Group Journal | Virginia Bryant | City University | (071) 253 4399 |
| | Rob Melville | City University | (071) 920 0111 |
| Long Term Planning | Bill Barton | The Rank Organisation Plc | (071) 706 1111 |

# CONTENTS

# NEW VENUE FOR MEETINGS