Transactions on Information and Communications Technologies vol 4, © 1993 WIT Press, www.witpress.com, ISSN 1743-3517

Quality within software developments: Research in the financial service sector

V.M.A. Edgar-Nevill, K. Murdoch, D.G. Edgar-Nevill
Department of Computer Sciences, Napier University, 219 Colinton Road, Edinburgh, UK

ABSTRACT

This paper is based on research carried out through a series of five case studies within the financial service sector. The role of information technology within the financial service sector is introduced and the requirement for quality software developments established. The paper describes the contribution quality standards ISO9001 & BS 5750 make towards establishing 'Best Practice', and the software developments studied review the quality management, control and assurance mechanisms found within the casestudy organisations. From this work one organisation was selected for an in-depth study of quality issues throughout the software development lifecycle. An analysis of the type of software projects undertaken is made considering the software development lifecycle and project management mechanisms. A model introducing the quality control techniques on top of the development lifecycle is presented. The paper concludes with a summary of the research findings and identification of further work required in the area.

INTRODUCTION

All businesses face the unrelenting pressures of a business environment characterised by intense global competition. Moreover, this increasingly competitive world has developed against a backdrop of difficult economic conditions; including long-term high inflation, high interest rates, and low real growth.

The interaction of these two variables - information technology economics and a challenging business environment - has generated what might be called the economic imperative of information technology. Organisations that do not take advantage of the growing opportunities provided by IT are likely to slip behind in the competitive business world. This was identified as early as 1984 by Benjamin, Rockart, Scott, Morton and Wyman [1]. Within the financial service industry, there is great uncertainty, and accelerated rate of change. Markets and the structure of firms and business as a whole are in a state of flux. De-regulation, legal and institutional changes have eroded the demarcation between different types of financial institution. There is now strong competition between firms that previously had reliable markets, as well as increasingly intense international competition.

The mid 1980s saw an increased acquisition of IT in banking. 1986 was a key year in the UK, containing as it did further deregulation and particularly the rush to automate in the City - the so-called 'Big Bang' - in time for the launch on October 27. This created pressure on the less technology-minded institutions to install computer systems as a defensive measure as documented by Holloway [2]. The essential ingredients of technological advancement and financial liberalisation have combined to produce an environment in which the imaginary barriers which once separated the traditional activities of banks, building societies and insurance companies no longer exist. For example, with the entry of the clearing banks into the mortgage market in 1982, the building societies' monopoly was broken for the first time. Deeply

alarmed by this development, the building societies fought back with uncharacteristic aggression, attacking the banks where they were most vulnerable; in the personal savings sector. The result of this onslaught was that between 1980 and 1984 building societies increased their share of new (retail) deposits from 47 per cent to 65 per cent, at the expense of the banks, whose share fell from over 40 per cent in 1980 to 19 per cent in 1984. Territorial boundaries no longer exist, and rivalry exists within the retail financial service section as discussed by Rothwell and Jowett [3].

While the impact of deregulation is still being felt, the advent of 1992, and the widening of the European common market, bring fresh competitive opportunities and rivalries.

Improved data processing system, such as personal computers and Local Area Networks (LANS), now present alternatives to the established computing strategies based on the central mainframe. Delivery systems are increasingly IT-based, embracing Automated Teller Machines (ATMs), home banking, on-line brokerage systems, Value Added Network Systems (VANS) and in the future, Electronic Funds Transfer Point of Sale (EFTPOS). Linked to these are decision making systems, developments in customer databases, expert systems and fourth generation languages, enabling the more effective exploitation of information sources. Unlike previous 'back office automation', profound implications for the structure and performance of the entire financial sector are anticipated from these advances - expanding the range of services, transforming the linkages between firms and ultimately changing the customer interface and the shape of the firm itself.

It was against this dynamic sector backdrop that we choose to undertake research work through case studies within the financial service sector service. It is a sector of vital importance to the UK economy, which due to external pressures is forced to look towards IT for business solutions. This sector also provides the perfect setting for the pursuit of software quality, the concept of dissatisfied customers (be they individuals or corporate) with erroneous statements, or software errors within the S.W.I.F.T. (Society for Worldwide Inter-Bank Financial Telecommunications International funds transfer) systems are simply too 'Image' damaging for the organisations concerned to tolerate.

SOFTWARE QUALITY

The need to improve the process of software engineering became clear in the early 1980s. The increasing instances of software projects being delivered over budget and late grew, together with the overwhelming costs of maintaining existing systems. These effects combine to become the software crisis which has been discussed by many authors (eg. Macro, Buxton [4]).

Fundamental to the problems found in software development is the notion of quality. If a suite of software is delivered to specification and to a high standard, there should be a minimal need for expensive maintenance; correcting errors which should not have been in the software in the first place. To achieve this it must be possible to ensure not only the quality of the product (in this case the software) but also the process used to create it. The concept of producing a product to an established and recognised quality standard has long been in practice within the electronics and engineering industries. However it is only in more recent years that standards of this type have been

specifically developed appropriate for software products.

The more widely accepted standards include DEF STAN 00-55, BS 5750 and ISO 9001. In May 1989 the UK Ministry of Defence issued interim defence standard 00-55 "requirements for the procurement of safety critical software in defence equipment" for comment; Ould [5]. The BS 5750 is the British standard for software quality developed with the British Computer Society as the main contributing professional body. ISO 9000 is the American equivalent, with the Institute of Electrical and Electronics Engineers playing a leading role. In each case a series of standards have been developed corresponding to the various processes within the systems life cycle, and an identification is made of quality assurance management as a functional role in it's own right.

For purposes of illustration the British BS 5750 will be used as a reference point, though surveys have shown a limited take-up within industry, Daily [6]. As discussed within NCC (National Computing Centre) Guides to BS 5750 Daily [7], quality assurance can be considered as having three levels: the interactions between the three levels is illustrated in Figure 1.

Quality control
This is the task of ensuring that a process or product has been done correctly ie. to requirements and to standard. It also covers the procedures and methods used for the software development, and a mechanism for recording that the quality control has been done. The complete process of Quality Control should be defined in the Quality Plan.

Quality assurance
This is a third party independent of the work area check that Quality Control has been performed, has been effective. The principal aim is to give confidence that the jobs or product will be of good quality.

Quality management
This implies the concept of a 'Quality System', as a stated policy objective. The quality management identifies the responsibilities and internal organisation which ensures that the Quality System is built effectively. The approach to quality is usually documented in a Quality Manual which details the organisation, methods and procedures in use to ensure quality.

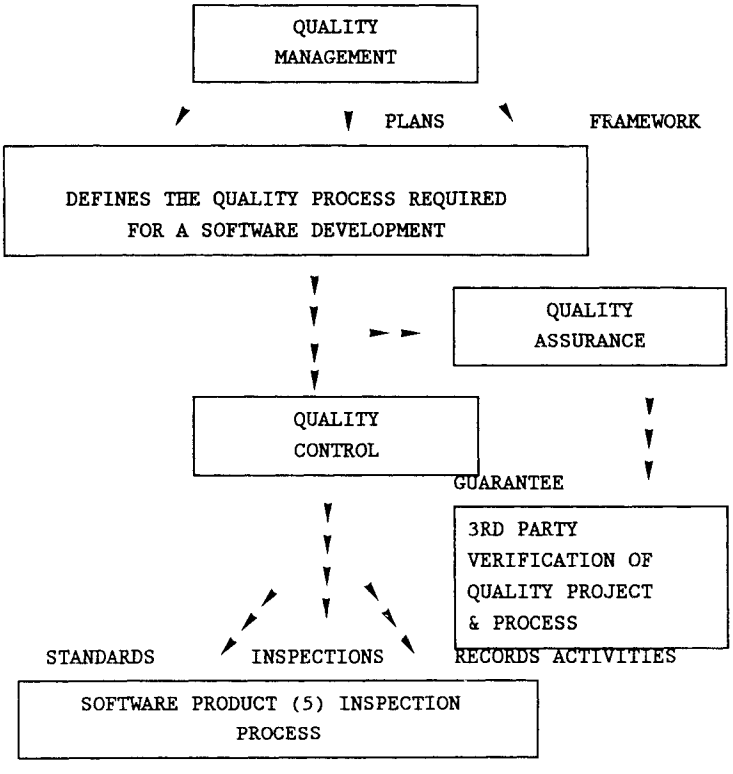


FIGURE 1 Inter-relationships between quality management, quality control and quality assurance.

Transactions on Information and Communications Technologies vol 4, © 1993 WIT Press, www.witpress.com, ISSN 1743-3517

BACKGROUND TO THE CASESTUDY RESEARCH

Five organisations within the financial service sector were visited as part of the casestudy work and a series of interviews took place at each. In all the organisations at least a software project manager and a senior information technology manager were interviewed. This work highlighted the key areas of concern and importance to companies within this industry sector with regard to software quality.

Once common issues and themes emerged, a fuller in depth casestudy was carried out with the co-operation and sponsorship of one of the organisations concerned. One of the authors visited the company on a weekly basis over a six month period conducting interviews, fact finding analysing software documentation, and sitting-in during the software development lifecycle of a selected 'typical' software development. This activity was mirrored by a company staff member with responsibility for standards and software quality.

Within the research activity key areas addressed included building a model of the organisations software development life-cycle, a review of the project metrics currently kept for software developments, a classification of the types and numbers of software developments over this past eighteen months and finally building a model of current software quality assurance practice on top of the software development life-cycle established. This research is introduced by giving essential background to the organisation concerned and introducing the management structure of it's information technology function. Each element within the casestudy research program is then discussed with the findings illustrated.

COMPARISON OF THE APPROACH TO SOFTWARE QUALITY BETWEEN THE CASESTUDY ORGANISATIONS

All the organisations felt that Quality Software was critical to both their business and operational success. However, considerable differences in approach were apparent; with a number of companies continually reviewing the quality process. To place the comparisons in prospective a brief outline of all the casestudy organisations is given.

Company A
This is the company described within the detailed casestudy description. The company specialises in the provision of management information services for owners and managers of globally invested portfolios. The whole of the company's operation depends on quality information provided by technology.

Company B
This is a large national bank, based in Edinburgh. Well-known for it's innovative use of technology, being among the fore-runners of the banking systems.

Company C
A large national insurance company based in Edinburgh. Traditional secure business facing increased competition. Responding rapidly to the situation by system improvement to provide a speedier, more customer friendly service.

Company D
A large national insurance company based in Edinburgh. At the time of the case study replacing both hardware and software. The new systems results from heavy customisation and tailoring of a large insurance package software product.

Company E
A relative newcomer to the insurance business, based in London. Fast expansion program, fully using information technology. The aim is to win customers by providing a fast, convenient and competitive service.

COMPARISON OF COMPANIES

While analysis of a sample of five organisations is insufficient for generalisation, some useful observations can be made. The casestudy work is summarised in figure 2. The larger the systems function, the more distinct this function was from the business operation and the more formal the line of communication. Otherwise the larger the systems department the more formal the hierarchial structure and the more likely the formation of centralised specialist groups for such functions as quality, training, standards, etc.

The observations are somewhat muddled by the very different recruitment and thus staffing policies of the organisation studied. The smallest systems group was in Company E. They had a deliberate policy of recruiting mature staff, preferring to recruit a new trainee from a government re-training scheme (such as TOPS), to a young computer studies graduate. They generally considered it more efficient to work with a small group of able, responsible well paid staff.

In contrast companies B, C and D rely heavily on graduate, or postgraduate conversion course diplomates as entrants. Most of the systems work is completed by larger very structured teams, with less individual responsibility. This is of course a gross oversimplification, and company B in particular have an elite team of highly skilled and experienced staff dedicated towards the evaluation of innovative software/hardware and business solutions. This team works with a maximum of structure. Company A, the main casestudy organisation falls somewhere between the two extremes. They recruit experienced systems professionals and also give systems training to business staff transferring into the information technology function. As a rapidly expanding systems department a need is perceived, and acted upon to provide staff with standard guidelines and software development aid. Concurrent at the time of research, senior management within the organisation had adopted a 'quality culture' which was gradually filtering its way down towards all levels of staff. Thus the concept of building quality into the product rather than achieving quality by control and inspection was paramount. In practical terms this was represented an abandonment of the centralised quality assurance function with the business development group managing the software development taking responsibility for embedding quality into the software process.

All the organisations recognised that there are costs associated with quality. Company E meet their objectives by paying the high quality mature staff. Company A tailored the level of quality control/cost to measure with the size, complexity and type of software development. Company B adopted different software development life cycles for different types of project. Company D employed an approach of introducing new employees to the company standards and then informally utilised the team leaders for policing them. Company C made the considerable commitment of establishing a sizeable standards/quality assurance centralised function who's personnel's experience is applied across all projects. None of the organisations studied had formalised the process of measuring quality, with the possible exception company A who did record throughout the development process, the numbers of different types of errors found during the inspection and auditing quality processes.

	Project Organisation Management Structure	Type(s) Software Project	Systems Development Lifecycle	Quality Management	Quality Control
A	Systems decentralised across single user major business divide. Pools of systems staff, utilised to form project teams for development duration.	Mainly enhancements and additional functions added to existing systems. During period of casestudy survey distorted by large system upgrade project.	Traditional systems development lifecycle modified at project initiation to be appropriate to each special software development.	At the start of the research this was a separate function but it was subsequently distributed to the project management groups.	Inspections are built in as appropriate to the project plan. (see figure)
B	Management services decentralised from the business. Very distinctive split between business analysis and software development role. Pools of latter form Project Teams.	Wide variety across very many teams. Significant amount of new development work.	Traditional systems lifecycle followed by most projects. However a prototyping all hand on deck approach towards innovative or high priority project work.	Build within a formal hierachical structure, for both teams and divisions.	Inspections performed by training department on new recruits work (for first 6 months) to ensure work performed compatible with the companies standards.
C	Large systems staff formed into sections by business function. Centralised responsibility maintained for Management Services functions.	Very traditional slowly moving towards end user systems. Half the project are new development work.	Business analysis function within user area provides communication channel between business and systems staff. Traditional lifecycle considerable use of software development tools.	Centralised function supporting about 16 ongoing projects.	Variety of approach taken in line with software development tool proliferation. New push to formalise quality control techniques implementation
D	Software development functions quite separate permanent project teams formed by business function	Two year hardware + software concession program unavailable at time of casestudy.	Use of 'Smart' Methodology formalised transfer between systems environments under implantation controls	Decentralised operates through very structured hierarchical project teams.	New staff members familiarised with company standards. Some team leaders employ inspection techniques.
E	Small team of mature staff. Organised as Multi disciplinary project teams centred around business function.	Development plus, ongoing support work allocated by functional area rather than classification.	Informal lines of communication, approach taken very project dependant with few restrictive standards or guidelines.	Mixed teams of business and systems staff oversee project as a working party.	No formal approach to quality assurance relies on small/medium sized project teams of able, mature motivated staff.

Figure 2: Comparison of Quality Approaches taken within the Casestudy Organisations

BACKGROUND TO THE INDEPTH CASESTUDY ORGANISATION (COMPANY A)

A private company formed 20 years ago from the computer systems division of stockbrokers. Launched originally to developed research services the venture blossomed rapidly through the development of innovative software to produce services that facilitated expansion from a provincial Edinburgh broking firm to one of the major players in London, and subsequently internationally.

The organisation specialises in the provision of management information services for the owners and managers of globally invested portfolios. They offer clients Investment Accounting, Valuation and Performance Measurement, tailored to meet the needs of their investment portfolios.

As the whole of the company's business and operation depends on quality information provided by technology, system control is critical. However, care has been taken not to stifle the systems professionals by over regulation and a dogmatic structured approach. (The Information Technology Director perceives this approach as common in financial institutions and the cause of staff demotivation). The concept of a project plan is in place. This is tailored as appropriate for each individual project at it's initial stage and includes checkpoints and reporting requirements both to the systems management and the user base. There is tremendous value in the existing software but through the continued enhancement process it is becoming evermore costly to maintain. The director considers a system re-write as simply a 'stalling operation' and has recognised that by the very nature of the business operation continual enhancements will always be necessary.

The line reporting structure is for the management, covering communication, holidays, and sickness. Staff act as a pool of resources assigned to individual projects or project stages. While there is a hierarchy of structure within the team, significantly a "culture of responsibility", is also encouraged. All staff recruitment is of graduate level, sometimes from within the business. In some ways the company considers it easy to give technical training to business staff rather than business knowledge to technical staff. However training is given to provide technologists with business appreciation.

The two applications development areas were organised a little differently, due for the most part to the different requirements of senior management within the different business divisions. However, during the period of research a reorganisation occurred merging the two application development areas into one. The original formation is described as being most significant in terms of the software development practice being researched. Within business area one business support analysts are a centralised user acceptance testing function under the control of technology. Originally these staff members were part of the business community spread across each of the user groups. This shift of the business support analysts into technology has not occurred within the business area two grouping. The equivalent staff are centralised within the business division (3 or 4 staff). Moving the Business Support Analysis into an improved technology is seen as giving them an improved career path.

This business area two requires some very large, complex mathematical systems. They generally deal with high level data units or subsets of the data detail found within the first business area systems. This data duplication is not considered significant however

there is very real duplication between the significant application systems in business area one. While some of the area one systems are perhaps 14 years old, more of the area two systems are more than 7 years old. Since completion of the casestudy research some upgrading of both hardware and software occurred.

Staff who move into the technology group from the business areas tend to be "Company" people, with a lower staff turnover than those who were bought for their technology skills. Many of the systems within the second business area are sensitive with confidential formula.

Company A: the management of software developments

Within the business area one very high level steering group called the BDG (Business Development Group), meets monthly and works within a three year planning time frame. All developments are reviewed by this group, which includes the business and technology director. At a lower level software developments are managed through a project management group. Within this group are:

Business Line Sponsor
User
Technology Group Representor
Stage Manager

This group reviews progress every 2 weeks, it approves plans, changes and provides assistance and direction. All requests for development work for systems go to the BDG and if approved the line manager prepares a justification paper. After the acceptance of this paper a feasibility study is initiated. The business analysts working in this stage, look at the system functionality, business justification and project plan. In general a detailed plan is done for the next stage and outline planning for the rest of the phases or stages. There is a normal backlog of two to three years. Priorities are reshuffled at meetings normally held monthly, working from a 20 month forward user plan which is prepared by the application manager. This rescheduling relies heavily on the advice of the application manager.

This structure and approval system operates less formally within business area two again due to the perceived different business requirements of the division. Rather than customised monthly meetings, the technology decision making is incorporated into the quarterly business meetings (two day affairs).

Company A: the software development lifecycle

The software development process within the organisation follows a standard "waterfall" approach with discrete stages and deliverables clearly identified within a standard framework.

The organisation within the casestudy has recognised for some time, that reliance on paper-based interim products throughout the development in the form of graphic and narrative documentation to demonstrate the system development is an inadequate medium often resulting in misinterpretation. They have invested in I.E. (Information Engineering) a set of casetools to assist with their development process. The use of this software engineering environment provides tools which help with some of the tedious chores inherent in software development (such as redrawing design diagrams, finding documentation etc). The impact of I.E. is to reduce the development time and increase documentation and standardisation of the new systems. The basic [5] factors of all packages is that they provide a structure for system investigations, data and process definition and an automatic documentation process. Within the organisation under study, the

structured development framework within I.E. does not map directly onto their waterfall model. This circumstance is currently under review.

The casestudy organisation's business operation revolves around the provision of information to clients. This circumstance complicated the implementation I.E. as the tool is driven by an organisation's vision not that of a multiple of clients. I.E. has a knowledge base at its core.

There is a clear statement within the development framework that a different staff member may be appointed for each stage of the development, and that these key individuals can be drawn from either the business or information technology functions. In addition stages of the software development process may be combined for small developments or those primarily of an enhancement or maintenance nature. This is discussed further under the classification of software developments.

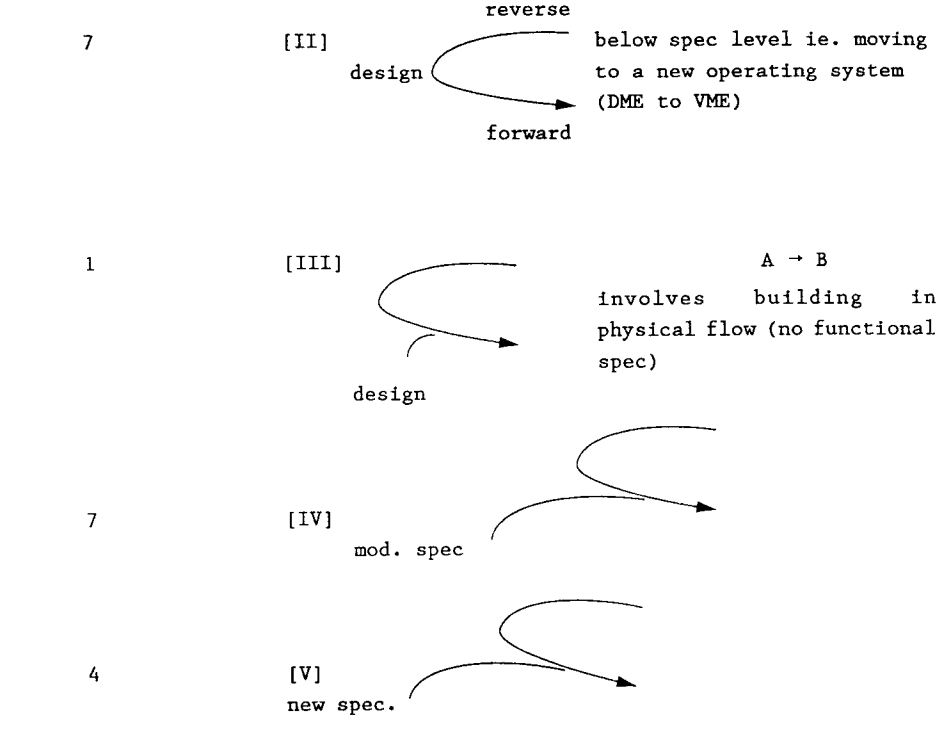
Company A: classification of software developments

Within the research program, the software developments in the organisation were identified over an eighteen month period. This project list included all developments which were initiated, or underway during the past twelve months, or projected as starting up within the upcoming six month period. Very many of the developments involve the enhancement through the addition of business functions to existing software systems. All software development work requiring less than two weeks man-effort was considered to be system support/maintenance and without the scope of the analysis. The groupings of project types is illustrated in figure 3. This summary was taken over an eighteen month period 1990/91.

The company generally undertakes less than twenty distinct software development projects within a year. However, in any one individual year this figure may be much distorted by the emergence of one larger project (eg. system conversion due to change of hardware or operating system). The company has increased the numbers of its systems professional staff from twenty to fifty over the past three years so yet again we are studying an organisation within a period of growth and increased dependency on Information Technology. As can be seen from Figure 3, almost all the project types identified include a sizeable element of reverse-engineering; moving from the systems software back within its development process to the system design or functional specification stage. Most often this activity will be partial in so much that a key area of the software will be identified rather than all of the system reverse engineered.

All project types except type II, which excludes the requirement specification stage, involve all stages of the systems life cycle. However, particularly for smaller projects, some stages are combined. Most often the last two stages, the installation and the operation are combined and managed as one stage within the system development life cycle.

The systems staff within the organisation involved within the research program were surprised by the project classification numbers, and suggested that perhaps software development activity within the department had changed in nature in very recent years. The profile is somewhat distorted by the large volume of resource allocated to an upgrade of the operating systems. A brief description of each of the project types is given;



[IV] and [V] come in two flavours:large and small. Small scale developments go to the small developments working party.

Figure 3. Project types within the casestudy organisation.

Project type I (forward software engineering) a standard new software development, where all the software is written to meet a specific system requirement. This is totally forward engineering of the classical type following the waterfall development lifecycle very closely.

Project type II (eg moving to a new operating system) this type of software project is geared towards replicating a system that is currently in operation on a different hardware or software support base. Thus the intention is to retain without alteration the status quo in terms of system functionality

Project type III (addition of a data stream) this is a straightforward enhancement of current software to handle additional data capture, manipulation and reporting. This project classification does not necessitate modification to the functionality of the existing system.

Project type IV (systems modification and enhancement) this is the same as classification III, with the added complexity of modifications being required to the existing software. This has a significant impact of software testing time allocation, and may have a detrimental affect on the original software decision in terms of quality.

Project type V (new specification) this is a new software function developed on top of an existing software system. The complexity of the system interface has the major impact on the project complexity and resultant quality issues.

COMPANY A: SUMMARY

Nearly all the software projects include a substantial element of 'reverse engineering' and are under six months in duration. A waterfall software development lifecycle approach is taken, with quality control built into the project plan, and stages of the lifecycle combined or eliminated to suit the individual project requirement. Very many of the 'best practice' approaches as highlighted by BS 5750 standards are in place, as is the concept of a quality culture.

The company does perceive a need to measure quality; in particular the impact of enhancement or modification projects (types C,D,E) on existing software is of concern. Further research work is on going with the company in the area of both project and quality metrics. However, the selection of suitable software development for further in depth casestudy seems to be limited to the larger projects. A recurring theme throughout the whole research program suggests that small projects are manageable and controllable, or not worthy of the cost overhead associated with any imposition of formalised quality control.

RESEARCH SUMMARY AND CONCLUSIONS

None of the casestudy organisations felt external accredited quality assurance was appropriate for their software development process or products. All had in place some mechanism for both quality management and quality control. The single biggest influence on how formalised these processes are the size of the software development department, project team, and software product. The significance of size seems to far outweigh any influence due to software complexity, or project risk analysis consideration. The level of quality control in place seemed to depend on the maturity and experience of individual staff members assigned to the software development. Surprisingly, the system

development methods employed did not seem the key issue. The casestudy work has identified the wide variety and mixture of approaches employed by organisations all within the financial service sector, towards the production of quality software. Further work is required to formulate the factors contributing to the software quality process, and to survey a wide spectrum of organisations within this sector.

Further research

Further research is underway in the area of software project and quality metrics, within Company A. This work is aimed at the full software development lifecycle, collecting metrics within a case study project. Most of the metric models currently available are only applicable to large scale forward engineering type software developments. However, within company A this type of project is not the normal, and model refinement is likely to be required.

REFERENCES

1 Benjamin, R., Rockart, J., Scott, M., Morton, Wyman, J., Information Technology: A Strategic Opportunity, *Sloan Management Review*, Vol.25, No.3 (1984).

2 Holloway, H., Information Technology and Company Policy: 4-Banking, *Journal of Information Technology*, Vol.3, No.4 (1988).

3 Rothwell, M., Jowett, P., Rivalry In Retail Financial Services, Chapter 2, *Retail Banking: The Big Four*, Page 19 MacMillan Press 1988.

4 Macro, A., Buxton, J., *The Craft of Software Engineering*, Addison Wesley 1987.

5 Ould, M., Software Development Under DEC STAN 00-55: A Guide, *Information and Software Technology*, Volume 32, Number 3, 1990.

6 Daily, K., "Quality Management Systems in Software Engineering - A Take-Up Study", *Proceedings of the Software Quality Workshop*, Dundee, June 1990.

7 Daily, K., "Managing with Quality Systems BS 5750 & ISO 9000", *NCC the National Centre for Information Technology*, April 1990.