DUNEDIN NEW ZEALAND

Industry Practices in Project Management for Multimedia Information Systems

Stephen MacDonell
Tim Fletcher

The Information Science Discussion Paper Series

Number 99/15
June 1999
ISSN 1177-455X

University of Otago

Department of Information Science

The Department of Information Science is one of six departments that make up the Division of Commerce at the University of Otago. The department offers courses of study leading to a major in Information Science within the BCom, BA and BSc degrees. In addition to undergraduate teaching, the department is also strongly involved in postgraduate research programmes leading to MCom, MA, MSc and PhD degrees. Research projects in spatial information processing, connectionist-based information systems, software engineering and software development, information engineering and database, software metrics, distributed information systems, multimedia information systems and information systems security are particularly well supported.

The views expressed in this paper are not necessarily those of the department as a whole. The accuracy of the information presented in this paper is the sole responsibility of the authors.

Correspondence

This paper represents work to date and may not necessarily form the basis for the authors' final conclusions relating to this topic. It is likely, however, that the paper will appear in some form in a journal or in conference proceedings in the near future. The authors would be pleased to receive correspondence in connection with any of the issues raised in this paper, or for subsequent publication details. Please write directly to the authors at the address provided below. (Details of final journal/conference publication venues for these papers are also provided on the Department's publications web pages: http://divcom.otago.ac.nz:800/COM/INFOSCI/Publctns/home.htm). Any other correspondence concerning the Series should be sent to the DPS Coordinator.

Department of Information Science
University of Otago
P O Box 56
Dunedin
NEW ZEALAND

Fax: +64 3 479 8311
email: dps@infoscience.otago.ac.nz
www: http://divcom.otago.ac.nz:800/COM/INFOSCI/

Industry practices in project management for multimedia information systems[1]

Stephen G. MacDonell and Tim Fletcher

Department of Information Science
University of Otago, New Zealand

Corresponding author: Dr Stephen G. MacDonell
Department of Information Science
University of Otago, P O Box 56, Dunedin, New Zealand
Tel: +64 3 479 8142 Fax: +64 3 479 8311
email: stevemac@infoscience.otago.ac.nz

Abstract

This paper describes ongoing research directed at formulating a set of appropriate measures for assessing and ultimately predicting effort requirements for multimedia systems development. Whilst significant advances have been made in the determination of measures for both transaction-based and process-intensive systems, very little work has been undertaken in relation to measures for multimedia systems. A small preliminary empirical study is reviewed as a precursor to a more exploratory investigation of the factors that are considered by industry to be influential in determining development effort. This work incorporates the development and use of a goal-based framework to assist the measure selection process from a literature basis, followed by an industry questionnaire. The results provide a number of preliminary but nevertheless useful insights into contemporary project management practices with respect to multimedia systems.

1. Introduction

With continuing advances in hardware and software technology, the feasibility of widespread multimedia use has reached a stage where almost all system developments can (and perhaps should) incorporate a variety of media components. In transaction-oriented systems, however, this generally extends only to the inclusion of graphics e.g. pictures of employees, or products for sale. As a result there remains a separate class of systems that is considered to be 'multimedia', used widely in education, entertainment and 'infotainment' (the interaction of the two prior domains), and incorporating some combination of sounds, animations, video clips, virtual reality and/or high quality graphics that, when effectively integrated, provide a semantically rich information environment. Whereas substantial research effort has been expended in measuring various aspects of traditional transaction processing and process

[1] A shorter version of this paper appeared in the *Proceedings of the Metrics'98 Symposium*, IEEE CS Press

control systems, little work has been performed in determining and evaluating measures for project management in relation to multimedia systems.

The question may reasonably be asked – is there a need for yet more measures? Could existing measures be used to assess aspects of multimedia systems? In our view there are a number of reasons why new measures are needed:

- *The software process employed in multimedia development is fundamentally different.*
  One of the most significant phases in multimedia systems development is that of content (or media) creation. It is at this point that the video clips are planned, recorded, edited and transformed for digital use; the graphics are sourced or painstakingly created and refined, often using subjective assessments of quality; the sounds are recorded, the voice-overs are produced; and the animations are developed frame-by-frame. Later in the process, all of these media components must be integrated, or 'authored', into a coherent, robust and temporally valid system. There may well be a need for storyboarding early in the development of multimedia systems, in order to effectively specify system content and navigation. These tasks have no directly corresponding activities in the development of transaction-centred or process control systems. Measurement models available from these domains are therefore insufficient to take into consideration these non-trivial multimedia-specific process steps.

- *The tools and techniques employed are specific to multimedia development.*
  Specialised content-oriented equipment and high-level visual authoring environments are widely employed in the creation and composition of multimedia systems, creating a different effort pattern than that found in systems development where 'standard' programming tools only are required, and about which we as software engineers know a great deal more. Moreover, whereas existing measures commonly have a basis in some form of design or specification notation (e.g. data flow diagrams, data models), these are seldom used in the development of multimedia systems.

- *Project management for multimedia systems may have a significantly different emphasis.*
  Personnel charged with the responsibility of managing software projects in the transaction processing domain have often themselves come through the developer ranks. As a result their focus is on issues such as lines of code and module error density, and they may be accustomed to using techniques such as COCOMO [1] or function point analysis [2] in terms of determining development effort and costs. Multimedia system project managers, however, are likely to come from a more diverse background – perhaps from graphic design, television or even film production. The models, priorities and assumptions that they bring to the task of project management may therefore be substantially different, and potentially more effective.

- *Multimedia development project teams are largely cross-disciplinary.*
  Related to the previous point, the personnel who make up multimedia development project teams may have substantially different skills and backgrounds, more so than in teams brought together to construct traditional business or scientific systems [3]. The fundamental building block of a multimedia system is the media component whose production could well require the skills of graphic designers, video producers and sound technicians, as well as those of software developers. Managing such a diverse team may be even more difficult than managing a more 'traditional' group of systems analysts and

programmers; as part of this management process, predicting, co-ordinating, monitoring and controlling the team's development effort may be equally more difficult.

- *The inherently different nature of multimedia systems.*
  Games aside, the role of many multimedia systems is one of 'information provider', rather than 'data processor'. In these systems there is little need for complex algorithmic processing, as one might find in sensor-control systems, or for robust database integrity controls, as encountered in transaction systems. Most existing effort assessment and estimation models, however, assume one or other of these situations. At the very least this would seem to demand a modified approach to effort measurement.

It is asserted here that, given the conditions just stated, the direct application of existing models and measures may not meet management needs for effort assessment and prediction in the multimedia systems domain. A new approach would seem to be warranted.

The remainder of this paper is arranged as follows: the next section describes some of the work undertaken to date in formulating measures for multimedia systems; this is followed by the summarised description of a small empirical case study that aimed to test for correlations between product and process measures in a set of student multimedia projects; the outcomes of this study encouraged further exploration of factors – this has led to the development of a structured measurement framework, a pilot study and a questionnaire, all described in Section 4; industry insights are presented and discussed in Section 5; finally the paper is summarised, along with a description of the next steps in our research.

2. Contemporary approaches to multimedia systems effort measurement

Few studies have addressed effort assessment specifically in the multimedia domain. This is due in part to the relatively recent expansion of multimedia into mainstream systems development, and to the corresponding dominance of text-based (i.e single-media) systems in large environments, where project control is perceived to be most important. In these situations, effort assessment and estimation have been based largely on measures of system size or functionality. Boehm's Constructive Cost Model (COCOMO) [1] uses estimated lines of code (LOC) in a size-based effort estimation approach, with adjustments for a number of 'cost drivers' related to technology, personnel, process and product. Marshall et al. [4] have created a version of the COCOMO technique for computer-aided learning (CAL) courseware, where the major inputs are course delivery time and a set of cost drivers under four classes: course difficulty, interactivity, development environment and subject expertise. Although some aspects of the adapted model seem intuitively useful, such as ranking the impact of content complexity on development effort, the model is specific to CAL-based material and as such, its wider applicability to other types of multimedia systems development is unclear. The inclusion of more than twenty cost drivers, some of which must be assessed subjectively, is also a significant disadvantage of such an approach. Furthermore, lines of code are largely irrelevant in terms of many multimedia products, as the development environments used in their creation are centred on the integration of media components as opposed to the production of program statements. The general suitability of LOC-based models in the multimedia domain is therefore doubtful.

The second major approach to effort assessment and prediction in systems other than multimedia is centred on function-based measures. Function point analysis (FPA) [2]

quantifies the weighted contribution of system inputs, outputs, enquiries and files to system scope and complexity, with subsequent adjustment based on attributes of the development and operating environments e.g. the need to distribute the system, the required extent of user interaction. Gao and Lo [5] suggest that this method will not be effective for multimedia systems without modification:

- whilst multimedia systems do make use of large databases these are not actually maintained by the system in the sense that they are manipulated in transaction-oriented systems (the focus of FPA);

- extensive component reuse is widely practised in the development of multimedia, an activity still sporadic in transaction systems development (and without comparison in terms of media content objects) – FPA fails to take account of this;

- the output forms for multimedia systems, incorporating sound, video and graphics, are far more complex than the largely textual output of standard business systems (for which FPA was developed).

As a result, Gao and Lo [5] have proposed a modified FPA method. Their approach takes into account the non-transaction orientation of the data manipulations performed and makes some adjustment for the use of pre-built components. However, it retains the equal weighting scheme for all forms of system output in spite of the contention that outputs such as video and animation may have significantly greater impact on systems development effort than pure text output.

A small number of other non-algorithmic 'models' have also been proposed to represent the relationship between system attributes and development effort in the multimedia application domain. For example, Marshall et al. [4] report the use of the ratio of development time to delivery time for estimating courseware production effort. According to Merrill et al. [6], a simple CAL program may require 100 hours development time for one hour of course delivery time, which may rise to 800 development hours per hour of delivery time if extensive use of media content is required [7]. Such ratios, however, are wholly anecdotal so their general applicability is uncertain [8]. Very simple system size measures, such as the number of screen or media components, have also been suggested as indicators of system scope. Whilst deserving of further investigation, some form of screen or component 'complexity' adjustment would seem to be essential if accurate predictions are to be obtained (although this is easier said than done – see Sections 3 and 4).

Other candidate measures may be borrowed from production domains that have similarities with multimedia. Cost per digital minute is often used to determine presentation value with a heavy video content. This measure has its origins in broadcast television [9]. The development time to delivery time ratio is similar to the person-hours to program hours ratio [10] which also has its origins in television. Although these benchmarks are again somewhat anecdotal they should not be totally discounted. Given the major differences between multimedia and transaction-oriented systems development, such approaches may play some part in helping us to determine the factors that do indeed influence development effort.

3. A preliminary empirical investigation

A small-scale experiment was undertaken with student subjects to explore the effectiveness of very simple size-based product measures in terms of their relationship with associated component and system development effort (see [11] for a complete report of this study). The aim of the work was to develop an algorithmic mapping to effort using product characteristics more appropriate to multimedia systems than lines of code, external files and the like (as used in existing models). The study was based on the assertion that multimedia systems development effort is a function of (i) building the system content and (ii) authoring the system. Each of these tasks was to be evaluated in terms of the components manipulated and the activities carried out:

(i) building the system content - for each media component created, the following data items were to be recorded: filename, media type (graphic, audio, video, animation, photograph, scan), status (original or pre-existing), creation effort (for original media), digitising effort (for scans, video and audio), editing effort, and component duration (for temporal media i.e. animation, sound and video). The assumption underlying this collection scheme was that each media form might have a different impact on development effort.

(ii) authoring the system - for the process of authoring, the screen name and authoring effort were to be collected for each screen. An inspection of each system was also to be conducted to ascertain the 'complexity' of each piece of media and each screen. The data to be collected for each screen were: the number of objects on the screen (including sounds), the number of links between that screen and other screens, the number of events on the screen, and the average number of actions per event. Procedures that respond to a mouse being clicked, or any other scripted actions, were considered as events. The associated task is normally a generic activity to be performed; typically most link buttons contain only two actions for the click event: play 'click' sound and go to another screen. The media complexity was to be based around graphics data: the number of objects on the component, whether it had been reused elsewhere in the project, and the form it took (button, toolbar, screen, background, component e.g. part of the foreground or a source in an animation sequence). This approach was based on the assumptions that a screen that incorporates a greater number of objects and events would take proportionally greater effort to develop, as would a more complex component.

Systems were developed by five 4th-year student groups as part of a joint course delivered by design studies and information science departments. The systems were delivered in two stages: prototype and final deliverable. As expected the information science students took charge when it came to the programming and technical issues whilst the design students generally concentrated on content and interface development.

A total of just 45 observations were recorded in relation to media component development. Data analysis revealed no correlation between development effort and either of the component variables – media type (graphic, audio, video, animation, photograph, scan) and media status (original or pre-existing). The screen authoring observations were more promising, but analysis was confounded by very small data sets (just 7 and 11 observations respectively). For one of the two authoring environments, correlations between development

effort and the number of onscreen objects and between development effort and the number of events were evident. As the data sets were so small and were related to student projects, however, these results cannot be extrapolated to form any generalisable conclusions. As a result we gained only minimal preliminary insights into effort modelling for multimedia systems.

4. Comprehensive assessment of factors

The empirical investigation just described highlighted a number of deficiencies in our approach. First, the selection of 'influential' characteristics had been based on our own considerations rather than on any systematic industry-driven determination of important attributes. Second, the data collection process needed to be more exploratory, as there was little literature evidence to suggest any widely accepted existing set of measures. Third, the use of student projects was not going to deliver results and conclusions that could be said to be indicative of multimedia systems development in industry. Therefore our aim shifted, to be focused on the development of a literature- and industry-based measurement framework for multimedia systems effort modelling followed by wider industry verification. Only once this has been achieved can we reasonably hope to undertake realistic and relevant empirical investigations.

4.1 Framework development

Our over-riding aim was to determine and measure the factors that influenced multimedia systems development effort, with a view to then estimating effort required for new developments. In an attempt to select appropriate measures in a more structured and rational way, the goal/question/metric (GQM) approach [12] was used. This is a decomposition-based method whose goal is to produce a hierarchy of measurement program levels. At the top, the program's goal is clearly stated. Questions are then asked that, when answered, will enable the goal to be achieved. Measures are then specified at the lowest level of the hierarchy – measures that are both sufficient and necessary to enable the questions to be answered. Our goal at this stage was (and is) to determine system and component characteristics that are considered by industry to be influential in affecting multimedia systems development effort. Given that this goal is largely exploratory, questions have been included (mainly from a basis in the literature) that enable the consideration of both empirical and anecdotal factors within the framework. Ultimately it may well be that only some of these factors will be used in any algorithmic model that is developed. The actual questions asked have come from several sources:

- texts on project management for multimedia information systems (e.g. [3])
- texts on project management from the film and video domains (e.g. [9])
- comments on important considerations from informal phone calls made to developers
- software metrics texts in which system size tends to be the dominant factor (e.g. [1]).

Part of the framework is shown in Figure 1.

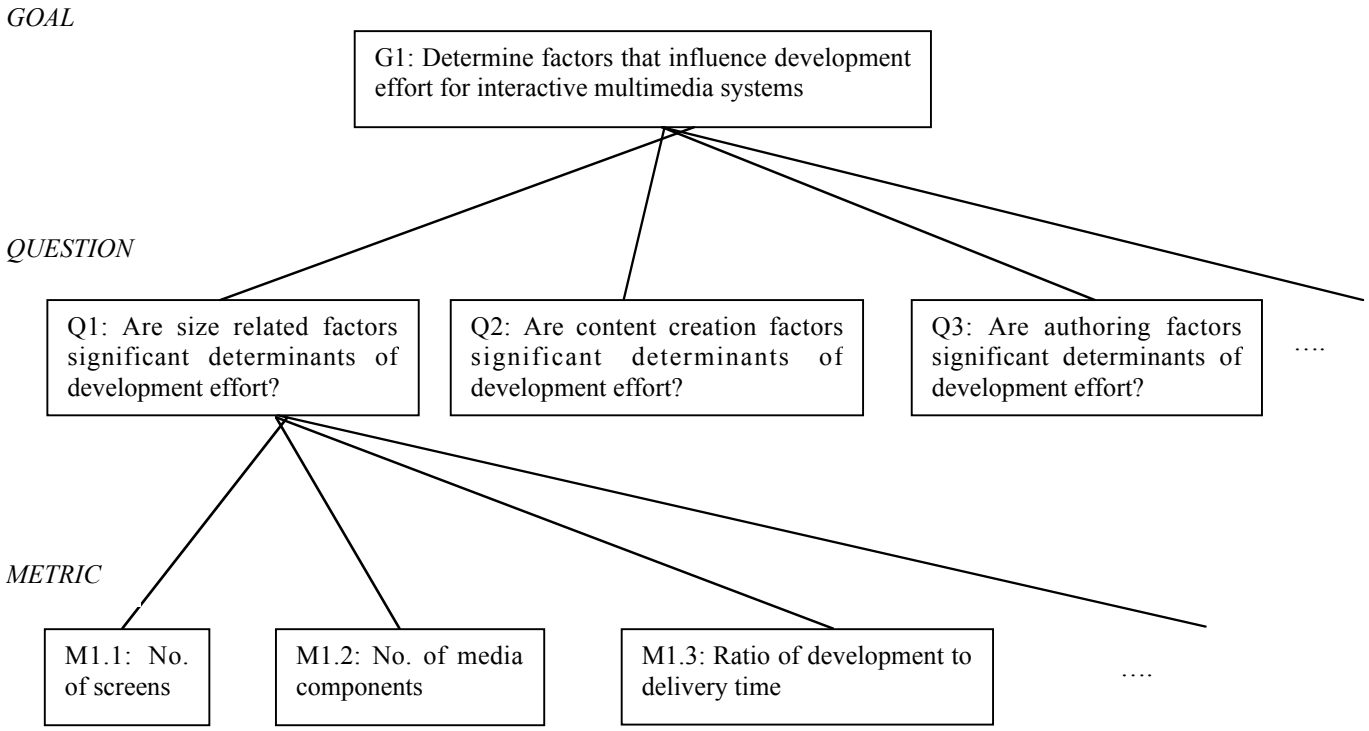

Figure 1: Part of the GQM framework for multimedia development effort

Thus a wide variety of factors have been incorporated into the overall framework, in order to make the approach as comprehensive as possible. As data is collected the framework will be refined to only include factors that are indeed found to have some significant impact on development effort. Some of the issues currently considered in the framework are now discussed.

Development tool issues – Generally the nature of a project will determine which development tool or environment is used. Whilst authoring tools are the most common, other tools are used, particularly where very complex processing is required. In these cases visual 3GLs, 'traditional' 3GLs, or a 3GL/authoring tool combination is often employed. Authoring tools generally offer visually based high productivity development under an event-driven development model. As such these environments are either codeless or utilise a very high-level scripting language. In addition, authoring tools provide features useful to multimedia developers such as media databases, built-in system optimisation tools and intelligent palette handling. Features of this nature can be significant in terms of development effort.

Delivery platform issues – As a rule the (often client-imposed) choice of target platform dictates the media format and optimisation required. For example, all content developed for use on CD-I must conform to either PAL or NTSC standard screen sizes and specifications, since CD-I systems use a television as the display unit. The issue of file formats is also important in terms of the need for conversion and the effort required for this task. Any analog format such as S-VHS video or DAT audio must be converted to digital format in order to be usable in a computer-based system.

Content development issues – Media content development is generally held to be the most labour-intensive part of multimedia systems development. Production effort may vary

according to the media mix (e.g. whether there is extensive use of the more complex media forms of video and sound), the number of media components and the 'complexity' of those components. The impact of each media type on overall content development effort therefore needs to be assessed. If the media component does not already exist, it must be created and this obviously adds to the content development effort. If a component does need to be constructed then its complexity will have a bearing on the associated construction effort – what actually *determines* media component complexity is, however, difficult to say. In the preliminary empirical study discussed above, screen complexity was said to be a function of the number of components and links it contained. How do we measure the complexity of a graphic, however? We are currently investigating graphics and visualisation research in order to assist us with such questions. Assessment difficulties may be further exacerbated by the fact that artistic judgement plays a significant role in the decision as to when a component is satisfactorily completed. The desire for aesthetic quality may mean that accurate prediction of development effort is made more difficult.

Organisational capability issues – Organisational factors are unlikely to directly affect development of individual projects but they will influence the general manner in which systems are developed. At a fundamental level the size of an organisation (in terms of personnel) may be of some interest. The breakdown of personnel across functional areas may also indicate a pattern of effort requirement. The media development technology employed by the organisation is also likely to have a significant impact on the ease with which they are able to effectively build multimedia systems.

Personnel issues – As successful multimedia development requires staff with a wide variety of skills, personnel often come from a more diverse background than just that of computing. It may therefore be useful to know the background of staff since this may affect how they develop systems, solve problems, plan, cost and estimate projects. England and Finney [3] state that the core members of a multimedia system development team are a producer, one or more programmers and one or more graphic artists. An extended team may consist of various other specialists: video and sound personnel, administrative support, script writers, instructional designers and subject matter experts. The effort of all of these personnel must be considered in any model that attempts to be comprehensive in its assessment.

4.2 Pilot study

In order to perform some preliminary verification of the framework, a small pilot study was undertaken. Structured interviews were conducted with three interactive multimedia development organisations. In terms of the recording of development effort data, only one of the three organisations made use of standard timesheets. In this case they were then able to employ a very simple effort estimation model for a system including *n* screens: *((n) screens) x (mean screen creation time)*. In the other two organisations effort estimation was performed based only on project managers' experience. Two of the three organisations built heavily graphical systems and, in their view, content creation was the major activity within development, and content specialists were the dominant personnel in their development teams. Within the three organisations few personnel came from a computing background – indeed the common backgrounds were in film or music.

Results also suggested that the type of project almost entirely determined the development environment and the workload among various development areas. Two of the three

organisations built reasonably simple (in terms of programming and functionality) content-centred systems delivered on kiosks, CD-ROMs and the Internet, and these were developed primarily using authoring tools. One organisation built more complex systems requiring custom-written data acquisition and data processing modules. This organisation used a visual 3GL environment because it provided rapid application development features with a powerful language. Only this organisation found programming to be their major development activity. Component reuse was practised extensively by all three organisations.

4.3 Industry Questionnaire

Results of the pilot study were combined with the components of the GQM framework to enable the development of a postal survey instrument. Again, the focus was on determining the factors that significantly affect development effort for multimedia systems. It is our hope that the eventual outcome of this study will be a set of industry-verified factors in terms of their impacting on systems development effort. Valid responses were received from more than 150 of the 200 organisations contacted, but just 22 of these considered themselves to be multimedia systems developers. (This apparently poor rate is due to the fact that the list of companies used for the mail-out inadvertently contained a high number of user group members who were not actually software developers of any form.) Whilst 22 is not a large number from which to draw conclusions, some trends were evident. These are now discussed.

Personnel in the respondents' organisations were drawn from a variety of backgrounds. Most common were those with a graphics design or arts background, employed at sixteen of the 22 organisations. Thirteen organisations employed personnel with a computing background, five employed those with television or film experience, and four employed personnel with music or audio engineering backgrounds.

Just ten of the twenty-two respondents confirmed that they used some form of formal methodology in their development of multimedia. Of these, four mentioned the use of storyboards and/or paper drafts and five described the incorporation of a content acquisition/development phase. Only one reported the use of a typical business-oriented life cycle type process.

In relation to the tracking of personnel effort, timesheets were used by fifteen of the respondents, most recording effort expended on development activities at the system level. A small number (five) recorded the screen and/or component name being worked on, and eight recorded the module name.

The survey recipients were then asked to specify the techniques they used to estimate system development effort. Nineteen answered that experience of similar past projects was used, the number of system screens was used by eight of the respondents, the amount of digital video was employed as a driver of effort by seven organisations, five respondents used a ratio of development time to delivery time and just one used a cost per digital minute measure. Of the five who claimed use of a development time to delivery time ratio, only two actually supplied the ratio employed: one at 1000:1, the other at 120:1. Two respondents remarked that they were still coming to terms with estimation.

Those who cited the use of experience of similar past projects in estimation were then asked what made projects similar. Fifteen of the group agreed that both project size and type were

important, fourteen of the 22 suggested that a similar mix of media elements was influential, and eight considered that a common target delivery platform had an impact on project similarity.

Component reuse was practised in an *ad hoc* sense by eighteen of the 22 respondents, although the use of a formally managed approach to reuse was not widespread (five sites). More importantly for this discussion, only thirteen of the sites made any adjustment to their effort estimates as a result of planned reuse. When asked to specify how this adjustment was performed, respondents provided a variety of answers describing 'unofficial' changes - it was clear that there was no systematic process in place for adjustment for reuse in any of the sites.

Twenty of the respondents confirmed that the selected development environment had a significant impact on development effort, largely in relation to familiarity with tools and methods. Recipients were then asked to rank (from 1 to 9) a list of nine factors in terms of how much they influenced development effort. The results, as summarised in Table 1, indicated that project size and experience of personnel ranked equal first, while testing requirements and platform optimisation ranked lowest of the factors provided. It is also interesting to note that respondents were generally in close agreement on the degree of influence of some of the factors (e.g. experience of personnel and platform optimisation) with low standard deviation and range values, whilst for others (particularly client's expected quality and testing requirements) there was much greater spread in the rankings given. Moreover, some factors produced data distributions with significant outlier rank values – these are shown in Figures 2 through 4.

Table 1: Ranking of factors that influence multimedia systems development effort

Factor	Mean rank	Median rank	Respondents	Std Deviation	Range
Experience of personnel	2.7	2.0	18	1.4	4
Size of project	2.7	2.0	18	2.0	6
Amount of original media	3.3	3.0	17	1.7	6
Programming complexity	3.7	3.0	17	1.6	5
Client's expected quality	5.1	5.0	18	2.4	8
Need for outside services	6.4	6.0	17	1.9	6
Choice of platform	6.3	6.5	16	1.8	6
Testing requirements	7.0	7.0	17	2.0	7
Platform optimisation	8.5	9.0	17	0.7	2

Eight factors considered to be associated with the difficulty of multimedia authoring and/or programming were listed and respondents were asked to rank these from 1 to 8. Once more, the experience of staff rated most significantly, and platform-related issues least influential, with the latter resulting in close agreement in rankings from the respondents.

Table 2: Ranking of factors that influence authoring or programming difficulty

Factor	Mean rank	Median rank	Respondents	Std Deviation	Range
Experience of staff	3.1	2.0	18	2.1	7
Type of tool used	3.5	3.0	17	2.1	7
Need for a custom solution	3.8	3.0	16	2.4	7
Special project requirements	3.8	3.0	17	2.4	7
Need for database handling	4.3	5.0	16	1.8	6
Other platform requirements	4.4	4.0	17	2.3	7
Need for a search engine	5.4	6.0	16	1.6	4
Platform optimisation needs	7.0	7.5	15	1.3	3

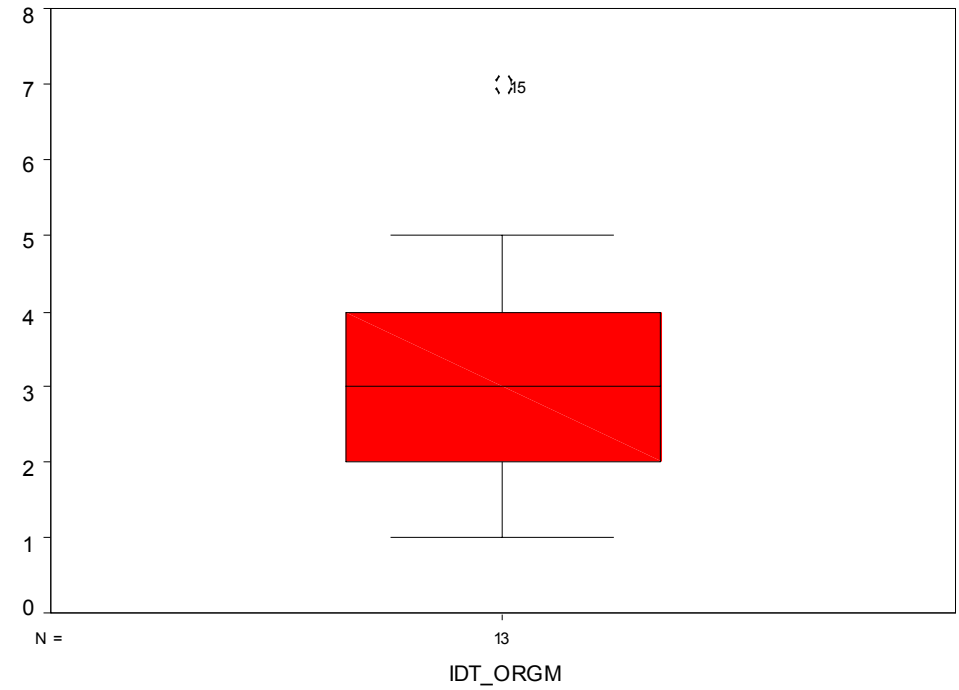

Figure 2: Boxplot distribution of rankings for amount of original media

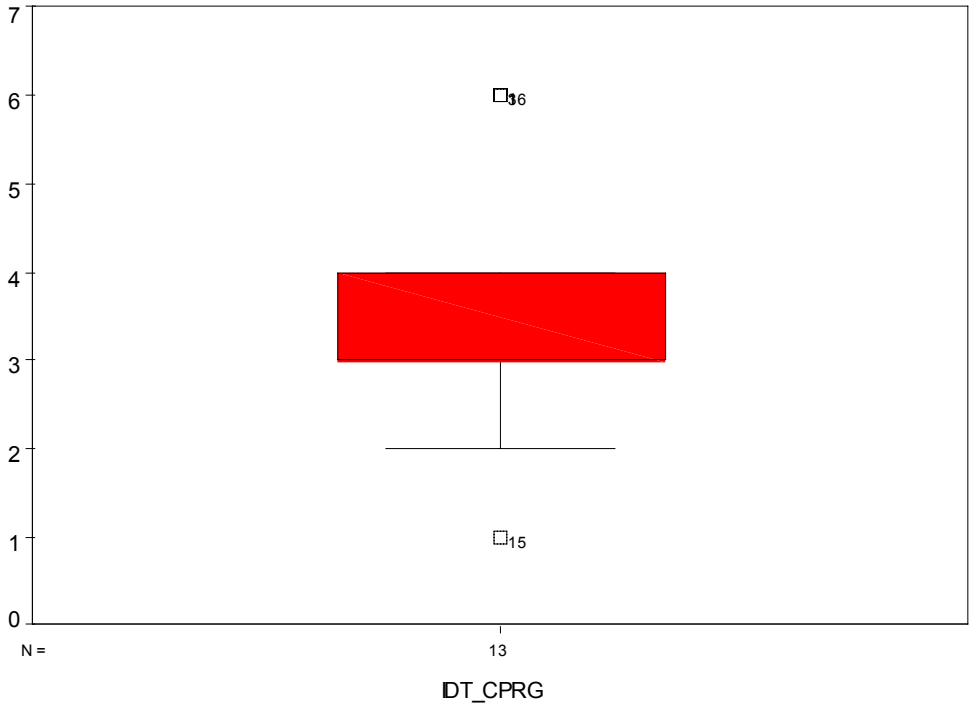

Figure 3: Boxplot distribution of rankings for programming complexity

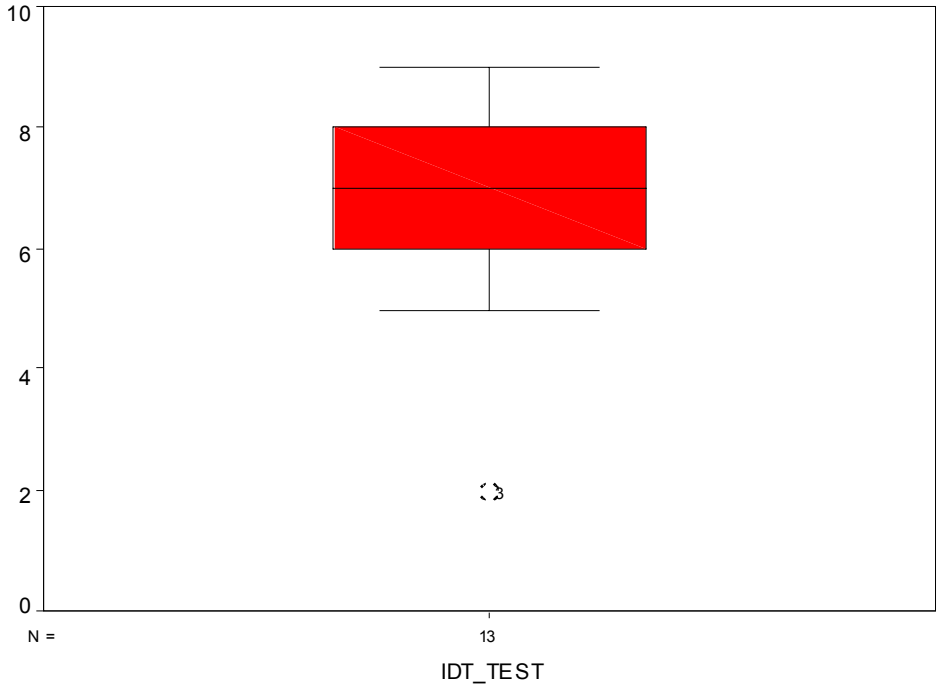

Figure 4: Boxplot distribution of rankings for testing requirements

When asked how their background helped them estimate project costs, one (fortunate) respondent mentioned a 'natural ability' for planning and estimation. More common answers included the application of prior experience, the use of a "hit and miss" strategy, a sense that no two projects are the same so estimation based on history is immaterial, and the broader feeling that background hadn't helped at all - "this is a totally new environment and costings have to be developed from scratch"; "very little estimation, costs tend to be high". One respondent in fact remarked that his/her background made estimation *more* difficult.

A more general question on the difficulties of project estimation elicited eight responses relating to a lack of realism in clients' perceptions of the ease of accommodating requirement changes and technology capability. Six respondents commented that the environments and tools used were still so new that accurate estimation was not yet possible. Other difficulties mentioned were other unexpected changes (not related to client needs) and external constraints, such as having to wait for days without rain for filming.

5. Discussion

Whilst the results above are preliminary, some insights can be gained. The fact that nearly half the respondents indicated no use of a formal methodology may suggest that, in terms of process maturity, they may be considered to be chaotic in their approach. When coupled with the widespread employment of personnel with non-computing backgrounds it is not entirely unexpected that product and process measurement has received only scant attention. In terms of methods used in estimation, none of the 'traditional' software metrics approaches (e.g.

COCOMO, function point analysis) were mentioned; rather, the clearly favoured method at present is to use experience of previous projects, particularly in terms of project size and type. Moreover, the tracking of effort is generally performed at a very high (normally system) level. This again fits the model of relatively immature project management practices. Quantitative methods derived from film and television are used in a very small number of sites.

As in many development domains, staff experience and project size are perceived to be the most significant factors in determining development effort. The most influential media-related factor, the amount of original media, ranked third of the nine factors listed. Issues related to the delivery platform, thought to be potentially significant in our preliminary investigation, appear to in fact be relatively unimportant.

The general comments obtained from the respondents further reinforce the currently *ad hoc* approach of most to estimation. Although some are attempting to improve their practices there is still significant uncertainty as to how this can be achieved, at least on the basis of the responses of the 22 development organisations. Whilst the same may be said in regard to the management of other development domains (e.g. in the real-time system domain) the *extent* of process immaturity in multimedia systems development seems far greater.

6. Summary and Conclusions

This research is ongoing – as such, it has produced only preliminary outcomes at this stage. We are encouraged by the positive industry verification of the framework through the pilot study and questionnaire. Further responses should provide greater insight into industry opinion on the factors that are related to multimedia systems development effort. Empirical analyses of the supported relationships should then take place with the aim of producing calibrated algorithmic models incorporating appropriate system factors.

References

1. B. Boehm, *Software Engineering Economics*, Prentice Hall, 1981.

2. A. J. Albrecht and J. E. Gaffney Jr, "Software function, source lines of code, and development effort prediction: a software science validation", *IEEE Transactions on Software Engineering* **9**(6) (1983) 639-648.

3. E. England and A. Finney, *Managing Multimedia*, Addison-Wesley, 1996.

4. I. M. Marshall, W. B. Samson. and P. I. Dugard, "A proposed framework for predicting the development effort of multimedia courseware", in *Multimedia/Hypermedia in Open Distributed Environments,* Eds. F. Kappe and W. Herzner, Springer-Verlag, 1994, pp. 161-180.

5. X. Gao and B. Lo, "A modified function point method for CAL systems with respect to software cost estimation", *Proceedings of the Software Engineering Conference SE:E&P'96*, 1996, pp. 125-132.

6. M. D. Merrill, Z. Li and M. K. Jones, "Limitations of first generation instructional design", *Educational Technology* **30**(1) (1991) 7-11.

7. P. Beautement, "Review of interactive video systems and their possible application to training in the 90's", *Interactive Learning International* **7** (1991) 45-54.
8. I. M. Marshall, W. B. Samson, P. I. Dugard and G. R. Lund, "The mythical courseware development to delivery time ratio", *Computers in Education* **25**(3) (1995) 113-122.

9. C. Fraser, *The Production Assistant's Survival Guide*, BBC Television Training, 1994.

10. R. Bretz, *Handbook for Producing Educational and Public-Access Programs for Cable Television*, Rand Corporation, 1976.

11. T. Fletcher, S. G. MacDonell and W. B. L. Wong, "Early experiences in measuring multimedia systems development effort", in *Multimedia Technology and Applications (Proc. ICMTM '96),* Ed. V. W. S. Chow, Springer, 1997, pp. 211-220.

12. V. R. Basili and H. D. Rombach, "The TAME Project: towards improvement-oriented software environments", *IEEE Transactions on Software Engineering* **14**(6) (1988) 758-773.