

Association for Information Systems AIS Electronic Library (AISeL)

ICIS 2002 Proceedings

International Conference on Information Systems
(ICIS)

December 2002

A Study of User Involvement in Packaged Software Selection

Debra Howcroft
University of Salford

Ben Light
University of Salford

Follow this and additional works at: <http://aisel.aisnet.org/icis2002>

Recommended Citation

Howcroft, Debra and Light, Ben, "A Study of User Involvement in Packaged Software Selection" (2002). *ICIS 2002 Proceedings*. 7.
<http://aisel.aisnet.org/icis2002/7>

This material is brought to you by the International Conference on Information Systems (ICIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ICIS 2002 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

A STUDY OF USER INVOLVEMENT IN PACKAGED SOFTWARE SELECTION

Debra Howcroft
IS Research Centre
University of Salford
Salford, UK
d.a.howcroft@salford.ac.uk

Ben Light
IS Research Centre
University of Salford
Salford, UK
b.light@salford.ac.uk

Abstract

This paper is concerned with the decision-making processes surrounding the adoption of packaged software in organizations. We begin by looking at its increasing utilization and consider some of the strengths and limitations of employing a standardized approach, particularly in relation to its consideration of end-user requirements. We note the highly problematic nature of installing a global standardized product in the local environment. Using a field study concerning the adoption of a customer relationship management package in a small organization, we go on to illustrate the limited amount of end-user involvement in the selection and procurement of the product. We argue that the art of salesmanship by the third party vendor and project team, which focuses on the interests of senior management, ultimately secures the selection and procurement of the software package.

Keywords: Packaged software, user participation, customer relationship management, action research

1 INTRODUCTION

Software development is replete with failures and much has been written about the *software crisis* (Brooks 1987) and the classic problems of late, over budget, poor quality systems that fail to adequately address users' requirements. In an attempt to address this situation, numerous solutions have been proposed, including the use of packaged software. For organizations facing the difficulties associated with embarking on custom development, a dedicated package that offers support for a particular business function seems like an ideal solution.

This paper describes the process of selection and procurement of a software package within a small organization, paying particular attention to issues concerning user involvement and participation. It is widely acknowledged that packaged software embodies implicit business rules. Consequently, it would be expected that the elicitation of organizational requirements would necessitate the involvement of users and, indeed, this is often recommended in packaged software studies (Bingi et al. 1999; Dolmetsch et al. 1998). In contrast, our study highlights the distinct lack of attention to user participation as *salesmanship*, geared to the wishes of senior management, configures the decision-making processes. In the next section, we discuss packaged software in terms of its phenomenal growth since the 1990s, its characteristics, and the role of user participation as compared with more traditional custom approaches. This is followed by details of the research method adopted for the empirical study. Section four considers the field study proper, before leading on to the conclusions of the paper.

2 LITERATURE REVIEW

There is an increasing body of evidence that suggests organizations are shifting from custom to standardized, packaged software¹ development for major applications (Deloitte and Touche 1996). Packaged software is generally sold as a tradable product (Carmel 1997) and can be purchased from a vendor, distributor or store. One major contributor to the growth of packaged software since the early 1990s is enterprise resource planning (ERP) applications, with licence revenues estimated in the region of tens of billions of U.S. dollars (Markus and Tanis 2000). With this in mind, it is worth noting that the costs of developing new software products are large, yet reproducing the additional copies of the product are negligible since the unit costs are minimal (Hanseth 2000). Therefore, the intellectual property that represents the packaged software is generally licensed for use rather than sold outright. The vendor retains ownership of the application and negotiates a licence governing its use with the purchaser (Carmel 1997). As a consequence, developers of packaged software tend to have a product (rather than process) view of development (Quintas 1994). That is, their focus is on developing and shipping a generic product, leaving the other (processual) activities associated with the systems development process such as implementation, system integration, and user acceptance to the purchasing organization to manage. This represents a considerable shift since, historically, user organizations have developed a large proportion of software, often with the IT function mediating between the real world problems of users and the information technology. Essentially, developers have operated in a service or mediating function. This traditional role is now being displaced as increasing numbers of organizations substitute the systems development process with the purchase and installation of standardized software products.

Yet, software development remains a relatively young and immature industry, particularly when compared with other scientific and engineering professions. Still perceived by many as little more than a cottage industry, it is only just beginning to progress into an intermediate maturity stage of commercialization (Carmel 1997). This commercialization and response to market pressures means that the packaged software industry is a highly competitive one. For the producers of packaged software, success is measured according to profitability, favorable product reviews, and market share (Carmel and Sawyer 1998). It is also equated with the achievement and prosperity of the product as opposed to user satisfaction with the process or integration of the new system into the organization. The latter is, to a great extent, the remit of the third-party vendor (often a consultancy group) that mediates between the software producer and the implementing organization. Still, the attraction of packaged software is easy to see. The ability to deliver complex systems in a relatively short period of time while standardizing applications across business functions has immense appeal. Couple this with a track record of success over a large installed base and minimal maintenance costs and it comes as little surprise that the packaged software industry is the fifth largest manufacturing industry in the United States (Carmel and Sawyer 1998).

However, a potential limitation of packaged software is the need for compromises as a result of adopting a product of finite configurability. In order to achieve the alignment of (local) organizational requirements with a (global) software package, the process involves compromises since the required functionality may be only partially met. To achieve greater alignment, the package will probably require modification in order to accommodate any unique requirements (Light 2001; Markus et al. 2000; Soh et al. 2000). Similarly, packaged software has been referred to as 'frozen organizational discourse' (Bowker and Star 1994) and its introduction into organizations may create additional problems in that the product itself is embedded or inscribed with assumptions, values and opinions about patterns of use, the nature of work, and organizational structures (Webster and Williams 1993). The research by Grint and Woolgar (1997) links software design to the construction (or configuration) of the user whereby he or she is configured to respond to the software in "sanctionably appropriate ways" (p. 93). It can be assumed that packaged software embodies scripts of particular behaviors, thus the organization must change its organizational practices in order to fit that prescribed behavior or commit resources to modify the package to match the needs of the organization.

Given the dominance of the American market in packaged software production (estimated at 75 percent of market share [Carmel 1997]), many of the inscribed assumptions embedded within the software will reflect the culture of this country. Carmel amply illustrates the point:

The American culture of packaged software is an intersection of characteristics unique to the (national) American culture and of the subculture of computer programmers and software professionals who share values, interests and life-styles relating to computers and software (Carmel 1997, p. 135).

¹Also known as commercial-off-the-shelf and standard software. It is perhaps interesting to consider the appropriateness of the usage of the term "packaged" software in the advent of other delivery methods of precoded standard software. It is highly unlikely that in future, standard software will be delivered "off the shelf" in a package—rather off a virtual shelf, down "a line."

The latter point, which relates to the subculture of the developers refers to a feature of computer development that is known as Mealy's Law, whereby "The eventual structure of the system reflects the structure of the organization that builds the system" (Yourdon 1976, p. 82). That is, the technical product that has been developed mirrors the organizational structure within which the developers operate (Bloomfield 1992). This clearly has implications for products that have widespread appeal in a global market.

One could expect that organizations involved in the process of selection and procurement of packaged software would be careful in choosing a product that readily met their requirements. Indeed, several studies and consultancy groups offer guidance for this process (e.g., Kunda and Brooks 2000; Lucas et al. 1988), and many of these guidelines are often rooted in rationalistic selection and procurement processes that include the sequential stages of acquisition and specification of requirements, assessment of packages and evaluation of the best fit, and selection of the optimal solution. Much of this literature assumes an unproblematic rational approach to IT implementation, involving rather mechanistic assumptions about the nature of organizations and how they function. Organizational change is seen as no more than the planned, intentional outcome of senior management decision making with the technology providing the optimal technical solution. Despite a number of studies to the contrary (Ciborra 1999; Orlikowski 1996; Suchman 1987), these assumptions constitute the dominant discourse within much of the literature.

In relation to the package software selection process, user involvement has been explicitly catered for, in spite of the oft-cited problems of the so-called communication gap and the difficulties of eliciting and accurately specifying user needs. It has been argued that the software package selection process should follow a linear pattern typically represented as follows: study existing needs, have vendors demonstrate the possible solution, contact other users of the package, and interview some future users (Chau 1995). Clearly, then, there is recognition of the need to understand the requirements of users and involve them throughout the process. This need becomes particularly acute when considering the organizational investment and the length of service the package is expected to provide.

In custom IS development the involvement of users is seen as critical to successful implementation (Markus 1983). However, despite the popularity of user participation, an obvious anomaly is that even *with* user participation, resistance still occurs and systems fail (Cavaye 1995; Olson and Ives 1981). Despite these findings, it has become accepted practice that *not* involving the users in design and implementation precludes success. From a more instrumental perspective, we see that as early as the 1970s user participation was recommended as part of the "solution" for overcoming resistance (Davis 1974). The strategy of salesmanship (Friedman with Cornford 1989), whereby users were taught what was good for them and convinced they should adopt new patterns of working, was a recommended strategy in much of the IS literature. But with packaged software development, the end-users are no longer seen as such an integral part of development as the effort shifts from a focus on the *process* of systems development (often in-house) to the development of a software *product* that has widespread appeal. These distinctions represent a considerable challenge to the way in which we traditionally consider user involvement in the development of IS.

In packaged software firms, developers are the primary revenue generators and so their needs are central to the profitability of the organization. The focus for software vendors is on developing products rather than systems, and adaptations are concerned with more accurately meeting the needs of their specialized market as opposed to a concern about a particular user organization. Accordingly, packaged software developers have a distant relationship with users involving little direct contact (Carmel and Sawyer 1998). Any communication that does occur is usually filtered through intermediaries such as help desk personnel and consultants (Keil and Carmel 1995). However, in order to acquire and maintain market share, software products need to be developed with end users in mind. In an attempt to understand user needs and the demands of the market, a bewildering range of customer-developer links have been developed including trade shows, user groups, and focus groups. Yet this is problematic as research highlights an over-reliance on indirect links such as intermediaries and customer surrogates (systems developers, technical support personnel) (Keil and Carmel 1995), which have been described as "ineffective conduits" (Grudin 1991). Couple this with the assertion that the knowledge of many software developers is often wanting in regard to their operational domain, the extent to which user needs are accurately reflected in packaged software products becomes highly questionable.

At the organizational level, we see a huge contrast between the process of purchasing packaged software as compared to the custom systems development life cycle model (SDLC). It has been argued that of the phases within the SDLC, the first and last—requirements and maintenance—are of most relevance to the purchase of packaged software (Butler 1999). The process of understanding and specifying requirements is seen as crucial to successful software selection. It has been recommended that some form of requirements analysis should be undertaken, at least at a high conceptual level, in order to identify discrepancies between user needs and packaged software before making a commitment (Lucas et al. 1988). With the purchase of packaged software, these steps are now expanded beyond the need to consider one's own functional requirements to include analysis of the available

software with an evaluation of its functionality as compared with organizational requirements. Those involved in this process require new skill sets to enable them to successfully communicate with professional software vendors and third-party intermediaries regarding selection of software, negotiation of licences, etc. The people drawn into this negotiation are rarely the end-users who will have the majority of the contact with the finished system; rather they are often representatives of management and system procurers, although they will often seek to represent user needs.

To summarize, we have witnessed a huge increase in the number of packaged software products that are being implemented across a host of organizations. Given the history of customized development and the well-documented software crisis, it is evident why these products carry such immense appeal. However, this enthusiasm for standardized products must be tempered by an awareness of their inherent limitations. Of particular concern here is the limited attention that is devoted to addressing user needs. Packaged software is very much product-focused rather than concerned with the (mediating) process of development and the corresponding needs of users at the organizational level and few mechanisms have been set up to address this problem.

3 RESEARCH METHOD

Given this context, we were interested in understanding if some of the limitations discussed above could be redressed at the local level by paying attention to user requirements in packaged software selection. The following research questions arise from this objective:

- *How is the packaged software selection process conducted at the local level?*
- *At the organizational level, how are users involved in the selection and purchase of packaged software products?*

To address these questions, we include a summary of findings from an action research project, a research method that has been noted for its appropriateness for IS research (Baskerville and Wood-Harper 1996). Throughout the research process, the role of the researcher ranges from that of detached observer to a fully engaged participant (Blaikie 1993); in this instance our role was to provide specific knowledge to address problems as and when they arose. Throughout the life-span of the project, we were viewed by the organization as temporary, part-time members over a given period of time (two years in total) and so we believe we gained an “inside view” (Walsham 1995) of activities, including access to sensitive information. Therefore, we saw our position as being part of the change process itself. This has been aptly described as “the original intent is to conduct research while effecting change” (Benbasat et al. 1987).

Although our presence in the organization was in a problem-solving capacity, it would be naïve to assume that the various stakeholders shared a set of unitary problems and equally welcomed our suggestions for improvements to practice. Our primary responsibility was to the IT manager, providing guidance to her, as opposed to a commitment to the company and its primary goal of increased efficiency and profitability. Clearly there are degrees of overlap here. As researchers, we were keen that the research process should advance in a participative and collaborative manner, while simultaneously offering advice on theoretical and methodological issues.

Meetings to discuss the project reported here began in September 2001. The study used multiple techniques of data collection including interviews, observation, and analysis of the various texts and documents associated with the process of packaged software selection. For the purpose of this analysis, we have identified three distinct groups of relevance to this project: those involved in (financially) approving the system implementation (senior managers or sponsors); the core project management team whose role was to persuade both sponsors and end-users of the merits of the new system; and, finally, the end-users who would operate the new system. The majority of meetings took place with the core project management team, although on occasion we were asked to make representations at senior management board meetings.

4 AN ILLUSTRATION

The field study is not intended to draw conclusions or even make recommendations, but simply open up channels for further discussion on the implications for user-developer relations with the move to increasing adoption of packaged software.

4.1 Background

The empirical study is directed at a small organization and its procurement of a customer relationship management (CRM) package. The study involves an owner-managed business (T.Co.) that offers a range of high quality career management services

covering executive outplacement, career management, and development. The company's primary customers are placed at the executive end of the market. The company was established in 1990 and operates from three different geographical locations in the United Kingdom.

Although T.Co. is a fairly small organization, the company is hierarchical with strong control and command structures. The owner-manager is a formidable character whose opinions were never questioned by low-level staff. Our observations of employees within the organization suggested they were under huge time pressures when trying to complete their everyday tasks. Indeed, with regard to the project of concern here, one of the documents noted, "Time has to be locked out of team member's diaries to be more productive. The project is a company priority so time should be dedicated accordingly" (minutes of project team meeting, November 16, 2001).

The company's strategic aim is to improve productivity and profitability while maintaining the high quality of service that their customers expect. To achieve this, they are embarking upon a fairly rapid program of expansion to enable them to service a wider geographical area. With this expansion comes a host of issues concerning communication (internal and external) and maximizing the benefits of their information system. The current IT-based business systems have been established in an *ad hoc* manner and are running independently of each other, resulting in problems of communication and control.

Historically, T.Co operated an Apple Mac platform as one of the founding members had a background in the publishing industry and was familiar with Apple technology. The founding members established this infrastructure and were responsible for purchasing the Claris Works and File Maker database, which they initially used to support the companies activities. File Maker was viewed as an ideal low-level product because, as the requirements evolved, the database could be developed incrementally, in a simplistic manner with the addition of new fields and layouts. This perceived advantage rapidly became problematic as File Maker was easy to adapt and minimal technical skill is required to develop the database, leading to a lack of database discipline and no documentation. The IT manager stated that "a realistic comment would be that it has run on, and on, without a real driver." However, this history has also been suggested as offering potential benefits. T.Co. has an element of organizational culture whereby users are able, and willing, to evolve their information system as necessary. Post-migration to the new PC-based environment, the IT manager has suggested that this history of user-led development would be a positive feature in terms of aiding the testing and piloting of any new packages that were purchased.

The project discussed here concerns the acquisition and installation of a new client tracking system in the research department. This department provides a personalized service for clients, described by senior management as a "unique selling point." It was intended that the new system would outline the sequence of activities that began when clients arrived at T.Co. and monitor them as they went through the process of client placement. The client tracking system consisted of two main stages: the first related to finding and securing sponsors (companies that would provide clients, usually as a part of their redundancy package); the second stage concerned monitoring client progress during their time at T.Co. Clearly, the quicker the client turnaround, the greater the profitability for T.Co. It was hoped that a standard CRM package would contribute to this, ultimately leading to enhanced efficiency and the leveraging of a greater market share. CRM packages are increasing in popularity as a standard way of conducting sales and marketing activities and are seen as "valid for any type of deal, business and country" (Ciborra and Failla 2000, p. 107). The aim within this particular organization was to standardize and streamline activities across the three geographic locations, while catering for the possibility of future planned expansion.

4.2 Chronology of Events

Initially it was perceived as crucial that the CRM implementation should begin with the research unit, considered as the most complex business function. At this stage, the end users were aware that a new software installation was planned and they viewed this as a panacea to their problems. Given the stressful nature of their working environment, the technology was seen as a solution to their (time-pressured) problems.

- **Project Start-Up.** During the initial meetings with the project management team, we advised of the importance of eliciting user requirements to enable a more informed evaluation of packaged products and to help narrow the options. While team members were accepting of this suggestion, little concrete effort was put into consulting end users. However, in order to aid their own understanding of user requirements, two members of the project management team conducted an analysis of the *client journey*. The primary purpose of this was the production of a high-level requirements document, which entailed mapping the business processes, information stores, and document flows, along with a textual description of what the journey comprises, resulting in a fairly complex diagrammatical representation. It was intended that this would assist the project

management team in selecting the package that most appropriately matched their business criteria. During this same period, a *focus day* with end-users was scheduled on a number of different occasions; this was intended to provide feedback on the project teams' understanding of user requirements and some additional clarification. For various reasons, the focus day never materialized.

Despite these limited attempts at user enrolment, the main concern of the project team seemed to lie with ensuring the support and (financial) backing of senior management. This was confirmed with much of the documentation that was produced which seemed imbued with rhetoric that appealed to the interests of senior management. These documents included statements declaring "Our aim is to introduce a flexible system that will streamline and improve our current business processes and...speed up the client journey...thus becoming more cost effective" (User Requirements Document, December 20, 2001). Similarly, the client tracking system was hailed as enabling "T.Co. to continue to provide a business class service and grow effectively in the future, whilst maintaining efficiency in all areas" (Board of Directors Document, January 22, 2002). Given that a commitment to purchase was seen as essential if the project was to progress, then their desire to secure the commitment of the senior management team is understandable.

- **The Product Search.** Concurrently, research was being conducted into a variety of packaged software products so that a number of potential suppliers could be short-listed. To help narrow down the options, T.Co held a number of discussions; these included ourselves as researchers at the university, some business acquaintances, and a T.Co employee who had used a CRM package in a previous role. They were also involved in searching the Internet and commercial product guides for reviews of various packages. Arising from this, initial negotiations were set up with three different vendors and the project management team. The vendors include Party A, a supplier for Sage workflow systems; Party B, a supplier for Goldmine; and Party C, another supplier for the Goldmine product. The project management team viewed the different product demonstrations and held meetings with the vendors to discuss their requirements. Communications with Party A were problematic from the outset (poor response on their part) and the product seemed comparatively expensive and so this company never went beyond initial negotiations with the project management team. The other two third-party vendors who sold the Goldmine product had a number of meetings with the project management team before demonstrating the product and discussing its capabilities with the Board.
- **The Beauty Parade.²** The first presentation to senior management involved Party B. Despite having carried out fairly detailed discussions on the nature of T.Co.'s requirements and having viewed the supporting documentation, this presentation was unsuccessful in that the salesperson simply demonstrated the standard product and paid no attention to the needs of the individual organization. Following this, the Board noted their concerns about the value of a CRM package, demanding research into the possibilities of developing their existing applications, which had been custom developed using a File Maker Pro database. The project management team investigated the potential for custom development while at the same time continuing their discussion with Party C. Their presentation to the Board went ahead and, on this occasion, the third-party vendor made extensive use of the background information that they had accumulated and as a result personalized much of the product terminology for their presentation. This was well received by the Board. The owner-manager was delighted with the presentation and shifted position from his initial suspicion of the product to completely embracing it. He now changed tack so that rather than installing the CRM package in the research department with the possibility of future implementation across the other business functions (and locations), the system was to be installed incrementally throughout the whole organization. Senior management resistance to cost seemed no longer relevant as the number of user licences increased and the costs were revised to over double the original estimates. Indeed, the cost of the package from Party C was marginally higher than the same product from Party B, but in the eyes of senior management, the latter were no longer a viable alternative.

The implementation will become progressively more difficult during the rollout plan beginning with sales and marketing, followed by finance, and finally the research department. In terms of business processes, the sales and marketing department are the most straightforward, whereas the other business functions, especially those in the research department (their unique selling point) are far from standard and we anticipate that these areas will be problematic in the future. Somewhat ironically, the research department has been allocated the least amount of time for requirements understanding and user training. To date, no end-user group (from sales, finance, or research) have seen a demonstration of any packaged software product. No requirements analysis has been conducted for the sales and marketing unit. Any discussions that took place were based at the head office; the other two satellite locations were made aware of the impending changes, but were never consulted.

²Term coined by T.CO.

The organization is currently negotiating the finalized contract with Party C prior to the release of funds. Interestingly, their quotation of costs for phase one (for the sales department) includes one day for understanding user requirements and one day for producing the requirements' specification. Detailed analysis will begin once the purchase has gone ahead and it is only during this process that users will, for the first time, become involved in evaluating how (and if at all) the standard software meets their requirements. The vendor will carry out all of the modifications to the software since they argue that this will then tailor the system to the unique requirements of T.Co.

4.3 Discussion

When an organization shifts from developing customized software to consuming standard software products, there are a number of implications. In relation to the field study outlined above, this will be discussed in relation to user involvement and the notion of salesmanship.

Given the well-documented history of the problematic and often conflictual nature of user-developer relations, one could assume that this communication gap (Mumford and Weir 1979) would be reduced (or even eradicated) when dealing with packaged software. Indeed, it seems sensible to assume that, as packaged software has been tested among a range of users over a large installed base, it is not necessary to conduct a detailed and complex analysis of user requirements. However, we argue that user participation remains a valuable activity. This is not because of the instrumentalism involved in reducing resistance, securing project champions, or increasing user commitment, since we believe that many users are adept at seeing through the facade of participation. Rather, we believe that end users are crucial in terms of understanding the operational aspects of the tasks performed and as a result their opinion as to the suitability of a product should at least carry equal weight with the opinions of senior management and systems developers. We suggest that ignoring their wishes could have potentially disastrous consequences.

The field study illustrates that the process of systems development is not a rational process resulting in a system that is an objective representation of true user needs, but is more a reflection of particular social relations and the exercise of professional power (Markus and Bjørn-Andersen 1987). With packaged software selection, power is not simply with the systems developers, since those people who have financial control may also impose their own views on the design process or impose constraints, which could conflict with user needs. So perhaps we are now witnessing a shift from the predominantly technical focus, beyond the consideration of organizational issues, to a market-oriented perspective (Sawyer 2001).

The field study illustrates the power of salesmanship. In relation to the third party vendors we see the power of selling in that two of the contenders represented the same product (Goldmine). Yet, the Party B presentation (of standard software) resulted in a lack of faith in the product on the part of senior management, while the Party C presentation (of configured, standard software) resulted in a complete shift of opinion. Both vendors sold licences for the same product and both vendors suggested how they could modify the software for T.Co.'s unique requirements, yet ultimately the configured presentation with its accompanying sales pitch won hands down. Management was clearly sold a product because of the salesmanship skills of the particular vendor and because of the impression that was created. The skills exhibited during a single demonstration from the vendors resulted in a complete turnaround of opinion among senior management: there was a shift from cynicism to enthusiasm about the product; there was a substantial increase in the number of user licenses as the new system was to extend from the research department to company-wide installation; and the implementation plan was completely reversed (beginning now with sales and marketing). Somewhat ironically, the *process* of salesmanship was elevated above the software *product* that was being purchased.

Interestingly, we also see how the selling of ideas or the act of persuasion operates on a number of different levels, beyond the simple vendor-buyer relationship. The project team aided the third party vendors in the selling activity by working with them to convey senior management requirements (not necessarily user requirements), in order to obtain project approval. This was probably due to the significant amount of pressure upon them to secure an implemented system. They then went on to persuade management (particularly the managing director) of the benefits of packaged software and opted for a solution that would satisfy senior management concerns, giving them a glimpse of project closure. Once the managing director had been won over, he became the project champion and worked on persuading the board of directors of the benefits in order to ensure they would endorse the release of funds. Somewhat ironically, the end users were only ever sold the idea on a very superficial level; they were never consulted nor involved beyond a superficial level. We believe that our presence at the company and the arguments we presented influenced the organization and their consideration of user involvement, although in actuality this never materialized beyond the level of rhetoric. Indeed, our influence could have negative repercussions in that as users began to anticipate a level of involvement and having a voice in the product selection, they may later feel frustrated when they realize their influence was

minimal at best. Our experience as researchers tells us that this lack of user involvement is a significant issue. Just how significant remains to be seen.

5 CONCLUSION

This paper began by commenting on the nature of packaged software, acknowledging the attraction of standardized products while noting the inherent limitations. Of particular interest is the product focus, which is centered on developing a generic product that has widespread appeal across applications, organizations, and nations. Paradoxically, the development of these products reveals that there may be very limited contact with users, despite the huge importance (certainly in terms of market share) attached to addressing their needs. With this in mind, we conducted a study to discover whether user needs were being addressed at the local, organizational level. Using a field study illustration, we showed that during the process of packaged software selection, user needs are further marginalized, as effort from the project management team is expended into securing the approval and authorization of capital investment from senior management. Any salesmanship that occurs is clearly aimed primarily at convincing senior management of the benefit of the CRM package. This salesmanship role concerns the project management team as well as the third party vendors. Not only did these skills secure approval for the intended system, but they also convinced management that the product should be launched across all business functions.

This has implications for our understanding of the systems development process. Historically, both researchers and practitioners concerned with custom systems development have been informed and influenced by the user participation literature, despite its limitations and problems with its effectiveness. The study outlined here illustrates the virtual irrelevance of end-user participation as their needs become marginalized and senior management concerns become prioritized. Thus the traditional role of the analyst/developer is changing from that of mediator (of user requirements and technology) throughout the development process to the role of negotiator of the suitability of software products between senior management and third-party vendors. As increasing numbers of standardized software products are installed, it will be interesting to see the consequences of this on user-developer relations and organizational working practices.

The process of selection and procurement of package software illustrated here contradicts the rational approach commonly reported in the IS literature. While the project management team attempted to identify user requirements and use these to inform the selection and procurement process, these efforts were quickly subsumed by social, political, and economic conditions that far outweighed what would be considered good practice. Ultimately, skilled salesmanship enabled mobilization of bias so that a solution was selected which may be completely inappropriate for both the organization and the end-users. Only time will tell.

6 REFERENCES

- Baskerville, R., and Wood-Harper, A. T. "A Critical Perspective on Action Research as a Method for Information Systems Research," *Journal of Information Technology* (11:3), 1996, pp. 235-246.
- Benbasat, I., Goldstein, D. K., and Mead, M. "The Case Study Research Strategy in Studies of Information Systems," *MIS Quarterly*, September, 1987, pp. 369-386.
- Bingi, P., Sharma, M. K., and Godla, J. K. "Critical Issues Affecting an ERP Implementation," *Information Systems Management* (16:3), 1999, pp. 7-14.
- Blaikie, N. *Approaches to Social Enquiry*. Cambridge, UK: Polity Press, 1993.
- Bloomfield, B. P. "Understanding the Social Practices of Systems Developers," *Journal of Information Systems* (2), 1992, pp. 189-206.
- Bowker, G., and Star, S. L. "Knowledge and Information in International Information Management: Problems of Classification and Coding," in L. B. Frierman (ed.), *Information Acumen: The Understanding and Use of Knowledge in Modern Business*. London: Routledge, 1994.
- Brooks, F. "No Silver Bullet: Essence and Accidents of Software Engineering," *IEEE Computer Magazine*, April, 1987, pp. 10-19.
- Butler, J. "Risk Management Skills Needed in a Packaged Software Environment," *Information Systems Management*, Summer, 1999, pp. 15-20.
- Carmel, E. "American Hegemony in Packaged Software Trade and the 'Culture of Software,'" *The Information Society* (13), 1997, pp. 125-142.
- Carmel, E., and Sawyer, S. "Packaged Software Development Teams: What Makes Them Different?," *Information Technology & People* (11:1), 1998, pp. 7-19.

- Cavaye, A. L. M. "User Participation in System Development Revisited," *Information and Management* (28), 1995, pp. 311-323.
- Ciborra, C. "A Theory of Information Systems Based on Improvisation," in W. L. Currie and R. Galliers (eds.), *Rethinking Management Information Systems*. Oxford: Oxford University Press, 1999.
- Ciborra, C., and Failla, A. "Infrastructure as Process: The Case of CRM in IBM," in C. Ciborra, K. Braa, A. Cordella, and B. Dahlbom (eds.), *From Control to Drift: The Dynamics of Corporate Information Infrastructures*. Oxford: Oxford University Press, 2000, pp. 105-124.
- Chau, P. Y. K. "Factors Used in the Selection of Packaged Software in Small Businesses: Views of Owners and Managers," *Information and Management* (29:2), 1995, pp. 71-78.
- Davis, G. B. *Management Information Systems: Conceptual Foundations, Structure and Development*. New York: McGraw-Hill, 1974.
- Deloitte and Touche. "1996 CIO Survey: Major Packages," 1996 (available online at <http://www.dttus.com>, accessed April 17, 1998).
- Dolmetsch, R., Huber, T., Fleisch, E., and Osterle, H. "Accelerated SAP: Four Case Studies," The Institute for Information Management, Universitat St. Gallen, 1998.
- Friedman, A. L., with Cornford, D. S. *Computer Systems Development: History, Organization and Implementation*. Chichester: John Wiley & Sons, 1989.
- Grint, K., and Woolgar, S. *The Machine at Work: Technology, Work and Organization*. Cambridge, UK: Polity Press, 1997.
- Grudin, J. "Interactive Systems: Bridging the Gap Between Developers and Users," *IEEE Computer*, April, 1991, pp. 59-69.
- Hanseth, O. "The Economics of Standards," C. Ciborra, K. Braa, A. Cordella, and B. Dahlbom (eds.), *From Control to Drift: The Dynamics of Corporate Information Infrastructures*. Oxford: Oxford University Press, 2000, pp. 56-70.
- Keil, M., and Carmel, E. "Customer-Developer Links in Software Development," *Communications of the ACM* (38:5), 1995, pp. 33-44.
- Kunda, D., and Brooks, L. "Identifying and Classifying Processes (Traditional and Soft Factors) That Support COTS Component Selection: A Case Study," *European Journal of Information Systems* (9), 2000, pp. 226-234.
- Light, B. "The Maintenance Implications of the Customization of ERP Software," *The Journal of Software Maintenance: Research and Practice* (13:6), 2001, pp. 415-430.
- Lucas, H., Walton, E., and Ginzberg, M. "Implementing Packaged Software," *MIS Quarterly*, December 1988, pp. 537-549.
- Markus, M. L. "Power, Politics, and MIS Implementation," *Communications of the ACM* (26:6), 1983, pp. 430-444.
- Markus, M. L., Axline, S., Petrie, D., and Tanis, C. "Learning From Adopters' Experiences with ERP: Problems Encountered and Success Achieved," *Journal of Information Technology* (15:4), 2000, pp. 245-265.
- Markus, M. L., and Bjørn-Andersen, N. "Power Over Users: Its Exercise by Systems Professionals," *Communications of the ACM* (30:6), 1987, pp. 498-504.
- Markus, M. L., and Tanis, C. "The Enterprise System Experience: From Adoption to Success," in R. W. Zmud (ed.), *Framing the Domains of IT Research: Glimpsing the Future Through the Past*. Cincinnati, OH: Pinnaflex Educational Resources, 2000, pp. 173-207.
- Mumford, E., and Weir, M. *Computer Systems in Work Design: The ETHICS Method*. New York: John Wiley, 1979.
- Olson, M., and Ives, B. "User Involvement in Systems Design: An Empirical Test of Alternative Approaches," *Information and Management* (4:4), 1981, pp. 183-196.
- Orlikowski, W. J. "Improvising Organizational Transformation Over Time: A Situated Change Perspective" *Information Systems Research* (7:1), 1996, pp. 62-92.
- Quintas, P. "A Product-Process Model of Innovation in Software Development," *Journal of Information Technology* (9), 1994, pp. 3-17.
- Sawyer, S. "Packaged Software: Implications of the Differences from Custom Approaches to Software Development," *European Journal of Information Systems* (9), 2000, pp. 47-58.
- Sawyer, S. "A Market-Based Perspective on Information Systems Development," *Communications of the ACM* (44:11), 2001.
- Soh, C., Kien, S. S., and Tay-Yap, J. "Cultural Fits and Misfits: Is ERP a Universal Solution?," *Communications of the ACM* (43:4), 2000, pp. 47-51.
- Suchman, L. *Plans and Situated Action*. Cambridge, UK: Cambridge University Press, 1987.
- Walsham, G. "Interpretive Case Studies in IS Research: Nature and Method," *European Journal of Information Systems* (4), 1995, pp. 74-81.
- Webster, J., and Williams, R. "Mismatch and Tension: Standard Packages and Non-Standard Users," in P. Quintas (ed.), *Social Dimensions of Systems Engineering: People, Processes, Policies and Software Development*. Hemel Hempstead, UK: Ellis Horwood, 1993, pp. 179-196.
- Yourdon, E. *Managing the Structured Techniques*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1976.