

# **An Initial Model of the Motivations of Packaged Software Developers and their Customers to Establish Close Relationships**

Felicity Murphy  
Peter Seddon

The Department of Information Systems  
The University of Melbourne  
Melbourne, Victoria  
e-mail: fmurphy1@pgrad.unimelb.edu.au

## **Abstract**

*Packaged software is a large component of the global market for software but to-date research into software development practices has mainly been driven by the interests of the developers of custom software. This paper examines the development of packaged software and focuses on the key area of difference between this and the development of custom software – user / customer participation. Partnerships being established between packaged software developers and their customers to develop the packaged software product are examined. A synthesis of associated literature is provided and a simple model for understanding the motivations of developers and customers in establishing close relationships is proposed.*

## **Keywords**

Packaged software, IS development, inter-organisational collaboration, partnership, relationship, joint-development

## **INTRODUCTION**

Packaged software is developed by specialised companies, e.g., SAP, Oracle, and Microsoft, and is usually leased or licensed as a commodity to respond to the needs of a market (both mass and special niches) (Dube 1998; Sawyer 2000, 2001; Barrett 2001). It is offered by a vendor who is typically trying to profit from it and is supported and developed by the vendor who retains the intellectual property rights to the software (Brownsword et al. 2000). Packaged software is also referred to as commercial, shrink-wrapped, commercial-off-the-shelf (COTS), and as software products (Sawyer 2000, 2001).

While the area of custom software development has been extensively studied the development of packaged software has been examined comparatively rarely (Carmel and Becker 1995; Potts 1995). Packaged software is, however, becoming an increasingly important part of the global economy and of organisations information systems choices (Carmel and Sawyer, 1998). As an indication of the importance of packaged software it is reported that between 1995 and 1998 the worldwide software market increased 260% from \$77 billion to \$200 billion and that packaged software sales increased as a component of this market from \$15.4 billion (20% of the total software market) to \$140 billion (70% of the total software market) (Sawyer 2000). Worldwide packaged software sales have continued to grow and in 2002 were approximately \$182 billion with one estimate suggesting that these will increase 6.3% to \$194 billion in 2003 (Kerstetter and Greene 2003). Given that organisations are increasingly using packaged software to satisfy their software needs this area is one that deserves greater attention.

This paper contributes to the limited existing understanding of packaged software development by focusing on the relationships being established between packaged software developers and certain key customers. The research question is as follows:

*Why do packaged software developers and their customers seek more intimate relationships?*

## **RESEARCH APPROACH**

The relationships between packaged software developers and their customers have not been extensively studied and as such understanding of these partnerships is limited. This study aims to remedy the situation by using a synthesis of literature to illuminate current practices and motivations for partnership. The literature drawn upon comes from both academic and practitioner sources in the field of information systems development as well as other related fields. The literature is used to describe the types of intimate relationships being established between software developers and their customers, with examples used to provide clarity. A model based on the

literature is presented to better frame our understanding of the motivations of both customers and developers in establishing such relations. It presents a set of issues which may contribute to a customer or developers inclination to seek involvement in a packaged software development partnership. The literature review provides the basis for the study and it is supplemented by a series of field discussions with packaged software developers which have been used to focus and make sense of the literature examined. These field discussions with software developers all took place in August and September of 2002.

The academic literature drawn upon comes from the fields of management studies, product development and innovation, information systems and software engineering. These literatures were consulted in order to develop an understanding of inter-organisational collaboration and 'customer intimacy' as general theories, the use of customer input to fuel product development and innovation, the use of user input in custom software development, and the packaged software development process. Trade press sources, such as MIS Magazine, Software Magazine, CIO, Information Week and Business Week, and marketing material from software development organisations were examined to understand the phenomena of customer interaction in a packaged environment and to develop a catalogue of current practices.

This literature is supplemented by the insights provided by three software development organisations through preliminary interviews. The organisations selected to provide some grounding to the study through these interviews had to meet the following criteria:

- to be developing packaged software,
- to have engaged in close customer relationships with multiple customers,
- to be developing software for corporate end-users – as opposed to for individuals or exclusively for other software organisations, and
- to be diverse in terms of company size, market scope (international and national), and software application.

The first company, "Dealership Co", develops configurable, network-based car dealership management software which is used to support all aspects of a dealerships business through new and used, car and part sales and servicing. It has integrated human resources, expert vehicle information and customer relationship management (CRM) capabilities. Dealership Co markets the software directly and has in excess of 500 customers throughout Australia and the Asia Pacific region with 23,000 users.

The second company, "ERP Co", develops configurable, mid-size enterprise resource planning (ERP) software. The software has core modules for finance and stock as well as optional modules integrating human resources, quality, plant maintenance and CRM processes. It also has industry-specific modules for distribution, manufacturing, service, project, retail and hire or rental organisations. They have in excess of 900 customers mainly in Australia and maintain a number of company offices and resellers outside of Australia from which they derive ten percent of their revenue.

The third company, "Design Co", develops software for the design of industrial components. It integrates with other computer-aided design (CAD) software packages and allows the user to test the manufacturability and to optimise the production process from design to manufacturing of the component. They have 4,100 mainly large customers around the world with international business accounting for more than 75% of total revenue.

## **CURRENT PRACTICES AND MOTIVATIONS FOR CLOSE CUSTOMER RELATIONSHIPS IN PACKAGED SOFTWARE DEVELOPMENT**

Discussion in this section is based on both academic and practitioner literature as well as preliminary interviews with three organisations<sup>1</sup>. It focuses on current practices and motivations for closer partnerships between customers and software development organisations.

### **Current Practices in this Field**

Since Peters and Waterman (1982) highlighted the association between organizations being close to their customers and "corporate excellence" it has become a common wisdom in many organizations that close relationships breed greater product or service quality, increased organizational flexibility and success (MacDonald 1995). In much of the research on customer closeness the concept has, however, been regarded as "no more than an aspect of marketing" (MacDonald 1995, p. 9) through which customer opinions and expectations are more closely monitored by the traditional customer interaction departments such as sales and

---

<sup>1</sup> As the preliminary interviews were with software developers rather than customers all conclusions drawn from these interviews regarding motivations for customers are those reported on by their vendor.

marketing. In the field of product development the concept has recently been elevated to include the direct involvement of customers in product research and development teams through programs such as joint development, for example: Kjaer and Madsen (1997), Campbell and Cooper (1999), Sawhney (2002), Neale and Corkindale (1998), Spina et al. (2002), Comer and Zirger (1997). In the field of packaged software development Carmel and Becker (1995) advocate this type of customer involvement in their prescriptive process model for the development of packaged software. They found in their study of twelve packaged software developers that the developers were reasonably comfortable involving customers in their development activities (Carmel and Becker 1995, p. 56). In practice some packaged software developers have partnered with customers in the process of developing their software, for example:

### **Product Development Direction Planning**

PeopleSoft solicits customer feedback on its development directions through its global special interest group. This group consisting of its major customers has been involved in setting the strategic product direction in conjunction with PeopleSoft. The group meets twice a year and have “been the most influential driver of the product ... [and that] they’ve become an extended research arm for us” according to PeopleSoft (Frye 1997, p. 63).

Design Co hold focus groups on a regular basis to get input from selected “strategic customers” about future development directions for the product. The input of these strategic customers was critical when Design Co merged with another software development company marketing a similar product and they had to merge the two software products. The “strategic customers” helped guide Design Co’s decisions about what functionality from both products should be supported in the new product and how other functions might be used to replace those being removed. Design Co reports that the role of strategic customer is one that is sought after, with customers having approached them to establish partnerships which might allow them insight into and control over the product development direction (Preliminary interview 2).

Dealership Co have six customers who are jointly involved in setting the direction of development as well as testing pre-beta releases of the software and providing comment and critique. This group is diverse, consisting of both large dealerships using the majority of the system functionality and smaller owner-operator dealerships with an interest in IT or who are using the majority of functionality or using the software in unusual ways (Preliminary interview 1).

### **Business Process Contribution**

QAD, a developer of object-oriented enterprise resource planning software, regularly interacts with their customers in the development of products. In the course of developing their On/Q supply-chain and logistics system they incorporated the recognised ‘best-practice’ business processes of five of their key customers. Adhikari (1997a, p. 130) reports that “each member of the consortium has developed a set of processes that are considered best-of-breed for a given facet of the supply/logistics chain, and each is contributing its processes to QAD for incorporation into On/Q”. One of these was the “world-class logistics and transportation management processes” of Hunt Manufacturing (Adhikari 1997b). This process has given QAD a blueprint from which to design an aspect of the software package and a marketable advantage in its product offering to other customers. Contribution of the process has allowed Hunt to tap the benefits of using packaged software without the usual limitation of having to change the way the organisation operates (Adhikari 1997b).

ERP Co partnered with one of its customers to jointly develop a business-to-business, web-based stock ordering system which encapsulated the customers ordering processes. The company report having been able to capture an estimated 95% of the total functionality currently in the packaged product through this partnership. The partnership enabled ERP Co to “get down to a level of detail we would never have achieved on our own”. The project was co-funded by ERP Co and its customer and ERP Co now owns the exclusive rights to the intellectual property and is marketing the system to its other customers (Preliminary interview 3).

### **Joint-Development**

Xerox undertook the joint-development of a document application system with Cornell University. They involved university staff from the beginning in all of the planning and development activities and had them participate in trialing a prototype system. Xerox found that this joint-development partnership enabled them to establish an ongoing relationship between their development team and the customer. The new system is now “enthusiastically represented by the customer to other institutions. The team continues to be invited back to the customer’s planning functions where further deployment of the system is contemplated” (Anderson and Crocca 1993, p. 53).

Computer Associates (CA) has a “Development Buddy” program through which a small, select group of development partners work “side-by-side with CA developers to create new CA-based solutions, as well as to enhance existing ones” (Computer Associates 1998). Customers offer input on the future product directions of CA, participate in development teams and commit to providing an environment in which to test the products before release. CA credit this program with accelerating their product development process and “ensuring the product meets the customer’s needs” (Gupta 1998, p. S-9).

In Australia, Technology One Corporation has made joint-development with customers a standard means by which to develop new software product offerings. The organisation teamed with Super Cheap Auto to develop their product Retail One and with Curtin University to develop Student One both of which are currently selling well to other organisations (Technology One 1999). These organisations provided the processes on which the software is based as well as assisting throughout development.

Dealership Co has established a partnership with one dealership group to develop a new component for the packaged system. The idea for the component came from the group having multiple dealership sites and wanting greater integration between them. This is a reasonably common phenomena among vehicle dealerships and as such the development is expected to benefit many other dealership groups in Dealership Co’s customer base. The customer in this instance contributed information about their requirements to the developer as well as helping throughout development and testing the pre-beta releases of the product (Preliminary interview 1).

### Motivations for Close Relationships

Having established that customer-developer partnership in the development of packaged software is occurring in practice, the motivations of both customer and developer organisations for participating are now examined. A simple model is used to synthesise the literature on partnership motivations and to frame the study of these relationships (see Figure 1). The model presents a set of issues which may motivate a customer or developer to seek involvement in a packaged software development partnership. The remainder of the paper is devoted to discussing these motivations.

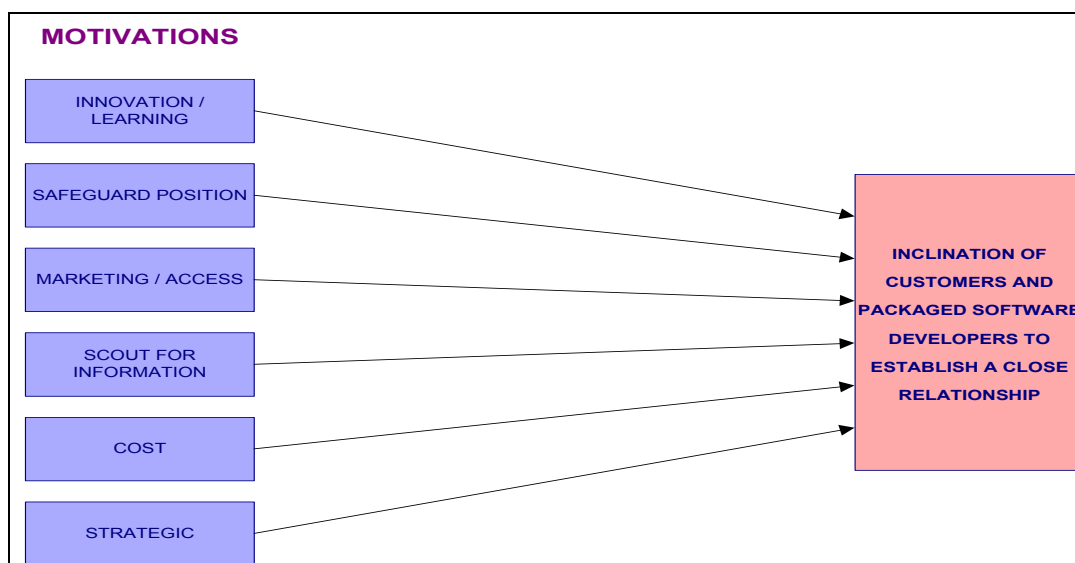


Figure 1: Motivations for Packaged Software Developers and Customers to Establish Partnerships<sup>2</sup>

### Innovation / Learning Motivations

#### Developers

The literature on innovation highlights the benefits of including customers as partners in the product development activities of organisations. Peters (1990) reports that bringing customers inside your organisation and specifically involving them in design and development activities speeds up the innovation process by “inject[ing] a sense of realism and urgency” (Peters 1990, p. 25). Partnership with customers is often cited as a key to increasing the innovative potential and creativity of a development team (Anderson and Crocca 1993; Chiesa and Manzini 1998; Campbell and Cooper 1999). This outcome of customer partnership is possibly a result of the introduction into the development process of new information, the mixing of internal and external

<sup>2</sup> The categories used to organise the review of motivations are loosely based on those used in the survey of value gained by suppliers in buyer-seller relationships in Walter, A., T. Ritter, et al. (2001).  
Murphy, Seddon (Paper #240)

information, which is a key to innovation. This internalisation of knowledge, or learning, through partnership makes the creation of new knowledge possible (Hagedoorn 1993; Inkpen 1996; Thomke and von Hippel 2002).

The benefits of customer involvement in product development are further highlighted in the studies undertaken by von Hippel and various associates of 'lead user' innovation activities across a diverse range of organisations. They find that directly involving leading customers in identifying product development opportunities is faster and cheaper than traditional methods of idea generation and that product outcomes are preferred by customers (Urban and von Hippel 1988; Herstatt and von Hippel 1991; Michelman 1996). von Hippel and Sonnack (1999, p. 12) note that the comparative success of lead customer partnerships in developing products stems from the fact that "user need information can be very 'sticky' – very complex and poorly encoded, and so very hard and costly to transfer from users to the manufacturer". The best way to learn from the customer is to have the product development team working closely with users.

Other partnership benefits include the ability to better monitor and respond to customer needs (MacDonald 1995; Chiesa and Manzini 1998), access to an environment in which to test the design and marketability of products (Neale and Corkindale 1998), and increased satisfaction with the products developed as a result of partnership (Walter et al. 2001; Sawhney 2002). Through partnership, developers have reported being able to refine product plans faster (Sawhney 2002) and achieve an overall reduction in the length of the product development cycle and improved product time-to-market (Hagedoorn 1993; Comer and Zirger 1997; Bailey et al. 1998; Chiesa and Manzini 1998; Campbell and Cooper 1999).

The points outlined above in favour of involving customers in development activities have also been raised in studies of packaged software developer and customer partnerships: for example, Anderson and Crocca (1993), Adhikari (1997a), and Carmel and Becker (1995). Overall, Carmel and Becker (1995) found that partnerships between customers and developers promoted innovation in the developer organisation. Partnerships also gave the developer the opportunity to more quickly learn about specific application domains and to glean insights into a customer's evolving work practices allowing greater anticipation of customer's future needs (Anderson and Crocca 1993). In an industry characterised by the necessity to reduce time-to-market, anticipation of future needs can help a packaged software developer to achieve this (Anderson and Crocca 1993; Gupta 1998).

In the packaged software industry there are many examples of innovation and learning occurring as a result of developer-customer partnerships. New and enhanced products have been developed through customers providing insights into the way they would like to be able to do business supported by packaged software. For example, Dealership Co has partnered with a customer who has provided information about a dealership integration process they wanted to implement which is common to many of Dealership Co's customers and has become a part of the core product (Preliminary interview 1). Products have been developed and enhanced through the codification of customer's processes and the embodying of these in the software, examples of this practice are the partnerships between Technology One and Curtin University and Super Cheap Auto (Technology One 1999) and QAD and Hunt Manufacturing (Adhikari 1997a) discussed above. There is some practical evidence that partnership has allowed the products to be developed better than they would have been by the software developer alone. For example, ERP Co in their customer partnership to develop a web-based ordering system were able to develop a highly detailed system given the domain knowledge provided by the customer. The system is now widely used by ERP Co's customers and only an estimated 5% extra functionality has needed to be added to accommodate other requirements and many of these additions have been as a result of very specific local requirements (e.g. the desire to have the software deploy on a PDA with a mobile phone) (Preliminary interview 3). There is also some evidence from the Computer Associates Development Buddy program that software products are developed more quickly and more accurately meet customer needs through partnerships (Gupta 1998).

#### Customers

Customers also report benefiting from participation in development partnerships by more intimately understanding the product than they would have simply by buying and using it (Anderson and Crocca 1993). The Computer Associates Development Buddy program offers its customer-partners early access to the product under development and to on-site training by CA staff (Computer Associates 1998, 2003). This access, to training skills especially, is expected to increase the customer organisations ability to understand and use the software product. The establishment of a more one-on-one relationship through partnership also means the software developer is more likely to put time and effort into helping the customer organisation learn to use the software.

In a situation where the customer has allowed the developer to codify their processes and use them in the software the customer has an advantage in learning to use the software as it will be uniquely representative of their, already familiar, business practices (Adhikari 1997b). Customer organisations could also experience a general increase in the technical proficiency of the employees who have been closely involved with the software

developers. Anderson and Crocca (1993, p. 54) note that “[c]odevelopment provides a unique learning experience for both users and engineers. In addition to learning the social and technical aspects of developing systems that affect organisational work practices, direct collaboration around using technology in actual work settings enlarges and enriches the work experience of both parties”.

### **Safeguard-Position Motivations**

#### Developers

Close relationships with customers can induce increased loyalty in these customers and discourage them from seeking out other suppliers (Walter et al. 2001). MacDonald (1995) notes that “the information the supplier obtains about the customer [through partnership] may help lock-in the customer ... The greater the integration, the harder it is for the customer to go elsewhere. The attraction for the supplier in terms of market security and reduction in uncertainty is obvious” (MacDonald 1995, p. 11). Walter et al. (2001) reinforce this notion with their observation that increased loyalty in a group of core customers reduces the organisations dependency on other customers and that this group can be held as insurance against hard times. MacDonald (1995) noted this type of behaviour by Logica (a software developer) when choosing potential customer partners. They sought those customers that “offered the most business and ... require least from Logica” (MacDonald 1995, p. 14). This choice runs counter to all other motivations for partnering except the notion that partnering will safeguard a developer’s market position. Similar motivations were evident in all three of the companies involved in the preliminary interviews who all mentioned that the market power of customers often had a bearing on their choice of partner.

Partnership may not only give the developer the opportunity to lock customers in but also to influence the direction of a customers demand. “The supplier may become close enough to influence the customers course of change. This is a much more powerful position than that achieved by interpreting his existing demand more accurately; the supplier moves from being reactive to being gently proactive” (MacDonald 1995, p. 10). Woolgar (1998) reflects the existence of this practice, which he calls ‘user configuration’, in his study of product development in a personal computer company. “The development process included a series of well-orchestrated meetings with key user representatives, where I observed with fascination the ways in which the users routinely made suggestions to the company for the development of products to which the company had already previously committed. It was clear that the producers of the new technology were not so much meeting users’ needs as teaching users what to want” (Woolgar 1998, p. 445). The ability to coax partnering customers in certain directions further reduces the uncertainty for a developer.

#### Customers

Customers may also find that partnership with a product developer ensures it some measure of loyalty from that developer. Customers who have contributed their ‘best-practice’ processes as models for the development of software should certainly be able to exert influence over the developer to offer continuing software support for that process. Partnership may also give customers some scope to influence the developer to extend the product in directions advantageous to them. MacDonald (1995, p. 11) notes that “the very integration that enlightens the supplier also allows the customer to become much more knowledgeable about the suppliers business, and may well enable the customer to influence the supplier’s business as much as the supplier influences the customer’s”. The goal in most supplier-customer partnerships seems to be for the customer to have some measure of control over the direction of development of the supplier. ERP Co gave an example of a customer partner who was able to influence the direction of development in its favour during a period in which the developer was experiencing a downturn in sales. This situation has changed but the agreements made remain (Preliminary interview 3).

### **Marketing / Access Motivations**

#### Developers

Relationships with customers may assist the developer in marketing his products or making contact with other important actors in the industry. Through the relationship the developer can often initiate contact with new customers or collect information about potential customers (Hagedoorn 1993; Chiesa and Manzini 1998; Littler et al. 1998; Campbell and Cooper 1999; Walter et al. 2001). Customers that have been involved with organisations in the development of products often agree to represent these products to other potential customers (Walter et al. 2001). In the packaged software industry, due to the nature of the software product, the establishment of customer reference sites is common. In the case of CA’s Development Buddies an integral part of the agreement that a Buddy enters into with CA is that he must “participate in the CA reference program after products become generally available” (Computer Associates 2003, p. 1). Customers also often voluntarily act as reference sites for a developer with whom they have had a good partnership experience. For example, as a result of the successful relationship with Cornell University, Xerox reports that “the customer commits time to help us

as we spread the word of the available opportunities within our own corporation” (Anderson and Crocca 1993, p. 53). Partnership activity can also bring the developer into contact with important others in the marketplace and enable them to promote the organisations interests through these channels (Moss Kanter 1994; Walter et al. 2001).

#### Customers

Partnerships between customers and developers can generate positive publicity for a customer, especially where they have been chosen to contribute their reputedly ‘best-practice’ processes. Some customers can also expect good publicity to be generated through their association with a high-technology company. Uttal (1993, p. 108) underlines this point by suggesting that “[t]he greatest enticement of all may be the glamour of associating with companies like Microsoft, Novell, and Oracle.”

Participating in a partnership with a developer can often bring a customer into contact with other organisations using the software. Adhikari (1997a) gives the example of a group of five organisations, including Hunt Manufacturing, who were involved with QAD to develop their On/Q package. Involvement in the group enabled the participating organisations to develop contacts with the other organisations involved in the project and Hunt specifically mentions the “benefits from the contributions of the other members of the group.” (Adhikari 1997a, p. 131). Even acting as a reference site can help a customer extend its network. As Faulkner (2001, p. 1) notes acting as a reference site can be “... about developing contacts so that for future deployments you have other people to talk to. It’s about shared knowledge”.

### **Scout-for-Information Motivations**

#### Developers

Partnership with customers often enables the developer to access information about the market and its competitors from a customers’ perspective (Hagedoorn 1993; Chiesa and Manzini 1998; Campbell and Cooper 1999; Walter et al. 2001). It can also provide the developer with access to information about impending technological changes which might affect its product offering (Bailey et al. 1998; Chiesa and Manzini 1998; Walter et al. 2001).

#### Customers

The customer in a relationship can expect similar outcomes in terms of access to information about the market and important competitors. Computer Associates highlights the ability of a customer to stay current with industry trends through participation as a Development Buddy (Computer Associates 2003). Where an industry-specific application is the focus of development a customer might also hope to access some information about the internal workings of other organisations in the same industry who may be participating in the development.

### **Cost Motivations**

#### Developers

The cost of development activities to the developer organisation can potentially be reduced by sharing them with customers through partnership (Hagedoorn 1993; Bailey et al. 1998; Chiesa and Manzini 1998; Littler et al. 1998; Campbell and Cooper 1999). Partnership can also potentially give developers access to the kinds of resources they have decided not to bring within their own boundaries but which are held by the partner organisation (Bailey et al. 1998; Littler et al. 1998; Araujo et al. 1999; Campbell and Cooper 1999; Thomke and von Hippel 2002).

Anderson and Crocca (1993) noted in their study of a co-development project that Cornell University (the customer) was prepared to contribute extensive resources to the development process. Cornell was happy with the outcome and continued to support the developer after the conclusion of the project in spite of it having “been more work than they anticipated” (Anderson and Crocca 1993, p. 56). Customer participation can also reduce the cost of testing the software by providing developers with access to a real-world test ground and testers (Anderson and Crocca 1993; Sawhney 2002).

As well as sharing the costs of development customer partnerships can provide the developer with a way to share and hopefully reduce the risk and uncertainty associated with development (Hagedoorn 1993; Bailey et al. 1998; Chiesa and Manzini 1998; Littler et al. 1998; Campbell and Cooper 1999).

#### Customers

In the course of partnership customers often expend extensive resources and in recognition of this, developers sometimes offer their partners discounts on the resultant products. QAD happily offered the five customers who contributed their processes to the software a discount on their licenses (Adhikari 1997a) and even customers

Murphy, Seddon (Paper #240)

who only act as a reference site for a software developer have been known to receive large discounts on software (Faulkner 2001). Radcliff (1999, p. 2) reports these reference site discounts are “sometimes in the order of 40% off-list.”

Customers involved in commissioning and jointly-developing new software products which are subsequently marketed to others by the software developer often find that although they partially fund the initial development they still come away with a system which is much cheaper than they could have bought elsewhere or developed in-house. This was the case for ERP Co’s customer in their co-funded partnership to develop a web-based ordering system. The customer’s contribution ended up being under \$500,000 which was much less than comparative systems available elsewhere (Preliminary interview 3). Customers have found that these types of partnerships can “provide a way to get the low costs and fast ramp-up times of off-the-shelf software, while adding to the mix a user’s specific requirements” (Adhikari 1997a, p. 129).

Another aspect of partnerships where the customer initially commits funds towards developing the system is that they can sometimes negotiate a share of the revenue attributed to the subsequent sale of the software product. Curtin University, through its partnership with Technology One, received a portion of the revenue from the sale of initial licences for the product to others up to the point where it has recovered its original development investment (Technology One 1999). Super Cheap Auto had a similar deal with Technology One but they shared initial license revenue beyond the recovery of their investment to a predetermined level (Technology One 1999). ERP Co agreed that this was a reasonably common practice and reported sharing profits from the sale of a jointly-developed product with customers on two occasions (Preliminary interview 3).

Customers often maintain in-house developed applications or customisations which integrate with the developers’ packaged system. In situations where customers and developers can jointly develop these as modules or components of the core software package the cost of maintaining the application or customisation devolve to the developer saving the customer future maintenance costs. This cost reduction for the customer was noted by Adhikari (1997a) as well as by the developer from ERP Co (Preliminary interview 3).

## **Strategic Motivations**

### **Developers**

Developers might also anticipate more strategic benefits to result from partnership with customers. Moss Kanter (1994) found that partnership with customers can create a greater internal client-focus within the organisation and both Anderson and Crocca (1993) and Kjaer and Madsen (1997) note that this new focus may require changes in the skills of the organisations’ staff. Anderson and Crocca (1993) found in their study of the Xerox-Cornell University joint-development that this customer-focus engendered greater commitment to the development project in the developers’ employees and increased their feelings of fulfilment in the workplace. It also encouraged them to “develop their own direct and informal contacts within their own and the users’ organisation” (Anderson and Crocca 1993, p. 49) hence increasing the network of experience, knowledge and skills on which they can call in future. This finding is reinforced by Campbell and Cooper (1999) and Bailey et al. (1998) who explain that developers may be motivated to partner in order to gain access to knowledge and skills which can be redeployed later. In this way partnership might be expected to have the strategic effect of changing an organisations’ capabilities and expanding its future opportunities.

### **Customers**

Customers may also anticipate strategic benefits to result from partnership with a developer, not least simply to be the one in their industry or area who has the opportunity to shape the product development direction for the future. Customers who contribute the processes on which the software is designed might expect to have an advantage over competitors who implement the software as it is designed around their work practices. In the case of the joint-development of software tools which have competitive implications for the customer there may be opportunities to request that the developer refrain from selling the software to direct competitors for an agreed period (Adhikari 1997b). This occurred at Dealership Co where they partnered with a leading car manufacturer to re-model their software package to accommodate the country-specific needs of dealerships in a location in which the software had not previously been sold. The customer was allowed exclusive rights to the software product for a period of one year from initial implementation (Preliminary interview 1).

## **CONCLUSION**

This study focuses on the packaged software industry which has not yet been the object of extensive examination. Given the immature understanding we have of development practices in the field of packaged software we compared this field with that of custom software and chose to focus on the area of key difference between them – user/customer participation during development. The study examined the establishment of



partnerships between packaged software developers and customers toward the development of software tools. An overview of the scarce literature in this area and a series of preliminary interviews with software developers found that partnership activity is occurring. Through a synthesis of related literature the study then attempted to make sense of the motivations of both developers and customers for engaging in partnerships. These motivations for developers seem to fall mainly in the areas of product innovation, safeguarding their customer base, giving them access to people and information both in the customer organisation and more widely, potentially sharing costs of development, and more strategic motivations, such as creating a customer focus internally and getting access to important resources and knowledge which could be deployed elsewhere. The motivations for customers seemed to be around increasing their knowledge of the product, having a forum within which to influence development direction in their interests and perhaps afford them some competitive advantage as a result of this, and to reduce the ongoing cost of maintenance on customisations or to get a system which meets their unique requirements more cheaply than was otherwise available.

The three main contributions of the study are: (1) that it focuses attention on packaged software development which as Potts (1995) notes has only rarely been a focus in the field of information systems and software engineering which has chosen to develop models and methods almost exclusively for use in the development of custom software; (2) that it highlights the differences between the customer-developer relationships in the two fields and perhaps encourages the research community to follow the advice of Carmel and Sawyer (1998, p. 16) to be aware of the differences between the fields "when studying development teams and activities as they will have implications for the methods advocated for use in development teams and for those practices which could be borrowed from the other field"; and (3) that it proposes a potentially useful model of motivations to frame studies of partnership between users and developers in the packaged software field.

The limitations of this study mainly revolve around the fact that it is a synthesis of the findings of related work. Given the shortage of prior work focusing on packaged software development, and customer participation in this field specifically, information directly relating to the topic under study was difficult to access and meant the study relies upon practitioner literature and the findings of research on partnerships in other areas. A second weakness is that the preliminary interviews used to make sense of the phenomena were solely conducted with software developers rather than their customers and any conclusions drawn about customer motivations were based on the observations and assumptions of the developer organisation. The study is also limited by its focus exclusively on the 'positive aspects of partnerships' or motivations. We are currently engaged in case studies focusing on the conduct and outcomes of these partnerships which should do greater justice to understanding the working of these complex relationships.

## REFERENCES

- Adhikari, R. (1997a). "Partner Power." *Information Week*(650): 129-132.
- Adhikari, R. (1997b). "Stay Ahead of the Competition." *Information Week*(650).
- Anderson, W. L. and W. T. Crocca (1993). "Engineering Practice and Codevelopment of Product Prototypes." *Communications of the ACM* 36(6): 49-56.
- Araujo, L., A. Dubois, et al. (1999). "Managing interfaces with suppliers." *Industrial Marketing Management* 28(5): 497-506.
- Bailey, W. J., Masson, R., and Raeside, R (1998). "Choosing Successful Technology Development Partners: a best-practice model." *International Journal of Technology Management* 15(1/2): 124-138.
- Barrett, R. (2001). "Labouring Under and Illusion? The labour process of software development in the Australian information industry." *New Technology, Work and Employment* 16(1): 18-34.
- Brownsword, L., Oberndorf, T., and Sledge, C. A (2000). "Developing New Processes for COTS-based Systems." *IEEE Software* 17(4): 48-+.
- Campbell, A. J. and R. G. Cooper (1999). "Do Customer Partnerships Improve New Product Success Rates?" *Industrial Marketing Management* 28(5): 507-519.
- Carmel, E. and S. Becker (1995). "A Process Model for Packaged Software Development." *IEEE Transactions on Engineering Management* 42(1): 50-61.
- Carmel, E. and S. Sawyer (1998). "Packaged Software Development Teams: what makes them different." *Information Technology and People* 11(1): 7-19.
- Chiesa, V. and R. Manzini (1998). "Profiting from the Virtual Organization of Technological Innovation - suggestions from an empirical study." *International Journal of Technology Management* 15(1-2): 109-123.

- Comer, J. M. and B. J. Zirger (1997). "Building a Supplier-Customer Relationship Using Joint New Product Development." *Industrial Marketing Management* 26(2): 203-211.
- Computer Associates (1998). *Computer Associates Annual Report 1998*. New York, Computer Associates.
- Computer Associates (2003). *CA Development Buddy Program*, Computer Associates. 2003.
- Dube, L. (1998). "Teams in Packaged Software Development: the Software Corp. experience." *Information Technology and People* 11(1): 36-61.
- Faulkner, N. (2001). "CIO Roundtable: to reveal or not to reveal?" *MIS Magazine UK* 11/01/01.
- Frye, C. (1997). "PeopleSoft Goes Global." *Software Magazine* 17(3): 62-66.
- Gupta, Y. (1998). *Developing R&D Buddy Programs*. R&D Magazine: S-9.
- Hagedoorn, J. (1993). "Understanding the Rationale of Strategic Technology Partnering: interorganizational modes of cooperation and sectoral differences." *Strategic Management Journal* 14(5): 371-385.
- Herstatt, C. and E. von Hippel (1991). *Developing New Product Concepts Via the Lead User Method: a case study in a "low tech" field*. MIT Sloan School of Management Working Paper. Boston: 1-15.
- Inkpen, A. C. (1996). "Creating Knowledge through Collaboration." *California Management Review* 39(1): 123-140.
- Kerstetter, J. and J. Greene (2003). "Software: getting its groove back." *Business Week* 13 Jan, 2003.
- Kjaer, A. and K. H. Madsen (1997). "Customer-vendor Co-operation." *Information Technology and People* 10(3): 205-223.
- Littler, D., Leverick, F., and Wilson, D. (1998). "Collaboration in New Technology Based Product Markets." *International Journal of Technology Management* 15(1/2): 139-159.
- MacDonald, S. (1995). "Too Close for Comfort - the strategic implications of getting close to the customer." *California Management Review* 37(4): 8-27.
- Michelman, P. (1996). *Home Bankers Prefer Ideas of "Lead Users"*, Research Shows, MIT Tech Talk.
- Moss Kanter, R. (1994). "Collaborative Advantage - the art of alliances." *Harvard Business Review* 72(4): 96-108.
- Neale, M. R. and D. R. Corkindale (1998). "Co-Developing Products - Involving Customers Earlier and More Deeply." *Long Range Planning* 31(3): 418-425.
- Peters, T. (1990). "Get Innovative or Get Dead." *California Management Review* 33(2): 9-26.
- Peters, T. and R. Waterman (1982). *In Search of Excellence: lessons from America's best run companies*. New York, Harper and Row.
- Potts, C. (1995). *Invented Requirements and Imagined Customers: requirements engineering for off-the-shelf software*. Proceedings of the Second IEEE International Symposium on Requirements Engineering, York, UK, College of Computing, Georgia Institute of Technology, Atlanta, GA.
- Radcliff, D. (1999). "Paths to a Better Deal." *Computerworld* 33(2): 61.
- Sawhney, M. (2002). "Don't Just Relate - Collaborate." *Sloan Management Review* 43(3): 96.
- Sawyer, S. (2000). "Packaged Software: implications of the differences from custom approaches to software development." *European Journal of Information Systems* 9(1): 47-58.
- Sawyer, S. (2001). "A Market-based Perspective on Information Systems Development." *Communications of the ACM* 44(11): 97-102.
- Spina, G., Verganti, R., and Zotteri, G. (2002). "A model of co-design relationships: definitions and contingencies." *International Journal of Technology Management* 23(4): 304-321.
- Technology One (1999). *Annual Report 1999*, Technology One.
- Thomke, S. and E. von Hippel (2002). "Customers as Innovators - a new way to create value." *Harvard Business Review* 80(4): 74-81.
- Urban, G. L. and E. von Hippel (1988). "Lead User Analyses for the Development of New Industrial Products." *Management Science* 34(5): 569-582.

- Uttal, B. (1993). "When to Be a Beta Site." *Fortune* 128(7 Special Issue SI): 108 ff.
- von Hippel, E. and M. Sonnack (1999). *Breakthroughs to Order at 3M*. MIT Sloan School of Management Working Paper. Boston: 1-21.
- Walter, A., Ritter, T., and Gemunden, H. G. (2001). "Value Creation in Buyer-Seller Relationships - theoretical considerations and empirical results from a supplier's perspective." *Industrial Marketing Management* 30(4): 365-377.
- Woolgar, S. (1998). "A New Theory of Innovation." *Prometheus* 16(4): 441-452.

## **COPYRIGHT**

Felicity Murphy and Peter Seddon © 2003. The authors assign to ACIS and educational and non-profit institutions a non-exclusive licence to use this document for personal use and in courses of instruction provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive licence to ACIS to publish this document in full in the Conference Papers and Proceedings. Those documents may be published on the World Wide Web, CD-ROM, in printed form, and on mirror sites on the World Wide Web. Any other usage is prohibited without the express permission of the authors.