Knowledge Management:
Concepts, Methodologies, Tools, and Applications

Murray E. Jennex
San Diego State University, USA

Information Science REFERENCE

INFORMATION SCIENCE REFERENCE
Hershey · New York

Assistant Executive Editor:	Meg Stocking
Acquisitions Editor:	Kristin Klinger
Development Editor:	Kristin Roth
Senior Managing Editor:	Jennifer Neidig
Managing Editor:	Sara Reed
Typesetter:	Amanda Appicello
Cover Design:	Lisa Tosheff
Printed at:	Yurchak Printing Inc.

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue, Suite 200
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-pub.com
Web site: http://www.igi-pub.com/reference

and in the United Kingdom by
Information Science Reference (an imprint of IGI Global)
3 Henrietta Street
Covent Garden
London WC2E 8LU
Tel: 44 20 7240 0856
Fax: 44 20 7379 0609
Web site: http://www.eurospanonline.com

Library of Congress Cataloging-in-Publication Data

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Chapter 8.13
Managing Intellectual Capital and Intellectual Property within Software Development Communities of Practice

Andy Williamson
Wairua Consulting Limited, New Zealand

David M. Kennedy
Hong Kong Institute of Education, Hong Kong

Ruth DeSouza
Wairua Consulting Limited, New Zealand

Carmel McNaught
Chinese University of Hong Kong, Hong Kong

INTRODUCTION

In this article, we will develop a framework for educational software development teams that recognizes the conflicts and tensions that exist between the different professional groups and will assist software teams to recognize the intellectual capital created by individuals and teams. We will do so by recognizing the inherent relationship between the tangible elements of intellectual property and the intangible organizational assets that form the basis of intellectual capital and by discussing how knowledge generated by a project team can become an explicit asset.

BACKGROUND

Universities are increasingly becoming developers of complex software-based applications. In-house development ranges from teaching aids and online learning resources to large information systems products that could ultimately become successful commercial ventures. Increased product complexity is easily recognized, yet research shows that the organizational aspects of a software development project are more likely to affect performance and outcomes than technical issues (Xia & Lee, 2004). Successful development and deployment of today's complex educational systems and environments comes with an imperative for an array of different and unique skill sets for the various stages of each project. One can view a software development team as a microcosm of the wider community of practice of software development professionals who work in information and knowledge management in higher education. As Wenger (1998) observes, such communities of practice are not random but constructed around required skills and through a process of negotiation based on mutuality and accountability.

Workforce mobility has increased: academic staff members regularly and easily move between institutions; development and design staff have many opportunities for contract-based work, move to other academic institutions or into the private sector. The ideas that lie behind a successful process or product are increasingly drawn from a wider pool of talent and, as people move around, these ideas are being taken with them and disseminated through informal and new work practices into a wider community of practice. How then does a team, formed to design and develop a technology-rich educational or systems environment, manage and control issues of intellectual capital and intellectual property such that all of those who contribute throughout the life of a project are acknowledged and rewarded fairly and appropriately for that contribution, even after they have left the project?

Team Formation and Relationships

Additional complexity leads to specialization (Jacobson, Booch & Rumbaugh, 1998). New ways of working bring with them a shift in power, where the academic expert will often lack the technical skills, time or resources to turn ideas into reality. Instead, they must rely on a team of experts from other disciplines to interpret their ideas, evolve them, and deliver the finished product. As complexity increases, communication between team members becomes paramount; specialist educational designers are required to translate pedagogy into functional specifications that can be understood by software developers and graphic designers. Modern software teams are project-based, where resources come and go as required.

Software development communities of practice exist within a larger organizational context. Roles and responsibilities will vary and are negotiated depending on the toolset and architecture used, the size of the project, and the culture of the organization (Phillips, 1997; Williamson et al., 2003). Project team members can be full- or part-time employees (academic or non-academic) or contractors retained specifically for the project. As such, these roles exhibit complex relationships and interfaces between each other and the project. In Figure 1, a range of typical roles and relationships found in a tertiary education software development project are shown.

During the various stages of the development process, various players move into prominent roles. One way to illustrate this shifting set of work responsibilities is to list the main players at each stage of the process. We will do this using the classic instructional systems design (ISD) model (Dick & Carey, 1990) as it is so well known. (There are many other models, many of which are discussed in Bannan-Ritland, 2003.) The key players at each stage of the ISD model are listed

Figure 1. Intra-project relationships in software development teams (Williamson et al., 2003, p. 345)

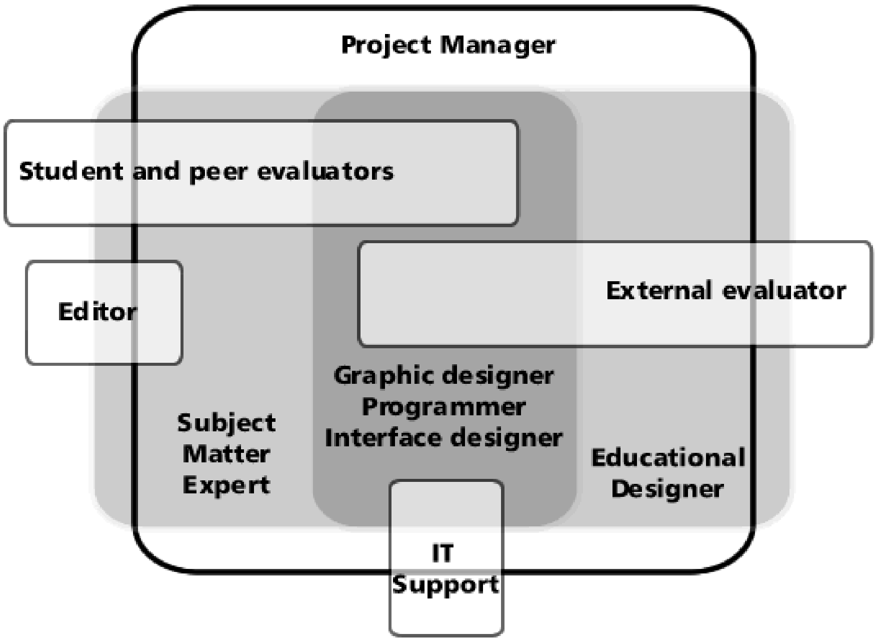


Table 1. Key players at each stage of the ISD model

Stage of the ISD model	Key players
Needs assessment	Subject matter expert
Analysis	Subject matter expert, Educational designer
Design	Subject matter expert, Educational designer, Project manager
Development	Project manager, Graphic designer, Programmer, Interface designer, Editor
Formative evaluation	Student and peer evaluators, Subject matter expert, Educational designer
Revision	Project manager, Graphic designer, Programmer, Interface designer, Editor
Implementation	Subject matter expert, IT support
Summative evaluation	External evaluator
Maintenance	IT support, Subject matter expert (Interface designer)

in Table 1. In reality, each team parcels out the work depending on the skill set of individuals in the team.

It is important to be aware of the different communities of practice that exist in this field and ensure that the role of individual team members is able to be promoted appropriately. Professional recognition can come through either publication, a portfolio of work or through the finished product, and the importance of a successful project to the career development of individuals should not be underestimated. It is important to ensure that academic dissemination of successful projects through publication recognizes the contribution made by all team members, including the non-academic members. Many myths persist in relation to acknowledging the veracity of contribution with regard to educational software, and these often have the potential to leave team members feeling their effort and ideas have gone unrecognized and, at worst, feeling they have been exploited (Williamson et al., 2003). In the second half of this article, we will develop a framework that ensures appropriate outlets for reward and recognition of individual contributions within academic software development teams. Before doing this, we will define what is meant by intellectual capital and intellectual property.

Defining Intellectual Capital and Intellectual Property

Florida (2002) argues that the principal factors for successful software development are talent, knowledge, and intellectual capital (IC). The connection of new ideas and existing knowledge within an organizational context leads to the creation of IC. Stewart (1999) defines IC as the sum of everything everybody in a company, organization, or team knows and which provides some advantage over their competitors. Davidson and Voss (2001) agree, describing individual IC as "the sum of individual imagination, intelligence and ideas" (p. 60). They then extend this definition to encapsulate a model for organizational IC that is based on the talent of individuals (human capital), the knowledge that is captured within systems and processes (structural capital), and the characteristics of relationships with customers and suppliers (customer/supplier capital). Organizational IC comes from the "interplay of all three (structural capital augments the value of human capital, leading to an increase in customer/supplier capital)" (p. 61). In terms of this discussion as it relates to the appropriate recognition and acknowledgement of individual contributions within software development teams, human capital is our primary focus. Human capital is "what walks out of the door at the end of the day" (p. 68); it is a vital intangible.

If IC is the intangible but invaluable contribution of human talent to a project, then Intellectual Property (IP) is a formal measurable subset. It is the tangible product that results from the idea. The UK Patent Office (United Kingdom Patent Office, n.d.) defines four formal types of IP:

- patents for inventions;
- trademarks for brands;
- designs for product appearance; and
- copyright for material (including software and multimedia).

This definition is then extended to cover a much broader and often more intangible grouping that extends to trade secrets, plant varieties, geographical indications, and performers rights. While many see copyright as a way of protecting IP, it is only a subset. Copyright provides recognition of their invention to the creators of software or multimedia products in order for them to be able to obtain economic rewards for their efforts (Macmillan, 2000). Historically, comparisons have been drawn between software development and the traditional arts, such comparisons reinforcing an argument that IP law is focused on the protection of software such that others are not able to modify the source product (White, 1997). It is important to note that copyright extends only to a tangible product, it does not lend protection to the more intangible areas of IC such as ideas and individual contribution. Since copyright has a primarily commercial imperative, it is a limited and perhaps inappropriate mechanism for acknowledging contribution. This is of greater importance in higher educational settings since copyright of educational materials can reside with the institution (particularly with off-campus courses), rather than the individual, and very few educational software products developed for specific content domains in higher education are ever commercialized (Alexander, McKenzie & Geissinger, 1998).

The Relationship between IC and IP

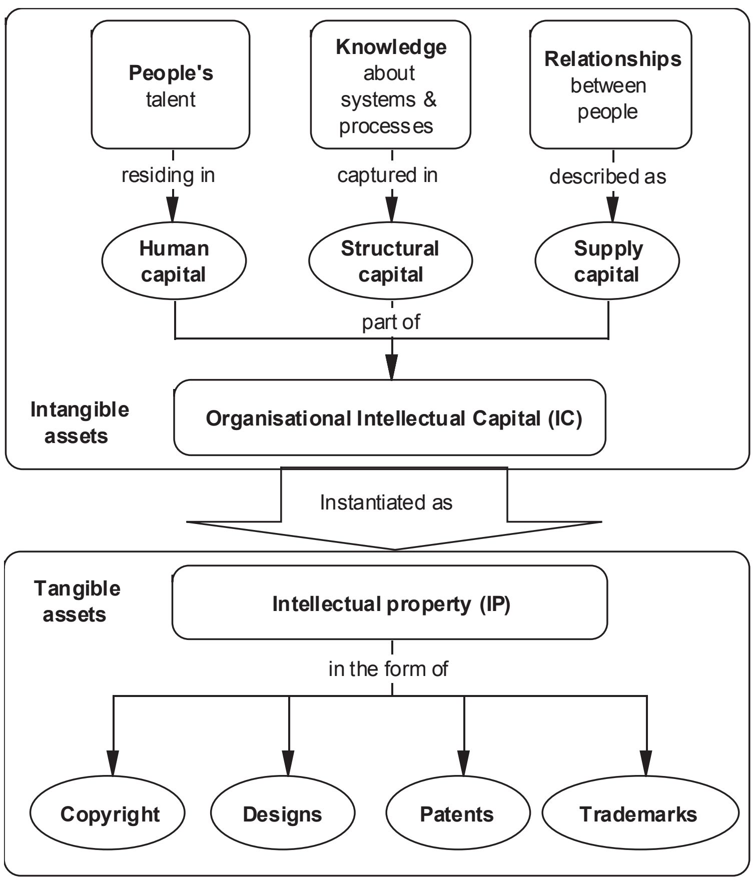
A relationship exists between the tangible elements of intellectual property and the three forms of intellectual capital (the intangible organizational assets) discussed in the preceding section. These are shown in Figure 2.

IC/IP Management Framework

Having addressed the complexity of educational software development teams and defined IC and IP within an educational software context, we will now develop a framework that can be used to ensure proper recognition and reward for individual and collective ideas in such a setting.

Given the critical value of IC in software development (Florida, 2002), it is important that the processes used within educational software development are strengthened and formalized through the adoption of a strong project man-

Figure 2. Intellectual capital and intellectual property (Williamson et al., 2002, p. 342)



agement framework. Project management is a key role in any project involving information and communication technologies and interactive multimedia software, and it requires specific skills and attributes. These include both the hard skills of contract negotiation, budgeting, scheduling, project definition, and scoping as well as the soft skills of human relations, team building, and facilitation (Burdman, 2000; Schwalbe, 2000). Successful teams work well together because they have clear roles and relationships and because the terms of engagement within the team and with external parties are well defined, understood, and agreed by all. This provides a solid platform for the explicit incorporation of IC and IP policies into project documentation so that such issues can be considered early on, preferably during the project scoping phase.

A process and framework are required to recognize knowledge as it is created so that it becomes explicit. Without doing so, knowledge remains tacit and cannot be rewarded or acknowledged, that is, credited to the appropriate team members in the future. Extending this concept, knowledge that is explicit within the team can remain tacit beyond team boundaries if no process is in place to ensure appropriate recognition of contribution. It is, therefore, necessary for teams to negotiate clear, up-front delineation of roles, responsibility, and ownership of both tangible and intangible outputs from the project. This does not prejudge what that ownership might be, merely that the agreement takes place before the project commences. It is important to consider how IC/IP generated during the project's life will be disseminated, in what form, and by whom. Such a clear articulation of roles and responsibilities has the benefit of helping to make the process of dissemination more visible. By doing so, it is hopefully the case that team members will recognize the significance of the different sources of acknowledgment. This in turn will result in up-front agreement on potential opportunities for dissemination of original ideas among the team.

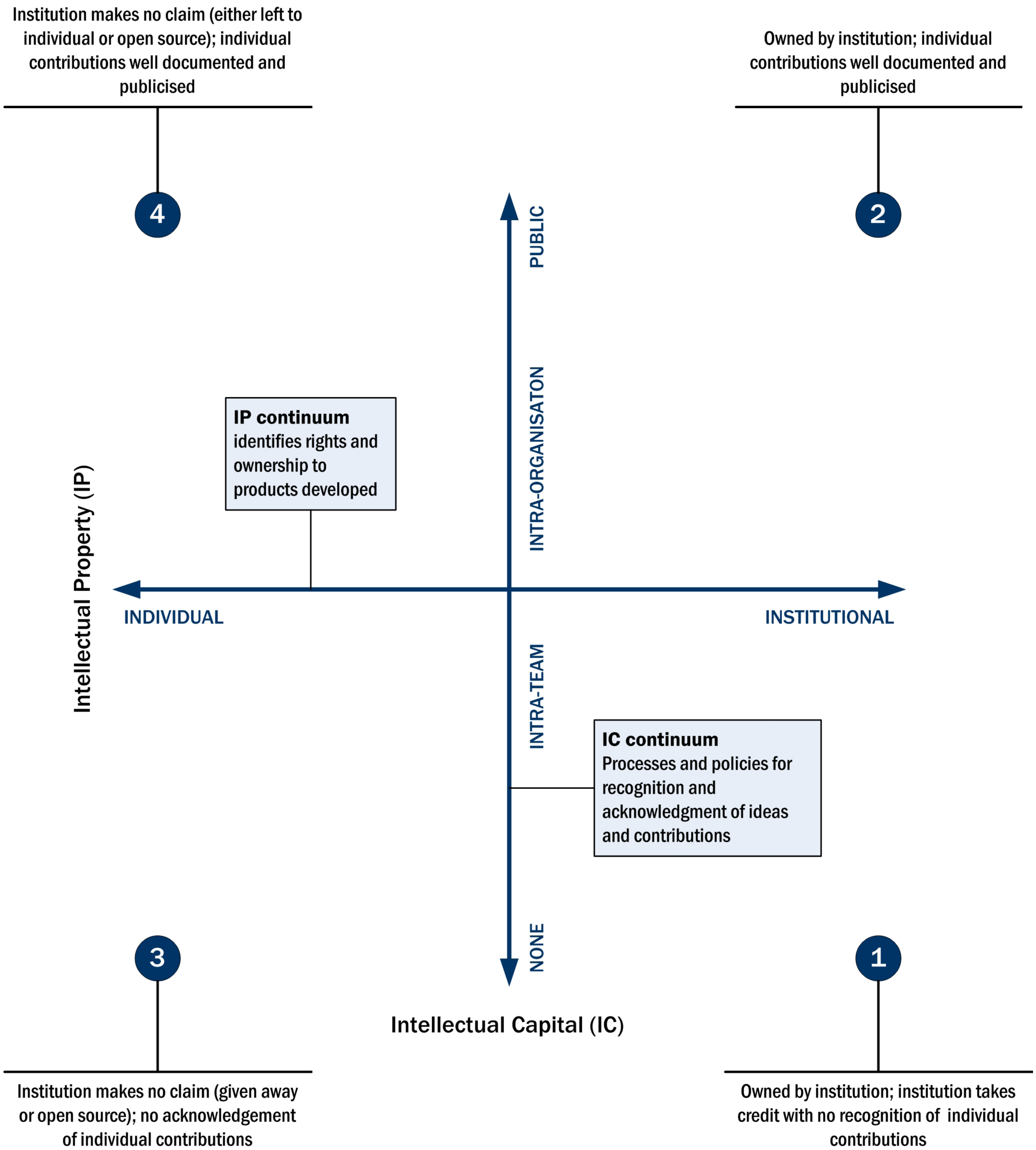
A seldom discussed aspect of the manner in which ideas might be disseminated (and credit obtained) is the potential synergy between individual team members. For example, among academic staff involved in the project, there is a possibility for cross-disciplinary publications.

This framework, shown in Figure 3, maps out two axes: the horizontal axis representing formal ownership of the tangible IP, the vertical axis representing a continuum of recognition for the IC generated during a project, ranging from no acknowledgment of individual effort and contribution to a full public acknowledgment. Intermediate steps include recognition at the team and institutional level.

Enacting the IC/IP Framework

Our discussion so far has shown that, regardless of the nature of the IP ownership, academics and professionals working in software development teams need appropriate recognition for their contributions, but certain factors can prevent this from happening. The challenge, therefore, is to identify a set of project attributes that can be used to inform project management practices such that institutions are cognizant of the need for appropriate recognition. In the following section, we identify seven key attributes of, or processes within, a successful project. The model is developed from a review of the authors' own experiences of software development teams where problems had occurred. This review led to the identification of which weaknesses in the process had resulted in these problems (Williamson et al., 2003). By ensuring that these seven attributes are recognized and actively negotiated by newly forming teams and enacted throughout the life of the team, this model can assist projects in identifying

Figure 3. IP/IC management framework



and filling gaps in the structure of development teams, hence future risk can be mitigated.

In essence, the IC embedded in the members of the project team is articulated in terms of the various IP contributions made by these team members. However, if the focus is exclusively on the tangible products of the process (e.g., software and papers) through only considering the IP, then the worth of ideas (IC) can be underplayed, and their potential may not be realized. We are suggesting that explicit application of these seven guidelines can ensure more successful project outcomes and positive professional outcomes for all members of the team.

The nature of an effective community of practice for software development teams is discussed in terms of Figure 3. The two major axes and the four examples in Figure 3 are used to frame the seven attributes.

Intellectual Property: Individual Affirmation to Institutional Affirmation

Have an IP Acknowledgement Strategy

Highly successful projects exhibit a strong team dynamic which arises when the expertise and knowledge of individual team members can be communicated and shared with others. Part of this process involves ensuring that ideas are fairly acknowledged within and outside the team, whether by portfolio (graphic artists), publication (academics), or product (project managers and programmers).

Have an IP Review Strategy

It does not matter for the purposes of academic critical review whether the subject of study is a written paper, a software product, or a portfolio. Contribution from individual team members needs to be acknowledged through an inclusive authorship policy which is regularly revisited in team meetings. This process can strengthen collegiality and reinforce mutual valuing between team members.

Have a Strategy to Separate IP from IC

The IP might be owned by an organization or institution, but the IC remains with the individuals in the team. Formal acknowledgment of where the ownership of IP lies is important and needs to be negotiated ahead of the commencement of the project. In many higher education institutions, this has become standard practice and involves retaining a competitive advantage and protecting the resources produced by employees of the organization. There are risks associated with key project contributors leaving (for example, a lead programmer) and either taking intellectual property with them or holding a software development team or institution to ransom by withholding access to code or other resources. In some organizations, the IC also remains with the organization via means of a nondisclosure agreement. Communities of practice might consider using a confidentiality agreement as part of a contract or offer of employment in order to keep this issue open and transparent.

Longevity Strategy: Ideas Remain

When a person leaves a team, they cede their IC to the project team or institution, and that contribution should continue to be recognized and acknowledged in project documentation, appropriate publications, and authorship in any finished product. In some projects, this may also involve ceding formal IP to the project (e.g., in the case of commercialization).

Intellectual Capital: No Formal Acknowledgment to Public Acknowledgment

Recognize the Emergent Nature of the Software Development Process and Its Impact on IC/IP for all Team Members

As software becomes more complex, it becomes less and less likely that the original academic imperative that led to the idea for the product will be instantiated in a form initially envisaged by the academic or the organizational unit that initiated the project. The development process and the end result will be strongly influenced by a wide range of individual and group contributions to the process and the product.

Ideas are Perishable

Software has a shelf-life, hence the IC that led to that product is also of limited use. The idea will become superseded and outdated as new ideas and new technologies emerge. For example, there are any number of commercial or free customizable online survey instruments (such as Survey Monkey, http://www.surveymonkey.com) that now exist. Learning Evaluation Online (LEO) was an early system that explored how customizable educational surveys could be developed online using an entirely Web-based interface (Kennedy & Ip, 1998). At the time this was an innovative approach, but it has since been superseded by more robust software. Thus, the IC for LEO has long since expired. The idea behind LEO has been

Table 2. Implications for IP and IC

#	Scenario	Implications for IP and IC
1	IC and IP is owned by institution; institution takes credit with no contribution of individual creativity and effort.	This is a very poor scenario for developing the IC of an institution. Without affirmation, individuals will seek employment elsewhere and take their IC with them.
2	IP and IC are owned by the institution; individual contributions are well documented and publicized.	This is the scenario in a number of institutions worldwide, particularly those involved in distance education. This scenario is problematic when commercial aspects enter the situation as in the case of patents.
3	Institution makes no claim (software is given away or open source); no acknowledgment of individual contributions.	This is often the result of small student projects (although many institutions claim the IP of all undergraduate student work, not postgraduate) undertaken during a course of study). Most software of this type has a very limited life although there are some exceptions (Gunn, 1995).
4	Institution makes no claim (either left to the individual or open source community); individual contributions are well documented and publicized.	This applies to postgraduate work in universities. In many institutions, postgraduate (especially doctoral) students own their IP, and it is up to the student and supervisor to disseminate the details of the project. This aspect is changing as universities try to gain a competitive advantage, and many postgraduate students working in a large research department would do well to consider how the results of their studies might be retained, negotiating with the university in the early phases of the project. For example, some student projects (see moodle.org and moodle.com) have become very high profile products (Dougiamas & Taylor, 2003).

taken up by others and reproduced using different software code. The code is the instantiation of the idea and is the only part of the project subject to IP rights.

Public Acknowledgment of IP/IC Requires the Source Material to be in the Public Domain

Acknowledgment of unpublished work or work not publicly available is not sufficient to acknowledge IC and IP issues in a publication. In the case of academics where affirmation and professional career progress is at least partially a result of publication in accredited arenas such as books and journals, this is clearly not sustainable. Graphic artists, on the other hand, have their portfolios of work with iterations of visual designs that they take with them to the next project or job; and programmers have compilations of code: for these professionals, the publication is less important or substantive in career development. A key issue for an institution is providing the process by which academic publications can be developed without compromising the IP of the individual or trade advantages in the marketplace.

In summary, the implications for the four scenarios in Figure 3 are shown in Table 2.

THE FUTURE: APPLICATION OF THE SEVEN ATTRIBUTES OF THE IC/IP FRAMEWORK

In order to see how these attributes can be enacted in practice, the example of a major Australian multimedia project, An@tomedia, will be used. An@tomedia was designed to support problem-based learning (PBL) of anatomy in the Faculty of Medicine at the University of Melbourne (http://www.anatomedia.com). A number of academic evaluations on the role of An@tomedia in this PBL learning environment have been published (e.g., Kennedy, Eizenberg & Kennedy, 2000; Kennedy, Kennedy & Eizenberg, 2001).

The software has been successfully commercialized by the four subject matter experts (core SMEs or core authors) after other members of the development team ceded any personal commercial claims to the group by means of a legal document to that effect.

Affirmation and acknowledgment involving publishing for academic members (Kennedy et al., 2000; Kennedy et al., 2001), contributions to portfolios for non-academic members, and public acknowledgments in the An@tomedia Web site for every person who contributed in any significant way to the project were not affected by this written agreement. The public affirmation (particularly important for non-academic members of the project team) is illustrated by the observation made by a reviewer of An@tomedia in The Lancet (Marušiæ, 2004) where she mentions the extensive list of credits for all the members of the team (over 60) involved with the project. This process was accomplished quite simply because matters of IP had been previously discussed in the course of project meetings, and the "prior art" that existed and underpinned the educational approach was well known to all project members. Table 3 summarizes the way in which the seven attributes worked in this project.

CONCLUSION

While formalized tools exist for capturing IP generated during a project, most software development teams lack formal explicit processes for ensuring that the IC generated is accurately and adequately apportioned. This article has raised issues relating to how software development project teams are recognized for their contribution and a simple framework for measuring recognition

Table 3. Example of the application of the seven attributes of the IP/IP framework

#	Attribute	Enactment
1	Have an IP acknowledgment strategy	The acknowledgment of IP was never an issue within the project group. Individual contributions were always acknowledged by the core development team, becoming part of the documentation of the project. The extensive documentation ensured that no one was left off the credits on the *An@tomedia* Web site.
2	Have an IP review strategy	The existence of "prior art" was established in the early phases of the project. While the final product did not resemble the initial designs, it was always clear in the project meetings that the team was involved in the instantiation of the educational vision of the project leader (one of the core SMEs).
3	Have a strategy to separate IP from IC	The strategy to separate IP from IC was undertaken by the four principal authors as *An@atomedia* was moved from an interesting project to a commercial product. The documentation resulting from meetings included discussions of commercialization of *An@atomedia* and the associated need to separate IP from IC. The strategy adopted involved consultations with the university's legal advisors and the project team. The IP for the sale and commercial rights to *An@tomedia* were ceded to the four key authors by the other team members; however, the IC remained with members of the project team to use as they required.
4	Longevity strategy: Ideas remain	The credits list contains a list of all members who contributed to the project over a period of many years, including those individuals who either retired (in one case) or moved to other institutions (a number of people). It is possible for all members of the project team to include evidence of contributions to *An@atomedia* by reference to either the Web site or the CD-Roms (the form in which *An@tomedia* is published and sold).
5	Recognize the emergent nature of the software development process and its impact on IC/IP for all team members	The development of *An@tomedia* occurred over a considerable period of time. The genesis of some the clinical approaches adopted in the project were developed by the project leader and occurred well before *An@tomedia* commenced (Eizenberg, 1988, 1991). The use of technology followed as a consequence of the need to develop more effective and engaging approaches to the teaching of anatomy (Driver & Eizenberg, 1995). As the final design of the software emerged, it was always clear in meetings and associated documentation that other members of the project team were involved in the instantiation of prior concepts and developments in new and innovative ways, but the underlying concept derived from the earlier work in paper-based media.
6	Ideas are perishable	*An@atomedia* received a number of awards for innovation and excellence after the first release (see http://www.anatomedia.com/credits.shtml). However, as people come and go from the project, the initial ideas will be superseded or altered to reflect teaching evaluations, changes in the medical curriculum, and improvements in technology. Solutions developed in 1999 or 2000 may not be suitable in 2005. What was once a good idea may not be appropriate in the future, but the three major methods of affirmation remain—publication, portfolio, and vitae for all contributions.
7	Public acknowledgment of IP/IC requires the source material to be in the public domain	The *An@atomedia* Web site provides definitive acknowledgment of the specific contributions of individuals, including the evaluators, programmers, educational consultants, photographers, medical consultants, project managers, dissectors, illustrators, interface and graphic designers, and research assistants, to name a few.

of contribution has been presented. Seven key project attributes or processes have been identified to assist project teams develop an awareness of how project roles and structures can be negotiated so that tacit ideas and knowledge generated can become explicit. Such a model must recognize that the requirements for, and process of, recognition will differ within different multiskilled teams. The application of the framework to one major multimedia project has been discussed.

REFERENCES

Alexander, S., McKenzie, J., & Geissinger, H. (1998). *An evaluation of information technology projects in university learning.* Canberra: Australian Government Publishing Services, Department of Employment, Education and Training and Youth Affairs.

Bannan-Ritland, B. (n.d.). The role of design in research: The integrative learning design framework. *Educational Researcher, 32*(1), 21-24.

Burdman, J. (2000). *Collaborative Web development: Strategies and best practices for Web teams*. Reading, MA: Addison-Wesley.

Davidson, C., & Voss, P. (2001). *Knowledge management: An introduction to creating competitive advantage from intellectual material.* Auckland, NZ: Tandem.

Dick, W., & Carey, L. (1990). *The systematic design of instruction.* Glenview, IL: Foresman/Little.

Dougiamas, M., & Taylor, P. (2003) Moodle: Using learning communities to create an open source course management system. In D. Lassner & C. McNaught (Eds), *ED-MEDIA 2003, Proceedings of the 15th Annual World Conference on Educational Multimedia, Hypermedia and Telecommunications* (pp. 171-178). Honolulu, Hawaii. Norfolk VA: Association for the Advancement of Computers in Education.

Driver, C., & Eizenberg, N. (1995). Constructing and deconstructing the human body: Applying interactive multimedia in the learning of anatomy. In J. M. Pearce, A. Ellis, C. McNaught, & G. Hart (Eds.), *Learning with technology: ASCILITE 95. Proceedings of the 12th Annual Conference of the Australian Society for Computers in Learning in Tertiary Education* (pp. 586-587). The University of Melbourne: The Science Multimedia Teaching Unit.

Eizenberg, N. (1988). Approaches to learning anatomy: Developing a program for pre-clinical medical students. In P. Ramsden (Ed.), *Improving learning: New perspectives* (pp. 178-198). London: Kogan Page.

Eizenberg, N. (1991). Action research in medical education: Improving teaching via investigating learning. In O. Zuber-Skerrit (Ed.), *Action research for change and development* (pp. 179-206). Avebury: Aldershot.

Florida, R. L. (2002). *The rise of the creative class: And how it's transforming work, leisure, community and everyday life.* New York: Basic Books.

Gunn, C. (1995). Useability and beyond: Evaluating educational effectiveness of computer-based learning. In G. Gibbs (Ed.), *Improving student learning through assessment and evaluation* (pp. 168-190). Oxford, UK: Oxford Centre for Staff Development.

Jacobson, I., Booch, G., & Rumbaugh, J. (1998). *The unified software development process.* Reading, MA: Addison-Wesley Longman.

Kennedy, D. M., Eizenberg, N., & Kennedy, G. (2000). An evaluation of the use of multiple perspectives in the design of computer-facilitated learning. *Australian Journal of Educational*

Technology, 16(1), 13-25. Retrieved May 5, 2005, from http://www.ascilite.org.au/ajet/ajet16/kennedy.html

Kennedy, D. M., & Ip, A. (1998). Learning Evaluation Online (LEO): A customizable Web-based evaluation tool. In C. Alvegard (Ed.)*, Computer aided learning and instruction in science and engineering. CALISCE'98 proceedings* (pp. 255-262). Goteborg: Chalmers University of Technology.

Kennedy, G. E., Kennedy, D. M., & Eizenberg, N. (2001). Integrating computer-facilitated learning resources into problem-based learning curricula. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning, 3*(1). Retrieved May 5, 2005, from http://imej.wfu.edu/articles/2001/1/02/index.asp

Macmillan, F. (2000). Intellectual property issues. In C. McNaught, P. Phillips, D. Rossiter, & J. Winn (Eds.), *Developing a framework for a usable and useful inventory of computer-facilitated learning and support materials in Australian universities. Evaluations and Investigations Program report 99/11* (pp. 189-205). Canberra: Higher Education Division Department of Employment, Education, Training and Youth Affairs.

Marušiæ, A. (2004). Media reviews: Interactive anatomy. *The Lancet, 363*(9404), 254.

Phillips, R. A. (1997). *A developer's handbook to interactive multimedia: A practical guide for educational applications.* London: Kogan Page.

Schwalbe, K. (2000). *Information technology project management.* Cambridge, MA: Course Technology.

Stewart, T. (1999). *Intellectual capital: The new wealth of organizations.* New York: Currency.

United Kingdom Patent Office. (n.d.). *What is intellectual property or IP?* Retrieved May 5, 2005, from http://www.intellectual-property.gov.uk/std/faq/question 1.htm

Wenger, E. (1998). *Communities of practice: Learning, meaning and identity.* Cambridge, UK: Cambridge University Press.

White, J. A. D. (1997). Misuse or fair use: That is the software copyright question. *Berkeley Technology Law Journal, 12*(2). Retrieved May 5, 2005, from http://www.law.berkeley.edu/journals/btlj/articles/vol12/White/html/reader.html

Williamson, A., Kennedy, D. M., McNaught, C., & DeSouza, R. (2003). Issues of intellectual capital and intellectual property in educational software development teams. *Australian Journal of Educational Technology, 19*(3), 339-355. Retrieved May 5, 2005, from http://www.ascilite.org.au/ajet/ajet19/williamson.html

Xia, W., & Lee, G. (2004). Grasping the complexity of IS development projects. *Communications of the ACM, 47*(5), 69-74.

This work was previously published in the *Encyclopedia of Communities of Practice in Information and Knowledge Management* edited by E. Coakes and S. Clarke, pp. 364-374, copyright 2006 by Idea Group Reference (an imprint of IGI Global).