Emergent Development of Web-based Education

David Jones, Dr Teresa Lynch and Kieren Jamieson
Central Queensland University, Rockhampton, Australia

d.jones@cqu.edu.au t.lynch@cqu.edu.au k.jamieson@cqu.edu.au

Abstract

There are problems with the adoption and use of many organizational implementations of Web-Based Education (WBE). This paper puts forward the view that the problems stem from a misfit between the development methodologies commonly used to implement these systems and the context in which they are being applied. This misfit contributes to a number of the problems and shortcomings faced with implementing WBE. The paper describes the application and use of emergent development methodologies of WBE and suggests that such methodologies are a better fit for many organizational level developments in WBE. The focus of this paper is WBE within the University sector.

Keywords: WBE, Emergent Development, Development Methodologies

Introduction

Organizational development of WBE within universities usually involves large-scale projects that require changes in existing educational practice through the use of information technology (Jones, 2000). There are problems surrounding the development of emerging technologies like WBE that could contribute to their level of success or failure. Amongst these problems are the fact the systems are continuously evolving, requirements are uncertain, and there are limited funds despite the organizational push to implement the system. This paper suggests that the cause of many of the problems and difficulties with implementing these projects is related to the use of methodologies that are particularly ill suited to the nature of the organization, the context in which it operates, and the nature of WBE. That is, there is a misfit between organizational needs and development approaches.

A response to this misfit between organizational needs and development approach has been a growing interest in the use of emergent (Truex & Klein, 1999) and agile development methodologies (Cockburn, 2001). These methodologies reject many of the fundamental traditional assumptions and aim to provide a better fit with the needs of the organization. This paper proposes that, given the current state of WBE, it is better suited to an approach based on emergent/agile methodologies than traditional development methods. The paper will demonstrate this through experience at the Faculty of Informatics and Communication (Infocom) at Central Queensland University (CQU). The paper begins by discussing the circumstances/characteristics of WBE within Infocom. It then discusses the symptoms and problems evident with traditional development methodologies and WBE development. On the basis of these problems, the paper goes on to describe the rationale for emergent/agile development and the situations in which it is more likely to work. This is illustrated with examples and experience of WBE developments within Infocom. Finally the paper draws some conclusions about how emergent development has been used within Infocom at CQU.

Paper Accepted as a Regular Paper

What is WBE at Infocom?

Infocom is one of five Faculties in CQU. Infocom courses are large complex offerings delivered across multiple campuses, multiple modes, and using many staff spread across these campuses. The Faculty is an eclectic mix of disciplines - including, but not limited to, mathematics, information systems, information technology, multi-media, contemporary communication, and journalism. Students are from a wide cultural spread with over 52% of all Faculty students coming from a non-English speaking background. The Faculty delivers its courses to 13 campuses both within Australian and off-shore, with 30% of students studying via print-based distance education.

Many teaching staff never meet face-to-face or even have voice contact with each other. Instead, staff rely on email as the main mode of communication. This is also true for those students who chose to study in distance mode. Some courses can have up to 30 staff spread over all 13 campuses. In the first half of 2002, Infocom offered 117 under-graduate and post-graduate courses. The largest course had over 1000 students with the average course having around 140 students. The Faculty intent is that course content and quality is the same no matter where it is delivered or how it is delivered, either on-campus or via distance delivery.

The principal use of WBE within Infocom can be classed as the use of Web-based Information Systems (WIS) to support staff and students in the process of education. This is not restricted to traditional teaching and learning process but includes the necessary administrative and support processes that enable teaching and learning. One example of WIS used by Infocom is the management of assignment results and databases that record who is teaching what courses and at which campuses. The complex nature of course delivery across multiple sites presents problems for the Faculty and appropriate information systems are seen as an important solution to the management of the complex processes. WBE within Infocom demonstrates many of the important characteristics of WIS (Gregor, Jones, Lynch, & Plummer, 1999) including: intra and inter-organizational issues; diverse users in terms of experience, geography and culture; reliance on the technologies, standards and tools of the Internet; reliance on hypermedia for its linkage mechanism; and a common Web-based interface.

Whilst CQU provides a range of organizational information systems and related infrastructure, the focus of many of these systems is in fulfilling institution level requirements such as enrolling and billing students, and reporting to government. The specific teaching needs of the staff within the Faculties are rarely considered at an organisational IS level. Faculties within CQU are essentially independent fiefdoms. This in turn allows them some freedom in choosing their own solutions to some problems. This paper discusses one such solution.

Before discussing the emergent/agile approach to software development an examination of the more traditional approach to software development is examined.

Development Methodologies for WBE/WIS

Common institutional approaches to the implementation and support of WBE generally demonstrate the characteristics of traditional information systems development methodologies and projects. Software development methodologies are generally formalized procedures or protocols which have been developed to assist software developers build systems that meet their clients' needs (Gregor et al., 1999).

A standard software development methodologies/project matches the following general steps (Jones, 2000)

1. An analysis and design period intended to identify abstract requirements and generate a formal specification of the new system.

2. A period of construction where the abstract requirements are implemented as an information system. In some cases the construction process is replaced with the purchase of a commercial system followed by modifications to fit local conditions.
3. A sign-off stage where the capabilities of the implemented/purchased system are ticked off against the formal specification.
4. A long period of low-cost, widespread operation and maintenance where the cost of the previous stages is recouped. There may be some low-cost modifications but in some cases this may not be possible.
5. At some stage the features of a system no longer match or support the needs of the organization at which stage there is a return to step 1.

These standard steps are based on and encapsulate the underlying assumptions of traditional software development methodologies. That is, there is a focus on the importance of up front design. It is believed that not only is it possible for a large analysis phase to identify all requirements but that it is compulsory since there is an assumption that making changes later in the lifecycle is more expensive. Consequently the analysis and implementation phases are long and costly. In order to recoup the costs of this investment it is necessary to have long periods of low cost use.

At some stage an organization's needs will have changed sufficiently such that the information system is no longer appropriate and is actually holding the organization back. At this stage the system development bias of traditional approaches means that the cycle returns to step 1.

The problems caused by these assumptions are discussed in the following section.

Symptoms and Problems with Traditional Approaches

Even though there are large number of institutions throughout the world reporting on successful WBE innovations there are also reports of problems with existing approaches to WBE (Jones, 2000). This section briefly describes some of the symptoms suffered by institutional development of WBE and suggests that problems with the traditional development methodologies used to implement WBE are the root cause of these symptoms.

There is a long history of failed technology-based innovations in education (Reeves, 1999). While most innovators pursue change for valid reasons, not simply for the sake of change (Smith, Prunty, Dwyer, & Kleine, 1987), most reforms fail to achieve their goals (Goodlad, 1984). The majority of WBE is badly done (Taylor, 1998) and makes less than effective use of the medium and does not use pedagogical approaches favoured by educational researchers (Mioduser, Nachmias, Oren, & Lahav, 1999). Instead there is a tendency to recreate old practices (Jones, Stewart, & Power, 1999). Morris (1997) observed that instructional innovations are likely to be restricted to imitations of others or of revisions of past technologies, repackaged by academics within or closely associated with the field of education. Many innovations are not institutionalized (Gibbs, 1996) within their home institution and, even less likely, to achieve widespread adoption amongst other institutions (Collis & Oliver, 1999).

The problems are not unique to WBE but are found in all types of IS development. The OASIG (Organizational Aspects Special Interest Group) (1996) study in the United Kingdom, found that around 40% of information technology systems developments failed or were abandoned. Laudon & Laudon (1996) report that up to 75% of all large systems involving change and information technology can be considered to be operating failures.

The issue of how to measure success or failure is not easy as the success or failure of a system is seen as a matter of interpretation and that interpretation can change over time (Myers, 1994). Despite this, in their review of the literature on system success, DeLone and McLean (1992) suggest that there are six

major dimensions or categories of information systems success. These are: system quality, information quality, use, user satisfaction, individual impact, and organizational impact. They suggest that these categories are six interdependent dimensions to information systems success rather than six independent success categories (p.88), and that information systems success is a multidimensional construct and should be measured as such. The symptoms of WBE problems addressed above fit in with these dimensions to IS success.

Research suggests that part of the problem contributing to the high failure rates may be related to software development methodologies. It has been suggested (Fitzgerald, 1998; Grudin, 1991; Russo & Stolterman, 1998) that many developers currently do not even follow software development methodologies when developing software systems.

It has been suggested that the traditional software development methodologies are not good matches for (1) the context in which systems are developed, (2) the needs of the developers and (3) the needs and participation of the clients.

The Context in which Systems are Developed

As stated previously, the assumption behind the traditional approach to software development is an underlying belief that not only is it possible for a large analysis phase to identify all requirements but that it is compulsory. Consequently the focus is on getting it right first time partly because of the assumption that changes that are made later in the process will cost more to implement and such changes are more likely to result in further problems and may even lead to failure. This approach makes it difficult when working in an uncertain world where experimentation, learning from mistakes, and deviating from a plan are often necessary ingredients to a successful outcome. The approach also assumes that the analyst or developer can anticipate everything and that if the plan is followed then success will follow also.

The world in which systems are developed is not static and entirely predictable - systems will need to be altered and maintained. Maintenance typically consumes about 40 to 80 percent of software costs and 60% of that maintenance cost is due to enhancement (Glass, 2001). That is, adding new capabilities that were not known of during the analysis phase. If maintenance is such a large part of system development the assumption of a period of low-cost maintenance to recoup costs from the analysis and development phases seems less valid. If an organization is operating in a continually changing context then a large investment in up front analysis is a poor investment as requirements change before the end of the analysis stage (Truex & Klein, 1999).

Organizations are surrounded by uncertainty and continual change, experimentation is often a good way forward, and deviation from a plan is often necessary and failure to deviate could result in a perfect system but one that was out of date. The rapid development of technology and global markets contributes to a need for constant change where organizations are no longer stable and must continuously adapt to their shifting environments (Truex & Klein, 1999).

In addition to the conceptual nature of the systems, there are often inherent risks and uncertainties associated with projects that are difficult to assess with any degree of reliability prior to the start of the projects. These risks may include the large size of the project, complexity of the problem domain, project members, unfamiliarity with new technology, unstable information requirements, and difficulties in integrating different components into a composite system (Ewusi-Mensah, 1997). All these factors can help contribute to failure of system completion and adoption.

The incredible variation of the needs, requirements, and tastes of students and teachers means that there is no one correct method for implementing a Web-based course (McCormack & Jones, 1997). The ability for teachers to customize an online course to identify their character, personality and teaching commitment can be vital to self-esteem and commitment (Brown, 1999). An online course and the informa-

tion system that enables it will need to handle change in response to changes in technology, significant differences across disciplines (Brown, 1999), cultural, pedagogical and organizational issues.

Balasubramanian & Bashian (1998) reported that political and cultural issues can confound the best laid plans for WIS projects, "Despite all our systematic efforts, our initial perception is the system is now less-than-successful due to non-technical factors" (p. 114). Romm & Wong (1998) noted that "The introduction of a Web technologies project to an organization is a far more complex process than is commonly assumed". Complexity that can be attributed to the political, organizational and culture variables involved.

We argue that the context in which the system is developed and used should influence the approach used for development.

The Needs of the Developers

Russo & Stolterman (1998) suggested that many developers currently do not even follow software development methodologies when developing software systems. They suggest that there is currently a 'misfit' between existing methodologies and the needs of developers. They believed that this is due to the changing nature of the types of systems developed. In his study determining the extent that developers currently use software development methodologies, Fitzgerald (1998) found that 60 percent of the respondents were not using a methodology. He commented that the general view was that methodologies are cumbersome and consume time and resources that were not always available. Like Grudin (1991), he suggested that many current methodologies are derived from practices and concepts relevant to older organizational environments and there is a need to reconsider their role in today's environment. There is a need for more rapid system delivery than that which is currently being achieved.

The older, more technological focused, development methods do not appear to be suitable for the current development environment. Many of the older, more established methodologies have a developer-focus. This approach assumes that a new product, either technical or non technical, will automatically replace inferior products or systems (Surry & Farquhar, 1997). Developers assume potential adopters will see the benefits just as they see them. The developer-based theories assume that there is no need to adapt the technology to the requirements of the people - rather everyone will adapt to the requirements of the technology. Davis et al (1992) determined that this type of attitude, in part, lead to the failure of an undemanding and elementary information system.

The Needs and Participation of the Clients

In contrast to the developer-focused approach, the adopter-focused approach focuses on the human, social, and interpersonal aspects of innovation diffusion (Surry & Farquhar, 1997). Developers are interested in the individual who will ultimately implement the innovation in a practical setting as the primary force for change. The adopter-based theories reject the assumption that superior products will automatically be attractive to potential adopters. They 'seek to understand the social context in which the innovation will be used and the social function the innovation will serve' (Surry & Farquhar, 1997). The adopter-focused approach emphasizes the importance of user involvement in the design and development process.

Grudin (1991) raised the concern that many of the software development methodologies currently available were developed before interactive end user applications became important. Because of this they do not provide for an early and continual focus on users - quite the contrary. He suggested that the traditional structured analysis approach actually relegates the task of establishing a 'man-machine interface' to one sub-phase of system development. Not only are users not consulted but also 'The designers of countless failed products anticipated user populations that did not materialize' (Grudin, 1991 p. 60). Intuition, he added, has become a less reliable guide to development. While contact with system users is

required he also acknowledged that determining how direct or extensive this contact need be and actually achieving it has been surprisingly difficult. However, for today's interactive systems developer, the reliance on specification documents imposes a 'wall' between users and developers that may impede user-based iterative design. Furthermore, developers who are isolated in large engineering laboratories may neither empathize nor sympathize with users who are inexperienced, non-technical, or have different values and work styles.

Brooks (1995) suggests that "software is pure thought-stuff, infinitely malleable as well as invisible and unvisualizable". Through user-participation, prototyping, and the release of early systems often users have something concrete to help them in the visualization of an abstract concept. The issue then is how to develop systems that evolve as the organization evolves and better represents the requirements of the clients. The paper now looks at the use of an alternative approach to software development in WBE.

Emergent Development of WBE

WBE within Infocom started with initial experiments by individual academics in 1994. The first "organizational" approach to WBE started in 1996 when one of the authors was relieved of teaching duties for a term to "develop some web stuff". Since then organizational approaches to WBE within Infocom has had little or not strategic direction from Infocom management and has often operated in spite of CQU organizational approaches. The development methodology used within Infocom since 1996 is best characterized as emergent development. Recently aspects of an agile development methodology, eXtreme Programming (Beck, 2000), has been adopted to add further rigor and structure to the emergent development approach.

This section describes the assumptions and goals that underlie emergent and agile development and how they differ from traditional approaches. Drawing on Infocom's use of these methodologies to develop and support an organizational approach to WBE this section will also illustrate how these goals work in action.

Emerging, Emergent and Goals

Emerging technologies like WBE threaten the status quo although there still exists the possibility that eventually the changes will result in a new possibility that will become accepted as a new status quo. Emerging is different than emergent. In emergent organizations, every feature of the organization, culture, meaning, social relationships, decision processes and so on, are continually changing as a product of constant social negotiation and consensus building (Truex & Klein, 1999). That is, change never ceases instead change enables more change. The aims and assumptions of traditional information systems development methodologies, the production of systems with long periods of stable operation, are ill suited to an emergent organization and will create long term problems (Jones, 2000).

Emergent development methodologies attempt to cope with the continual change within an emergent organization. As a result it needs to adopt a different set of goals than those of traditional methods. The new goals for emergent development include: continual analysis, dynamic requirements negotiations, incomplete but useful specifications, continuous redevelopment, and the ability to adapt (Truex & Klein, 1999). The following sections explain these goals in more detail and illustrate them with examples from the development of WBE within Infocom.

Continual Analysis

With stable organizations and applications that have low volatility in requirements it is possible for stable, precisely designed systems to satisfactorily operate with minimal changes for long periods (Truex & Klein, 1999). Consequently, a large initial investment in analysis, design and construction is followed by recoupment during a long low-maintenance phase. During this maintenance phase little or no analysis

occurs to feed into system changes. With an emergent organization and an application area like WBE this large initial investment in an analysis and design phase is wasted. By the time such a phase is completed the organization or desired application of the technology has moved on. Analysis is no longer a part of a specific project but an ongoing part of the maintenance activity (Truex & Klein, 1999).

Within Infocom, continual analysis is implemented by a combination of methods intended to increase awareness of the requirements of the system clients and context. Analysis of factors external to CQU involves reading and making contributions to the literature. Analysis of factors external to Infocom but internal to CQU involve participation in committees but more importantly the development of close and wide-ranging personal contacts with people throughout the institution. Analysis of factors internal to Infocom is performed through being an integral part of the Faculty rather than being seen as outsiders; observation and reflection on the work of others; and through the use of other more formal means such as surveys and interviews.

In 1999 a WBE innovation, course barometers (Svensson, Andersson, Gadd, & Johnsson, 1999), were observed at a conference. The features of this innovation provided possible solutions to some of the problems being faced within Infocom and so was implemented on the Infocom website. Over the period of three years initial ad hoc use grew to become more systematic. However, there were on-going problems with the level of participation of students. This problem was analyzed through the use of student surveys and identified that the primary cause for the low level of student participation was the student perception that their contributions to the course barometers generated little visible change (Jones, 2002). Consequently steps were taken to address these problems. As a result of these changes further issues have been identified which in turn will be addressed.

As well as being driven by external innovation or internal problems, other WBE innovations implemented within Infocom have come from observing interesting and useful practices of individual staff. Practices that are particularly useful have been adopted and integrated into the Infocom organizational web system for use by all staff.

Dynamic Requirements Negotiation

The upfront analysis phase of traditional development methodologies produces a static requirements document that summarizes all the needs of the users. The intent is that a sufficiently correct requirements document will enable a system to be built that will satisfy all users and be checked off by the organization at the end of the development. However, the length of time taken to generate the requirements document and build such a system means that, within an emergent organization, the requirements have changed by the time construction is complete.

Within emergent organizations the needs of users unfold rapidly and in directions that are poorly understood by the users themselves (Truex & Klein, 1999). A stable system that cannot be changed will frustrate and anger users as it fails to deliver the services they require. Emergent development relies on a degree of conflict between users and the information system to drive requirements generation. As users interact with the information system the shortcomings of the system identify new requirements to be implemented.

One of the most successful applications of WBE within Infocom is online assignment submission and management. In 2000, this system provided a simple management interface that allowed a single staff member to collect and mark student assignments. As student numbers grew this limited interface began creating large amounts of work. This generated a requirement to provide a management interface that supports multiple staff distributed across multiple campuses.

At the same time, changes in the student cohort increased staff perception that plagiarism amongst students was growing at an increasing rate. Initial staff requirements have been for systems to aid in the detection and management of plagiarism cases. However, as staff experience and discussion grows, staff

are realizing that this is treating the symptoms and not the cause (Jones & Behrens, 2003). Consequently there is now a growing requirement amongst staff for information systems that support alternate assessment schemes that are more likely to avoid plagiarism.

Incomplete but Useful Specifications

Complete and unambiguous specifications are only possible in an organization that is not changing. Developing specifications within an emergent organization becomes little more than a history lesson in past organizational states or abstractions of guesswork about future organizational states (Truex & Klein, 1999). Rather than waste resources on less than useful complete specifications, emergent development is based on usefully incomplete specifications that can be easily adapted to changing requirements and ongoing emergence.

The Infocom web team does not maintain separate system specifications, documentation, and implementation. Instead the specifications, documentation and implementation are combined and automation is used to aid in the generation and maintenance of this documentation.

Continuous Redevelopment

Traditional software development methodologies have a replacement-mentality, a hold-over from the time when computer-based systems replaced manual systems, where each information system has a limited lifespan and must be replaced when it no longer meets the need of the organization (Truex & Klein, 1999). In emergent development an information system should never become obsolete as it is being continuously redeveloped to meet the changing needs of the organization.

The information system used to implement the Infocom website goes under the label Webfuse. Webfuse, is not a single piece of software, instead it is an eclectic collection of software from various commercial and non-commercial sources, integrated into a single common system and interface through the use of a structure based on hypermedia templates (Jones, 1999). First designed and developed in 1996 (Jones & Buchanan, 1996), Webfuse has gone through a period of continuous redevelopment to meet the changing needs of Infocom.

For example, Webfuse's authentication and access control facility is used to identify visitors to the website and what actions they are allowed to do, started out as a very simple group-based system that relied on a database of Webfuse specific accounts (usernames/passwords). Since that time this facility has gone through a period of continuous redevelopment in response to changes in requirements and context. The changes resulting from this continuous redevelopment includes:

- Addition of student accounts.
  Access was gained to CQU's student records system, which included a unique username/password for each student and access to student enrolment data. This enables website areas to be restricted to students enrolled in a particular course.
- Advent of CQU domain accounts.
  In the late 1990s CQU implemented central infrastructure that provided all staff with a single unique username/password. Webfuse was modified to use these accounts. As a result the number of passwords staff needed to remember was reduced. Additionally, the workload for the Infocom web team was significantly reduced since account management was taken over centrally.
- Addition of role-based access control.
  As Infocom's supporting information systems became more refined a stage was reached where information about which Infocom staff were teaching which courses was being stored in databases. At this time the Webfuse access control mechanism was changed to support role-based access control. Though significantly different, this scheme allowed for increased flexibility and reduced

work. Role based control allocates a set of standard permissions to specific roles. For example, a course coordinator automatically has a range of permissions associated with that course.

- Change in CQU student records system.
  In 2001 CQU adopted a new student records system. This system used different course numbers, different database schemas and different technology. The student account system used by Webfuse was modified to use a combination of the old system and the new system. This was done to reduce the amount of risk during the change over process.
- Adding locally cached student information.
  Even with this duplication of student data, the change over process was difficult, especially due to the fact that the new student record database was unavailable for long periods of time. This caused significant problems for the Infocom website which relied on this database to grant access to services. To address this problem a local cache of the data, independent of the central system, was created. The system now uses this local cache first and if there are any discrepancies will attempt to use the central student records system (if available).

All of these changes have occurred without any change from the user's perspective and limited change in the programming interface.

The Ability to Adapt

In emergent information systems development, the information systems infrastructure must undergo continuous redevelopment. In order to support this continuous redevelopment of the information systems infrastructure, development approaches and organization must be able to be easily adapted to dynamic requirements. The underlying assumption of traditional development methodologies, that changing a system is more expensive the later in the life-cycle it occurs, is a hang-over from the past where the technology being used was at a lower level and knowledge about constructing flexible and maintainable systems was significantly less.

The Infocom web team makes use of a number of practices and schemes to increase its ability to adapt quickly to the changing requirements of the organization. These include 1) design patterns, 2) scripting languages, 3) open source software, 4) hypermedia templates, and 5) eXtreme Programming. A brief description of how these practices and schemes are incorporated in Webfuse follows.

A design pattern is ‘a generic approach to solving a particular problem that can be tailored to specific cases. Properly used, they can save time and improve quality’ (Fernandez, 1998). The use of patterns provide a number of benefits including making it easier to reuse successful designs, make proven techniques more accessible to developers, enable choice between alternatives, and improve the documentation and maintenance of existing systems (Gamma, Helm, Johnson, & Vlissides, 1993).

Administration of a Web site, requires the constant development of small tools, scripts, and "glue applications" to make everything work together (O'Reilly, 1999). Scripting languages allow very rapid development of applications via the gluing together of existing applications, 5 to 10 times faster than through the use of traditional systems programming languages (Ousterhoust, 1998). Webfuse (Jones, 1999) makes use of hypermedia templates and the Perl scripting language to enable the gluing together of disparate applications, many of it open source software, into a standard infrastructure and interface.

This approach means that the focus of the Infocom web team is not on developing its own version of application software. Instead it attempts to identify and integrate software development by other people and organizations into the interface provided by Webfuse. For example, since 1996 three different web-based bulletin board systems have been integrated into Webfuse. Each replacement system offering increased functionality that better suited the changing requirements of Infocom staff.

eXtreme programming (Beck, 2000) is one of a number of development methodologies that fall under the term agile development (Cockburn, 2001). eXtreme programming is a highly disciplined approach to developing software within a context where requirements are changing rapidly. It offers a range of interlocking practices that enable the disciplined development of emergent systems.

Does it Work?

As mentioned previously, measuring the success or failure of an information system is a difficult task due in large to its subjective nature and tendency to change as time progresses. Measuring success in an emergent organization is harder again since in part emergent development relies on user dissatisfaction to drive on-going development. It is possible to suggest that the Infocom emergent approach to the development of WBE is successful on each of the six major dimensions of information systems success identified by DeLone & McLean (1992): system quality, information quality, use, user satisfaction, individual impact, and organizational impact. This section offers only some brief examples of this.

During the 2002 the small, four person, Infocom web team supported the creation and maintenance of course websites for 331 different course offerings. The online quiz facility was used by 4227 individual students to make over 36,000 attempts at quizzes in 20 different course offerings. 28 different course offerings used online assignment submission and marking where 3989 individual students submitted and received assignments. 407 individual teaching staff made use of these services to support their teaching and learning. The weekly average requests for the Infocom website increased from 1.3 million per week in the first major term of 2002 to 1.6 million per week in the second major term.

In the first half of 2002 the Infocom web development process developed websites for over 100 Infocom courses. These default course websites, first developed in late 2001, are now accepted by students and staff as a standard part of Infocom's operations. During early 2002 over 1700 students in 13 of these courses used online assignment submission via the Infocom website. At the same time over 1900 students in 14 courses used the Infocom website for online quizzes. During this period the Infocom website was averaging around 1.3 million file requests each week and over 100 Infocom staff made modifications to material on the site.

In 2001 a system for the uploading of final results into CQU's student records system was developed in response to serious shortcomings of the centralised student records system provided by the university. The Infocom system has proven to be so useful, and the central system so difficult, that parts of the Infocom system are being institutionalized at the request of the Dean's of the other CQU faculties.

Discussion

The work performed by the Infocom web development team did not involve a large-scale, up-front, analysis and requirements cycle. There has never been any strategic direction given to the web team by Infocom management. Instead the web team has identified what and when it develops services based on its on-going analysis and interpretation of users' needs. While this approach has been reasonably successful there have been problems where the requirements of some staff have not been fully achieved. This has become increasingly common as the size, variety and geographic spread of Infocom staff grows. Recently organizational funding of this team has started to catch up with the demands allowing the team to adopt practices that were previously not possible.

A major enabler of the emergent development approach adopted by the Infocom web team has been the support of the Infocom Dean. For most of its recent history Infocom has been led by a Dean who is a firm believer that Infocom is an emergent organization and that its information systems must support this. With a different type of person in this position the approach discussed here would not have been possible. Even with the support of the Dean it is possible that central CQU initiatives may overwhelm the work of the Infocom web team.

Some people react to emergent and agile development methodologies with the retort that it is a return to the "lone-ranger" undisciplined development that characterized much early information systems development and much early use of WBE. Nothing could be further from the truth. Done well, emergent/agile development methodologies require significantly greater discipline than traditional methods. The rejection of traditional measures of discipline, e.g. voluminous project documentation, does not indicate a lack of discipline. Rather it indicates that emergent/agile development methodologies value different measures of discipline: measures which proponents of these methodologies believe are better suited to the context in which they work.

By its very nature emergent development is never complete. There are always changes in requirement or capability that require additional work. However, there are still some common problems that need to be addressed. Within Infocom there is a need for greater agreement that the Faculty is an emergent organization and recognition that such an organization has different requirements. Many staff are yet to understand this distinction and thus have problems appreciating certain characteristics of the Infocom web team development approach.

A more important problem yet to be addressed within Infocom is one identified by Green (Green, 1999). Most campuses in the United States have IT development programs (75.8 percent) and campus support centers (65.9 percent) to assist faculty in bringing technology resources into their courses. However, just one-seventh (13.7 percent) of colleges and universities have a formal, institutional program to recognize and reward the use of information technology as part of the faculty review process. The 1999 data show little change on these measures over the past few years.

When it comes to information systems development there is no silver bullet. There is no approach that works equally well in all situations and contexts. Instead it is up to the organizations participants to identify and adopt development methodologies that suit their organizations unique needs and capabilities. Certain characteristics of the Infocom situation including: enormous growth, presence of academics who are software developers, sufficient but not overwhelming resources, a supportive Dean, and organizational independence amongst other factors means that emergent development is a good fit for Infocom. Common characteristics of Universities and WBE mean that emergent development may be a good fit for other Universities examining ways to implement organizational WBE.

Conclusion

In this paper we have suggested that many of problems with the organizational implementation of Web-Based Education (WBE) within Universities can be blamed on the misfit between the development methodologies being used and the nature of the organization. We have shown how traditional development methodologies are based on assumptions and goals that are inappropriate for an emergent organization. We have shown that information systems development methodologies for emergent organizations are based on a different set of assumptions and goals and illustrated this difference by relating our experience since 1996 with emergent development within the Faculty of Informatics and Communication at Central Queensland University. Our experience has shown that emergent development can offer a valuable alternative to traditional methodologies but like all approaches is not without its challenges.

References

Balasubramanian, V., & Bashian, A. (1998). Document management and web technologies: Alice marries the mad hatter. *Communications of the ACM, 41*(7), 107-115.

Beck, K. (2000). *Extreme Programming Explained: Embrace Change*: Addison-Wesley.

Brooks, F. (1995). *The mythical man-month: Essays of software engineering, Anniversary Edition* (2 ed.): Addision-Wesley.

Brown, C. (1999). *From the What and Why to the How of Course Support Systems - The Value of the Teacher's Perspective Provided by These Systems.* Paper presented at the WWW-Based Course Support Systems Seminar, EdMedia'99, Seattle, Washington.

Cockburn, A. (2001). *Agile Software Development*: Addison-Wesley.

Collis, B., & Oliver, R. (1999). *Preface.* Paper presented at the Ed-Media, Seattle, Washington.

Davis, G. B., Lee, A. S., Nickles, K. R., Chatterjee, S., Hartung, R., & Wu, Y. (1992). Diagnosis of an information system failure. A framework and interpretive process. *Information and Management, 23*, 293-318.

DeLone, W. H., & McLean, E. R. (1992). Information systems success: The quest for the dependent variable. *Information Systems Research, 3*(1), 60-95.

Ewusi-Mensah, K. (1997). Critical issues in abandoned information systems development projects. *Communications of the ACM, 40*(9), 74-80.

Fernandez, E. B. (1998). *Building Systems Using Analysis Patterns.* Paper presented at the Third International Workshop on Software Architecture, Orlando, Florida.

Fitzgerald, B. (1998). An empirical investigation into the adoption of systems development methodologies. *Information and Management, 34*(6), 317-328.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1993). *Design Patterns: Elements of Object-Oriented Software.* Reading, Massachusetts: Addison-Wesley.

Gibbs, G. (1996). Institutional strategies for implementing resource-based learning. In S. Brown & B. Smith (Eds.), *Resource Based Learning* (pp. 10-21). London: Kogan Page.

Glass, R. (2001). Frequently Forgotten Fundamental Facts about Software Engineering. *IEEE Software*, 110-112.

Goodlad, J. I. (1984). *A place called school: Prospects for the future*. New York: McGraw-Hill.

Green, K. (1999). *The 1999 Campus Computing Survey*.

Gregor, S., Jones, D., Lynch, T., & Plummer, A. A. (1999). *Web Information Systems Development: Some Neglected Aspects.* Paper presented at the Proceedings of the International Business Association Conference, Cancun, Mexico.

Grudin, J. (1991). Interactive Systems: Bridging the Gaps between Developers and Users. *IEEE Computer, 24*(4), 59-69.

Jones, D. (1999). *Webfuse: An Integrated, Eclectic Web Authoring Tool.* Paper presented at the Proceedings of EdMedia'99, World Conference on Educational Multimedia, Hypermedia & Telecommunications, Seattle.

Jones, D. (2000). *Emergent Development and the Virtual University.* Paper presented at the Learning'2000, Roanoke, Virginia.

Jones, D. (2002). *Student Feedback, Anonymity, Observable Change and Course Barometers.* Paper presented at the Proceedings of EdMedia'2002, Denver, Colorado.

Jones, D., & Behrens, S. (2003). *Online Assignment Management: An Evolutionary Tale.* Paper presented at the Proceedings of HICSS'36, Hawaii.

Jones, D., & Buchanan, R. (1996). *The Design of an Integrated Online Learning Environment.* Paper presented at the Proceedings of ASCILITE'96, Adelaide.

Jones, D., Stewart, S., & Power, L. (1999). *Patterns: Using Proven Experience to Develop Online Learning.* Paper presented at the Proceedings of ASCILITE'99, Brisbane.

Laudon, K. C., & Laudon, J. P. (1996). *Management Information Systems*. New Jersey: Prentice-Hall.

McCormack, C., & Jones, D. (1997). *Building a Web-Based Education System.* New York: John Wiley & Sons.

Mioduser, D., Nachmias, R., Oren, A., & Lahav, O. (1999). *Web-Based learning environments (WBLE): Current state and emerging trends.* Paper presented at the Ed-Media, Seattle, Washington.

Morris, D. R. (1997). Adrift in a sea of innovations: A response to Alexander, Murphy and Woods. *Educational Researcher, 22*, 22-26.

Myers, M. (1994). A disaster for everyone to see: An interpretive analysis of a failed IS project. *Accounting, Management and Information Technology, 4*(4), 185-201.

OASIG. (1996). *The performance of information technology and the role of human and organizational factors* (Report). Sheffield: University of Sheffield.

O'Reilly, T. (1999). Lessons from open-source software development. *Communications of the ACM, 42*(4), 32-37.

Ousterhoust, J. (1998). Scripting: Higher Level Programming for the 21st Century. *IEEE Computer, 31*(3), 23-30.

Reeves, T. (1999). *A Research Agenda for Interactive Learning in the New Millennium.* Paper presented at the Proceedings of EdMedia'99, Seattle, Washington.

Romm, C. T., & Wong, J. (1998). The dynamics of establishing organizational web sites: Some puzzling findings. *Australian Journal of Information Systems, 5*(2), 60-68.

Russo, N. L., & Stolterman, E. (1998). *Uncovering the assumptions behind information systems methodologies: Implications for research and practice.* Paper presented at the 6th European Conference on Information Systems, Aix-en-provence, France.

Smith, L. M., Prunty, J. P., Dwyer, D. C., & Kleine, P. F. (1987). *The fate of an innovative school*. London: Falmer Press.

Surry, D. W., & Farquhar, J. D. (1997). Diffusion Theory and Instruction Technology. *Journal of Instructional Science and Technology, 2*(1), 269-278.

Svensson, L., Andersson, R., Gadd, M., & Johnsson, A. (1999). *Course-Barometer: Compensating for the loss of informal feedback in distance education.* Paper presented at the EdMedia'99, Seattle, Washington.

Taylor, P. (1998). Institutional Change in Uncertain Times: Lone Ranging is Not Enough. *Studies in Higher Education, 23*(3), 269-278.

Truex, D., & Klein, B. (1999). Growing Systems in Emergent Organizations. *Communications of the ACM, 42*(8), 117-123.

Biographies

David Jones is a Senior Lecturer with the Faculty of Informatics and Communication at Central Queensland University. For the last two years his primary responsibility has been supporting Faculty staff in the use of online teaching and learning. His research interests are mainly focused around the problem of implementing and supporting information systems in organizations, in particular those suitable for online education in tertiary environments.

Teresa Lynch is Head of the Computing and Information Systems School in the Faculty of Informatics and Communication at Central Queensland University. Teresa's research interests include the adoption of information technology in developing countries, the adoption and use of intelligent support systems in agriculture, and user involvement in system development. Teresa has also published in the areas of participation of women in tertiary information technology courses, computer mediated communication and computer based learning.

Kieren Jamieson is a Lecturer in the school of Computing and Information Systems, Faculty of Informatics and Communication at Central Queensland University. He has extensive experience in the development of Web-based Information Systems as part of Corporate Information Systems as well as a long association as a user and some-time developer of the Faculty's online systems. His research interests are in Information System adoption, selection and implementation.