

2004

The Utility of a Rapid Application Development (RAD) Approach for a Large Complex Information Systems Development

Hilary Berger

University of Wales, Hberger@uwic.ac.uk

Paul Beynon-Davies

University of Wales - Swansea, p.beynon-davies@swan.ac.uk

Pat Cleary

University of Wales, pmcleary@uwic.ac.uk

Follow this and additional works at: <http://aisel.aisnet.org/ecis2004>

Recommended Citation

Berger, Hilary; Beynon-Davies, Paul; and Cleary, Pat, "The Utility of a Rapid Application Development (RAD) Approach for a Large Complex Information Systems Development" (2004). *ECIS 2004 Proceedings*. 7.

<http://aisel.aisnet.org/ecis2004/7>

This material is brought to you by the European Conference on Information Systems (ECIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ECIS 2004 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

THE UTILITY OF A RAPID APPLICATION DEVELOPMENT (RAD) APPROACH FOR A LARGE COMPLEX INFORMATION SYSTEMS DEVELOPMENT

Berger, Hilary, University of Wales Institute Cardiff, Business School, Colchester Avenue, Cardiff, CF23 9XR, UK, Hberger@uwic.ac.uk

Beynon-Davies, Paul, University of Wales Swansea, EBMS, Singleton Park, Swansea, SA2 8PP, UK, p.beynon-davies@swan.ac.uk

Cleary, Pat, University of Wales Institute Cardiff, Business School, Colchester Avenue, Cardiff, CF23 9XR, UK, PMCleary@uwic.ac.uk

Abstract

Rapid Application Development (RAD) as a development methodology has its origins based within the commercial arena. As a result individual philosophies and perceptions of its rationale and applicability have led to considerable debate about its appropriateness for large complex Information Systems (IS) development. Even though RAD is becoming an increasingly accepted approach to IS development, existing literature does little to clarify the position and continues to question its suitability for large complex development projects.

Contrary to published beliefs, a RAD type approach is being adopted for a large complex IS that is currently being implemented within UK Regional Government. This paper describes the case study that presents an interesting and atypical opportunity to examine the use of RAD within such a complex development environment.

This research adopts an interpretive approach using an ethnographic style of qualitative research that literature posits has been effectively used for the study of information systems. It looks at the application of the development approach, considers the problems identified with such an approach and highlights the issues that impact and impinge upon the utility of RAD for such milieux.

keywords: RAD, Utility, Information Systems Development, Development Methodology, Case Study.

1 INTRODUCTION

The nascent status of RAD as a development approach and its original commercial emphasis has resulted in individual philosophies and perceptions of its rationale and application that have led to considerable debate about its suitability as a development approach for large complex Information Systems (IS). Existing literature does little to clarify the debate and posits that the lack of academic research in this area identifies a need for further evaluation of RAD type projects

A large, complex Information Systems Project adopting a RAD type development approach that is currently being implemented within UK Regional Government is being used as an on-going case study for this research in progress paper.

This paper provides an analytical viewpoint of RAD as a development approach and presents an understanding of the research problem through a literature review of the current status of the debate concerning the suitability of RAD within complex environments. It describes the research methodology adopted, explains both the Organisational and IS Project contexts involved, and gives an overview of the system being developed.

The outsourced developers own in-house commercial Iterative Application Development (IAD) approach is defined and aligned against particular features associated with RAD to identify which elements have been implemented. It discusses those that have impacted and impinged upon the development process, whether positively or negatively. In conclusion it presents some preliminary analysis that enable the Researcher to respond to some of the views and opinions expressed in the literature reviewed.

2 WHAT IS RAPID APPLICATION DEVELOPMENT (RAD)?

RAD originated from rapid prototyping approaches and was first formalised by James Martin (1991), who believed that it refers to a development life cycle designed for high quality systems with faster development and lower costs than the traditional lifecycle provided. By the mid 1990s the definition of RAD became used as an umbrella term to encompass a number of methods, techniques and tools by many different vendors applying their own interpretation and approach. This unstructured and extemporised ad hoc evolution of RAD means that the rationale behind its use is not always clear. It is perceived as an IS system methodology, a method for developers to change their development processes or as RAD tools to improve development capabilities. (Beynon-Davies 1999).

Literature reports that RAD centres on prototyping and user involvement where the analysis, design, build and test phases of the development life cycle are compressed into a sequence of short, iterative development cycles. This was seen as a remedy to perceived flaws with the traditional lifecycle because the iterative approach encourages effectiveness and self-correcting as each increment is refined and improved. It necessitates the collaboration of small and diverse teams of developers, end users and other stakeholders (Martin 1991, Tudhope 2001, Beynon-Davies 1996, Elliott 1997). The public domain RAD standard is the DSDM (Dynamic Systems Development Method, see section 9) but the specific method described in the context of this paper is Iterative Application Development (IAD) a vendor specific method (Described in Section 3).

RAD projects are sometimes distinguished in terms of intensive and non-intensive forms. A non-intensive approach refers to projects where system development is spread over a number of months involving incremental delivery compared to the intensive RAD where project personnel are closeted away to achieve set objectives with a 3 - 6 week timeframe (Beynon-Davies 1999). The case study concerns a non-intensive approach that is considered to be more adaptable for larger projects because development can be organised into separate blocks for incremental development and phased delivery.

3 ITERATIVE APPLICATION DEVELOPMENT METHODOLOGY

This section describes the development method used, the Researcher's aim is to emphasise aspects of the development process that are distinctive as far as RAD is concerned not to focus on the product.

The developers adopted their own in-house commercial Iterative Application Development (IAD) approach to promote a controlled, structured but flexible development methodology aimed at providing incremental delivery. This involved a series of time-boxed mini iterations and a number of software 'release' and test iterations to provide flexibility to meet the recognised volatile needs of the business environment. They believe this methodology offers all the main benefits of a RAD type approach and is suited to the uncertainty of, and continually changing business requirements. IAD, like RAD, involves prototyping and iterative delivery but without the problems of lack of rigour, creeping scope and overrun that are perceived as associated with RAD and an iterative development life cycle. Its similarity is extended to using the same main features i.e. JAD (Joint Application Design) workshops, time-boxing, prototyping, intensive user involvement, iterative development and incremental delivery, which they maintain are increasingly used for system functionality development.

The developers believe that a major benefit of an iterative approach to development is that it affords early visibility of the system being developed. Thus early validation of the system by the users and the business analysts provides the flexibility to incorporate user feedback and handle any new or changing requirements within the volatile business environment – a key goal of the RAD approach.

4 RAD - THE DEBATE

As a systems development approach RAD has both critics and supporters whose opinions, in some cases, are fundamental to individual philosophies and perceptions of its rationale. Existing literature exposes particular themes of discussion within the RAD arena and a prominent area of debate concerns the scalability of RAD across large and complex environments. Although the lack of provenance is reflected by the limited availability of published material, there is substantial reporting of its application and considerable debate about its appropriateness for different types and sizes of systems development. (Osborn 1995, Beynon-Davies 1999, 2000). RADs origins as a development process can be placed more within a commercial development arena than an academic one and literature considers it more appropriate for small to medium simple, highly interactive development projects rather than for environments that are also computationally complex such as the case study involved. It is further thought that its success is linked to the project management approach, level of management commitment, degree of end-user involvement and the ability of the team to make fast authoritative decisions (Beynon-Davies 1998). It is also suggested that RAD projects necessitate cultural and managerial changes because people are required to behave in a different way than in the more structured traditional environments. Consequently without radical shifts in organisational attitudes and structures and peoples' mindsets many projects may fail because the change to new methodologies, methods and techniques did not fit within the culture (Hirschberg 1998, McConnell 1996). Hence the potential of a RAD development and delivery approach to meet information systems requirements in uncertain and volatile business settings of complex system development environments is questioned. Critics advocate that the need for high levels of user involvement, stakeholder collaboration, lack of project control and rigour are major issues to its success (Martin 1991, Osborn 1995, Beynon-Davies 1996, 2000, Elliott 1997, Cross 1998, Boehm 1999, Highsmith 2000).

5 ORGANISATIONAL CONTEXT

In 1999 under the UK Government's Devolution legislation a UK Regional Government department took on the devolved functions formally carried out by the Welsh Office. It became responsible for managing the expenditure of EC grants and subsidies to customers through a number of Common

Agricultural Policy (CAP) schemes across the region. Headquarters is centrally located and operations run from a number of Divisional and Area Offices acting as powerhouses of management functions. The customer base consists of farmers, farming businesses, and other citizens.

The case study concerns the development of the new IT system aimed at improving the current effectiveness and efficiency of the agricultural grants and subsidy administration. The system moves away from the former discrete individual scheme administration processes towards a Generic Process (Figure 1.) that integrates the core processing of the common activities of the separate schemes.

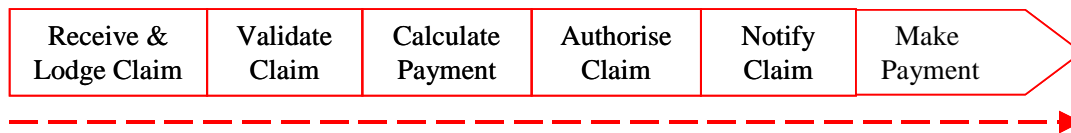


Figure 1. *New Scheme Generic Process.*

Each scheme details the EC rules and conditions that apply, however the schemes do not exist independently of each other, but acquiesce to a ‘network’ of complex interdependent relationships that are also individually dependent on specific payment windows within an annual cycle. All payments must conform to EC legislation and EC regulatory control mechanisms that undergo continual change.

Development was outsourced to a commercial company who were provided with a requirements specification of the core processes that would form the basis of development activities. This was drawn up from an analysis of the existing ‘As – Is’ legacy system and described a high level view of the Generic Process core activities for the replacement ‘To – Be’ system. Although outsourced the project environment remains within a central location (Cardiff) where both the clients and developers are co-located on the same site throughout the project duration. Consequently both public and commercial cultures co-exist within the same environment. The project structure comprises of a management project board and teams of project workers with a pre-defined reporting structure. Teams are both integrated across the two cultures and are subject/specialist specific according to need.

6 PROJECT CONTEXT

The project was initially planned for a period of 2 years, March 2001 to March 2003 and divided into 4 sequential development stages. Stage 1 involved generating the ‘To-Be’ Generic Process Model. Stages 2, 3, & 4 consisted of iterations of IAD development cycles, testing and acceptance. However an outbreak of the Foot & Mouth Disease affecting the agricultural industry early 2001 impeded development work as key project personnel were re-assigned to deal with it. Consequently the Project was extended into a 3rd year and is still ongoing. Access to the project environment was granted by both client and development Project Managers and permission given to approach all project personnel.

The Project is described as large in terms of finance – an initial estimate of £10m; size of the project team – a core team of 50+; the volume of business rules developed aligned to business processes – currently in excess of 4,000; and the extent of its agricultural customer base throughout Wales.

7 RESEARCH METHODOLOGY

The Researcher adopted an interpretive stance as advocated by Walsham (1997) aimed at producing an understanding of both the context of the IS and the process in which the IS influences and is influenced by its context. Ethnography was selected as a style of interpretative qualitative research for intensive data collection that allows a rich and deep interpretation (Myers 1999, Orlikowski and Baroudi 1989). Its suitability is reflected by its association with other IS development projects (Loftland and Loftland 1984, Strauss and Corbin 1990, Gill and Johnson 1991, Beynon-Davies 1997, Johnson and Duberley 2000). This methodology included direct non-participatory observation and

indirect observation, informal and formal semi-structured interviews, shadowing of key informants, focus group meetings and spontaneous conversations that allows cross checking and yields stronger substantiation of analysis.

Secondary research reflects two levels. Firstly, an in-depth exploration and analysis of existing literature from both academic and practitioner perspectives to present a foundation for an understanding of the current situation in the field of Rapid Application Development. Secondly, an examination of existing project documents, discourse and artefacts.

A Case Study database is being created using QSR NUD*IST Vivo (Nvivo), a qualitative software product to store and analyse the range of qualitative data collected (Myers 1999, Yin 2003). Initial data analysis was driven by the data rather than the Researcher and concerned 'open coding' that involved content analysis where data were analysed and categorised into themes. Further investigation necessitated establishing how these categories might inter-relate and link into sub-categories that reflect the associated dimensions and conditions. This is commonly known as axial coding and was used to uncover the relationships within the categories. Strauss and Corbin (1990) substantiate the importance of understanding the relationship between structure and process, which they believe are inextricably linked, in order to comprehend the phenomenon being studied.

The Researcher recognises that as both the IS development project and this research are being funded by the Regional Government involved, the data collected may be affected. Additionally, to deal with potential ethical problems confirmation was sought from the Project Board at the onset that participant confidentiality would be maintained, and that analysis from the information gathered would not be used out of context by them.

8 HIGH LEVEL SYSTEM DESCRIPTION

Customers apply for CAP grants and subsidies by submitting completed scheme specific application claim forms. Before being archived the 'data set' of that claim form are captured through a scanning data capture process and stored in a database. A Business Rules Engine contains the CAP schemes and EU regulations as specific business logic rules that are launched dynamically during run time enabling automated workflow processes to apply to the generic processes i.e. acceptance, validation, calculate payment, and any failure procedures against each customer's application claim form. A 'perfect claim' without errors or anomalies that satisfies all the relevant business rules and workflow steps is passed through the system for 'payment processing' via an interface to the current finance systems' Accounting Matrix without manual intervention. However, 'data sets' of forms that fail are placed into a process work queue within the workflow procedure to await the attention of the teams of multi-skilled users with specific scheme expertise. Once resolved, the 'data set' is returned to workflow process for further processing.

9 CASE STUDY PROJECT AND THE APPLICATION OF A RAD APPROACH

This section describes aspects of the context of the IS development project. It also describes critical elements of the development work important to the Researcher's explication of RAD.

Set out below in Table 1 are the 9 fundamental principles that the DSDM Consortium have established constitute a RAD type methodology. These were used to determine the application of a RAD-ish approach across on-going development of the Case Study Project.

DSDM Principles		Case Study Project		
		Year 1	Year 2	On-going
1	Active user involvement	Yes	Yes	Yes
2	Teams empowered to make decisions	No	Yes	?
3	Frequent delivery of products	Yes	Yes	Yes
4	Fitness for business purpose	No	?	?
5	Iterative and incremental delivery	No	Yes	?
6	Changes are reversible	Yes	Yes	Yes
7	High level requirements	Yes	No	?
8	Integrated testing during lifecycle	No	Yes	?
9	Stakeholders co-operation	No	No	?

Table 1. *DSDM 9 Principles as Applied to the Case Study Project*

There is significant evidence that the first 8 of the principles that constitute a RAD process occurred during the Project, however although co-operation was achieved with the majority of stakeholders, a serious constraint on the RAD development process was the inflexibility of the EC as a major stakeholder. Conformity to EC legislation and regulations where the 'fit for purpose' requirement reflected almost 98% of business needs impacted on the time-boxing concept and the de-scoping element of a RAD development process that proved difficult to achieve.

As the literature suggests a successful RAD approach necessitates cultural and managerial changes (Hirschberg 1998, McConnell 1996). In this particular case study a radical shift in the mindsets of 'organisational' people was needed to promote the Generic Process model. Evidence exposes the difficulty that these people had in moving away from their previous 'silo' attitude to buying into the Generic Process concept. This was highlighted by their inability to prioritise scheme development work as each Scheme Manager believed theirs was paramount. The inability to make empowered decisions about business needs was a key concern for the developers who needed to meet time-boxed development deadlines. It is felt that this issue is linked to the identified need for strong project management with a RAD approach. Analysis of views from both the Development Team and the client Project Team indicate that this was a Project Management issue that should have been resolved through direct supervision of those concerned to prioritise business needs. However an influencing factor reveals that the Scheme Managers are deemed to 'own' their scheme business processes and as such did not respond to the level of Project Management supervision that was applied. Consequently, Project Management is viewed as reactive rather than proactive and considered weak since the authority exerted was not sufficient or effective enough to handle the impact this issue had on initial development delay. Conversely, Project Management believe that this problem reflects the 'civil servant' culture inherent in government environments where decision making is deferred to a higher authority in response to a perceived 'blame culture' environment. The Researcher believes that low visibility of project management rather the perceived lack of project control may be the issue.

Perhaps the most notable argument being addressed is the view expressed in the literature that RAD is unsuitable for complex IS development. The complexity of this IS system is reflected by the CAP schemes that are not independent of each other, but acquiesce to a 'network' of highly complicated interdependent relationships that are also individually dependent on specific payment windows within an annual cycle and subject to continual change. These difficulties have been successfully addressed through the use of business rules logic. Each business rule contains an action that gets performed when a condition is met. In other words business rules are concerned exclusively with a business action that needs to be performed and the circumstances that trigger that action. In this way a business rule represents a part or parts of a process of a particular scheme that satisfies a specific business need.

The developers decompose the business rules into individual statements that are underpinned by 'Java' coding. These are then saved and stored for re-use to provide a flexibility where the business rules can be modified by the organisations people without the need to alter the Java coding. This has enabled appropriate employees to meet changing and new business needs, a RAD concept, and conform to the EC legislation and EC regulatory control mechanisms. However, this only accommodates simple straightforward regulatory changes such as modifying payment dates or percentage values set within a business rule. For more complex interwoven changes whose impact cascades across a number of inter-related schemes the need still exists for specialist IT knowledge that the organisations people do not have. Consequently this creates a heavy reliance on the developers that is seen as a potential problem.

10 CONCLUSION

The research question concerns the utility of using a RAD approach for a large complex IS development. Contrary to the published beliefs, a RAD type approach has been adopted for a large complex IS that is currently being implemented within UK Regional Government even though existing literature continues to question its suitability for this type of development environment. The case study is significant in that it presents an atypical and valuable opportunity to provide an analytical insight of RAD type development approach within such a complex environment, and thus respond to the research question and views expressed in the literature. Additionally, it addresses the identified lack of academic research in this field whilst adding to the body of existing knowledge.

Evidence suggests some distinct areas where the RAD approach has been successful. Most notable is the iterative nature of the design, build and integrated test phases of the development life cycle that enabled the system to accommodate the complexities and interwoven relationships inherent in the CAP schemes. This in conjunction with the application of business rules logic provides a flexibility for the system to evolve inline with EC legislation and regulatory controls that is seen as a critical factor in its perceived success. This is a keystone goal of a RAD approach i.e. to deliver effective solutions for volatile business environments.

The JAD workshops, a main feature of RAD, that were used for requirements elicitation were of particular importance in the success of this development approach. They were responsible for drawing out the crucial business understanding and perception that was not documented but only existed as tacit knowledge of individual scheme experts. This coupled with the iterative design, build and testing cycles is seen as a critical success factor in meeting the degree of conformity required by the EC.

A major success factor is that this project is perceived as a re-usable investment that will provide future benefits since the system itself has potential for wider applications across similar arenas within UK Regional Government.

However there are also areas where the RAD approach has been unsuccessful. Strong effective management is considered key to the success of RAD. However, it is evident that Project Management here was ineffective in managing the cultural and managerial changes necessary for such an approach. Evidence exposes the difficulty experienced by key people in shifting away from a 'silo' attitude to buying into the Generic Process concept that was counter cultural and hence difficult for them to accept. Considerable delay resulted from their inability to make empowered decisions that, it is felt, could have been controlled through stronger management supervision and pressure exerted.

An external influence outside the projects control provided a further serious constraint on the RAD development process. The inflexibility of a major stakeholder, the EC, whose rigidity necessitated high levels of conformity to business needs that impacted severely on development deadlines.

Although business rules logic is successful in meeting straightforward system changes, more complex changes require involve a software solution. Thus the need still exists for specialist IT knowledge that the organisation's people do not have. Consequently this creates a heavy reliance on the developers that is seen as a potential problem once the project has reached completion.

To date, both the clients and the developers believe that the RAD type development approach has been successful for this case study development environment, particularly in light of its evolving and volatile nature, and suggest that had a traditional 'Waterfall' development approach been adopted the project would have been cancelled early during second year. It is believed that this research has provided a useful starting point to critique the utility of using a RAD type development approach for large complex IS development and the Researcher expects to continue research and analysis to clarify this issue further.

References

- Beynon-Davies, P, Mackay, H, Slack, R & Tudhope, D (1996), *Rapid Applications Development: the future for business systems development?*, Published proceedings of BIT96 Conference, November 7th, Manchester Metropolitan University,
- Beynon-Davies, P (1997), 'Ethnography and Information Systems Development: Ethnography of, for and within IS development', *Information and Software Technology*, vol. 39, pp. 531-540.
- Beynon-Davies, P (1998), *Rapid Applications Development (RAD)*, Briefing Paper, Kane Thompson Centre, University of Glamorgan.
- Beynon-Davies, P, Carne, C, Mackay, H & Tudhope, D (1999), Rapid application development (RAD): an empirical review, *European Journal of Information Systems*, vol. 8, pp. 211-223
- Beynon-Davies, P, Mackay, H & Tudhope, D (2000), 'It's lots of bits of paper and ticks and post-it notes and a case study of a RAD project', *Information Systems Journal*, vol. 10, pp. 195-216.
- Boehm, B (1999), 'Making RAD work for your Project', *IEEE Computer*, March, pp113-117.
- Cross, SE (1998), 'Toward Disciplined Rapid Application Development', *Software Technical News*, viewed 02/01/2002. <http://www.dacs.dtic.mil/awareness/enesletters/technews22-1/disciplined.html>
- DSDM Consortium, (1995), *Dynamic Systems Development Method*, viewed 14/02/2002. <<http://www.dsdmna.org/en/about/principles.asp>>
- Elliott, E (1997), Rapid Applications Development (RAD): an odyssey of information systems methods, tools and techniques, *4th Financial IS Conference*, Sheffield Hallam University, U.K.
- Gill, J, Johnson, P (1991), *Research Methods for Managers*, Paul Chapman Publishing Ltd, London.
- Hammersley, M, Atkinson, P (2000), *Ethnography – Principles in Practice*, Routledge, London.
- Highsmith, J (2000), *Agile Software Development Ecosystems*, Addison-Wesley, London.
- Hirschberg, MA (1998), 'Rapid Application Development (RAD): A Brief Overview', *Software Tech News*, vol. 2, no.1, pp. 1-7.
- Johnson, P, Duberley, J (2000), *Understanding Management Research*, SAGE Publications, London.
- Loftland, J, Loftland LH (1984), *Analysing social settings: A guide to qualitative observation and analysis*, Wadsworth Publishing, Belmont, CA.
- Martin, J (1991), *Rapid Application Development*, MacMillan, New York.
- McConnell, S (1996), *Rapid Development – Taming Wild Software Schedules*, Microsoft Press, Washington.
- Myers, MD (1999), Investigating Information Systems with Ethnographic Research, *Communications of the AIS*, vol. 2, Article 23.
- Orlikowski, WJ & Baroudi, JJ (1991), Studying Information Technology in Organisations, *The Institute of Management Sciences*. Vol. 2, pp 1-28.
- Osborn, CS (1995), 'SDLC, JAD and RAD: Finding the Right Hammer', *Centre for Information Management Studies*, Working Paper, pp. 95-107.
- Strauss, AL & Corbin. J (1990), *Basics of Qualitative Research*, Sage publishing, CA.
- Tudhope, D, Beynon-Davies, P, Mackay, H & Slack, R (2001), 'Time and representational devices in Rapid Application Development', *Interacting with Computers*, vol. 14, no. 4, pp 447-466.
- Walsham, G (1997), *Interpreting Information Systems in Organisations*, Wiley, NY.
- Yin, R (2003), *Case study research: Design and Methods*, 3rd edn, Sage Publishing, CA.