A/B Dashboard: The Case for a Virtual Information Systems Development Environment to Support a RAD Project

Panos Louvieris  
School of Management  
University of Surrey  
Guildford GU2 7XH  
United Kingdom.

Jan Powell-Perry  
School of Management  
University of Surrey  
Guildford GU2 7XH  
United Kingdom.

Paul Beynon-Davies  
European Business Management School  
University of Wales Swansea  
Singleton Park Swansea SA2 8PP  
United Kingdom.

Abstract

The impact of digitization on organizational structures and processes is transforming relationships and work practices. The dynamics of the digital economy fuelled by the high rate of innovation in digital technologies [2] and changing customer expectations [3] demand that firms adopt an equally rapid and responsive information systems development model/paradigm to keep pace. With reference to Rapid Applications Development (RAD), this case study considers the degree to which an iterative and participative information systems development process is amenable to being conducted in a virtual manner. Using participatory action research, this paper investigates the potential for deploying a virtual learning environment as a CASE tool for the development of web-based applications. In this instance the final IT artefact is an operational decision support prototype - the A/B Dashboard. The purpose of the paper is to demonstrate that a virtual information systems development (vISD) environment can successfully support a RAD/DSDM approach, resolving some of the existing problems associated with RAD.

1. Introduction

The impact of digitization on organizational structures and processes remains unabated, transforming relationships and work practices. These days organizations are operating in rapidly changing environments requiring equally flexible and adaptable organizational forms [1] to ensure their future survival. The dynamics of the digital economy fuelled by the high rate of innovation in digital technologies [2] and changing customer expectations [3] demand that firms adopt an equally rapid and responsive information systems development (ISD) model/paradigm to keep pace. Osborn [4] predicted the emergence of networked organizations with the promise of shorter information paths and cross-functional knowledge support as an environment for delivering information systems that reflect a balance between flexibility and stability. He cites evidence of emergent strategies as a gauge of a firm’s competitive agility, often representing unexpected, bottom-up ideas not considered during the formal planning of intended strategies [5]. Similarly, this paper presents a case study that is set in such circumstances.

Using a rapid applications development (RAD) approach, this case study considers the degree to which an iterative and participative ISD process is amenable to being conducted in a virtual manner. The ISD was conducted as a collaborative research project bringing together industry and academe into a single research and development entity. Using participatory action research, the potential for deploying a virtual learning environment (VLE), such as Lotus LearningSpace, as a computer aided software engineering (CASE) tool for the development of web-based applications is investigated. In this instance the final IT artefact is an operational decision support prototype - the A/B Dashboard. The purpose of the paper is to demonstrate that a virtual information systems development (vISD) environment can successfully support a RAD.
approach, resolving some of the existing problems/standstills associated with RAD. The implementation of the A/B dashboard operational prototype was judged successful in accordance to RAD best practice principles employed in Dynamic Systems Development Method (DSDM). Implicit to the approach taken herein is the notion of the “reflective practitioner” [6], so the research method is necessarily rooted in the action-reflection paradigm. In any event, the investigation of information systems with action research is well documented [7].

The paper is organised as follows: First, we address the literature concerning systems development methods, including the use of Dynamic Systems Development Method (DSDM) as a framework for a Rapid Applications Development (RAD). We then turn our attention to the tools and techniques that are an essential part of RAD. We will demonstrate how a VLE can be used as a CASE tool, revealing new opportunities for formalising RAD methods. These issues frame our research questions and provide the focus for this paper. Thus we explore the degree to which the iterative and participative principles of DSDM are feasible in a vISD environment to support a RAD project approach. Emphasis is placed upon:

- the high visibility of decision-making through textualisation that this approach affords;
- using the vISD to capture transitory groups, arresting the dissipation of RAD teams and managing the instability of actors and relations that are typical in a RAD development;
- the vISD as a self-documenting approach that captures case history; and
- the communication opportunities of the vISD to facilitate time/place management in both synchronous and asynchronous modes.

These research questions are addressed within the case study description of the A/B Dashboard as an example of a successful vISD project (action), followed by analysis and evaluation (reflection), using the nine principles of DSDM as an evaluative framework.

2. Literature review

The problems with information systems development in large organisations are well-documented in the literature [8]. Such authors frequently cite issues to do with organisational bureaucracy, high inertia and overly formal development processes. For example, the inadequacies and drawbacks of the traditional waterfall development method [9] are well known [10], [11] with observers noting particularly the relatively slow speed of information systems development to business demands and requirements inflexibility. While RAD as an evolutionary/prototyping method emerged as a popular way forward [12], some researchers comment that in spite of increases in the speed of ISD, the claimed increase in flexibility is questionable [13]. Eva [14] reports that two models of RAD are evidenced in practice: one method claims to be disciplined while the other is described as Just Do It (JDI). The differentiating factors can be found in the rigour of the requirements analysis and user participation, hence he proposes an umbrella definition of RAD that differentiates it from JDI.

“A method of developing information systems that involves: a development team incorporating end-user clients and IS specialists, the use of rapid development software tools, and the staged delivery of a working system by means of iterative prototyping of solutions to business requirements. A RAD project will also involve the use of high level workshops such as Joint Applications development (JAD) and Joint Requirements Planning (JRP) and the use of timeboxing for each delivery in the project” [14].

In many ways RAD techniques are ideally suited to the acquisition of user requirements. The positivist notion within traditional ISD that a requirement is a discrete, identifiable entity is being superseded by a more epistemological view that a requirement emerges from the social interaction between user and analyst [15]. It follows that the more perspectives considered during the early stages of an ISD, the more likely that unexpected requirements will be identified, reflecting a plurality of views as a sound basis for requirements prioritisation. Notwithstanding Eva’s attempts to separate RAD from JDI, concerns in the IS community over RAD’s efficacy have resulted in an industry-led initiative to validate RAD within a professional framework; DSDM. This initiative to develop an industry standard for RAD was taken by the DSDM Consortium in 1994. The resulting and still evolving framework for the tightly controlled management of RAD projects is known as DSDM [16] and its success is deemed to be governed by the nine principles given in Table 1 below.

| Table 1. The DSDM principles [16] |
|-----------------|------------------|
| I.              | Active user involvement is imperative. |
| II.             | DSDM teams must be empowered to make decisions. |
| III.            | The focus is on frequent delivery of products. |
| IV.             | Fitness for business purpose is the essential criterion for acceptance of deliverables. |
V. Iterative and incremental development is necessary to converge on an accurate business solution.

VI. All changes during development are reversible.

VII. Requirements are baselined at a high level.

VIII. Testing is integrated throughout the life-cycle.

IX. A collaborative and co-operative approach between all stakeholders is essential.

The aim is to ensure that RAD can be seen as a valid, disciplined approach to ISD, rather than as a shortcut. To this end, DSDM formalises the iterative nature of RAD by building in explicit testing cycles against fit-for-purpose business criteria. In many respects DSDM/RAD is an inversion of traditional thinking about ISD approaches. In fact, by adhering to the DSDM principles, primacy is given to speed of delivery as shown in Figure 1.

The principal difference between a traditional ISD and DSDM/RAD is that the overall project time and resources remain fixed, but flexibility of requirements is allowed. Timeboxes are the means for managing requirements flexibility, and all timeboxes must fit within the overall project time frame. When the scope of requirements conflicts with the time limit, *the scope is reduced to fit the time limit* - in other words *'the cloth must be cut to suit'* creating greater pressure for effective requirements negotiation. There are a range of methods available for eliciting and negotiating requirements. Eva [14] uses the ACRE (ACquisition of REquirments) framework to demonstrate that RAD methods such as prototyping and JAD can be used effectively to elicit all knowledge types, including tacit and semi-tacit. This is in contrast to traditional fact-finding techniques, such as 1-1 interviews and questionnaires that are more suited to eliciting explicit knowledge. A word of caution is needed here in that the successful adoption of RAD techniques to capture and refine requirements is dependent upon the commitment and involvement of stakeholders and in this sense, DSDM’s stakeholder emphasis is especially beneficial.

Because RAD is a participatory paradigm, JAD has an important role to play. Conventional use of JAD is to bring together all relevant stakeholders to agree information system requirements in the design phase of the project [17]. Given the dynamics of contemporary business environments and the emergent nature of requirements, then a VLE-facilitated JAD will have a greater role to play in managing requirements flexibility and eliciting feedback about prototypes. Such use of the VLE should enhance what Giddings [18] refers to as the management of uncertainty inherent in ISD, at the same time acknowledging the evolutionary nature of ISD [19]. Further, it places emphasis on RAD as knowledge work [20] which can only contribute to building the professional standing of DSDM.

3. Research background

The relationship between Organization A and Organization B is typical of many between academe and industry. Both organizations operate in the tourism domain; Organization B is an international Applications Service Provider (ASP) of tourism and hospitality systems and Organization A’s roots lie in the service sector management arena, historically specialising in tourism. Thus Organization A’s world of teaching and learning benefits from the industrial relevance that the relationship affords and Organization B gains from access to a world-class research institution.

The A/B systems development team was formed as an emergent strategy to facilitate the development of a software system for customer relationship management (CRM). At the time Organization B was in the throws of a major project to re-engineer its business processes, with the attendant problems of disparate and legacy information systems. Hence the organisation’s computer staff were necessarily occupied and unable to respond rapidly to this new initiative. Senior management of Organization B were aware of Organization A’s distinctive competence in crucial areas required by the project, including innovation in ISD and expertise in the design of business metrics such as the use and aggregation of performance measures as critical success factors (CSFs). Importantly, Organization A’s expert knowledge of tourism facilitated a common Universe of Discourse (UoD) of the application domain between users, clients and developers; vital in minimising the communication gap that so frequently impedes successful systems development [21].
The geographical separation of the development team (US/Phoenix, US/Dallas, US/Denver, UK/Birmingham, UK/London and UK/Guildford) appeared to be a significant project constraint. Many researchers (e.g., Venkatraman, [22]) have observed that such organizational structures will precipitate an increase in the adoption of information technology to co-ordinate and inform teams in dispersed locations. In this case Organisation A’s interest in VLEs surfaced as an opportunity for vISD as an enabler for collaborative ISD in a distributed RAD environment. As a result, the Lotus LearningSpace VLE was given the role of a CASE tool to support the mixed development team during the systems development process. Moreover, we see ISD/RAD as an organisational process that has itself been impacted by digitization and which the case study herein will elucidate upon.

To complete the background, the key objective of the CRM dashboard system is defined as follows:

“To provide a consolidated and reliable source of summary analysis and detailed information on each of organisation B’s customers to support the management of the customer relationship”.

4. Methods

The research and development methods employed in the case study are outlined below.

4.1. Research method

With reference to the research objectives and background to the A/B Dashboard development, the research method employed in this paper is collaborative/participative action research [22], [23], [7]. From the outset, the A/B research and development contract was explicitly specified as a co-research (collaborative research) project. The dichotomy between the researchers and the research subjects was eliminated in accordance with the participatory approach. The A/B team members became co-researchers and co-subjects, and participated fully in the iterative action-reflection process cycle during the inquiry. The principal medium for data collection and conducting the collaborative inquiry was the VLE. This enabled explicit textualisation of discussions between members of the geographically dispersed development team. It provided the means by which qualitative and interpretative data was gathered during each timeboxed activity. Furthermore, the nine DSDM principles were adopted as an ex post evaluative framework for participant co-researchers to determine the success of the intervention once the project was completed.

4.2. Development method

Within the context of the DSDM framework (see Figure 2), the development method employed in this case study to create the dashboard operational prototype was RAD. The RAD techniques used were prototyping, JAD and CSF-based information engineering. The novelty of this development is the virtual manner in which RAD was conducted using the VLE Lotus LearningSpace (LLS) as a CASE tool. The VLE was the primary focus of interaction between members of the mixed development team of co-researchers/co-subjects comprising of users, clients and developers. Moreover, the iterative and collaborative processes of DSDM/RAD (esp. Principles V, IX) are entirely in congruence with the participative action research method.

Before turning our attention to the A/B case study, we briefly describe the LLS VLE and outline the main reasons for its adoption in the next section.

5. The virtual learning environment

LLS is the Web-based discussion forum and document database/repository employed for the Dashboard development. The LLS application is based upon Lotus Notes Web-based technologies, providing a secure environment for the project. Its functional components for DSDM/RAD are indicated below.

1. Timeboxes are specified and managed using the ‘Schedule’.
   (A ‘timebox’ is a time limit to deliver a set of prioritised requirements. Providing the essential minimum set of requirements are delivered the remainder may be dropped if time runs out. Timeboxes are determined at the start of the project and remain fixed.)
2. Documents and other media are stored in the ‘MediaCenter’.
3. Participant Discourse is conducted in the ‘CourseRoom’.

Figure 2. DSDM framework [16]
4. Participant Profiles are kept in ‘Profile’. LLS is an asynchronous vISD environment that enables the textualisation and full capture of the international development team’s communications. The entire text-based discourse concerned with the Dashboard development is automatically threaded, author-tagged and time-stamped providing a visible and trackable audit trail. Therefore, LLS is fully self-documenting whereby any documentation created, exchanged and discussed is automatically captured as case history. Using LLS as a CASE tool enables the synthesis of all participants’ contributions, overcoming many of the requirements and design problems typically encountered in RAD projects, including poor access to stakeholders, low availability of clients and users in development team, dissipation of commitment, geographically distributed participants and transitory development teams. [8], [11].

6. A/B dashboard case: the intervention

The project to develop an operational prototype for managing client relationships was launched in Spring 2001 with the only face-to-face meeting of the complete design and development cycle. The high level requirements were specified as follows:

- Effective client portfolio management
- Elimination of gatekeepers i.e. account managers protecting their domain
- Providing senior management with direct access to performance information using a dashboard output built from complex CSF modelling

The central RAD team consisted of eight members, four from each of the partner organizations paired within the following roles: project management, data modelling, business and information engineering, and prototyping. This integration of the user/client with the development/research team proved a significant advantage and is especially relevant to the DSDM principle of empowered teams. All members were registered within the VLE at the start of the project. Inevitably as the development progressed through the various project phases and development cycles, other stakeholders were identified and included within the VLE. Peripheral team members were either given full access to the VLE, providing a complete history of the project development, or in some circumstances, restricted access was deemed appropriate. A key point to note here is the ease with which the VLE supports flexible access to the development history and assists liaison between team members within and across roles. This is essential given the iterative nature of the RAD/DSDM development process.

The VLE’s Schedule was used as the primary project management device because of its suitability for tracking timeboxes through project stages. Timeboxes were developed at the beginning of the project and, once agreed by the RAD team, remained constant throughout. A high level view of the timeboxes can be seen in Figure 3 and the LLS Schedule management of them is shown in Figure 4.

![Figure 3. Project timeboxes](image3)

<table>
<thead>
<tr>
<th>Timebox</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sep</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Refined and developing master and measures</td>
<td>✔️</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Data model prototype specification</td>
<td>✔️</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Marketing tools process and definitions</td>
<td>✔️</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Implementation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Monitoring and evaluation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Current report</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Critical Success Factors</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8. Collect individual CSFs</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9. Aggregate and analyse individual CSFs</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10. Physic test CSS framework</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11. Collect measurements and build model process</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12. Outline and begin managing processes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13. Finalize and deliver in 6 steps and project management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14. Evolution and development (dashboard development phase)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15. Decision report</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Figure 4. LLS Schedule](image4)

Key CourseRoom discussion threads were also established early on, with links set from the Schedule to facilitate regular team input. As the project progressed, the CourseRoom was used as the team’s primary communication channel, where all members could initiate new discussions or contribute to existing threads. In addition, work-in-progress was regularly presented for discussion via the CourseRoom. An extract from this discourse can be seen in Figure 5.
The MediaCenter was used to store key documents as they emerged, with links from both the Schedule and the CourseRoom, ensuring that team members could find the documents they needed, when they needed them. Taking the development of the data model as a case in point, the VLE provided an invaluable tool for managing the delivery and discussion of the emerging design. At the project management level, the Schedule signalled the time start and time stop of the data modelling phase, as per the planned timebox. Both the logical and physical models were presented to the team within the VLE discussion area as work progressed, allowing for easy access to all iterations of the design. Not only are the final models stored within the MediaCenter as formal documents, but also the full development cycle (including all design iterations and accompanying discussions) is now captured as case history in the CourseRoom. An unexpected outcome of this textualisation of debate is the visibility of “information politics” [25]. This is best illustrated with an example from the data modelling phase. At one point the positioning of a single attribute within the overall model emerged as the focus of attention as it became clear that operational control over the variable in question, to set performance targets for individual clients, would inevitably result in a transfer of power. Thus what could have been a hidden agenda became visible in the VLE, further ensuring a plurality of views and open discussion as a sound basis for decision making.

The information engineering phase for the CSF method was managed in the same way, and again, the full history of the emerging algorithm and discussion is encompassed within the VLE. An example output, based upon the method, is shown in Figure 6.

7. Evaluation

The Dashboard development was evaluated using the nine DSDM principles bearing in mind Checkland's [26] ‘3Es’ Performance Criteria.

1. **Efficiency** – Are we achieving the transformation at the least possible cost (time and money)?
2. **Effectiveness** - Is the transformation achieving its intended purpose? In other words does it satisfy customer need?
3. **Efficacy** - Do the means work?

Evaluation was sought from both client/user and development/research organizations and these are presented in table 2 below. The resulting concensus evaluation from the client/user meetings, held locally at the client organisation, were placed in the the LLS CourseRoom as a completed assignment. The client/user evaluation comments as shown in Table 2 are unedited. The academic researchers completed a similar exercise and their comments are shown alongside the client/user comments.
### Table 2. Evaluation of the development/ intervention

| III. | The focus is on frequent delivery of products.  
Client/User evaluation: The Dashboard development’s primary deliverables were a data model and a prototype design. Preliminary versions were created and delivered at key points during the project lifecycle but there was not substantial scope in this project for frequent product delivery. However, a timebox approach was adopted to ensure that product delivery remained the key driving point at each stage of the development.  
Developer/Researcher evaluation: The short information paths within vISD helped to ensure maximum output within project timeboxes and therefore optimum delivery of products was achieved. The whole development team was able to visualise and experience interim ISD products. The Web-based development medium in this case study was also the product delivery medium which meant the lag in the action-reflection cycle was minimised.  

| IV. | Fitness for business purpose is the essential criterion for acceptance of deliverables.  
Client/User evaluation: Evaluations of the match between the Dashboard prototype, the data model and the defined business purpose were carried out at several key stages of the project. The feedback at each stage was incorporated into the next element of the development process. Again, this activity was greatly facilitated by the use of Lotus LearningSpace.  
Developer/Researcher evaluation: vISD helped to control and formalise deliverable acceptance through all stages/phases of the project development. Textualisation of Client/User acceptance of each timebox deliverable was explicit and visible to all stakeholders. The security and author-tagging features of the LLS/vISD ensured authenticity, integrity and nonrepudiation. The adaptability and flexibility of the vISD approach ensured that changes in business needs were more easily accommodated and securely tracked.  

| I. | Active user involvement is imperative.  
Client/User evaluation: In this development the user involvement was significant. The concept and design principles were constantly evaluated and verified by representatives of the user community.  
Developer/Researcher evaluation: The direct input from the user/client via the paired roles and liaison across roles allowed for frequent/full discussion in a time/cost effective way. The asynchronous communication in vISD proved powerful, overcoming time/distance issues by allowing the client 'Ambassador/Advisor Users' [27] to accommodate the dashboard development within their daily jobs. Thus the project did not suffer the restriction of non-availability that results from the synchronous participation demanded by conventional JAD workshops. The virtualisation of DSDM using vISD enabled frequent/higher levels of participation than previous RAD approaches.  

| II. | DSDM teams must be empowered to make decisions.  
Client/User evaluation: The team contained all the representative elements necessary to ensure effective decision making and was empowered with full control over the project. Decision-making efficiency was enhanced by the use of Lotus Learning Space.  
Developer/Researcher evaluation: Ease of access to all stakeholders (as above) is confirmed as a fundamental requirement for a fully empowered team. The vISD proved an effective enabler of high levels of relevant participation for credible, consensus decision-making.  

| |
V. Iterative and incremental development is necessary to converge on an accurate business solution.

Client/User evaluation: The iterative approach ensured that as the data model evolved and new factors in the design emerged the project team were able to adapt the solution. Some of the original conceptual thinking had to be adjusted as the development proceeded. However, the resulting deliverables provided a much more accurate business solution.

Developer/Researcher evaluation: vISD ensured a collaborative solution via fast cycling through requirement/design/build iterations, ensuring confidence in the solution from both the development team and the client group. Overall, vISD increased the development tempo and enhanced product quality without sacrificing reliability.

VI. All changes during development are reversible.

Client/User evaluation: This principle was demonstrated to a certain extent in the data modelling exercise where assumptions had to be revisited and changed in order to reflect business processes correctly. However, in general it was not found to be necessary to reverse changes during the development. The accurate progress tracking and documentation which was facilitated through Lotus Learning Space ensured a complete picture of the current status of the project was available to all project participants throughout the development.

Developer/Researcher evaluation: High levels of participation enabled full functionality to be captured with few reversals. Where reversal occurred, vISD ensured high visibility and reasons for back-tracking or reconstruction were documented.

VII. Requirements are baselined at a high level.

Client/User evaluation: The key is to find the balancing point when defining requirements. On the one hand, they should not be set at so high a level that a considerable portion of early development time has to be spent refining the business requirement. Conversely, requirements must not be so detailed and rigid that they suggest only one path to a solution as this can result in reduced efficiency of development and a less effective solution.

Developer/Researcher evaluation: vISD facilitated involvement of all relevant stakeholders for baselining high level requirements at the project outset, concomitantly with the timeboxing exercise (see figure 5). “MoSCoW Rules” [27] were involved in the prioritisation of requirements.

VIII. Testing is integrated throughout the life-cycle.

Client/User evaluation: Testing of both assumptions and functionality was integrated into the development process. This contributed to the team’s confidence in creating a solution which would meet the business need.

Developer/Researcher evaluation: On-line testing was integrated throughout the life-cycle. vISD conveniently lent itself to testing this on-line application. May not be so easy when developing off-line applications.
A collaborative and co-operative approach between all stakeholders is essential.
Client/User evaluation: The project brought together a diverse group of stakeholders with a common objective. The collaboration and co-operation between team members was essential to the success of the project. This was established early in the development through the definition of the requirement, the assignment of resources and the flexibility of the approach which was adopted.

Developer/Researcher evaluation: vISD can be considered groupware, with attendant advantages i.e. vISD enabled virtual colocation of users, clients and developers facilitating a highly flexible and hyperparticipative approach. The vISD was quintessential to this international project requiring the collaboration and co-operation of a geographically distributed development team. Dashboard could not have been delivered within the project time/cost constraints without vISD support.

8. Conclusion

Referring to Table 2, the virtualisation of DSDM/RAD is considered successful in delivering the Dashboard product. The case study has illustrated the potential for the LLS VLE as a CASE tool in the development of web-based applications. Specifically, this co-research has demonstrated that vISD-enabled asynchronous participation is key to overcoming some of the current standstills encountered in DSDM/RAD. Moreover, vISD can increase the development tempo and enhance product quality without sacrificing reliability. This ties in well with the demands for greater flexibility and adaptability, without losing effect, now placed upon organizations.

Whilst it is acknowledged that vISD is not a 'silver bullet' solution to the challenges facing RAD, it does resolve many problems. For example, the twin notions of dynamic and transitory RAD teams pose problems in that accountability for RAD developments is not always clear. In vISD the teams are captured within the VLE and the high visibility of decision-making through textualisation serves to validate RAD as a disciplined approach to developing systems. Rigour in terms of requirements acquisition is also enhanced in vISD. According to DSDM principles, stakeholder negotiation throughout all iterative cycles is essential for effective requirements re-appraisal. This poses a problem in traditional RAD projects, where JAD workshops are the means for bringing stakeholders together. Thus a dilemma is created – should system functionality be compromised within the timebox in favour of full stakeholder representation in the JAD, which may be difficult to organise? The asynchronous nature of vISD eliminates this dilemma. Stakeholders contribute when it is convenient to them. All that is required is access to the Internet and the development of mobile Internet access devices will bring collaboration ever closer, with opportunity for extended JAD across all timeboxed stages of the project. Asynchronous communication imposes reflection and this may help to elicit requirements that mirror tacit and semi-tacit knowledge. This kind of representative and reflective participation can only assist in the early capture of full system functionality and therefore fewer reversals. Where they do occur, the rapidity of action-reflection cycles in timeboxes reduces uncertainty and each cycle is an experiment in knowledge elicitation that is captured and fully documented.

Organizations wishing to capitalise on the benefits of digitization for vISD may wish to consider the implications of this case study.

9. References


