Optimising the Potential of Educational Computing Research in Emerging Countries

Ronald Bisaso

This chapter explains the possibility of merging approaches to educational computing research for the benefit of developing regions. It is on the premise that educational researchers, policymakers, and practitioners agree that educational research is often divorced from the problems and issues of everyday practice (The Design-Based Research Collective, 2003). Moreover, educational effectiveness and efficiency is a factor of Information and Communication Technology (ICT) especially the computer – a ‘tool of our time’. Developing countries have seen the potential of educational computing as a catalyst in the enhancement of knowledge acquisition and management, but approaches to educational ICT research in these settings have not been explored. It is noteworthy that software developers have either developed materials on the basis that they are conversant with all the contextual factors, or have remained oblivious of the advantages accruing from understanding the change process. Consequently, in this chapter, Bisaso proposes a blended approach where two strands, namely (rapid) prototyping and basic research ought to sit side by side if the exceeding relevance of educational computing research is to be sustainable and its returns fully exploited in the developing regions.

Introduction

Interest in developing educational technology solutions to curriculum and instruction problems has been ever increasing. Czerniewicz, L. and Carr, T. (2005) report that there is evidence of diverse emerging practices in the use of educational technologies across Southern Africa, in both the school and higher education sectors, hence suggesting that institutions at all levels in the developing regions have seen the benefits that this field of study offers that each is considering installation of new educational computing hardware and software. Governments at all levels have expressed support for educational technology initiatives, for example the Uganda Ministry of Education SchoolNet project, and the CurriculumNet project at the National Curriculum Development Centre (NCDC), the latter tasked to develop content online or computer-based curriculum instructional materials (National Curriculum Development Centre, 2004). It is thus not difficult to find environments in which to evaluate educational computing research and development owing to the prevalent support for educational technology and the anticipated prospects it renders to effective schooling in the developing regions. Rath and Hsu (1999) argue that it is the existence of this potential that motivates the necessity for progress in productivity and quality in educational computing research and development. Yet Nieveen and Gustafson (1999) note that data documenting
the advantages accruing to the practicality and effectiveness of educational computer support systems are still insufficient. Rath A. (1999) acknowledges that, in education, we are struggling to link research with development in a practical and productive manner.

Reeves (2000) recounts that some instructional technologists appear to have immense dedication to basic research, irrespective of whether it has any practical value, perhaps because basic research seems more scientific or they believe that it is someone else’s role to figure out how to apply the findings of basic research. Others seem to believe that the value of basic research in a design field such as IT is limited and that ICT research should therefore have direct and clear implications for practice. Basic research is characteristically work that emphasises fundamental processes. Development is typically work that emphasises the design and development of curriculum materials (Rath and Hsu, 1999). Tyler (1978, in Rath and Hsu, 1999), for example, stated that “Research is an activity seeking to gain greater understanding of a phenomenon, while development is an effort to design a system that will achieve the desired ends under specified conditions or constraints” (p. 94). In this vein, the two approaches to investigation are amalgamated into research and development, and consequently globally defined as a “process that links basic research on curriculum, instruction, and learning with the design and development of educational products and processes for specific teaching and learning situations” (Rath and Hsu, 1999). According to Reeves (2000), Stokes (1997) called for more “use-inspired basic research” rather than either pure basic or applied research. “Use-inspired basic research” for instructional technologists is what is labelled “development research” (Akker, 1999).

Notwithstanding the inventiveness in educational computing, we ought to uncover ways to qualitatively and quantitatively enhance the current state of invention and inquiry. It should be borne in mind that educators are limited in choice when it comes to selection of tested and proven educational computing systems in the developing regions. This explains the need for guaranteeing sustainability of research and development to the advantage of both the theorists and designers, remaining keen on their interdependence (International Federation for Information Processing, 2005). Development researchers are also committed to constructing design principles and producing explanations that can be widely shared. Instructional technologists engaged in development research are above all reflective and humble, cognisant that their designs and conclusions are tentative in even the best of situations (Reeves, 2000). Moreover, educational computing research and development is both a process and a tool. It thus remains questionable how one is either a theorist or a designer and successfully develops well-situated products. Although it may seem to work, this naivety explains why educational computing products and processes developed in faculties or Schools of Education hardly trek past the development phase (Rath and Hsu, 1999). The subsequent sections of this paper discuss the invention and innovation approaches in detail.
Invention (prototyping) research

Nieven (1999) argues that to reach product quality, the suitability of the prototyping approach is beyond question. Nieven (1999) quotes Smith (1991, p.42) who defines a prototype as “a preliminary version or a model of all or a part of a system before full commitment is made to develop it.” Prototypes are all interim products that are incrementally designed before the final product is constructed and fully implemented in practice (Moonen, 1996). Nieven (1999) notes that based on several such cycles, the computer system evolves towards a high quality final product.

De Hoog, de Jong and de Vries (1994, in Nieven, 1999) are reported to propose a non-linear approach to development in which several parts of the product are developed independently of each other. Nieven (1999) notes that this approach envisages an improvement in the product under construction with its parts showing a correspondence with the set quality criteria, and remaining consistent with the other components undergoing development. The output of this method is in all probability regarded exemplary and consequently very useful as innovations unfold in educational settings. These software materials can, for instance, illustrate the basic meaning of change at hand, provide the would-be users with a (probable) chance to interface with these materials or better still elicit interaction among the users on their relevance or irrelevance. Nieven (1999) recounts that only when these materials are of “good quality” can we predict with certainty that they will fulfil the obligation mentioned already.

Even then, the aspect of quality remains hazy because of its varied interpretations (Nieven, 1999). Some often imply robustness (working well) while others effectiveness (better results). Either way, a robust or effective product is cherished and valued but this ought to be grounded on the tenets below:

a) Validity: This refers to the extent to which materials developed are congruent with the state of the art knowledge (content validity) and consistent in linkage between the diverse sections of the materials (construct validity). Educational software can only be valid on meeting this standard.

b) Practicality: Software for learning or educational computing products will be regarded to be of quality when there is compatibility between software utilisation among the intended audience (teachers or other experts) and the software developers’ intent.

c) Effectiveness: Users’ experience with the products developed is such that it adds value to their current way of doing things in a way that is in agreement with the intentions of the developer.

However, in an educational context, it must be remembered that ICTs are basically tools for facilitating learning and teaching. Their success does not depend on the technology itself, but on the correctness of its application. High expectations for ICT applications may cause disappointment among their users, if they do not take full account of the actual educational contexts, hence the importance of understanding innovations.
Innovation (basic research)

Fullan (1999) attempts to discuss the interactive factors affecting implementation in educational settings, and he posits that the characteristics of the change, namely Need, Clarity, and Complexity, are vital ingredients in the implementation of innovations. Educational computing innovations are no exception. Need refers to how much scrutiny is made in relation to the software ‘addressing’ the most imperative demands. Equally important is the establishment of the edge of the software product under development over other similarly essential issues; otherwise, the beneficiaries of the status quo may reject it as soon as it is rolled out. On clarity, Fullan notes that “people often become clearer about their needs only when they start doing things, that is, during implementation itself”. The question, have software developers taken the trouble to deal with making their clients “clear” before they get “clearer”? The level of difficulty, relevant skills needed to grapple with the change, and alteration in beliefs, instructional strategies, and application of technologies constitute the complexity attribute. A software developer who has distanced himself from these characteristics thrives on relativism rather than objectivism during his research in educational computing.

Rogers (2003) recounts that it is not uncommon for societies to adopt relevant practices external to their environments, and adapt them to meet their own needs, and discard the rest; and according to Constant (1984), and Rath and Hsu (1999), “Community and tradition are the locus of what is called technological change” (p. 29). It is impeccably acknowledged that practitioners are the major determinants of the change process (Rath and Hsu, 1999). Dourish (1995) argues that “the design process does not end with the delivery of a system to some community of users. Instead, it continues as they use and adapt the system” (p.44). This serves to concretise the view that the software development process is a never-ending journey which undergoes several refinements as the context may dictate. Even in web technologies, an area common in educational computing, Powell (1998) notes that the value of software depreciates over time unless it is altered in response to changing needs.

Development research (a blended approach)

Development research refers to the successive approximation of interventions in interaction with practitioners (Akker, 1999). Its aim is both practical and scientific contributions but the innovation challenge is usually quite substantial. Moreover, it involves an iterative process of ‘successive approximation’ of the ideal intervention is desirable. The argument is that direct application of theoretical underpinnings is a necessary but not a sufficient method to solve these rather complicated problems. Hake (2004) argues that proponents of design-based research are willing to attempt to address, simultaneously and iteratively, the scientific processes of discovery, exploration, confirmation, and dissemination, resulting in an active innovation and intervention in the classrooms (Kelly, 2003). Moonen (1999) reported on the rational and relational approaches; the former entailing problem-disintegration into sub-problems and eventually tackling these in a micro manner. On the whole, problems are well defined and consensus is obtained with regard to the solution.
Moonen (1999) on the contrary recounts that design methods that are only lop-sided to this strand are inappropriate. This, he argues, is a result of the growing influence of the stakeholders. Shön (1993, in Moonen, 1999) is reported to have suggested an extension in the execution of this process to embrace “reflection-in-action” in which means and ends are dealt with iteratively. Kessels (1993) refers to this as the relational approach – “gradually evolving correspondence between the product and the intended user”.

Moonen (1999) further questions the way to go – either ‘rational’ (invention) or ‘relational’ (innovation). His argument is that it all depends on the situation. He further argues that once there is clarity of purpose then a rational approach is more appropriate. However, when the contrary is the case, a relational inclination serves better. Moonen concludes that “it is a wise choice to combine in a global design strategy, the strengths of both approaches”. This is represented in the model in Figure 1. Fusion of these two methods should bear in mind that a) there is often uncertainty about how to proceed, b) the context has considerable influence, and c) many design activities have to do with adaptation (Moonen, 1998). Kessels (1993) in his study on curriculum consistency in corporate education labels this as internal and external consistency respectively, implying rational (invention) and relational (innovation) approaches.

Discussion and conclusion

The preceding discussion and the model suggest that educational computing research and development should focus on the product through the lenses of basic research hence producing educational software that has been thoroughly tested on both fronts. Much as it is costly to develop a software product, to solve some of the educational and training challenges, an informed approach of involving the stakeholders through analysing the context (what hardware and software limitations), the task at hand and the users themselves is inevitable (Reeves, 2000). Muianga (2005) reports on an on-site study into the development and application of a course management system (CMS) for online learning in Mozambique, which clearly shows the application of development research. Teachers and students were involved in the use of the TeleTop environment, a CMS, which enriched both the design aspects and the innovation challenge, especially on the part of the teachers. Moreover, the Apple Advance Technology Group (Sphorer 1998, in Moonen, 1999) reported that users had to be involved in two perspectives, namely the cognitive fit (for usability) and social fit (for dissemination).

The authors acknowledge that the social fit is appreciably longer than the cognitive fit. Hence, educational technologists should establish the critical determinants for successful adoption and adaptation as they strive to evolve well-situated products for the developing regions. It is of paramount importance that educational technologists in emerging regions are duty-bound to develop models as frameworks for adoption and diffusion. Miller’s model, a synthesis of a variety of technology-implementation models, has been successfully tried out in South Africa and Mozambique to study the introduction of computers in schools (Cossa and Cronje, 2004). These two southern countries have been mapped onto the stages in Miller’s model. Emerging countries
also ought to explore possibilities of at least designing software according to their local priorities; for example, in the Southern African region, South Africa is becoming a producer of software and ICT-related services within the global market (Hodge and Miller 1997; Otter 2005, in Czerniewicz and Carr, 2005).

References


