Forms and processes of information systems evolution

Conference or Workshop Item

Forms and processes of information systems evolution

Magnus Ramage, Open Systems Research Group, The Open University
m.ramage@open.ac.uk

Abstract

The way in which software evolves over time has been much studied and is now fairly well-understood. What has been less thoroughly studied are the processes by which information systems – containing software as one component, but also with significant human and organisational aspects – evolve. In many organisations, few information systems are built at all from scratch, but rather are modified from or built on top of existing ones or bolted together from third-party components. In practice, the old division between design, implementation and maintenance has largely disappeared. In this paper, I discuss the nature of IS evolution. I make a distinction between planned (intentional and strategic) evolution, for which we can formulate a clear process; and unplanned (emergent and externally-driven) evolution, where we can simply study the dynamics of the process and be ready for events.

Keywords

Information systems evolution, planned and unplanned change, legacy systems, evaluation, scenario planning, coevolution.

Introduction

Rapid change is constant and continuous in organisations today. Lewin's classic three-phase model of change (unfreeze-change-refreeze), with its expectation that the changed state of the organisation remains in place for a prolonged period is now widely regarded as too simplistic, although Burnes (2004) observes that Lewin was aware of some of the inadequacies of his model. The rapidity of change is widely observed, and yet it is poorly reflected in both IS theory and practice.

In particular, the traditional model of the information systems development lifecycle – with its stages of analysis, design, implementation and maintenance – is increasingly problematic and a poor reflection of the needs of real information systems use. The life cycle model is largely predicated on the bespoke development of information systems from scratch. However, few information systems arise in this way – most are built on top of an existing IS (legacy systems), or bought in from external suppliers (commercial off-the-shelf systems). Even those which are developed in-house frequently use third-party components, building upon the past work of the developers of those components. Furthermore, it has long been known that the maintenance phase of the IS lifecycle can consume more than half of the total lifetime costs of an information system (Bennett, 1996).

These phenomena taken together lead to the conclusion that instead of focusing on IS development, we should instead focus on IS evolution: the ongoing revision of information systems, gradually changing over time, building upon existing components

(organisational and technical), to meet changing business needs. This focus resembles but goes beyond calls for a focus on software evolution.

Software evolution derives from two main streams of work. First, there is an attempt to allow the maintenance phase of the IS lifecycle to be regarded as more significant, given the proportion of costs and resources it occupies. This has led to a reconceptualisation of the software lifecycle as being primarily driven by the evolution of existing software systems (Bennett and Rajlich, 2000). Second, there is an analysis of the processes by which software evolves over time, principally at the level of program code, using methods of system dynamics (Lehman and Ramil, 2003).

Both these streams of work come from within computer science, and so their discussions are confined to software as an artefact, regardless of the context of use. However, as is well-known within IS research, software is only one component of the wider information system – others including organisations, actors and the way we conceptualise interactions between these and software technologies (Checkland and Holwell, 1997). A model of IS evolution will therefore need to take into account these wider components and interactions. It is the purpose of this paper to discuss such a model.

Towards a Model of IS Evolution

The concept of IS evolution is complicated by the fact that the term conflates two different forms of evolution: that which an organisation intends to happen and builds into their decision making; and that which occurs whether or not organisations choose it. I will refer to these two forms as *planned* and *unplanned* evolution. In practice, both forms of evolution will be happening at any one time within most settings – in particular, even organisations that have a clear plan for the evolution of their information systems will encounter external events that change the IS without their intention. However, it is useful to draw the distinction between these forms, as they represent different activities with quite different organisational responses.

Planned evolution is part of the normal organisational decision-making process. It must therefore fit with an organisation's strategy and timescales. It should not be thought of as an unusual event, but something that happens regularly and frequently. It is rational, systematic and incremental, rather than unusual and haphazard.

However, because it happens over time and is planned by different people and according to different goals, the overall outcome of planned evolution may not appear entirely rational. However carefully organisations try to establish strategies and to control events, there is inevitably a considerable slippage between intention and reality. As Mintzberg (1994) puts it, "strategic planning often spoils strategic thinking, causing managers to confuse real vision with the manipulation of numbers … the most successful strategies are visions, not plans" (p.107).

Thus it is not the case that a planned evolution process will always appear rational, if looked at from a distance or over a long time. Part of this is because information systems on the whole are changed incrementally, in small-scale stages – it is an evolutionary process rather than a revolutionary one.

This raises the interesting question as to whether there is in fact such as thing as a revolutionary change to an information system. In fact, these are remarkably rare. Almost all organisations have some sort of information system in place (not necessarily computerised) which serves as the basis for change. Even when organisations start from scratch with new forms of technology, or new business processes that throw away the old ones completely, there are still many residual aspects of the previous IS – the organisational culture, the skills people have built up working with or in the old IS environment, the form of business that the organisation is in, the people outside the organisation with whom it has dealings. Some of these things will stay the same, many will change somewhat as a result of the IS change, and only a very few will change completely. Certainly there will always be radical aspects of many IS change programmes, but they rest upon the framework of the status quo and are surrounded by aspects that are only changing in small ways or not at all. In that sense even the largest IS change is still evolutionary.

Unplanned evolution reflects the limitations of planning in a different way from the implied by Mintzberg. Information systems evolve in lots of ways that are not intended by their 'owners'. The people in the organisation change, and thus their technical skills, or knowledge of how to handle some sorts of information, or knowledge of how a piece of technology works, are lost to the organisation. The organisation changes – new strategies might be developed, or new business areas entered, or management might change and so affect many decisions, or the organisation (if it is commercial) might be taken over by another organisation. The technology also changes – an operating system or application software might receive updates from its suppliers, hardware might need upgrading, new platforms for application development become available. These are all changes to the information system, or to its immediate environment, but they are not changes made by the IS decision makers.

However, decision makers can be in a position to respond to the changes. This may be done through an analysis of the dynamics of IS evolution; through an examination of historical trends; or through strategic techniques for future readiness such as scenario planning.

Both planned and unplanned evolution can be understood further as a series of steps, and these are examined in the following sections.

A Framework for Planned Evolution

Planned evolution happens in various ways in different organisations.

- evaluation of the organisational impact of existing information systems
- examination of the value of existing ('legacy') software, and looking at ways to evolve that software
- decision making about which changes to the existing systems are appropriate, taking into account issues of participation and power
- making the IS ready for future changes ('future proofing'), and addressing risks that arise in the process of making changes.

The cyclical relationship between these stages is summed up in Figure 1 below. In general it is likely to be the case that the evolution process begins at the top of the cycle (with evaluation) and then proceeds step by step. However, it is possible to begin at another point in the cycle and work around it, and it is possible to consider two stages simultaneously (e.g. evaluation and legacy systems). This model is not intended to be followed as a 'recipe' for planned evolution, but rather as a simple context within which to place the four stages.

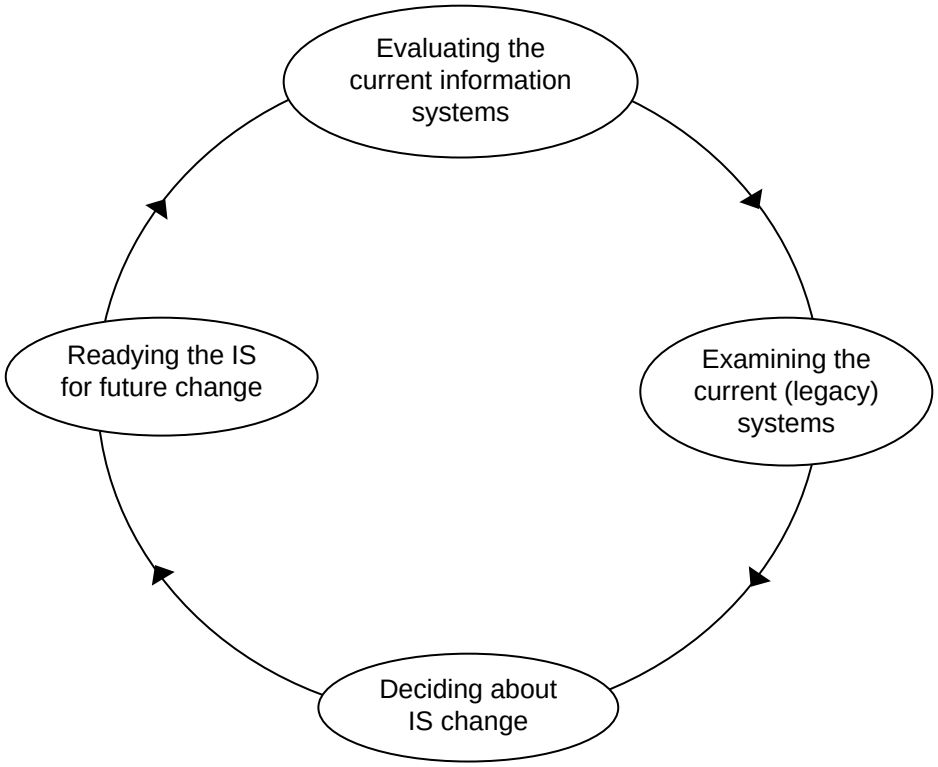

Figure 1: A model of planned evolution

Let us examine each of these four phases in slightly more detail.

The starting point in any process of planned evolution is to look at the existing information systems and to examine their appropriateness – that is, to evaluate them. IS evaluation is a topic that has been written about in great detail over a number of years. Many methods exist for IS evaluation, with various perspectives:

- financial, such as cost-benefit analysis and effective use of investment (e.g. Farbey et al., 1999);
- strategic, such as links between organisational needs and IS capability (e.g. Kaplan and Norton, 2004);
- technical, with questions of usability and technological effectiveness (e.g. Cox and Greenberg, 2000);

- interpretivist, with focus on multiple interpretations by different participants (e.g. Serafeimidis and Smithson, 2003);
- participatory, with explicit involvement of multiple stakeholders (e.g. McAulay et al., 2002);

None of these perspectives can give the whole truth on the current state of an information system, given the large number of different issues that are conflated under the term 'evaluation'. It is clear, however, that evaluation happens from a particular perspective (of the person or people making the judgement) – it cannot be neutral; that it happens in a particular context – in one setting, at one time, with one group of people involved; and that it happens using some kind of criteria and processes of making judgements. While some authors have combined some of the above perspectives (e.g. Ross et al., 1995), these combined methods are still contingent upon the context of the evaluation.

Having looked at IS evaluation, we move on to ask more technical questions around the effectiveness of legacy systems. This term is much misunderstood, being taken to refer simply to old software (perhaps running on previous versions of operating systems, or written in old programming languages). A more helpful definition of legacy systems is given by Sommerville (2001, p.598) as "old system[s] that still provide essential business services" – that is, legacy systems continue to have importance to the organisation making use of them.

A variety of technical solutions exist to legacy systems, summarised by Bennett et al (1999) as discarding (building a new system from scratch), outsourcing to a specialist organisation, freezing them (deciding to carry out no further work), carrying on maintaining them (despite difficulties), wrapping them within more modern programs that provide additional functionality (encapsulation), and reverse engineering (building new software on a different platform which exhibits the same functionality as the existing software). While these are largely technical options, the decisions must be taken on a socio-technical basis, such as the Renaissance method which assesses both the business value and the technical quality of the legacy system and makes decisions based on these two factors in combination (Warren, 1998).

Having evaluated the existing IS and its legacy systems, the next stage of planned evolution is to make decisions about the future of the IS. Decision making in organisations is always complex, ambiguous and has unpredictable consequences (March, 1991). While it is often taken to be a largely rational activity, it is deeply bound up with issues of power and politics, which in IS research has been emphasised by the recent resurgence of interest in critical theory (Brooke, 2002). For decision making around IS to be effective and lasting, it must take into account multiple stakeholder interests and power issues, rather than being treated as disinterested and rational.

Once decisions have been taken, they must be robust to possible future changes in the organisational environment. This process of 'future proofing' can take a number of forms. One route is to try to forecast the future of the information system, especially its technological evolution, using methods such as Delphi (Linstone, 1999) or roadmapping (Galvin, 2004), which in limited ways can be fruitful, although it fails to deal with the basic unpredictability of the future. A second route is that of strategic alignment –

ensuring that the organisational and technical change routes are closely connected – although the usefulness of this concept is highly contested (Hirschheim and Sabherwal, 2001; Ciborra, 1997). A third route to future proofing can be found through risk management – identifying potential risks to an information system through changes in its organisational environment, and finding ways to minimise or ameliorate those risks (Charette, 1996; Ould, 1999).

Activities in Unplanned Evolution

As briefly discussed above, there are clear limits to the usefulness of planned evolution, in particular arising from the complexity and unpredictability of the organisational environment. Three possible responses to these limits are discussed in this section: studying evolutionary dynamics, taking a historical approach, and scenario planning.

The first, and most theoretical, approach is to model the dynamics of unplanned evolution – to study the processes by which they change. This can be done using a number of methods, each of which has been used in studying IS, including system dynamics (Wolstenholme, 2003; Lehman and Ramil, 2003), complex adaptive systems (McMillan, 2004), coevolution (Mitleton-Kelly and Papaefthimiou, 2002; Cecez-Kecmanovic and Kay, 2001), and punctuated equilibrium theory (Saberwahl et al., 2001). Of these methods, coevolution seems to be particularly relevant, and fairly straightforward, allowing one to consider the interrelated changes between the different components of an information system (technology, people, organisation) as they each evolve over time. Peppard and Breu (2003, p.747) argue that coevolution “allows us to frame the process of mutual adaptation and change between business and IS strategies not just as a matter of alignment but as a dynamic interplay of coevolving interactions, interrelationships and effects”. Studying evolutionary dynamics is a complex approach, which in its current state as much involves theoretical development as practical application, but it is of considerable promise. Further integrative work has been done by others, such as McGann and Lyytinen (2004), who discuss the importance of improvisation in IS evolution.

A second analytical approach to unplanned evolution is to look at historical trends in IS change, and to try to draw conclusions from these trends. Through works such as Ward and Peppard (2002), we can identify key trends in IS history, such as embededness in organisations (the shift from support of specific functions to organiser of core business); application types (the shift from holding data to shaping the business); technology (from big and fixed to small and portable); and software development (from bespoke to tailoring). These trends can be taken as guides to possible future evolution of IS, although inevitably not in any simplistic way – Peppard and Ward (2004), for example, discuss the rise in IS capabilities as a rising trend that might usefully be taken into account.

A final route to considering unplanned evolution is the use of scenario planning – being aware of many possible futures and in a strategic position to respond to them (Matzdorf and Ramage, 1999; van der Heijden et al., 2002). By constructing possible scenarios of the environment surrounding their IS and organisation, decision makers may expand their awareness of possible forthcoming changes. In this sense, scenario planning improves the robustness of an IS to external change. While there are many analyses of

scenario planning in the general organisational literature, there are surprisingly few examples of its specific use within an IS context (perhaps partly due to confusion with the unrelated use of the term ‘scenario’ in human-computer interaction and software engineering, as discussed by Go and Carroll, 2004). Some exceptions include the extensive use of scenarios by ICL in the 1980s and 1990s (Ringland, 1998); a set of scenarios produced for the future of e-learning (Bell et al., 2004) and the organisational scenarios approach of Brooke (2000).

Of course, these approaches are not mutually incompatible, and each can inform the others, but they are not a cycle of activity in the way that can be seen for planned evolution. This is appropriate: unplanned evolution is, by its nature, unexpected, and responses to it necessarily need to vary according to circumstances.

Further Work

In this paper, I have examined the importance of focusing on the evolution of information systems. I have argued that evolution occurs in two forms, planned and unplanned, and that a particularly fruitful way of looking at unplanned evolution is by studying the coevolution of information systems and the organisations within which they are located. Truex et al. (1999, pp. 123) have argued that:

> [S]ystems should be under constant development, can never be fully specified and, like the organizations for which they are built, are subject to constant adjustment and adaptation. Since organizational change has become so important to organizational survival, IT systems must also incorporate continuous change. This incorporation goes beyond adaptable systems, and includes creating support for organizations that cannot help but emerge.

It is this concept of continuous development and emergent information systems that a focus on IS evolution, helps to bring us.

There is much further work to be done on the various models described above. While there seems to be a clear usefulness in delineating different forms of IS evolution as planned and unplanned (or perhaps intentional and unintentional, or proactive and reactive), the boundary between these types is much more porous than the above discussion allows, and examining this boundary is important. So too is understanding the nature of planned evolution, and determining whether the above model is in fact a good way of describing its form. Lastly, the analysis of the dynamics of unplanned evolution has much further to go, and methods are still needed that combine some of the above approaches. However, taking an evolutionary stance on IS change is definitely helpful, in assisting us in understanding IS responses to our highly complex and ever changing world.

Acknowledgements

This paper builds on conversations and work with a number of colleagues at the Open University. I would especially like to thank Clive Savory, Tony Nixon and Sue Betteridge.

References

Bell, M., Martin, G. and Clarke, T. (2004). Engaging in the future of e-learning: a scenarios-based approach. *Education+Training, 46(6/7)*, 296–307.

Bennett, K.H. (1996). Software evolution: past, present and future. *Information and Software Technology, 38*, 673–680.

Bennett, K.H., Ramage, M. and Munro, M. (1999). Decision model for legacy systems. *IEE Proceedings on Software, 146(3)*, 153–9.

Bennett, K.H. and Rajlich, V.T. (2000). Software maintenance and evolution: a roadmap. In Finkelstein, A. (Ed.), *The Future of Software Engineering* (pp. 73–87). New York: ACM Press.

Brooke, C. (2000). A framework for evaluating organisational choice and process redesign issues. *Journal of Information Technology, 15(1)*, 17–28.

Brooke, C. (2002). Critical perspectives on information systems: an impression of the research landscape. *Journal of Information Technology, 17(4)*, 271–83.

Burnes, B. (2004). Kurt Lewin and the Planned Approach to Change: A Re-appraisal. *Journal of Management Studies, 41*, 977-1002.

Cecez-Kecmanovic, D. and Kay, R. (2001). IS–organization coevolution: the future of information systems. *Proceedings of the 22nd International Conference on Information Systems,* 363–71.

Charette, R.N. (1996). The mechanics of managing IT risk. *Journal of Information Technology, 11(4)*, 373–8.

Checkland, P.B. and Holwell, S. (1998). *Information, Systems and Information Systems: Making Sense of the Field*. Chichester: John Wiley.

Ciborra, C. (1997). De profundis? Deconstructing the concept of strategic alignment. *Scandinavian Journal of Information Systems, 9(1)*, 67–82.

Cox, D. and Greenberg, S. (2000). Supporting collaborative interpretation in distributed groupware. *Proceedings of the ACM Conference on Computer-Supported Co-operative Work (CSCW '00)*, 289–98.

Farbey, B., Land, F., Targett, D. (1999). IS evaluation: a process for bringing together benefits, costs, and risks. In Currie, W. and Galliers, B. (Eds.), *Rethinking Management Information Systems*. Oxford: Oxford University Press.

Galvin, R. (2004). Roadmapping – a practitioner's update. *Technological Forecasting and Social Change, 71(1–2)*, 101–3.

Go, K. and Carroll, J.M. (2004). The blind men and the elephant: views of scenario-based system design. *interactions, 11(6)*, 44–53.

Hirschheim, R. and Sabherwal, R. (2001). Detours in the path towards strategic information systems alignment. *California Management Review, 44(1)*, 87–108.

Kaplan, R. and Norton, D. (2004). Measuring the strategic readiness of intangible assets. *Harvard Business Review, 82(2)*, 52–63.

Lehman, M. and Ramil, J. (2003). Software evolution – background, theory, practice. *Information Processing Letters, 88(1–2)*, 33–44.

Linstone, H.A. (1999). *Decision Making for Technology Executives: Using Multiple Perspectives to Improve Performance*. Boston: Artech House.

March, J.G. (1991). How decisions happen in organizations. *Human–Computer Interaction, 6(2)*, 95–117.

Matzdorf, F. and Ramage, M. (1999). Out of the box – into the future. *Organisations and People, 6(3)*, 29–34.

McAulay, L., Doherty, N. and Keval, N. (2002). The stakeholder dimension in information systems evaluation. *Journal of Information Technology, 17(4)*, 241–55.

McGann, S. and Lyytinen, K. (2005). How Information Systems Evolve by and for Use. *Sprouts: Working Papers on Information Environments, Systems and Organizations, 5(1)*, 33-48. http://sprouts.case.edu/2005/050103.pdf

McMillan, E. (2004). *Complexity, Organizations and Change*. London: Routledge.

Mintzberg, H. (1994). The fall and rise of strategic planning. *Harvard Business Review, 72(1)*, 107–14.

Mitleton-Kelly, E. and Papaefthimiou, M.C. (2002). Co-evolution of diverse elements interacting within a social ecosystem. In Henderson, P. (Ed.) Systems engineering for business process change: new directions (pp. 253–273). London: Springer,.

Ould, M. (1999). *Managing Software Quality and Business Risk*. Chichester: John Wiley.

Peppard, J. and Breu, K. (2003). Beyond alignment: a coevolutionary view of the information systems strategy process. *Proceedings of the 24th International Conference on Information Systems,* 743–50.

Peppard, J. and Ward, J. (2004). Beyond strategic information systems: towards an IS capability. *Journal of Strategic Information Systems, 13(2)*, 167–94.

Ringland, G. (1998). *Scenario Planning: Managing for the Future*. Chichester, UK: John Wiley.

Ross, S., Ramage, M. and Rogers, Y. (1995). PETRA: participatory evaluation through redesign and analysis. *Interacting With Computers, 7(4)*, 335–60.

Saberwahl, R., Hirschheim, R. and Goles, T. (2001). The dynamics of alignment: insights from a punctuated equilibrium model. *Organization Science, 12(2)*, 179–97.

Serafeimidis, V. and Smithson, S. (2003). Information systems evaluation as an organizational institution – experience from a case study. *Information Systems Journal, 13(3)*, 251–74.

Sommerville, I. (2001). *Software Engineering (6th edition)*. Harlow, UK: Addison-Wesley.

Truex, D.P., Baskerville, R. and Klein, H. (1999). Growing systems in emergent organizations. *Communications of the ACM, 42(8)*, 117–23.

Van der Heijden, K., Bradfield, R., Burt, G., Cairns, G. and Wright, G. (2002). *The Sixth Sense: Accelerating Organizational Learning with Scenarios*. Chichester, UK: John Wiley.

Ward, J. and Peppard, J. (2002). *Strategic Planning for Information Systems (3rd edition)*. Chichester, UK: John Wiley.

Warren, I. (1998). *The Renaissance of Legacy Systems*. London: Springer.

Wolstenholme, E.F. (2003). The use of system dynamics as a tool for intermediate level technology evaluation: three case studies. *Journal of Engineering and Technology Management, 20(3)*, 193–204.