

# Information Technology Governance and Decision Support Systems

Rob MEREDITH

*Centre for Decision Support and Enterprise Systems Research, Monash University  
PO Box 197, Caulfield East, Victoria 3145, Australia  
Rob.Meredith@infotech.monash.edu.au*

**Abstract.** Information technology governance is the set of organizational structures that determine decision-making rights and responsibilities with regard to an organisation's information technology assets. Although an important sub-field of information technology, little research has been done on the issues relating to the governance of decision support systems. This paper argues that decision support systems are significantly different to other kinds of information technology, and that this means there is a need to consider issues specific to their governance. Orlikowski's (1992) theory of the structuration of technology is used to highlight the fundamental differences between decision support systems and other kinds of information technology, and their respective relationships with organizational structures. Some preliminary recommendations and suggestions for further research into issues of decision support systems governance are made.

**Keywords.** IT governance, DSS, evolutionary development, structuration theory

## Introduction

Weill & Ross (2004) write that "IT governance is an issue whose time has come" (p.216). As a result of a renewed interest in corporate governance in general, as well as criticisms of information technology (IT) as a strategic tool (Carr, 2003), there is increasing pressure on the IT industry to demonstrate value to organisations, and to put in place organizational structures that can help to ensure that IT, as a corporate asset, is managed responsibly and effectively.

Most IT governance literature focuses on the governance of IT in general, giving little consideration to the characteristics of a given technology, rather focusing on political power structures in organisations and how this relates to particular governance decision structures (Brown, 1997). In particular, attention has focused on whether the IT department or non-IT business units should dominate the determination of IT policy.

Decision support systems (DSS) are a significant aspect of today's IT industry. Although the current industry term for the concept is 'business intelligence' (BI) (Arnott & Pervan, 2005), DSS has a rich academic and industry history that stretches back to the 1970s. However, despite both corporate and IT governance receiving a significant amount of research attention, little has been written on governance issues for DSS specifically. The exception is a small body of literature on data warehousing governance (see Ang & Teo, 2000; Suritpayapitaya, Janz, & Gillenson, 2003; Watson, Fuller, & Ariyachandra, 2004, for example).

The purpose of this essay is to argue that DSS are different to other kinds of information systems in that they are 'chaotic' and 'subversive.' It will further argue that governance is, in large part, about control and enforcement. Given that approaches to controlling and managing chaotic and subversive processes or systems are significantly different to the control and management of predictable, stable processes or systems, it follows that approaches to DSS governance need to be different to the governance of other kinds of IT.

## **1. Decision Support Systems**

### *1.1. Kinds of DSS*

The term 'Decision Support System' is commonly attributed to Gorry & Scott Morton (1971), who describe a framework of decision-support tools based on decision categories of "operational control," "management control," and "strategic planning" and a dichotomy of "programmed" versus "un-programmed" decision problems. Subsequently, various kinds of DSS have emerged providing varying levels of support (Jelassi, Williams, & Fidler, 1987; Keen, 1987) ranging from the purely informational (i.e. passive support) to normative tools that recommend a course of action.

Industry practice has been dominated, at various times, by various kinds of DSS. In addition to spreadsheet-based DSS, systems such as Executive Information Systems (EIS), Group Support Systems (GSS), Negotiation Support Systems, Intelligent DSS, Knowledge Management-based DSS, and Data Warehouses have all been used to support managerial decision-making (Arnott & Pervan, 2005). Current industry practice focuses on the use of data warehouses (see, for example, Kimball, Reeves, Ross, & Thornthwaite, 1998) to provide the data infrastructure for so-called Business Intelligence (BI) systems. BI is the current term for what are functionally equivalent to the EIS of the 1980s and 1990s (Arnott & Pervan, 2005).

DSS therefore differ on a number of dimensions, including the technological approach adopted, the kind of support offered, the level of 'structuredness' of the decision supported, the level of management supported and the number of decision makers involved (one or many). They range from small, informal systems through to large-scale systems similar in nature to enterprise resource planning (ERP) systems. This is not to say that the size and scale of the DSS are positively correlated with their impact: some short-lived, small-scale DSS have had a profound impact (see Arnott, 2004, for a description of so-called "ephemeral" systems).

### *1.2. DSS as Chaotic Systems*

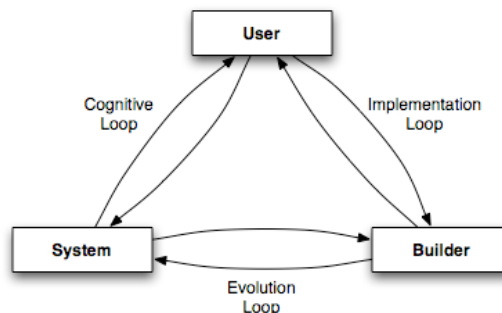
Arguments that DSS are different to other kinds of information systems have been made since at least Gorry and Scott Morton (1971). This difference is particularly relevant when thinking about the development lifecycle for DSS. Traditional, engineering-based systems development processes such as the 'waterfall' model (Royce, 1970) do not cope well with the dynamic design requirements and usage patterns typical of DSS.

Keen (1980) was the first to articulate the key factors that make development of decision support systems of any variety different to the development of other kinds of information systems. The primary reason for this difference is that any decision

problem that benefits from the kind of analysis that a DSS can provide (specifically, semi- and un-structured decision problems) necessarily involves ambiguity and uncertainty. This makes the initiation and analysis phases in Royce's (1970) waterfall model difficult to complete. The kind of requirements specification that needs to occur in these phases comes from a detailed understanding of the task that the system is being designed to support. In a semi- or un-structured decision situation, it is this very understanding that the DSS is supposed to provide assistance with.

The result is that any DSS designed to assist with these kinds of decisions will be developed with an initially incomplete understanding of the users' requirements. The system itself "shapes" the users' understanding of the decision problem, and therefore the users' information and support needs (Keen, 1980). This in turn leads to novel, unanticipated uses of the system, and a need to evolve the functionality of the DSS.

Keen (1980) conceptualized the development environment for any DSS using the framework depicted in Figure 1 below. In particular, Keen showed that interaction between the user and the system drives a need for evolutionary change as the system helps formulate the user's understanding of the decision problem, and the user utilizes the system in novel and unanticipated ways as a result.



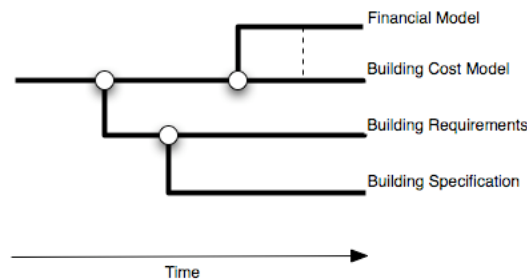
**Figure 1.** Keen's Adaptive Framework for DSS (Adapted from Figure 1, Keen, 1980).

This "cognitive loop" is the basis of the difference between transaction-processing systems and DSS, and Keen's (1980) argument is that a need for evolutionary development and use necessarily holds true for any kind of DSS: if not, then the system cannot possibly provide meaningful 'support.' The evolutionary process itself – the act of changing the system through close interaction with the user as they use the system – as well as system use, provides insight to the decision problem.

Development of DSS must, therefore, be evolutionary. There have been various development methodologies proposed for DSS, and most incorporate this idea of evolutionary adaptation to user requirements to a varying degree. Sprague and Carlson (1982), for example, describe four kinds of "flexibility" required of a DSS: flexibility for the user to solve the decision problem; flexibility to change the functionality of the DSS; flexibility to adapt a new DSS application; and flexibility to evolve the underlying technology. Similarly Arnott (2004) described two different kinds of DSS adaptation: within- and between-application evolution.

The adaptation process can take place quite rapidly. Arnott (2004) describes a DSS developed over a period of six weeks. The DSS evolved into four distinct systems used to investigate various aspects of the decision problem, including spreadsheet-based financial models and CAD-based architectural plans, shown in Figure 2. The development path was characterized by opportunism and unpredictability. DSS

evolution is often dependent on factors outside of the control of the developers and the organisation, including the user's ability to understand the decision problem as well as technical and political disruptions (Arnott, 2004).

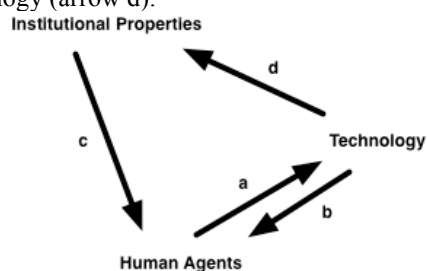


**Figure 2.** Evolution of a Building Project DSS (Adapted from Figure 4 in Arnott, 2004).

This unpredictability makes DSS a kind of ‘chaotic’ system in the sense that future properties and characteristics of the DSS design cannot be foreseen. This contrasts with the comparative stability and predictability of transaction-processing systems.

### 1.3. DSS as Subversive Systems

The way in which organizational structures are embedded in technology, or the way in which technology itself influences organizational structures frames how we understand the design and use of information systems (DeSanctis & Poole, 1994; Orlikowski, 1992). Orlikowski (1992), drawing on Giddens’ (1984) structuration theory, demonstrates that the influence between technology (as a tool and mediator of human agency) and organizational structure is bi-directional. Orlikowski’s (1992) “structurational model of technology” is depicted in Figure 3. Technology results from human actions such as design and development (arrow a), but also acts as a medium for human action (arrow b) through technology use. Institutional properties both influence what human actions are acceptable and/or possible with regards to technology (arrow c) and are shaped by technology (arrow d).



**Figure 3.** Orlikowski's (1992) Structurational Model of Technology

Structuration theory is based on the three concepts of ‘signification’ (signs and language), ‘legitimation’ (norms and values, accepted ways of doing things), and ‘domination’ (means of enforcing and controlling human action) (Giddens, 1984). Together, these three constitute organizational structures, and Orlikowski (1992) argues that technology acts upon each of them in one of two ways: either through reinforcement or transformation. Generally, technology is intended to reinforce existing organizational structures, rather than transform them (Orlikowski, 1992). Arguably,

even when the intent is to transform organizational structures, the intent of the technology is to embed and reinforce the new structure. Orlikowski (1992, p.411) asserts: “[users] are generally unaware of their role in either reaffirming or disrupting an institutional status quo.”

Further, when technology is used to transform rather than reinforce, it is usually in situations characterized by “high levels of stress, ambiguity and unstructured ... situations” (Orlikowski, 1992, p. 412). In such situations, workarounds and other anticipated uses of the technology ‘subvert’ organizational structures.

Gorry and Scott Morton (1971) differentiated between systems that are developed to support semi- or un-structured decisions (DSS) and systems designed to support structured decision problems (“structured decision systems”, or SDS). While SDS support recurring, unambiguous decisions, DSS are designed in an unstructured (i.e. novel, ambiguous, and stressful) environment. The unanticipated uses of transformative technologies and consequent subversion of organizational structures described by Orlikowski (1992) have a direct parallel in DSS use as described by Keen (1980).

DSS are inherently subversive. While other kinds of technology can be transformative, their subversion of organizational structures is often unanticipated and unplanned. DSS are *intentionally* subversive since, by design, they directly influence decisions on organizational goals, policies, activities and direction.

## 2. IT Governance

Information technology (IT) governance outlines the “decision rights and accountability framework [that] encourage desirable behavior in the use of IT” (Weill & Ross, 2004, p.8). It defines the principles, procedures, responsibilities and other normative aspects of managing an organisation and its resources (Weill & Ross, 2004).

As a subset of corporate governance, IT governance takes into account general corporate governance doctrines and strategies and applies them in the context of IT (Weill & Ross, 2004). Much of the debate in the academic literature has, in the past, been concerned with power-structure issues, such as whether a given governance arrangement was centralized or decentralized, or a hybrid form of the two (Brown, 1997; Sambamurthy & Zmud, 2000; Suritpayapitaya et al., 2003).

The power-structure view is a relatively narrow lens through which to view all of the issues related to IT governance. The work of Weill & Ross (2004; 2005) expands the scope of debate on IT governance to include a range of decision-types in addition to who specifically makes those decisions. These two issues – what decisions need to be made, and who should make them – are the basis of a matrix used to analyze the IT governance arrangements in a number of different organisations. Weill & Ross (2004) define the following IT governance decisions:

- *IT Principles*. How IT should support the business.
- *IT Architecture*. Requirements for organizational standards and integration of systems.
- *IT Infrastructure Strategies*. Requirements for supportive services for IT applications.
- *Business Application Needs*. Requirements for information systems to support the business, whether developed internally or purchased.
- *IT Investment*. Selection and funding of IT initiatives.

They also outline the following archetypal arrangements for making these decisions:

- *Business Monarchy*. Centralized decision making by senior business managers/executives.
- *IT Monarchy*. Centralized decision making dominated by the CIO / IT department.
- *Feudal*. Decentralized decision making by business unit managers.
- *Federal*. A hybrid approach combining decision making by senior executives as well as business unit managers. This may or may not include the IT department.
- *IT Duopoly*. Decision making by the IT department and one other entity – either business unit management, or senior executives.
- *Anarchy*. Isolated, uncoordinated decision making by individuals or small groups.

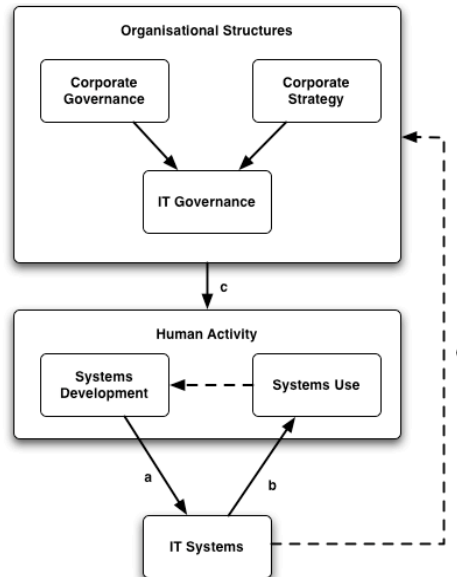
Table 1, below, shows the most common governance structures found from Weill & Ross' (2004) research:

**Table 1.** Recommended IT Governance Arrangements from Weill & Ross (2004).

	<b>IT Principles</b>	<b>IT Architecture</b>	<b>IT Infrastructure Strategies</b>	<b>Business Application Needs</b>	<b>IT Investment</b>
Business Monarchy					
IT Monarchy					
Feudal					
Federal					
IT Duopoly					
Anarchy					

Weill & Ross (2004) point out that, in addition to the two issues described above, there is the further question of how governance decisions are to be implemented. Typical governance instruments include the use of formal policy statements, project management methodologies, documents outlining standards, advisory councils and committees, chargeback structures, appointment of business/IT relationship managers and service level agreements (Weill & Ross, 2004). All of these instruments are means of controlling IT activities in an organisation. As with other management instruments, the typical intent is to increase managerial control, coordination and predictability.

As a subset of corporate governance, IT governance is used to ensure that an organisation's information systems are consistent with and embody organizational structures. Corporate governance principles and the strategic direction of the firm dictate IT governance principles, which in turn dictate the features, functionality, operation and use of individual information systems, as depicted in Figure 4.



**Figure 4.** Flow of Corporate Governance Principles Through to IT Systems, adapted from Figure 3. Arrow labels have been maintained from Figure 3 for cross-reference. Dashed lines indicate a ‘weaker’ influence.

Based on Figure 3, Figure 4 decomposes Orlikowski’s (1992) “institutional properties” into corporate governance, strategy and IT governance, renaming them ‘organizational structures.’ It also decomposes Orlikowski’s “human activity” into systems development and systems use. Corporate governance and strategy filter through IT governance and its corresponding governance instruments, which, through the activity of designing and developing the system (arrow c, then a), ensures that these structures are reflected in the deployed information system. In turn, this encourages a use of the system that is consistent with corporate organizational structures (arrow b).

An arrow has been introduced between systems use and development to reflect system redesign based on users’ experiences. Both this, and d have been shown as dashed lines to indicate the relative weakness of the influence. In the case of the influence of use on development, modifications will typically be minor and more along the lines of maintenance rather than a full redesign. In the case of d, this influence is typically one of reinforcement, or, when transformative, it is often unintentional (Orlikowski, 1992) and so not as direct or deliberate as the other influences.

### 3. IT Governance and DSS

The assumption behind Weill & Ross’s (2004) framework is that each intersection of governance decision and type of organizational culture represents a coherent, homogenous approach to IT governance within the domain of the relevant decision-maker’s mandate. That is, in the case of a feudal information culture, each business unit determines the IT principles etc that apply to all IT within that business unit. Likewise where the mandate is enterprise-wide, all IT within the organisation will adhere to the same set of principles, architecture, and so on.

This assumption is present, not only in Weill & Ross (2004), but also in the earlier IT governance literature. Although the IT governance debate acknowledged the possibility of different governance models for different parts of an organisation (at least under a decentralized or hybrid approach), the assumption is that governance arrangements will differ according to aspects of the organizational structure such as business units or the relative dominance of a central IT department. There is no recognition that approaches to governance may need to differ according to characteristics of the technology itself, in addition to characteristics of the organisation.

In section 1, it was argued that DSS are different in the way that they are developed and used compared to other kinds of IT. Whereas transaction-processing systems are developed to enforce and control one or more business processes, with the intent that the system will be as stable as possible, DSS are necessarily unstable and intentionally subversive.

The chaotic and subversive nature of DSS implies a different set of relationships between a DSS and organizational structures compared to other kinds of IT. There is an inherent tension between a governance structure predicated on control and predictability needed for transaction-processing systems, and the unpredictable development process that is necessarily a characteristic of DSS. Not only is control and predictability difficult to enforce with DSS, it is undesirable: “The label ‘Support System’ is only meaningful in situations where the ‘final’ system must emerge through an adaptive process of design and usage” (Keen, 1980, p.15).

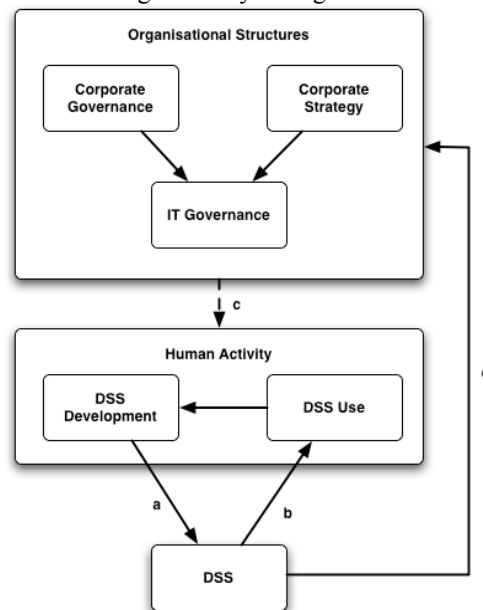
The chaotic nature of DSS development and use has two implications for the model in Figure 4. The first is that the influence between systems use and systems development is much stronger and more explicit than with other kinds of IT. It should be expected that as a DSS is being used, the user learns and experiments and that this will drive pressure for system change. IT and corporate governance instruments, put in place to manage and control systems development and use, can have a deleterious effect on this dynamic. One such case is described in Arnott (2006) where a business intelligence project at a financial services company failed, in large part, because of the project management methodology that was employed. The administrative overheads associated with the methodology “throttled” (Arnott, 2006, p. 720) development on the business intelligence project to the point where the project was cancelled fourteen months after it began, with no DSS functionality actually delivered.

This leads to the second implication for the model in Figure 4. The link between organizational structures and DSS development and use should be less controlling (and more enabling) than for other kinds of IT. Although Weill & Ross acknowledge the need for an enabling aspect to IT governance (Weill & Ross, 2004, pp.20-21), they don’t see this occurring outside the normal IT governance structures in the organisation. While innovation is possible in such situations (Weill & Ross cite several examples), it is unreasonable to expect that this would typically work for the rapid, continuous and chaotic evolution required for DSS. DSS users and developers need the freedom and to be able to evolve the system as needs be, without having to continuously second guess or report to a ‘stifling’ (Arnott, 2006) layer of bureaucracy. This devolution of power to small teams of DSS developers and users suggests that for DSS, an ‘anarchic’ decision-making structure would be more appropriate than the more structured approaches recommended by Weill & Ross (2004). This is supported by Arnott (2006) where a subsequent, anarchic project was undertaken successfully at the same organisation.

The subversive nature of DSS also means a much more explicit and deliberate influence of the DSS on organizational structures. Because of the nature of decisions

that DSS support, this often means that decision-makers are directly considering some aspect of organizational strategy or structure. By definition, a decision-making process is one where a commitment is formed to a particular course of action. In an organizational setting, this often means that a new course of action for the organisation is being committed to, thereby directly affecting organizational structures.

Figure 5 incorporates these changes to Figure 4 for DSS. Arrow c is now less deterministic, while the influences of systems use on development and the system on organizational structures are both significantly stronger.



**Figure 5.** Structuration Theory Applied to DSS

### 3.1. Implications for the Governance of DSS

The chaotic and subversive nature of DSS implies that the necessary assumptions for the governance of DSS are different to those for the governance of other IT resources. Where the latter is intended to provide a capacity for control, predictability and conformance with corporate governance structures, the former can only be successful in an environment that encourages flexibility and experimentation.

There is a parallel between the idea of managing DSS development and use as a learning process (Keen, 1980), and managing creative processes in organisations, characterized by Ewing, Napoli, & West (2000-2001) as “idiosyncratic, unpredictable, random, [and] anarchic” (p. 163). Such processes require very different managerial mindsets to other organizational processes (Amabile, 1998; Ewing et al., 2000-2001), and it is reasonable to assume that approaches to managing creative processes in organisations hold insight for managing DSS, and by extension, governing DSS.

Management of creative processes tends work better when not characterized by direct control and supervision (Ewing et al., 2000-2001). Amabile (1998) also points to the need for freedom and encouragement to explore and experiment. Excessive administration, surveillance, and a lack of autonomy tend to restrict such processes

(Nagasundaram & Bostrom, 1994). The same is true of DSS development and use: there is a need for flexibility to deviate from the governance structures put in place to manage and control other kinds of IT.

This is not, however, an argument for a completely free reign. Rather, it is an argument for an approach to the governance of DSS that places trust in the DSS team (including the users) to make development decisions within well-defined boundaries. Clearly, it is not desirable to arbitrarily violate IT management procedures in such a way as to negatively impact on the operation of other IT systems. Each deviation from accepted IT governance principles in the organisation should be made deliberately, in full awareness of the potential implications. The corollary of this is that the organisation must have clearly articulated IT governance structures already in place.

Weill & Ross (2004) also acknowledge the need for some degree of creative IT experimentation in organisations (pp. 41-42), and argue that this kind of experimentation should be undertaken in an environment with explicit boundaries. The DSS team needs to be clear about what can or cannot be done. In other words, DSS development and use should operate in a kind of governance ‘sandbox’ where the team is free to experiment and adapt the system outside of normal governance structures.

DSS governance can therefore be characterized by the following points:

1. An organisation should have clear and explicit IT governance structures generally.
2. DSS should not be strictly bound by these structures.
3. There should be a clearly defined scope for DSS development and use, including budget and resources, goals and anticipated benefits.
4. This scope should not be overly constraining, and should be revised regularly.
5. There should be trust placed in the DSS team to develop the DSS as they see fit within the broad scope defined above.

#### **4. Conclusion and Directions for Future Research**

With a renewed focus on corporate governance in recent times as a result of legislative changes such as the Sarbanes-Oxley Act of 2002 in the US, and related legislation in other jurisdictions (Weill & Ross, 2004), as well as questions regarding the strategic benefit IT can deliver to organisations (Carr, 2003), IT governance is currently an important issue for the IT industry. IT governance helps to ensure that an important organizational resource is managed effectively, and that organizational structures are enforced through the systems that people use to do their work.

To date, however, little consideration has been given to the relationship between governance structures and the characteristics of the technology being governed. While extensive consideration has been given to the relationship between governance and organizational power structures, the same cannot be said of how principles inherent in a given technology (such as evolutionary development for DSS) integrates with various governance structures.

The motivation for IT governance in general is control and the enforcement of organizational structures. For most IT systems – especially transaction-processing systems – this is a reasonable aim. However, this is not the case for DSS.

Unlike other kinds of IT, DSS development and use is chaotic. DSS use subverts organizational structures. The typical assumptions behind the governance of IT in general are incompatible with these two characteristics.

DSS, therefore, may require a more flexible approach to governance: one that trusts DSS developers and users to use their judgment to assess the appropriateness of changes to the design and use of the system, rather than having to go through the bureaucratic procedures appropriate for other kinds of systems. This autonomy, though, should be used in light of clear boundaries and requirements: DSS governance should not be *carte blanche*.

The issues highlighted in this essay raise a number of interesting questions for future research:

- To what extent do different kinds of DSS require the kinds of governance recommendations in section 3.1? Do large-scale DSS such as business intelligence systems and data warehouses benefit from the same governance freedoms as smaller scale personal DSS?
- To what extent does the decision problem, or task, determine governance requirements for DSS?
- How much scope should be given, or conversely, how restricted should governance boundaries be for DSS development and use?
- What mechanisms are appropriate to encourage DSS evolution to maximize benefits to the organisation?
- How can conflicts between DSS governance and other organizational structures (including general IT governance) be resolved? How can other IT assets be protected from changes in, and resource demands by, DSS?
- What is the relationship between characteristics of a given technology and the assumptions that underpin its governance?

## References

- Amabile, T. M. (1998). How to Kill Creativity. *Harvard Business Review*, 76(5), 76-87.
- Ang, J., & Teo, T. S. H. (2000). Management Issues in Data Warehousing: Insights from the Housing and Development Board. *Decision Support Systems*, 29, 11-20.
- Arnott, D. (2004). Decision Support Systems Evolution: Framework, Case Study and Research Agenda. *European Journal of Information Systems*, 13, 247-259.
- Arnott, D. (2006). *Data Warehouse and Business Intelligence Governance: An Empirical Study*. Paper presented at the 2006 IFIP WG8.3 International Conference on DSS.
- Arnott, D., & Pervan, G. (2005). A Critical Analysis of Decision Support Systems Research. *Journal of Information Technology*, 20(2), 67-87.
- Brown, C. V. (1997). Examining the Emergence of Hybrid IS Governance Solutions: Evidence From a Single Case Site. *Information Systems Research*, 8(1), 69-94.
- Carr, N. (2003). IT Doesn't Matter. *Harvard Business Review*, 81(5), 41-49.
- DeSanctis, G., & Poole, M. S. (1994). Capturing the Complexity in Advanced technology Use: Adaptive Structuration Theory. *Organizational Science*, 5(2), 121-147.
- Ewing, M. T., Napoli, J., & West, D. C. (2000-2001). Creative Personalities, Processes and Agency Philosophies: Implications for Global Advertisers. *Creativity Research Journal*, 13(2), 161-170.
- Giddens, A. (1984). *The Constitution of Society*. Berkeley: University of California Press.

- Gorry, G. A., & Scott Morton, M. S. (1971). A Framework for Management Information System. *Sloan Management Review*, 13(1), 55-70.
- Jelassi, M. T., Williams, K., & Fidler, C. S. (1987). The Emerging Role of DSS: From Passive to Active. *Decision Support Systems*, 3(4), 299-307.
- Keen, P. G. W. (1980). Adaptive Design for Decision Support Systems. *Data Base*, 12, 15-25.
- Keen, P. G. W. (1987). Decision Support Systems: The Next Decade. *Decision Support Systems*, 3(3), 253-265.
- Kimball, R., Reeves, L., Ross, M., & Thornthwaite, W. (1998). *The Data Warehouse Lifecycle Toolkit*. New York: Wiley.
- Nagasundaram, M., & Bostrom, R. P. (1994). The Structuring of Creative Processes Using GSS: A Framework for Research. *Journal of Management Information Systems*, 11(3), 87-114.
- Orlikowski, W. J. (1992). The Duality of Technology: Rethinking the Concept of Technology in Organizations. *Organizational Science*, 3(3), 398-427.
- Royce, W. W. (1970). *Managing the Development of Large Software Systems*. Paper presented at the IEEE/WESCON Conference, August 1970.
- Sambamurthy, V., & Zmud, R. W. (2000). Research Commentary: The Organising Logic for an Enterprise's IT Activities in the Digital Era - A Prognosis of Practice and a Call for Research. *Information Systems Research*, 11(2), 105-114.
- Sprague, R. H., & Carlson, E. D. (1982). *Building Effective Decision Support Systems*. Englewood Cliffs, New Jersey, USA: Prentice Hall.
- Suritpayapitaya, S., Janz, B. D., & Gillenson, M. (2003). The Contribution of IT Governance Solutions to the Implementation of Data Warehouse Practice. *Journal of Database Management*, 14(2), 52-69.
- Watson, H. J., Fuller, C., & Ariyachandra, T. (2004). Data Warehouse Governance: Best Practices at Blue Cross and Blue Shield of North Carolina. *Decision Support Systems*, 38, 435-450.
- Weill, P., & Ross, J. (2004). *IT Governance: How Top Performers Manage IT Decision Rights for Superior Results*. Boston, USA: Harvard Business School Press.
- Weill, P., & Ross, J. (2005). A Matrixed Approach to Designing IT Governance. *Sloan Management Review*, 46(2), 25-34.