Implementation Differences Between DSS/BI and Other Types of Information Systems

Efrem G. Mallach, University of Massachusetts Dartmouth, North Dartmouth, MA 02747; E-mail: emallach@umassd.edu

ABSTRACT

This paper considers concerns that arise in the implementation of TPS and those that arise in the implementation of DSS/BI systems, noting differences between the two. Important management focus areas in implementing TPS include low-level user training, extensive user testing, change management and cut-over strategy. With DSS/BI, attention should focus on data quality, higher-level user training, having an executive "champion" and adapting the system to its users. Managers approaching an implementation project must be aware of these differences, as should academics who write textbooks or who teach from textbooks that do not cover them adequately.

1. INTRODUCTION

No business information system is of value unless people use it to achieve business objectives. An unused system may be an impressive monument to technology but cannot justify the use of corporate funds. Therefore, no matter how well a system was designed and developed, it must be put to use before its goals can be achieved. That is the role of *implementation*.

However, not all systems are created equal in this regard. The critical success factors for implementing one type of system are not necessarily the same as those for implementing a different one. While this may seem obvious when stated, there are two practical problems:

1. Research tends to focus on one type of system in isolation. It does not compare the conclusions of one study, for one type of system, with the conclusions of research on other types of systems.
2. Even what is known about implementation of different types of systems has yet to make its way into the textbooks used to teach future practitioners and users of IS.

2. DEFINITIONS AND DISCUSSION OF IMPLEMENTATION

Implementation, as used here, is the process of preparing an information system for use. Kwon & Zmud (1987), in their seminal paper on the subject, define it as:

...an organizational effort directed towards diffusing appropriate information technology within a user community.

A similar definition comes from a popular introductory MIS textbook (Laudon & Laudon 2004):

...all organizational activities working toward the adoption, management and routinization of ... a new information system.

Both these definitions focus on organizational aspects. By the time implementation begins, technical work is (or should be!) complete.

This is not the only possible definition. The problem arises from different views of when something is ready for use. The major dichotomy is between the user and vendor communities. Vendors see a system, or whatever system component they sell, as ready for use when it's ready to ship to customers. User organizations see it as ready for use when it has been customized as needed, its databases fully populated, and its client software installed on individual users' desks. The user's implementation starts where the vendor's ends. The vendor's view includes more technical activities than does the user's. Attempts to resolve this conflict, without offending anyone who considers a given activity to be part of implementation, result in defining it to cover virtually everything after drawing data flow and entity-relationship diagrams—a "definition" so inclusive as to be nearly meaningless.

We will use the user view here, not the vendor's. This reflects our focus. It is not a value judgment of "rightness" or "wrongness."

Accordingly, we consider areas such as database preparation, system conversion, training, and other activities that occur after programming and database development are done.

The questions this paper addresses are: Are there significant differences in these areas between (a) decision support and business intelligence (DSS/BI) systems and (b) other enterprise-level information systems? (These latter will be referred to as "transaction processing systems," TPS for short, recognizing that this term often has connotations of older technology which are not intended here.) If there are, what does this imply?

The distinction here is between systems that update the organization's operational database and those that use it to support management activities but do not update it. The former are typically used at lower levels in the organization to carry out routine activities; the latter are used by managers and knowledge workers to support decision making and related higher-level tasks. The terminology *DSS/BI* versus *TPS* is a convenient way to express this but is not meant to restrict the discussion to specific technologies. Modern integrated enterprise-level systems such as ERP, CRM and SCM increasingly include DSS/BI capabilities that would have been provided by separate packages not long ago. Even when this is the case, however, the two sets of functions have different user communities and thus different implementation requirements. Phrasing the question in terms of modules of an integrated system, rather than separate systems, would change some of the discussion but would not affect either the underlying issues or the conclusions.

3. IMPLEMENTATION OF TPS

The stages of implementation for Enterprise Resource Planning systems (ERP), a prime example of a large-scale, enterprise-level TPS, have been studied extensively. Over 15 years ago, the implementation of its Material Requirements Planning (MRP) predecessor was studied in (Cooper & Zmud 1990). Rajagopal (2002) discussed a model for ERP implementation. Lai (2006) provides guidelines for successful ERP implementation, based on experience in China.

An example of TPS implementation is given in (Mandal & Gunasekaran 2002). They discuss a large (6000-employee) Australian wine producer and exporter's implementation of SAP R/3 ERP software. The system focused on online inventory control. In addition to the customization that any SAP installation requires, a separate bar-coding system was also developed and integrated with SAP. Key aspects of implementation here included:

- The system was tested extensively by its prospective users, with formally maintained lists of open issues and timing to complete all tasks.
- Shop floor personnel, many of whom had never before used a computer, were trained on the new system before its roll-out.

- Additional on-site training and support was provided after roll-out.

Some employees, whose jobs had not previously required them to use a computer, felt their new job requirements justified additional compensation. This was handled by providing overtime payment during the learning phase, returning to normal pay levels once computers were no longer seen as anything special.

Muscatello & Parente (2006) iterate the importance of training as part of ERP implementation, noting that "future employees will have to have a significantly greater skill set than previous employees," and also noting the frequently-mentioned need to be willing to modify business processes in order to achieve a better fit to the technology in use. While this is not an implementation issue *per se,* the need to adapt to changed business processes as well as to new technology can complicate the human aspect of the implementation process.

4. IMPLEMENTATION OF DSS/BI

Implementation of DSS/BI[1] systems has been studied less than that of TPS, with (Kivijarvi & Zmud 1993) being one example with a Finnish focus. The state of research was summarized in (Mora et al. 2002). Little has been reported that provides much practical guidance or contrasts DSS/BI issues with those that arise for TPS.

Fortunately, specific DSS/BI implementation cases have been reported. We can draw meaningful conclusions from them. Summarizing a conference at which executives reflected on their BI implementation experience, Havenstein (2006) reports that the two key implementation issues in this sphere are data quality and executive buy-in. These are less critical with TPS: data quality, while not unimportant, is more of a technical and clerical issue than a managerial one. Executive sponsorship may improve the acceptance of business process changes that a TPS requires, if it requires any, but those who are to use a TPS will ultimately do so because their jobs require it—not because they are inspired by the CEO's vision. Their attitude toward using it may be important as regards morale and motivation, but the usage itself is not optional if they want to keep their jobs.

An example of a DSS implementation that raises some of the same personnel issues that typically arise in the TPS context is provided by (Botha & Atkins 2006). In that study the target users were New Zealand farmers, not corporate managers and knowledge workers. One reason for the system's poor adoption was overestimating user capability vis-à-vis system complexity. This example is, in some ways, the exception that proves the general rule. Managers and knowledge workers in an office environment will usually be more computer-literate than this user community.

5. CONCLUSIONS

There are significant differences between the implementation of TPS and of DSS/BI systems. It is important for those who manage such implementations, and for those who will teach future managers their trade, to be aware of them.

These conclusions involve *focus*. It's not that low-level computer training (to use the first bullet below as an example) is always totally unnecessary with DSS. Sometimes it is, perhaps for an unusual user community (e.g., the New Zealand farmers of (Botha & Atkins 2006)) or perhaps for a small number of users within a group that is more typical overall. However, organizations have finite resources. They must allocate those resources where they will do the most good. Our purpose here is to point out (to continue with that example) that low-level training tends to be an important focus area with TPS, but not with DSS/BI.

5.1. With TPS:

- **Low-level computer training** is mandatory. TPS users may not have used computers before, though this is changing as teenagers flock to the Web, e-mail, online chat and personal networking sites. Even if they have used computers, they often do not have the professional sophistication to adapt easily to a new information system.
- **Extensive user testing** is essential, as the system carries the operational database which drives all business activities. Errors may not be detected until they have caused havoc down the line. (Testing by developers can never be complete since they only test for situations they can foresee, and which the software was therefore designed to handle. Users create situations that developers did not foresee.) DSS/BI output is reviewed by humans before it is acted upon, providing a level of protection against erroneous outputs.
- **Change management**. Users of an existing TPS often see their value to the organization as tied to their expertise with that system. They know no more about the new system than a new hire would, perhaps less. First- and second-level user management must be careful to show them that the new system will be to their advantage, or serious negative consequences may result. This responsibility cannot be passed off to technical support staff or "bucked up the ladder."
- **Cut-over strategy**. Since it is impractical to use two TPS in parallel[2], and is likewise impractical to give users a choice as to which they will use, it is necessary to move an organization systematically from the old to the new. Conversion approaches are beyond the scope of this paper but can be found in (Palvia et al. 1991), among other places.
- **Executive sponsorship** is important, as noted above, to the extent that the new TPS imposes cultural changes (including major business process changes) on the organization. This was common when organizations moved to their first ERP system but is becoming less of an issue as more and more companies have had one for a while. New TPS may involve usage changes, such as going from "dumb terminals" to PCs or to a Web-based user interface, but increasingly leave the processes behind the interfaces alone.

5.2. With DSS/BI:

- **Data quality** is more of a concern. DSS/BI systems often use data warehouses whose inputs come from multiple systems, some of them perhaps incompatible with others. Using data developed with inconsistent assumptions, based on different time periods, structured in different ways, etc., can lead to problems even if each individual system is internally consistent and (in its own context) fully correct. This raises broad data governance issues, often for the first time.
- Training is needed, but at a higher level. Today's knowledge workers are comfortable with technology in general. They need to be brought to a comparable comfort level with the technology of DSS/BI. As a Holland America Line BI project manager was quoted in (Songini 2003), they need to "spend the time to get to know the data." This is *information* literacy, not computer literacy. "Getting to know the data" would be a waste of time for most TPS users.
- Having **an executive "champion"** is vital. Whereas an airline check-in agent cannot issue a boarding pass without the appropriate computer system, a buyer can in principle decide how many blue shirts to order by "gut feel." Such users will adopt a DSS/BI system more readily if the executive suite uses it, or at least visibly backs its use.
- For similar reasons, it is more important to **adapt a DSS/BI system to its users** than the other way around. Xu & Quaddus (2005) point this out, suggesting that a pilot project permits this without impacting the majority of users. While this may result in a roll-out procedure that resembles pilot conversion of TPS, the motivation for using the pilot approach is totally different.

6. RECOMMENDATIONS

6.1. For IS Faculty

Information systems textbooks, in discussing implementation, do not differentiate by type of system[3]. To them, implementation is implementation. However, as we have seen, this is not the case. It falls to IS faculty to make up for deficiencies of the text in use, including this one. If a faculty member thinks implementation is worth more than the briefest notice, he or she must point these differences out. (I personally think it's worth more than a brief notice, since most introductory MIS students are not MIS majors, and implementation is an area where users play a big part.)

6.2. For IS Textbook Authors

Many IS textbooks spend a great deal of time discussing alternative system development methods: SDLC, various flavors of prototyping, modern variations such as agile development, RAD and extreme programming, etc. The descriptions are followed (in the better books) by discussions of where each fits best. That's good as far as it goes, but it doesn't go far enough. Implementation methods should be treated the same way: by discussing alternatives and when to use each.

6.3. For IS Practitioners

Most experienced practitioners intuitively have a good idea of what to do in this area, as the conference report of (Havenstein 2006) shows. One danger is that, in

reading the "one size fits all" recommendations of textbooks and perhaps hearing them repeated in an MBA classroom by an instructor with little real-world experience, experienced practitioners can become convinced that their intuition is at fault when it isn't.

However, we can *and must* say more than "get 20 years' experience, then trust your gut." Professional papers are published, in part, so that practitioners will learn from them. They must recognize, in reading a paper about system implementation, that its conclusions are probably limited to the type of system being studied. They generalize in finite ways, if at all, to others. Their limitations may be even narrower than system type, as in (Lai 2006) where the author cautions about generalizing his (already ERP-focused) results beyond China, or (Kivijarvi & Zmud 1993) where the limitation of the sample to Finland may limit the validity of extrapolation beyond its borders.

REFERENCES

Botha, N. & K. Atkins. (2006). The design, utility and adoption of decision support systems in the New Zealand pastoral industry. *Proceedings*, APEN International Conference (March).

Cooper, R. & R. Zmud. (1990). Information technology implementation research: a technological diffusion approach. *Management Science,* 36, 123-139.

Havenstein, H. (2006, October 2). Data governance, exec buy-in are keys to BI adoption. *Computerworld,* 40, 40, 8.

Kivijarvi, H. & R. Zmud. (1993). DSS implementation activities, problem domain characteristics and DSS success. *European Journal of Information Systems Research,* 2, 3, 159–168.

Kwon, T., & R. Zmud. (1987). Unifying the fragmented models of information system implementation. In *Critical Issues in Information Systems Research* (R. Boland & R. Hirschheim, Eds.). Hoboken, N.J.: Wiley.

Lai, I. (2006). The critical success factors across ERP implementation models: an empirical study in China. *International Journal of Enterprise Information Systems,* 2, 3, 24–42.

Laudon, K. & J. Laudon. (2004). *Management information systems* (9e: Managing the digital firm). Upper Saddle River, N.J.: Prentice-Hall.

Mallach, E. (2006). System conversion: teaching versus reality. *International Journal of Information and Communication Technology Education,* 2, 2, 17–26.

Mandal, P., & A. Gunasekaran. (2002). Application of SAP R/3 in on-line inventory control. *International Journal of Production Economics,* 75, 47-55.

Mora, M., F. Cervantes-Pérez, O. Gelman-Muravchik, G. Forgionne, M. Mejía-Olvera & A. Weitzenfeld-Reitel. (2002), DMSS implementation research: a conceptual analysis of the contributions and limitations of the factor-based and stage-based streams. In *Decision Making Support Systems: Achievement, Trends and Challenges* (M. Mora, G. Forgionne & J. Gupta, Eds.). Hershey, Pa.: Idea Group Publishing.

Muscatello, J., & D. Parente. (2006). Enterprise resource planning (ERP): A postimplementation cross-case analysis. *Information Resources Management Journal,* 19, 3, 61–80.

Palvia, S., E. Mallach & P. Palvia. (1991). Strategies for converting from one IT environment to another. *Journal of Systems Management* 10, 23.

Rajagopal, P. (2002) An innovation-diffusion view of implementation of enterprise resource planning systems and development of a research model. *Information and Management,* 40, 87–114.

Songini, M. (2003). Cruise line changes BI tack. *Computerworld* 37, 40, 32

Xu, J. & M. Quaddus. (2005). A six-stage model for the effective diffusion of knowledge management systems. *Journal of Management Development* 24, 4, 362–373.

ENDNOTES

1. The term *DSS* is older and more common in academic circles. The newer *BI* (often credited to Gartner's Howard Dresner in 1989) is more often found in industry. While one can argue that DSS is a broader concept that subsumes BI as well as other types of IS, that distinction is not important here. They share the characteristics—user community, nature of input data, use of output, etc.—that affect implementation.
2. Parallel conversion was an industry staple 30+ years ago. Lamentably, it persists in most MIS texts. It is impractical when both old and new systems are online and fundamentally flawed when input timing can affect output. This is discussed further in (Mallach 2006).
3. The author has examined over 25 MIS and systems analysis texts from major publishers over the past few years. Twenty are listed in (Mallach 2006). Others appeared since that paper was written. This statement applies to all, without exception. Listing them here would serve no purpose other than increasing the length of this paper's reference list.