December 2003

Business Process Modelling Using System Dynamics: reflections on Some Field Applications

Gerard McGrath
Victoria University

Follow this and additional works at: http://aisel.aisnet.org/pacis2003

Recommended Citation
http://aisel.aisnet.org/pacis2003/50

This material is brought to you by the Pacific Asia Conference on Information Systems (PACIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in PACIS 2003 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
Business Process Modelling Using System Dynamics: Reflections on Some Field Applications

G. Michael McGrath

School of Information Systems Victoria University
PO Box 14428, Melbourne City, Australia, MC8001
michael.mcgrath@vu.edu.au

Abstract

Experiences with a number of field applications of system dynamics (SD) are detailed. While many of the benefits claimed by proponents of the paradigm were experienced, recommendations flowing from modelling exercises were not implemented in many cases. Moreover, simplistic prescriptions found in the SD literature (such as ensure key stakeholders are involved) were found to be impractical in various real-world situations. Our studies lead us to believe that a considerably more sophisticated approach to change management is required if organizations are to gain maximum benefit from their SD endeavours.

Keywords

Business process modelling, system dynamics, change management.

1. Introduction

System dynamics (SD) has been used to very good effect in many domains over recent years and, in particular, its application in software engineering process modelling has attracted a great deal of attention (see, especially, Abdel-Hamid and Madnick, 1991). One particularly attractive feature of SD is its capacity to represent some aspects of the critical “people-related”, organization behaviour issues so crucial to the successful implementation of any new business process or system. That is, most tools, in general, address the who, whom, what, where and when of process modelling, but not the how or the why. Specifically, they enable the recording of the details of the individuals and teams enacting a process, the artefacts produced and the sequencing of the functional tasks, but not the norms, beliefs, motives etc. of the staff carrying out the process: i.e. hard factors but not soft factors.

Partly because of the above reasons, we have employed SD extensively in a number of business process modelling field studies over the past 5-6 years. In general, we have found our experiences with SD modelling (and a number of increasingly excellent software products based on the approach) to be rewarding. For the most part, however, any difficulties experienced in model development pale in comparison to problems we have experienced in having our recommendations implemented. This is our focus here, where we report on our experiences with four field studies; the first of which commenced in 1997 through to the fourth study, which is yet to be completed.
We employ a well-known change management framework (Leonard-Barton, 1988) to compare and contrast implementation characteristics of our studies. This analysis is presented in Section 3, following a brief presentation of each of our studies in Section 2. Concluding comments are presented in Section 4. A basic knowledge of SD and its modelling conventions is assumed. The reader requiring an introduction is referred to (Maani and Cavana, 2000).

2. Case Studies

The case studies described in this paper are taken from a collaborative research project that was undertaken with a large Australian information services provision company (which we shall refer to as Gigante). In all, the research was carried out over a 6-year period and began with a study into problems the company was experiencing at its customer interface. During this (rather lengthy) study, events occurred that gave rise to additional research opportunities and, while the scope of each of these overlapped the original investigation to a greater or lesser extent, each was treated as a separate study within the overall project. SD techniques were employed in all our work and we briefly outline four of our case studies in the following sub-sections.

2.1 Case Study No. 1: Customer Service Interface Problems

As noted, the catalyst for our series of case studies were problems that Gigante was experiencing in its customer interface operations. These problems were causing significant revenue losses, and previous investigations had indicated that the root causes were the high workload, plus a lack of training, combined with a high rate of new product introduction. In its attempts to remedy the situation, the company had poured much effort, resources and dollars into the development of a (flowchart-based) business process model. A simplified version of the relevant portion of that model is presented in Figure 1.
Figure 1. Gigante’s information services provision function.

Customers call a Customer Service Centre (CSC) and may request one or more information services. One Customer Service Representative (CSR) takes a customer’s order, comprised of one or more products (identified by product codes). The CSR then enters the order into the provisioning system. Because there are several hundred products, many errors occur at this point. Product codes entered into the provisioning system are translated into: 1) service codes and sent to the billing system; and 2) network reconfiguration data and sent to the engineers responsible for maintaining Gigante’s own internal network. In many cases where an incorrect product code is entered, the network will, nevertheless, be configured correctly. From this time, the customer has access to the requested services and may make calls. However, because of the incorrect service code in the billing system, call charge records (CCRs) will be rejected and a bill for the revenue involved cannot be issued until the original provisioning error is corrected.

CSRs must correct their own errors but customer calls take priority. As the rejects database grows, the workload increases. This places more pressure on CSRs which, in turn, results in more errors and a vicious cycle is thus established. More errors, of course, also mean more customer enquiries and complaints, leading to a further workload increase. To cap it all off, Gigante releases new products on a regular basis. This places a further strain on CSRs - new products in themselves mean more errors but, with no time left for training (because of the workload), the situation is exacerbated.

The development of an SD-based process model as a key component of our study was a worthwhile exercise and led to many of the benefits claimed by proponents of the modelling paradigm (see, e.g., Senge, 1990; and Vennix, 1996). In particular, the model (and, more specifically, the modelling process itself) greatly assisted us in familiarising ourselves with the study domain and seems to have ensured that all participants in the modelling process developed a common understanding of the key concepts. In turn, this deeper understanding of the inter-relatedness of the component parts of the total system resulted in a greater awareness of the flow-on effects of local decisions and actions to other areas of the organization. However, we experienced a number of difficulties with the approach.
Specifically, SD models do not readily decompose – they spread. This means that adding detail adds to the complexity of the model and inhibits validation and understanding. There is also a tendency for inexperienced modellers to “model the system” rather than the problem (Anderson, Richardson and Vennix, 1997). This, again, adversely affects validation and understanding, both of which are based on the model being useful for the purpose for which it was intended and require explicit identification of the problem boundary (Forrester 1961; Forrester and Senge, 1980; Legasto and Macariello, 1980; Bell and Senge, 1980; Barlas, 1996; Coyle and Exelby, 2000). SD models are also most effective when variables are aggregated to a very high level and analysis is restricted to determining/testing policies rather than operational questions (Forrester, 1961). They are not suitable for capturing fine-grained detail. Our major problem with this exercise, however, was lack of implementation of recommendations. This problem is not unique to this particular modelling exercise, nor to the SD community (Campbell and McGrath, 1999). It is this issue that is the major concern of this paper.

2.2 Case Study No. 2: A Pricing Policy Decision

Gigante’s information services are provided via electronic voice and other telecommunications facilities. The organization’s policy has always been that, for connection times less than 5 seconds, the customer would not be charged. Gigante’s Accounting Department had analysed usage data and discovered that over 40% of calls fell into this category. Consequently, they proposed that the charging threshold be reduced to 2 seconds. Our client was concerned that this could lead to a substantial increase in billing enquiries and, thereby, lead to even more CSR errors. Making a number of simplifying assumptions, we were able to rapidly (within 2 days) develop an *ithink* model that demonstrated that the policy change could, indeed, lead to significant problems (which we were able to quantify as lost and delayed revenue estimates). This sufficiently concerned the organization that they decided not to proceed with the proposed pricing policy change.

![Causal loop model of reduced threshold impacts.](image)

*Figure 2. Causal loop model of reduced threshold impacts.*
Key factors and relationships in the subsystem investigated are illustrated in the causal loop diagram presented in Figure 2. Essentially, a decrease in the charging threshold does indeed produce extra revenue but, at the same time, it also generates extra billing enquiries. This, in turn, creates extra work for CSRs and a consequent staff shortfall, meaning that extra staff resources are required (thus adding to costs). The staff shortfall also means that existing staff are put under more pressure, resulting in an increase in errors. Based on the situation in CSCs outlined earlier, some revenue involved in these errors will almost certainly be lost (or, at least, its collection will be substantially delayed). The staff shortfall also increases customer wait-time leading, eventually, to customer defections and more lost revenue. Note also that customer dissatisfaction over Gigante's inability to quickly resolve errors is another source of defections and that errors, in themselves, generate further enquiries (thus producing the pressure on CSR staff even further). Hence, the diagram illustrates that, while the proposed threshold reduction would clearly have produced extra revenue, there were several potential sources of reduced profit - specifically: increased CSR costs; additional unresolved errors (or delays in resolving errors); customer defections due to increased wait-time; and customer defections due to dissatisfaction with error resolution.

Our causal loop model was translated into an ithink stock-flow model and simulations were run. These clearly demonstrated that profitability is extremely sensitive to the level of initial increase in enquiries: e.g. it was projected that a 10% initial increase would result in a net loss of $27.0M over 10 months. On the basis of this modelling and simulation exercise, the research team recommended that the proposed threshold decrease should not proceed until more was known. This recommendation was accepted by the company – in fact, while the report was still in draft form! This is in stark contrast to the previous case study, where much more effort was spent in modelling (and, from a technical viewpoint, a much superior model was produced) but very few of the research team’s recommendations were implemented.

2.3 Case Study No. 3: An Outsourcing Initiative

One of Gigante’s most successful areas was its International Division. Rapid response to new circumstances was the key to their success. In particular, communications lines were flexible and uncluttered and scant regard was paid to formal policies and procedures. Nowhere was this style of working more evident than in International’s IT operations. Rapid changes to production lines, the customer base and specific customer demands meant that billing, orders and customer management systems had to be updated frequently and quickly. Typically, a salesman working offshore would place an urgent call to the IT Manager and: 1) request information required to finalize a quote; and 2) advise of system changes required to support his prospective customer’s particular product demands. Unless the quote information was provided overnight and the necessary system changes made within a month (at the outside), it was highly likely that the deal would fall through. These sales support requests were given top priority by the IT Department, activities were carried out “on-the-fly” and, despite (or because of) its small size (12 people), the Department had compiled an excellent and admirable record in meeting its deadlines. As such, it was considered to be a major source of competitive advantage for International and was generally held in high regard by colleagues.

This idyllic state of affairs came to an abrupt end in 1997 when Gigante entered into an outsourcing deal with Worldwide Information Technology (WIT). A major player in IT outsourcing, WIT had a hierarchical organization structure, a predominately bureaucratic mode of working and a culture similar in many respects to that of Gigante. In this environment, the very features that were the essence of International’s strengths and
successes now were jeopardized. In particular, to have any systems maintenance or
elevation work undertaken, work orders had to be prepared, and estimates and program
specifications had to be developed. All these were then passed upwards through three layers
of interface management on the Gigante side, then downwards through a similar number of
layers on the WIT side and, finally, each work order had to be vetted by WIT’s Legal
Department.

Unconsulted prior to the establishment of the outsourcing arrangement or during the
preparation of detailed operating procedures (which were far from complete in a number of
important respects), International found itself in a very difficult position. With the procedures
as they stood, there was no way that its sales force or other operations management and staff
could continue to receive the level of IT service they had become accustomed to and required:
its staff lacked both the skills and the will to prepare the necessary documentation and to
negotiate their way through the system; unacceptable delays were intrinsic in the procedures
themselves; and the final blow came when their champion (the General Manager) was
promoted to an offshore position.

In arguing their case, International’s IT Department felt that they needed some quantitative
support. Intuitively, they were certain that outsourcing would have a major negative impact
on their bottom line but needed some means of demonstrating this. We were consulted and it
was agreed that, using SD, an ithink model would be developed and that we would employ its
powerful simulation capabilities to demonstrate the impact that introducing delays into their
processes would have.

Having developed the model, we were then able to run various forms of sensitivity analysis.
For example, Figure 3 illustrates the impact of variations in initial delays relative to
competitors on market share (with graphs 1-5 corresponding to 4 weeks faster, 2 weeks faster,
no difference, 2 weeks slower and 4 weeks slower respectively). A number of other graphs
were produced as a result of our analyses and, collectively, these dramatically demonstrate
the impact of both types of delay on International's market share and revenue: i.e. the
modelling exercise produced precisely the type of "ammunition" the IT Department was
seeking.
However, the IT Department's attempts to utilise these results met with little success. In retrospect, there was always a fair chance this would eventuate, given that key decision-makers were not included in the model building process (Vennix, 1996). Vennix's advice is sound and, in an ideal world, key stakeholders would always be involved in the development of important decision support models. The reality, though, is that, in many modelling situations, this is simply not possible. For example, in the case under review here, the size of the organization, stakeholders' other responsibilities, stakeholders' geographical dispersion and severe internal and external pressures on Gigante at the time of the study, all mitigated against our attempts to get "buy in" and active participation from the more influential decision makers. Moreover, even if we had managed to realize our desired levels of participation, we doubt it would have made a great deal of difference to the eventual outcome: i.e. other factors - notably power/political considerations were always going to make life very difficult for both International and, particularly, its IT Department. We turn our attention to these factors in the following section.

2.4 Case Study No. 4: A Model of Software Procurement

In this study, we ‘went over to the opposition’ – i.e. we worked with a team within Gigante’s outsourcing firm, WIT, in an effort to assist them to obtain a better understanding of their interface with their client and to abstract our findings into a more general model of the software procurement process (a very much under-researched area).

Our study was concerned mainly with the IS operations of Gigante’s International Operations Division (Glnt). Six software firms were contenders for a multi-million dollar ERP contract to be let by Gigante with one bidder, WIT, involving both its Sydney and Melbourne branches (to some extent, as competitors). The study covered a period from June 1996 through to end-2000. As noted, in June 1996 Gigante outsourced all its internal IS development and operations work to WIT. Glnt's systems had all been converted to run on a specific ERP product base during the early 1990s and these (100+) systems, together with the
40 ERP specialists responsible for their development and maintenance, were transposed to WIT as part of the outsourcing deal.

Despite the (supposedly) exclusive nature of this deal however, GInt (in 1999) went out to tender for a complete upgrade of its ERP systems. Eventually, WIT were successful but felt that they did not understand the software procurement process sufficiently well. As a result, we were asked to generalise from this case study to a model that might be employed for decision support and training purposes.

![Figure 4: An episodic/encounter model example.](image)

In a particularly interesting piece of work focussed on COTS, Heiskanen, Newman and Simila (2000) present a ‘social model of software development’ based on user-developer interaction. This interaction is defined as relatively stable “state progress passages” over periods of time (episodes) punctuated by “critical events” (encounters). Encounters have the potential to radically change various attributes of user-developer interaction; such as developer access to users, influence and user satisfaction with developer performance.

Longitudinal representations of interactions are portrayed in diagrams similar to Figure 4. In this example, the user organization is initially equivocal about its new system. Then, at encounter No.1, something happens (e.g. a major disagreement over project direction) that causes users to reject the system. Later, at encounter No.2, rejection changes to acceptance (because e.g. the initial version is released and users are very happy with it). At encounter No.3, the user organization again rejects the system (due e.g. to a sharp increase in leasing and maintenance fees).
These diagrams are very useful but key features of the episodic/encounter model seem to be eminently well-suited to an SD representation. In particular, the ability of SD modelling tools capture the complex feedback loops intrinsic to software processes and to automatically simulate system behaviour over time may be used to good effect. In addition, more subtle aspects of the vendor-client relationship can be neatly modelled, simulated and illustrated. Here, there is already a substantial and rapidly-expanding body of work dealing with the use of SD for (in-house) software engineering process modelling; the seminal work in this field being that of Abdel-Hamid and Madnick (1991).

Figure 5: A model of software procurement.

The initial version of our software procurement model is presented in Figure 5. The model is underpinned by two theories from the management science literature; expectancy theory and the ‘garbage can’ model of organizational decision making. Expectancy theory (DuBrin, 1995) is actually a group of motivation theories founded on a rational-economic view of people but all versions are based on: i) the expectation that an act will result in a given outcome; and ii) the attractiveness of that outcome to the party. Effort is linked to performance and performance, in turn, is linked to rewards. If a breakdown occurs at any point in the chain then, ultimately, less effort will be put in (i.e. motivation will suffer). These linkages and loops are clearly evident in our model presented in Figure 5. Specifically: i) a vendor puts effort into managing a client relationship (which costs money); ii) the effort expended is a major determinant of the level of access (LOA) that the vendor has to the client
(performance); and iii) the level of access, in turn, will have an impact on the amount of work the vendor secures from the client (rewards). If, however, the vendor puts a lot of effort into the relationship and this does not bear fruit (in both improved access to the client and more work), then he or she is liable to be demotivated. Finally, in our base model, the user (playing the role of the vendor) has the opportunity to vary the effort put into maintaining the relationship at different points during a simulation run. The temptation to do this may be considerable because relationship management is expensive and consequences (reduced level of access and less work) may not become evident for some time.

The parameters of our base model have been established such that vendor effort will, indeed, deteriorate gradually over time – behaviour quite consistent with the poor customer relationship management characteristic of much COTS implementation and maintenance (Heiskanen et al., 2000). Obviously, if this continues for long enough without major intervention (or encounters, using the terminology of Heiskanen et al., 2000), then the vendor is liable to be at a major disadvantage if (for example) the opportunity to bid for a major new contract arises. As noted, our client’s intention was to employ our model as a training tool and decision making aid. To date though, this has not occurred to any great extent.

3. Comparison and Discussion

While there is a growing body of literature in the change management field, it is deficient in a number of ways including: a failure to differentiate between initiating change and successfully implementing such change (Watkins, Ellinger and Valentine, 1999); a real understanding of how change is well introduced and implemented (Buchanan and Badham, 1999); and neither has there been adequate addressing of the variable dimensions of depth, size, pervasiveness of change and the diverse perspectives from which such change is viewed and managed (Bolman and Deal, 1997; Wing, 2001). What is reflected in our case studies and gradually being emphasised in the literature is the need to address innovation and paradigm shifts (Chakravarthy, 1997) and to actually track as we have endeavoured to do, actual longitudinal, organizational case studies.

Given such confusion in the research and literature the well-tried Leonard-Barton (1988) change model is an appropriate change management tool; not only because of its clarity, but also because of its focus on change management in high-technology organizations. While her framework is by no means complete, over recent years we have found it to be an excellent starting point for assessing the probability of success of proposed organizational innovations. Here, we employ the framework to retrospectively contrast and compare the four SD case studies detailed in the previous section.

The basis of Leonard-Barton’s model is that: i) any technologically-based change initiative has implementation characteristics; and ii) that these characteristics set conditions within which change managers must operate and, in turn, dictate implementation tactics. The principal characteristics are:

- **Transferability**, which includes:
  - **Preparedness**, the extent to which the technology has been proven; and

---

1 This behaviour is built into the feedback loops from Work and LOA to Effort.
Communicability, the extent to which the technology’s features can be communicated to end-users and decision makers.

- Implementation complexity, which includes:
  - Organizational span, the number of people affected by the innovation; and
  - Organizational scope, the number of organization units affected.

- Divisibility, which includes the degree to which the innovation implementation can be partitioned according to:
  - Modularization (i.e. partition the innovation itself); and
  - Individualization, meaning implement the total innovation in one part of the organization at a time.

Table 1 compares and contrasts the four case studies according to the Leonard-Barton framework. While the retrospective analysis of our case studies suggests that the predictive capability of the framework is reasonably sound. It is our view, however, that its real value lies in its capacity to highlight areas of the implementation domain that require special attention in change management planning. Here, Case Study 3 provides an excellent illustrative example and we now revisit this study.

<table>
<thead>
<tr>
<th></th>
<th>Case Studies</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Transferability:</td>
<td></td>
</tr>
<tr>
<td>Preparedness</td>
<td>Low</td>
</tr>
<tr>
<td>Communicability</td>
<td>Low</td>
</tr>
<tr>
<td>Complexity:</td>
<td></td>
</tr>
<tr>
<td>Org. span</td>
<td>High</td>
</tr>
<tr>
<td>Org. scope</td>
<td>High</td>
</tr>
<tr>
<td>Divisibility:</td>
<td></td>
</tr>
<tr>
<td>Modularization</td>
<td>Medium</td>
</tr>
<tr>
<td>Individualization</td>
<td>Low</td>
</tr>
<tr>
<td>Chances for success</td>
<td>Low</td>
</tr>
</tbody>
</table>

Table 1: Case Study prospects for success according to the Leonard-Barton framework.

Gigante's International Division (and, particularly, its IT Department) could clearly see the devastation the outsourcing decision would wreak on their business. In their attempts to alleviate the impact of this decision, they employed rational arguments. However, much decision making in organizations is not rational. In this instance, International's concerns were dwarfed by a much bigger 'game' and, here, their lack of any real political clout counted very much against them. In the causal-loop diagram presented in Figure 6, we have attempted to represent some of the major influences we detected as part of this wider game.

Organizations enter into IT outsourcing agreements for many reasons. During this study, we heard considerable conjecture as to the real reasons behind the Gigante/WIT deal, but press
reports at the time consistently nominated major cost savings as Gigante's principal motive for entering into the contract (one of the largest IT contracts ever signed by an Australian company).

A special Program Management team (abbreviated as pm in Figure 6) was formed to manage Gigante's side of the alliance. From Figure 6, it can be seen that their performance had a major effect on the success of the outsourcing venture. Furthermore, since most of Gigante's products require substantial IT support, outsourcing operations had a significant impact on product performance and this, in turn, directly influenced Gigante's bottom line. In addition, the direct link from outsourcing perf to Gigante profit indicates that outsourcing, in itself, was expected to contribute to profit through greatly decreased IT costs. Thus, this portion of the diagram effectively mirrors Gigante's official outsourcing policy and views.

![Figure 6. Impact of allowing outsourcing exceptions.](image)

However, other important factors were also at work. In particular, Gigante's executive, the Project Management team and the outsourcing agent (WIT) all stood to gain (and lose) much from the outsourcing venture and links from Project Management team and agent performance to rewards/profit are clearly identified in Figure 6. In this context, it is easy to see that exceptions (exemptions from the outsourcing arrangement) could clearly not be tolerated. That is, apart from reducing the agent's profit, every exception allowed was likely to have a damaging impact on general perceptions of Project Management team performance. Thus, Project Management implemented a policy to the effect that exceptions would not be allowed under any circumstance - no matter what benefits specific cases might have for
individual products (and the systems and personnel that supported these). To complete the picture, outsourcing can be a very risky business (Aubert et al., 2001) and there were many (inside and outside Gigante) who doubted the wisdom of this particular venture. Consequently, perceptions of the performance of Gigante's executive were closely linked to both the outsourcing operations themselves and the Project Management team.

Looked at in a (seemingly) rational light, the decision not to exempt International and their systems from the outsourcing deal seems bizarre - ensuring as it did the eventual destruction of International's leadership in their particular product market, plus the additional loss of a number of committed, scarce and valued IT specialists. If we view the situation from a power/political perspective, though, the events that transpired begin to make sense. In particular, Pfeffer (1981; 97-135) presents a classification scheme for power sources. Using this scheme, we may investigate the impact of a decision to allow outsourcing exceptions on the power sources of the three parties discussed earlier in this section.

Clearly, all parties stood to lose substantially. The Project Management team and the agent would have lost (shared) control over the provision of important resources (International's systems and IT personnel) to the organization at large and, in addition, the agent would have received less funds for its services. Furthermore, without control over these systems and specialist personnel, the two parties' total level of expert knowledge (a vital source of power in organizations) would have suffered. Gigante's executive, however, would have been largely unaffected (in a direct sense) with respect to these power sources.

Perhaps, most interesting of all is that reputations are built upon perceptions Pfeffer (1981; 54-57) and all parties would have suffered here. As noted previously, allowing exceptions would have had a major negative impact on perceptions of Project Team performance. Actual performance also has an effect on perceived pm perf and both these factors are clearly identified in our causal-loop diagram. The link between perceptions of Project team and Gigante's executive performance is also identified. However, in Pfeffer's scheme, there is also a clear link from perceptions (of both power and performance) to prestige and extending our model to specify this additional relationship is a relatively simple exercise. Finally, where an organization unit or group has a strong, united, common view on issues, they derive power from consensus (Pfeffer, 1981; 122-124). Pfeffer emphasises that this is a particularly formidable power source but one that can easily be dissipated by significant change. Allowing exceptions would certainly fit into this category.

4. Conclusion

The experiences described in this paper with four SD modelling exercises would appear to indicate that implementation outcomes are not determined by the amount of effort expended in modelling. Nor are they dependent on the validity, or accuracy, of models. Rather, they appear to be more dependent on the effectiveness of change management approaches adopted.

We believe, Case Study 3 was particularly instructive. Organization change increases the turf warfare of change management - and part of this includes the more macro dimensions of organization prestige and reputation. In the realm of any organization’s corporate communication, the most critical function is that of the organization’s image and identity, important within the organization but perhaps even more vital to the external community and many of its key stakeholders. Diverse stakeholders may have varying images of an organization but reputation and identity should be consistent, a hallmark that distinguishes it
instantly in a globally competitive environment where attracting the right customers, investors and employees is crucial. Image, identity, prestige and reputation are today hallmarks of the qualitative intangibles based in perceptions and hard fought for. They are part of an organization’s intellectual capital in the broad sense and can be irrevocably damaged even by one instance of poor management. Put simply, the organization is the message and poor communication (arising, for example, from the politically-motivated turf warfare referred to above) can destroy it instantly!

References


