December 1998

Investigating the Delivery of Information Technology Services in the Presence of Cultural Conflict

Daniel Braunstein
Oakland University

Steve Elliot
University of New South Wales

Graham Low
University of New South Wales

Richard Webby
University of New South Wales

Follow this and additional works at: http://aisel.aisnet.org/amcis1998

Recommended Citation
http://aisel.aisnet.org/amcis1998/240

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 1998 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
Investigating the Delivery of Information Technology Services in the Presence of Cultural Conflict

Dan Braunstein  
School of Business Administration  
Oakland University

Steve Elliot  
Graham Low  
Richard Webby  
School of Information Systems  
University of New South Wales

Case studies from three organizations demonstrate the potential applications of theory and method for investigating the presence of cultural conflict in delivering information technology [IT] services. Critical incidents highlight the case applications. A complete paper is available, in which a model influenced by Romm et all [7] and Trice and Beyer [8] organizes the observations.

Culture provides some degree of order and continuity in organizational life, and therefore makes it manageable. Components of culture include systems of beliefs, values and norms, the ideologies, plus artifacts, the cultural forms. Research into organizational culture typically investigates members’ responses to the ideologies using interviews and surveys, as well as careful observations of the cultural forms. Gordon [2] has identified three classes of industry variables that can help us identify potential cultural conflicts: competitive environment, customer requirements, and societal expectations. Each of these characterizes a significant group of ideological issues which can affect IT services, the cultural forms. If the IT vendor experiences internal conflict in ideology, services can be negatively impacted. The following is a list of potential conflict issues:

- systems development approaches, leading to choices between object-oriented and structured development methodologies;
- available implementation technologies, including alternative policies and practices concerning the availability of re-usable libraries of routines;
- senior management policies regarding the use of productivity tools, e.g. CASE viewed as providing a strategic benefit; a source of added value; or merely a means of cost displacement and/or cost avoidance.
- quality management policies, arguments defining quality standards as the basis for review procedures, affecting the monitoring of software and system development process accordingly.
- prototyping practices: how early shall system component prototypes be delivered, for the purpose obtaining immediate reactions of clients for feedback to system development personnel?

List developed from a critique by Yourdan [9].

A Case of a New Joint Venture With Emerging Cultural Conflict

Consider the potential for cultural conflict when two leading corporations, each from a different national culture, both with a substantial business interest in information technology, announced the formation of a joint venture that would take control of both companies’ IT services throughout a major part of the world. The new entity began functioning in a variety of IT businesses, including facilities management, systems integration, business recovery, application software, networking, information kiosk services. In order to be successful in these diverse businesses, the new corporation had to build effective applications development teams. Resolution of cultural conflict would be central to this process [1].

The new joint venture began life with a mixture of staff from its parents: two distinctly different corporate cultures from two countries. Not only were there substantial staff hired from the home countries of the two parent corporations, there was also hiring from the various countries in which the new subsidiary began doing business. Two "critical" incidents suggest potential conflicts in ideology. The first concerns computer hardware purchasing policy. The two corporate parents had very different approaches. Parent A had an ideology of central control which required all equipment to budgeted for in advance, and insisted on higher level approvals to change the budget. Parent B had a much more flexible policy, in which the development team had authority to make ad hoc purchases as long as they would not affect projected profitability. A second incident involved contrasting policies with regard to training. This conflict resulted from different values which two corporate cultures place upon staff training experiences. Staff from parent B had almost unlimited access to training, while staff from parent A, where centralized control was a fundamental value, had very restricted access. Resentment developed among parent B staff when tighter controls were instituted. Perhaps a history of perceived opportunistic threat on the part of the parent corporations [5] may have endangered the outcome of the joint venture.
A Case of a New Corporate Venture and a Major Client

Two of the corporate officers of a software development company known for its relational data base products left that firm to form a new venture. The company would specialize in the development of interactive software, which according to their business plan, could be produced in a much shorter cycle time than is typically required for most applications. The creative engine driving the company’s products was a new version of a powerful development language, minimizing the need to program in basic code. In its short time in business, the new company was successful enough to obtain a prominent financial services client. When the firm began work on a complex software system which would track brokerage fees and commissions, and serve as the actual clearinghouse for trades, difficulties emerged.

Communications between the IT firm and its client were not effective, thus jeopardizing software development. Furthermore, some software engineers with significant project responsibilities had poor managerial skills. Planning and time management were very casual: “what should our development team be doing next week?” A project leader might have little understanding of the information requirements of the financial services business, and moreover did not value the importance of spending time gaining detailed knowledge of the client’s needs. On many occasions, software engineering personnel argued about the virtues of various forms of the development language, but did nothing to resolve these difficulties in terms of financial services business requirements.

While the new venture’s early success was based upon its IT expertise, the culture associated with this expertise may have created conflicts with its client. The financial industry works under considerable time stressors and substantial information overload, risking large sums of money for short durations. It is difficult for outsiders to understand the trading business, especially the format of information requirements. The founding officers of the firm were in a bind: how could they hire specialists who could both develop the technology as well as work with financial services clients to develop applications? They realized that they had become victims of the technical culture which originally was responsible for their success. This technical culture, which offers considerable promise for complex interactive software systems, was also jeopardizing client relationships.

A Case of Conflict Between Vendor Development Teams

A software vendor’s cultural ideology may defeat the intended goal of integrating a complex information system. This case involved a vendor whose software engineering teams were used to considerable autonomy in their development efforts. The vendor sold an avionics software system to a client while agreement was yet to be reached regarding maintenance features of the system components. The object-oriented applications could not be be efficiently maintained by the client due to their complexity. For example, documentation of the required code ran to over 3,000 pages. According to one insider, the system was actually too complex for CASE tools.

Since the system had 17 major components, different development teams were required to work on each component. However, there was no standard format for code specification. When the client brought in a third-party consultant, he immediately looked ahead 20 years and concluded that such a system would be a code maintenance nightmare. Coordination attempts with the vendor’s in-house developers became frustrated because everyone jealously guarded their independent coding standards. Since the vendor’s culture was very much like that of a University organization, each component development team worked relatively independently of the others. At points in the entire system development, when teams were asked to make coding changes which would benefit the system as a whole, members claimed that their intellectual property rights were being violated!

Component reliability testing was not able to identify defects in a timely fashion because team members were afraid to criticize defects found in other team’s components. A design document was written which contained a very elaborate mathematical proof of how an electronic filter should work within the system, but no code was ever developed which specified how the filter would be embedded within the program. The team responsible for filter design did not think it was also necessary for them to also develop the code, but no other team understood how to do it.

Suggestions for Research on Cultural Conflicts in Delivering IT Services

We have observed conflict in three IT vendor organizations which has interfered with IT service delivery. In each case, certain key ideological issues in the software development organization’s culture were left unresolved. Gordon’s classification of industry variables can help us summarize these unresolved conflicts: (1) In the case of the cross-national joint venture, different national competitive environments created the conflicts in managerial control. (2) In the case of the firm whose business plan was based upon the software development language, the challenge of meeting customer requirements opened unresolved ideological conflicts; (3) In the case of the avionics software, different expectations associated with a basic research and development organization’s flexibility of inquiry led to the emergence of conflict when a coordinated effort was required in order to satisfy contractual requirements.

The field research we suggest is conceived as a series of iterative cycles, in which grounded theory [4] would inform both research team and corporate management of the necessity to deal with potential conflicts in provision of IT services. Pinsonneault and Kraemer [6] urge that “more mixing of research methods is needed…. This is particularly important for explanation and exploration studies. Surveys should be used more with case studies and field observations in order to develop a richer, more detailed, and complete understanding…”[p. 97]. By comparing survey results with interview data based on the
critical incidents, checks can be made for analytical rigor of any newly developed models [3]. A table in the complete paper contains a sample of suggested interview questions and survey scales for both cultural ideologies and cultural forms. The concluding section deals with research limitations of current definitions of organizational culture, as well as ethical considerations in data collection.

Acknowledgements
The authors acknowledge the assistance of Marcia Beck and Wayne Cox in the collection of case materials.

References