

Is Emotion Relevant to Requirements Engineering?

Daniel M. Berry, U. of Waterloo, Canada
Isabel Ramos, U. do Minho, Portugal

A Bit of History

Originally, software systems (SWSs), actually programs, were introduced merely to automate existing manual tasks for collecting, processing, or distributing information.

Duplicated Original Processes

A program was used

- to speed up,
- to reduce errors, or
- to do both

for existing clerical tasks without changing the basic processes in which these tasks were carried out.

Duplicating, Cont'd

If the original task produced a report on paper, the automated version produced the same report, so that the automated task could be carried out, albeit faster and with fewer errors, *in the original task's place.*

First E-Type Systems

However, as observed by Meir Lehman [1980], the introduction of these SWSs began to affect the processes in which these automated tasks were embedded.

Different and better processes were enabled by the automated tasks, even to the points of

- making the automated task unnecessary, and
- eliminating some people's jobs!

New Peripherals

New peripheral devices were introduced to allow the computer to sense and control more than just data, e.g., production lines and aircraft.

New Peripherals Allowed

These peripherals allowed the introduction of SWSs to change more and more processes and ...

to fundamentally alter the way things are done in many real-world, man-made systems not at all related to computing.

Example of Fundamental Change

Originally, commercial systems evolved in the presence of only paper and carbon paper.

Joint evolution of:

- the form to contain all information that might ever be required about a transaction, and
- creating multiple copies of a document.

Example, Cont'd

Creating and distributing multiple copies of a form became the way to make sure that all potentially needed information about a transaction were distributed rapidly and in parallel to all involved in the transaction.

At first, with photocopiers and later with printers, the production of the multiple copies was automated.

Example, Cont'd

However the basic workflow, and in particular the distribution of the copies, was unaltered.

Then, new high-resolution screens and networks became available.

Creative SW engineers noticed that it was no longer necessary to print and distribute multiple paper copies of all possible needed information.

Example, Cont'd

Instead, it suffices to put a network of computers, each with a high-resolution screen, on the desks of all persons involved in all transactions.

Each is able to access directly precisely the information he or she needed for his or her part of the transaction at the time he or she needed it.

Making any paper copy of the information is totally unnecessary.

Advantages

On one hand, the operation of the transaction has improved.

The information needed for the transaction is distributed instantaneously to all involved in the transaction.

A person involved in the transaction looks up only the information that he or she needs.

Advantages, Cont'd

If the source of the information on the old form is readily available, there is no need for an electronic version of the form to be filled out in advance with all information that might conceivably be needed in the transaction.

Only the usual information is requested with the assumption that in the rare case additional information is needed, the source can be queried on the spot.

Thus, the transaction is made more efficient.

Disadvantages

On the other hand, there are some negative consequences.

No paper copies are distributed (good!).

Therefore, all who were involved with production, purchasing, distribution, & disposal of paper and printed reports find their jobs reduced and possibly eliminated.

Organizational Transformation

The organization has been transformed.

The introduction of any SWS has the potential to transform any organization into which it is introduced.

OT, Cont'd

According to Dickson & DeSanctis [2000], in general, the introduction of SWSs has become a driver for

- innovative work practices and
- new models of management and organization,

i.e., organizational transformation (OT)

OT, Cont'd

Palmer & Hardy [2000] define organizational transformation (OT) as the process of fundamentally changing an organization's processes in order to allow it to better meet new challenges.

Potentially Rebellious Employees

In the past, each employee ...
got his or her information from ...
a particular place on one copy of all possible
needed information.

Rebellious Employees, Cont'd

Now, that employee ...
needs to work online ...
on a computer with a windowing system with
...
a program that allows querying for the needed
information, ...
but in a language probably derived from
predicate calculus.

Rebellious Employees, Cont'd

One or more of these employees could rebel
against this job change because he or she

- hates computers,
- hates mathematics or anything reeking of it, or
- refuses to learn any newfangled way to do something that he or she has done for years in a perfectly good simple way!

“If it ain't broke, don't fix it!” says the employee.

Emotion Rears Its Ugly Head

Clearly, emotional issues are coming into play in the use of SWSs and the OT they breed.

Goals of OT

Generally, innovative, organization transforming SWSs are introduced with the laudable goals of

- improving organizational efficiency and effectiveness,
- reducing costs,
- improving individual and group performance,
- and even enabling individuals to work to their potentials,

However

However, it is sometimes very difficult to get these SWSs to be used successfully and effectively.

Resistance

As was observed by Lyytinen, Mathiassen, and Ropponen [1998], Iivari, Hirschheim, and Klein [1998], and Bergman, King, and Lyytinen [2002]:

Some people in some organizations resist the changes.

They resist using the systems, misuse them, or reject them.

Results

As a result,

- the goals are not achieved,
- intended changes are poorly implemented, and
- development budgets and schedules are not respected.

The Role of EVBs

Misplaced, as opposed to normal, emotions, values, and beliefs (EVBs) are often offered as the causes of these problems:

Bergman, King, and Lyytinen observe, [2002, p. 168] “Indeed, policymakers will tend to see all problems as political, while engineers will tend to see the same problems as technical. Those on the policy side cannot see the technical implications of unresolved political issues, and ...

those on the technical side are unaware that the political ecology is creating serious problems that will show up in the functional ecology.” They go on to say, (p. 169) “We believe that one source of opposition to explicit engagement of the political side of RE is the sense that politics is somehow in opposition to rationality. This is a misconception of the nature and role of politics.

Political action embodies a vital form of rationality that is required to reach socially important decisions in conditions of incomplete information about the relationship between actions and outcomes.”

Other Cases

We think it's fair to say that the emotions of the stakeholders played a big role in the failed deployments of the London Ambulance System (LAS) and of CAPSA, Cambridge University's new on-line accounting system.

LAS

It appears to us, for example, after reading the Finkelstein *et al* reports on the LAS deployment, that sabotage by stakeholders, e.g., ambulance personnel, who had been left out of the decisions regarding the new system contributed heavily to the failed deployment.

Exacerbating Effects

Exacerbating these potential emotional problems is the fact that environments into which the SWSs are introduced are incredibly complex.

Thus, it is next to impossible to predict the effect that the introduction of a new SWS will have on the organization and its users.

RE During this History

Now, let us consider the changing nature of requirements engineering (RE) for SWSs.

Initially, RE needed to consider only the input-output behavior, i.e., the functionality, of programs.

RE History, Cont'd

Certainly, NF issues, especially performance, could be important too.

If an automated task were to take too much time, space, or both, the task would not be effectively automated.

RE History, Cont'd

As SWSs began to change the real-world systems in which they were embedded, ...

RE needed to consider the effects of the changes, and ...

in some cases, to predict and even alter likely effects.

RE History, Cont'd

As SWSs began to transform organizations deploying them, ...

RE needed to consider how the organization should be transformed and ...

to project organizational-level requirements onto the requirements for the SWSs that were to effect these transformations.

RE History, Cont'd

As more and more technical issues in computing were solved, ...

as computing technology became more stable, and ...

as people began to be more sophisticated about what computing technology could and could not do, ...

emotional issues, which had been overshadowed by technical issues, percolated to the forefront.

RE History, Cont'd

Finally, as users' emotions began to affect how well, and even if, the deployed SWSs would be used in the transformed organizations,

RE needed

- to consider emotional issues and
- to project them onto the requirements for the SWSs triggering the emotional responses.

The Rest of this Talk

The rest of this talk attempts to describe the roles of emotions and of closely related values and beliefs in determining acceptance of deployed organization-transforming SWSs.

It reports case studies in which emotions, values, and beliefs (EVBs) of users and other stakeholders inhibited the success of the deployment of at least the originally conceived SWS.

Values and Beliefs

OT concepts and practices often require changes in or abandoning cherished and long-held beliefs and values.

An example illustrates this issue.

Suppose that Dan dislikes MS Windows (MSW).

No one can force him to like it.

V&Bs, Cont'd

He can be forced to show some appearance of liking it, ...

but then there is no transformation in his fundamental OS preferences, ...

and the forcing only increases his dislike for MSW.

He could be brainwashed, but brainwashing is hardly an enlightened technique.

V&Bs, Cont'd

Dan may be convinced of the advantages of liking MSW, thus ensuring his motivation to cooperate with the transformation.

However, not even Dan can guarantee the transformation of his fundamental OS preferences.

V&Bs, Cont'd

Nevertheless, if Dan is motivated to cooperate, there are some strategies to improve the chances of a successful transformation:

- by constructing pleasant views of Dan's past and future involving MSW, e.g., of playing a pleasant game of solitaire, or
- by addressing the dislike head on, i.e., finding out what Dan dislikes about MSW and then fixing the problem by modifying preferences or even modifying the SW.

V&Bs, Cont'd

Thus, an OT process must plan on spending resources to improve the chances of making the process a success.

Impacts of OT on EVBs

Nowadays, many OTs are initiated by management and fostered by Information and Communication Technology (ICT) gurus.

In the name of so-called best practices, there is little consideration for their ethical and moral implications

Impacts, Cont'd

The implementation of complex systems, such as enterprise resource planning (ERP) systems, are rarely preceded by considerations about employees' job-related fears.

Fears

These employees fear

- for their jobs;
- changes in comfortable procedures;
- continual, unremitting changes;
- degraded meaning of work, as they believe that they will no longer think and will just operate computers that are perceived to do the thinking;

Fears, Cont'd

- degraded meaning of work that can lead to depression;
- increased stress and uncertainty in pursuing task and career interests;
- degraded informal communication that normally brings friendship, trust, a feeling of belonging, and self respect;

Requirements Affected by EVBs

Ramos examined a number of organizations around Portugal, in businesses and universities, that were attempting ICT-based OTs.

She found many examples of

- SW features that raised fears in some stakeholders and
- some SWS development processes that were affected by emotion-driven agendas of some stakeholders.

Four Examples

We examine four of these examples here.

- **Mistake logging system**
- **Giving users more autonomy**
- **Charismatic leader with own agenda**
- **CSCW freak out**

Mistake Logging System

A SWS that was to store information about mistakes and who was responsible for them stressed out many potential users.

They were stressed out to the extent that the mistake-logging features had finally to be removed.

Giving Users More Autonomy

A university library was installing a centralized SWS.

In it, all staff members would have access to all stored information.

Therefore, it would be easier for all to participate in decision making.

All would have more opportunities for autonomy and creativity.

More Autonomy, Cont'd

These effects were worthy goals.

However, this increased access stressed out some potential users, ...

who really preferred not to have access to information not specifically related to their own tasks.

More Autonomy, Cont'd

They were not interested in having more responsibility and more autonomy and in being more creative.

They were comfortable with their current low-responsibility jobs that allowed them to get paid regularly for doing jobs that required very little thinking and no initiative.

They resisted the new work practices.

Leader with Own Agenda

One company had installed an off-the-shelf (OTS) ERP system that met the company's requirements.

The company was trying to motivate people to use it to its full potential.

The charismatic leader of the resource planning department viewed the SWS as reducing the influence of his department and as inhibiting his own promotion as director.

Leader, Cont'd

He began to actively, but surreptitiously, sabotage the use of the new SWS.

This leader's sabotage efforts were so successful that the installation was shelved.

This leader convinced the company that the OTS product was bad and then began an in-house development of a SWS of nearly identical functionality.

Leader, Cont'd

The advantage of the in-house SWS over the OTS SWS was that the slow development would give the leader and his staff a chance to learn and to teach the SWS more gradually, and thus become indispensable to the company.

In other words, the OTS SWS was NIH!

Leader, Cont'd

The in-house development made the charismatic leader so strong that he was able to defeat the intentions of the company's administration.

It was ironic at the end, that after the leader won his war and got his promotion as the director of the department, he proposed the installation of a new ERP system to better control his own staff's work and performance.

CSCW Freak Out

In a university, a computer supported cooperative work (CSCW) system was introduced to the classroom to allow

- the students of a team to work together and
- the faculty member to observe the students' progress.

CSCW, Cont'd

This CSCW system stressed out students

- who had never worked before in teams,
- who did not work well in teams,
- who did not trust others not to mess up their own work, which had been made accessible to others,
- who did not like the idea of instructors observing their work closely in real time, or
- who freaked out when they found files they were editing being modified, as they were working, by others.

Summary of Cases

In each of these cases, the proposed or current SWS caused fears among users because the SWS stood against their EVBs.

The introduction of the SWS was a failure or was not as successful as had been hoped.

Summary, Cont'd

It would have been useful to have had a way to detect these EVB problems

- before the implementations of the SWS had even started, during requirements analysis, or at least,
- before the implementation had progressed very far.

Summary, Cont'd

Detecting these problems early enough would allow properly addressing them either

- as changes to the SWS's requirements or
- by better employee and user training and relations.

Just Managerial Issues?

Some who have heard this talk before accept that there indeed may be the kind of problems mentioned when deploying job-changing ICT applications.

However, they regard them as managerial problems and not as requirements problems.

Just Managerial Issues?, Cont'd

In one sense, these people are right.

The responses to these problems often require action by management, addressing social issues.

Just Managerial Issues?, Cont'd

However, any problem that can prevent the successful deployment of a SWS, whether it be

- incorrect function,
- failure to notice tacit assumptions,
- or anything else

should be identified as early as possible so that dealing with it can permeate the entire system design and development process.

Just Managerial Issues?, Cont'd

Perhaps,

- a so-called managerial problem caused by emotion can be solved by a simple change in functionality or user interface, e.g., by eliminating a hated feature entirely.
- managers and colleagues of the employees that hate the feature should clarify both the business strategy supported by the feature and the benefits of the feature to these employees.

Just Managerial Issues?, Cont'd

Delaying consideration of any problem drives up the cost of solving the problem once it is identified [Boehm 81].

When viewed this way, all such problems become requirement problems.

Just Managerial Issues?, Cont'd

In the end, it may be that the decided-upon solution to an identified problem may be a managerial solution, e.g.,

- educating users and their managers,
- providing incentives for adopting,
- etc.

Just Managerial Issues?, Cont'd

However, such solutions, especially that of educating users, may be applied also to what might appear to be a functional or user-interface issue.

For example, NASA occasionally simply trains users to follow different steps during control of an unmanned deep-space vehicle rather than modify the on-board SW as a solution to a detected failure of the SW to meet its original requirements [Lutz & Mikulski 2004].

Other Observations of EVBs in RE

- Boehm & Ross
- Goguen
- Krumbholz & Maiden
- Bergman, King & Lyytinen
- Huff (Games)
- Sickenius de Souza & others (UIs)
- Boehm & Huang
- Holzinger (Prototyping)
- Pentland, Fletcher & Hasson
- Rost (Sabotage)

Boehm & Ross

Perhaps the earliest, albeit implicit, recognition of the role of emotion in determining requirements for a SWS in the SW engineering research literature was Barry Boehm and Rony Ross's [1989] Theory W.

Theory W and Win-Win conditions [Boehm *et al* 1994] are methods of negotiating requirements so that each stakeholder for a SWS ends up winning ...

winning in the sense that he or she has at least some of his or her requirements satisfied.

Normal alternative has some stakeholders having none of their requirements satisfied in order that others have most of their requirements satisfied.

This alternative is thus termed *Win-Lose*.

Clearly understood reason for preferring Win-Win to Win-Lose:

When all stakeholders win, they all buy into the system, and there is less chance that some will reject the system as not meeting their requirements.

There is less chance of sabotage by losing stakeholders.

Goguen

Joseph Goguen [1993] observed that “It is not quite accurate to say that requirements are in the minds of clients; it would be more accurate to say that they are in the social system of the client organization. They have to be invented, not captured or elicited, and that invention has to be a cooperative venture involving the client, the users, and the developers. The difficulties are mainly social, political, and cultural, and not technical.”

This is from a noted theoretical computer scientist, well known for his contributions to algebraic specifications of SW [Goguen 1979],

Note that this quotation is from a draft that preceded publication as a chapter in a book [Goguen 1994]. The quotation did not survive into the book chapter. However, by e-mailed personal communication, Joseph Goguen assures us that he still believes in the contents of the quotation and that he does not disown it.

Krumbholz & Maiden

Krumbholz, Maiden, *et al* [2000, 2001] investigate the negative impact on user acceptance of ERP induced OT that results from a mismatch between the ERP system's actual and perceived functionality and the users' requirements, including those motivated by their values and beliefs.

Bergman, King & Lyytinen

M.B. Bergman, J.L. King and K. Lyytinen [2002, p. 153] “examines the problem of stating and managing requirements for large system development in terms of heterogeneous engineering, focusing especially on issues of power and interest amongs principals involved in a project.” They observe that requirements engineering (RE) is a political process.

Bergman, King, and Lyytinen note that there is extensive coverage of this view of RE in the IT literature [e.g. Markus 1983] and in the organizational theory literature [e.g., Provan 1980]. They observe that more recently, the SE and RE literature has begun to explore this view as well [e.g. Boehm 1989, Robinson 1998].

Huff

C. Huff [2002] observes that the user orientation of educational and game SW is thought to depend on the gender of the user.

In particular, educational and game SW writers follow gender stereotypes.

Moreover, data show that users do indeed seem to have difficulty with SW for the wrong gender when they use the SW in public view, although not when they use the SW in private.

Clearly, the user interface of SW is a requirements issue, and there are emotional issues coming to play with these user-interface requirements.

Sickenius de Souza *et al*

Sickenius de Souza, Prates, and Barbosa [2003] describe lessons learned from developing information technology to be used by volunteer social workers in Brazil.

The developers recognized that volunteers are motivated not for money or advancement, but for their own satisfaction.

The developers knew that they would have to understand such emotional factors in designing a new interactive system for the volunteers to use to do their work.

The developers decided to use the underlying discourse unveiling method (UDUM), developed originally for clinical psychological research.

UDUM uses open-ended interviews to allow people to talk freely. UDUM (p. 174) “focuses on grasping and analyzing hidden or implicit fears, desires, motivations, aspirations, conflicts, and other deep feelings experienced by individuals.”

Boehm & Huang

Boehm and Huang [2003] describe a new method for tracking a project’s adherence to its schedule and budget.

They observe that the traditional earned-value management process performs well when tracking how closely a project is meeting its original plan.

However, there are difficulties.

(p. 36) “A project can be tremendously successful with respect to cost-oriented earned value, but an absolute disaster in terms of actual organizational value earned. This frequently happens when the product has flaws with respect to user acceptability, operational cost-effectiveness, or timely market entry.”

Thus, at least one of the possible budget, schedule, and value wrecking flaws is user acceptability, an emotional issue.

Indeed, Boehm and Huang cite an example: (p. 36) “the initiative to implement a new order-entry system to reduce the time required to process must convince the sales people that using the new system features will be good for their careers. For example, if the order-entry system is so efficiency-optimized, it doesn’t track sales credits [which prove who sold what], the sales people will fight using it. The salespeople also must be trained to use the system effectively.”

Holzinger

Andreas Holzinger [2004] reports on an effort to use rapid prototyping to identify requirements for a virtual medical campus interface.

He observed that users’ emotions can affect even their willingness to critique a proposed user interface and thus how effective an RE effort is.

(p. 98) “The goal in prototyping is to evaluate the interface’s function and flow, not its look. HTML prototypes, for example, create a polished look but don’t change how quickly and easily users can accomplish a task. The paper prototype proved successful because users felt more comfortable critiquing a paper prototype than a polished coded prototype. When something doesn’t work well in a beautiful interface, users will more likely blame themselves or their lack of experience than the prototype.”

Here, a user's fear of appearing stupid is interfering with rational behavior.

Pentland, Fletcher & Hasson

Sandy Pentland, Richard Fletcher, and Amir Hasson [2004] describe their efforts to deploy a full-coverage broadband wireless infrastructure to allow poor, remote villages in rural India to be connected to the Web without the need to lay expensive wiring to physically connect all the villages to the Internet for instant delivery of packets.

They used the buses that were regularly and frequently circulating among the villages to carry packets, uploading and downloading all accumulated bytes from and to each village as it drove by the village.

The average bandwidth of this delivery is higher than that of a phone line that would provide one telephone per village and synchronous access to the Internet.

The high-bandwidth asynchronous delivery of packets allowed fillable and printable forms to be transmitted and supported asynchronous messaging such as voice mail or e-mail. These asynchronous communications were recognized as more useful to the villagers than a dedicated telephone and phone line would be.

Occasionally, messages would have to be hand-delivered by messengers to recipients.

In recognition of social norms among the villagers, (p. 80) “women were chosen as the community operators ..., since it was socially acceptable for women to deliver messages to everyone in the village.”

In other words, they paid attention to the users’ emotions, values, and beliefs in devising the functionality of the system, and they avoided using methods that might have turned the villagers away from using the system.

Rost

Finally, Johann Rost [2004] writes about political reasons for failed SW projects.

In particular, with several concrete examples, he describes

- how emotions towards a SWS can lead to subversive behavior and
- how this subversive behavior can sabotage SW projects.

Summary

Because most current OTs bring with them the adoption of complex SWSs that support new work concepts and practices, *the elicitation of the requirements of those systems must include the understanding of the involved emotions, values, beliefs, and interests.*

Bibliography

Bergman, M.B., King, J.L., Lyytinen, K.: “Large-Scale Requirements Analysis Revisited: The need for Understanding the Political Ecology of Requirements Engineering”, *Requirements Engineering Journal* 7:3 (2002) 152–171

Boehm, B.W.: *Software Engineering Economics*, Prentice-Hall, Englewood Cliffs, NJ (1981)

Boehm, B.W., Ross, R.: “Theory-W Software Project Management: Principles and Examples”, *IEEE Transactions on Software Engineering* 15 (1989) 902–916

Boehm, B., Bose, P., Horowitz, E., Lee, M.J.: "Software Requirements As Negotiated Win Conditions", *Proceedings of the First International Conference on Requirements Engineering (ICRE94)*, pp. 74–83, IEEE Computer Society, Colorado Springs, CO (April 1994)

Boehm, B.W., Huang, L.G.: "Value-Based Software Engineering: A Case Study", *IEEE Computer* 36:3 (March 2003) 33–41

Dickson, G.W., DeSanctis, G.: *Information Technology and the Future Enterprise: New Models for Managers*, Prentice Hall, Englewood Cliffs, NJ (2000)

Finkelstein, A.C.W.: "Report of the Inquiry Into The London Ambulance Service", Report to The Communications Directorate, South West Thames Regional Health Authority, Original ISBN No: 0 905133 70 6 (February 1993)
<http://www.cs.ucl.ac.uk/staff/A.Finkelstein/las/lascase0.9.pdf>

Finkelstein, A.C.W., Dowell, J.: "A Comedy of Errors: the London Ambulance Service Case Study", *Proceedings of 8th International Workshop on Software Specification and Design, IWSSD-8*, IEEE Computer Society Press (1996) 2–4

Goguen, J.A.: "Requirements Engineering as the Reconciliation of Technical and Social Issues", Technical Report, Computing Laboratory, Oxford University, Oxford, UK (1993)

Goguen, J.A.: "Requirements Engineering as the Reconciliation of Technical and Social Issues", In: Goguen, J.A., Jirotko, M.: *Requirements Engineering: Social and Technical Issues*, Academic Press, London, UK (1994) 165–199

Goguen, J.A., Tardo, J.J.: "An Introduction to OBJ: A Language for Writing and Testing Formal Algebraic Program Specifications", *Proceedings of a Conference on Specifications of Reliable Software*, Cambridge, MA (April 1979) 170–189

Huff, C.: "Gender, Software Design, and Occupational Equity", *Inroads, SIGCSE Bulletin* 34:2 (June 2002) 112–115

Holzinger, A.: "Rapid Prototyping for a Virtual Medical Campus Interface", *IEEE Software* 21:1 (January/February 2004) 92–99

Iivari, J., Hirschheim, R., Klein, H.K.: "A Paradigmatic Analysis Contrasting Information Systems Development Approaches and Methodologies", *Information Systems Research* 9:2 (1998) 164–193

Krumbholz, M., Galliers, J., Coulianos, N., Maiden, N.A.M.: "Implementing Enterprise Resource Planning Packages in Different Corporate and National Cultures", *Journal of Information Technology* 15 (2000) 267–279

Krumbholz, M., Maiden, N.A.M.: "The Implementing of ERP Packages in Different Organisational and National Cultures", *Information Systems Journal* 26:3 (2001) 185–204

Lehman, M.M.: "Programs, Life Cycles, and Laws of Software Evolution", *Proceedings of the IEEE* 68:9 (September 1980) 1060–1076

Lutz, R.R., Mikulski, I.C.: "Empirical Analysis of Safety-Critical Anomalies During Operations", *IEEE Transaction on Software Engineering* SE-30:3 (March 2004) 172–180

Lyytinen, K., Mathiassen, L., Ropponen, J.: "Attention Shaping and Software Risk—A Categorical Analysis of Four Classical Risk Management Approaches", *Information Systems Research* 9:3 (1998) 233–255

Markus, M.L.: "Power, Politics and MIS Implementation", *Communications of the ACM* 26 (1983) 430–444

Palmer, I., Hardy, C.: *Thinking about Management*, Sage, Thousand Oaks, CA (2000)

Pentland, A.S., Fletcher, R., Hasson, A.: "DakNet: Rethinking Connectivity in Developing Nations" *IEEE Computer* 37:1 (January 2004) 78–83

Provan, K.G., Beyer, J.M., Kruytbosch, C.: "Environmental linkages and power in resource-dependence relations between organizations", *Administrative Science Quarterly* 25 (1980) 200–225

Robinson, W.N., Volkov, V.: "Supporting the Negotiation Life Cycle", *Communications of the ACM* 41 (1998) 95–102

Rost, J.: "Political Reasons for Failed Software Projects", *IEEE Software*, 21:6 (November+December 2004) 104, 102–103

Sickenius de Souza, C., Prates, R.O., Barbosa, S.D.J.: "Adopting Information Technology as a First Step in Design, Lessons Learned from Working with Brazilian Social Volunteers", *ACM Interactions* X:2 (March+April 2003) 72–79