Supporting Global Software Development in Open Source Ecosystems: A Role for Actability in the Pragmatic Web

Pär J. Ågerfalk
Lero – The Irish Software Engineering Research Centre
University of Limerick, Limerick, Ireland
par.agerfalk@ul.ie

Abstract: New forms of collaboration between organizations based on open source principles are rapidly emerging. The collaboration is typically done in a spirit of co-opetition whereby companies, often SMEs, share cost and risk by developing software jointly and openly. The paper elaborates how this emerging phenomenon of open source ecosystems can be understood from the perspective of actability and the Pragmatic Web. The concept of open source ecosystems as a form of global software development is explored and actability is presented as a useful concept for articulating design criteria for the required collaborative tools. In doing so, a possible research agenda for pragmatic web research in this domain is outlined.

1 Introduction

The open source software (OSS) landscape is a rapidly changing one. While OSS and its Free Software antecedent were primarily driven by ideology and individual commitment, the main driving force of OSS today is commercialization and opportunities for inter-organizational collaboration [Fi06]. OSS is no longer mainly about enthusiasts contributing to SourceForge projects, but increasingly about commercial organizations developing software in 'co-opetitive ecosystems' [ÅFH06]. Many small and medium-sized enterprises (SMEs) see this mode of working as a way of sharing risks and costs and an opportunity to participate in cutting edge research that does not require a significant R&D budget [ÅDF05]. This trend is in line with recent studies which show that a significant proportion of contributors to OSS projects are paid employees [FL02, DWA03] and supported by many recent commercial OSS events[1]. Understanding the underlying communicative mechanisms of such ecosystems and designing information technology (IT) to support the collaboration is a challenging task for information systems research.

From a software/information systems development perspective, this collaboration (or rather co-opetition since the collaborators are often also competitors in the marketplace) can often be seen as a particular form of globally distributed software

[1] See, for example, www.calibre.ie and www.osbc.com.

development [LLÅ06]. This mode of software development, often referred to as GSD (global software development), emphasizes the problems associated with increased geographical, temporal and socio-cultural distance between people [ÅFH05]. The implications of increasing distance are also central in recent work on information systems actability [Åg04] which emphasizes that the introduction of IT for communication in business processes also increases distance between people. Particularly, communication through and by means of IT moves the business interaction away from an "ideal" face-to-face setting [Cl96] towards an IT mediated one [Åg03, Åg04]. Actability is founded on a pragmatic view of IT use and thus appears as a fruitful concept in a discussion about the Pragmatic Web. The notion of the Pragmatic Web, as it has been proposed by Schoop and colleagues [SMD06] captures many of the problems faced in open source ecosystems generally and in supporting their GSD efforts particularly: "to augment human collaboration effectively by appropriate technologies … in [distributed] communities of practice."

This essay introduces the concept of open source ecosystems as a form of GSD and explores how actability principles can be useful in articulating design criteria for the required collaborative tools. In doing so, a research agenda for Pragmatic Web research in this domain is outlined. Given that OSS communities have traditionally been viewed as collections of loosely coupled individuals with a common 'itch worth scratching' [Ra99], it is important to emphasize that the kind of OSS development we are concerned with here is open collaboration between commercial organizations. Such collaboration is more structured and organized than the traditional 'bazaar style OSS development model' [Fi06] and thus more in line with the sort of institutionalized setting that the actability concept has been explored in previously (i.e. a work context).

2 GSD in Open Source Ecosystems

Recent research on the open source software (OSS) phenomenon suggests that we are currently witnessing an ongoing shift from OSS as community of individual developers to OSS as community of commercial organizations, often SMEs, operating as symbiotic ecosystems in a spirit of co-opetition [ÅDF05, ÅFH06, Fi06]. These ecosystems are not restricted to software development collaboration, but rather draw on open philosophies in a wide range of domains (open content, open knowledge, open standards, etc). However, the impact on software development organizations is of particular interest since OSS can be seen both as an approach to inter-organizational collaborative software development and as the result of such collaboration to be employed by other, non software developing organizations and by the software developing organizations themselves. Interestingly, many commercial organizations involved in this emerging second wave of OSS can be characterized as 'secondary software companies' [ÅFD05]. These are companies whose

main product is not software, but of which software plays an important part; telecom, automotive and medical devices are good examples.

The OSS development model is by its very nature globally distributed [LLÅ06, ÅFH06]. Globally distributed software development (GSD) has been characterized as a setting where increased geographical, temporal and socio-cultural distance affects the processes of communication, coordination and control in software projects [ÅFH05]. Temporal distance is here thought of as 'a directional measure of the dislocation in time experienced by two actors wishing to interact', caused by, for example, time zone difference; geographical distance is 'a directional measure of the effort required for one actor to visit another at the latter's home site'; and socio-cultural distance is 'a directional measure of an actor's understanding of another actor's values and normative practices'. In the context of GSD, the communication process concerns transfer of information and creation of common understanding, and the tools used to facilitate such interaction. Coordination has been defined as 'the act of integrating each task with each organisational unit, so the unit contributes to the overall objective.' [CA01] The coordination process thus concerns interdependencies between actors: 'Two people have a coordination problem whenever they have common interests, or goals, and each person's actions depend on the actions of the other.' [Cl96] Finally, control is 'the process of adhering to goals, policies, standards, or quality levels.' [CA01] The control process thus concerns the management and reporting mechanisms needed to make sure a development activity is progressing. Table 1 summarizes these concepts and points at some of the characteristics of GSD in an OSS context.

Process	Dimension		
	Temporal Distance	*Geographical Distance*	*Socio-Cultural Distance*
Communication	Practically all routine communication asynchronous, through the Internet.	Typically, developers acting as the market. Internet used creatively for communication channels.	Responsive communities of motivated, self-selected contributors.
Coordination	Preponderance of modular, plug-in style architectures, reducing the need for coordination.	Dynamic and flexible labour pools. High critical task awareness throughout the community.	Common environments based on free, lightweight tools and lightweight process infrastructures.
Control	Typically 24x7 working. Control primarily through the commit process.	Mechanisms in place for identifying and addressing the issue of non-active key members.	Shared project goals and no project forking. Protection of OSS values. High level of activity on mundane tasks.

Table 1: Characteristics of GSD in an OSS context [LLÅ06].

As can be seen from Table 1, communication, and, as a consequence, mechanisms for coordination and control, are primarily implemented through IT-based solutions and predominantly over the Internet. Typically, the technologies used provide fairly low-level support and include simple configuration management tools such as CVS, mailing lists, and IRC.

3 Actability Principles

To fully support workpractice action and communication, it is important to see IT as a tool that mediates social action. To understand the particular features of IT in such settings, we may draw on the casual face-to-face conversation made by Clark [Cl96]. Clark suggested this model as a benchmark for understanding other communication situations, and by using this setting as an ideal type we can see how the introduction of IT changes the situation. To understand the face-to-face situation, Clark suggests ten typical features: *co-presence* (participants share the same physical environment), *visibility* and *audibility* (they see and hear each other), *instantaneity* (they recognize each other's action at no perceivable delay), *evanescence* (the medium fades immediately), *recordlessness* (actions do not leave any record or artefact), *simultaneity* (participants may receive and produce at once and simultaneously), *extemporaneity* (actions are formulated and executed in real time), *self-determination* (participants determine for themselves what actions to take when), *self-expression* (participants take actions as themselves) [Cl96]. Table 2 summarizes an interpretation of these features from the perspective of actability, as used in the remainder of this paper and defined below. The comparison and interpretation is based on the notion that workpractice communication is a special type of norm-based context (a work context) and that IT is a special type of medium for conversation (speech act exchanges).

The concept of actability was introduced as a way of conceptualizing the use of IT in organizations from a pragmatic point of view. Within actability theory [GÅ02, Åg03, Åg04] IT is viewed primarily as a tool for social (inter-personal) action and communication. This is in line with the language/action perspective which suggests that the real power of computers is to support communication, not computation *per se* [Fl98, Sc01, Di03, Di04]. Actability can be defined as: 'an information system's ability to perform actions, and to permit, promote and facilitate the performance of actions by users, both through the system and based on information from the system, in some work context.'

In essence, actability promotes systems that are 'easy to use' and also explicit about what actions are possible to perform. They should furthermore encourage users to benefit from acting through them so that an organizational action memory can be maintained by the IS. That is, information systems should *permit, promote and facilitate the performance of actions by users.*

Feature	*Face-to-face conversation*	*Workpractice action through IT system*
Co-presence	Participants share the same physical environment	Participants may not share the same physical environment
Visibility	Participants can see each other	Participants may not see each other
Audibility	Participants can hear each other	Participants may not hear each other
Instantaneity	Participants perceive each other's actions at no perceptible delay	Participants may perceive each other's actions with considerable delay
Evanescence	Medium is evanescent – it fades quickly	Medium is persistent – it may stay until the system is shut down
Recordlessness	Participants' actions leave no record or artefact	Participants' actions may leave a record in an "action memory" (e.g. a database)
Simultaneity	Participants can produce and receive at once and simultaneously	Participants either produce or receive as separate acts
Extemporaneity	Participants formulate and execute their actions extemporaneously, in real time	Participants may formulate and execute their actions reflectively during extended amounts of time
Self-determination	Participants determine for themselves what actions to take when	Workpractice norms and system design determine (to a large extent) what actions to take when
Self-expression	Participants take actions as themselves	Participants may take action on behalf of other people and their organization

Table 2: Features of face-to-face conversations [Cl96] as compared to workpractice communication through and by means of IT. (After [Åg04])

Actions are here thought of as 'social actions', i.e. intentional actions that takes into account the behaviour of others [We78], or, more specifically, as speech acts [Au62, Se69] or communicative actions [Ha84], following the language/action perspective. Within actability theory, such actions are referred to as elementary communicative actions (or e-actions), which generates action-elementary messages (or ae-messages) communicated through and by means of the system. Ae-messages are elementary information units carrying a propositional content (what is talked about) and an associated action mode (representing the speaker's intention, or what Searle termed 'illocutionary point' [Se69]). The concept of the ae-message is thus based on the fundamental language/action thesis that language use is not restricted to making descriptions of reality, i.e. to refer and to predicate. Rather, language is often used to perform actions. People do things when speaking, such as promising, ordering and declaring [Au62, Se69]. Someone performs an e-action to effect a social change (which may or may not have material consequences). This creates an action relationship between the speaker and one or more listeners [Ha84, GÅ02, Åg02].

Saying that information systems have an ability to perform actions implies that they can be seen as agents performing actions on behalf of some human actor. There is always a human actor ultimately responsible for such 'automatic actions', which are always derived from predefined rules. Actions can also be performed *through the system*, such as when a user performs actions with an information system as a tool for communication. Conversely, *based on information from the system* implies that an information system can also be used to create possibilities for action. The actability achieved in a certain situation is always related to a particular *work context* in which the information system is used. The work context includes actors' pre-knowledge and skills regarding both the information system and the work tasks performed. [GÅ02]

To facilitate actability design and evaluation, nine actability dimensions have been suggested along which the pragmatic usefulness of IT can be analysed [Åg04]. These dimensions should be understood in the light of many years of research into the success of information systems from a use perspective [Da89, DM92, GT95, Be99, DM03]. Several lists of criteria focusing on usability and user interface design and its relation to user and task characteristics have been proposed, including Nielsen's ten usability heuristics [Ni94] and Shneiderman's eight golden rules [Sh98]. Such criteria are often grounded in cognitive psychology and tend to employ an overly instrumental view on IT use [ÅE06]. The actability dimensions, on the other hand, concern institutionalized settings in which IT is used as a tool to perform communicative business action. As shown below, each dimension highlights a number of criteria for design and evaluation by means of questions to ask. The actability dimensions and associated criteria/questions are derived from the concept of actability and the set of ten significant features of casual face-to-face conversations introduced above (Table 2). Note that the following description of the nine dimensions is only meant as a brief summary and the reader is referred to [Åg04] for more in-depth elaboration.

The *action elementariness dimension* reflects the notion that information systems are systems for handling messages as (semiotic) results of communicative actions. This dimension addresses questions such as:

- Is it clear who says what to whom with what intentions, and, that this is done on behalf of someone else, if that should be the case?
- Are separate messages kept separate? That is, are users forced to do (or to make sense of) more than one thing at a time?

The *recorded action dimension* reflects the notion that users' actions may leave a record in the action memory of the information system. This dimension addresses questions such as:

- Does the system store and provide access to what has previously been said and done using the system?

- Does the system keep track of who said what to whom?

The *action potentiality dimension* reflects the notion that an information system can be understood as the set of communicative actions it affords and supports. This dimension addresses questions such as:

- Are required actions afforded?
- Are known and understandable effects of possible actions communicated?
- Are expressive interactive user interface components used (icons, labels, *et cetera*)?
- Is information that the system requires from users meaningful and easily provided to the system?
- Is information shown adequate (necessary and sufficient) so that it can be readily used as a basis for action?
- Does the language used correspond to the users' professional language?
- Does the system support justification by explanations, and possibly negotiation, of the action potential and its communicative validity?

The *structured action dimension* reflects the notion that business rules to a large extent determine what actions to take, and when to take them. This dimension addresses questions such as:

- Does the system admit focus and work task changes?
- Is the navigation style made explicit?
- Are sequence restrictions enforced when necessary and desirable, and only then?
- Does the system assist performers in knowing what they are doing, and what they are supposed to be doing?
- Is choice of course of action to take legibly informed by the system?
- Does the system support the following-up of previous commitments made?

The *irrevocable action dimension* reflects the notion that business messages may be formulated and executed reflectively during extended amounts of time. This dimension addresses questions such as:

- Is the system explicit about when a social action is actually performed?
- Is rollback (undo) provided as far as socially acceptable?

The *remote activity dimension* reflects the notion that participants may not share the same physical environment. This dimension addresses questions such as:

- Is the receipt and interpretation of messages possible at desired places?
- Is the receipt and interpretation of messages possible in desired ways?
- Is action potential provided where and when it is needed?

The *delayed interpretation dimension* reflects the notion that participants may perceive each other's actions with considerable delay. This dimension addresses questions such as:

- Can we (always) tell when an (important) action was performed?
- Do messages reach intended interpreters in due time?

The *delayed feedback dimension* reflects the notion that communicating users either produce or receive, but not simultaneously. This dimension addresses questions such as:

- Do users understand that no feedback on communication effects is given until a message has been delivered, interpreted and acted upon?
- Is delayed feedback on communication effects minimized and, if known or anticipated, communicated to users?

The *delegated action dimension* reflects the notion that users and systems may take action on behalf of other people and of their organization. This dimension addresses questions such as:

- Is performance of action allocated to human actors and IT systems so that users gain maximum support?
- Are descriptions and explanations of the system's performed and scheduled future action(s) readily available?
- Are users aware of their action relationships?

4 A Case for Pragmatic Web Research

The actability dimensions were developed and validated primarily in a traditional business information system context. Such a setting means that organizational norms and business rules are, if not well-known at least possible to elicit and document; as is actors and actor roles with associated action responsibilities. When moving into the realm of open source ecosystems, many of these properties do not hold. For example, as in any 'virtual community', the actual people involved and their norms, values, needs, etc are not always possible to identify and explicate beforehand. Similarly, the communication structures and interaction patterns evolve during the course of evolution of the ecosystem. To support GSD efforts in these emerging ecosystems of people and organizations there is a great need for collaborative environments that can adapt to local needs, yet provide facilities for implementing the required control structures. We believe that the concept of actability and its associated dimensions are crucial in such an endeavour. This is because, as described above, actability immediately addresses the challenges of increased distance and provides detailed design guidelines for how to make sure that communication through and by means of IT is supported at both the semantic and the pragmatic level [Åg04, ÅE04, ÅE06]. Also, the pragmatic orientation of actability goes well with a value-based phenomenon such as OSS. Understanding the

underlying values (for example, the conflict between business/economic value and 'open' community values) is arguably key for co-opetition to be successful [Fi06]. In a Pragmatic Web context this would leverage the current use of Internet-based technologies in these ecosystems while making sure that new tools support emerging social action contexts properly.

A possible mapping of the actability dimensions to the open source GSD ecosystem context and a number of areas where the different dimensions could be particularly useful are outlined in the remainder of this section (and summarized in Table 3). Indeed, this is an initial tentative mapping and further research is required to fully realize the potential of actability in the Pragmatic Web in this context. Also, as indicated above, the actability dimensions and associated design guidelines need to be tailored to cater for the emergent and dynamic nature of open source ecosystems as compared to traditional in-company information systems.

Process	Dimension		
	Temporal Distance	*Geographical Distance*	*Socio-Cultural Distance*
Communication	*Recorded Action* to make sure that the asynchronous communication through the Internet is captured and maintained at both semantic and pragmatic level.	*Remote Activity* to explore what type of communication solutions best support different ecosystems.	*Action Elementariness* to facilitate bridging of possible socio-cultural mismatch and creation of appropriate pragmatic ontologies to support communication across organizations.
Coordination	Although typical OSS architectures reduce coordination needs, understanding properties of *Structured Action* and *Irrevocable Action* becomes critical in more formalized projects, which appears to becoming the norm.	Given high critical task awareness throughout the community, understanding effects of *Delayed Interpretation* and *Delayed Feedback* is crucial to maintain proper coordination.	*Action Potentiality* and *Delegated Action* to support the proper design of the required common environments and to understand the delegation structures implemented and supported by those.
Control	Although control in OSS is primarily through the commit process, understanding the *Structured Action* and *Irrevocable Action* is crucial in more formalized projects.	*Delegated Action* as a tool to understand how to implement mechanisms for identifying and addressing the issue of non-active key members and other emerging issues related to the agency of the ecosystems.	*Delegated Action* as a way of understanding the agency properties of the ecosystems and to make sure that all agents work towards common goals.

Table 3: GSD in open source ecosystems from an actability perspective.

4.1 The Temporal Distance Dimension

Since communication in open source ecosystems is generally asynchronous over the Internet, the *recorded action* dimension could be useful in designing a communication infrastructure such that relevant aspects of this communication is captured and maintained at both semantic and pragmatic level. Today's communication tools, mailing lists and IRC, for example, are typically not utilizing the potential in a pragmatic approach to structuring information so that previous conversations and commitments are easily traceable and retrievable.

As OSS projects are becoming more formalized, understanding properties of *structured action* and *irrevocable action* becomes critical. Although typical OSS architectures reduce coordination needs, communication support should be aligned with the software development process used. Since OSS projects are typically using flexible and lightweight process infrastructures, there is a need to understand these dimensions in light of fluid and agile action structures. Still, making sure that everyone knows when an action has been performed and how that action relates to other actions (previous and future) is essential. This is probably especially important if coordination of the project relies on the commit process, which is often the case in OSS development. A relevant question to ask is, for example, what does it mean, socially, to commit a changed software artefact to a repository.

4.2 The Geographical Distance Dimension

Each open source ecosystem will have their own particular needs when it comes to supporting action and communication in the software development process. It is becoming increasingly common in OSS development to organize events where developers get together and work collocated for short periods. This should of course be considered in the development of open source ecosystems, but may not be suitable in every case. The *remote activity* dimension allows for exploration of what type of communication solutions best support different ecosystems.

Understanding effects of *delayed interpretation* and *delayed feedback* is likely to be crucial in maintaining proper coordination in open source ecosystems. Especially since coordination in OSS development often relies on high critical task awareness throughout the community.

The *delegated action* dimension should be explored in order to understand how to implement mechanisms for identifying and addressing the issue of non-active key members and other emerging issues related to the agency of the ecosystems. In general, since agent technology is likely to be of increasing importance, understanding what action responsibilities that are created and maintained throughout the ecosystem is essential.

4.3 The Socio-Cultural Distance Dimension

A key problem in GSD is the possible socio-cultural mismatch between the people involved. Although this is less of a problem in OSS, perhaps because OSS developers are often self-selected and share the same 'itch', the notion of *action elementariness* could help facilitate bridging socio-cultural gaps. One way to do this could be to create appropriate pragmatic ontologies to support communication across organizations and cultures. This way, intentions, understandings and ways of working have to be explicitly agreed upon and codified.

Investigating the *action potentiality* and *delegated action* dimension could support the design of collaborative work environments and help ensuring that these are aligned with the delegation structures implemented and supported in these environments.

Finally, and returning to *delegated action*, understanding the agency properties of the ecosystems and making sure that all agents work towards common goals is probably significant in order to bridge possible socio-cultural differences.

5 Conclusion

This essay has introduced the emerging phenomenon of open source ecosystems and explored how actability and ideas underlying the Pragmatic Web can be useful for supporting global software development in such a context. We have identified a number of areas where actability principles may be useful in this endeavour and also pointed at some issues that may require tailoring of these principles for this new and emerging context. The suggested mapping between actability dimensions and GSD in open source ecosystems (Table 3) can thus be seen as a proposed agenda for research on actability in the Pragmatic Web to support this emerging mode of collaboration (or co-opetition) between competitors.

Acknowledgements

This work has been financially supported by the Science Foundation Ireland Investigator Programmes B4-STEP (Building a Bi-Directional Bridge Between Software ThEory and Practice) and Lero – The Irish Software Engineering Research Centre, and by the EU FP6 projects CALIBRE (Coordination Action for Libre Software) and OPAALS (Open Philosophies for Associative Autopoietic Digital Ecosystem).

References

[Åg02] P. J. Ågerfalk. Messages Are Signs of Action: From Langefors to Speech Acts and Beyond, In Barjis J, et al. (eds.), *Proc. 7th Int. Workshop on the Language-Action Perspective on Communication Modelling* (Lap 2002), pp. 81–100, Delft, The Netherlands, 2002. Delft University of Technology.

[Åg03] P. J. Ågerfalk. *Information Systems Actability: Understanding Information Technology as a Tool for Business Action and Communication*, Doctoral dissertation, Department of Computer and Information Science, Linköping University, Linköping, Sweden, 2003.

[Åg04] P. J. Ågerfalk. Investigating Actability Dimensions: A Language/Action Perspective on Criteria for Information Systems Evaluation, *Interacting with Computers*, 16(5): 957–988, 2004.

[ÅDF05] P. J. Ågerfalk, A. Deverell, B. Fitzgerald, L. Morgan. Assessing the Role of Open Source Software in the European Secondary Software Sector: A Voice from Industry, In M. Scotto, G. Succi (eds.), *Proc. 1st Int. Conf. on Open Source Systems*, pp. 82–87, Genoa, Italy, 2005.

[ÅE04] P. J. Ågerfalk, O. Eriksson. Action-Oriented Conceptual Modelling, *European Journal of Information Systems*, 13(1): 80–92, 2004.

[ÅE06] P. J. Ågerfalk, O. Eriksson. Socio-Instrumental Usability: IT Is All About Social Action, *Journal of Information Technology*, 21(1): 24–39, 2006.

[ÅFH05] P. J. Ågerfalk, B. Fitzgerald, H. Holmström, B. Lings, B. Lundell, E. Ó Conchúir. A Framework for Considering Opportunities and Threats in Distributed Software Development, In *Proc. Int. Workshop on Distributed Software Development (DiSD 2005)*, Paris, France, 2005.

[ÅFH06] P. J. Ågerfalk, B. Fitzgerald, H. Holmström, E. Ó Conchúir. Open-Sourcing as Offshore Outsourcing Strategy, In *Proc. 29th Scandinavian Information Systems Research Seminar (IRIS'2006)*, Helsingør, Denmark, 2006.

[Au62] J. L. Austin. *How to Do Things with Words*. Oxford University Press, Cambridge, 1962.

[Be99] N. Bevan. Quality in Use: Meeting User Needs for Quality, *Journal of Systems and Software*, 49(1): 89–96, 1999.

[CA01] E. Carmel, R. Agarwal. Tactical Approaches for Alleviating Distance in Global Software Development, *IEEE Software*, 18(2): 22–29.

[Cl96] H. H. Clark. *Using Language*. Cambridge University Press, Cambridge, 1996.

[Da89] F. D. Davis. Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology, *MIS Quarterly*, 13(3): 319–340, 1989.

[DM92] W. H. DeLone, E. R. McLean. Information Systems Success: The Quest for the Dependent Variable, *Information Systems Research*, 3(1): 60–95, 1992.

[DM03] W. H. DeLone, E. R. McLean. The DeLone and McLean Model of Information Systems Success: A Ten-Year Update, *Journal of Management Information Systems*, 19(4): 9–30, 2003.

[Di03] J. L. G. Dietz. The Atoms, Molecules and Fibers of Organizations, *Data & Knowledge Engineering*, 47(3): 301–325, 2003.

[Di04] J. L. G. Dietz. Towards a LAP-Based Information Paradigm, In M. Aakhus, M. Lind (eds.), *Proc. 9th Int. Working Conf. on the Language-Action Perspective on Communication Modelling (Lap 2004)*, pp. 59–76, Rutgers University, The State University of New Jersey, New Brunswick, NJ, USA.

[DWA03] P. A David, A. Waterman, S. Arora. *FLOSS-US: The Free/Libre/Open Source Software Survey for 2003*. Stanford University, California, USA, Available online at http://www.stanford.edu/group/floss-us/report/, Accessed 2006-08-03.

[Fi06] B. Fitzgerald. The Transformation of Open Source Software, *MIS Quarterly*, 30(3): 587–598, 2006.

[Fl98] F. Flores. Information Technology and the Institution of Identity: Reflections since Understanding Computers and Cognition, *Information Technology & People*, 11(4): 352–372, 1998.

[FL02] *Free/Libre and Open Source Software: Survey and Study*, FLOSS Final Report, International Institute of Infonomics, University of Maastricht, The Netherlands and Berlecon Research GmbH, Berlin, Germany, Available online at http://www.infonomics.nl/FLOSS/report/index.htm, Accessed 2006-08-03.

[GÅ02] G. Goldkuhl, P. J. Ågerfalk. Actability: A Way to Understand Information Systems Pragmatics, In K. Liu et al. (eds.), *Coordination and Communication Using Signs: Studies in Organisational Semiotics 2*, pp. 85–113, Kluwer Academic Publishers, Boston, 2002.

[GT95] D. L. Goodhue, R. L. Thompson. Task-Technology Fit and Individual Performance, *MIS Quarterly*, 19(2), pp. 213–236, 1995.

[Ha84] J. Habermas. *The Theory of Communicative Action*. Polity Press, Cambridge, 1984.

[LLÅ06] B. Lundell, B. Lings, P. J. Ågerfalk, B. Fitzgerald. The Distributed Open Source Software Development Model: Observations on Communication, Coordination and Control. In *Proceedings of ECIS'2006*, Gothenburg, Sweden, 2006

[Ni94] J. Nielsen. Heuristic Evaluation, In J. Nielsen, R. L. Mack (eds.), *Usability Inspection Methods*, pp. 25–64, John Wiley & Sons, New York, NY.

[Ra99] E. S. Raymond. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*, O'Reilly, Sebastopol, CA, 1999

[Sc01] M. Schoop. An Introduction to the Language-Action Perspective, *ACM SIGGROUP Bulletin*, 22(3): 3–8, 2001.

[SMD06] M. Schoop, A. de Moor, J. L. G. Dietz. The Pragmatic Web: A Manifesto, *Communications of the ACM*, 49(5): 75–76, 2006.

[Se69] J. R. Searle. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, Cambridge, 1969.

[Sh98] B. Shneiderman. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, 3rd ed. Addison Wesley Longman, Reading, MA, 1998.

[We78] M. Weber. *Economy and Society*. University of California Press, Berkeley, CA, 1978.