

Co-Design Metaphors and Scenarios - Two Elements in a Design Language for Co-Design

Lars Albinsson

Calistoga Springs Research Institute/School of Business and Informatics,
University College of Borås, Sweden
lars@maestro.se

Prof. Olov Forsgren

School of Business and Informatics,
University College of Borås, Sweden
Olov.forsgren@hb.se

Abstract

Design Metaphors and Co-Design Scenarios are discussed as important tools for a Co-Design approach for IS. The advantages of an electronic assistant metaphor are indicated. Research is based on a number of case studies exemplified with the EU fifth framework project Avanti.

Keywords: Co-Design, Design Language, Design Metaphor.

1 Introduction

Within IS development there has been a gradual increment of the importance of stakeholder involvement. Back in the 60:ies and 70:ies IS design was a job for analytical experts. In the 80:ies the arrival of personal computers and graphical user interfaces brought forward ideas of user participation in the design process. The idea of Participatory Design and the “Scandinavian School” has been rather influential. [1] They realised that not only the users of IS are stakeholders but also the management of organisation the users belong to. They, however, took the political position that workplaces should be democratised and advocated strongly that only the user’s/worker’s perspective should guide the development.

The increased use of IS in market communication brought the customer in as potential stakeholder gradually during the 90:ies, with the web boom as the peak of the fireworks. In the late 90:ies IS was also beginning to be viewed as important strategically in business development; hence management also become a stakeholder in more than just the cost cutting perspective. As a response traditional marketing techniques as market surveys, customer focus groups as well as traditional management tools like competitions models, product life cycle models, the process metaphor etc were brought into IS development projects.

The consequence of this can be described as a mess. The projects were staffed with people from disciplines that had no history of working together and the professionals had no common (design) language. Each groups did (and indeed still does) argue their perspective to be the most important and hence their language as the principal in project. Typically the one stakeholder with no voice in this battle is often the client/citizen.

Studies of IS project failures [1] clearly states that “unclear and changing requirements” are the main culprit. This problem is mostly described as miscommunication between managers,

users, IS developers and clients/citizens (in case they were at all involved.) Research on eGovernment also states that many citizens are unable to express requirements in the “language” that is used in the current methodologies. [2] Löwgren and Stolterman also argue the need for more work on Design Languages. [3]

In this paper we are developing some ideas about how this communication problem in the design process can be supported by the use of good design languages; Languages that makes it possible for the stakeholder interests to express themselves and also to imagine a possible new upcoming situation as a result of the design process. Before that we will just briefly mention some historic notes on the theoretical justification of why it is important to involve stakeholder interests in the design process.

1.1 LAP and Co-Design two perspectives advocating stakeholder involvement

One of the most important problems during the era of IT-Solutions (ITS) in business and organisations has been how to get the solutions used so that they give value for made investments. During the years we have seen many approaches to this problem, from more engineering scientific approaches saying that if we only have a precise well defined method guiding the process of developing IT, then we will get good solutions, all the way to design approaches saying that it is a matter of the right look and feel, and that is an art.

We can trace a major shift in the approaches, from “perspective unconscious” approaches to “perspective conscious” approaches. Or to put it another way: In the early and mid eighties there were a number of attempts to apply and develop modern social constructivist and design philosophies of knowledge and language into the area of ITS in order to see if this could generate more ITS value. [4] One of these approaches was formulated by Winograd and Flores as the LAP-Language Action Perspective. [5, 6] This work was inspired by Flores earlier work at Berkeley. [7] Flores himself was embedded in a supervising team with Heinz von Forster as an intellectual dominant. [8] The conversation base for this team, including Huberto Maturana, was the American Association for Cybernetics and the Second Order Cybernetics Group[9]. Also in that group was Olov Forsgren [10] and while inspiration sources for this group were many, one of the more important was C West Churchman. [11]

Very much inspired by Churchman and the Churchmanian perspective Forsgren and others later has continued to develop a perspective conscious approach to ITS and named it Co-design(Samskapande.)[12] Co-design has a lot in common with the LAP perspective. Most important is part of the same strong criticism of rational or depictive, analytical approaches and philosophies. Also the idea that the LAP or CO-design perspective, as such, is just one of many possible perspectives is common.

But there are also some differences worth mentioning. The LAP-perspective as well as the Churchmanian perspective is integrative, or sweep-in oriented. The main idea is that the more perspectives we use, the better understanding we get. The Co-design perspective is more pragmatic, it says at some level you have to stop sweeping in and then act. Based on reactions a new sweep-in process of new perspectives may take place and so on. When to stop the sweeping in process is a question of the right feeling, a conscious decision that has to be made. The other difference is that The LAP perspective is focused on language as action. The co-design perspective is as much focused on action as action. Finally the most important difference is that the Co-design perspective looks at itself as a designed and evolving perspective, which can be implemented and reflected upon, from other perspectives. It is successful by its own criteria, which means if the main defined stakeholder’s views are involved and satisfied in an evaluation conversation about the result of the implementation of co-design ideas, then the co-design perspective regard itself as successful. The original Churchmanian perspective is more doubting. It says we can never know that we have defined

the most important stakeholder's views, so here we have to trust God. The Co-design perspective says, we can not know for sure but we have to make design decision and then act, after that reflection and so on. We are God when we make those design decisions.

The original LAP perspective has hard to follow this conversation. It is still in a view where as many as possible complementary views together forms a more complete "representation" of the studied phenomena. At the same time the LAP perspective is allergic to the concept "representation" because it leads us directly back to the rational or depictive, analytical approaches and philosophies. In that way the original LAP-perspective involves deep contradictions, which is really hopeful for the future from a co-design perspective.

This was some past flashing memories from the conversations way back in the Second Order Cybernetics Group in the beginning of the eighties. After that both the co-design and the LAP perspectives have evolved, but still with common roots in the first attempts to formulate an alternative to the "rationalistic (depictive)¹ view of knowledge and understanding as mental representations of an objectively existing world" [14]. A central theme is the interest in involvement of stakeholder's views and that is what this paper is about.

2 Design Languages

In every design process there is some way that the design is expressed in order to communicate and explore it. This we will refer to as the "design language". In every discipline there is a design language, we also use them in everyday life, most often without being aware of it. For instance in architecture it is common to produce sketches and blueprints. In filmmaking synopsis and storyboards are way of communicating ideas. Many approaches to Information Systems (IS) development make use of requirement specifications and data models. In management many different models are used, for instance Porters competition model, the product lifecycle model with Cash Cows, Stars, dogs and Question Marks, the process (re)engineering models, etc

2.1.1 Co-Design Languages should require little or no training

Co-Design encourages stakeholders to engage in the design process. If the engagement are going to be useful there needs to be some common language for the stakeholder to think about the design as well as to present and discuss ideas. This active involvement aims at a stronger contribution than merely reading and accepting requirements statements, something that often takes place without much thought to how those statements will be interpreted by people with different backgrounds. While many design professions have well established design languages, most of these require training and experience to be useful. The Co-Design process however prefers to have as few prerequisites on the participating stakeholders as possible, not to limit the participation to people of certain vocations. Co-design metaphors and scenarios are two elements in a design language that meet these requirements.

2.2 Co-design metaphor

When thinking about something new, one needs a way to talk about something that doesn't exist. Metaphors are important when talking about new things in terms of older things, which makes them popular among IS developers, poets and other people struggling with innovation. Let us give an example from the Avanti project, a projecting aiming at creating eGovernment services for people who have no experience of using computers. [15]

¹ Authors Notes

When the elderly in Kista was asked for the first time what kind of computer support they wanted, their answer was to a high degree empty eyes- and a comment: “we are too old for this”. They couldn’t think of any good ideas what Computer support was all about.

In another session we said – do not think of computers, instead regard this as your assistant that almost can do whatever you want the assistant to do. Then you could see some light in the eyes of the elderly. One of the services they wanted the most was getting help from the assistant to connect with other elderly with similar interests – and with glimmering eyes they asked -“is this really possible?” After that they did continue to come up with other things they could use the assistant for.

In this example we can see that the initial Design Metaphor “computer support” was rather useless. The Elderly had no experience with computers and hence no model or expectations. In the second session the “assistant” metaphor is introduced and proven more successful. The Elderly had experience of assistants, it gave them a model and they could start to have ideas for such. The Design Metaphor gives them a language to think and talk about “computers”, something they had no experience of. This is a basic power of the metaphor.

Another example of the power of the metaphor did show up in another stage of the project. The first prototype we built in the Avanti Project used a photorealistic humanoid character as an avatar. One observation during the test was that the elderly did “tell” elaborate stories of their problems to the avatar which frequently did not understand them. This would make them angry. “Stupid... eh...machine?” Later we replaced the humanoid avatar with a cartoon dog. The elderly were much briefer with the dog character, putting short, precise sentences; like they would talk to a child. This of course increased the probability that the avatar did “understand” them. When it didn’t they would laugh rather than be angry. “Ha ha, of course, it’s just a dog.”

The humanoid metaphor or the computer as a humanoid did not work so well. The cartoon dog carried far less expectations and since many people talk in very simplistic manners with dogs, the computer as a dog metaphor did perform better in this case.

The case also gives us a hint on connections between the metaphor and the design. The dog metaphor resulted in a Co-Designed Dog avatar with abilities to help and guide the elderly. Less obvious is also that the metaphor also stimulates the design of the design process with its tools and methods. In fact the main part of the Avanti project was about the development of a software design environment for electronic assistants. This work mainly performed by Fujitsu and Microsoft led to one possible design environment for electronic assistants now used in many other projects. In more general terms is it easy to imagine a design environment for Dog avatars that allows you to select types of dogs and added to those possible actions and services that the dog can perform. As another step we can imagine marketplaces or communities for exchange of dog avatar actions and services. In this direction of research we are working together with Microsoft in rather exiting projects that we leave to another report.

2.2.1 Design Metaphors are IMPORTANT in all kinds of design

We may think that metaphors are more important in IS design than in for example architecture because the “IS” is a far more abstract matter than houses. That may be the case, but to us the metaphor is there to support the thinking about something that does not exist. And it’s equally difficult to think about a non-existent building as it is to think about a non-existent IS. At the same time one imagines a house it does begin to exist in the mind and is pretty hard to un-think.

A “house” can be regarded as a Design Metaphor in architecture. It supplies a model and invokes assumptions. Indeed, many architects struggle to get rid of the assumptions carried with “house” and may try other metaphors, like “space”, “box” or “personal sphere”. For instance many architects designing galleries or art museums struggle to get away from the

assumption that these building should contain “white cubes” as exhibition places. All these metaphors supply different models and carry different assumptions. It’s for example common for architects to challenge the idea assumptions carried in metaphor “room”.

So even if we don’t think of “house” as a metaphor, because we are so used to house being concrete, in the design situation it becomes a metaphor, with its model and assumptions. Design Metaphors are not limited to IS, which is a useful quality, enabling IS to be designed in a context.

2.2.2 The Design Metaphor war IS on

Today many IS developers for instance use the “database” as a Design Metaphor, in the same way as many would talk about a “house”. It is also easy to see that implemented metaphors have advantages and limitations. The good part is that people feel at home and know what to expect from the IS. The limitation part is that all implemented metaphors have limitations. For instance in early days of web design, graphic designers would use “brochure” and “page” metaphors based on their experience from design of paper artefacts. IT people would use “database” and “program” metaphors based on their experience with PCs. Last but not least “management” people would use “marketing campaign” and “store” metaphors based on traditional business. In a way this situation, when different metaphors are put forward as the best solution, can be described as a war between Metaphor camps. [16]

This war can easily be observed in every company today: Databases are competing with web-pages and decision support, with Intranet, and so on, as the “best” view of different IS. We are, at this point, participating in the war by advocating the electronic assistant as one powerful design metaphor basically because it has the power to include important stakeholders in a conversation about a possible future. Another strong side of the electronic assistant metaphor is that it has few limitations other than it has to act as part of an electronic system. It can inform elderly, but it can also help dial telephone numbers and watch after the elderly during the night. Finally, and may be most importantly, an electronic assistant is assistant to someone or some ones with their special interests. The assistant is in itself a result of a co-design process where different interests and related perspectives have been implemented. Following this argument we could say that a co-designed electronic assistant should explicitly be able to tell which interest were thought of when it was designed. In Estonia it has, for example, been suggested that e-voting assistant systems should have an inbuilt documentation on stakeholder interest as a security feature. We could also summarise this argument by saying that an electronic assistant always acts on Co-Designed perspectives in line with basic Co-Design ideas.

2.3 The Co-Design Scenario

Since Co-Design is about making people, interests and perspectives positive, constructive forces in the design process, it’s vital to explore the stakeholder’s perspectives in the design. This should be done in a way that allows a rich exploration, not just functional or economic aspects but also aesthetical, political and ethical aspects. The exploration should also facilitate discussions and contributions. In this sense the metaphor is limited in its power. The house metaphor or the electronic assistant metaphor just gives the first ideas of what could be. When we develop the metaphor into more detailed episodes of what we are doing with for example an electronic assistant we’ve found it most useful to work with Co-Design Scenario. [17]

Co-Design Scenario is a first person story about a client of the organisation and his/her experience. The Co-Design Scenario should include the overall relationship between the design and the stakeholders, as well as the situation where the particular artefact will be used. The Co-Design Scenario should therefore include other interactions, services and staff that

might be part of organisation's response to the clients matter. The wide scope of the Co-Design Scenario is needed for other stakeholder's perspectives to be considered, visualised and explored in the design process.

It's fair to say that all people are familiar with stories, so the Co-Design Scenario as a "language" does not exclude anyone per se. To maintain this quality the Co-Design Scenarios have to be written in a plain language and in a way that makes the interesting and illustrative. It's often useful to create one or more personas, fictive but typical clients, for the Co-Design Scenario. (That is; if personas weren't created already at beginning of the design, when stakeholders are chosen.) The richer the personae are the more aspects of the design can be explored.

The Co-Design Scenario can also live in flexible design process. It allows one to design the whole before the parts. The first version doesn't have give any details or parts of the IS or whatever is designed. It can just describe a desired series of desired actions. The design can evolve as the parts are expanded, while still maintaining an overall idea.

2.3.1 The Avanti case – combining metaphor and scenario

We have argued that an electronic assistant metaphor can have some advantages in that it does not limit the thinking in a way that for example a database metaphor does, it also introduces a design language possible to use for many groups of people. In one part of the Avanti project we developed a prototype "exemplar" using the electronic assistant as a Co-Design metaphor, but we also used the Co-Design scenario technique to arrive to the final design of the electronic assistant. One of the conclusions from the "exemplar" project also reported from all other test sites in the Avanti project was that the use of an electronic assistant as a co-design metaphor was that it forced the designers to put themselves into all aspects of the live and thinking of the clients. The electronic assistant metaphor really stimulated creation of Co-Design scenarios and a client driven design thinking. From earlier projects we have noticed that some designers have much easier than others to "live the life" of the clients and create Co-Design scenarios. It seems like the use of an electronic assistant metaphor supports or almost forces the designers to "live the life" of the clients. These findings were regarded as one of the most important results in the Avanti project as a whole. [18]

2.3.2 Dramatising Co-Design Scenarios

According to the basic Co-Design ideas we refer to there is a need to give feed back to the stakeholders and keep involvement in the design process. One effective way to do this in our experience is to use Co-Design Scenarios dramatised into stories, movies, interactive multimedia etc. In our earlier projects we have produced many such scenarios in the form of multimedia movies. These would typically illustrate one or more client persona, the actual situation the client would be in and how the interaction with IS and the IS provider would be. Also in the Avanti project we produced an exemplar multimedia movie. This movie was also one of the key results of the project and it was used both as a feedback to stakeholders but may be as important it has been used as inspiration for new projects with electronic assistants all over the world, which has placed the Avanti project as one of the most successful EU fifth framework IST research projects.

Even if the Co-Design scenario in this theory is viewed as design language to involve different stakeholders, we have also found that a dramatised Co-design scenario can be an effective management tool in the transformation of an organisation. That goes for both private business and in public organisations. This result can also be related to the new research on effectiveness of storytelling management.

3 Conclusion

In the Co-Design perspective is born in the same “mud” as the LAP perspective and for both the involvement of stakeholders are of great importance in the design process. The awareness of Design Metaphors, the active selection and creation of them can work as important support for such involvement. We should choose those that are useful to as many of the stakeholders as possible. We can not rely on the luxury of poets to just address an audience who favours her/his language. Since we start out by selecting a group of stakeholder, we must choose the languages in accordance with them. In the cases described here we have used service for elderly and the electronic assistant Metaphor. The choice of Design Metaphors is critical on all levels, the first being the objective of the design process. Just think of the difference between “this project should design a new website” and “this project should design a new service”.

Expanding the Metaphor into Co-Design Scenarios can serve as a forceful design language for communication between stakeholders. They allow the designer to work on the whole before the parts, equally allowing parts to be explored in details. When dramatised they can be used to engage large numbers of stakeholders in a cost effective way to simulate both IS and its use and consequences. They can serve as an important driver for transformation of an organisation or a company.

References

- [1] Glass, Robert. (1998.) *Software Runaways*. Prentice-Hall. New Jersey.
- [2] Stolterman, J. L. a. E. (2004). *Thoughtful interaction design: a design perspective on information technology*, Cambridge, Mass.: MIT Press.
- [3] Löwgren, J. & Stolterman, E. (1998). *Design av informationsteknik - materialet utan egenskaper*. Studentlitteratur. Lund.
- [4] Nurminen, M. (1987). Different Perspectives: What are They and How can They be Used? In P. Docherty & K. Fuchs-Kittowski & P. Kolm & L. Mathiassen (Eds.), *Systems Design for Human Development and Productivity: Participation and Beyond*. Amsterdam: North-Holland.
- [5] Winograd, Terry (1987), "A language/action perspective on the design of cooperative work," *Human-Computer Interaction* 3:1 (1987-88), 3-30
- [6] Winograd, Terry and Fernando Flores, *Understanding Computers and Cognition: A New Foundation for Design*, (220 pp.) Norwood, NJ: Ablex, 1986. Paperback issued by Addison-Wesley, 1987.
- [7] Flores, C. F. (1981). *Management and communication in the office of the future*. Unpublished doctoral dissertation, University of California at Berkeley.
- [8] von Foerster, H. (1981). *Observing systems*. Seaside, CA: Intersystems.
- [9] Maturana, H., & Varela, F. (1980). *Autopoiesis and cognition: The realization of the living*. Dordrecht: Reidel.
- [10] Forsgren O., (1986) Co-Constructive Cybernetics, Invited talk at: *American Society for Cybernetics, Special European Conference on Cybernetics, St Gallen*
- [11] Churchman, C. W. (1971). *The design of inquiring systems: Basic principles of systems and organization*. New York: Basic Books.

- [12] Forsgren O., (1988) *Samskapande Datortillämpningar: En systemteoretisk ansats för lösning av vissa förändringsproblem vid administrativ datoranvändning*, (Doctoral thesis), Umea university, Sweden.
- [13] Goldkuhl G., (2005) *Socio-Instrumental Pragmatism: A Theoretical Synthesis for Pragmatic Conceptualisation in Information Systems*, in *Proceedings of the 3rd Intl Conf on Action in Language, Organisations and Information Systems (ALOIS)*, University of Limerick
- [14] Regoczei, S. (1987). Understanding computers and cognition: A new foundation for design. (Review) *Computational Linguistics*, 13(3-4), 340-344.
- [15] Moussalli, A., & fl, m. (2001). *Avanti - IST-2000-28585*.: EU - IST 5 RP
- [16] Grönlund, Å., Forsgren, O., Albinsson, L., & a.o. (2000). *Managing Electronic Services*. London: Springer
- [17] Albinsson, L., & Forsgren, O. (2004). *Who's at the wheel of user driven projects when user can't drive?* Presented at the 3:rd International Conference on Systems Thinking in Management, University of Pennsylvania.
- [18] Forsgren, O., a.o (2003). *Final report - Lessons Learned*. London: Avanti - 5th framework - European research project.