Computational alternatives in participatory design - Putting the T back in Socio-Technical research

Henrik Korsgaard
Department of Computer Science, Aarhus University
Aarhus, Denmark
korsgaard@cs.au.dk

Clemens Nylandsted Klokmose
School of Communication and Culture, Aarhus University
Aarhus, Denmark
clemens@cavi.au.dk

Susanne Bødker
Department of Computer Science, Aarhus University
Aarhus, Denmark
bodker@cs.au.dk

ABSTRACT

This paper takes its starting point in a concern that Participatory Design (PD) and PD research have lost interest in innovating and reshaping technologies. We examine decades of projects and the current state of affairs and propose computational alternatives as a means of questioning the state of affairs and reintroducing a technical research interest into PD. Computational alternatives are used to systematically question the technological status quo and peak into a possible future; they are material manifestations of our focus and curiosity and can aid us in inquiring into possible socio-technical alternatives. Ultimately we focus on whether (and how) it is possible to maintain a technological research agenda in participatory and user-centered design, without giving up on pursuit of strong conceptual and theoretical insights.

CCS Concepts

•Human-centered computing → HCI theory, concepts and models;

Keywords

Computational alternatives; prototypes; participatory design

1. INTRODUCTION

In the summary of the second UTOPIA report, the authors describe the project as *"[...] both a development project for technology and a sociological experiment in understanding the conditions relating to that development."* [31, p.5]. This socio-technical agenda took the form of a sociological criticism of technology, in particular how it was introduced into the workplace, and based on the criticism the project builds new disciplinary understandings of the development of technology *and* novel alternative systems. The report also lists the envisioned achievements of the project, and the first is *"The development of alternative systems"*. Although there were a strong emphasis on the development of new systems, the contributions from early PD projects that have been picked up by the community focus on the early stages of the design process; involving users and techniques to support this [10, 20, 49], and the (political) critique of and subsequent interventionist approach in development processes and adoption of technology [3].

We can only speculate about the reasons: The publications from the work emphasize the process and participatory centric focus [24, 10] and resonated with contemporary movements in related areas (e.g. [2, 32, 27]). Early PD provided a space for increasingly multi-disciplinary research where a number of non-technical disciplines stepped in and made a home (from ethnography and sociology to e.g. architecture). The movement away from technology as the sole object of interest toward the social conditions of and role of users in the development of technology gave less technical disciplines an opening to contribute and investigate the impact of technology as it spread outside the workplace, a deliberate "branding" of the first PDC [21]. Finally, technology development itself made the "solutions" developed within the projects seem ephemeral to the increasingly less-technical PD community; the methodological (and ideological) contribution were simply more actionable across multiple disciplines, than technical implications and results.

As a result, it seems to us that the interest in doing technological research has largely disappeared from PD research, at an expense of increased focus on process and method. Whereas we have nothing against a design method focus, or for that matter a social science one or an activist attempt to give people a technology they want, we sadly miss a concern also for technology development, technological alternatives, etc. as it was found in earlier years of PD. With the proliferation of the web, mobile technologies, social media, surveillance, data mining, machine learning etc., developing technologies that challenge and expand on existing use and conceptions of technology are more important than ever. This holds both for using technology research as a way of practising a constructive criticism of contemporary computational technology and use, and building upon and from the theoretical achievements of the field. To put it more bluntly, when focusing on

PDC '16, August 15 - 19, 2016, Aarhus, Denmark

ISBN 978-1-4503-4046-5/16/08…$15.00

DOI: http://dx.doi.org/10.1145/2940299.2940314

process, community building, workshops, participation as a goal in itself, and even "feel good processes" [1], we not only miss out on an opportunity for examining the implications for systems design in detail, we make the theoretical contributions less relevant by not being able to show how our research findings might have an impact on technology design. Our argument is, that in order to do so we need to re-introduce the technology research concern into PD research.

The aim of this paper is doing exactly that, discussing how we might re-introduce a technological research concern into PD, or rather, regain the balance between the social and the technical that is a defining trait of PD, as it was defined in the early projects. We do this through a combined discussion of the historical concerns regarding technological research in relation to PD, the current state of research in PD and in suggesting computational alternatives as a prominent focus for PD research. A computational alternative is, in accordance with the use of alternative in UTOPIA, a (paraphrasing [31, 24]) technical and social design alternative that challenges existing socio-technical conceptions of technology, how it is designed, implemented and used to support practices.

2. SOCIO-TECHNICAL BALANCING ACT

A defining trait of Scandinavian PD in particular is the commitment to socio-technical alternatives. The first generation of projects focused on developing competencies within the workers' unions that enabled these to participate in assessing, negotiating and questioning management strategies for, introduction of computers within the workplace, and develop requirements for future systems. One of the earliest projects [44] were to analyse existing systems in use, the conditions for the workers and the possibilities for the union to influence company operations in relation to planning and control systems, and based on that, develop a system desiderata. In the project they changed strategy from describing the situation and produce proposals, to initiating a process within the Metal Workers Union (MWU) to gather experiences and prepare future action. The MWU project inspired several following projects, DEMOS in Sweden, DUE in Denmark and later a collaboration in UTOPIA. One of the technical-oriented outcomes of the MWU project was the DELTA language [34]. DELTA was designed to support communication between system analysts, people influenced by the system, trade union members, computer programming experts and people working in interdisciplinary teams. DELTA is closely tied to Simula, the first object-oriented programming language [22], the development of which Nygaard participated in. Central to this tradition was the idea (that dated back from Simula's roots in operation systems research) that modelling of human activity had to be a central component in building better systems. However, many experiments with user-centered/driven systems descriptions proved that there is no easy technical outcome of such processes. Simply because these systems descriptions were carried out in a formalism that resembles programming language does not necessarily lead to a good implementation [42].

In UTOPIA, the researchers shifted toward a more offensive and design-oriented strategy. The development of alternative systems came first in the list of objectives, with training and education, and union initiatives as the second and third [31, 24]. The UTOPIA project focused on how the introduction of computers in newspaper production changed the conditions for typographers. Layout computers in the newsroom meant that journalist and editors slowly took over work from the typographers. Researchers from the Swedish Center for Working Life, the Technical University in Stockholm, and from Aarhus University collaborated with typographers in formulating ways in which computers could enhance their skill and quality of newspaper printing. The project outlined *"technical and organisational design alternatives"* that would allow *"a peaceful coexistence between typographers and journalists"* [24, p.171]. The researchers developed mock-up and prototyping techniques that allowed the participants to explore possible future designs and practices. In many ways UTOPIA became famous for its methodological contributions and use of prototypes and workshop formats as the focal point for collaboration among the participants, and to some degree the theoretical contributions, which count both theories on participation in system design and concrete insight on the relationship between work, artifacts and interfaces. Despite developing a system that was marketable and envisioned as a concrete alternative for unions to point to, it is the methods and techniques, and the strong offensive (critical) approach that stands as the primary contribution when reviewing UTOPIA's later influence.

In the Florence project [6, 7], the researchers had two goals: Developing and testing techniques for user participation in systems development, and building a computer system to support the daily work of the nurses. A core concept was the 'application perspective', a perspective that emphasised that computers should be understood in the use context and its value would be demonstrated in use. The basis for the project was that the workers should control the development and use of computers in their work, and that computer systems should be based on the professional language and skill of the users, in this case nurses. Through an approach emphasising mutual learning, the computer scientists were to learn about the practice and daily work of nursing in specific wards, and teach the nurses about different kind of computer technologies. The outcome was two prototypes and a pilot system called the "Work Paper System". The nurses made the specification and the researchers did the implementation. As a result of this approach, the implementation became a technical challenge and the suggestions by the nurses had *"some heavy technical implications"* [6, p.261]. The primary contribution to PD was the idea of mutual learning and collaborative prototyping [17], and the application perspective and its insisting on the importance of professional knowledge and the dependency of the organisational and physical design of the use context. The developed system was in use some time after the project ended, but from the researchers perspective, the system (as a product) was later described as a side effect of the project and *"it was necessary to develop a computer system in order to create a setting of cooperation with the nurses"* [7, p.167].

The Great Belt project aimed at developing more generic CSCW applications supporting large scale project groups. Although having a very technology-oriented goal, the project build upon previous findings and techniques for understanding the context and involving users. In the initial phases of the project the researchers initiated a long range of activities involving participants from the organisation and future users. Through interviews and multiple workshops [30, figure 82] the participants and researchers explored the potential issues related to collaborating and managing heterogeneous (information) material in the large project teams. Based on the work, the researchers implemented three demonstrator prototypes including a hypertext prototype. This was developed through several iterations of meetings and feedback with the users, leading to the project's contribution to the Dexter model of hypermedia [28]. The prototype was deployed in a three month pilot-test and saw some continued use within the organisation afterwards. The work contributed with insight into the development of hypermedia systems within a larger organisational context [30] and detailed insights on hypermedia and hypertext concepts [28] in addition to work on augmented paper [39] and telemedicine.

We chose a few of the early examples of PD projects from Scandinavia that exemplify how early PD projects consciously balanced a socio-technical approach and outcome. Reviewing the early PDC proceedings [43, 40], it seems that these examples are unique, with rare exceptions, e.g. Trigg's work in the Workplace project [30]. They illustrate and share qualities of a socio-technical research perspective we take inspiration from and find missing in current PD research. First, all the projects made an effort in presenting socio-technical agenda as part of the research focus in the initial project descriptions. In one end of the spectrum the MWU project wanted to examine the conditions governing the process of adoption of technology in the workplace and develop what could be characterised as specifications for an ideal system, and in the other end, the Great Belt project had an explicit technological focus but did employ and contribute to experiences and the understanding of PD processes from the earliest projects. Second, all the projects made contributions to our conceptual and theoretical understanding of PD processes and techniques for involving and co-designing with future users, and with concrete technical systems and knowledge. In some cases (e.g. Florence) the technical systems "remained" within the context, while in UTOPIA and Great Belt the findings were more widely reported. Third, with the exception of NWU, each project deployed systems of a fidelity that allowed interaction, pilot studies, technical experiments and analysis beyond simple prototypes.

Although the UTOPIA project might be most known for the "cardboard computers" [24] to some, the researchers did develop prototypes of a high enough quality to examine new ideas related to graphical user interfaces and interaction, hence positioning themselves in upcoming areas of image processing and human-computer interaction [47, 23]. Before the era of freezing of raster-graphical displays and desktop computers into icons, windows and menus, the UTOPIA project explored a model that would provide the best possible quality of text and images on the display, while providing tools for graphical users to utilize their professional skills, using lenses in addition to a wide selection of custom-designed, alternative mice/pointing devices.

3. THE MISSING TECHNOLOGY FOCUS

Some attention has over the recent years been given to the integration of PD with those of software development methods such as agile development: Whittle reviews five PD projects and concludes *"Curiously, whilst there has been some research on adopting PD practices and principles within software development, there has been little consideration of incorporating agile methods into PD."* [49, p.129]. Together with others he focuses on software method integration rather than development of innovative technological systems, tools and platforms as such. Mogensen & Wollsen [41] however, work to expand PD processes beyond early analyses and methodological concerns. Pilemalm & Timpka in their analyses of generations of PD projects within health point out that *"An initial focus on needs analysis and requirements leads to technologies remaining abstract in the PD process."* [46, p.332]. Prototypes may improve on this situation, there are no recent examples (or very few) asking if these prototypes raise technological research challenges beyond software development methods. In reviewing the PD literature from 2002 to 2009, Halskov & Brodersen [33] identify 9 out of 101 publications with focusing on technology, indicating a trend in the community similar to [49, 46]. Balka argues that *"within the PD community we have gotten so focused on processes of participation, that we have forgotten about project outcomes."* [1, p.78]. To which Whittle adds: *"The charge to the PD community is that participation has become "a goal in itself" and has led to an obsession with methodologies for engendering participation and a willingness to see success in terms of "feel good processes" rather than any long-term, sustained outcome."* [49, p.121]

The field of Human-Computer Interaction has in the last two decades seen a decline in research contributions based on interactive systems development and architecture. Olsen argues that *"[t]here are three reasons for this decline in new systems ideas. The first is that, unlike those early days, there are essentially three stable platforms (Windows, Mac, Linux) upon which virtually all software is built and those platforms have dictated the user interface architecture. This is in contrast to the state of UI research 15 years ago when there were many competing toolkits and platforms. The second is that the stability of these platforms has lead to a new generation of researchers who lack skills in toolkit or windowing system architecture and design. The third reason is the lack of appropriate criteria for evaluating systems architectures."* [45, p.251] We believe a similar analysis is in place for the reason of a decline in PD projects with strong technological contributions. The stability of platforms (now also including iOS and Android) together with software that has matured for decades (Microsoft Word (33 years), Microsoft Excel (30 years), Adobe Photoshop (25 years), MatLab (32 years) to name a few) has led to an entrenchment of software practices and a conservatism both on the behalf of software developers, designers and end-users but also in the training of researchers and practitioners in our field. This means that it is easier to build upon available platforms and applications, than to critically rethink whether the entrenched practices are suitable or just taken for granted. Although there are arguments for basing solutions on existing frameworks within the user domain (integration, sustainability, familiarity, licensing etc.), this development comes with implications that are important in PD research. If technologies are chosen based on the researchers' (and users') taken-for-grantedness, familiarity and/or convenience, and later result in recommendations for, or, a finished system, it must be implicitly assumed that our current technologies are adequate for local practices. For this reason alone, we insist that alternatives are needed. Alternatives help both users and designers imagine beyond the taken-for-granted. Uncritically adoption may make researchers and user insensitive toward the ideological premise embedded within the (commercial) platforms discussed above. To us, this stands in opposition to the original ethos of Scandinavian PD, where the local knowledge of professionals are the focal point and the importance of questioning the conditions under which technologies is developed and introduced.

4. COMPUTATIONAL ALTERNATIVES

Now we turn to the notion of computational alternatives as our approach to incorporate and start thinking systematically about the role of computational artifacts in PD research projects. At a first glance, the arguments presented in this paper could be seen as a technology-driven. This is far from the case. Rather, we find the balancing act between understanding the conditions where under technology is produced and used, its relationship with practice and the passion for exploring socio-technical alternatives present in the origins of PD an ideal position. But reviving these positions requires, for the present, investigating the role computational artifacts and novel technology may play in PD research. The research we do and our position are strongly embedded in, and shaped by, traditional Scandinavian PD, as we have discussed. To us, PD is not a 'toolbox', a collection of design techniques or a matter of choice; it is the modus operandi, a tradition in the strongest sense. This is why we are concerned with the early PD projects, which considered technological alternatives as part of PD research. In addition we believe that it is actually from within the tradition itself that we get the help needed to understand the role of technological research in specific processes and projects.

In continuation of [37, 29, 30, 9], we see prototypes as computational alternatives in our research practice, developed iteratively in specific cases and more generically beyond that [29, 30]. When a prototype serves as a computational alternative it raises questions, and makes us see what *is* in a new light. A computational alternative is not designed to showcase a new technical solution to a well-known problem, but to elucidate problems in the otherwise taken for granted. Wartofsky [48] refers to artifacts with such qualities as 'tertiary' artifacts; artifacts that make us see possible worlds alternative to the actual world. These 'worlds' are simultaneously connected to and inseparable from the artifact and its use, and the practice they are embedded in throughout a research project. Computational alternatives are concrete technology, and a concrete practice. They are not new technology detached from a social practice, nor a social experiment detached from critical technological development. To understand this further we will now examine computational alternatives as a prototype, practice and mediator.

4.1 Computational alternatives as a prototype

Computational alternatives are prototypes in the simplest form; they are the first of their kind and an attempt for explore and formulate an alternative to an existing product, system and/or activity. Just as prototypes are *"manifestations of design ideas that concretize and externalize conceptual ideas."* [38, p.5], computational alternatives are manifestations of alternative ideas on how technology is currently used, like we have seen from many of the historical cases. They question what is otherwise taken for granted, or demonstrate what can be made possible with technology from a perspective of use. This includes questioning how the conceptual models are translated into a system through design choices, and exploring how both existing models and novel alternatives can act, in generative and exploratory ways, as a *"catalyst to elicit good ideas and promote a creative co-operation"* [26, p.6]. This is a familiar perspective on prototyping in PD. Kyng [37] discusses prototypes (and mock-ups, scenarios and other representations in the design process) as representations of the system being designed *and* representations of the the future use. In discussing the difference between low-tech prototypes, he points out that while these low-tech tools and techniques allow users to take on an active design role, the final system will be implemented in some form of computational system. Thus, it is necessary to be able to manifest the ideas of alternative use and alternative computational design in actual computer systems throughout the process. Lim et al. [38] provide a framework for understanding the needed fidelity of the prototype and what components of the prototype needs to be developed in order to examine the qualities and ideas in which the designers are interested. In their framework they focus on prototyping as framing and exploring a design space, where the purpose is not to identify or satisfy requirements but finding manifestations that in their simplest form filter the qualities in which designers are interested, without losing focus on the understanding of the whole. They are for traversing a design space, *"leading to the creation of meaningful knowledge about the final design as envisioned in the process of design"* and they *"are purposefully formed manifestations of design ideas"* [38, p.3]. They emphasize the economic principle of prototyping whereby *"the best prototype is one that, in the simplest and most efficient way, makes the possibilities and limitations of a design idea visible and measurable. If we keep the economic principle of prototyping in mind, determining the values of the manifestation dimensions – that is, the materials, resolution, and scope of the prototype – can be approached in a rational and systematic way."* [38, p.3]. With this in mind, the fidelity of a computational alternative is filtered by what we want to investigate and what it should convey from a research perspective. Not only must a computational alternative have a high enough fidelity to establish a credible practice in order for users to be able to experience and assess the proposed (work) practice represented by the prototype, the level of fidelity should also make it possible to assess the value in the alternative computational aspects of the proposed design. The socio-technical research agendas we describe here, may require that we have underlying systems in place that hold more in common with a finished product than a traditional prototype.

4.2 Computational alternatives as practice

As discussed above, prototypes both represent concrete design ideas related to the form and function of a particular (future) system and, in more subtle ways, its future use and ideas about the practice wherein it will be inserted. They represent a specific understanding of the existing practice and possible future changes. In representing possible futures to participants and researchers, computational alternatives serve as Engeström's springboards: *"A springboard is a facilitative image, technique or socio-conversational constellation [...] misplaced or transplanted from some previous context into a new [...]"* [25, p.287]. Springboards do not come about smoothly or automatically, and they are not as such solutions to a problem that one is facing. They are starters that may lead to an expansive solution. Bødker & Christiansen [9] use scenarios as means of making hypotheses or qualified guesses about the future computer application, as embodiments of it. While they consider scenarios as the backbone of design, they also see them as closely interlinked with prototypes that facilitate this embodiment. Whereas much has been said about social and psychological expansion in relation to design and prototyping [5, 8, 25] the focus on technological expansion has been considered much less. The notion of springboards and the idea of facilitative images, transplanted into new contexts, however, allow for thinking about building technologies not only to replace existing ones, but also to take a known technological idea from one context and explore it in a new, possibly without the concern that it should or could ultimately provide the final solution to a socio-technical challenge in the new context. Nonetheless, the fundamental challenges of understanding and developing computational alternatives could usefully be understood as part of such expansion, and hence as springboards in research as well as in design (which is the role in which they have been considered so far).

A computational alternative establishes a microcosm, which Engeström in his work on expansive learning defines as *"[...] social test bench and a spearhead of the coming culturally more advanced form of the activity system. The conscious formation of a microcosm as a sub-step of expansive research corresponds to the formation of a vehicle for transition from cooperation to reflective communication. In other words, the microcosm is supposed to reach within itself and propagate outwards reflective communication while at the same time expanding and therefore eventually dissolving into the whole community of the activity."* [25, ch.5]. The microcosm allows a community of potentially diverse stakeholders to peek into an alternative future, and importantly for us researchers, to study this potential, alternative future, its socio-technical tensions and possible resistances towards it.

4.3 Computational alternatives as a mediator

Computational alternatives become instruments mediating use [4, 17] as well as design and research, and in this mediation lie both facilitation and resistance or backtalk. Backtalk is a double loop where the technology talks back in the use situation and then in the research process. But not only that: Computational alternatives talk directly back to research, through the technological challenges that are to be addressed in order to develop a somewhat final and self-sustained prototype that may work in the use situation. One may say that it is in the meeting and confrontation between the double-loop and the direct mediation that the interesting happens for the kind of research that we address here.

Béguin talks about how various forms of mediation punctuate mutual learning in a design process: *"Semiotic mediation occurs when a symbolic language is used to generate graphic descriptions such as maps and diagrams. But mediation also comes in other forms, such as scale models, mock-up, prototypes, etc. [...] Let us call these productions 'intermediaries' insofar as they link the individual and collective dimensions of design."* [4, p.713]. Béguin primarily discusses prototypes and technology probes as design intermediaries, yet we argue that they are also research intermediaries because, in the way he describes it, the researchers are also designers, who set out to build technologies that are instruments for the researcher, albeit driven by a different type of ideas, or rather questioning those. With the notion of 'punctuating' mutual learning he uses a term that on the one hand talks about disrupting mutual leaning, on the other about bracketing and closing something with the purpose of mutual learning. Bødker states: *"Thus, I propose another dilemma: Design representations must be sketchy and incomplete to be used here and now (the napkin); yet to hold on to history and to be handed over, they need to be complete and rigid. To paraphrase Brown and Duguid [18], they need to (re-) create the context of design."* [8, p.118]. With this in mind we are concerned with prototypes that are in a state and quality that can create punctuation in both understandings of the term. They are intermediaries rather than versions of a final system, and help establish a microcosm. At the same time they are also prototypes that are open as to be redeveloped both technically and in relation to use. From the perspective of the concrete prototype, constructing a computational alternative may involve going beyond how we, in a design process, typically use low-fidelity prototypes, or even beyond high-fidelity prototypes, into prototypes that have a fidelity high enough and a scope that is large enough to establish a convincing microcosm for study. This does not mean that the computational alternatives are fully-fledged systems, rather that they are punctuating intermediaries.

In summary, computational alternatives are prototypes setting out to elucidate problems otherwise taken for granted, through concrete technical development. They are manifestations of research and design ideas as well as demonstrations of possible ways to move ahead. They help provide springboards to carry out technology-supported expansion of user practices. They are part of exploratory research processes, rather than versions of a future system. They are functional in particular microcosms, at the same time as they support the investigation of more general alternative futures. They provide backtalk and punctuation, and not least are they the simplest means of filtering and manifesting alternatives of a specific use setting.

5. CASES

5.1 Local Area Artworks

Local Area Artworks (LAA) [11, 12] was developed to study how information technology could support audience participation in interpreting and curating an art exhibition. LAA was part of ongoing research in how to apply existing technologies and infrastructure, i.e. personal devices and local area wireless networks, to support and enable participation at large. With LAA, a part of the usual curatorial activity of authoring interpretive descriptions for artworks was opened up for the visitors, artists, curators, staff, etc. to participate – effectively anyone physically present in the exhibition space. Hence, LAA made the existing interpretative role of the audience explicit and visible by enabling co-interpretation among audience members in the physical space. A central idea was to use people's personal devices as a means for participation, to create a sense of familiarity allude to visitors' existing skills and experiences with their devices.

LAA was developed in dialog with artists and staff at the venue, and through these dialogs an idea was formed about Wikipedia-inspired collaborative authoring in and about a local space. The staff of the art venue, furthermore, shared an interest with the researchers in anchoring a digital layer to the local space and in this project the digital layer consisted of the interpretive texts associated to the artworks. The installation was deployed and ran for the duration of a month-long Easter exhibition at Kunsthal Aarhus in Aarhus, Denmark and was connected to six selected artworks.

In LAA, the conventional curatorial descriptions of artworks were replaced by texts on digital panels collaboratively written and rewritten by visitors during the exhibition mediated by their own personal devices. Using WiFi proximity detection, the system detected when visitors were in close proximity of an artwork and redirected their web-browser on their personal device to the respective editable text [35]. Making use of personal devices can require significant bootstrapping on the side of the user in the form of downloading and installing apps. In LAA we wanted the barrier of participation as low as possible and required zero installation on the user's device. We hypothesized that contributions about local matters would flourish best when people write about what they immediately see and experience. This led us to a design requiring physical proximity of the user to an artwork in order to allow editing its associated text thereby strengthening the coupling between physical and digital layer. Therefore, LAA sought to make navigating between different artworks in the exhibition as 'automagic' as possible by basing it on the user's location in the gallery. Finally, the digital panels next to each artwork gave the digital activity a physical representation in the space.

The requirements of zero installation and 'automagic' proximity-based navigation posed significant technical challenges, as traditionally positioning-based systems require custom software installed as an application or app on the user devices. This challenge was overcome and the results were document as a technical research paper [35].

Bødker et al. [11] document the outcome of studying Local Area Artworks in use. The study was particularly focused on how visitors of the exhibition understood and appropriated the system, and what background experiences they used to orient themselves towards the system. We observations that when the traditionally curatorial practices were challenged through the computational alternative, it led to surprising metaphors for what people reported participating in. Some expressed that they were participating in a dialogue about the art through a stream of commentaries, while others that they participated in the artistic expression of the exhibition. We had applied a Wikipedia metaphor for the collaborative authoring on the interpretation panels in the design of the system, but this did not carry through to the visitors. The use of personal devices did provide familiarity in the interaction, and the 'automagic' navigation blended the physical and digital space together more or less seamlessly. Yet, we also observed how the panels shifted involvement and changed group dynamics from happening between people physically present together, to interaction with people who had been visiting before (or would visit in the future), and changed the pattern for how people would physically move about in the space.

5.2 City Bug Report

City Bug Report (CBR) was developed for the Media Architecture Biennale 2012 [36]. The project was collaboration between the Participatory IT centre, the city of Aarhus, Media Architecture Institute, and a local business intelligence company. The design process only lasted a few months and the design was developed at a two-day workshop involving researchers, designers, representatives from the municipality, the local open data project, the region and local companies with an interest in open data. In the project we developed two prototypes: A large media facade installation on the city hall tower of Aarhus showing an animation of four years of data on civic communication between the city departments and the citizens on a 5.500 LED display wrapped around the tower. The animation visualised incoming and outgoing communication filtered by case numbers and the visualisation was designed to illustrate how efficient the city departments responded to incoming request from the citizens. The other prototype was a web-application that allowed citizens to report issues they encountered within the city. The project borrowed the term 'bug' from software development, as a way of articulating and framing urban issues. When reporting a bug, the citizens could pick a predefined category reflecting city departments, add a description and possible solution. Once reported, the bug was added to a public list and citizens could share the issues on social media.

With this case, the municipality wanted to show their digital ambitions to the public, embrace new technologies and use civic data as a way of increasing transparency, as well as give access to and use these data to potentially change how the municipality works. From a research perspective we wanted to investigate three aspects of open data and civic participation. First, how open data and media architecture would challenge conceptions of transparency and use of civic data. Second, how open web-platforms would encourage citizens to report issues that are important to them and potentially change the way city operations identify and prioritise issues. Third, understand the process of moving public sector data from a municipal database, to an open data portal and onto a media facade and the socio-technical implications involved. The research produced three primary insights: First, getting access to and opening up data from municipal systems represent a socio-technical challenge. Not only is it difficult to give access to data deeply embedded within the municipal IT systems, the dataset in addition may contain information that, when made accessible outside the practice wherein it is normally used, exposes tacit work processes and sensitive information. Although the participants from the municipality assured us that access was a formality and that the dataset in question was already public and checked (on a field name level) for sensitive data, it was later discovered that as part of the existing internal use of the data set, caseworkers added sensitive data to free text fields. Second, at the workshop and in the initial phases, the representative from the city departments was enthusiastic regarding the potential in using citizens as a knowledge resource in identifying (and potentially solving) city issues. As the 'bug' reports started coming in, it became apparent to the participants that involving citizens in identifying issues came with a (legal) obligation to address the issues within a short time frame. This would short-cut the existing way the individual municipal departments prioritised maintenance and work, planned their budgets and their organisation. Inviting citizens to participate in city operations and integrating such a tool would require a major change on all organisational levels. Third, transparency works in both directions. As the project became more concrete and the actual data was shown on the media facade of the city hall tower, the participants slowly became more conscious on the potential implications of exposing the internal work processes on the highly visible outside of the building. One participant noted that the project created a sense exposure inside city hall.

CBR was the first local experiment involving citizens in identifying urban issues and the media facade was an installation developed specifically for the Media Architecture Biennale 2012. On a local level, both prototypes explored how transparency, open data and citizen participation might challenge how the municipality is organised, from an individual department level and up. The research outcomes partly inspired work on the role of urban design, participation and policy [19] and ongoing work on open data and implications related to working with data produces across contexts and practices. This would not have been possible without the fidelity of the final systems. In order to explore the taken-for-grantedness (easy access to data and citizens as a knowledge resource), it was necessary to have prototypes that would require access and allow citizens to report issues. In order to understand what transparency based on open data means, we need to make open data transparent.

5.3 CaseLine

The initial focus in CaseLine was to explore collaborative information sharing across boundaries between citizens and caseworkers using web based tools and infrastructure [13, 16]. The explicit focus on parental leave, applying for parental leave funding and the planning thereof, as it is a process that involves many potential stakeholders: Parents need to coordinate the leave plan between themselves, which in turn is affected by the parents' respective agreements with their employers. As the leave plan potentially spans over a period of nine years, the plan for one child and its parent may overlap or collide with the leave plans of other children and previous partners.

The design of the timeline tool, CaseLine, was based on insights from empirical studies of parents and municipal caseworkers and a PD process with parents and caseworkers. This design crosses the boundaries between leisure and work-life and CaseLine plays several roles on these boundaries: It is a shared planning and visualisation tool that may be used by parents and caseworkers alone or together, it serves as a contract and a sandbox, as a record and a plan, as inspiration for planning and an authoritative road, as a common information space and a fragmented exchange.

This required a different architecture than the municipal systems supports, a different way of incorporating the information already existing within the system, what is needed across stakeholders and what the individual actors provide to the system. This was reflected in how the architecture was developed and how the shared objects formed the basis for both the visualisation and the collaborative side of the prototype (see [14, fig. 1]). The design moved the thinking about the coordination of parental leave away from a series of adaptive documents [15] and records moving back and forth between the actors, to seeing it as a timeline visualisation incorporating more complex manipulation and more open, tangible plan [14]. Caseline led to a challenging discussion among the caseworkers regarding the loss of direct control over what information was given to parents. Among parents, too, parallel discussions regarding privacy and sharing over time, as well as to the possibilities of more generic sharing (e.g. on Facebook) of people's own parental plans.

CaseLine was the first of its kind in that the entire collaboration between stakeholders, not least the city officials and citizens was not mediated by technology before, if we exclude letters and telephones. The idea that one could share a plan on-line that would connect to all necessary documents, was also not described in literature, let alone the more technology-centered ideas of adaptive documents, collaboration over time, and timeline-based web-browsing. The prototypes developed were at times rough sketches leading to a more thorough high-fidelity prototype [13], prioritizing to build prototypes that were sufficing to show and users explore the ideas at various times. These prototypes served as springboards at several organizational levels in the municipal organization, both among the caseworkers, and vis-a-vis e.g. management and web-maintenance. The parents explored possibilities of the sandbox exploration among themselves, as well as notions of sharing with friends as well as with employers. Research wise this led to a new (current) focus on privacy and security.

6. ANALYSIS

Despite being functioning systems deployed and running for an extended period of time, none of computational alternatives presented above provided viable, sustainable solutions to concrete problems within the respective domains. Instead, each of them illuminated challenges both technically, organizationally and in use.

Local Area Artworks demonstrated that it was doable and relatively inexpensive to enable audience participation mediated by personal devices in an art exhibition. However, it also pointed out that the facilitation of *what* visitors should participate in and *why* does not come by it self. It would require the staff (and the artists) to take an active role in the dialogue with the participating visitors. Also, that any introduction of technology, even if it is done discreetly changes the dynamics of the praxis, in this case visiting and art exhibition, significantly for good and for ill.

CaseLine demonstrated an alternative to the traditional forms and spreadsheets inspired municipal interfaces for the public, pointing to a wider set of organizational matters in the municipal organization, as well as interesting concerns regarding sharing and privacy over time, within the community of new parents, as well as across the borders to employers, friends, family and government agencies.

CBR demonstrated that it was possible to use municipal data to create a sense of transparency by visualising civic communication on the city hall tower. It also demonstrated that citizens are willing to participate and contribute by reporting urban issues. The case also indicated that transparency also creates a sense of exposure and that accommodating day-to-day citizen participation requires rethinking municipal organisation and work processes.

Each of the above cases exemplifies the use of computational alternatives as a means for socio-technical research. Each prototype embodied both technical challenges and conceptual challenges within the domain. They all worked with both high level concepts and the necessary technical steps, decisions and designs that were required to concretize the underlying design and research ideas. They all represented a number of design hypothesis, open questions related to use and research hypotheses. In Local Area Artworks we hypothesized that we could stimulate a Wikipedia-inspired collaborative writing in a local space and that personal devices as mediators for participation would create a sense of familiarity. In CaseLine we wanted to explore collaboration and the notion of shared objects and plans rather than transactional interactions around records and information. In CBR we had series of questions relating to both the installation and the web platform. Some of these were very basic: Will the data visualisation be intelligible on such a low resolution display? And, will the citizens even use the bug reporting platform? Others were more intermediate and related to the kinds of issues the citizen would report and how the city department would handle these in the future, and finally we hypothesised that concept as a whole would provoke reactions from the involved stakeholders, institutions and the public around the central concepts explore in the project.

The three cases each warrented different levels of maturity and scope required for a prototype to establish a microcosm. CaseLine addressed activities that potentially spanned years, hence a self-sustained prototype was not feasible. Instead scenarios were played with high-fidelity interactive prototype with simulated data. This of course meant that the established microcosm was not representative of a complete alternate future, but instead hinted at what such future could bring. Similarly, even though the system deployed in Local Area Artworks was self-sustained and ran without the presence of researchers, the scope was limited in that the exhibition was temporary. However, in both cases the microcosm, exposed unforeseen tensions and resistances for the specific use situation as well as for the wider potential of the computational alternative. CBR, on the other hand had an extremely simplistic visualization of data on the tower of the city hall, so simplistic that it was more or less unintelligible by the passers-by on the street, but as a microcosm, it required substantial extra work getting this established with the specific dataset and the media facade in particular. Yet, the established microcosm of an alternative future where municipal data is exposed so prominently pushing internal conceptions of transparency, puncturing or stirred the municipal hornet's nest in a way we believe would be difficult to have achieved without making it real and concrete.

CaseLine and CBR both tapped into municipal systems and workflows, by changing how we use and represent data, in the interface as well as in the architecture. CaseLine moved more isolated bits and pieces of information (records) up into a shared information space, and CBR pushed data from within the depths of a municipal database into a visualisation displayed on the huge low-resolution media facade enclosing the city hall tower. Similarly, they both played with changing ownership and the right to define aspects of municipal case flow, either by providing citizens with a platform that potentially turns the process of identifying and prioritizing important city issues inside out, or in CaseLine by combining information from multiple sources and allow parents to experiment with and change the elements of the parental leave more fluent and continuously. Local Area Artwork played with renegotiating who can and should curate and produce the text describing the artworks in the space of an art exhibition. Again, ownership over parts of the institutional information space was delegated to the visitors in an attempt change how we participate and engage with an art exhibition. As with the other cases, this required both tailor-made infrastructure and/or architecture, utilization of (web) technology, and for Local Area Artwork, the development of a zero-install proximity system to make this new relationship between the viewer (now potentially writer) and the art works a spatial one on the technical side as well. As such, each case is a computational alternative, as they both attempted to explore and tackle socio-technical challenges, while focusing on the reciprocal relationship between both the concrete praxis and the technology.

LAA and CaseLine both demonstrated how the praxis of use encouraged critical development of technology. In LAA a technique was devised to allow participation using personal devices without requiring a lengthy app installation, while also allowing for WiFi proximity detection to simplify navigation and only allow the editing of texts when in close proximity of an artwork. In CaseLine the challenges of collaborating around planning over long time spans, challenged the traditional document centric model of the web, resulting in the development of novel timeline based interaction for the web.

Local Area Artwork also played with very mundane concepts partly introduced by how we think web technology and networks, questioning if the taken for granted global access is always an ideal or participation could be a matter of being situated or proximate. This renegotiation of rights based on infrastructure also prompted a negotiation of who writes the texts in the space of an art exhibition and created a tensions related to ownership and roles. Similarly, CBR and CaseLine created multiple tensions on ownership over public records, information flow and division of work. Who is the planner, when we move from documents to a collaborative timeline or delegate the right to define urban issues and matters of concern to citizens?

7. DISCUSSION

With the notion of computational alternatives we point to the need for and potential in reinvigorating a socio-technical research agenda within PD wherein technology development play a central role. When reviewing the technological contribution from the early PD projects, we see that strong PD research can contribute to the development of programming languages, models, graphical interface design, work-flow systems, hypermedia models etc. The same work give some indications to what is to be done if PD research have ambitions of making similar contributions. Common for the work is a) having the focus on socio-*technical alternatives* as part of the initial research agenda [31, 7], b) engaging in technology production as part of the collaboration with stakeholders [42, 24, 7], c) having the technical insight to identify, formulate and propose 'deep' technical implications and shortcomings in contemporary models [30, 22]. This does not imply abandoning understanding the process of technology development or the existing positions in PD, rather, to recognise the value in and necessity of developing computational systems as part of furthering those research perspectives as well. The researchers in the Florence project outline this relationship in a simple way: *"Knowledge of system development was the overall goal of the research project, more important than any products or computer systems. However, it was necessary to develop a computer system in order to create a setting of cooperation with the nurses."* [7, p.167].

We find the alternative, to *not* engage in technology development in PD research, problematic. As Kyng [37] rightly points out, the final system (or design and research insights from the PD process) will be implemented on a computer in some form or another, at least if we still claim that PD is an important position in designing computational artifacts. This means that computers should be a part of the process and that it is important to show how to move beyond early analysis and methodological concerns [41, 46]. Not questioning existing technologies, either in the process or by proposing computational alternatives, could be interpreted either as an instrumental position toward technology and/or as an insensitivity to the representations of work, collaboration, participation, sharing, community etc. already embedded contemporary technologies (from devices, over operating systems, to applications). Do contemporary technologies adequately represent work? Not having the above in mind or never moving beyond the early phases might also indicate that the outcome is insignificant and all is about the community work and feel good processes [1, 49]. Is it?

8. CONCLUSION

Through an examination of early PD research projects we show that Scandinavian PD is defined by how it balanced socio-technical alternatives. We have argued that a strong technical commitment has faded in PD, and propose computational alternatives as a perspective to return to and maintain a technology research agenda from within the PD tradition. This is based on recent discussions in PD and related fields on the lacking technology focus in PD research, and through analysis and discussion of three recent cases.

9. ACKNOWLEDGEMENTS

We would like to thank the reviewers for insightful comments throughout the review process. This work is funded by the interdisciplinary Participatory IT centre.

10. REFERENCES

[1] E. Balka. 2010. Broadening discussion about participatory design. *Scandinavian Journal of Information Systems* 22, 1 (2010), 77–84.

[2] L. Bannon. 1991. From human factors to human actors: The role of psychology and human-computer interaction studies in system design. *Design at work* (1991), 25–44.

[3] J. Bansler. 1989. Systems development research in Scandinavia: three theoretical schools. *Scandinavian Journal of Information Systems* 1, 1 (1989), 3–20.

[4] P. Béguin. 2003. Design as a mutual learning process between users and designers. *Interacting with computers* 15, 5 (2003), 709–730.

[5] O.W. Bertelsen. 2000. Design artefacts: towards a design-oriented epistemology. *Scandinavian Journal of Information Systems* 12, 1 (2000), 2.

[6] G. Bjerknes and T. Bratteteig. 1988a. Computers—utensils or epaulets? The application perspective revisited. *AI and SOCIETY* 2, 3 (1988), 258–266.

[7] G. Bjerknes and T. Bratteteig. 1988b. The memoirs of two survivors: or the evaluation of a computer system for cooperative work. In *Proc. Computer Supported Cooperative Work*. ACM, 167–177.

[8] S. Bødker. 1998. Understanding representation in design. *Human-Computer Interaction* 13, 2 (1998), 107–125.

[9] S. Bødker and E. Christiansen. 1997. Scenarios as springboards in CSCW design. *Social Science, Technical Systems, and Cooperative Work: Beyond the Great Divide* (1997), 217–234.

[10] S. Bødker, K. Grønbæk, and M. Kyng. 1993. Cooperative design: techniques and experiences from the Scandinavian scene. In *Participatory design. principles and practices*. Lawrence Erlbaum Associates.

[11] S. Bødker, C.N. Klokmose, M. Korn, and A.M. Polli. 2014. Participatory IT in Semi-public Spaces. In *Proc. NordiCHI*. ACM, New York, NY, USA, 765–774. DOI: http://dx.doi.org/10.1145/2639189.2639212

[12] S. Bødker and A.M. Polli. 2014. *Proc. COOP*. Springer International Publishing, Cham, Chapter Between Initial Familiarity and Future Use: A Case of Collocated Collaborative Writing, 137–154. DOI: http://dx.doi.org/10.1007/978-3-319-06498-7_9

[13] M. Bohøj, N. Gandrup Borchorst, N.O. Bouvin, S. Bødker, and P.O. Zander. 2010. Timeline Collaboration. In *Proc. Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 523–532. DOI: http://dx.doi.org/10.1145/1753326.1753404

[14] M. Bohøj and N.O. Bouvin. 2009. Collaborative Time-based Case Work. In *Proc. Hypertext*. ACM, New York, NY, USA, 141–146. DOI: http://dx.doi.org/10.1145/1557914.1557940

[15] M. Bohøj, N.O. Bouvin, and H. Gammelmark. 2012. A Framework for Interactively Helpful Web Forms. *J. Web Eng.* 11, 1 (March 2012), 1–22. http://dl.acm.org/citation.cfm?id=2230876.2230877

[16] N.G. Borchorst and S. Bødker. 2011. "You probably shouldn't give them too much information"–Supporting Citizen-Government Collaboration. In *Proc. European Conference on Computer Supported Cooperative Work*. Springer, 173–192.

[17] T. Bratteteig. 1997. Mutual learning: enabling cooperation in systems design. In *Proc. IRIS'97*, Vol. 20. 1–20.

[18] J.S. Brown and P. Duguid. 1994. Borderline issues: Social and material aspects of design. *Human–Computer Interaction* 9, 1 (1994), 3–36.

[19] M. Brynskov, J.C. Carvajal Bermúdez, Manu Fernández, Henrik Korsgaard, Ingrid Mulder, Katarzyna Piskorek, Lea Rekow, M Waal, and others. 2014. *Urban interaction design: Towards city making*. UrbanIxD/Booksprints.

[20] A. Clement and P. Van den Besselaar. 1993. A Retrospective Look at PD Projects. *Commun. ACM* 36, 6 (June 1993), 29–37. DOI: http://dx.doi.org/10.1145/153571.163264

[21] Paul Czyzewski, Jeff Johnson, and Eric Roberts. 1990. Introduction: Purpose of PDC-90. In *Proc. Conference on Participatory Design*, Vol. 90.

[22] O. Dahl. 2002. The roots of object orientation: the Simula language. In *Software pioneers*. Springer, 78–90.

[23] P. Ehn and M. Kyng. 1986. A tool perspective on design of interactive computer support for skilled workers. *DAIMI Report Series* 14, 190 (1986).

[24] P. Ehn and M. Kyng. 1991. Cardboard Computers: Mocking-it-up or Hands-on the Future. In *Design at work*. Lawrence Erlbaum Associates.

[25] Y. Engeström. 2001. Expansive learning at work: Toward an activity theoretical reconceptualization. *Journal of education and work* 14, 1 (2001), 133–156.

[26] C. Floyd. 1984. A systematic look at prototyping. In *Approaches to prototyping*. Springer, 1–18.

[27] K. Grønbæk, J. Grudin, S. Bødker, and L. Bannon. 1993. *Achieving cooperative system design: shifting from a product to a process focus*. Lawrence Erlbaum Associates, Hillsdale, NJ.

[28] K. Grønbæk, J.A. Hem, O.L. Madsen, and L. Sloth. 1994. Cooperative Hypermedia Systems: A Dexter-based Architecture. *Commun. ACM* 37, 2 (Feb. 1994), 64–74. DOI: http://dx.doi.org/10.1145/175235.175240

[29] K. Grønbæk, M. Kyng, and P. Mogensen. 1995. Cooperative Experimental System Development: Cooperative Techniques Beyond Initial Design and Analysis. In *Proc. Aarhus Conference Computers in Context: Joining Forces in Design*. 20–29.

[30] K. Grønbæk and R.H. Trigg. 1999. *From Web to workplace: designing open hypermedia systems*. MIT Press.

[31] UTOPIA Project Group. 1981. *Training, Technology, and Product from the Quality of Work Perspective*. Technical Report Utopia Report no 2. Swedish Center for Working Life, Stockholm, Sweden.

[32] J. Grudin. 1999. CSCW and groupware: Their history and trajectory. *Designing Communication and Collaboration Support Systems* 2 (1999).

[33] K. Halskov and N.B. Hansen. 2015. The diversity of participatory design research practice at PDC 2002–2012. *International Journal of Human-Computer Studies* 74 (2015), 81–92.

[34] P. Håndlykken and K. Nygaard. 1980. The DELTA System Description Language–Motivation, Main Concepts and Experience from Use. In *Proc. Software Engineering Environments Symposium*.

[35] C.N. Klokmose, M. Korn, and H. Blunck. 2014. WiFi Proximity Detection in Mobile Web Applications. In *Proc. Symposium on Engineering Interactive Computing Systems*. ACM, New York, NY, USA, 123–128. DOI: http://dx.doi.org/10.1145/2607023.2610281

[36] H. Korsgaard and M. Brynskov. 2014. City Bug Report: Urban Prototyping As Participatory Process and Practice. In *Proc. Media Architecture Biennale Conference*. ACM, New York, NY, USA, 21–29. DOI: http://dx.doi.org/10.1145/2682884.2682896

[37] M. Kyng. 1995. Making representations work. *Commun. ACM* 38, 9 (1995), 46–55.

[38] Y.K. Lim, E. Stolterman, and J. Tenenberg. 2008. The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. *Transactions on Computer-Human Interaction* 15, 2 (2008), 7.

[39] W. E. Mackay, D. S. Pagani, L. Faber, B. Inwood, P. Launiainen, L. Brenta, and V. Pouzol. 1995. Ariel: Augmenting Paper Engineering Drawings. In *Conference Companion on Human Factors in Computing Systems (CHI '95)*. ACM, New York, NY, USA, 421–422. DOI: http://dx.doi.org/10.1145/223355.223763

[40] Sarah Kuhn Michael J. Muller and Judith A. Meskill (Eds.). 1992. *Proc. of Participatory Design Conference*.

[41] P.H. Mogensen and S. Wollsen. 2014. Participatory Realisation?: PD in a Complex, Large-scale, and Commercial Context. In *Proc. Participatory Design Conference (PDC '14)*. ACM, New York, NY, USA, 21–30. DOI: http://dx.doi.org/10.1145/2661435.2661443

[42] A. Munk-Madsen. 1978. *Systembeskrivelse med brugere*. Datalogisk Afdeling, Aarhus Universitet.

[43] Aki Namioka and Douglas Schuler (Eds.). 1990. *Proc. of Participatory Design Conference*.

[44] K. Nygaard and O. Terje Bergo. 1975. The trade unions-New users of research. *Personnel Review* 4, 2 (1975), 5–10.

[45] D.R. Olsen Jr. 2007. Evaluating user interface systems research. In *Proc. User interface software and technology*. ACM, 251–258.

[46] S. Pilemalm and T. Timpka. 2008. Third generation participatory design in health informatics – making user participation applicable to large-scale information system projects. *Journal of biomedical informatics* 41, 2 (2008), 327–339.

[47] Y. Sundblad. 1984. Bilevel display screen representation of text, graphics, and images for page make-up work. In *Proceedings of Eurographics*, Vol. 84.

[48] M.W. Wartofsky. 1979. Perception, representation, and the forms of action: Towards an historical epistemology. In *A Portrait of Twenty-five Years*. Springer, 215–237.

[49] J. Whittle. 2014. How Much Participation is Enough?: A Comparison of Six Participatory Design Projects in Terms of Outcomes. In *Proc. Participatory Design Conference (PDC '14)*. ACM, New York, NY, USA, 121–130. DOI: http://dx.doi.org/10.1145/2661435.2661445