Check for updates

Intelligent Computing in Personal Informatics: Key Design Considerations

Fredrik Ohlin and Carl Magnus Olsson

Department of Computer Science
Malmö University
Malmö, Sweden

Internet of Things and People Research Center
Malmö University
Malmö, Sweden

{fredrik.ohlin, carl.magnus.olsson}@mah.se

ABSTRACT

An expanding range of apps supported by wearable and mobile devices are being used by people engaged in personal informatics in order to track and explore data about themselves and their everyday activities. While the aspect of data collection is easier than ever before through these technologies, more advanced forms of support from personal informatics systems are not presently available. This lack of next generation personal informatics systems presents research with an important role to fill, and this paper presents a two-step contribution to this effect. The first step is to present a new model of human cooperation with intelligent computing, which collates key issues from the literature. The second step is to apply this model to personal informatics, identifying twelve key considerations for integrating intelligent computing in the design of future personal informatics systems. These design considerations are also applied to an example system, which illustrates their use in eliciting new design directions.

Author Keywords

Personal informatics; quantified self; cooperative computing; augmented computing; ambient computing; cooperative action orchestration

ACM Classification Keywords

H.5.m Information Interfaces and Presentation (e.g. HCI): Miscellaneous; I.2.m Artificial Intelligence: Miscellaneous

INTRODUCTION

Activity trackers, smart watches, and large selections of sensor-powered smartphone applications all exemplify current technology supporting personal informatics. Using such technology, people can explore, strive to understand, and improve their behavior in ways not previously feasible due to the cumbersome data collection otherwise. Further contributing to the wave of interest in this technology is the *quantified self* movement, wherein enthusiast champion a 'data-driven life' [41], and publicly reflect upon appropriate practices and tools through meetups and on the web. Aside from private consumer use, the medical area has started showing significant interest in the possibilities of patient-driven longitudinal data collection [42] using this technology. Computing system support is therefore widely considered as an enabling factor for personal informatics [25, 26], and as a driver for adoption [38].

Nevertheless, current personal informatics systems often holds a simple focus on recording data and displaying it in statistical form to the user for post-activity viewing. In such cases, system support mostly takes the form of automation of previously manual tasks with automatic data collection being the most prominent example. More advanced forms of system support, informed by techniques from intelligent computing, are currently underexplored. This becomes a notable omission when considering that personal informatics is an area where engaged users willingly contribute increasingly rich data (cf. [8]) and thus offers a strong case for intelligent computing to be applied.

The purpose of this paper is to illustrates how intelligent computing could be included as a core part of personal informatics and thereby act as a catalyst for new forms of system support that may provide a richer understanding of the tracked activities to users than present systems allow. We begin our work towards this goal in the next section, by identifying related challenges and opportunities from current personal informatics research. The following section then contains a survey of established techniques and approaches from several areas of intelligent computing. The findings of this review is then used to inform a new model of cooperative action orchestration for human-centered intelligent computing, which is developed in the subsequent section. The model collates issues regarding how human and computing system both may drive cooperation, and mutually revise their behavior as part of a cooperative process. Finally, we present our main contribution of this paper, which is to apply the cooperative action orchestration model to personal informatics in order to identify key design considerations for incorporating intelligent computing into this area. As the design considerations are general to personal informatics, we also illustrate their use in eliciting new design directions for the specific example system *Ski Tracks* (shown in Figure 1).

IUI 2015, March 29–April 1, 2015, Atlanta, GA, USA.

ACM 978-1-4503-3306-1/15/03 ...$15.00.
http://dx.doi.org/10.1145/2678025.2701378

Figure 1. Ski Tracks is an example of a traditional personal informatics system. It provides convenient data collection via GPS, and simply presents the resulting data for post hoc analysis.

PERSONAL INFORMATICS

A substantial part of the current personal informatics research is focused on examining the real-world practices of users, and framing the resulting insights in terms of possibilities for system design. As a prominent example, Li et al. [25] describe five stages of personal informatics processes. Before collecting any data, users are in what Li et al. describe as the preparation stage, which deals with motivations for data collection, and defining what and how to collect data. The subsequent collection stage is when data is recorded during everyday use, and is followed by the integration stage where data is prepared for analysis. When data is deemed to be in a suitable form for analysis, users may then transition to the reflection stage. This reflection stage is were analyzed data is considered both in terms of reliability and to decide upon potential action-taking. This action stage is the final stage where users make informed choices of what to do given the insight obtained through the personal informatics process. The stages Li et al. describe are intended as iterative stages, and each stage has many potential barriers to users. These barriers cascade through the sequence of stages, meaning that problems in an earlier stage is likely to negatively impact the latter ones. Users may address barriers as part of their ongoing practices, or – if they fail to overcome them – simply give up. Identifying and overcoming barriers subsequently becomes one of the most important aspects of successful personal informatics practices.

Notably to the context of this paper, the stage-based model [25] of personal informatics includes the notion that each stage can be user-driven, system-driven, or driven in a hybrid fashion. This means that system autonomy and proactive behavior already are implied as key aspects of personal informatics systems. However, the stage-based model shares a limitation with other research, as we shall see in a moment, as it does not elaborate on the system autonomy implications. We do know that including system autonomy in personal informatics is not simple, as shown by Choe et al. [8]. From their analysis of 52 expert presentations from quantified self meetup groups, they find a tension between the convenience of automation and user engagement. They find this even for automated data collection, and ”envision striking a balance between fully automated sensing and manual self-report that can increase awareness, achieve better accuracy, and decrease mental workload” [8].

Parts of personal informatics is criticized by Rooksby et al. [35] for being overly technology-centric, and for assuming rational and systematic behavior that does not match the messy and intertwined activities and contexts of everyday life. As they put it, ”do not expect people to act as rational data scientists” [35]. This is in line with Choe et al. [8], who argue that users need system support designing their self-tracking experiments. Consequently, Rooksby et al. [35] recommend evaluating personal informatics systems not only by the resulting behavior change, but also from a broader experiential perspective. Calvo and Peters [7] also point to complexities in how people use personal informatics data, by using frameworks and evidence from psychology. Specifically, they show that goals may become obstacles to behavior change – contrary to the intention of them – and that we continually reinterpret data, leading to reduced predictability in how we behave.

An important discussion related to personal informatics is about the extent to which system design should be normative. The question posed is essentially: Should the system provide an objective starting point for personal reflection, or should it promote behavior found to be positive by its designers? The latter approach corresponds to what is argued for and described as persuasive technology. Such technology overlaps with what is used in personal informatics and is particularly common in health-related applications (cf. [17]). The former approach is more consistent with research examining self-reflection in personal informatics or interactive systems in general [34, 2, 22]. In their review, Baumer et al. [2] find such self-reflection to be positive for increased self-knowledge, which is the traditional goal of personal informatics [25].

Difficulties in performing sound data analyses is a barrier in many personal informatics systems. While different appropriate data visualizations techniques play a key role (cf. [14]), Bentley et al. [4] note that the general population has low chart literacy. This poses a considerable problem to personal informatics as such techniques dominate data visualization presently. Furthermore, and as individuals may be uncomfortable analyzing their own data [27], Bentley et al. [4] instead propose a system that provides textual descriptions of correlations found in the data. More conventional graphs are thus combined with statements such as ‘on weeks when you are happier you walk more (quite likely)’. They argue that this is a promising direction in cases where the personal informatics system acts as a cooperative agent.

Summarizing the review of personal informatics research, we find several challenges and opportunities related future directions informed by intelligent computing.

- Various forms of computing system support is key in personal informatics. There are opportunities for autonomous and proactive system behavior, both for collection and analysis of data. The full range of appropriate forms of system behavior is not known.
- Personal informatics relies on active user engagement and ongoing reflection. System design needs to strike a balance workload-reducing automation and promoting user engagement. Autonomous system functionality specifically to promote engagement and reflection is underexplored.
- System design needs to be mindful of the types of behavior change it promotes – and consider how this can to evolve in cooperation with the user.
- The need for flexibility is apparent from several perspectives – individualization between users, and over time.

HUMAN-CENTERED INTELLIGENT COMPUTING

Intelligent computing is being integrated in everyday life through appliances, entertainment systems, wearable and mobile devices, and similar everyday use technology. As a consequence, there is a shift in focus from ‘performing tasks efficiently and safely’ to ‘enhancing the lived experiences of individuals’. This experiential shift can be seen in much contemporary research, including ambient intelligence [10, 18]; cyber-physical systems [9, 37]; human-automation interaction [20]; and intelligent user interfaces. What can be learnt, so far, from this growing body of research? To answer this, we performed a review of related literature that identifies key design issues for intelligent computing and its integration in everyday life. In later sections we will use these show their usefulness in informing intelligent personal informatics systems.

In the survey of related research, three basic relationships between human and computing were found. In the first relationship, *cooperation*, intelligent computing is viewed as an agent of its own and with which humans interact (cf. [29, 16, 5]). In the second relationship, *augmentation*, human experience is partly shaped through the computing system (cf. [43, 13, 12]). The third form of relationship centers on system autonomy, where intelligent computing exists *ambient* to humans by being embedded in the environment and acting in the background of human actions (cf. [10, 18, 9]). The human-computing relationship is one major factor in the design of intelligent computing, and the following review of the literature consequently includes all three relationship types. The relationship types are perspectives on design in and of themselves, and their related research serves to highlight partly different concrete issues.

Cooperative relationship

Humans and computing systems have fundamental, and complementary, asymmetries [21, 6]. In essence, the argument this relies upon is that many tasks are best achieved by human and computer working collaborating based on the inherent strengths of each agent. Determining how such a collaboration can be designed is part of an ongoing discourse of autonomous computing, with early research dating back to the 50s and 60s [20].

One common concept is describing and analyzing the *level of automation*, commonly classified according to a ten level scale such the one given by Miller et al. [29]. This allows discussion and analysis of the degree of manual supervision required for a given system, ranging from “1. Human does it all”, via “5. Computer executes alternative if human approves”, to “10. Computer acts entirely autonomously” [29]. Automation classification schemes are often (e.g. [15]) informed by the early work of Sheridan and Verplank [36]. This early work did not intend for a specific level of automation to be assigned to systems as a whole. Instead, their original ten level scale was rather intended to be applied for a delimited step in the cooperative process.

From the discourse on scale as an elemental step follows the second key concept – the *concern of automation*. Parasuraman et al. [32] identified four concerns (acting as stages) of automation, from ‘information acquisition’, and ‘information analysis’, to ‘decision and action selection’, and ‘action implementation’. These represent a step towards acknowledging how humans and automation can cooperate in more nuanced ways, and are often cited to show how the level of autonomy can vary within one system.

A third key concepts is *adaptive autonomy* [19], which dictates that the appropriate level of autonomy should be decided based on the specific situation. This can reduce the need for human involvement, e.g. by only requiring supervision on tasks that are determined highly critical, or increase the level of automation when human performance is degrading. Miller et al. [29] show how adaptive autonomy is related to tradeoffs with regards to the human operator, namely to unpredictability and mental workload. The workload “can be reduced by allocating some functions to automation, but only at the expense of increased unpredictability” [29]. A corollary is that increased human management, described as adaptable autonomy, may sometimes be preferable. For instance, Ball and Callaghan [1] show adaptable autonomy to be necessary based on large variations of user preferences.

At this point, a final set of concepts relevant to a cooperative relationship between human and computer is starting to emerge from the discourse on autonomous computing. Similar to adaptive and adaptable autonomy is the notion of *mixed-initiative systems*, which focuses on effective and natural man-machine cooperation [16, 5]. In mixed-initiative systems, we can consider actions as possible, available or obligated – either independently by each actor or through joint effort [5]. Ferguson and Allen [16] describe how mixed-initiative systems rely on the human and the computing system agreeing on allocation of responsibility, and jointly committing to achieving tasks. The system can then exhibit proactive behavior in service of goals committed to, including communicating with the user to gather new knowledge deemed

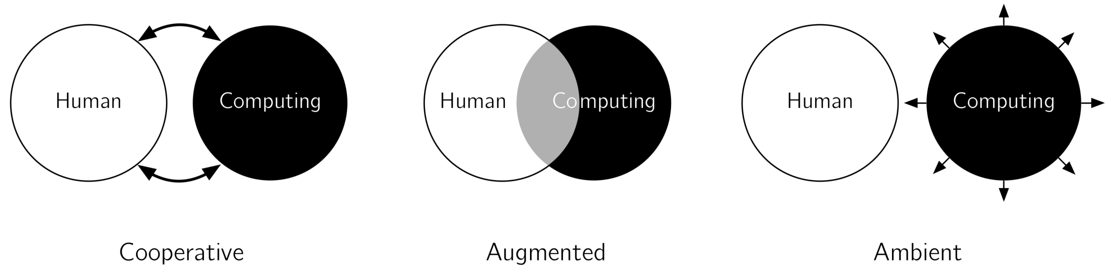

Figure 2. Three relationships between human and intelligent computing.

required. Shared awareness of the process, and alignment of the parties' intentions are identified as key issues. This is not unlike the shared task model [30] (extended further in [29]) which focuses on achieving a shared vocabulary of goals and plans - or 'task hierarchies' as the authors refer to these. More recently, Pacaux-Lemoin and Vanderhaegen [31] present a general model of human-machine cooperation, which features a core focus on mutual awareness between human and computing agent.

Augmented relationship

Augmentation is a second type of relationship between humans and intelligent computing. From this perspective, intelligent computing is not a separate agent, but something integrating with and enhancing human capabilities. The augmentation literature often uses Engelbart's [13] early work on 'augmenting human intellect' as a starting point, and his notion of 'coupled processes' still serves to highlight the quest for tight integration of human and artifact.

In light of the emergence of wearable computing, with intelligent computing facilitating personalized functionality, Xia and Maes [43] show that a personal enhancement view of augmentation is appropriate. They point to key questions that emerge from such a view, notably "How would I like to change myself?" and "What program can I employ to change myself?" – making clear connections to personal informatics. In their elaboration if augmentation research in that direction, Xia and Maes show how three cognitive domains are especially relevant: memory, motivation, and decision making. Each of these domains contains processes suitable for augmentation:

- Memory: event recording (information should not be lost or distorted); handle attachment (relevant handles should be attached to facilitate retrieval of stored information); handle usage (handles relevant to the current situation should be identified); and event playback (store information should be read correctly).
- Motivation: self-evaluation of performance (relate personal performance to goals); reminder of goal (remember and stick to goals); task identification (convert long-term goals into actionable tasks); and task evaluation (determine whether a task is worthwhile).
- Decision making: knowledge acquisition (consider knowledge relevant to the decision); value system formation; decision recognition (determine decision appropriate to context); decision framing (relate decision to time and choices available); and value system reconciliation.

Augmentation is also one emerging direction within personal informatics. Swan [40] uses the notion of 'the extended exoself' to refer human senses heightened by the real-time integration of personal informatics. This could mean more than simply showing current sensor values – the current status can also be related to goals, projections, and previous experiences. Swan joins Xia and Maes [43] in pointing to the importance of interaction technologies that support augmentation, discussing smart glasses and haptics as notable current examples.

Personal informatics augmentation can in this light be seen as a specialized version of situation awareness [39] – a version that is focused on the *self*. Traditional situation awareness is focused on enhancing comprehension of the current situation, with strong ties to fields like aerospace, military, and safety-critical systems (cf. [12, 28]), and does not highlight the self as focal point in the same manner as personal informatics. Situation awareness is commonly characterized according to three levels, each of which can be enhanced through computing system support: perception, comprehension, and projection [12]. Perception is the sensing and gathering of potentially important information, comprehension is the subsequent interpretation and understanding of this information, and projection is the forecasting of future events.

Ambient relationship

In the third relationship, intelligent computing exists in the background, ambiently acting autonomously as a mindful monitoring agent of human activity. Often, this is framed as placing the intelligence in the environment, as is done within ambient intelligence systems–defined by Cook et al. [10] as "a digital environment that proactively, but sensibly, supports people in their daily lives". Several types of environments make up the focus of ambient intelligence, including the home, workplace, healthcare, and transportation [10]. In all cases, this includes sensing, reasoning and acting based on the behavior of people present.

Enabling automatic sensing is a strength in personal informatics, but there is much to learn from the approaches to computational reasoning found within ambient intelligence. Key issues in this regard include the design of domain and activity models, as well as activity recognition and predictions [10].

Related to this is the importance of system designers understanding the situation of use [18].

Intelligent computing coexists with people not only as embedded in the physical environment, but also as part of the *digital environment*. The ongoing realization of ubiquitous computing and the Internet of Things results in a convergence of the digital and physical [9, 11]. This means that the digital part of everyday life increases as the worlds become less separated and subsequently not easily opted out of or ignored. Intelligent computing can be used to guard the interest of individuals through processes, preferences, and filtering mechanisms running in the background. A key issue is enabling people to understand and control such processes, and creating a scrutable system can be an explicit design goal. Making system models scrutable can serve to increase engagement and trust, as well as to facilitate user-driven identification of inappropriate system behavior [24].

A key issue for most forms of ambient intelligent computing is striking a balance between the system actively interacting with the user ('intervening') and autonomous action [10, 18]. This also relates to the broader discussion within ambient intelligence and ubiquitous computing of 'calm' or 'invisible' services. For instance, Jonsson [23] shows that users may strive to make such services more visible, in spite of designer intentions of keeping the computing in the background. Bellotti and Edwards [3] further establish intelligibility as a key design considerations for all forms of context-aware systems – meaning to design for the ability of computing systems to "represent to their users what they know, how they know it, and what they are doing about it".

COOPERATIVE ACTION ORCHESTRATION

The relationships presented earlier are all concerned with people working (or living) together with intelligent computing. The behaviors of the human actor and the computing system are, to varying extents and forms, dependent on each other in the three relationships. The computing system may make certain actions more likely, for instance by interjecting with a now-relevant piece information or recommendation. This in turn may cause the human to interact with the computing system to instruct or otherwise affect future system behavior, and so forth. This ongoing process can be characterized as a form of *cooperative action orchestration*.

In this section, we introduce a model of action orchestration which draws upon the human-computing relationships described earlier. We use the term *orchestration* as it emphasizes a mindful stance (from user as well as system) towards future progression of the cooperative process, rather than for instance a master-slave stance where one agent (human or system) would blindly follow the other's instructions. The model seeks to inform design of intelligent computing which enables such a stance from both individual and computing system. At the core of the model, there are four basic types of action (Table 1). These types are identified based on the work by Parasuraman et al. [33] and Bradshaw et al. [5], and separates the initiating party of an action from the executing party to identify the action types this results in.

		Initiating party	
		H	C
Acting party	H	$I_H A_H$ Human autonomy	$I_C A_H$ Computing-directed human action
	C	$I_H A_C$ Human-directed computing system action	$I_C A_C$ Computing system autonomy

Table 1. Four basic types of cooperative actions (H = Human, C = Computing system).

As a result of separating initiating from executing party, autonomous action may be performed by both the human ($I_H A_H$) and the computing system ($I_C A_C$). Such autonomous actions are by nature self-initiated and self-regulated, but they also occur in an environment of other agents. In the human case, action is situated, meaning that conscious and unconscious thought is given to the current context as the action is taken. Computing system autonomy similarly relies on system perception, as dictated by sensors and computing models for the state of the world as viewed by the system.

The computing system can also be instructed, or otherwise explicitly be engaged with, in order for the human to initiate an action ($I_H A_C$). Doing so not only depends on what the computing system's capabilities are, but also the interfaces through which the capabilities are exposed. The human user must have a way to express intent or instructions in mutually understandable form. The computing system can also ask the human to take action ($I_C A_H$), or more subtly provide information which indirectly affects human behavior. Computing-directed human action is limited and enabled by the interaction channels available to the computing agent. It also highlights issues of when it is appropriate for the computing agent to interject.

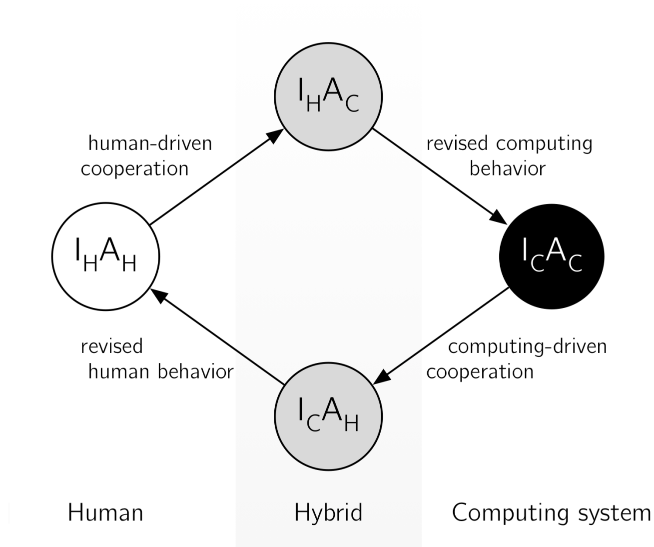

Figure 3. Transitions facilitating cooperative action orchestration.

While the types of cooperative action (Table 1) are important to note for orchestration, the possible transitions between the action types should also be considered as an ongoing process (Figure 3). By examining these transitions we can reveal aspects that constrain or enable the cooperative process to progress. For example, the human actor must be able to move from autonomous action to directing the computing system, which in turn must be able to revise its behavior. This revised behavior may then directly or indirectly affect future human interaction with the system, which illustrates why this is an ongoing process. The same example may be considered in the other direction as well, where the computing system must be able to engage with the user to affect its future behavior, and so forth.

In an intelligent computing informed personal informatics system, the four transitions are played out during the entire life-time of the human-system and not only during active use. Below, key considerations in terms of constraints and enablers are presented for each transition. These are not meant as an exhaustive lists of all possible considerations but rather to cover a core set of aspects from the literature. Prior to each set of aspects and related considerations, we provide a summary of the general issues.

Human-driven cooperation: $I_H A_H \rightarrow I_H A_C$

For transitions that show human-driven cooperation, an individual may contribute and drive the cooperative process based on the understanding this individual has of how the computing system analyses (i.e. makes sense) of human actions, as well as the individual's understanding of the orchestration process as a whole. An important issue therefore concerns how to facilitate human knowledge of the capabilities that the computing system holds, and understanding of the computing system's current state. Considerations related to this include making the system models visible and understandable [24]; explicitly highlighting current choices available [12, 43]; and the human learning based on observable behavior [31].

At a more basic level lies the issue of the conceptual design of cooperation and the subsequent framing of this that is presented to the human. One direction would be an outcome-focused presentation, meaning that goals are established and worked towards. Goals can concern either – or both – the human and the computing system (cf. [16, 29]). Cooperation can also hold a process-focused presentation, emphasizing how the human and the computing system influence each other's behaviors. Further, an assistance-focused presentation of cooperation represents a background relationship between human and computing system which is enabled by computational reasoning techniques [10]. Regardless which presentation form is used, it is not the specific design and inherent quality of this design of computational intelligence that is in focus but rather the *effects* of such choices upon the user.

For the human to actively engage with the computing system, there must be enabling interaction technologies, e.g. a smartphone application or physical controls (cf. [40, 10]). The user interfaces provided through these technologies further enable and constrain what is possible for the human to express.

Aspect	General issues	Considerations
Human understanding and motivation	Knowledge of computing system capabilities Knowledge of computing system state Trust in computing system Understanding current options for interacting with the computing system	Scrutable models [24] Intelligibility [3] 'Know-how', 'Know-how-to-cooperate' [31] Situation awareness: perception and comprehension [12] Decision framing augmentation [43]
Outcome-focused cooperation	Goals of H documented to C Goals of C programmed by H Shared goals	Mixed initiative: allocation of responsibility [16] Adaptive autonomy [29] Task identification augmentation [43]
Process-focused cooperation	Behavior of C programmed by H Behavior of H regulated by C	Concern of automation [32] User-adaptable autonomy [29] Task recognition [10]
Assistance-focused cooperation	Proactive stance from computing system	Concern of automation [32] User-adaptable autonomy [29] Task recognition [10]

Table 2. Key issues and related considerations for enabling human-driven cooperation.

Revised computing system behavior: $I_H A_C \rightarrow I_C A_C$

While the previous transition deals with enabling human-driven input into the cooperative process, the transition that follows such input is concerned with how the computing system revises its behavior as a result. This means that the transition covers the issues of mapping interpretations of human intentions – as communicated via what the interaction interface permits – to the internal functions and models of the computing system.

System designers must here tackle the possible scope of revised behavior (as in: what is the action space?), and what the sources of revised behavior are. This means considering possible and appropriate levels of automation, how and in which situations the level of autonomy changes, and the levels of explicit user control (cf. [19, 29, 5]).

Enabling more intelligent autonomous action requires understanding of both application domain and user behavior (cf. [10, 37]). Related considerations concern the design of models used by the system, which may have a very limited scope and dealing with only a limited set of highly specific concepts. This can come at the expense of being too constraining for the user, and models could therefore instead opt to include rich descriptions of the domain and individual user, at the possible risk of becoming overly complex and thus hard to understand and use. Related design considerations also concern the intended levels of flexibility between different users and over time (cf. [24, 29]).

Aspect	General issues	Considerations
Source of revised behavior	Explicit instruction Independent learning Mapping of user interface vocabulary to computational models	Levels of automation [19, 29] Adaptive and adaptable autonomy [29]
Scope of revised behavior	What can the computing system achieve?	Independently possible, available and obligated actions [5] Predictability [29]
Enabling autonomous action	Understanding of domain and user behavior Understanding of role in cooperation	Domain and activity models [10, 37]
Flexibility	Variations between users Variations over time	Scrutability [24] User-adaptable autonomy [29]

Table 3. Key issues and related considerations for enabling revised computing system behavior.

Computing-driven cooperation: $I_C A_C \rightarrow I_C A_H$

This transition covers considerations related to how the computing system move from existing ambiently in the background to actively engaging with the human. For such interjections to be possible, there needs to be supporting interaction technologies such as the push notifications on smartphones or haptic feedback from a wearable bracelet (cf. [10]). Different forms of interaction are associated with different user expectations that need to be taken into account. Experience and understanding of the particular computing system further affect such expectations [31].

Aspect	General issues	Considerations
Human openness to interaction	Knowledge and expectations on how the computing system behaves Possibilities to interact	'Know-how-to-cooperate' [31] Variation of interaction technologies [10]
Promoting human engagement	Knowledge about actions available Ease of affecting computing system behavior Actionable interjections Frequency of interjections	Decision making augmentation [43] Intelligibility [3] Adaptive autonomy [29]
Cost of interjection	Immediate cost: is interjection contextually appropriate Long term cost: is the ongoing amount and type of interjections appropriate	Automation: unpredictability and mental workload [32, 5] Type of interaction technology [10] User-adaptable autonomy [29]

Table 4. Key issues and related considerations for enabling computing-driven cooperation.

An underexplored but important aspect of this transition concerns the design of computing system interjections for promoting future human engagement (cf. [43, 29]). This includes facilitating more immediate engagement through actionable interjections, and promoting long-term engagement through techniques such as adaptive autonomy [29].

Too frequent or irrelevant interjections may incur an higher than acceptable cost to the user experience. The cost of interjections on human engagement may either be immediate, such as when they come at an inappropriate time (and thus go ignored or cause frustration). The cost/value ratio of interjections is also experienced over longer periods of time, meaning that a slight value of each interjection may not be enough to justify the process as a whole. Overall, there is a need for human-centered interjection protocols as the user should experience interactions as valuable, either through some instrumental value, or through understanding of the interjections' larger role in promoting end-user value (cf. [32, 5, 10, 29]).

Revised human behavior: $I_C A_H \rightarrow I_H A_H$

The final transition in the cooperative action orchestration process concerns how human practice changes as a result of the interactions with the computing system. The intent of computing system interactions needs to be understandable during use–e.g. is it a call to action to perform an immediate task, or is it simply a piece of information to consider, or is it only used to show that the computing system is active (cf. [12, 31])?

Revised behavior can also stem from a increased insight into how current behavior relates to possible outcomes. It can, however, be challenging to ongoingly understand one's behavior in terms of beneficial goals, which points to several considerations related to augmentation technology (cf. [12, 3, 43]).

Aspect	General issues	Considerations
Cooperative stance	Understanding of and engagement in cooperation Shared goals	Mixed initiative: agree on responsibilities [16] Shared task modelling [30, 29] Independently possible, available and obligated actions [5]
Immediate action	Communication of computing system intent Actionable interjections	Situation awareness: direction of attention [12] 'Know-how-to-cooperate' [31]
Informed change	Understand relationship between behavior and set goals Frame behavior in terms of possible goals	Situation awareness: perception and comprehension [12] Feedforward [3] Motivation augmentation [43]

Table 5. Key issues and related considerations for enabling revised behavior.

Additionally, human actors can of course revise their behavior independently of explicit interactions from the system. Revised human behavior may or may not affect the computing system and the cooperative process, depending on the system's ability to note this behavior change and adapt accordingly, rather than continue along the progression and data analysis path that the system had predicted previously as relevant. System designers therefore have considerations related to increasing the coupling between behavior change and cooperation with the system. In essence this implies striving to enable human behavior change that promotes further system cooperation (cf. [16, 30, 29, 5]).

INTELLIGENT COMPUTING IN PERSONAL INFORMATICS

Having identified key considerations for intelligent computing it is time to position these towards personal informatics. To do this, we again use the basis of the cooperative action orchestration model and the aspects of human-centered intelligent computing it highlights. The broad goal of this step is to move personal informatics beyond convenient data collection and visualization, to include proactive and cooperative behavior from the computing system. While our introduction of the cooperative action orchestration model earlier is an important step of this paper, it is how we use the model (to identify key considerations from intelligent computing) and its application to personal informatics (i.e. this section) that is the core contribution of this paper.

The resulting design considerations are described with a practical design orientation, as our intention is for them to be useful guides for actual design and development of future intelligent computing informed personal informatics systems, and not strictly as academic reflection guidelines. As the area of personal informatics is quite heterogenous – spanning from casual logging of everyday activities, via professional sports tracking, to health-critical applications – it may be inappropriate to strive for too overarching and generally applicable guidelines. As a consequence, we have kept to design considerations that need elaboration in the specific case. In their elaboration, these considerations will serve to highlight questions not commonly reflected in personal informatics.

From our review of personal informatics research, we know that people may benefit from system support in all stages of their personal informatics practices (cf. [25]). Research on e.g. automation design [5] and mixed-initiative systems [16] indicates that intelligent and proactive system behavior should not be approached as 'adding a feature' but rather a core conceptual design which should define *what the system is*. This points to possibilities for a broader reframing of what constitutes a personal informatics system, and we return to discuss this direction as a concluding element of this section. Such a reframing can also be seen as a theme behind the presentation of design considerations which follows. We will start, however, by identifying what the four transition stages of the cooperative action orchestration process mean in terms of personal informatics systems.

The key design considerations we present below for each transition are based on a synthesis of the existing research presented earlier in this paper. While the considerations are clearly situated within the context of personal informatics, Tables 2-5 contain the corresponding issues and references from intelligent computing.

The first transition from the cooperative action orchestration model concerns how to enable human-driven cooperation. In terms of personal informatics, this means enabling users to think not only of themselves and their behavior, but to couple these things with potential system support. Based on the intelligent computing issues reported for this transition (Table 2), the following considerations may be viewed key for personal informatics systems:

1. *Explore framings of personal informatics as cooperation between human and computing system.* Goals in personal informatics are traditionally framed as a concern for the human, and the computing system's role is to 'keep track' of them. In the alternative framings of cooperation, presented as part of this first transition, goals are an equal concern of both parties, and serve to establish agreements on who should do what.

2. *Consider conceptual designs which shift the focus of behavior change to a joint human-system effort.* Insight gained from data analysis would then equally be used to update future behavior of the personal informatics system. This may require user interfaces for data analysis to better highlight possible contributions from the system.

3. *Consider the effect of increased focus on cooperation in terms of human engagement and motivation.* Expose what the autonomous capabilities of the personal informatics system are, as well as its earlier contributions.

4. *Evaluate when and how the computing system should to be available to the human, and focus on interaction technologies accordingly.* What will the relationship between human and personal informatics system be (cooperation, augmentation, coexistence) and how will this vary over the course of use?

The second transition from the cooperative action orchestration model concerns enabling revised behavior from the computing system. For personal informatics systems, this can mean an increased focus on richer models of both application domain and user activity as these enable independent reasoning. Key design considerations thus include (see Table 3 for the underlying issues from intelligent computing):

5. *Focus not only on models of the domain (e.g. specific type of exercise or biometric factor), but also on models of how people use the personal informatics system itself.* This would enable reasoning about the cooperative process and more sophisticated system behavior like adaptive autonomy.

6. *Include explorations autonomous behavior in the design process of personal informatics systems.* Further consider collecting ongoing use data or feedback to identify pain points, e.g. in terms of too much manual work.

In the third transition of cooperative action orchestration, focus is on enabling the computing system to initiate interactions. As with the human-driven perspective, this requires supporting interaction technologies which make the human available. Unless the personal informatics system has its own hardware component, it must rely on the equipment commonly used – which will likely mean the smartphone remains a primary enabler for the foreseeable future. Regardless of the enabling interaction technologies, several design considerations relate to the design of the system interjections themselves, as they make up a primary channel through which the system can keep the user in the loop. Key design considerations for computing-driven cooperation in personal informatics systems (based on the core issues in Table 4) are:

7. *Explore forms of real-time system-provided feedback to decrease the need for sophisticated post hoc data analysis by the user.* This implies a shift of personal informatics system from context-recording to context-aware.

8. *Explore ways of grounding real-time feedback in past performance, projections, and/or goals.* Simply showing current sensor data readings does not sufficiently promote increased insight into potential behavior change – designs should strive for heightened understanding of the current situation.

9. *Explore ways including users in processes of automatic data collection.* Insight into the autonomous behavior of the system may promote user engagement, although the frequency and type of interjections will likely have to vary (related to next point).

10. *Consider ways of letting users give feedback on interjections from the personal informatics system.* System interjections provide a natural opportunity to allow the user to tweak system behavior, potentially making for low cost user-adaptable autonomy.

The final transition of cooperative action orchestration, revised human behavior, establishes a new perspective on behavior change in personal informatics by making it inclusive of user-system cooperation. We have already discussed the possibility of adopting conceptual designs in this direction, which would promote the aspect of cooperative stance found in this transition as well. Considerations specific to this transition are as follows (see Table 5 for the underlying issues from intelligent computing):

11. *Include possibilities for immediate action with interjections from the personal informatics system.* This can facilitate the user opportunistically engaging in the cooperative process.

12. *Explore more sophisticated relationships between personal informatics data and goals.* One direction is towards answering the user question: based on my current status, what goals can be appropriate to work towards? (Or, as has been discussed, which goals should *we*, the user and the personal informatics system, adopt together?)

Taking a step back from the four transition stages, we can now return to the matter of incorporating intelligent computing as more than adding a feature. Overall, our analysis points to a reframing of what constitutes a personal informatics system. New systems powered by intelligent computing are not simply advanced versions pen and paper, but represent active participants in the ongoing personal informatics practice. In this reframing, hybrid actions between user and computing system – which make up the core of the cooperative action orchestration model – subsume the earlier human-only, or tool-only, focus. As a result, the established notions of 'activity tracking' and 'self-quantification' may be inadequate for future personal informatics systems informed by intelligent computing.

Example: Applying the Considerations in Practice

The presented design considerations are intended to be applied in the context of a specific personal informatics system. To illustrate such a process and potential outcome, we will use the Ski Tracks smartphone application shown in the introduction of this paper (Figure 1). The application may be described as continuously tracking the user's movements via GPS, once the user has activated it. When the skiing session is over, the user stops the tracking and may then access key statistics and simple visualizations that are relevant to downhill skiing. The application thus enables convenient logging and analysis that would be cumbersome or impossible to do manually. The application is made with particular care to the needs of skiers and snowboarders, and can be considered a strong implementation and functional application when compared to other similar applications. Thus, what would it mean to include lessons from intelligent computing in the particular case of Ski Tracks? To answer this, we will elaborate on each of the presented design considerations (C1-C12).

C1 and C2 both suggest that we consider what the application is, in terms of how it is experienced by the user. As is the case for many personal informatics systems, Ski Tracks is conceptually designed as tool for data tracking, and in light of this, personal informatics becomes human-driven. C1 asks us instead to explore how Ski Tracks can be framed as a cooperative agent with more active participation. One primary direction to consider, as included in C1, is designing goals such that they are of concern for both human and computing system. Goals could be included in Ski Tracks as part the current logging and analysis. For instance, the user might configure a goal of 10000 meters descent per day, with an evaluation of this goal being added to the analysis screen ("Achieved 81 % of 10 k descent goal"). This would not, however, promote increased participation of the computing system, as C1 asks. Instead, we can expand on the goal example to include the application helping in goal achievement. This may entail configuring Ski Tracks to notify of progress ("Skiing for 3.5 hours and currently at 51 % of descent goal with 3 hours before the lifts close") or advise on behavior ("Do 3 more runs to achieve descent goal" or "Stop current break soon to if you want to achieve descent goal at your average pace for today").

Further, C2 then asks for ways to include contributions from the computing system as a perspective during data analysis, complementing data describing only the user. This implies a future design that complements visualization of ski data by also showing when system notifications occurred. Such a change would allow the user to evaluate if revised system behavior is appropriate, and may also lead the user to notice where the system could have helped but did not.

Ski Tracks taking a more active role may have both positive and negative effects on user engagement and motivation, which C3 asks us to consider. Specifically, C3 suggest managing user expectation by clearly exposing what system-initiated behavior is possible. To address this, the application could for instance include a textual description when creating the type of goal illustrated above ("Ski Tracks will let you know if you have to increase the pace to reach the day's goal").

Skiing is an activity which inherently exemplifies an activity that may prohibit typical phone usage, which means that system-driven interactions are particularly important that they come at a contextually appropriate time, such as when stopping to get on a lift rather than during the actual descent. C4 asks us to examine what those situations may be, and what supporting interaction technologies to use as a consequence. In this situation, the phone notification model may be considered inadequate for ongoing interaction, which instead suggest that future designs explore more substantial changes such as integration with smart watches or goggles with heads-up display, thereby facilitating an augmentation relationship (cf. Figure 2).

The Ski Tracks application includes models of the activity of skiing allowing it to e.g. determine when the user is riding a lift. C5 asks us to go beyond this and also explore models of how the application itself is used. For instance, this can include adapting what data is highlighted based on previous user behavior during data analysis. C6 further focuses on identifying user pain points that can be alleviated through autonomous system behavior. In its current design, users must activate Ski Tracks manually, or re-activate after pausing, and this seemingly simple step may result in significant data loss if not performed. A simple form of proactive system behavior could target this specific pain point, e.g. by considering time of day or specific user instructions.

C7 and C8 both concern real-time feedback, as opposed to Ski Tracks' current focus on post hoc analysis. In exploring contextually-relevant feedback in real-time, starting with determining contexts which the application already has data to identify – such as "having stopped after the fastest run of the season", or "going on the same slope for the third time in a row" – would pragmatically be useful. However, providing contextual feedback is further enabled by the available interaction technologies (as discussed with C4) and may thus not be feasible only through the smartphone. Similarly, when considering C9, which asks for increased user engagement in automatic data collection, we may find it unnecessary for this particular application. This is because Ski Tracks, in contrast with general always-on fitness trackers, has a very clear connection to a particular type of activity. In other words, users are likely to be highly engaged in the activity of skiing, with which the application is closely tied. If system-initiated interactions are added to Ski Tracks, C10 suggests including the option of immediate user feedback – e.g. "do not show notifications like this" or "keep me posted on this goal".

The two final concerns relate to revised human behavior. C11 emphasizes that notifications from the application should not only be understandable, but also actionable when appropriate. In listing what types of actions that may be appropriate for Ski Tracks to suggest, we can note that some actions are human-only (e.g. skiing a particular slope), while others are cooperation with the system (e.g. "compare future runs to this previous one"). Finally, in considering C12, choosing and setting goals in Ski Tracks should not be the sole responsibility of the user. Given some initial use, the application may suggest goals that are likely to be attractive (based on the domain of skiing), and fitting (based on user data).

As presented here, the design considerations for personal informatics systems have been applied to the Ski Tracks application, in order to illustrate how the consideration may be used to elicit future design directions that build upon state-of-the-art opportunities that are recognized within intelligent computing. This implies that the presented design directions do not represent a complete list, as they are based on the domain for inquiry identified as relevant to this paper. Additional domains for inquiry would add further design considerations, and represents an obvious way to expand upon this paper.

CONCLUSIONS

This paper set out to explore how intelligent computing could be included as a core part in personal informatics and thereby act as a catalyst for new forms of system support that may provide a richer understanding of the tracked activities than present systems allow. To form a basis for this exploration, we have surveyed current research on personal informatics to more concretely reveal the related challenges and opportunities. We have also presented established techniques and approaches from several areas of intelligent computing, resulting in the new cooperative action orchestration model.

The model of cooperative action orchestration is combined with the insights from personal informatics to form the domain-specific key design considerations. These can be used to elicit new design directions for individual personal informatics systems, as illustrated with the Ski Tracks example. Neither the list of design considerations, nor the core action orchestration model, can ever be said to be fully exhaustive. This does not diminish the value of what has been presented, however, and we see a promising research direction in iteratively revising both model and considerations as part of future design and development. In light of these results, we also see the need for a consistent vocabulary facilitating clear comparison between different personal informatics systems. Another intriguing possibility is using the action orchestration model to form consideration for domains other than personal informatics, thus contributing broader insight into how humans and computing systems can complement each other in practice.

REFERENCES

1. Ball, M., and Callaghan, V. Explorations of Autonomy: An Investigation of Adjustable Autonomy in Intelligent Environments. In *Proc. IE 2012*.
2. Baumer, E. P. S., Khovanskaya, V., Matthews, M., Reynolds, L., Sosik, V. S., and Gay, G. Reviewing reflection: on the use of reflection in interactive system design. In *Proc. DIS 2014*.
3. Bellotti, V., and Edwards, K. Intelligibility and Accountability: Human Considerations in

Context-Aware Systems. *Human-Computer Interaction 16*, 2 (2001), 193–212.

4. Bentley, F., Tollmar, K., Stephenson, P., Levy, L., Jones, B., Robertson, S., Price, E., Catrambone, R., and Wilson, J. Health Mashups: Presenting Statistical Patterns between Wellbeing Data and Context in Natural Language to Promote Behavior Change. *Transactions on Computer-Human Interaction (TOCHI) 20*, 5 (2013).
5. Bradshaw, J. M., Feltovich, P. J., Jung, H., Kulkarni, S., Taysom, W., and Uszok, A. Dimensions of Adjustable Autonomy and Mixed-Initiative Interaction. In *Agents and Computational Autonomy*, M. Nickles, M. Rovatsos, and G. Weiss, Eds. Springer Berlin Heidelberg, 2004, 17–39.
6. Bradshaw, J. M., Sierhuis, M., Acquisti, A., Feltovich, P., Hoffman, R., Jeffers, R., Presott, D., Suri, N., Uszok, A., and van Hoof, R. Adjustable Autonomy and Human-Agent Teamwork in Practice: An Interim Report on Space Applications. In *Agent Autonomy*. Springer US, 2003, 243–280.
7. Calvo, R. A., and Peters, D. The irony and re-interpretation of our quantified self. In *Proc. OzCHI 2013*.
8. Choe, E. K., Lee, N. B., Lee, B., Pratt, W., and Kientz, J. A. Understanding quantified-selfers' practices in collecting and exploring personal data. In *Proc. CHI 2014*.
9. Conti, M., Das, S. K., Bisdikian, C., Kumar, M., Ni, L. M., Passarella, A., Roussos, G., Tröster, G., Tsudik, G., and Zambonelli, F. Looking ahead in pervasive computing: Challenges and opportunities in the era of cyber-physical convergence. *Pervasive and Mobile Computing 8*, 1 (2012), 2–21.
10. Cook, D. J., Augusto, J. C., and Jakkula, V. R. Ambient intelligence: Technologies, applications, and opportunities. *Pervasive and Mobile Computing 5*, 4 (2009), 277–298.
11. Crabtree, A., and Rodden, T. Hybrid ecologies: understanding cooperative interaction in emerging physical-digital environments. *Personal and Ubiquitous Computing 12*, 7 (2008).
12. Endsley, M. R. Theoretical Underpinnings of Situation Awareness: A Critical Review. In *Situation Awareness Analysis and Measurement*, M. R. Endsley and D. J. Garland, Eds. Lawrence Erlbaum Associates, 2000, 1–24.
13. Engelbart, D. C. Augmenting Human Intellect: A Conceptual Framework. Tech. Rep. AFOSR-3223, Oct. 1962.
14. Epstein, D., Cordeiro, F., Bales, E., Fogarty, J., and Munson, S. Taming data complexity in lifelogs: exploring visual cuts of personal informatics data. In *Proc. DIS 2014*.
15. Fereidunian, A., Lehtonen, M., Lesani, H., Lucas, C., and Nordman, M. Adaptive autonomy: Smart cooperative cybernetic systems for more humane automation solutions. In *Proc. SMC 2007*.
16. Ferguson, G., and Allen, J. Mixed-initiative systems for collaborative problem solving. *AI Magazine 28*, 2 (2007), 23–32.
17. Fritz, T., Huang, E. M., Murphy, G. C., and Zimmermann, T. Persuasive technology in the real world: a study of long-term use of activity sensing devices for fitness. In *Proc. CHI 2014*.
18. Gross, T. Towards a new human-centred computing methodology for cooperative ambient intelligence. *J Ambient Intell Human Comput 1*, 1 (2009), 31–42.
19. Hancock, P. A., Chignell, M. H., and Lowenthal, A. An adaptive human-machine system. In *Proc. SMC 1985*.
20. Hancock, P. A., Jagacinski, R. J., Parasuraman, R., Wickens, C. D., Wilson, G. F., and Kaber, D. B. Human-Automation Interaction Research: Past, Present, and Future. *Ergonomics in Design: The Quarterly of Human Factors Applications 21*, 2 (2013), 9–14.
21. Hoffman, R. R., Feltovich, P. J., Ford, K. M., and Woods, D. D. A rose by any other name...would probably be given an acronym. *Intelligent Systems, IEEE 17*, 4 (2002), 72–80.
22. Huldtgren, A., Wiggers, P., and Jonker, C. M. Designing for Self-Reflection on Values for Improved Life Decision. *Interacting with Computers 26*, 1 (2014)
23. Jonsson, K. Making IT Visible: The Paradox of Ubiquitous Services in Practice. In *Industrial Informatics Design, Use and Innovation Perspectives and Services*. IGI Global, 2010, 30–43.
24. Kay, J., and Kummerfeld, B. Creating personalized systems that people can scrutinize and control: Drivers, principles and experience. *Transactions on Interactive Intelligent Systems (TiiS) 2*, 4 (2012).
25. Li, I., Dey, A., and Forlizzi, J. A stage-based model of personal informatics systems. In *Proc. CHI 2010*.
26. Li, I., Dey, A. K., and Forlizzi, J. Understanding my data, myself: supporting self-reflection with ubicomp technologies. In *Proc. UbiComp 2011*.
27. Mamykina, L., Mynatt, E., Davidson, P., and Greenblatt, D. MAHI: investigation of social scaffolding for reflective thinking in diabetes management. In *Proc. CHI 2008*.
28. Matheus, C. J., Kokar, M. M., and Baclawski, K. A core ontology for situation awareness. In *Proc. Fusion 2003*.
29. Miller, C. A., and Parasuraman, R. Designing for Flexible Interaction Between Humans and Automation: Delegation Interfaces for Supervisory Control. *Human Factors: The Journal of the Human Factors and Ergonomics Society 49*, 1 (2007), 57–75.

30. Miller, C. A., Pelican, M., and Goldman, R. "Tasking" interfaces to keep the operator in control . In *Proc. Human Interaction with Complex Systems 2000*.

31. Pacaux-Lemoine, M.-P., and Vanderhaegen, F. Towards Levels of Cooperation. In *Proc. SMC 2013*.

32. Parasuraman, R. Designing automation for human use: empirical studies and quantitative models. *Ergonomics 43*, 7 (2000), 931–951.

33. Parasuraman, R., Sheridan, T. B., and Wickens, C. D. A model for types and levels of human interaction with automation. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 30*, 3 (2000), 286–297.

34. Pirzadeh, A., He, L., and Stolterman, E. Personal informatics and reflection: a critical examination of the nature of reflection. *Proc. CHI EA 2013*.

35. Rooksby, J., Rost, M., Morrison, A., and Chalmers, M. C. Personal tracking as lived informatics. In *Proc. CHI 2014*.

36. Sheridan, T. B., and Verplank, W. L. Human and Computer Control of Undersea Teleoperators. Tech. rep., 1978.

37. Sheth, A., and Anantharam, P. Physical cyber social computing for human experience. In *Proc. WIMS 2013*.

38. Shilton, K. Participatory personal data: An emerging research challenge for the information sciences. *Journal of the American Society for Information Science and Technology 63*, 10 (2012), 1905–1915.

39. Smith, K., and Hancock, P. A. Situation Awareness is Adaptive, Externally-Directed Consciousness. In *Situational Awareness in Complex Systems*, R. D. Gilson, D. J. Garland, and J. M. Koonce, Eds. Aeronautical University Press, 1994, 59–68.

40. Swan, M. The Quantified Self: Fundamental Disruption in Big Data Science and Biological Discovery. *Big Data 1*, 2 (2013), 85–99.

41. Wolf, G. The Data-Driven Life *New York Times Magazine*, Apr. 2010.

42. Wright, A. Patient, heal thyself. *Commun. ACM 56*, 8 (Aug. 2013).

43. Xia, C., and Maes, P. The design of artifacts for augmenting intellect. In *Proc. AH 2013*.