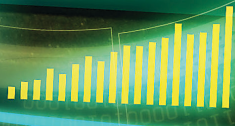




48
mph



- /Autonomous
- /Sensing
- /Communication
- /Battery
- /Navigation
- /Mirrorless
- /Ecology



Intervention User Interfaces: A New Interaction Paradigm for Automated Systems

Insights

- Machine learning and artificial intelligence enable devices, applications, and systems that act with great autonomy.
- Interaction design for autonomous systems inherently requires joint control, which is not well addressed by classical user interfaces.
- We introduce the intervention user interface paradigm to address challenges for creating novel user interfaces in autonomous systems.

Computer science has been driving automation in the workplace and the home. Automated processes and autonomous systems are having an impact on our experience with technology. Will we still need humans in the loop? Will HCI as a discipline get sidelined? In a 2015 keynote, Yvonne Rogers raised the question “User-less or User-full Interfaces?” discussing the consequences of leaving the user out of the loop (<http://muc2015.mensch-und-computer.de/programm/keynotes/>).

Automation takes on tedious, boring, or error-prone tasks using machines and algorithms. This vision proclaims that humans will then be free to do more exciting and

more cognitively challenging tasks. Recent breakthroughs in machine learning and artificial intelligence imply that more and more tasks can be automated. This leads to the fundamental question “What can humans do better than computers and robots?” being discussed in both the science community and greater society. In our view, this is the wrong question!

It is not about machines replacing humans. We see a major paradigm shift in how we as humans use technology and what types of interaction are appropriate. The crucial question is how a team of humans and machines collaborating is better than either humans or

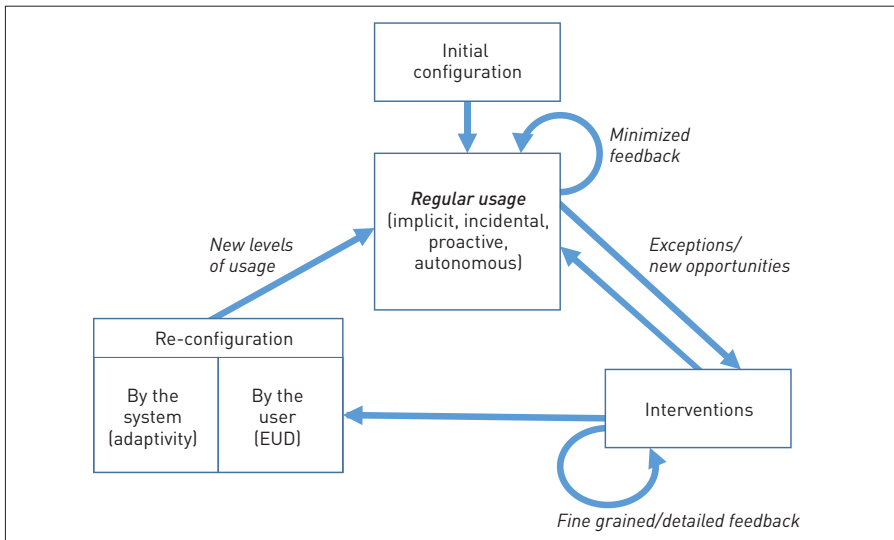


Figure 1. The relationship between regular usage, intervention, and configuration.

technology alone. This calls for tasks where humans using machines outperform machines. These tasks are not obvious. For example, which type of chess computer helps teams of chess players to outperform other teams or systems like Deep Blue (<http://www-03.ibm.com/ibm/history/ibm100/us/en/icons/deepblue/>)? These developments in artificial intelligence and in cyber-physical systems force us to fundamentally rethink how tasks are divided between humans and computers.

We believe that a large class of automated and autonomous systems allow for joint control, where the majority of decisions are automated but where users can intervene. To address this challenge, we introduce the *intervention user interface paradigm*. It describes a new approach for interaction with technologies that are predominantly based on automation and artificial intelligence. Designing for joint control is widely applicable and relevant for various cases.

We define the concepts of *intervention*, *intervention user*

interfaces, and *granularity of control*. Using examples, we explain the complexity in designing interfaces for automated systems and highlight the challenges when moving from continuous interaction to intervening interaction. We refer to the well-known golden rules of Ben Shneiderman to derive six design principles for intervention user interfaces.

INTERVENING INTERACTION AND INTERVENTION USER INTERFACES

With the shift from command-based dialogues and direct-manipulation interfaces toward more automation and to autonomously acting agents, conventional dialogue structures no longer adequately support the interaction requirements. Systems are configured prior to operation, and in many applications further interaction may not be apparent. In many areas, automation has removed the need for continuous, fine-grained interaction. Similar to direct-manipulation systems and proactive computers, the initiative for interaction can be taken by the

human or the machine. A challenge is that humans must be made aware and understand that there is an opportunity for interaction.

As we know from airplane pilots, the concept of continuous monitoring and interaction is hard and not appropriate, for two reasons. First, it is difficult for humans to effectively monitor systems that work automatically over a long time and react in a moment of crisis. Second, there is no sense in staying in the interaction loop while autonomous operations run according to the user's expectations. We believe, however, that humans should still have an appropriate level of control. Hence, a new interaction metaphor is required to complement the configuration of systems: intervention.

Thus, we define the following concepts:

Intervention in human-computer interaction is an action by the user that takes place during the usage of an automated system and initiates a diversion from the predefined behavior.

Intervening interaction allows the user to alter the behavior of a process that is regularly highly automated, and continuously proceeds according to a plan or to situational awareness without the need for any interaction. This ad hoc change through exceptional control takes place in accordance with emerging user needs or situations.

Intervention user interfaces allow for unobtrusive awareness of automated processes, suggest options for intervention, and provide elements that allow the user to change the behavior of the currently running processes with immediate effect. Intervention user interfaces are robust and support the exceptional nature of interventions.

Interventions are typically due to needs that are not foreseeable by the automated system. Allowing intervention makes the system's implementation and configuration easier since the developer, designer, or user can pay less attention to all kinds of exceptions that might occur.

Intervention and configuration are closely linked: The autonomous behavior of systems is pre-set, typically by a combination of factors, including the implementation by

We believe that a large class of automated and autonomous systems allow for joint control, where the majority of decisions are automated but where users can intervene.

software developers, algorithms, the configuration during installations, and settings by service providers, users, or learning mechanisms. Configuration precedes usage and usually supports regular tasks. Configuration and implementation are based on anticipation of what will be required and are related to foreseeable contextual influences.

Intervention can compensate for the limited situational awareness of an autonomous system. Therefore, autonomous behavior can be enlarged since users can intervene to compensate for the system's inability of reacting in an unknown situation.

Interventions have the following characteristics:

- They are not planned and occur only exceptionally.
- They are fast to initiate and impact the system immediately.
- They cyclically help to improve automated behavior.

The notion of control is central to interaction with automated and autonomous systems. At the same time, when we assume a world with a large number of devices (e.g., the ubiomp vision), then it is apparent that traditional fine-grained control is no longer a viable option.

Granularity of control refers to the number of actions needed during interaction to achieve the desired impact of the behavior of a system or application.

As an illustration, consider parking a car. In a traditional car, the user parks herself, exercising a continuous and fine-grained level of control. Every 100–500 milliseconds, she assesses the situation and controls or corrects the steering, acceleration, and braking. In a car with parking assistance, the driver still controls parking, but needs only two steps: choosing a parking spot and activating the parking procedure. The fine-grained control (steering, driving) is automated. In an autonomous car, the user exits the car and it autonomously finds a parking spot, manifesting even greater granularity.

Figure 1 shows the relationship between regular use in the automated case, interventions, and configuration (see the Related Concepts sidebar). During regular use, the system will work autonomously; with an

intervention, it moves to fine-grained control and feedback. Repeated interventions may be a trigger for reconfiguring the system, either by automated adaptivity or by end-user development (EUD).

With respect to our concept of intervention, we can say that intervention is relevant for exceptions within routinized tasks. According to Jens Rasmussen [1], these tasks are controlled subconsciously. If it turns out that a routinized task has to be adapted to meet exceptional needs, this requires conscious control. However, while routinized tasks require a high degree of repetition for humans who carry out these tasks (e.g., the daily trip to work), this does not necessarily apply to autonomous vehicles and their related infrastructure. The first trip by a single user of a car from his home, perhaps located in the Bay Area, to a campground at Yosemite National Park may appear as a task that requires conscious control. However, an autonomous vehicle could refer to a thousand similar trips that have already taken place by other travelers—and it could even provide a checklist of the items to bring to the campground. The user would not have to apply conscious control even though this trip is not a regular, repetitive task for him. However, he should know whether he might have special requirements that

differ from the majority of people doing the same trip, and if this is the case, he should be ready to intervene and take over control with respect to certain aspects of the trip.

THE COMPLEXITY OF INTERACTION WITH AUTOMATED SYSTEMS

The design space for interaction with automated systems is surprisingly complex. Even interventions with simple systems, such as an automated light switch using a motion and daylight sensor (Figure 2), are conceptually complex. An autonomous car illustrates how automation and intervention can be combined. A basic set of questions includes:

- How should automation and explicit user input be combined?
 - How are conflicts between user input and automated behavior managed?
 - How is the opportunity for interventions and their impact communicated?
 - What is the temporal characteristic of an intervention?
- More complex automation procedures lead to more options; hence, creating interfaces for intervention becomes more complex.

Consider an autonomous car for your commute to work. On an abstract level, autonomous driving is similar to highly automated industrial manufacturing or robot-based



Figure 2. Automated light that is reactive to motion in the dark. Configuration allows the setting of thresholds for sensors and timings. Interventions prolong or terminate the light exceptionally.

RELATED CONCEPTS

Intervention user interfaces are inspired by other reflections on interaction. What the concepts of implicit [2], incidental [3], and proactive interaction [4] have in common is that they support autonomous behavior by observing the changes of a system's context, partially with sensors. These concepts are relevant in ubiquitous computing, autonomous vehicles and traffic systems, and automated manufacturing (https://en.wikipedia.org/wiki/Industry_4.0), where the role of humans is challenged by transferring control to systems and distributing and decentralizing decisions between various components.

Concepts such as intervening use [5] build a contrast with usage for routine tasks and emphasize the interdependency between intervention and explorability. Intervenability as a means of privacy protection [6] emphasizes the modification of personal datasets and profiles of users.

RELEVANT CASES FOR INTERVENTION

Autonomous vehicles in networked traffic systems: Control can be temporarily handed over from the vehicle to the driver.

Smart homes and ambient assisted living: Implicit interaction based on context (e.g., motion or infrared sensors); control of heating, ventilation, and air conditioning.

Automated manufacturing and smart factories: Predefined maintenance intervals can be adjusted if urgent orders have to be dealt with.

Personal assistance with wearable and mobile devices: They offer sensor-based data tracking and contextual awareness.

Machine learning and cognitive services: Interventions can influence the sets of data and profiles aimed at improving the service used.

services. Those cases have in common that sensors and actors are included, complex components are connected in a network, and data is continuously produced and triggers machine-learning-based improvement. Autonomous cars are part of a traffic system, exchanging data between participating components and sensors. Interventions can be on different levels, including a stop at a coffee shop, the demand for a detour to pick someone up, or changing priorities (e.g., speed versus fuel economy).

The interaction design for interventions raises the questions of how options are visible and how consequences are comprehensible.

One option for intervention interfaces in the car may be to stick with the steering wheel and pedals. However, what would the semantics for intervention be using these controls? Using the brake pedal to signal a need to stop, tilting the steering wheel for a change in direction, and hitting the accelerator pedal to increase speed? In what temporal way would this intervention be combined with the autonomous behavior? What does it mean if users stop hitting the accelerator pedal? Does it mean that they want to reduce speed or that they intend to switch back to the automated speed control? These problems point to an essential characteristic of intervention

For intervening interaction, it has to be clear whether a change is short term (e.g., changes in how I drive to work today) or is meant as a configuration (e.g., my workplace has permanently changed).

interaction: The automated routines must eventually be resumed after a certain time span or through an explicit interaction. A real and dramatic example in which this binary solution (automation on or off) fails is the switching off of automated, distance-based braking systems in trucks. Drivers who switch off distance braking and forget to switch it back on again regularly cause accidents on highways. Similarly, one can discuss the semantics of manual steering.

For intervening interaction, it has to be clear whether a change is short term (e.g., steering changes in how I drive to work today) or is meant as a configuration (e.g., my workplace has permanently changed).

GOLDEN RULES AND INTERVENTION DESIGN

Ben Shneiderman's golden rules (<https://www.cs.umd.edu/users/ben/goldenrules.html>) have proven over the years to be a substantial orientation point for user interface design. Since the underlying concepts still have great relevance, we discuss them in the context of autonomous systems. We suggest

some reconsiderations to understand the differences that are related to the paradigm of intervention.

Strive for consistency. Consistency requires the same or similar sequences of actions in similar situations. Since fine-grained control and actions are drastically reduced in automated systems, the consistency of system behavior should be perceptible across different situations.

The rule for autonomous systems is: *Strive for dynamic and contextual consistency and ensure expectability and predictability.* Deviations from expected behavior (e.g., an autonomous car always takes the same route) *should be understandable and explainable. If inconsistency is perceived, intervention must be offered.*

Enable frequent users to use shortcuts. Shortcuts make frequent and repetitive actions more efficient. Automation already removes needs for frequent actions and explicit interaction becomes an exception.

Thus we propose: *Replace the need for frequent explicit actions and interventions by automation.*

Frequent users should benefit from the system's self-reconfiguration (see Figure 1). Because of the required immediateness, we claim: *Create shortcuts for interventions that combine a sequence of interactions.*

Offer informative feedback.

Originally, feedback accompanies every (major) action and is drastically reduced when using a whole infrastructure of automated systems.

Designing the balance of automation and intervention has to address the feedback dilemma: Calm environments and minimizing the attention required are key aspects; feedback directs attention to unexpected behavior. *Feedback about the impacts of automated behavior and the temporary potentials for intervention must be offered.*

This is complemented by: *Provide feedback on whether or not intervention is occurring and: Offer feedback (implicit/explicit) on the impact of intervention.*

Design dialogue to yield closure.

Users can understand how their activities contribute to a cycle of task performance and how far it is completed. With automated systems, this is mainly about starting and terminating a process, while intervention means to switch back to fine-grained control.

Therefore, we propose: *Design the start and control of intervention in conjunction with clear and simple options for completing and terminating it.* Interventions should be designed to have a limited temporal impact.

Offer simple error handling. The rule conventionally covers human errors. With automated systems, the challenge is about whether the user lets the system go on when errors happen or may be expected.

Thus the context of automation requires: *Allow for immediate intervention to avoid the occurrence or repetition of unsolicited automated behavior.*

Permit easy reversal of actions.

This feature encourages users to try efficient ways and unfamiliar features. This remains an important rule, but literal reversal of human action may not be feasible; reversing its *impact*, however, may be.

Thus we state: *Combine the means*

for reversing the impact of automated actions with intervention interfaces and: Allow for simple means for reversing the impact of interventions.

This kind of reversibility is crucial to encourage users to explore the system.

Support internal locus of control.

The user should feel like they are the one who controls the system. With automated systems and implicit interaction, the relationship between controlling action and reaction has to be newly balanced.

It has clearly to be communicated:

- *How the impacts of interventions are related to the goals being pursued by the automated processes*

- *How control is distributed between the automated system and the user.*

Possibilities of intervention offer flexibility for control and enable ad hoc changes.

Reduce short-term memory load.

The amount of information to be kept in mind to efficiently interact with a system has to be reduced. Little information should be needed as long as active control is not necessary.

We propose: *Do not require the user to remember a previous system status.* And between the interventions: *Minimize required attention and design for default behavior.*

The system should be designed in such a way that the user could intervene and go back to automated behavior without remembering and noting the current status for future use.

SIX DESIGN PRINCIPLES FOR INTERVENTION USER INTERFACES

Based on the discussion, we propose the following design principles:

- **Expectability and predictability.**

Ensure that users are not surprised by automated behavior and that they understand how it develops.

- **Communicate options for interventions.** Make options for interventions that may be context-aware visible and understandable for users in an unobtrusive way.

- **Exploration of interventions.** Allow the safe and enjoyable exploration of interventions and their potential impacts, e.g., by simulation or previews on future system statuses.

- **Easy reversal of automated and intervention actions.** Offer a simple means to reverse the impact of the system's automated behavior or of the results of interventions.

- **Minimize required attention.**

Minimize the user attention required to operate the system by implicitly controlled feedback.

- **Communicate how control is shared.** Clearly communicate the distribution of responsibilities, as well as the actual control between the human and the machine.

Automation and autonomous systems are inevitably coming in many domains. We hope our proposed principles help to develop usable intervention interfaces and kick-start a fundamental discussion on interaction with autonomous systems.

ENDNOTES

1. Rasmussen, J. The human as a systems component. In *Human Interaction with Computers*. H.T. Smith, T.R.G. Green, eds. Academic Press, London, 1980, 67–96.
2. Schmidt, A. Implicit human computer interaction through context. *Personal Technologies 4*, 2 (2000), 191–199.
3. Dix, A. Beyond intention-pushing boundaries with incidental interaction. *Proc. of Building Bridges: Interdisciplinary Context-Sensitive Computing, Glasgow University* (Vol. 9), 2002.
4. Tennenhouse, D. Proactive computing. *Communications of the ACM 43*, 5 (2000), 43–50.
5. Herrmann, T. Support of intervening use. *Ergonomics of Hybrid Automated Systems 3* (1992), 289–294.
6. Rost, M. and Bock, K. Privacy by design and the new protection goals. *DuD* (Jan. 2011).

Albrecht Schmidt is a professor of human-computer interaction and cognitive systems at the University of Stuttgart. His research interests are at the intersection of ubiquitous computing and human-computer interaction, including large-display systems, mobile and embedded interaction, and tools to augment the human mind. He has a Ph.D. from Lancaster University.
→ albrecht.schmidt@acm.org

Thomas Herrmann is a professor of information and technology management at the University of Bochum, Germany. Current research interests include design methods for sociotechnical systems in various areas such as healthcare, computer-supported collaboration, knowledge management, process management, and industry 4.0. He has a Ph.D. from the Technical University of Berlin.
→ thomas.herrmann@ruhr-uni-bochum.de