

By focusing on users' abilities rather than disabilities, designers can create interactive systems better matched to those abilities.

**BY JACOB O. WOBROCK, KRZYSZTOF Z. GAJOS,
SHAUN K. KANE, AND GREGG C. VANDERHEIDEN**

Ability-Based Design

RECALL THE LAST time you took a trip out of town. Perhaps you were traveling to a conference far from home. Remember the many forms of transportation you endured: cars, buses, airplanes, and trains. Not only were you responsible for moving yourself over a great distance, you had to move your things as well, including books and baggage. Remember the cramped spaces, sharp elbows, body aches, and exhaustion. Feel again your desire to simply be at your destination with your possessions intact . . .

Such journeys remind us of our physical embodiment in the physical world, that much of our lived experience is fundamentally physical, and that

we must contend with the world on physical terms. As computing professionals, we might be tempted to forget this, as our keystrokes summon data instantly from across the globe. But as humans, we still interact with that data through physical devices and displays using our physical senses and bodies. We and the world interact physically.

Civilization's story of technological progress is in no small part the story of an increasingly built physical environment, from the pyramids to roads to skyscrapers to sanitation systems. Much of our energy, collectively and individually, goes into moving and shaping material for such purposes, altering the physical landscape and our movement through it. Some of our most thrilling experiences come by way of changing our bodies' relation to that landscape: bungee jumping, skydiving, scuba diving, and riding a rollercoaster all provide radically new experiences for our bodies in the world.

As designers and builders of interactive systems for human use, we also play a central role in defining people's relationship to and experience of the physical world.^{2,13,30} When we design things, we take mere ideas, things without form, and embody them in the world, whether simple sketches or cardboard mockups. They could be pixels on a screen or functioning digital devices. Regardless of the medium, to design and build things is to embody ideas that are then encountered and

» key insights

- **Ability-based design is a new design approach for interactive systems that focuses on people's abilities in context, on what people *can* do, rather than on what they *cannot* do.**
- **Ability-based design scrutinizes the "ability assumptions" behind the design of interactive systems, shifting the responsibility of enabling access from users to the system.**
- **People's abilities may be affected not just by disabilities but by disabling situations; designing for abilities in context leads to more usable, accessible systems for all people.**



IMAGE BY FXQUADRO

used by other embodied people.

This design-and-build activity is profound. It was not long ago in human history that giving form to the formless was considered the purview of the divine. In fact, the English verb “to create” comes from the Latin “creare,” which means to bring “form out of nothing.” When we design and build systems, we bring form out of nothing.

Unfortunately, unlike the divine, we cannot anticipate all the ways our designs will affect the people who encounter them. And when a mismatch arises, the world can become a very rigidly embodied place (see Figure 1).

Many of the great breakthroughs in interactive computing have come as improved embodiments capable of transforming the way people experience the digital world. Sutherland’s interactive display and light pen in SketchPad,³¹ Engelbart’s and English’s mouse in NLS,⁴ and Apple’s iPhone

all represent breakthrough embodiments. But a vital engineering insight is that they, as with all interactive technologies, include certain “ability assumptions” that must be met by human users. These assumptions are often unstated but alienating if they cannot be met.

An everyday example makes the point. In the student union building at the University of Washington in Seattle, wall-mounted touchscreens function as information kiosks for visitors (see Figure 2). In the on-screen operating instructions, a particular word stands out—“just,” as in, “*just* touch the screen.” In fact, touching the screen requires many abilities, including closing one’s hand, extending one’s index finger, elevating one’s arm, seeing the target, landing accurately, holding steady, and lifting without sliding—along with the ability to read and understand the instructions in the

first place. There is clearly no “just” about it.

Where do ability assumptions come from? Designers and developers make assumptions from their own abilities, from the ones they imagine other people have, or the ones of the supposed “average user.”²² Unfortunately, each source of such assumptions is flawed. The first two are prone to bias and unrepresentative; the third, insidious for its statistical façade, does not reflect the diversity of human life.

On that point, Rose²⁵ offered an anecdote from the U.S. Air Force. After World War II, it frequently lost pilots and planes in peacetime crashes—incredibly, 17 on one particular day—so it decided to redesign its cockpits to reduce “pilot error.” Air Force engineers measured 4,063 pilots along 140 dimensions, averaging these values to create cockpits to fit the mathematically average pilot. But a young Air Force scien-

tist, Lt. Gilbert Daniels, questioned this approach. He took just 10 of the most important dimensions, added a tolerance of 30% of their ranges around their means, and compared every individual pilot to them to see how many of the 4,063 pilots aligned. The surprising result? Zero. Even among pilots recruited for their congruity, human diversity dictated that individual differences ruled. Only when the Air Force created pilot-

configurable cockpits covering the 5th to 95th percentile of pilot measurements did the crashes decline.

Motivated by a need to make interactive computing systems that better match users' abilities, we formulated "ability-based design,"^{37,38} aiming to create accessible technologies for people with disabilities and for people in disabling situations (such as in the dark or while walking in the cold or encum-

bered). Following our work on adaptive user interfaces⁹⁻¹¹ and technologies for people on the go,^{15,24,32,33} ability-based design pursues an ambitious vision—that anyone, anywhere, at any time can interact with systems that are ideally suited to their situated abilities, and that the systems do the work to achieve this fit. Here, we expound this vision and describe the steps we have taken toward achieving it.

Ability and Disability

It helps to be explicit about the term "ability." For our purpose, a useful definition comes from the Oxford dictionary: "Possession of the means or skill to *do* something"^a (emphasis ours). The focus is on acting in the world, not just thinking about it.

Defining the term "disability" is thornier. In 1976, the World Health Organization (WHO) defined disability as, "Any restriction or lack ... of ability to perform an activity in the manner or within the range *considered normal for a human being*"³⁹ (emphasis ours). Thankfully, in 2001, this normative language yielded to the International Classification of Functioning, Disability, and Health,^b authored and adopted by WHO, identifying disability as a complex interaction among an individual, activity, society, and the environment, both social and physical. Indeed, research has illuminated just how much social factors play a role in the experience of disability.^{28,29}

When considering disability, ability-based design goes further. If "ability" is about having the means or skill to do something, then "disability" means simply being unable to do something. Disability becomes something one experiences rather than something someone has or is. Following such a view, everyone experiences disability, because everyone lacks the means or skill to do quite a few things, at least in certain circumstances. Designing for abilities applies to all people.

We call this perspective the "positive affirmation of ability," namely that all people have abilities, some more than others, and designers and

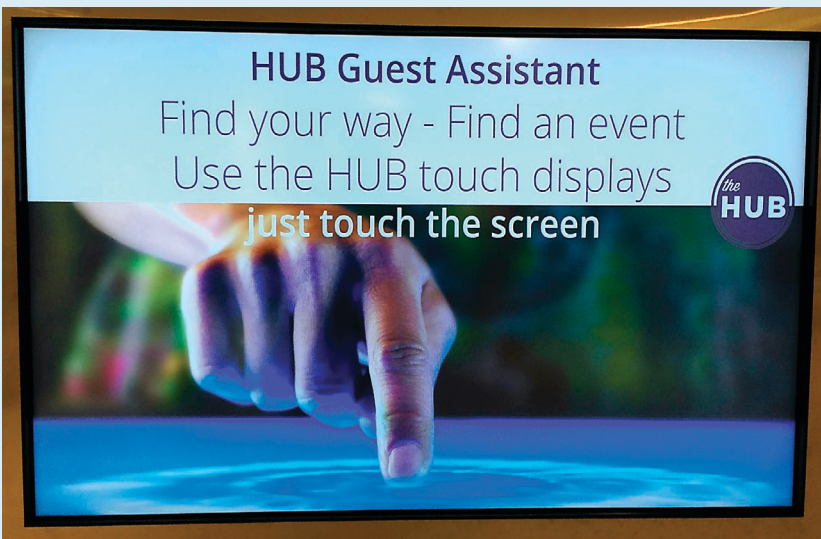
a <https://en.oxforddictionaries.com/definition/ability>
 b <http://www.who.int/classifications/icf/en/>

Figure 1. A person in a wheelchair facing a flight of concrete stairs.

Does the challenge lie in his inability to walk or in the stairs' requirement that he must? Regardless, people should expect and receive more accommodation from their interactive computing systems than they do from an immutable flight of concrete stairs.



Figure 2. A wall-mounted touchscreen instructing users to "just touch the screen," though a great many abilities are required to do so.



developers ought to create systems for people with abilities of all kinds and degrees. Likewise, Newell²² referred to “extra-ordinary abilities,” saying, “common sense and observation show us that every human being has . . . abilities, some of which can be described as ‘ordinary’ and some of which are very obviously extra-ordinary.” The focus is not on disability but on the diversity of human ability.

Ability is thus like weight or height—it is positive-valued only. Nobody has dis-weight or dis-height; neither are there disabilities, only abilities. Any experience of disability is not attributable to a person but to a mismatch between a person’s abilities and the ability assumptions of the environment. Like the proverbial water in a glass half full, abilities are only present and “designed for,” not absent and “filled in.”

This view of “design for” rather than “fill in” is not the historical view. Filling in for lost abilities has been the norm. From early human history through World War II and after, the approach has been to restore whatever was lost (such as an arm or a leg). People were expected to adapt themselves to the environment, whether physical or social, as they found it, with little hope that society would meet them halfway.

Although such attitudes have improved, designers and developers still often take a similar stance with interactive computing systems. When users’ abilities fail to match the ability assumptions underlying today’s interactive computing systems, the burden usually falls on the users to make themselves amenable to those systems, and the systems remain oblivious to the users doing it (see Figure 3).

Ability and Situation

The experience of disability applies to us all. With the proliferation of smartphones, tablets, and wearables, we increasingly interact with systems in situations that challenge our abilities.

Consider how the physical environment of “the computer user” has changed from the 1980s to today. A typical computer user in the 1980s would have been seated at a stable work surface with ample lighting, controlled temperatures, quiet surroundings, and relatively few distractions. Today, with computing pervading so many as-

Figure 3. Users adapting themselves to the ability assumptions of their input devices—keyboards and trackballs—which are oblivious to their contortions.



(a)



(b)

pects of life, “computer users” interact off-the-desktop while adapting to dynamic, distracting environments and their movements through them.⁷ An example is how users interact in “four-second bursts”²⁴ when walking with smartphones, constantly diverting their attention from and returning to their screens. And yet, with the exception of a few research prototypes (such as in Mariakakis et al.¹⁹), smartphones are oblivious to users’ behaviors, unchanging from the street to the café to the library to the office.

Researchers have identified “situational impairments” caused by changing situations, contexts, and environments, using the language of

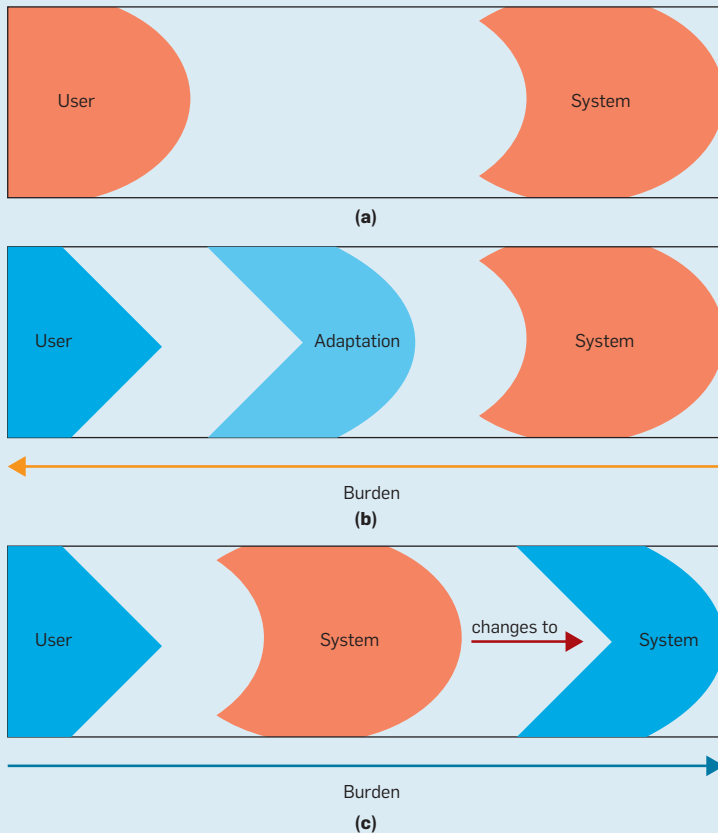
disability and accessibility.^{7,22,27,33,38} Sears and Young²⁷ said, “Both the environment in which an individual is working and the current context . . . can contribute to the existence of impairments, disabilities, and handicaps.”

This observation has grown even more relevant in the 15 years since it was made. In Stockholm, Sweden, city officials have erected street signs alerting drivers to watch out for people texting while walking. In Seoul, South Korea, some sidewalks are divided into two lanes, one for those intent on walking while staring at their phones, and the other for those who promise to refrain. In the U.S., the Utah transit authority imposed a \$50

Table 1. Situational factors that can limit our physical and cognitive abilities and affect our interactions with technology.

Vibration	Cold temperatures
Divided attention	Impeding clothing (such as gloves)
Distraction	Encumbering baggage
Diverted gaze	Rainwater
Device out-of-view	Light levels (such as darkness and glare)
Intervening objects	Ambient noise
Bodily motion (such as walking)	Social behaviors (such as interruptions)
Vehicular motion	Multitasking
Uneven terrain	Stress
Physical obstacles	Fatigue
Awkward postures or grips	Haste
Occupied hands	Intoxication

Figure 4. User abilities and a system's ability assumptions: (a) user abilities match a system's ability assumptions; (b) in assistive technology, the user acquires an adaptation to remedy a mismatch; and (c) in ability-based design, user abilities drive changes in the system.



fine for “distracted walking,” including walking while texting. And the city of Honolulu adopted the Distracted Walking Law, banning even just looking at a screen while in a crosswalk. Alarming, the Federal Communications Commission estimates that at any daytime moment in the U.S., 660,000 people are interacting with

their smartphones while driving.⁶

If we are to design for human ability, disabling situations must be addressed. Unfortunately, our interactive computing systems know little about their users’ abilities, attention, situations, con-

c <https://www.fcc.gov/consumers/guides/dangers-texting-while-driving>

texts, and environments. A great many factors can impair use (see Table 1), yet few of them are detected, accommodated, or used as a basis for discouraging or deferring interaction.

Toward Ability-Based Design

Addressing such concerns while providing a unified approach to designing for people of all abilities is why we pursued ability-based design,^{37,38} a design approach in which the human abilities required to use a technology in a given context are scrutinized, and systems are made operable by or adaptable to alternative abilities. Emerging from our work on adaptive user interfaces,⁹⁻¹¹ ability-based design is characterized by the designer’s focus on what people can do, rather than on what they cannot do, and on systems and environments adapting to users rather than the other way around. Examples include desktop interfaces that customize their designs based on how a user moves a mouse,¹⁰ touch surfaces that observe complex motor-impaired touch sequences and resolve intended touch points,²¹ and mobile touch keyboards that sense and accommodate walking to improve accuracy.¹²

Strategies

Ability-based design is pragmatic, concerned with abilities insofar as they are useful for design. It is thus strategy-agnostic, embracing multiple methods for achieving successful user-technology fits. Strategies include automatic ability-based adaptation; high configurability by the end user; ability-specific customization by a third party; and having multiple designs for alternative abilities. Regardless of which one is employed, ability-based systems do the work to match users’ abilities, not burdening users with having to satisfy a system’s rigid ability assumptions.

Employing a visual language developed by Edwards,³ we outline a successful user-system fit in Figure 4a, where a user’s abilities match a system’s ability assumptions. In traditional assistive technology, when they do not match, as in Figure 4b, the burden falls on the user to become amenable to the system by procuring an adaptation. The adaptation fits and makes the user “seem normal” to the system. With ability-based design, this burden is reversed

(see Figure 4c); it is the user's abilities that dictate what the system must do to make itself amenable to the user. For example, the system might adapt or be adapted to match the user's abilities.

Ability-based design differs from traditional assistive technology by eschewing user-procured adaptations like the one in Figure 4b in favor of on-board adaptability. When on-board adaptability is not possible or practical, assistive technologies can still meet the objectives of ability-based design if they are well matched to the user's abilities and not burdensome to procure. In cases where assistive technologies are used, ability-based systems should be aware of their use and do whatever they can to make that use as uninhibited as possible.

Ability-based design also relates to universal design.¹⁸ Arising from the field of architecture, universal design readily applies to built structures and spaces and has been extended to physical and digital products as well. Universal design is the process of designing places and things so they are usable by people with the greatest range of abilities possible. Ability-based design

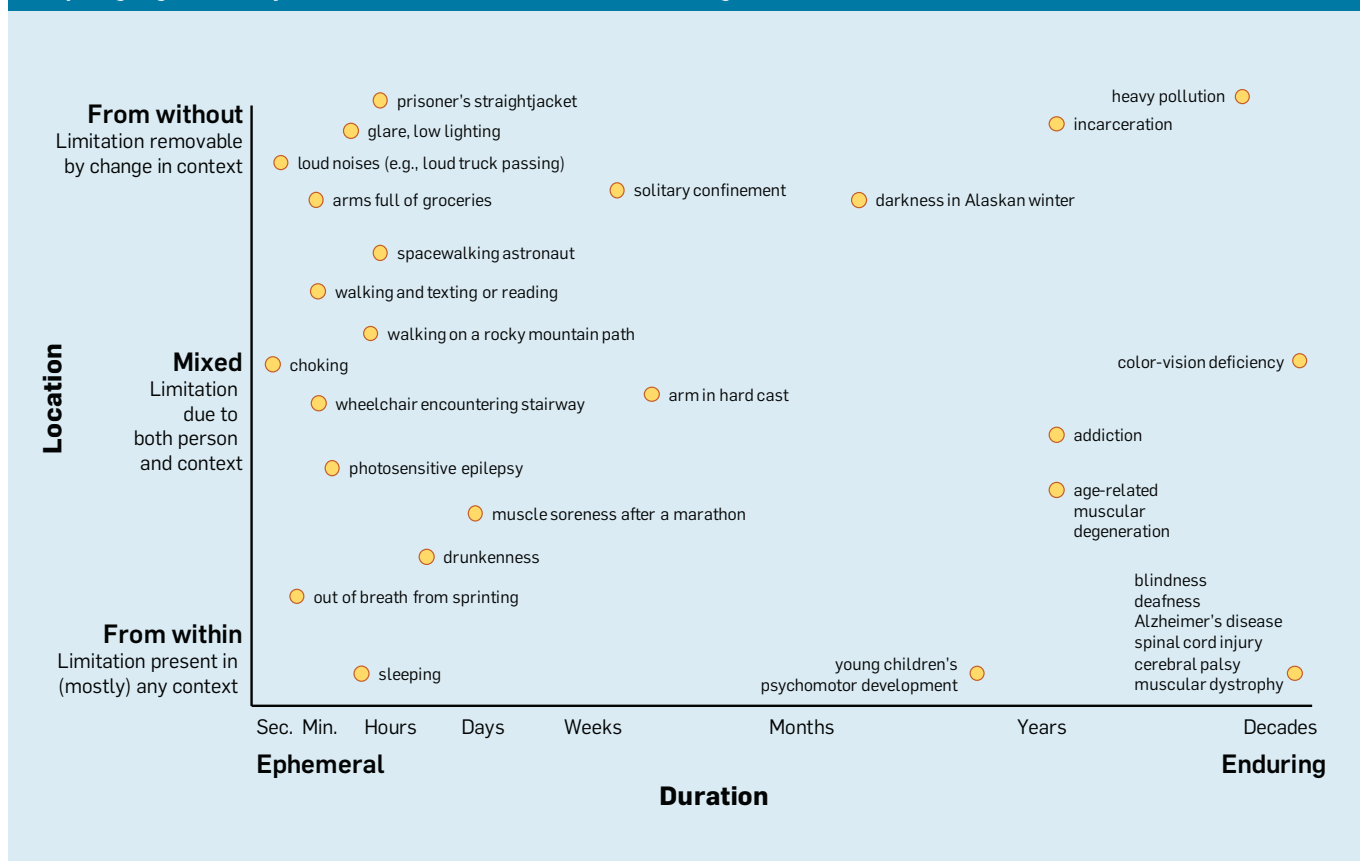
creates designs that match the abilities of individual users to the greatest extent possible. Ability-based design is thus one way to realize the ambitions of universal design. Unlike universal design, however, we created ability-based design with interactive computing in mind, so sensing, adapting, and configuring are presumed technology possibilities. While ability-based design might not natively apply to immutable concrete stairs, as in Figure 1, it would thus ask how future stairways (or wheelchairs) might use sensing, adapting, and configuring to prevent accessibility barriers.

Other strategies for designing for diverse abilities exist and are similar to ability-based design insofar as they consider users' abilities and the role of the environment. For example, inclusive design^{16,23} seeks to eliminate design choices that cause exclusion by revealing designer biases through participatory methods, field observations, and empathy building. Among the foci of inclusive design is understanding user capabilities, similar to ability-based design.

A key difference between ability-based design and both universal design and inclusive design is one of focus and approach. Universal design and inclusive design focus on creating designs that are for general widespread use, including by people with specific interface needs. Ability-based design promotes creating general interfaces with the flexibility to address a range of users, as well as tailored interfaces specific to subgroups or even to an individual user. Ability-based design potentially has broader reach since it embraces both flexible-general and tailored-specific interfaces in its scope and approach.

With ability-based design, there is also a subtle but important difference in focus by the researcher, designer, or developer. With universal design or inclusive design, the focus is on creating an interface that can accommodate as many people as possible. With ability-based design, the focus is on the abilities of the individual user. All three approaches might at times produce similar designs, but with ability-based design, the focus is on optimizing the

Figure 5. Contexts that impair one's ability to use technology are defined by location and duration. What advances in sensing and computing might enable systems to better serve their users across a range of contexts?



experience for individual users according to their abilities and contexts.

Contexts Limiting Technology Use

Ability-based design considers a broad range of contexts that impair technology use. We define a space with two axes: location and duration (see Figure 5). The location of a limitation ranges “from within the self” to “from outside the self.” Limitations arising from within the self are present in almost any context. Examples are a spinal cord injury, a toddler’s undeveloped psychomotor control, and being asleep. Changing a person’s context has little effect on the limitations arising from such internal states.

In contrast, limitations arising from outside the self are present primarily due to context, and therefore changeable. Astronauts have remarkable physical abilities, but while spacewalking, expressing many of those abilities is quite difficult. Even an Olympic athlete can do little when confined to a prisoner’s straightjacket. The external context severely limits the person’s expressible abilities.

Intermediate points also exist on the location axis, where the mixture of self and environment limit ability. One example of a mixed-location limitation is photosensitive epilepsy, where a flashing light might induce seizures. If not for the flashing light, seizures would not be triggered. In this example, a part of the person and a part of the environment combine to pose a possible limitation.

On the other axis, the duration of a limitation ranges from “ephemeral” to “enduring.” An ephemeral limitation lasts only briefly and changes quickly; one example is the lack of a usable arm because a person is carrying an infant. Next, short-term limitations can arise from many causes, including inebriation, illness, and an ankle sprain. Limitations might even be enduring or even lifelong, as with, say, those caused by age-related declines, spinal cord injuries, incurable diseases, lifetime imprisonment, or irreversible brain damage.

Our argument is not that the lived experience of a person with one arm is the same as that of a person carrying an infant. Situational impairments are neither subjectively nor

objectively anything like long-term limitations. Rather, the argument is that technology designs that are useful to people with certain long-term limitations might also be useful to people in certain disabling situations. A technology design for a person with one arm also might be useful for a person carrying an infant. Using an ability-based lens helps one recognize such design opportunities.

Assistive technology focuses mainly on compensating for long-term limitations within a person, as in Figure 5, bottom right. Ability-based design considers a larger space of limitations that impair technology use.

Design Principles

By adopting ability-based design in numerous projects, we have formulated and refined seven design principles to guide our work (see Table 2). The first three are required of any ability-based design project and relate to the designer’s attitude and approach, or “stance.” The next two relate to adaptive or adaptable user interfaces, and the final two to sensing and modeling users and contexts. Taken together, they can help guide designers and developers creating ability-based systems.

Example Projects

Our development of ability-based design was and continues to be highly iterative and inductive, arising from research projects that both preceded and followed its initial formulation. Here, we highlight a number of projects to illustrate the possibilities for ability-based design:

SUPPLE. SUPPLE⁹⁻¹¹ was an automatic user-interface generator that used decision-theoretic optimization to help choose interface widgets and layouts that were optimized for a user’s preferences, visual abilities, and motor abilities. For optimizing motor performance, SUPPLE first presented the user with a series of basic pointing, clicking, dragging, and list-selection tasks.¹⁰ It then built regression models capturing the relationship between task parameters and user performance, using these models to guide the optimization process such that the interface being generated was predicted to be the fastest to operate by the user. Each user thus received a cus-

tom user interface, optimized for that user’s particular abilities.

In a quantitative study in 2008 involving people with motor impairments,¹¹ SUPPLE’s custom interfaces were 26% faster and 73% more accurate to use than the default interfaces provided by manufacturers of popular desktop software applications. SUPPLE thus helped close more than 60% of the performance gap between people with and people without motor impairments, making access more equitable. Qualitatively, it was apparent how SUPPLE was optimizing interfaces based on different abilities; for example, SUPPLE gave people with muscular dystrophy interfaces with small, densely packed targets able to support slow, short, deliberate movements. In contrast, SUPPLE gave people with cerebral palsy interfaces with large, spread-out targets divided among different tabs, compatible with fast but error-prone movements. SUPPLE had no declarative knowledge of either muscular dystrophy or cerebral palsy, generating its user interfaces solely from observed input performance.

The SUPPLE approach was used in subsequent projects. For example, in SPRWeb,⁶ SUPPLE’s personalized optimization approach was used to recolor websites, adapting them to the individual color-vision abilities of users with color-vision deficiencies. SPRWeb also aided users in color-limiting or color-altering situations, including glare and low-light conditions.

SUPPLE exhibited the first six principles of ability-based design and was the original system that inspired many of the ideas now found throughout ability-based design.

Slide Rule. Slide Rule¹⁴ was a mobile screen reader that made touchscreens accessible to blind users by leveraging multi-touch gestures and audio feedback. It was an example of making systems usable to people with abilities different from what device manufacturers originally intended. Slide Rule addressed a pressing challenge emerging in 2007 from the advent of touchscreen smartphones: How would a blind person interact with a phone having buttons that person could not feel? At the time, smartphones had little or no acces-

sibility support, and many people presumed touchscreens could not be made usable for blind people. Slide Rule developed a set of gestures and the first finger-driven screen-reading techniques to enable blind people to access and control smartphone touchscreens.

We became aware from a personal communication in 2010 that Slide Rule inspired aspects of Apple’s VoiceOver screen reader for iOS. Indeed, Slide Rule’s finger-driven screen reading, swipe gestures, and second-finger tap can all be found in VoiceOver today.

Slide Rule exhibited the first three principles of ability-based design; it also exhibited the fourth and sixth principles, as its screen reader could adapt to the speed of users’ movements, tailoring its performance to theirs. The underlying principles demonstrated in Slide Rule have survived into today’s touchscreen systems.

Walking user interfaces. Today’s smartphones are portable but not truly mobile because they support interaction only poorly while moving; for example, walking divides attention,²⁴ reduces accuracy,¹⁷ slows reading speed,²⁶ and impairs obstacle avoidance.³² We conducted multiple projects to improve interaction while walking, focusing on people’s abilities while on the go.

In our early exploration of walking user interfaces,¹⁵ we studied level-of-detail (LoD) adaptations, where the interface shown while a user was standing had high detail and the interface shown while a user was walking had low detail, with larger fonts and bigger targets. When a user moved from standing to walking and vice versa, the interface changed. We compared this adaptive interface to component static interfaces for both walking and standing, finding that walking increased task time for static interfaces by 18%, but with our adaptive interface, walking did not increase task time. We also found that the adaptive interface performed like its component static interfaces; that is, there was no penalty for the LoD adaptation.

In our subsequent project, called WalkType,¹² we made mobile touch-based keyboards almost 50% more accurate and 12% faster while walking. Touch-based features like finger

Table 2. Seven principles of ability-based design, updated and revised from previous versions.^{37,38}

	Principle	Description
Designer Stance (required)	Ability	Designers focus on users’ abilities, not disabilities, striving to leverage all that users can do in a given situation, context, or environment.
	Accountability	Designers respond to poor usability by changing systems, not users, leaving users as they are.
	Availability	Designers use affordable and available software, hardware, or other components acquirable through accessible means.
Adaptive or Adaptable Interface (optional)	Adaptability	Interfaces might be adaptive or adaptable to provide the best possible match to users’ abilities.
	Transparency	Interfaces might give users awareness of adaptive behaviors and what governs them and the means to inspect, override, discard, revert, store, retrieve, preview, alter, or test those behaviors.
Sensing and Modeling (optional)	Performance	Systems might sense, monitor, measure, model, display, predict, or otherwise utilize users’ performance to provide the best possible match between systems and users’ abilities.
	Context	Systems might sense, monitor, measure, model, display, predict, or otherwise utilize users’ situation, context, or environment to anticipate and accommodate effects on users’ abilities.

location, duration, and travel were combined with accelerometer features like signal amplitude and phase to train decision trees that reclassified wayward key-presses. WalkType effectively remedied a systematic inward rotation of the thumbs caused by whichever foot was moving forward as the user walked.

Performing input tasks is only one challenge while walking. Consuming output is another. In SwitchBack,¹⁹ an attention-aware system for smartphones, a smartphone’s front-facing camera was used to track eye-gaze position on the screen to aid task resumption. For example, when a user was reading and looked away, SwitchBack remembered the last-read line of text; when the user’s gaze returned to the screen, that same line was highlighted to draw the user’s attention for easy task resumption.

These three walking user interfaces exhibited all seven principles of ability-based design to varying degrees.

Global Public Inclusive Infrastructure

Ability-based design has been applied mostly at the level of individual systems and applications, but for greater impact, a new infrastructure that extends beyond the user’s own device is

needed. Although the Global Public Inclusive Infrastructure (GPII),^{34,35} with its cloud-based auto-personalization of information and communication technologies, was formulated independent of ability-based design, its objectives are the same—enable interfaces to be ideally configured to match each user’s situated abilities.

The GPII is built on three technological pillars.³⁵ The second, “auto-personalization,” is the one of interest here.^d Its long-term goal is to ensure that any digital interface a person encounters instantly changes to a form that can be understood and used by that person. The GPII’s auto-personalization capability uses a person’s needs and preferences, which are stored in the cloud or on a token, to automatically configure the interface of each device for that individual.^{34,36} Its “one size fits one” approach is designed to help each person have the “best fit” interface possible. Since interface flexibility on current devices and software is limited, GPII auto-personalization uses both built-in features and assistive technologies (AT) (on the device and in the cloud)

^d The two other pillars make it easy for people to determine what they need or prefer, ensuring solutions exist for everyone.


to achieve each best-fit interface. For example, accessibility features located in five layers—operating system features, installed AT, browser features, cloud AT, and Web app features—can be configured to work together to provide best-fit user interfaces, with features at each level being invoked (or not) in order to meet the user’s needs and preferences.

GPII auto-personalization supports interfaces that self-adapt, as well as configuration of interfaces and adaptations, to match a user’s needs. By combining auto-adjusting interfaces, preference-configured interfaces, and user-selected-and-configured AT, the GPII can function as a bridge among these approaches, maximizing the utility of each one for an individual at any point in time. The GPII also supports auto-configuration based on contextual changes.⁴⁰ The GPII thus meets all seven principles of ability-based design.


Taking Up the Challenge

Pursuing these and other projects, some patterns have emerged for us. For example, we noticed a perspective shift as we began to actively seek out the abilities people have, inspiring an openness to consider how we could create or change technologies to suit different abilities. We also noticed a seamlessness between designing for people with limited abilities and designing for people in ability-limiting situations. We realized accessibility is indeed a worthy goal for all users. Because we were looking to modify systems, not users, we deemphasized assistive hardware add-ons. Customization arose from a powerful sequence of sensing, modeling, and adapting; it also arose from support for end-user configurability, as with the U.S. Air Force cockpits mentioned earlier. We thus made our interactive systems more aware of their users and contexts.

Where does ability-based design go next? One way to answer is to treat the vision of ability-based design as a grand challenge and ask what it would take to create a world in which anyone, anywhere, at any time could interact with technologies that are ideally suited to his or her situated abilities. Achieving the “anyone any-



What would it take to create a world in which anyone, anywhere, at any time could interact with technologies that are ideally suited to his or her situated abilities?



where any time” part will require systemwide infrastructure of the kind pursued by the GPII. Ability-aware operating systems infused with SUPPLE-like user-interface generators could help create personalized applications. Improved sensing and modeling of users’ abilities and contexts, as in walking user interfaces, could enable mobile and wearable systems to better support diverse contexts of use. One challenge is to avoid explicit task-based training and calibration in favor of implicit observation and modeling from everyday use, as in Evans and Wobbrock⁵ and Gajos et al.⁸

To date, ability-based design has focused primarily on single-user experiences, but the social lives of users could also lend themselves to collaborative support. How should the abilities of a pair, group, team, crowd, or organization be considered? For service arrangements, what would it look like to have an ability-based design for services?

Moreover, abilities exist on many levels, from low-level sensorimotor and cognitive abilities, to mid-level abilities for daily living, to high-level social, occupational, professional, and creative abilities. Such abilities form a hierarchy paralleling Maslow’s hierarchy of needs,²⁰ whereby each need corresponds to an ability to meet it. Ability-based design seems applicable throughout such a hierarchy, but the range has yet to be explored.

Concerning “adaptivity,” providing each individual with a unique user interface raises several pragmatic issues, as in, say, authoring help documentation, provision of customer support, and making the design process of personalized experiences consistent with accepted design practice. These challenges are real but, as we discuss elsewhere,⁹ solvable.


With the vast range of human abilities from which to draw, adaptivity based on sensing and modeling is a powerful way to realize custom designs that, while inevitably imperfect, nonetheless provide good user-system fits at scale. Adaptive interfaces can remember users’ abilities and preferences and draw on them when generating interfaces for both familiar and unfamiliar systems, providing more satisfying and effective access

for each individual user. We thus see an important and continuing role for adaptivity and personalization within ability-based design.

We close with a quote from Frank Bowe (1947–2007), professor and disability-rights activist who helped instigate the Americans with Disabilities Act of 1990 (<https://www.ada.gov/>). Writing in *MIT Technology Review* in 1987, he emphasized the importance of focusing on what people are able to do, not on what holds people back:¹ “When society makes a commitment to making new technologies accessible to everyone, the focus will no longer be on what people cannot do, but rather on what skills and interests they bring to their work. That will be as it always should have been.”

We could not agree more.

Acknowledgments

We wish to thank our co-authors on the projects we covered here, especially Jeffrey Bigham, Leah Findlater, Jon Froehlich, Mayank Goel, Susumu Harada, Alex Mariakakis, Shwetak Patel, and Daniel S. Weld. This work was supported in part by the Mani Charitable Foundation and the National Science Foundation under grants IIS-0952786 and CNS-1539179. Any opinions, findings, conclusions or recommendations expressed in this work are those of the authors and do not necessarily reflect those of any supporter or collaborator. 

References

1. Bowe, F. Making computers accessible to disabled people. *MIT Technology Review* 90 (Jan. 1987), 52–59, 72.
2. Dourish, P. *Where the Action Is: The Foundations of Embodied Interaction*. MIT Press, Boston, MA, 2001.
3. Edwards, A.D.N. Computers and people with disabilities. Chapter 2 in *Extra-Ordinary Human-Computer Interaction: Interfaces for Users with Disabilities*, A.D.N. Edwards, Ed. Cambridge University Press, Cambridge, England, 1995, 19–43.
4. Engelbart, D.C. and English, W.K. A research center for augmenting human intellect. In *Proceedings of the AFIPS Fall Joint Computer Conference* (San Francisco, CA, Dec. 9–11). AFIPS, Los Alamitos, CA, 1968, 395–410.
5. Evans, A.C. and Wobbrock, J.O. Taming wild behavior: The input observer for obtaining text entry and mouse pointing measures from everyday computer use. In *Proceedings of CHI 2012* (Austin, TX, May 5–10). ACM Press, New York, 2012, 1947–1956.
6. Flatta, D.R., Reinecke, K., Gutwin, C., and Gajos, K.Z. SPRWeb: Preserving subjective responses to website colour schemes through automatic recolouring. In *Proceedings of CHI 2013* (Paris, France, Apr. 27–May 2). ACM Press, New York, 2013, 2069–2078.
7. Gajos, K.Z., Hurst, A., and Findlater, L. Personalized dynamic accessibility. *Interactions* 19, 2 (Mar.–Apr. 2012), 69–73.
8. Gajos, K.Z., Reinecke, K., and Herrmann, C. Accurate measurements of pointing performance in situ observations. In *Proceedings of CHI 2012* (Austin, TX, May 5–10). ACM Press, New York, 2012, 3157–3166.

9. Gajos, K.Z., Weld, D.S., and Wobbrock, J.O. Automatically generating personalized user interfaces with SUPPLE. *Artificial Intelligence* 174, 12–13 (Aug. 2010), 910–950.
10. Gajos, K.Z., Wobbrock, J.O., and Weld, D.S. Automatically generating user interfaces adapted to users' motor and vision capabilities. In *Proceedings of UIST 2007* (Newport, RI, Oct. 7–10). ACM Press, New York, 2007, 231–240.
11. Gajos, K.Z., Wobbrock, J.O., and Weld, D.S. Improving the performance of motor-impaired users with automatically generated, ability-based interfaces. In *Proceedings of CHI 2008* (Florence, Italy, Apr. 5–10). ACM Press, New York, 2008, 1257–1266.
12. Goel, M., Findlater, L., and Wobbrock, J.O. WalkType: Using accelerometer data to accommodate situational impairments in mobile touch-screen text entry. In *Proceedings of CHI 2012* (Austin, TX, May 5–10). ACM Press, New York, 2012, 2687–2696.
13. Hummels, C. and Lévy, P. Matter of transformation: Designing an alternative tomorrow inspired by phenomenology. *Interactions* 20, 6 (Nov.–Dec. 2013), 42–49.
14. Kane, S.K., Bigham, J.P., and Wobbrock, J.O. Slide Rule: Making mobile touch screens accessible to blind people using multi-touch interaction techniques. In *Proceedings of ASSETS 2008* (Halifax, Nova Scotia, Canada, Oct. 13–15). ACM Press, New York, 2008, 73–80.
15. Kane, S.K., Wobbrock, J.O., and Smith, I.E. Getting off the treadmill: Evaluating walking user interfaces for mobile devices in public spaces. In *Proceedings of MobileHCI 2008* (Amsterdam, the Netherlands, Sept. 2–5). ACM Press, New York, 2008, 109–118.
16. Keates, S., Clarkson, P.J., Harrison, L.-A., and Robinson, P. Towards a practical inclusive design approach. In *Proceedings of CUU 2000* (Arlington, VA, Nov. 16–17). ACM Press, New York, 2000, 45–52.
17. Lin, M., Goldman, R., Price, K.J., Sears, A., and Jacko, J. How do people tap when walking? An empirical investigation of nomadic data entry. *International Journal of Human-Computer Studies* 65, 9 (Sept. 2007), 759–769.
18. Mace, R.L., Hardie, G.J., and Place, J.P. Accessible environments: Toward universal design. Chapter 8 in *Design Intervention: Toward a More Humane Architecture*, W.E. Preiser, J.C. Vischer, and E.T. White, Eds. Van Nostrand Reinhold, New York, 1991, 155–176.
19. Mariakakis, A., Goel, M., Aumi, M.T.I., Patel, S.N., and Wobbrock, J.O. SwitchBack: Using focus and saccade tracking to guide users' attention for mobile task resumption. In *Proceedings of CHI 2015* (Seoul, South Korea, Apr. 18–23). ACM Press, New York, 2015, 2953–2962.
20. Maslow, A.H. A theory of human motivation. *Psychological Review* 50, 4 (July 1943), 370–396.
21. Mott, M.E., Vatavu, R.-D., Kane, S.K., and Wobbrock, J.O. Smart touch: Improving touch accuracy for people with motor impairments with template matching. In *Proceedings of CHI 2016* (San Jose, CA, May 7–12). ACM Press, New York, 2016, 1934–1946.
22. Newell, A.F. Extra-ordinary human-computer interaction. Chapter 1 in *Extra-Ordinary Human-Computer Interaction: Interfaces for Users with Disabilities*, A.D.N. Edwards, Ed. Cambridge University Press, Cambridge, England, 1995, 3–18.
23. Newell, A.F. and Gregor, P. User-sensitive inclusive design: In search of a new paradigm. In *Proceedings of CUU 2000* (Arlington, VA, Nov. 16–17). ACM Press, New York, 2000, 39–44.
24. Oulasvirta, A., Tamminen, S., Roto, V., and Kuorelahti, J. Interaction in 4-second bursts: The fragmented nature of attentional resources in mobile HCI. In *Proceedings of CHI 2005* (Portland, OR, Apr. 2–7). ACM Press, New York, 2005, 919–928.
25. Rose, L.T. *The End of Average: How We Succeed in a World That Values Sameness*. HarperCollins, New York, 2015.
26. Schildbach, B. and Rukzio, E. Investigating selection and reading performance on a mobile phone while walking. In *Proceedings of MobileHCI 2010* (Lisbon, Portugal, Sept. 7–10). ACM Press, New York, 2010, 93–102.
27. Sears, A. and Young, M. Physical disabilities and computing technologies: An analysis of impairments. Chapter 25 in *The Human-Computer Interaction Handbook, First Edition*, J.A. Jacko and A. Sears, Eds. Lawrence Erlbaum, Hillsdale, NJ, 2003, 482–503.
28. Shinohara, K. and Wobbrock, J.O. In the shadow of misperception: Assistive technology use and social interactions. In *Proceedings of CHI 2011* (Vancouver, BC, Canada, May 7–12). ACM Press, New York, 2011, 705–714.
29. Shinohara, K. and Wobbrock, J.O. Self-conscious or self-confident? A diary study conceptualizing the social accessibility of assistive technology. *ACM Transactions on Accessible Computing* 8, 2 (Jan. 2016), article 5.
30. Stienstra, J. Embodying phenomenology in interaction design research. *Interactions* 22, 1 (Jan.–Feb. 2015), 20–21.
31. Sutherland, I.E. Sketchpad: A man-machine graphical communication system. In *Proceedings of the AFIPS Spring Joint Computer Conference* (Detroit, MI, May 21–23). AFIPS, Santa Monica, CA, 1963, 329–346.
32. Vadas, K., Patel, N., Lyons, K., Starner, T., and Jacko, J. Reading on-the-go: A comparison of audio and handheld displays. In *Proceedings of the Eighth Conference on Human-Computer Interaction with Mobile Devices and Services* (Helsinki, Finland, Sept. 12–15). ACM Press, New York, 2006, 219–226.
33. Vanderheiden, G.C. Anywhere, anytime (+anyone) access to the next-generation WWW. *Computer Networks and ISDN Systems* 29, 8–13 (Sept. 1997), 1439–1446.
34. Vanderheiden, G. and Treviranus, J. Creating a global public-inclusive infrastructure. In *Proceedings of the International Conference on Universal Access in Human-Computer Interaction, Lecture Notes in Computer Science, Vol. 6765* (Orlando, FL, July 9–14). Springer, Berlin, Germany, 2011, 517–526.
35. Vanderheiden, G., Treviranus, J., Ortega-Moral, M., Peissner, M., and de Lera, E. Creating a global public-inclusive infrastructure (GPII). In *Proceedings of the International Conference on Universal Access in Human-Computer Interaction, Lecture Notes in Computer Science, Vol. 8516* (Heraklion, Crete, Greece, June 22–27). Springer, Berlin, Germany, 2014, 506–515.
36. Vanderheiden, G.C., Treviranus, J., Usero, J.A.M., Bekiaris, E., Gemou, M., and Chourasia, A.O. Auto-personalization: Theory, practice and cross-platform implementation. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (Boston, MA, Oct. 22–26). Human Factors and Ergonomics Society, Santa Monica, CA, 2012, 926–930.
37. Wobbrock, J.O. Improving pointing in graphical user interfaces for people with motor impairments through ability-based design. Chapter 8 in *Assistive Technologies and Computer Access for Motor Disabilities*, G. Kouroupetoglou, Ed. IGI Global, Hershey, PA, 2014, 206–253.
38. Wobbrock, J.O., Kane, S.K., Gajos, K.Z., Harada, S., and Froehlich, J. Ability-based design: Concept, principles, and examples. *ACM Transactions on Accessible Computing* 3, 3 (Apr. 2011), article 9.
39. World Health Organization. *Document A29/INFDOCI/1*, 1976. Geneva, Switzerland.
40. Zimmermann, G., Vanderheiden, G.C. and Strobbe, C. Towards deep adaptivity: A framework for the development of fully context-sensitive user interfaces. In *Proceedings of the International Conference on Universal Access in Human-Computer Interaction, Lecture Notes in Computer Science, Vol. 8513* (Heraklion, Crete, Greece, June 22–27). Springer, Berlin, Germany, 2014, 299–310.

Jacob O. Wobbrock (wobbrock@uw.edu) is a professor of human-computer interaction in the Information School and, by courtesy, in the Paul G. Allen School of Computer Science & Engineering at the University of Washington, Seattle, WA, USA.

Krzysztof Z. Gajos (kgajos@eecs.harvard.edu) is a Gordon McKay Professor of Computer Science at the Harvard Paulson School of Engineering and Applied Sciences at Harvard University, Cambridge, MA, USA.

Shaun K. Kane (shaun.kane@colorado.edu) is an assistant professor in the Department of Computer Science and, by courtesy, in the Department of Information Science, at the University of Colorado Boulder, Boulder, CO, USA.

Gregg C. Vanderheiden (GreggVan@umd.edu) is a professor and Director of the Trace R&D Center in the College of Information Studies at the University of Maryland College Park, MD, USA, and Co-Director of the Global Public Inclusive Infrastructure.