Generative Theories of Interaction

MICHEL BEAUDOUIN-LAFON*, Université Paris-Saclay, CNRS, Inria, France
SUSANNE BØDKER*, Aarhus University, Denmark
WENDY E. MACKAY*, Université Paris-Saclay, CNRS, Inria, France

Although Human-Computer Interaction research has developed various theories and frameworks for analyzing new and existing interactive systems, few address the *generation* of novel technological solutions, and new technologies often lack theoretical foundations. We introduce *Generative Theories of Interaction*, which draw insights from empirical theories about human behavior in order to define specific concepts and actionable principles, which in turn serve as guidelines for analyzing, critiquing and constructing new technological artifacts. After introducing and defining Generative Theories of Interaction, we present three detailed examples from our own work: Instrumental Interaction, Human-Computer Partnerships, and Communities & Common Objects. Each example describes the underlying scientific theory and how we derived and applied HCI-relevant concepts and principles to the design of innovative interactive technologies. Summary tables offer sample questions that help analyze existing technology with respect to a specific theory, critique both positive and negative aspects, and inspire new ideas for constructing novel interactive systems.

CCS Concepts: • **Human-centered computing → HCI theory, concepts and models**.

Additional Key Words and Phrases: Theory; Generative principles; Generative theory

ACM Reference Format:
Michel Beaudouin-Lafon, Susanne Bødker, and Wendy E. Mackay. 2021. Generative Theories of Interaction. *ACM Trans. Comput.-Hum. Interact.* 28, 6, Article 45 (September 2021), 54 pages. https://doi.org/10.1145/3468505

1 INTRODUCTION

Human-Computer Interaction (HCI) is a multi-disciplinary research discipline that draws from the *natural sciences*, which seek to explain naturally occurring phenomena; from the *social sciences*, which seek to explain human phenomena; from *design*, which seeks to explain how designs emerge to support human activity; and from *engineering*, which seeks to enable and guide the creation of innovative technology. HCI has produced a large body of such knowledge, including a deeper understanding of the behavioral phenomena involved in human-computer interaction; the creation of a variety of novel devices and interaction techniques; and extensive empirical know-how about the design process.

45

Mackay & Fayard [112] describe the relationship between theory and empirical studies in the natural sciences, where scientific research involves a back-and-forth cycle between theoretical constructs and empirical observation. They note that HCI research includes a third element — artifact design — making it what Simon [141] calls a "science of the artificial". Fig. 1 shows how a specific HCI research project may start at any level, whether theory development, artifact design, or empirical observation, and proceed along different paths through other types of research activity.

*Authors are listed in alphabetical order and contributed equally to this work.

Authors' addresses: Michel Beaudouin-Lafon, mbl@lri.fr, Université Paris-Saclay, CNRS, Inria, LISN, Bât 650, Université Paris-Saclay, Orsay, F-91400, France; Susanne Bødker, bodker@cs.au.dk, Aarhus University, Aabogade 34, Aarhus, DK-8200, Denmark; Wendy E. Mackay, mackay@lri.fr, Université Paris-Saclay, CNRS, Inria, LISN, Bât 650, Université Paris-Saclay, Orsay, F-91400, France.

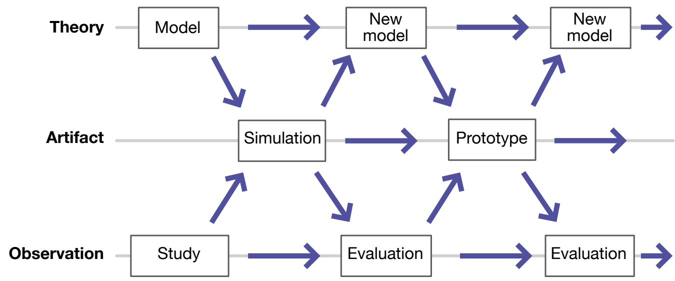

Fig. 1. HCI research involves cycling through theory development, artifact design and empirical observation — it may start on any level, and involves a full back-and-forth cycle through the three levels. (Adapted from Mackay & Fayard [112].)

Legitimate outcomes of HCI research may thus include new empirical discoveries, innovative interactive artifacts, or novel theory.

When we examine HCI theory from this perspective, we argue that a piece is missing: we lack *generative* theories that provide a direct link from empirically based theories in the natural and social sciences to *HCI-specific* constructs that suggest and inspire novel forms of human-computer interaction. This paper introduces the concept of *Generative Theories of Interaction*, which provide analytical, critical and constructive power for explaining existing interactive systems and inspiring new ones.

We engage with generative theory at multiple levels, from understanding human interaction with technology, to identifying HCI-specific concepts, to articulating generative principles that operationalize these concepts, and finally to testing these theories in real-world applications. Our primary audience consists of HCI researchers who are interested in either building new theory or reframing existing theories in new and generative ways. We focus specifically on those HCI researchers who achieve this by investigating particular groups of users or exploring how to improve interactive systems. Although we do not specifically address design practitioners, we expect that Generative Theories of Interaction will help HCI researchers transmit their results by teaching students who later become design practitioners and apply these concepts and principles in real-world settings.

This paper first defines *Generative Theories of Interaction*, and explains how they draw inspiration from well-established scientific theories that they then apply to HCI-specific research questions. After positioning Generative Theories of Interaction with respect to related work, we introduce three specific generative theories of interaction and show how we applied their generative principles to HCI research questions, to produce new empirical findings, novel artifacts and refined theory. We conclude with a discussion of the value of Generative Theories of Interaction and directions for future work.

2 GENERATIVE THEORIES OF INTERACTION

We view HCI research not so much as problem solving [77], but rather as generating insights and opening up new possibilities for design in the face of constantly changing technology. Critically, individual users' interactions with technology are dynamic, and evolve as the user's practice, experience and specific, real-world context change over time [23, 31]. Since the user experience is never definitive, we need HCI theories, methods and tools that embrace these dynamics, both to identify new design ideas and to create new artifacts that assume that interaction changes over time.

This resonates with a number of familiar perspectives in HCI, such as Schön's [137] reflective practitioner and activity theoretical HCI [11, 35], as well as Kaufman's [88] concept of the "adjacent possible" from biology. Johnson [85] generalized Kaufman's concept to address any generative process, describing it as: "*a kind of shadow future, hovering on the edges of the present state of things, a map of all the ways in which the present can reinvent itself ... [it] captures both the limits and the creative potential of change and innovation.*" When applied to HCI research, each new insight about users or technological invention has the potential to enlarge the adjacent possible surrounding interactive systems, creating an ever-increasing design space of possibilities to explore.

2.1 Definition

We introduce *Generative Theories of Interaction* to encourage this exploration of the "adjacent possible": Each generative theory of interaction builds upon ideas from empirically based scientific theory, and introduces HCI-specific concepts and actionable principles that, when applied to specific research questions, help generate new insights about users and inspire novel design directions. Each generative theory of interaction offers what Brown [41] calls a *conceptual lens* for understanding interaction within a specified scope, helping HCI researchers to conceptualize and articulate research questions, and to bridge the gap between HCI research and artifact design. The longer term goal is to help transform the ever-growing body of HCI knowledge into actionable principles that can be taught to students and practitioners, and productively guide the exploration and production of novel technological artifacts.

We define a Generative Theory of Interaction as a construct that is:

(1) grounded in a theory of human activity and behavior with technology;
(2) amenable to analytical, critical and constructive interpretation; and
(3) actionable through the theory's concepts and generative principles.

(1) Generative Theories of Interaction should not be based solely on intuition or anecdotal evidence, but grounded in descriptive, predictive or prescriptive theories of human activity and behavior from the natural and social sciences, especially biology, experimental psychology, sociology and anthropology. This helps ensure that we "stand on the shoulders of giants" [115] to build a relevant body of knowledge. Although a variety of theories provide suitable foundations, we particularly value theories that address the dynamic aspects of human behavior in relation to technology.

(2) Generative Theories of Interaction involve three successive lenses to help HCI researchers both understand current use of technology and suggest future interactive systems: The *analytical* lens provides a description of current use and practice; the *critical* lens assesses both the positive and negative aspects of a system given different needs and contexts of use, thus providing avenues for improvement or re-design; and the *constructive* lens inspires new ideas relative to the critique, expressed in terms of the generative theory's concepts and principles.

(3) The three lenses are enabled through the theory's concepts and principles. We do not intend generative theory to somehow *automatically* generate new insights or possibilities, e.g. with generative algorithms. Instead, each generative theory's concepts and principles provide tools that enable the principled assessment, exploration and expansion of the design space related to a particular research problem and its possible solutions.

Figure 2 shows how concepts and generative principles associated from a particular generative theory of interaction enable analytical, critical and constructive lenses for analyzing existing and new artifacts. Observations of the use of existing artifacts can also inspire new generative principles, which then inform the design of new artifacts.

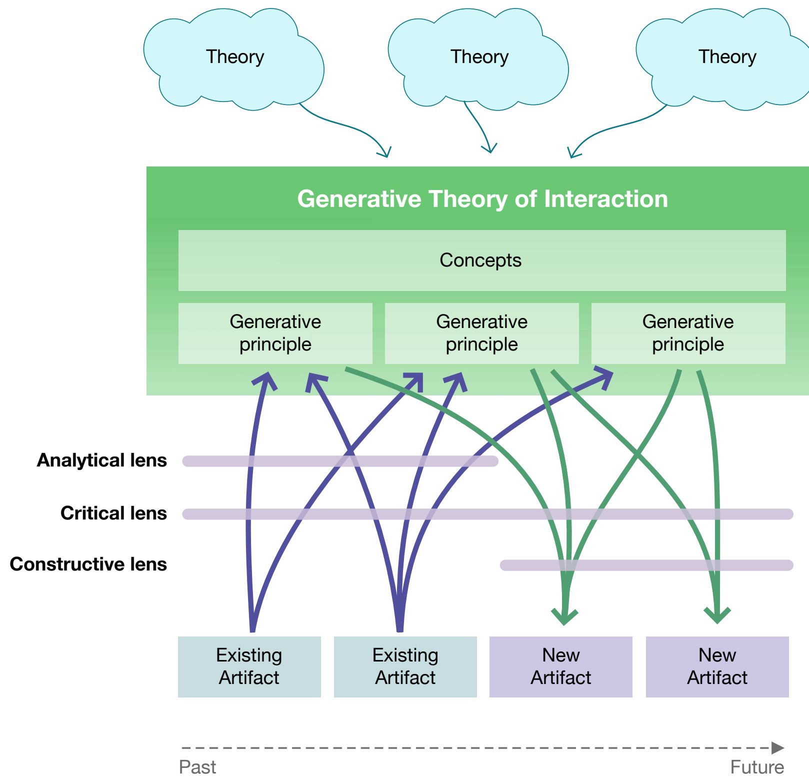

Fig. 2. A Generative Theory of Interaction is grounded in existing theories of human activity and behavior and consists of concepts and generative principles that can be used to analyze, critique and construct interactive artifacts.

2.2 Scope

Theory comes in many forms, with different emphases on concepts, principles or empirical grounding, and different goals with respect to precision, context, and generality [136]. HCI research includes a particularly wide variety of underlying theory, ranging from predictive Fitts' law [60] analysis of pointing behavior, typically studied with lab-based hypothesis-testing experiments, to qualitative assessments of the human experience and technology in society, typically studied in the field. We borrow Mackay & Fayard's [112] characterization of scientific theories that involve an interplay between theoretical and empirical investigation. The associated research activities may include qualitative and/or quantitative evidence and result in descriptive, predictive and/or prescriptive theory. As shown in Figure 1, they may start with empirical results that lead to testable theories or with theories subject to further empirical investigation.

Figure 3, adapted from Mackay & Fayard [112], illustrates how generative theories of interaction link theory and artifact design (upper green area), just as HCI design methods [111] link artifact design and empirical observation (lower blue area). They encourage a change-oriented perspective by providing HCI researchers with conceptual tools for analyzing technologies in use or exploring novel future solutions. Although creating or refining Generative Theories of Interaction is clearly an activity best suited for HCI researchers, not practitioners, we are confident that individual generative theories of interaction can be applied productively to real-world design practice.

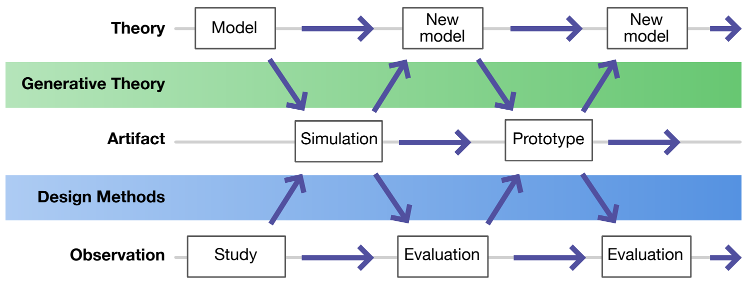

Fig. 3. Generative theory involves applying theory to artifact design (green); whereas design methods involve empirical observation of the use of artifacts (blue). HCI researchers engage in both. (Adapted from Mackay & Fayard [112].)

Instead of seeking universal generality, generative theories of interaction should be bounded in scope and address a clearly defined context or set of issues, at an intermediate level of abstraction. The emphasis here is on what might be called "micro" HCI research, whose goal is to produce specific outcomes within an interaction design process, as opposed to a more "macro" view that seeks broader explanations of human behavior and activity with technology.

Note that we do not define specific criteria for *which* theories are eligible to ground a generative theory of interaction, beyond ensuring that they are based on an empirical foundation that justifies their core concepts and principles. Nor do we claim that all HCI theories *should* be generative, or that all HCI designs should result from applying generative theory. While we believe that generative theories can consolidate HCI knowledge in useful ways, we also recognize the many other valid ways of conducting research and design in HCI. We cannot speak to the appropriateness of these other types of theory for creating Generative Theories of Interaction, but they clearly offer potential for future research.

3 RELATED WORK

Human-Computer Interaction is a multidisciplinary field that includes aspects of natural science, social science, design, and technology development, with correspondingly different approaches to theory. Neuman [120, p. 30] describes theory as "*a system of interconnected ideas that condense and organize knowledge*". However the term "theory" has multiple definitions that may vary across or even within research disciplines. For example, Abend [1] offers a nuanced analysis of seven distinct meanings of the term "theory" in Sociology, and the confusion this has caused. Here, we are explicitly interested in empirically based theories that involve qualitative and/or quantitative data. This makes the first three of Abend's theory definitions suitable: 1) establishing relationships between two or more variables; 2) explaining particular social phenomena; and 3) describing the meaning or significance of phenomena in the social world.

To bridge the resulting cross-disciplinary gaps, venues such as the flagship ACM/CHI conference ask authors to consider the relationship between empirical findings and technological novelty. Empirical contributions are encouraged to include "implications for design", despite Dourish's [55] argument that few social scientists have the necessary technical or design background. Technical contributions are encouraged to demonstrate the effectiveness of the proposed technique or system, which is tricky when behavioral measures fall outside the standard experimental paradigm. Despite these issues, maintaining this on-going tension between conceptually, empirically and technically oriented papers has served the HCI community well.

A small percentage of the HCI literature proposes or discusses theory explicitly, with different approaches, motivations, means and disciplinary perspectives. However, authors rarely try to explain why we need theory, and instead offer theory as a tool for categorizing and describing different aspects of HCI. We review two main strands of theorizing in HCI that relate to our approach, namely the development of theory to better understand how people interact with technology, and theory that supports the innovation process.

3.1 Theory related to understanding human interaction with technology

Carroll [46] and Rogers [133] each describe and assess common theories in HCI research. Interestingly, the vast majority of these theories have their roots outside of HCI, such as applied perception, motor learning, distributed cognition, ecological psychology, ethnomethodology, and activity theory. However, several of these theories were developed in conjunction with ongoing HCI research. For example, qualitative descriptive theories, such as distributed cognition [78] and situated action [144, 145] were originally grounded in anthropology. However, each was developed in the context of HCI to challenge the disembodied nature of 1980's cognitive science and what Suchman calls the "plans" that characterize rule-based artificial intelligence research. Similarly, activity-theoretic HCI [87] originally draws from activity theory, but continues as a key HCI theory [104] for describing human behavior with respect to technology. More quantitative theories include Card, Moran and Newell's GOMS (goals-operators-methods-selection rules) model [44] and Appert's CIS (Complexity of Interaction Sequences) [5], both of which are framed in terms of causal inference, with methodology from experimental psychology to design controlled laboratory experiments and develop quantitative HCI-related theory.

Rogers [133] distinguishes between first generation theories that aim to be prescriptive by directly advising how to construct a user interface (p.91), and newer, second-generation theories that do not share this aim. She suggests that we consider these theories in terms of their formative and generative power: to build overarching frameworks for HCI is to provide a set of concepts "*from which to think about the design and use of interactive systems.*" The idea is to "*stimulate new ideas, concepts, and solutions. In this sense, the theory can serve both formative and generative roles in design*" (p. 121). We agree with Rogers' assessment that the formative and generative roles of theory are fundamentally different, and further, we suggest explicit strategies for creating generative theory.

Gergen [65] also argues in favor of generative theory that "*can provoke debate, transform social reality, and ultimately serve to reorder social conduct.*" This is a wider and more radical way of thinking about generative theory than Rogers' more design-oriented approach, but nonetheless equally relevant, because it suggests that generative theory should not only help practitioners be creative, but also reframe research accordingly. Gergen, who coined the term "social constructivism" in psychology, belongs to a group of authors, including Shotter [140] and Garfinkel [64], who were inspired by Wittgenstein to consider a generative stance. Their work connects this perspective with foundational work in HCI, e.g. Ehn [57]. Gergen further suggests that we "*consider competing theoretical accounts in terms of their generative capacity, that is, the capacity to challenge the guiding assumptions of the culture, to raise fundamental questions regarding contemporary social life, to foster reconsideration of that which is "taken for granted", and thereby to furnish new alternatives for social action.*" This opens the possibility of generative engagement with one theory, as well as using and comparing the generative potential of parallel or competing theories. Like Gergen, our goal is to reconsider that which is "*taken for granted*" and "*thereby to furnish new alternatives*", although at a different scale than social and cultural theories.

As suggested by Rogers [133, 134] and Shneiderman [139], generative theories focus on enhancing creativity in interaction design by employing grounded concepts, methods and principles. They

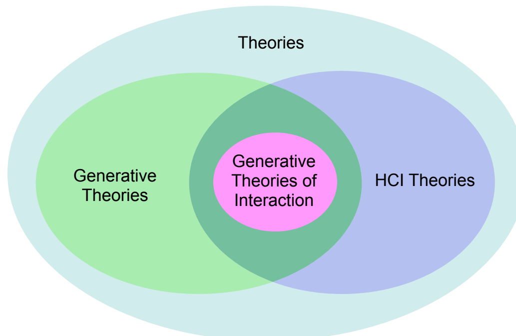

Fig. 4. Generative Theories of Interaction sit at the intersection between generative theory and HCI-specific theory, specifically targeting the interaction between human users and interactive technology within the general space of theory in the natural and social sciences.

also inform technological alternatives both theoretically and empirically, thus leading to more principled, coherent designs.

Compared with previous uses of generative theory in HCI, Generative Theories of Interaction feature an explicit link with their existing grounding theories, and include explicit concepts and generative principles that can be applied to specific research questions. They thus constitute a well-defined subset of the generative theories as defined by Rogers [133] (see Figure 4).

3.2 Theory related to the innovation process

A key role for generative theories is to help spark innovation and suggest new directions for designs that better meet the needs of users in different contexts. Gaines [63] proposes the BRETAM model of the information systems innovation process: After an initial **B**reakthrough that changes how we think of technology, other researchers **R**eplicate or slightly modify the ideas as point designs. Next comes **E**mpricism, where the community codifies the work into lessons learned and design rules. This makes it possible to generate **T**heory, with causal reasoning and testable hypotheses, followed by **A**utomation, as theories are accepted, and **M**aturity when the theories are in widespread use. Greenberg [68] argues that HCI has begun moving along Gaines's BRETAM trajectory, but that many areas are "*throttled at the replication stage*", minimizing their impact on product invention and innovation.

Even so, a few HCI areas have moved further along Gaines's trajectory, such as Wellner's Digital Desk [153], a **B**reakthrough demonstration of how to seamlessly link digital information and physical paper. This inspired research in multiple research areas, including tangible computing, mixed and augmented reality, for which it received ACM/UIST's first lasting impact award. Various researchers **R**eplicated the digital desk or explored variations. As the necessary technology decreased in size and increased in power, new opportunities arose for "intelligent" objects and "wearables" and mixed physical-virtual environments, which Mackay [112] characterizes as points within a design space where technology can augment the user, the object, or the environment or any combination of the three (**E**mpiricism). Early formulations of **T**heory in that space include Ishii & Ullmer's Tangible Bits [80], which capture what is now referred to as tangible computing. However, such theories remain rare and are not at the stage where they are **A**utomated or universally accepted, much less sufficiently **M**ature for widespread use by practitioners.

Kostakos [96] analyzes the HCI literature in terms of "motor themes", and shows how researchers who publish at CHI conferences rarely focus on a particular topic to advance it. Instead, they move from topic to topic without further substantial development. HCI as a field arguably overvalues **B**reakthroughs, with little reward for **R**eplication and **E**mpiricism, much less **T**heory. Unfortunately, Kostakos [96] offers little help in framing and motivating these motor themes and theoretical contributions. Kostakos' analysis led Oulasvirta & Hornbæk [127] to view HCI research as problem solving, offering an interesting take on how empirical, constructive and conceptual research (should) play together as categories in HCI research:

> Conceptual problems are non-empirical; they involve issues in theory development in the most general sense [...] Conceptual problems might involve difficulties in explaining empirical phenomena, nagging issues in models of interaction, or seeming conflicts between certain principles of design [...] Work on a conceptual research problem is aimed at explaining previously unconnected phenomena occurring in interaction. (Oulasvirta & Hornbæk [127, page 4958])

Following this work, Hornbæk & Oulasvirta [77] define the term "interaction" by examining seven key concepts that highlight how conceptual research has developed in HCI over the past few decades. They define the "constructive power" of an HCI theory as "*the scope, validity, and practical relevance of the counterfactual reasoning it permits that links the conditions of interaction and its events*" [77, page 5048]. While this is more restrictive than our notion of a constructive lens in generative theories of interaction, we share their concern that over-emphasizing empirical and constructive research at the expense of conceptual research contributes to the paucity of abstraction and theory in the HCI literature, and agree that combining conceptual with empirical and constructive research is needed to advance the field.

Another perspective is Höök & Löwgren's [76] "strong concepts", which generate intermediate-level knowledge and play a direct role in the creation of new designs. Strong concepts act as only one of several possible intermediate-level forms of knowledge that can emerge from design-oriented HCI research. Others include HCI-specific design patterns [53] from the software engineering literature. We see these contributions as possible building blocks for the concepts and principles of future generative theories of interaction.

With a focus on creating building blocks, Bødker [22] offers a conceptual toolbox based on situation concepts and technological constructs that transfer the lessons learned from two large European research projects to inform specific designs, provoke ideas during the design process, and encourage change. With a similar focus on making change, and based on extensive interviews with HCI researchers and designers, Zimmerman et al. [158] propose four criteria for evaluating interaction design research contributions: process, invention, relevance and extensibility. Unlike the quest for validity in the behavioral sciences, they argue that interaction designers should demonstrate how their contribution relates to and improves upon the current state of the art and how others can leverage the knowledge gained, similar to the replication and experience stages of Gaines's BRETAM model [63]. Although these and similar efforts suggest useful generalizations from design work, they offer few clues as to how other interaction designers might actually accomplish these goals. Therefore we do not consider them "actionable" in our definition of a Generative Theory of Interaction.

By contrast, Lottridge & Mackay's [107] "generative walkthroughs", which link socio-technical principles directly to story-based design prototypes, explicitly bridge the gap between theory and design. After learning principles such as distributed cognition and situated action, designers systematically critique each step of a storyboard with respect to each principle, and then apply those principles to inspire new design alternatives. This is closest to our approach, but lacks the transition between high-level external theory and HCI-specific theory.

In summary, Generative Theories of Interaction explicitly target the innovation process in a research setting. Compared with the work outlined above, they strive to operationalize theoretical constructs into concepts and generative principles that can then be applied to specific research questions, providing both a path from theory to artifact and a principled method for exploring the research design space.

4 THREE EXAMPLES OF GENERATIVE THEORIES OF INTERACTION

In order to illustrate the notion of Generative Theory of Interaction, we present three examples are drawn from research carried out by the three co-authors over many years. Each follows a common structure that begins by briefly introducing a specific generative theory of interaction and its theoretical foundations, and explains how it influenced the core HCI *concepts*. We then present a set of actionable *generative principles* for analyzing and critiquing existing interactive technologies, showing how they lead to new insights about users and generate ideas for innovative interactive systems that address these critiques. Each project began with observation of or interviews with target users to better understand the initial design problem. We then systematically applied the principles from the associated generative theory of interaction to inspire new technology designs, which we evaluated relative to the users' needs.

All three examples are based on our common interest in a particular phenomenon in HCI, namely understanding artifacts as being created by human beings with future use in mind and used by people to achieve certain goals. The artifact therefore plays the role of a *tool* or *mediator* among people involved in the activity. In HCI, the tool perspective dates back to the 1980s and the extensive discussion on automation and Artificial Intelligence that took place at the time. For example, Weizenbaum [152] analyzed computing as a means, not an end for human beings, whereas Winograd & Flores [154] identified the fundamental role of tools in human action, and how humans generate the world by developing both tools and their use, emphasizing the importance of breakdowns as sources of learning and the shift between tool and objects. Ehn & Kyng [58] and Göranzon [67] promoted skill as an essential quality in developing the tool perspective, pointing out that human tool use does not happen in isolation but is situated in human practices and materials. Several authors were inspired by Heidegger's discussions of how the tool, in use, extends the human body [74] and the dialectics between the tool "ready-at-hand", i.e. that extends the hand, vs. "present-at-hand", i.e. being the object of interest, developing over time in the hermeneutic circle. While the three example theories presented here share these roots, they are in no way a defining feature of Generative Theories of Interaction.

5 EXAMPLE 1: INSTRUMENTAL INTERACTION

The first example targets traditional desktop interfaces and focuses on the design of more efficient and powerful interaction techniques for graphical user interfaces. The Graphical User Interface (GUI) was created in the late seventies [84] to help executive secretaries carry out office tasks. Forty years later, computers come in many forms and are used for a wide range of tasks by a wide variety of users, yet the most common interfaces are still based on applications, documents, files and folders, with the same menus, buttons and dialog boxes and the same input devices as the early systems. GUIs are obviously limited in the face of the new challenges of ever larger and more complex data and the wide range of uses and users of computers.

GUIs are based on the principles of *direct manipulation* [79, 138] where physical actions on visual representations of the objects of interest replace the text-based commands of earlier command-line interfaces. However, direct actions in current GUIs are typically limited to selecting content, e.g. with a click or a tap, and moving it, with a drag. More complex commands use widgets such as menus, scrollbars and dialog boxes that break the fundamental principle of "*direct manipulation of* *the object of interest*" [138]. They enact a form of *mediated interaction* where the user's actions are transformed and interpreted before being applied to the target object. Some techniques, such as dialog boxes, create a strong decoupling between the users' actions and their effects, while others, such as the use of tools in a tool palette, resemble the familiar use of physical tools, such as using a pencil or a brush to draw or paint.

Instrumental Interaction [13] was borne out of the observation that our interactions in the physical world are also often mediated by tools, devices and instruments, and are rarely direct. The goal of Instrumental Interaction is therefore to acknowledge this mediated interaction in the digital world and to give the mediators, called *instruments*, the status of first-class objects in the interface.

5.1 Theoretical Foundations

While humans are not the only species that use tools, they are unique in their ability to actively and intensively create new ones. Evidence of tool use by early humans dates back at least 3.3 million years [72], including traces of older adults training younger ones to create and use tools. Throughout human history all the way to modern life, human activity has been mediated by physical tools that extend and enhance our capabilities, or simply make life easier. Surprisingly, relatively few studies of human tool use can be found in the Psychology and HCI literature.

5.1.1 Human tool use in Psychology. In his theory of affordances, Gibson [66] mentions tools only briefly but makes a key observation: "*When in use, a tool is a sort of extension of the hand, almost an attachment to it or a part of the user's own body, and thus is no longer a part of the environment of the user.*" [66, p40] For Gibson, "*The affordances of the environment are what it offers the animal, what it provides or furnishes, either for good or ill.*" [66, p127, emphasis by the author]. Holding a tool thus redefines the affordances of the environment, since it changes the person's capabilities. Recent work supports this theory that a tool becomes an extension of the body schema, as if it were an integral part of the body [89]. This property provides a psychologically and cognitively sound basis for creating extensible systems, where new digital tools reveal new affordances.

Cognitive scientists have also studied human tool use to understand the cognitive processes at play when selecting and using tools. Osiurak et al. [126] contrast two competing views of tool use. The computational approach states that perceived properties of the tool are translated into physical actions for use, whereas the ecological approach states that the affordances of the tool (what can be done with it) are perceived directly. More recently, Osiurak [125] reports on experiments with patients suffering from several forms of apraxia (action disorganization syndrome) that suggest that tool use involves "technical reasoning", a specific type of reasoning that complements the direct perception of what the tool can do.

Technical reasoning is based on abstract technical laws, which, unlike skills acquired through procedural learning, do not require constant reinforcement. For example, people do not forget how to ride a bike, even after many years without practice, but commonly forget simple procedures, such as changing the time on a digital clock. Technical reasoning relies on identifying the technical properties of objects and their appropriate combination to perform a task and reach the desired goal. It explains unanticipated uses of objects as tools, such as using a knife to undo a screw when a screwdriver is not available. Both procedural learning and technical reasoning are clearly useful in everyday life. However, given that current computer systems are heavily based on procedural learning, we focus on the ability of technical reasoning to enable appropriation.

This difference between procedural learning and technical reasoning can be a powerful basis for creating (digital) tools that are more "natural", in the sense that they do not need constant re-learning. An interesting and open research question is to identify which types of digital tools, if any, elicit technical reasoning versus procedural learning. A sensible hypothesis is that the digital tools found in tool palettes are more likely to involve technical reasoning while the use of menus and dialog boxes seem more likely to involve procedural learning.

5.1.2 Human tool use in HCI. As mentioned in the previous section, the tool perspective dates back to the 1980s with the works of, e.g., Weizenbaum [152], Winograd & Flores [154] and Ehn [57]. Ehn & Kyng [58] and Göranzon [67] see human craftmanship and skill as essential qualities in developing the tool perspective, pointing out that human tool use does not happen in isolation but is situated in human practices and materials. Surprisingly however, the role of tools in human-artifact interaction is absent from seminal works in HCI such as Card, Moran & Newell [44], Suchman [145] or Norman [123].

In their taxonomy of the concept of interaction, Oulasvirta & Hornbaek [77] include tool use as one of seven core conceptions of interaction. Indeed, several authors have focused on the mediating role of tools (or instruments) in the relation between human users and their objects of interest, addressing issues such as in-directness. Bødker [22] emphasizes how tool use is embedded in human practice, with individual and cultural/collaborative sides that mirror each other. Rabardel [130, 131] analyzes the genesis of instruments as either directed towards the subject ("instrumentation") or towards the artifact ("instrumentalization"). Beaudouin-Lafon [13, 14] argues for the design of interaction models that can be used to describe, evaluate and generate interaction styles and techniques, and introduces Instrumental Interaction, a model based on explicit digital tools.

In summary, while human tool use has not been a mainstream topic of research in Psychology nor HCI, it brings an enlightening perspective on the role of mediating artifacts in our interaction with both the physical and digital worlds.

5.2 Concepts of Instrumental Interaction

The main concept of Instrumental Interaction [13] is that interaction with digital content should be mediated by digital tools called *instruments* that operate on the *objects of interest*. In line with the above theories of human tool use, we define instruments as objects that can be manipulated by the user to affect other objects. In order to elicit technical reasoning, and therefore user appropriation, instruments must have clear technical properties that can be combined with those of target objects. In the same way that a pen can write on different types of surfaces, even those it was not necessarily designed for, such as a wall, a digital pen should work with different types of content.

A number of current computer interfaces feature "tools", usually grouped into tool palettes: the user selects a particular tool and can then apply it to the object(s) of interest. These tools do not meet the above definition of instruments, though, because they typically only work with pre-defined types of objects and are trapped inside their applications.

Consider a simple, ubiquitous digital tool in current interfaces: the color picker (Fig. 5). Color pickers vary within a single application, e.g. text vs. highlighting in Microsoft Word; within a single suite, e.g. Microsoft Word vs. Excel; and across suites, e.g. Microsoft Office vs. Adobe Creative Suite. Users must constantly adapt, since the Adobe Photoshop selector cannot be used in Microsoft Word, nor vice versa, and the text coloring tool cannot be used to pick a highlight color. The only way to copy a color across applications is through a separate tool, e.g. the eyedropper. Colors cannot be applied to objects that were not designed to be user-colorable, e.g. the window frame or document background. Color pickers often feature color palettes and color swatches, but these are rarely first-class objects that can be saved, loaded, exchanged, or attached to a document. Conversely, a user cannot designate an existing object as a color swatch, turning it into a tool. Finally, in multi-device environments, color pickers cannot be used collaboratively or migrated to a handheld device, e.g. to adjust color on a wall-sized display from a handheld touch tablet. In fact, current color pickers do not address the needs of professionals who use color for their everyday work [82].

Fig. 5. Incompatible color pickers from Microsoft Word (for highlighting), Microsoft PowerPoint (for text), Affinity Designer, and Adobe Photoshop

Instrumental Interaction challenges current digital tool design by separating instruments from the information or documents they operate on. The goal is to both leverage our everyday skills in interacting with the physical world through tools and make the digital world more flexible by letting users choose the tools they want to use, create their own tools, and compose their own digital environments. Instrumental Interaction is grounded in the theories outlined in the previous section, in particular the notion of tool as mediating artifact, as extension of the body, and as construct that turns an object into an agent for action.

The generative power of Instrumental Interaction can be readily illustrated with the above example of the color picker: in an instrumental interface, any color picker could be used with any content that has color properties, both to extract and apply color to objects; existing objects could be turned into a color picker, such as a photo or the color histogram extracted from it, as in Histomages [49]; color swatches would be first-class objects that can be moved, resized and combined to assess color in context, as in ColorLab [82].

We now introduce the generative principles that we developed for Instrumental Interaction, and illustrate their analytical, critical and constructive power with two examples.

5.3 Generative Principles of Instrumental Interaction

We introduced the three principles of Instrumental Interaction, *Reification*, *Polymorphism* and *Reuse*, in 2000 [17], together with Instrumental Interaction itself [13], and have since then applied them extensively.

5.3.1 Reification. The key principle of Instrumental Interaction is *Reification*: an instrument *reifies* an abstract command or concept, i.e. it embodies it into an object that the user can interact with to manipulate a target object. For example, a scrollbar reifies the action of navigating a document by mediating the user's actions: the user acts upon the scrollbar by moving the thumb, which acts upon the document, changing which part of the document is displayed in the window.

Reification is the fundamental generative principle for creating new instruments: when confronted with the problem of providing a given functionality, designers can imagine an instrument that reifies it, in order to make it concrete for the user. This contrasts with the traditional approach where a new functionality is typically bound to a new menu item that acts as a "magic word" to invoke the functionality, typically forcing procedural learning of which command does what and when it is available. By contrast, a "properly designed" instrument leverages technical reasoning:

the tool should have a technical effect that makes it more understandable and memorable than the mere arbitrary mapping between a menu entry and a command.

Of course, not every reification of a command into an instrument will be successful. Applying the principle still requires skill, creativity, luck, and the patience to iterate. Nevertheless, as we will show in the next sections, *Reification* has been instrumental (so to speak) in helping us design interfaces that are both more powerful yet simpler to learn and operate than their non-instrumental counterparts.

5.3.2 Polymorphism. *Reification* generates new instruments, calling for an additional principle to control the potential explosion of the number of instruments in the environment. The second generative principle, *Polymorphism*, helps create instruments that can be applied to objects of different types.

For example, a coloring instrument can work with different types of objects, and even different parts of objects. Many drawing tools have a different command to change the border color vs. the fill color of an object, and the color of text vs. the color of text highlighting. A single coloring instrument can replace these commands, by acting differently depending upon whether it is applied to the border or interior of a graphical object, to text, or to a highlighting brush.

As in programming languages, *Polymorphism* can be used for better or for worse. To be successful, a polymorphic instrument must have some internal consistency that makes its behavior with different types of objects predictable. Here too, we expect to leverage technical reasoning: "*I want to change the color of a folder in the file explorer, I should be able to use the same color tool as in my text editor*", rather than procedural learning: "*I have to remember that to change the background color of the document I need to go to the document settings*".

5.3.3 Reuse. The third generative principle of Instrumental Interaction is *Reuse*: the ability for users to reuse previous actions (*Input Reuse*) or the results of these actions (*Output Reuse*). This principle is based on the observation that computer tasks are often repetitive and users prefer starting from existing content and modifying it rather than starting from scratch. Historically, many physical tools have been created to facilitate reuse and make repetitive tasks more efficient. For example, pins were replaced by staples to bind documents, and then paper clips were introduced to facilitate reuse and to easily unbind documents without damage [128].

Output Reuse is already present in many interfaces through the "copy-paste" and "duplicate" commands. However, designing with *Output Reuse* in mind can help extend the power of these commands, for example with the ability to copy the tools themselves in order to create variants, e.g. brushes with different patterns. Similarly, *Input Reuse* is often present in the form of a "redo" command, and sometimes with the ability to access and selectively re-execute past actions. Combining *Input Reuse* with *Reification* leads to creating instruments that embody a sequence of past actions, empowering users to create their own tools through use.

5.3.4 Power in combination. Beyond their individual generative power, the three principles are even more powerful in combination: *Reification* creates more objects that can be targeted by *Reuse*, *Polymorphism* expands the types of objects that can be targeted by a given instrument, including groups of heterogeneous objects, and *Reuse* facilitates the creation of new objects and instruments by end users.

We now illustrate how these principles directly guided the design of novel interaction techniques in two recent research projects. We describe the analytical, critical and generative power of Instrumental Interaction with two examples drawn from our own research: StickyLines [51], for graphical editing, and Textlets [71], for text editing.

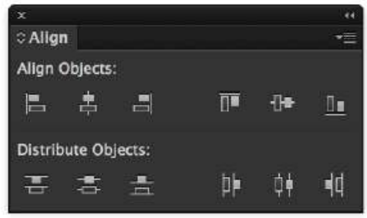

Fig. 6. Typical palette for alignment and distribution (Adobe Illustrator)

5.4 Applying the Generative Principles: StickyLines

StickyLines [51] addresses the issue of aligning and distributing graphical objects. It refines and extends the notion of *magnetic guidelines*, which arose in the context of applying these principles to the design of the CPN/Tools editor and simulator for Colored Petri Nets (CPNs) [16, 18]. StickyLines is the result of participatory design workshops with graphic designers and users of tools such as Adobe Illustrator, where we used the generative principles to address their specific needs.

5.4.1 Analytical and Critical Power. Our studies of users of Colored Petri Nets showed that they use graphical alignment extensively to create diagrams that are both visually pleasing and semantically meaningful. The tool they used, Design/CPN, featured alignment commands similar to most graphical tools (Fig. 6): a set of menu items or buttons in a tool palette to align the selected objects horizontally or vertically, along their centers or one of their sides.

The fact that these commands are not reified into a manipulable object means that users spend up to 25% of their time aligning and re-aligning objects, or selecting a group of aligned objects before moving them together so as to keep them aligned. While Design/CPN, like most graphical tools, let them group a set of objects to manipulate them as a single object, they seldom used this feature because many objects are part of multiple, overlapping alignments. We observed similar issues with professional designers using tools such as Adobe Illustrator. Moreover, they also often needed to adjust the alignment or distribution of shapes to be visually correct: when aligning logos, for example, the visual center is rarely the geometric center of the bounding box. Similarly, when distributing objects, the visual size is often not the bounding box of the object. Graphic designers therefore routinely adjust alignment and spacing manually. Unfortunately, the system does not keep track of these adjustments, which users need to perform again every time they re-align or re-distribute the objects.

The alignment commands of tools such as Design/CPN and Adobe Illustrator are also insufficiently polymorphic. While they can be used to align different types of shapes, they are limited to horizontal and vertical alignments. Some Petri Nets and graphical designs lend themselves to, e.g., circular layouts, but users can only create those coarsely, by hand. Most tools also do not support aligning the control points of polylines or curves, but only the entire object. According to the principle of *Polymorphism*, the "align" and "distribute" commands should be applicable to any object with a position and size and to any shape, not just horizontal and vertical lines.

Finally, the align and distribute commands result in poor *Reuse*. For example, it is not possible to extract the alignment and/or distribution properties of a set of objects to apply them to another set.

5.4.2 Constructive power. StickyLines is the result of applying *Reification* to the align command: a magnetic guideline is an object that can be created, moved and deleted like any other object in the diagram[1]. The user can attach an object to the guideline by dragging the object and snapping it

[1]See a full video of StickyLines at https://www.youtube.com/watch?v=jIDDWJ1xptE.

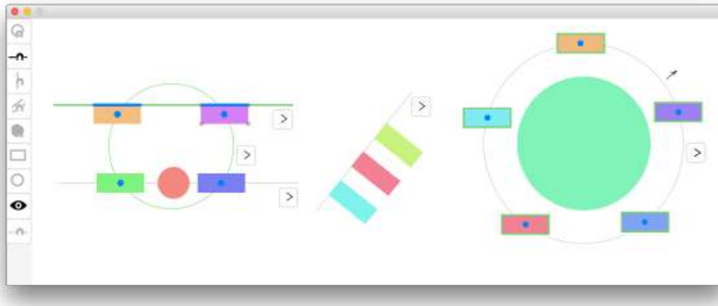

Fig. 7. Circular and linear StickyLines (left), parallel StickyLine (center) and Stickyline created from an existing shape (right) and resized. The user attaches shapes to a StickyLine by moving them onto it

to the guideline, and can detach an object by dragging it away from the guideline (Fig. 7). Crucially, the objects attached to a guideline move with it when the guideline itself is moved, maintaining the alignment.

StickyLines also supports the even distribution of objects along a guideline: when objects are added to or removed from it, or when the endpoints of the guideline are moved, the objects move along the StickyLine so that they stay evenly distributed. Finally, StickyLines reifies the concept of *adjusting* the alignment of an object: When using the arrow keys to move an object attached to a StickyLine, a purple line linking the object to the line, called a "tweak", shows the visual offset applied to the object (Fig. 8 left). The object is still attached to the line, and moves with it. The tweak itself is an object (*Reification*) that can be selected, copied and pasted (*Reuse*), or deleted. The same principle applies to the bounding box of an object when distributing objects (*Polymorphism*): the box can be made visible and directly modified, changing the system's notion of the object's bounding box and therefore altering the distribution accordingly (Fig. 8 right).

Another use of *Reification* is the ability to change the distribution function that maps the rank of objects on a guideline into their position. StickyLines can show this mapping as a curve, which can be edited to create a layout where objects are progressively closer or further away from each other (Fig. 9).

StickyLines support *Polymorphism* by allowing any object to be attached to them, including geometric shapes, text, and the corners of multi-segment lines and arrows. In addition, any shape can be turned into a guideline, including shapes that are part of the diagram itself (Fig. 7 right). We also found it useful to let guidelines be the target of other commands, such as setting the fill or border color of shapes. Rather than applying these attributes to the guideline itself, it applies them to each shape attached to the guideline, as if they were a group.

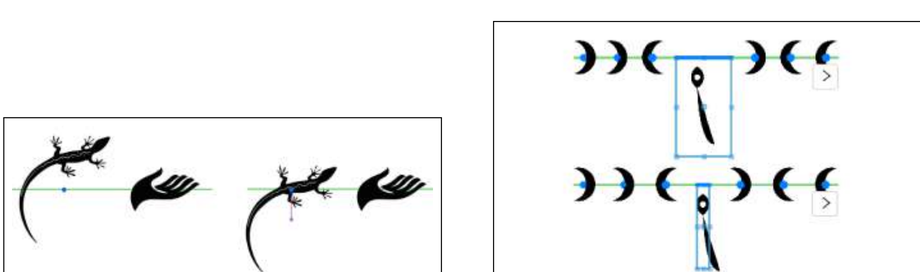

Fig. 8. Left: Tweaking the position (purple line) for visually correct alignment; Right: Tweaking the bounding box (blue box) for visually correct distribution

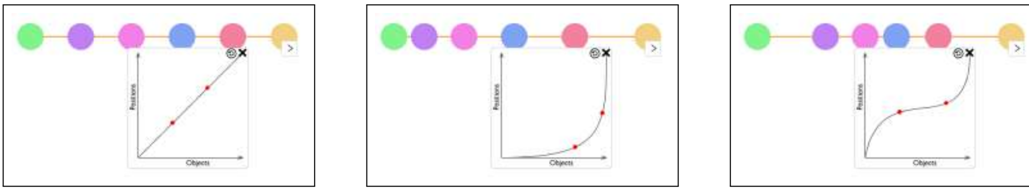

Fig. 9. Controlling the distribution of objects along a StickyLine. The red dots control the shape of the curve and therefore the spacing between adjacent shapes on the guideline. Left: default (linear) mapping; Center: exponential mapping to make objects on the left closer together; Right: sigmoid mapping to make objects in the middle closer together

Since guidelines and tweaks are objects that can be directly manipulable by the users, they naturally support *Reuse*, such as copy-paste and duplicate. Copying a guideline copies the objects attached to it, but it is also possible to copy just the guideline, with its attribute. This is particularly useful when the guideline is used for distribution, e.g. to copy the distribution function.

We found that StickyLines blur the distinction between tools and objects of interest: a StickyLine is used as a tool when the user moves it to move the aligned objects, or when it "grabs" objects as the user moves and releases the guideline over them; but it is also the object of interest when the user snaps an object onto it, or when a new StickyLine is created automatically when an object alignment is detected. A StickyLine is therefore not just an object "ready-at-hand" that works only when explicitly manipulated, it is an object "present-at-hand" that affects the behavior of other objects as they come near it.

Our performance evaluation of StickyLines showed that it was up to 40% faster than traditional align and distribute commands [51]. This is due mainly to the fact that StickyLines keep objects aligned, and that users can manipulate aligned objects as groups even when individual objects are attached to multiple StickyLines.

We also observed that graphic designers used StickyLines in creative and unexpected ways. While tweaks were intended as small adjustments to an object's position, we observed some users creating giant tweaks so that they could move a set of non aligned objects together by moving the guideline, reifying the notion of a group. A user also asked if the concept of distribution could be generalized to 2D, to automatically lay out (and tweak) objects within an area instead of along a line. We also observed that Petri Net designers started to create guidelines before creating the diagram itself, as a way to plan the overall layout. This shows how the generative principles not only help generate creative and powerful solutions, but also how they inspire users to take them even further. It also provides evidence that users were employing technical reasoning to appropriate the properties of the guidelines in unexpected ways.

5.5 Applying the Generative Principles: Textlets

Text editing is a basic feature of many interactive systems, and word processors are one of the most heavily used applications. Yet text editing has not changed very much over the past decades. In the context of the European Research Council project ONE (Unified Principles of Interaction), we interviewed two groups of professional users who rely on word processors for their work, namely contract lawyers and patent lawyers, to study how they manage the internal constraints and relationships that are critical to these documents, such as using the proper vocabulary consistently, meeting constraints such as word length, or maintain consistency between the claims and the body of a patent [71].

5.5.1 Analytical and Critical Power. We found that while all participants used Microsoft Word extensively, they did not use built-in features that could address their needs, even if they knew these features. For example, many participants do not use global search and replace because they cannot check its effects easily, and they hesitate to use incremental search and replace through fear of missing a change. Some prefer numbering items in a contract by hand, even if it also means renumbering them and their references by hand every time an article is added, moved and removed. They rarely use styles, as these tend to pollute documents as they circulate among users, and they often manage versions and variants by hand, e.g. by using colored text rather than Word's review mode.

Our interviews showed that users prefer these manual operations because it gives them a sense of control. The fact that the effect of Microsoft Word commands such as search-and-replace are not directly visible and editable, i.e. reified into objects that they can inspect and revise, makes them wary that "it did not do the right thing", with possible dire legal consequences. They find that checking that the command worked is more tedious than performing the changes manually.

The lack of *Polymorphism* for Microsoft Word features that are seemingly identical, such as numbering list items, section titles, references and footnotes leads to cognitive overload: users must rely on procedural learning rather than technical reasoning as each command works differently and has different settings.

Finally, the lack of *Reification* limits *Reuse*. For example, styles are not first-class objects that can be directly copied and pasted, e.g. across documents. The participants in our study sometimes create their own templates, and use copy-paste extensively, but fight with the side effects such as "style pollution".

5.5.2 Constructive Power. In order to address some of these issues, we applied *Reification* to the most common operation in text editing after typing text: selection. In current text editors, selection is transient: the user selects some text, invokes a command that affects the text, e.g. turns it bold, and the selection is lost as soon as the user clicks elsewhere. By reifying text selection into a first-class object, called a *textlet* [71], we turn the transient selection into a persistent object that users can return to at any time. Textlets are highlighted in the document and displayed in a side panel. Clicking a textlet reselects the associated text.

The power of TEXTLETS derives from the ability to assign *behaviors* to them. For example, the *word count* behavior adds a counter that displays the number of words in the textlet in real time, as the content of the textlet is edited (Fig. 10). Similarly, a *variant* behavior lets users record multiple alternatives for the content of the textlet, and pick the one being displayed. A more sophisticated behavior is *search-and-replace* (Fig. 11): a search textlet dynamically creates textlets for each occurrence of the search text and lets users replace occurrences one by one, in the order they want, or all at once. Replaced occurrences are displayed in a different color, and the user can undo and redo any change. Another example is *numbering* (Fig. 12), which creates automatically numbered textlets according to a pattern defined by the user; each such textlet can then manage references to it as separate textlets.

Behaviors represent a different form of *Reification* than that of STICKYLINES or the basic textlet introduced above. Rather than turning a command into an object, they turn a command into a

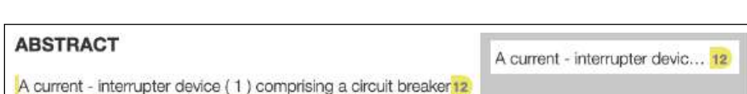

Fig. 10. Textlet for counting words. The word counter appears both in the text itself and in the side panel

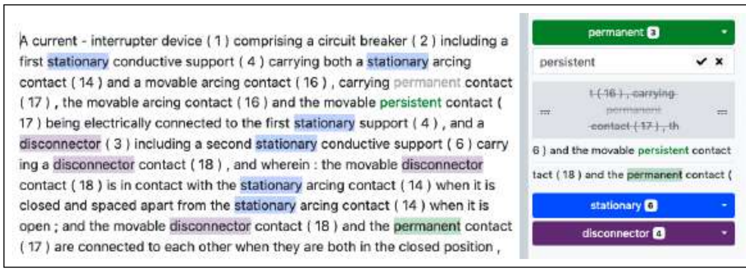

Fig. 11. Three textlets for searching and replacing text. The side panel lets users specify the replace string, exclude matches from a global replace, and perform and undo global and individual replacements. In the document, matching occurrences are highlighted with the same color as the textlet, e.g. "stationary"; replaced occurrences use the color of the textlet, e.g. "persistent"; excluded matches are in grey, e.g. "permanent"

dependency that is re-evaluated and updated when the content of the surrounding document changes, as in a spreadsheet. Attaching the command to the textlet makes it more concrete, observable and manipulable. For example, a word-counting behavior turns a textlet into a real-time word-counting instrument.

Behaviors make textlets Polymorphic: different behaviors can be attached to a textlet, possibly in combination. For example, combining the variant and word counting behaviors lets users try alternatives of, e.g., an abstract limited to 150 words, and see the word count in real time as they edit the abstract and switch between the alternatives. At a more abstract level, the notion of textlet as *Reification* of selection is polymorphic in that it can be applied to other types of content than text, e.g. the graphical shapes of a drawing editor or the file icons of a file manager.

Textlets support *Reuse* at a very basic level: by remembering the transient state of a selection, the user can instantly re-select the text by clicking the textlet (*Input Reuse*). At a higher level, textlets and their attached behaviors can be copied and pasted, and attached to a different selection (*Output Reuse*). Turning transient selections into persistent objects also makes it possible to, e.g., manage multiple simultaneous searches whereas the traditional approach only supports one search string at a time. This makes it easy to reuse a previous search.

The participants in our study found Textlets particularly easy to use and efficient for their tasks, giving them a sense of control and appropriate feedback on their actions [71]. As with StickyLines, we observed appropriation of Textlets by users. For example, some participants created search textlets for forbidden words, to track words that they had decided should not be used, e.g. in a patent. Typing such a word would immediately highlight it, warning the user in a non-obtrusive way. We also brainstormed many other behaviors that could be attached to textlets, such as indexing

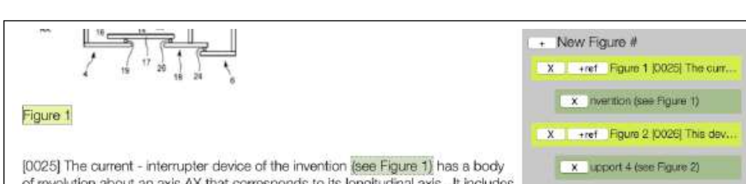

Fig. 12. Textlet for numbering items. The '+' button creates a new instance of the template (here 'Figure #'); the '+ref' button creates a reference to the enclosing textlet; the 'X' buttons delete textlets and the corresponding text. Numbering updates automatically as the text is edited and as items are added, moved and removed

words for a glossary, creating spreadsheet-like formulas for calculated text, hiding, summarizing or translating text, etc.

5.6 Discussion

The original article [13] introduced Instrumental Interaction introduced as an *interaction model* and demonstrated its descriptive, evaluative and generative power, similar to the analytical, critical and constructive dimensions we are using in this article. The followup article [17] introduced the three principles of *Reification*, *Polymorphism* and *Reuse*. However, the formal articulation of the link between the notion of instrument and the theoretical concepts of affordance and technical reasoning was made several years after these articles and remains an ongoing research topic.

Since these original papers, we have used Instrumental Interaction in many projects and for teaching. Students have applied the generative principles successfully in many applications, such as reifying paths in map applications, reifying layout in presentation software, or reifying channels in communication applications [69]. Instrumental Interaction has also been used beyond desktop interfaces, including touch-based interfaces [42, 155], multi-surface/multi-device interaction [90], collaborative systems [91], and tangible interfaces [83]. These examples demonstrate Instrumental Interaction's ability to suggest novel and powerful techniques. Table 1 summarizes questions to ask and issues to address, analytically, critically and constructively, to apply the concepts and principles of instrumental interaction.

Instruments transform user actions into operations on the objects of interest, and therefore act as modes: the same user actions can have different effects according to the instrument being used. While modes are often considered harmful [132, p.42][147], instruments reduce the risks of mode errors [122] because they are "ready-at-hand", i.e. they are internalized by the user. Indeed, Larry Tesler, creator of cut-copy-paste and a strong opponent to modes, recognized that "*modes can be good when they support a metaphor like picking up a brush*" [147].

Tool-based interaction can make interfaces more intuitive by enabling technical reasoning, where users fetch the tool with the proper technical effect to get the result they seek. Most interactive software systems rely instead on procedural knowledge, which requires learning and remembering commands, instead of technical reasoning, which develops through practice and can be transferred from one context to another. We can find evidence of technical reasoning in the fact that users spontaneously develop creative and unexpected uses of digital instruments, as illustrated with StickyLines and Textlets.

At a larger scale, the notion of instrument challenges the notion of application that is a cornerstone of today's interactive environments [15]. Instruments should not be trapped into applications, and the content managed by applications should be more interoperable. Ideally, a StickyLine should be available not only for graphical editing, but also to align the icons in a file manager, the windows on the screen, or the margins in a word processor. Similarly, Textlets should be available anywhere there is text, such as the names of files or the result of an on-line search. The power of Instrumental Interaction therefore goes beyond incremental improvements to existing systems. Instead, it helps HCI researchers to deeply re-think interaction and create a new breed of interactive environments that are both more flexible and powerful yet simpler to use.

6 EXAMPLE 2: HUMAN-COMPUTER PARTNERSHIPS

The second example explores the concept of a Human-Computer Partnership, which operates at a different time scale than Instrumental Interaction. Instead of examining how to create individual instruments that accomplish specific functions such as alignment and selection, we shift the focus to the process of use over time. Users establish stable interaction patterns with time spans that range from seconds and minutes to months or even years. This process involves at least two inter-related

Table 1. Questions that can be asked to apply the concepts and principles of Instrumental Interaction analytically, critically and constructively.

Instrumental Interaction	Analytical	Critical	Constructive
Concepts			
Object of Interest	What are the objects visible and directly manipulable by the user?	Do these objects match those of the users' mental models?	Are there other objects of interest, e.g. styles in a text editor? Should some objects of interest turned into instruments?
Instrument / Tool	What functions are available as tools, e.g. in tool palettes, as opposed to commands, e.g. menu items?	Do the tools actually work as such i.e. by extending users capabilities? Do the tools enable technical reasoning?	Which commands can be turned into tools? Are they related with physical tools?
Principles			
Reification	Which concepts/commands are reified into interactive objects/tools? How can these objects be manipulated?	Are the reified concepts effective? Are the objects directly manipulable?	Which concepts/commands should be reified? Into which objects/tools? What manipulations should be available?
Polymorphism	Which commands/tools apply to objects of different types? Do they apply to collection of heterogeneous objects?	Should commands/tools apply to multiple object types? Which types?	How to make each instrument (more) polymorphic? How to create groups of heterogeneous objects?
Reuse	Which commands/objects can be reused?	Which commands/objects should be reusable?	How to make commands reusable (input reuse)? How to make objects reusable (output reuse)?

phenomena: When faced with a new technology, users must discover its capabilities and adapt to its existing features. Yet users can also appropriate technology to meet specific needs or to generate personalized output. Effective human-computer partnerships must account for both, which becomes even more complex when interacting with "intelligent" systems that learn from or influence the user's behavior. We need to consider how users and systems can share agency, while letting users retain control over the interaction. The challenge is how to create simple, easy-to-understand forms of interaction for novices, while providing an incrementally learnable path to achieve the flexibility and power of an expert.

6.1 Theoretical Foundations

Early computers took up entire rooms and required a small army of human technicians to program and maintain them. However, today's computer technology is so small and inexpensive that many people own multiple devices and actively construct personal software ecosystems over time [69]. Users are deeply influenced by the computational ecology they grow up with — Apple users tend to stick with Apple devices, PC users stick with Microsoft environments, and Linux users stick with Linux — deriving their skills and expectations from previous experience, which affects their future interactions. HCI's historical emphasis on point designs does not encourage researchers to consider how users and their corresponding software ecologies [56] co-develop. The notion of

Human-Computer Partnerships builds on theory from the natural sciences to better inform this dynamic relationship.

6.1.1 Co-Adaptation in Evolutionary Biology. The biological phenomenon of *Co-adaptation* specifies how species are both affected by and affect the environments in which they live. The more familiar term *co-evolution* describes how the actions of certain species reciprocally affect each others' behavior over millennia. Each species in a co-evolutionary relationship exerts selective pressures on the other, thus influencing how the other evolves. For example, anaerobic bacteria released oxygen into the atmosphere, which created a new, oxygenated habitat where aerobic bacteria evolved that can breathe oxygen [108]. More relevant to human-computer interaction are the changes that occur within an organism's lifetime. In *The Origin of Species* [52], Charles Darwin defined co-adaptation as "*the usually beneficial relationship between different species*", where the "*skills, habits, actions, or form of one species matches and uses the skills, habits, actions, or form of another species.*" Darwin explains that co-adaptation refers to the interactions among individual organisms, whereas co-evolution refers to their evolutionary history.

The key idea here is that organisms do not simply *adapt to* their current environment and the other organisms within it. Rather, organisms also actively participate in the evolutionary process: their actions *adapt* and modify the environment, such that they and other organisms within that environment evolve and change together. Co-adaptation emphasizes this on-going, potentially asymmetrical process of mutual influence between organisms and the environment, where "survival of the fittest" is not simply a matter of an organism adapting to a changing environment, but also of it physically changing that environment to ensure its survival.

6.1.2 Co-Adaptation in HCI. The concept of Human-Computer Partnership builds upon Mackay's [109] observations of Co-adaptation with interactive systems, where users do not passively react to a technology's features, but also intentionally appropriate it to meet their needs. Some systems are designed to reduce this potential for customization, such as the purposefully limited options available on automated teller machines. However, other systems encourage exploration and innovation, such as the underlying structure of spreadsheets that supports far more innovative activities than simply adding up columns of numbers [119]. Co-adaptation takes the user's perspective, highlighting the dual relationship users establish with any interactive technology, where they discover how it works but also modify it for new purposes. When we take the system's perspective, especially with an intelligent system, we see a similar, although not identical, relationship, where some systems adapt to the user, e.g. smartphones that suggest words based on the user's earlier input [156, 157], while others adapt the users behavior, e.g. to teach users new skills. In summary, Co-adaptation offers a lens for examining the dual role of the user's interaction with technology, which shifts the focus from what Bill Buxton calls "getting the design right" to "getting the right design" [43].

6.2 Concepts of Human-Computer Partnerships

The concept of Co-adaptation [109] accounts for how users are continually learning from and modifying technology over time. It challenges the idea that interaction designers can maintain full control over "the" user experience, since any product may be experienced in ways that were never predicted [62]. Carroll [45] argues that users struggle with technology in a series of attempts to engage their prior knowledge and skill to accomplish meaningful results. This means that users should be viewed as active interpreters of their technology who not only assess how the system is "supposed" to be used, but also how to customize it for current or future needs. We argue that true Human-Computer Partnerships should help users:

(1) *adapt to* the system by supporting *discovery* of how they work; and

(2) *adapt* the system by supporting *appropriability* of its characteristics or by generating *expressive* output.

For example, creativity support applications include sets of tools with user-modifiable parameters. Non-experts should not only be able to discover which tools perform which functions, but, importantly, which of their needs can be addressed by which tools. The system might help users find or perform relevant commands, or reveal how it interpreted a particular command, or how to correct possible errors. Similarly, it is important to support user innovation by letting users take advantage of technical reasoning [125] to infer the system's properties, and re-purpose them to create novel solutions or different forms of output.

These two properties have clear correlates in the physical world. For example, chopsticks are tools designed for eating rice or noodles, whose physical properties — size, shape, weight, rigidity — are easily discerned. Although efficient use of chopsticks requires knowledge and practice, non-chopstick users can still learn through trial and error, although the process may be slow and messy. Over time, users discover and master the finer points of chopstick use, such as how to balance them, and which fingers should move or remain static. Experienced chopstick users can use technical reasoning to adapt objects with the appropriate properties to perform the same task. For example, a thin, round, rigid pencil is about the right length and will do in a pinch. Another form of adaptation is to take advantage of the tool's physical properties for new purposes, such as creating an impromptu drumstick or tying back long hair. Users can also adapt an object's properties for artistic expression, such as the artist who created a full-size portrait of Jackie Chan[2] by assembling bundles of chopsticks of varying sizes into a large, tightly spaced grid to create the effect of pixels of varying shades of color intensity. The challenge is to make interactive systems similarly understandable and adaptable.

6.2.1 Reciprocal Co-Adaptation. The above examples involve interaction with "non-intelligent" objects that have no agency of their own. An artist can hone the edge of a pencil to achieve a particular effect, but the pencil will not suggest which edge will produce the best result. By contrast, intelligent systems can act independently, with their own agency. This suggests that such systems should be able to:

(3) *adapt to* users, by learning from their behavior; and
(4) *adapt* users, by changing their behavior.

Here, we see the same basic co-adaptive relationship as before, although the specific natures of human and system intelligence are very different. The third property asks the system to capture patterns of the user's behavior over time, so it can adjust its responses to meet the specific needs of that individual. However, this also implies that users should be able to discover what the system has inferred from their behavior, and change it if it is unwanted or incorrect. The fourth property asks the system to modify the user's behavior, ideally with the user's consent. This can have major benefits for the user, as when educational software [8] improves the user's skills. More controversially, some systems, e.g. persuasive technologies [61, 113], modify users' behavior without their explicit knowledge or permission: the challenge remains how such systems can offer users "informed consent" [40].

To be effective, a Human-Computer Partnership with an intelligent system requires careful thought about how users and the system share agency, especially how to allocate control. Users reason about tasks differently than computers, and may or may not be aware of the system's agency, which makes intelligent tools significantly more complex to design and understand. We use the

[2] http://redhongyi.com/portfolio/jackie-chan-chopsticks-portrait/

term *Reciprocal Co-adaptation* to describe how these four relationships cross-influence each other to form a Human-Computer Partnership.

6.3 Generative Principles for Human-Computer Partnerships

Successful Human-Computer Partnerships are co-adaptive: they allow users to successfully adapt to the specific characteristics of the system, and also adapt the system to meet their current needs and desires. This suggests three generative principles:

6.3.1 Discoverability. When first approaching a new system, users usually have an idea of how it works and what they seek to accomplish with it. Even so, they need help figuring out where expected commands are located, and how to invoke them, as well as finding out which new commands or features are available. They need feedback about their previous actions to understand how the system interpreted them, and also feedforward, to discover how they can achieve particular effects. When we consider Reciprocal Co-adaptation, the system should also learn from the user's behavior, and use that information to inform the user or achieve specified goals.

6.3.2 Appropriability. When people try accomplish a particular task, they look for appropriate tools in the system. If they cannot find them, they should be able to modify or re-purpose commands or features to meet their needs. Ideally, as with physical systems, users should be able to re-purpose various properties of the interface to create new tools. This may be as simple as remapping gestures to new commands, or taking advantage of certain features to accomplish new tasks. When we consider Reciprocal Co-adaptation, the system should explicitly influence the user's behavior, either through teaching new skills or subtly modifying their actions.

6.3.3 Expressivity. Each person generates individual variation every time they interact with technology. For example, physical handwriting reflects both unconscious variation, identifiable as an individual's handwriting style, and conscious choices, such as carefully lettering a love letter. Systems should be able to capture this input variation and transform it into correspondingly rich, expressive output, under the user's control. When we consider Reciprocal Co-adaptation, the system should reflect individual aspects of the user's behavior, such as generative design tools that create dynamic patterns guided by the user's input.

These three principles address different aspects of Co-adaptation in order to better support Human-Computer Partnerships, and can be summarized as:

(1) *Discoverability*: reveal how the system interprets the user's recent behavior (*feedback*) and which commands are now possible (*feedforward*);
(2) *Appropriability*: modify the system's behavior by *customizing* its characteristics for new purposes; and
(3) *Expressivity*: create rich, personalized output generated from individual user-controlled input variation.

6.3.4 Power in combination. Each principle addresses users' specific needs as they interact with the system over time. Although effective individually, as with Instrumental Interaction they achieve greater power when combined. Discoverable systems encourage exploration, which can lead to new insights and ideas for appropriation. Appropriable systems offer user-modifiable properties that users can manipulate to create more expressive commands or mappings between user input and rich output. Expressive systems reveal details of how the system interprets the user's behavior, which can lead to discovery of new features (*Discoverability*), that can in turn be modified for new purposes (*Appropriability*).

6.4 Applying the Generative Principles: CommandBoard

Under the auspices of the European Research Council CREATIV project, we explored gesture-based interaction on mobile devices, with special emphasis on gesture typing [156]. We observed that, although today's smartphones are extremely powerful computers, they offer an impoverished form of screen-based interaction, restricted primarily to tapping, swiping and pinching. We developed several systems, culminating in CommandBoard [2], that use gestures to increase users' power of expression while preserving simplicity of interaction. We ran multiple formative studies, including interviews and participatory design workshops, and applied the principles of *Discoverability*, *Appropriability*, and *Expressivity* to generate novel design solutions.

6.4.1 Analytical and Critical Power. Gesture-based interaction allows users to express a huge vocabulary in a limited space, and advances in machine learning make it possible to reliably recognize a wide variety of gestures. Because the "*nervous system has the capacity to form multiple long-term (> 24 hours) motor memories*" [97], experts rarely forget these learned gestures, resulting in highly efficient interaction.

Given the speed and expressive power of gesture-based interaction, why is it so rare? One major difficulty lies in the transition from novice to expert, which requires first discovering, then mastering each gesture. Users struggle to remember mappings between gestures and commands, and rely on separate "cheat sheets" with pairs of command names and gestures [12]. Most gesture-based commands lack in-context *Discoverability* — users usually only receive binary feedback as to whether or not a particular gesture was recognized, although a few also provide a confidence level or suggest possible alternatives.

Gesture typing addresses some of the problem by drawing gestures over a soft keyboard. Here, users trace from letter to letter of their desired word, which is "*visually guided, closed-loop, and relatively slow [but] easy because it does not require any prior memory*" [157]. However, over time, experts develop memory-driven gesturing which is "*recall-driven, open-loop, efficient, and fast*" [157]. However, despite being up to 40% faster than tap-typing [99, 156], adoption rates are low. Although gesture-typing is easier than learning free-form gestures, it still takes time, usually about 40 hours: "*If you [make the swipe gesture] enough times, that gesture sticks in your motor memory. It becomes automatic and involuntary, and by repeatedly doing that single motion for a single word you become a shorthand expert*" [98].

Some systems with gesture-based commands offer limited *Appropriability*, usually through a dialog box that lets users change the mapping between a gesture and a particular command. But most systems simply offer users the choice of turning default gestures on or off. For example, Apple iPad users can traverse a series of menus to toggle whether or not using four or five fingers will "pinch to the home screen", but cannot specify that this command should work with only three fingers. Of course, users can use technical reasoning to re-purpose gestures, beyond their "official" purpose. For example, we observed graphic designers draw free-form lines that were not intended as part of the final drawing, but rather served as scaffolding for positioning other graphical elements. Systems such as Knotty Gestures [149], SketchSliders [148] and Knotation [50] build on this idea by allowing users to draw free-form lines that become interactive controllers of music, video or other data.

Many creativity support tools explicitly encourage *Expressivity*, by capturing user-based variation and reflecting it in the output. For example, with a pressure-sensitive pen and the right software, users can control the thickness or the fuzziness of the line as it is drawn. Similarly, systems designed to support musical expression can dynamically change the sound based on the user's movements [21]. Users should be able to choose between creating uniform "professional" output, or revealing the nuanced details of the gesture in the form of personalized results. Including

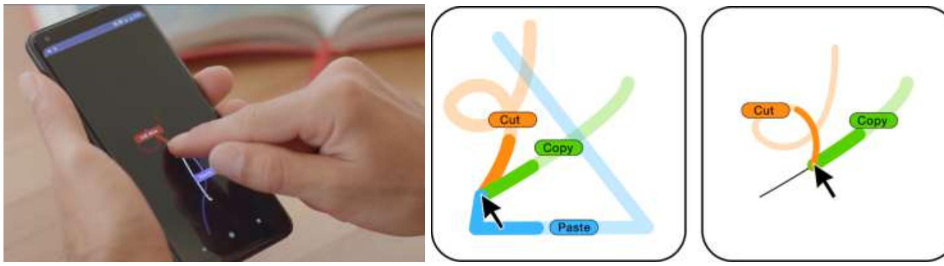

Fig. 13. The OCTOPOCUS dynamic guide [12] reveals the remaining possible completions of a gesture to produce a recognizable command. The dynamic guide appears only when the user hesitates; experts can just perform the gesture directly. OCTOPOCUS offers feedback as to the gesture thus far, and feedforward indicating the remaining possible commands.

Expressivity as a generative principle allows us to capture the individual richness of the user's input, and treat it as a resource for creating varied output, under the user's control.

6.4.2 Constructive power. COMMANDBOARD [2] expands gesture typing on a soft keyboard, taking advantage of earlier techniques that support *Discoverability*, *Appropriability*, and *Expressivity* to provide users with the power of a command-line interface.

Discoverability: When novice users need to perform a gesture-based command, they need to know both which commands are currently available, and how to perform the current gesture. We designed OCTOPOCUS [12] as a dynamic guide that helps novice users discover these mappings, without interrupting expert users. We borrowed the Marking Menu [100] technique of only reacting when the user indicates uncertainty by pausing the gesture. At that point, the system displays *feedforward* (Figure13:left) to show which currently available gestures will produce which commands, as well as *feedback* to show the initial part of the user's gesture. Pausing at the start of the gesture displays all possible commands (Figure 13:center); whereas pausing partway through displays only the remaining possible gesture-commands (Figure 13:right). The variation in the width of each gesture reveals the likelihood that the system will successfully recognize the gesture. Displaying too many gestures creates visual overload, since users can only differentiate about 16 gestures at a time. However, OCTOPOCUS accounts for the current context by displaying only the most relevant gesture-commands, which reduces visual clutter.

Appropriability: Users sometimes want to create their own personal gesture-commands, ones they find easy to remember. However, the system must be able to reliably recognize each user-generated gesture command, and distinguish it from others in the gesture vocabulary. We created

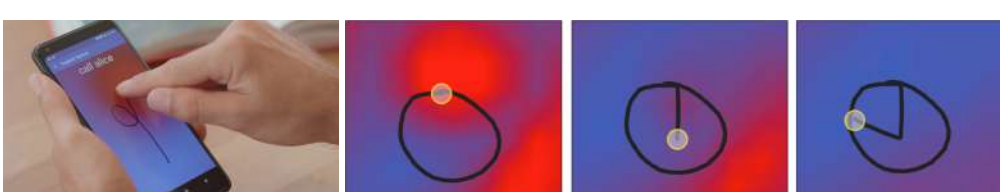

Fig. 14. FIELDWARD [114] displays a dynamic heatmap that displays feedback as to whether the current gesture already exists and feedforward indicates how to create an acceptable gesture. The heatmap changes color as the user draws: Red areas show that the gesture is not unique, purple denotes ambiguity, stopping in a blue area creates a recognizable gesture.

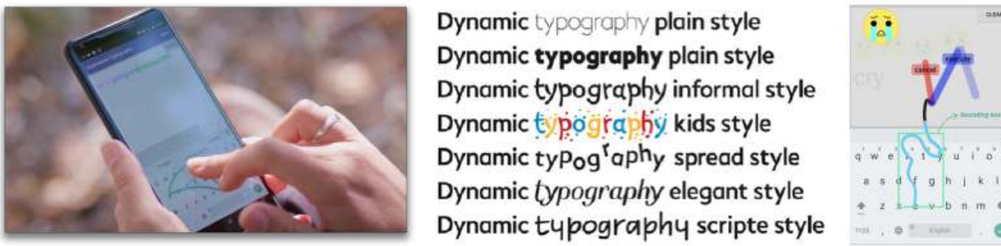

Fig. 15. The EXPRESSIVE KEYBOARD [3] (left) transforms small variations in the user's gesture as they input text into expressive output, in this case variations in the text color (left) or variations in the font (center). MOJIBOARD [4] (right) maps input variation to different emoji expressions [4].

FIELDWARD [114] (Figure 14), a dynamic guide that provides both feedback and feedforward as the user draws a proposed gesture. The FIELDWARD heatmap changes color: If the user stops in a red area, the gesture is not unique and the user should either extend the gesture by moving towards a blue area or try again. If the area is purple, the gesture is ambiguous and cannot be recognized reliably. However, if the user stops in a blue area, the current gesture is both available and easy for the system to recognize. This strategy helps users discover the acceptability of each newly proposed gesture, while revealing if it is similar to one that already exists. Users can also discover features of the recognition algorithm, such as that upward and downward strokes are treated as distinct gestures, despite leaving the same trace. Our studies showed that some users appropriated the system. For example, instead of creating unique gestures for compound commands, such as "Message Mom" and "Skype Ann", one participant developed his own personal syntax with a set of gestures for communication apps (phone, text message, video chat), and another set for key contacts (Mom, Dad, girlfriend) which resulted in an easy-to-remember grammar.

Expressivity: Although users can pre-select particular fonts, gesture-typers cannot vary them as they write, as they would with physical handwriting. Sophisticated machine learning algorithms capture the details of each gesture, but focus only on identifying the "correct" word, and discard individual variation as noise. We created EXPRESSIVE KEYBOARD [3] (Figure 15:left) to transform this variation into user-controlled, personalized output. We first collected samples of how people gesture-type under different physical conditions — walking, sitting, standing — for different purposes, and extracted three common features — curviness, speed variation and size. We then mapped these to red, green, and blue, the three components of RGB color. This lets users generate the full spectrum of RGB colors by simply modifying how they draw from letter to letter, without losing the benefit of word recognition.

The EXPRESSIVE KEYBOARD is clearly expressive, but is also discoverable. Although intuitively it sounds easier to control an explicit feature, such as explicitly increasing the gesture's size to increase the level of red, users actually learned more quickly when they explored gesture variations to see if they could "make it red" or "make it green". We also mapped the characteristics of each gesture to variations of professionally designed fonts (Figure 15:center) and emoji expressions (Figure 15:right). In each case, users enjoyed having fluid control over their output, as well as the option of later adjusting the level of variation. It would be interesting to make EXPRESSIVE KEYBOARD more explicitly appropriable, by letting users remap its expressive output features to different input characteristics.

COMMANDBOARD [4] offers users a dynamic Human-Computer Partnership that combines the three principles, *Discoverability*, *Appropriability* and *Expressivity*. It supports gesture typing as well as providing a free-form interaction space above the soft keyboard, where users can draw gestures that invoke commands. If the user is unsure of how to perform a gesture command, she pauses to display an Octopocus dynamic guide. Alternatively, she can gesture-type any currently available command name, followed by the v-shaped "execute" gesture above the keyboard, to invoke it. She can create personally memorable, but recognizable gestures with Fieldward, and map them to available commands, or commands with parameters. For example, a single, continuous gesture can invoke the "color" command, which displays a color bar she can choose from. Users can modify the shapes of their gestures as they gesture type to produce a range of expressive output, with varying fonts, colors and emoji expressions. Users enjoy several methods of discovering its various capabilities, but also appropriate it and generate expressive output, adapting it to meet their individual needs.

6.5 Applying the Generative Principles: ImageSense

As part of the aforementioned CREATIV ERC project, we observed and collaborated with a variety of creative professionals, including graphic designers [82, 103] and landscape designers [151], with special emphasis on the ideation process [39]. Within this context, we ran multiple studies to better understand how designers use both physical and digital mood boards to express visual ideas. The results of these studies, combined with the theory of Reciprocal Co-adaptation, contributed to the design of ImageSense [94], a collaborative digital mood board. ImageSense illustrates Human-Computer Partnerships that explicitly share agency between user and intelligent agents, with a focus on *Discoverability* from both perspectives, as well as *Appropriability* and *Expressivity*, primarily from the user's perspective.

6.5.1 Analytical and Critical Power. Mood board design is an iterative process that involves three key phases: *collection*, *composition*, and *reflection*. Designers first seek visual inspiration by collecting images from physical magazines or on-line image depositories. Here, *Discoverability* refers to the problem of discovering appropriate images, as well as understanding how the system searches for those images. Next, designers crop, resize and transform images, adapting them to correspond to their design concept. Here, *Appropriability* refers both to the ways they modify these images, as well as how they appropriate other objects, such as fabric swatches or dried flowers, to convey their ideas. Finally, designers find words that convey the mood board's meaning, so they can explain it to external stakeholders [47]. Both latter phases involve *Expressivity*, as they compose images and text to communicate their personal vision. The overall process is highly iterative, requiring fluid transitions across phases, and also highly co-adaptive, as designers adapt their ideas to the images that inspire them, while also adapting those images to better express their ideas.

When we apply a critical lens to mood board design, our studies of designers suggests that *Discoverability* poses the most problems. Few current tools support serendipitous image discovery, beyond browsing through images curated by other designers, e.g. on *Pinterest*. Computer-based search forces designers to transform vague, yet-to-be-articulated visual ideas into concrete, text-based search terms: they struggle to find the right words to describe images they have not yet seen. Search engines do, of course, return many images that the designer can choose from, but this poses an additional discovery challenge, since the system never reveals which criteria were used to select those particular images. Worse, the designer cannot fine-tune or adapt future searches to achieve more suitable results: subsequent searches may return different images or more of the same, depending on hidden machine learning algorithms — the choice is never under the user's explicit control. Finally, search engines often generate large numbers of highly similar images, whereas what the designer really wants are related images that differ from each other as much as possible.

Appropriability is also an issue, since designers must adapt features from different creation tools to achieve desired effects. For example, Adobe *Photoshop* modifies and transforms images,

Fig. 16. ImageSense [93] includes both a central *Mood board Canvas* ⓐ and *Maybe area* ⓑ. The four image collection methods include: *Search Bar* ⓒ with *Search Results* ⓓ; *Upload* area ⓕ; and *System Suggestions* ⓖ. Composition and reflection tools include: *Color* palette ⓗ; *Tool* palette ⓘ; and *Semantic Tag Clouds* ⓙ.

whereas Adobe *Illustrator* is better for experimenting with different layouts, and pen-based drawing programs are better for making quick sketches. Designers sometimes adapt less obvious tools, such as using *Powerpoint* to crop images, because the designer already knows the tool. Designers also appropriate characteristics of certain images, such as generating a color palette from a favorite image, or using an existing color palette to modify other images to match.

Expressivity is clearly fundamental to the whole process, since each mood board reflects the individual designers' choices, influenced by the design brief, other collaborator's opinions, their access to appropriate images, and their own individual taste. However, since current digital mood boards support only one phase of the design process at a time, designers must switch tools, which risks breaking their creative flow. Many designers also have difficulty expressing their visual concepts with words, and struggle to articulate a story that explains the semantics of clusters of images or the mood board's overall meaning [47]. Although designers can choose from a wide variety of search and creativity support tools, none offer intelligent support for discovering appropriate images and their meanings (*Discoverability*), nor do they offer integrated tools for adapting images and text (*Appropriability*) to express their ideas (*Expressivity*).

6.5.2 Constructive power. Our studies show that mood designers' needs vary: sometimes they can articulate what they are looking for; sometimes they are stuck and need inspiration; and sometimes they want to successively refine their ideas, alternating between offering and receiving suggestions. We designed ImageSense [93] as a human-computer partnership that offers all three types of intelligent assistance across the collection, composition, and reflection phases. The key challenge is how to let designers control the level of user- and system-based agency, such that the system's actions benefit designers, without interrupting them. ImageSense achieves this through three embedded systems, ranging from fully guiding the process with ImageCascade, to alternating control with Semantic Collage [94], to reacting to suggestions from May AI [92]. This creates

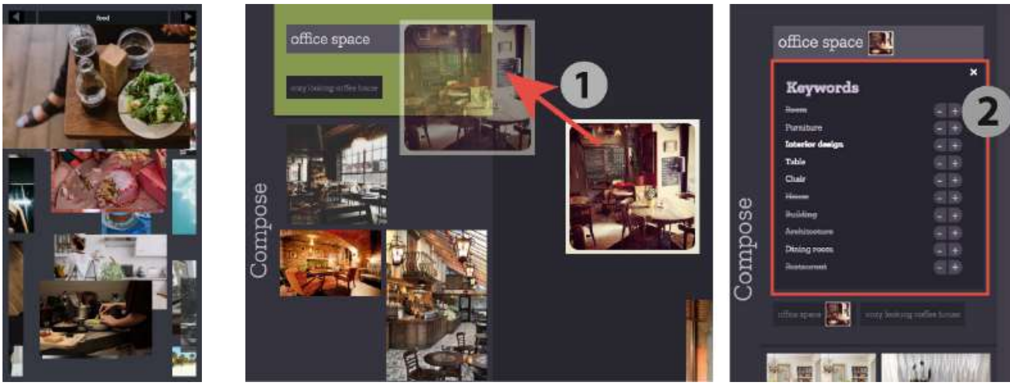

Fig. 17. Left panel toggles between (left) ImageCascade [93], a descending series of human-curated images; and (right) Semantic Collage [94] with ① a mixed-media search bar combines images and text; and ② adjustable associated semantic labels.

a Human-Computer Partnership where designer and system each discover and react to relevant information from the other, and adapt images and text of the mood board accordingly.

Figure 16 shows the ImageSense collaborative mood board, with a shared central canvas ⓐ for composing images and text, and a "maybe" area ⓑ for storing potentially interesting images. It supports *Discoverability* by letting designers search with combinations of images and text ⓒ, to discover new images ⓓ. Designers can also proactively upload their own images ⓕ. When the designer seeks inspiration, the ImageCascade (Figure 17:left) supports serendipitous discovery of new images drawn from human-curated collections. They can select from their own images or relevant sets of high-quality curated images, which they can drag to the main canvas.

When the designer wants to actively refine a search, Semantic Collage (Figure 17:middle) lets users combine multiple images and text, for example, combining an image of a restaurant table and the keyword "office space". In addition to producing relevant images, the system also generates semantic labels extracted from the images and text in the search box, in descending order of importance (Figure 17:right). The designer can respond to the system's suggestions, e.g. by removing the word "room", or adding a new term, e.g. "cozy looking coffee house". In this form of mixed-initiative *Discoverability*, the user and system each interpret the other's actions, then produce output that the other interprets in turn. Finally, if the user is stuck, May AI offers system-initiated interaction via the "suggest-o-matic" dial ⓖ, which suggests new images, search terms, weighted tag clouds, and color palettes ⓗ. Designers can choose among different agents, e.g. asking for "less of these" images, or adjust the type of semantic analysis, e.g. to emphasize emotional content rather than identifying physical objects.

The various tools for adapting and modifying images and other mood board elements support *Appropriability*, including a tool palette ⓘ for cropping, resizing, recoloring and transforming images and text, an interactive color palette derived from selected mood board images ⓗ. The user's ability to modify both the type of semantic analysis and the resulting semantic labels, color palettes and tag clouds generated by the system, as well as the immediate reanalysis of each image as it is cropped, resized or otherwise transformed, allows users to adapt the content to meet current needs. Here, the system alternates agency with the user, adjusting its interpretations of the images as the designer changes them, while the designer reacts to the system's updated semantics and suggested images.

Finally, ImageSense supports *Expressivity*, with the emphasis on human agency for composition and reflection tasks. Designers retain full control when choosing and arranging images, text labels and other decoration, but may also seek system-initiated information, such as word clouds based on their preferred semantics, or color palettes derived from chosen images. In summary, ImageSense supports a highly iterative design process, that lets the user move fluidly through collection, composition and reflection phases, while fully controlling when and how to access intelligent assistance, creating an effective Human-Computer Partnership.

6.6 Discussion

The principles of *Discoverability*, *Appropriability*, and *Expressivity* capture different aspects of Human-Computer Partnerships as they evolve over time. Users require *Discoverability* to understand not only how the system works, but also how to modify their behavior to adapt to the constraints of the system. A key insight concerns the benefits of dynamic, in-context *feedforward*, where the system not only reveals its current interpretation of user input, but also shows future possibilities, helping users to better understand and master the system.

Users also need *Appropriability* to adapt interactive systems, from simple customization to more extensive remapping of input or output. However, the original discussion of co-adaptation [109] suggests something more, when users adapt systems in ways the system designers did not anticipate. This creates an interesting paradox: How can such designers create systems that users can adapt in unexpected ways? Yet this is not really a paradox, but rather the observation that, given appropriate characteristics, users adapt both physical and digital tools anyway, using strategies such as technical reasoning, to suit their needs. The challenge is how to create interactive systems with sufficiently rich underlying structures that users can usefully appropriate.

Finally, users, especially creative professionals, need *Expressivity*, such that variations in their input may be reflected in the output. Unfortunately, few systems view *Expressivity* as a goal, and instead create uniform, standardized output, with batch-oriented rather than dynamic, user-controlled personalization. Even more problematic are machine learning-led systems that throw away individual variation as "noise", rather than treat it as a rich source of information that both users and the system can learn from and exploit.

Table 2 summarizes which questions to ask and issues to address, analytically, critically and constructively, when applying the concepts and principles of Human-Computer Partnerships. These generative principles offer analytical and critical lenses for evaluating the learnability and interpretive flexibility [124] of systems, as well as for identifying missing features or characteristics of the system. They also serve as a useful tool for inspiring new interactive systems, asking designers to consider, at each point in the interaction, whether the user can discover relevant aspects of the system (*Discoverability*), appropriate them (*Appropriability*), or use them for individual expression (*Expressivity*). These principles also encourage designers of intelligent systems to reflect on how their systems both learn from and modify the user's behavior. When combined, these principles encourage new ways of designing effective Human-Computer Partnerships that account for the dynamic relationship between users and systems over time.

Note that Human-Computer Partnerships explore the ever-changing relationship between users and intelligent systems, without implying that they are coequal partners. Clearly human intelligence differs fundamentally from system "intelligence", with very different forms of awareness and agency. Incorporating system intelligence into a Human-Computer Partnership is not simply a question of choosing the most efficient algorithm, since the same algorithm can actively increase users' skills and enjoyment, or end up deskilling or replacing them, depending on the nature of the interaction. Ideal Human-Computer Partnerships take optimal advantage of each partner's strengths while accommodating their weaknesses, such that the combination is greater than either alone. The

Table 2. Questions that can be asked to apply the concepts and principles of Human-Computer Partnership analytically, critically and constructively.

Human-Computer Partnerships	Analytical	Critical	Constructive
Concepts			
Co-Adaptation	Can users reveal, interpret or modify the system's behavior?	Which aspects of the system are discoverable, appropriable and expressive?	How can we help users to both *adapt to* the system, and *adapt* it for new tasks and creative expression?
Reciprocal Co-Adaptation	Does the system reveal, interpret or modify the user's behavior?	Which characteristics of the user can the system discover, appropriate or display?	How do users and the system *adapt to* and *adapt* each other's behavior?
Principles			
Discoverability	Does the system reveal how it interpreted user behavior and show what options are currently available?	Can users discover and understand the system? Can the system interpret aspects of the user's behavior?	How can we present the system's interpretation of users' actions and reveal user-relevant features?
Appropriability	Does the system permit customization of the system or its features?	Can users create or modify the commands and features they need?	How can we help users personalize or redefine the system or its features?
Expressivity	Does the system transform individual input variation into expressive output?	Can users control how the system interprets their actions so as to generate rich or expressive output?	How can we help users dynamically control their expressive output?

generative principles of *Discoverability*, *Appropriability*, and *Expressivity* offer a path for creating such partnerships, while letting human users remain in control.

7 EXAMPLE 3: COMMUNITIES AND COMMON OBJECTS

The two previous examples show Generative Theories of Interaction for dealing with the relationship between a human and a computer in HCI. The challenge that motivates this third example is to address the multiplicities of human users and technologies rather than a single one, i.e. to embrace communities of practice surrounded by multiple technological tools and objects of work.

This perspective is grounded in the understanding that people always organize themselves together around shared objects and artifacts/instruments. Objects have been critically examined, e.g. by Heath & Luff [73] who propose to move beyond a focus on simple transactions of objects "*to support a range of capabilities from the simple exchange of objects, through the sharing of objects and to common views of the same object.*" They call for a more complex and dynamic understanding of objects and their roles between people in particular practices, such as the sharing of records of various sorts.

By focusing on objects as multiple and multi-layered entities, this perspective offers better ways of focusing on shareability over time, in communities of various kinds, by addressing the objects that these communities share, develop and interact through. How communities develop technology over time is a process of collaborative appropriation where the staring point is the shared practices and artifacts (see also [6]).

The perspective of Communities & Common Objects aims to study and construct qualities and mechanisms that help empower users to better understand and develop the technologies they use collaboratively. The goal is to innovate computing technology based on these ideas at both the conceptual and design levels. This perspective supports multiplicity and diversity in technology-supported communities by developing the means and language for more diverse communities of users to apprehend and develop their own joint technologies.

7.1 Theoretical Foundations

The perspective has its roots in the Russian socio-cultural research tradition of activity theory, dating back to Vygotsky [150] and Leontiev [105, 106]. It has been the basis of extended work since the 1980's to understand and develop perspectives on human shared practice and its development as well as on the side of tools, language and societal structures (Engeström [59]). This perspective also shares a foundation with, e.g., Dewey, Mead, Strauss and Lave (see Star [142]) through the joint focus on matter as conditions for situations. With a concept such as communities of practice these perspectives have been used to embrace the development of technology in real-life situations; they share a focus on dialectics and the collective as well as a pragmatic focus on situated actions.

7.1.1 Activity Theory in Psychology, Anthropology and Sociology. Activity theory has led to a wide and thriving interdisciplinary research community focusing often on learning and development, with a specific focus on collectives and the development of situated practices. This community has strong member groups all over the world in psychology, anthropology, sociology, organizational theory and more (see, e.g. [135]).

7.1.2 Activity Theory in HCI. Since the mid 1980's, activity theory has been explored as a basic perspective on human-computer interaction [22, 118]. In an attempt to break with cognitive science-based HCI, a theoretical platform has been established, using dialectical materialism [59, 116] to focus on human beings acting with technology in real-life situations [145, 154] (see also Star [142]).

Human activity is the analytic starting point. Human beings undertake actions and operations together, mediated by objects used as artifacts. For example, a group of people convenes in the Zoom video-conferencing system to plan their daily work. They may show each other what they are doing by sharing spreadsheets or documents through screen sharing. Operations from earlier generations of activity are crystallized, or reified into next generation artifacts. In Zoom, one participant is an owner of the meeting and all other participants can raise their hand to participate. In addition, artifacts are representations of certain modes of acting in the activity [22]. The Zoom example illustrates that, no matter how people actually collaborate in Zoom, it is based on a model with an owner who has a different status than the other participants and, e.g. cannot/is not supposed to use the raise-hand function. How this reification happens is illustrated in Bærentsen's analysis of the historical development of the hand gun [7]. The analysis points out how new organizing principles (war formations), new technologies (e.g. gun-powder and cartridges), training of soldiers, new affordances, etc. all have developed in a dialectical relationship with the hand gun, and hence have left traces in hand guns as they have developed historically. In a similar way, online meetings are developing alongside their tools, including, e.g. shared documents and spreadsheets. People collaborate and this collaboration is mediated through technology; more generally human activity is mediated by artifacts. Human beings are embedded in praxis, in routines and activities that they learn, do, and change, together and through their interaction with material objects and artifacts [22]. Technological mediation of human activity is hence the primary principle and focus of this perspective.

Artifacts mediate human relationships with technological objects, and artifacts are themselves such objects. They have the attention of human users in breakdown situations [19, 22, 24, 129] or in more deliberate design and building situations [24]. Objects are outcomes of building processes but they are also artifacts, or mediators that help users act on other objects, in ways they could not without using the mediator [13, 22]. Bødker & Klokmose [29] focus on the plurality of objects and artifacts and talk about *artifact ecologies*. In addition, Nicolini et al. [121] propose to understand objects as performing at least three types of work: Objects motivate and allow collaboration; they allow participants to work across different types of boundaries; and they constitute the fundamental infrastructure of the activity. They are also multiple, heterogeneous and potentially conflictual. Bardram [10], Sørgaard [146] as well as Bødker & Lyle [33] and Larsen-Ledet et al. [101, 102] have characterized cooperation as happening through interactive objects, as coordination around the object and co-construction of future use with the object.

Bødker & Klokmose [29], in developing the argumentation for mediation as a key principle, emphasize dialectical thinking as a way to understand things concretely in all their movement, change and interconnection, in unity with their opposite and contradictory sides. Since movement and change are essential parts of dialectical thinking, this approach focuses on development of use. Bødker & Klokmose address the development of artifact ecologies beyond singular artifacts in use by communities. This perspective is further developed by Bødker et al. [32, 34]. The focus of the *Communities & Common Objects* perspective is on the *social*, addressing both the shared practices that keep communities together and develop over time and the fact that people are always collaborating. Hence we talk about *common* objects (Figure 18).

In other words, common objects must be understood as always changing, and existing in a field between the well-known and the epistemic, i.e. how they seed the future. Accordingly it is important to always understand and address practices and technological objects *historically*, by looking back as well as looking towards the future and what they *may become*, i.e. how technological objects may change human practice. Addressing technological objects in use means focusing on *breakdowns* that happen in use [22, 129], when the common object stops supporting the activity. It is equally important to understand and address the seeds of future use embedded in the objects, which is really the core of a generative theory. These *epistemic objects* embody what is not yet known, and provide motivation for the creation of new knowledge [121].

Technological objects never exist alone, and *ecologies and their dynamics* are important, both analytically, critically and constructively. Technological objects exist in ecologies together with other objects used by people, not in singular and isolated activities. They participate in webs of activities and act as boundary objects [143] across activities.

However, the perspective does not only provide an analytic framing. By virtue of its emphasis on dynamics, it also provides a framework for critically apprehending the breakdowns and challenges of an empirical situation with or without the deployment of a particular technological design, and for constructing such designs with an emphasis on their future use.

7.2 Concepts of Communities & Common Objects

Communities & Common Objects is a generative theory that addresses both future technologies and re-understanding of the past. This holds even when one aims to analytically understand human activities and artifacts at specific points in time. Artifacts are noticed, as objects, in breakdown situations. Hence, the processes of designing such artifacts must emphasize how mediating artifacts are backgrounded in (future) use, and must critically and constructively explore breakdowns that involve both material objects and communication. Objects influence users' activity (whether intended or not) and they get appropriated and developed further in use (see also Kaptelinin & Bannon [87]). As a consequence, from this perspective, generative research processes always balance:

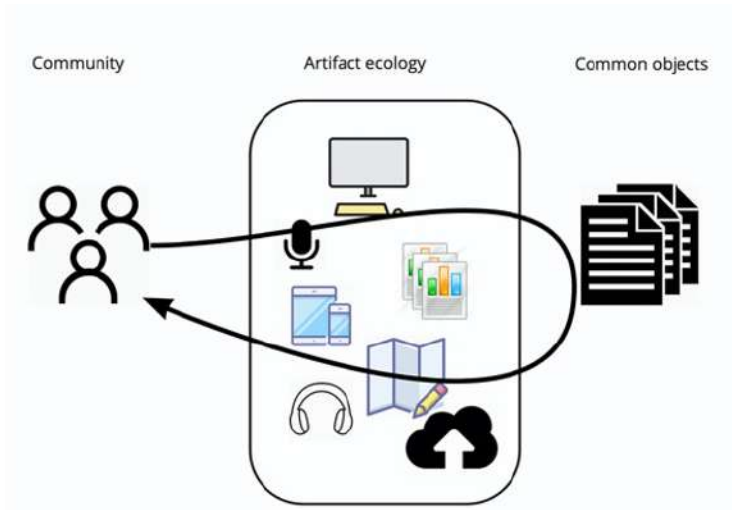

Fig. 18. Artifact mediation through common objects

(1) Empirical and participatory research with communities of human users, looking back, analytically and critically, at their past and current use activities, and looking ahead, constructively exploring and seeding possible change;
(2) Theoretical research to seed appropriate interim concepts and principles into the empirical process and to reflect critically and constructively upon the empirical findings; and
(3) Seeding of technological possibilities and alternatives to bridge theory and empirical research.

As introduced above, Communities & Common Objects is based on three concepts: *Common objects* are the center of activity for people in *communities* with joint practices, carrying out their joint activities through *artifact ecologies.*

It is important to empower users to better understand and develop the technologies they use together. This entails technological means and a theory-informed language for more diverse communities of users to share common objects, whether within or across communities. The generative principles below are therefore important for both the design process and the objects constructed for users to use. We utilize these principles to explore and assess technological designs for communities with common objects.

7.3 Generative Principles of Communities & Common Objects

Based on these theoretical underpinnings and concepts, we introduce the generative principles of Communities & Common Objects as follows (see also [6]).

7.3.1 Mediation through Common Objects. The underlying principle of this perspective is mediation. Common objects mediate human activity, as artifacts, in accordance with the basic principles of activity theoretical HCI. Common objects stand between the human users, they are material that gets turned into outcome, and they serve as mediating artifacts of the collaborative activity.

7.3.2 Material and Communicative Mediation. Common objects are simultaneously material, social and communicative, because common objects stand between people, both materially and communicatively, and mediate their joint activity. A particular focus of common interactive objects is the language and concepts related to cooperation and transfer of experiences across a community.

Bødker & Klokmose [30] point towards the use of conceptual blends to constructively focus on this aspect.

7.3.3 Malleability over time. Interaction needs to be understood as it unfolds over time and in the collaborative activities between people using technological artifacts and objects. The unfolding over time is a matter of development of community practice, individual routines, technology in use with respect to the *history*, the *present* and the *epistemic*, i.e. the not yet known, possible and desired future. Appropriation relates to development in use of both the common objects and the collaborative use activities. Objects can be collaboratively tailored and appropriated in use [11], and hence common objects should be understood as changeable and malleable over time.

7.3.4 Multiplicity of Artifact Ecologies. Human use of particular artifacts and objects happen across activities, configurations of people, applications and devices, and it is important for the interaction to embrace such transitions and substitutions [29].

7.4 Applying the Generative Principles: CaseLine

Prototypical examples of communities with interaction centered around common objects include everyday artifacts, family calendars [28], parental leave planning [37, 38], shared art curation [36] and community artifact ecologies for food sharing [32, 34]. The mediating qualities of such common objects, and their potential for sharing and boundary drawing within and between communities have been the starting point for developing the generative theory.

In the following, we present an example that has been instrumental in formulating the principles. This study was originally seen as a project to address the interaction and collaboration involved in the planning and control of parental leave in Danish municipalities (summary from Bødker & Grönvall [28]). Planning and control of parental leave involves several citizens (the parents), a municipal office, and several other stakeholders such as the parents' employers and labor unions. We developed CaseLine as part of this project. For details regarding the study background, the design rationale, the method and the system itself, see [37, 38].

7.4.1 Analytical and Critical Power. With the general concern for *mediation*, the principle of *Material and communicative mediation* focuses on the parental leave planning process as a product of the negotiation between the parents and the surrounding stakeholders, in particular how the parents may use their rights for parental leave, as determined by national legislation and managed by the municipality, while making the most of the payment that either parent gets from their employers. The legislation constrains this planning by the fact that the rights of the parents are interwoven with one-another. The municipal office is an important source of advice as well as the body that approves the final plan.

As a critical lens, the principle of *Material and communicative mediation* specifically raises issues regarding establishing the best possible solution in terms of total leave-time, split between mother and father, total income during the leave period, the possibility of spending leave-time together, and saving leave-time for later. Conversations and thoughts about "what if" scenarios are mediated through common objects, shared only by the parents. Parents often do not have all the information necessary to calculate these scenarios, and the legislation is difficult to work with due to its inherent flexibility. As part of the counseling process, these "what if" scenarios can be explicitly shared with a caseworker.

The principle of *Malleability over time* specifically focuses on the fact that the planning period for parental leave is up to nine years per child. Parents may desire to change their plan, e.g. when the time comes for the child to start daycare. Over the nine-year period, many events may happen that make a change of plan necessary: New jobs, additional children, etc. Consequential changes

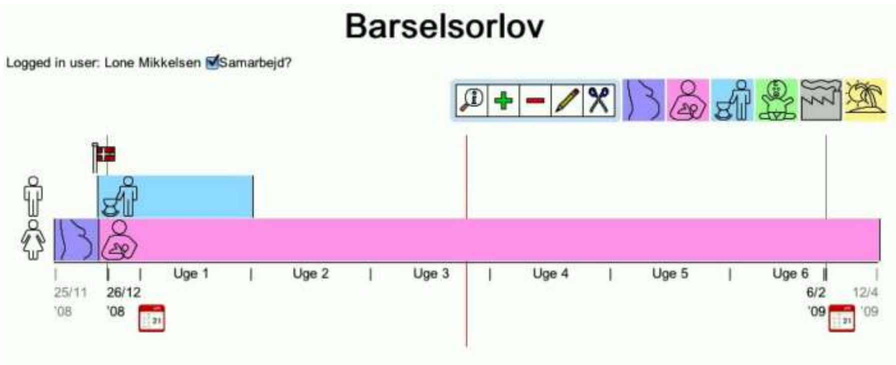

Fig. 19. The timeline-based CaseLine calendar (Sandbox mode). The tick-box makes explicit sharing with caseworker possible. See details in [37].

need to be reiterated with all stakeholders and it is, from a critical lens, quite complicated to keep track of these changes since both the parents and the municipality mainly documents them as time slices, with documents that formalize the here and now at the time when a version of the plan was agreed.

Due to the complexity of plans and their economic consequences, sharing of experiences is attractive among friends and groups of parents. However, since the actual plans are highly dependent upon specific details, it is difficult to apply plans across pregnancies, and the municipal workers caution against such replication of plans. The actual care plan can be changed regarding the future, but time and money spent cannot be taken back. This balance illustrates the critical use of the principle of *Malleability over time*.

Multiplicity of artifact ecologies emphasizes how, when parents have decided on the best solution, it has to be communicated to and negotiated with the respective employers and added, e.g., to their HR systems. The manifest objects of the activity typically consist of a number of rule-books, and a number of documents that contractually specify the actual plan/agreement for each parent. While the parents may have discussed among themselves some kind of informal calculation of their options, typically the "what if" scenarios are explored orally over the telephone with a municipal caseworker. Sharing is therefore complicated and oftentimes the municipal office gets questions from expectant parents who cannot understand why and how their situation differs from that of their friends and relatives. This is also true for parents who search for information in the limited available Internet sources.

7.4.2 Constructive Power. We used *Material and communicative mediation* to address the many challenges of joint parental leave planning, resulting in a new concept and early prototypes of CaseLine [37]. CaseLine is a timeline/calendar-based planning and overview tool to be shared between parents, and towards employers and public authorities (Figure 19). CaseLine enables citizens to help themselves and each other in understanding, planning, and applying for parental leave funding through a timeline design. CaseLine is designed as a Common Object to facilitate communication and collaboration between citizens and municipal caseworkers, and ultimately also between the parents and, e.g., their employers and unions.

CaseLine helps families manage and visualize shared time resources by introducing two mediating Common Objects: The parental leave plan, which was until then rather implicit in the interaction between the parties, and the timeline itself, which maps the plan and its possible changes. Sharing these objects with authorities and employers is made explicit in the tools/instruments available. The parental leave time chunks can be placed on the timeline in order to explore particular plans, and the time chunks that are not yet accounted for remain visible. The timeline has a calendar and can be zoomed in and out. In some versions of the design we experimented with attaching the legal contractual documents to points on the timeline.

Applying the *Malleability over time* principle, CaseLine was designed to support and explore interaction in collaboration between the parents and with their municipality, in ways that could extend to other communities, from sharing parental plans with friends to sharing income agreements with labor unions and employers. CaseLine served as an epistemic object in that it seeded discussions among parents about, e.g. how they could share their plan or the principles behind it with their relatives, or with wider circles of friends. These discussions also related to the possibility of sharing CaseLine plans e.g. on Facebook groups or on a Labor Union or company website.

CaseLine supports open exploration between parents, as well as controlled sharing with the municipality, when parents allow access to their plan. In addition, the municipality can provide general plan elements on-line. The CaseLine elements may be tailored by, e.g. employers and unions, and used by parents as a basis for their agreement with the municipality, or by each parent in their agreement with their employer. However, a shared timeline raised a number of concerns even among the parents: "*Do parents of a child wish to share with each other all information about their interaction with employers, and government? Even if this is the case, can such openness also be assumed if the parents are divorced, but share joint custody over the child?*" [37]. Consequently, each citizen needs to be able to make a clear distinction between exploring possibilities and sharing information with other stakeholders such as employers and the municipality. CaseLine tools, hence, provide both an experimental sandbox and a more formal part where the actual agreements are made visible. CaseLine builds on malleable objects such as plans and rules, but also works with the notion that some objects need to be frozen as decisions are made, so that time and money spent do not come back when one tries to change the plan. Shared overview and mutual awareness were important for the cooperation between parents as well as when it came to sharing with municipal case-workers.

Multiplicity of artifact ecologies. CaseLine introduced the plan as a Common Object to mediate the parents' planning process as well as the communication with the municipality and the parents' employers. This object keeps track of the leave time spent and the one planned, and can be manipulated (and explored in sandbox mode) by the parents together or one at a time. The plan can be shared with the municipality, and also serves as a record of the formal contracts that have been made between the parties, something that is currently difficult for parents to do over an extended timescale, leading to numerous phone calls to the municipality.

In this presentation of the case study, we have made the generative principles more explicit than they were at the time of conducting the empirical research. Yet the basis of the theoretical concepts was present from the outset, and the notion of CaseLine was due to a critical analysis of the municipal interactive setup where documents, rather than Common Objects, were the driving force. Designing CaseLine as a Common Object helped investigate both the ability for parents to explore and optimize parental leave plans over the years, and the ability to share plans with other expectant parents as well as employers and municipal officers. The principles further helped critically appraise, e.g. how boundaries change when parents change employer or get divorced.

7.5 Applying the Generative Principles: InPlenary

The second case study was set up to explore Communities & Common Objects in the context of university lectures. The research was carried out by Henrik Korsgaard in collaboration with Clemens Klokmose and others [95].

InPlenary was developed using a Participatory Design process and tried out in four lectures in order to explore the potential and implications of the particular design. The project aimed to investigate how well the lecture situation as a whole was supported, how to distribute interfaces and functions, and how students and lecturers experienced the learning activities. Two colleagues were recruited to use InPlenary in two of the lectures and two of the authors of the study conducted the other two. One of these lectures was an invited lecture where students were asked to participate outside their normal study program. This allowed for a more in-depth focus on the study of use than a normal lecture.

7.5.1 Analytical and Critical Power. The principle of *Material and communicative mediation* is used to analytically address how, in a traditional lecturing situation, the first step in achieving an adequate understanding of the subject matter requires exposure to the common basis for the lecture with objects such as books, concepts, models and lecture slides and notes on the blackboard produced by the lecturer. Even abstract models and concepts are made commonly available through diagrams, examples and metaphors. Modern lecturing activities are *mediated* by more complex sets of artifacts and objects, for good or ill. Lecture theaters are typically equipped with projectors that the lecturer uses to project her presentations, made with, e.g. Microsoft Powerpoint or Apple Keynote. Secondary screens and blackboards/whiteboards may be available as well. Lecturers often share materials and course plans with students through on-line learning platforms. The students bring a number of technological devices to the lecture, to have access to such material, to take notes (for themselves or for sharing with their class or study group), or even to entertain themselves if bored (Facebook and YouTube are well-known in this capacity, but Bødker & Christiansen [27] even mention a more "exotic" case of poetry reading on an iPhone).

The starting point for InPlenary is largely based on a critical appraisal of the authors' own lecturing experiences. The critical analysis is summarized as follows [95]: When entering a lecture hall or a classroom, students have some common understanding of how to behave and what is about to happen. In case they have forgotten, the physical space gives strong indications toward its purpose and serves as an implicit common information space. In contrast, when seated and opening a laptop, the realm of possibilities is much more disconnected and detached from the primary activity. InPlenary critically tackles this detachment by focusing on the physical space as the scope for designing a digital system that supports co-located learning activities. Based on this, the principle of *Material and communicative mediation* has been key for the critical analysis of how to understand and change the possibilities for future lecturing.

Analytically, it is often the case that the class materials are predefined, and in that sense static, and that neither lecturers nor students have a chance to share their notes and other material during or after the class. This emphasizes the principle of *Multiplicity of artifact ecologies.*

Modern lecturing activities are characterized by *Multiplicity of artifact ecologies*: Projectors, the lecturer's PC, Powerpoint or Keynote presentations, additional screens, blackboards and whiteboards for everybody in the room to view, as well as materials and course plans shared through on-line learning platforms, student laptops and devices for note-taking, social media, etc. A critical investigation of *Multiplicity of artifact ecologies* available in and around the lecturing situation is a good starting point for further development.

Analytically, it is interesting to note how much Facebook and other tools are used for entertainment and as back-channels through which students interact with each other during the lecture. This has not been investigated specifically for this case study, although it points to elements of *Malleability over time* among students. In contrast, none of the above technologies offer much possibility for collaborative appropriation and object malleability, in particular across students and

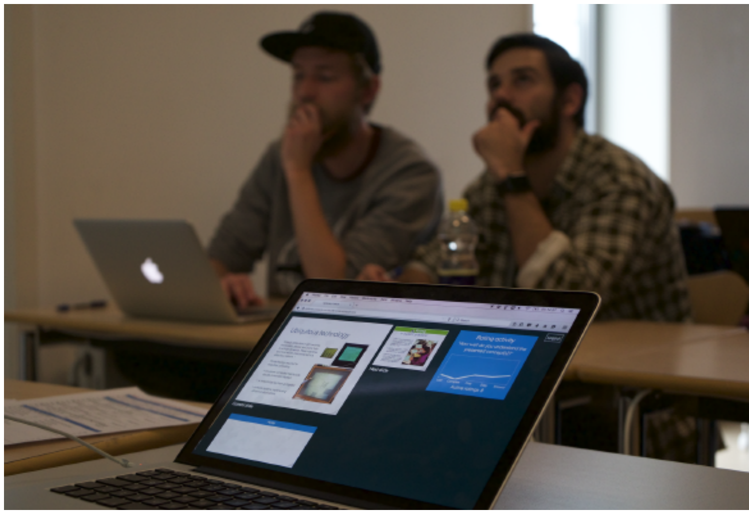

Fig. 20. InPlenary used in class. (Picture by Henrik Korsgaard)

lecturer. How to hand over control between lecturer and students and the role of the physical place can therefore benefit from an analysis through the *Malleability over time* principle.

7.5.2 Constructive Power. InPlenary [95] is a prototype system (Figure 20) built to support and develop university lectures to be actively mediated by the students' and lecturers' personal computers and software. It is based on the idea that it is possible to activate students in collaborative activities focusing on the lecture topic through their various brought-in devices in lectures and classes. InPlenary uses existing infrastructure and personal devices to distribute the lecture presentation across multiple devices, embed learning activities within the lecture, use personal devices as an entry-point for active participation, co-develop the lecture presentation as a common lecturing artifact throughout the lecture, and couple users to common lecturing objects based on their connection to a wireless access point.

The users of InPlenary are teachers and students. Its Common Objects are designed to mediate human activity by supporting existing practices of both lecturers and students, as illustrated in Figure 20. By activating students' personal devices as part of the primary activities, InPlenary applies the principle of *Mediation through common objects*. It leverages the duality of *Material and communicative mediation* to encourage active participation: InPlenary requires physical presence in the lecture for participation and subsequent access to the common objects produced during the lecturing activity. Participation in InPlenary is anonymous, but physical co-location serves as social moderation.

InPlenary integrates with existing teaching and note-taking practices to support broad participation, both from the perspective of the lecturer and the students, thereby applying the principle of *Multiplicity of artifact ecologies*. Using the system should add something extra to the lecture, without destroying traditional lecturing and use of slides. This also implies supporting as many personal devices as possible to avoid exclusion by incompatibility. InPlenary is set up to support the process of building a mutual understanding during the lecture and sees the slides and the information generated in the lecturing activities as common resources, following the principle of *Malleability over time*. As mentioned above, the collaborative appropriation and object malleability has yet to be constructively explored further.

7.6 Discussion

The principles and examples of Communities & Common Objects are all based on principles of collective human practices that are tied together by human action through tools on materials and through the coordination and communication that occurs when sharing tools and objects. The emphasis and fundamental focus of the approach is on sharing and collaboration. Hence, a basic assumption in both examples is the need to deal with multiple users, over multiple artifacts, places and times, with Common Objects as center pieces and shared foci.

The Communities & Common Objects perspective has roots in Activity theoretical HCI that dates back to the 1980's. This includes the basic principle of *Mediation through common objects*, the understanding of learning and breakdowns as important for human-computer interaction, and the principle of development over time. Other principles are newer, and some (object malleability and collaborative control over common objects) are still being explored in the second author's European Research Council project Common Interactive Objects (CIO).

The CaseLine example demonstrates that the Communities & Common Objects perspective can be used analytically, to understand how parents and organizations are dealing with the parental leave planning and rule compliance; critically, to reframe the then-existing document focus towards objects that persist over time and can be shared across persons and communities in various forms; and constructively, to build and explore prototypes that function through common objects, hence also realizing when this notion has limitations. Time is in several ways central to CaseLine, in all three lenses, since the main object of the activity, the leave plan, extends over long periods of time. Object malleability was explored in the construction of CaseLine regarding, e.g. how income information could be added by employers across cases, or how the structure of a parental leave plan (without personal information) could be shared among groups of expectant parents.

The InPlenary example more actively deployed the generative principles. It is based on an overall critical analysis of isolated applications as the main artifacts of a lecture, with the notion of common objects at the outset. The three lenses were used to further focus on the coming and going of a multiplicity of devices, on the duality of *Material and communicative mediation* that is happening specifically in and around lectures, on the control over shared objects understood through the different roles of lecturers and students, and on the possibilities of collaborative appropriation of the tools for specific lecture series.

The Communities & Common Objects perspective is actionable in various ways. First, each of the principles gives a particular analytical, critical and constructive focus depending on where in the research process the activity is found. Second, the principles have made it possible to derive useful questions: For example, the principle of *Malleability over time* leads to analytic recommendations such as: Identify traces of past routines, past artifacts and past collaboration in the activity; Identify the current activity with its actions and operations, and consider who does what and which breakdowns happen as the activity unfolds [23]; Are there uses of the activity artifact that point towards the future, and even seed how this change may happen? A critical application of the principle asks questions about what hinders the application of historic artifacts and past routines: What are the boundaries, time-wise, of the activity, and are those time boundaries appropriate, as in the case of InPlenary? What in the activity and artifacts can make a break away from the current possible? Constructively, the principle supports the development of new routines, new uses, new artifacts and new forms of collaboration over time (see also, e.g. Bardram & Bertelsen [11]).

The two examples are different since CaseLine focuses on extensive analytic work regarding the use context, whereas the starting point for InPlenary comes out of a critical analysis. They both lead, however, to constructive design of Common Objects to be used for *mediation* within and across communities, hence emphasizing the generative nature of the approach.

Since Common Objects are inherently epistemic and hence point towards not-yet-understood uses, challenges, breakdowns and trouble, it is evident that use develops over time, and that

Table 3. Questions that can be asked to apply the concepts and principles of Communities & Common Objects analytically, critically and constructively

Communities & Common Objects	Analytical	Critical	Constructive
Concepts			
Common Object	What common objects can be identified and how do they serve the joint activity?	Do common objects work for users' joint activity?	Are other common objects possible/interesting?
Community	What is the community of use and its joint practices?	What limits the community and its joint practices and activities?	Can the community and its practices be developed?
Artifact Ecology	Which artifacts are used by the community in their joint activity? How are the artifacts connected in the ecology?	What are the limitations and possibilities of these artifacts?	Could new artifacts be introduced to aid the development of the joint activity? Could existing artifacts be reconsidered for new purposes?
Principles			
Mediation through common objects	What mediators and objects are present? Who are the collaborators? What are the joint practices? What is the joint purpose of the collaborative activity?	Are mediators and objects appropriate for collaborative practices, and joint purpose?	How may mediators and objects serve joint practices and the purpose of the joint activity better?
Material and communicative mediation	How is material and communicative mediation supported in the joint activity? How do common objects reflect and support this?	What helps or hinders material and communicative mediation in collaborating around common objects?	How may concepts and objects be used to better support collaboration around common objects e.g. though conceptual blending?
Malleability over time	What traces of past routines are present? What breakdowns happen in use? What seeds of future use can be identified? How is learning happening?	How may traces of the past be made visible? What alternatives exist for breaking away from existing practices, mediators and objects? How is learning hindered?	What new forms of collaboration can be mediated? How may changed epistemic objects be supported? How may learning happen?
Multiplicity of artifact ecologies	How are the activity, mediators and objects in focus embedded with other activities, artifacts and objects? How do mediators apply to multiple activities and objects?	(How) may replacement and substitution of objects and mediators support the joint activity in order to remove breakdowns and hindrances?	(How) may objects and mediators from other parts of the artifact ecology be used to support the particular joint activity?

design needs to concern itself with development in and of use, e.g. appropriation. Both examples show that prototypes can be used in this capacity as they help users get hands-on experience and deploy current and past practices as well as imagine the future [22, 117] and foster further research [86]. This is not true of standard prototyping processes. Indeed, it is common in HCI to consider prototypes as mainly objects that can be analyzed and tested against the immediate practice of the users.

In table 3 we summarize questions to ask and issues to address, analytically, critically and constructively, from the perspective of the principles of Communities & Common Objects. On the one hand, these questions focus on common objects and mediators, while on the other hand they open up to connected joint activities, objects and mediators, history and context. These questions represent a very simplified summary of many more cases than discussed here and of 35 years of theoretical research from the very early ideas of acting through the interface [22] and the emergence of the area of Computer Supported Collaborative Work. In this time frame many technological mediators have come and gone, e.g. online virtual worlds, and others have dramatically changed the lives of us all, in particular smartphones and mobile technologies [25, 26]. Some of these changes have been slow, while others have happened fast and with dramatic effects. As a generative theory of interaction, this perspective helps account and design for such time-unfolding processes.

8 DISCUSSION

Our goal is to evolve the field of Human-Computer Interaction from the creation of individual "point designs" to a more grounded, theory-driven approach. After first outlining the broad similarities and differences among the three examples described above, we discuss the creation of Generative Theories of Interaction according to the three criteria defined in section 2.1:

(1) grounded in a theory of human activity and behavior with technology;
(2) amenable to analytical, critical and constructive interpretation; and
(3) actionable through the theory's concepts and generative principles.

8.1 Similarities and differences

The three example theories share a concern for the (mediating) role of technology in human use, in contrast to, for example, models of information transmission or interaction as dialogue [77]. In particular, they share the fundamental goal of understanding tool use, which we embrace in a broad sense: from embodiment and types of experience to the interaction among people through technology. All three examples seek to understand how users appropriate technology, and the ways that technological artifacts allow and support appropriation. Each explores how to design for appropriation during interaction and over time, and how routines developed in practice can be reified into future artifacts.

Beyond these similarities, each theory addresses a different set of concerns, which we lay out in the simplified design space shown in Figure 21, where the three axes represent time (x-axis), people (y-axis), and location (z-axis). This design space captures different time scales: from the immediacy of a specific interaction technique; to patterns of interaction that develop as a user performs complex creative tasks over minutes or hours; to the ways in which users collaborate with each other, mediated by technology over months or years. Similarly, we can accommodate individuals, pairs, small groups, or large numbers of users, and consider interaction at a single location, a few selected locations, or a large number of locations. Different points along these axes result in significantly different design challenges and decisions. Selecting points within this design space affects not only how we characterize the details of the interaction, but also the units of analysis and the measures of success, which, in turn, affect the choice of underlying theory.

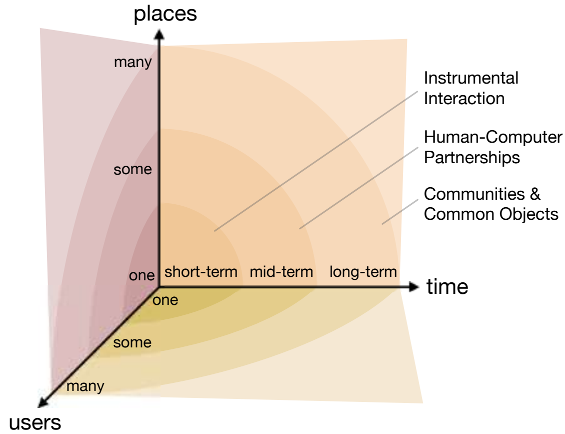

Fig. 21. Interactive systems increase in complexity when they encompass multiple points in time, multiple users, and multiple places. The three examples address different, but overlapping areas within this design space.

Each of the three generative theories of interaction covers a different area within this design space: Instrumental Interaction focuses on an individual's needs, in one location, at one point in time (central area in Figure 21); Human-Computer Partnerships focus on an individual's or a small group's needs, in a few locations, over extended periods of time (middle area in Figure 21); and Communities & Common Objects focus on multiple users' needs, in multiple locations, over even longer periods of time (larger area in Figure 21). Of course, these different areas of the design space overlap and influence each other, and the three theories build upon and affect each other. We chose this diversity of examples to illustrate the generality of Generative Theories of Interaction. By explicitly considering a multiplicity of use contexts within this rich design space, we to provide a constructive approach for addressing the ever-increasing complexity of interactive systems.

8.2 Theoretical grounding

Together, the three examples illustrate the breadth of existing theories of human behavior that can be recruited to create a generative theory: From ecological psychology (affordances) and cognitive science (human tool use, technical reasoning) to social sciences (activity theory) and evolutionary biology (co-adaptation). We believe that many other theories are potential candidates for generative theories, some well-known and close to HCI, others less so. For example, theories that are familiar to HCI researchers such as Distributed Cognition [75], Situated Action [145] or Embodied Cognition [54] appear to be good candidates, since they typically lack actionable principles that make them directly usable in a constructive way. More generally, most existing HCI theories focus on analytic and descriptive elements and do not clearly suggest or guide a constructive approach. In our experience, theories that recognize the dynamic aspects of technology and changes in use over time have proven particularly suitable for generative theories of interaction.

A number of existing frameworks and interaction models could benefit from a firmer theoretical grounding, such as Tangible Interaction [80] and Reality-Based Interfaces [81], to improve their generative power. In our experience, seeking such theoretical support helps strengthen the model or framework by asking new questions about it, finding additional relevant theory, and identifying or distilling more powerful concepts and principles.

8.3 Analytical, critical and constructive interpretation

For each theory, we illustrated the importance of using the analytical, critical and constructive lenses when applying the concepts and principles to generate new insights. We do not claim, however, that all concepts and principles apply equally well to all lenses or all applications. This is partly due to the different levels of specificity of the concepts and principles, and partly because some empirical cases may be better suited to some concepts, principles or lenses. Even so, we find it fruitful to explore as many combinations as possible, since this inspires new perspectives and original ideas.

The analytical, critical and constructive lenses connect prototypes and other artifacts to the theoretical basis on the one hand, and to empirical settings on the other. All contribute to the generative quality of the theory by feeding into each other: The analytical lens helps frame existing artifacts and phenomena in terms of the theory's concepts and generative principles, feeding into the critical lens, which helps discover breakdowns and missed opportunities; which in turn feeds into the constructive lens, which helps envision new solutions.

Tables 1, 2 and 3 illustrate how these lenses can be framed in terms of questions that can be asked of the artifacts or phenomena being studied, and we invite readers to ask them directly when applying these theories. Although these theories address different foci, at different levels of scale, they are sufficiently consistent that they can be explored in pairs (or more), both to identify critical differences as well as generate new ideas. We have found that these three lenses offer an efficient way to activate theory-driven work in empirical settings, especially when engaging users in participatory design activities such as brainstorming or hands-on prototype exploration.

8.4 Actionable through the theory's concepts and generative principles

A key goal of generative theories of interaction are that they are "actionable", i.e. a particular theory's concepts and generative principles can be applied within the HCI researcher's design process to produce new insights. Since technologies and their use change constantly, an important part of the HCI research process is to reflect on the past: how activities have been routinized and appropriated, how users have developed technological workarounds, and how reification traces have been left in technologies over time. These questions are equally important to the design process and several authors have attempted to make these types of analysis actionable in design [23, 31]. However, as stated earlier, this paper focuses on the HCI *research* design process which seeks to expand the "adjacent possible" [85] set of design possibilities, rather than on design *practice*, where one must address the needs of a client.

Generative Theories of Interaction help address the future in constructive ways. We have systematically applied these theoretical concepts to construct alternatives, e.g. Bertelsen & Nielsen's [20] exploration of augmented reality in a wastewater plant and Mackay's [110] exploration of novel interactive paper applications. We also apply these concepts in participatory design workshops, and find, time and again, that they inspire participants to re-frame their designs and generate more innovative solution spaces. The authors all teach advanced interaction design courses where students use these concepts to critique systems in the HCI literature and generate specific, novel ideas for improving them.

Since these concepts and principles are presented at different levels of detail, it seems likely that some will be more or less appropriate for particular design activities. For example, the principle of *Polymorphism* addresses a fine level of detail and can be used to analyze and generate the individual commands and tools of an interactive system. By contrast, the principle of *Development and Malleability* addresses a coarser level of detail and will more likely influence the overall concept of a system.

8.5 Generating generative theory

Creating a generative theory of interaction requires an iterative process where concepts and principles are constructed, embedded in new designs, evaluated and critiqued. This involves connecting a theory of human activity that offers abstract knowledge but little or no generative power ("theory" level, Fig.3), to design innovations from products or the HCI literature ("artifact" level, Fig.3).

For example, consider the extensive HCI literature on pointing. The theory level includes a large body of literature on human aimed movement, including the well-known Fitts' law [60]. This theory, even though based on extensive empirical work, is abstract with respect to generating more efficient pointing techniques. The artifact level includes a large collection of pointing techniques, what we call "point designs", with well-known advantages and limitations. We can construct a generative theory of interaction by identifying specific concepts and actionable principles that inspire novel pointing techniques. For example, Balakrishnan [9] distinguishes between the concepts of target-aware vs. target-agnostic techniques, depending upon whether or not the system needs full knowledge of the potential targets; and identifies the principles of improving pointing by either reducing the distance to the target or increasing the target width. These concepts and principles led to a more systematic exploration of the design space and to more efficient pointing techniques, such as the Bubble Cursor [70] and DynaSpot [48].

We believe that many other areas within HCI are amenable to creating or improving generative theories of interaction. We recommend the following framework or meta-model to progressively shape a new theory:

- Focus on a single underlying theory (or several closely related theories);
- Choose concepts and principles for their consistency and complementarity;
- Assess the analytical, critical and constructive power of the concepts and principles (Fig. 3);
- Create specific questions to ensure that concepts and principles are actionable when exploring new designs;
- Pay attention to where concepts and principles resist or break down and revise them accordingly; and
- Engage with users throughout the process.

Note that we do not necessarily recommend starting from an existing theory and "making it generative". Indeed, the three theories presented in this article were framed progressively, based on successive projects, sources of inspiration, trial and error. Even though we introduced each of the three examples above in a "top-down" fashion, starting with the underlying scientific theory, followed by the concepts and principles of the generative theory, and concluding with their application in novel interactive systems, historically this is not how these theories came to be.

Instead, these research projects all started in different places, and each generative theory of interaction emerged from earlier projects and intuition as much as from an existing scientific theory. Instrumental Interaction began as an interaction model inspired by empirical observations of human tool use and the frustration of so-called direct manipulation interfaces. It became a generative theory once we identified the principles of *Reification*, *Polymorphism* and *Reuse*, and later strengthened the links with human tool-use theories from psychology and cognitive science. Human-Computer Partnerships began with observations of how people customize their tools and re-purpose different features to address current needs. This led to observations about how people both adapt the software, but also adapt to it, and share both types of understanding with others, which echoes the process of co-adaptation in evolutionary biology. The notion of reciprocal co-adaptation was based on the observation that current machine learning and other AI techniques can both learn from and adapt to users as well as affect their behavior. Communities & Common

Objects, like Instrumental Interaction, started with an interest in tools, but focused on the practices in which they are used. Realizing that practices and collaborative tool use develop dynamically led to an explicit focus on change over time.

We encourage HCI researchers to develop new generative theories of interaction by taking advantage of other empirically based concepts from a variety of disciplines. For example, it would be interesting to see if theories related to distributed cognition [79] or boundary objects [143] could be transformed into generative theories of interaction. Note that we restricted our focus here to empirically grounded theories, but other researchers might find it useful to explore alternative types of grounding theory. For example, Abend [1] identifies four other types of sociological theory—historical, conceptual, normative, and philosophical—that might be more or less amenable to developing generative theories of interaction.

We also need to establish methods for evaluating, comparing and measuring the effectiveness of specific generative theories of interaction. This is difficult since they do not, by themselves, guarantee better designs, but always involve the creativity of the designers who use them. Our assessments thus far have been empirical, based on how well they inspire novel design ideas, and the effectiveness of those ideas. We find that the sequence of applying analytical, critical and constructive lenses sparks ideas and leads us in new directions. These lenses also provide a basis for comparing and assessing the value of each theory in principled ways.

Ideally, future work should address the translation of Generative Theories of Interaction into specific design practice. We have taught the generative theories of interaction presented in this paper to our students for years, and have extensive informal evidence that the generative principles help them create more innovative solutions, as assessed by outside juries unfamiliar with these principles. Practical design methods such as Generative Walkthroughs [107] suggest an approach for incorporating these theories into practical design methods and processes for practitioners.

9 CONCLUSION

This article introduces Generative Theories of Interaction, a specific type of generative theory that spans high-level theory derived from the natural and social sciences, through HCI-specific concepts that can be operationalized into actionable principles, which can then be applied using analytical, critical and constructive lenses. Each Generative Theory of Interaction should be grounded in established scientific theory rather than on intuition or anecdotal evidence, and bounded in scope to address a specific context or set of issues. We have found it productive to engage at multiple levels of abstraction, revisiting different theoretical concepts and principles as we create interactive systems that illustrate or test our theories or inspire other work.

In their analysis of the roles of theoretical concepts in HCI, Oulasvirta & Hornbæk [77] argue that concepts must inform design towards better choices. They call for "*propositions about interaction: theoretic statements that link the determination between humans and computers, given certain starting conditions and boundary conditions*" and argue that "*those conditions link theories to design and equip researchers/designers with some foresight that has been missing.*" Generative Theories of Interaction contribute to this goal by providing a more principled approach to the creative aspect of designing novel interactive systems.

Our contributions include a definition of Generative Theories of Interaction, a rationale for creating them, and three specific generative theories of interaction that show how this approach can be applied in different domains. We discuss each theory with respect to its analytical, critical and constructive potential, drawing examples from a diverse set of projects. Together, they demonstrate the scope and flexibility of Generative Theories of Interaction, and how they bridge the gap between empirical investigation, theory development and artifact creation.

The goal of this article is to provide inspiration and a specific template for creating generative theories of interaction that address new areas within the design space of interactive and socio-technical systems. We encourage HCI researchers to embrace Generative Theories of Interaction as tools for enhancing creativity, bringing new possibilities to life and exploring them in a principled way.

Future work includes creating new generative theories of interaction, devising better evaluation criteria, and developing design methods that explicitly embrace generative theory, both for the exploratory design processes needed in research and more widely for design processes outside academia. The ultimate value of a generative theory of interaction is to affect practice and lead to tools and systems that are better adapted to their users and contexts of use. We call upon the research community to critically scrutinize existing point designs in the light of theoretical constructs and to explore and synthesize new generative theories. We invite both HCI researchers and practitioners to join us in discussing, critiquing and formulating Generative Theories of Interaction to establish stronger scientific foundations for the field of Human-Computer Interaction.

ACKNOWLEDGMENTS

We thank the following people for their direct contributions: Henrik Korsgaard for the work on inPlenary, Gopinath Kannabiran for finding and helping us discuss Gergen's work; as well as Caroline Appert, Mirzel Avdic, Kim Halskov, Gopinath Kannabiran, Clemens Klokmose, Ida Larsen-Ledet, Nolwenn Maudet, and Nicolas Taffin, who read and commented on earlier versions of this article. We also thank the participants in the CIO Sandbjerg seminar for fruitful discussions and the anonymous reviewers for their detailed and constructive feedback.

This work was partially supported by European Research Council (ERC) Advanced Grants № 321135 "CREATIV: Creating Co-Adaptive Human-Computer Partnerships" (P.I. Wendy Mackay), № 695464 "ONE: Unified Principles of Interaction" (P.I. Michel Beaudouin-Lafon) and № 740548 "CIO: Common Interactive Objects" (P.I. Susanne Bødker) under the European Union's Framework 7 and Horizon 2020 research and innovation programmes.

REFERENCES

[1] Gabriel Abend. 2008. The Meaning of 'Theory'. *Sociological Theory* 26, 2 (2008), 173–199. https://doi.org/10.1111/j.1467-9558.2008.00324.x arXiv:https://doi.org/10.1111/j.1467-9558.2008.00324.x

[2] Jessalyn Alvina, Carla F. Griggio, Xiaojun Bi, and Wendy E. Mackay. 2017. CommandBoard: Creating a General-Purpose Command Gesture Input Space for Soft Keyboard. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17)*. ACM, New York, NY, USA, 17–28. https://doi.org/10.1145/3126594.3126639

[3] Jessalyn Alvina, Joseph Malloch, and Wendy E. Mackay. 2016. Expressive Keyboards: Enriching Gesture-Typing on Mobile Devices. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. ACM, New York, NY, USA, 583–593. https://doi.org/10.1145/2984511.2984560

[4] Jessalyn Alvina, Chengcheng Qu, Joanna McGrenere, and Wendy E. Mackay. 2019. MojiBoard: Generating Parametric Emojis with Gesture Keyboards. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI EA '19)*. ACM, New York, NY, USA, 1–6. https://doi.org/10.1145/3290607.3312771

[5] Caroline Appert, Michel Beaudouin-Lafon, and Wendy E. Mackay. 2004. Context matters: Evaluating Interaction Techniques with the CIS Model. In *People and Computers*. Springer Verlag, Leeds, United Kingdom, 279–295. https://doi.org/10.1007/1-84628-062-1_18

[6] Mirzel Avdic, Susanne Bødker, and Ida Larsen-Ledet. 2021. Two Cases for Traces: A Theoretical Framing of Mediated Joint Activity. *Proc. ACM Hum.-Comput. Interact.* 5, CSCW1, Article 190 (April 2021), 28 pages. https://doi.org/10.1145/3449289

[7] Klaus B. Bærentsen. 1989. Mennesker og maskiner. In *Et virksomt liv*, Mariane Hedegaard, Vagn Rabøl Hansen, and Sven Thyssen (Eds.). Aarhus Universitetsforlag, Aarhus, Denmark, 142–187.

[8] Ryan S. Baker. 2016. Stupid Tutoring Systems, Intelligent Humans. *International Journal of Artificial Intelligence in Education* 26, 2 (2016), 600–614. https://doi.org/10.1007/s40593-016-0105-0

[9] Ravin Balakrishnan. 2004. "Beating" Fitts' Law: Virtual Enhancements for Pointing Facilitation. *Int. J. Hum.-Comput. Stud.* 61, 6 (Dec. 2004), 857–874. https://doi.org/10.1016/j.ijhcs.2004.09.002

[10] Jakob Bardram. 1998. Collaboration, Coordination and Computer Support: An Activity Theoretical Approach to the Design of Computer Supported Cooperative Work. *DAIMI Report Series* 27, 533 (May 1998), 1–246. https://doi.org/10.7146/dpb.v27i533.7062 Ph.D. Thesis.

[11] Jakob Bardram and Olav W. Bertelsen. 1995. Supporting the Development of Transparent Interaction. In *Selected Papers from the 5th International Conference on Human-Computer Interaction (EWCHI '95)*. Springer-Verlag, London, UK, UK, 79–90. http://dl.acm.org/citation.cfm?id=646183.682900

[12] Olivier Bau and Wendy E. Mackay. 2008. OctoPocus: A Dynamic Guide for Learning Gesture-Based Command Sets. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology* (Monterey, CA, USA) *(UIST '08)*. ACM, New York, NY, USA, 37–46. https://doi.org/10.1145/1449715.1449724

[13] Michel Beaudouin-Lafon. 2000. Instrumental Interaction: An Interaction Model for Designing post-WIMP User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (The Hague, The Netherlands) *(CHI '00)*. ACM, New York, NY, USA, 446–453. https://doi.org/10.1145/332040.332473

[14] Michel Beaudouin-Lafon. 2004. Designing Interaction, Not Interfaces. In *Proceedings of the Working Conference on Advanced Visual Interfaces* (Gallipoli, Italy) *(AVI '04)*. ACM, New York, NY, USA, 15–22. https://doi.org/10.1145/989863.989865

[15] Michel Beaudouin-Lafon. 2019. A World Without Apps. Vision talk at *UIST 2019, ACM Symposium on User Interface Software and Technology*, New Orleans, LA. Video at https://www.youtube.com/watch?v=ntaudUum06E.

[16] Michel Beaudouin-Lafon and Henry Michael Lassen. 2000. The Architecture and Implementation of CPN2000, a Post-WIMP Graphical Application. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology* (San Diego, California, USA) *(UIST '00)*. ACM, New York, NY, USA, 181–190. https://doi.org/10.1145/354401.354761

[17] Michel Beaudouin-Lafon and Wendy E. Mackay. 2000. Reification, Polymorphism and Reuse: Three Principles for Designing Visual Interfaces. In *Proceedings of the Working Conference on Advanced Visual Interfaces* (Palermo, Italy) *(AVI '00)*. ACM, New York, NY, USA, 102–109. https://doi.org/10.1145/345513.345267

[18] Michel Beaudouin-Lafon, Wendy E. Mackay, Peter Andersen, Paul Janecek, Mads Jensen, Michael Lassen, Kasper Lund, Kjeld Mortensen, Stephanie Munck, Anne Ratzer, Katrine Ravn, Søren Christensen, and Kurt Jensen. 2001. CPN/Tools: A Post-WIMP Interface for Editing and Simulating Coloured Petri Nets. In *Applications and Theory of Petri Nets 2001*, José-Manuel Colom and Maciej Koutny (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 71–80.

[19] Olav Wedege Bertelsen and Susanne Bødker. 2003. *Activity Theory*. Vol. Chapter 11. Morgan Kaufmann, San Francisco, CA, USA, 291–324.

[20] Olav Wedege Bertelsen and Christina Nielsen. 2000. Augmented reality as a design tool for mobile interfaces. In *Proceedings of the 3rd conference on Designing interactive systems*. ACM, New York, NY, USA, 185–192. https://doi.org/10.1145/347642.347714

[21] Louis Bigo, Jérémie Garcia, Antoine Spicher, and Wendy E. Mackay. 2012. PaperTonnetz: Music Composition with Interactive Paper. In *Proceedings of the 9th Sound and Music Computing Conference*. SMC Network, Copenhagen, Denmark, 219–225. https://hal.inria.fr/hal-00718334

[22] Susanne Bødker. 1990. *Through the Interface: A Human Activity Approach to User Interface Design*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA.

[23] Susanne Bødker. 1993. Historical Analysis and Conflicting Perspectives - Contextualizing HCI. In *Selected Papers from the Third International Conference on Human-Computer Interaction (EWHCI '93)*. Springer-Verlag, Berlin, Heidelberg, 1–10. http://dl.acm.org/citation.cfm?id=646181.682738

[24] Susanne Bødker. 1999. *Computer applications as Mediators of Design and Use*. WorkingPaper. Department of Computer Science, Aarhus University.

[25] Susanne Bødker. 2006. When Second Wave HCI Meets Third Wave Challenges. In *Proceedings of the 4th Nordic Conference on Human-Computer Interaction: Changing Roles* (Oslo, Norway) *(NordiCHI '06)*. ACM, New York, NY, USA, 1–8. https://doi.org/10.1145/1182475.1182476

[26] Susanne Bødker. 2015. Third-Wave HCI, 10 Years Later—Participation and Sharing. *Interactions* 22, 5 (Aug. 2015), 24–31. https://doi.org/10.1145/2804405

[27] Susanne Bødker and Ellen Tove Christiansen. 2012. Poetry in motion: appropriation of the world of Apps. In *Proceedings of the 30th European Conference on Cognitive Ergonomics, ECCE '12*, Phil Turner and Susan Turner (Eds.). ACM, New York, NY, USA, 78–84. https://doi.org/10.1145/2448136.2448152

[28] Susanne Bødker and Erik Grönvall. 2013. Calendars: Time coordination and overview in families and beyond. In *ECSCW 2013*, Olav W. Bertelsen, Luigina Ciolfi, Maria Antonietta Grasso, and George Angelos Papadopoulos (Eds.). Springer London, London, United Kingdom, 63–81. https://doi.org/10.1007/978-1-4471-5346-7_4

[29] Susanne Bødker and Clemens Nylandsted Klokmose. 2011. The Human-Artifact Model: An Activity Theoretical Approach to Artifact Ecologies. *Human-Computer Interaction* 26, 4 (2011), 315–371. https://doi.org/10.1080/07370024.2011.626709

[30] Susanne Bødker and Clemens Nylandsted Klokmose. 2016. Dynamics, Multiplicity and Conceptual Blends in HCI. In *CHI '16*. ACM, New York, NY, USA, 2538–2548. https://doi.org/10.1145/2858036.2858530

[31] Susanne Bødker and Clemens Nylandsted Klokmose. 2012. Dynamics in Artifact Ecologies. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design* (Copenhagen, Denmark) *(NordiCHI '12)*. ACM, New York, NY, USA, 448–457. https://doi.org/10.1145/2399016.2399085

[32] Susanne Bødker, Henrik Korsgaard, and Joanna Saad-Sulonen. 2016. 'A Farmer, a Place and at Least 20 Members': The Development of Artifact Ecologies in Volunteer-based Communities. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing* (San Francisco, California, USA) *(CSCW '16)*. ACM, New York, NY, USA, 1142–1156. https://doi.org/10.1145/2818048.2820029

[33] Susanne Bødker and Peter Lyle. 2017. Community end-user development: Patterns, platforms, possibilities and problems. In *IS-EUD 2017 6th international symposium on end-user*, Khan J.V. , Soute I.A.C., De Angeli A., Piccinno A., and Bellucci A. (Eds.). Technische Universiteit Eindhoven, Eindhoven, Netherlands, 76–81.

[34] Susanne Bødker, Peter Lyle, and Joanna Saad-Sulonen. 2017. Untangling the Mess of Technological Artifacts: Investigating Community Artifact Ecologies. In *Proceedings of the 8th International Conference on Communities and Technologies* (Troyes, France) *(C&T '17)*. ACM, New York, NY, USA, 246–255. https://doi.org/10.1145/3083671.3083675

[35] Susanne Bødker and Marianne Graves Petersen. 2001. Design for Learning in Use. *Scand. J. Inf. Syst.* 12, 1-2 (Jan. 2001), 61–80. http://dl.acm.org/citation.cfm?id=372668.372682

[36] Susanne Bødker and Anna Maria Polli. 2014. Between initial familiarity and future use – a case of Collocated Collaborative Writing. In *COOP 2014 - Proceedings of the 11th International Conference on the Design of Cooperative Systems, 27-30 May 2014, Nice (France)*, C. Rossitta, L. Ciolfi, D. Martin, and B. Conein (Eds.). Springer, Cham, Germany, 137–154. https://doi.org/10.1007/978-3-319-06498-7_9

[37] Morten Bohøj, Nikolaj Gandrup Borchorst, Niels Olof Bouvin, Susanne Bødker, and Pär-Ola Zander. 2010. Timeline Collaboration. In *Proceedings of the 28th International Conference on Human factors in computing Systems. CHI '10*. ACM, New York, NY, USA, 523–532. https://doi.org/10.1145/1753326.1753404

[38] Nikolaj Gandrup Borchorst and Susanne Bødker. 2011. "You probably shouldn't give them too much information" – Supporting Citizen-Government Collaboration. In *ECSCW 2011*, S. Bødker, N. O. Bouvin, W. Lutters, V. Wulf, and L. Ciolfi (Eds.). Springer London, London, United Kingdom, 173–192. https://doi.org/10.1007/978-0-85729-913-0_10

[39] Adrien Bousseau, Theophanis Tsandilas, Lora Oehlberg, and Wendy E. Mackay. 2016. How Novices Sketch and Prototype Hand-Fabricated Objects. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 397–408. https://doi.org/10.1145/2858036.2858159

[40] Barry Brown, Alexandra Weilenmann, Donald McMillan, and Airi Lampinen. 2016. Five Provocations for Ethical HCI Research. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 852–863. https://doi.org/10.1145/2858036.2858313

[41] John Seely Brown and Paul Duguid. 1991. Organizational Learning and Communities-of-Practice: Toward a Unified View of Working, Learning, and Innovation. *Organization Science* 2, 1 (1991), 40–57. http://www.jstor.org/stable/2634938

[42] Frederik Brudy, Steven Houben, Nicolai Marquardt, and Yvonne Rogers. 2016. CurationSpace: Cross-Device Content Curation Using Instrumental Interaction. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces* (Niagara Falls, Ontario, Canada) *(ISS '16)*. ACM, New York, NY, USA, 159–168. https://doi.org/10.1145/2992154.2992175

[43] Bill Buxton. 2007. *Sketching User Experiences: Getting the Design Right and the Right Design* (first ed.). Morgan Kaufmann, San Francisco, CA, USA.

[44] Stuart K. Card, Allen Newell, and Thomas P. Moran. 1983. *The Psychology of Human-Computer Interaction*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA.

[45] John M. Carroll. 1997. Reconstructing Minimalism. In *Proceedings of the 15th Annual International Conference on Computer Documentation* (Salt Lake City, Utah, USA) *(SIGDOC '97)*. ACM, New York, NY, USA, 27–34. https://doi.org/10.1145/263367.263373

[46] John M. Carroll (Ed.). 2003. *HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[47] Tracy Cassidy. 2011. The mood board process modeled and understood as a qualitative design research tool. *Fashion Practice* 3, 2 (2011), 225–251. https://doi.org/10.2752/175693811X13080607764854

[48] Olivier Chapuis, Jean-Baptiste Labrune, and Emmanuel Pietriga. 2009. DynaSpot: Speed-Dependent Area Cursor. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. ACM,

New York, NY, USA, 1391–1400. https://doi.org/10.1145/1518701.1518911

[49] Fanny Chevalier, Pierre Dragicevic, and Christophe Hurter. 2012. Histomages: Fully Synchronized Views for Image Editing. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. ACM, New York, NY, USA, 281–286. https://doi.org/10.1145/2380116.2380152

[50] Marianela Ciolfi Felice, Sarah Fdili Alaoui, and Wendy E. Mackay. 2018. Knotation: Exploring and Documenting Choreographic Processes. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. ACM, New York, NY, USA, 1–12. https://doi.org/10.1145/3173574.3174022

[51] Marianela Ciolfi Felice, Nolwenn Maudet, Wendy E. Mackay, and Michel Beaudouin-Lafon. 2016. Beyond Snapping: Persistent, Tweakable Alignment and Distribution with StickyLines. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. ACM, New York, NY, USA, 133–144. https://doi.org/10.1145/2984511.2984577

[52] Charles Darwin (Ed.). 1859. *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life.* John Murray., Albemarle Street, London, England.

[53] Andy Dearden and Janet Finlay. 2006. Pattern Languages in HCI: A Critical Review. *Human–Computer Interaction* 21, 1 (2006), 49–102. https://doi.org/10.1207/s15327051hci2101_3

[54] Paul Dourish. 2001. *Where the Action is: The Foundations of Embodied Interaction.* MIT Press, Cambridge, MA, USA.

[55] Paul Dourish. 2006. Implications for Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Montréal, Québec, Canada) *(CHI '06)*. ACM, New York, NY, USA, 541–550. https://doi.org/10.1145/1124772.1124855

[56] Sebastian Draxler and Gunnar Stevens. 2011. Supporting the Collaborative Appropriation of an Open Software Ecosystem. *Computer Supported Cooperative Work (CSCW)* 20, 4 (08 May 2011), 403–448. https://doi.org/10.1007/s10606-011-9148-9

[57] Pelle Ehn. 1988. *Work-oriented design of computer artifacts.* Ph.D. Dissertation. Umeå University, Faculty of Social Sciences.

[58] Pelle Ehn and Morten Kyng. 1984. A tool perspective on design of interactive computer support for skilled workers. In *Proceedings from the seventh Scandinavian Research Seminar on Systemeering*, M. Sääksjärvi (Ed.). Helsinki School of Economics, Helsinki, Finland, 211–242.

[59] Yrjö Engeström. 2014. *Learning by Expanding: An Activity-Theoretical Approach to Developmental Research* (2 ed.). Cambridge University Press, Cambridge, United Kingdom. https://doi.org/10.1017/CBO9781139814744

[60] Paul M. Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology* 47, 6 (1954), 381–391. https://doi.org/10.1037/h0055392

[61] BJ Fogg. 2009. A Behavior Model for Persuasive Design. In *Proceedings of the 4th International Conference on Persuasive Technology* (Claremont, California, USA) *(Persuasive '09)*. ACM, New York, NY, USA, Article 40, 7 pages. https://doi.org/10.1145/1541948.1541999

[62] Jodi Forlizzi and Shannon Ford. 2000. The Building Blocks of Experience: An Early Framework for Interaction Designers. In *Proceedings of the 3rd Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques* (New York City, New York, USA) *(DIS '00)*. ACM, New York, NY, USA, 419–423. https://doi.org/10.1145/347642.347800

[63] Brian Gaines. 1991. Modeling and Forecasting the Information Sciences. *Information Sciences* 57–58 (1991), 3–22. https://doi.org/10.1016/0020-0255(91)90066-4

[64] Harold Garfinkel. 1991. *Studies in Ethnomethodology.* Wiley, Cambridge, UK. https://books.google.dk/books?id=zj_leg8-tIEC

[65] Kenneth Gergen. 1978. Toward generative theory. *Journal of Personality and Social Psychology* 36 (Nov. 1978), 1344–1360. https://doi.org/10.1037/0022-3514.36.11.1344

[66] James J. Gibson. 1979. *The ecological approach to visual perception.* Houghton, Mifflin and Company, Boston, Massachusetts.

[67] Bo Göranzon. 1988. The Practice of the Use of Computers. A Paradoxical Encounter between Different Traditions of Knowledge. In *Knowledge, Skill and Artificial Intelligence*, Bo Göranzon and Ingela Josefson (Eds.). Springer, London, United Kingdom, 9–18.

[68] Saul Greenberg. 2007. Toolkits and Interface Creativity. *Journal of Multimedia Tools and Applications (JMTA)* 32, 2 (2007), 139–159. http://www.jstor.org/stable/2634938

[69] Carla F. Griggio, Midas Nouwens, Joanna McGrenere, and Wendy E. Mackay. 2019. Augmenting Couples' Communication with Lifelines: Shared Timelines of Mixed Contextual Information. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. ACM, New York, NY, USA, Article 623, 13 pages. https://doi.org/10.1145/3290605.3300853

[70] Tovi Grossman and Ravin Balakrishnan. 2005. The Bubble Cursor: Enhancing Target Acquisition by Dynamic Resizing of the Cursor's Activation Area. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*

(Portland, Oregon, USA) *(CHI '05)*. ACM, New York, NY, USA, 281–290. https://doi.org/10.1145/1054972.1055012

[71] Han L. Han, Miguel A. Renom, Wendy E. Mackay, and Michel Beaudouin-Lafon. 2020. Textlets: Supporting Constraints and Consistency in Text Documents. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. ACM, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376804

[72] Sonia Harmand, Jason E. Lewis, Craig S. Feibel, Christopher J. Lepre, Sandrine Prat, Arnaud Lenoble, Xavier Boës, Rhonda L. Quinn, Michel Brenet, Adrian Arroyo, Nicholas Taylor, Sophie Clément, Guillaume Daver, Jean-Philip Brugal, Louise Leakey, Richard A. Mortlock, James D. Wright, Sammy Lokorodi, Christopher Kirwa, Dennis V. Kent, and Hélène Roche. 2015. 3.3-million-year-old stone tools from Lomekwi 3, West Turkana, Kenya. *Nature* 521, 7552 (2015), 310–315. https://doi.org/10.1038/nature14464

[73] Christian Heath and Paul Luff. 1996. Documents and Professional Practice: 'Bad' Organisational Reasons for 'Good' Clinical Records. In *Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work* (Boston, Massachusetts, USA) *(CSCW '96)*. ACM, New York, NY, USA, 354–363. https://doi.org/10.1145/240080.240342

[74] Martin Heidegger. 1962. *Being and Time*. Harper & Row, New York, NY, USA. https://books.google.fr/books?id=VKS7p3Vy6gAC

[75] James Hollan, Edwin Hutchins, and David Kirsh. 2000. Distributed Cognition: Toward a New Foundation for Human-computer Interaction Research. *ACM Trans. Computer-Human Interaction* 7, 2 (June 2000), 174–196. https://doi.org/10.1145/353485.353487

[76] Kristina Höök and Jonas Löwgren. 2012. Strong Concepts: Intermediate-level Knowledge in Interaction Design Research. *ACM Trans. Comput.-Hum. Interact.* 19, 3, Article 23 (Oct. 2012), 18 pages. https://doi.org/10.1145/2362364.2362371

[77] Kasper Hornbæk and Antti Oulasvirta. 2017. What Is Interaction?. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. ACM, New York, NY, USA, 5040–5052. https://doi.org/10.1145/3025453.3025765

[78] Edwin Hutchins. 1995. *Cognition in the Wild*. MIT Press, Cambridge, MA, USA.

[79] Edwin L. Hutchins, James D. Hollan, and Donald A. Norman. 1985. Direct Manipulation Interfaces. *Hum.-Comput. Interact.* 1, 4 (Dec. 1985), 311–338. https://doi.org/10.1207/s15327051hci0104_2

[80] Hiroshi Ishii and Brygg Ullmer. 1997. Tangible Bits: Towards Seamless Interfaces Between People, Bits and Atoms. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '97)*. ACM, New York, NY, USA, 234–241. https://doi.org/10.1145/258549.258715

[81] Robert J.K. Jacob, Audrey Girouard, Leanne M. Hirshfield, Michael S. Horn, Orit Shaer, Erin Treacy Solovey, and Jamie Zigelbaum. 2008. Reality-based Interaction: A Framework for post-WIMP Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. ACM, New York, NY, USA, 201–210. https://doi.org/10.1145/1357054.1357089

[82] Ghita Jalal, Nolwenn Maudet, and Wendy E. Mackay. 2015. Color Portraits: From Color Picking to Interacting with Color. In *Proceedings of the ACM International Conference on Human Factors in Computing Systems, 2015* (Seoul, South Korea). ACM, New York, NY, USA, 4207–421. https://doi.org/10.1145/2702123.2702173

[83] Yvonne Jansen, Pierre Dragicevic, and Jean-Daniel Fekete. 2012. Tangible Remote Controllers for Wall-Size Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. ACM, New York, NY, USA, 2865–2874. https://doi.org/10.1145/2207676.2208691

[84] Jeff Johnson, Teresa Roberts, William Verplank, David Smith, Charles Irby, Marian Beard, and Kevin Mackey. 1989. The Xerox Star: A Retrospective. *Computer* 22, 9 (Sept. 1989), 11–26. https://doi.org/10.1109/2.35211

[85] S. Johnson. 2010. *Where Good Ideas Come From*. Penguin Publishing Group, New York, NY, USA. https://books.google.fr/books?id=3H2Xg5qxz-8C

[86] Gopinaath Kannabiran and Susanne Bødker. 2020. Prototypes as Objects of Desire. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference* (Eindhoven, Netherlands) *(DIS '20)*. ACM, New York, NY, USA, 1619–1631. https://doi.org/10.1145/3357236.3395487

[87] Victor Kaptelinin and Liam J. Bannon. 2012. Interaction Design Beyond the Product: Creating Technology-Enhanced Activity Spaces. *Human–Computer Interaction* 27, 3 (2012), 277–309. https://doi.org/10.1080/07370024.2011.646930

[88] Stuart Kauffman. 1995. *At Home in the Universe: The Search for Laws of Self-Organization and Complexity*. Oxford University Press, Oxford, United Kingdom.

[89] Roberta L. Klatzky, Brian MacWhinney, and Marlene Behrman. 2008. *Embodiment, ego-space, and action*. Psychology Press, New York, NY, USA.

[90] Clemens Nylandsted Klokmose and Michel Beaudouin-Lafon. 2009. VIGO: Instrumental Interaction in Multi-Surface Environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. ACM, New York, NY, USA, 869–878. https://doi.org/10.1145/1518701.1518833

[91] Clemens N. Klokmose, James R. Eagan, Siemen Baader, Wendy Mackay, and Michel Beaudouin-Lafon. 2015. Webstrates: Shareable Dynamic Media. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*

(Charlotte, NC, USA) *(UIST '15)*. ACM, New York, NY, USA, 280–290. https://doi.org/10.1145/2807442.2807446
[92] Janin Koch, Andrés Lucero, Lena Hegemann, and Antti Oulasvirta. 2019. May AI? Design Ideation with Cooperative Contextual Bandits. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. ACM, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300863
[93] Janin Koch, Nicolas Taffin, Michel Beaudouin-Lafon, Markku Laine, Andrés Lucero, and Wendy E. Mackay. 2020. ImageSense: An Intelligent Collaborative Ideation Tool to Support Diverse Human-Computer Partnerships. *Proc. ACM Hum.-Comput. Interact.* 4, CSCW1, Article 045 (May 2020), 27 pages. https://doi.org/10.1145/3392850
[94] Janin Koch, Nicolas Taffin, Andrés Lucero, and Wendy E. Mackay. 2020. SemanticCollage: Enriching Digital Mood Board Design with Semantic Labels. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference* (Eindhoven, Netherlands) *(DIS '20)*. ACM, New York, NY, USA, 407–418. https://doi.org/10.1145/3357236.3395494
[95] Henrik Korsgaard. 2016. *Toward Place-Centric Computing: Making Place With Technology Together.* Ph.D. Dissertation. Aarhus Universitet.
[96] Vassilis Kostakos. 2015. The Big Hole in HCI Research. *interactions* 22, 2 (Feb. 2015), 48–51. https://doi.org/10.1145/2729103
[97] John W. Krakauer and Reza Shadmehr. 2011. Consolidation of Motor Memory. *Trends in Neuroscience* 21, 1 (Jan. 2011), 58–64. https://doi.org/10.1016/j.tins.2005.10.003
[98] Per Ola Kristensson. 2015. Next-Generation Text Entry. *Computer* 48, 7 (2015), 84–87.
[99] Per-Ola Kristensson and Shumin Zhai. 2004. SHARK2: A Large Vocabulary Shorthand Writing System for Pen-Based Computers. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology* (Santa Fe, NM, USA) *(UIST '04)*. ACM, New York, NY, USA, 43–52. https://doi.org/10.1145/1029632.1029640
[100] Gordon Kurtenbach and William Buxton. 1994. User Learning and Performance with Marking Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, Massachusetts, USA) *(CHI '94)*. ACM, New York, NY, USA, 258–264. https://doi.org/10.1145/191666.191759
[101] Ida Larsen-Ledet and Henrik Korsgaard. 2019. Territorial Functioning in Collaborative Writing. *Computer Supported Cooperative Work (CSCW)* 28 (08 May 2019), 391–433. https://doi.org/10.1007/s10606-019-09359-8
[102] Ida Larsen-Ledet, Henrik Korsgaard, and Susanne Bødker. 2020. Collaborative Writing Across Multiple Artifact Ecologies. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. ACM, New York, NY, USA, 1–14. https://doi.org/10.1145/3313831.3376422
[103] Germán Leiva, Nolwenn Maudet, Wendy Mackay, and Michel Beaudouin-Lafon. 2019. Enact: Reducing Designer–Developer Breakdowns When Prototyping Custom Interactions. *ACM Trans. Comput.-Hum. Interact.* 26, 3, Article 19 (May 2019), 48 pages. https://doi.org/10.1145/3310276
[104] Wassily Leontief. 1986. *Input-Output Economics.* Oxford University Press, Oxfort, United Kingdom.
[105] A.N. Leontiev. 1978. Activity, Consciousness, and Personality. http://marxists.org/archive/leontev/works/1978/index.htm
[106] Aleksei N. Leontiev. 1974. The Problem of Activity in Psychology. *Soviet Psychology* 13, 2 (1974), 4–33. http://www.comnet.ca/~{}pballan/Leontiev%281979%29.htm
[107] Danielle Lottridge and Wendy E. Mackay. 2009. Generative Walkthroughs: To Support Creative (Re)Design. In *Proceedings of the Seventh ACM Conference on Creativity and Cognition* (Berkeley, California, USA) *(C&C '09)*. ACM, New York, NY, USA, 175–184. https://doi.org/10.1145/1640233.1640261
[108] James E. Lovelock and Lynn Margulis. 1974. Atmospheric homeostasis by and for the biosphere: the Gaia hypothesis. *Tellus. Series A. International Meteorological Institute* 26, 1–2 (1974), 2–10.
[109] Wendy Mackay. 2000. Responding to cognitive overload : Co-adaptation between users and technology. *Intellectica* 30, 1 (2000), 177–193. https://doi.org/10.3406/intel.2000.1597
[110] Wendy E. Mackay. 1998. Augmented Reality: Linking Real and Virtual Worlds: A New Paradigm for Interacting with Computers. In *Proceedings of the Working Conference on Advanced Visual Interfaces* (L'Aquila, Italy) *(AVI '98)*. ACM, New York, NY, USA, 13–21. https://doi.org/10.1145/948496.948498
[111] Wendy E. Mackay. 2021 (in preparation). *DOIT: The Design of Interactive Things.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
[112] Wendy E. Mackay and Anne-Laure Fayard. 1997. HCI, Natural Science and Design: A Framework for Triangulation Across Disciplines. In *Proceedings of the 2nd Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques* (Amsterdam, The Netherlands) *(DIS '97)*. ACM, New York, NY, USA, 223–234. https://doi.org/10.1145/263552.263612
[113] Wendy E. Mackay, Ernest Holm Svendsen, and Bjarne Horn. 2001. Who's in Control? Exploring Human-Agent Interaction in the McPie Interactive Theater Project. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems* (Seattle, Washington) *(CHI EA '01)*. ACM, New York, NY, USA, 363–364. https://doi.org/10.1145/634067.634282
[114] Joseph Malloch, Carla F. Griggio, Joanna McGrenere, and Wendy E. Mackay. 2017. Fieldward and Pathward: Dynamic Guides for Defining Your Own Gestures. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing*

Systems (Denver, Colorado, USA) *(CHI '17)*. ACM, New York, NY, USA, 4266–4277. https://doi.org/10.1145/3025453.3025764
[115] Robert K. Merton. 1993. *On the Shoulders of Giants: A Shandean Postscript*. University of Chicago Press, Chicago, IL, USA). 1–348 pages.
[116] Reijo Miettinen and Jaakko Virkkunen. 2005. Epistemic Objects, Artefacts and Organizational Change. *Organization* 12, 3 (2005), 437–456. https://doi.org/10.1177/1350508405051279 arXiv:https://doi.org/10.1177/1350508405051279
[117] Preben Mogensen. 1992. Towards a Provotyping Approach in Systems Development. *Scandinavian Journal of Information Systems* 4, 1 (1992), 31–53. http://aisel.aisnet.org/sjis/vol4/iss1/5
[118] B.A. Nardi. 1996. *Context and Consciousness: Activity Theory and Human-computer Interaction*. MIT Press, Cambridge, MA, USA. https://books.google.dk/books?id=JeqcgPlS2UAC
[119] Bonnie Nardi and James Miller. 1991. Twinkling lights and nested loops: Distributed problem solving and spreadsheet development. *International Journal of Man-Machine Studies* 34, 2 (Feb. 1991), 161–184. https://doi.org/10.1016/0020-7373(91)90040-E
[120] W. Lawrence Neuman. 2000. *Social Research Methods: Qualitative and Quantitative Approaches*. Vol. 30. Allyn and Bacon, Boston, MA, USA. 1–594 pages.
[121] Davide Nicolini, Jeanne Mengis, and Jacky Swan. 2012. Understanding the Role of Objects in Cross-Disciplinary Collaboration. *Organization Science* 23, 3 (2012), 612–629. https://doi.org/10.1287/orsc.1110.0664 arXiv:https://doi.org/10.1287/orsc.1110.0664
[122] Donald A. Norman. 1983. Design Rules Based on Analyses of Human Error. *Commun. ACM* 26, 4 (April 1983), 254–258. https://doi.org/10.1145/2163.358092
[123] Donald A. Norman. 1988. *The psychology of everyday things*. Basic Books, New York, NY, USA.
[124] Wanda J. Orlikowski. 1992. The duality of technology: Rethinking the concept of technology in organizations. *Organization Science* 3, 3 (1992), 398–427.
[125] François Osiurak. 2014. What Neuropsychology Tells us About Human Tool Use? The Four Constraints Theory (4CT): Mechanics, Space, Time, and Effort. *Neuropsychology Review* 24, 2 (2014), 88–115. https://doi.org/10.1007/s11065-014-9260-y
[126] François Osiurak, Christophe Jarry, and Didier Le Gall. 2010. Grasping the affordances, understanding the reasoning. Toward a dialectical theory of human tool use. *Psychological Review* 117, 2 (2010), 517–540. https://halshs.archives-ouvertes.fr/halshs-00485348
[127] Antti Oulasvirta and Kasper Hornbæk. 2016. HCI Research As Problem-Solving. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 4956–4967. https://doi.org/10.1145/2858036.2858283
[128] Henry Petroski. 1994. *The Evolution of Useful Things*. Vintage Books, New York, USA.
[129] Volkmar Pipek and Volker Wulf. 2009. Infrastructuring: Toward an Integrated Perspective on the Design and Use of Information Technology. *J. AIS* 10 (May 2009). https://doi.org/10.17705/1jais.00195
[130] Pierre Rabardel and Pierre Beguin. 2005. Instrument mediated activity: from subject development to anthropocentric design. *Theoretical Issues in Ergonomics Science* 6, 5 (2005), 429–461. https://doi.org/10.1080/14639220500078179 arXiv:https://doi.org/10.1080/14639220500078179
[131] P. Rabardel and G. Bourmaud. 2003. From computer to instrument system: a developmental perspective. *Interacting with Computers* 15, 5 (Oct. 2003), 665–691. https://doi.org/10.1016/S0953-5438(03)00058-4
[132] Jef Raskin. 2000. *The Humane Interface: New Directions for Designing Interactive Systems*. ACM Press/Addison-Wesley Publishing Co., USA.
[133] Yvonne Rogers. 2004. New theoretical approaches for human-computer interaction. *Annual Review of Information Science and Technology* 38, 1 (2004), 87–143. https://doi.org/10.1002/aris.1440380103 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/aris.1440380103
[134] Yvonne Rogers. 2012. *HCI Theory: Classical, Modern, and Contemporary* (1st ed.). Morgan & Claypool Publishers, San Rafael, CA, USA.
[135] Wolff-Michael Roth. 2004. Activity Theory and Education: An Introduction. *Mind, Culture, and Activity* 11, 1 (2004), 1–8. https://doi.org/10.1207/s15327884mca1101_1
[136] P.J. Runkel and J.E. McGrath. 1972. *Research on Human Behavior: A Systematic Guide to Method*. Holt, Rinehart and Winston, New York, NY, USA. https://books.google.dk/books?id=3FdqAAAAMAAJ
[137] Donald A. Schön. 1993. Generative metaphor: A perspective on problem-setting in social policy. In *Metaphor and Thought* (2 ed.), AndrewEditor Ortony (Ed.). Cambridge University Press, Cambridge, United Kingdom, 137–163. https://doi.org/10.1017/CBO9781139173865.011
[138] Ben Shneiderman. 1983. Direct Manipulation: A Step Beyond Programming Languages. *Computer* 16, 8 (Aug. 1983), 57–69. https://doi.org/10.1109/MC.1983.1654471

[139] Ben Shneiderman. 1997. *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (3rd ed.). Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[140] John Shotter. 1992. Bakhtin and Billig: Monological Versus Dialogical Practices. *American Behavioral Scientist* 36, 1 (1992), 8–21. https://doi.org/10.1177/0002764292036001003 arXiv:https://doi.org/10.1177/0002764292036001003

[141] Herbert A. Simon. 1996. *The Sciences of the Artificial (3rd Ed.).* MIT Press, Cambridge, MA, USA.

[142] Susan Leigh Star. 1996. Working together: Symbolic interactionism, activity theory, and information systems. In *Cognition and Communication at Work*, Yrjo Engeström and David Middleton (Eds.). Cambridge University Press, Cambridge, United Kingdom, 296–318. https://doi.org/10.1017/CBO9781139174077.013

[143] Susan Leigh Star and James R. Griesemer. 1989. Institutional Ecology, 'Translations' and Boundary Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39. *Social Studies of Science* 19, 3 (1989), 387–420. https://doi.org/10.1177/030631289019003001 arXiv:https://doi.org/10.1177/030631289019003001

[144] Lucy Suchman. 2006. *Human-Machine Reconfigurations: Plans and Situated Actions* (2 ed.). Cambridge University Press, Cambridge, UK. https://doi.org/10.1017/CBO9780511808418

[145] Lucy A. Suchman. 1987. *Plans and Situated Actions: The Problem of Human-machine Communication.* Cambridge University Press, New York, NY, USA.

[146] Pål Sørgaard. 1989. A Framework for Computer Supported Cooperative Work. *DAIMI Report Series* 17, 253 (March 1989), 1–23. https://doi.org/10.7146/dpb.v17i253.7609

[147] Larry Tesler. 2012. A Personal History of Modeless Text Editing and Cut/Copy-Paste. *Interactions* 19, 4 (July 2012), 70–75. https://doi.org/10.1145/2212877.2212896

[148] Theophanis Tsandilas, Anastasia Bezerianos, and Thibaut Jacob. 2015. SketchSliders: Sketching Widgets for Visual Exploration on Wall Displays. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. ACM, New York, NY, USA, 3255–3264. https://doi.org/10.1145/2702123.2702129

[149] Theophanis Tsandilas and Wendy E. Mackay. 2010. Knotty Gestures: Subtle Traces to Support Interactive Use of Paper. In *Proceedings of the International Conference on Advanced Visual Interfaces* (Roma, Italy) *(AVI '10)*. ACM, New York, NY, USA, 147–154. https://doi.org/10.1145/1842993.1843020

[150] Lev Vygotsky. 1962. *Thought and Language.* MIT Press, Cambridge, MA.

[151] Andrew M. Webb, Hannah Fowler, Andruid Kerne, Galen Newman, Jun-Hyun Kim, and Wendy E. Mackay. 2019. Interstices: Sustained Spatial Relationships between Hands and Surfaces Reveal Anticipated Action. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. ACM, New York, NY, USA, Article 588, 12 pages. https://doi.org/10.1145/3290605.3300818

[152] Joseph Weizenbaum. 1976. *Computer power and human reason: From judgment to calculation.* W. H. Freeman & Co, Oxford, England.

[153] Pierre Wellner. 1993. Interacting with Paper on the DigitalDesk. *Commun. ACM* 36, 7 (July 1993), 87–96. https://doi.org/10.1145/159544.159630

[154] Terry Winograd and Fernando Flores. 1986. *Understanding Computers and Cognition: A New Foundation for Design.* Ablex, Norwood, NJ.

[155] Haijun Xia, Bruno Araujo, Tovi Grossman, and Daniel Wigdor. 2016. Object-Oriented Drawing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 4610–4621. https://doi.org/10.1145/2858036.2858075

[156] Shumin Zhai and Per-Ola Kristensson. 2003. Shorthand Writing on Stylus Keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Ft. Lauderdale, Florida, USA) *(CHI '03)*. ACM, New York, NY, USA, 97–104. https://doi.org/10.1145/642611.642630

[157] Shumin Zhai and Per Ola Kristensson. 2012. The Word-Gesture Keyboard: Reimagining Keyboard Interaction. *Commun. ACM* 55, 9 (Sept. 2012), 91–101. https://doi.org/10.1145/2330667.2330689

[158] John Zimmerman, Jodi Forlizzi, and Shelley Evenson. 2007. Research Through Design As a Method for Interaction Design Research in HCI. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '07)*. ACM, New York, NY, USA, 493–502. https://doi.org/10.1145/1240624.1240704

Received July 2020; revised April 2021; accepted May 2021