16

Introduction to interaction

By now, it should be clear that the term *interaction* covers a lot of ground in human–computer interaction (HCI). We have used it to discuss various applications, from a user typing on a smartphone to a team of information workers communicating via email. It has been used to describe individuals, groups, and communities using computers. Moreover, interaction takes place at different timescales. Pressing a button takes about a hundred milliseconds; adopting an information system in a large organization can easily take months. Interaction also occurs in different contexts, including work, leisure, and in-between contexts such as commuting. If "interaction" covers all of this, of what use is it as a construct for HCI? What *is* interaction?

To answer this fundamental question, one might turn to dictionary definitions of the term. However, as pointed out over three decades ago by Winograd [898], that may not be helpful. He noted that "Webster defines 'interaction' as 'mutual or reciprocal action or influence.' Clearly, humans act on computers and computers influence humans. But how? In what dimensions?" [pp. 444–445]. We need a more nuanced understanding of interaction, one that sheds light on phenomena occurring across timescales, units of analysis, and contexts.

Since Winograd's question, researchers have begun to articulate more specific accounts of interaction [357, 383]. This part reviews the well-known views listed in Table 16.1. They provide a *general* conceptual basis for understanding interaction. Specifically, they expound *mutual determinacy* in interaction. This focus distinguishes these theories from theories of people as users of computers, which we covered in Part II. Theories of interaction are also different from the concept of user interface. In the next part of this book (Part V), we will look at user interfaces—the *technology* through which interaction with computers takes place. That part will discuss the technology necessary for input and output in interactive systems. User interfaces are technologies that mediate interaction and are inherently part of it; they are not the same as interaction. Interaction is a dynamic phenomenon that unfolds over time as users and computers influence each other.

16.1 What is a theory of interaction?

What makes a theory *a theory of interaction*? Or more specifically, what theories of interaction could be relevant to the situation in Figure 16.1? To answer that, we first need to understand what a theory is. We then need to scope "interaction" as the subject of theory. Finally, we need to discuss what makes a good theory in HCI.

In general, a theory consists of a set of *propositions*, or statements. A proposition is a claim about the world. In the case of particle physics, the propositions that make up a theory may concern the nature and behavior of particles. HCI theories contain statements that link humans and technology and possibly some outcomes (e.g., poor usability, high user experience). Propositions characterize entities and link them to other entities, some of which are conceptual. For example, they can talk about information, difficulty, working memory, and so on.

Introduction to Human-Computer Interaction. Kasper Hornbæk, Per Ola Kristensson, and Antti Oulasvirta,
Oxford University Press. © Kasper Hornbæk, Per Ola Kristensson, and Antti Oulasvirta (2025).
DOI: 10.1093/oso/9780192864543.003.0016

Figure 16.1 Theories of interaction illuminate interactive phenomena beyond intuition and help solve design and engineering problems in HCI. Even seemingly mundane activities such as entering text have rich structures that can be uncovered by these theories. Photo by Jonas Leupe.

In HCI, propositions come in different forms. Consider the following propositions of Fitts' law, which we discussed in Chapter 4:

1. "Human motor system is a limited capacity information channel."
2. "Attempts at reaching a target are limited by the speed and variance (accuracy) of the movement involved."
3. "If the user tries to increase speed, accuracy will be compromised, and vice versa: An increase in accuracy reduces speed."
4. "The difficulty of selecting a target is proportional to its distance and inversely proportional to its width (index of difficulty)."
5. "Average movement time can be predicted as linear regression to the index of difficulty."

Here, a progression can be observed. The theory starts with a high-level statement: The motor system is a limited-capacity channel. From there, it derives propositions about the relationship between performance and interface design through the construct of a target.

In HCI, propositions must generally link aspects of people with aspects of technology or design. In other words, they must be about mutual determinacy.

16.1.1 Mutual determinacy

The philosopher of science Bunge defined *mutual determination* as a special type of causal relationship applied in scientific explanation [110]. You may normally think of causality in terms of cause and effect. Consider pressing a button with your finger: When you press the button cap (cause), it triggers a command (effect). However, such a description does not count as a theory of interaction because it does not link the effect *back* to the cause. It is silent on how the design of the button affects its pressing. The point of mutual determination is exactly this: What happens in interaction is mutually determined by the human and the computer. In other words, what happens in interaction cannot be attributed solely to the human or the computer—the two must be considered together.

The theories we discuss in this part all commit to this idea. For example, interaction-as-tool-use focuses on this idea (Chapter 19). Tools change people and their activities; in turn, this changes the tools. This has profound effects on how we think about computer-based tools.

According to Bunge, causal determination is only one of the forms of causal relationships used in scientific theories. Bunge analyzed theories in scientific fields, from biology to physics, and developed a rich typology of mutual determination. In *mechanical determination*, an antecedent determines a consequent. The pressing of a button would count as a mechanistic determination. Here, neural events in the brain determine physiological events leading to the contraction of relevant muscles, leading to physical contact with the button that then triggers an event in the computer. In *statistical determination*, there is a stochastic relationship between the two entities. For example, statistical models, such as Fitts' law, describe a relationship that is considered a statistical determination. Here, the statistical model (regression) links the time it takes for a user to elicit a response to the design of the task environment (distance and width of buttons). However, this relationship is statistical and not definite. In *structural determination*, the end results of the interaction are jointly determined by multiple causes that make up the whole. For example, in interaction-as-rationality, courses of action emerge as a *joint* function of the user's goals and capabilities and the properties of the environment. Finally, in *quantitative determination*, interaction is described as a continuous unfolding of states. Each state dynamically leads to a new state depending on the forces involved. For example, in control-theoretical analysis, we see the interaction as dynamically changing states.

One type of determination is shared by all theories of interaction in HCI: *teleological determination*. In teleological determination, goals or purposes determine interaction in some way. In Greek, *telos* means goal or purpose. Even in interaction-as-transmission, the user has an *intended* message in mind. All theories of interaction discussed in this part assume *intentionality*: People have goals that have a (mutually) causal role in interaction. Intentionality is a litmus test for HCI theories. However, computers can also have goals or at least algorithmic objectives. This adds another level to teleological determination. An interactive AI system can possess an objective function it pursues when acting, for example, when correcting a character it flags as a typo. To conclude, Bunge's typology is useful in understanding the landscape of HCI theories. We discuss these types in the rest of this part.

16.1.2 **Example: Six views of text entry**

To provide an overview of theories of interaction, we view the task of text entry through the six theories discussed in this part.

First, text entry can be seen as the transmission of typed information to a computer. This concept, communication of information, is rooted in information theory (Chapter 17). Information theory provides a rigorous formalism to understand and quantify interaction via the concept of passing messages through a noisy channel. In text entry, the theory provides us with concepts and metrics that aid our understanding of typing performance. For example, the information rate of a text entry method can be quantified using the concept of throughput from information theory. As another example, text entry often relies on language models, and such modeling is based on information theory.

Table 16.1 The theories of interaction reviewed in this part can be delineated by their focus, time span, and scope.

Theory	Interaction phenomenon	Time span	Scope
Information and control	Interaction is viewed as the user and the computer sending messages across a noisy channel to communicate and control each other.	Seconds	Individual
Dialogue	Interaction is structured as communication turns between the computer and the human, with each turn changing the context of the subsequent turn.	Seconds to minutes	Individual
Tool use	Interaction involves users using computers as tools to pursue their goals in an effective, efficient, and satisfying way.	Minutes to days	Individual in context
Automation	Interaction involves users striving to achieve their goals with the assistance of an AI system that can autonomously carry out aspects of tasks, such as acquiring information, analyzing information, deciding an action, or carrying out an action.	Minutes to days	Two or more autonomous or semi-autonomous actors
Rationality	Interaction is about users choosing actions with a computer that they believe will maximize utility for them. Their success is limited by the environment (computer, context) and their abilities.	Seconds to weeks	Individual
Practice	Interactions are situated in and affected by organizational and societal contexts.	Weeks to years	Teams and organizations

Another classical view of interaction, originating in cybernetics, is to view it as a control problem (Chapter 17). Interaction is understood as actions and reactions that minimize the discrepancy between the present state and the goal state. Viewing text entry as a control problem can provide a more in-depth understanding of the human perceptual and motor control aspects of text entry, such as a user's ability to touch individual letter keys on a touchscreen keyboard or a user's ability to perform a touchscreen gesture. Moreover, control theory offers a broad view of interaction; it is not limited to input. Many things we do with computers, such as listening to music or posting messages, are about control. We want to have certain effects on the world; that is, we want to control it. For example, we may choose a song to play to regulate our emotions or influence co-present others.

Text entry can also be seen as dialogue (Chapter 18). The essence of dialogue is that communication is organized in sequences of turns. When you enter a letter (a turn), a small pop-up appears for you to confirm your press (a turn). A word prediction list may show possible completions of the word. At every press, these are updated, over time forming a graphical dialogue between the user and the system. Text entry can also be carried out via spoken dialogue. Conversational agents (e.g., ChatGPT, Siri, Alexa, Cortana) allow users to enter text messages via commands and feedback.

Text entry is also a good example of tool use (Chapter 19). A text entry method is a tool that allows the user to communicate with someone or something, typically other people or a service, using asynchronous text messages and longer documents. Thinking about text entry as tool use allows us to assess text entry in terms of its utility—how well it supports users in that pursuit—and its usability—how easy it is to enter text and learn how to enter text. Tool use also triggers the concept of accessibility. Persons with varying capabilities and backgrounds need to effectively enter text, or these systems fail as tools. For example, text entry methods such as eye typing are designed to allow nonspeaking users with motor disabilities to enter text using their eye movements only.

Text entry can also be seen as a task where different subtasks are shared between the human and the computer (Chapter 20). One example is autocorrect, which automatically corrects typing errors while the user is typing. Another example is the use of word predictions, which allow the user to select a word from a set of word suggestions instead of typing out the word in full. Such techniques provide different levels of automation. The effect of such automation critically depends on how well its design appreciates the actor's unique capabilities. For example, word prediction usage may not be beneficial if the word suggestions are poorly suited to the individual's language or if the user can type words faster than the time required to scan and select among word suggestions.

In this part, we will also look at interaction as a rational pursuit. The basic assumption underlying theories of rationality is that people choose how to act according to their benefit; in other words, they are rational. However, the design of the artifact and the user's cognitive limitations both *bound* (limit) what can be achieved in practice. In text entry, users do their best to hit the right buttons; however, because of the limits of the perceptual and motor systems, they may need to slow down to decrease the error rate. They also need to be rational in how they share their visual attention. On the one hand, visual attention is needed to check whether the typed text is right; on the other hand, it is needed to guide the fingers to the right keys. The users change their behavior to achieve optimal bounded behavior. In Chapter 21, we cover different ways to predict user behavior based on their environments and abilities.

Finally, we look at factors that affect interaction beyond the interface. A text entry method is affected by practice (Chapter 22). The practice of someone typing is shaped by norms and social factors, for example, the expected division of labor between communication partners—who is supposed to say or do what in a conversation. As another example, the use of emojis is affected by social norms that have evolved since the advent of mobile phones with text entry capability. Factors that define a practice evolve over time and may not be immediately observable; however, they heavily shape what happens and why.

16.2 **Phenomena explained by HCI theories**

Let us now look beyond text entry at the bigger picture of HCI theories. We look at their differing scopes and the key phenomena they explain in interaction.

16.2.1 **Commanding**

One goal of interaction is *communicating intentions*, for example, issuing commands to a computer, selecting graphical elements in image processing software, or entering text.

Chapter 17 presents two views of such interaction. The first view is that interaction can be seen as a communication system. This communication system is used to model the interaction as a source (e.g., the user) sending messages over a noisy channel to a receiver (e.g., a computer). Importantly, these messages can be quantified, typically by the number of bits required to encode them. This makes it possible to reason about the efficiency of interaction by defining and quantifying messages. As a consequence, this view has been used to reason about achievable limits in text entry, the time it takes a user to consider several choices, and the movement time required for a user to select a target.

The second view presented in Chapter 17 is that of interaction as control, which models interaction as a closed-loop control system. The information-centered model views the interaction as discrete bits that are transmitted over a noisy channel. By contrast, the control theory model views interaction as a user moving their body continuously in space and time to articulate their intention in such a way that it has the desired effect on the computer. A central notion in viewing interaction as control is *feedback*: The user regulates action as a response to information obtained via their senses, such as touch and vision. The control view of interaction allows for reasoning about mechanisms that underpin interaction. An example of this in Chapter 17 is the control view of modeling the movement time required for a user to select a target. Unlike the information-centered model, which merely predicts the movement time for this task, the control model can explain how actions (feedforward) and feedback together produce pointing movements moment by moment.

16.2.2 **Communicating**

Interaction can also be viewed as a form of communication. While Chapter 17 is concerned with interaction as sending messages or responding to feedback, Chapter 18 introduces dialogues as a way to understand interactions that go beyond a single communication act. It does so by discussing four different perspectives of a dialogue as a model of interaction. The computational perspective models dialogues as finite-state machines. This formalism allows designers to reason about emergent properties of dialogues, such as how many steps the user must take to recover from a mistake. The perspective of dialogue as a goal-directed action views interaction as a user driving a system into an intended state using a turn-based dialogue. Central to this perspective of interaction is the assumption that the main challenge for the user is understanding the system to such a degree that the user can take appropriate actions.

This perspective of interaction as goal-directed dialogue has been challenged by another perspective: dialogue as embodied action. The idea is that goal-directed dialogue reduces the users to agents who passively react to a set of options generated by the system. Dialogue as embodied action rectifies this by introducing interactive elements into goal-directed dialogue. For example, a user learns about various options by exploring the user interface. The final perspective views interaction as communication. This means accepting that human–computer dialogues are similar, yet distinct, from human–human dialogues. Similar to how we use a different language in formal settings compared to when we interact with friends, users use a different language when interacting with computers. Understanding the context of dialogue allows for the improved management of people's expectations and their preferred style of interaction.

16.2.3 **Tool use**

A fundamental goal of interaction is to provide users with tools that allow them to achieve goals that they would otherwise not have been able to achieve. Chapter 19 discusses the general topic of tool use and elucidates how users use the computer as a tool to accomplish a variety of tasks. This leads to a discussion around the utility, usability, fit, and accessibility of tools. Utility captures how well a tool supports users in achieving their goals. Usability refers to the ease with which a tool is operated. Accessibility is an extension of usability to ensure that as many people as possible find a tool easy to operate, regardless of individual capabilities. Understanding tool use from these aspects allows designers to ensure their user interfaces are effective, safe, and easy to learn. "Fit" refers to the match between a tool's capabilities and the user's abilities and task-related requirements.

Chapter 21 discusses the use of computers as rational acts, that is, as actions that users take for their benefit. Theories of rationality, from classical decision-making to computational rationality, offer a rich toolbox for understanding *why* users act the way they do. Underlying all of these theories are two core assumptions. The first is that a user's actions are choices from sets of options with different pros and cons for the user; the second is that actions are selected to maximize the expected utility for the user. Theories differ in terms of what kinds of limitations, or bounds, limit the user's ability to pick the best possible action.

16.2.4 **Augmenting capabilities**

Tools can also provide a degree of automation to augment human capabilities. Chapter 20 introduces a framework for understanding four types of automation: acquisition, analysis, decision, and action. Acquisition automation refers to the automated registration and collection of input data. Analysis automation involves performing some level of inference based on the input data, for example, extrapolating a trajectory. Decision automation refers to various degrees of automated support for deciding among several alternatives, for example, a system that automatically recommends a suitable route from point A to point B. Finally, action automation means that the system is, to some degree, automatically carrying out an action in response to a decision. For example, a photocopier can automatically sort and collate copies. These types of automation can range from fully automated, meaning the system acts autonomously, to no automation, meaning the user has to make decisions or choose actions from a set of alternatives. Designers decide the type and level of automation appropriate for a system based on a set of primary and secondary evaluation criteria.

16.2.5 **Situated use**

Chapter 22 discusses interaction as it happens in real life to advance users' goals, be it career goals, life goals, or something else. Importantly, interaction occurs in social contexts and physical spaces; these and other factors shape interaction in many ways. Investigating practice means understanding how people handle variations, challenges, breakdowns, and other events as part of their activities, which are governed by rules, plans, visions, and other norms. This implies that it is insufficient to understand interaction as the mediation of human activities by merely examining social structures, such as work hierarchies and delegation chains, or individual behaviors. It is necessary to look at the practice as a whole. Therefore, when we view interaction as practice, we accept that when users interact with systems, they interpret information, modify elements of systems, and work around limitations to reach their goals. In addition, viewing interaction as practice means realizing that interaction with systems is embedded in wider organizational and social settings and may involve many interactive systems that affect each other. Finally, viewing interaction as practice means studying how interaction affects people over time in different settings, such as work or leisure.

16.3 What makes a good HCI theory?

HCI theories construe interaction by reference to different mechanisms. They refer to events occurring at different timescales, at different levels of granularity, or that are mediated by different types of causal relationships. Knowing about these mechanisms is important because they help us understand and improve interaction in ways that go beyond intuition.

16.3.1 Theories should be informative

Any theory should be informative to be of relevance; in HCI, a theory should tell us something about interaction that is not obvious. The *explanatory power* of a theory refers to the empirical accuracy and coverage of the explanations offered by the theory. The more accurate and broader the coverage, the higher the explanatory power. For the sake of argument, one can compare Fitts' theory of aimed movements with Norman's theory of interaction, both of which we discuss in this part. Fitts' theory has high statistical accuracy but low coverage; Norman's theory has high coverage but low accuracy. So, how is such explanatory power achieved?

HCI theories gain their explanatory power by shedding light on *latent factors* in interaction. A latent factor is something that affects observations about interaction without being directly observable. Latent factors are the inner mechanisms of theories; they relate otherwise unrelated observations together. Famous latent factors in HCI theories include:

1. **Interaction-as-dialogue:** Norman's gulf of evaluation, or the cognitive appraisal that occurs when a computer changes its state. "Is this taking me closer to my goal?"
2. **Interaction-as-information:** The limited information capacity channel of the human motor system, which imposes a trade-off between the speed and accuracy of aimed movements in interaction.
3. **Interaction-as-control:** A controller is a latent process in the user's mind that tries to pick motor actions that minimize the distance between the present and goal states.
4. **Interaction-as-practice:** Contextual and normative factors that affect the user's choice of action.
5. **Interaction-as-bounded-rationality:** Utility or a reward that a user is trying to obtain via interaction.

Every HCI theory aims to explain how some surprising or important phenomenon *emerges* as interaction unfolds. Emergence refers to how a theory suggests that interesting high-level properties arise in interaction. In physics, the properties we attribute to water—such as its felt wetness and its properties as a fluid—emerge from low-level physical events. Many key HCI concepts emerge in interaction: usability, user experience, accessibility, fit, and so on. These constructs cannot be attributed to the interface or the human; they emerge through their interaction. Interesting theories should tell us something about their emergence. Interaction is thus more than simple cause and effect.

16.3.2 Theories should make predictions

Theories can also make predictions. In a *prediction*, a future state is estimated from a given starting condition. In HCI, we are particularly interested in predictions concerning users: their performance, errors, experiences, and so on. We need theories that link such predictions to factors we can measure or affect, such as characteristics of the task, the user interface, or the users.

Obviously, predictions are not "free." A theory needs to relate to the situation at hand and be instantiated in some way. In practice, this is often achieved via a *model*. A model is a formally expressed set of propositions that follows some axiomatic system, such as algebra, logic, or a programming language. Models contain "hard" and "soft" parts. The hard parts are mechanisms (theoretical postulates) that link observations, often via latent factors. Hard parts stay the same from one instance of a model to another; in other words, such mechanisms are invariant. The soft parts are parameters that can be changed to fit the situation at hand. Changing a parameter changes how the mechanism works but not the mechanism itself.

For example, earlier in this chapter, we noted the connection between Fitts' theory of a limited-capacity motor system and Fitts' law. Fitts' law is a model consisting of a regression equation (Chapter 4). Its hard part is the first-order regression model and the algebraic structure of the second term ($ID = \log_2(2D/W)$). Its soft part is the two coefficients a and b.

Common modeling approaches in HCI include:

- Regression models
- Probability theory
- Logic and rule-based systems
- Deep learning and other machine learning models
- Simulations.

A model stands between a theory (including its propositions) and the world:

Theory–model–data–world.

There are two possible directions in modeling: forward (from model to data) and inverse (from data to model). In forward modeling, we produce data from a model. For example, we input the starting conditions that describe a choice reaction task (Chapter 4): How many options does the user have to choose from? The model, in this case, the Hick–Hyman model, predicts the choice reaction time. In inverse modeling, we do the opposite: We fit the parameters of a model to match the data. A model's *predictive power* refers to the accuracy and precision with which it predicts events that are not in the data used to determine its parameters.

16.3.3 **Theories should aid evaluation**

The theories, methods, models, and even the views of interaction covered in this part of the book can be used in various ways to assist in designing and evaluating interaction.

Chapter 17 shows that it is possible to quantify aspects of interaction such as the rate of entering text, the time it takes a user to select a target, or the time it takes a user to select among multiple choices. When an interaction can be quantified, it is possible to optimize it, for example, by investigating which parameters and design choices reduce the average response time. In addition, it is possible to evaluate the interaction by observing how its metrics, such as the information rate, change as the design is modified.

To support users in exploring their options, interaction can be viewed as a dialogue (Chapter 18). Dialogue views interaction as a series of communication turns that require the user to infer what the other agent intends to communicate or how one's communication acts might influence the other agent.

Viewing interaction as the use of tools (Chapter 19) allows designers to understand the utility, usability, and accessibility of interaction. This can help designers evaluate interaction and support the design of interaction, for example, by helping designers identify and mitigate accessibility issues.

The automation framework covered in Chapter 20 allows designers to analyze each function in a design to identify suitable types and levels of automation. It can both inform design and serve as an evaluation tool.

Models of rationality (Chapter 21) can serve as tools for determining how to distribute and shape information in a user interface to allow users to maximize utility. This can be achieved, for example, by ensuring information is retrievable and presented in such a way that the cost–reward structure for the user is optimal. These models assume that users are as rational as they can be in light of the given constraints, such as the time available to make a decision or the resources available, including cognitive resources. Operating under such constraints, the user may be unable to reach the optimal solution. The term *satisficing* captures the idea that a user may settle for the first option that is deemed satisfactory given some constraints. Chapter 21 presents alternative ways to analyze interaction from this viewpoint. Such analysis can evaluate whether the cost–reward structure of the user interface allows the user to implement the optimal strategy.

Finally, Chapter 22 discusses how users personalize, tailor, and appropriate interaction as part of interacting with computer systems that are situated in larger contexts—communities, organizations, and societies. Such understanding can help designers understand design issues and requirements and assist in studying how interaction is interrelated with the use of interaction in practice. For instance, people may have many objectives in mind when trying to achieve a task; if the design of interaction does not consider this, this may lead to unexpected side effects.

16.3.4 **Theories should guide measurements**

Theories can help us decide what to measure. Information theory, for example, places emphasis on speed and accuracy in the entry of words. It offers precise measurements such as throughput and words per minute. Theories of practice, on the other hand, call for observations on the social processes that surround the sending and receiving of a message. Theories of tool use add to these by allowing us to measure the utility and acceptability of technology.

16.3.5 **Theories should inform design**

Finally, HCI theories should help us make practical decisions in design. After all, HCI as a field is concerned with design and engineering. To achieve this, theories need to offer some way of *reasoning* that links propositions to practical questions.

The logical basis of theory use in design has been traditionally attributed to *abduction*, which goes from observations to theory. For example, "Why did Aaron give up in the tenth level of the game?" Among others, this observation can be explained by reference to a motivational mechanism. Such an explanation can then be reasoned with to draw implications for the redesign of the game level.

Another form of reasoning is *deduction*. Here, we go from a theory to the world. For example, control theory suggests that feedback provided to the user should be informative; in particular, it should help the user correct their course of action. This implies that the design of feedback should aim to maximize informativeness for such corrections.

A third form of reasoning is *counterfactual reasoning*. This type of reasoning goes from a theory to a possible—but not yet existing—world. Counterfactual claims are "what might" or "what ought" claims. A counterfactual proposition can be expressed as follows:

If design were *a*, then interaction would be *c*.

The argument has two parts, the antecedent *a* and the consequence *c*, linked by the counterfactual step. This step should be given by the theory.

For example, human performance models, cognitive models, and many other models follow a general form that maps task conditions or design (antecedents *a*) to some effect on users (consequents *c*). This counterfactual effect is marked using the symbol >:

a (task conditions) > c (user behavior, performance, or experience).

In another variant, a feature of the design is what affects the consequent:

a (feature of design) > c (user behavior, performance, or experience).

Theories can also inform design by simply illuminating factors that are in play in a situation. For example, the theories of practice are not expressed as models; yet, they expose critical factors that one might dismiss when thinking about the design of a user interface. A theory can also inspire and help one consider new perspectives. For example, Palen and Dourish [635] proposed a theory of privacy which countered the then-prevailing view of privacy as something determined by system design. They proposed that privacy is something we control and coordinate with other people interactively. This view inspired a new take on how to support privacy via interface mechanisms.

Summary

- Interaction is a core notion in HCI and refers to the mutual influence between people and computers.
- Phenomena in interaction are emergent, that is, they are not attributable to the user or to the computer alone.
- Theories of interaction explain the emergence of interactive behavior by referring to factors related to the user, the computer, and the environment. Some factors may be latent, that is, not directly observable.
- Theories of interaction have different theoretical commitments and they encompass different timescales and purposes.
- Theories of interaction help design and evaluate interactive systems.