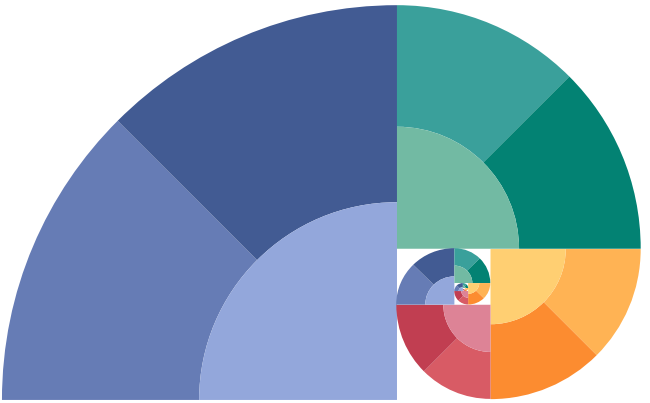
WENDY E. MACKAY

DO IT: THE DESIGN OF INTERACTIVE THINGS

CHI 2023 Preview



EX)SITU

DO IT: The Design of Interactive Things

Wendy E. Mackay

CHI 2023 PREVIEW
EX)SITU

Published with the support of Inria – https://inria.fr.

This work was supported by European Research Council (ERC) Advanced Grant No 321135 "CREATIV: Creating Co-Adaptive Human-Computer Partnerships" (Principal Investigator Wendy Mackay).

Table of contents

How to design, test and improve interactive systems?

DO IT: The Design of Interactive Things provides a practical, hands-on guide for designing interactive technology. Based on a coherent set of design principles, the book offers rapid and effective techniques from multiple disciplines that students, system designers and researchers have used to create successful startups and publish award-winning research.

This book includes a subset of the material in the ***Do It*** book, intended for students in the CHI'23 Course on the *Design of Interactive Things*.

To our beta readers...

This booklet is a Special Preview of the forthcoming *DO IT: The Design of Interactive Things* book. This preview was created for Wendy E. Mackay's CHI 2023 course *C17: The Design of Interactive Things: Selected methods for quickly and effectively designing interactive systems from the user's perspective*. If you find a typo, or want to provide feedback, please feel free to contact the author at this address: mackay@lisn.fr. Thank you.

Introduction

How to design interactive systems from the user's perspective

"A good design is better than you think."
— Rex Heftman

Interaction designers can choose from a wealth of existing design methods, but it is often difficult to decide which to use when. This book presents the ***DoIt*** process, which integrates a variety of interaction design methods into a coherent whole. Created for participants at the *CHI'23 Conference* in Hamburg, Germany, this book contains four methods from the forthcoming book entitled: *DOIT: The Design of Interactive Things* by Wendy E. Mackay (2024).

Each method was developed and refined by the author (Mackay 2002, 2019) to highlight the *interaction* aspect of "*Human-Computer Interaction*". All are grounded in social science research and were selected for their relevance to both User Experience (UX) designers and Human-Computer Interaction (HCI) researchers. Each has been successfully employed in both corporate and academic settings by novice and professional interaction designers.

If you are a novice designer, you will be able learn and apply these methods immediately. If you are a more advanced designer, you will be able to develop custom variations that reflect the underlying principles and adapt them to the needs of your current project.

When combined, the ***DoIt*** methods offer an effective, iterative process for creating interactive systems from the user's perspective. The full ***DoIt*** book will place each method within

the larger design context, with a description of its origins, when and why it is useful, how it relates to other methods, followed by detailed instructions and examples that show how to perform the method well.

The book will also explain how to modify, extend and create new methods, how to actively involve users in *participatory design* throughout the project, and how to incorporate methods dedicated to *generative redesign*.

The four methods chosen for this book preview — *story interviews*, *video brainstorming*, *video prototyping* and *generative walkthroughs* — emphasize the importance capturing and representing users' lived experiences in the form of stories that inspire innovative ideas and accommodate potential breakdowns. Each takes advantage of video to capture those stories, generate ideas, explore designs and evaluate and redesign the resulting system.

The Design of Interactive Things

The field of Human-Computer Interaction draws from the natural and social sciences for methods that help us understand humans; and from computer science and engineering for methods that help us design and implement technology. However, interaction designers must also consider something more subtle and difficult to pin down: **interaction**.

Just as musicians need to learn to "hear" music before they can play compositions; and artists need to "see" faces before they can paint portraits; interaction designers need to "experience" interaction from the user's perspective before they can create successful interactive systems.

One of your key roles as an interaction designer is to take the user's perspective and act as their primary advocate. Your designs should not only enable users to perform specified tasks, but also help them make sense of and enjoy using the system in real-world situations, and accommodate their wants and needs as they evolve over time.

The ***Do It*** design process involves a set of interconnected methods that capture the *interaction* between human users and the technology. The ***Do It* wheel** (Figure 1) is composed of four

quadrants that explore "*Who is the user?*", "*What is possible?*", "*What should it be?*", and "*Does it work?*". Each is subdivided into methods for collecting, representing interpreting information relevant to that quadrant.

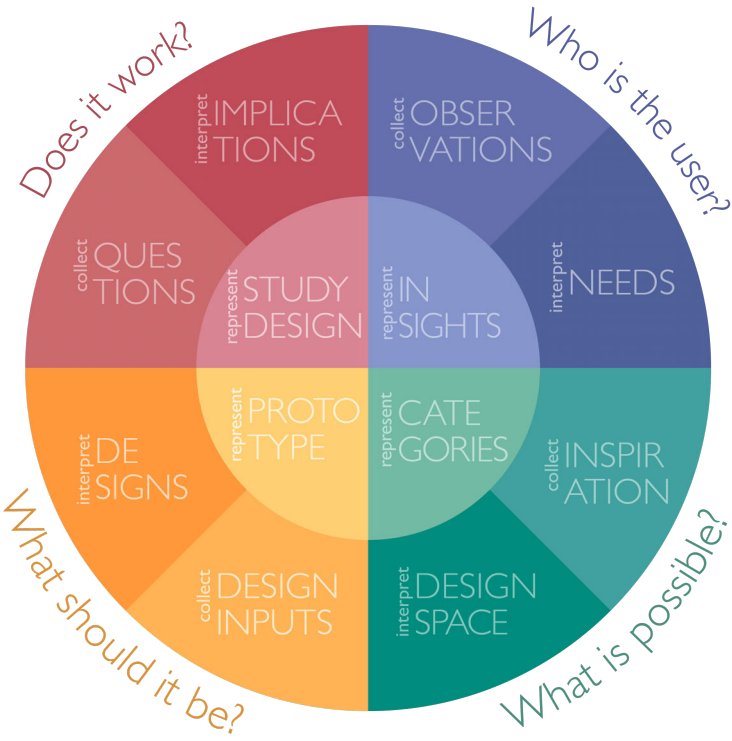


Figure 1.
The ***Do It* Wheel** includes finding out about users, inspiring new ideas, prototyping interactive designs and evaluating systems.

Unlike the British Design Council's oft-cited "*Double Diamond*" (2005) timeline, the ***DoIt*** process is shown as a circle. While the former rightly highlights the key observation that design alternates between expanding the *design space* and making specific choices, it de-emphasizes the iterative and interconnected nature of these design methods. By contrast, the ***DoIt*** wheel can be traversed in any order, so that design artifacts produced in one quadrant affect design activities in another.

One of your key challenges as a designer is to choose the "right" design method at the right time. Although you may perform the same methods multiple times, the goals and details

will vary, influenced by the results of previous design activities and your current project needs. Some methods are most appropriate in early design phases, others will only make sense in the middle and late stages of the project.

Experienced designers know that no "perfect" design method exists: a method suitable at one point in the design lifecycle may be irrelevant before or after. The purpose of this book is not therefore to define an ideal set of methods, especially since HCI researchers and UX designers will continue to generate new ones. Instead, the goal is to provide a solid foundation for understanding the interaction design process and the inherent trade-offs among these approaches, so that you can choose or adapt the appropriate method for the design challenge at hand.

Book overview

This book is organized into four main chapters which address each of the four quadrants of the ***Do It* wheel**. Each summarizes the key design activities relevant to that quadrant: *collecting* new information, either gathered or generated; *representing* that information as a design artifact; and *interpreting* the resulting design artifacts.

Some methods are *divergent*: they help you expand the space of possibilities. Others are *convergent*: they help you select among design alternatives and directions. Some methods are system-oriented and focus on the characteristics of the technology being designed. Others are story-oriented and focus on how users will interact with the proposed design under circumstances that change over time. Each approach raises different design challenges but all are valuable and necessary.

Each chapter provides an overview of a particular method category — *interviews*, *brainstorming*, *prototyping* and *walkthroughs* — and either compares variations or shows how several methods can build upon each other.

Next, each chapter introduces a specific method — *story interviews*, *video brainstorming*, *video prototyping* or *generative walkthroughs* — with directions for how to prepare *before* the activity, what to do *during* the activity and what to produce *after*.

These chapters also include multiple examples that illustrate each method.

Although many design methods emphasize the features of the system being designed, here the emphasis is on story-based methods that will help you understand your design from the user's perspective. You will learn to explore *interaction* — how users interpret, interact and feel about the system under different circumstances — to define the user experience.

These methods demonstrate the benefits of "sketching with video", not to create highly produced and edited video productions, but rather as a lightweight tool for capturing and expressing interaction, with no post-session video editing required.

Advice

Interaction designers almost always work in interdisciplinary design teams that benefit from different types of expertise. However, differences in background, experience, assumptions and personality can contribute to misunderstandings and slow the process down.

The methods in this book are crafted to help minimize disputes among team members and enhance your time together. You will learn how to alternate rapid, concrete action with reflection and redesign, while minimizing excessive debate. Learning these techniques will build your confidence as a designer and help you avoid getting stuck.

The following advice will help you perform the methods quickly and well, with minimal friction among the design team.

**"Just do it!...
then make it better!"**

1. Listen to and value diverse perspectives and rotate through the various roles within and across methods. Remember that interaction design works best as a collaborative activity with small groups of mixed expertise.
2. Prepare for every design activity in advance. Ensure that you have an appropriate space to work and have sufficient materials to run the session.

3. Ensure that all group members participate on an equal basis. None of these activities require special skills and all are accessible to everyone, including (or especially!) users.
4. Avoid "*analysis paralysis*"! The methods in this book emphasize *doing* over talking. Instead of arguing about the perfect solution, start with an imperfect first guess and iterate from there.
5. If you find yourself arguing about a design issue, stop arguing and sketch or paper prototype several alternatives. You will find that improving ideas is far easier than facing a blank sheet of paper and will save time over arguing about hypothetical problems. Your motto should be: "*Just do it! … then, make it better!*"
6. Avoid post-hoc editing after a video-based design session. Instead, capture the video during the session in a form that everyone can review at the end of the session.
7. Schedule debriefing time at the end of every session and be sure to capture at least a minimal record of what you accomplished. This can be quick, such as voting for your favorite *brainstormed* idea or noting three key insights after a *story interview*. However, even methods such as a *breakdown analysis* that require an additional session will benefit from a quick review of the key insights you noted during the initial interview.
8. Schedule time for reflection later, after the session. For example, replay your *video brainstormed* ideas. You need not perform an in-depth analysis after every activity, but try find time for reflection time soon after the main activity. The longer you wait, the less you will remember.
9. Reuse your design artifacts — make every design activity count! Do not forget about the artifacts you already created. Instead, consider how to reuse and modify them to improve your design.

Human-Computer Interaction and *User Experience design* emphasize the iterative nature of design, with a phased process of creation and assessment, where designers explore different alternatives and select the best designs.

Some of the methods described here will seem familiar if you have a background in HCI or UX design, but each offers a

unique twist. All complement and build upon each other and can be varied to meet your individual needs. The next four chapters cover each of the four quadrants in the ***DOIT* wheel** shown in Figure 1, with sample methods for you to try.

Figure 2. The *Grande Vitrine de Noël* interactive exhibit.

Bibliography

1. British Design Council (2005) *Eleven lessons: managing design in eleven global brands. A study of the design process.* British Design Council. London, UK.
2. Mackay, Wendy E. (2002) *Using Video to Support Interaction Design* In *ACM/CHI 2002 Extended Abstracts*, 88 pages. Inria Multimedia Services, Roquencourt, France.
3. Mackay, Wendy E. (2019) *Designing with Sticky Notes.* In *Sticky Creativity: Post-It Note Cognition, Interaction and Digitalization* Christensen, Halskov, and Klokmose, (Eds.) Explorations in Creativity Series, Academic Press.
4. Mackay, Wendy E. (2024) *DO IT: The Design of Interactive Things.* (in preparation).

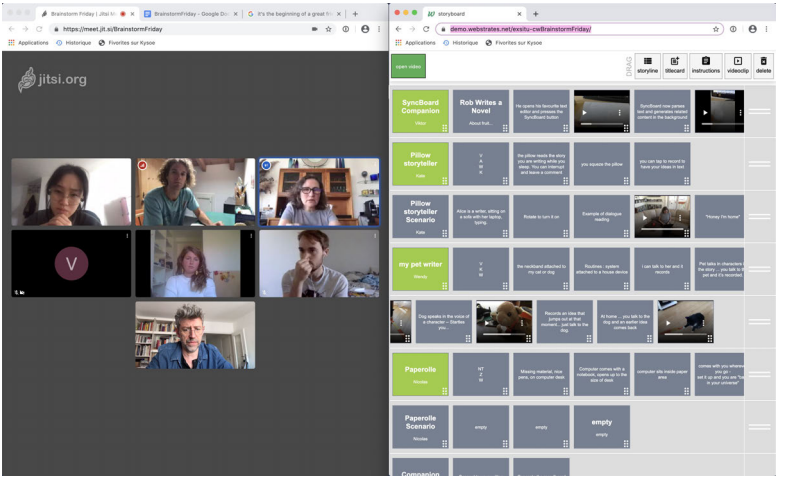


Figure 3. Remote *video brainstorming* during the pandemic worked, but it is better in person!

Discovery

Who is the user?
Learning about users and their experiences

"Finding the question is often more important than finding the answer."
— John Tukey

Introduction

Before you can design a successful interactive system, you need to consider the users of that system: Who are they? What do they need? How does the context of use change their activities?

HCI researchers and UX designers borrow from a variety of social science methods to address these questions. However, whereas academic social scientists are interested in developing general theories that explain human cognition, emotion and behavior; HCI researchers are more interested in developing principles about interactive systems and UX designers in industry must focus on meeting specific customer needs.

Most current HCI methods were transferred from another discipline to meet an HCI-specific goal, that is, to inform design. We use them to better understand how people interact with technology and produce new design directions through actionable "*implications for design*". However, this means that the rules and emphasis of many HCI methods differ from social science methods. Although we seek sufficient understanding of users to improve the effectiveness of our design concepts, we need not automatically adopt the social science goal of developing general theories abut human beings. Instead, we need HCI-specific methods dedicated to HCI goals.

This chapter summarizes methods for gathering information about users in Table 1, including how to *collect* information from and about users; *represent* it to characterize users and *interpret* it to understand situated use.

Next, the chapter contrasts the three types of *interviews* most relevant to HCI researchers and UX designers: *story interviews*, which capture realistic details of a user's current experiences in a real-world context; *tutorial interviews*, which capture how an existing system is "supposed" to work and *marketing interviews*, which elicit users' general opinions about a particular system.

This chapter concludes by explaining how to conduct a well-structured *story interview*, including a detailed description of how to prepare *before* you start; what to do *during* the activity and what to *produce* afterwards to create reusable design artifacts. Each method is illustrated with multiple examples from a sample design project.

Gathering information about users

How can we discover what users need and want? The *Discovery Methods Table* below summarizes diverse activities for understanding users. Individual methods may involve one or more of three key activities: *collecting*, *representing*, and *interpreting* ideas.

Designers have multiple options for collecting design ideas. You can gather existing information by reading academic literature or user manuals, as well as more informal sources, such as watching documentary films and even reading novels. However, successful need-finding also requires generating your own specific information about target users. For example, you can observe them directly or ask them questions in the form of live interviews or questionnaires.

Once you have collected information about users, *represent* it in a form that contributes to your design process. You can list general *user requirements* or identify the characteristics and contexts of individual *personas* (Pruit & Adlin, 2006). You can also sketch *interaction snippets* (Mackay, 2019) that illustrate the user's interaction with the current system, or draw a *journey map* (Zemke & Bell, 1985) that shows what the user does, thinks

and feels about the current system over days or weeks. More advanced methods require creating specific events that actively engage users, such as asking users how they use a *technology probe* in their homes (Hutchinson et al., 2003) or setting up an *interactive thread* (Mackay, 2004) at a conference.

Finally, you need to *interpret* your user data. You can describe users by creating a general *user profile* or develop a *current scenario* that shows the user's current experience. You can organize video footage of users into *themed videos* that illustrate particular themes, or you can edit them into a *current video* to illustrate a story about a user's experience. Finally, you can use inferential statistics to *interpret* quantitative data such as a keystroke log, or use descriptive statistics to analyze responses to a *questionnaire*. You can also analyze rich qualitative data such as interviews with methods such as Braun & Clark's (2016) *thematic analysis*.

Note that some methods focus on the *system*: How can we get an overall picture of what users need from the technology? Other methods focus on the *story*: What is the context in which the user will interact with the system and how will that evolve over time? Table 1 lists these in two columns: System-oriented methods, such as *user requirements* and *user profiles*, emphasize general user needs relative to the system. Story-oriented methods, such as *story interviews* and *current scenarios*, seek to understand the user's lived experience by capturing or generating stories and story elements situated in a specific place and time.

Understanding users	System	Action	Story
Collect	Literature review Introspection Questionnaire	READ OBSERVE ASK	Novels, films Observation Story interviews
Represent	User requirements Interaction snippets Cultural probes	LIST SKETCH ENGAGE	Persona Journey map Interactive thread
Interpret	User profile Themed video Descriptive statistics	DESCRIBE SHOOT ANALYSE	Current scenario Current video Thematic analysis

Table 1. Discovery Methods

About Interviews

One of the most informative methods for finding out about users is an *interview*. Not only do *interviews* provide richer and deeper insights than *questionnaires*, they also let you probe for more information. This is critical for interaction designers who need to capture and understand the details of both the design problem and its solution. A few well-conducted interviews can provide a wealth of design ideas and inspiration, and it is extremely difficult to create an effective design without them.

We all assume we know how to interview people. After all, we have all seen journalists' interviews and may have been interviewed ourselves. But interview techniques differ greatly according to the desired outcome: a journalist's interview should challenge the interviewee whereas a talk show host's interview should entertain. A police detective's interview should uncover new facts about a crime, whereas a sociologist's interview should contribute to a deeper, more general understanding of human behavior.

Interaction designers need something else: they need to hear grounded, real stories about users' actual experiences. *Story interviews* (Mackay, 2019) offer the most effective method for capturing an individual users' current experiences in a realistic context. These stories be transformed into a series of actions and reactions that can then be represented as *interaction snippets*. These can in turn be arranged into representative *current scenarios* that describe how users react in real-world situations. *Current scenarios* serve as the foundation for *future scenarios* which both inspire and represent new design concepts. Example 7 in Chapter 3 shows an example of a *current scenario* that was inspired by the *story interview* shown in Example 1 of this chapter.

Comparing Interview Methods

Designers can choose among three types of interview questions. A *story interview* is intended to elicit specific stories of what has actually happened; a *tutorial interview* (or "*how to interview*") explains what is supposed to happen and an *opinion*

interview reveals what the user thinks should happen or what goes wrong. The chapter focuses on *story interviews*, which always begin with story questions and only ask tutorial or opinion questions later, if at all.

Story interview

Story interviews ask the user for a step-by-step account about a recent noteworthy event or experience, with as much detail as possible. They form the most useful type of interview for interaction designers, since they capture real examples of interaction in context. You can easily identify answers from a *story interview* because they are always told in the past tense. The interview

Figure 4. *Story interviews* ask the interviewee to demonstrate what they did, step by step.

format lets the designer dig deeper and deeper to find out not only what worked well but also how the interviewee dealt with any problems that arose.

The most effective story questions take advantage of human memory by asking about notable events, with as much context as possible. Flanagan (1954) initially proposed this approach as the *critical incident technique*. He established a protocol for pilots who had survived a plane crash (the original "critical incident") and asked them to systematically remember the details of every incident and action that led up to the crash. Mackay (2002) applied this approach to HCI and then expanded the method (Mackay, 2019) to include stories about "bright spots" or significant positive events, as well as routine but extremely recent activities. *Story interviews* are also useful for investigating "critical objects" where the interviewee describes the steps they took to create a physical or digital artifact.

To gain maximum insight into the user's experience, *story interviews* should not only inquire about what went right, but more importantly, about what went wrong. They should examine each *breakdown* to discover its cause and whether or not the interviewee was able to create a successful *workaround*. It is also good practice to also look out for *user innovations* — novel solutions that serve as a source of inexpensive, field-tested ideas you can incorporate into your design.

Learning how to conduct *story interviews* is the most important method in this book. Although it takes some practice to master them, you will find that *story questions* come naturally because interviewees enjoy talking about their personal history. If you gather actual stories rather than generalities about what "should" happen, you will discover a wealth of inspirations for design.

Unfortunately, since you are surrounded by other types of interviews, you may find it hard to recognize when your *story interview* is veering into a *tutorial* or *opinion interview*. You will find specific suggestions below for how to conduct a successful *story interview* and how to rescue it if it starts to go wrong.

Once you master the *story interview* technique, you will begin to gather diverse examples of users' recent, memorable interactions with technology. Although *tutorial* and *opinion* interviews can also provide some useful information, as an interaction

designer, you will find that *story interviews* contribute the most. You designs will be much better if they are influenced by real stories of what actually happened, rather than stereotypes of typical behavior.

Tutorial interview

Tutorial interviews result in descriptions of how the system is supposed to work, when everything goes according to plan. They are especially useful when you are unfamiliar with the field or the details of the current system, and need to learn about it from an expert. However, they can also be misleading, since they make it more difficult to uncover the underlying problems that your system should solve.

Be careful! You can end up with a *tutorial interview* when your *story interview* goes wrong. You can tell if a *story interview* has turned into a *tutorial interview* if, rather than describing a

Figure 5. Tutorial inteviews work best when interviewing experts.

specific activity, the interviewee invents a description of how that activity is "supposed" to be performed. Phrases such as *"I usually do this"* or *"Every time this happens, I do that"* indicate that the interviewee has shifted from telling an actual past story and is now teaching you an idealized version of what happens now.

Tutorial interviews are most relevant when you need to understand what typical behavior looks like, usually at the beginning of a project. For example, if you are designing a health app, you can conduct a *tutorial interview* to interview health professionals. Once you understand the current approach and issues, you can conduct *story interviews* with a wider range of users to gain new insights about what to design.

Designers who rely solely on *tutorial interviews* risk designing for a stereotypical rather than actual users, which can lead to naive designs that break down under actual use. We always begin our *tutorial interviews* with an initial *story question*. This helps the interviewee remember more details and provides a better-grounded, more realistic summary of the activity.

Opinion interview

Opinion interviews are the most common but least useful interview type for interaction designers. This is counterintuitive, since we are all familiar with marketing surveys that seek users' opinions. However, although identifying positive features and "pain points" is important, *opinion interviews* offer few clues as to their underlying causes and offer far less inspiration for design.

Worse, people tend to generate opinions on the spot, without grounding them in their actual experience. Everyone can offer an opinion, what you want are opinions derived from the user's reflections on their own behavior. You will collect better, more grounded opinions if you ask for them at the end of a *story interview*, after they have remembered what they did in the past. Opinions that arise naturally during a *story interview* better reflect a user's lived experiences than those generated "off the top of the head".

The Story Interview Method

In a nutshell: Ask the user questions that elicit recent stories about specific events or objects relevant to the design brief.

Participants: pair
Level: intermediate
Resources: none
Supplies: notes, log
Access: users

As the name implies, *story interviews* are a story-based design method that captures detailed information about user experiences in order to produce actionable implications for design. When performed well, they will not only improve your general understanding of your target users' needs, but also will also provide a rich source of material from which you can generate new ideas.

The main reason that *story interviews* are so valuable stems from the fact that you can generate abstractions from detail, but you cannot generate detail from abstractions. If you let users abstract their experiences for you, you must rely on what *they* think is important to abstract. You risk hearing about poorly considered stereotypes, rather than real experiences. Although you may hear complaints about general problems, you will find it harder to probe for the reasons why these various breakdowns occurred.

Although some designers initially find *story interviews* tricky to do well, once mastered, they are easy and enjoyable for both interviewer and interviewee. People rarely have the opportunity to explain (or complain!) about the issues they face when using technology, and enjoy having a sympathetic audience who is interested in both the details of what went wrong and any clever solutions they may have found.

"You can generate abstractions from detail, but you cannot generate detail from abstractions."

Before: Prepare for a Story Interview

Choose a topic, the more specific the better. If your goal is to improve an existing system, find current users of the technology and if possible, users of competing systems. Even if you are

developing an entirely new system, you still need to understand how people manage today as they perform related activities.

Recruit members of your target audience. Some users are difficult to find, especially if your system is intended for people with advanced skills, such as doctors; people with particular challenges, such as the handicapped; or people with strong interests, such as fitness buffs. The best strategy is to find a member of the target audience who can introduce you to others. Take advantage of social networks or special interest mailing lists, or go where they gather, such as at a class or conference.

Practice your interviewing skills to get the most from each interview. *Story interviews* can provide extremely useful information if done well, but each takes time and you should avoid "wasting" interviewees.

Prepare an interview sheet with the participant ID, the date and time, setting and a short phrase summarizing the topic. Include your name, organization and project name. Next, write several *story questions* that will serve as your guide during the interview. If you are just learning this interview technique, include a few reminders about what to do if you hear "red flag" phrases, such as *"Usually, I ..."* or *"Every time I..."*. Since you cannot anticipate all the possible branches of a question, note several possible ideas for follow-up questions.

Decide which background information to collect, such as level of computer experience or length of time in the job. This works best during the recruitment process, but you can also ask at the end.

Choose Roles

Decide if you will interview as an individual or as a pair, where one person asks the questions and the other takes notes and/or records the interview.

- ***Interviewer:*** asks questions to a member of the target audience.
- ***Interviewee:*** answers questions from the interviewer. (Should be a member of the target audience.)
- ***Scribe:*** takes notes during the interview, ideally hand-written on paper. May also record audio or video.

During: Conduct a Story Interview

Duration: 15-60 minutes
Skills: video
Roles: interviewer, scribe
Contributors: designers, users
Audience: interviewer, scribe

Interview as a pair, if possible. The advantage of interviewing as a pair is that the interviewer can concentrate on formulating questions and listening to answers, while the scribe takes notes.

First introduce yourself, thank the interviewee for talking to you and explain the purpose of the interview.

Obtain informed consent from each interviewee by explaining the goal of the interview and how you will handle their data. Do not confuse "informed consent" with signing a legal consent form. You are morally obligated to ensure that the interviewee understands the consequences of being interviewed and still agrees. (See Mackay, (1995) for different perspectives on the ethics of using video in HCI.) If your university or company has an Institutional Review Board, follow their procedures. Even if you work for a small startup or are taking a class, follow basic ethical guidelines.

***Always* begin with a specific question** that encourages the user to describe a real experience, as recent as possible. You can ask about a particularly negative experience, a so-called "critical incident", which is the easiest to remember, but very positive experiences or "bright spots" are easy to remember as well. You can also ask the interviewee to "walk you through" their current or very recent experience when they created a specific artifact or "critical object". Finally, you can ask about what happened, step by step, earlier today or yesterday, even if it was a typical experience. Students often feel more comfortable if they say that the interview is for a class and that they are learning new interviewing techniques.

Avoid starting with a general question to "break the ice". While this is tempting, it sets the wrong tone and leads to general answers. You will find it very difficult to recover the interview, even if your next question is about a recent event, because the interviewee will assume that you are still interested in generalizations.

Begin with planned questions then probe for more detail, especially if relevant to your design problem. If the interviewee replies with "yes-or-no" or very short answers, change your question style. Listen to make sure their answers are in the past

tense: "First I did this ...", not present tense: "When I do this ..." The former involve real stories, the later are tutorials.

Choose different questions to elicit different results. Always start with *story questions* and only ask *tutorial* or *opinion questions* once you have obtained at least one good story. Order matters! Never ask for opinions or general descriptions first. However, do not worry if the interviewee offers an opinion in the middle of a story. Interviewees often offer opinions spontaneously and if they do not, you can always ask for them at the end. Interestingly, opinions that arise as they describe a real, specific activity are better grounded in their actual experience, and are more likely to truly reflect what they believe.

Pay attention to what they say and nod in acknowledgement. Interviewees usually enjoy talking about the details of what they do and the frustrations they face, since they rarely find someone who is interested enough to listen. Be that person!

Remember to pause to give the interviewee time to answer. Pauses encourage people to talk, so avoid jumping in with a new question immediately after their initial response.

Record both your questions and their answers.

Ask if the story was typical or unusual after the first story. Then, ask for another story about the alternative experience.

Figure 6. Probe for more details.

Assume that your questions will evolve in response to the interviewee's answers and that new topics will come up.

Always aim for specific details first. Only ask for generalizations after you have two or three interesting stories. Be sure to listen for "red flags", such as when the interviewee tells you what they usually do or offers a general opinion. If you hear, *"Normally, I ..."*, listen politely, then say: *"That's interesting, but can you tell me exactly what happened this particular time?"* If you hear: *"I think that ..."*, again, listen politely to their opinion, but follow up with: *"Great, can you describe a specific example?"*

Listen for *breakdowns*, *workarounds* and *user innovations*. Each can be the basis for a story that leads to new insights for design.

Record the interview if you can, but only with the interviewee's permission. Video will be easiest to watch afterwards, but interviewees find audio far less distracting.

Take notes, ideally on paper, as you go. However, be careful, since note-taking can distract you from listening to the person. Hand writing notes is less distracting during the interview, and makes it easier for the interviewer to look at the interviewee as they answer questions, as opposed to looking at a computer screen. If you must choose between listening and taking notes, either draw a quick symbol and focus on the person, or ask them to pause for a minute while you catch up. Note that typing on a computer usually requires more of your attention than writing by hand.

Handle data carefully. Consider both the size of the potential risk if the data is released, as well as the likelihood that its release will cause harm. For example, if your interview might reveal embarrassing information or raise legal issues for interviewees, let them know in advance and always give them the option of dropping out at any time.

Anonymize your interview data even if the risk of harm is negligible. Assign a unique participant identifier to each interviewee and analyze your data with participant IDs, not names. To meet European data requirements, we keep only one master list, on paper, that links users' names and personally identifiable data with the participant ID. For example, If Marie Dupont is assigned *P07*, her name will appear only once on the master sheet. All other data will refer to her as *P07*.

After: Create Reusable Design Resources

Format: story

The most important action immediately after an interview is to quickly to record your three most surprising or interesting insights. Try to give yourself additional time, ideally the same day, to review your notes. Use a different colored pen and fill in any missing details or anthing else that you remember. Star or highlight anything that strikes you as potentially important.

Later on, consider transforming the most interesting findings into "interaction snippets". (See Example 5 in Chapter 3.) Treat these as miniature storyboards that describe the interaction sequence: What did the user do, how did the system react and how did the user react? Or, alternatively, what did the system do, how did the user react and how did the system react to the user?

First, explain the interaction in the upper title: What does the user want to accomplish? Next, sketch the interaction in the upper boxes and add a text description below. (Interaction snippets should be short, but may include more than three elements.) Focus on surprises, especially *breakdowns*, *workarounds* and *user innovations*. Try to extract three to five interaction snippets from each story. Clearly, longer stories may include many more *interaction snippets*.

Story Interview Example

All of the examples in this book are based on a design project assigned for the *Design of Interactive Technology* course at the Université Paris-Saclay. Because the university is undergoing a great deal of construction, finding one's way around the campus can be difficult, especially for non-French speaking students.

The design brief was to design a new app that helps people navigate to different places via diverse transportation options (walking, biking, bus, train, car) despite the blocked roads and changed routes that arise to due to construction or poor weather.

Design brief:

"*Create a novel map application for a smartphone that improves user navigation.*"

Example 1. Story Interview

Interview date: 10 March 2023
Location: Café du Théâtre, Paris
Interviewer: Wendy Mackay
Interviewee: Participant 3

Initial question: *"Can you remember the last time you had a problem with a map application? Can you tell me what happened?"*
Answer: *"Last Thursday, my son asked me to help him deliver a table to a friend's house. He texted me the address and I copy-pasted the address from the text into Apple Maps to get a sense of how far away it was and generally how to get there. When it was time to leave on Saturday, I opened Apple Maps but the address had disappeared. So I had to go back to my son's text, scroll until I found the address, and then re-enter it. Since I took the car, I then had to manually re-enter the address from my phone, and kept both open, since they often show things differently."*

Probe question: Did you use the GPS and the phone at the same time?
Answer: *"Yes, since they show different things. Unfortunately, the phone was in "dark mode", which made it hard really hard to read in the daylight. It also showed lots of irrelevant information, such as local restaurants that I didn't care about, but sometimes a landmark was useful, especially since some of the physical street names are hard to see. In one case, I couldn't see the street name and wasn't sure I was in the right place. I zoomed in, but the landmark disappeared, so that didn't help. It wasn't clear which ones would stay and which would go away. I had to wait to get to the next marked intersection to be sure I was on the right track."*

Probe question: Did you have any trouble finding the address?
Answer: *"Yes. I missed the turn at a complex intersection that was really confusing with several branching streets. I'm still not sure if the car was wrong because it messed up the tracking, since the map did not turn as quickly as the car. Was it out of date and didn't know that one street was one-way? I've noticed that the accuracy within the city isn't great. Anyway, I went down the wrong street to avoid the one-way street, and had to loop around to get to the right address."*

Probe question: Do you have an example of something innovative you did?
Answer: *"Not sure if it's innovative, but I took a photo of the car's GPS display because it isn't cluttered with irrelevant restaurants and sent it to myself. I then overlaid written directions and mailed it to [name] so they could see the most direct route without lots of extra stuff."*

Example 1 shows an edited transcript of the actual questions and answers of one *story interview*, with both the opening question and additional questions that probed for more detail. Example 2 shows a breakdown analysis of this interview, including four situations where things went wrong; one workaround

Example 2. Breakdown Analysis

Breakdowns:

• The map didn't remember the address that was put in two days earlier, so had to re-find it and re-enter it.

• The phone was in "dark mode" so it was hard to see in the daylight.

• The map display updates more slowly than the physical movement of the car, so it was not clear which street was correct.

• Address appeared as a single point, but really spanned a whole block, so it was hard to figure out where to meet.

Workaround:

• Took both Apple Maps and the car GPS map sine they have different info.

User innovation:

• Took a snapshot of uncluttered GPS map in the car and added written directions.

that partially solved a problem; and one user innovation with a clever alternative solution.

Example 3 shows how a *story interview* can yield a wealth of useful information and give you specific, useful ideas about what users might need. Figure 7 shows a student rereading the results of his breakdown analysis. The next step will be to create implications for design.

Wrap-up

Designers can choose from a wide variety of methods for discovering information about users and their needs, but we have found *story interviews*, once mastered, to be the simplest and most effective approach. HCI researchers can find inspiration for novel design ideas, and UX designers can discover what users actually need. Asking people to generate specific stories almost always leads to new insights that aid design in a way that tutorial- and opinion-based interviews do not.

As with all design methods, it is important to assess the trade-offs of *story interviews* according to the current needs of the project. *Story interviews* are extremely useful for discovering real examples of what users actually do. Our research projects

Example 3. Implications for Design

- Make it possible to remember previous addresses.
- Make it possible to change the visual display to accommodate different lighting conditions.
- Account for inherent errors, such as the map being out of date or updating too slowly.
- Show addresses that cover more than a single location.
- Allow users to coordinate maps, so that they can see when each will arrive.
- Let different apps communicate with each other, such as from Apple Maps and car GPS.
- Allow users to easily annotate and share maps.

usually involve 10-12 *story interviews*, which we combine with other methods, such as *story questionnaires* and *participatory design workshops*.

Although the main benefit of a *story interviews* are the resulting rich, qualitative descriptions of users' real-world experiences, *story interviews* are also time-consuming to conduct and analyze. You may find it difficult to recruit and schedule appropriate members of the target audience and a truly rigorous analysis of qualitative data takes time and skill.

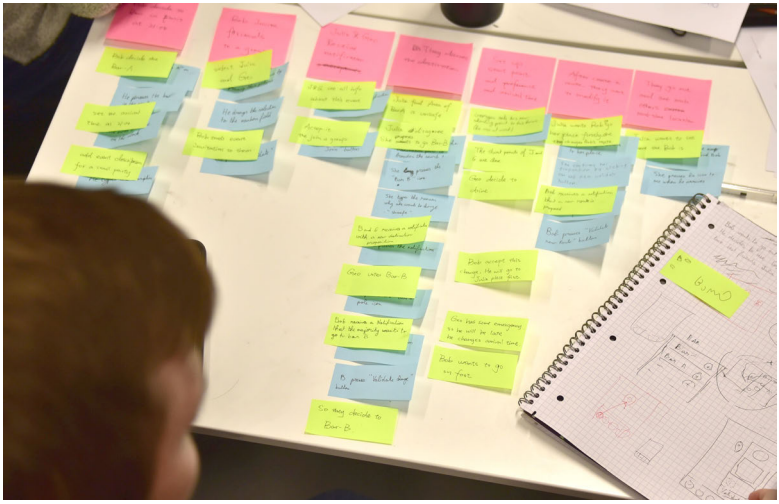


Figure 7. A *breakdown analysis* categorizes interview data into *breakdowns*, *workarounds* and *user innovations*.

Trade-offs

PRO: Can capture long, open-ended answers and probe for more information, in greater depth.	**CON:** Time consuming to find and interview a small number of users, requires interviewing skill, time consuming to analyze data.

Remember: Before you begin an interview, it is important to ask for permision. Be sure to set realistic expections: make sure the interviewee does not expect you to design a custom-made solution to their interface problems! When you ask questions, be sure to start by asking for a specific, recent story about a memorable event and use follow-up questions to gather as much additional information as you can.

Be careful! If you hear phrases in the present tense, such as *"I usually do this..."*, the interviewee has stopped telling you a specific story and is now explaining how things usually work. Shift the conversation back to their story. Finally, remember to focus on stories, not opinions; avoid "obvious" or leading questions and wait until the end of the interview to ask for their opinions.

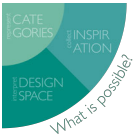
Advice

Ask permission, set realistic expectations, begin with real, recent story, probe for detail. **Caution!** If you hear "usually I ..." it's no longer a story, it's a tutorial..	**Did you remember to...** focus on recent or highly memorable stories? Start by asking for a specific story, not opinions? Avoid asking 'obvious' questions? Ask open-ended questions only at the end?

Bibliography

1. J.P. Djajadiningrat, WIlliam W. Gaver and J.W. Frens (2000) Interaction Relabelling and Extreme Characters: Methods for Exploring Aesthetic Interactions, In *Proceedings of ACM DIS'2000, Designing Interactive Systems*, Brooklyn, N.Y.: ACM Press. pp. 66-71.
2. Flanagan, John C. (1954). The Critical Incident Technique. *Psychological bulletin* 51(4) pp. 327-358.
3. Hutchinson, Hilary, Mackay, Wendy E., Westerlund, Bosse, Bederson, Ben, Druin, Alison, Plaisant, Catherine, Beaudouin-Lafon, Michel, Conversy, Stephane, Evans, Helen, Hansen, Heiko, Roussel, Nicolas, Eiderbäck, Børne, Lindquist Sinne and Sundblad, Ynge (2003) Technology Probes: Inspiring Design for and with Families. In *Proceedings of ACM CHI 2003*, 5(1). New York, NY: ACM Press, pp. 17-24.
4. Mackay, Wendy E. (1995) Ethics, Lies and Videotape. In *Proceedings of ACM CHI 1995*. Denver, CO: ACM Press. pp. 138-145.
5. Mackay, Wendy E. (2002) *Using Video to Support Interaction Design* In ACM/CHI 2002 Extended Abstracts, 88 pages. Inria Multimedia Services, Roquencourt, France.
6. Mackay, Wendy E. (2004) The Interactive Thread: Exploring Methods for Multi-disciplinary Design. In Proceedings of ACM/DIS'04. Conference on Designing Interactive Systems. Cambridge, MA: ACM Press, pp. 103-112.
7. Mackay, Wendy E. (2019) *Designing with Sticky Notes.* In *Sticky Creativity: Post-It Note Cognition, Interaction and Digitalization* Christensen, Halskov, and Klokmose, (Eds.) London, UK: Explorations in Creativity Series, Academic Press.
8. John Pruitt and Tamara Adlin (2006) *The Persona Lifecycle*, New York, NY: Morgan Kaufmann.
9. Zemke, Ron and Bell, Chris (1989) *Service Wisdom: Creating and maintaining the customer service edge.* Minneapolis, MN: Lakewood Books.

Inspiration



What is possible?
Finding inspiration and generating new ideas

"The best way to have a good idea is to have lots of ideas."
— Linus Pauling

Introduction

Once you have a clear understanding of who your target users are and what they need, you should gather or generate ideas about how to address their needs with a new design. Which existing design concepts and technologies are relevant to the current design problem? What kinds of interaction techniques will be most useful? What is the space of design possibilities?

This requires drawing from various sources of inspiration, including the web and the work of other designers. You must assess which solutions already exist and then decide what must be developed for your specific project. Another critical source of inspiration are the users themselves. Take advantage of your studies of users not only to discover their needs, but also to inspire solutions. Learning about one user's context and how that affects their use of the system may suggest ideas for other users in related or even radically different contexts.

Users who face breakdowns in the technology often come up with useful workarounds. Treat these not only as a source of problems to solve, but also as inspiration for different types of solutions. Finally, although you should never treat users as *the* source of design solutions, you may be lucky enough to find some users who develop free, field-tested *user innovations*.

Talking directly to users offers you a rarely tapped source of ideas for design solutions that meet actual user needs.

"Some users develop free, field-tested *user innovations*."

This chapter summarizes methods for finding sources of inspiration in Table 2, including how to *collect* and brainstorm ideas; *represent* them at different levels of granularity; and *interpret* them in the form of a *design space*.

Next, the chapter explains how to *brainstorm* ideas, first by describing how *classic brainstorming* generates as many ideas as possible and then how *video brainstorming* builds on those ideas by transforming them into short video clips (*interaction snippets*) that illustrate how a user might interact with a simulated version of the idea.

This chapter concludes by explaining how to conduct a well-structured *video brainstorming* session, including a detailed description of how to prepare *before* you start; what to do *during* the activity and what to *produce* afterwards to create reusable design artifacts. Each method is illustrated with multiple examples from a sample design project.

Finding Sources of Inspiration

Where can we find or generate sources of inspiration? The *Inspiration Methods Table* below summarizes diverse activities for creating new ideas relevant to the design problem. Individual methods may involve one or more of three key activities: *collecting*, *representing* and *interpreting* ideas.

Designers have multiple ways of *collecting* design ideas: you can gather ideas from the web, the research literature and, importantly, users themselves. For example, commercial websites such as *Pinterest* include a wealth of design ideas, websites such as *datavizproject.com* and *dear-data.com/theproject* offer curated data visualization examples, and our own *HCI Museum* (*hci-museum.lri.fr*) summarizes interaction techniques from the HCI research literature. You can also *brainstorm* or generate your own ideas, both about new technologies or about new techniques for interacting with a system.

Designers can also choose from multiple ways of *representing* ideas. You can select the most promising ideas to create an

idea archive or a set of *interaction snippets* that combine sketches and text to describe the back-and-forth interaction between a user and a new system. The next level of representation includes *paper* and other *prototypes*, where designers or users act out the interaction to simulate the proposed user experience. Here, the physical act of trying out the idea adds significant additional depth to everyone's understanding of the idea.

Other live methods for exploring ideas in greater depth include embodied methods such as *improvisation* and *bodystorming*. Interestingly, *video brainstorming*, which involves shooting a video of the above interaction, affects how the designer or user interacts, usually with a correspondingly deeper consideration of the details of the idea. Finally, various methods of *engaging* users, such as *cultural probes* and *technology probes*, involve creating prototypes for users in the context of their daily lives, and offer a rich though time-consuming, method of finding inspiration.

Finally, you need to *interpret* your design ideas. One of the first steps is to *classify* them according to categories or along *dimensions* relevant to the design brief. These may be relevant to the overall *system* design, or to situations that arise from *stories* such as types of breakdowns. A more elaborate process involves *sketching* a *design space*, which lays out the different design dimensions in relationship to each other and identifies gaps or opportunities for new ideas. All of these methods are designed to produce useful takeaways, in the form of reusable design artifacts, that contribute to later design activities.

Inspiring Ideas	System	Activity	Story
Collect	Web links Brainstorm ideas	GATHER IDEATE	Literature review Brainstorm interactions
Represent	Idea archive Video brainstorming Improvisation Cultural probes	SELECT SIMULATE EMBODY ENGAGE	Interaction snippets Video brainstorming Bodystorming Technology probes
Interpret	Idea dimensions Design space	CLASSIFY SKETCH	Breakdowns Interaction snippets

Table 2. Inspiration Methods

About Brainstorming

The most common term for generating ideas is *brainstorming*, which was initially introduced by Alex F. Osborn (1953) to help design teams generate new ideas. His key insight was to emphasize quantity over quality — to generate as many ideas as possible without criticizing them.

Today, designers can choose from a wide variety of brainstorming methods, each with different goals and methods. For example, the "*Five Whys*" method was developed by Taiichi Ohno, (2021) at Toyoto to dig deeper and deeper into discovering the root cause of a problem. From there, designers brainstorm solutions.

Another popular method is "*Crazy 8*" brainstorming, which asks each participant in a design sprint to sketch eight ideas: "*How might we...*" perform a particular project-related task. Next, everyone "plays back" or explains their ideas to the group. Finally, everyone places their sketches on the wall and everyone votes for their three favorite ideas.

This brainstorming approach takes advantage of academic research (Diehl & Storebe, 1987) that demonstrates that *individual brainstorming* generates a larger, more varied set of ideas. On the other hand, *group brainstorming* benefits from building on other designers' ideas and learning about each others' perspectives, which adds to team spirit.

Although a wide variety of brainstorming methods are available, the basic rules still apply: avoid criticizing ideas, produce as many ideas as possible, build on each other's ideas, and encourage stupid, wild or exaggerated ideas.

Combining Brainstorming Methods

Video brainstorming (Mackay, 2002) builds on *classic brainstorming* by selecting the most promising ideas to prototype and experimenting with how they will look and feel.

Classic Brainstorming

Classic brainstorming is based on the observation that people do not enjoy being criticized or made to feel stupid. This is why the fundamental rule of *brainstorming* is to avoid criticizing or discussing ideas. Instead, the goal is to generate as many ideas as possible in the allotted time. To reinforce the rule that it is impossible to produce "bad" ideas, many *brainstorming* methods insist that every participant include at least one idea that they think is stupid — but not identify which one it is. This results in more lively sessions where "stupid" ideas serve as the spark for new design directions.

You should always schedule *brainstorming* for a limited period of time, since it is hard to maintain your energy for more than about 30 minutes. Think of *group brainstorming* as popping corn. Usually, ideas are generated slowly at first, as everyone thinks. Then, ideas come faster and faster as new ideas inspire other ideas. But listen for when everyone starts to gets tired and slow down — stop before you burn out … or the popcorn burns!

Another issue is what to do when you get blocked and cannot think of any new ideas. One useful technique is to read a list of "opposites", such as "fast vs. slow", "cheap vs. expensive", "tiny vs. huge" or "smart vs. stupid". Then, revisit your existing ideas and push them to both extremes: *What would be a "fast" version of this idea? What would be the "slow" version? What is would be a "cheap" idea? What is the really "expensive" version?*

Mackay et al., (2000) shows that how you represent an idea affects what kind of contribution it will make to the design. Ideas represented as short text phrases are the fastest to produce, but yield the least understanding, and are usually hard to interpret later. Hand-drawn sketches are almost as fast, especially for visually oriented designers, but have the potential to convey ideas in a richer, more evocative way and are easier to re-interpret later. Video brainstormed ideas offer the deepest insights, but take the longest to produce. The rapidity of generating the idea is thus inversely proportional to the benefits of the idea for future design. All are useful — choose what best serves your project's needs.

Video Brainstorming

Video brainstorming (Mackay, 2002, 2019) builds upon ideas generated in a *classic brainstorming* session to illustrate and explore the details of how a user might interact with your new system. Although both methods try to generate as many ideas as possible, the use of video lets you highlight the *interaction* between the user and your idea.

Unlike static wireframes that encourage you to focus on screen layout at the expense of interaction, ideas captured on video inspire you to imagine interaction from the user's perspective. Simulate each idea by first creating a paper prototype and then asking another team member to interact with it.

"Create simple *special effects* by simulating time-lapse photography."

You can create simple *special effects* by simulating time-lapse photography. For example, record a user pressing a button. Next, pause or stop the camera while keeping it steady, ideally on a tripod or else braced against the camera person's body and insert a sticky note that represents a dialog box. Restart the camera to create the illusion of immediate feedback, as if the dialog box popped up in response the user's button press.

Be inventive! With a little ingenuity and experience, you can illustrate a wide variety of interaction effects. You can simulate a mouse cursor by cutting a strip from a transparency and marking an arrow at one end, or use a wooden skewer with an arrow drawn on a sticky note stuck to the end. You can also simulate typing by slowly pushing a line of pre-written text through a slit cut in the paper. Refer to our *ProtoTips* website for more ideas (*prototypes.lri.fr*).

Clearly *video brainstorming* sessions require more preparation and resources than *classic brainstorming*. Although they emphasize generating a large quantity of ideas, unlike *classic brainstorming*, they also consider quality. Individual ideas need not look polished, but it is important to shoot them so that others can understand what is happening.

The Video Brainstorming Method

In a nutshell: Record a video of one or more interaction snippets that illustrate how users would interact with a new design.

Participants: team
Level: beginner
Resources: brainstorm votes
Supplies: camera, kit
Access: none

The goal of *video brainstorming*, like *classic brainstorming*, is to generate as many ideas as possible. However, here the focus is on interaction: create short, named video clips that illustrate how a user would interact with the system, based on your idea.

Before: Prepare for Video Brainstorming

First, run a *classic brainstorming* session. You will save time, since participants will be able to choose their favorite ideas from the earlier session. However, participants should always feel free to come up with new ideas.

Gather prototyping supplies, typically colored paper, post-it notes, pens, transparencies, scissors and tape, plus any other office supplies that seem relevant.

Make sure you have tables and enough space to work and to shoot. Try to find a quiet area with natural light.

Break large groups into smaller groups ideally with four people each. A team of four can work in parallel, with two directors working on two ideas at the same time, while borrowing other team members as needed.

Ensure that each group has its own video device. Although you can use smartphones and tablets, a video camera on a tripod is easier to manipulate, especially if you can attach the camera so that it shoots downwards towards the table. This lets everyone see what is happening, and allows team members to manipulate the *paper prototype* in response to the actor's actions.

Limit session length to 90 minutes or at most two hours. All team members should be able to shoot at least two ideas per person, and experienced teams will shoot significantly more.

Choose Roles

Video brainstorming only works if each idea has exactly one director who decides everything about the idea, including its name, what the prototype looks like, how it will be shot, and who will perform what role. It is important that everyone supports the director. The best way to handle disagreements is not to argue about them, but rather to wait your turn and when you become the director, shoot the idea your way.

- ***Director:*** directs the video shoot, including deciding who performs which design roles.
- ***Actor:*** performs specified interactions with the *paper prototype* before the camera.
- ***Camera person:*** operates the camera during a design activity.
- ***Maker:*** creates paper prototypes or mockups.
- ***Scribe:*** takes notes during the session.

During: Run a Video Brainstorming Session

Duration: 30-90 minutes
Skills: sketch, video, prototype
Roles: moderator, scribe, participant
Contributors: designers, users
Audience: moderator, scribe, participant

Assign one director per idea. The director assigns roles to other team members, including who will create prototypes and manipulate them, as well as who will act, take notes and operate the camera. The director is also responsible for choosing the name of the idea and writing a brief, one-phrase description on the titlecard. The director also figures out how to frame each shot and any special effects.

Avoid arguments: This is the director's idea! The easiest way to waste time is to argue about how to shoot the idea. Remember that the goal is to shoot as many ideas as possible in a limited time.

Shift roles for every new idea, with a new director each time. Everyone should have a chance to try each role.

Shoot different variations if you disagree. For example, if you disagree with how the director is shooting idea #3, then take on the role of director yourself and shoot idea #3b.

Consider exploring a "theme and variations", where you all explore the same idea from different perspectives. You may

Figure 8. The Wizard of Oz technique lets you simulate complex interactions with paper prototypes.

decide to progress systematically through a set of ideas, or explore maximally diverse ideas to push the limits of the design space.

Label each idea by first shooting a colored titlecard. Use different colored paper to create paper titlecards, or use the built-in titlecards in *Video Clipper*. The titlecard should include the director's name, idea name and number, and a one-phrase description of the idea. This will not only save time as you shoot, but will also help you find individual ideas later, as you fast-forward through your collection of *video brainstormed* ideas.

Create a master idea list that summarizes the design session, including date, place, topic and participants and keep a running list of the ideas as they are created. You should also place any paper titlecards in a folder, in order, to serve as a record of the ideas.

Ensure the text and images are large enough to read. Use a tripod or rest the phone or tablet on a stable surface to make sure the camera does not move while shooting.

Keep the ideas short, ideally 20 - 60 seconds, and only shoot one idea at a time.

Recreate the user's experience by showing what the user sees and experiences when using the idea.

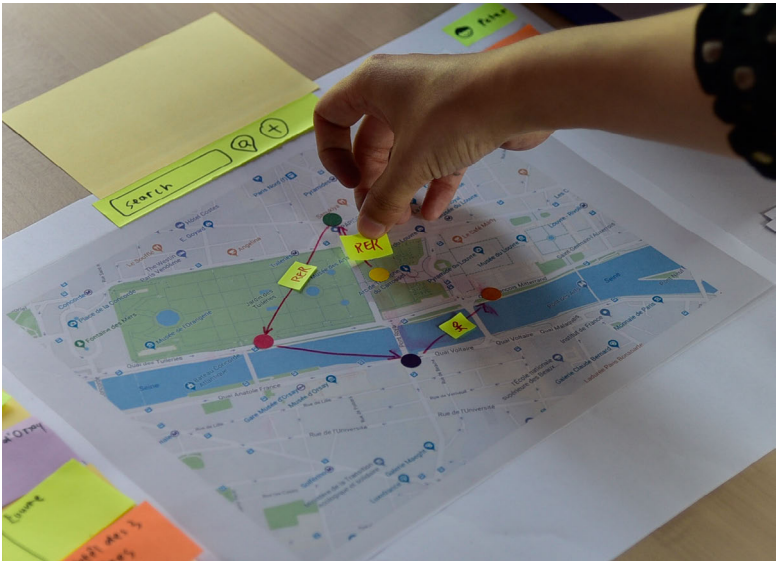


Figure 9. Set up your video prototype so that you can manipulate the interactive elements.

Let actors and prototype operators practice first before shooting the clip.

Consider "over-the-shoulder" shots that take the user's point of view when interacting with a phone or a laptop.

If you make a mistake, reshoot the clip. Editing in the camera, even if you have good video editing tools takes longer and shifts your focus from interacting to editing.

Use *special effects* to simulate different interactions (Check out our *ProtoTips* website — *Prototips.lri.fr* — for a large number of examples.)

Avoid using your fingers to point out what is happening. Your finger should only appear if it is directly part of the interaction, such as when dragging an icon on a tablet.

Use a voice over to explain anything that is not clear from the visuals. However, try to make the interaction understandable without audio.

Do not edit the video you just shot. Avoid trimming or rearranging video clips during the session. Remember: your goal is to capture user interaction with your future design.

Shoot as many ideas as possible. Do not agonize — shoot!

Example 4. Classic Brainstormed Ideas

- Show overall path with a focus circle around current location
- Above idea, but allow multiple waypoints, with close-up circles for each
- Send locations of multiple people to show up on everyone's map
- Highlight confusing intersections and show closeup circle to show where to go
- If street name isn't visible, show a local landmark
- Show different landmarks for people who are walking, biking or driving
- Snap a photo of directions on a laptop and upload as a map to the phone
- Do the opposite: send a map from the phone to a laptop
- Navigation arrows from phone onto smart watch
- Communicate from phone to a drone to show navigation

After: Create Reusable Design Resources

Format: list

Schedule a 10-minute debriefing at the end of the session. Show everyone the groups' *video brainstormed* ideas and let everyone vote for their favorites. Use the master list of ideas as a guide for remembering the ideas and finding a relevant video clip. This will prove especially useful if you later reuse one of these ideas in a *video prototyping* session.

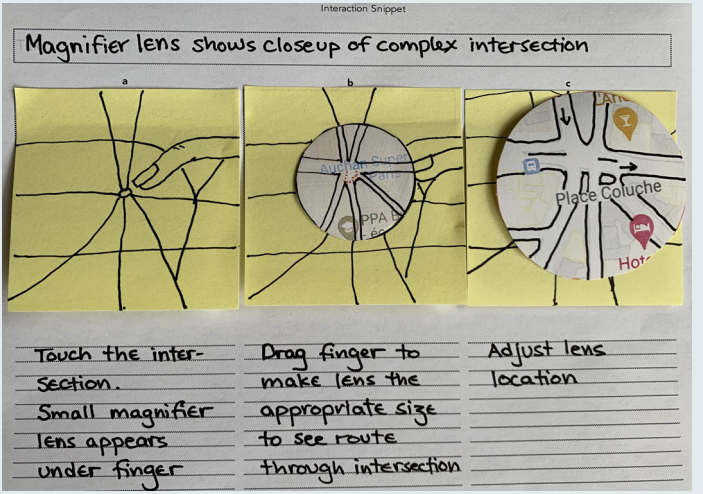
Video Brainstorming Example

The next set of examples are directly influenced by the examples shown in Chapter 2. All are based on the same design brief: "*Create a novel map application for a smartphone that improves user navigation.*" Example 4 shows the set of brainstormed ideas generated after a *classic brainstorming* session, which was inspired by the *story interview* in Example 1.

Example 5 shows an *interaction snippet* that illustrates how the user expands the magnifying lens by dragging her finger. Example 6 is a still image shot from the *video brainstormed* magnifying lens idea. Here, the user interacts with *DynaRoute* on her smartphone. She taps the route and when the magnifying lens appears, she drags her finger outwards to make the corresponding intersection larger and easier to read.

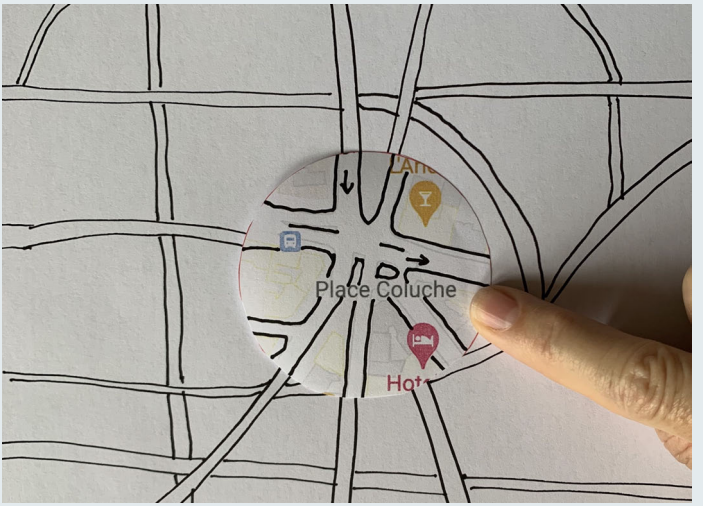
Example 5. Interaction Snippet

Figure 10. Interaction snippet shows how to create and enlarge a magnifing lens.



Example 6. Video Brainstormed Idea

Figure 11. User drags the edge of the magnifing lens to make it bigger.



Wrap-up

As with any method, it is important to assess the trade-offs associated with running a *video brainstorming* session, based on the current needs of the project. The key benefit is that *video brainstorming* helps designers generate ideas that focus on how users will interact with the system, not just the functionality or how the system will look. They help designers think about the design from the user's perspective and facilitate communication within the design team.

Video brainstormed ideas also last longer and have greater impact than hand-written notes or sketches on sticky notes. We have reviewed *video brainstormed* ideas years later as a useful snapshot of what the design team was working on at the time. (We also uncovered some useful ideas that we had forgotten.)

Short video clips make it easy to communicate interaction ideas to other designers, users and developers. The latter especially welcome *video brainstormed* ideas because they offer a much clearer idea of how the system should actually work.

When designers follow the rules — primarily to stop arguing and let one person direct — experienced design teams can create an idea in a few minutes, and a well-prepared team of four designers can shoot a dozen ideas in an hour.

Even so, *video brainstorming* sessions take significantly more time and effort to run, which is why they are most effective when combined with a *classic brainstorsming* method. Our projects usually involve multiple *video brainstorming* sessions and are an especially fun and informative activity in the context of a *participatory design workshop* with users.

Trade-offs

PRO: Generates reusable videos that explore the details of interaction.

CON: Generates fewer ideas.

Be sure to run a *classic brainstorming* session first where you either generate ideas invididually or as a group to establish a solid base of ideas to choose from. Spend a bit of time thinking about how to shoot each idea and use "special effects" such as pausing the camera between the actor's movements to make objects seem to appear and disappear.

Video brainstormed ideas are short, usually 20 to 40 seconds each, and should not be confused with longer *video prototypes* that tell a story according to a *future scenario*. Be sure to choose one director per idea and follow the director's lead. Shoot different variations rather than arguing about them.

Advice

Select brainstormed ideas, create paper prototypes and shoot the interaction.
Caution! Keep ideas short, do not create full future scenarios! Shoot variations if you disagree.

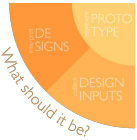
Did you remember to... choose one director for each idea? Avoid arguing, and follow the director's lead? Shoot variations to capture disagreements?

Figure 12. Use a tablet to try out different ideas.

Bibliography

1. Diehl, A. and W. Stroebe (1987) Productivity loss in brainstorming groups: toward the solution of a riddle. *Journal of Personality and Social Psychology*, Voume 53, Number 3, September, pp. 497-509.
2. Ohno, Taiichi, Bodek & Lowenstein, Norman. (2021) *Toyota Production System: Beyond Large-Scale Production* Milton Park, UK: Taylor & Francis Productivity Press.
3. Alex F. Osborn. (1953) *Applied Imagination: Principles and Procedures of Creative Thinking*, New York: Scribner's.
4. Mackay, Wendy E., Ratzer, Anne, and Janecek, Paul. (2000) *Video artifacts for design: Bridging the gap between abstraction and detail.* In *Proceedings of ACM DIS 2000, Conference on Designing Interactive Systems.* Brooklyn, New York. ACM Press. pp. 72-82.
5. Mackay, Wendy E. (2002) *Using Video to Support Interaction Design* In ACM/CHI 2002 Extended Abstracts, 88 pages. Inria Multimedia Services, Roquencourt, France.
6. Mackay, Wendy E. (2004) The Interactive Thread: Exploring Methods for Multi-disciplinary Design. In Proceedings of ACM/DIS'04. Conference on Designing Interactive Systems. Cambridge, MA: ACM Press, pp 103-112.
7. Mackay, Wendy E. (2019) *Designing with Sticky Notes.* In *Sticky Creativity: Post-It Note Cognition, Interaction and Digitalization* Christensen, Halskov, and Klokmose, (Eds.) Explorations in Creativity Series, Academic Press.

Design



What should it be?
Prototyping designs from the user's perspective

"A good design is better than you think."
— Rex Heftman

Introduction

Once you understand your users and have a sense about what is possible, you need to create and explore a *design concept*. What features do users need? What will make sense to users? How will they interact with it in different situations and over time?

This requires exploring multiple alternatives, usually in the form of *prototypes*. Prototypes help you think: they suggest alternatives, reveal problems and help refine ideas. Even if not realistic in every detail, prototypes let designers, users and other stakeholders envision and comment on what the future design could look and feel like. Prototypes let designers quickly investigate alternative design directions from the user's perspective, and ensure that they have adequately explored the design space.

The act of building prototypes helps designers shift fluidly between refining the details of a design and enlarging the space of possibilities, what Buxton, (2007) refers to as "getting the design right and the right design". Mackay & Beaudouin-Lafon (2023) present a taxonomy of the different kinds of prototypes relevant to interaction design, from informal paper prototypes to functional systems, and explains their roles at different phases of the design process.

This chapter summarizes methods for gathering information about users in Table 3, including how to *collect* design alternatives; *represent* them as prototypes and *interpret* them in a synthetic way.

Next, the chapter offers a general explanation of prototyping, and describes how to build on a series of methods, including: a *current scenario*, which tells a realistic story of users' current experiences; a *future scenario*, which retells the *current scenario* as if the proposed design existed and a *storyboard*, which enhances the *future scenario* with *titlecards*, sketches, descriptions and shooting instructions. The *storyboard* then serves as a guide for shooting a *video prototype*, which shows how users will interact with the proposed design in a realistic context.

This chapter concludes by explaining how to conduct a well-structured *video prototyping* session, including a detailed description of how to prepare *before* you start; what to do *during* the activity and what to *produce* afterwards to create reusable design artifacts. Each method is illustrated with multiple examples from a sample design project.

Designing user-centered interaction

How do you design an interactive system from the user's perspective? The *Design Methods Table* below summarizes diverse activities for designing interactive systems. Individual methods may involve one or more of three key activities: *collecting*, *representing* and *interpreting* ideas.

Most design projects start with a *design brief*, which provides a concise summary of the goals of the system and its target audience. You may also gather or generate a more specific list of *design requirements*. You can also take a story-based approach by gathering or generating *personas* and *current scenarios* based on the design artifacts you created in the discovery phase to establish concrete examples of users' needs in particular contexts. Be sure to gather any relevant design resources from earlier design phases, especially user characteristics and needs, selected brainstormed ideas and any design implications or requirements.

The next step is to *represent* your design through various system- and story-based prototypes. One of your roles as a

designer is to decide on the appropriate type and focus of each prototype, according to the design needs of the current phase of the project.

Early in the design process, you should start by sketching *paper prototypes* and *mockups*. You can also shoot a *tutorial video* that shows how the system should work; simulate a user's interaction with specified tasks using the *Wizard-of-Oz* technique; or code a *functional prototype*.

Story-based representations are particularly helpful for ensuring a truly user-centered design approach. Begin by transforming a *current scenario* into a *future scenario*; then sketching a *storyboard* and shooting a *video prototype*.

You can also simulate various types of interaction by exploring live scenarios with the *Wizard-of-Oz* method, or a *fixed path* prototype where the designer presents the user with a series of *wire frames* or an online sequence written in a language such as *Figma* or *Powerpoint*.

Finally, you will need to interpret your prototypes in the form of a written or sketched overview of the system. For example, you can describe the system as an *interaction table* to ensure completeness; or sketch a *design diagram* to illustrate how the system is supposed to work. Alternatively, you can used a story-based approach to describe the *design concept* or sketch a *flow diagram* to show the various paths the user can take over time.

Designing Prototypes	**System**	**Action**	**Story**
Collect	Design brief Design requirements	GATHER GENERATE	Persona Current scenario
Represent	Functional specification Paper prototype Tutorial video Wizard of Oz (tasks) Functional prototype	WRITE SKETCH SHOOT SIMULATE CODE	Future scenario Storyboard Video prototype (story) Wizard of Oz (scenario) Fixed-path prototype
Interpret	Interaction table Design diagram	DESCRIBE SKETCH	Concept description Flow diagram

Table 3. Design Methods

About Prototyping

Design is an active process that combines divergent thinking, where designers generate new ideas that expand the design space, with convergent thinking, where the goal is to refine ideas and select among them. Prototypes offer a flexible means for exploring the design space, helping designers discover new insights, try out new design directions, and consider the consequences of their design decisions.

"Prototypes are useful in all phases of the design cycle."

Although most commonly associated with design, prototypes are useful in all phases of the design cycle. The act of creating concrete artifacts not only helps designers express their ideas, but also helps them discover new insights about users and user experiences, and can also inspire new design directions. Prototypes, even early-stage ones, are also critical for evaluating the system.

Prototypes also offer a highly effective way to communicate design ideas, not only within the design team, but also with developers, managers, customers and, critically, users. Just as you can choose from a variety of brainstorming methods, you also have many prototyping options each with different goals, strengths and weaknesses. Mackay & Beaudouin-Lafon (2023) classify prototypes along five primary dimensions:

- **Representation** describes the physical form of the prototype, from rough sketches to wireframes to full computer simulations.
- **Precision** describes the level of detail, from rough and informal to highly polished.
- **Interactivity** describes the level of interaction with the prototype, from pre- recorded video or animations to fully interactive.
- **Lifecycle** describes the expected evolution of the prototype, from rapid "throw-away" prototypes to components of the final system.
- **Scope** refers to the part of the final system that is covered by the prototype, including breadth-first horizontal prototypes, depth-first vertical prototypes and path-based story prototypes.

Some prototypes are meant to be created quickly, then thrown away. Others serve an active role throughout the design process. All have the potential to add to your understanding of the design problem and the design solution: treat them as a key resource for future design activities.

Building on Earlier Prototyping Methods

Video prototypes build on a series of earlier design artifacts. For example, after creating a *current scenario* that tells the story of how one or more *personas* engage with the technology in a series events, each represented as an *interaction snippet*, to accomplish a particular task or address a specified problem.

After developing a *design concept*, the design team transforms the *current scenario* into a *future* scenario that envisions

Figure 13. Design teams can explore their ideas using *paper prototypes*.

Figure 14. Multiple designers create the illusion of interacting with the new design.

how the user will interact with the new system. From here, the design team can quickly illustrate the idea with a *paper prototype*, and use the *Wizard-of-Oz* method to simulate how a user would interact with the design.

Paper Prototype

The fastest form of prototyping involves paper, transparencies and sticky notes that represent aspects of an interactive system. Designers can get a quick idea of a variety of different layout and interaction alternatives. They can also create low-cost "special effects": a tiny triangle drawn on a transparency makes a handy mouse pointer, and sticky notes make great buttons and pull-down menus.

Paper prototypes are faster to create than carefully-drawn computer images, and encourage designers to explore a wider range of ideas. They are also useful in early design stages with users, because they automatically imply that the design open to interpretation and can still be modified. Paper prototypes and mock-ups are an excellent starting point for developing other types of prototypes.

Scenario

One of the methods for representing user data collected during the *Discovery* phase of a project is to create a *current scenario*, essentially a short, one-act play where personas who share characteristics of the target user population experience a series of events that illustrate key design challenges. The *current scenario* serves as the foundation for creating a *future scenario* in which the personas face the same circumstances, but instead interact with the new design concept.

Storyboard

The next step is to represent the *future scenario* as a *storyboard*, which combines quick sketches and text descriptions to convey how the personas will act and interact in each situation. The *storyboard* acts as a guide for shooting the *video prototype*, and also includes *titlecards* that describe key moments in the story, as in a silent movie, as well as how to frame and shoot each event.

Figure 15. Creating a *storyboard*.

Video prototype

Video prototypes, introduced by Mackay (1988, 2000), illustrate how users will interact with different aspects of the proposed design. However, unlike *video brainstorming*, the goal is not to expand the design space, but rather to refine it — to explore how users will interact with a specified set of design choices in a particular context.

Use *video prototypes* to challenge your design, not explain or sell it. Note how they differ from *tutorial prototypes*, which describe how the future system is "supposed" to work; and *marketing pitches*, which offer an idealized view of the system when everything works perfectly.

Video prototypes focus instead on exploring *breakdowns* and any other issues that require additional design. They will help you understand different aspects of your design — not only what works well, but just as importantly, what does not. Be sure to embed potential breakdowns into your *storyboard* and either deal with them directly in that *video prototype* or think of them as placeholders to remind you of which issues need further design iterations.

Figure 16. *Paper prototype* on a smart phone.

Figure 17. Shooting your own interaction with the prototype is tricky, but possible!

Video prototypes can take advantage of any of a variety of physical or computer-based prototyping techniques, from paper prototypes and physical mockups to fully functional working systems. However the focus is always on how users will react to and control the system, under realistic use conditions.

Treat *video prototypes* as quick and easy "interaction sketches" for communicating design ideas to various stakeholders, not only the design team, but also users and developers. For example, you can collaborate with users to design *video prototypes* in a participatory design workshop to help them envision and experience your design before it exists. We have also successfully used *video prototypes* as technical specifications for developers to help them implement the look and feel envisioned by the design team for a widely disseminated, open-source software product.

"Use *video prototypes* to challenge your design, not explain or sell it."

A *video prototype*, if well prepared, can be shot and viewed in a single design session, with no post-hoc editing. However, this requires more initial preparation than the methods described in the previous two chapters, and requires building on a series of earlier design artifacts. You will need both a clear *design concept* to explore, and a specific *current scenario* grounded in real user experiences gained from interviews and observations.

The Video Prototyping Method

Participants: team
Level: beginner
Resources: storyboard, video brainstormed ideas
Supplies: camera, kit
Access: none

In a nutshell: Record a video scenario of how users would interact with the new design in a realistic setting.

Video prototypes use video to illustrate how users will interact with a new system. The goal is to refine a single system concept, making design choices that highlight and explore a particular design path.

Unlike the earlier methods in this book, each *video prototype* requires extensive initial preparation that builds on previous design artifacts, beginning with a *future scenario*, usually transformed from a *current scenario*, followed by a *storyboard* that illustrates how user(s) interact with different aspects of the design by alternating between *titlecards* that explain the story and *interaction snippets* that show the interaction.

By themselves, each of the above methods help you explore, understand and communicate your system from the user's perspective. However, *video prototypes* go even further, by embodying the user's experience through interaction with a prototype.

Before: Prepare for Video Prototyping

Prepare the workspace with enough room for everyone to spread out as they create paper prototypes. Tables should be large enough for four people to work comfortably with large sheets of paper. Also, ensure that participants have access to additional space to shoot video. This might be a hallway, an extra room, or any other unusual space that can be appropriated for the current video.

Gather prototyping supplies. Although you should arrive with previously prepared paper or other prototypes, it is helpful to have white and colored paper of various sizes, sticky notes, transparencies, various types of pens, scissors, painter's tape and glue. This will let you add new ideas or variations as needed.

Choose Roles

Choose who will serve as the first director, then rotate the director role to other team members as you shoot the *video prototype*. Note that this is unlike in *video brainstorming*, where the director has complete control over the idea being shot. Here, the director must collaborate with the team, but has ultimate decision power in case of a disagreement.

One successful strategy for handling differences of opinion about particular interaction points is to create a "*branching storyboard*" that lets you compare the two alternatives. The branches might consist of different technology solutions, but might also show different user reactions to the same situation.

- ***Director:*** directs the video shoot, including deciding who performs which design roles.
- ***Actor:*** performs specified interactions during a video brainstorm or prototype.
- ***Camera person:*** operates the camera.
- ***Maker:*** creates paper prototypes or mockups.
- ***Scribe:*** takes notes during the session.

During: Run a Video Prototyping Session

Duration: 60-120 minutes
Skills: sketch, video, prototype
Roles: moderator, scribe
Contributors: designers, users
Audience: moderator, scribe

Use the storyboard as a guide for shooting the *video prototype*. Do not try to shoot a *video prototype* without a storyboard — figuring out shots as you go causes errors, invites arguments and is always much slower. *Storyboards* are flexible, so you can adjust them as circumstances change, but you need them to organize your time. They are also important for allowing the design team to work effectively together or in parallel.

Discuss how to present the new system in context including the initial establishing shot. Simulate settings with whatever is around or sketch simple cues about the environment on a whiteboard or flip chart. For example, simulate being in a car by drawing a steering wheel and a windshield on a whiteboard and placing a chair facing it. Shoot over the actor's shoulder as she pretends to move the steering wheel. Use transparencies or sticky notes to show what changes in the view through the windshield.

Test the prototype to make sure you can illustrate the idea. Rehearse once first, then shoot the *titlecard*, then shoot the interaction. Illustrate the idea through action as much as possible.

Ensure that *titlecards* tell the story, including the user(s) motivation and the key story incidents.

Consider how to handle audio. For example, will you need a voiceover to explain what is happening? Will actors speak dialog? Do you need ambient noise to communicate the situation?

Avoid adding music unless it is a part of the design.

Avoid trying to be funny. *Video prototypes* of people using *paper prototypes* are amusing in their own right, but explicitly trying to be funny rarely works and just distracts the audience.

Shoot the main titlecard including the project name, date and team members, then shoot a *titlecard* to introduce the personas and another to describe the initial situation.

Simulate the setting either with objects around you or draw relevant cues on a whiteboard or flip chart. For example, draw a large steering wheel on a white board to simulate being in a car.

Shoot an establishing shot to show the persona(s) in the initial setting.

Alternate *titlecards* and video sequences. *Titlecards* should tell the story and the *video sequences* show users interacting with the prototype.

Remember to include breakdowns! One of the key benefits of a *video prototype* is to explore what will happen under realistic use settings, including figuring out what can go wrong so you can fix the design.

Avoid wasting time on post-hoc video editing. Use the *storyboard* as a guide for shooting the *titlecards* and video clips in order so that everyone can watch the *video prototype* at the end of the session. If your video requires extensive post-hoc editing, you are no longer shooting a *video prototype* that helps you reflect on your design but have instead shifted to creating a *marketing pitch* for selling your idea.

After: Create Reusable Design Resources

Format: story

Review the full *video prototype* at the end of the session. If you used either the "editing-in-the-camera" technique (Mackay,

2002) or a system such as *Video Clipper*, you should have no trouble creating a finished, if rough, design artifact that will support future design iterations.

Note that being able to reflect on the *video prototype* at the end of the session in which it was shot greatly increases its value. Participants usually leave motivated and excited about the next design steps.

Video Prototyping Example

Showing how to create all of the intermediate steps that lead to a *video prototype* is beyond the scope of this book. However, the following examples will help you understand how to create a *video prototype*.

Example 7. Current Scenario

Personas
Lola is a 25-year old Masters student in HCI who is moving to Paris.
Bob is a 28-year old student in Lola's class who lives in Paris and goes everywhere by bike.
Carl is Bob's father. He lives in a different part of Paris and deliver a table with his car.

Situation: Last Thursday, Lola texted Bob her new address and asked him to arrive at 10:00. Bob emailed the address to Carl and asked him to bring the old dining table from the garage. Carl retyped the address from Bob's text into *Google maps* on his laptop. He saw that it should only take about 20 minutes by car, so he decided he should plan to leave at 9:40.

Breakdown: On Saturday morning, Carl reclicks on *Google maps* where he had entered the address, but it is gone. He spends several minutes trying to refind Bob's mail message to get the address, and then has to re-enter it into *Google maps*.

Next he enters the address in *Google maps* on the phone. When he gets in the car, he looks up the address on his phone and types it into the car's GPS system.

Breakdown: Carl arrives at a complex intersection and is not sure which direction to take. He double checks his phone and sees that the GPS is suggesting a different route. Worse, he realizes that the middle branch he was planning to take is a one-way street. He passes the "correct" branch and takes a "wrong" branch that is at least going the right way. He then has to figure out how to navigate back to the right route.

The design team chose examples from their *story interviews* to develop a *current scenario*. They created three personas with evocative names: Lola is local, Bob rides a bike, Carl drives a car. They selected problems their interviewees faced when using current map applications and compressed multiple breakdowns into a shorter time span to inspire specific design challenges.

Example 7 shows the first part of the *current scenario* that was written based on the *story interviews* in chapter 2. The full *current scenario* is available, along with empty worksheets on the *CHI'23 DO IT* course website.

Next, the design team created a new *design concept* inspired by *Instrumental Interaction* (Beaudouin-Lafon, 2000a) and the principles of *reification*, *polymorphism* and *reuse* (Beaudouin-Lafon & Mackay, 2000b). Refer to Beaudouin-Lafon, Bødker & Mackay (2021) for an explanation of how to use these and other *generative theories of interaction* to generate novel but grounded design ideas.

Example 8. Design Concept

DynaRoute is a persistent, interactive route that users can save, manipulate and share.

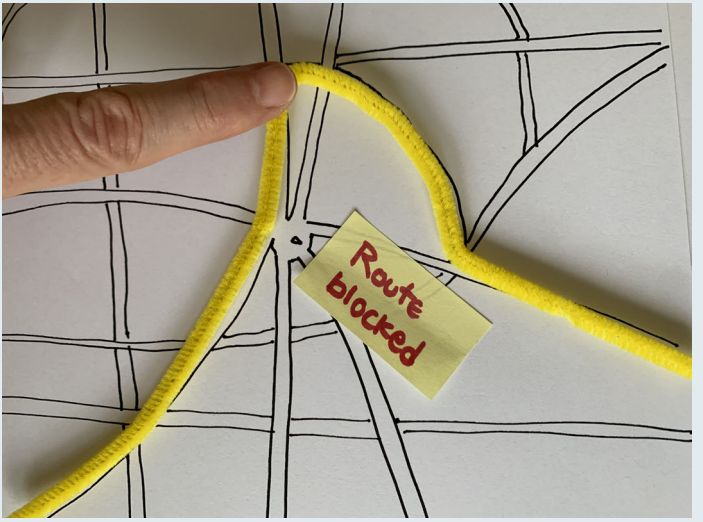


Figure 18. The user can manipulate the *DynaRoute* directly.

Example 8 shows the concept of a "*DynaRoute*", which is defined as a "*persistent,interactive route that users can save, manipulate and share.*" When the *DynaRoute* app proposes a route, the user can "reify" it or turn into a persistent object, with a starting point, intermediate way points and an ending point. Each *DynaRoute* appears differently according to the mode of transportation (bike, car, etc.) and can include a "magnifier" at any way point.

The team then transformed this *current scenario* into a *future scenario* that retells the story of the same *personas* in the same situation, but instead imagines how they might take advantage of *DynaRoute* to address the various breakdowns. The *future scenario* in Example 9 was based on the *current scenario* in Example 7 with the same personas, situation and breakdowns. This segment analyzes the design team's solution to the intersection mismatch.

"The design team introduced multiple breakdowns to push the limits of the design."

The next step was to transform the *future scenario* into a *storyboard*, composed of a series of *titlecards* to tell the story and a series of *interaction snippets* that show the new interaction, either based on previously brainstormed ideas or new ones.

Example 9. Future Scenario breakdown analysis

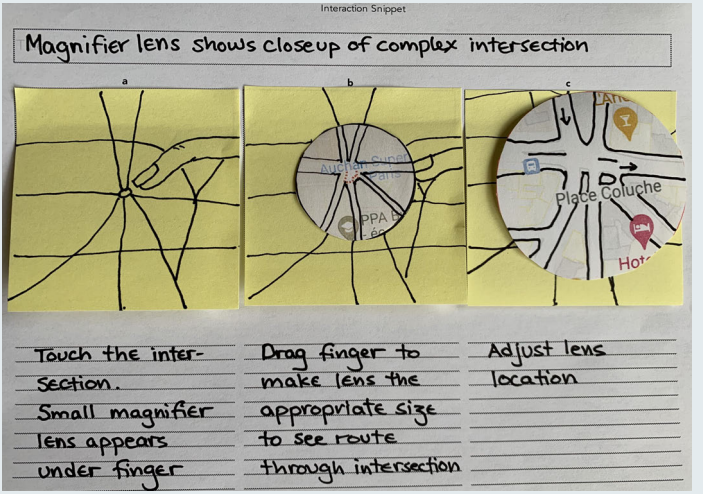
Breakdown solution: Lola has sent Carl and Bob each a customized *DynaRoute*, with "car" and "bike" routes marked accordingly. She used the magnifier feature to warn Carl about one of the complex intersections. When Carl approaches the intersection, *DynaRoute* shows that that the first branch is the optimal route.

Problems with the solution: The system has to figure out in advance where the "bad" intersections are. Carl will have trouble using the magnifier (although a passenger could do it easily).

Possible fixes: Since *DynaRoute* is an interactive object, Lola could send it to Bob and Carl, with "bike" and "car" variations, and highlight complex way points that she knows will be tricky. They could adjust the starting point to their separate addresses, or move *DynaRoute* to follow a better path. For example, Bob might prefer to bike through a local park.

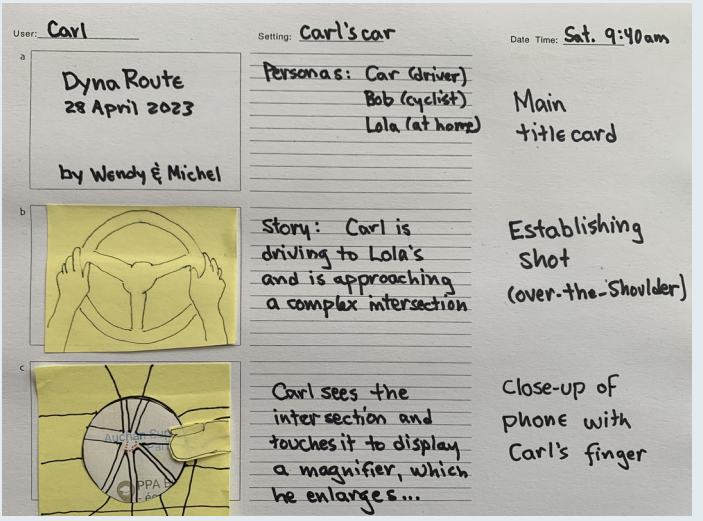
Example 10. Interaction Snippet

Figure 19. *Interaction snippets* combine sketches and text to show a user interacting with the new design.



Example 11. Storyboard

Figure 20. The *storyboard* guides how to shoot the *video prototype*.



Example 10 shows how the user drags the edge of the magnifier to enlarge it to the appropriate size. This makes it easier to see and follow the correct route at a complex intersection. The magnifier can also be dragged to adjust the location.

The next step was to create a *storyboard* (Example 11) using a combination of *titlecards*, sketches and text descriptions to delve deeper into the design details and examine exactly how the user would interact in each situation. The design team introduced multiple *breakdowns* to push the limits of the design, either by modifying the circumstances of the story or by considering possible problems caused by the new design. The full storyboard acted as a guide for shooting the *video prototype*.

Wrap-up

Designers can chose from a wide variety of prototyping methods to explore their design concept. *Video prototypes* are especially useful, since when mastered, they offer a rapid, compelling and enjoyable way to explore the design in context, identify and resolve potential breakdowns, and clearly communicate the design to users, developers, managers and other stakeholders in a form that remains open to ideas and suggestions.

However, as with any method, you need to assess the pros and cons of creating a *video prototype* according to the current needs of the project. The key benefit is that they provide a contextualized story of how users will experience the proposed design, which makes it easy to identify potential problems in the design. However, *video prototypes* can take time to produce, although far less than creating a working prototype.

We usually create multiple *video prototypes*, often in the context of *participatory design workshops*. They provide a fun and highly interactive way of collaborating with users.

Trade-offs

PRO: Provides a reusable video story of context-specific user interaction that is easy to evaluate.

CON: Time consuming to produce.

Remember to focus your storyboard on how users will *interact* with the system, and include *breakdowns* to explore problems that might arise when the design is used in a real context.

Remember to treat the *storyboard* as a guide for shooting the *video prototype*. Shooting a *video prototype* directly will take much longer, even though it seems faster. Avoid using your finger to point out things on the screen. If the interaction involves touching a tablet with a finger, we should see a finger. But if it involves using a mouse cursor, use a simulated cursor and avoid other types of pointing. Finally, use *video prototypes* as a low-cost way to push the limits of the design.

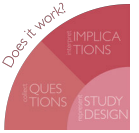
Advice

Shoot video of a storyboard that shows how users would interact with the new system.
Caution! Do not be afraid to shoot breakdowns, they can inspire new ideas and solutions!

Did you remember to ... shoot based on the storyboard? Distinguish user interaction from pointing? Include situations that push the designs limits?

Bibliography

1. Beaudouin-Lafon, Michel (2000) Instrumental Interaction: An Interaction Model for Designing Post-WIMP User Interfaces. In *Proceedings of ACM CHI 2000*, pp. 446-453.
2. Beaudouin-Lafon, Michel & Mackay, Wendy E. (2000) Reification, Polymorphism and Reuse: Three Principles for Designing Visual Interfaces. In *Proceedings of AVI 2000, Advanced Visual Interfaces*, Palermo, Italy. pp. 102-109.
3. Beaudouin-Lafon, Michel, Bødker, Susanne, & Mackay, Wendy E. (2021) Generative Theories of Interaction. *ACM Transactions on Computer-Human Interaction* (January 2021), 28(6), December 2021, Article 45. 54 pages.
4. *Sketching User Experiences: Getting the Design Right and the Right Design.* Morgan Kauffmann, San Francisco, CA.
5. Mackay, Wendy E. (1988) Video Prototyping: A technique for developing hypermedia systems. In *Proceedings of ACM CHI 1988 Extended Abstracts.*, New York: ACM.
6. Mackay, Wendy E. (2002) *Using Video to Support Interaction Design* In ACM/CHI 2002 Extended Abstracts, 88 pages. Inria Multimedia Services, Roquencourt, France.
7. Mackay, Wendy E. (2019) *Designing with Sticky Notes.* In *Sticky Creativity: Post-It Note Cognition, Interaction and Digitalization* Christensen, Halskov and Klokmose, (Eds.) Explorations in Creativity Series, Academic Press.
8. Mackay, Wendy E. and Beaudouin-Lafon, M. (2023) Prototyping Tools and Techniques. In Vanderdonckt, J., Palanque, P. and Winckler, M. (Eds), *Handbook of Human-Computer Interaction.* Switzerland: Springer Nature, 40 pages. (to appear).
9. Williams, Robin (1994). *The non-designer's design book: Design and typographic principles for the visual novice.* Berkeley, CA : Peachpit Press.



Evaluation (and Redesign)

Does it work?
Applying socio-technical principles for evaluation and redesign

"To design is always to redesign."
— Bruno Latour

Introduction

Once you have created a design, you need to assess whether or not it will work for users. Can they understand it? Does it meet their needs? What are the potential breakdowns and can users easily resolve them?

As in the discovery phase, HCI borrows many evaluation methods from the social sciences. Some are quantitative, usually to measure performance. Others are more qualitative and focus on the user's perception of the system.

Academic HCI researchers traditionally run more formal studies, with rules for rigorously conducting and evaluating them. UX designers can relax these rules, but still benefit from understanding why the rules exist. Your choice of evaluation will vary greatly according to the current design phase, your available time and resources and your access to users. The study's audience also matters: you can run simple studies for yourselves, but will need other approaches to engage users or demonstrate results to external stakeholders.

This chapter summarizes methods for evaluating interactive systems in Table 4, including how to *collect* study alternatives; *represent* them as study designs and *interpret* the results of the study to produce implications for design.

Next, the chapter shifts its focus to redesign methods. It describes how to combine a *design walkthrough*, which offers a step-by-step critique of a *scenario* or *video prototype* and *principled brainstorming*, which uses *socio-technical principles* to help generate ideas grounded in our understanding of users from the social sciences. When combined, these methods create a *generative walkthrough* that evaluates and extends design artifacts in a principled way.

This chapter concludes by explaining how to conduct a well-structured *generative walkthrough* session, including a detailed description of how to prepare *before* you start, what to do *during* the activity, and what to *produce* afterwards to create a reusable artifact for future design activities. Each method is illustrated with multiple examples from a sample design project.

(Re)designing interactive systems

How can you tell whether or not your design "works" for users? The *Evaluation Methods Table* below summarizes diverse activities for assessing different aspects of your design. Individual methods may involve one or more of three key activities: *collecting*, *representing* and *interpreting* ideas.

HCI researchers and UX designers can choose from methods for gathering and analyzing data, both quantitative and qualitative. However, before you decide what kind of evaluation study is most appropriate, you need to decide what research questions to ask.

You can define research or study questions either to assess system features or to determine whether the system can be used effectively under different contexts. Once you have a clear idea of which questions to ask, you will need to *represent* those questions the form of a study. You can design a *user study* with particular tasks for users to perform, and then *ask* them questions with a *questionnaire* about system features or *interview* them about their experiences. If you have a story-based design

artifact, you can run a *design walkthrough* that evaluates the effectiveness of the system from the user's perspective.

Some HCI researchers run *controlled experiments* to evaluate the participant's performance with respect to different features or interaction techniques. Experiments ask participants to perform pre-defined tasks that vary according to specified criteria, and produce quantitative data such as speed, accuracy and user preferences. Another option is a *structured* or *comparative observation* (Mackay et al., 2023). Here, study participants are exposed to comparable experiences, and then asked to compare and reflect on the advantages and disadvantages of each.

Designers can also run *field studies* where participants try out the new design in a real-world setting for days or weeks. Measures range from highly quantitative, such as *keystroke logs*, to highly qualitative, such as open-ended post-hoc *interviews*. Designers may also choose to run a *diary study*, where participants note their reactions to the design as they experience it in the field.

Finally, you need to *interpret* or analyze the data from these studies. Studies that gather quantitative measures of performance are usually analyzed with *inferential statistics*. Opinion data can be analyzed with *descriptive statistics* or *visualization tools*. Studies that gather subjective, qualitative data may be analyzed with qualitative analysis methods such as a *thematic analysis*. Some designers also like to create *mind maps* that illustrate the relationships across the data. Finally, designers need to identify additional *design requirements* and possible *implications for design*.

Finding Problems	System	Action	Story
Collect	Research questions Study questions	DEFINE	User contexts User activities
Represent	Questionnaire User study Controlled experiment Field study	ASK CONDUCT	Interview Design walkthrough Structured observation Diary study
Interpret	Descriptive statistics Inferential statistics Design requirements	ANALYZE	Thematic analysis Mind Map Design implications

Table 4. Evaluation Methods

About Redesign

Although the primary goal of an evaluation method is to identify the problems with the system, a second very important goal is to suggest how to improve it. Most HCI methods emphasize one of goals of the four quadrants, but we also need methods that are explicitly targeted at redesign.

The complete ***Do It*** book will include one chapter devoted to *evaluation* and another devoted to *redesign*. However, because evaluation methods are much better known in HCI, this book describes a more unusual method — a *generative walkthrough* — that was explicitly developed to support redesign. You cannot run an effective *generative walkthrough* unless you have already developed a design concept and story-based artifacts that explore how that design will be used in realistic contexts. Thus *generative walkthroughs* only make sense as part of an active redesign process.

One of the most effective ways to think about redesign is to consider the how a particular design concept will be used in the context of a larger "socio-technical system". Socio-technical design (Mumford, 2006) embraces a value system that actively takes social aspects into account when designing systems. The idea is that systems can only be understood and improved if both the "social" and "technical" aspects are considered together as interdependent components of a complex system.

A key challenge for socio-technical design is how to bridge the gap between abstract concepts and concrete action. Social scientists have contributed relevant theories about human behavior and cognition to HCI, but although we have adopted many social science *methods*, HCI researchers and UX designers often have difficulty translating their *findings* into specific implications for design (Dourish, 2006).

This is why it is important to derive concrete *socio-technical principles* from more general social science theory. This chapter describes how to effectively combine two methods, *design walkthroughs* and *principled brainstorming* into a single redesign method — *generative walkthroughs* — that successfully incorporates *socio-technical principles*.

Design Walkthrough

Yourdan (1979) introduced *structured walkthrough* as an efficient and effective method for programmers to perform peer reviews of their code in order to obtain feedback and improve the quality. The goal of a *structured walkthrough* is to work through a section of code, step-by-step, to identify as many problems as possible. Yourdan established a simple, but important set of rules: Groups should be small (3-7 people), the author should select and prepare the material in advance, everyone must be on time and the review should be limited to at most one hour. Members of the group should be at the same level in the organization — *structured walkthroughs* do not work well if the boss is evaluating the programmers while everyone else is evaluating each others' code.

Like *brainstorming*, the emphasis is on maximizing the quantity of bugs within the given time period. Participants should focus on finding problems rather than discussing solutions. However, unlike brainstorming, participants need not defer judgement. Instead, they should quickly specify potential issues and move on. Constructive criticism is always more effective and comments should be as specific as possible, referring to the material at hand, not generalities.

Design walkthroughs (Mackay, 2002) follow the basic principles of a structured code walkthrough, but can be applied to any sequentially organized design artifact. You can perform a *design walkthrough* on a written document, such as a *design brief* or a *user profile*, but they are especially effective for story-based artifacts, including *future scenarios*, *storyboards* and *video prototypes*.

Design teams can critique the design artifact from different perspectives, for example when team members adopt the roles of user, client, developer and marketing. Participants may also apply different heuristics or design principles and of course, should always include their own personal contributions.

Figure 21. A *design walkthrough* of a *video prototype.*

Principled Brainstorming

Lottridge & Mackay, (2009) describe how to transform insights gleaned from five social science theories into a story-based HCI design method called a *generative walkthrough*. They distill aspects of each general theory into a set of *socio-technical principles* that are directly relevant to interaction design.

The method itself combines the step-by-step critique found in a *design walkthrough* with *principled brainstorming* inspired by one or more *socio-technical principles*. They take advantage of the story-based nature of *scenarios*, *storyboards* and *video prototypes* to help designers reflect on how the user would interact with the proposed design in a particular context in light of each *socio-technical* principle.

Beaudouin-Lafon et al., (2021) describe the more general "*generative deconstruction*" approach for translating social science theory into HCI-relevant concepts and actionable principles. First, designers *analyze* the interaction events in a story-based design artifact to determine whether a particular *socio-technical principle* exists.

Next, they *critique* each event to determine whether or not the principle exists and if it has been applied effectively. Finally, they *construct* a new version of the artifact by brainstorming ideas inspired by the principle(s).

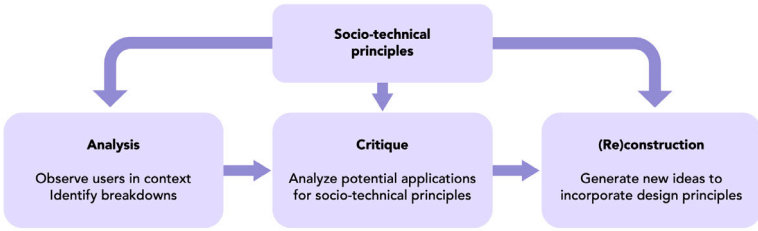


Figure 22. The process of "generative deconstruction" involves analysis, critique and (re)construction.

This book presents one principle — *Distributed Cognition* — to illustrate the approach. Hutchins, (1995) is an anthropologist who realized that human knowledge and cognition are not confined to the individual. The concept of *Distributed Cognition* applies models of cognition beyond the confines of physical skill to include groups of people, artifacts, places and culture. In other words, we aid our own memories by explicitly placing information on objects in the environment or by sharing it with other people.

Hutchins studied sailors in the U.S. Navy as they managed the navigation of their ship. He found that the work is highly coordinated and that different members of the group understood and took responsibility for separate parts of each task. Artifacts were essential for effective work coordination and members interacted with artifacts in different ways based on their background and experience. The theory of *Distributed Cognition* is informed by the observation that we use other objects and other people, to reduce our cognitive load for memory or communication tasks. Lottridge & Mackay, (2009) identified two key principles from this theory:

Memory aids: Memory is not entirely "in the head". Instead, we use physical objects and the environment to help us remember. This also lets us comfortably forget things as long as we know where to find them in the future. For example, if you place a sticky note with a shopping list on your front door, you can forget about the list until you leave to go shopping, at which point, you will remember to pick up some milk.

Boundary objects: People may share the same object, but interpret them differently (Star & Griesemer, 1989). The same object can have different meanings for people with different backgrounds or contexts. For example, a sticky note placed next to a coffee machine that explains how to change the filter will be understood differently by someone who has done it many times before and just needs a quick reminder, and someone else who has never done it and does not understand how the coffee maker works.

Generative walkthroughs distill these insights into a key directive: Help users reduce their "cognitive load" by incorporating objects and other people to support memory and communication tasks. By applying the principle of *Distributed Cognition* to each event of a series of events, you can imagine not only how you can improve the design at each interaction point, but also how the principle might lead you to reframe your design concept or even radically redesign it.

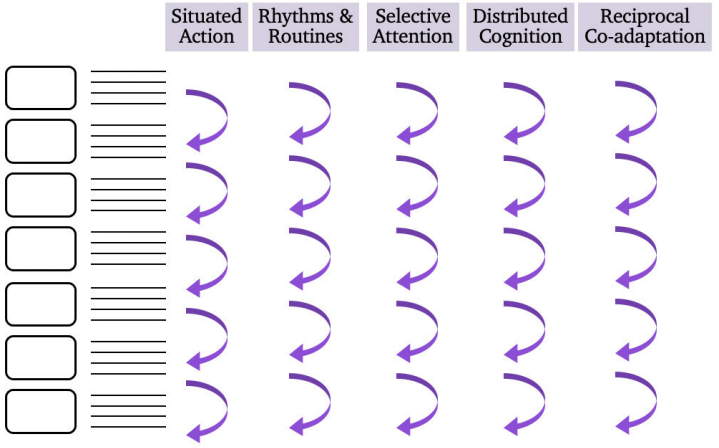


Figure 23. The *generative walkthrough* process illustrated with five socio-technical principles.

The Generative Walkthrough Method

In a nutshell: Apply sociotechnical principles at each step of a story-based design artifact to identify problems and suggest novel design possibilities.

Participants: 1 or 2 teams
Level: intermediate
Resources: future scenario, storyboard, video prototype
Supplies: notes, prototype
Access: none

Generative walkthroughs link *socio-technical principles* to story-based design prototypes in order to explicitly bridge the gap between theory and design. They are most useful in the middle phase of a design project, when the initial design already exists but merits a detailed consideration of the system from the user's perspective.

The combination of the step-by-step analysis of a *design walkthrough* and the generation of ideas through *principled brainstorming* will help you reassess and improve your design.

Perform the analysis at each interaction point in a story-based design artifact, in this case your *video prototype*: First determine whether or not the principle already exists, then consider whether the principle should or should not be applied here. Finally, brainstorm ideas about how the principle could improve the current design or suggest a new design direction.

Before: Prepare a Generative Walkthrough

Duration: 15-30 minutes
Skills: none
Roles: observer, scribe, videographer
Contributors: designers, users
Audience: observer, scribe, videographer

Find a quiet room with a table big enough for everyone to sit comfortably. Ensure that everyone can see the *video prototype*, which can be presented from a tablet or laptop, or projected onto a wall or screen.

Choose the design artifact, usually a *scenario*, *storyboard* or, in this case, a *video prototype*.

Choose the design principle(s) to focus on. If you choose several principles, decide whether to address each principle at each interaction point, or whether to make multiple passes through the prototype, one per principle.

Ensure that the scribe can write comfortably as they record all comments. Other participants can take notes as well.

Choose Roles

The presenter is responsible not only for running the session, but should be able to explain the relevant *socio-technical principle(s)*. As in brainstorming, it is important to make sure that all participants fully participate. Perhaps even more important, presenters need to stop arguments and long discussions and focus on generating critiques and ideas.

- ***Presenter:*** explains the principle(s) and presents the full prototype. Steps through each interaction point and keeps track of time.
- ***Scribe:*** records all comments from participants.
- ***Participants:*** critiques each interaction point and generates ideas according to the design principle(s).

During: Conduct a Generative Walkthrough

Remind everyone of the rules: constructive critique first, then generate an idea according to the design principle(s).

Remind everyone about the socio-technical principle(s). The presenter summarizes the relevant aspects of the principle to apply to the design artifact.

For example: *The principle is "_Distributed Cognition", which is based on the observation that we use physical objects and other people to reduce the cognitive load of remembering things. We treat physical objects as part of our memory, embedding them in the environment so that we can forget the details but know how and when to retrieve them when we need the information. We also rely on other people to reinforce our memory, sharing information and asking them when we need to remember something.*

Play the complete sequence first, without comments to familiarize everyone with the concept and the scenario.

Examine each *interaction snippet* step by step: First, identify any problems with the concept or the prototype. Second, analyze the *interaction snippet*: Does the principle already exist? Third, critique it: Does the lack of the principle cause a problem or offer an opportunity? Fourth: (re)construct it: Does applying the principle improve the design?

Figure 24. The presenter ensures that everyone can see the *video prototype* during the *generative walkthrough*.

Take the user's perspective when you brainstorm and generate concrete examples of how a real person might interact with the proposed technology. Try to generate at least three new ideas at each stopping point.

Critique designs, not people!. Start with specific critiques, positive or negative, then more general issues and suggestions.

Avoid defending your design choices. No design is ever perfect and early stage *video prototypes* are explicitly created to explore potential issues and breakdowns. As an author, if you disagree with a critique or a suggestion, note it down and consider it later. Your goal is to gather as many new insights and ideas as possible, not justify or apologize for what you did.

The scribe should read the list of design issues and ideas at the end of the session. Note that some ideas may suggest a radical design change. If you have time at the end of the session, this is the best time opportunity to discuss the new direction.

After: Create Reusable Design Resources

At the end of the session you should have a list of critiques and new ideas associated with each interaction point in your *video prototype*. You will be able to transform this list into a set of implications for design.

Format: list

Generative Walkthrough Example

Example 12 is a segment of notes from a *generative walkthrough* of the *video prototype* described in Chapter 4, which applied principles from *Distributed Cognition* to the *DynaRoute* concept.

The presenter began by playing the entire three-minute *video prototype* to give everyone an idea of the basic scenario and the design being assessed. Next, the presenter showed each interaction point in turn.

Example 12 shows the first set of comments about using the magnifier lens. In the analysis phase, the group decided that the current version of *DynaRoute* does not show any particular evidence of *Distributed Cognition*. When they critiqued it, they realized that it does not help the user remember anything and it works exactly the same for everyone. (Which is not necessarily a bad thing.)

Finally, they started brainstorming ideas about how using *Distributed Cognition* might improve the magnifier. They first considered using the magnifier as a "memory aid". One person suggested that the magnifier could record its own history, so that the user could see it again when the same situation arose in the future. Another idea was to create magnifiers that could be attached to "problem intersections", and have them pop up as needed.

When the group considered the magnifier as a potential "boundary object", they realized that what consistitutes a "problem intersection" is different for different people. A local person like Lola already knows how to navigate it, but newcomers like Bob and Carl will want it to act differently. Also, the magnifier should react according to their mode of transportation, since Bob is on a bike and Carl is in a car.

After the session, the group decided to rethink both *DynaRoute* in general and the *magnifier* feature in particular to make them easier to share and reuse.

Wrap-up

Although designers can chose from a wide variety of *evaluation* methods, few are intended specifically for *redesign*. *Generative walkthroughs* offer a principled approach to critiquing your design from the user's perspective, in light of insights gained from social science research.

As with any method, you should assess the pros and cons of running a *generative walkthrough* for your project. The key benefit is the systematic analysis of your design concept to discover problems and generate new ideas. *Generative walkthroughs* are a redesign method and are therefore only appropriate in the middle or later stages of a design project. They also require at least a limited understanding of the relevant *socio-technical principles*.

Trade-offs

PRO: Applies socio-technical principles to find problems and generate ideas.	**CON:** Requires (limited) understanding of socio-technical principles.

Choose a story-based artifact to analyze, such as a *future scenario*, *storyboard* or *video prototype*. Be sure to go through the entire artifact once, before starting your step-by-step assessment. Start with critiques, then transform them into new ideas.

Example 12. Generative Walkthrough Comments

Event 1: Using the magnifier lens
***Analysis*:** No evidence of distributed cognition.
***Critique*:** The magnifier does not really help remember anything, nor does it act differently for different users.
***Ideas*:** Let the user leave a trace of past uses of the magnifier, so they all pop up as needed. Consider sharing magnifiers associated with problem intersections with people who are unfamiliar with the area. (Locals will already know and will not need them.)

Be sure to avoid destructive comments: use constructive criticism and remember that the word "critique" can involve both negative and positive comments. Also, avoid getting defensive about your ideas — listen to the comment, note it down, and move on. As the designer, you are responsible for deciding which comments to implement, so you do not need to argue about the merits here. Finally, identify problems first, then brainstorm ideas, one socio-technical principle at a time.

Advice

Go through a story-based design artifact, identify problems and brainstorm ideas.
Caution! Be careful to separate critiques from criticisms.

Did you remember to... Identify problems first, then generate ideas? Explore one socio-technical principle at a time?

Bibliography

1. Beaudouin-Lafon, Michel, Bødker, Susanne and Mackay, Wendy E. (2021) Generative Theories of Interaction. *ACM Transactions on Computer-Human Interaction*, 28(6), December 2021, Article 45 54 pages.
2. Dourish, Paul. (2006) Implications for Design. In *Proceedings of ACM CHI 2006*, ACM. pp. 541-550.
3. Hutchins, Ed. (1995) *Cognition in the Wild.* MIT Press: Cambridge, MA.
4. Latour, Bruno. (2008) "A *Cautious Prometheus? A Few Steps Toward a Philosophy of Design.*" Keynote lecture given at History of Design Society, Falmouth, U.K., September 2008
5. Lottridge, Danielle and Mackay, Wendy E. (2009). Generative Walkthroughs: to support creative redesign. In *Proceedings of ACM Creativity & Cognition*. ACM, pp. 175-184.
6. Mackay, Wendy E. (2002) *Using Video to Support Interaction Design* In ACM/CHI 2002 Extended Abstracts, 88 pages. Inria Multimedia Services, Roquencourt, France.
7. Mumford, Enid. The story of socio-technical design: Reflections on its successes, failures and potential. *Information Systems Journal*, *16*, pp. 317-342.
8. Yourdan, Edward (1979) *Structured Walkthroughs*. NY: Prentice-Hall.
9. Star, S.L. and Griesemer, J.R. (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) pp. 387-420.

Conclusion

What's next?

"Seriously, create prototypes."
— Madeo Piervincenzo

This book was created for the ***DoIt*** course at the CHI'23 conference, with selected extracts from the forthcoming book entitled: *DOIT: The Design of Interactive Things*, by Wendy E. Mackay. It describes how the ***DoIt*** approach to interaction design highlights *interaction* from the user's perspective, and offers useful advice for working successfully as an interaction design team.

The four subsequent chapters each cover one quadrant of the ***DoIt* wheel**: finding out about users, generating ideas, prototyping designs and evaluating systems. The chapters then compare method variations or demonstrate how methods can build upon each other. Finally, each chapter presents a detailed description of a design method chosen for its relevance to story-based design, with multiple examples from a sample design project.

The complete *DOIT: The Design of Interactive Things* book will explain the design process in greater detail and how the methods relate to each other within a larger context. The book will also provide detailed summaries of methods from each quadrant, including their origins, why they are useful for which kinds of design challenges, and when to use them.

In addition to providing detailed instructions for how to perform each method, the book will also provide examples that illustrate what to do — or not do. Additional chapters will include "*Theme & Variations*", which describes how to modify, extend and create new methods; "*Redesign*", which describes

methods dedicated to generative redesign; and "*Participatory Design*", which describes how to involve users throughout design process.

The complete ***Do It*** book will also address ethical issues and practical advice on sketching, prototyping and shooting video, as well as provide worksheets for each primary method and pointers to online resources and tools.

Figure 25. Paris creARTathon'22: Design a creative, intelligent, interactive object

This book was written using Asciidoc (https://asciidoc.org) and typeset by Nicolas Taffin in April 2023, in a web browser with the help of CSS for paged media, javascript and the PagedJS polyfill (https://pagedjs.org).

Fonts in use are Peter Bilák & Nikola Djurek's Zilla Slab (https://www.typotheque.com) and Meredith Mandel's Chunk Five (https://www.theleagueofmoveabletype.com).

Fonts and software used are under a free licence.

NOTES & SKETCHES

NOTES & SKETCHES

NOTES & SKETCHES

NOTES & SKETCHES

NOTES & SKETCHES

NOTES & SKETCHES

Printed in France by Nouvelle imprimerie Laballery (Clamecy)
Print number 303.776

WENDY E. MACKAY

DO IT: THE DESIGN OF INTERACTIVE THINGS

CHI 2023 Preview

***DO IT: The Design of Interactive Things* provides a practical, hands-on guide for designing interactive technology.**

Based on a coherent set of design principles, the book offers rapid and effective techniques from multiple disciplines that students, system designers and researchers have used to create successful startups and publish award-winning research. This book includes a subset of the material in the *Do It* book, intended for students in the CHI'23 Course on the *Design of Interactive Things*.

Wendy E. Mackay is a Research Director, *Classe Exceptionnelle* at Inria Saclay and the 2021-22 Annual Chair of Computer Science at the Collège de France. She received her Ph.D. from MIT and has published over 200 research articles in Human-Computer Interaction. She directs the ex)situ research lab at Inria and teaches HCI at the Université Paris-Saclay.

She has been innovating user-centered tools and participatory design methods since the 1980s, when she helped launch IVIS, the first commercial interactive video system. After developing over 30 interactive video products, she returned to academia and has published award-winning research in tangible computing, multimedia communication, generative theories of interaction and human-computer partnerships. She has been teaching user-centered research and design in academic and corporate settings for four decades, and her user-centered methods have been widely adopted by the HCI community.

This work was supported by European Research Council (ERC) Advanced Grant No 321135 "CREATIV: Creating Co-Adaptive Human-Computer Partnerships" (Principal Investigator Wendy Mackay).

15 €
Printed in France
https://ex-situ.lri.fr/