Story Interview

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

INTRODUCTION

Phase: discovery
Activity: collect

Story interviews capture a detailed description of how a user interacted with a particular system at a particular time. The goal is to record a step-by-step account of what happened, with as much detail about the user's interaction as possible.

Never start with a general question about how they like the system. For example, you might ask a user about a recent, memorable situation that required a map. Start with a semi-structured *critical incident* or *critical object* question, and then ask follow-up questions to probe for more detail.

Specific stories lead to design insights; tutorials and opinions do not. Listen carefully. If the user describes how something 'usually' works, in the present tense, it is a *tutorial*. If the user says what they like or dislike, it is an *opinion*. In both cases, shift back to what actually happened, with details about a specific past event, so that you end up with a detailed *story* of the interaction.

Always ask for permission if you record the interview and always take notes. Highlight surprises, paying particular attention to breakdowns, workarounds, and user innovations.

Background: 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, based on 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 an event, whereas a sociologist's interview should contribute to a deeper, more general understanding of human behavior.

Design-oriented interviews serve yet another goal–to improve your understanding of behavior so as to directly influence your design. We distinguish among three types of questions which produce three different types of results: stories, tutorials and opinions, in order of usefulness for design.

Story interviews are the most useful, since they capture detailed examples of real-world interaction. Stories are always told in the past tense: the user recounts what happened, step-by-step, with as much detail as possible. The most effective story questions take advantage of human memory by asking the user to recount a highly memorable event–a *critical incident*, or describe how they created a specific artifact–a *critical object*, or describe a routine but extremely recent activity. The resulting stories offer designers maximum insight into the user's experience, including examples not only of what goes right, but, more importantly, what goes wrong. What caused any breakdowns? What workarounds did the user try and were those workarounds successful? Finally, did the user create a novel solution or *user innovation*? The latter, while relatively rare, are worth seeking out, since one user's novel perspective can lead to an inexpensive, field-tested new market for the system.

Tutorial interviews result in descriptions of how the system is supposed to work, when everything goes according to plan. While they are sometimes useful, especially when you

are unfamiliar with the field or the details of the current system, they can also be misleading, because they make it difficult to uncover the problems that occur in real-world contexts. Be careful, because they are often the result of a story interview gone wrong. You can tell if a story interview has turned into a tutorial interview if, rather than describing a specific activity, the user invents a description of how that activity is supposed to be performed. If you hear phrases such as: "I usually do this" or "Every time this happens, I do that", the user has shifted from telling a story to treating you as a student and teaching you an idealized version of what should happen. Tutorials are useful when you need to understand what 'normal' behavior looks like, but offer few insights for design. Although they are the most common, they often encourage designing for stereotypical rather than real behavior. If you do decide to conduct a tutorial interview, ask for at least one specific story first. The user will remember more details and provide a better-grounded overview of the idealized activity.

Opinion interviews are the most common but least useful for design. This is counterintuitive, since both users and designers are all familiar with marketing surveys and assume that opinions are what is expected. While it is true that users' opinions can help to identify features that work well or 'pain points' that must be addressed, opinion intervews offer few clues as to the underlying causes, nor do they provide inspiration for design. Worse, users generate these opinions on the spot, making them less grounded in the details of the user's actual experience. If you want to collect better opinions from the user, ask for them at the *end* of the interview, after you have first obtained specific stories and tutorial examples. Remember that opinions that arise naturally as the user talks about what actually happened will be far better grounded in their real experience, unlike opinions offered 'off the top of the user's head'.

Learning how to conduct story interviews is the most important method in this book. Story interviews take a bit of practice, but once you learn how to direct your interview questions back to capturing more details of the user's story, you will find the resulting interviews offer a wealth of design ideas. Unfortunately, since you are surrounded by other types of interviews, in the beginning, you may have difficulty recognizing that the interview is going poorly. However, you will know that the interview went well if you discover new insights about the user that contribute to your design.

In summary, if you master the story interview technique, users will tell you stories about recent, memorable interactions with technology. You will be able to draw from these to gain new insights that affect your design. While sometimes useful, tutorials and opinions may contribute, your designs will be much better if they are influenced by real stories of what actually happened, rather than general stereotypes of what usually happens.

How to prepare (before)

Choose a topic, the more specific the better. Clearly, if the design brief is to improve an existing system, you should focus on users of the current system, as well as users of any competing systems. If you are trying to create an entirely new system, you still need to have an idea of what people do today, without your system, so try to identify people who perform related activities.

CHOOSE TOPIC
Team: pair
Level: intermediate
Resources: none
Supplies: notes, log
Access: users

Recruiting: Finding people to interview can be tricky if you do not have easy access to your potential target population. This is especially true if your system is designed 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 someone who is a member of that target audience who can introduce you to others

(and perhaps give you a tutorial interview.) You can take also advantage of relevant social networks or special interest mailing lists. Ideally, go where these users gather, such as at a conference, meeting, or where they live.

Be sure to practice your interviewing skills first, so you get the most out of each interview. Interviews can provide extremely useful information, if done well, but they take a great deal of time and you should avoid 'wasting' interview subjects.

Ethics and Informed Consent: Before you interview a user, you must explain the purpose of the interview and how you will handle their data. This is not simply a matter of getting them to sign a legal consent form. You are morally obligated to ensure that the user understands the consequences of being interviewed and still agrees.

If your university or company has an existing Institutional Review Board, follow their procedures. But even if you work for a small startup or are taking a class, you should follow basic ethical guidelines, and always be very careful when handling interview data. You must consider both the risk of harm to the user and the likelihood of that risk. For example, if your interview might reveal embarassing information or raise legal issues for them, you must let them know in advance and always give them the option of dropping out at any time.

If the risk of harm is small, but the data is easily accessible, you should still restrict data access, by anonymizing the data and keeping it separate from personal identifiers before you share it with anyone. You should assign a unique participant identifier to each person you interview, and use that identifier when you analyze your data. To meet European data requirements, we keep a single record, paper not electronic, that links users' names and personally identifiable data with the participant ID. For example, Marie Dupont might be assigned 'P07' as her participant ID. Her name should be paired with this ID once, on the project sheet, and all the remaining data should be labeled P07. The summary sheet should be kept in a locked cabinet, separate from the other data, with limited access. Ensure that there is no online method of linking her name with her data. See the Ethics chapter for more details.

Interview sheet: For each interview, prepare a 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…”. You will not be able to anticipate all the possible branches of a question–you will have use follow-up questions to probe for more detail. If possible, take notes during in the interview, either the interview sheet to take notes during the interview.

A single story interview may contain different types of questions that elicit different results: stories, tutorials and opinions. However, always capture stories first, then tutorials and then opinions, in that order! If you let the user start with a generalized description of an activity, or worse, ask for their general opinions about an existing system, you will find it very difficult to bring them back to telling specific stories. Worse, their tutorials and opinions will be poorly grounded and perhaps invented. If you keep urging the user to get to the specifics of a specific, ideally recent, account of what they did, it will be relatively easy for them to later summarize everything in tutorial form. Most users spontaneously give their opinions during the course of the interview, but if not, you can ask directed questions at the end. User’s opinions that arise as they describe a real, specific activity will be better grounded in their actual experience, and are more likely to truly reflect what they believe.

Story questions elicit detailed stories, ideally including breakdowns, workarounds, successful activities and user innovations. The key is to obtain a step-by-step account of what happened, with as much detail about the interaction as possible.

Figure 1. Story Interview: Interview in pairs so that one person can concentrate on interviewing and formulating questions while the other person takes notes.

What to do (during)

ASK FOR USER'S STORIES
Duration: 15-60 minutes
Skills: video
Roles: interviewer, scribe
Contributors: designers, users
Audience: interviewer, scribe

First introduce yourself, thank them for talking to you, and explain the purpose of the interview. While it is tempting to begin with general questions to 'break the ice', this usually sets up the wrong tone for the interview, and encourages the interviewee to answer in general terms. Explain that you are interested in collecting specific anecdotes or stories of what actually happens, and that the details are interesting to you. Students often feel more comfortable if they say that the interview is for a class, with a specific interviewing technique. Begin with a question that encourages the user to explain a specific 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 are easy to remember as well. You can also ask the interviewee to 'walk you through' their current or very recent experience in creating or working with a particular object. Finally, you can ask about what happened, step-by-step, earlier today or yesterday, even if it was a typical experience.

Begin with the questions you planned, and probe more deeply if you get 'yes/no' or very short answers. Try to get the interviewee to tell you a story of what actually happened and ask additional questions that gather more detail, especially any details that are relevant to design. Record the sequence: question/answer/ question/answer. Do not forget to ask about related stories, both typical and unusual, that contrast with the initial story.

As you talk, be sure to pause to give the interviewee time to answer. Avoid jumping in with a new question immediately after they give an initial response. Give the person time to reflect and answer. Pauses encourage people to talk more, so use them. Also, as they speak, 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!

Assume that your questions will evolve in response to the interviewee's answers and that new topics will come up. Consider which background information to collect, such as the

user's level of computer experience or length of time in the job. Think of additional questions (Who, What, Where, Why and How) as they answer, to obtain more detail about how the interviewee uses the system to support their work. Successful interviews should include descriptions of both normal and unusual uses of the system. Remember, your goal is to obtain concrete, specific examples and generalize from there. Try to envision the user setting and make some predictions about possible uses.

Take notes, ideally on paper, as you go. However, be careful, since note-taking can distract you from listening to the person. If you have to 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 is very distracting for some people and usually requires more of your attention than hand writing.

What to produce (after)

STORIES
Format: story

Capture the most interesting 'interaction snippets' from your interviews. 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? Then sketch the interaction in the upper boxes and add a text description below. (Interaction snippets are short, but may include more than three elements.) Focus on surprises, especially breakdowns, workarounds, and user innovations. Try to extract 3-5 interaction snippets from each story. (Clearly, longer stories may have many more interaction snippets.)

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

WHAT TO DO?

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?