INTERVIEW with Gitta Salomon

Gitta Salomon is a consultant interaction designer. She founded Swim Interaction Design Studio (swimstudio.com) in 1996 as a consultancy company to assist clients with the design of interactive products. Over the years, her clients have included established and start-up companies, developing web-based and other interactive products and services. These companies contact Swim with new product initiatives or when a partially-developed product needs interaction design work. Swim has consulted for a range of clients, including Jeppesen, GoPro, Nike and Gap, Inc.

YR: What is your approach to interaction design?
GS: I've devised my own definition: interaction design is the design of products that reveal themselves over time. Users don't necessarily see all the functionality in interactive products when they first look at them. For example, the first screen you see on a cell phone doesn't show you everything you can do with it. As you use it, additional functionality is revealed to you. Same thing with a web-based application or a desktop application—as you use them you find yourself in different states and suddenly you can do different things. This idea of revealing functions over time is possible because there is a microprocessor behind the product and usually there is also a dynamic display. I believe this definition characterizes the kind of products we work on—which is a very wide range, not just web products.

YR: How would you say interaction design has changed in the years since you started Swim?
GS: I don't think what we do has changed fundamentally, but the time frame for product development is much shorter. And more people believe they need interaction design assistance. That has definitely changed. There are more people who don't necessarily know how to go about doing interaction design, but they know they need to make their products usable and appealing.

YR: So what were the kinds of projects you were working on when you first started Swim?
GS: They were less web-centric. There was more software application design and a few hardware/software type things. Between 2001 and 2004, the focus shifted to almost exclusively web-based applications. However, these were quite similar to software applications—they just have different implementation constraints. At this point, we are seeing a lot of interest in migrating to small platforms (and off of the desktop)—for example, mobile phones

and tablets. Also, hardware/software products are starting to pick up again. The nature of the problems we solve hasn't changed much; it's the platforms and associated constraints that change.

YR: What would you say are the biggest challenges facing yourself and other consultants doing interaction design these days?
GS: One of the biggest challenges is remembering that half of what we do is the design work and the other half is the communication of that design work. The clients almost never bridge the gap for us: we need to bridge it. We always have to figure out how to deliver the work so it is going to have impact. We are the ones who need to ensure that the client is going to understand the deliverable and know what to do with it. That part of the work is oftentimes the most difficult. It means we've got to figure out what is going on internally with the client and decide how what we deliver will be effective. In some cases you just start seeing there is no place to engage with the client. And I think that is a very difficult problem. Most people right now don't have a product development process. They are just going for it. And we have to figure out how to fit into what is best described as a moving train.

YR: And what do you use when you try to communicate with them? Is it a combination of talking, meetings, and reports?
GS: We do a number of different things. Usually we will give them a written document, such as a report or a critique of their product. Sometimes we will give them interactive prototypes using any variety of tools—Acrobat, Keynote, Flash, HTML—to provide them with simulations of what the product experience would feel like. In the written materials, I often create a taxonomy, we name the things that we all need to be talking about. Then at least we all have a common language to discuss things. It is a measure of our success if they start using the words that we gave them, because then we truly have influenced their thinking. A lot of times we'll give them a diagram of what their system is like, because nobody has ever visualized it. We serve as the visualizers, taking a random assortment of vaguely defined concepts and giving shape to them. We'll make an artifact, which allows them to say "Yes, it is like that" or "No, it's not like that, it's like this...." Without something to point to they couldn't even say to each other "No, that is not what I mean" because they didn't know if they were talking about the same thing. Many times we'll use schematic diagrams to represent system behavior. Once they have these diagrams then they can say "Oh no, we need all this other stuff in there, we forgot to tell you." It's often the case that nobody is writing complete lists of functionality, requirements, specifications, or complete documentation. This means the product ideas stay in somebody's head until we make them tangible through visualization. Recently, we've had to create more visually-oriented specifications to support offshore teams that aren't fluent in English, Reducing our designs to many diagrams and few words is a tough problem and requires more communication time (vs. design time) than we've encountered before.

YR: So this communication process is just as important as the ideas?
GS: I think it is, a lot of the time.

YR: So, how do you start with a client?
GS: For clients who already have something built, I find the best way for us to get

started is to begin with a comprehensive demo of their product. Depending on the size of the problem and nature of the target audience, we typically spend one day to a week collecting information. Besides the demo, we usually ask our clients to tell us about their target market, competitors, long range goals, etc. Over a longer period of time, we use the product and observe other people using it to get a much broader picture. Because the client's own vision of their product is so narrow, we have to step back from what they initially tell us.

YR: So do you write notes, and then try and put it together afterwards, or—what?
GS: We use all kinds of things. We use notes and video, and we sit around with paper, pencil and pen. When reviewing the materials, I often try to find the overarching themes. It's often mind-boggling to bring a software product that's been thrown together into any kind of coherent framework. It's easy to write a shopping list of observations, but we need to assemble a larger structure and framework, an overall point of view, and that takes time to construct. We need to reflect and stew on what was done and what, maybe, should have been done. We need to highlight the issues and put them into some kind of larger order. If one always operates at a low level of detail—like worrying and critiquing the size of a button one ends up solving only local issues. You can't get to the big interaction problems that way; you don't end up addressing the issues that should have been solved first. We often find our clients have spent the bulk of their time at that low level of detail.

YR: If you're given a prototype or product to evaluate and you discover that it is really bad, what do you do?
GS: Well, I almost never have the guts to go in and say something is fundamentally flawed. And that's not the best strategy anyway, because it's your word against theirs. Instead, I think we always need to make the case for why something is wrong or flawed, to provide a logical argument the client can wrap their heads around. Sometimes I think we are like lawyers. We have to assemble the case for what's wrong with the product. We have to make a convincing argument, one that allows the client to get past their biases and grasp where and why things have gone wrong. ■