BusinessWeek

Design May 18, 2007, 11:04AM EST text size: TT

# Why Products Fail

## Author and Microsoft researcher Bill Buxton proposes a solution to the tech industry's software development problem

by Jessie Scanlon

*Sketching User Experience* is, nominally, a book about product design. But it would be just as accurate to say that it's a book about software development, or, more generally, about the often broken process of bringing new products to market, with examples ranging from the iPod to an orange juicer.

"Hardly a day goes by that we don't see an announcement for some new product or technology that is going to make our lives easier, solve some or all of our problems, or simply make the world a better place," writes Bill Buxton in the preface to his book. "Few of these products survive, much less deliver on their typically over-hyped promise." Why, Buxton asks, are we not learning from these expensive mistakes? Why are we not fundamentally rethinking the process of product development?

*Sketching User Experience* (Morgan Kaufmann, 2007) is a book born of the frustration of an industry insider. Principal researcher at Microsoft Research (MSFT) since late 2005, Buxton has spent decades in the trenches of computer science—including a stint at the storied Xerox PARC (XRX)—and at the front lines of the software industry.

### Not Just Improved, New

It was after eight and a half years as chief scientist at Alias Wavefront (now owned by Autodesk (ADSK)) that Buxton threw up his hands. During that time, he reveals over lunch in Boston, the company introduced only two products of which he's genuinely proud—an interactive digital corkboard called the Portfolio Wall and Alias Sketchbook. Yet he came to realize, he says, that the problem wasn't with Alias Wavefront—the whole software industry was broken.

While companies were very good at what Buxton calls "N+1" development, or pumping out improved versions of existing products, most were quite bad at developing the new products that are essential for sustainable long-term growth. He points to Adobe (ADBE), one of the largest software companies in the world, as one example of the industry's modus operandi: most new products are the result of mergers or acquisitions.

Buxton left Alias Wavefront in 2002 and spent the following years thinking about the problem with traditional software—and more broadly product—development, and how to fix it. *Sketching User Experience* is the result of that effort. It's a book written primarily for designers, but one that could and should be read by any engineers and executives who share Buxton's desire for better and more successful products.

## Holistic Approach

And while Buxton's insights are geared towards companies making software or products that depend on software, the book is jargon-free, and most anyone in the business of creating products will learn from it.

"My belief is that one of the most significant reasons for the failure of organizations to develop new software products in-house is the absence of anything that a design professional would recognize as an explicit design process," writes Buxton.

The aim of his book is to help change this situation. He argues for a holistic approach to experience-based design, showing how the weakness of software product development can be complemented by the strengths of traditional product design and vice versa. But mostly, he argues strongly for an explicit and distinct design process that's integrated into the larger organization and supported by executive leadership.

## New Levels of Complexity

He illustrates the importance of executive support with a little-mentioned fact about Apple (AAPL): The design team that created the iPod, the iMac, and all of the other products that punctuated Apple's rise, is the same team that created all of the products released during the company's long slide from Sculley through Amelio. The difference between before and after is a chief executive officer who made design an integral part of the product-development process.

For Buxton, the need to rethink the development process by inserting design into the front-end is all the more urgent because new technology, such as embedded processors and wireless networking, introduce new levels of complexity to the challenge of product design. He argues that designers are uniquely trained to focus on the human side of product development, to consider the behaviors and experiences associated with or enabled by these new technologies. "To design a tool, we must understand the larger physical, social, and psychological context in which it will be used. And that's something designers are trained to do," he says.

This focus on "what designers do" rather than on the murkier question of "what design is," is one of the things that makes *Sketching User Experience* worthwhile for non-designers. Buxton offers his straightforward analysis of the design process, in concrete language and with plenty of real-world examples.

## From Sketch to Prototype

This approach leads him to focus on the archetypal activity of design: sketching. Sketches, he argues, are quick, inexpensive, disposable, plentiful, offer minimal detail, and suggest and explore rather than confirm. (It should be noted that he doesn't limit "sketches" to pen on paper—a sketch might be digital or three-dimensional.) The value of sketching is less in the artifacts themselves than in the cognitive process of working through dozens of ideas, of considering as many options as possible, and allowing each option to raise new questions.

Buxton takes pains to distinguish sketches from prototypes, which are more detailed, more expensive, and more focused on testing or proving a single idea. If sketching is about asking questions, prototyping is about suggesting answers. Sketching takes place at the beginning of the development process, prototyping only later.

For an executive more comfortable with hard data, the value of these scraps of drawings, these glorified doodles, might not be obvious. But Buxton makes a case that could easily be expressed in a spreadsheet. Sketching is less expensive than prototyping, and far less expensive than trying to fix problems late in the development cycle.

## Into the Lion's Den

Does that mean that companies shouldn't invest in prototypes? Of course not. But it does suggest that investing more money and resources up front to allow a small team of designers adequate time for product ideas will save significantly higher costs of trying to correct problems later in the game, when the production team has swelled to include as many as 50 engineers and marketers, and delays can cost millions. In fact, it was the delay of a software project by two years that broke Buxton's spirit at Alias.

With the book finished, he has rejoined the industry at a company that's hardly known for good design. "As I was working on the book, I figured the ultimate test of my ideas would be to give them as talks at Microsoft," he says. "So I walked into the lion's den and ranted."

The good news for the company, as well as its millions of customers, is that the lion didn't rip him to shreds. Instead, it gave him a home. Based in Toronto, he spends one week a month in Redmond—where he says he spends more time with the product teams than doing research—or visiting one of Microsoft's labs in Silicon Valley; Beijing; Cambridge, Britain; or Bangalore, India, where he weighs in on various projects.

"A lot of my friends certainly raised an eyebrow when I took the job," he admits. "But part of part of my job is to put design on equal footing with technology. If I'm successful, the impact will be phenomenal."

Jessie Scanlon is the senior writer for Innovation & Design on BusinessWeek.com.