



Mark A. Hart, Visions Launch Editor

Launch Pad

In developing and launching a software product, try an *underdo* strategy

Mark A. Hart, Visions Launch Editor, NPDP, President, OpLaunch

When creating new software products, sometimes it seems almost impossible to compete. For instance, in the project management software category, how can anyone compete with Microsoft Project—the entrenched leader. And, if the company creates a competitive product, how can they launch it for maximum impact? Visions Launch Editor Mark Hart shares insights from a team that employs a unique commercialization philosophy.

It is tempting to plan a product launch by mimicking the techniques of successful companies. Will 20th century tactics produce the best results? Probably not. Chris Anderson discusses the future of business in the book, *The Long Tail*¹. He notes that in the 21st century, two important changes are occurring that impact Product Development and commercialization. One is the democratization of tools. Globally, more people have access to the tools required to produce new products and services. The result is the creation of more products. Second comes the explosion of information. Today a confusing matrix of communication channels, such as the Internet-enabled social networks, facilitate the distribution of information from product producers, reviewers, and consumers. This drives development decisions. For example, online communities and Internet search engine results can have a larger impact on product positioning and branding of a product than advertising.

In markets where there are more products and more information, which products become successful? According to Jason Fried, the CEO of 37signals, a software company, sometimes the answer is *underdo*—to provide fewer product features than the market leader.

Necessity sparks innovation

In late 2003, the team at 37signals, which was at the time a design company, needed better project management tools for their small, geographically dispersed team. They didn't need a tool that had more features than Microsoft Project or one that produced a greater variety of charts and graphs. They were too small for an enterprise-sized solution. They were reluctant to dedicate an administrator to manage specialized desktop software running on one computer. They didn't want to spend time updating HTML

pages on a private extranet. Relying on unstructured project information in various e-mail accounts was not efficient, so they created their own tool. The tool's focus was to facilitate communication between their team members and clients. The tool had to accommodate the chaos of real projects. The tool they created not only solved their problem but it was appealing to many of their clients.

In February 2004, 37signals launched *Basecamp*, their project collaboration, Web-hosted software after approximately four months of design and development. Besides the software, two other things emerged from

this effort: their "Getting Real" philosophy and the toolset used to build Web-based applications.

The 37signals method for building better software is outlined in their book, *Getting Real*². 37signals (www.37signals.com) started as a design company but morphed into a software company that produces Web-based software products and an open source Web application framework, *Ruby on Rails*. Two of their products, *Basecamp* and *Backpack*, were awarded Best of the Web 2005 by the editors of *Business Week* magazine.

Development philosophy overview

The Getting Real philosophy recommends small teams of experienced, enthusiastic, multidisciplinary designers and developers. Team members that have effective, concise writing skills are preferred. To encourage effective communication, teams are intentionally small. The work environment minimizes interruptions.

To build a Web application, 37signals advocates minimizing items that represent the product, (such as charts, schematics, wireframes, and functional specifications) and actually building the real thing. They begin by creating the interface in HTML because this is the environment for the actual screens that customers will use. They demand more than a target user's approval of a static screens. They evaluate how the product is used to solve the target problem.

In terms of functionality, 37signals advocates *underdo*—offering fewer features than the competitive products. They strive to deliver what the majority of customers must have and eliminate anything else. The price of a product developed by this method is lower because the development expenses are lower. When the appropriate reduced feature set is targeted, a small company with fewer resources has a better chance of competing successfully against better-resourced, larger competitors. A lower-priced product has a larger potential number of users.

Having working prototypes very early in the development process is a major advantage to this method. This provides more opportunities for iteration with the target audience. Once they achieve the appropriate user experience using the prototypes, they build the rest of the software backwards from the interface.

37signals does not have a marketing, sales, or technical support department. This ensures that the development team is immersed in all aspects of the commercialization.

Implications for launch

The 37signals team can complete and test a lightweight, agile application quickly. They can react to changes in the market. The time-to-market for their products will be the lowest it can be.

37signals believes that the best promotional tool is a great product. A

“Which products become successful? ... sometimes the answer is *underdo*—to provide fewer product features than the market leader.”

great product conforms to the customer's preferences and workflows. For the target customer, the product must be easy to use.

They advocate a Hollywood-style launch—teaser, then preview, and then launch to generate buzz and anticipation. In addition, their Web site provides product tours, screen captures, and videos that show the practical use of the product in multiple scenarios. To attract new customers, 37signals offers free versions of all of their products. The products are made available in the “software as a service” model and can be used on any computer with a browser and Internet connection. They have an easy signup and cancellation policy for subscribers.

They don't purchase advertising or use traditional PR methods. 37signals cites blogging as their most effective promotional tool. They ensure that

their product blog contains help advice, tips, tricks, and links.

Since their applications are Web-hosted, applications can be updated frequently and customers are always using the latest version of the software. Because blogs facilitate two-way communication, the development teams quickly learn about customer sentiment and have the chance to address

problems rapidly. Because the applications are lean, changes can be made and tested in a few days.

“The price of a product developed by this method is lower because the development expenses are lower.”

The result of the right strategies

Today, over 500,000 people and small businesses use *Basecamp*. This group includes freelance software developers, advertising agencies, architects, authors, teachers, and musicians. These are customers who otherwise would be using mashup methods such as scraps of paper plus filing cabinets, spreadsheets plus e-mail, or Microsoft Project plus shared file directories. 37signals can produce great software faster and more efficiently than most of their competitors.

37signals doesn't use traditional advertising or PR to persuade potential customers that they have a great product. By using an interconnected marketplace to identify the best solutions from a multitude of alternatives, they rely on the Voice of the Marketplace to drive both sales and product development.

Mark A. Hart, Visions Launch Editor, NPDP, President, OpLaunch (mark_hart@oplunch.com)

Endnotes

- 1 Anderson, Chris, *The Long Tail: Why the Future of Business Is Selling Less of More*. New York, Hyperion: 2006.
- 2 37signals, *Getting Real: The Smarter, Faster, Easier Way to Build a Successful Web Application*, self published and available at www.37signals.com, 2006.

Exhibit 1: Software Development and Commercialization Comparisons

	<i>Traditional software development</i>	<i>37signals software development</i>
Development	Develop the majority of a complex product, based on predictions about the market. Announce product through press releases. Concurrently finish development, finalize interface design, integrate components, build installation routine, and test to generate the release to manufacturing candidate.	Develop interface very early in development. Iterate until user experience goals are realized with target customers. Develop support code. Perfect core features. Provide new, working versions of software online for recurring evaluation by target customers.
Pre-Launch	Marketing, PR, and sales staff plan a big budget introduction event at a large venue. Promote the number of features in your product. Buy print advertisements. Submit informative articles to trade journals.	No separate marketing, PR, or sales staff. Low budget, online introduction. No paid advertising. Emphasize the utility of the product. Promote upgrade opportunities from free to subscription versions on the Web site and within the applications.
At-Launch	Invite influential customers and media representatives to the introduction event and wait for the reviews to be written.	Generate online buzz and anticipation with teaser, preview, and launch method. Strive to maximize the links to product sites. Provide “golden tickets” for early availability of product.
Post-Launch	Collect leads from trade shows attendees and evaluate early sales data. Manage customer relationships using separate organizations and data mining. Release new versions periodically.	Team members participate in blog communications regarding product. Monitor customer reactions and online sales trends. Adapt very quickly.

Source: The author