Public Beta Tests as a Launch Tool

a Centercode whitepaper



INTRODUCTION

Public beta tests have become a staple of many software launches, including websites, online games, and both desktop and mobile apps. In this whitepaper we'll be providing an in-depth look at the risks, goals, and best practices that need to be considered when defining your public beta strategy.

Before we begin, it's important to first define the two primary forms of beta testing: public and private.

Public Beta Tests

Public betas (also called **open betas**) are a form of beta test in which a product is made available to the general public without the need to be invited or selected in order to participate. Public beta tests typically include anywhere from thousands to millions of testers and include little to no structured feedback loop. They often focus on data analytics (stress testing, feature usage, etc.) over actual customer feedback.

Public betas may have different owners based on their goals. If the focus is on awareness, the project is likely owned by marketing. If the project is focused on providing early adopters access to upcoming fixes, the project is likely owned by product management, engineering. or even support. Given their size and need for scalability, public betas are only conducted with software-based products (desktop, web, mobile, and gaming).

Private Beta Tests

Private betas (also called **closed betas**) are a function of product management and/or quality, intended to improve the product through feedback from a selected group of individuals who closely match the target market of the product. These are typically much smaller (25 to 500 testers), confidential, and focus heavily on finding and squashing bugs. Private beta tests are an industry standard phase of both hardware and software development that originated within IBM in the 1960s, and are currently used globally with technology products of every size and shape.

PRIVATE BETA RESOURCES

We also offer a number of high quality resources and templates to help improve your private beta test phases. These can be downloaded at no cost via www.centercode.com/library.

THE HISTORY OF PUBLIC BETA TESTS

While marketing-focused public beta tests have long been a staple of massively multiplayer online (MMO) games and website applications, Google was the first company to widely introduce the concept with Gmail and other web apps nearly ten years ago. In the specific case of Gmail, Google used a creative viral strategy to initially draw in the public via invitations that grew in availability with each new member.

The result was a product that spread extremely quickly and gained an enormous audience before it was ever officially released. Gmail had more than 100 million "beta users," making it the largest public beta test of all time with a massive five-year (2004 to 2009) public beta period.

This strategy accomplished two important goals. First, it built a sense of exclusivity around being invited to use a product that wasn't yet available to the general public. Second, it provided an extended excuse for any customer-facing bugs because "it's a beta!" Accomplishing these two objectives while millions of people were actively using (and loving) the product was simply marketing genius.



WHAT HAPPENED TO GOOGLE BETAS?

Public betas were a core expansion tactic when Google was primarily focused on providing free (ad-supported) products to consumers. Eventually Google expanded Gmail and many of the other Google apps (Google Docs, Calendar, Drive, etc.) from focusing on free products for consumers to commercial products for businesses. Companies are much less likely to rely on (and pay for) products which are deemed "beta", thus Google strategically removed the label.

Read the whole story:

http://googleblog.blogspot.com/2009/07/google-apps-is-out-of-beta-yes-really.html

THE RISKS OF PUBLIC BETA TESTS

While the public beta strategy worked well for Google, it had the side effect of changing the public perception of what beta testing means; many had no idea what a beta was prior to Google's logo-tagged use of the term. Given how well the Google "beta" products performed in live use (all of which went through rigorous QA and private beta tests prior to reaching their public beta status), the pre-production and likely unreliable "beta state" of the product lost its meaning. The public came to read beta as "early," but still very stable and usable.

While this seems reasonable, since you clearly shouldn't hand your product to thousands of your prospective customers if it's not in a highly-usable state, it can also backfire.

Richard Garriott, the creator of the extremely successful and long-running Ultima video game series experienced this firsthand prior to releasing his MMORPG, *Tabula Rasa*. He found that by releasing true traditional beta (early/unfinished) builds under the banner of a public beta, a substantial portion of his audience formed an ugly impression of the product, which ultimately hurt the product's release. In his words from an IGN interview[1]:

> *"We burned out some quantity of our beta-testers when the game wasn't yet fun. As we've begun to sell the game, the people who hadn't participated in the beta became our fast early-adopters. And the people who did participate in the beta, we've had to go back to and say 'look, look, we promise: we know it wasn't fun two months ago, but we fixed all that. Really, come try it again.' We've had to go out and develop free programs to invite those people back for free before they go buy it. So the beta process, which we used to think of as a QA process, is really a marketing process."*

Another potential risk of a public beta test is overwhelming your team with feedback and support requests. While a much smaller percentile of testers will provide feedback in a public beta compared to a private beta, you could be receiving hundreds of repetitive bug reports and thousands requests for support if the public beta is large enough. Depending on your team's size and preparation, this could bog down your entire staff and risk delaying the product's launch.

In a nutshell, if your product and your team aren't ready for the public spotlight, then a public beta test may do more harm than good by hurting the product's reputation before it's even out the door. The best way to mitigate these risks is to put some time into planning out the public beta test so you can ensure that you have the right resources dedicated to meeting your goals. Only then will your team be able to make the most of the public exposure.

1 http://www.1up.com/news/richard-garriott-beta-testing-hurt

THE GOALS OF PUBLIC BETAS

Planning a public beta test starts with setting some keys goals. Like private beta tests, public betas can be designed to achieve a wide variety of objectives. Most public beta tests aim to achieve one or more of the following goals:

Launch Awareness	One of the most well-known forms of public betas, these projects focus on raising awareness of a product prior to its release. This is most commonly used by companies launching video games and websites.
Beta In Name Only (BINO)	This type of public beta uses the term "beta," but the product is effectively feature complete. In this case, the beta tag serves to both excuse minor issues and promote a sense of "getting in on the ground floor."
Viral Release / Buzz	Popularized by Google, this type of beta has become prominent with web applications. The general idea is that the product is available by way of a viral invite system prior to an official release.
Early Adopter Access	In this scenario, a release is made available to early adopters (most often technical individuals) on an opt-in basis in order to receive access to early fixes for issues they may be having, with an understanding that other issues may be present. While these are beta builds, they're generally not considered beta tests as they don't solicit feedback.
Data Collection	Some public beta tests function as a tool to collect back-end data that can be used to analyze customer behavior, which will in turn influence the design or launch of the product.
Load/Stress Testing	Some public betas exist in order to test the threshold or scalability of an infrastructure. Given this requirement, these beta tests are carefully coordinated to execute over a short window of time.

These objectives are merely a starting point. If there are additional goals you want to achieve, be sure to think about what they are, what data you need to collect in order to achieve them, and how you aim to collect that data. You may also want to consider whether you have goals that rely heavily on receiving detailed feedback from testers. If that's the case, your goals may be better served by a private beta test.

PUBLIC BETAS BY PRODUCT TYPE

Different types of software products utilize public betas in unique ways. The following chart describes some of trends as they relate to different product types.

Websites	New websites commonly use the "beta" tag within or adjacent to their logo as an indication of an early or limited release, despite the product being publicly available. In many cases the site is a "Beta In Name Only," being functionally complete and customer-ready. In other situations the site may truly be in its early stages, and utilizing Agile and/or MVP (minimum viable product) development and release strategies.
Online Games	Massively multiplayer online (MMO) games commonly employ public betas in the form of launch awareness campaigns, partnering with sites like IGN to distribute beta keys to large audiences. In some cases, these public betas are available for unlimited access, while others are limited to scheduled windows (e.g. "weekend betas"). Games also provide access to public betas as a perk for pre-ordering the final product, effectively using the public beta as both a sales tool and a soft launch.
Hardware Drivers	It's fairly common for new iterations of hardware drivers to be released sporadically throughout the lifecycle of a hardware product. New drivers have the possibility of introducing unforeseen issues which require the rapid release of new beta builds. These fixes may not address problems which are widespread, and therefore only apply to a small portion of the product's audience. In these cases, beta drivers are often released early and aimed at the users experiencing specific issues.
Desktop Software	Early public beta releases of upcoming desktop software build buzz, attract enthusiastic early adopters as ambassadors, and gain press coverage.
Mobile Apps	Mobile app public betas are generally focused on load testing the online components of the app. Often these tests begin by limiting the testers to a designated region or country (e.g. Canada) in order to test their infrastructure prior to a global release.

COMPARISON: PUBLIC AND PRIVATE TESTS

As we mentioned earlier, public and private beta tests require very different strategies. This table outlines the key attributes that differentiate public (open) and private (closed) beta tests, so you know your options when you're planning your beta program.

	Private Beta	Public Beta
Owner	Product Management or Quality	Marketing
Access	Invite by company	Open or invite by other testers
Participants	Dozens to hundreds	Thousands to millions
Goals	Bugs, performance, accuracy, acceptance	Marketing, load testing, data collection
Feedback Loop	Constant engagement	Minimal (forums/email) or none
Feedback Type	Bugs, features, suggestions, tasks, surveys, forums	Bugs, forums, surveys, email, or none
Product Type	All technology products	Online games, websites, some apps
Also Known As	Field trial, UAT, CAT, customer validation, prerelease	Open beta, marketing beta, stress test, load test
Tester Qualifications	Target demographic specific to product	Limited or no requirements
Duration	Fixed time (weeks to months)	Open length of time until release
Incentives	Project incentives for tester engagement	None
Product Security	NDA, test agreement, license agreement	License only
Recruitment	Staged (recruit, select, NDA, test)	Flat (recruit direct to access)
Distribution	Pre-release system or production-ready	Production-ready usually required
Exposure	Secret and undisclosed	Public and promoted

BUILDING A PUBLIC BETA STRATEGY

With careful planning, a public beta can serve as a very successful marketing campaign. The following are key best practices to consider when planning and executing your public beta test.

1. Begin seeding your pool early

It's never too early to start building a list of people who are interested in your product. As soon as it's announced, consider putting up a landing page that collects the email addresses of people who'd like know more about your product and it's future availability. There are companies and inexpensive tools available that specialize in these types of services, such as LaunchRock, Launch Effect, and LaunchGator. Alternatively, you can make this feature a part of the announcement itself. Either way, when you're ready to launch your beta, you already have a group of enthusiastic early adopters that you can leverage to get your beta started quickly.

2. Run your private beta test in parallel

While public betas are executed to increase awareness, companies will often run private beta tests at the same time. This enables the company to work with a small, focused group of testers to accomplish more traditional beta goals. As you receive feedback or impressions from the public beta users, you can take those issues to the private beta testers for a more in-depth understanding of the problem or use case. This allows you to get the benefits of both public and private beta tests.

3. Make sure the product is ready for mass market usage

Given the shift in perception of public betas, you can no longer rely on the beta moniker to protect your product or brand from giving off a negative impression before it's commercially available. While private betas are designed to limit the risk of repelling your market, most public betas are designed specifically to provide very wide exposure for your new product. With this in mind, it's essential that your product is as stable and as near to release quality as possible. This means that the product is mature, having already completed a full QA cycle, alpha test phase, at least one phase of customer-focused private beta, and has undergone load testing tools prior to launching in public beta.

4. Have a launch and promotion plan

Since a common goal of most public betas is to get as many people as possible to use your product, announcing the beta correctly is critical. Therefore, a public beta launch should be approached just like any other marketing activity. It needs to involve working with your public relations (PR) resources to get as much coverage as possible from relevant websites and blogs during your beta launch. You can also leverage partnerships or customer communication channels to announce the public beta opportunity. However you decide to reach out to your target market, make sure it's planned well in advance so that your announcement gets the most coverage and testers possible.

PROMOTE EARLY EXCLUSIVITY TO GROW YOUR AUDIENCE

Unless your product is heavily anticipated by an established core audience, chances are that just opening the gates to your product won't automatically result in a firehose of users. In these cases, it can pay to utilize the exclusive nature of viral invites (similar to Google's aforementioned strategies) to encourage growth from your initial early adopters. Leveraged correctly, a few hundred users can turn into thousands and tens of thousands.

5. Release in stages

It's very likely that you'll only have one free shot to impress your market. If you launch to 200,000 customers and the product breaks after the first 5,000, there's a good chance that a huge portion of your users won't be returning anytime soon. Unless your product is a hot commodity, public beta testers have little to no investment in you, so they're extremely fickle and should be treated with kid gloves. Plan your public beta in stages that ramp up based on previous success. If handle properly, this strategy will also increase anticipation as people excitedly wait for their turn to join the beta.

STRESS TESTING IN PUBLIC BETA

It can seem nearly impossible to predict server load requirements for a public beta test, as proven by nearly every massively multiplayer game beta. It's important to make sure that your infrastructure can handle the thousands of testers you're about to unleash on your product **before** your test begins. We suggest running a stress testing of your infrastructure, beginning with a number far below what you think your system can handle. If you think the system will break at 1,000 users, start with 250. From our experience, most systems break much earlier than expected when first put under realistic load conditions. It's better to include an extra testing phase than it is to frustrate 70% or more of your testers. You will also benefit from extra confidence when your public beta starts, knowing that your system is going to be able to handle all of the excited participants.

6. Include a feedback loop

Regardless of your own ambitions for your product, public beta tests should always include some sort of feedback loop. While individual customer feedback may not be your top priority, the sentiment of your early adopters is extremely valuable and should be taken into consideration. Equally important is the fact that customers will continue to use a product that they feel they have a stake in, and a beta feedback loop provides that stake.

7. Plan your support level

Most public beta versions of software don't include the robust support of a released product. Depending on your goals, you'll want to consider how much support you can offer or would like to have in place for your users. If users have no support or way of letting you know about a bug or issue, they may get frustrated and turn against your product. At the same time, you most likely are not in a position to provide full support at this point in your development. Whatever you decide to do, make sure to communicate it clearly to your testers when they sign up so that they know what to expect out of your test.

8. Use private forums

The tools you use to collect feedback during your public beta can have a lasting effect on your product's reputation. For example, some companies will use public forums to collect opinions from testers during their public beta. The problem with these public forums is that they're (you guessed it) *public*, which means that Google will crawl and catalogue them for years to come. If your testers are all complaining about bugs or features that affected their beta experience, that forum will show up in the Google search results for your product, even if you fixed the problem before launch. As a result, the negative buzz from your public beta could hamper your product indefinitely.

9. Consider load and analytics in context

If your product is online, there's no reason not to collect as much data as possible. That said, the design of your public beta could greatly influence the nature of that data collected. It's important to consider all data in the context of the project. Did the testers have access to the entire product? Did they have enough time to fully experience it? Was their behavior influenced in some way? Were they the correct target market for your product? These factors could help identify trends or skewing in the data that isn't reflective of how your real customers will use the released product.

10. Consider a time bomb

When releasing your product in a public beta, it will be exposed to the public for unknown quantities of users to download and use. If your product fails to expire, you'll effectively lose your chance to sell the product to a large portion of your audience. When you include a time bomb in your product, make sure that you allocate enough time to account for false starts, late starts, or extensions to the planned test period. Depending on your goals, you may also consider releasing a feature-limited version to be tested; just keep in mind that this will result in limited feedback on your product.

CONCLUSION

With the ever expanding and increasingly viral nature of the internet, public betas play an important role in the modern product development cycle. They can be a powerful marketing tool that gives your product a running start. Even though they have very little in common with traditional private beta tests, they still contribute to the same goal: achieving customer validation prior to a product's release. They can also be uniquely challenging due to their sheer size and scope. If you keep your goals uppermost in mind and plan carefully, however, you can get tons of helpful feedback and market buzz during a public beta. Hopefully the ideas and tips in this whitepaper will help you avoid the common pitfalls so that you can make public beta testing a powerful part of your launch strategy.

HOW CENTERCODE CAN HELP

We offer a range of beta testing solutions for technology companies of all sizes.

Beta Management Platform

The Centercode beta test management platform is a complete system for managing beta programs of all sizes, including large public beta tests. It's easily scalable and includes everything you need to recruit users, distribute releases, collect and manage feedback, build powerful reports, and much more. You can schedule a brief online demo of our software and services to learn more.

Managed Beta Tests

Our Managed Betas Team has the experience and resources to run large scale public beta tests. From goal setting to stress testing, we know how to ensure successful public beta tests. We'll even build you a free beta test plan prior to any commitments, where we can discuss your situation in more detail.

Beta Testers

Betabound is our growing panel of more than 90,000 prequalified beta test candidates that are always looking for new products of all types to test. In the past we've used this pool to execute successful stress tests, hardware driver tests, and a variety of other public betas. Best of all, this panel is available as a resource to both Centercode software licensees and our Managed Beta customers.

Free Beta Resources

We're constantly developing and publishing free high quality beta resources in an effort to help improve beta programs everywhere. This includes both our beta test management blog and our growing library of eBooks, whitepapers, templates, and other beta-related resources.