



# What every product manager needs to know about product analytics

The data we glean from product analytics tells us how users actually use the product.



BY SAM TARDIF

## BROWSE TOPICS

- Agile manifesto
- > Scrum
- > Kanban
- > Agile project management
- > **Product Management**
  - Overview
  - Product Roadmaps management
  - > Agile at scale
  - > Software development
  - > Design
  - > The agile advantage
- DevOps
- > Agile Teams
- > Agile tutorials
- > About the Agile Coach
- All articles

As product managers, we take every opportunity we get to learn more about our customers because understanding their needs is critical to building useful products. This means conducting [customer interviews](#), running surveys, and examining in-product analytics. The data we glean from product analytics tells us how users actually use the product – not what they want to do, how they think they're using them, or even how we think they are using them.

Where software development differs, and home building could definitely benefit, is the use of agile methodology. Agile allows multiple teams to respond to changes, quickly. So how can agile, a method based on frequent, continuous delivery exist with long-term, big-picture planning? Is it possible to create a realistic forecast over a long period of time, knowing that the one constant is change?

As a PM, questions like, "How much time do users spend in the every day?" "What actions do they take most?" "Which features get used least?" are incredibly valuable for understanding your users and give us clues as to how to make their experience better. In this post, I'll explain what product analytics are and why you should use them; how to gain a true understanding of your users so you can pay off "empathy debt"; and how to use analytics to help guide new feature development.

Let's get started!

## An intro to product analytics

In order to get a quantitative understanding of what users are doing with your product, the first step is instrumenting it with analytics. The idea is to fire an event for every action that a user can take in your product so you get an aggregated view of how many

## RELATED TUTORIAL

[Create an agile board in Jira Software](#)

[Try this tutorial →](#)

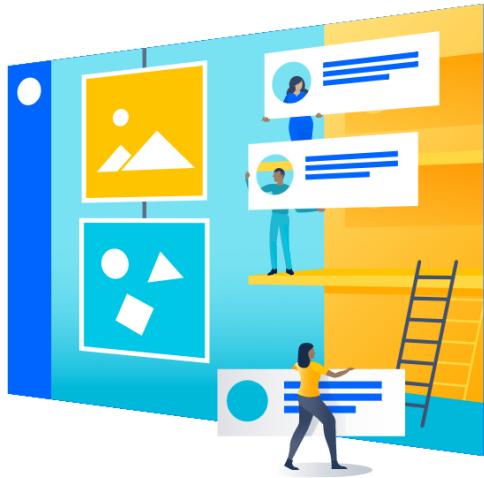
## SUBSCRIBE

[Sign up for more articles](#)

Email

[Subscribe](#)

product so you get an aggregated view of how many users use a feature, and how often they're using it. For example, if you want to track the number of times a user clicks a specific button, you might fire an event called "big-red-button.click." From there you can see which features need work, which are your most important, and use that information to prioritize changes.



PROTIP:

There are a ton of solutions out there that give you a framework for adding analytics events and tracking them. Check out Google Analytics or KISSmetrics as a starting point.

At Atlassian we've tried to make it as easy as possible for everyone to get at data and be able to run their own queries and reports. We use some internally developed tools to provide these services, but tools Google Analytics will get you started too. This has lead to everyone, from developers to PMs to design, asking questions about usage and trying to understand the impact of what we build.

## “Empathy debt”: the newest kind of debt

Lets try out this new term, “empathy debt.”



**Empathy debt: a gauge of whether you understand your users and how they use your product.**

In-product analytics can help you pay off empathy debt in two ways: with qualitative feedback gathered through activities like concept testing and customer interviews; and with quantitative data collected in-product with things like product analytics and NPS

product with things like product analytics and NPS surveys.

As an example, Confluence has been around for a fairly long time now and it has a lot of features that have little to no analytics. One of those is the dashboard, which is the beginning of most people's journey with Confluence. We are actually in the process of overhauling it right now. We had some feedback about the dashboard from customer interviews, but we didn't have all the product analytics needed to really understand usage from a quantitative perspective. We had a lot of unanswered questions, like:

- How much usage does the dashboard get? How many times do people visit the dashboard in a typical Confluence session?
- What do people actually use the dashboard for? The All Updates feed? The Popular feed? Navigating to a space?
- What do people want on the dashboard? Can we determine the best things to surface based on what people do after visiting the dashboard?

These are some pretty fundamental questions that we needed answers to before embarking on a change to one of the most visited pages in Confluence. If you don't have analytics in your product, or even a specific feature you're looking to change, then you're in the same boat and should be very wary about making any decisions. It's time to pay off that empathy debt!

In our dashboard testing we learned that one of the most common actions taken on the dashboard was viewing "favorite pages." This was a super important finding, and one that wasn't necessarily in our initial hypothesis. Which brings us to the main takeaway here: Pay off your empathy debt as soon as you can – if you don't have analytics in your product, add them in ASAP and start using data to help inform your product decisions. Otherwise you're making important decisions in the dark. And remember that analytics don't lie! They show us exactly what users are doing with the product, but try and dig a bit deeper and use analytics to understand what users really want.

## Testing the future before it's here

While adding product analytics can be valuable for understanding how users use existing features, they're also extremely valuable for testing new features and experiences. If you have a clear goal for how much you want your feature to be used, having product analytics

helps you work towards that old agile mantra of failing fast and iterating until you succeed.

The process that we use generally looks like this:

- Define a clear hypothesis for a product change – e.g. "By increasing the size of the comment box we expect to see a 5% increase in commenting."
- Build the cheapest possible implementation of this change, loaded with any analytics events we need, that will allow us to test our hypothesis.
- Deploy the change to a subset of customers in an A/B test.
- Twiddle our thumbs while we wait for results.
- Do a breakdown of the results, with the help of an analyst in the case of more complex changes, and decide whether the change was successful.

For our dashboard changes we ended up designing three very "opinionated" dashboards, each promoting a different use case and set of behaviours. We ran them through this process (though our hypothesis was somewhat more complicated) and it worked really well for us. But there are some common gotchas we've learned – sometimes the hard way – that you'll want to think about before testing new features this way.

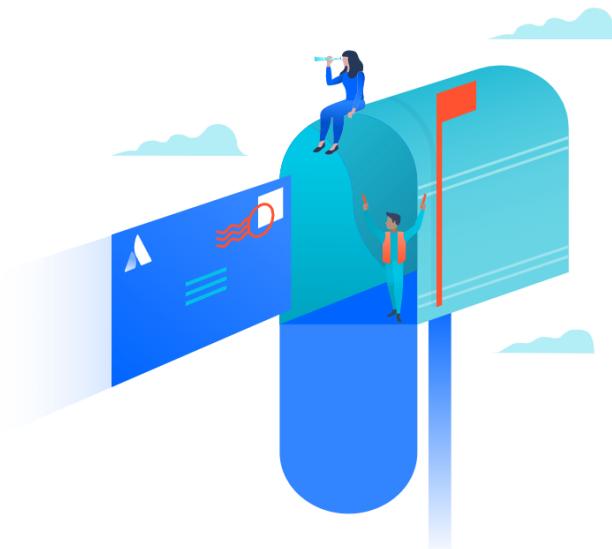
#### SOME ANTI-PATTERNS TO WATCH OUT FOR:

- There's nothing worse than getting to the end of an experiment and realizing you don't have all the events you need... Try to do your analysis before you run the experiment using some dummy data, you'll quickly see any gaps in what you're capturing.
- Coming up with a hypothesis can be time consuming, but you need to make sure you have one and that you're confident you can prove or disprove it with the product analytics you have before you launch. Doing an analysis on dummy data before launching will help you test this, too.
- Make sure you're testing on enough users and for a long enough period of time. You want to make sure your results are statistically significant.
- Be prepared to throw away bad ideas! As I mentioned, you want to test features out as cheaply as possible and run these tests as quickly as possible. Failing fast is good.

**Just don't forget to listen to your**

## users, too

As I mentioned above, it's great to be data-informed, but being entirely data-driven can sometimes leave you blind to the overall experience that you're creating for users. Being dependent entirely on data can also be a bit crippling when it comes time to make a decision and you don't have all the data you need.



Product analytics expose the raw reality of how people use the product, or even a particular feature, but it can be very one-dimensional. Combining what you think you know from product analytics data with qualitative feedback in customer interviews, concept testing workshops, and sparring will give you a more complete picture of what's happening so you can build the best product possible.

SHARE THIS ARTICLE



SAM TARDIF

I'm a developer-cum-growth-hacker-cum-developer-again, now leading the Confluence team. At work, my passions are data-driven product development and simple, elegant products. Outside of work I stay away from data, spending more time with puppies and making my way through every Japanese restaurant Sydney has to offer.



TUTORIAL

## Create an agile board in Jira Software

Find out how to create agile boards in Jira with this step by step guide

[Try this tutorial →](#)

ARTICLE

## How to build a kick-ass agile team

Explore agile team dynamics, culture & collaboration, and build a great agile team.

[Read this article →](#)

### Agile Topics

Agile project management

Scrum

Kanban

Design

Software development

Product management

Teams

Agile at scale

DevOps

Sign up for more agile articles and tutorials.

Email

[Subscribe](#)



Up Next  
[Remote product management →](#)