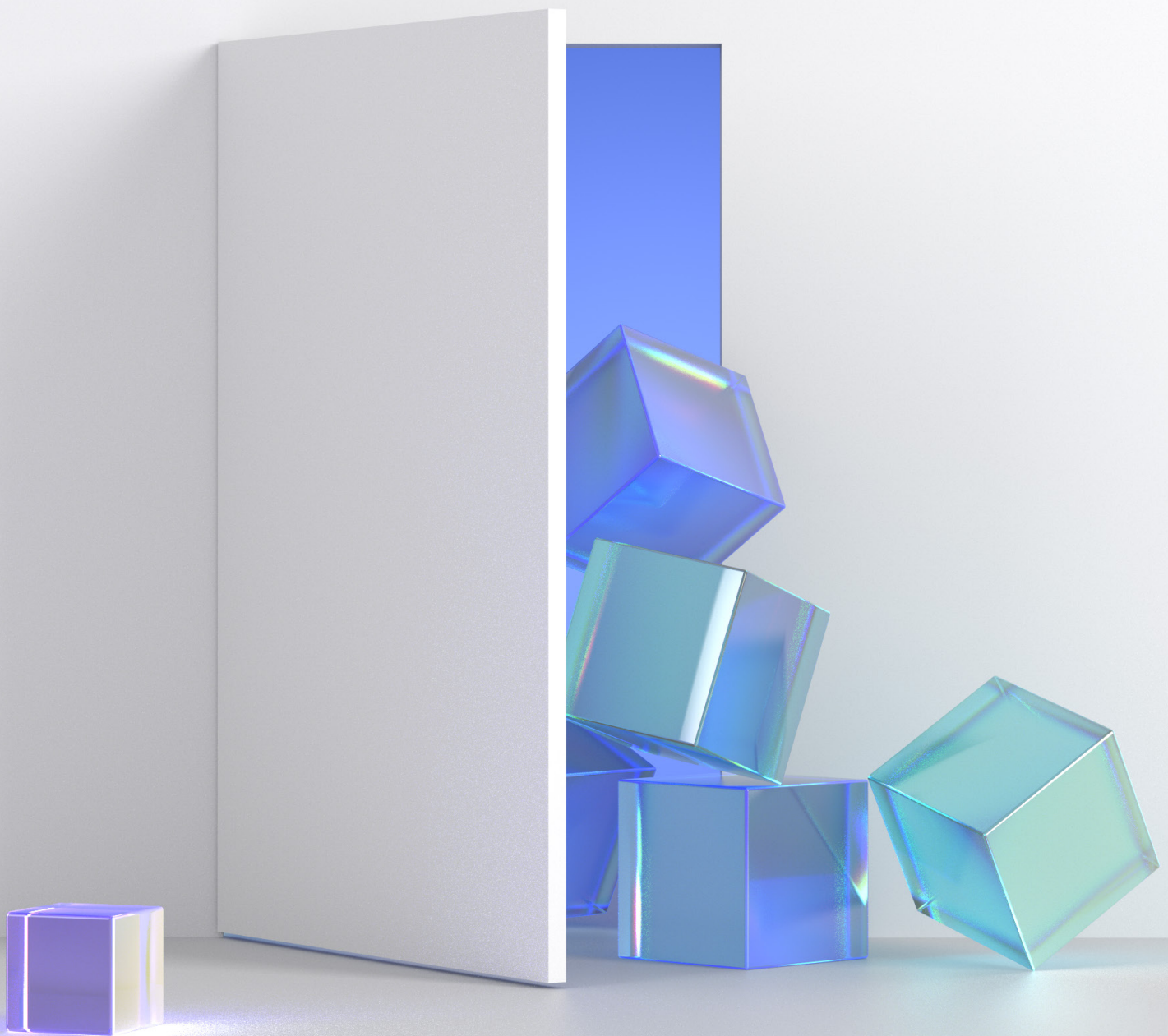


THE ROUNDTABLE DEBATE

airfocus PRESENTS

How Can Product Teams Stop Being “Feature Factories”?



The Roundtable Debate: How Can Product Teams Stop Being “Feature Factories”?

Having more features than their competitors doesn't automatically make your product successful. Apple, Google, and Amazon, for example, have earned their position because their features solve users' actual problems, are well thought out prior to release, and are closely maintained post-release.

While it's safe to assume that people in product are generally aware of this in theory, our recent re-search discovers that around 86% of global product managers still feel like they are working in a “feature factory” - a term first introduced by John Cutler - churning out features and focusing on quantity over quality. Why is that the case? And more importantly, how can teams break out of the vicious circle of feature shipping?

In April 2022, airfocus invited three renowned and respected product coaches and leaders - C. Todd Lombardo, Tim Herbig, and Quadri Oshibotu - to a roundtable to share their experiences and insights on this topic. They discussed the problems caused by working in a feature factory, how PMs can actively become more outcome-focused, and how to approach the situation with the higher-ups.

ON THE PANEL WERE:



C. TODD LOMBARDO
VP of Product & Design
at Openly



TIM HERBIG
Product Management
Coach & Consultant



QUADRI OSHIBOTU
Founder
of Product Hall



The panel was moderated by **PAUL ALLEN**,
Sarum PR

PAUL ALLEN:

What is so bad about becoming a „Feature Factory,“ and what are some problems this situation can bring to the product team and the wider organization?

QUADRI OSHIBOTU:

A feature factory describes a company that consistently releases products, features, and enhancements, focusing mainly on the quantity of the outputs rather than the quality of the outcomes generated. By only focusing on releasing features, you miss out on continuous discovery opportunities, understanding what your customers and users are looking for, or how the released features and enhancements tie into your product strategy. You could waste time and resources for your organization with poorly implemented features or bloated products. Ultimately, this situation can move you away from developing towards providing more value for your users and customers because you're not exactly taking the time to understand and work on meeting their needs.

TIM HERBIG:

Quadri summed that up very accurately, so maybe I'll just add to that; another way being a feature factory may harm your company is the effect it has on your teams. When your team realizes that they are constantly just chasing after new features and are only doing what they are told, a lack of motivation and drive may start building amongst the team members. Such lack of motivation can lead to employee churn or teams made of people who just obey, would never challenge you, and never get back to you on a program, a product, or a feature idea. The lack of feedback will leave you with a subpar quality of your overall product. So I think this lack of motivation is a severe consequence of the feature factory situation.

C.TODD LOMBARDO:

Perhaps a bit contrary to the popular opinion of my colleagues here, I think there are times when being a feature factory has its uses. There may be times when you've got a bunch of things to do because you know that this feature set or that will be truly useful to your customer base. That is when you go into the feature factory mode, just working on getting it done. The challenge lies in ensuring this does not become your long-term modus operandi and that your focus does not switch to producing output. That's where the problem lies, making sure you only become a feature factory when it is beneficial to you and your customers.

PAUL ALLEN:

What are the telltale signs that you have been drawn into a feature factory situation?

TIM HERBIG:

I usually look for two major red flags, the first being the measure of your team's success. If your measure of success is output-focused, like sprint velocity, the number of deadlines met, or the number of features produced, you found your first red flag. The other one is the ratio between the number of features released and the number of customer behaviors directly influenced as a result. What I like to do with teams is to have them line up all the features of projects they're currently working on and have them connect each feature to the specific change in user behavior they're trying to cause. You're certainly a feature factory if you are just following a list of features that are not directly linked to a real change for your customers.

QUADRI OSHIBOTU:

Another significant telltale sign, in my opinion, is when features aren't enhanced upon release. In a feature factory situation, once the team releases a set of features, they immediately focus on the next batch to release. There simply isn't the capacity to go back and look at the recently released features, chat with customers to understand whether these features are actually solving their problems, and see what the data and the metrics show. This ties to another significant issue, which is minimal, if any, customer discussion. Customer discussions are vital, especially for customer development. Being in constant touch with your customers as you not only build your products, features, and enhancements ensures that your product continues to address their pain points as they evolve. If you are just focusing on producing more features without really considering their impact, you are a feature factory.

C.TODD LOMBARDO:

To me, you are a feature factory when you are shipping a lot of stuff, and you're not paying attention to the outcomes; to what happens after you ship it. A definitive telltale sign here is that you might not see enough analytics. A couple of companies I've worked with recently had zero instrumentation in their products. So they had no insight into whether their last release actually worked as intended. From the beginning, you should ask yourself: do we have tools to measure what this was supposed to be doing? No? Okay, that's the first thing you will set up - the analytics and get some dashboards around what you will be measuring. It is one thing to have a goal, but how do you know how to measure it? You really need data. This lack of a data-driven approach is something you'll see in organizations that fell into the feature factory situation.

PAUL ALLEN:**How should a product manager react to new feature ideas from within the team or feature requests from their big customers?****C.TODD LOMBARDO:**

You don't want to be the product manager who always says „No!“ so I often recommend approaching this tactically. Say: „well, that's really interesting. Can you tell me more about that?“ Those three little words „Tell Me More“ can open up a massive conversation around why they are pushing for this feature? Is it the salesperson trying to close a deal? Okay, let's talk more about that with the customer too. By deep-diving into the request, you might find out that you have something that might be able to address that specific customer problem already being pushed for or on your roadmap. Let's see this as an example. I had a client where many of their customers and Sales Team reps were asking for a better Excel file export for one of the products from a few years back. I just heard it over and over and over, but I didn't understand why exactly they were asking this. So I asked them to tell me more about it and explain what they are looking for.

It turns out that the customers were downloading data into Excel, but they needed these in graphs to send to their boss. And then it hit me - they didn't need a feature for a better Excel export; they needed a way to visualize and share the data. And we already had some of that on our near-term roadmap! So we just said, „Hold on,“ and we started pushing for the release of this feature. And voila, the problem was solved, even though it wasn't a direct response to what the originally requested feature was supposed to be. So always think of the „why?“. Then, find what problem that feature request is solving and see if you can solve that problem directly.

QUADRI OSHIBOTU:

Feature requests should always be documented, including any additional information that may be required to assist the product manager or PM team in making the prioritization decisions. One key thing, especially for those working within a feature factory and trying to escape it, is always to understand why stakeholders are requesting new features. It should never be the case that you give out features just because you have a big, important customer asking for something. Always ask - why? Ask "why" multiple times to understand the underlying problem - why they're asking for what they're asking for. Simultaneously, you should also assess whether solving that specific problem will actually assist your company in reaching its goals for the quarter or the year.

TIM HERBIG:

Indeed, my thoughts are pretty much in line with what Quadri said here. It is vital for your organization to have some kind of an objective, like strategy OKRs or a roadmap artifact, that acts as a benchmark, an anchor that helps you to say yes or no to specific requests. However, I'm not a big fan of rigidly pre-planned maximal capacity, so I believe this evaluation should be done flexibly. For example, maybe you receive a feature request that is so valid or that the problems behind it are so new or severe that it will make you change your organization's goals. So long as this is a conscious, thought-through, and well-discussed decision that aligns or improves your company strategy, it is good to say yes to the big clients. An added layer certainly is the nature of your industry, especially in B2B. A particular case is sales-led organizations. If your business model relies on direct sales, you have another stream for the ad hoc requests of your customers in the form of salespeople. They, being measured by closing deals and sales revenue, have their incentives to push for certain potential lead-closing feature requests. But, as Teresa Torres once said, „you're never just one feature away from

success or closing a deal". So, I think it's important not to fall for the idea that if you just push this one thing out, you will succeed. Instead, evaluate the requests based on how they align with your company goals.

PAUL ALLEN:

Why is „outcome over output“ so important for effective product management? And how can product teams convince the board and the wider organization that outcome over output is a smarter approach to product management?

QUADRI OSHIBOTU:

There is a critical distinction between outcomes and outputs. The output doesn't get you anywhere per se. The only thing that you can do is brag about it. When you focus just on the output, you don't have the correct information needed to improve your processes or make any changes to your features. Having outcomes that you're tracking is what's going to differentiate you from your other competitors. And I think this is a key distinction that needs to be made when speaking to the executives. Are we here to focus on how many features we're getting out and how fast? Or are we here to discuss how our products are impacting lives? How are we making the lives of our customers and our users a bit better? How are we reaching our business goals? You can ask these questions when you have objectives that you're tracking and outcomes that you actually want to work towards.

C.TODD LOMBARDO:

The relationship between outputs and outcomes is a fragile one, in my opinion. We need to acknowledge that the outputs drive the outcomes. You need to create outputs, but you should continuously monitor that outputs are conceptualized and executed with outcomes in mind. Now, there are plenty of times an executive team comes saying: „hey, go build these features“, because in their minds, they believe that those features are going to drive an outcome. And sometimes they do. That’s the challenge - because the suggestions are not always 100% bad or 100% good. So as a product manager, instead of immediately accepting or dismissing a request, you could say: „Okay, you told me to ship X, but in service of what?“ Demand clear outcomes from those who request new features from you, wherever the request comes from. If a clear outcome was not defined, accept the input, but you then make the call whether to act upon it or not. Essentially, it boils down to how you create and subsequently measure the value of your product. Are you going to build a business around a list of particular features, or around a list of outcomes you want to provide? The answer will determine how you’re going to drive value and how the output-outcome relationship will look in practice.

PAUL ALLEN:

So, if you’re an organization, a product manager, or a product team in an organization that has been dragged into a feature factory situation, how can you break out of that vicious cycle of feature shipping?

TIM HERBIG:

I think there’s a „chicken or the egg“ problem here. On the one hand, you have to educate and upskill the people you’re working with to help them figure out how to identify an outcome worth building a feature for. Your teams need to be taught the skills and use the right tools to track, measure, and translate outcomes into future actions. On the other hand, I think all the hard skills and tools are almost useless if you don’t give your teams the autonomy to explore the outcomes. Allow them to talk to your customers and conduct research into the outcomes. If all your teams ever get from the management are feature lists, they will start asking „What do I have the skills for?“ So I think there has to be a balanced environment where people are not bound to promised feature requests for 12 to 24 months in the future. At the same time, they are not left in the void of autonomy or don’t have the skills to identify and execute outcomes.

QUADRI OSHIBOTU:

I would say there are a few ways of getting out of a feature factory situation. Number one, fostering a culture where team members ask questions is essential. Why are we working on this? Who asked for this? Is this the most important thing to work on at the moment? Having your team challenge the feature release will clarify it for you and get your team to think of it more collaboratively. Number two is to start leveraging customer development and continuous discovery. Speaking with your customers constantly to understand what they’re looking for and build alongside their current needs. If you maintain this feedback loop even as you’re building your features, you get feedback before release, rather than simply releasing it, getting their feedback, and then working retrospectively. But the ultimate means of escaping the feature factory trap is the ability to tie whichever features you’re working on to specific outcomes or objectives that you want to accomplish for your organization.

C.TODD LOMBARDO:

Precisely as Quadri said, although this should not be limited to tying just the business outcomes, it's tying the feature to a business and the user outcome - you don't want to have it be one or the other. That can be a big problem. As a PM, you want to balance the business outcome with the user outcome by placing your solutions in the middle. Think of this as a Venn diagram - where there is the overlap; there is your feature solving for both the company and the client. This is why product management is hard. Sometimes you might be one or the other. Okay, fine. But overall, you should be making sure you're solving for both the business and the user simultaneously.

PAUL ALLEN:

As you said, product management is hard, I believe that will resonate with a lot of people who are watching today. And I think generally, people will agree that the right tools are important for product management. How do you go about choosing the right platform for your organization's particular requirements?

QUADRI OSHIBOTU:

When looking for features, one should remember that product management is not the same from company to company. So make sure to look for modular and customizable tools so you can set up your tool stack according to how your company operates and the needs of the various teams you work with. And, very importantly, you must ensure the integration options. So whether you're working with your design team, your development team, or your customer success team, you need to make sure your tools can integrate with their tools as well. That way, you can have a 360-degree view of everything happening with the products you manage across the various teams you work with.

C.TODD LOMBARDO:

When selecting your tools, you want to ensure you get the job done. Thankfully we are now past the stone age of PM, where we ran the show with just Excel and PowerPoint, and there are many intelligent tools to help you achieve what you need. The thing that I think about a lot with my clients is how you centralize feedback in one spot. There are many different streams of communication and so many ways for customers, users, and teammates to give feedback and offer suggestions. You need to have a tool to put them all in one spot and tag them appropriately. Centralizing feedback is super, super helpful. And from there, find a way to help you plan and act on this. Then, you can integrate those with planning tools to help you determine your short and long-term milestones. Just be careful about changing the tools too frequently, as that can confuse your team. I speak from experience here, as we have gone through three different sets of tools in the last 12 months, and it's been a little challenging.

TIM HERBIG:

For me, I like to pay attention to differentiating between thinking General and Specialized. Generalist tools are more flexible and allow you to map the broad strokes in your processes. They help you with the more creative parts like running workshops and, at times, the more ad hoc solution tools, which still integrate well with some of the more technical activities you're doing. And then, you need the Specialist tools like the ones C.Todd mentioned for storing user feedback. Specifically, you need to have a wireframing probably for user testing, for prototyping, for whatever your discipline or your domain requires that these tools can provide. And finally, you want to make sure that the specialist tools can integrate well with the generalist tools to limit the frictions.

PAUL ALLEN:

Can you give some perspective on internal enterprise products and the corresponding feature factory resulting from an influential leader's push?

QUADRI OSHIBOTU:

So whether it's an internal-facing product or an external-facing product that you're offering to customers, not just the employees of your organization, I think the process would be the same. I think when it comes to internal products, some of the hesitations or issues may be that as a product manager for an internal-facing product, you're concerned that because management is making the request you have to deliver it. But that's not necessarily the case; you are a product manager for a reason. You are the one, to a certain extent, that makes the final decision on whether this should be done or not. So even if management or a very important customer is requesting features, you have to be able to understand and articulate why specifically we are going to work on this. So I think the advice that we shared ties into internal products as well as understanding the business outcomes. But likewise, the customer or the internal user outcomes as well.

TIM HERBIG:

To build on what Quadri said - for most product teams or product managers, it can be difficult to view their colleagues as customers or users and treat them the same way. You have to learn to apply that same rigor and genuine curiosity you would bring to a standard user interview when sitting down with a colleague. So I think that can be a challenge, but as Quadri said, I don't think it makes sense to change the fundamentals. The fundamentals are the same, which also leads to a great internal product. It just might be more challenging to apply them due to the friendship or colleague bias built into your audience relationship.

PAUL ALLEN:

So with regards to measuring the impact that delivery had, what are some best practices to keep track in a B2B environment where the impact might be delayed over time?

C.TODD LOMBARDO:

When it comes to understanding outcomes, it rarely happens that the right outcomes are a trailing indicator. Many people think, "Alright, I've shipped the feature, I'm done". No, you're not. You as a product manager need to pay attention to what happens next. You can't expect that as soon as you flip the release switch, that suddenly 100% of your users will be using that new feature. Have you educated them? Have you told them? How long does it take? They might not have logged in for like four or five days. And you also need to think about the data to measure your expectations of the feature performance and customer behavior. You should be tracking those data points. If we don't see that we hit those metrics, something's wrong; we didn't educate enough, the feature wasn't quite right, and we didn't solve the problem. So we need to iterate. And that's your indicator saying: Okay, we didn't solve this problem; let's figure out another way to look at it and measure it.

QUADRI OSHIBOTU:

I'd also add that it may take some time for you to understand the outcomes based on the metrics. So, to speed it up, it is always beneficial that upon releasing new features, products, and enhancements, you hop on a call with your customers and ask for their feedback. You know, how's the design experience? Is this assisting your team with accomplishing their goals? And likewise, are there any challenges that you're facing as well? So as you're waiting for those metrics, at least you can obtain that qualitative feedback from them.

PAUL ALLEN:

If you have to choose one key takeaway for the audience from this session, what will it be?

TIM HERBIG:

Invest in decision-making criteria that you can actually turn to when you have to make decisions. Again, whether that is some kind of objectified framework, one pager alignment, document, or whatever you need in your company - invest in this explicit alignment and understanding among your teams so you can use it whenever you have to make decisions.

C. TODD LOMBARDO:

I think the biggest thing is making sure that you have very clear outcome-driven goals, and that you have the data to measure your progress towards them. If these two things are completely absent, that'll be one thing, but oftentimes, we all have some gray area in the middle. And so we need to sharpen those goals to make them very much outcome-driven, and make sure we can get the data to measure that progress.

QUADRI OSHIBOTU:

Similar to what C. Todd mentioned, just fostering a culture where the focus is on outcomes. If there's any feature that you're working on, ensure that you can tie them to specific outcomes and that those are the right outcomes that your company and your teams are working towards.

To learn more about outcome-focused product management

please visit www.airfocus.com

