What is Developer Marketing?

appsembler

Table of Contents

What is Developer Marketing?

The answer isn't as straightforward as you might think.

Developer marketing aims to accomplish more than just capture the attention of software developers; it's the collection of principles, strategies, and tactics that companies use to increase their product's awareness, adoption, and advocacy across various technical audiences.

Ultimately, an effective developer marketing program will educate technical audiences on your product's features, help them solve their problems using your product, with the final effect of increasing product adoption and usage.

appsembler
Build it with education.

The 5 Benefits of Developer Marketing

eBook: The 5 Benefits of Developer Marketing

Why is Developer Marketing Important?

There are three (3) reasons why you should be paying attention to developer marketing:

1 Developers are controlling more and more software purchasing decisions;

2 The developer population is increasing; and

3 Traditional marketing methods don't work on developers.

Let's first talk about the developer's growing influence in software purchasing decisions. According to Evans Data, a market research firm specializing in the software development industry, 95% of developers have some role in purchasing. Another survey by DeveloperMedia suggests that 60% of developers have the ability to approve or reject a technology purchase, while Stack Overflow's 2020 Developer Survey (comprising over 65,000 developers) discovered that 57% of developers exert a great deal of influence over technology purchases. And of course, there's the dreaded "dark horse" effect — something that's probably all too familiar with your sales team and Account Executives — where a technical evaluator vetoes a product purchase at the 11th hour (34% of opportunities are lost as a direct result of developer influence). All of these studies point toward the same conclusion: developers are controlling more and more software purchasing decisions.


Source: Why developers have influence today? by Max Katz

Developers are influencing more and more purchasing decisions — so what? They are a small subset of the population and represent a small % of my company's total addressable market (TAM). Well, that depends on what you mean by "population" and what represents your TAM. Obviously, if you're selling a product that is strictly sold to consumers, developer marketing would be a useless exercise. But if you're marketing a product or service that is directly or indirectly tied to software, then there's a good chance that a software developer is part of your product's evaluation process. In this case, you should know that SlashData's State of Developer Nation Report (20th Edition) estimates that the total global developer population stands at 24.3 million as of

Q1 2021. And at a DevMarCon talk titled "The Marketer's Guide to Building a Developer Program," Dinesh Chandrasekhar, Cloudera's Head of Product Marketing, noted that $40 billion in products are either consumed, influenced, or purchased by developers.

Finally, developers are unlike any other audience you've ever marketed to. Traditional B2C (or even B2B) marketing won't work in **business-to-developer (B2D) marketing**. Developers are very analytical and critical, and are likely working on a problem (which your product may or may not solve — they'll decide that) that they understand at a level of detail that surpasses your marketing team's understanding of the problem. What this means is: developers **don't** want to learn about your product's benefits or the problems it can solve, they just want to know and see its features. They're not interested in hearing about your product's benefits because they'll decide (through hands-on experience) if your product actually has the benefits it touts. And they're not interested in the problems your product solves because they're already working on that problem. So when marketing to developers, keep these perspectives in mind and try to keep the fluff to a minimum. Be descriptive, accurate, and efficient with your words. And more importantly, give them frictionless access to a hands-on environment of your product. A hands-on product experience is worth significantly more than any marketing materials you can try to have developers to download.
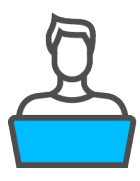
A **virtual lab** is a virtual environment for learners to have hands-on experience with the same software they will eventually use on the job.

What are the Benefits of Developer Marketing?

So we've established that developers influence a large percentage of software purchasing decisions, are a growing population, and have a persona quite different from other audiences you've marketed to. With all of these hurdles set in front of us, why should software companies consider doing developer marketing?

There are five (5) main reasons which we summarize below. If you want a more detailed discussion on the benefits of developer marketing, download The 5 Benefits of Developer Marketing eBook. The eBook is especially aimed at marketers who are struggling to articulate the benefits of developer marketing to non-technical and business stakeholders.

eBook: The 5 Benefits of Developer Marketing

The 5 Benefits of Developer Marketing (Summarized)

1. Developer Adoption

Your developer marketing program should help increase the number of developers who initially sign up for your product, access an API, or start building applications for a marketplace. Developer marketing is so much more than just generating developer sign-ups, but your stakeholders and high-level executives will likely pay close attention to this KPI — so work on measuring it and understanding its drivers.

The success of your developer marketing program may not be directly tied to developer adoption, but it's a crucial KPI that you should measure in order to build stakeholder support and runway for your developer marketing initiative. It's also an effective way of measuring how well your developer marketing efforts are at attracting developers.

2. Developer Success

Developer success helps your developer community become more educated with your product, API, marketplace, etc. While developer success historically hinged on products like GitHub Pages, Read the Docs, Tettra, and other platforms to host and distribute documentation, developer documentation geared for the modern-day developer should be interactive and hands-on. Hands-on product environments (often called playgrounds) will ensure that developers get a learn-by-doing environment that lets them analyze your product to determine if it solves the problem they're working on.

3. Developer Usage

Increasing the developer community's ongoing usage is what helps preserve the long-term growth and success of your product. This is an important theme to consider as you build-out your developer marketing program, because you've exerted a lot of effort acquiring developers — so you want to make sure they use your product for the long-term. To help improve long-term developer usage of your product, reduce the friction between your product and the developer's learning experience, provide helpful product context when necessary, and give incentives that will encourage developers to continually return to your products.

4. Lowering Developer Support Costs

As more and more developers adopt your product, the number of support tickets they generate will inevitably rise. Normally, this would require your internal team to spend valuable time and resources tending to the requests and questions of your developer community. But when handled properly, your developer advocacy team can pinpoint common issues being raised within the community and build product courses to address the common themes in your developer community's support tickets.

5. Orchestrating Activities with Marketing

You've successfully managed to get developers using and enjoying your products. Next, you'll want to consider how your developer marketers, traditional marketers, and sales staff communicate and engage with developers. From product training offers, product roadmap announcements, or a call to discuss potential license purchases, there's a wide variety of topics that your teams can discuss with your developer community. It's important to take the time to determine when you should be reaching out to certain developers, what content you should be sending them, and who should be responsible for handling those communications.

3 Drivers for Business-to-Developer (B2D) Marketing Success

1 **Lower the friction for developers**

Friction refers to the amount of effort a developer has to exert in order to access or use your product. Time is precious within the developer community, which makes it absolutely critical that you make it as easy as possible for developers to reach a live product experience. When developers have frictionless access to a live product experience, they can analyze your product's effectiveness and reach a purchasing decision faster.

2 **Educate, *Don't Market or Sell***

Developers hate marketing. If they get the sense that you're trying to push a product on them or throw vague marketing jargon their way, they won't think twice about looking elsewhere for assistance. Instead, aim to educate developers and provide them with insightful information regarding the effectiveness of your product. An educated developer is more likely to adopt your product when they feel fully-educated on its capabilities, potential, and limitations.

To learn more about the importance of education when it comes to effective developer marketing tactics, check out the following Forbes article — **Why Education Is Key To A Developer Marketing Strategy.**

3 **Adopt a Community-Centric Mindset**

Your goal with developer marketing should not be to turn developers into leads or MQLs. Ultimately, your developer marketing team is striving to demonstrate that your product has the backing of a community of experienced developers. Having a large community of developers backing your product or platform means that an ecosystem exists where companies can get help answering technical support questions, learn best practices about your product, hire product experts, and — perhaps most important — that your product has a low risk of being deprecated. This developer ecosystem will give technical buyers the confidence they need to adopt your product, integrate it into their systems, and make it a critical part of their business infrastructure.

5 Challenges to Doing Effective Developer Marketing

1. Traditional Means of Marketing Won't Work

Developers will not respond to traditional forms of marketing or advertising. In fact, many developers have (by default) an ad blocker that puts a wall between themselves and your company's marketing and advertising efforts. Additionally, they rarely look at external emails from unfamiliar businesses, with the exception of educational product newsletters that they've subscribed to.

Instead of superfluously speaking about what makes your product great, educate developers on how others are using your product and the ways in which it's helping other developers solve their problems. Just be sure that your language doesn't slip into mainstream marketing lingo and keep the content solely focused on empowering developers on how to solve their problems.

2. Letting Your Product (and its Features) Do the Talking

Instead of high-level, assumptive messaging about the benefits that your product might deliver, developers prefer reading a straightforward and detailed description about your product's features. Don't make assumptions about the problems developers have; if you effectively communicate the features of your product, developers will be able to connect the dots and determine if your product fits their needs.

Also, make sure your product isn't only accessible behind a gated sales demo. This goes back to providing developers with easy access to a hands-on environment of your product, which will give them first-hand experience with your product and what it can do.

3. Ensuring Ease of Access

Making your product as easy as possible for developers to access is imperative to earning their trust. When you add gates to your product, demo, or content, you're creating an unnecessary barrier between the developer and your offerings. At best, this deters them from learning more about what you have to offer — and at worst, it creates the speculation that your product needs to "hide" behind a sanitized sales demo in order for it to perform well.

There will be plenty of time down the road to gather contact information about your developers. But when making an initial impression, focus on ensuring that developers have easy and streamlined access to your product.

4. Crafting the Right Messaging Approach

Crafting the best user experience for developers requires you to adopt a different mindset when writing content. You never want to treat developers as leads, but rather as a workforce that's eager to learn, improve, and utilize the best tools that solve their problems.

It might be helpful to see how other organizations approach their developer marketing efforts, and take notes as to how their messaging works to resonate with the developer community (we provide several examples below). Your messaging approach should be centered around providing educational information and resources that enable developers to improve their understanding of your product.

5. Avoiding the Early "Sell"

If you're calling developers the moment they sign-up for a product demo or trial, you're doing it wrong. Developers hate being sold to in the traditional sense, and won't respond well to your marketing or sales team reaching out to gauge their interest in a license purchase.

Developer marketing is a slow burn and your initial efforts should be focused around empowering and educating your audience on your product. Make sure your developer advocacy team is aligned in how you aim to move developers through the funnel. Only when a developer makes it explicitly clear that they're ready to work with you should your sales team be reaching out. With developers, when it comes to initiating a purchase conversation, it's better to be too late than too early.

Developer Marketing and Relations: The Essential Guide

Go deeper into creating a developer outreach program or measuring your B2D success with SlashData's essential guide. Arranged in an order that takes you from strategic issues to more tactical issues, you can drop into specific chapters or read the guide from start-to-finish.

Learn More
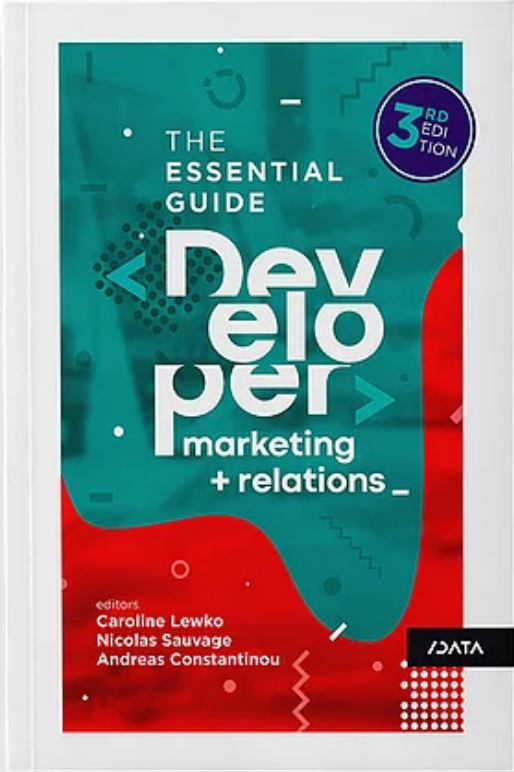
Creating a Developer Outreach Program

Your First Key Hires

An effective developer outreach program starts with building the right team to make it happen. From the onset, you'll need a technical content writer that is well-versed in the product, API, or marketplace that you're focusing your developer marketing efforts on. Often times, the best technical writers are internal engineers who may be looking for a change in roles. Since they already understand the ins and outs of your product, they can hit the ground running when it comes to crafting content aimed at a technical audience.

Eventually, you'll also need a community or marketing manager who will be responsible for building out the analytics and measurement framework for your developer marketing initiative. They should also be able to amplify the audience that your technical content reaches, as well as be responsible for the logistics around developer events that work to boost product awareness and developer engagement.

One area that we did not cover in this eBook, however, is the challenge of building broader stakeholder understanding within an organization that currently does not have a Developer Marketing initiative.

Measuring the Success of Your Program

How do you know whether your developer marketing efforts are actually working? There are a number of ways to track the performance of your program, all of which will help guide your organization to continually improving your tactics over time. Below are some key performance indicators that can measure your developer marketing success

1 **Awareness**

Awareness refers to how many developers your developer marketing program is reaching. This can usually be measured by looking at the number of:

- ⚪ # of registrants to your developer newsletter
- ⚪ # of impressions to your developer portal or content
- ⚪ Social media subscribers (assuming you have a developer-specific channel)
- ⚪ Hackathon and developer event attendees
- ⚪ Blog article views & reads
- ⚪ Social media mentions

Together, these KPIs will paint a picture of how much awareness you have in the developer community. Awareness is the first step towards adoption, so be sure that you're getting noticed by the right developers before focusing on efforts further down the funnel. For example, it's probably not a good idea to increase your product's awareness amongst iOS developers if it's primarily aimed at helping Android developers.

2 Adoption

Your adoption rate analyzes how many developers are actually signing-up for your product. Awareness is a pivotal starting point to getting noticed by prospective developers, but ultimately, your adoption rate will be a much better indicator of whether your awareness efforts are driving results. The following indicators play into your adoption rate:

○ How many virtual labs or developer playgrounds are being spun-up

○ How many trials (attributed to your B2D efforts) are being spun-up

○ How many applications are being built or have been built

○ Number of applications per developer

○ Number of 3rd party integrations onto other platforms

Once developers begin using your product, you'll start to gain a better sense of how they're enjoying the experience and how often they're returning to it.

3 Engagement

While adoption rate looks at the number of developers making that initial first touch with your product, your engagement rate analyzes how frequently developers are using it. This is an important metric to determine whether developers are continually returning to use your product, or if they just wanted to give it a test run. Engagement rate will require you to look at:

○ How often are developers using your product

○ How often specific features within a product are being used

○ Number of API calls (assuming your company sells or markets an API)

○ Average number of logins per developer

○ Number of developers engaging with your product 30 days after sign-up

Engagement rate helps you understand whether you've built a product that developers will regularly use in their daily workflows, so be sure to keep an eye on how many initial sign-ups convert to engaged, long-term users.

4 Community

Your developer community is the foundation that your product stands on. Building a strong community depends on your organization's ability to participate in insightful conversations with developers and add value as like-minded professionals and thought-leaders. The community metrics you should be taking note of include looking at the number of:

○ Monthly growth of your developer community

○ # of developer groups and # of new groups added per month

○ # of developer meetings per month

○ # of developers per group and # of total developers

○ Active developer tokens

5 Developer Satisfaction

You've attracted the right developers and engaged with them in a meaningful way. Now it's time to consider how satisfied they are with your product. Determining your Developer Satisfaction Score will require you to look at two factors:

○ Your Net Promoter Score (NPS)

○ Time to First Hello World (this is a common acronym used in developer marketing circles to refer to the time it takes for developers to experience a product's value. The lower the TTFHW, the better)
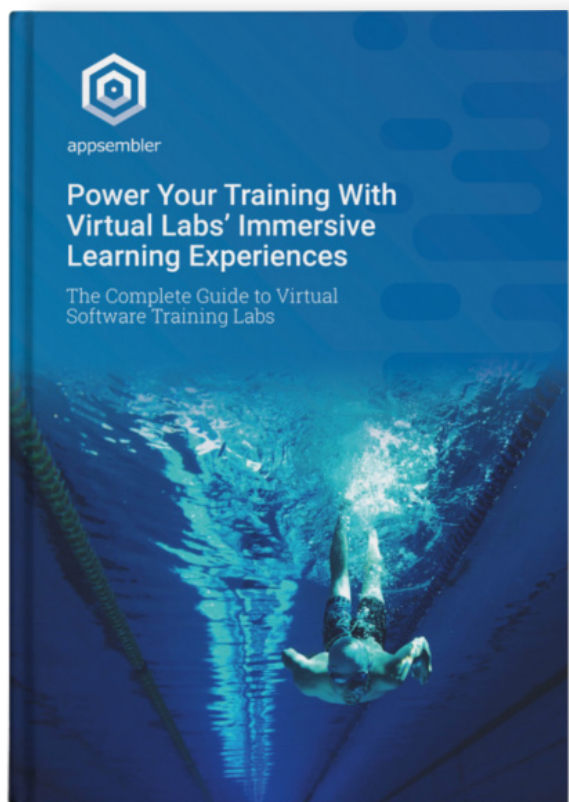
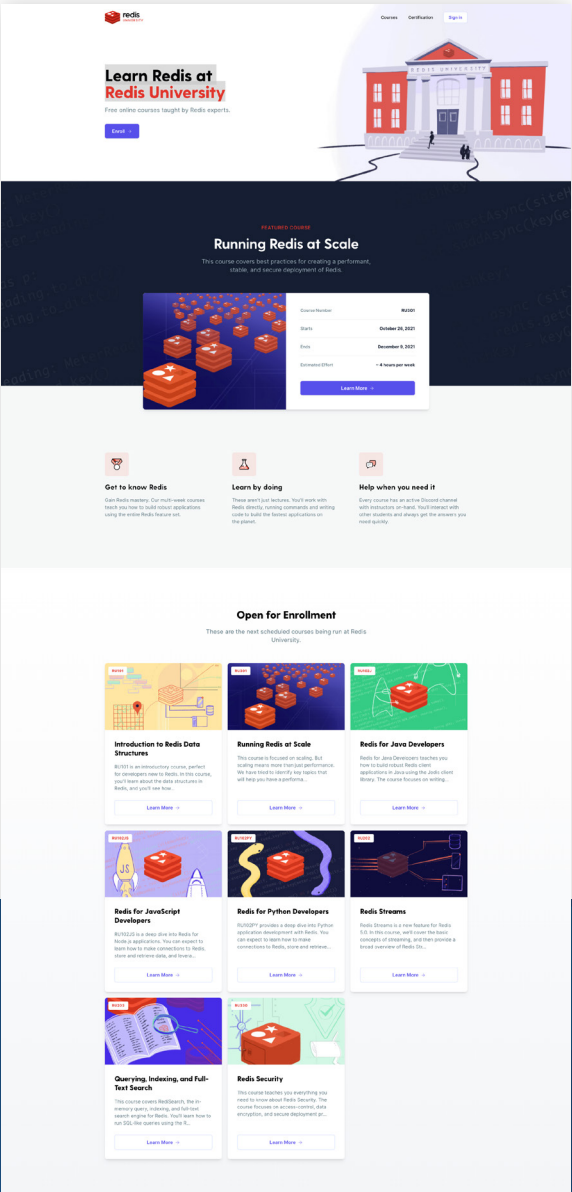# The Complete Guide to Virtual Labs

Learn about the underlying technology enabling companies to deliver hands-on versions of their product.

Traditionally used for training, virtual labs are being repurposed by software companies for developer education, developer marketing, build product awareness, and drive product adoption.

Download the Guide

appsembler

Power Your Training With Virtual Labs' Immersive Learning Experiences

The Complete Guide to Virtual Software Training Labs

Examples of Developer Marketing Initiatives

1

Redis

Redis is the open source home and commercial provider of Redis, the world's leading real-time in-memory data platform that's used on a global scale. Through Appsembler for Developer Marketing, Redis launched Redis University, where Redis users could take online learning courses to collaborate with peers and gain insights from the world's most renowned database experts.

Check out our webinar to get an inside view of how and why Redis created their dedicated education team to take advantage of Appsembler and use their online learning courses as effective marketing collateral.
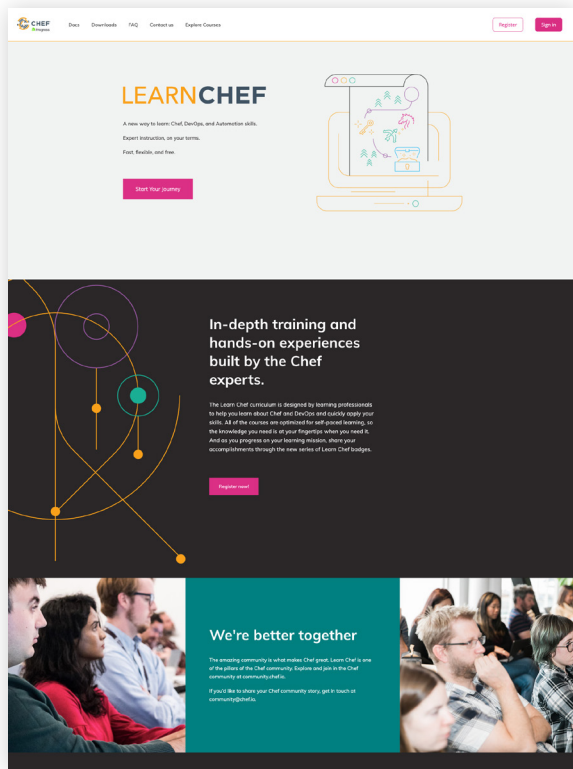
2

Chef Software

[Chef](#) is the leader in Continuous Automation software, an innovator in application automation, and one of the founders of the DevSecOps movement. Providing commercial solutions around the open source Chef Software project, Chef needed to implement non-traditional marketing efforts that were tailored specifically to their savvy technical buyers.

Chef chose Appsembler for Developer Marketing to provide hands-on educational product experiences to their audience, turning [Learn Chef](#) into a reliable and high-volume lead generation channel.
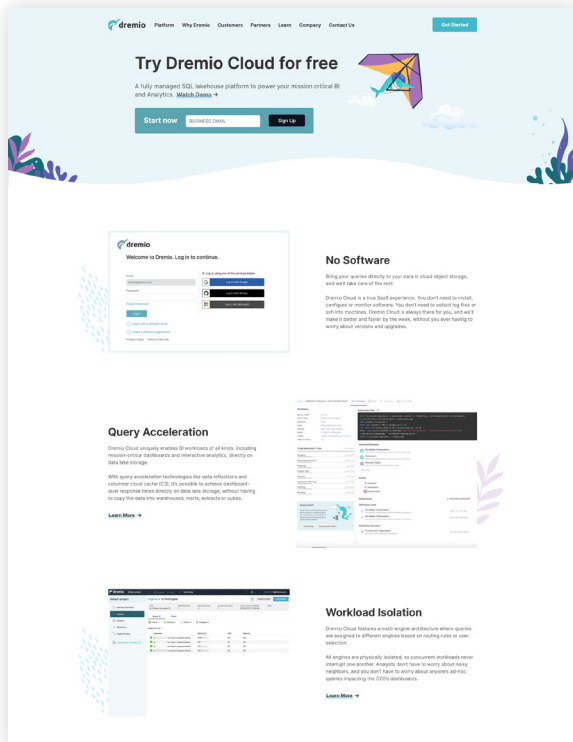
3

Dremio

[Dremio](#) is the SQL lakehouse platform that enables high-performing BI and analytics directly on data lake storage, liberating its customers' data with live, interactive queries. Dremio had a great product, but in order to trial them, potential buyers had to work through a high-friction, multi-step installation process. This was leading more people to see the product trial through and ultimately resulted in fewer conversions.

Appsembler for Developer Marketing provided a reliable solution, helping Dremio increase the volume of product trials delivered to potential buyers by streamlining the trial installation process and prioritizing the user experience above all else.

appsembler
Build it with education.

Speak to Appsembler

Start a conversation with Appsembler and discuss examples of modern, developer marketing initiatives. We'll provide a walkthrough of each initiative and talk about specific results experienced by leading developer marketing and product marketing teams.

Contact Us