



Effective Devops: Building a Culture of Collaboration, Affinity, and Tooling at Scale

Jennifer Davis , Ryn Daniels

[Download now](#)

[Read Online ➔](#)

Effective Devops: Building a Culture of Collaboration, Affinity, and Tooling at Scale

Jennifer Davis , Ryn Daniels

Effective Devops: Building a Culture of Collaboration, Affinity, and Tooling at Scale Jennifer Davis , Ryn Daniels

This practical guide addresses technical, cultural, and managerial challenges of implementing and maintaining a DevOps culture by describing failures and successes. Authors Katherine Daniels and Jennifer Davis provide with actionable strategies you can use to engineer sustainable changes in your environment regardless of your level within your organization.

Effective Devops: Building a Culture of Collaboration, Affinity, and Tooling at Scale Details

Date : Published May 2015 by O'Reilly

ISBN : 9781491926307

Author : Jennifer Davis , Ryn Daniels

Format : Paperback 410 pages

Genre : Science, Technology, Computer Science, Programming, Nonfiction



[Download Effective Devops: Building a Culture of Collaboration, ...pdf](#)



[Read Online Effective Devops: Building a Culture of Collaboration ...pdf](#)

Download and Read Free Online Effective Devops: Building a Culture of Collaboration, Affinity, and Tooling at Scale Jennifer Davis , Ryn Daniels

From Reader Review Effective Devops: Building a Culture of Collaboration, Affinity, and Tooling at Scale for online ebook

Michael says

Focusing on the Culture of DevOps, Effective DevOps is a welcome addition to the literature on modern software development practice. It is a rare book in the field that begins with people and process, then moves to technology. The co-authors have anchored considerations of effectiveness in a broader social science perspective that I found very compelling. Recognizing that there is much new in the DevOps movement, they also place the movement in the broader context of software history. Everything you knew about highly functioning cultures is still true. In this light DevOps is more an evolution than a revolution, a perspective that I tend to agree with.

Sebastiaan Bekker says

Good and broad primer for anyone wanting to start. Wide array of topics that are important for a successful implementation.

Miguel says

A great introductory book on devops (note the capitalisation) and what it stands for as a culture. If you're looking for a technical book this is not it.

I believe labeling this a IT manager handbook would have been more apt. It delves deep into real cultural policies and it's impact on your organization

Thomas says

This book is not a technical book at all, but rather a guide for building well-functioning teams and company cultures that encourage inter-team collaboration. Rarely gives specific recommendations on tooling, but rather how to choose the tools that are right for your team.

Brian Palmer says

The subtitle ("Building a culture of collaboration, affinity, and tooling at scale") is important here, because the book spends a lot more time there than on what I was expecting. Automation is briefly discussed, and even scaling is almost all about scaling hiring practices and team growth rather than technical scaling. It doesn't even spend much time on the portmanteau's "development" vs "operation," preferring to generalize to multiple teams collaborating. The book ends up wandering quite a bit through what the authors believe are best practices in collaboration and affinity (very little discussion of tooling even at a high level).

The case studies are the best part, as they are fairly focused and directly relate to companies' self-reported success in devops. Moreover, they include some different types of companies ("tech" companies to government to retailers).

One brief passage in the book, however, felt completely out of place, and knocked me into a far more critical read of the entire rest of the book. In a discussion about "group membership," the authors discuss "real names" in social networks (spoiler: they're against), in effect blaming the decision on insufficiently diverse teams. It's something the authors clearly have a deep opinion on; heck, it's something I have a deep opinion on. But it's a product decision that involves a variety of arguments which they didn't acknowledge: instead, it's a slam out of nowhere.

A lot of the book is spent on fostering diversity; I felt again in places that they stacked the deck (for example, on remote work,), and floating above issues where affinity groups conflict with inclusivity. There is a lot of advice here for managers looking to attract members of more traditionally underrepresented groups in tech (in particular women), but almost none of it seems devopsy (unless devops = "collaborative culture", which I think could be a fair takeaway from this book if not others).

I probably wouldn't recommend this book to most people, but it does have some meat within for people trying to improve cross-team collaboration and discerning sticky points inside a team.

Terry says

Decent book. Think some folks were put off by many of the "soft skill" items mentioned at the beginning of the book instead of the presumptive battery of tools to "Do Devops" they may have expected. Common sense is not common, so I think many of the recommendations about inclusivity, diversity, and analyzing team makeup, to me seemed like "Duh" but felt that these sorts of discussions are needed because oftentimes people *dont* get it. The valley is the valley for a reason so I feel like the book has done a great job in broaching topics that those in managerial roles may not have brought to the front of their minds. I'm waiting for me.

Good book, and I'm hungry for more.

Liamarcia Bifano says

The book draws a clear line between what is tooling / processes and what is actually devops culture. It makes you think about the usefulness and goals of processes and tools to build a devops culture rather than just throw a bunch of buzz words about devops world. Sometimes it is easy to get lost in this sea of new/crazy methodologies and forget why we do things and its essence.

It also addresses the affects and consequences that a blameful culture will cause and how to avoid this kind of situation. Besides that, it also touches a delicate subject inside companies where "developers" have more prestige than other employees and how harmful it can be. In addition, the book also talks about how a "rock star" culture might not be the right track and how it can be inefficient for the company as a whole.

Razvan says

After first 100 pages I already want to throw it away. It mentions so often the word bias I started to have a bias for this word. Fortunately, it has also good advices. I will keep reading it, but for the moment just 2* for repeating to every few pages about the racial/sexual problems of some people.

Tomasz Gawlik says

Seems like the book has a few points and ideas which are being covered over and over again.

Vitaly says

A more correct title would be "Effective Operations and Organisational Culture", as the book, being well written, tries to touch upon all aspects of company's operations, people collaboration, and so on. Probably a good read for managers, but most likely they read 80% of it somewhere else already, and the other 20% is an overview of DevOps as a culture, and potential issues and perks that come with it. That 80/20 ratio between non-devops/devops topics is why the book feels bloated (at 410 pages!). But my view of the material doesn't correlate with what the author sees/understands as DevOps.

Manu Hortet says

If you dislike this read because the attention it pays to the so-called soft skills, you're probably missing the whole point. The term DevOps is treated, explained and analyzed as a cultural evolution of individual good practices to maintain an agile and sustainable work environment. This culture-focused understanding of the term is way more realistic and functional. Concise but detailed read, greatly explanatory and revealing.

Btw, It's fantastic to read all those female historical references. Appreciate it. :)

Janet says

Not really what I expected. I was looking for guidance on changing company culture and persuading leaders to embrace the efficiencies.

Pretty good, as far as it goes. Doesn't deliver the information I looked for.

YaMan says

Some of the observations were spot on and show me new way of understanding things that are going on at our company. The stories helped to explain presented ideas. I would maybe enjoy it even more if it was shorter :-D probably due to parts that I didn't find so interesting. Overall I'm happy that I came across it and

would recommend it to everyone working in IT.

Mark Honomichl says

It took me a while to get through this book. Rather than laying out some good information on how to effectively build out a Devops organization, it seemed to be more like the annual workplace sensitivity training that one takes at most large organizations.

Yevgeniy Brikman says

This book has two problems: first, they don't define DevOps; second, they don't define the target audience. As a result, while the book contains a lot of interesting content, it's arranged in a way that makes it very difficult to learn.

Now, you might argue, "but wait, they do define DevOps, right in Part 1!" OK, let's look at their official definition:

"Devops is a cultural movement that changes how individuals think about their work, values the diversity of work done, supports intentional processes that accelerate the rate by which businesses realize value, and measures the effect of social and technical change. It is a way of thinking and a way of working that enables individuals and organizations to develop and maintain sustainable work practices. It is a cultural framework for sharing stories and developing empathy, enabling people and teams to practice their crafts in effective and lasting ways."

OK, close your eyes, and tell me, what did that paragraph say?

It's hard to repeat, isn't it? That's because their DevOps definition is vague and deliberately avoids any concrete details. If I remove the word "Devops" from that paragraph above, it could be about anything. The rest of that first part makes you feel like you're trying to hold on to a slippery fish: they spend a ton of time defining what DevOps is not, repeating dozens of times "there is no one true DevOps", and actively dodging and denying any concrete details to the point where, no matter how much you try, you can't grasp it. They even acknowledge this fact in the book itself:

"There has been some discussion in the devops community as to whether or not devops has lost its direction. Critics of the movement say that it is too defined by negative spaces, by people saying what devops isn't rather than what it is (or not providing a concise definition for it at all)."

Yup, I am one of those critics. It seems like the authors attempted to be as all-inclusive as they could be, but the result is that this isn't a book about "Effective DevOps," but rather, a haphazard collection of things the authors believe lead to "Effective Companies." And while one of the goals of DevOps is to make a company more effective, you can't really claim that everything that makes companies effective should be put under the umbrella "DevOps". Their definition is too broad, and as a result, the message of this book is very diluted.

It's also diluted in the sense that the book tries to speak to a bunch of different audiences. Some parts are targeted at developers; some at operations people; some for managers; some for HR; some for the CEO;

some for the legal team. Depending on who you are, you'll find a few parts interesting, and the rest will feel largely irrelevant.

I'll also mention that while the book talks a lot about avoiding unconscious biases, it falls to some itself.

For example, one is the odd stereotype that a "10x engineer" is always a "10x asshole" that no one wants to work with. The *real* 10x engineers are 10x precisely because others **love** working with them and are **more** productive as a result.

Another one is the assumption that the differences between tools (e.g. programming languages, cfg mgmt systems, etc) don't matter. All that matters is how you use the tools. It's certainly true that with the wrong workplace culture, even the best tools will be ineffective. But the opposite is not true: even the best culture won't succeed with the wrong tools. There is a reason Etsy isn't written in assembly; there is a reason you use a config mgmt tool like Chef or Puppet and not manual shell scripts; there is a reason you store data in an RDBMS like MySQL and not flat text files. There is a right tool for the job and it's worth the time to find it.

All of this is a shame, as there is some excellent content lurking in these pages. There are great discussions of diversity and minorities in tech organizations. The authors could extract those discussions, go into more detail, and turn them into an great standalone book focused on that one topic. The case studies scattered throughout this book are intriguing too, as they contain real-world stories, with concrete details (!) of how DevOps actually works. Again, the authors could extract those stories, go into more detail, and create yet another interesting, standalone book on "DevOps stories." Chapter 18 has lots of interesting stories about promotions, transparency, evaluating organizations, and cultural debt (what an awesome concept!). Again, a standalone book on Cultural Debt would've been a great read!

But as it is, all of this intriguing content is crammed into a single book, organized poorly, and only explored at a surface level.

As always, I jot down interesting quotes as I read. Here are some of the best ones from this book:

"There is a sea change happening in software development and operations, and it is not simply the introduction of a new word into our lexicon—it's much more than that. It is a fundamental shift of perspective in the design, construction, and operation of software in a world where almost every successful organization recognizes that software is not something you simply build and launch—it is something you operate."

"If someone isn't on the right track with something that they're doing, waiting up to a year for their next annual review isn't good for anyone involved. They will likely go through this time thinking they are doing well, leading to a nasty surprise come review time. The psychology of getting feedback shows that people generally react to these sorts of negative surprises emotionally rather than intellectually, a phenomenon known as amygdala hijacking. As a result, people are less likely to fully understand and be able to act on the feedback they are being given."

"One of the differentiating factors between a group and a team is the presence of trust."

"James Stewart, director of technical architecture at GDS, recounted the general approach for tool selection,

shown in Figure 14-4, via a quote from JP Rangaswami:

For common problems use Opensource.

For rare problems use Buy.

For unique problems use Build."

"Disallowing remote work reflects a culture that values the appearance of doing work more than the effectiveness of the actual work."
