

[← Back](#)

Aug 26th, 2024

# Coolify's rise to fame. And why it could be a big deal.

To this day, I remember deploying my first Next.js app on Vercel. It was a magical moment.

Before that, I had only known pain. From renting a server, dockerizing, setting up a proxy, SSL certificates, monitoring—you name it!

Then came Next.js and Vercel. Sure, they weren't the first to promise seamless deployments. Products like Heroku, AWS Elastic Beanstalk, etc. all tried to remove the pain from deploying software on the internet.

However, it never felt as magical as connecting my repository to Vercel and pressing that sweet, sweet deploy button.

What happened next with Next.js and Vercel is far less magical... but back to the blog post.

## Coolify Clicked

While the internet is in an AI craze, I quietly had another magical moment. This time, it wasn't fueled by a multimillion-dollar company like Google, AWS, or Vercel.

It was fueled by [Andras Bacsai](#), who turned down over 30 investors to build Coolify as a community-funded project. It is a true revolution as I will try to explain later in this blog post.

During the last weeks, I was on the lookout for the best solution to deploy the [Api-Fiddle](#) REST API. After considering [fly.io](#), a friend pointed me to [Coolify](#). The name rang familiar—I'd seen it on X before!

Coolify describes itself as:

An open-source & self-hostable Heroku / Netlify / Vercel alternative.

hours to get Coolify running on my server.

## My Coolify Moment

My Coolify moment happened right then and there.

Coolify is still in its early days and the UI can be a bit rough. But with little effort I became the CEO of my own Vercel-like, fully-featured deployment platform.

Deploying a PostgreSQL instance took three clicks and 30 seconds. Configuring a running instance of Grafana? Another 30 seconds.

When it was time to deploy my app. I pointed Coolify to my GitHub repository and specified the domain.

After a bit of fiddling my app was deployed with a database, hourly backups, and a [Caddy](#) proxy. Just like that.

Once you get the hang of it, deploying any app that is either dockerized or [Nixpacks-compatible](#) is a breeze.

This alone makes me talk about Coolify with excitement. But I believe there is more to this story.

Coolify has the potential to change how small to mid-sized projects and companies - deploy internal and external tooling.

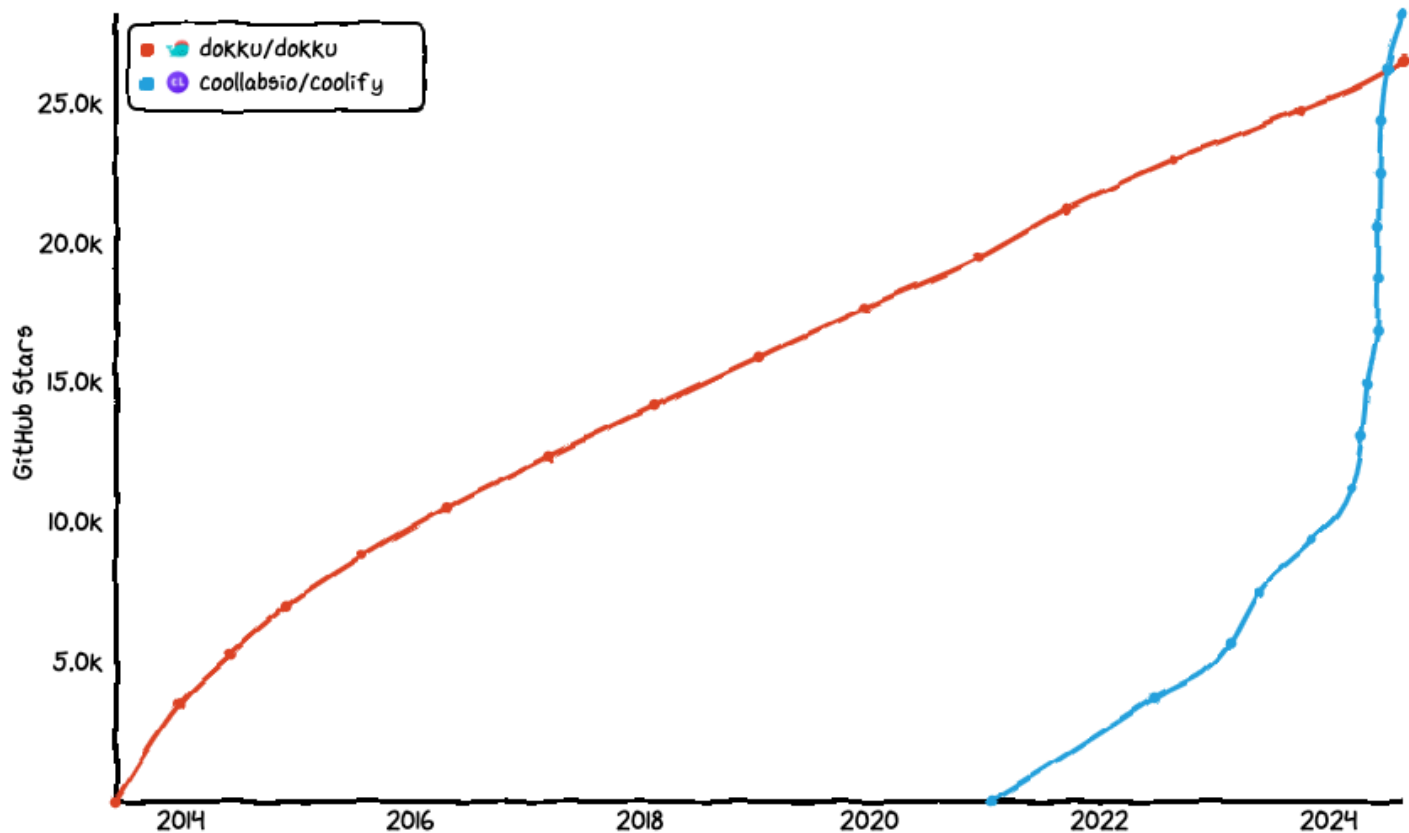
## The Rise to Fame

Looking back, I've only had one moment in my software career that felt like a turning point: My first deployment to Vercel. Now, I've had another one with Coolify. Before you disagree, let me explain.

No, Coolify isn't something entirely new. Like many new things, it's a wrapper around existing technologies. Now and then, you could piece together these technologies yourself and, depending on your skill and luck, end up with something very similar to what Coolify offers.

You can achieve this with foundational tools like Kubernetes or Docker Compose and additional layers to support rolling updates, webhooks, etc.

Additionally, Coolify isn't the first attempt to build an open-source Platform as a Service (PaaS). Tools like Dokku have been around since 2014 (Coolify began in 2021).



[Check out this graph.](#)

To paint the full picture of Coolify's rise to fame, we also need to talk about the recent rise of self-hosted software in general.

There's a wide variety of open-source, self-hostable software out there—from Calendly alternative [Cal.com](#) to email and marketing tools like [Mailcoach](#).

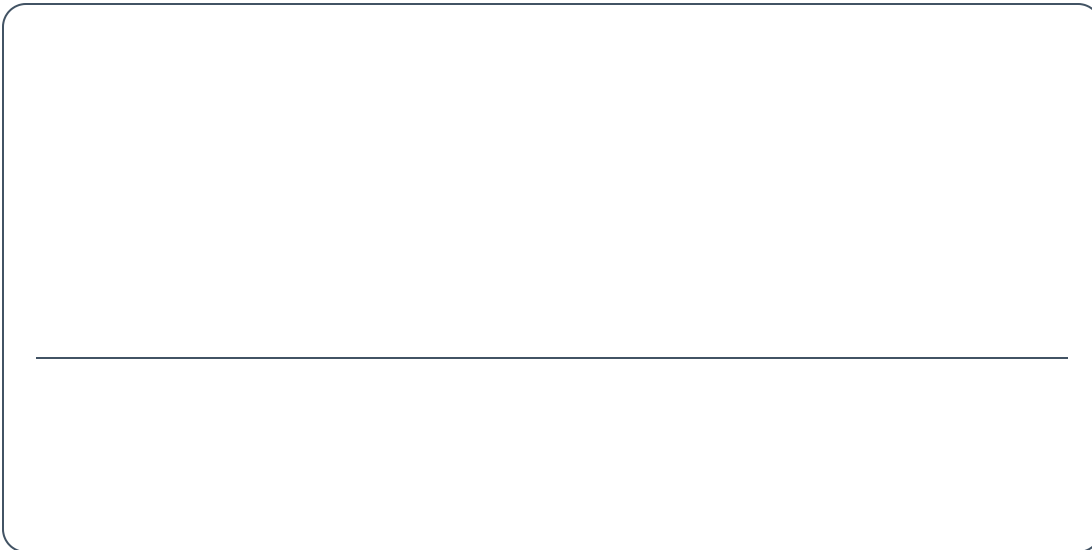
In recent years, many software companies have developed a business model around self-hosted versions of their products.

While altruism surely plays a role in open sourcing software, it can be good for business too. Small companies often open source products so they get adopted by large organizations quicker! This model is called [Commercial Open Source Software](#) (COSS).

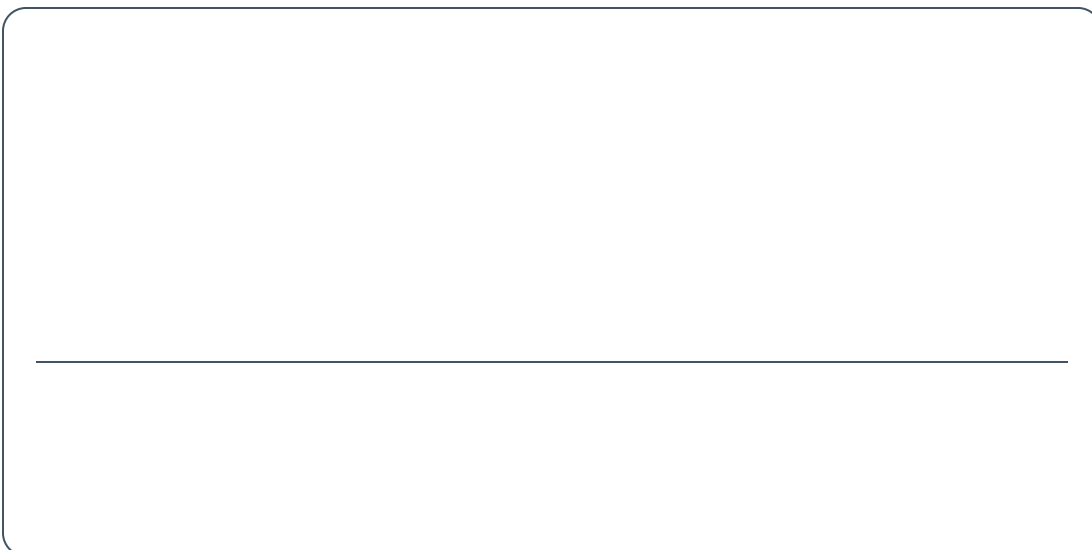
Large enterprises self-host tools and retain ownership of their data. In return, they sign a service contract with the vendor. This saves companies years of security certifications and enterprise sales cycles. All for the price of open sourcing their product that no small company in their right mind would ever self-host...**Right? Right!**

In the past, most self-hosting (aside from WordPress and a few other examples) was cumbersome for small companies. It was easier to wire \$100 to Salesforce and hope Slack will do right by them and their data.

a couple of hours (I'm not sure if he uses Coolify):



Here's another Tweet from Theo, a larger creator:



More and more developers seem to take advantage of free, self-hostable software.

My argument is this: The increasing availability of well-designed, self-hostable software combined with tools that make self-hosting trivial has the potential to shake things up.

## **Dorothy, are we still in Kansas?**

One could be tempted to call this a full circle. We came from a world of on-premise (On-Prem) Software, moved to the Cloud, and ended up in a Software as a Service (SaaS) world. Now,

While the perspective has merit, there's another way to view it. We can choose to celebrate the wealth of tools and deployment options available, allowing us to optimize for cost efficiency and data ownership.

## Final Thoughts

This is an overwhelmingly positive review of Coolify, but it's important to remember that despite its wide usage, Coolify is still in its early stages.

This article was not sponsored by Coolify, and I have no affiliation with the product other than being a user.

If you're looking to get started with Coolify, the best place to start is likely [this video](#) along with their [website](#).



## API definitions and bad APIs

So we created a better way. API-Fiddle is an API design tool with first-class support for DTOs, versioning, serialization, suggested response codes, and much more.

Try, it's free →