The top five [signs you need DevOps]

Grrrrr... *Grrrrr...*

The DevOps methodology unites software development (Dev) and production and operations (Ops) into a single, integrated, and continuous process. While Dev focuses on innovation and speed, the mandate to Ops is about stability, control, and predictability. Left to themselves, Dev and Ops struggle to talk to each other, much less collaborate. Is the ongoing battle between Development and Operations holding you back? Here are five signs you may need to bring them together with DevOps.

[1]
Your customers find major defects

A defect found by a customer in production can be 1,000 times more expensive to fix. But when there is a hard deadline looming, the application often gets thrown over the wall into production, no matter how incomplete the testing.

Your software melted my computer.

Try unplugging it, waiting five minutes then plugging it back in.

[2]
You can't do anything until you have everything

I can't start until he does.

One team waits for code being built by a second team, which is waiting on a third, which is waiting on the first. Or developers and testers need access to the production environment, available for two hours a week with data they can't touch. The results? Hurry up and wait. Then take a guess on whether the code will work as designed when it goes into production.

[3]
No one talks to each other

Because Ops, Dev, and Testing are in different groups and use different systems to manage their environments, the feedback loop that should be in place between Ops and Dev breaks down. Code is just thrown over the wall to Ops, and the production requirements are never communicated to Dev.

WELCOME TO OPERATIONS PLACE NEW WORK HERE

DEV

[4]
No matter how agile you become, going live takes as long as ever

On the other hand, I believe we just invented a cross between coding and yoga.

Agile development speeds software development. But Dev alone doesn't get the app into the hands of the customer. Because Ops often hasn't been involved in development and doesn't understand what is needed, there is a lot of trial and error in deploying to production, wasting time. And if your software release isn't automated, it makes getting a new app into production that much slower and error-prone.

[5]
Any problems lead to finger pointing

If things go wrong, does everyone scramble to show that it was someone else's fault? DevOps is all about collaborating and cooperating as a team, and pointing fingers is poisonous to a collaborative working environment.

He's not a good coder. But he's awesome at finger pointing.

If you found yourself laughing or crying at these cartoons, this is a sign you need to take immediate action! Find out how you can fix these problems at:

www.ca.com/devops-ebook

Download the eBooklet now!
Dysfunction Junction: A Pragmatic Guide to Getting Started with DevOps

ca technologies