

DevOps anti-patterns

The author

Jeffrey D. Smith is a system administrator with experience in web-based protocols, who specializes in team building, workflow automation, documentation, etc. and is the author of the book "Operations Anti-Patterns, DevOps Solutions".

What is DevOps?

DevOps is a set of practices related to software development and other functions, which joins people and processes towards the entirety of the development process. The concept is related to development and IT operations.

The author defends that DevOps is held by the following four pillars:

- Culture: The ability to change the rules.
- Automation: Making every member of the team empowered enough to make their work autonomous and not needing the 11:30 pm calls on a Friday.
- Metrics: It is vital to measure how well (or how badly) things are working. Binary results (it works, it doesn't) should not be a decisive indicator of team efficiency.
- Sharing: Team sharing knowledge is normally the best way of learning things.

Information hoax (Only Brent Knows)

Information hoarding happens whenever an employee of a company holds information to themselves without sharing it with others. Brent is someone that knows everything about the topics that are important in the organization. Information hoax happens when information sharing is not well adopted inside of the organization, such as documentation. There are two types of information hoarding: Intentional and Unintentional.

- Intentional hoarding: happens when a manager or engineer decides to keep information to himself as a source of benefit, this is mainly the result of organizational influences combined with poor incentives.
- Unintentional: people who do not realize that they are keeping information to themselves.

Every question that comes to Brent is filtered through his own perspective, which could be limiting. This concentration of knowledge is not necessarily bad, but it could generate artificial barriers around knowledge in the teams. The main points stated by the author about how to stop your team or organization from having information hoarding are:

- Allowing knowledge sharing to occur in different formats.
- Create learning rituals as an alternative to standard written documentation.
- Create structure around documentation repos to make info retrieval easy.

Alert fatigue

Alert fatigue is a phenomenon when one is exposed to many alarms and gets desensitized. Happens in many stressful fields such as medicine, air traffic controlling and IT management.

When launching a system into production, system administrators due to lack of experience, often overthink or failing to understand the ways the system might break can spend a lot of time creating alarms for all possible scenarios. This may generate a lot of noise in the alerting

system and quickly become ignored, treated as the normal rhythms. This will cause missed or ignored alerts or delayed responses.

To fight alert fatigue, you can start by reducing the number of alerts and prioritizing them based on their severity.

Non-actionable alerts should be avoided. Alerts that do not require actions will be soon ignored and tire up the worker. This fatigue comes from making a precedence to ignore them.

Insistence should fit the problem. Repeating alerts shouldn't be the way to work.

Use monitoring dependencies to avoid alert storms. Setting up dependency relationship so that alerting is suppressed if an overarching issue would cause other alerts that depend on it.

You can also automate the process of alerting and use machine learning to identify patterns and reduce false positives.

Culture by decree

The "culture by decree" antipattern happens when an organization's culture is defined by a verbose statement or a "plaque that hangs in the main lobby" but does not apply to reality, in any tangible way.

Organizations will define its culture on paper, but the culture will not exist. This way of managing can lead to a toxic culture and issues throughout the organization. For example, senior leadership may not solve any of the problems that derive from that mistaken reality. It can be thought that your organization is open, transparent, friendly, and open to hearing diverse voices. However, it could be found that people are burned out and employees are not speaking their voices or opinions because they feel intimidated.

The company's mission statement could not be aligned with the company's actions. You need to understand what your culture is to fix it. Your culture and your mission statement need to match to be attractive to new employees and customers. It is important that the company explores ways to create that desired culture and promote their values.

The empty toolbox

It is known by everyone that mistakes can happen during development and deployment of an application. Some of them may be due to human errors such as having to sometimes act as a computer executing tasks one by one, just to, for example, deploy a new update. This is why all projects should have at least a minimum level of automation.

Normally if a process is done multiple times and requires no human factor, it should be automated. In the book some factors are mentioned as benefits of efficient automation.

- Queue time: Time that a task waits for a human to interact with it.
- Time to perform: Removing the human part of executing commands not only makes it more satisfactory, but reduces the time spent, since a computer processes the tasks faster than you could even write them.
- Frequency of performance: Not only executing a task repeatedly may be disruptive, but sometimes it could get in the way of something important.
- Variance in performance: Also, it must be considered that human factors in tasks increase variation in outcomes.