


# DevOps Has Evolved Beyond Shift Left

Written by: Tim Johnson

September 15, 2022

5 min read

 Share

[Subscribe](#)

"Shift left" as a best practice for catching code issues earlier in the development cycle and thus reducing costs has been around for more than 20 years. It's a great concept, born of the waterfall era where the security group became known jokingly as the "Release Prevention Department." Developers would work hard putting together a new product only to have it blocked just prior to release, or worse, have customers find serious errors in production.

It wasn't until later that shift left entered the security space, becoming a mantra of the DevOps era at the first Rugged DevOps special interest group that met in San Francisco in 2017. That was quickly followed by the emergence of DevSecOps, where not only was testing done sooner, but security teams were 'embedded' in the process to bring their expertise to the table and speed the release of quality software.

So, has shift left delivered on this promise? Has moving security and compliance – security's close cousin – to a point earlier in the process made them automated, self-service non-events? Are security teams, developers, and

auditors joyously eating lunch together and telling war stories about the 'old days' before we all shifted left?

No, not really.

## New Data on Shift Left, Enterprise Security, and Compliance

This year's [CloudBees Global C-Suite Security Survey](#) reveals data that shows shift left hasn't delivered on its promise. For instance, while 33% of respondents say they are shifting left and 44% say they are "probably" shifting left, nearly 60% said it was a burden on developers. Shifting left is just one piece of the puzzle, however. The same study revealed that about half of executives believe compliance and security processes (56%) and knowledge related to security and compliance (47%) is what is stopping their development team from spending more time on the activities they believe should be the priority. Specifically, they believe that security (75%) and compliance (76%) requirements hinder innovation.

The attempt at shifting left is having a significant impact on both delivering software and the developer experience overall. Across the board, executives say their teams are spending more than half their time on risk and technical debt, and less than 30% of their time on innovation. So placing additional security and compliance burdens on developers will only serve to reduce the time available for value-added activities.

### How did we get to this point?

When the concept first arose, code was a monolith. Environments were well-documented and well-known, and there were only a few testing tools at our disposal. Regulatory and governance requirements changed slowly, environments changed even more slowly, and code was almost completely developed in house. Everything moved at a leisurely pace. It was relatively easy for a developer to understand the results of a test, make the required updates to code, and move on.

Nowadays, companies only write about 30% of new code. Open source makes up the bulk of applications, and environments, requirements, and standards change at a blinding rate. Add the plethora of testing tools developers are now

required to run and the need to decipher the torrent – or as we call them ‘alert storms’ – of findings from all those tools, and it is clear it is becoming impossible for one developer or team to reasonably handle these tasks. How can they understand how to respond, identify the real issues against false positives, and prioritize their work. This is one example of what we call the Compliance Tax.

## Shift Left is a DevOps Anti-Pattern

At the end of the day, **shift left in its current state has become a DevOps anti-pattern**. Instead of making things simpler, increasing flow, and making improvement easier, teams are spending more time on non-value-add work than on innovation. We have yet to meet a developer who loves trying to translate a governance policy into code, decipher the thousands of critical security alerts they’ve received, or is happy to see Slack messages from the audit team.

We could write a book (or a whole series of blogs) alone on the burden shift left causes for risk management, CISOs, and executives, too. It’s a different story but just as ugly. It’s too hard to find data, what they do have is out of date, and they can’t see the whole process or know for certain their risk posture. Suffice it to say, the Compliance Tax is looked at as a leading impediment to delivering ROI on digital transformation efforts.

## Shift Left, Done Right

Catching problems early and fixing them before they slow the process or get into production is still the ultimate goal of shift left. However, shift left needs to be reimagined with a new mindset, a new approach, and some innovative automation to deliver on the promise.

So, how can shift left be done right? We see three core attributes:

1. Security and compliance teams that declaratively state what “safe and secure” means for the organization and then map their policy prose to automation;

2. A system that runs continuously across the entire organization and software delivery lifecycle (SDLC), including production, comparing the digital estate against those policies and regulatory requirement;
3. And, a system that provides context of any detected threats or problems in relation to the stage of the SDLC, risk profile of the application, and the impact on the business' critical services.

Another way to think about it is to look at the outcomes. You'll know you've done shift left correctly when:

1. Developers are relieved of alert storms, allowing them to focus on innovation
2. Security and compliance teams improve the organization's risk posture AND speed delivery, changing them from the "department of slow" to the "department of go."
3. Executives can assess their organization's risk posture and compliance status on demand, and assert that status with confidence.

For more information on this data, [download the CloudBees C-Suite Security Survey report](#). To see "shift left, done right" in action, [book a demo](#).



---

All Blog Articles

## Stay up to date

We'll never share your email address and you can opt out at any time, we promise.

[Sign Up](#)

### Related Content



[Whoa the Woes and Fix Your Infrastructure](#)

[BLOG](#)



[CI and Jenkins](#)

[BLOG](#)



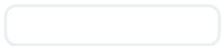
[Managing Jenkins: Consistency Isn't Boring](#)

[BLOG](#)



[Using CloudBees to Achieve a Continuous ATO](#)

[BLOG](#)



[Choosing the Right Platform](#)

[BLOG](#)



[Managed Jenkins—Where Have You Been All My Life?](#)

[BLOG](#)



## Capabilities

[Analytics](#)

[Continuous Integration](#)

[Continuous Delivery](#)

[Feature Flagging](#)

[Release Orchestration](#)

## Resources

[Blogs](#)

[Customer Stories](#)

[Podcasts](#)

[Whitepapers](#)

[Events](#)

## Why CloudBees

[About](#)

[News](#)

[Meet the Bees](#)

[Join the Hive](#)

[Contact us](#)



By clicking "Accept All Cookies", you agree to the storing of cookies on your device to enhance site navigation, analyze site usage, and assist in our marketing efforts.

[Cookies Settings](#)

[Reject All](#)

[Accept All Cookies](#)