



3 ways to manage technical debt with Pega

Best practices to keep you innovating



IT Modernization is a hot topic – and it should be. Technology is constantly changing. For enterprises though, this rapid pace of change leads to technical debt that limits their ability to keep up with the competition – and more importantly, their customers. Critical to a modernization strategy is how you will change your operations once your technology is current, all to prevent future technical debt. In this eBook, we will tackle the three things you can do to prevent a build-up of technical debt, keeping your enterprise modern and competitive.

What is technical debt?

Identifying the invisible threat

Technical debt, also called code debt, are code or configurations that don't follow current best practice based on modern software capabilities and features. Essentially, technical debt is created when you don't prioritize adopting new features as technology advances. What might have been an excellent customization to solve a particular problem may conflict with advancements that render the customization obsolete. This is also true for how your platform is built or configured. The cost of technical debt is that engineers lose time to innovate software, costing value to the business in speed or customer experience.

When operating Pega Infinity™ on-premises, or in your own cloud environments, this debt can be compounded by infrastructure and operations costs. Gartner estimates that through 2023, infrastructure and operations leaders who actively manage and reduce technical debt will achieve at least 50% faster service delivery times to the business¹.

“Engineers spend 33% of their time dwealing with technical debt.”

Alex Omeyer
“The Cost of Technical Debt,”
July 28, 2021, DZone

Developing with speed & quality

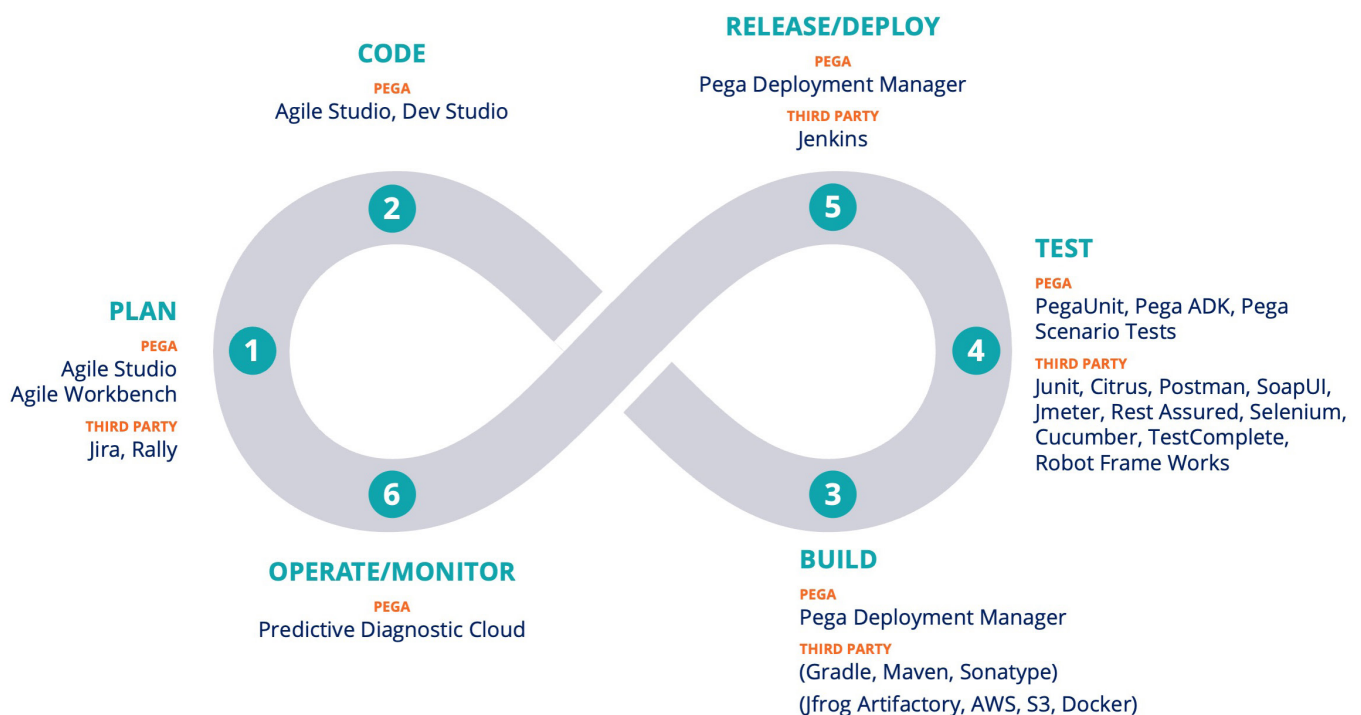
The role of DevOps in managing technical debt

Often, people speak of DevOps as a particular stage of the lifecycle, like automated testing, or a deployment pipeline. To effectively modernize your development operations, you need to embrace DevOps best practices as a key tool to aid your evolution. By adopting DevOps based on discipline, you can realize value that both eliminates and mitigates technical debt.

The full DevOps lifecycle brings development (including low-code developers) together with operations to plan changes to applications, execute the code, build the updated application, deploy it to a staging environment, fully test it, release it, and monitor your progress. Release scope becomes smaller and cycles shorter as you mature. With this model, you can quickly identify existing or potential technical debt, assess its impact, and plan to mitigate it.

DevOps best practices help you to:

- Foster collaboration between development teams and operations teams to manage the fast pace of changes.
- Apply automated testing early in the development cycles, which helps to quickly identify potentially major problems so you can fix them before they slow you down.
- Use automated pipelines to automate common tasks, freeing your development team to focus on innovation.
- Address technical debt continuously, so it doesn't get out of control.



Getting the most from software updates

Patching and updating to keep clear of technical debt

The benefit of subscription-based software licenses is access to ongoing software updates. With Pega, clients can choose to adopt a tolerance, also known as basic, update to their software or adopt a compliance update.

A basic update is not best practice because this method is a source of technical debt. While your application may run on the latest version, a basic update does not include updating changes to the rulebase of Pega Infinity and the application remains unchanged. Repeated basic updates compound both feature and code debt in applications.

Compliance updates are considered best practice for updating Pega. In this method, the rulebase is updated and the application is refactored to take advantage of new features, while eliminating deprecated or retired features.

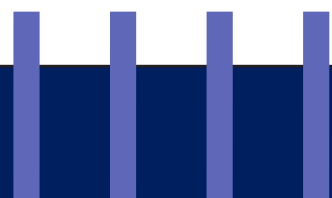
Embracing innovation

Position yourself for continuous innovation

It's an exciting time in the software industry. New technologies like generative AI are driving excitement for their potential in natural language processing – with a wide array of potential use cases. But if you are mired in technical debt, new innovations may feel out of reach.

Aligning your strategy to look to new innovations with each release of software is a forcing function that addresses things like custom code continuously to avoid technical debt. By planning innovation into your regular cycle you'll not only improve business outcomes, you'll position yourself to manage technical debt.

18% of respondents to the Forrester Analytics Business Technographics® Software 2 Survey, 2021, said that their biggest software challenge is that technical debt is taking energy away from innovation.



Managing customization

How refactoring for out-of-the-box capabilities eliminates technical debt

Sometimes the market can outpace technology advancements. In cases like these, Pega developers have the freedom to create customizations to satisfy requirements. This solves an immediate need and creates value to the enterprise at the time of release. However, as Pega continues innovation, these customizations become a major source of technical debt.

When Pega adds or enhances features to the Pega Platform™, we fully test these against our full breadth of capabilities, features, and products. Because customizations are our client's intellectual property, they are not tested against future releases.

When out-of-the-box features are released to meet a requirement previously satisfied by a customization, application refactoring can replace customizations – rendering your application both more stable and more updatable.

A 2015 customization that a European organization built to satisfy GDPR eventually caused extensive memory usage, causing slow application performance. Refactoring the application to Pega's subsequently released out-of-the-box capabilities eliminated performance challenges.

Conclusion

As most enterprises move to modernize their IT stack, technical debt is a major factor impeding innovation. To fully modernize, business and IT can embrace a modern operating model that incorporates DevOps best practices, regular application refactoring to take full advantage of software updates, and replace required customizations with out-of-the-box features when they become available.

Summary

Whether we call it technical debt or “continuous application health,” it’s an essential subject affecting agility, innovation, and ultimately, value.

- Technical debt exists and discussing it is not a criticism. The only criticism should be pretending otherwise.
- The MLP/MVP delivery method is great at delivering value early, but we need to be careful not to over prioritize agility over quality. When an MLP has been delivered to provide early value, the associated technical debt should be documented and addressed when appropriate.
- Make deployment choices that are in the wider best interest of the organization. Tactical choices made without context in the wider business risk Pega being perceived negatively, with larger consequences that dwarf any benefits seen.

Final thought... Consider how much technical debt is appropriate to carry. On one side you have low technical debt with ease of change, on the other is high debt that is much harder to change. The most important thing to do is to start somewhere and make managing technical debt a conscious decision that is weaved into your development culture.



About Pegasystems

Pega provides a powerful low-code platform that empowers the world’s leading enterprises to Build for Change®. Clients use our AI-powered decisioning and workflow automation to solve their most pressing business challenges – from personalizing engagement to automating service to streamlining operations. Since 1983, we’ve built our scalable and flexible architecture to help enterprises meet today’s customer demands while continuously transforming for tomorrow.

For more information, please visit us at www.pegasystems.com