

THERE ARE UNICORNS ON THE MAINFRAME



I just finished reading the "[The Unicorn Project: A Novel About Developers, Digital Disruption, and Thriving in the Age of Data](#)" by Gene Kim. If you are not familiar with the book, the story follows Maxine and Kurt, employees of a car parts retailer and manufacturer named Parts Unlimited. After a production issue, Maxine is reassigned to the Phoenix Project, a massive transformative project that spans almost every platform in the company. When Maxine tries to do a build on the Phoenix Project and is unable to get it working, she is introduced to Kurt and the Rebellion. The Rebellion is a group of engineers within IT trying to transform the company and the way they do work. In working with the Rebellion, Maxine and Kurt lead a total transformation of Parts Unlimited by focusing on the "Five Ideals."

- The First Ideal: Locality and Simplicity
- The Second Ideal: Focus, Flow, and Joy
- The Third Ideal: Improvement of Daily Work
- The Fourth Ideal: Psychological Safety
- The Fifth Ideal: Customer Focus

At first glance, you might think that the book would advocate removing older legacy platforms like the mainframe, but that is not at all the case. The book is about modernizing, improving the developer experience, and eliminating technical debt. All of these goals are possible and part of the modern mainframe.

First let's tackle the idea of "modernizing the mainframe." I do not like this term, since it implies that

as a platform, the mainframe is not modern. I am here to say that is not the case. The mainframe is a modern and vibrant platform. With the release of IBM z16[®], as well as Db2[®] v13, the platform is as modern as any cloud, web, or mobile platform. Enhanced capabilities around artificial intelligence and machine learning (AI/ML), quantum computing, and hybrid cloud are just a few of the advancements built into the latest release of the mainframe, bringing it in line with any modern platform today.

Instead, the discussion should really be focused on modernizing application development and architecture on the mainframe, and modernizing the development environment. This may seem like a subtle distinction, but it is a distinction nonetheless. Modernizing the application development space is absolutely necessary and in line with the message in "The Unicorn Project." The most business-critical applications run on the mainframe, so updating and modernizing them is essential to becoming a high-performing organization.

Let's look at modernizing application development through the lens of "The Unicorn Project" and the Five Ideals.

The First Ideal: Locality and Simplicity

Simply stated, this ideal relates to the degree to which a development team can make code changes in a single location without impacting various teams. Teams should be able to make simple changes in a single place and test them without affecting or involving other application teams.

I think of all the Five Ideals, this one could potentially be the most difficult to achieve for mainframe application teams. Over the years, many mainframe applications have become very interwoven with other applications, and it's become increasingly difficult to make simple changes given their size and complexity. Refactoring and simplifying these applications will be challenging. Many of the subject matter experts who wrote these applications are either retired or close to retirement, so the expertise on these systems is dwindling.

The immediate task of refactoring and simplifying applications can seem very daunting, but the payoff for that work will set application teams up to reap the benefits of DevOps, and more importantly, make things simpler to maintain and change for the most critical applications in an organization. And by doing so, organizations will also modernize and simplify your most critical applications and make working on them easier and more fulfilling.

The Second Ideal: Focus, Flow, and Joy

This ideal is all about how daily work feels. To boil it down into a couple of words: developer experience. How does it feel to come into work each day and try to do your job? Is it a joy, are you working with top-of-the-line systems and processes that aid in getting work done, or is it a challenge each day? Does it take days or weeks waiting on processes or system to be available?

Focus, flow, and joy make your work a delight and help organizations attract and retain top talent. Like the first ideal, this one has fallen by the wayside for a lot of mainframe application development. As an example, many teams are still using green-screen technology and following older waterfall processes that make it more difficult to work. This need not be the case.

Mainframe developers can have the same focus, flow, and joy as their distributed counterparts. Modern integrated development environments (IDEs) like [BMC AMI DevX Workbench for Eclipse](#) or

Visual Studio Code are available for doing day-to-day work. Mainframe source code management and deploy systems like [BMC AMI DevX Code Pipeline](#) are built with agility and the developer experience in mind. For those application teams who can and want to make the transition to Git, ISPW fully integrates with Git systems. Achieving focus, flow, and joy will bring new life to existing developers and attract an entirely new generation onto the mainframe.

The Third Ideal: Improvement of Daily Work

While this ideal sounds a lot like things we touched on in the first and second ideal, this is a different concept. This ideal is about paying down technical debt that has built up over the years and re-architecting your applications to be more efficient. The first ideal is about decoupling applications so they're easier and simpler to change. The second ideal is about making it easier for developers to do their work.

For the third ideal, technical debt needs to be treated as a priority and paid down and your architecture needs to be modernized so your development teams can flow and push out changes faster and easier. This can be a difficult proposition for mainframe teams. For years, technical debt on the mainframe has been ignored and allowed to build up, so it will take a lot of time and effort to dig back out of that hole, but it is well worth it. By improving your daily work on the mainframe, you make it approachable, easy, and smooth to deliver value. This is a necessary step for the advancement or development on the mainframe.

The Fourth Ideal: Psychological Safety

This is the idea that people should feel safe to speak up about issues, concerns, and problems. If teams do not feel they can bring up issues without fear of repercussions, then no problems or concerns will be discussed and issues will continue to arise and never get fixed. An open dialogue is one of the top indicators of team performance and is essential to building a high-performance team. Regardless of the platform, organizations need to create a culture of open communication at all levels to be able to confront hard problem and resolve issues. This is true on the mainframe or any other platform.

The Fifth Ideal: Customer Focus

This ideal highlights the idea of "[context and core](#)," as described by Geoffrey Moore. Core is what customers can and are willing to pay for—the center of your business. Context is the systems that customers do not care about—backend systems like marketing or HR. They are very important to the business and should be treated as such, but they are not essential to the customer or the customer experience.

Context should never kill core. Your focus should always be on the systems that face and are important to the customer. In almost all of the organizations we talk with, mainframe systems are core. They are the center of the business and essential to the customer experience. By focusing on mainframe applications and modernizing the development experience, organizations are keeping the fifth ideal.

The Five Ideals in "The Unicorn Project" are not exclusive to the four "technology stock leaders—Meta, Amazon, Netflix, and Alphabet. They are for any organization, on any platform, with any application, that wants to make a change and become a high-performing organization.

Mainframe is just another box in the datacenter and the applications on it are like any others that need to be modernized and reconstructed to stay relevant in today's ever-changing market.