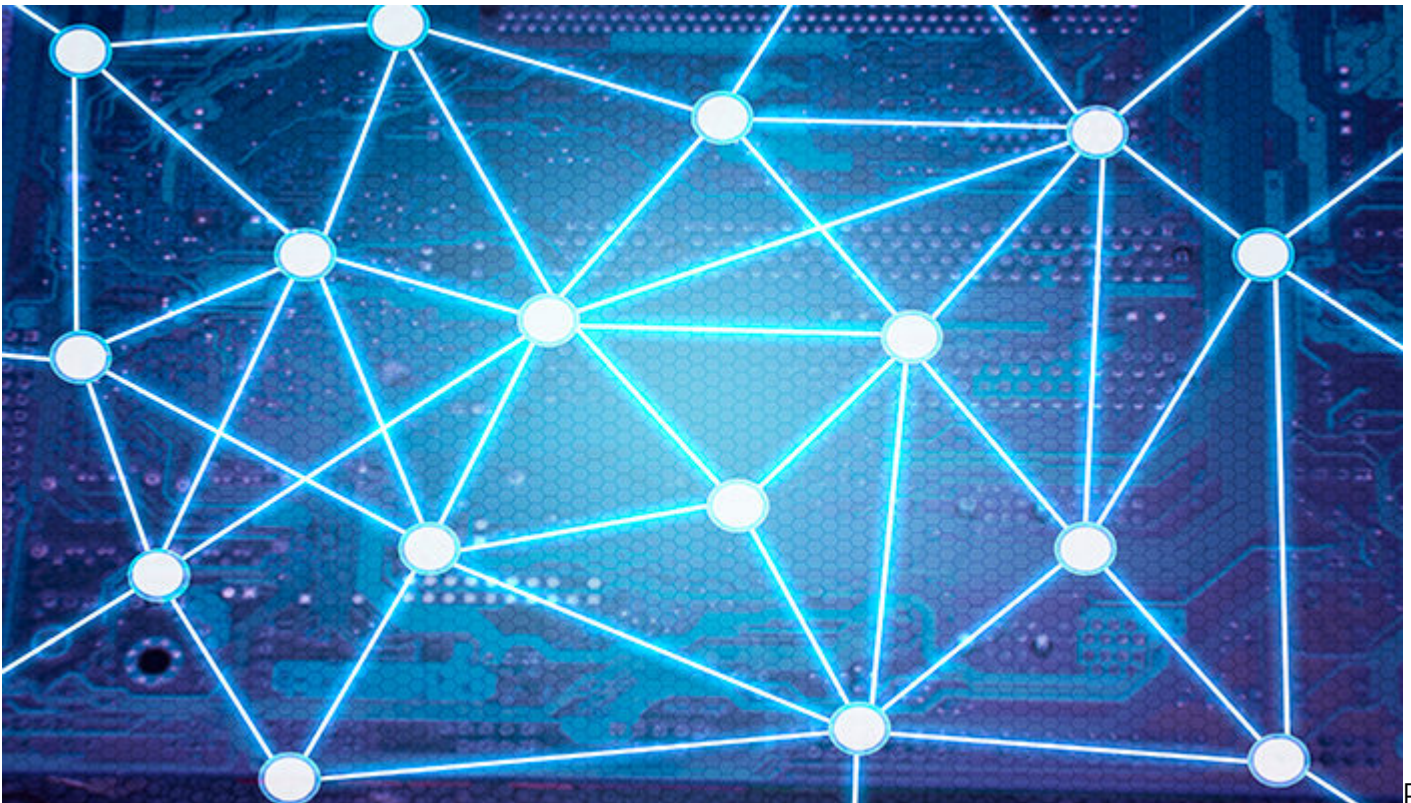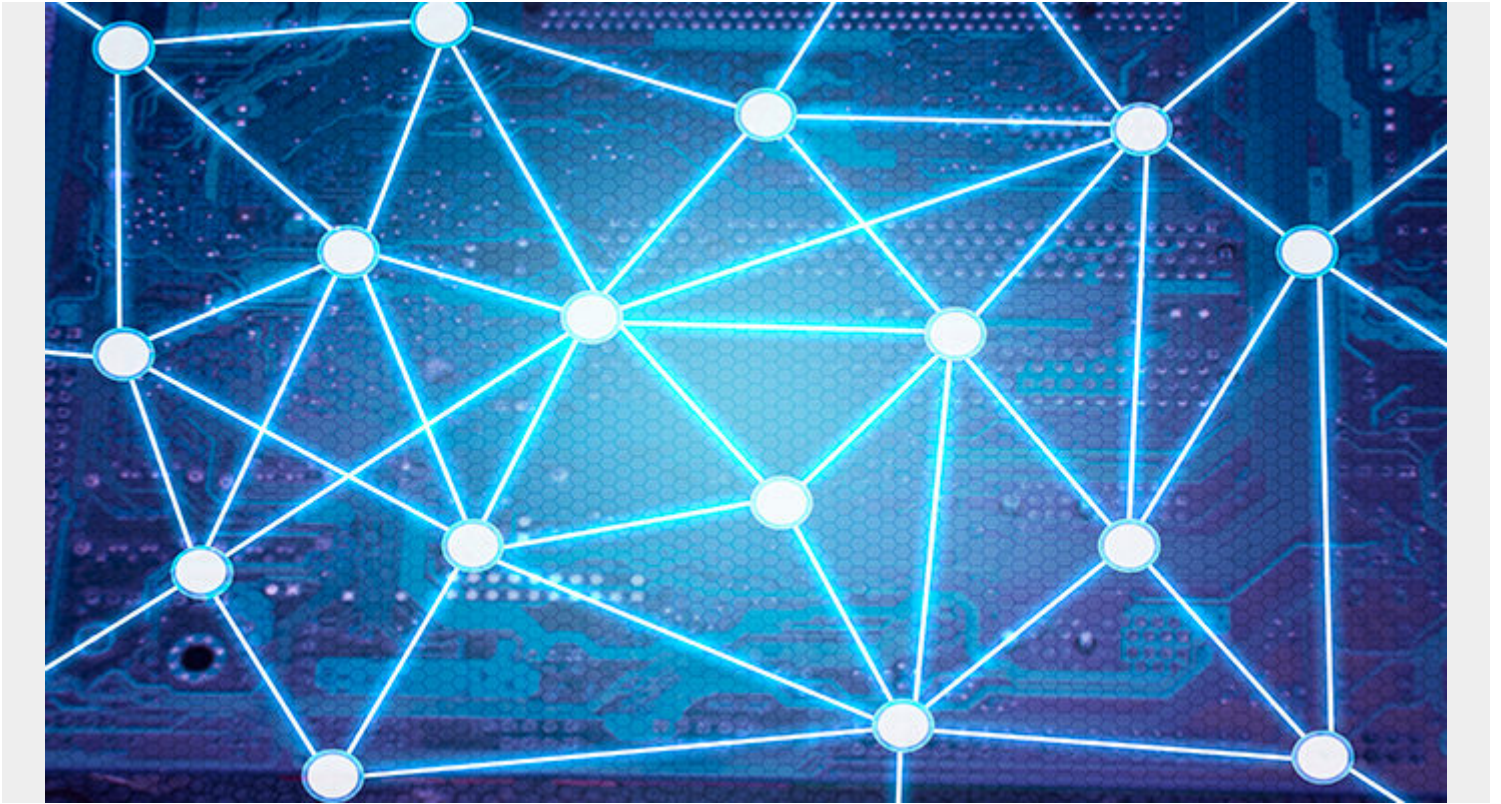# HOW PAYPAL SUPERCHARGES APP DEVELOPMENT WITH DEVOPS AND CONTROL-M

PayP

al Highlights:

- $13,500+ in payments processed each second
- 2,600+ applications
- 4,500+ developers
- 42,000 batch executions per day

With 210+ million active users (in the highly competitive global financial services market), PayPal must constantly find new and innovative ways to help their customers connect and transact. Yet, the highest levels of quality, security and compliance must be maintained. How do they do it? They've embraced DevOps and created self-service culture that empowers developers to make better applications faster.

Last November, Rama Kolli, a product manager for PayPal (@PayPal) spoke to attendees at the 2017 DevOps Enterprise Summit in San Francisco (@DOES_USA) about how the company has dramatically sped up application development. PayPal's story is filled with valuable takeaways for any company facing the same challenges. Here's some of what I learned.

A few years ago, a PayPal developer's life was all about logging support tickets.

- If a developer needed an estimate for production capacity, they logged a ticket.
- If they needed certifications for a new app, they logged a ticket.
- If they needed a release window, they logged a ticket.

You see where I'm going. It took **days** to create a new application, **weeks** to deploy a test server, and **months** to move the app into production. Developers were drowning in tickets and struggling to keep up with the innovation the business required. The process was clearly broken.

So, PayPal joined the DevOps revolution and built a custom self-service platform designed to give developers full control over every phase of the Software Development Lifecycle (SDLC) while still adhering to corporate standards. This approach completely shifts the developer's day to day activities from the chaos of support tickets to a one-stop shop for all their requests.

One area of this self-service lifecycle that I am going to focus on is scheduling.
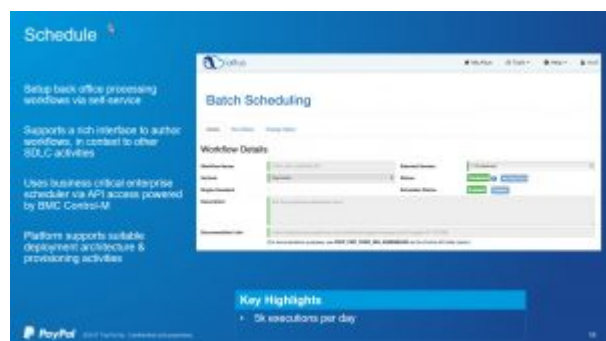
PayPal leverages BMC's Digital Business Automation Platform, Control-M, to enable developers to orchestrate their batch workflows. Developers already have self-service access to build their batch applications. It's been so successful, PayPal is building a self-service experience for scheduling batch jobs using Control-M Automation API.

Because batch applications are integrated between Control-M and their self-service platform, agents are provisioned along with the applications and get propagated to all the different

environments that batch apps are deployed to. And by using the Control-M's feature-rich editor, developers can define their batch schedules, execute dry runs of the schedules, and publish the schedule for continuous execution. This means that batch definitions are completed much earlier in the SDLC, reducing the amount of rework needed in the release and deploy phase while increasing the quality of the app in production.

So, what's the end result?

PayPal is building and deploying applications faster than ever – in some cases in less than two weeks. To date, over 1,000 applications have been built in their self-service platform. Developers feel empowered and can actually focus on development, not logging and following up on tickets. The rate at which change is deployed to production is no longer bound by platform limitations. Now it's all about the speed at which developers can write their own code, certify it and deploy it.

Rama also shared a few lessons that PayPal learned along the way.

1. **Site Operations is a key partner in the DevOps journey**. Increasing development agility with self-service capabilities can never come at the expense of site stability. It's important to work directly with your site operations partners to make sure their concerns are addressed and that their tools are built into the platform.
2. **DevOps raises the bar for development**. With self-service, developers get used to a much faster work pace and no longer tolerate delays. This makes it easy to identify bottlenecks and opportunities to invest in automation.
3. **Freedom isn't free**. PayPal's 4,500 developers have built over 12,000 test playgrounds. If these test instances are not being used by developers efficiently, there's a significant cost to the business. It's important that DevOps platforms give developers visibility into the cost of the resources they have provisioned.
4. **Invest in the user experience (UX) to lower support costs**. The more intuitive you can make your platform, the more developers are able to be self-sufficient. Pain points in the UX will quickly show up in the form of added service costs. Strive to create a unified, user-friendly experience and you'll keep service and training costs down.
5. **Productivity is key**. PayPal is investing in tracking capabilities to ensure that the DevOps platform is available to developers whenever they need it and that they can complete tasks as fast as possible. Availability and usability will drive productivity and efficiency.

DevOps has revitalized the way PayPal approaches development, and BMC is proud to be a partner on that journey. I hope these insights are helpful as you navigate your DevOps journey. Are you currently implementing or planning to implement DevOps strategies and tools at your company? I'd love to hear your feedback and questions!

[Click here](#) to watch Rama Kolli's entire presentation '*Supercharging Application Development at PayPal to Democratize Financial Services*"