

The Workflow of Disciplined DevOps

Defining DevOps

Why Disciplined DevOps

The Workflow of Disciplined DevOps

DevOps Strategies Throughout Disciplined Agile®

Adopting Disciplined DevOps

Critical success factors

As an industry we are nowhere near agreement as to what DevOps really is, and there are several complimentary visions. Let's work through these visions one at a time so as to build to a coherent vision for Disciplined DevOps that addresses the challenges faced by modern organizations. These visions are:

- The BizDevOps Vision
- The DevSecOps Vision
- The DataDevOps Vision
- Making Release Management and Support Explicit
- Disciplined DevOps

1 The BizDevOps Vision

First and foremost, a key improvement over the basic DevOps vision is to explicitly bring your customers into the picture. This is something that is commonly referred to as “BizDevOps” (or BusDevOps) and its workflow is depicted in Figure 1. There are two important differences in this diagram:

- We're clear that DevOps isn't just for teams following an agile or continuous delivery lifecycle but is potentially applicable to any team following a lifecycle that supports incremental delivery. Having said that, a continuous delivery approach to development is certainly preferred.
- The workflow includes Business Operations which are the activities of delivering of products and services to your organization's customers. There is little value in having a responsive IT organization if the rest of your enterprise isn't able to take advantage of it. BizDevOps seeks to streamline the entire value stream, not just the IT portion of it.

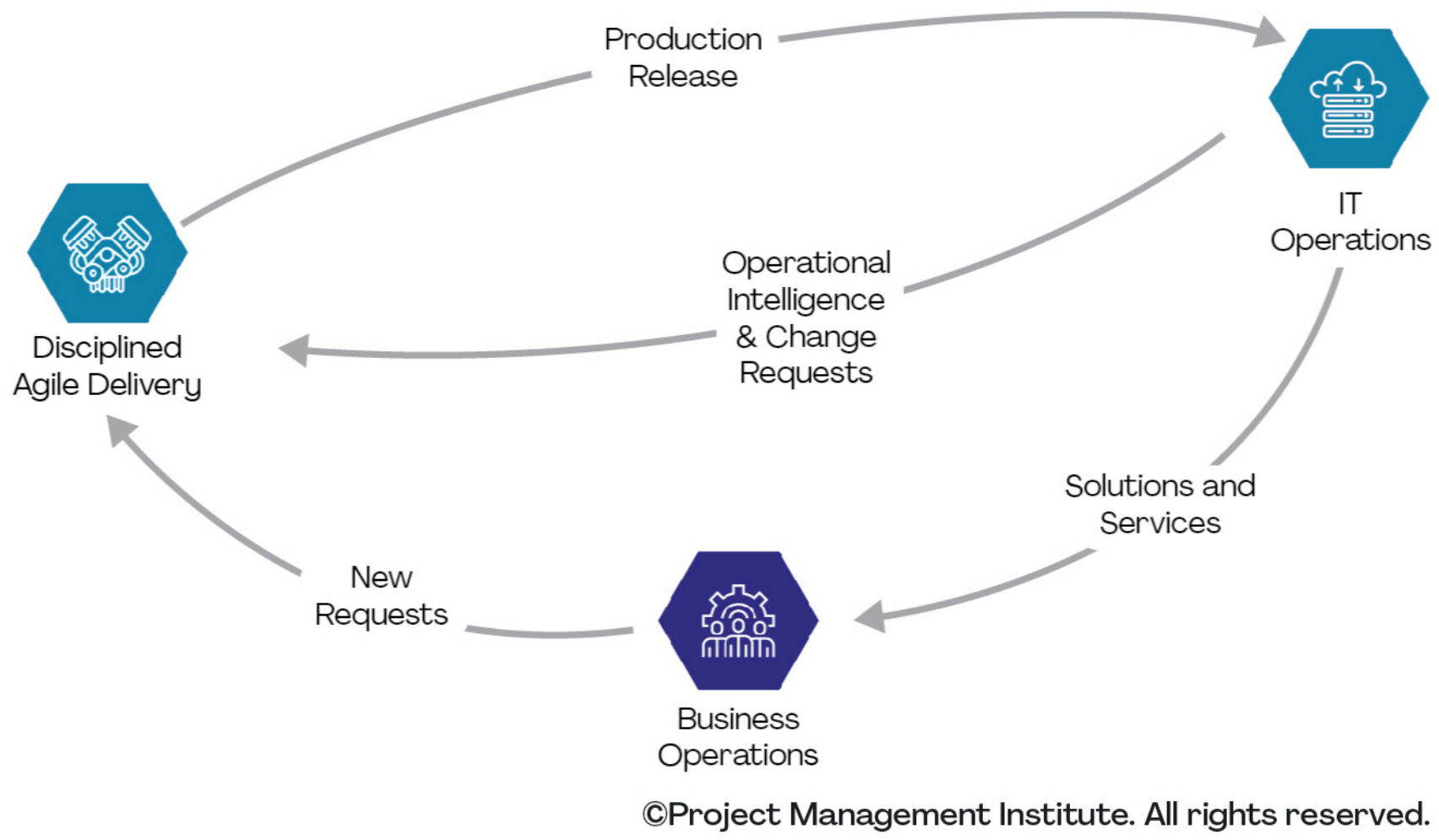


Figure 1. The workflow of “BizDevOps”

2 The DevSecOps Vision

Now let's go in a different direction. Another common improvement over the basic DevOps vision is something called DevSecOps, the workflow for which is overviewed in Figure 2. The goal of DevSecOps is to ensure data security through improving the awareness and understanding of security issues, by adopting proactive security practices, and by incrementally identifying and addressing the most urgent security gaps [DevSecOps]. Security strategies that support DevOps includes collaborative security engineers, exploit testing, real-time security monitoring, and building “rugged software” that has built-in security controls. In some ways security and DevOps have competing goals — Security wants to keep everything safe where DevOps wants to enable quick, responsive changes to the marketplace. Ensuring safety will slow things down yet being responsive increases the chance of inadvertently introducing security holes. Because DevOps will never be a replacement for formal security practices it is important to find a viable middle ground as provided by DevSecOps.

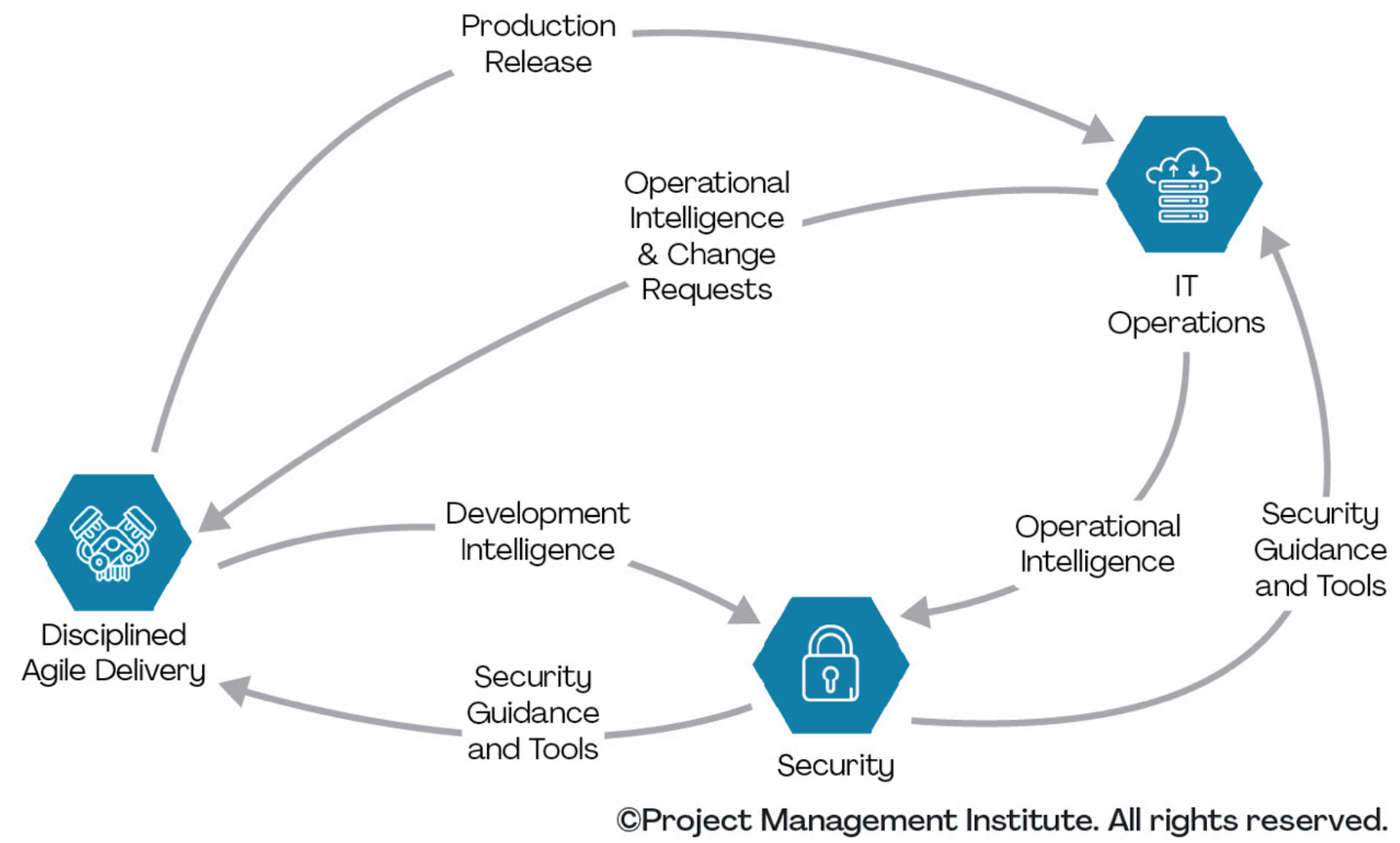


Figure 2. The workflow of DevSecOps

3 The DataDevOps Vision

Similarly, Data Management is often missing from the DevOps picture. To our knowledge no one has coined the term “DataDevOps” so we'll do so for the sake of expediency. The goal of DataDevOps is to ensure a fair balance between the competing needs of data management in that it wants to provide timely and accurate information to your organization and DevOps in wanting to be responsive to the marketplace. Where DevSecOps is a safety vs flexibility tradeoff, DataDevOps is an accuracy vs. flexibility issue. The DataDevOps workflow is depicted in Figure 3. Supporting data management activities include the definition, support, and evolution of data and information standards and guidelines; the creation, support, evolution, and operation of data sources of record within your organization; and the creation, support, evolution, and operation of data warehouse (DW)/business intelligence (BI) solutions.

The workflow of DevDataOps

Figure 3. The workflow of DataDevOps

4 Making Release Management and Support Explicit

Let's consider a fourth viewpoint to extending DevOps. Some people choose to include Release Management and Support (help desk) activities in with the IT Operations efforts. Although this is a perfectly fine decision, our experience is that in doing so you risk downplaying these important activities. Furthermore, in large organizations Release Management can become a critical activity. When you have a handful of delivery teams coordinating their releases, it is straightforward. But when there are dozens or even hundreds of solution delivery teams working in parallel it can be very challenging to ensure that their releases go smoothly. In Figure 4 we explicitly call out Release Management, IT Operations, and Support to make the workflow clear.

The workflow of DevOps with explicit Release Management and Support

Figure 4. The workflow of DevOps with explicit Release Management and Support

5 The Disciplined DevOps Vision

We're now in a position to understand what a Disciplined DevOps strategy actually entails. Figure 5 depicts the workflow of Disciplined DevOps, which is a combination of the workflows of Figures 1 through 4. Our point is that the BizDevOps, DevSecOps, and DataDevOps strategies all have their merits as does the strategy of making Release Management, Support, and IT Operations explicit.

The workflow of Disciplined DevOps

Figure 5. The workflow of Disciplined DevOps

Share Share Tweet Share

Related Resources

- Citizen Developer
- Disciplined DevOps Layer
- Manifesto for Agile Software Development
- Solutions for Software Developers
- What is a process blade?