**Get started with Azure DevOps**

* 25 min
* Module
* 5 Units

 4.7 (7,005)

Beginner

DevOps Engineer

Administrator

Developer

Solutions Architect

Azure DevOps

Explore what DevOps is (and isn't) and learn how to get started with Azure DevOps.

**Learning objectives**

In this module, you will:

* Learn what DevOps is and identify what separates elite performers from low performers
* Identify what services Azure DevOps provides
* Set up an Azure DevOps organization

**Introduction**

* 1 minute

Azure DevOps can help your team release code in a more efficient, cooperative, and stable manner. Let's rejoin the team as they learn about DevOps. Together, we'll discover what it takes to get an Azure DevOps practice started.

In the previous module , you met the team at Tailspin Toys. They're working on a new game but they've got many problems with their release process. These problems are affecting the quality of the products they deliver to their customers. The team knows they need to change but they don't know how.

The newest team member, Mara, believes DevOps will help. Her goal is to convince her teammates. She'll explain more about the value stream mapping (VSM) exercise she started in the previous module. Hopefully, her explanation will show the team why DevOps is the way forward.

**Learning objectives**

In this module, you will:

* Learn what DevOps is and identify what separates elite performers from low performers
* Identify what services Azure DevOps provides
* Set up an account in Azure DevOps

**Prerequisites**

The modules in this learning path form a progression. We recommend you start at the beginning of the Evolve your DevOps practices  learning path before you start this module.

**Meet the team**

You met the *Space Game* web team at Tailspin Toys in the previous module. As a refresher, here's who you'll work with in this module.



Andy is the development lead.



Amita is in QA.



Tim is in operations.



Mara just joined as a developer and reports to Andy.

Mara has prior experience with DevOps and is helping the team adopt a more streamlined process by using Azure DevOps.

# What is DevOps?

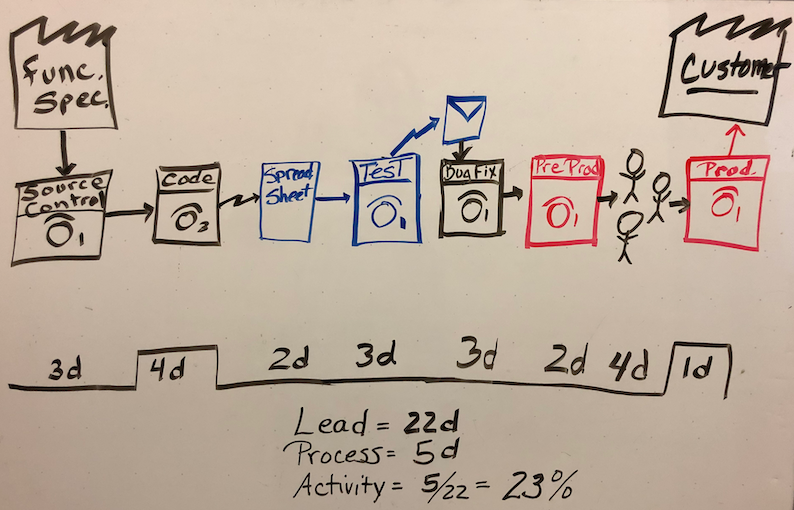
* 5 minutes

DevOps is the union of people, process, and products to enable continuous delivery of value to our customers. But what exactly does that mean? Let's join the team as Mara explains what DevOps is, what it isn't, and what makes elite performers successful.

Mara has called a short meeting with her teammates. Everyone has shown up but no one wants to be there. She's put a box of donuts on the table.

**Mara:** Hi, thanks for coming. I wanted to talk more about our value stream map and how we can make our processes more efficient.

Mara's value stream map is still on the whiteboard from their previous meeting:



**Mara:** Our value stream map shows where we lose efficiency in delivering value to our end users. Just like everyone else, we can improve. And we can decide which areas to tackle first.

**Andy:** This shows us where we have problems but not what to do about them.

**Mara:** Right, it's an exercise that helps point us in the right direction. As for what to do about our problems, I think DevOps will help us. At my last company, our deployment rates went way up, lead times were much faster, and operations had far fewer incidents. It took us a while to get there but it was worth it. DevOps isn't a quick fix.

**Tim:** I know someone who just got a job as a DevOps engineer. I think it's more for developers. That sounds like you, Andy.

**Mara:** DevOps isn't a job title.

**Amita:** Is there any software program we can get that will help us along, or a template? Maybe there's a DevOps spreadsheet.

**Mara:** DevOps isn't a piece of software.

**Andy:** It's more like a methodology.

**Mara:** Not really.

**Andy, Amita, Tim**: So what is it?!

**Mara:** Here's the definition I like to use:

DevOps is the union of people, process, and products to enable continuous delivery of value to our end users.

In fact, Abel Wang, a Cloud Advocate at Microsoft, has a great set of videos with quick answers to some of our big questions. Let's see how Abel defines DevOps.

**Ask Abel**

Our goal is to give our customers games they'll love. We do that by working together with a shared set of practices and tools.

**Amita:** What does that mean? What shared practices? What shared tools?

**Mara:** Here's what I mean by practices:

* **Agile planning**. Together, we'll create a backlog of work that everyone on the team and in management can see. We'll prioritize the items so we know what we need to work on first. The backlog can include user stories, bugs, and any other information that helps us.
* **Continuous integration (CI)**. We'll automate how we build and test our code. We'll run that every time a team member commits changes to version control.
* **Continuous delivery (CD)**. CD is how we test, configure, and deploy from a build to a QA or production environment.
* **Monitoring**. We'll use telemetry to get information about an application's performance and usage patterns. We can use that information to improve as we iterate.

**Amita:** I don't know about automated testing. My tests are manual and I do them after Andy hands off the code to me. I don't have time to change how I do everything.

**Tim:** There's no way I'm letting any of you deploy to production.

**Andy:** This will scare management. They never think further than the next release and they always want it yesterday.

**Mara:** I know what you mean about management. I put together this handout about what makes an elite performing team.

## What makes an elite performing team?

Here's the handout that Mara prepared. The information is based on DevOps research reports and surveys conducted with technical professionals worldwide.

DevOps helps companies experiment with ways to increase customer adoption and satisfaction. It can lead to better organizational performance, and often to higher profitability and market share.

It uses metrics to create four categories by which to compare elite performers with low performers.

Elite performers:

* **Deploy more frequently**

In fact, some teams deploy up to dozens of times per day.

Practices such as monitoring, continuous testing, database change management, and integrating security earlier in the software development process help elite performers deploy more frequently, and with greater predictability and security.

* **Reduce lead time from commit to deploy**

Lead time is the time it takes for a feature to make it to the customer. By working in smaller batches, automating manual processes, and deploying more frequently, elite performers can achieve in hours or days what once took weeks or even months.

* **Reduce change failure rate**

A new feature that fails in production or that causes other features to break can create a lost opportunity between you and your users. As high-performing teams mature, they reduce their change failure rate over time.

* **Recover from incidents more quickly**

When incidents do occur, elite performers are able to recover more quickly. Acting on metrics helps elite performers recover more quickly while also deploying more frequently.

How you implement cloud infrastructure also matters. The cloud improves software delivery performance, and teams that adopt essential cloud characteristics are more likely to become elite performers.

Outsourcing can save money and provide a flexible labor pool, but it must be used in the correct areas. Low-performing teams are more likely to outsource whole functions (like testing and operations) than their high-performing counterparts.

### The bottom line

DevOps is a key reason many elite performers are able to deliver value to customers, in the form of new features and improvements, more quickly than their competitors. In this short video, Abel explains why you should learn more about DevOps.

**Ask Abel**

## What DevOps is not

When considering what DevOps is, it's also important to make sure we understand what it's not. DevOps is not:

* A methodology
* A specific piece of software
* A quick fix for an organization's challenges
* Just a team or a job title (although these titles are reasonably common in the industry)

**What is Azure DevOps?**

* 4 minutes

Azure DevOps provides several tools you can use for better team collaboration. It also has tools for automated build processes, testing, version control, and package management. That's quite a bit to cover! We'll get to all the tools eventually. For now, let's follow the team as they begin with an overview of what Azure DevOps is and how they can get started.

**Mara:** Amita asked about tools, and I'm proposing we use Azure DevOps.

**Andy:** How can we use something for Azure if we're not deploying to the cloud? Plus, we deploy to Linux. Does that matter?

**Mara:** These tools are great whether you're in the cloud or on-premises. It also doesn't matter if we're deploying to Linux or Windows or another platform. Azure DevOps is a suite of services that provide a solution for anyone who wants an enterprise-grade tool chain. Those tools will help us implement all the practices we just talked about. Here's what you get:

| **TABLE 1** | |
| --- | --- |
|  |  |
| Azure Boards | Azure Boards . These are agile tools that help us plan, track, and discuss our work, even with other teams. |
| Azure Pipelines | Azure Pipelines . These will let us build, test, and deploy with CI/CD that works with any language, platform, and cloud. |
| Azure Test Plans | Azure Test Plans . These are manual and exploratory testing tools. |

Those are the three I was thinking about using right now. There are two other services we can think about later.

| **TABLE 2** | |
| --- | --- |
|  |  |
| Azure Repos | Azure Repos . These provide unlimited, cloud-hosted private, and public Git repos. |
| Azure Artifacts | Azure Artifacts . These let us create, host, and share packages. |

**Amita:** This sounds like a lot. Where do we start?

**Mara:** Let's just try to do some planning by using Azure Boards. We'll see how that goes. You don't have to use every service Azure DevOps offers. You just use what you need.

**Tim:** What do we have to do?

**Mara:** It's easy. All we do is set up an account and an organization. The whole process only takes a couple minutes.