DEVOPS TEAM STRUCTURE

The focus on products over projects is one hallmark of digital transformation. And as companies seek to be quicker in responding to evolving customer needs as well as fend off disruptors, the need to better manage the end-to-end product lifecycle has become a crucial differentiator.

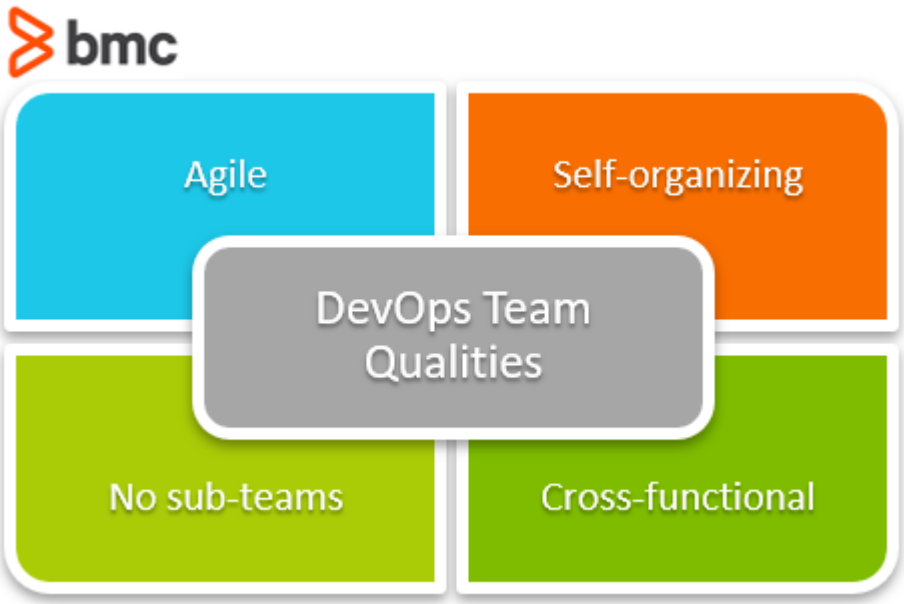
In recent times, DevOps has emerged as the go-to approach for organizing product teams in order to gain visibility over the end-to-end value chain, resulting in enhanced delivery speed and improved quality, among other benefits. The 14th annual state of agile report reported that:

- 76% of respondents currently have a DevOps initiative or are planning one in the next 12 months.
- 90% of respondents said DevOps transformation was important in their organization.

If DevOps is to deliver value to organizations, how should they structure their teams to gain the most from this approach? Is there a standard formula that all should adhere to, or is it adaptable to the specific and unique needs of the organization?

This article attempts to address this topic.

(This article is part of our DevOps Guide. Use the right-hand menu to navigate.)



Qualities of a DevOps team

A DevOps team mindset differs from traditional IT or scrum teams as it is an engineering mindset geared towards optimizing both product delivery and product value to the customers throughout a product's lifecycle.

For this mindset to deliver value, the team has to adopt the following qualities as defined by DASA:

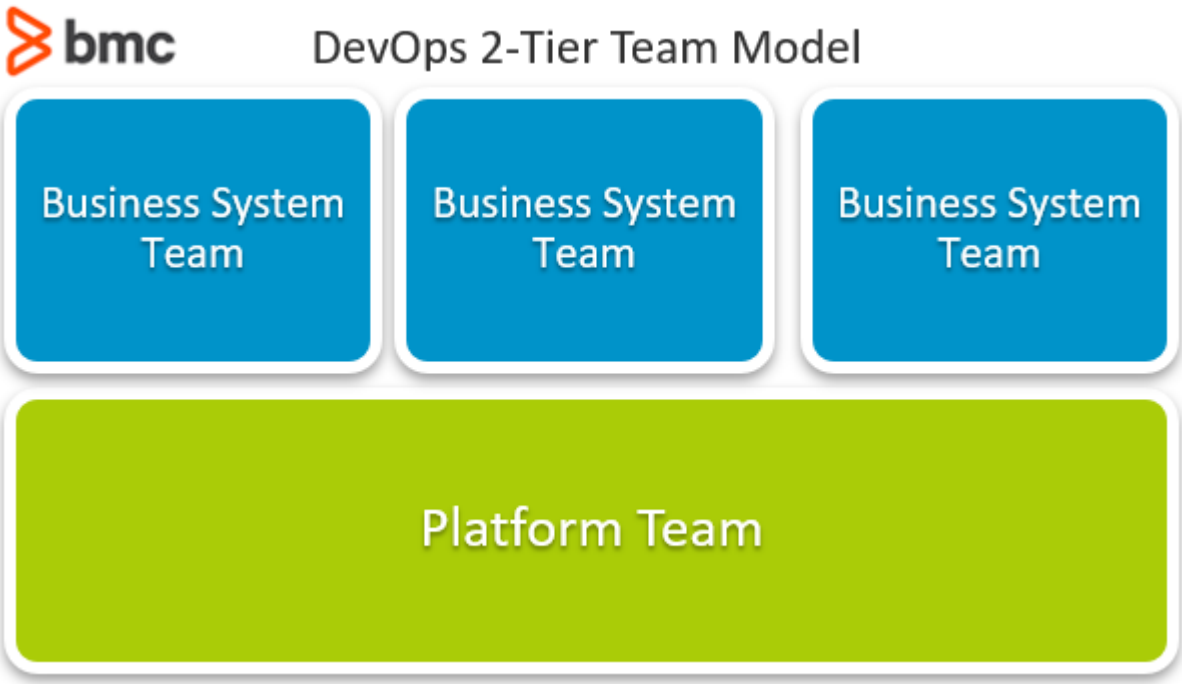
- **Agile.** The team is focused on creating customer value according to the committed time, quality, and value. They are transparent on performance, progress, and impediments, with a constant and relentless push towards improvement through feedback.
- **Self-organizing.** The team is autonomous within set boundaries and is aligned to other teams through a clear vision and goal definition therefore is interdependent on others.
- **Cross-functional.** All required competencies to develop and manage products should be within the team. Attainment of comb-shaped competencies is preferred for all team members, as well as continuous knowledge sharing and collaboration.
- **No sub teams.** The team works optimally as one unit and does not split into separate teams to address work concerns.

(Explore different types of IT teams.)

DevOps organizational model

The structure that is now most appreciated for DevOps organizations is a two-tier mode comprising:

- **Business System Teams** who take full responsibility of the product lifecycle end-to-end, as well as managing business and end users.
- **Platform Teams** who manage the underlying platforms and infrastructure and present these as a self-service to business system teams via APIs.



Product-focused DevOps teams are multidisciplinary, having every capability required for the entire product lifecycle. This reduces expensive handovers and delays, with the team focused on:

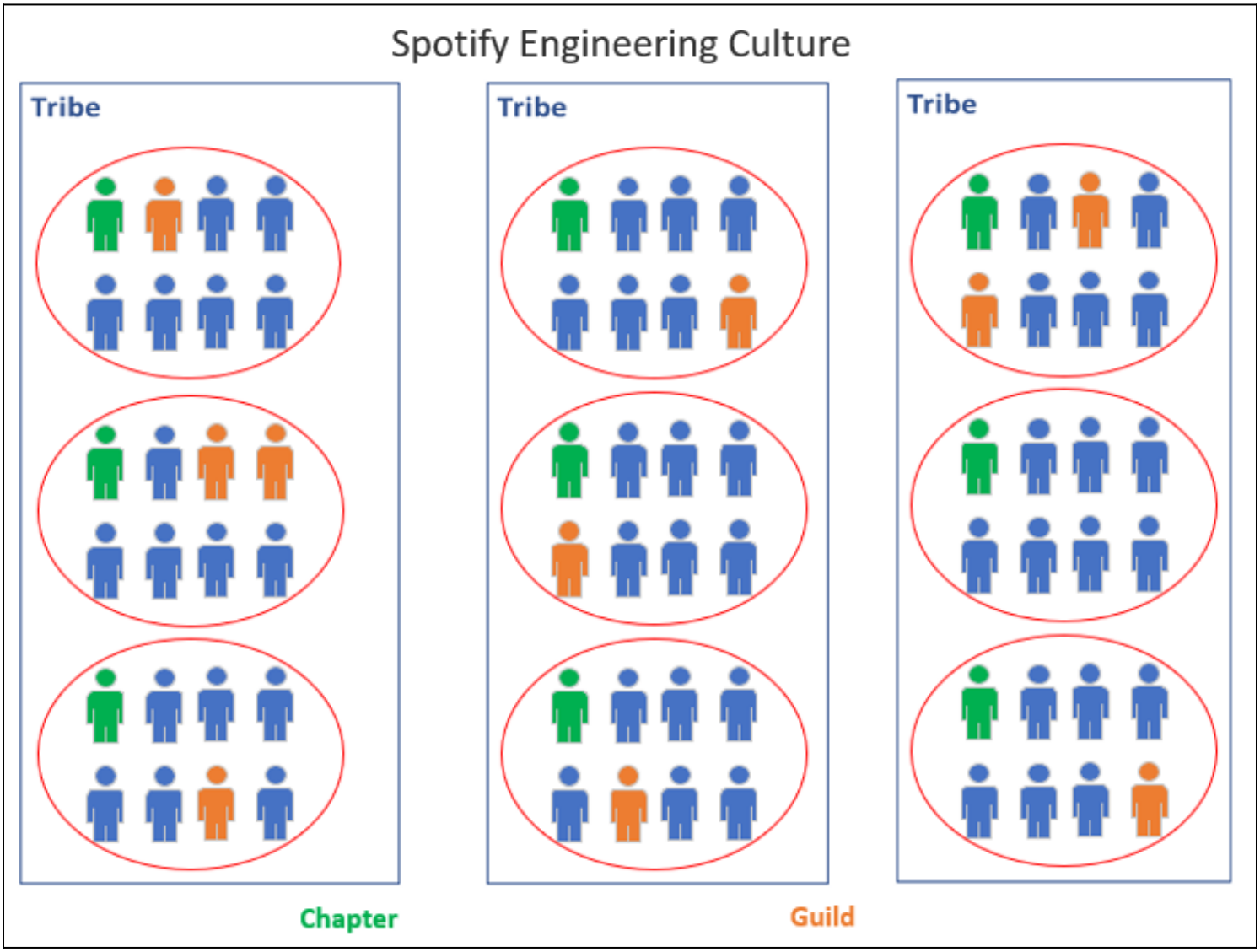
- Delivering product features to customers as quickly as possible
- Incorporating feedback autonomously without going through bureaucratic hurdles from other siloed entities

Modern DevOps teams employ value stream mapping to visualize their activities and gain necessary insights in order to optimize the flow of product increments and value creation.

Taking an example from Spotify, the business teams are called squads, who handle specific services (e.g., search, playlist, player etc.). These squads are small, multidisciplinary and autonomous. They sit together and act as a mini-startup, incorporating every component required to support a service throughout its lifecycle.

The squads are loosely coupled yet tightly aligned. Where multiple squads collaborate on a feature, they form tribes. And for specialists, such as DBAs or JavaScript developers, there are:

- Chapters for aligning standards
- Guilds for communities that want to share best practices cutting across the organization



Platform teams at Spotify are called Operations teams. But their role is dissimilar to the traditional IT operations team involved in deployment and support. Their core mandate is to:

1. Build the pipeline to production used by the business teams.
2. Provide the infrastructure and automation tools that the business developers require for releasing and supporting the code themselves.
3. Ensure the underlying infrastructure and platforms can effectively support the services through capacity and availability planning, monitoring, and optimization.

DevOps team number

The optimal size of a DevOps team is subject to debate. Of course, a small team is preferable for a few reasons:

- Coordination is easier
- Decision making is not protracted
- Gelling would not take long

However, the risk with small teams means that getting all the required expertise might be a challenge, and loss of a team member might significantly impair the team's throughput. A general agreement is that team sizes should range between 5 and 12.

DevOps team roles

As the DevOps team is expected to have end-to-end responsibility of the service, some of the expected roles within the team include:

1. The **Product Owner** manages the interaction with the customer to understand the requirements and work with the rest of the team to prioritize their delivery and incorporate feedback.
2. The **Team Lead** provides oversight and guides the team based on the chosen approach (e.g. scrum, Kanban, lean etc.). Also facilitates coordination with other squads.
3. The **Solution Architect** figures out how the requirements will be designed in line with the organization's environment and existing systems.
4. **Developers** translate the design into code, and are involved in supporting the code through testing and deployment.
5. **Quality Assurance (QA)** validates the product to ensure it meet both customer and organizational requirements throughout the development and deployment phases.
6. The **Platform Engineer** supports the platform teams to ensure that the environment supports the products effectively, and uses the tools provided to automate integration and deployment.

Since every organization is different, the roles might be handled by individuals or be mixed, depending on the team's capabilities. But as the team continues to work together, then cross-pollination will take place as knowledge and responsibilities are shared. The 2021 DevOps Institute Upskilling report ranked the most important skills that a DevOps team member requires include:

- Automation skills: Replacing manual activities, tasks, or events
- Human (soft) skills: Collaboration, communication, empathy
- Technical skills: Coding, cloud, microservices, APIs
- Specific automation tools skills: CI/CD, orchestration
- Functional Knowledge: development, operations, security
- Process skills and knowledge: SAFe, SRE, ITIL®
- Business skills: Planning, finance, sales

Dig deeper into DevOps job titles, roles, and responsibilities, the next article in our DevOps Guide.

Related reading

- BMC DevOps Blog
- Product Owner vs Product Manager: What's the Difference?
- Agile Roles & Responsibilities
- Common Roles in Cloud Computing
- Tiger Teams: How To Solve Critical Problems