LEADERS AND MANAGERS

Over the years, our research has investigated the effects of various technical and Lean management practices on software delivery performance as well as team culture. However, in the early years of the project, we hadn't directly studied the effects of leadership on DevOps practices.

This chapter will present our findings on the role of leaders and managers in technology transformations, as well as outline some steps that leaders can take to improve the culture in their own teams.

TRANSFORMATIONAL LEADERSHIP

Not sure of how important technology leadership is? Consider this: by 2020, half of the CIOs who have not transformed their teams' capabilities will be displaced from their organizations' digital leadership teams (Gartner).

That's because leadership really does have a powerful impact on results. Being a leader doesn't mean you have people reporting to you on an organizational chart—leadership is about inspiring

and motivating those around you. A good leader affects a team's ability to deliver code, architect good systems, and apply Lean principles to how the team manages its work and develops products. All of these have a measurable impact on an organization's profitability, productivity, and market share. These also have an impact on customer satisfaction, efficiency, and the ability to achieve organizational goals—noncommercial goals that are important for profit-seeking and not-for-profit organizations alike. However, these effects on organizational and noncommercial goals are all indirect, through the technical and Lean practices that leaders support in their teams.

In our opinion, the role of leadership on technology transformation has been one of the more overlooked topics in DevOps, despite the fact that transformational leadership is essential for:

- Establishing and supporting generative and high-trust cultural norms
- Creating technologies that enable developer productivity, reducing code deployment lead times and supporting more reliable infrastructures
- Supporting team experimentation and innovation, and creating and implementing better products faster
- Working across organizational silos to achieve strategic alignment

Unfortunately, within the DevOps community we have sometimes been guilty of maligning leadership—for example, when middle managers or conservative holdouts prevent teams

from making changes needed to improve software delivery and organizational performance.

And yet, one of the most common questions we hear is, "How do we get leaders on board, so we can make the necessary changes?" We all recognize that engaged leadership is essential for successful DevOps transformations. Leaders have the authority and budget to make the large-scale changes that are often needed, to provide air cover when a transformation is underway, and to change the incentives of entire groups of technical professionals—whether they are in development, QA, operations, or information security. Leaders are those who set the tone of the organization and reinforce the desired cultural norms.

To capture transformational leadership, we used a model that includes five dimensions (Rafferty and Griffin 2004). According to this model, the five characteristics of a transformational leader are:

- **Vision.** Has a clear understanding of where the organization is going and where it should be in five years.
- **Inspirational communication.** Communicates in a way that inspires and motivates, even in an uncertain or changing environment.
- **Intellectual stimulation.** Challenges followers to think about problems in new ways.
- **Supportive leadership.** Demonstrates care and consideration of followers' personal needs and feelings.
- **Personal recognition.** Praises and acknowledges achievement of goals and improvements in work quality;

personally compliments others when they do outstanding work.

What Is Transformational Leadership?

Transformational leadership means leaders inspiring and motivating followers to achieve higher performance by appealing to their values and sense of purpose, facilitating wide-scale organizational change. Such leaders encourage their teams to work toward a common goal through their vision, values, communication, example-setting, and their evident caring about their followers' personal needs.

It has been observed that there are similarities between servant leadership and transformational leadership, but they differ in the leader's focus. Servant leaders focus on their followers' development and performance, whereas transformational leaders focus on getting followers to identify with the organization and engage in support of organizational objectives.

We also selected transformational leadership as the model to use in our research because it is more predictive of performance outcomes in other contexts, and we were interested in understanding how to improve performance in technology.

We measured transformational leadership using survey questions adapted from Rafferty and Griffin (2004):[1]

My leader or manager:

- (Vision)
  - Has a clear understanding of where we are going.
  - Has a clear sense of where he/she wants our team to be in five years.
  - Has a clear idea of where the organization is going.
- (Inspirational communication)
  - Says things that make employees proud to be a part of this organization.
  - Says positive things about the work unit.
  - Encourages people to see changing environments as situations full of opportunities.
- (Intellectual stimulation)
  - Challenges me to think about old problems in new ways.
  - Has ideas that have forced me to rethink some things that I have never questioned before.
  - Has challenged me to rethink some of my basic assumptions about my work.
- (Supportive leadership)
  - Considers my personal feelings before acting.
  - Behaves in a manner which is thoughtful of my personal needs.
  - Sees that the interests of employees are given due consideration.
- (Personal recognition)
  - Commends me when I do a better than average job.
  - Acknowledges improvement in my quality of work.
  - Personally compliments me when I do outstanding work.

Our analysis found that these characteristics of transformational leadership are highly correlated with software delivery performance. In fact, we observed significant differences in leadership characteristics among high-, medium-, and low-performing teams. High-performing teams reported having leaders with the strongest behaviors across all dimensions: vision, inspirational communication, intellectual stimulation, supportive leadership, and personal recognition. In contrast, low-performing teams reported the lowest levels of these leadership characteristics. These differences were all at statistically significant levels. When we take our analysis one step further, we find that teams with the least transformative leaders are far less likely to be high performers. Specifically, teams that report leadership in the bottom one-third of leadership strength are only half as likely to be high performers. This validates our common experience: though we often hear stories of DevOps and technology transformation success coming from the grassroots, it is far easier to achieve success when you have leadership support.

We also found that transformational leadership is highly correlated with employee Net Promoter Score. We find transformational leaders in places where employees are happy, loyal, and engaged. Although our research didn't include measures of transformational leadership and organizational culture in the same year, other studies have found that strong transformational leaders build and support healthy team and organizational cultures (Rafferty and Griffin 2004).

A transformational leader's influence is seen through their support of their teams' work, be that in technical practices or

product management capabilities. The positive (or negative) influence of leadership flows all the way through to software delivery performance and organizational performance. We show this in Figure 11.1.

Said another way, we found evidence that leaders alone cannot achieve high DevOps outcomes. We looked at the performance of teams with the strongest transformational leaders—those with the top 10% of reported transformational leadership characteristics. One might think that these teams would have better than average performance. However, these teams were equally or even less likely to be high performers compared to the entire population of teams represented in survey results.

This makes sense, because *leaders cannot achieve goals on their own*. They need their teams executing the work on a suitable architecture, with good technical practices, use of Lean principles, and all the other factors that we've studied over the years.

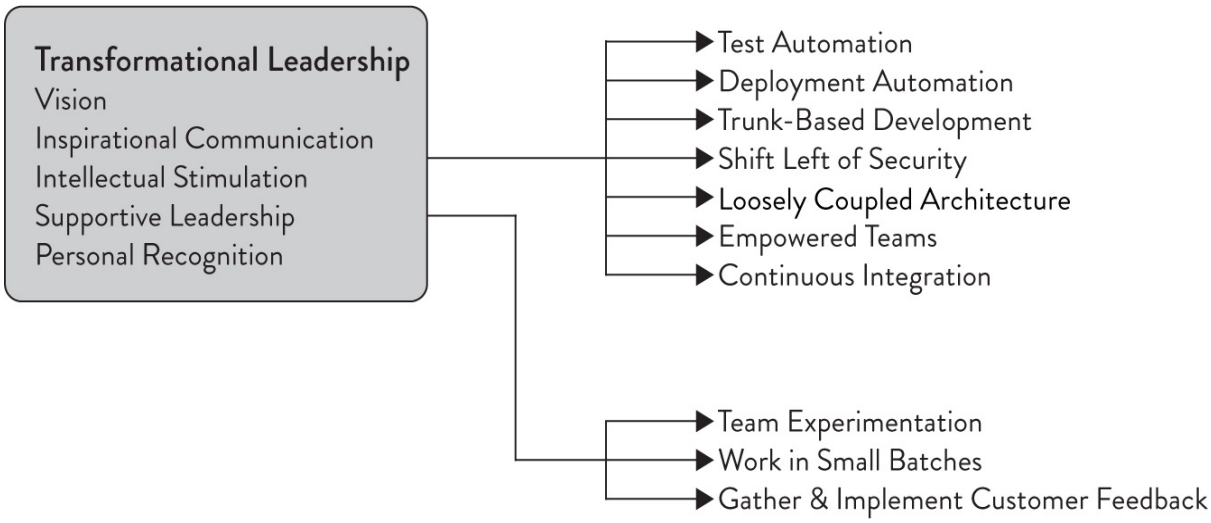

Figure 11.1: Impacts of Transformational Leadership on Technical and Lean Capabilities

In summary, we found that leadership helps build great teams,

great technology, and great organizations—but indirectly, leadership enables teams to rearchitect their systems and implement the necessary continuous delivery and Lean management practices.

Transformational leadership enables the practices that correlate with high performance, and it also supports effective communication and collaboration between team members in pursuit of organizational goals. Such leadership also provides the foundation for a culture in which continuous experimentation and learning is part of everybody's daily work.

The behavior of transformational leaders thus enhances and enables the values, processes, and practices that our research has identified. It is not a separate behavior or a new set of practices—it just amplifies the effectiveness of the technical and organizational practices we have been studying over several years.

THE ROLE OF MANAGERS

We see that leaders play a critical role in any technology transformation. When those leaders are managers, they may have an even bigger role in affecting change.

Managers are those who have responsibility for people, and often budgets and resources, in organizations. In the best case, managers are also leaders and take on the characteristics of transformational leadership outlined above.

Managers, in particular, play a critical role in connecting the strategic objectives of the business to the work their teams do.

Managers can do a lot to improve their team's performance by creating a work environment where employees feel safe, investing in developing the capabilities of their people, and removing obstacles to work.

We also found that investment in DevOps is highly correlated with software delivery performance. When it comes to culture, managers can improve matters by enabling specific DevOps practices in their teams and by visibly investing in DevOps and in their employees' professional development.

Managers can also facilitate big improvements in software delivery performance by taking measures to make deployments less painful. Last but not least, managers should make performance metrics visible and take pains to align these with organizational goals, and should delegate more authority to their employees. Knowledge is power, and you should give power to those who have the knowledge.

You may be asking yourself: What could investment in DevOps initiatives and my teams look like? There are a number of ways technology leaders can invest in their teams:

- Ensure that existing resources are made available and accessible to everyone in the organization. Create space and opportunities for learning and improving.
- Establish a dedicated training budget and make sure people know about it. Also, give your staff the latitude to choose training that interests them. This training budget may include dedicated time during the day to make use of resources that already exist in the organization.

- Encourage staff to attend technical conferences at least once a year and summarize what they learned for the entire team.
- Set up internal hack days, where cross-functional teams can get together to work on a project.
- Encourage teams to organize internal "yak days," where teams get together to work on technical debt. These are great events because technical debt is so rarely prioritized.
- Hold regular internal DevOps mini-conferences. We've seen organizations achieve success using the classic DevOpsDays format, which combines pre-prepared talks with "open spaces" where participants self-organize to propose and facilitate their own sessions.
- Give staff dedicated time, such as 20% time or several days after a release, to experiment with new tools and technologies. Allocate budget and infrastructure for special projects.

TIPS TO IMPROVE CULTURE AND SUPPORT YOUR TEAMS

As the real value of a leader or manager is manifest in how they amplify the work of their teams, perhaps the most valuable work they can do is growing and supporting a strong organizational culture among those they serve: their teams. This allows the experts that work with and for them to operate at maximum effectiveness, creating value for the organization.

In this section, we list some easy ways managers, team leads, and even engaged practitioners can support the culture in their teams. Our research shows that three things are highly correlated with software delivery performance and contribute to a strong team culture: cross-functional collaboration, a climate for learning, and tools.

Enable cross-functional collaboration by:

- **Building trust with your counterparts on other teams.** Building trust between teams is the most important thing you can do, and it must be built over time. Trust is built on kept promises, open communication, and behaving predictably even in stressful situations. Your teams will be able to work more effectively, and the relationship will signal to the organization that cross-functional collaboration is valued.
- **Encouraging practitioners to move between departments.** An admin or engineer may find as they build their skills that they're interested in a role in a different department. This sort of lateral move can be incredibly valuable to both teams. Practitioners bring valuable information about processes and challenges to their new team, and members of the previous team have a natural point person when reaching out to collaborate.
- **Actively seeking, encouraging, and rewarding work that facilitates collaboration.** Make sure success is reproducible and pay attention to latent factors that make collaboration easier.

Use Disaster Recovery Testing Exercises to Build Relationships

Many large technology companies run disaster recovery testing exercises, or "game days," in which outages are simulated or actually created according to a pre-prepared plan, and teams must work together to maintain or restore service levels.

Kripa Krishnan, Director of Cloud Operations at Google, runs a team that plans and executes these exercises. She reports, "For DiRT-style events to be successful, an organization first needs to accept system and process failures as a means of learning . . . We design tests that require engineers from several groups who might not normally work together to interact with each other. That way, should a real large-scale disaster ever strike, these people will already have strong working relationships" (ACMQueue 2012).

Help create a climate of learning by:

- **Creating a training budget and advocating for it internally.** Emphasize how much the organization values a climate of learning by putting resources behind formal education opportunities.
- **Ensuring that your team has the resources to engage in informal learning and the space to explore ideas.** Learning often happens outside of formal education. Some companies, like 3M and Google, have famously set aside a portion of time (15% and 20%, respectively) for focused

free-thinking and exploration of side projects.

- **Making it safe to fail.** If failure is punished, people won't try new things. Treating failures as opportunities to learn and holding blameless postmortems to work out how to improve processes and systems helps people feel comfortable taking (reasonable) risks, and helps create a culture of innovation.
- **Creating opportunities and spaces to share information.** Whether you create weekly lightning talks or offer resources for monthly lunch-and-learns, set up a regular cadence of opportunities for employees to share their knowledge.
- **Encourage sharing and innovation by having demo days and forums.** This allows teams to share what they have created with each other. This also lets the teams celebrate their work and learn from each other.

Make effective use of tools:

- **Make sure your team can choose their tools.** Unless there's a good reason not to, practitioners should choose their own tools. If they can build infrastructure and applications the way they want, they're much more likely to be invested in their work. This is backed up in the data: one of the major contributors to job satisfaction is whether employees feel they have the tools and resources to do their job (see Chapter 10). We also see this in our data as one of the predictors of continuous delivery: teams that are empowered to choose their own tools drive software

delivery performance (see Chapter 5). If your organization must standardize tools, ensure that procurement and finance are acting in the interests of teams, not the other way around.

- **Make monitoring a priority.** Refine your infrastructure and application monitoring system, and make sure you're collecting information on the right services and putting that information to good use. The visibility and transparency yielded by effective monitoring are invaluable. Proactive monitoring was strongly related to performance and job satisfaction in our survey, and it is a key part of a strong technical foundation (see Chapters 7 and 10).

While many DevOps success stories highlight the fantastic grassroots efforts of the technical teams involved, our experience and our research shows that technology transformations benefit from truly engaged and transformational leaders who can support and amplify the work of their teams. This support carries through to deliver value to the business, so organizations would be wise to see leadership development as an investment in their teams, their technology, and their products.

[1] Our analysis confirmed these questions were good measures of transformational leadership. See Chapter 13 for a discussion of latent constructs and Appendix C for the statistical methods used.