

WHITEPAPER

TRANSFORMING ORGANIZATIONAL CULTURE WITH PLATFORM-AS-A-SERVICE

Krain Arnold, Principal consultant, Red Hat

INTRODUCTION

The traditional technical and cultural challenges that large, hierarchical IT organizations face become more visible during times of IT transformation, such as the adoption of cloud technologies, DevOps practices, and agile methodologies.

Persistent organizational culture is organic, and it can be extremely difficult for IT leaders to dictate it effectively from the top down. IT leaders can, however, use technology to encourage proper incentive structures in all levels of the organization.

This paper explores the role of OpenShift by Red Hat in creating an atmosphere where creative and passionate developers can succeed frequently and quickly. Their success means decision-makers can impact organizational culture and more rapidly achieve their goals. In this way, IT leaders can harness the efficiency and persistence of spontaneous organization, despite the inherent limitations of hierarchical systems.

CHALLENGE

Technology changes exponentially. People, culture, and policies change incrementally. Larry Downes – an Internet industry analyst, business strategist, and author – defines this dynamic as the Law of Disruption.

“According to the Law of Disruption, technology changes exponentially, but social, economic, and legal systems change incrementally.”

IT leaders migrating from old to new practices will recognize this, whether they are moving from waterfall to agile or making a more complex move from traditional organizational silos to a DevOps-oriented culture.

The majority of complications associated with organizational evolution are rarely due to any particular product or the quality of code. More often, they are due to organizational culture, processes, and policies. Agile and DevOps methodologies are not something you can buy, they are something you must do, and you should be suspicious of anyone selling a quick fix.



facebook.com/redhatinc
[@redhatnews](https://twitter.com/redhatnews)
linkedin.com/company/red-hat

redhat.com

¹ Downes, L. (2009). *Digital Life. In The laws of disruption harnessing the new forces that govern life and business in the digital age* (p. 17). New York: Basic Books.

Agile and DevOps methodologies are not something you can buy, they are something you must do.

The impact of cloud technologies on policy and culture will be substantial. Benefits like on-demand self-service, rapid elasticity, and automated application scaling will necessarily change how things are done. For example, when application scaling is automated, an organization that requires multiple forms and change requests in order to provision a virtual machine will have to either change the policy or automate the existing request process. The latter is rarely successful.

In modern IT organizations, technology is less of a problem compared to the cultural issues that accompany its implementation. Nothing exposes poor internal communication more quickly and effectively than an attempt to automate cross-organizational practices.

When a process is fully automated and delivered in a fraction of the time it took manually, there is no room for error. Siloed systems and archaic processes (or procedural scar tissue) collected over years of manual execution rarely remain obscured when illuminated by an automation effort.

This becomes evident when a technology like OpenShift by Red Hat is introduced to a culture that is not ready for it. While the product itself may be operational, reducing the provisioning overhead from weeks to minutes can be negated by old policies requiring multiple approvals from various levels of management.

Culture problems can be difficult to diagnose, and even harder to resolve. When an IT leader implements a new way of thinking, the organizational structure can greatly impact the success of such a mandate. Divisions within the organization often prioritize reducing their own culpability should the implementation fail. This results in blame shifting, conflicting information, and confusion.

A common response to this is a blanket mandate to implement change by a given date. While this may seem effective, it can create a perverse incentive for sub-divisions to take any means necessary to comply. Teams may follow the letter of the mandate without truly adhering to its spirit. A common result is that the vocabulary of the methodologies is adopted, but underlying issues are not addressed.

Consider a large Federal institution given a mandate to transition to agile development practices. There are cultural elements necessary for such a transition, including:

- Changing management's attitude toward failure.
- Allowing for public discussion of current procedural limitations without fear of retribution or blame.
- Deeper trust between developers and management.

Complications should not be viewed merely as costs associated with progress. On the contrary, disruption, creative destruction, and change should be viewed as opportunity.

This particular organization had none of these elements. Nonetheless, developer teams and management attempted to comply with the mandate. However, the changes they implemented were entirely semantic. Existing 2-hour daily meetings became "standups," traditional developer cycles remained months long but were renamed "sprints," the project manager became the "scrum master," and a comprehensive description of the end product was expected during the first "sprint planning."

Most importantly, the developer teams were not given the opportunity to react to lessons learned as the project progressed, arguably the most valuable tenant of agile methodology. Any changes to the initial design of the project had to be approved through the same inefficient and tedious bureaucratic process as before. Six months and several million dollars later, they had successfully mapped agile terms to traditional waterfall practices and achieved nothing but an extremely expensive checked box.

This example highlights some of the complications inherent in attempting to drive cultural change from above. Much of this is due to information costs. Some of these costs are merely consequences of being a large institution, but some result from intentionally spreading misinformation or concealing truth because of a benefit to the sub-organization in question. Understanding and navigating the complexities of shaping organizational culture requires a proper comprehension of the incentive structures that govern your organization's decision-making processes and impacting those incentive structures positively.

OPPORTUNITY

Disruptive technology is not a misnomer. New technology and the practices associated with it are creative destruction at its finest. Inefficiencies and irrelevant, legacy practices will be ripped away, replaced by streamlined, productive practices better suited for the modern datacenter.

This change is absolutely necessary for any IT organization that wishes to be relevant in five years, but it would not be realistic to expect that it will happen without complication. However, these complications should not be viewed merely as costs associated with progress. On the contrary: disruption, creative destruction, and change should be viewed as opportunity.

An automation effort is an example of creative destruction as an opportunity. In the initial stages, when procedural inefficiencies are encountered, it is common to blame the automation, but that is short-sighted. It is more likely that the issues are being exposed by the automation effort, not caused by it. Inefficient practices are easily obscured by emails, meetings, verbose forms, and tribal knowledge. Streamlining and automating something as standard as building a server subjects the process to a higher level of scrutiny. It illuminates areas ripe for improvement. This is an opportunity for any decision-maker to shake things up, eliminate comfort zones, and expose organizational weaknesses.

APPROACH

IT is changing fast, and learning to use that volatility to improve your organization is an art. Through all of this, one of the most valuable traits an IT leader can exhibit is patience. Change is more likely to be achieved through evolution than revolution.

A patient leader is substantially less likely to issue a counterproductive mandate. There is an important difference between issuing a mandate to implement DevOps by the end of the month and one that creates an environment in which DevOps can become a reality.

A productive mandate, for example, would declare that all reviews of current policies and procedures must be done in an environment where open discussion of assumptions and opinions is encouraged, without fear of retribution.

There is some level of procedural inefficiency in your organization. You can either choose to conceal it and suffer or expose it, learn from it, and improve. If you make it clear that you prefer the latter approach, and reward those who follow suit, you will create an environment where team members can examine themselves and each other honestly and learn more quickly from mistakes. Failure is the quickest path to success when you learn from it effectively.

New technology and practices can be a catalyst for cultural complications, but technology used wisely can also be a powerful tool for mitigating these complications.

The basic premise of Platform-as-a-Service (PaaS) assumes individual responsibility, and the flexibility of OpenShift in particular allows your team to explore that responsibility.

The values that shape organizational culture are all about people. A key to changing or maintaining a particular culture is to identify the type of individuals that you want to shape your organization and empower them as necessary.

OpenShift by Red Hat is one example of a technology that can help creative and passionate members of your organization make the principles discussed in this paper a reality. The basic premise of Platform-as-a-Service (PaaS) assumes individual responsibility. The flexibility of OpenShift in particular ensures that your team will be given the freedom to explore that new responsibility. The following examples demonstrate ways that OpenShift can impact organizational culture incrementally, in addition to its more obvious benefits.

PRODUCTIVITY TOOLS

A common challenge when implementing new technologies is finding ways they can add value immediately, in addition to viewing them as a long-term investment.

For example, the suite of community-developed productivity tools available to OpenShift users is an excellent way to gradually socialize new concepts and technologies among your developer teams. These tools, available as quickstarts and cartridges, can increase developer productivity without requiring a redesign in the software development life cycle. Many tools fall into this category, but the following list highlights some of the available options:

- **Etherpad:** A hosted web service that allows real-time document collaboration for groups of users.
- **OwnCloud:** An enterprise file sharing solution for online collaboration.
- **Redmine:** A web-based project management and bug-tracking tool.
- **Nexus:** A flexible repository management tool.
- **Kanbanik:** A free and open source kanban board used for organization and task management.
- **Ghost:** A simple, powerful publishing platform.
- **Jenkins:** The leading open source continuous integration server.

Making OpenShift available to your organization lets developers experiment with these productivity tools, encouraging familiarization with the product with minimal initial disruption. This increases the likelihood that your developers will be receptive.

Using OpenShift in this way may provide further benefits for organizations dealing with rogue or shadow IT. The following is not a defense of the use of these risky outside resources, as the danger and downsides are well understood. It may, however, be helpful to try and understand rogue IT as a symptom of a greater problem.

The lack of flexibility in a fixed technology portfolio can be frustrating to an engineer who simply wants to build something great. In some cases, unsanctioned products are used to simplify workflow and develop faster. We can learn from a developer who chooses unsanctioned technology, though that learning does not justify the risks. Introducing the flexible and secure portfolio of productivity tools available with OpenShift can assist you in stamping out the use of rogue IT by eliminating the need.

Making OpenShift available to your organization will allow developers to experiment with these productivity tools in a way that encourages familiarization with the product with minimal initial disruption. This increases the likelihood that your developers will be receptive.

Workloads will thrive as PaaS natives, leveraging the true power of the modern data center while strategically avoiding the complications of forced adaptation

INCREMENTAL FAMILIARIZATION

It takes time to adjust to the concepts of cloud technologies, especially when dealing with ideas like rapid elasticity and on-demand self-service. Your organization will not embrace the principals of ephemeral workloads overnight.

Organizations often protect their virtual machines because their experience replacing them has been either monetarily or bureaucratically painful. It is easier to encourage members of an organization to destroy an old server after they have seen how easy it is to get a new one.

Providing OpenShift as an open sandbox allows your developers and administrators to spin up test servers for infrastructure or code in minutes. This familiarizes your organization with the idea of treating servers as ephemeral objects. This cultural benefit is in addition to the more obvious value: empowering developers to clone servers to isolate and test variables or create infrastructure tools for preliminary testing (like Jenkins or Nexus), without sending a single email.

Furthermore, because a new server is available at the press of a button, the incentive to hoard old servers is eliminated. This not only frees up organizational resources, but also reduces the number of poorly updated servers with security vulnerabilities.

NATURAL ADAPTATION

As developers benefit from these new practices, more uses will manifest naturally. This will vary from organization to organization, but one common use case is to replace the first environment of an organization's software development life cycle with OpenShift. For example, you could replace the DEV environment in a system that uses DEV, TEST, and QA. In many situations, this means replacing the developer's own sandbox.

By embracing convention over configuration, a developer sandbox becomes standardized. In addition, because it is being provided by OpenShift, it will be available immediately. This streamlines the on-boarding process for new hires.

A common pitfall during the transition to cloud technologies is forcing traditional workloads into Infrastructure-as-a-Service (IaaS) or PaaS environments. By giving developers an opportunity to develop on OpenShift, you smooth the path from development to production and allow the process to evolve naturally. Workloads will thrive as true natives of a PaaS environment can take advantage of the true power of the modern datacenter, while strategically avoiding the complications of forced adaptation. Throughout this process, you will also continue to reap the benefits of consistently exposing your engineering team to cloud concepts and practices.

SUCCESS

As you move through this process, the next steps will start to become more clear. At some point, you will have a collection of stateless, horizontally scalable applications that can function efficiently in a cloud environment. Your organization is developing cloud-aware workloads. This is a pivotal decision point for the IT decision-maker.

One may be tempted at this point to force a full transition to using OpenShift in production, but a more patient strategy may not only satisfy organizational goals, but also genuinely shape organizational culture from the bottom up. Instead of mandating the use of OpenShift in production, allow your developer teams to opt in.

Mastering the art of shaping organizational culture is contingent upon allowing that culture to be defined from the bottom up while strategically guiding it from the top.

We briefly discussed the concept of choosing convention over configuration, but the principle is quite simple: The smaller the menu, the faster the progress. But forcing a limited menu of server configuration options on a culture that is accustomed to autonomy can be exhausting. Instead, provide the following options:

1. Developers who intend to go to production with pre-approved server configurations can use OpenShift and have their servers immediately. They no longer need to concern themselves with approval processes or delays from manual configuration.
2. Those who feel they are not ready or require a configuration that is not yet standardized can still use the traditional process.

In this way, you create a clear path to success for the type of development team that you want to define your organizational culture. As they succeed, and consistently deliver higher quality code in less and less time, they will naturally become the standard by which new projects are judged.

CONCLUSION

Organizational and cultural change takes time, strategy, and patience. By critically examining your needs, you can find ways to introduce new ideas that add value immediately and pave the way for future success.

Being an effective leader in any capacity is always a challenge. Being an effective leader in IT carries the additional complication of constantly being on the forefront of innovation. Mastering the art of shaping organizational culture is contingent upon allowing that culture to be defined from the bottom up while strategically guiding it from the top.

By introducing OpenShift incrementally, IT leaders can create incentive structures that foster an environment that creates a path for success for the individuals with the character traits that you want to define your culture. At the same time, it avoids abrasive and potentially counter-productive mandates.

OpenShift will continue to change, especially as we move into exciting areas like containerization and the increasing integration between OpenShift and technologies like Red Hat Enterprise Linux OpenStack Platform and Red Hat CloudForms. However, the general principles discussed here will continue to be applicable as OpenShift evolves, and are equally applicable across all new technologies.

Change is inevitable, but there are important choices to be made about how you guide your organization to respond to that change. You can't opt out of the future, but you can decide if it's something that happens to you or if it's something that you and your organization create.

ABOUT THE AUTHOR

Krain Arnold is a Red Hat consultant specializing in IaaS and PaaS implementation, automation, and business process analysis and architecture. At Red Hat, he leads and participates in engagements with a wide variety of clients, focusing on large organizations and government. He is a U.S. Army veteran and holds a degree in Economics from George Mason University. He currently resides in Sweden with his wife and 2 children.

ABOUT RED HAT

Red Hat is the world's leading provider of open source solutions, using a community-powered approach to provide reliable and high-performing cloud, virtualization, storage, Linux, and middleware technologies. Red Hat also offers award-winning support, training, and consulting services. Red Hat is an S&P company with more than 80 offices spanning the globe, empowering its customers' businesses.



facebook.com/redhatinc
@redhatnews
linkedin.com/company/red-hat

redhat.com
#12666597_1214

NORTH AMERICA
1 888 REDHAT1

**EUROPE, MIDDLE EAST,
AND AFRICA**
00800 7334 2835
europe@redhat.com

ASIA PACIFIC
+65 6490 4200
apac@redhat.com

LATIN AMERICA
+54 11 4329 7300
info-latam@redhat.com