2022 Planning Guide for IT Operations and Cloud Management

By Gregory Murray, Lydia Leong, Simon Richard, Tony Iams, Paul Delory, Marco Meinardi, Douglas Toombs, Venkat Rayapudi, Dustin Hassemer, Steve White
2022 Planning Guide for IT Operations and Cloud Management

Overview

Key Findings

- As business functions are increasingly adopting digital technologies, effective I&O teams adapt their operating models to balance between enabling business autonomy and ensuring production readiness.

- The business-led technology adoption and the continued growth of agile development is forcing I&O to deliver IT services faster, while championing principles and practices of reliability.

- Innovative I&O teams are expanding their cloud operations and ITSM governance beyond just infrastructure and are enabling developers to leverage advanced cloud services to deliver more value with less time and effort.

- Automation and analytics can help but are most effective with a commitment to their full life cycle and maintenance.

Recommendations

To address the key technical trends of 2022, technical professionals responsible for I&O strategies within their organizations should:
IT Operations and Cloud Management Trends

Through 2022, technical professionals will need to deliver processes and platforms that support the shift of IT technologies to a more business-driven and application-focused model. Building these new processes and skill sets is hard enough, but the challenge is amplified by a need to do this faster than ever and without impacting compliance, availability and quality.

While most organizations were already shifting business models and processes to online services, COVID-19 suddenly made these efforts critical and overdue. Even the most traditional organizations were forced to immediately scale to meet the needs of a large remote workforce and rely, in some cases exclusively, on digital platforms to reach customers. Improving the customer experience provided by applications and services suddenly became everyone's priority — not just the application team. The future is clear: In 2022 and beyond, traditional business roles will be critical stakeholders in both the strategy and execution of IT technologies and processes.

The good news is that the challenges ahead are fairly universal, which has fostered lots of commercial solutions and an I&O community succeeding with innovative practices to adapt to current business expectations. This Planning Guide describes the trends resulting from the ongoing innovation, acceleration and transformation. For each trend, this guide highlights the strategies and techniques that high-performing I&O teams are using to tackle these challenges.

- Adopt a product-centric view of I&O, where application teams and business users are treated as customers, and use agile techniques to develop governance practices that meet the pace of the customer demands.
- Embrace hybrid as the default footprint and take advantage of tools, technologies and processes that make hybrid operations easier.
- Treat their valuable automation assets as software projects, rather than ad hoc efforts, to elevate automation from reactive responses to foundational capabilities.
- Shift their focus to applications by combining monitoring tools and AIOps solutions that provide actionable insights, enable observability and focus on the customer experience. This shift helps improve response efficiency and alert relevance.
- Tackle increasing demand for self-service and decentralization by augmenting governance and ITSM practices with automated incident management, effective problem management, modern CMDB practices and reliable change management.
I&O leaders should use this Planning Guide to identify likely and upcoming challenges, get insight into how Gartner clients are overcoming them and to identify additional Gartner resources to help address these challenges. The most visible trends for 2022 are:

- Accelerating application release velocity will demand new operational and tooling strategies.
- Increasing complexity of hybrid environments will require new architectural primitives and a mature cloud operating model.
- A shift from infrastructure to platforms and services will require disciplined automation as the foundation of I&O.
- The growing need for both increased observability and operational efficiency will accelerate APM, DEM and AIOps adoption.
- Increasing self-service, autonomy and decentralization will require mature and multidisciplined governance and ITSM practices.

Figure 1 shows the trends and associated planning considerations for IT operations technical professionals in 2022.
Figure 1: 2022 Key Trends in IT Operations and Cloud Management

For cloud-specific planning trends, please see Gartner’s 2022 Planning Guide for Cloud and Edge Computing.

**Accelerating Application Release Velocity Will Demand New Operational and Tooling Strategies**

In 2022, the digital engagement trends that originated in the pandemic — with customers increasingly interacting with businesses online through digital products and services — will not only continue but accelerate (see The 2021 CIO Agenda: Seize This Opportunity for Digital Business Acceleration). When business leaders refer to the desire for “IT agility,” what they often mean is “faster delivery of new applications and new application features,” accelerating application release velocity, especially in the context of delivering digital business initiatives.
Rapid release velocity not only improves time to market, but it also enables lower-cost experimentation. The business fundamentally gains flexibility if an application team can rapidly deploy a change, measure and analyze its business impact, and then either quickly roll it back or build upon it. Continuous integration and continuous delivery (CI/CD) — both in the cloud and on-premises — has become especially important for many organizations. Digital business requires early and frequent product releases that can only be achieved by DevOps teams following agile practices.

To support application teams in these efforts, I&O technical professionals must:

- Improve the alignment with applications owners by managing IT assets and services as products and application teams as customers.
- Enhance platform ops by balancing container management between I&O and DevOps teams.
- Apply immutability and infrastructure as code as core tenets of cloud-native application operations and tooling.
- Evaluate container management tools with a view toward hybrid and multicloud capabilities.

Planning Considerations

Improve the Alignment With Applications Owners by Managing IT Assets and Services as Products and Application Teams as Customers

Adopt a culture that defines business consumers as customers and strive to delight them by understanding and exceeding their expectations. There are two key changes that you’ll need to adopt to promote this cultural and organizational mindset:

- A shift toward a product-centric operating model. The mantra “products over projects” emphasizes a continuous approach to strategy execution and an incremental delivery of value. Product-centric delivery empowers standing business-IT teams (aka product teams) to explore and deliver incrementally to desired outcomes. Empowered teams can pivot more quickly based on new conditions in the market, customer preferences or priorities.
There are lots of ways to build and run a platform ops team. Prepare for this product-centric execution by understanding the common approaches and be ready for the new roles. The most common approaches to platform operations teams are as follows:

- **A shift toward delivering IT as a set of productized services.** When IT organizations build platforms for internal service delivery, these platforms are increasingly managed as if they were products. In I&O, this manifests itself organizationally in a platform operations (“platform ops”) approach, in which a team is responsible for agile delivery and operations of a platform that is defined and managed as a product. A product manager is responsible for that platform and strives to satisfy its customers. The customers are typically application teams or other technical end-user teams (such as teams of data scientists, researchers or engineers).

There are lots of ways to build and run a platform ops team. Prepare for this product-centric execution by understanding the common approaches and be ready for the new roles. The most common approaches to platform operations teams are as follows:

- **A developer platform ops team** delivers, maintains and improves a self-service application platform, including the CI/CD toolchain, for multiple agile application teams delivering custom-built software.

- **A cloud platform ops team** treats cloud management capabilities as a “product offering,” with a formal structure of products and services offered that imitates the capabilities of an external cloud managed service provider (MSP). The team delivers a software platform for cloud operations, along with cloud managed services and cloud consulting services, to the rest of the business.

- **Data, analytics and AI/ML operations teams** may also use a platform ops approach.

All platform operations approaches share a customer-centric mindset. Therefore, focus on the needs of the internal teams that the platform serves. Shift your thinking — from metrics to value stories, from widgets to productized services, and from “you get what we give you” to a focus on customer satisfaction. Finally, measure success through user feedback:

- **Customer satisfaction score**: How satisfied is the user with the platform and with their interaction with the personnel who support the platform?

- **Customer effort score**: How easy is it to interact with platform service and support?

- **Value enhancement score**: Does the platform deliver direct value to the user, and do interactions result in greater confidence in the platform?
Platforms drive standardization and efficiency, but the emphasis on the satisfaction of the internal teams that use those platforms shifts the definition of success from “we delivered reliable and efficient infrastructure” to “we delighted our application teams through the cloud capabilities we enabled.”

Related Research

- Quick Answer: When Should CIOs Use Projects Versus Products for I&T Delivery?
- Using Platform Ops to Scale and Accelerate DevOps Adoption
- Guidance Framework for Implementing Cloud Platform Operations
- Demystifying XOps: DataOps, MLOps, ModelOps, AIOps and Platform Ops for AI

Enhance Platform Ops by Balancing Container Management Between I&O and DevOps Teams

Delivering software fast enough to meet the demands of business requires solid teamwork between software teams and IT operations. DevOps has been a key practice for improving the interaction between developers and I&O. With DevOps, cross-functional product teams take end-to-end ownership of the applications they deliver — from design to deployment and operations and retirement. One of the big challenges in scaling DevOps is the intersection between I&O and development is neither purely IT operations nor purely development. Figure 2 shows how container orchestration with Kubernetes establishes a consistent unit of exchange between the platform ops team and the DevOps teams.
Containers and Kubernetes solves this challenge by providing a standard framework for I&O and application development to collaborate, deploy and operate applications at scale. The self-service capabilities in Kubernetes platforms also give product teams the ability to deploy services without involving I&O. Where you have multiple teams practicing DevOps, adopt platform ops to gain even greater efficiency by delivering a consistent, shared container management platform.

With the platform ops approach, the team responsible for the container management platform should focus on the following tasks:

- Provide application development teams with tools, processes and policies that they can use to deliver software products in a consistent, agile manner.
- Define any necessary abstractions for the application teams to practice DevOps on the platform on a self-service basis using CI/CD processes.
Invest in skills that make platform ops more efficient and effective. The platform ops team needs significant software engineering skills. Platform ops is easier to implement when using containers and Kubernetes, but you could include virtual infrastructure on-premises, cloud infrastructure as a service (IaaS), or serverless function platform as a service (fPaaS) in the cloud. Start a platform ops initiative with a targeted scope, like a Kubernetes environment or a CI/CD deployment, and focus on the minimally viable product that meets the needs of your customers.

The long-term value of adopting platform ops is the ability to smoothly scale the number of teams practicing DevOps. Measure the impact systematically, using the advice of Google's DevOps Research and Assessment (DORA) annual report, which advises to base software delivery performance on four metrics — two covering velocity and two for quality. Encourage your development teams to provide feedback through metrics that measure their ability to develop and deliver code into the deployment pipeline. Use metrics such as the overall stability of the platform and self-service productivity to gauge results and drive iterative improvements.

Related Research

- Solution Path for Continuous Delivery With DevOps
- New Roles and Skills for I&O Professionals in DevOps
- Using Platform Ops to Scale and Accelerate DevOps Adoption
Apply Infrastructure as Code and Immutability as Core Tenets of Cloud-Native Application Operations and Tooling

Infrastructure as code (IaC) brings principles of developer automation to I&O. Developers use continuous integration and continuous delivery to treat application release automation as a mechanized assembly line. Code artifacts are pulled in, assembled, packaged and validated automatically. Developers make changes to artifacts and, as shown in Figure 3, the automation incorporates these changes by reassembling the application with the updated artifacts. I&O can leverage this approach by using IaC for infrastructure automation.

IaC can be combined with immutability to address some common use cases that have been difficult to automate. Increasingly, environments that enable automated software-defined deployment support the immutable pattern. For example, immutable instances are no longer patched in production. When an application needs to be patched, the base image is patched and the artifact is picked up by a CI/CD pipeline. The unpatched instance is destroyed and replaced with the newly assembled instance, with the updated artifact. This pattern is common for container environments, where both the containers and the container configuration are treated as immutable artifacts.
This approach has given rise to GitOps, in which the assets and configuration of an environment, like a Kubernetes cluster, is kept in a Git repository. Container images are immutable and stored in the repository, along with the YAML manifests for container orchestration. All of the assets are maintained in this environment, bringing version control and proven automation release through CI/CD.

Declarative systems are particularly powerful in this environment. Declarative systems express the desired state, and then an active controller uses a closed-loop feedback system to automatically establish and maintain the desired state under varying operating conditions. With Kubernetes, this approach is usually used to orchestrate deployments of containers, but it can also be extended to other IT operations, such as deploying Kubernetes clusters themselves or maintaining virtual infrastructure hosting VMs.

As changes are committed to the repository, an orchestration component (such as Argo or Flux) responds to that event and applies the change to the cluster. The declarative representation of state and its continuous enforcement automated the change without operator intervention. Automatically triggering infrastructure reconfiguration based on version control events is not a new idea, but the deep configurability of declarative systems, as in Kubernetes with YAML files and its active control plane make the platform especially well-suited for the GitOps approach. While not all organizations are ready to adopt this degree of automation, a successful adoption of GitOps will simplify the operation of Kubernetes clusters.

The success of adopting IaC consistently with both applications and infrastructure can be measured in the reduction of time to deploy new releases and the time it takes to recover from unsuccessful releases of applications. The approach helps to avoid “configuration drift” from unrecorded system changes, which ultimately results in systems that are difficult or impossible to replicate if they have to be rebuilt. Use immutability, based on software artifacts, to enable faster and safer upgrades, simplify change management and troubleshooting, and reduce operational errors. When coupled with the appropriate CI processes, practicing immutability also helps to implement canary and blue/green releases, in which new releases of an application can be tested in production with a subset of users.

Related Research

- To Automate Your Automation, Apply Agile and DevOps Practices to Infrastructure and Operations
Evaluation Container Management Tools With a View Toward Hybrid and Multicloud Capabilities

When selecting container management tools, start by determining the role of cloud-based container services. Where compliance and functional requirements allow, deploy Kubernetes as a service managed by cloud service providers. This will both accelerate deployment and address any skill gap for production Kubernetes operations. Also, as shown in Figure 4, apply consistent operational processes to managing the state and life cycle of all Kubernetes clusters, whether they are on-premises, in the cloud or across cloud providers. One often-quoted reason for the adoption of Kubernetes is that it offers an abstraction that clients expect will solve their hybrid and multicloud consistency requirements. It does not.
In general, containerized applications managed by Kubernetes are relatively portable but, most often, lock-in comes from application integration with other provider services. All of the major cloud services offer managed versions of Kubernetes that are based on the same upstream open-source code, but each service has its own operational interfaces and differentiating capabilities. Although users generally prefer a consistent Kubernetes experience across locations, they may also want to define discrete policies and management constructs for cloud, on-premises and edge clusters separately in order to address the unique parameters of a particular location.

Depending on the use case for distributing Kubernetes, I&O teams responsible for deploying Kubernetes clusters on-premises and in one or more public cloud services must address two operational issues:
At present, the Kubernetes architecture does not define a complete standard approach for distributing its functionality across multiple clusters that is ready for production usage. Key open-source projects for enabling multicluster deployments have not yet reached version 1.0 releases. These include Kubernetes Cluster Federation (KubeFed for short) and Cluster API, which provides declarative APIs to simplify provisioning, upgrading and operating multiple Kubernetes clusters. Therefore, any solution for hybrid or multicloud Kubernetes will require either commercial products or custom solutions based on open-source software components.

Avoid focusing on “portability” as the primary driver of Kubernetes initiatives.

Instead, adopt Kubernetes for agility, standardization and consistent control of distributed systems. If portability requirements are unavoidable, collaborate with development teams to build a flexible CI/CD process that can deploy to different targets, rather than moving containers from one provider to another. The pipeline and configuration will need to abstract platform-specific parameters to allow deployments to different providers. Similarly, invest in tools or platforms to automate the life cycle management of heterogeneous Kubernetes products and services based on consistent processes, using immutable practices to simplify operations.

Successfully arriving at a solution for deploying containers with a hybrid and/or multicloud approach will provide an abstraction layer that enhances application portability by separating container orchestration mechanisms from the details of the underlying infrastructure. Even when the container environment is on-premises or in a single provider, these abstractions can help facilitate container DR across data centers or enable multitenant hosting environments by using separate clusters for each tenant.

Related Research
Increasing the Complexity of Hybrid Environments Will Require New Architectural Elements and a Mature Cloud Operating Model

The trend of adopting platforms and solutions across providers continues to accelerate, and managing complex, multicloud environments is becoming universal. If you haven't already, adapt to a state where hybrid management and operations is the primary use case. Organizations that have made this transition build effective multicloud management using a combination of technologies, practices and tools.

Landing zones have become a compelling architectural construct that delivers the rare combination of both agility and control. A “landing zone” is a fully equipped set of hierarchical constructs, policies, network, and identity configurations where infrastructure and platform resources can land safely. In other words, a landing zone is the baseline catalog of requirements for hosting workloads within a cloud provider. A well-architected “landing zone” in public cloud IaaS, along with the right cloud management practices and strategies, will lead to superior governance, lower costs and autonomy within the guardrails of the landing zone.

Maximizing the value of cloud will require changes in the way we manage and operate these distributed assets. Unfortunately, there is no single best approach to managing cloud environments. However, the cloud platform operations (CPO) pattern works well for a wide array of organizations, business needs, application portfolios and levels of cloud maturity. CPO blends characteristics of an external managed service provider (MSP) offering, with formally “productized” automation, engineering and operations capabilities. Adapting the cloud operating model is an important early step in facilitating cloud adoption — both within IT and within the developer community.
Over time, and as developers and other technical end users gain cloud-related skills, they will prefer the self-service experience of cloud and integrate that experience with toolchains for CI/CD. Mature organizations adopt policy-based governance across providers by separating policy definitions from enforcement style and provider-specific implementation. This approach is more flexible than per-provider rules and helps balance competing constraints like risk, agility, freedom of choice, complexity, cost, functionality and scalability. Cost, in particular, continues to get lots of attention, fueled by the financial disruption of the pandemic response. The right policies and the right kind of accountability will help cloud consumers control their cloud spending, especially in a self-service environment. Use both native and third-party tools to give your consumers the resources to create forecasts, monitor costs and pursue their own optimization opportunities.

I&O technical professionals can embrace the change and thrive by:

- Using cloud landing zones to enhance management and governance practices.
- Choosing a pragmatic approach to cloud operations.
- Assessing cloud provider management tools based on extensibility to on-premises, edge or other cloud providers.
- Combining native tools, third-party tools and tagging to control cloud costs and drive accountability.

Planning Considerations

Use Cloud Landing Zones to Enhance Management and Governance Practices

Cloud landing zones are becoming a core component of enabling autonomy and agility in cloud environments. A landing zone defines an environment — ideally, provisioned through code — that is ready to host a deployment. A landing zone has many hierarchical elements, including policies and policy assignments, templates, identity and network configurations. Use landing zones to ensure that cloud deployments are bound within established guardrails to govern usage and mitigate risks. Given their role in policy enforcement, network routing and segmentation strategies, it is important to develop the right landing zone architecture before scaling out significantly. For 2022, prioritize increasing your architectural and automation skills for cloud landing zones.

Figure 5 shows how the setup of a landing zone must be the first step in a cloud workload deployment life cycle.
A landing zone is created by the team leading cloud brokerage and governance. Then, the landing zone is handed off to cloud consumers, who can land their resources in it autonomously and with confidence that mandatory policies are in force. In the past, the setup of a landing zone has been primarily a manual task. With the continuous increase in complexity and scale of cloud services and deployments, in 2022, organizations must focus on automation to make the rapid deployment of landing zones repeatable and reliable.

Although hierarchical placement and policy attachments are the most common elements of a landing zone, this can also be characterized by additional configurations. A landing zone may also contain:

- A list of predefined, curated orchestration templates that consumers can leverage to provision their resources.
- A set of roles with predefined permissions that users can assume to perform their tasks.
- Networking elements, such as a dedicated or shared virtual network (such as a VPC), configured per the organization’s network policy. Depending on the type of landing zone being provisioned, inbound or outbound public internet access may or may not be provisioned.
- The default enablement of key services, such as audit logging, or the creation of security resources, such as a key vault or recovery vault.
Choose a Pragmatic Approach to Cloud Operations

When organizations use cloud IaaS+PaaS, many aspects of infrastructure and platform delivery are automated and delivered as a service. However, this does not entirely — or even mostly — free the organization of operational concerns. The organization must still possess cloud architecture, cloud engineering, and cloud security skills in order to properly architect, build, run and secure cloud environments. At the very least, the appropriate cloud configuration options must be selected. The organization is not freed of all hands-on operational concerns — including but not limited to monitoring, incident management, problem management, systems administration and security operations.

Even though application teams frequently want to embrace cloud self-service, very few application teams have the cloud skills and operational capabilities to be fully self-sufficient in a “You build it, you run it” approach. Consequently, almost all I&O organizations need to include a cloud operations function.

Many organizations begin with ad hoc cloud operations. Most organizations drift into a formal structure for event, incident and problem management for SaaS, but this is relatively straightforward because the operational burdens of SaaS are usually minimal. The troubles typically begin when the organization begins to use IaaS or PaaS. When IaaS or PaaS is adopted by individual application development teams or digital product teams, it may be sufficient to embed IaaS and PaaS operations within those teams or to distribute related responsibilities into the general I&O functional technology silos. Some organizations instead create a cloud center of excellence (CCOE) and task the personnel within that organization with taking on operations, architecture and governance responsibilities. Although these approaches can work when the organization is at the initial stages of cloud adoption, they often do not scale well as the organization moves further into its cloud journey.
CIOs often direct cloud architects to formalize a strategy for building and scaling a cloud operations function. This can, however, result in an idealized fantasy of perfection that will take years and dozens (if not hundreds) of people to accomplish. Gartner recommends taking a pragmatic approach to delivering what your organization needs now, while laying the foundations for maturing cloud operations as your cloud adoption increases and evolves.

Although Gartner has identified 11 common patterns for cloud operations, not all of these patterns are equally successful. The majority will outsource IaaS+PaaS provider management to an MSP, but for those who do not want to do so, the most successful pattern — the “cloud platform operations” approach — uses an internal I&O team that mimics the approach of the leading cloud MSPs. The team is divided into three aspects:

- **Software platform**: An agile development squad develops and maintains software tools and scripts — or extends open-source and commercial cloud management software — and integrates them together into a platform for cloud management.

- **Managed services**: Cloud operators perform operational tasks that cannot be (or haven’t yet been) automated. This may include operational troubleshooting, systems administration, configuring automation, enforcing policies that aren’t automatically enforced (such as addressing complex cost-management violations and security violations), and addressing other “toil.”

- **Consulting services**: DevOps engineers with cloud expertise perform cloud engineering projects. This may include assisting application teams with technical architecture, or building cloud IaC and orchestration templates — preferably side-by-side with developers. They may also be responsible for reviewing cloud configurations, IaC and templates produced by application teams for conformance with operations and security best practices.

This structure is illustrated in Figure 6, which contains an example of one possible way of assigning responsibilities to a cloud platform operations team.
Choosing an approach to cloud operations is not merely a question of organizational structure. It has significant technical implications. The cloud operations approach will impact not only how cloud engineers, operators and administrators do their work, but also how application developers (and other technical end users of cloud services, such as data scientists and engineers) do their work. It impacts the approach to security and regulatory compliance. It affects the entire application life cycle's processes and tools. It has implications for the implementation of DevOps within the organization. It impacts most agile initiatives, especially CI/CD, which is an increasingly common CIO-level priority. One of the key advantages of the cloud platform operations model is its adaptability to different operational needs, different levels of cloud maturity in different parts of the organization, and differing levels of maturity in agile development and DevOps adoption.

Related Research

- Comparing Cloud Operations Approaches
Assess Cloud Provider Management Tools Based on Extensibility to On-Premises, Edge or Other Cloud Providers

As hybrid environments proliferate, cloud providers have started to decouple their management tools’ value proposition from their platforms. Their management toolsets are now delivering value into locations such as private, on-premises data centers, edge and even other cloud providers. This provides significant benefits to customers that rely heavily on one cloud provider’s management toolset. Consistent interfaces enable reusability and extensibility of skills across providers. Customers that want to take advantage of such consistency must assess cloud provider management tools also based on their extensibility to external hosting platforms. Examples of the extensibility of cloud providers’ management tools include:

- **AWS CloudFormation** registry is a repository of plug-ins open to external contributions. Third-party vendors and client organizations can add providers to describe new resources and their management workflows. The CloudFormation registry is currently supported only by CloudFormation and AWS Config. I&O teams can use CloudFormation to orchestrate operations and AWS Config to manage and remediate policies on resources that are outside of the Amazon Web Services (AWS) platform.

- **Azure Arc** is an extension of the Azure Resource Manager (ARM) and uses the concept of resource providers to extend beyond the Microsoft Azure environment. Azure Arc requires an installed agent and a controller to create a subset of Azure services on a local Kubernetes deployment. With Azure Arc, clients can manage their legacy SQL Server (by installing the agent) or spin up Azure services such as Azure Kubernetes Service (AKS) and Azure Data Studio. Once deployed, Azure Arc services are unified within the Azure portal and can be managed with the same tools and workflows as the other services in the provider’s public cloud platform.

- **Google Cloud Anthos** extends the Google Kubernetes Engine (GKE) control plane to other deployments of the Kubernetes-based service that may run in other locations. Google Anthos provides the management layer that can tie together Kubernetes applications under a common management framework.
Although these three initiatives have significant differences, they all focus on delivering a single control plane. These initiatives are complementary to cloud providers’ platforms to edge offerings, like AWS Outposts, Microsoft Azure Stack and Google GKE on-prem. In 2022, organizations developing multicloud or hybrid management strategies should evaluate the extensibility of providers’ control planes and possibly rethink their tooling approaches accordingly. Next year, keep your tooling strategy flexible to accommodate for these new capabilities as they mature to support more use cases.

Related Research

- Differences Between AWS Outposts, Google Anthos, Microsoft Azure Stack and Azure Arc for Hybrid Cloud

**Combine Native Tools, Third-Party Tools and Tagging to Control Cloud Costs and Drive Accountability**

For most organizations, managing cloud costs has always been a challenge. Now, with the pandemic accelerating cloud adoption initiatives and elevating the appeal of technology autonomy, the challenge is greater than ever. Many organizations simply overlooked the challenges of managing cloud spending to meet time constraints and, over time, this will affect TCO and could jeopardize their cloud adoption success.

In 2022, prioritize the development of cloud cost management capabilities. Maturing cost management will allow better cost control while expediting cloud adoption strategies. With cloud cost management, organizations can create estimates, track spending on a daily basis, implement chargeback, identify overspending and quickly address cost optimization opportunities. Furthermore, they will be able to influence the behavior of their cloud consumers and, ultimately, make them accountable for their own cloud spend.

Avoid simply buying a tool and relying on its features to implement cloud cost management. Such an approach can work in the short term at cutting down costs, but it fails in building long-term strategic capabilities. Adopt the Gartner Public Cloud Cost Management Framework to manage spending without slowing down growth and innovation. Figure 7 depicts the framework.
Successful cost management relies on a solid resource classification strategy. Define and implement a classification strategy through the native provider's tagging. Then, use this classification to break down cloud charges and attribute costs to the right department or application.

Start with a tagging dictionary and the mechanisms, which might be different across environments, for enforcing the regular and correct use of the defined tags. Where possible, incorporate automation to facilitate adoption. Then, audit the implementation early in the process, promote the benefits of using tagging and recognize individual tagging efforts.

Even with a robust tagging strategy in place, comprehensive cost management will require multiple tools. No single tool can implement the full framework in Figure 7. Build the complete cost management solution from the following types of tools:

---

**Gartner Public Cloud Cost Management Framework**

<table>
<thead>
<tr>
<th>Plan</th>
<th>Define Requirements</th>
<th>Architect With Cost in Mind</th>
<th>Choose Pricing Models</th>
<th>Forecast Consumption</th>
<th>Deploy Pilot Application</th>
<th>Establish Budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>Track</td>
<td>Design Native Hierarchy</td>
<td>Implement Tagging Strategy</td>
<td>Allocate Costs of Shared Resources</td>
<td>Define Metrics to Track</td>
<td>Alert on Anomalies</td>
<td>Implement Chargeback and Showback</td>
</tr>
<tr>
<td>Reduce</td>
<td>Dispose Unused Resources</td>
<td>Schedule Services</td>
<td>Rightsize Allocation-Based Services</td>
<td>Leverage Discount Models</td>
<td>Upgrade Instance Generation</td>
<td>Establish a DevOps Feedback Loop</td>
</tr>
<tr>
<td>Optimize</td>
<td>Use Preemptible Instances</td>
<td>Set Up Data Storage Life Cycle Policies</td>
<td>Implement Horizontal Autoscaling</td>
<td>Balance Usage of Consumption-Based Services</td>
<td>Use Serverless Technologies</td>
<td>Modernize Your Application for PaaS</td>
</tr>
<tr>
<td>Evolve</td>
<td>Adopt Tooling</td>
<td>Onboard New Providers</td>
<td>Broker Cloud Services</td>
<td>Shift Budget Accountability</td>
<td>Incentivize Financial Responsibility</td>
<td>Correlate Costs to Business Value</td>
</tr>
</tbody>
</table>

Source: Gartner
72001_C

---

**Figure 7: Gartner Public Cloud Cost Management Framework**
Gartner recommends adopting cloud providers’ native tools to get started with cloud cost management and finding the “low-hanging fruit” of cost savings. The providers’ native cost optimization tools will continue to mature in 2022 and beyond, but they will continue lagging behind the capabilities of purpose-built third-party tools. Many organizations find that they exhaust the savings opportunities found by the provider’s native tools as their cloud estate (and their experience) increases. This becomes motivation to begin evaluating other tools that can assist with further cost optimization and management. At this point, evaluate whether to augment these native tools with third-party or DIY solutions.

**Related Research**

- How to Manage and Optimize Costs of Public Cloud IaaS and PaaS
- Implementing a Tagging Strategy for Cloud IaaS and PaaS
- Key Services Differences Between AWS, Azure and GCP: Governance and Policy Management
A Shift From Infrastructure to Platforms and Services Will Require Disciplined Automation as the Foundation of I&O

Delivering digital platforms and services at the speed of business, and on budget, requires extensive automation. For decades, operators and administrators have created their own libraries of custom-made run book automation and scripts to accelerate common tasks but haven't had much in the way of formal life cycle management. This ad hoc approach doesn't scale and doesn't reliably deliver the level of quality required of foundational automation that makes platform and service delivery effortless.

This trend is a continuation and expansion of the ongoing IT automation adoption. For 2022, things are shifting toward the principles and practices that elevate and mature automation development, seeing automation as a fundamental requirement rather than a way to improve efficiency of an existing process.

This shift is inspired by the successes that application teams in accelerating their release cycles by adopting formal software development life cycle (SDLC) practices. I&O technical professionals are recognizing that the value and impact of these practices address the core issues of the ad hoc approach to automation.

For Gartner clients with impactful automation practices, this shift takes the form of treating all forms of automation as value-centered products, applying robust and rightsized software development life cycle practices and being intentional with the automation architecture, implementation and metrics. Replicate their automation successes with the following:

- Use DevOps and development practices to make automation a resilient foundation for IT and business.
- Use product value streams to make data-driven decisions.
- Align automation practices and use cases to minimize technical debt and defend ROI.

Planning Considerations

Use DevOps and Development Practices to Make Automation a Resilient Foundation for IT and Business
For over a decade now, CI/CD has been the gold standard for software delivery. CI/CD employs a fully instrumented, staged delivery pipeline designed to implement a series of changes in sequence while also validating each action before proceeding to the next one. Its action can be continuous because each change is tested for production readiness before being deployed. Therefore, with CI/CD, incremental changes can be deployed rapidly but safely. In software development, the quality and throughput benefits of CI/CD are well-understood (though perhaps not widely deployed).

The DevOps movement aims to extend CI/CD to infrastructure. Indeed, a principal goal of DevOps is to manage infrastructure as code by building infrastructure automation pipelines based on CI/CD techniques. Gartner calls this continuous infrastructure automation (CIA), and it represents state-of-the-art infrastructure automation. Forward-looking organizations have achieved transformative successes by adopting CIA, and CIA has only grown more important as infrastructure automation has assumed a place at the heart of modern operational models.

But infrastructure automation is not the only domain in which CI/CD techniques are being adopted. DataOps, for example, extends CI/CD techniques to data and analytics pipelines. Data engineers face many of the same challenges as infrastructure engineers when building reusable automations in a collaborative workspace, and therefore achieve many of the same benefits from CI/CD pipelines. Furthermore, CI/CD patterns are beginning to emerge even in general-purpose business process automation. Low-code/no-code tools still generate artifacts that must be managed and maintained. Critical business processes may come to depend entirely on humble scripts or excel macros; CI/CD techniques manage these like the production software they are.

Figure 8 illustrates the CIA workflow, which has eight sequential steps:

1. Trigger the workflow via a change to a software version control repository or via a change management system.

2. Query the source of truth to obtain the context data the automation pipeline needs to execute.

3. Build a package, such as a configuration snippet that will affect the desired change. The package should also include a rollback procedure.

4. Validate the package via an automated suite of unit tests designed to evaluate its effectiveness and safety.
5. Deploy the change package using a fleet of tools incorporated into the pipeline.

6. Validate the change holistically, via an automated suite of integration tests designed to evaluate its production readiness.

7. Notify the relevant parties that the change has occurred.

8. Update the source of truth, as needed, to account for the changes made.

**Figure 8: The CIA Workflow**

When done right, pipeline-based automation can increase stability and availability both during and after a change. The automated test suite also forestalls potential problems before they occur and helps in troubleshooting bugs that do make it into production.
In 2022, I&O continues to expand the use of DevOps techniques, replacing older automation methods such as scripting. Meanwhile, data engineers and automation architects should explore extending CI/CD techniques to their own domains. This pervasive automation translates directly to business agility and faster time to market. There are many secondary benefits, as well, like improved reliability by ensuring that every change is subject to the same type and degree of testing and improved transparency through version control.

Recommended Research

- Solution Path for Infrastructure Automation
- To Automate Your Automation, Apply Agile and DevOps Practices to Infrastructure and Operations
- Demystifying XOps: DataOps, MLOps, ModelOps, AIOps and Platform Ops for AI

Use Product Value Streams to Make Data-Driven Decisions

In response to the demands for consistent quality, rapid response and business autonomy, the role of I&O is transitioning from “building widgets” to “building factories that build widgets.” In this context, there are some techniques that we can borrow from physical factories. One of the main lessons IT practitioners have learned from looking at physical factories is the factory’s intense scrutiny of each product’s value stream.

The value stream is the specific sequence of activities needed to deliver customer value. It represents the end-to-end process stretching from the initial customer request, through product design, through the delivery of functional software, to the ongoing maintenance and updating of the software in production. Managing the value stream means creating an integrated and highly automated delivery pipeline that enables the flow of work across the entire product life cycle.

High-functioning factories and, increasingly, high-functioning IT product teams, place central importance on value streams. They use value stream metrics to assess quality, delivery velocity and risk, among other critical markers of success. Value stream metrics, as expressed by tools such as real-time dashboards, alerts, reporting and interactive data exploration, result in data-driven decision making.

Figure 9 is a summary view of a product value stream and a six-step process for using value stream mapping to find and prioritize automation use cases, based on Goldratt’s Theory of Constraints.
The concept of a value stream is disarmingly simple, even commonsensical. Yet most IT organizations have no such holistic view of product delivery. Product pipelines remain fractured across many different people and teams. Often, there are multiple independent pipelines, each specific to a person or group, with little integration and even less visibility.

Value stream thinking promises a better way. It emphasizes transparency and visibility across the entire value stream. The idea has gained real currency in the DevOps community; more and more practitioners are focusing on value streams as a way to find bottlenecks, manage defects, eliminate redundancy or waste, improve cross-functional collaboration, and more. In the "15th Annual State of Agile Report," for example, over half of the respondents indicated they have either adopted value stream metrics already or plan to do so.

There is even an emerging ecosystem of DevOps value stream delivery platforms (VSDPs). These aim to provide automation throughout the entire continuous delivery pipeline — from creating project plans and backlogs to builds, packaging and releases — all expressly oriented around the value stream. This market is immature, and none of the tools has yet reached the goal of a full-life cycle VSDP. But familiar DevOps tools from vendors such as Atlassian, GitLab and JFrog are all trending in this direction.
If your DevOps efforts are falling short of your vision, adopt value stream metrics to identify and target opportunities for improvement. This will help identify process impediments, making clear which barriers need to be broken down and improving efficiency and time to value of product delivery. At the same time, a transparent value stream instills confidence and trust in end users, who can see clearly the process by which IT delivers value. By prioritizing the important issues that create constraints and bottlenecks, IT product teams can deliver faster and/or more frequently. This maximizes the business impact of their work.

Related Research

- Guidance Framework for Choosing What to Automate to Increase Application Delivery Agility
- Analyze Value Stream Metrics to Optimize DevOps Delivery
- Choose the Right Metrics to Drive Your Agile, DevOps and Continuous Delivery Initiatives

Align Automation Practices and Use Cases to Minimize Technical Debt and Defend ROI

Since automation can apply to nearly every aspect of an enterprise, automations can take many distinct forms. Business process automation, for instance, might rely on robotic process automation, business process management tools and low-code application platforms. IT automation, on the other hand, uses a different set of automation tools, like scripting in PowerShell or Python, HashiCorp Terraform, Jenkins, and run book automation tools. Across the variety of use cases, users and technologies, though, there are distinct domains with different tools, skills and capabilities.

While these different automation domains might have different practices for selection, development and maintenance, there are some key principles that transcend disciplines and create more impactful automations, such as:

- Treat automation assets as software products. Govern them with a complete software development life cycle that delivers the right type and degree of both risk and quality management.
- Optimize and standardize processes before automating them. A more reliable automation increases return on investment (ROI) and decreases total cost of ownership (TCO), which can offset the cost of any process improvements.
Focus on mechanical, not creative, processes. Straightforward, single-path automations are more reliable and easier to create and maintain than those that attempt to adapt to conditions within the process.

Choose the right automation tool for the use case. Align the use case to the tool. Since each approach will have different strengths and weaknesses, each tool will deliver a different ROI. Selecting the optimal automation approach ensures each implementation minimizes technical debt that can erode ROI.

The last principle, use case alignment, is becoming much more common going into 2022 and an important element in driving greater value from the automation practices. Gartner clients are finding that skipping this step drives up licensing costs, reduces reliability or creates technical debt that drives up maintenance. In low-maturity automation practices, developer preference and convenience often dictate the automation implementation. As an automation engineer, help mature the automation practice by defining which use cases are well-suited and poorly suited for your automation techniques. Develop the automation alignment framework by incorporating this definition into the broader practice and foster the flexibility to realign poorly suited use cases to where they might perform better.

There is no single, best automation tool and, in most cases, an automation might have a few different viable approaches, each with different development, maintenance and licensing costs. Therefore, identify, investigate and establish adjacent automation practices that address gaps in your automation portfolio and provide better ROI where existing practices aren't meeting goals. Use rejected use cases and poorly performing automations to justify the additional practice.

The practice of establishing the capability and optimizing the application of multiple automation technologies is called “hyperautomation” and can be the difference between just automating and automating well.
Remember, too, that business constraints, like time to market, might force a suboptimal implementation and create some technical debt. This happens all the time, usually because of a resource constraint within the development team. It is important, when you make this kind of compromise, to adjust the metrics and KPIs for the automation. For example, defining success based on the quality of an automation whose top priority was time to market will usually reflect poorly.

Several Gartner clients have already begun their hyperautomation journey and use the use case alignment process to provide the following benefits:

- More reliable automations by alleviating the constraints of developer bias
- Lower TCO by reducing technical debt
- Stronger and more visible ROI by adapting KPIs when compromising implementation

Related Research

- Decision Point for Process Automation Platforms
- Essential Skills for Automation Engineers
- Essential Skills for Automation Architects

The Growing Need for Both Increased Observability and Operational Efficiency Will Accelerate APM, DEM and AIOps Adoption

The term “observability” has become a buzzword, but few know what it really means in practice. Many monitoring vendors use the term observability as if it were a capability or feature. More often than not, they are using “observability” as a synonym for monitoring. Observability is a property, rather than a process. You do monitoring; you don’t do observability. Instead, observability exists when the state of a system is sufficiently expressed that meaningful problems are captured in enough detail for them to be understood and resolved. Therefore, a monitoring solution, alone, cannot provide observability; instead, monitoring tools contribute insight that creates an understanding of state. Establishing observability will be an important step in enabling the visibility and forensics across assets, environments and products without having to inspect every piece of telemetry.
At the same time, agile methodologies have become common, creating a constantly changing catalog of assets and requiring nonstop adaptation of monitoring tools and practices. One of the ways that I&O teams are tackling this churn is to shift the focus from an infrastructure-centric to a customer-centric mindset. This perspective provides more stability and reduces the volume of instrumentation. Without this, telemetry and alerts can grow out of control and eventually limit our ability to assess the impact of anomalies. In a world of digital business and business-led IT initiatives, data analytics and automation will become critical strategies for improving visibility. In the years ahead, scaling IT through people and manual processes will become a competitive disadvantage. Therefore, begin augmenting and accelerating human labor with automation and artificial intelligence.

While the full promise of AIOps may be years away, there are immediate use cases and investments that can facilitate the realization of that promise over time. Therefore, in 2022, IT operations professionals should begin to focus on the following:

- Combine modern application and infrastructure tools to shift toward monitoring user experience and enabling observability.
- Use the combination of AIOps tools, features and platforms to capture immediate value from AIOps with minimal investment.
- Supplement your operations team with SRE practices to support DevOps teams.

**Planning Considerations**

**Combine Modern Application and Infrastructure Tools to Enable Observability**

Siloed approaches to monitoring do not meet the agility requirements of modern application and infrastructure observability. Modern application architectures spanning multiple cloud providers, data centers, and teams need unified visibility and context across all monitoring domains. Today’s monitoring tool landscape (APM, DEM, NPMD, ITIM and AIOps), where each tool has siloed storage, makes it difficult to support this cross-domain visibility, as shown in Figure 10.
Modernize application and infrastructure monitoring tools, enable observability and prepare for unified analysis with AIOps by consolidating data collection, storage, and analysis across metrics, logs and traces.

Modern monitoring and observability methods include combining the golden signals from various data sources, such as:

- **Metrics**: For many monitored elements, the most visible golden signals come from time-series metrics. A wide variety of tools are available to gather the metrics data generated by infrastructure and applications, such as Prometheus. These time-series metrics can be combined with other signals to be viewed with cross-domain visualization tools such as Grafana.
Platforms that collect and analyze all three types of instrumentation can rightly be characterized as observability tools. However, no tool can “add observability,” especially to a system that does not generate data necessary to understand the pathological behavior. Therefore, prioritize monitoring tools that extract the most diverse set of metrics, logs or traces. Store this data centrally, where possible, and apply analytics and AIOps to help address increasing data volumes.

**Related Research**

- Guidance Framework for Deploying Centralized Log Monitoring
- Monitoring and Observability for Modern Services and Infrastructure
- Solution Path for Modern Infrastructure and Application Monitoring

**Use the Combination of AIOps Tools, Features and Platforms to Capture Immediate Value From AIOps With Minimal Investment**
AIOps has many meanings and takes many forms. Everything from chatbots to monitoring tools is called “AIOps.” For now, we’ll focus on AIOps use cases that analyze operations data and telemetry to improve IT service delivery and operations.

The first step for I&O is to see the range of AIOps solutions as three distinct categories:

- **AIOps tools** — These are solutions that enable a skilled operator to create targeted and custom analyses. Log file analysis tools, like Splunk Enterprise and Elastic, fit into this category, along with other data analysis tools, business intelligence (BI) tools and DIY machine learning approaches. These solutions require specific skills and are best used when you understand the data involved and how that data relates to the problem at hand. Most Gartner clients report that their I&O team has at least one AIOps tool but most struggle with the complexity of building ambitious solutions that capture much of the potential of AIOps.

- **AIOps features** — Today, just about every monitoring tool is laying claim to the “AIOps” label. These tools have incorporated some anomaly detection, correlation or classification into their monitoring domain to provide a more dynamic analysis of telemetry than static thresholds. In general, these solutions work well in their domain and leverage the domain expertise to capture useful patterns within the monitoring data. The challenge with AIOps features is that they are inherently limited to the scope of the monitoring. This means that, while they might do a great job of finding localized anomalies, they lack the breadth of data required for data-center-wide AIOps and advanced use cases like intelligent remediation. AIOps features are a great way to capture some tactical value without having to rely on data science skills to build tailored solutions.
**AIOps platforms** — For most clients, AIOps platforms come closest to delivering on their AIOps vision. AIOps platforms consume data across monitoring and operational tools. Monitoring tools send their events, which might be triggered by an AIOps feature within the tool, to the AIOps platform. AIOps platforms also integrate with other operational tools — like CMDB, CI/CD orchestration, incident management systems and more. The AIOps platform assembles a topology from these data sources and learns event patterns associated with incidents and, through integration with incident management, learns the human response to these incidents. AIOps platforms don’t replace monitoring tools or human operators. Instead, they augment the human capacity for detecting cross-domain issues and accelerating the response by highlighting similarities to previous incidents. Over time, as confidence increases through additional observations, the AIOps platform can trigger an automated response to a particular event pattern, leading to automated, even predictive, remediation.

Start architecting your AIOps environment by classifying existing and potential AIOps solutions into these three categories. This classification will clarify the role in the overall solution, the limitations and the capabilities. Then, evaluate the capabilities to see whether an AIOps platform is required or whether the silos of AIOps features will meet your immediate needs for AIOps solutions. Figure 11 shows common capabilities of AIOps solutions and how they can be layered to take advantage of the tools, features and platforms.
Figure 11. AIOps Layered Architecture

Keep in mind a few critical points — the AIOps platform impact is only as good as your data. Limiting the data sources to silos of telemetry will create blind spots and limit the ability to correlate across additional dimensions. Also, AIOps solutions that use machine learning will require observations, over time, to identify useful patterns. In some cases, particularly for rare events, this will delay the time to value. Expect the most immediate impacts from more consistent and frequently occurring incidents.

Related Research

- Solution Path for Adopting AIOps
- Solution Criteria for AIOps Platforms

Supplement Your Operations Team With SRE Practices to Support DevOps Teams
Site reliability engineering (SRE) emerged within Google in 2003 as a novel approach to service management designed to reduce friction and foster collaboration between product development, operations and business teams. This is exactly what many I&O teams are facing today. Since its unveiling, SRE has developed into a discipline that enables cross-functional teams to design and operate scalable, resilient systems with complex modern architectures. Gartner defines SRE as a collection of systems and software engineering principles used to build and operate resilient distributed systems at scale. SRE practices help organizations manage operational risks by promoting resilience, accountability and innovation.

Key callouts on SRE principles are:

- Designed with data-driven operating philosophies.
- Balance business risks and reliability to drive innovations faster.
- Approach everything from a customer lens that prioritizes product quality and experience.
- Strive to provide freedom from the operational burden by applying software engineering techniques to solve operational problems.
- Adopted widely by “cloud-first” organizations as part of their digital transformation journeys.
- Better-suited for high-velocity custom software development environments.

Benefits of establishing, adapting and adopting SRE principles include:

1. **Customer-centric objectives**: SRE principles provide a framework for defining business-centric objectives and outcomes, and they reduce silo tension between development and operations teams. In this environment, engineers gain in-depth knowledge of service and its dependencies, which include hardware, network, storage and upstream service calls that enable the ability to analyze incidents quickly and efficiently with low dependency on development teams.
2. **Ensures application reliability for high-velocity software releases**: SRE principles enable software delivery velocity by defining service-level indicators, service-level objectives, error budgets and actionable monitoring. This enables organizations to focus on growth and improve the customer experience. With the heavy focus on reliability and resiliency of the applications, SREs enable organizations to be more agile and fail forward.

3. **Scale operations without directly relying on staff expansion**: With heavy focus on automation, engineering work and a self-service mindset, SRE provides a scalable operating model that does not need to scale operations staff proportionally with organizations’ systems and application growth.

4. **Quantifies effective monitoring and system availability**: Organizations often struggle with targets for availability and performance, as well as how to measure them. This is exactly the problem that SLOs seek to resolve. By ensuring that service-level indicators are tied to achievable SLOs, product and platform operations teams have measurable objectives and can understand at a glance whether they are being met.

Organizations must make a thorough assessment of strengths, weaknesses and commitments of SRE first. SRE does not solve all operational problems and is not a magic bullet. Evaluate SRE principles for organizational cultural fit, and create a list of drivers for SRE before embarking on the SRE journey. Be strategic in introducing SRE because, if the teams do not understand the SRE concepts well, the natural tendency is to reject. Start with a single-app pilot and then apply SRE to additional applications.

**Related Research**

- Assessing Site Reliability Engineering (SRE) Principles for Building a Reliability-Focused Culture
Increasing Self-Service, Autonomy and Decentralization Will Require Mature and Multidisciplined Governance and ITSM Practices

The core concepts behind IT management frameworks are not lost in this era of distributed environments, business autonomy and DevOps agility. There are still “incidents” in the most mature DevOps environments. There are still software changes that need to be managed. The difference is that, now, these processes have different pipelines, workflows and checkpoints. Likewise, inventory tracking, knowledge and problem management and other IT service management goals are still relevant even in a fully cloud-native environment. These more modern operating models demand more automation and integration and less centralized teams and manual processes.

In 2022, IT operations professionals must embrace automation and governance to achieve the right balance between speed and risk. Focus your automation, integration and governance efforts on the following areas in the months ahead:

- Define governance principles and build programmatic controls or policies to enforce them.
- Inject policy-based governance into application and infrastructure deployment pipelines.
- Evolve ITSM practices to meet the demands of digital business.

Planning Considerations

Define Governance Principles and Build Programmatic Controls or Policies to Enforce Them

One of the most important goals of governance within an organization is to prevent chaotic or reckless behavior on the part of the organization as a whole or any of its members individually — whether by accident or on purpose. However, discussions about IT decision rights and authority/autonomy can be tough conversations. As a result, these topics often simply go unaddressed until so many problems and conflicts have emerged that they can no longer be ignored.

Any hope of achieving comprehensive governance in the era of cloud computing and digital business will require broad agreement on a shared set of “principles" within the organization. Every member of the organization should not only agree to these principles, but feel a sense of responsibility to follow them in their day-to-day decision making. For example, a core principle for a European organization that does no business outside of the EU may state something like "As an EU business with only EU operations, our systems and data should never reside outside of the EU." Well-designed principles should:
Once common governance principles have been developed and agreed on (which can be a challenge), implement programmatic controls to enforce them, wherever those controls are available. In the previous example, the major IaaS and PaaS cloud providers allow administrators to programmatically limit which data center regions are available to a given user.

However, not every governance principle developed by the organization may have a readily available programmatic control mechanism that can be easily implemented. In those circumstances, define and implement corporate data protection policies to fill the gap. Develop these policies in collaboration with human resources to have substantive enforceability (much like a “dress code policy” is ultimately backed by HR enforcement if ignored).

**Related Research**

- Solution Path for Public Cloud Governance

**Use Policy as Code to Enforce Security and Compliance**

In the most mature delivery pipelines, I&O engineers spend most of their time on optimization, governance and compliance. They no longer build and maintain infrastructure. That work has all been automated and turned over to end users through self-service. In this environment, developers and product teams create their own infrastructure, while Ops ensures that infrastructure conforms to architectural standards, corporate policies, industry regulations and — not least — budgets.
Ops teams must now enforce a set of policies that control the behavior of both users and infrastructure. To further this important work, a new generation of policy as code (PaC) tools has emerged. PaC expresses governance and compliance rules as code, so they can be enforced programmatically by automation tools. PaC languages are often domain-specific and declarative. With PaC, policies are treated as software, making them subject to version control, code review and functional testing. The most mature PaC tools can render any business logic in code. You can use them to enforce architectural standards, corporate policies, regulatory requirements, budgets and more. There is a growing market of dedicated PaC tools, including Chef InSpec, the Cloud Native Computing Foundation's Open Policy Agent, and HashiCorp Sentinel. The major hyperscale cloud providers also all have their own native policy engines.

Policy as code is the natural progression from infrastructure as code (IaC). IaC languages define what the state of the infrastructure is; PaC languages define what the state of the infrastructure should be. Both are equally important.

Compared to IaC tools, PaC tools are generally less capable and less mature. But PaC tools, like Chef InSpec or the Open Policy Agent, are now production-ready for certain use cases. You still need to start small and target well-understood workflows that have ready-made implementation templates. (Ideally, there would be publicly available downloadable content for your chosen PaC tool.) But this describes an increasing number of critical security- and compliance-focused use cases.

In 2022, I&O, security and compliance teams should collaborate to deploy PaC for the following use cases:

- Enforcing CIS benchmarks
- Kubernetes ingress controls
- API authorization
- Triggering role-restricted automation tasks/workflows
PaC use can be expanded over time. As the tools mature, the library of downloadable content expands, and users grow more comfortable. This will align security and compliance teams to interface directly with automation pipelines to ensure conformance and visibility, while reducing the overhead of creating and enforcing policies. As more and more companies embrace DevOps and DevSecOps, they inevitably encounter the hard problems of mature automation pipelines. PaC can automate pipeline governance, helping these new and better operating models succeed and thrive.

Related Research

- Using Cloud-Native ‘Policy as Code’ to Secure Deployments at Scale
- How to Protect Your Clouds With CSPM, CWPP, CNAPP and CASB
- “Policy as code” innovation profile in Hype Cycle for I&O Automation, 2021

Evolve ITSM Practices to Meet the Demands of Digital Business

In 2022, IT operations professionals should continue to adapt IT service management in order to optimize service delivery and achieve the right balance between speed and risk. Given the scope, this is no small task, but there are some impactful areas that should be at the top of the priority list. When searching for opportunities to optimize ITSM, IT operations professionals should:

- Automate incident management
- Enable self-services
- Reduce incident volume and impact with problem management
- Balance speed and risk with change management
- Decentralize configuration management

Automate Incident Management

Organizations that rely solely on centralized, manual incident management processes will be overrun with the volume of incidents and struggle to meet increasingly stringent service-level objectives. Labor-intensive support models are expensive and unresponsive to fluctuating demand, and can lead to excessive downtime.
Organizations should automate incident routing, response, triage and resolution. Automating these four cases is an effective way to both increase the speed and capacity of support and improve key performance indicators, like service availability, mean time to respond and mean time to repair.

Take the following actions to begin augmenting your support staff and incident management processes with automation:

- **Automate incident routing and response.**

- **Shift high-volume, manual incident management tasks to automation by first introducing human in the loop (HITL) automation, then human out of the loop (HOOTL) automation.**

- **Prioritize the automation of tasks that will free up resources that can be reinvested back into the organization to accelerate your strategic initiatives.**

Gartner recommends automating incident response processes with solutions like Opsgenie, PagerDuty, VictorOps and Everbridge. Figure 12 shows an overview of common incident response automation capabilities.
Enable Self-Services

Improve IT service speed, convenience and efficiency by empowering end users with knowledge-backed self-services and an easy-to-use IT product and service catalog. Self-service offers a convenient, cost-effective alternative to live agent support and reduces the demand on IT staff.

Take these steps now to bolster your support model with self-services:

- Leverage your ITSM platform to drive the adoption of self-service and to decentralize the management of self-service solutions, like the knowledge base and the IT Product and Service Catalog.
- Create knowledge artifacts for the most common issues to reduce support contacts and cost.
- Deploy an intuitive IT product and service catalog that helps end users get the solutions they need, when they need them, without manual IT effort.
Reduce IT Incident Volume and Impact With Problem Management

When a problem management practice is singularly focused on the most severe incidents, the cause of lower-severity incidents is never addressed and new incidents pile on old problems. Eventually, there are more things to fix than there are people to fix them, and IT resources become consumed with closing recurring incidents. These issues show up as missed project deadlines, breached service-level agreements and dissatisfied customers.

However, when a problem management practice is used to actively seek out and address the causes of recurring issues, the volume and impact of incidents can be dramatically reduced. Organizations must address the causes of their incidents to reduce the number of recurring incidents and their impact on the organization.

Take these steps now to reduce the volume and impact of future IT incidents:

- Transition to a problem management practice that actively seeks out and addresses the common causes of high-volume incidents. Eliminate these issues to improve the health of your services and to reduce the demand on your support staff.
- Utilize a simple method for identifying and prioritizing problems with your domain of control.

Balance Speed and Risk With Change Management

Organizations must balance the speed/accuracy trade-off by defining what optimal means relative to your organization's tolerance for risk. Overly restrictive change controls hamper innovation by slowing down the release cycle. If controls are too loose, the risk may exceed the risk threshold of the business. Finding that balance is critical.

Gartner’s approach to balancing speed and risk is based upon the following near-term actions:

- Identify highly successful, low-risk changes and transition them to preauthorized standard change models, see Figure 13. Using standard change models will increase the velocity of change and increase change reviewer capacity.
- Leverage dependency mapping to document the relationships between applications and infrastructure. Use this information to assess the potential impact of change to determine risk.
- Begin automating high-volume change review and coordination tasks to increase the velocity of change management while driving down the risk of individual changes.
Figure 13: Identifying Candidates for Standard Change

Identifying Candidates for Standard Change

| Success Rate | Number of Reviews | Source: Gartner 720845_C |

Decentralize Configuration Management

Traditional configuration management databases (CMDBs) struggle to keep pace with the rate of change brought about by cloud-based solutions and agile practices. Subsequently, organizations should begin transitioning to a federated model that leverages multiple sources of configuration data to provide an up-to-date and complete view of the environment.

To accomplish this, recast the CMDB as part of a system of multiple configuration data sources. In this configuration management system (CMS), the CMDB contains configuration item (CI) records that have relatively static attributes and references to sources of dynamic attributes. By retaining the CI record, the CMDB is able to meet the requirements of workflows that rely upon CI records. And, by pointing to the external sources for dynamic data, the CMDB provides a reference to accurate, up-to-date configuration data.

Gartner’s approach to moving to a CMS is based upon the following near-term actions:

- Reduce overhead by eliminating unnecessary attributes from CI records.
- Link CI records to authoritative sources of dynamic attributes to federate configuration data.
Because most organizations will need to serialize investments and won’t be able to tackle everything outlined in this 2022 Planning Guide, they will need to prioritize. The assumption in developing prioritization is that democratization of IT and automation/speed are the overarching priorities.

Where that is true, here are priorities for investment and focus:

- Determine how overall organizational governance goals can be maintained in a diverse landscape of IT- and business-owned applications in the future. These challenges are often more organizational and political in nature, and often are not easily solved due to conflicting internal priorities and perceptions.

- Decide on what is the right cloud-operating model for your organization — whether it is more COTS-centric, DevOps, SRE, Platform Ops or some other model. Differing models can work together but require coordination in order to do so effectively.

- Elevate automation to a “must have.” Although automation does take extra effort, it should be viewed as an investment that will pay off in the years ahead — much like properly commenting source code.

**Related Research**

- ITSM Best Practices: How to Optimize IT Incident Management With Automation
- ITSM Best Practices: Problem Management
- Automate Incident Response To Enhance Incident Management
- How to Implement a Modern IT Change Management Practice
- How to Modernize the Configuration Management Database

**Document Revision History**

Recommended by the Authors

Some documents may not be available as part of your current Gartner subscription.

- Assessing Kubernetes for Hybrid and Multicloud Application Portability
- Solution Path for Implementing Containers and Kubernetes
- Guidance Framework for Implementing Cloud Platform Operations
- How to Manage and Optimize Costs of Public Cloud IaaS and PaaS
- Solution Path for Infrastructure Automation
- Guidance Framework for Choosing What to Automate to Increase Application Delivery Agility
- Analyze Value Stream Metrics to Optimize DevOps Delivery
- Solution Path for Modern Infrastructure and Application Monitoring
- Assessing Site Reliability Engineering (SRE) Principles for Building a Reliability-Focused Culture
- Solution Path for Public Cloud Governance
- Automate Incident Response to Enhance Incident Management
- How to Implement a Modern IT Change Management Practice
- How to Modernize the Configuration Management Database
Actionable, objective insight

Position your IT organization for success. Explore these additional complimentary resources and tools for I&O and IT leaders:

**Resource Center**
**Cloud Strategy**
Discover insights, advice and tools to help address your top challenges for cloud.

**Roadmap**
**2021-2023 Emerging Technology Roadmap**
Make technology investment decisions with confidence.

**Tool**
**IT Score for I&O**
Evaluate I&O capabilities to drive better business outcomes.

**eBook**
**2022 Leadership Vision for Infrastructure & Operations**
Explore a data-driven view of 3 strategic priorities I&O leaders must act upon.

Already a client?
Get access to even more resources in your client portal. Log In

Gartner®
Connect With Us

Get actionable, objective insight to deliver on your most critical priorities. Our expert guidance and tools enable faster, smarter decisions and stronger performance. Contact us to become a client:

**U.S.:** 1 855 811 7593

**International:** 44 (0) 3330 607 044

Learn more about Gartner for IT leaders

gartner.com/en/it

Stay connected to the latest insights

Attend a Gartner conference

View Conference