Hype Cycle for Monitoring and Observability, 2023

Published 10 July 2023 - ID G00791091 - 110 min read

By Analyst(s): Pankaj Prasad, Padraig Byrne

Initiatives: I&O Operations Management

> Observability is becoming central to how modern businesses understand customer and employee experience. New technologies are emerging to deal with the complexity presented by operational telemetry. This Hype Cycle helps I&O leaders plan their observability initiatives for success.

Additional Perspectives

- Summary Translation: Hype Cycle for Monitoring and Observability, 2023 (25 September2023)

Analysis

What You Need to Know

This document was republished on 14 July 2023. The document you are viewing is the corrected version. For more information, see the Corrections page on gartner.com.

The rapid evolution and complexity of IT architectures is contrasted by the slow pace of development in the traditional monitoring tools market. Some new IPs in this Hype Cycle are indicative of how I&O leaders are unshackling their organizations from the rigidity of traditional monitoring approaches by adopting observability and a shift-left approach. Examples are taking control of instrumentation via an internal development portal (IDP), using monitoring as code to define metrics needed for analysis and leveraging telemetry pipelines to capture the resulting telemetry data at scale.

Continued digital transformation investment is driving comparatively higher growth in the digital experience monitoring (DEM) market due to the need for improved customer experience and better employee engagement. At the same time, challenges in realizing concrete value have driven AIOps platforms into the Trough of Disillusionment.

For more information about how peer I&O leaders view the technologies aligned with this Hype Cycle, see Infographic: 2023 Technology Adoption Roadmap for Infrastructure and Operations.

The Hype Cycle

This research is one of several Hype Cycles for I&O leaders focused on IT operations. I&O leaders should also refer to Hype Cycle for ITSM, 2023 and Hype Cycle for Agile and DevOps, 2023.

This Hype Cycle delivers advice on IT monitoring and observability practices, tools and technologies with information regarding their visibility and market adoption.

Gartner's 2022 Signature I&O Role Survey results show the top three goals of I&O organizations in the next 12 months are a focus on increasing agility, improving infrastructure resilience and alignment with business outcomes. In line with these results, there has been a corresponding growth of interest and investment in autonomous workload optimization, automated systems, and low-code and monitoring as code.

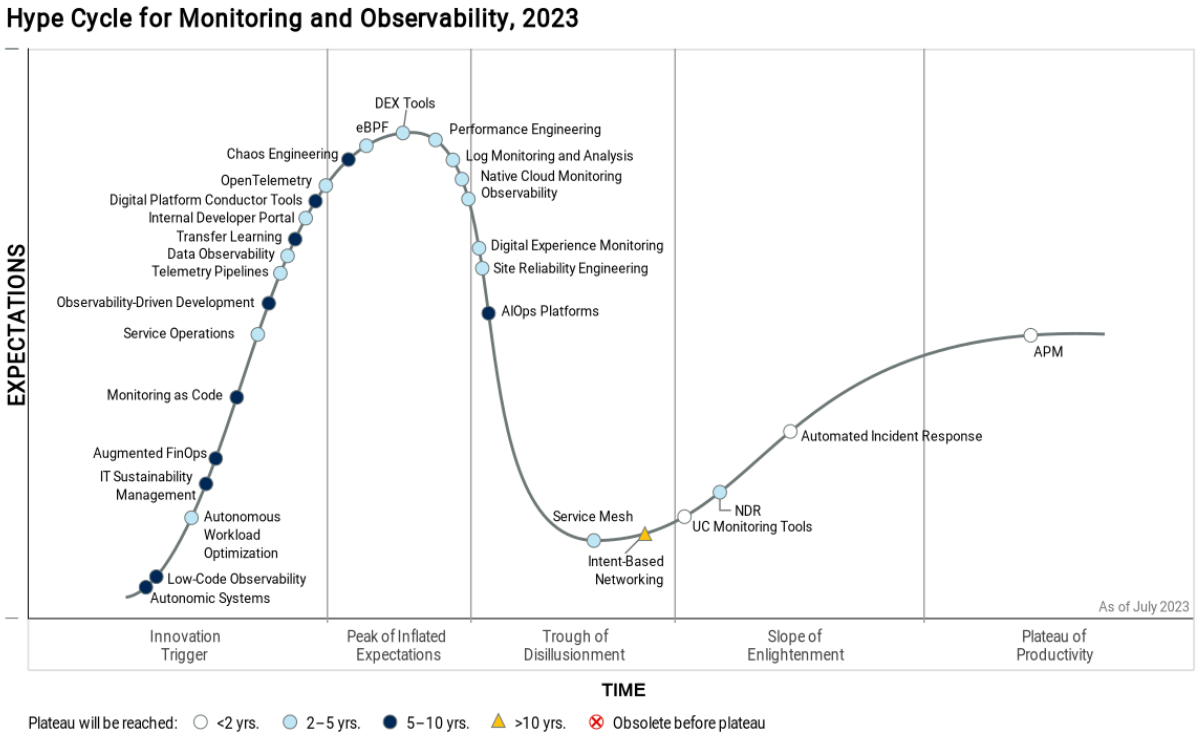
In parallel, the growing interest and efforts toward achieving environmental sustainability goals are seeping into IT and I&O leaders' remit. This is leading to their exploration of ways and means to monitor and measure IT's contribution in areas like power consumption and greenhouse gas (GHG) emissions.

This year we added eight new profiles to the Hype Cycle. These additions are reflective of the continued innovation and investment in making monitoring more efficient and easy to leverage across the entire life cycle from design, development, operations and automation to measuring value-add through sustainability. New profiles for 2023 include:

- Autonomic systems
- Autonomous workload optimization
- Internal developer portal
- IT sustainability management
- Telemetry pipelines
- Transfer learning
- Low-code observability
- Monitoring as code

This year, the profile “AI-enabled log monitoring” has been renamed to “Log monitoring and analytics” in response to the changes in the monitoring landscape, mainly log-analytics becoming commoditized in some existing technologies .

Figure 1: Hype Cycle for Monitoring and Observability, 2023



The Priority Matrix

The Priority Matrix maps a technology’s time to maturity on an easy-to-read grid format. It accounts for two critical considerations:

- The value an organization will receive from investing in a technology.
- The time required for the technology to be mature enough to provide this value.

For most profiles listed, the truly transformative impact is realized only by interlocking technology adoption with people and process frameworks that are aligned with clear business objectives.

Most of the transformational innovation profiles on this Hype Cycle are at the trigger or climbing the peak. The disillusionment with slow evolution of existing monitoring markets has led to the emergence of new technologies to enable monitoring across the life cycle of IT artifacts and applications. Early adopters will benefit from the ease of a shift-left approach that can be applied to monitoring starting from the design phase in DevOps through the adoption of internal developer portal and monitoring as code.

Very few innovation profiles are entering the trough this year. AIOps platforms is the most notable. Over the past few years, AIOps platforms enjoyed a relatively high demand, but the slower pace of innovation and longer time to value are causing enterprises to take a cautious approach when planning investments in this space.

The Priority Matrix illustrates innovation profiles of some monitoring technologies that are at the mature mainstream phase of adoption, for example application performance monitoring (APM), unified communications monitoring (UCM) tools, and network detection and response (NDR) — signifying a level of commoditization in technology. Mature IT organizations can successfully adopt these technologies and derive efficiency and reliability faster than indicated in this matrix. Unrealistic expectations and a lack of collaboration when leveraging some technologies like Observability will lengthen the time to adoption.

Table 1: Priority Matrix for Monitoring and Observability, 2023

(Enlarged table in Appendix)

Benefit ↓	Years to Mainstream Adoption: Less Than 2 Years ↓	2 - 5 Years ↓	5 - 10 Years ↓	More Than 10 Years ↓
Transformational		Observability Service Operations Site Reliability Engineering	Augmented FinOps Autonomic Systems Digital Platform Conductor Tools Low-Code Observability	
High	APM Automated Incident Response UC Monitoring Tools	Autonomous Workload Optimization Data Observability DEX Tools Digital Experience Monitoring Internal Developer Portal Log Monitoring and Analysis OpenTelemetry Performance Engineering	AIOps Platforms Monitoring as Code Observability-Driven Development Transfer Learning	
Moderate		eBPF Native Cloud Monitoring NDR Telemetry Pipelines	Chaos Engineering IT Sustainability Management	Intent-Based Networking
Low		Service Mesh		

Source: Gartner (July 2023)

Off the Hype Cycle

A few profiles have been dropped from this year's Hype Cycle:

- Virtual desktop infrastructure monitoring (VDIM) — The capabilities of VDIM are subsumed by digital experience monitoring and digital employee experience (DEX).
- Cloud management platforms (CMPs) — The market for this technology has matured onto the Plateau of Productivity.

- Distributed Tracing — This is one of the three flavors of telemetry under the Open Telemetry IP included in this Hype Cycle.

On the Rise

Autonomic Systems

Analysis By: Erick Brethenoux, Nick Jones, David Cearley

Benefit Rating: Transformational

Market Penetration: Less than 1% of target audience

Maturity: Embryonic

Definition:

Autonomic systems are self-managing physical or software systems, performing domain-bounded tasks, that exhibit three fundamental characteristics: autonomy (execute their own decisions and tasks autonomously without external assistance); learning (modify their behavior and internal operations based on experience, changing conditions or goals); agency (have a sense of their own internal state and purpose that guides how and what they learn and enables them to act independently).

Why This Is Important

Autonomic systems are emerging as an important trend as they enable levels of business adaptability, flexibility and agility that can't be achieved with traditional AI techniques alone. Their flexibility is valuable in situations where the operating environment is unknown or unpredictable, and real-time monitoring and control aren't practical. Their learning ability is valuable in situations where a task can be learned even though there is no well-understood algorithm to implement it.

Business Impact

Autonomic systems excel where:

- Conventional automation applying composite AI techniques is inadequate, or using fixed training data is impractical or not agile.
- It is impractical to provide real-time human guidance, or training conditions can't be anticipated.
- We cannot program the exact learning algorithm, but the task is continuously learnable.

- Continuously or rapidly changing tasks or environments make frequent retraining and testing of ML systems too slow or costly.

Drivers

Autonomic systems are the culmination of a three-part trend:

- Automated systems are a very mature concept. They perform well-defined tasks and have fixed deterministic behavior (e.g., an assembly robot welding cars). The Increasing number of use cases around automation using AI techniques is a strong base for autonomous systems.
- Autonomous systems go beyond simple automation to add independent behavior. They may exhibit some degree of adaptive behavior, but are predominantly under algorithmic control (e.g., self-driving cars or a Boston Dynamics' Spot robot 1 that has its overall route and goals set by a remote human operator but has substantial local autonomy over how it achieves them). Adaptive AI capabilities are a necessary foundation for autonomic systems and should accelerate the adoption of autonomic systems.
- Autonomic systems exhibit adaptive behavior through learning and self-modifying algorithms (e.g., Ericsson has demonstrated the use of reinforcement learning and digital twins to create an autonomic system that dynamically optimizes 5G network performance. It learns from network behavior and local conditions and adjusts software and physical network control parameters to optimize performance). This trend is showing the feasibility of such systems and early learning about carefully bounded autonomic systems will build trust in their capabilities to operate independently.

Longer-term drivers include:

- Autonomic behavior is a spectrum. For example, chatbots learn from internet discussions; streaming services learn which content you like; delivery robots share information about paths and obstructions to optimize fleet routes. The advantages of systems that can learn and adapt their behavior will be compelling, and many examples will involve physical devices.
- Substantial academic research is underway on autonomics, which will result in more widespread use.

Obstacles

- **Nondeterminism:** Systems that continuously learn and adapt their behavior aren't predictable. This will pose challenges for employees and customers who may not understand how and why a system performed as it did.
- **Immaturity:** Skills in the area will be lacking until autonomics becomes more mainstream. New types of professional services may be required.
- **Social concerns:** Misbehavior, nondeterminism or lack of understanding could generate public resistance when systems interact with people.
- **Digital ethics and safety:** Autonomic systems will require architectures and guardrails to prevent them from learning undesirable, dangerous, unethical or even illegal behavior when no human is validating the system.
- **Legal liability:** It may be difficult for the supplier of an autonomic system to take total responsibility for its behavior because that will depend on the goals it has set, its operating conditions and what it learned.

User Recommendations

- Start by building experience with autonomous systems first to understand the constraints and requirements (legal, technical and cultural) that the organization is subjected to. Pilot autonomic technologies in cases where early adoption will deliver agility and performance benefits in software or physical systems.
- Manage risk in autonomic system deployments by analyzing the business, legal and ethical consequences of deploying autonomic systems — which are partially nondeterministic. Do so by creating a multidisciplinary task force.
- Optimize the benefits of autonomic technologies by piloting them in situations, such as complex and rapidly changing environments where early adoption will deliver agility and performance benefits in either software or physical systems.

Sample Vendors

Adapdix; IBM; Latent AI; Playtika; Vanti

Gartner Recommended Reading

Top Strategic Technology Trends for 2022: Autonomic Systems

Low-Code Observability

Analysis By: Pankaj Prasad, Cameron Haight

Benefit Rating: Transformational

Market Penetration: 1% to 5% of target audience

Maturity: Embryonic

Definition:

Low-code observability enables easy and rapid configuration of monitoring and analytics tools by minimizing the need to write queries or scripts. It enhances observability with a model-based graphical approach that helps with flexible data capture, analysis and dashboard design. These features eliminate the need for training, while enabling self-service, rapid insights and customization for different use cases and personas.

Why This Is Important

As organizations deploy increasingly complex IT architectures and tools, infrastructure and operations (I&O) teams are becoming frustrated with the complexity of the required monitoring, observability and analytics tools, which demand more advanced technical and management skills. In extreme scenarios, the cost of managing these complex tools may exceed that of managing the IT architecture and IT tools themselves.

Business Impact

Low-code observability enables I&O teams to adapt better to emerging technologies and to pursue new use cases without major delays or backlogs. The self-service capability and agility it enables could help various users, including business leaders, make better decisions by providing relevant insights faster. Additionally, the reduced complexity associated with low-code observability can enable I&O teams to focus on “grow” and “transform” initiatives.

Drivers

- **Reduction in skills gap:** Low-code observability reduces the need for I&O teams to rely on complex monitoring and analytics tools, which require scarce management and maintenance skills, to deal with the increasing complexity of IT.
- **Speed and agility:** Low-code observability can increase organizations' speed and agility by means of back-end models and intuitive graphical front ends that help automate workflows and ease configuration.
- **Self-service:** I&O teams gain the ability to provide self-service observability and analytics, which enables other teams to get relevant insights faster.

Obstacles

- **Resistance from IT professionals:** IT professionals who view low-code technology in general as a threat to their influence will be reluctant to adopt low-code observability.
- **Vendor lock-in:** Much low-code observability capability is "trapped" within vendor platforms, which makes it difficult to transfer to other products.
- **Governance:** Most I&O teams lack the level of governance required to avoid the potential risk of data loss or service degradation while ensuring low-code observability spans multiple user personas and initiatives.

User Recommendations

- Be open to experimenting with low-code approaches to observability.
- Embrace and encourage a self-service culture.
- Work on governance to ensure adoption of low-code observability can mature beyond the proof-of-concept stage. For example, determine how to open up access to monitoring data, while overcoming the challenge posed by the need to anonymize private and sensitive data.
- Start small, but with a continuous review cycle to ensure a low-effort, high-value approach becomes a consistent practice.

Sample Vendors

Datadog; Dynatrace; New Relic; ServiceNow

Autonomous Workload Optimization

Analysis By: Pankaj Prasad, Manjunath Bhat, Hassan Ennaciri

Benefit Rating: High

Market Penetration: 1% to 5% of target audience

Maturity: Embryonic

Definition:

Autonomous workload optimization tools maximize performance while minimizing resources through one or more of the three approaches. First, resource and cost optimization, including minimizing compute resources. Second, performance, including optimizing code for running applications, e.g., fine-tuning configuration of the underlying runtime. Third, dynamic workload placement, including reshuffling jobs and allocating resources to workflows for maximum utilization and minimizing idle time.

Why This Is Important

Optimizing IT costs has always been on the agenda for infrastructure and operations (I&O) leaders. Additional pressures include becoming energy efficient and simultaneously ensuring optimal performance to customers. For the results to be impactful, this has to be done at scale while also dealing with the trade-offs required to optimize across the dimensions of cost, performance and energy. The scale and trade-off challenge is better addressed via autonomous operations and algorithms.

Business Impact

Automated workload optimization tools enable organizations to optimize their IT costs, and avoid wastage of IT resources, while balancing the performance requirements of applications to ensure that customers are not impacted. These tools can enable organizations to become more energy efficient, and in the process, take a step closer toward their sustainability goals.

Drivers

Once organizations achieve their targets for availability and performance, the next goal is optimization along a few dimensions:

- **Optimize performance**: Enterprises should aim for an optimal range of performance where cost is balanced while avoiding negative impact to customer experience, for example, by leveraging error budgets.
- **Optimize workloads and cost**: Mapping application resource utilization to demand patterns, especially in virtualized and cloud-native architectures, enterprises can dynamically project resource requirements thereby also optimizing their IT spend.
- **Energy efficiency**: I&O leaders are committing to sustainability goals and achieving greater energy efficiency is one of their primary targets. This is also a driver for sustainable software engineering.
- **Speed and scale**: Manual intervention in optimization goals will not be sustainable in the long run. The need to speed and scale is a driver for autonomous tools and processes.

Obstacles

- **Lack of maturity**: Ad hoc workloads and lack of appropriate tools to capture the right metrics are inhibitors toward realizing the full potential of workload optimization tools.
- **Standardization**: Many enterprises have a nonstandardized IT architecture where these tools will only tackle simple optimizations and require high efforts for significant results.
- **Collaboration challenges**: Data sharing between customer-facing, preproduction and production teams is crucial to ensure no negative impact due to optimization efforts. For e.g., to identify the appropriate error budgets and to ensure trade-offs do affect other operation areas like increase in errors.
- **Disconnect with business**: Business leaders' buy-in is crucial for continued investment, and is hampered due to a lack of appropriate reporting, i.e., translating resource optimization gains in dollar terms, proper tracking and trend analysis for comparisons.
- **Narrow focus**: Most of the tools in this space have a narrow focus on virtual operating systems or Kubernetes, limiting their adoption to workloads that leverage such architectures.

User Recommendations

- Start small and get some perspective on resource utilization vs. performance to identify patterns and correlations. Use this data to pilot autonomous optimization for non-mission-critical workloads to prove value and capability before rolling out autonomous optimization for all applications.
- Use autonomous optimization as a driver for improving maturity in monitoring metrics and to speed up standardization initiatives across IT architectures and processes to maximize the value of optimization objectives.
- Improve collaboration across customer-facing, preproduction and production teams to ensure regular review and appropriate data exchange, including, but not limited to customer engagement, performance, IT resource requirement and utilization patterns, and capacity.
- Collaborate with business leaders and IT stakeholders to ensure measurements and benefits are suitably conveyed and business and technical objectives are appropriately captured.
- Optimize continuously, which means reviewing results, revisiting targets and refining goals since this has to be an ongoing effort.

Sample Vendors

Akamas; CAST AI; Cisco Systems (Opsani); Control Plane; Google; Granulate; IBM; Sedai; StormForge

Gartner Recommended Reading

Market Guide for Digital Platform Conductor Tools

Market Guide for AIOps Platforms

IT Sustainability Management

Analysis By: Autumn Stanish

Benefit Rating: Moderate

Market Penetration: 5% to 20% of target audience

Maturity: Emerging

Definition:

IT sustainability management refers to both the practice of using IT monitoring/management tools for sustainability, as well as new and emerging solutions designed specifically for the purpose of measuring the environmental footprint of IT. Specific functionality includes asset life cycle optimization, energy consumption and efficiency tracking, and optimized workload balancing.

Why This Is Important

Enterprise sustainability programs require technology solutions to accelerate and scale reporting and performance against environmental impact reduction goals. New IT monitoring tools are emerging to offer simple, clear and transparent ways to both measure and action sustainable IT objectives.

Business Impact

Stakeholders continue to demand better reporting and transparency of the environmental footprint of IT operations. IT leaders can use IT sustainability monitoring to easily track and report on IT's efforts and progress toward their organization's sustainability goals. Most include data integration capabilities feed data into ESG tools and reporting frameworks to significantly reduce overhead.

Drivers

- Sustainability has become a large societal concern globally, which is driving examination of environmental performance in all aspects of the business — especially IT.
- Internal carbon taxes are placing pressure on IT leaders to identify and reduce technology-related carbon emissions.
- The IT industry has not yet standardized sustainability metrics, making it difficult for IT leaders to create a baseline measurement and prioritize efforts.
- IT sustainability monitoring tools are emerging from various hardware and software vendors to help organizations easily collect emissions and energy consumption data to inform sustainability initiatives and investment decisions.
- In response to the dual need for reduced costs and improved environmental performance, monitoring tool vendors also include the ability to track and reduce the energy consumption of employee devices, and extend device lifespans to avoid unnecessary purchasing.
- IT leaders need more real-time analytics and data-integration to streamline reporting up to the ESG practice and reallocate their time toward delivering on their sustainable initiatives.

Obstacles

- Energy efficiency calculations are based on each country's averaged energy grid mix, therefore, there is a varying margin for error depending on the region and access to renewables.
- Most tools will only report on scope 1 and 2 emissions. Scope 3 calculations are considered in some specific use cases, however, they are absent for most devices given the inaccuracy of product carbon footprint calculations and complexity of IT equipment life cycle measurement.
- The cost and overhead of adding another tool into the environment.
- The efficacy of the tools in accurately measuring across all types of technologies.
- Many IT leaders are still challenged by availability, performance and customer experience. To prioritize sustainability, more maturity and stability is needed in I&O.

User Recommendations

- Check with any existing data center infrastructure monitoring (DCIM) and/or digital employee experience management (DEX) tool vendors to see if they have a module or add-on product that can easily begin tracking the energy, waste and emissions impact of this equipment. The two biggest contributors to IT greenhouse gas emissions are employee devices and servers.
- Question the sources of inputs. Ensure that any data averages used in the tool are tailored to the device or facility's specific make, model and region.
- Ensure the tool can integrate its data with existing ESG-tracking software in the organization to streamline reporting.
- Prioritize solutions that can provide recommendations and automation, rather than just monitoring.

Sample Vendors

1E; Dynatrace; IBM; Lakeside Software; Nexthink; SentryOne

Augmented FinOps

Analysis By: Adam Ronthal, Dennis Smith

Benefit Rating: Transformational

Market Penetration: Less than 1% of target audience

Maturity: Embryonic

Definition:

FinOps applies the traditional DevOps concepts of agility, continuous integration and deployment, and end-user feedback to financial governance, budgeting and cost optimization efforts. Augmented FinOps automates this process through the application of artificial intelligence (AI) and machine learning (ML) practices — predominantly in the cloud — to enable environments that automatically optimize cost based on defined business objectives expressed in natural language.

Why This Is Important

In the cloud, it is now possible to assess the cost of a specific workload or collection of workloads assigned to a project. However, price/performance — the primary measure of cloud efficiency — is difficult to assess due to the complexity and diversity of choice in underlying cloud infrastructure and service offerings and a lack of consistency in pricing models. Augmented FinOps can automate this process by applying AI/ML techniques.

Business Impact

The automation of cloud budget planning and financial operations will allow businesses to express their objectives — ideally in natural language — and allow their cloud ecosystems to automatically optimize the underlying cloud resources to meet those objectives. This will result in more efficient use of resources and, therefore, optimal spend by reducing/eliminating misaligned or poor use of cloud infrastructure and service offerings.

Drivers

- Practitioners are increasingly realizing that cloud is fundamentally a complex cost optimization exercise.
- Cloud adopters have a strong desire for transparency into cloud spending.
- Buyer inexperience is leading to either under-provisioning and associated resource contention or overprovisioning and spending more than is needed.
- Vendors are positioning cost-effectiveness as a competitive differentiator in their go-to-market strategies.
- Practitioners need to reduce the unpredictability of cloud spending when using cloud infrastructure and services for analytics, operational database management systems (DBMSs), data lakes and other applications, including custom IT infrastructure.
- Consumption-based usage remains common in earlier stages of cloud adoption, driving the need for augmented FinOps, although commit-based usage mitigates some unpredictability.
- Cost overruns are often obscured, downplayed, or dismissed by line of business implementers, requiring augmentation to achieve holistic and comprehensive cost optimization.
- Automation of financial governance controls in cloud environments provides increased predictability and cost optimization with less operational effort.
- Solid financial governance frameworks are positioning organizations to take advantage of FinOps.
- Emergence of specific roles — like FinOps practitioner or cloud economist — focused on FinOps practices and cost optimization means organizations have the expertise to address augmented FinOps.
- Owing to their complexity, cloud environments are ideally suited for the application of ML and AI methods to automate processes and track price and performance.
- Core FinOps capabilities are being delivered in three ways: Homegrown solutions, cloud service provider (CSP) instrumentation and third-party vendors. Increasingly practitioners are seeking out third-party or CSP tools to address their needs. All of these have a broad objective of adopting augmented capabilities as a means of competitive differentiation.

Obstacles

- Cloud service provider pricing models remain needlessly complex and diverse.
- Cloud ecosystems are (and will remain) open to third-party participants, which implies multiple commercial arrangements with multiple providers.
- Standards for cloud cost, usage and billing data like the FinOps Foundation's FOCUS proposal have yet to be broadly adopted. APIs for communicating performance data within the context of a broader ecosystem have yet to emerge. Both of these are required to assess the primary measure of success: price/performance.

User Recommendations

- Seek out service offerings to automate (via AI/ML) performance, consumption and pricing options. Increasingly, incorporate these capabilities into cloud data ecosystems that will learn from consumption patterns as they seek to optimize the underlying resources, and by extension, cloud spending through orchestration and optimization.
- Apply Gartner's FinOps Maturity Model to assess FinOps offerings in terms of their ability to address the following core capabilities: Observe, report, recommend, predict and optimize. The last three introduce augmented FinOps capabilities.
- Plan to use multiple tools to address the full scope of requirements. Many tools are broad in reach, but do not go deep into prescriptive recommendations. Others are tightly scoped and provide very targeted optimizations. Expect to spend time combining multiple tools to achieve broad and deep capabilities.

Sample Vendors

Acceldata; Anodot; Apptio; Capital One Software; Densify; Enteros; Finout; OtterTune; Sync Computing; Unravel Data

Gartner Recommended Reading

How to Identify Solutions for Managing Costs in Public Cloud IaaS

A Guidance Framework for Selecting Cloud Management Tools

Emerging Tech: Data Management Product Leaders Must Implement Augmented FinOps in Their Cloud Solutions

CDAOs and CFOs Must Drive Business Value in the Cloud Through Collaboration

Financial Governance Is Essential to Successful Cloud Data and Analytics

Monitoring as Code

Analysis By: Hassan Ennaciri, Pankaj Prasad

Benefit Rating: High

Market Penetration: 20% to 50% of target audience

Maturity: Emerging

Definition:

Monitoring as code (MaC) is the process of applying software principles to monitoring, meaning the configuration of monitoring is designed to enable its management, like software. With MaC, the configuration of monitoring is codified, version-controlled, tested and automated. This flexibility offers DevOps teams the option to apply a shift-left approach for fast and consistent monitoring across systems.

Why This Is Important

Traditional monitoring approaches require manual and, in some cases, inflexible configurations that hinder DevOps teams' ability to support continuous delivery with frequent deployments. Manually changing monitoring introduces risk of misconfigurations and inconsistencies that can extend the mean time to detect failures, causing a larger impact on customers.

Business Impact

By leveraging MaC, teams can collaborate, share, optimize and speed up monitoring configurations to ensure a consistent, efficient approach to monitoring that can support high-velocity changes, while reducing the risk of missed alerts. MaC enables DevOps teams to automate monitoring configuration by leveraging the continuous integration/continuous delivery (CI/CD) toolchains or DevOps platforms they use to deliver features or infrastructure.

Drivers

- **Customers' Expectations for Reliability and Performance:** This makes monitoring very important, as it ensures reduced impact on customers during unexpected events. Monitoring as code provides a consistent way to manage monitoring configuration at scale.
- **Agile, DevOps and GitOps:** These methodologies and new ways of working drive the need for automation to support the velocity and high frequency of changes. MaC meets these needs because it leverages the modern architecture's attribute of instrumentation, while adding agility and efficiency to monitoring configuration.
- **Cloud-Native Architectures:** Modern cloud-native architecture, microservices and distributed nature drive the need for complex monitoring configurations.
- **Infrastructure Platform Engineering:** Delivering infrastructure services via infrastructure-as-code pipelines drives the need to automate monitoring configurations and take advantage of CI/CD processes to change, test and deploy the applications.

Obstacles

- **Resistance to Change:** Traditional monitoring is done by the IT operations team and is manual. This new modern approach is a paradigm shift that requires collaboration and automation.
- **Security and Compliance:** In-depth monitoring requires privileged and granular role-based access. MaC is relatively new and will require security teams to reassess governance and apply more stringent evaluations to ensure the organization will not be in breach of compliance requirements.
- **Skills and Expertise:** MaC is an advanced practice that relies on software engineering skills, which many I&O teams lack. IT organizations will require new skilled resources or training.
- **Lack of Support With Traditional Monitoring Tools:** This will require investment in new tools that support MaC.

User Recommendations

- **Collaborations:** Monitoring complex systems with modern architecture is a team activity. Teams need to continuously collaborate to ensure the relevant features and metrics are monitored.
- **Tools:** Invest in tools that support infrastructure as code.
- **Configurations via DevOps:** Treat the MaC as software, taking advantage of version control, testing and automated deployments. Teams should leverage the same DevOps toolchains to manage the monitoring code.
- **Continuous Improvements**: Continuously seek operational feedback, and use qualitative and business KPIs to improve the quality of monitoring configurations, as well as the process.

Sample Vendors

AppDynamics; Checkly; Datadog; Dynatrace; Grafana; New Relic; Splunk; Zoho

Service Operations

Analysis By: Mark Cleary

Benefit Rating: Transformational

Market Penetration: 5% to 20% of target audience

Maturity: Emerging

Definition:

Service operations is the convergence of the infrastructure and application monitoring environments with the ITSM incident management practice to create a more effective and optimized mechanism for diagnosing and resolving incidents. The combination of the two environments with the context of AI can lead to a significant reduction in both the number and impact of incidents.

Why This Is Important

The ability to quickly diagnose and resolve incidents is critical for organizations as they develop and implement digital services. Intelligent automation can remove monitoring and service desk silos, creating the concept of service operations. This ensures that incident detection through monitoring can be better integrated with ITSM platforms, leading to incidents that can be more readily identified and resolved creating higher availability and more resilient services.

Business Impact

The convergence of ITSM and monitoring environments results in a more productive and seamless incident management practice. The contextual analysis of alerts resulting in anomaly detection and escalation can result in fewer incidents and improved mean time to resolve (MTTR), especially when integrated with the ITSM platform. Products and services are more available and less likely to suffer from poor performance or outages and disruption, costs are reduced, and overall productivity is improved.

Drivers

- Business pressure for reliable and stable services drives the demand for incidents to be either eliminated before they can cause disruption or be diagnosed and resolved as quickly as possible.
- The closer integration of monitoring with the service desk creates a far more effective and seamless environment in which outages and disruption can be better and more effectively managed.
- ITSM platforms now feature observability and correlation functions to identify incidents as part of the monitoring capabilities now available.
- ITSM platforms can now enrich the events by applying the context of recent incidents and changes to the impacted configuration items (CIs) to radically improve the identification of the probable root cause.
- The rise in the use of AI allows anomalies and events from different monitoring systems to be swiftly correlated and analyzed and presented to technicians via the ITSM platform.

Obstacles

- AI-based monitoring systems can take longer than expected to understand the context of the services, the traffic, and the environment creating delays in improving availability.
- Improved diagnostics and resolution times require time and effort to achieve due to dependencies including effective service mapping, the degree of automation and the quality of monitoring. This requires a concerted effort to achieve a holistic approach.
- Aligning data feeds, telemetry and log information from different sources creates data disparity leading to delays in identifying and correlating incidents.
- Accurate service mapping and comprehensive, up-to-date configuration management databases are necessary to support end-to-end incident processes if enhanced availability and resilience is to be achieved.
- Automated responses need to be carefully chaperoned to ensure that their decisions match the use case.
- Many monitoring and service desk teams operate in silos, making it difficult to create a seamless experience.

User Recommendations

- Develop a strategy for service operations that focuses on effective incident management. Consider the implications on the monitoring and technical support staff (including the service desk), the range of tools available, how they will be integrated, and whether the service practices need to be amended.
- Identify the tool and process dependencies and build a roadmap to define and implement the solution.
- Develop or purchase runbooks as scripted solutions for common incident scenarios.
- Explore vendor options for solutions offering automated responses and self healing to ensure alignment and integration with your environment.
- Adopt a pragmatic approach and pilot the initiative on a critical business service to understand the capabilities and drawbacks.
- Use collaboration techniques such as communities of practice to bring the various teams together to understand the respective roles and responsibilities, and challenge them to reduce the cycle time to identify and process an incident.

Sample Vendors

Atlassian; BigPanda; BMC; DataDog; Moogsoft; PagerDuty; ServiceNow

Gartner Recommended Reading

Market Guide for AIOps Platforms

Observability-Driven Development

Analysis By: Manjunath Bhat

Benefit Rating: High

Market Penetration: 5% to 20% of target audience

Maturity: Emerging

Definition:

Observability-driven development (ODD) is a software engineering practice that provides fine-grained visibility and context into system state and behavior by designing observable systems. ODD works by instrumenting code to unravel a system's internal state with externally observable data. As part of a shift-left approach to software development, ODD makes it easier to detect, diagnose and resolve unexpected anomalies early in the development life cycle.

Why This Is Important

Building observable systems can expedite issue resolution as observability data is a useful debugging aid. Designing for observability also amplifies the benefits of other resilience engineering practices, such as site reliability engineering (SRE) and chaos engineering. ODD enables software engineers and product owners to understand how the software is performing and how it is being used. However, the practices to institutionalize ODD are still emerging.

Business Impact

Built-in observability helps develop reliable software. ODD reduces the time to troubleshoot issues in production and preproduction environments. This feature helps software engineering teams reduce cycle time, making developers more productive during testing. ODD also helps ship code confidently since observable data makes it easier to troubleshoot production issues, thus minimizing downtime. In business terms, it translates to compliance with regulatory and contractual SLAs.

Drivers

- Organizations adopting SRE and chaos engineering practices need the data provided by observability design. These practices are fundamentally data-driven and support software reliability.
- User experience (UX) is subjective and difficult to measure. It depends on various factors that span the complete technology stack. Some factors, such as last-mile network connectivity, affect UX but are difficult to optimize unless systems are designed to be observable and extract data related to such signals, like response rates and latency.
- Mobile and edge environments present unforeseeable challenges since applications can run on potentially unknown devices and untrusted environments. These production environments can significantly differ from the local environments used to test the application. Therefore, ODD can help capture and investigate the “unknown unknowns” at runtime.
- Distributed systems exhibit emergent behavior and are difficult to predict. Therefore, the need for observability increases as issues can arise due to unexpected component interactions. Troubleshooting issues in distributed applications requires fundamentally different techniques than monolithic or client and server applications. Building observability narrows down the problem domain and helps engineers inspect the problematic component.
- OpenTelemetry is an open standard that has seen increased open-source implementations and vendor adoption over the past year. This has improved consistency when adopting ODD across products. Organizations also favor using open standards and protocols to ingest telemetry data. This makes ODD accessible to most developers even when they lack the budget for commercial tools.

Obstacles

- Monitoring and, by association, observability, is commonly viewed as an operational responsibility. Software engineers often lack expertise with observability as a practice and with the tools and frameworks used to implement observability.
- Software engineering leaders must overcome the perception that observability can be achieved merely by implementing an observability tool. Observability is a domain that must be designed and built into systems to ensure that it provides business benefits.
- A piecemeal approach to observability may thwart efforts to adopt ODD at scale. As a standard practice, ODD provides greater benefits when all system components are designed with an observability mindset. For example, distributed tracing requires that all components contributing to the trace are “instrumented” and propagate context for diagnosing response-time issues. Platform teams entrusted with driving ODD at scale can help overcome this obstacle.

User Recommendations

- Adopt ODD as a standard software engineering practice to handle unexpected and unforeseeable system behaviors and anomalies.
- Make observability a priority since it provides critical insights to resolve production errors. It provides continual feedback to understand how the software is being used.
- Keep up with the pace of innovation in observability by using open standards and open-source technologies, such as OpenTelemetry.
- Be wary of vendor hype regarding observability merely to provide access to logs, metrics and traces. Use ODD as a fundamental software engineering practice that improves UX and resilience with granular insight into system state and behavior.

Sample Vendors

Chronosphere; Datadog; Dynatrace; Grafana; Honeycomb; Logz.io; New Relic; observIQ; ServiceNow; Splunk

Gartner Recommended Reading

How to Identify and Resolve Application Performance Problems

Improve Software Quality by Building Digital Immunity

Telemetry Pipelines

Analysis By: Gregg Siegfried

Benefit Rating: Moderate

Market Penetration: 1% to 5% of target audience

Maturity: Emerging

Definition:

Telemetry pipelines are solutions that provide a uniform and holistic mechanism to manage the collection, ingestion, enrichment, transformation and routing of machine data (telemetry) from source to destination. These solutions can be consumed on a self-managed, SaaS-managed or hybrid basis. Telemetry pipelines may be stand-alone products or part of a vendor's broader portfolio of monitoring solutions.

Why This Is Important

When applications and services are distributed across a wide area, and involve multiple service providers, their context is as well. Telemetry pipelines, sometimes called observability pipelines, enable organizations to collect, transform, enrich and route health, performance and security telemetry more efficiently from sources (workloads, monitoring agents, platforms) to destinations (analysis and investigation tools, event management solutions, and long-term storage).

Business Impact

Telemetry pipelines improve efficiency by:

- Normalizing taxonomy across disparate sources, improving correlation.
- Ensuring telemetry of sufficient quality before analysis.
- Managing analysis cost by storing telemetry according to its purpose.
- Reducing the number of agents collecting telemetry at the source.
- Simplifying alert generation by managing granularity and cardinality before ingestion into analysis tools.

- Optimizing bandwidth utilization through compression and deduplication.

Drivers

- **Telemetry volume is increasing:** Modern workloads generate significant amounts of telemetry which takes many forms and may originate in different locations. Telemetry pipelines provide a mechanism to unify them.
- **Cost:** Moving and storing data can be expensive. Many telemetry insight platforms charge based on ingest volume. Being more intentional about telemetry and only moving, ingesting and storing what you need can help manage costs.
- **Bulk long-term storage:** Cloud-based object storage has become a ubiquitous, secure and reasonably priced way to store "colder" data. Some log monitoring products have built seamless support for object storage and maintain rapid reporting access as well.
- **Most telemetry pipeline products support OpenTelemetry:** Transforming traditional forms of telemetry into OpenTelemetry for ingestion and analysis by a modern tool is one way to reap some of the benefits of OpenTelemetry without substantial changes to code or configuration.

Obstacles

- **Additional tools add to administrative cognitive load:** Telemetry pipeline products have their own learning curve, particularly when configuring them to transform and enrich data as they pass through.
- **Potential compatibility:** Given the variety of telemetry sources and analysis back ends available, choosing a telemetry pipeline product that meets current and future needs may pose a challenge.

User Recommendations

- Consolidate or bridge silos of telemetry by deploying telemetry pipelines. Understanding the data that you have will support use-case adjacencies such as a telemetry data lake.
- Unify operational and security-related telemetry collection by deploying telemetry pipeline products. The analysis back ends may be different, but a reduction in agents can be a win.
- Emphasize consistency by limiting the output formats. Although telemetry pipelines are inherently many-to-many solutions, make “many to fewer” the objective.
- Use telemetry pipelines to optimize and standardize the data enrichment and transformation needs without overloading the data source.

Sample Vendors

Calyptia; Cribl; Datadog; Edge Delta; Mezmo; observIQ

Gartner Recommended Reading

Predicts 2023: Observing and Optimizing the Adaptive Organization

Innovation Insight for Telemetry Pipelines

Enhance Monitoring and Observability by Using Telemetry Pipelines

Data Observability

Analysis By: Melody Chien, Ankush Jain

Benefit Rating: High

Market Penetration: 1% to 5% of target audience

Maturity: Emerging

Definition:

Data observability is a technology that supports an organizations' ability to understand the health of an organization's data, data pipelines, data landscape, and data infrastructure by continuously monitoring, tracking, alerting and troubleshooting issues to reduce and prevent data errors or system downtime. It tells us what went wrong based on agreed upon SLAs for data quality and usage; reasons; assesses the impacts; and recommends solutions. Data observability improves reliability of data by increasing our ability to observe changes, discover unknowns and take appropriate actions.

Why This Is Important

Data observability uses data profiling, AI/ML, lineage and active metadata to provide the following benefits:

- **Monitor & Detect:** Provide a holistic view to determine how components of data pipelines are operating, evaluate whether data quality meets expectations, and detect data related issues.
- **Alert & Troubleshoot:** Send right alerts to the right people at the right time and perform root cause analysis.
- **Resolve & Prevent:** Provide recommendations to fix the issues or optimize data pipelines to meet business requirements with the goal to prevent downtime or critical data issues before affecting business.

Business Impact

- Data observability allows technical teams to gain visibility of the health of data pipelines and infrastructure. They can identify possible drifts in various areas, and minimize the time to investigate and solve issues, preventing unplanned outages or critical data errors.
- Business users will also gain visibility of data quality and associated financial impacts. This will ensure appropriate use and management of data to meet governance requirements.
- Data observability allows facilitation and improvement of the data fabric with continuous observations and evaluations of the data and analytics ecosystem.

Drivers

- Data and analytics leaders face a growing number of mixed data stacks, diversity of datasets, unexpected data drifts such as change in schema or business context, high demand for data quality and near zero tolerance of downtime. All these add to the challenges in data management. They need a holistic view of the state of data quality and data pipelines within interconnected systems.
- Data pipelines move data from point to point and deliver data to consumers. This journey can be disrupted by unexpected events such as data quality issues or a lack of infrastructure resources. The data that flows through these pipelines needs to be monitored for loss of quality, performance or efficiency. Organizations need to be able to identify points of failure before they have a chance to propagate. Data observability automatically detects important events and analyzes various signals to troubleshoot the issues, and provides actionable insights of what to do next.
- Data observability goes beyond traditional monitoring. It provides a multidimensional view of data including performance, quality, usage and financial impacts to the downstream applications. Leveraging active metadata, lineage of data and AI/ML, data observability generates real-time insight by monitoring the business context and analyzing data pattern, comparing history, and developing a semantic understanding of the data. It provides an end-to-end observability to help organizations be better equipped to handle critical events and prevent business disruptions.
- This capability is essential to the data fabric design concept and becomes an important building block to further automation in data management practices.

Obstacles

- There is no standard definition of what constitutes a data observability solution. Vendors offer a range of different capabilities often branded as data observability which is causing confusion in the market and leading to issues adopting the tools.
- The current vendor landscapes are very fragmented based on coverage areas and data environments supported. Most vendors focus on observability of the data quality and data pipelines, and are less concerned about data usages and financial impacts. The full end-to-end observations are not quite there yet from individual vendors.
- Most data observability tools only support the modern data stack. This limits their application in large enterprise environments with more complex data environments in many cases using legacy data management tools.
- Most data observability tools target the data engineer persona and are positioned as IT tools. Though business users receive important insights from data observability tools, they may find them less user-friendly.
- Organizations are embracing the concept of “observability.” But the actual adoption of the tools is not straightforward. The consideration of how they connect to the overall ecosystem and connecting this to data governance strategy is still a concern.

User Recommendations

- Identify the data elements or data pipelines which require high standards or SLA in quality, uptime, latency and performance. Pinpoint the gap of current monitoring capabilities vs. desired capabilities to support the requirements.
- Evaluate data observability tools available in the market that can enhance your observability based on priority of business requirements, primary users and interoperability with the enterprise data ecosystems.
- Pilot data observability program by building a monitoring mechanism as a starting point to increase visibility over the health of data. Invest in observability capabilities in a cloud environment first, as it's commonly supported by vendors and is faster and easier to demonstrate value.
- Include both business and IT perspectives when evaluating data observability tools by engaging with both personas early on in the evaluation process.
- Partner with business stakeholders to evaluate and demonstrate business value of data observation practices by tracking improvement of data quality, reduction in downtime and ability to meet SLAs to show tangible benefits.

Sample Vendors

Acceldata; Ataccama; Bigeye; Collibra; IBM; Kensu; Monte Carlo; Soda; Unravel

Gartner Recommended Reading

Data and Analytics Essentials: Data Observability

Quick Answer: What Is Data Observability?

The State of Data Quality Solutions: Augment, Automate and Simplify

Market Guide for DataOps Tools

Transfer Learning

Analysis By: Ben Yan, Shubhangi Vashisth, Radu Miclaus, Wilco van Ginkel

Benefit Rating: High

Market Penetration: 1% to 5% of target audience

Maturity: Adolescent

Definition:

Transfer learning reuses previously trained machine learning models as an advanced starting point for new purposes, in order to reduce the learning time and data required to attain acceptable performance.

Why This Is Important

Transfer learning is attractive. It enables rapid training, reduces the amount of data needed and may provide better predictive performance than models trained from scratch. A starting point is a repository of models, and these models can be customized based on internal or external data. Transfer learning advances the broader field of AI, as it allows AI to generalize — i.e., use what is learned in one task to more quickly learn another, related task. Transfer learning can also be used to further refine existing models with smaller amounts of data.

Business Impact

The business impact of transfer learning can be summarized as follows:

- It will impact how organizations apply machine learning (ML), especially for natural language processing (NLP) scenarios.
- It promises to attain acceptable performance with less training data, significantly less computational overhead (green IT) and faster development speed.
- It utilizes the model from the source (data-rich) domain, opening ML use cases that were previously infeasible due to lack of data.

Drivers

- **Broader application in foundation models:** The popularity of ChatGPT and image generation models is driving attention to the fine-tuning (a form of transfer learning) of foundation models. Compared with prompt engineering, fine-tuning techniques can incorporate more data into the models, and build customized models for organizations.
- **Increase in model marketplaces and community:** The availability of repositories, such as Hugging Face, allows developers to find and reuse pretrained models with easy community collaboration. The open-source repositories also enable organizations to build models with fewer barriers to entry.
- **Applicability to multiple use cases and verticals:** We see transfer learning being used across a number of model types (language, computer vision, predictive and multimodal) in many domains, such as finance, healthcare, gaming, autonomous driving and e-commerce.
- **Proliferation of AI models within organizations:** Many AI models can be reused. As more models are created, opportunities for transfer learning increase. These models and their datasets can be reused between departments, or even in external organizations.

Obstacles

- The adoption of transfer learning highly depends on the availability of existing models and the relevance/similarity between domains. It is hard to determine upfront whether transfer learning works.
- Transfer learning today is a capability embedded into existing platforms or a method applied by systems integrators and analytics consultancies.
- Transfer learning remains a technical challenge. Fine-tuning foundation models is even harder, and requires proper data, computing resources and talents. The ROI of fine-tuning customized models needs to be measured case by case.
- As the AI field becomes more regulated, documentation of source data and model lineage may be required to support explainability and trust. Not all model providers could provide sufficient information.
- In the quest for more explainable, fair and transparent AI, transfer learning can be seen as a further complication to the AI development process.

User Recommendations

- **Maintain repositories of AI models and datasets:** Work with your data and analytics leaders to utilize metadata management initiatives to identify AI models and their datasets. Document the successful transfer learning examples. AI centers of excellence (COEs) or similar should facilitate.
- **Explore useful internal and external models:** Seek transfer learning opportunities to reuse AI models. Organizations with a more mature level of AI adoption should additionally assess how their current models might be reused in related domains and/or similar tasks.
- **Check the ML tools you use to create and train models, and determine their support for transfer learning:** ML tools should include capabilities that facilitate transfer learning, such as fine tuning.
- **Loop in CSOs, legal teams and business stakeholders:** Teach them about transfer learning to develop your initial position on AI risk. For example, educate them on the lineage of the original data for the base model and the security risks of open base models you may use.

Sample Vendors

4Paradigm; Alibaba Group; Amazon; Google; H2O.ai; Hugging Face; IBM; Microsoft; NVIDIA; OpenAI

Gartner Recommended Reading

Innovation Insight: Transfer Learning

Transfer Learning in China: Increase the Value of Your Data Every Time You Use It

Three Steps to Boost Data for AI

AI Design Patterns with ChatGPT

Digital Platform Conductor Tools

Analysis By: Roger Williams, Dennis Smith

Benefit Rating: Transformational

Market Penetration: Less than 1% of target audience

Maturity: Embryonic

Definition:

Digital platform conductor (DPC) tools coordinate the various infrastructure tools used to plan, implement, operate and monitor underpinning technology and services for applications and digital products. They enable digital business, regardless of the environments used or who owns them. DPC tools provide a unified view of underpinning technologies and their connection to applications. This augments strategic decision making and improves the value obtained from technology investments.

Why This Is Important

Traditional, cloud and hybrid infrastructure management tools do not inherently provide an integrated view of infrastructure across all environments. Moreover, as infrastructure and operations (I&O) leaders struggle to manage their portfolio of investments to enable composable business, optimize costs and reduce risks, they need help with filling the gaps in visibility, assurance and coordination. DPC tools promise to help close these capability gaps and are improving in their ability to do so.

Business Impact

DPC tools deliver the following benefits not inherent in more focused infrastructure management toolsets:

- Visualizing digital platform performance across all life cycle stages — planning, implementing, operating and monitoring.
- Enabling continual optimal performance and placement of workloads in all environments — on-premises, in the cloud or at the edge.
- Ensuring tangible business value from improvement efforts across all technology architectures — compute, storage, middleware and network layers.

Drivers

- Difficulty in maintaining a coherent view of all technology infrastructure resources and their dependencies that are aligned with changes to services, applications and components, as well as the configuration of their promised performance levels.
- Lack of transparency into spending on hybrid digital infrastructure and how resource capacity aligns with actual application workload demand.
- Need to guide where workloads are processed (data center, public cloud, colocation facility, etc.) based on requirements, including capacity, cost and dependency dynamics.
- Challenges with estimating the value, efficiency, quality and compliance delivered by hybrid digital infrastructure based on aggregated data from performance analysis tools and other hybrid digital infrastructure management (HDIM) toolset data feeds.
- Desire for a single point of entry and reporting for digital platform resource requests, and routing them to appropriate HDIM tooling for fulfillment.
- Desire to reduce the level of skills and effort required within initiatives to improve operations and digital employee experiences.
- Gaps, duplication and conflicts in data to support application workload migration and business continuity goals, as well as protection of data from accidental deletion or malicious activities.
- Inability to confirm compliance of application workloads and digital platforms to identity requirements and security baselines as part of the organization's cybersecurity mesh approach.
- Poor credibility of business cases for digital platform improvements, including: assessing business impact; measuring gaps between current and desired performance; providing oversight of improvement efforts; and validating benefits delivered.

Obstacles

- Lack of interoperability: Tool sprawl and difficulties in integration inhibit DPC tool adoption. The technology landscape is littered with failed approaches that were intended to support data sharing between vendors.
- Lack of data credibility: The desire for a complete, accurate view of all technology as a precondition for decision making has been around for decades, yet is no closer to being realized. Customers that demand perfect data before they act, and vendors that require complete and accurate data for their tools to function properly, will continue to co-create expectations that will not be met.
- Lack of budget: DPC tools may be viewed as “overhead” that does not have a compelling business case. No one likes paying for something that does not appear to address specific pain points felt today.
- Lack of vendor commitment: Many vendors will be tempted to “DPC wash” their existing offerings and claim that these capabilities are already addressed or can be added for very little cost.

User Recommendations

- Build a DPC tooling strategy that supports digital business ambitions by defining the management elements, environments and technology layers required to meet the organization’s infrastructure needs now and in the future.
- Address measurement and coordination gaps by working with key stakeholders to identify infrastructure value and risk and cost objectives, and by making targeted investments in integration, dependency mapping and continuous improvement capabilities.
- Plan for DPC tooling investments by determining which DPC capability aspects are needed in the short, medium and long term. Compare these capabilities to current and future vendor offerings for infrastructure management tooling that can provide initial DPC tool functionality.
- Ensure that DPC tooling investments can deliver sustained value by requiring that DPC tool marketers show how the tool will address current organizational pain points and how it will adapt to future needs as organizational requirements evolve.

Sample Vendors

Cloudsoft; Flexera; HCLTech; IBM (Turbonomic); Oomnitza; OpsRamp; ReadyWorks; Snow Software; Virtana

Gartner Recommended Reading

Market Guide for Digital Platform Conductor Tools

3 Steps to Improve the Reliability of Large, Complex and Distributed IT Systems by Leveraging SRE Principles

Internal Developer Portal

Analysis By: Manjunath Bhat

Benefit Rating: High

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Definition:

Internal developer portals serve as the interface that enables self-service discovery and access to resources in complex, cloud-native software development environments. They can include software catalogs, scorecards to benchmark software quality, scaffold templates, product documentation, plug-ins for extensibility and automation workflows. Developer portals help improve developer productivity, operational efficiency and enhance governance by providing shared visibility across multiple teams.

Why This Is Important

Internal developer portals help software developers navigate infrastructure complexity, understand service interdependencies and enable faster release cadence in at least three ways. First, they serve as a common viewpoint for multiple teams of developers. Second, they provide developers with self-service access to underlying platform components and environments. Third, they provide a centralized place to score applications and measure progress against reliability and security requirements.

Business Impact

Developer portals can have the following business impacts:

- Developer experience and productivity: Help development teams improve their delivery cadence by improving developer experience, reducing cognitive load and shortening the onboarding time.
- Reliability and resilience: Aim to provide visibility to application health and include scorecards to assess their production readiness.
- Security and governance: Include prebuilt toolkits, templates and curated libraries that help create “paved roads” with built-in compliance, security and audit policies.

Drivers

- Platform engineering: Organizations are adopting platform engineering principles and creating platform teams to scale cross-cutting capabilities across multiple development teams. Platform teams curate internal developer platforms to abstract away the complexity of siloed systems and processes. Internal developer portals serve as an interface through which developers can consume the capabilities of internal developer platforms.
- Backstage: Backstage is one of the first open-source frameworks for building developer portals. It was created at Spotify and is now a Cloud Native Computing Foundation (CNCF) incubating project. The thriving open-source community supporting Backstage has largely contributed to its enormous mind share and rapid adoption. Hundreds of organizations have adopted Backstage since it was open-sourced in 2020. Backstage’s success continues to drive interest, momentum and competition in this space.
- Developer experience: With software at the core of all digital innovation today, a great developer experience that accelerates software development becomes a key competitive advantage. Therefore, software engineering leaders are increasingly focused on minimizing developer friction and frustration. The ability to curate and provide customizable, developer-friendly experiences within the developer portal and reign in complexity will drive their appeal for both product and platform teams.

- Innersource: To enable rapid innovation and facilitate greater collaboration and knowledge sharing, software engineering leaders are adopting innersource approaches to software development. However, innersource requires an easy way for other teams to discover and search for existing projects within their organization. This is why organizations adopting innersource are turning to internal developer portals to make projects available and discoverable by other teams. See InnerSource Portal.

Obstacles

- Prerequisites: The successful adoption of internal developer portals goes beyond deploying a tool and requires certain prerequisites to be in place. For example, application services and their dependencies must be organized with consistently defined metadata that helps track their usage, performance and team ownership.
- Absence of platform teams: A dedicated platform team to manage and evolve the portal as a product is necessary to ensure the portal meets desired objectives. The absence of a dedicated platform team, and more so, led by a platform product owner to manage the portal as a product results in a disconnect between developer expectations and the portal's capabilities.
- Lack of developer buy-in: Although the developer portal serves as the "window" to the underlying platform's capabilities, it should provide "paved roads" and not "forced marches" — portal use should remain the choice of the development team. Trying to force development workflows into organizationwide blueprints for building developer portals without involving developers is a recipe for failure.

User Recommendations

- Use internal developer portals to scale cross-cutting software engineering capabilities across multiple development teams and streamline the software delivery life cycle.
- Do not assume that internal developer portals are turnkey solutions — they require a lot of prerequisites to be in place and many cases involve several weeks of prework activities. For example, Backstage requires codification of service-related metadata in YAML files before the content shows up in the software catalog.
- Ensure that the platform team includes internal developer portals in their charter. Continuously innovate portal capabilities by appointing a platform product owner for the developer portal to manage its roadmap, gather feedback and market its capabilities.

Sample Vendors

Atlassian; Calibo; CodeNOW; configure8; Cortex; Mia-Platform; OpsLevel; Port; Roadie

Gartner Recommended Reading

Innovation Insight for Internal Developer Portals

Drive Innovation by Enabling Innersource

Adopt Platform Engineering to Improve the Developer Experience

Cool Vendors in Platform Engineering for Improving Developer Experience

OpenTelemetry

Analysis By: Gregg Siegfried

Benefit Rating: High

Market Penetration: 5% to 20% of target audience

Maturity: Early mainstream

Definition:

OpenTelemetry (OTel) is a collection of specifications, tools, APIs and SDKs that describe and support the implementation of an open-source instrumentation and observability framework for software. The initiative, curated by the Cloud Native Computing Foundation (CNCF), defines multiple flavors of telemetry — traces, metrics, logs and, soon, performance profiles. OTel is widely supported by commercial and open source monitoring solutions.

Why This Is Important

OpenTelemetry introduces a portable way to instrument, generate, collect and export telemetry data about application health and performance. It has changed the way application performance monitoring (APM) solutions are assessed, deployed and used. OpenTelemetry has garnered widespread support and adoption. It is currently second in velocity only to Kubernetes within the CNCF ecosystem. Many software vendors, cloud providers and observability tools have released or announced support for OTel.

Business Impact

OpenTelemetry promises to:

- Enable deeper visibility into application health and performance, even across application and service provider boundaries.
- Facilitate the use of multiple vendors or the migration between vendors.
- Benefit product owners, SREs and platform operators by allowing them to "instrument once, analyze anywhere."
- Standardize encoding, transport, and delivery of telemetry between sources and targets, thus improving reliability and scalability as more vendors become OpenTelemetry Protocol (OTLP) compliant.

Drivers

- **Uniformity of instrumentation:** Traces provide a rich, sequenced perspective to request handling in distributed software, but are not always enough to fully identify and resolve anomalies. By including support for correlating metrics and logs with traces, OpenTelemetry incorporates a more complete dataset for application observability use cases.
- **Software architecture:** Microservices, containers and functions are powerful constructs that serve as the basis for modern applications. Loose coupling facilitates the build, test and release of independent components.
- **Cost of APM solutions:** Many organizations cannot afford to monitor all of their applications with commercial APM offerings and either reduce the monitoring footprint or leverage a low-cost, secondary APM tool. OpenTelemetry facilitates the latter by allowing the same telemetry to be flexibly routed to different solutions.
- **Site reliability engineering:** In many organizations, Site Reliability Engineers (SREs) are responsible for health and performance management and are the most likely to demand insights from OpenTelemetry. Interest in building an SRE role is on the rise.

Obstacles

- **Maturity:** OpenTelemetry is evolving rapidly, and is beginning to be used widely in the field. It is still most suitable for early adopters. There may be a cost in time and effort associated with adopting it now that will diminish over time.
- **Implementation variations:** The need to ship support for an emerging set of specifications has led some vendors to make assumptions about work in progress, which risks multivendor compatibility. In April 2023, Elastic donated their Elastic Common Schema to the OpenTelemetry project, which may help reduce these instances.
- **Roadmap:** While the tracing and metrics specifications are completely stable, OpenTelemetry logging is partially "experimental" as of this writing. These specifications often stabilize earlier, but full implementation of all flavors will be spotty through development. An initiative to add continuous performance profiling as a fourth type of telemetry is in the very early stages.

User Recommendations

- Prefer vendors that are committed to OpenTelemetry when selecting monitoring solutions.
- Embrace OpenTelemetry for distributed tracing today when building trace instrumentation into your custom application software.
- Instrument your cloud-native applications using OpenTelemetry SDKs as they are available for your languages and frameworks.
- Augment existing APM solutions with OpenTelemetry for hybrid workloads.

Sample Vendors

Amazon Web Services (AWS); Cisco Systems; Dynatrace; Elastic; Honeycomb; Lightstep; New Relic; Splunk

Gartner Recommended Reading

Assessing OpenTelemetry's Impact on Application Performance Monitoring

Monitoring and Observability for Modern Infrastructure and Applications

Solution Path for Modern Infrastructure and Application Monitoring

At the Peak

Chaos Engineering

Analysis By: Jim Scheibmeir, Hassan Ennaciri

Benefit Rating: Moderate

Market Penetration: 20% to 50% of target audience

Maturity: Early mainstream

Definition:

Chaos engineering is the use of experimental and potentially destructive failure testing or fault injection to uncover vulnerabilities and weaknesses within a distributed system. Chaos engineering tools provide the ability to systematically plan, document, execute and analyze an attack on components and whole systems throughout the life cycle of the system. This planning may include the injection of random timing or attack executions.

Why This Is Important

Many organizations rely on test plans that overemphasize functionality and underemphasize validating the system's reliability and resilience. The distribution and complexity of systems makes understanding them more difficult. Chaos engineering (CE) shifts the focus of testing a system from the "happy path" toward testing how it can degrade gracefully or continue to be useful and secure while under various levels of impact. Applying CE enables improvements to system knowledge and documentation.

Business Impact

CE is aimed to minimize time to recovery and the change failure rate, while maximizing uptime and availability. Addressing these elements helps improve customer experience, satisfaction, retention and acquisition. Gartner inquiries regarding CE increased by over 11% between 2021 and 2022.

Drivers

- Increased complexity of systems and increasing customer expectations are the two largest drivers of CE and the associated tools.
- As systems become more rich in features, they also become more complex in their composition and more critical to digital business success.

- Overall, CE helps organizations become more resilient across their processes, knowledge and technology.
- Teams often lack the confidence to handle failures and the psychological safety to take action to resolve incidents. CE can help build that confidence.

Obstacles

- Within many organizations, the predominant view of CE is that the practice is random, first implemented during production, and increases, rather than reducing, risk.
- Organizational culture and attitudes toward quality and testing can present barriers to the adoption of CE. When quality and testing are only viewed as overheads, there will be a focus on feature development over application reliability.
- It can be challenging just to secure the time and budget to invest in learning CE and associated technologies. Organizations must reach minimum levels of expertise so that value is returned.

User Recommendations

- Utilize a test-environment-first approach by practicing CE in preproduction environments.
- Incorporate CE into your system development, CI/CD or testing processes.
- Build out incident response protocols and procedures, as well as monitoring, alerting and observability capabilities, in tandem with the advancement of the CE practice.
- Utilize scenario-based tests — known as “game days” — to evaluate and learn about how individual IT systems would respond to certain types of outages or events.
- Investigate opportunities to use CE in production to facilitate learning and improvement at scale as the practice matures. However, Gartner believes that very few organizations purposely use CE in their production environments.
- Formalize the practice by adopting a platform or tool to track the activities and create metrics to build feedback for continuous improvements.

Sample Vendors

Amazon Web Services; ChaosIQ; Gremlin; Harness; Microsoft; Steadybit; Verica

Gartner Recommended Reading

Quick Answer: What Metrics Should We Use to Assess and Improve Software Quality?

Predicts 2023: Observing and Optimizing the Adaptive Organization

Top Strategic Technology Trends for 2023: Digital Immune System

Predicts 2023: How Innovation Will Transform the Software Engineering Life Cycle

eBPF

Analysis By: Simon Richard

Benefit Rating: Moderate

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Definition:

Extended Berkeley Packet Filter (eBPF) is an enhancement to the Linux operating system kernel that allows specific instruction sets to run (sandboxed) inside the kernel. It allows companies to add features to Linux without changing kernel source code or requiring kernel modules.

Why This Is Important

eBPF increases the extensibility of Linux. It allows users to create hooks that are triggered by Linux kernel events. This offers a safer and simpler way to add capabilities, such as performance, security and visibility, in Linux. Technology vendors like ISVs and cloud providers use eBPF to avoid kernel-level modules, which carry inherent risks. eBPF is used in production at scale by hyperscalers, including AWS, Facebook and Netflix, and content delivery networks (CDN) such as Cloudflare.

Business Impact

eBPF improves observability, security and performance for applications. However, most enterprises will not use eBPF directly. Technology vendors do use eBPF as an underpinning technology in their products and services to improve the performance and safety of programs that run on Linux. eBPF allows extremely technically savvy organizations to safely and quickly make changes to Linux, compared to using alternative approaches, such as Linux kernel modules or upstreaming to the Linux distribution.

Drivers

- eBPF usage is driven by hyperscalers using it to deliver more efficient cloud offerings, as well as networking, monitoring and security vendors that use it in their products.
- Hyperscalers use eBPF to remediate kernel vulnerabilities without patching to address Day 0 vulnerabilities, and to more efficiently handle distributed denial of service (DDoS) attacks.
- Organizations are looking to accelerate the development speed of software that runs on Linux via avoidance of the requirement for upstream inclusion into the Linux distribution.
- Organizations are looking to improve the performance, security and monitoring capabilities of software running on Linux.
- eBPF is popular among technologically advanced companies, including technology vendors and hyperscalers, because it provides a standardized interface, supported kernel portability and requires less in-depth kernel programming knowledge.
- eBPF helps overcome scale and visibility limitations of iptables, which is the default networking stack in Linux. eBPF helps optimize and customize Linux network packet handling by processing them earlier in the cycle.
- Vendors are increasingly using eBPF in their career network infrastructure (CNI) software to improve performance, security and network visibility.

Obstacles

- While it is realistic for technology vendors and hyperscalers, most enterprises lack the expertise and skills necessary to build and integrate eBPF-based functions.
- Most enterprises do not have the awareness, need or risk tolerance to tackle Linux kernel challenges directly.
- Many older Linux kernels don't support eBPF, or only partially support the latest features.
- Security and system reliability concerns will severely limit what organizations are willing to deploy using eBPF, as poorly written eBPF programs can directly impact the operation of the Linux kernel.
- Integration challenges and backward compatibility with existing non-eBPF-enabled products.

User Recommendations

- Migrate to more modern platforms for organizations that are still using Linux distributions with limited or no eBPF support.
- Seek eBBF-based Kubernetes CNI solutions when scale, performance, visibility and security are top of mind.
- Use Linux variants that provide eBPF support to enable network performance, visibility and security products.
- Explore whether eBPF can meaningfully address the organization's performance or visibility challenges by supporting technologically advanced enterprises.
- Invest in eBPF to improve performance and visibility, to avoid falling behind competitors, for networking and network security vendors.

Sample Vendors

Aqua; Cloudflare; Isovalent; New Relic; Splunk; Sysdig; Tigera

Gartner Recommended Reading

Cool Vendors in Cloud Networking

Using Emerging Service Connectivity Technology to Optimize Microservice Application Networking

DEX Tools

Analysis By: Dan Wilson, Autumn Stanish, Stuart Downes, Tom Cipolla

Benefit Rating: High

Market Penetration: 20% to 50% of target audience

Maturity: Early mainstream

Definition:

Digital employee experience (DEX) tools help IT leaders measure and continuously improve the performance and employee sentiment toward company-provided technology. Near-real-time processing of aggregated data from endpoints, applications, employee sentiment and organizational context surfaces actionable insights and drives self-healing automation, optimized support and employee engagement. Insights and self-healing can also enhance IT support.

Why This Is Important

Accelerated digital workplace investment has highlighted gaps in objective measurement and continuous improvement of DEX. Client interest in DEX has steadily increased since the start of 2021. Primary use cases focus on tactical and technology issues however mature digital workplaces are expanding to include more strategic use cases. Their cross-functional DEX strategy directly targets reduced IT overhead and improved DEX as a way to retain and attract top talent.

Business Impact

DEX tools shift focus from technology management to more business value-added work. Specific impacts include:

- Fewer IT issues that disrupt and impede employee productivity.
- Reduced IT overhead through automation.
- Improved endpoint configuration and patch compliance.
- Better balance of objective and subjective success measures, including technology adoption, performance and employee sentiment.
- IT becoming more proactive and human-centric.

- Increased ability to retain talent.

Drivers

- DEX is a major influencer of the overall employee experience.
- Organizations are increasingly dependent on technology to perform their work.
- Employees are suffering in silence by living with or working around issues rather than reporting issues to IT.
- IT leaders seek broader measurement and management capabilities as internally focused activity KPIs have proven incomplete.
- IT administrators are looking for better visibility into how hybrid workers' devices are performing.
- Employee sentiment toward technology cannot be measured effectively with periodic or transactional surveys alone. Feedback must also include how employees feel about and engage with specific devices or apps, and how technology changes impact their work.
- Service desk and other IT support analysts require faster access to device configuration and performance data to offset an increase in support interaction volumes and wait times.
- Increasing threat of cyberattacks demands faster identification and remediation of configuration issues and missing patches.
- Increased focus on sustainable IT is promoting consumption- and performance-based device life cycles in place of refreshing devices on a schedule.
- AI and machine learning have significantly increased the value and capability of SaaS-based DEX tools.

Obstacles

- Legacy culture that does not trust the tool's insights or sees automation as a threat.
- SaaS- or cloud-averse organizations will be limited to less capable on-premises offerings.
- Low-maturity IT support or end-user computing (EUC) organizations may not be ready for DEX tools.
- An "ignorance is bliss" mindset fearing that a sudden unveiling of the massive volume issues will make IT leadership look bad.
- The cost to acquire, implement and integrate new tools.
- Insufficient staffing levels or skills required to operate a DEX tool.
- Failure to adjust IT staff rewards and recognition to promote new behaviors and DEX tool adoption.
- The need to account for legislative, regulatory, industry or labor union limits on data collection and use.
- The lack of maturity and feature parity among representative and similar tools including common APIs for integration.
- Smaller organizations have limited options given that many DEX tools target larger enterprises.

User Recommendations

In its third year on the Hype Cycle, DEX tools have reached the Peak of Inflated Expectations. Market penetration and maturity have also advanced. Organizations that have not invested in DEX tools should:

- Build a broader team by collaborating with business and IT peers to define IT and non-IT use cases.
- Ensure the business case focuses on objective and measurable impacts by minimizing reliance on vendor-provided ROI templates.
- Choose a DEX tool that best fits your needs and budget by using the Market Guide for DEX Tools.

- Assign dedicated ownership and allocate dedicated resources to deploy and drive DEX tool adoption and ROI. Resources can be reallocated from IT support roles as proactive automation reduces support volumes.
- Incentivize new behaviors by adapting IT performance measures to focus more on outcomes than activities.
- Avoid diminishing returns by adding features and use cases as the team and DEX tool matures.

Sample Vendors

1E; ControlUp Technologies; HP Inc.; Ivanti; Lakeside Software; Nanoheal; Nexthink; Riverbed Technology; Tanium; VMware

Gartner Recommended Reading

How to Successfully Deploy a DEX Tool

Market Guide for DEX Tools

Employee Enablement Is Key to Digital Workplace Services Leaders' Survival

Performance Engineering

Analysis By: Joachim Herschmann

Benefit Rating: High

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Definition:

Performance engineering is a systematic approach to developing software applications to ensure they meet the application performance objectives of the business. It focuses on the architecture, design and implementation choices that will affect application performance, and encompasses practices that help mitigate associated risks before progressing to subsequent phases.

Why This Is Important

The ability to consistently deliver products that satisfy end-user expectations of scalability, stability, quality of service (QoS), availability and performance has become crucial for digital businesses. Performance engineering promotes a holistic and proactive approach, with performance requirements driving the design, development and delivery of products as part of a continuous quality strategy.

Business Impact

Performance engineering includes both “shift left” and “shift right” testing practices and significantly improves an organization’s ability to consistently deliver solutions that exceed customers’ performance expectations. Organizations can improve application observability by using the insights gained through performance engineering. This includes adding metrics and telemetry, or deploying new application monitoring tools so alerts can be raised before a performance bottleneck impacts users.

Drivers

- Increased end-user expectations for application quality, specifically operational characteristics such as performance efficiency.
- The need to ensure business continuity under changing usage patterns, network topologies and data volumes.
- The need to optimize the use of modern microservices-based architectures, as well as the automation and integration capabilities of modern application platforms.
- Support for different migration scenarios, such as lift and shift, replatforming or refactoring the architecture of packaged or on-premises apps.
- The need to manage performance and scalability across different cloud providers, such as Amazon Web Services (AWS), Google or Microsoft Azure, to ensure the ability to shift from one operator to another without a change in user experience.
- Cost optimization for SaaS/PaaS services that makes use of dynamic infrastructure to spin up (and down) testing resources as needed.

Obstacles

- Many organizations focus only on tools and technology. Performance engineering requires a change in organizational culture. Tools enable quality but won't solve problems on their own.
- Departmental silos, traditional top-down management structures and a lack of experience with managing quality continuously can impede adoption.
- Successful performance engineering requires clear goals that are aligned with the priorities of the business. The absence of established service-level indicators (SLIs) and objectives (SLOs) leads to a lack of realistic, quantifiable service availability or performance metrics.
- Performance engineering requires engaging stakeholders throughout the organization and empowering them to be more accountable and to seek out opportunities for improvement. Such a holistic approach can be seen as restrictive and requires consensus across all team members.
- Performance engineering includes designing with performance in mind, building the product with clear performance objectives and facilitating the discovery of issues early in development.

User Recommendations

- Create awareness of nonfunctional or operational characteristics, such as performance efficiency or the Application Performance Index (Apdex), an open standard developed by an alliance of companies to measure the performance of software applications in computing. ISO/IEC 25010 provides a template for understanding quality characteristics and includes performance efficiency as one of the top-level nonfunctional domains.
- Establish SLIs and SLOs as part of a performance engineering strategy. An SLI is a realistic, quantifiable measure of service availability or performance. An SLO is the target for the SLI over a fixed period.
- Foster a proactive performance quality strategy that makes performance an explicit requirement and verifies that performance goals are met and user satisfaction meets expectations.
- Allocate ownership and appoint staff with the skills needed for performance engineering by identifying the required roles, technologies and practices.
- Establish performance quality metrics based on the joint objectives that the business and IT are trying to accomplish.
- Collaborate with I&O leaders and establish a feedback loop using performance information from production and real users.

Sample Vendors

AppDynamics; Dynatrace; Keysight (Eggplant); Micro Focus; Perforce Software; Tricentis

Gartner Recommended Reading

How to Identify and Resolve Application Performance Problems

Quick Answer: Which Non-Functional Software Quality Characteristics Can Make or Break Your Product?

Improve Software Quality by Building Digital Immunity

Log Monitoring and Analysis

Analysis By: Pankaj Prasad, Gregg Siegfried

Benefit Rating: High

Market Penetration: 20% to 50% of target audience

Maturity: Early mainstream

Definition:

Log monitoring and analysis solutions provide log ingestion, storage, interactive search, dashboards, alerts and advanced diagnostic analysis. Log-monitoring solutions apply advanced analytics or machine learning (ML) to reduce the operator's cognitive load through contextualization, correlation and analysis of large volumes of log data from multiple data sources.

Why This Is Important

Storing and searching log files is reactive and relies on human operators, which is not effective for IT architectures currently being deployed. The focus of traditional log monitoring is entity-centric anomaly detection and manual diagnosis. Automated log analytics uses statistical analysis, pattern recognition, correlations and machine learning to accelerate identification and resolution of service-impacting anomalies.

Business Impact

Log monitoring and analysis will impact:

- **Customer journeys** — Log analysis can help in behavior analysis and mapping user journeys, enabling better decisions.
- **Enhanced anomaly detection** — This enables monitoring of logs for patterns, which provides better insights into potential anomalies over static thresholds.
- **Operational efficiency** — Logical and temporal correlations of log data across multiple data sources enables better diagnosis and faster root-cause analysis.

Drivers

- Data explosion with modern architectures — As organizations adopt geographically distributed, container- and microservice-based architectures, the amount of log data generated — already in the petabyte per day range at some organizations — will start approaching exabyte levels. At scale, collection, storage and gaining insights from logs is not feasible without automated analysis.
- Operational challenges — I&O organizations need to improve mean time to repair (MTTR) by reducing manual correlation effort time.
- Root cause analysis — I&O and site reliability engineering (SRE) teams looking for ways and means to enhance the resilience of their IT architectures need a faster way to identify correlating patterns and relevant logs beyond the system of interest.

Obstacles

- Implementation — ML models demand heavy investment for either supervised or unsupervised training to be accurate over time, thus increasing the time to value for these solutions. The noncentralized nature of log data in many organizations further hampers the quality of outcomes.
- Basic use cases — These products do not enable ease of higher-order use cases beyond I&O due to the lack of an advanced interface that can be used by a data-scientist-type persona.
- Cost — Whether a log-monitoring solution is deployed on cloud or on-premises (as either an open-source or proprietary option), the licensing or maintenance costs increase with the volume of log data and data-retention policies. Balancing total cost of ownership (TCO) for log monitoring solutions and preserving historical information for analysis is a prevalent problem for I&O organizations.

User Recommendations

- Simplify log analysis by establishing a log governance model that standardizes field names and data formats throughout multiple sources of log data. This facilitates the pooling of log data from multiple sources and results in more efficient queries.
- Evaluate advanced analysis and out-of-the-box capabilities and use cases during product assessment for log monitoring and analysis.
- Choose a solution that supports multiple sources of logs and broad use cases, and deploy them centrally. Log collection and analysis should not be deployed in silos.

Gartner Recommended Reading

Guidance Framework for Deploying Centralized Log Monitoring

Market Guide for AIOps Platforms

Infographic: AIOps Architecture for Analyzing Operational Telemetry

Native Cloud Monitoring

Analysis By: Padraig Byrne

Benefit Rating: Moderate

Market Penetration: 20% to 50% of target audience

Maturity: Early mainstream

Definition:

Native cloud monitoring refers to the monitoring tools and services that are offered within the cloud portfolio. While the primary focus of native cloud monitoring is the vendor's own cloud services, vendors offering these services have expanded to support hybrid environments in limited circumstances.

Why This Is Important

Organizations continue expanding their usage of cloud services as part of digital transformation and shifting to agile processes. Developers have shown a desire to stay within the cloud ecosystem, selecting native tools like databases, messaging systems or machine learning tools to build applications. By extending native monitoring capabilities, cloud vendors allow developers and operations teams the same choice for their IT monitoring.

Business Impact

Organizations that use native cloud monitoring capabilities gain value through:

- **Unified monitoring platform:** Improved visibility of overall cloud applications and services with logs, metrics and traces in a centralized repository and platform.
- **Simplified pricing model:** Pricing in most cases is based on data volume and often cheaper than third-party monitoring solutions.

Drivers

- **Native integration:** A cloud monitoring solution that natively integrates into the overall cloud services ecosystem becomes an enticing first preference for organizations.
- **Modern architectures:** The ephemeral nature of modern microservices and container-based workloads, coupled with the explosion of telemetry data, makes traditional monitoring unsuitable for monitoring cloud services.
- **Adoption of PaaS:** Traditional agent-based monitoring techniques cannot monitor PaaS services.
- **Cost:** Modern monitoring solutions like APMs tools can be costly. By leveraging the cloud provider's native monitoring tools, organizations can take advantage of combining the monitoring costs in the overall cloud spend.

Obstacles

Organizations that plan to use native cloud monitoring solutions should plan to address the following challenges:

- **Limited reach:** Native cloud monitoring solutions are primarily focused on monitoring their own ecosystem and may not be best to extend for multicloud monitoring strategies.
- **Misconfigurations:** While native tools can drive down overall monitoring costs, misconfiguration of the telemetry settings can lead to excessive costs.
- **Multicloud:** Many organizations have capabilities in more than one cloud provider. By using only native cloud monitoring tools, they limit the ability to get a cross-platform view of all their applications and services. Costs can also increase due to the use of different tools in each cloud, as well as the need to skill staff in more than one toolset.
- **Costs:** While native cloud monitoring solutions often reduce costs for monitoring, challenges can arise in complex architectures around scaling the costs for the solution.

User Recommendations

I&O leaders and technical professionals responsible for monitoring and observability should:

- Evaluate both native cloud monitoring and APM solution capabilities with your organization's criteria to select the best option.
- Implement cost management tools (either native to the cloud provider or third party) in order to maintain control over monitoring costs.
- Choose third-party solutions that take advantage of cloud-vendor-provided monitoring APIs — especially in hybrid or multicloud monitoring situations.
- Leverage broad geographic coverage of cloud provider's regions and availability zones for your in-region data residency requirements that require the telemetry data to reside in specific locations.

Sample Vendors

Alibaba; Amazon Web Services; Google; Microsoft; Oracle

Observability

Analysis By: Padraig Byrne, Gregg Siegfried

Benefit Rating: Transformational

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Definition:

Observability is the characteristic of software and systems that enables them to be understood, based on their outputs and enables questions about their behavior to be answered. Tools that facilitate software observability enable observers to collect and quickly explore high-cardinality telemetry using techniques that iteratively narrow the possible explanations for errant behavior.

Why This Is Important

The inherent complexity of modern applications and distributed systems and the rise of practices, such as DevOps, has left organizations frustrated with legacy monitoring tools and techniques. These can do no more than collect and display external signals, which results in monitoring that is, in effect, only reactive. Observability acts like the central nervous system of a digital enterprise. Observability tools enable a skilled observer to explain unexpected system behavior more effectively.

Business Impact

Observability tools have the potential to reduce both the number of service outages and their severity. Their use by organizations can improve the quality of software, because previously invisible (unknown) defects and anomalies can be identified and corrected. By enabling product owners to better understand how their products are used, observability supports the development of more accurate and usable software, and a reduction in the number and severity of events affecting service.

Drivers

- The term “observability” is now ubiquitous, with uses extending beyond the domain of IT operations. Although the 2020s are now the “decade of observability,” care must be taken to ensure the term retains relevance when used beyond its original range of reference.
- OpenTelemetry’s progress and continued acceptance as the “observability framework for cloud-native software” raises observability and its toolchain.
- Traditional monitoring systems capture and examine signals (possibly adaptive) in relative isolation, with alerts tied to threshold or rate-of-change violations that require prior awareness of possible issues and corresponding instrumentation. Given the complexity of modern applications, it is unfeasible to rely on traditional monitoring alone.
- Observability tools enable a skilled observer, a software developer or a site reliability engineer to explain unexpected system behavior more effectively, provided enough instrumentation is available. Integration of software observability with artificial intelligence for IT operations (AIOps) to automate subsequent determinations is a potential future development.
- Observability is an evolution of longstanding technologies and methods, and established monitoring vendors are starting to reflect observability ideas in their products. New companies are also creating offerings based on observability.

Obstacles

- In many large enterprises, the role of IT operations has been to “keep the lights on,” despite constant change. This, combined with the longevity of existing monitoring tools, means that adoption of new technology is often slow.
- Enterprises have invested significant resources in their existing monitoring tools, which exhibit a high degree of “stickiness.” This creates nontechnical, cultural barriers to adopting new practices such as those based on observability.
- Costs associated with observability tools have grown as companies struggle to keep up with the explosion in volume and velocity of telemetry.

User Recommendations

- Assess software observability tools to integrate into their continuous integration/continuous delivery (CI/CD) pipelines and feedback loops.
- Investigate problems that cannot be framed by traditional monitoring by using observability to add flexibility to incident investigations.
- Enable observability by selecting vendors that use open standards for collection, such as OpenTelemetry.
- Tie service-level objectives to desired business outcomes using specific metrics, and use observability tools to understand variations.
- Ensure IT operations and site reliability engineering teams are aware of updates to existing monitoring tools and how they may take advantage of them. Many traditional application performance monitoring vendors are starting to incorporate observability features into their products.
- Avoid the conclusion that observability is synonymous with monitoring. At minimum, observability represents the internal perspective, rather than external.

Sample Vendors

Chronosphere; Grafana; Honeycomb; Lightstep; Observe; VMware

Gartner Recommended Reading

Monitoring and Observability for Modern Infrastructure and Applications

Magic Quadrant for Application Performance Monitoring and Observability

Sliding into the Trough

Digital Experience Monitoring

Analysis By: Mrudula Bangera, Padraig Byrne

Benefit Rating: High

Market Penetration: 20% to 50% of target audience

Maturity: Mature mainstream

Definition:

Digital experience monitoring (DEM) technologies monitor the availability, performance and quality of experience for an end user or digital agent as they interact with an application and the supporting infrastructure. Users can be external consumers of a service, internal employees accessing corporate tools, or a combination of both. DEM technologies seek to observe and model the behavior of users as a continuous flow of interactions in the form of “user journeys.”

Why This Is Important

DEM helps organizations address visibility in two key areas:

- **Remote employees’ experience:** Instrumenting the corporate network is relatively easy. Doing the same for a home or coffee shop network ranges from challenging to impossible.
- **Web applications:** Visibility into the performance of as-a-service-based applications (including e-commerce) presents a unique challenge, due to the location of the application and difficulty in instrumenting cloud-based environments.

Business Impact

RUM and STM technologies in DEM allow businesses to understand how the users (customers) are interacting with the brand across mobile and web. The endpoint monitoring technology gives organizations increased flexibility to gain visibility into the endpoint, network and service of the user, irrespective of where workers are located, and without requiring extensive instrumentation of the physical environment.

Drivers

- **User experience:** Organizations are coming to the realization that metrics tell only part of the story. If the user is having a less-than-ideal experience, then whatever the metrics say are meaningless. DEM can help provide visibility into not just the metric-based performance, but also the subjective portion of the user experience.
- **SaaS:** As organizations move from on-premises-based applications to SaaS-based applications, they lose visibility into, and control over, the performance of these applications. A user of a SaaS-based application in one location using a specific endpoint (such as a laptop or mobile) may have a totally different experience from a different user at a different location using a different endpoint. Even the same user at the same endpoint may have very different experiences, depending on where they are located at the time. DEM enables organizations to understand where the performance bottlenecks are, so they can be addressed.
- **Work from anywhere:** The massive changes in workforce location brought on by the COVID-19 pandemic are driving infrastructure and operations (I&O) teams to adopt endpoint monitoring technologies to analyze and optimize remote workers' access to, and use of, applications.
- **"Last mile" in full-stack observability:** Monitoring of applications from the server side is important, but I&O teams need to understand the end-user journey and the corollary experience. Endpoint monitoring through DEM tools allows I&O teams to track performance from the endpoint's connectivity to Wi-Fi through service provider networks and beyond.
- **Commercial off-the-shelf (COTS) and virtual desktop infrastructure:** Organizations often rely on COTS applications for critical business operations. The very nature of these solutions makes them difficult (if not impossible) to instrument from an application perspective. I&O teams rely on the visibility provided by DEM tools to provide information on performance from the end user's perspective.

Obstacles

- There are very few DEM vendors that provide functionality across all three pillars of DEM (synthetic monitoring, endpoint visibility and real-user monitoring), making it difficult to choose a vendor that can provide a complete solution.
- Most DEM visibility comes from an agent installed on the endpoint, which can represent a challenge for organizations that are already running numerous endpoint agents.

- Large organizations may struggle with the management of tens or hundreds of thousands of endpoints via a DEM tool user interface.
- Due to the sheer volume of data generated by DEM tools, organizations without a robust analytics approach may struggle to make sense of all the data. Few vendors use analytics to enable a proactive approach in this space.
- User experience can be enhanced through autorectification of anomalies. However, very few DEM vendors provide the ability to automate remediation.

User Recommendations

- Gain a holistic view of digital experience by choosing and deploying DEM solutions that gather sentiment alongside other data points.
- Minimize endpoint performance impacts by evaluating DEM capabilities from vendors and tools you already own (for example, DEM capabilities from a unified endpoint management , security or remote access vendor).
- Enable insight-driven automation by choosing DEM solutions that provide analytics and remediation functions.
- Measure SaaS application performance by choosing DEM solutions that can perform real-user monitoring and synthetic transaction monitoring.
- Gain transparency into employee experience by monitoring as many endpoints as possible.

Sample Vendors

Apica; Catchpoint; Cisco; Fortinet; Kadiska; Lakeside Software

Gartner Recommended Reading

Market Guide for Digital Experience Monitoring

How to Monitor and Troubleshoot Remote Workers' Application Performance

3 Ways to Optimize Observability and Monitoring of Digital Services in the Cloud

Use DEM to Understand and Enhance Your Employees' Work-From-Home Experience

Use Synthetic Monitoring to Enhance User Experience for Hosted and SaaS Applications

Site Reliability Engineering

Analysis By: George Spafford, Daniel Betts

Benefit Rating: Transformational

Market Penetration: 5% to 20% of target audience

Maturity: Adolescent

Definition:

Site reliability engineering (SRE) is a collection of systems and software engineering principles used to design and operate scalable resilient systems. Site reliability engineers work with the customer or product owner to understand operational requirements and define service-level objectives (SLOs). Site reliability engineers work with product or platform teams to design and continuously improve systems that meet defined SLOs.

Why This Is Important

SRE emphasizes the engineering disciplines that lead to resilience; but individual organizations implement SRE in widely varying ways such as a defined role or a set of practices. SRE teams can serve as an operations function, and nearly all such teams have a strong emphasis on blameless root cause analysis. This is to decrease the probability and/or impact of future events and to enable organizational learning, continual improvement and reductions in unplanned work.

Business Impact

The SRE approach to improving reliability and resilience is intended for products and platforms that need to deliver customer value at speed at scale while managing risk. The two primary use cases are to improve the reliability of existing products/platforms or to create new products or platforms that need reliability from the start.

Drivers

- Clients are under pressure to meet customer requirements for reliability while scaling their digital services and are looking for guidance to help them.
- While Google originated what became known as SRE and continued to evolve it, practitioners are developing and sharing new practices as well. Potential practitioners looking for pragmatic guidance to improve the reliability of their systems have a rich body of knowledge they can leverage that works well with agile and DevOps.
- Organizations are adopting highly skilled automation practices (usually DevOps), and usage of infrastructure-as-code capabilities (which usually requires a cloud platform) to deliver digital business products reliably.
- The most common use case based on inquiry calls with clients is to leverage SRE concepts to improve the reliability of existing systems that are not meeting customer requirements for availability, performance or are proving difficult to scale.

Obstacles

- Insufficient internal marketing to understand what agile, DevOps or product teams need or would value and then explaining how the value SRE can deliver will justify the costs and risks incurred. Without marketing its benefits, SRE adoption tends to be less certain or slower. The SRE concept by itself is insufficient — people must continuously believe it is worthwhile.
- Finding SRE candidates who have the right mix of development, operations and people skills is a big challenge for clients. Impacts on initial adoption and scaling efforts as well.
- Rebranding of a traditional operations team without changing to adopt SRE practices, only SRE in name.
- Clients have voiced problems with product owners who overly focus on functional requirements and not nonfunctional requirements thus slowing improvements and support of SRE within the organization.

User Recommendations

- Leverage practices pragmatically based on need. Don't feel that you must implement SRE exactly the way Google does it, learn what works for you.
- Detect an opportunity to begin that is politically friendly, will demonstrate sufficient value and has an acceptable risk profile.
- Start small, focus, learn, improve, and demonstrate value — do not try to change everything at once.
- Work with the customer or product owner to define clear, obtainable SLOs based on their needs.
- Implement monitoring and improve observability to objectively report on actual performance relative to the SLOs.
- Product owners must be accountable for functional and non-functional requirements of their products.
- Instill collaborative working between site reliability engineers, developers and other stakeholders to help them learn how to design, build and evolve their products to meet SLOs.
- Create a community, implement effective organizational learning practices and evolve SRE practices.

Sample Vendors

Atlassian; Blameless; Datadog; Dynatrace; New Relic; OpsRamp; PagerDuty; Splunk

AIOps Platforms

Analysis By: Matt Crossley, Matthew Brisse

Benefit Rating: High

Market Penetration: 20% to 50% of target audience

Maturity: Early mainstream

Definition:

Gartner defines AIOps platform as the application of AI/ML and data analytics at the event management level in order to augment, accelerate and automate manual efforts in the event management process and associated procedures. AIOps platforms are defined by the key characteristics of cross-domain event ingestion, topology assembly, event correlation and reduction, pattern recognition, and remediation augmentation.

Why This Is Important

The combination of increasing application complexity, monitoring tool proliferation, and increasing volumes and varieties of telemetry has shifted complexity from gathering data to interpreting data. AIOps platforms apply machine learning (ML) and data analytics to classify and cluster cross-domain events in near real time, at scale, and in ways that can exceed human capacity. These inferences can augment human analysis, accelerate human response, or automate a process to resolve an issue.

Business Impact

AIOps platforms deliver value through:

- Agility and productivity: By reducing alert fatigue through identification and correlation of related events, operators can focus on fewer, more critical events.
- Service availability and triage cost: By reducing the time and effort required to identify root causes and augmenting, accelerating, or automating remediation.
- Increased value from monitoring tools: By unifying events from siloed tools and learning actionable event patterns across domains.

Drivers

Demand for AIOps platform capabilities is accelerating and is fueled by:

- **Increasing complexity:** Organizations use an increasingly complex mix of IT assets that rely on a highly integrated combination of on-premises assets, cloud IaaS/PaaS providers and SaaS platforms to deliver solutions.

- **Increasing monitoring expectations:** Investments and improvements in monitoring and the pursuit of observability are generating more data from more sources. Increasing demand and advances in monitoring trends, like application performance management (APM) and digital experience monitoring (DEM), present operators with extremely detailed views into their business applications and the end-user experience. Effective use of this additional data requires near-real-time analysis and rationalization of events from related assets and services.
- **Demands for reliability:** Shifts in roles and responsibilities driven by modern operating models, like DevOps and SRE, in the pursuit of greater availability and faster incident resolution. AIOps platforms enable agility by offloading some of the mechanical tasks of event triage, root cause analysis and solution identification. This both accelerates response for common issues and frees up human creative capacity for novel events and business priorities.

Obstacles

- **Unrealistic expectations:** Hype is a major obstacle to AIOps platform adoption. Clients struggle to separate claims of AI and magical automation from achievable use cases. This impacts demonstrating value of AIOps platforms, specifically quantifiable return on investment.
- **Maturity of dependencies:** Benefits of AIOps platforms beyond event correlation requires maturity in dependencies such as automation.
- **Time to value:** AIOps platforms learn through observation, modeling normal data patterns, and associate a solution with these patterns. This can take time depending on the frequency of occurrence. Developing accurate detection models for rare events can take months.
- **Market shifts and maturity:** Monitoring vendors are moving up the stack, AIOps platform vendors are reaching into monitoring domains, and ITSM vendors use AIOps capabilities to extend their reach. Expect further convergence and market shifts to change the definition of “state of the art.”

User Recommendations

- Establish clear, realistic use cases for an AIOps platform pilot and validate them individually, rather than all at once. This approach helps reveal pockets of potential value that might be missed when evaluating only the aggregate impact. Ultimately, this fundamental step underpins an eventual strategy, while scoping the vendor landscape, clarifying technical and process dependencies, and separating hype from reality.
- Layer the AIOps features within monitoring tools with the cross-domain analysis of an AIOps platform. This approach enables efficient data ingestion and analysis, and the surfacing of insights across domains.
- Do not require automation outcomes for all AIOps applications. There is tremendous value in accelerating and augmenting human activity. These approaches often avoid the challenge of the probabilistic uncertainty combined with automated change in production environments.

Sample Vendors

BigPanda; BMC Software; Digitate; IBM; Interlink; Moogsoft; OpsRamp; PagerDuty; ServiceNow; Splunk

Gartner Recommended Reading

Market Guide for AIOps Platforms

Deliver Value to Succeed in Implementing AIOps Platforms

Infographic: Artificial Intelligence Use-Case Prism for AIOps

Infographic: AIOps Architecture for Analyzing Operational Telemetry

How Do I Plan for Migrating My Data Center Infrastructure Into an XaaS Model?

Service Mesh

Analysis By: Anne Thomas

Benefit Rating: Low

Market Penetration: 1% to 5% of target audience

Maturity: Adolescent

Definition:

A service mesh is a distributed computing middleware that manages communications between application services — typically within managed container systems. It provides lightweight mediation for service-to-service communications and supports functions such as authentication, authorization, encryption, service discovery, request routing, load balancing, self-healing recovery and service instrumentation.

Why This Is Important

A service mesh is lightweight middleware for managing and monitoring service-to-service (east-west) communications — especially among microservices running in container management systems, such as Kubernetes. It provides visibility into service interactions, enabling proactive operations and faster diagnostics. It automates complex communication concerns, thereby improving developer productivity and ensuring that certain standards and policies are enforced consistently across applications.

Business Impact

Service mesh is one of many management technologies that provide software infrastructure for distributed applications. Service meshes are most often used with services deployed in container management systems, such as Kubernetes. This type of middleware, along with other management and security middleware, helps provide a stable environment that supports “Day 2” operations of containerized workloads. However, the technology is complex and often unnecessary for smaller deployments.

Drivers

- Microservices and containers: Service mesh adoption is closely aligned with microservices architectures and container management systems like Kubernetes. Service mesh supports useful functionality in ephemeral environments, such as dynamic service discovery and mutual Transport Layer Security (mTLS) between services.
- Observability: As microservice deployments scale and grow more complex, DevOps teams need better ways to track operations, anticipate problems and trace errors. Service mesh automatically instruments the services and feeds logs to visualization dashboards.
- Resilience: A service mesh implements the various communication stability patterns (including retries, circuit breakers and bulkheads) that enable applications to be more self-healing.
- Bundled feature: Many container management systems now include a service mesh, inspiring DevOps teams to use it. The hyperscale cloud vendors provide a service mesh that is also integrated with their other cloud-native services.
- Federation: Independent vendors such as Buoyant, greymatter.io, HashiCorp, Kong and Solo provide service meshes that support multiple environments.

Obstacles

- Not necessary: Service mesh technology can be useful when deploying microservices in Kubernetes, but it's never required.
- Complexity: It's complex to use and administer, and there are increasing discussions on why not to use a service mesh in technology discussion groups and social media.
- Redundant functionality: Users are confused by the overlap in functionality among service meshes, ingress controllers, API gateways and other API proxies. Management and interoperability among these technologies is still nascent within the vendor community.
- Overhead: Service mesh technology consumes resources and typically adds overhead to the interactions it manages. Some vendors now support alternate architectures, such as a shared-agent model to reduce overhead, but this solution reduces the observability benefits.
- Competition with "free": Independent service mesh solutions face challenges from the availability of platform-integrated service meshes from the major cloud and platform providers.

User Recommendations

- Determine whether the value you might get from a service mesh in terms of improved security or observability is worth the increase in complexity and administration of the service mesh. A service mesh becomes more valuable as the number of service-to-service (east-west) interactions increases.
- Favor the service meshes that come integrated with your container management system unless you have a requirement to support a federated model.
- Reduce cross-team friction by assigning service mesh ownership to a cross-functional platform engineering team that solicits input and collaborates with networking, security and development teams.
- Accelerate knowledge transfer and consistent application of security policies by collaborating with I&O and security teams that manage existing API gateways and application delivery controllers.

Sample Vendors

Amazon Web Services; Ambient Mesh; Buoyant; Google; HashiCorp; Istio; Kong; Microsoft; Solo.io

Gartner Recommended Reading

How a Service Mesh Fits Into Your API Mediation Strategy

Intent-Based Networking

Analysis By: Andrew Lerner

Benefit Rating: Moderate

Market Penetration: Less than 1% of target audience

Maturity: Emerging

Definition:

The IETF defines intent-based networking (IBN) as a set of operational goals and outcomes defined in a declarative manner without specifying how to achieve or implement them. Gartner further specifies IBN as a closed-loop system to design, provision and operate a network based on business policies. IBNs translate business policies to network configurations, automate network activities, maintain an awareness of network health and provide continuous network assurance and dynamic optimization.

Why This Is Important

Intent-based networks simultaneously improve network agility and reliability while enabling a common policy across multiple infrastructures. Unfortunately, the term is used loosely by network vendors. Thus, most offerings marketed as intent-based fall short of the complete functionalities of intent-based networking. Instead, we observe an incremental adoption of the subcomponents of an IBN, including network automation, configuration validation and network assurance.

Business Impact

A complete IBN implementation can reduce the time taken to deliver network infrastructure to business leaders by an estimated 50% to 90%. It can simultaneously reduce the number and duration of outages by an approximate 50%. However, there has been limited impact to date due to low real-world adoption and a lack of viable, easy-to-use full IBN products.

Drivers

- There is a desire to make networks more agile in conjunction with cloud deployments and digital business.
- Intent-based networking offers a reduced operating expenditure (opex) associated with managing networks. Therefore, more senior-level network resources are free to focus on more important strategic tasks.
- There is a desire to simplify network administration amid the increasing complexity of networking with overlays, cloud environments and containers.
- IBN allows real-time self-documentation, which also includes the rationale — that is, the intent behind design or configuration decisions.
- IBN can lead to improved compliance and simplified auditing. This is due to the algorithmic correctness of configurations that provide self-validation, direct mapping to the business intent and ongoing, dynamic and real-time validation.
- IBN can help to reduce the impact of enterprises that are not able to hire or retain senior-level network engineers and architects.
- In October 2022, the IETF published the informational document, RFC9315, which helps to streamline IBN terminology and definition. This research document can reduce confusion, foster consistency and consequently clear hurdles to adoption.

Obstacles

- Only a limited number of vendors offer complete IBN capabilities. Very few offerings translate a higher-level intent into network configuration — we estimate that there are fewer than 1,000 full deployments.
- Network automation, AI networking and AIOps all deliver some of the value intent-based networks offer, and are often simpler to implement. This limits, prevents or delays IBN.
- Vendors are increasingly delivering recommendation engines and predictive capabilities in their management products, which limits or delays IBN.
- Very few vendors provide full intent-based networks. Instead, vendors release products that deliver some discrete individual benefits, often with limited integration between them.
- Full IBN is restricted to greenfield or very homogeneous environments that are deployed in a very prescriptive, structured and specific way. This inflexibility limits adoption.
- It is challenging to define intent for preexisting network deployments.

User Recommendations

- Tune out vendor marketing of products listed as “intent-based” or “intent-driven.” Instead, invest in products that enable network automation and provide network assurance or prescriptive predictions with specific recommendations.
- Invest in network products that provide specific and actionable recommendations down to the device and configuration level when purchasing equipment and tooling solutions. The combination of these investments can help to enable closed-loop automation and operations.

Sample Vendors

Gluware; Juniper Networks; NetBrain

Gartner Recommended Reading

Market Guide for Network Automation Tools

Climbing the Slope

UC Monitoring Tools

Analysis By: Lisa Pierce

Benefit Rating: High

Market Penetration: More than 50% of target audience

Maturity: Early mainstream

Definition:

Unified communications monitoring (UCM) tools collect, analyze and report on the performance of voice, video, and messaging sessions from vendor-supplied data sources, such as call quality metrics and real-time end-user data from devices/clients. Some tools use standard APIs to extract relevant data from UC vendor databases and repositories. Advanced tools also collect session and packet data, decode voice and video codecs, and employ synthetic session/call testing.

Why This Is Important

UC and unified communications as a service (UCaaS) monitoring and management tools are essential to assure a satisfactory end-user experience, across an array of work environments. These tools must combine both application performance monitoring (APM) and network performance monitoring (NPM), by employing synthetic testing with real-time endpoint and end-user monitoring. UCM tools identify and rectify voice, video or messaging sessions performance problems in real time.

Business Impact

Enterprises initially invested in UCM tools, while migrating from on-premises UC platforms to cloud services, to ensure performance targets were met. These new deployments are mostly cloud-based UCaaS solutions. They also link to APIs, which allow third-party monitoring tools to ingest the data. These tools allow administrators to isolate and anticipate problems across a range of real-time services and worker environments, making them essential to eliminate bottlenecks and challenges that impede productivity.

Drivers

- Deployment of UCaaS has been a major driver of adoption of advanced UCM solutions that use digital experience monitoring (DEM) technologies for synthetic testing, and real-time endpoint and end-user monitoring.

- The use of telework solutions that employ residential broadband service, coupled with the rapid adoption of cloud-based videoconferencing, have revealed the fragility of UC applications' performance. Thus, awareness of and appreciation for the role UCM tools play has escalated.
- Organizations are again supporting remote and hybrid workers, in addition to frequent travelers. UC monitoring tools can assess performance across all these environments.
- Implementing long-term remote work plans requires that infrastructure and operations (I&O) shops must support real-time network and application performance capabilities across diverse work environments. Employees who are traveling or are working from home can no longer be treated with benign neglect, especially since tools are available to help deliver the desired experience.

Obstacles

- Very few UCaaS providers offer advanced real-time and synthetic session or call monitoring capabilities, so clients must employ third-party UCM specialist tools.
- Some UCaaS providers offer monitoring tools for their services. However, their monitoring capabilities may not be as strong as those provided by third-party monitoring specialists.
- Some DEM tools provide both APM and NPM functionality, but these tools are not as comprehensive as third-party specialists.
- Niche vendors focus on providing deeper UC insight, often from specific UC vendors. But they often cannot support other applications, or discern root-cause-absent APM and NPM tools. This results in acquiring and using an increasing number of tools.
- Some UCaaS monitoring tools are cloud-only and they do not look at premises-based UC performance. Others are outgrowths of tools that assess the performance of premises-based UC systems, making them less than ideal for cloud-first services.

User Recommendations

- Use UCM tools that blend APM and NPM (real-time monitoring from endpoints and end-users, and synthetic call agents). These solutions may be supplied by dedicated UCM vendors, broader DEM performance monitoring tools, tools provided by UCaaS providers, or a small number of security and software-defined WAN (SD-WAN) vendors. Clients may also employ these four strategies in combination.

- Validate that third-party UCM tools collect and analyze API data provided by the specific UCaaS provider. Third-party tools are helpful in monitoring multivendor or hybrid UC environments, and may provide superior monitoring functions than those offered by the UCaaS provider.
- Assess DEM suppliers' UCM capabilities against UCM specialists and the monitoring capabilities supplied by UCaaS providers, since DEM is applied to many IT assets.
- Affirm that the chosen vendor supports the required proprietary codecs and protocols, when using browser-based UC applications.

Sample Vendors

8x8; ATSG (Optanix); Catchpoint; ExtraHop; IR; Lakeside Software; Nectar Services; RingCentral; TeleMate; Unisys (Unify Square); Vyopta

Gartner Recommended Reading

Market Guide for Digital Experience Monitoring

Negotiate Your Unified Communications as a Service SLAs, Focusing on These Key Terms

Consider These Key Functional Areas for Application Performance Monitoring and Observability

How to Monitor and Troubleshoot Remote Workers' Application Performance

Market Guide for Infrastructure Monitoring Tools

NDR

Analysis By: Jeremy D'Hoinne, Nat Smith, Thomas Lintemuth

Benefit Rating: Moderate

Market Penetration: 20% to 50% of target audience

Maturity: Early mainstream

Definition:

Network detection and response (NDR) products detect abnormal system behaviors by applying behavioral analytics to network traffic. They continuously analyze raw network packets or traffic metadata for both internal (east-west) and “public” (north-south) networks. NDR can be delivered as hardware and software sensor, and software or increasingly SaaS management console. Organizations rely on NDR to detect and contain postbreach activity, such as ransomware, or insider’s malicious activity.

Why This Is Important

NDR focuses on detecting abnormal behaviors, with less emphasis on signature-based controls detecting known threats. NDR is effective in detecting weak signals and previously unknown behavior from traffic on networks such as lateral movement or data exfiltration. NDR solutions expand to hybrid networks, adding new detections. Automated response capabilities, provided natively or through integration remain important, but incident response workflow automation becomes an increasing area of focus.

Business Impact

NDR solutions provide visibility into network activities to spot anomalies. The machine learning algorithms that are at the core of many NDR products help to detect anomalies in traffic that are often missed by other detection techniques. The automated response capabilities help to offload some of the workload for incident responders. NDR products also help incident responders with their threat hunting by providing useful context and drill-down capabilities.

Drivers

- Detecting postbreach activity: NDR complements traditional preventative controls by detecting activities based on deviations from baseline. This allows the security team to investigate insider's activities resulting from breaches without relying on having observed a previous occurrence of the same activity.
- Low risk — high reward: Implementing NDR products is a low-risk project because the sensors are positioned out-of-band, so they don't represent a point of failure or a "speed bump" for network traffic. Enterprises that implement NDR products as a proof of concept (POC) often report high degrees of satisfaction because the tools provide much-needed visibility into network traffic and enable even small teams to spot anomalies.
- Monitoring cloud traffic: A growing number of NDR vendors offer the ability to monitor IaaS traffic and M365 by leveraging available APIs from the cloud providers. Organizations expanding their cloud presence use NDR to avoid creating gaps in their ability to monitor interactions between their systems.

Obstacles

- Enterprises with a lower maturity security operation program might struggle to justify the expense for a technology that cannot simply be evaluated by counting the number of alerts it triggers.
- The response features of the NDR products are more rarely deployed or narrowed down to specific use cases, such as ransomware, due to a risk of false positives. Many organizations postpone their implementation until they understand how to use the NDR tool better.
- NDR is expanding to support more detections in the cloud but have yet to prove they are the right tool for the use case.
- False positives are inevitable with any behavioral-based detection tool. NDR tools might require fine-tuning of the configuration to reduce the amount of false positives, especially in early days of the deployment. This explains why response capabilities are more rarely deployed initially.
- NDR increasingly competes for budget with consolidated platforms such as SIEM and extended detection and response (XDR).

User Recommendations

- Develop a strong understanding of the overall traffic patterns and specific traffic patterns in your enterprise network to gain maximum value from NDR.
- Carefully plan sensor types and deployment locations so that the most relevant network traffic can be analyzed. Proper positioning of the NDR sensors is critically important to limit the number of false positives and control the cost of the deployment.
- Tune out false positives in the implementation phase (false positives may be triggered by vulnerability scanners, shadow IT applications and other factors that may be specific to your environment).
- Plan for ongoing tuning as new detection models are deployed from the vendor.
- Select sensor capturing capacity that is sized appropriately for your network.

Sample Vendors

Cisco; Corelight; Darktrace; ExtraHop; Fortinet; IronNet; MixMode; Plixer; Trend Micro; Vectra

Gartner Recommended Reading

Market Guide for Network Detection and Response

Emerging Tech: Top Use Cases for Network Detection and Response

Automated Incident Response

Analysis By: Pankaj Prasad, Padraig Byrne

Benefit Rating: High

Market Penetration: 20% to 50% of target audience

Maturity: Mature mainstream

Definition:

Automated incident response (AIR) centralizes alert or incident routing through a policy or rule-based engine, on-call scheduler and streamlined collaboration. AIR solution capabilities improve operational efficiencies with action-oriented insights, shorter incident durations and automated workflows for event routing, easier collaboration, remediation and escalations.

Why This Is Important

Manual processes for incident resolution is a challenge, especially when multiple experts need to be involved, time is of essence and the organization wants to improve efficiency. For DevOps teams, the juggling of contact lists and lack of seamless collaboration inhibit speedy delivery of application features, as well as the stability of features after the release. AIR solutions solve this by automating most of the incident response process and collaboration, and enabling iterative improvement.

Business Impact

AIR solutions deliver value through:

- Automated incident communication to the relevant recipient and visibility across the organization.
- Quick incident resolution minimizing customer impact.
- A well-integrated incident management practice that meets DevOps requirements.
- Insights into incidents and their responses, which helps improve process and operational efficiency.
- Automated workflows that eliminate fatigue and human errors and reduce the turnaround time.

Drivers

- **Incident communication and visibility challenges:** With geographically distributed teams, remote workforce, complex on-call schedules and notification channel preferences, incident triage teams often have difficulty engaging responders quickly. Incident communication itself may lack all the relevant inputs or rely on multiple sources of incident data.
- **Automation of incident response processes:** AIR reduces mean time to acknowledge (MTTA) by automating the process of identifying and contacting the relevant domain experts, and speeds up the resolution process.
- **DevOps and site reliability engineer (SRE) requirements:** Traditional incident management models cannot meet the needs of agile cultures because of manual tasks in the incident response workflow. AIR caters to the need for seamless collaboration across various groups enabling DevOps to underpin its offerings with an effective, consistent IT service management (ITSM) practice.
- **Transparent review and analysis:** AIR tools capture an incident's progress from identification through resolution, including the handoffs needed across various teams. This includes the time and action taken at each step of the incident, and provides vital information for postincident review (PIR) and process review for further enhancements.
- **Workflow automation:** These tools can automate workflows that are part of processes like creating incidents for actionable alerts, opening a communications channel in instant messengers for collaboration, updating on a web-portal and one-click remediation for existing runbooks.

Obstacles

- **Overlapping capabilities:** Although AIR solutions offer differentiating features, they also overlap with ITSM and event management systems, making it difficult to articulate the value of investing in AIR.
- **Service definitions:** Service definitions that connect alerts to responder teams are often challenging to configure as it involves interpretation of a problem based on the notification to identify the domain experts that need to be engaged. Service definitions are also a complex part of AIR onboarding.
- **Portability between solutions:** Migrating from one AIR vendor to another is a reset process, with no defined migration path. The integrations, team and service definitions, responder preferences, and role-based access controls must be reconfigured without sophisticated import/export mechanisms.
- **Maturity in I&O:** Few organizations have the required I&O maturity to quantify impact due to time lost in contacting the right personnel for resolving an issue to justify investing in these tools.

User Recommendations

- Invest in a centralized AIR solution for automating incident management workflows and on-call capability for major incidents and critical events with wide integrations for holistic incident response management.
- Integrate monitoring solutions and service desk systems with bidirectional synchronization to incident response systems, which keeps the incident status synchronized across systems.
- Leverage automation for remediation and to extend incident response capabilities that can integrate with DevOps toolchains.
- Improve incident communication and collaboration by integrating incident workflow processes with ChatOps tools, such as Slack or Microsoft Teams.

Sample Vendors

AlertOps; Atlassian; Derdack; Everbridge; OnPage; PagerDuty; ServiceNow; Splunk

Entering the Plateau

APM

Analysis By: Padraig Byrne, Mrudula Bangera

Benefit Rating: High

Market Penetration: More than 50% of target audience

Maturity: Mature mainstream

Definition:

Application performance monitoring (APM) enables the observation of an application's behavior, dependencies, users and business KPIs throughout its life cycle. The application being observed may be developed internally, as a packaged application or as software as a service (SaaS).

Why This Is Important

The APM market continues to evolve beyond its core root of server-side application monitoring as organizations seek to optimize business outcomes, enhance user experience and improve application performance. It is no longer sufficient to monitor one aspect of the technology stack; nor is it enough to deploy proprietary technologies to collect performance data. Modern APM implementations are becoming more tightly integrated with observability platforms.

Business Impact

APM solutions enable businesses to examine modern applications' end-to-end performance, coupled with detailed inspections to quickly identify service-impacting outages. As organizations continue to embrace digital transformation, their need increases for agility in order to succeed with their transformation initiatives. APM solutions can be perceived as more than another monitoring tool, supporting the need for agility and aiding in its acceleration and effectiveness.

Drivers

- **Unified monitoring:** New application monitoring and observability tools are becoming more unified. This approach requires platforms that share common data models to conduct correlation analysis and other critical functions of application performance monitoring.

- **Holistic monitoring:** Modern tools are becoming more holistic in terms of the types of data they can ingest, analyze and integrate. The continued adoption of new application development and operations technologies requires monitoring teams to constantly test the limits of their monitoring products.
- **Integration with DevOps and site reliability engineering (SRE):** Testing in preproduction and integration with continuous integration/continuous delivery (CI/CD) tools have become the new norm, increasing the quality and robustness of the finished product.
- **Intelligent monitoring:** The use of logs, traces, metrics and multiple other types of telemetry is enabling operations and monitoring teams to find unexpected patterns in high-volume, multidimensional datasets using artificial intelligence for IT operations (AIOps) technologies.
- **Business monitoring:** APM tools can be used to derive business metrics, such as abandoned shopping carts or average spend per customer for retailers. Representing such critical business information means an increased likelihood of further investment in these tools.

Obstacles

- Traditional implementations of APM often fail to provide a complete solution, requiring organizations to pivot between tools, wasting time and resources, while struggling to find the root cause of the problem.
- Modern architectures such as containers and microservices and cloud-native environments are coming to IT operations environments faster than monitoring strategies are evolving to handle them. This is leading to visibility gaps and performance challenges.
- Clients increasingly cite cost as a significant challenge for implementation of APM tools. Costs for larger organizations can run into millions per year, representing a significant percentage of IT spend.
- Many IT monitoring teams still rely on manually invoked runbooks or scripts to remediate problems, hindering I&O leaders' ability to deploy and monitor new technologies. There is often a major disconnect between what I&O monitors and what the business cares about, which can have significant negative implications for the business.

User Recommendations

- Choose vendors that assist in relating application performance to business objectives and serve not only IT operations (ITOps), but also DevOps, application owners and lines of business, providing value throughout the application life cycle. Select a vendor that provides actionable answers and not just endless drill-downs to more data.
- Choose products based on their ability to support: mapping and monitoring of customer and business journeys; bidirectional integration with the DevOps toolchain; new emerging standards in instrumentation, such as OpenTelemetry; cloud-native monitoring with an API-first approach; application security; and integrations with your existing or planned IT service management (ITSM) and configuration management database (CMDB) tools.

Sample Vendors

Cisco; Datadog; Dynatrace; Elastic; Grafana; Instana; New Relic; Splunk

Gartner Recommended Reading

Magic Quadrant for Application Performance Monitoring and Observability

Critical Capabilities for Application Performance Monitoring and Observability

Appendixes

See the previous Hype Cycle: Hype Cycle for Monitoring, Observability and Cloud Operations, 2022.

Hype Cycle Phases, Benefit Ratings and Maturity Levels

Table 2: Hype Cycle Phases

(Enlarged table in Appendix)

Phase	Definition
Innovation Trigger	A breakthrough, public demonstration, product launch or other event generates significant media and industry interest.
Peak of Inflated Expectations	During this phase of overenthusiasm and unrealistic projections, a flurry of well-publicized activity by technology leaders results in some successes, but more failures, as the innovation is pushed to its limits. The only enterprises making money are conference organizers and content publishers.
Trough of Disillusionment	Because the innovation does not live up to its overinflated expectations, it rapidly becomes unfashionable. Media interest wanes, except for a few cautionary tales.
Slope of Enlightenment	Focused experimentation and solid hard work by an increasingly diverse range of organizations lead to a true understanding of the innovation's applicability, risks and benefits. Commercial off-the-shelf methodologies and tools ease the development process.
Plateau of Productivity	The real-world benefits of the innovation are demonstrated and accepted. Tools and methodologies are increasingly stable as they enter their second and third generations. Growing numbers of organizations feel comfortable with the reduced level of risk; and the rapid growth phase of adoption begins. Approximately 20% of the technology's target audience has adopted or is adopting the technology as it enters this phase.
Years to Mainstream Adoption	The time required for the innovation to reach the Plateau of Productivity.

Source: Gartner (July 2023)

Table 3: Benefit Ratings

Benefit Rating ↓	*Definition* ↓
Transformational	Enables new ways of doing business across industries that will result in major shifts in industry dynamics.
High	Enables new ways of performing horizontal or vertical processes that will result in significantly increased revenue or cost savings for an enterprise.
Moderate	Provides incremental improvements to established processes that will result in increased revenue or cost savings for an enterprise.
Low	Slightly improves processes (e.g., improved user experience) that will be difficult to translate into increased revenue or cost savings.

Source: Gartner (July 2023)

Table 4: Maturity Levels

(Enlarged table in Appendix)

Maturity Levels	Status	Products/Vendors
Embryonic	In labs	None
Emerging	Commercialization by vendors Pilots and deployments by industry leaders	First generation High price Much customization
Adolescent	Maturing technology capabilities and process understanding Uptake beyond early adopters	Second generation Less customization
Early mainstream	Proven technology Vendors, technology and adoption rapidly evolving	Third generation More out-of-box methodologies
Mature mainstream	Robust technology Not much evolution in vendors or technology	Several dominant vendors
Legacy	Not appropriate for new developments Cost of migration constraints replacement	Maintenance revenue focus
Obsolete	Rarely used	Used/resale market only

Source: Gartner (July 2023)

Evidence

2022 Gartner Signature Infrastructure and Operations Role Survey. This survey was conducted to understand the biggest goals and challenges of I&O leaders for 2022 and 2023. The survey also explored the prioritization of investments for the I&O leaders. The research was conducted online from 6 May through 13 June 2022. In total, 207 respondents were interviewed across Asia/Pacific (n = 73), Western Europe (n = 68) and North America (n = 66). Qualifying organizations operated in multiple industries and reported enterprisewide revenue for fiscal year 2021 of at least $50 million or equivalent. Qualified participants belonged to the functional areas of executive leadership, infrastructure and operations, and enterprise architecture and technology innovation. All respondents led, participated in or had visibility into the strategies and decisions for making investments in infrastructure and operations.

Disclaimer: The results of this survey do not represent global findings or the market as a whole, but reflect the sentiments of the respondents and companies surveyed.

Document Revision History

Hype Cycle for Monitoring, Observability and Cloud Operations, 2022 - 20 July 2022

Hype Cycle for Monitoring, Observability and Cloud Operations, 2021 - 16 July 2021

Hype Cycle for IT Performance Analysis, 2020 - 17 July 2020

Hype Cycle for IT Performance Analysis, 2019 - 30 July 2019

Hype Cycle for IT Performance Analysis, 2018 - 18 July 2018

Hype Cycle for IT Performance Analysis, 2017 - 20 July 2017

Hype Cycle for IT Infrastructure Availability and Performance Management, 2016 - 30 June 2016

Hype Cycle for IT Infrastructure Availability and Performance Management, 2015 - 23 July 2015

Recommended by the Authors

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

Understanding Gartner's Hype Cycles

Tool: Create Your Own Hype Cycle With Gartner's Hype Cycle Builder

Magic Quadrant for Application Performance Monitoring and Observability

Market Guide for AIOps Platforms

Market Guide for Digital Experience Monitoring

Infographic: Artificial Intelligence Use-Case Prism for AIOps

Monitoring and Observability for Modern Infrastructure and Applications

Table 1: Priority Matrix for Monitoring and Observability, 2023

Benefit ↓	*Years to Mainstream Adoption*			
	Less Than 2 Years ↓	***2 - 5 Years*** ↓	***5 - 10 Years*** ↓	***More Than 10 Years*** ↓
Transformational		Observability Service Operations Site Reliability Engineering	Augmented FinOps Autonomic Systems Digital Platform Conductor Tools Low-Code Observability	
High	APM Automated Incident Response UC Monitoring Tools	Autonomous Workload Optimization Data Observability DEX Tools Digital Experience Monitoring Internal Developer Portal Log Monitoring and Analysis OpenTelemetry Performance Engineering	AIOps Platforms Monitoring as Code Observability-Driven Development Transfer Learning	
Moderate		eBPF Native Cloud Monitoring NDR Telemetry Pipelines	Chaos Engineering IT Sustainability Management	Intent-Based Networking
Low		Service Mesh		

Benefit	Years to Mainstream Adoption			
↓	Less Than 2 Years ↓	2 - 5 Years ↓	5 - 10 Years ↓	More Than 10 Years ↓

Source: Gartner (July 2023)

Table 2: Hype Cycle Phases

Phase	*Definition*
Innovation Trigger	A breakthrough, public demonstration, product launch or other event generates significant media and industry interest.
Peak of Inflated Expectations	During this phase of overenthusiasm and unrealistic projections, a flurry of well-publicized activity by technology leaders results in some successes, but more failures, as the innovation is pushed to its limits. The only enterprises making money are conference organizers and content publishers.
Trough of Disillusionment	Because the innovation does not live up to its overinflated expectations, it rapidly becomes unfashionable. Media interest wanes, except for a few cautionary tales.
Slope of Enlightenment	Focused experimentation and solid hard work by an increasingly diverse range of organizations lead to a true understanding of the innovation's applicability, risks and benefits. Commercial off-the-shelf methodologies and tools ease the development process.
Plateau of Productivity	The real-world benefits of the innovation are demonstrated and accepted. Tools and methodologies are increasingly stable as they enter their second and third generations. Growing numbers of organizations feel comfortable with the reduced level of risk; and the rapid growth phase of adoption begins. Approximately 20% of the technology's target audience has adopted or is adopting the technology as it enters this phase.
Years to Mainstream Adoption	The time required for the innovation to reach the Plateau of Productivity.

Phase ↓	Definition ↓

Source: Gartner (July 2023)

Table 3: Benefit Ratings

Benefit Rating ↓	Definition ↓
Transformational	Enables new ways of doing business across industries that will result in major shifts in industry dynamics.
High	Enables new ways of performing horizontal or vertical processes that will result in significantly increased revenue or cost savings for an enterprise.
Moderate	Provides incremental improvements to established processes that will result in increased revenue or cost savings for an enterprise.
Low	Slightly improves processes (e.g., improved user experience) that will be difficult to translate into increased revenue or cost savings.

Source: Gartner (July 2023)

Table 4: Maturity Levels

Maturity Levels ↓	*Status* ↓	*Products/Vendors* ↓
Embryonic	In labs	None
Emerging	Commercialization by vendors Pilots and deployments by industry leaders	First generation High price Much customization
Adolescent	Maturing technology capabilities and process understanding Uptake beyond early adopters	Second generation Less customization
Early mainstream	Proven technology Vendors, technology and adoption rapidly evolving	Third generation More out-of-box methodologies
Mature mainstream	Robust technology Not much evolution in vendors or technology	Several dominant vendors
Legacy	Not appropriate for new developments Cost of migration constraints replacement	Maintenance revenue focus
Obsolete	Rarely used	Used/resale market only

Source: Gartner (July 2023)