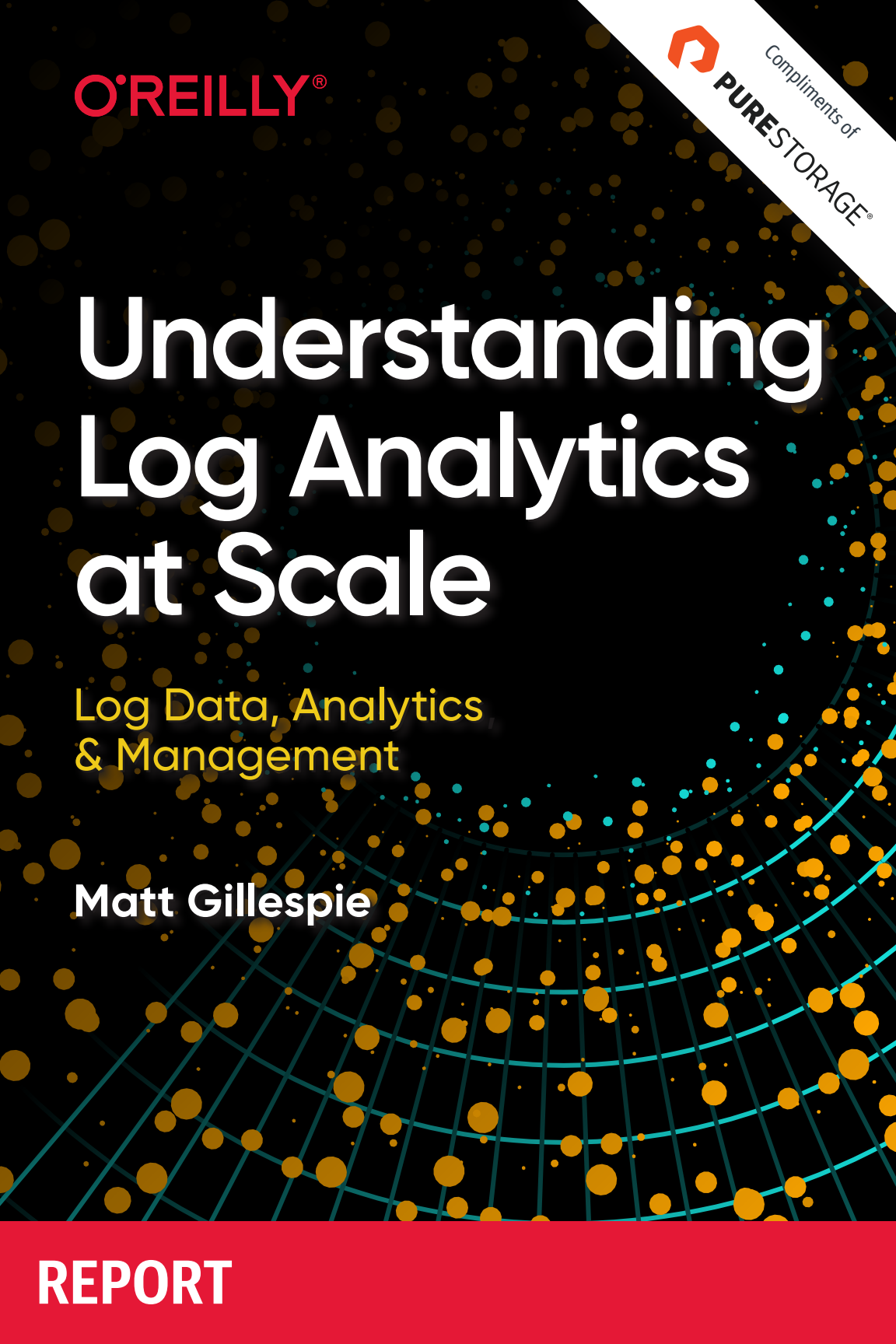
O'REILLY®
Compliments of
PURESTORAGE®
Understanding Log Analytics at Scale
Log Data, Analytics & Management
Matt Gillespie
REPORT

Understanding Log Analytics at Scale

Log Data, Analytics, and Management

Matt Gillespie

Beijing • Boston • Farnham • Sebastopol • Tokyo O'REILLY®

Understanding Log Analytics at Scale
by Matt Gillespie

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://oreilly.com*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Acquisitions Editor: Jessica Haberman
Development Editor: Michele Cronin
Production Editor: Kristen Brown
Copyeditor: Octal Publishing, LLC.
Interior Designer: David Futato
Cover Designer: Karen Montgomery
Illustrator: Rebecca Demarest

January 2020: First Edition

Revision History for the First Edition
2020-01-23: First Release

978-1-492-07622-3

[LSI]

Table of Contents

Understanding Log Analytics at Scale

The humble machine log has been with us for many technology generations. The data that makes up these logs is a collection of records generated by hardware and software—including mobile devices, laptop and desktop PCs, servers, operating systems, applications, and more—that document nearly everything that happens in a computing environment. With the constantly accelerating pace of business, these logs are gaining in importance as a contributor to practices that help keep applications running 24/7/365 as well as analyzing issues faster to bring them back online when outages do occur.

If logging is enabled on a piece of hardware or software, almost every system process, event, or message can be captured as a time-series element of log data. Log analytics is the process of gathering, correlating, and analyzing that information in a central location to develop a sophisticated understanding of what is occurring in a datacenter and, by extension, providing insights about the business as a whole.

The comprehensive view of operations provided by log analytics can help administrators investigate the root cause of problems and identify opportunities for improvement. With the greater volume of that data and novel technology to derive value from it, logs have taken on new value in the enterprise. Beyond long-standing uses for log data, such as troubleshooting systems functions, sophisticated log analytics has become an engine for business insight as well as compliance with regulatory requirements and internal policies, such as the following:

- A retail operations manager looks at customer interactions with the ecommerce platform to discover potential optimizations that can influence buying behavior. Complex relationships among visit duration, time of day, product recommendations, and promotions reveal insights that help reduce cart abandonment rates, improving revenue.
- A ride-sharing company collects position data on both drivers and riders, directing them together efficiently in real time as well as performing long-term analysis to optimize where to position drivers at particular times. Analytics insights enable pricing changes and marketing promotions that increase ridership and market share.
- A smart factory monitors production lines with sensors and instrumentation that provide a wealth of information to help maximize the value generated by expensive capital equipment. Applying analytics to log data generated by the machinery increases production by tuning operations, identifying potential issues, and preventing outages.

Using log analytics to generate insight and value is challenging. The volume of log data generated all over an enterprise is staggeringly large, and the relationships among individual pieces of log data are complex. Organizations are challenged with managing log data at scale and making it available where and when it is needed for log analytics, which requires high compute and storage performance.

NOTE

Log analytics is maturing in tandem with the global explosion of data more generally. International Data Corporation (IDC) predicts that the global datasphere will grow more than fivefold in seven years, from 33 zettabytes in 2018 to 175 zettabytes in 2025.[1] (A zettabyte is 1021 bytes or a million petabytes.)

What's more, the overwhelming majority of log data offers little value and simply records mundane details of routine day-to-day operations such as machine processes, data movement, and user

1 David Reinsel, John Gantz, and John Rydning. IDC, November 2018. "The Digitization of the World From Edge to Core." *https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf*.

transactions. There is no simple way of determining what is important or unimportant when the logs are first collected, and conventional data analytics are ill suited to handle the variety, velocity, and volume of log data.

This report examines emerging opportunities for deriving value from log data, as well as the associated challenges and some approaches for meeting those challenges. It investigates the mechanics of log analytics and places them in the context of specific use cases, before turning to the tools that enable organizations to fulfill those use cases. The report next outlines key architectural considerations for data storage to support the demands of log analytics. It concludes with guidance for architects to consider when planning and designing their own solutions to drive the full value out of log data, culminating in best practices associated with nine key questions:

- What are the trends for ingest rates?
- How long does log data need to be retained?
- How will regulatory issues affect log analytics?
- What data sources and formats are involved?
- What role will changing business realities have?
- What are the ongoing query requirements?
- How are data-management challenges addressed?
- How are data transformations handled?
- What about data protection and high availability?

Capturing the Potential of Log Data

At its core, log analytics is the process of taking the logs generated from all over the enterprise—servers, operating systems, applications, and many others—and deducing insights from them that power business decision making. That requires a broad and coherent system of telemetry, which is the process of PCs, servers, and other endpoints capturing relevant data points and transmitting them to a central location.

Log analytics begins with collecting, unifying, and preparing log data from throughout the enterprise. Indexing, scrubbing, and normalizing datasets all play a role, and all of those tasks must be completed at high speed and efficiency, often to support real-time analysis. This entire life cycle and the systems that perform it must be designed to be scalable, flexible, and secure in the face of requirements that will continue to evolve in the future.

Generating insights consists of searching for specific pieces of data and analyzing them together against historical data as well as expected values. The log analytics apparatus must be capable of detecting various types of high-level insights such as anomalies, relationships, and trends among the log data generated by information technology (IT) systems and technology infrastructure, as shown in Figure 1.

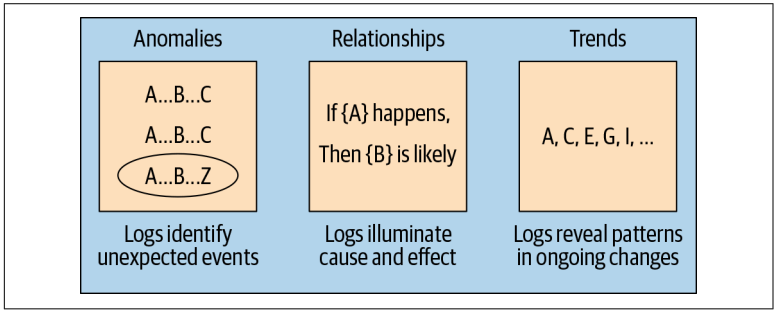

Figure 1. High-level types of insights discoverable from log data

Following are some examples of these types of high-level insights:

Anomaly
: Historically, 90% of the traffic to a given server has come from HR. There is now an influx of traffic from a member of the sales department. The security team might need to investigate the possibility of an insider threat.

Relationship
: The type of spike currently observed in traffic to a self-serve support portal from a specific customer often precedes losing that customer to the competition. The post-sales support team might need to ensure that the customer isn't at risk.

Trend
: Shopping cart abandonment rates are increasing on the ecommerce site for a specific product type. The sales operations team might need to investigate technical or marketing shortcomings that could be suppressing that product's sales.

In addition to detecting these high-level insights, the log analytics apparatus must be capable of effective reporting on and visualization of those findings to make them actionable by human administrators.

Your Environment Has Too Many Log Sources to Count

Log data is generated from many sources all over the enterprise, and deciding which ones to use for analytics is an ongoing process that can never be completed. The following list is representative, as opposed to exhaustive:

Servers
: Operating systems, authentication platforms, applications, databases

Network infrastructure
: Routers, switches, wireless access points

Security components
: Firewalls, intrusion prevention systems, management tools

Virtualization environments
: Hypervisors, orchestration engines, management utilities

Data storage
: Local, virtualized, Storage Area Network (SAN), and/or Network-Attached Storage (NAS) resources

Client machines
: Usage patterns, data movement, resource accesses

Although they derive from a shared central concept, implementations of log analytics are highly variable in scope, intent, and requirements. They can run the gamut from modest to massive in scale, with individual log entries that might be sparse or verbose, carrying all manner of information in an open-ended variety of formats that might not be readily compatible, as shown in Figure 2. All share the challenge of tightening the feedback loop between sifting

through and interpreting enormous numbers of events, often in real time, to generate insights that can optimize processes.

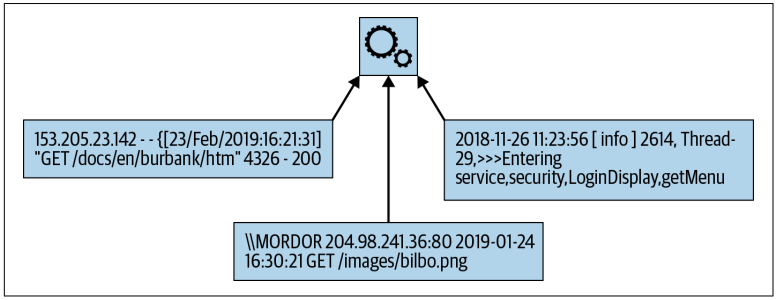

Figure 2. Challenge of bringing together nonstandardized log file formats

Storing log data enables analysts to go back through the repository of time-series data and re-create a series of events, correlating causes and effects after the fact. In addition to casting light on the past, identifying historical patterns also helps illuminate present and future dangers and opportunities. The sheer volume of that data and the need to be able to effectively query against it places significant demands on storage systems.

Treating Logs as Data Sources

The contents of logs are less a series of metrics than they are text strings akin to natural language, in the sense that they are formatted imprecisely, with tremendous variation depending on who created the log-writing mechanism. In addition, because log entries are only semi-structured, they must be interpreted and then parsed into discrete data points before being written to a database.

Telemetry from thousands of different sources might be involved, from simple sensors to enterprise databases. In keeping with that enormous diversity, the structure, contents, and syntax of entries vary dramatically. Beyond differences in format and syntax, various logs contain discrete datasets, with mismatched types of data. Transforming and normalizing this data is key to making it valuable.

Analytics can be performed on log data that is either streaming or at rest. Real-time or near-real-time analysis of logs as they are generated can monitor operations and reveal emerging or existing problems. Analysis of historical data can identify trends in quantities

such as hardware utilization and network throughput, providing technology insights that complement business insights more broadly and help guide infrastructure development. Using older data to create baselines for the future also helps to identify cases for which those ranges have been exceeded.

Logs Versus Metrics

Both logs and metrics are essentially status messages, which can come from the same source. They are complementary but distinct, as represented in Figure 3.

Logs	Metrics
Less structured	More structured
Verbose descriptions	Quantitative data points
Triggered by events	Collected at regular time intervals
Best for root-cause analysis	Best for direct numeric analysis

Figure 3. Comparison between logs and metrics

Logs are semi-structured, defined according to the preferences of the individual developers that created them. They are verbose by nature, most often based on free text, often resembling the natural language from which they derive. They are intended to give detail about a specific event, which can be useful in drill-down root-cause analysis of scenarios such as system failures or security incidents.

Metrics are quantitative assessments of specific variables, typically gathered at specific time intervals, unlike logs, which are triggered by external events. Metrics have a more structured format than logs, making them suitable for direct numerical analysis and visualization. Because their collection is governed by time rather than events, volumes of metrics data tend to scale more gradually than logs with increased IT complexity and transaction volume.

Of the two, logs are far messier. Although they are semi-structured, recovering that structure requires parsing with specialized tools. Metrics, by contrast, are inherently highly structured. Logs and metrics can work together, with different functions that reflect their respective structures.

For example, metrics reveal trends through repeated measurement of the same quantities over time. Referred to as the aforementioned "time series," this sequence of data points can be plotted as a line

graph, for example, where increases in query response time might indicate deteriorating performance of a database. The greater level of forensic detail in the corresponding logs can be the key to determining why that deterioration occurred.

Log data is messy, both in the organizational sense of mismatched formats of logs from various sources, as well as in the hygienic sense of misspellings, missing data, and so on. Beyond the need to interpret the structures of entries and then parse them, the transformations applied to log data must also account for quality issues within the data itself. For example, log analytics systems typically provide the ability to interpret data so that they can successfully query against data points that might include synonyms, misspellings, and other irregularities.

Aside from quality issues, data logs can contain mismatches simply because of the way they characterize data, such as one security system tagging an event as "warning" while another tags the same event as "critical." Such discrepancies among log entries must be resolved as part of the process of preparing data for analysis.

The need to collect log data from legacy systems can also be challenging. Whereas legacy applications, operating systems, and hardware are frequent culprits in operational issues, they can provide less robust (or otherwise different) logging than their more modern counterparts. Additional layers of data transformation might be required by such cases in order to normalize their log data to that of the rest of the environment and provide a holistic basis for log analytics.

The Log Analytics Pipeline

As log data continues to grow in volume, variety, and velocity, the associated challenges require structured approaches to infrastructure design and operations. Log analytics and storage mechanisms for machine data based on tools such as Splunk and the Elastic Stack must be optimized across a life cycle of requirements, as illustrated in Figure 4. To perform these functions effectively, it must be possible to draw data from anywhere in the environment, without being impinged by data silos.

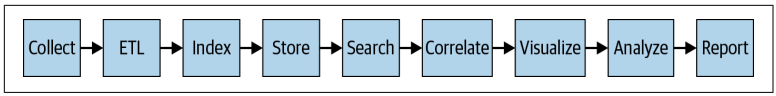

Figure 4. Pipeline for assembling and driving value from log data

This pipeline represents the process for transforming log data into actionable insights, although in practice, their order might be rearranged or only a subset of the steps listed here might be used. Common operations performed on log data include the following:

Collect
: From its dispersed sources, log data must be aggregated, parsed, and scrubbed, such as inserting defaults for missing values and discarding irrelevant entries.

ETL (Extract, Transform, Load)
: Data preparation can include being cleaned of bad entries, reformatted, normalized, and enriched with elements of other datasets.

Index
: To accelerate queries, the value of indexing all or a portion of the log data must be balanced against the compute overhead required to do so (as discussed below).

Store
: Potentially massive sets of log data must be stored efficiently, using infrastructure built to deliver performance that scales out smoothly and cost effectively.

Search
: The large scale of the log data in a typical implementation places extreme demands on the ability to perform flexible, fast, sophisticated queries against it.

Correlate
: The relationships among various data sources must be identified and correlated before the significance of the underlying data points can be uncovered.

Visualize
: Treating log entries as data means that they can be represented visually using graphs, dashboards, and other means to assist humans in understanding them.

Analyze
: Slicing and dicing log data and applying algorithms to it in a structured, automated way enables you to identify trends, patterns, and actionable insights.

Report
: Both predefined and ad hoc reporting must be powerful and flexible so that users can rapidly produce outputs that illuminate their business needs.

NOTE

The framework of steps given here is a guideline that could easily be expanded to more specifically call out actions such as data parsing, transformation, and interpretation, among many others. The point of this life cycle description is to provide a workable overall view rather than the most exhaustive one possible.

Getting your arms fully around the challenges associated with implementing log analytics is daunting. The potential sources and types of log data available are of open-ended variety, as are the possible uses of that data. Although the specific implementations at every organization will be different, they share general technical requirements as well as the potential to be applied to common business needs and use cases.

Log Analytics Use Cases

Technologists have been analyzing machine logs for decades, from the earliest days of tuning or troubleshooting their environments. Over time, the industry has found ways to increasingly automate that analysis, leading to the emergence of log analytics as we know it today. Now more than ever, log analytics can help businesses run more efficiently, reduce risk, and ensure continuity of operations.

The use cases described in this section illustrate some examples of how log analytics has taken on new importance in the past several years, demonstrating how it can deliver unprecedented value to organizations of all types and sizes. Factors that have contributed to that growing importance include the following:

- *Data growth* provides greater opportunities for log analytics as well as challenges. The scale of data analysis will grow further as we continue to drive intelligence into the world around us. A

single self-driving car is estimated to generate multiple terabytes of data each day, while a smart factory might generate a petabyte per day.[2]

- *Greater variety among types of endpoints* has already reached unprecedented levels as the IT environment has become more complex. As the pace of change accelerates and the Internet of Things (IoT) adds billions of new devices online, the insights to be gained by bringing together multiple data sources will continue to increase.
- *Technology evolution*, making log analytics feasible at greater scale than before. In particular, the mainstream emergence of flash storage offers faster read/write speed than conventional spinning hard disk drives (HDDs), and low-cost compute capacity offers high performance with commodity servers.

With the increased scope and prevalence of log analytics as a whole, a growing set of common use cases have emerged. The remainder of this section discusses several prevalent ones, grouped here under the categories of cybersecurity, IT operations, and industrial automation. While many other use cases are possible and indeed prevalent, these provide a representative sample.

Cybersecurity

Securing IT and other systems is a classic application of log analytics based on the massive numbers of events that are logged and transmitted throughout a large organization. Cyber protection in this area draws from log data and alerts from security components such as firewalls and intrusion detection systems, general elements of the environment such as servers and applications, and activities such as user login attempts and data movement. Log analytics can play a role in multiple stages of the security life cycle:

Proactively identifying and characterizing threats
: Log analytics can iteratively search through log data to detect unknown threats that conventional security systems are not designed to identify, creating testable hypotheses.

2 Richard Friedman. Inside HPC, May 31, 2019. "Converging Workflows Pushing Converged Software onto HPC Platforms." *https://insidehpc.com/2019/05/workflows-converged-software-hpc/*.

Detecting and responding to attacks and other security events
: When abnormal indicators arise, log analytics can help to identify the nature and scope of a potential breach, minimize exposure, and then neutralize and recover from the attack.

Performing forensic analysis after a breach has occurred
: A robust log analytics platform helps identify the point-in-time log information that should be brought into a post-mortem investigation as well as making that data available and acting on it.

Detecting anomalies

Cyber processes often use analytics to define a typical working state for an organization, expressed as ranges of values or other indicators in log data, and then monitor activity to detect anomalies. For example, an unusual series of unsuccessful authentication attempts might suggest attempted illicit access to resources. Unusual movement of data could indicate an exfiltration attempt.

The sheer volume of log data makes it untenable for humans to interpret it unaided.

With thousands of events per minute being documented by hardware and software systems all over the computing environment, it can be difficult or impossible to determine what is worthy of attention. Machine learning models can help analytics engines cull through these huge amounts of log data, detecting patterns that would not be apparent to human operators.

Those processes can occur automatically, or they can be initiated by ad hoc queries by analysts or others. Their outputs can be used to identify items of interest for human analysts to investigate further, allowing them to focus their attention where it is the most valuable. A common application is that threat hunters often use log analytics to help identify potential threats, look more deeply into them, and determine what response, if any, is required.

AI Is Invaluable to Anomaly Detection

The twin limiting factors in detecting anomalies in log data for security usages are massive data volumes and the necessity of looking for undefined patterns. The data itself is messy, consisting of many different formats and potentially containing misspellings,

inconsistencies, and gaps. The anomalous patterns being looked for can be subtle and easy to overlook.

All of this makes humans poorly suited to anomaly detection at scale. Sustained massive levels of event volumes quickly become overwhelming, and a significant proportion of those events are irrelevant. At the same time, software tools might also not be successful, given that the effectiveness of its detection is limited by the accuracy of its assumptions, which are likely to be predetermined and static. Over the past 5 to 10 years, the industry has developed sophisticated dashboards to provide real-time views that help identify potential security incidents.

Machine learning and artificial intelligence (AI) are increasingly viable for improving those human monitoring approaches, overcoming some key limitations and turning massive data stores from a liability into an asset for helping to train AI models. Algorithms can use both historical and real-time data to continually update their vision of what "business as usual" looks like and use that moving baseline as the standard against which they interpret emerging log events.

In recent years, predictive analytics have become more common in a variety of usages. Based on all data received up to the current moment, a machine learning model can predict expected parameters of future events and then flag data that falls outside those ranges.

Working from that set of detected anomalies, the algorithm can correlate them with other incidents to limit the universe of events to be considered and to illuminate patterns in real time. By alerting security analysts to those anomalies and patterns, the system can pare the scope of alerts that must be investigated by human operators down to a manageable level. As a result, IT staff can focus on innovation and adding value to the organization rather than just maintaining the status quo.

Identifying and defeating advanced threats

One of the issues confronted by modern security teams is the subtlety and long time horizons associated with today's stealthy attacks. Advanced persistent threats operate by moving laterally as quietly as possible through an organization to gain access to additional resources and data, in a process that often elapses over a matter of months.

The key to detection often lies less in identifying any specific event than in overall patterns.

In practice, a security analyst might begin with a suspicious activity such as a service running from an unexpected file location and then use various log data to uncover additional information surrounding that event to help discover whether it is malicious or benign. For example, other activities during the same login session, connections from unexpected remote IP addresses, or unusual patterns of data movement captured in logs can be relevant.

Treating log data as a coherent whole rather than natively as a disparate collection of data points enables analysts to examine activities anywhere across the entire technology stack. This approach also enables analysts to traverse events backward and forward through time to retrace and analyze the behaviors of a given application, device, or user. This capability can be vital in cases such as understanding the behaviors of persistent cyber threats that operate over the course of weeks or months.

Data context consists of information about each data point's connections to others, which must be encoded along with the data itself, typically in the form of metadata created to describe the main data. This context enables analysis to identify the significance of a given data point in relation to the greater whole.

Statistical analysis against bodies of machine log data can reveal relationships that would otherwise remain hidden. Those insights help security teams more confidently categorize events in terms of the levels of threat they represent, enabling faster, more precise responses that help limit negative impacts on operations, assets, and reputations. In the context of a smart factory, for example, that analysis can help avoid unplanned outages that would otherwise lead to lost productivity and profitability.

First, Prepare the Organization

Because sources of log data often cross organizational boundaries, even within the same company, setting the foundations for a log analytics practice involves more than technology. For example, a single transaction might involve technology components that are managed and controlled by many different individuals and departments, as illustrated in Figure 5.

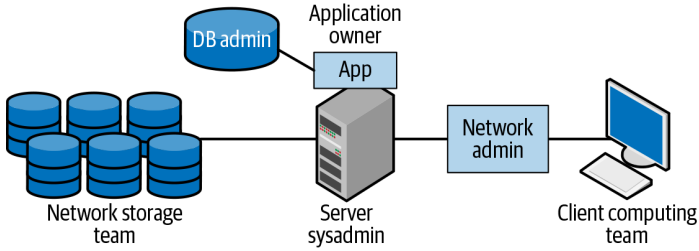

Figure 5. Resources involved in a single transaction, managed by separate teams

The expertise of people who know the systems best can be invaluable in determining what logs are available and how to get data from them for use in analytics. Cooperation with all of these entities is essential at all points along the transaction chain. Administrators of everything from enterprise databases to network hardware will need to enable logging and provide access to the log files generated.

This reality makes the buy-in and support of senior management essential. Involving them early in the decision-making process is sound advice, and it illustrates the value of presenting them with use cases that support their business interests. In particular, the cyber security potential of log analytics is a strong inducement to cooperate that crosses organizational boundaries.

IT Operations

Even though IT operations has always been challenging, today's business and technology climate makes it more so than ever. Very high standards for the quality of end-user experience have become the norm; meanwhile, the pace of change has accelerated and the ability to turn on a dime is taken for granted. At the same time, many of these organizations are being forced to do more with less in the face of budgets that might be flat or even decreasing.

The technology environment itself has also become more complex and varied. In place of traditional client-server computing stacks that were relatively uniform and static, dynamic, heterogeneous infrastructures change in real-time cadence with varying workloads. A growing proportion of the network is defined in software, creating new management challenges, and software paradigms such as microservices and containers challenge basic assumptions about

enterprise applications. And all of this unfolds against the backdrop of unprecedented data volumes.

Log analytics can help address the need for IT operations to be highly responsive to each diverse part of the enterprise, maintaining smooth operations and avoiding downtime. Speed is of the utmost importance, with operations teams finding and remediating as many issues as possible automatically without human intervention, and addressing others quickly. This capability is a prerequisite for meeting Service-Level Agreements (SLAs), delivering a good end-user experience, and maintaining uninterrupted, highly responsive access to resources.

Enabling logs in infrastructure and applications enables visibility into factors that reveal insights about performance and availability. Both machine-to-machine and machine-to-human modalities can make use of analysis based on that log data to identify potential issues before they arise and to tune the environment on an ongoing basis, to deliver the best results possible in the face of changing business requirements. For troubleshooting, the log analytics stack helps accelerate root-cause analysis by bringing together time-series data from all over the environment.

Infrastructure monitoring and troubleshooting

As IT infrastructure becomes more complex, the challenges with providing uninterrupted, smooth operation and an excellent end-user experience become more acute. In particular, a single operation can involve a large number of different systems, which might involve a combination of resource types, as shown in Figure 6, such as far-flung collections of sensors, on-premises systems, public cloud, and Software as a Service (SaaS). Platforms can vary dramatically, services and applications might be controlled by different parts of the organization, and the information available from each can be inconsistent.

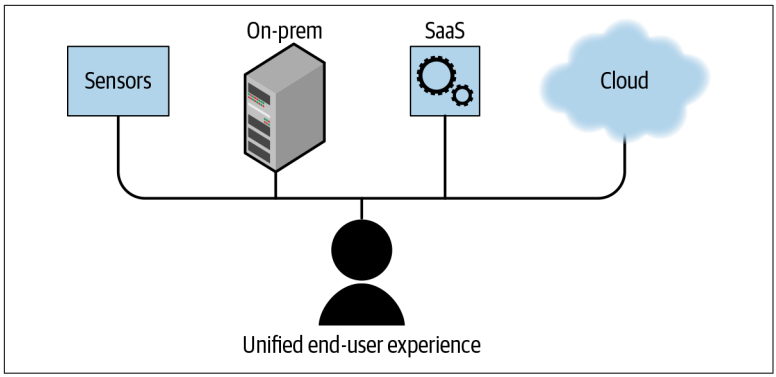

Figure 6. Diverse resource types contributing to a single end-user experience

Traditionally, systems were monitored manually and independently, calling on the familiar image of an administrator scanning multiple displays arrayed at a monitoring station. Unfortunately, this approach of eyeing alerts as the basis for keeping systems in good working order becomes less viable as the environment becomes larger and more complex. Both the large numbers of systems to keep watch over and the multidimensional interactions among them defy the abilities of human operators. What's more, the dispersed nature of the log data being generated under this model makes it difficult to discern relationships among them.

For example, when doing root-cause analysis of a performance issue with a database application, there are many places to look at once. The application itself can be a contributor, as can the database platform, the server hardware that both run on, and the availability of network resources. There might be contributing factors associated with all of these, or even another external entity that might not be immediately apparent, such as a single sign-on (SSO) mechanism, firewall, or DNS server.

Aggregating all of that data together provides the basis for more efficient and sophisticated analysis. Having access to a composite picture of all the factors potentially contributing to the issue lets administrators do troubleshooting with a holistic approach, rather than having to look at various resources in a piecemeal fashion. For example, staff are able to look at all of the events that occurred on the entire body of related systems at a specific point in time,

correlating them to discover the issue—or combination of issues—behind the problem.

Software development, optimization, and debugging

Applying log analytics within software-development organizations arms developers with information about how their code behaves and interacts with the rest of the environment. This insight can help them optimize the quality of their software while also letting them act more deliberately, breaking the cycle of running from fire to fire. In addition to enabling the development organization to be proactive rather than reactive, the organization as a whole is saved from the impacts of avoidable issues in software.

The success of a development organization is directly tied to how well its software functions in a variety of situations, including edge cases and unforeseen circumstances. Identifying potential issues before they affect the production environment is a critical capability. For example, a minor inefficiency in an application's operation could become an unacceptable bottleneck as usage, data volumes, and integration with other systems grow. An intermittent delay in one process can affect other processes that are dependent on it over time, and the cascade effect can eventually become untenable.

Log files can provide early indications of emerging issues, long before they would normally become apparent to users or administrators. For example, a gradual trend toward a database server taking a progressively longer time opening a set of database records can indicate growing response-time issues. Log analytics can help detect this trend and then determine its root cause, whether it is a simple capacity issue or a need for performance tuning in application code. Anticipating the usability issue before it affects users allows for the necessary steps to be taken in a timelier fashion and business impacts to be avoided.

Apart from avoiding exceptions, log analytics can also aid in capacity planning by helping predict how a system will continue to perform as it scales. By tracking the time needed to perform a given operation as the number of database records increases, it's possible to estimate how those trends will continue in the future. That information can give you a better idea of the viability of a specific piece of code going forward, which can feed strategy about when a new approach to achieving the desired result might be needed.

In the context of a DevOps practice, log analytics can help teams ensure compatibility of their code with complex distributed systems. Applied in the early stages of your Continuous Integration/Continuous Delivery (CI/CD) pipeline, log analytics can show how the code interacts with the rest of the environment before it is released to production. By anticipating issues at this stage, we can often fix them with less effort and expense than if those problems didn't emerge until post-deployment.

Application performance monitoring

The purpose of Application Performance Management (APM) is to maintain a high standard of user or customer experience by measuring and evaluating the performance of applications across a range of domains. Common capabilities include analysis of transaction throughput and response time, establishing baseline levels of those quantities, and monitoring to generate alerts when performance strays outside set limits. Real-time data visualizations are typically employed to help conceptualize the significance of emerging events, as an aid to identifying potential problems and their root causes (hopefully before they occur).

Log analytics can also play a key role in the related field of A/B testing, for which log data related to usage is collected separately for two versions of an application. The two sets of log data are then compared side by side to identify how the changes made between the two versions of the application affect factors such as the user experience (UX).

Exerting control and visibility across applications has become more complex in recent years as modern applications have transmuted from monolithic stature to distributed collections of microservices, containers, and other components loosely coupled together using a variety of application programming interfaces (APIs). In addition to forming a complex structure for the application, these components can be hosted using a combination of on-premises resources and multiple cloud environments.

Together, these factors make the task of integrating and consolidating log data to track application performance more challenging than in the past. Accordingly, APM approaches have shifted to meet the needs of modern applications and development practices. Because APIs are central to the architectures of distributed applications,

monitoring and managing API performance is critical to the success of APM as a whole. Likewise, the growing prominence of containers, especially for cloud-distributed applications, makes monitoring container performance an important consideration, as well.

APM practices should allow for robust prioritization of performance problems for resolution, identifying the most urgent ones for triage. That capability is often assisted by machine learning algorithms that are trained to recognize the signs of common performance issues. At the same time that triage is necessary, the log analytics toolset must also provide precise insights that enable deeper root-cause analysis. This capability is essential to avoid wasting time and effort to solve a secondary or ancillary issue without addressing the underlying root cause.

Industrial Automation

In many cases, log analytics for industrial automation begins with adding connectivity to rich data collection mechanisms that already exist on industrial systems. The computing and control apparatus often captures detailed log data, stores it for a specified period of time, and then discards it. Technical staff can manually access that data in response to outages or performance problems, although in many organizations, there is no day-to-day usage of that log data beyond watching for exceptions or other issues.

Enabling Industry 4.0

The essence of the fourth industrial revolution (Industry 4.0[3]) is taking full advantage of the connected computer systems that underlie industrial processes. Data exchange among different parts of the environment, including cyber-physical systems, is a key aspect of that development. The potential for using log data to enable Industry 4.0 depends on integrating data from both IT and operational technology (OT) systems, generating insight from it, and applying that insight to optimize efficiency, profitability, and competitiveness.

3 Tom Merritt. Tech Republic, September 3, 2019. "Top Five Things to Know about Industry 4.0." *https://www.techrepublic.com/article/top-5-things-to-know-about-industry-4-0/*.

For example, analyzing data gathered from automated manufacturing equipment can enable sophisticated systems for preventive maintenance. Such measures monitor operating logs and metrics from industrial mechanisms in production and optimize maintenance to avoid unplanned outages and maximize the working lifespan of capital equipment.

The mechanisms to facilitate those processes are sometimes known as the physical-to-digital-to-physical (PDP) loop,[4] which is illustrated in Figure 7. In the physical-to-digital stage of the PDP loop, log data is captured from cyber-physical systems to create a record of operational details. In the digital-to-digital stage, that data is gathered centrally so that analytics and visualizations can be applied to it, generating insights about how operation of the physical systems can be optimized or enhanced. The digital-to-physical stage provides the novel aspect of this process that distinguishes Industry 4.0, namely to provide a feedback loop back to the cyber-physical systems, which can then act on those insights.

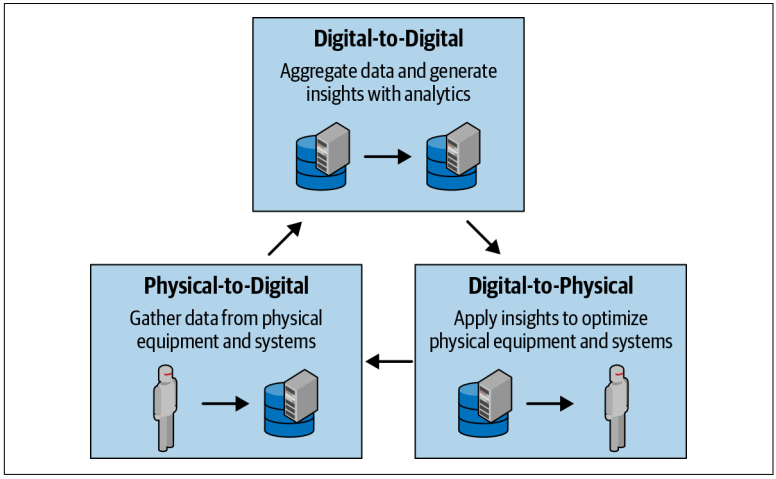

Figure 7. The Industry 4.0 physical-to-digital-to-physical loop

4 Mark Cotteleer and Brenna Sniderman. Deloitte Insights, December 18, 2017. "Forces of change: Industry 4.0." *https://www2.deloitte.com/us/en/insights/focus/industry-4-0/overview.html*.

Operating in real time, the information flowing through the PDP loop enables industrial automation equipment to be continually self-tuning. The resulting efficiency improvements help to ensure that companies can extract the maximum value out of their capital investments.

In place of the physical equipment in the preceding description of the PDP loop, we can modify the model to use a digital twin, which is a sensor-enabled digital replica of an industrial cyber-physical system. The twin is a dynamic digital doppelganger that is continuously updated by means of log data that is collected from the physical device so that it accurately represents the system in real time. We can use the simulation to observe and investigate the operation of equipment under different conditions and analyze them together with historical data to predict maintenance requirements in advance.

Industrial Internet of Things

Instrumentation and telemetry are nothing new in industrial applications, for which data has been collected for decades from sensors on equipment that ranges from oil rigs to assembly lines to jet engines. It is common for millions or even billions of dollars' worth of equipment to be in use within a large industrial operation. That high value of capital equipment and its importance to profitability has led to increasingly richer and more plentiful information being provided by these telemetry systems.

The motivation for collecting all of this information is both to make the equipment operate as efficiently as possible—generating more profit more quickly—and to extend the working life of the equipment itself, maximizing return on investment. Improvement in the tools and other technologies that support log analytics has driven the ability to get more granular telemetry, to aggregate and analyze it more effectively, and to more directly implement the outcome. The goal, then, is to tighten the feedback loop between microevents that occur in the environment, looking at millions of such events to deduce significance, and to use that insight to optimize some process.

Part of the challenge of log analytics in an Industrial Internet of Things (IIoT) context is the enormous variety of data. The rapidly evolving nature of this field means that many new players are

emerging, and it might be some time before standard data formats emerge. It is not uncommon to have thousands of different types of sensors, all with different data formats and different ways of communicating information. All of that information needs to be brought together and normalized in such a way that either a human or an algorithm can make sense of it to reveal the information buried inside.

It is common in IIoT deployments for large numbers of sensors and other data sources to be located far from the network core. In such cases, it is often not practical or desirable to transfer the full volume of data generated over a wide-area connection.

In response, "edge analytics"—the practice of performing analytics close to the data source—is becoming increasingly prevalent and can have multiple advantages. In autonomous vehicles and other real-time usages, for example, the results of an algorithm being applied to log data are required as near to instantaneously as possible. Long-range transfer of data is also incompatible with latency-sensitive usages, such as real-time control of manufacturing-line equipment.

Performing edge analytics on log data helps support the ability to analyze that data in a variety of ways for different business needs. For example, monitoring for safety issues or controlling machine settings in real time might be performed at the edge, whereas analysis of operational data across the enterprise might be better suited to a centralized analytics apparatus.

Similarly, this combination of approaches allows for both real-time inquiry to discover and address problems as they happen as well as longitudinal studies, including the addition of historical information, to discover long-term trends and issues. That work depends on having a centrally available repository of historic data. On their way to that central store, log data streams might pass through complex, multistage pipelines.

The data and various pieces of metadata typically come from sensors to a staging point where the data is collected and transformed in various ways so that it can be more readily stored in a database and compared against other data that initially might have had an incompatible structure and format. For example, this staging point could be a central information hub on an oil field where data from dozens of wellheads is collected.

From there, the data can be aggregated at a regional hub that incorporates all of the data from several oil fields, performing further transformations on the data and combining it into an aggregate stream that is sent to a larger collection point, and so on, all the way to the network core (which increasingly includes private, hybrid, or public cloud resources).

Sending the data to a remote location to perform calculations on it may introduce unacceptable transport latency, even on high-speed connections. Moreover, the need to send a massive stream of raw data as free text can be prohibitive in terms of bandwidth requirements, particularly as the volumes of log data being collected continue to increase.

Predictive maintenance

Fine-tuning maintenance schedules for large-scale industrial equipment is critical to getting the full value out of the capital they represent. Taking a machine out of service for maintenance too soon cuts into efficiency by creating unneeded planned downtime; whereas, stretching out the time between maintenance intervals carries the risk of unplanned downtime and interrupted productivity.

By increasing the amount of instrumentation built in to industrial equipment, a richer set of data is generated, supporting sophisticated log analytics that provide insights about optimal maintenance timing. Rather than a simple break-fix approach that addresses problems on an exception basis, or even a regular schedule designed to prevent unplanned outages, predictive maintenance can respond to actual conditions and indicators, for greater accuracy.

Machine learning models can help transform telemetry into insight, helping predict the most cost-effective approaches to physical maintenance, responding to the needs of an individual piece of equipment rather than averages. In fact, this approach to predictive maintenance has implications far beyond the industrial context, a few examples of which include the following:

- Vehicle fleets and their replaceable components such as filters and lubricants
- IT physical infrastructure, including components of servers, storage, and network devices

- Field equipment needs that range from leaking pipelines to vending machine replenishment

In any of these spheres of operation, among many others, applying log analytics to predictive maintenance can optimize operational expenses by increasing efficiency and productivity while offering capital expense advantages, as well, in the form of longer equipment life cycles.

When building out storage infrastructure, you should make scalability a primary design requirement. In addition to storage capacity, you need to recognize the importance of growing performance requirements as automated and ad hoc query volumes increase over time with new business needs.

Tools for Log Analytics

Tasks along the log analytics pipeline ingest disparate log data, transform it to a more usable state, draw insights from it, and output those insights either as machine-to-machine communications or in human-consumable forms such as visualizations and reports. A large number of toolsets are available to perform these functions, both proprietary and open source. The largest market shares among these for log analytics are held by Splunk, the Elastic Stack, and Sumo Logic, some characteristics of which are summarized in Table 1.

Table 1. Vital statistics for a few popular log analytics tools

	Splunk	Elastic Stack	Sumo Logic
Open source/proprietary	Proprietary	Open source	Proprietary
SaaS option	Yes	Yes	Yes
On-premises option	Yes	Yes	No

All of these toolsets utilize compute and storage resources to perform search, analysis, and visualization that are suited to the needs of log analytics, as illustrated in Figure 8. These solutions place a premium on the ability to ingest data directly from virtually any source, provide high-throughput flexible analytics on it, and scale as data volumes grow, in terms of both capacity and performance to drive increased query complexity and volume.

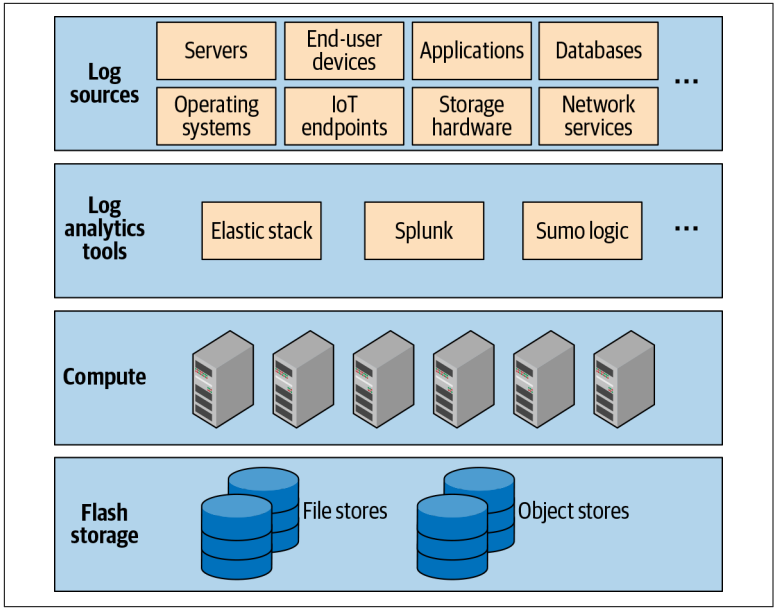

Figure 8. Placement of log analytics tools within the broader solution stack

Even though each individual implementation will have its own unique requirements for compute resources, they might have characteristics in common. For example, because log analytics workloads tend to depend on high throughput of small packets, many implementations use processors with large numbers of relatively lightweight cores. In addition, applications that require fast response time will benefit from large allotments of system memory, possibly holding data close to the processor with an in-memory data store. However, as data sizes become larger—including with regularly used historic data—flash storage can play an increasing role.

Splunk

Splunk is a proprietary offering that is available for either on-premises or SaaS implementations, with the primary difference being where the data is stored: on-premises or in the cloud, respectively. It offers the largest variety of plug-ins (around 600) for integration with external tools and platforms among the products discussed here, and it has the most established ecosystem,

documentation, and user community. Splunk also provides an extensive collection of developer tools.

Splunk the company has been public since 2012. It focuses on the enterprise segment from the standpoints of feature set, scalability, and pricing, and it targets IT operations, security, IoT, and business analytics usages. The platform is feature-rich, although the complexity that accompanies that richness can create a relatively steep learning curve for new users. Splunk implements machine learning for advanced analytics capabilities such as anomaly detection and forecasting.

Elastic (formerly ELK) Stack

The Elastic Stack is a combination of open source tools that can be implemented either on-premises or in the cloud, with options for the latter that include the Elastic Cloud platform or AWS Elasticsearch Service, a hosted solution offered by Amazon Web Services (AWS). Also available is Elastic Cloud on Kubernetes, which enables the use of containers infrastructure to deploy, orchestrate, and operate Elastic products with Kubernetes. Elastic the company has been public since 2018, and the primary components of the Elastic Stack[5] include the following:

- Elasticsearch, the search and analytics engine at the core of the Elastic Stack
- Logstash, a data-ingest and transformation pipeline
- Kibana, a visualization tool to create charts and graphs

Customers can download and implement the Elastic Stack for free or choose from a variety of subscription options that offer varying levels of security hardening and technical support (either within business hours or 24/7/365). Higher subscription levels also offer more sophisticated search and analytics options, including machine learning capabilities in areas such as anomaly detection and alerting, forecasting, and root-cause indication.

5 The Elastic Stack was known as the ELK stack (named for the initials of Elasticsearch, Logstash, and Kibana) until the data-shipping solution Beats was added, causing the acronym to implode.

Sumo Logic

Another proprietary solution for log analytics, Sumo Logic is offered as a cloud-based service, without an on-premises option. Thus, although customers don't need to maintain their own infrastructure to support their Sumo Logic implementation, they are compelled to transfer their data offsite to Sumo Logic's AWS-based cloud network. The security implications of that requirement can be a blocking factor for some organizations. In some geographies, the ability to depend on always-available high-bandwidth, dependable, and secure internet connectivity might also be limited.

Sumo Logic targets small and medium businesses in addition to large enterprises, placing a premium on simplicity of set up and ease of use. Multiple subscription tiers are offered, including free, professional, and enterprise, with successive tiers adding progressively higher levels of alerting, integration, support, and security-focused functionality. The toolset implements machine learning algorithms to continually investigate log data for anomalies and patterns that can produce insights and provide 24/7 alerting in response to events or problems.

Topologies for Enterprise Storage Architecture

At its most fundamental level, the value of log analytics depends upon drawing on the largest universe of data possible and being able to manipulate it effectively to derive valuable insights from it. Two primary requirements of that truism from a systems perspective are capacity and performance,[6] meaning that the systems that underlie log analytics—including storage, compute, and networking—must be able to handle massive and essentially open-ended volumes of data with the speed to make its analysis useful to satisfy business requirements.

As datacenter technologies have evolved, some advances in the ability to handle larger data volumes at higher speed have required little

6 Of the canonical three legs of the data tripod—volume, velocity, and variety—this context is concerned primarily with the first two because of its focus on system resources, as opposed to how those resources are applied. More broadly speaking, all three are intertwined and essential not only to storage architecture, but to the entire solution stack.

effort from an enterprise architecture perspective. For example, storage drives continue to become larger and less expensive, and swapping out spinning HDDs for solid-state drives (SSDs) requires little effort or planning. Datacenter operators continue to deploy larger amounts of memory and more powerful processors as the generations of hardware progress. These changes enable new horizons for what's possible, with little effort on the part of the practitioner.

Accompanying changes to get the full value out of hardware advances require more ingenuity. As a simple example, advances in performance, security, and stability enabled by a new generation of processors might require the use of a new instruction set architecture such that software needs to be modified to take advantage of it.

More broadly and perhaps less frequently, changes to enterprise architecture are needed to take full advantage of technology advances as well as to satisfy new business needs. The rise of compute clusters, virtualization and containers, and SANs are examples of this type of technological opportunities and requirements.

Like other usage categories, log analytics draws from and requires all these types of changes. Much of what is possible today is the direct result of greater processing power, faster memory, more capable storage media, and more advanced networking technologies, compared to what came before. At the same time, seizing the full opportunity from these advances is tied to rearchitecting the datacenter. Incorporation of cloud technologies, edge computing, and the IoT are high-profile examples that most conference keynotes at least make mention of.

One critical set of challenges inherent to enabling massive data manipulations such as those involved in log analytics is the ballooning scale and complexity of the datacenter architecture required to collect, manipulate, store, and deliver value from the data. Platoons of system administrators are needed just to replace failed drives and other components, while the underlying architecture itself might not be ideal at such massive scale.

NOTE

Log analytics place performance demands on the underlying systems that are an order of magnitude greater than those of a general-purpose datacenter infrastructure. That reality means that the same architectures—including for storage—that have met conventional needs for years need more than an incremental degree of change; evolution at the fundamental architecture level might be indicated.

One example of the need to fundamentally revise architectures as new usages develop is the emergence of SANs, which took storage off the general network and created its own, high-performance network designed and tuned specifically for the needs of block storage, avoiding Local-Area Network (LAN) bottlenecks associated with passing large data volumes. Now, log analytics and enterprise analytics more generally are examples of emerging, data-intensive usages that place challenges beyond the outer limits of SAN performance.

Newer network-scale storage architectures must be able to handle sustained input/output (I/O) rates in the range of tens of gigabytes per second. Hyper-distributed applications are demanding heretofore unheard-of levels of concurrency. The architecture as a whole must be adaptable to the ongoing—and accelerating—growth of data stores. That requirement not only demands that raw capacity scale out as needed, but to support that larger capacity, performance must scale out in lockstep so that value can be driven from all that data. Complexity must be kept in check, as well, lest management of the entire undertaking become untenable.

To consider the high-level arrangement of resources as it pertains to these challenges, three storage-deployment models play primary roles (see also Figure 9):

DAS
: This is the simple case in which each server has its own local storage on board that is managed by some combination of OS, hypervisor, application, or other software.

Virtualized storage
: Pools storage resources from across the datacenter into an aggregated virtual store from which an orchestration function dynamically allocates it as needed.

Physically disaggregated storage and compute
: Draws from the conventional NAS design pattern, where storage devices are distinct entities on the physical network.

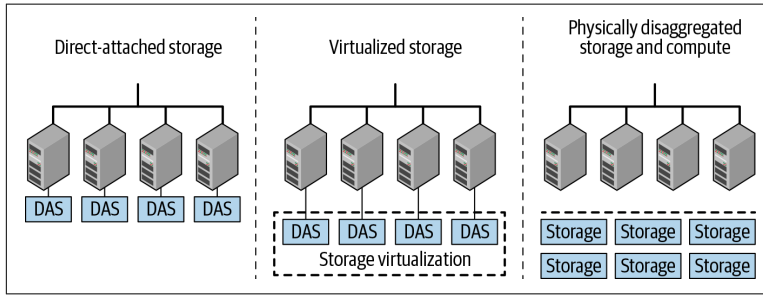

Figure 9. Common models for storage architecture in the enterprise

DAS

In many organizations, log analytics has grown organically from the long-standing use of log data on an as-needed basis for tasks such as troubleshooting and root-cause analysis of performance issues. Often, no specific initiative is behind expanded usages for logfiles; instead, individuals or working teams simply find new ways to apply this information to what they are already doing.

Similarly, no specific architecture is adopted to support log analytics' potential, and the systems that perform log analytics are general-purpose servers with DAS that consists of one or more HDDs or SSDs in each server unit. Scaling out in this model consists of buying more copies of the same server. As the organization adopts more advanced usages to get the full value from log analytics, a sprawling infrastructure develops that is difficult to manage and maintain. Requirements for datacenter space, power, and cooling can also become prohibitive.

In addition, this approach can constrain flexibility, because the organization must continue to buy the same specific type of server, with the same specific drive configuration. Moreover, in order to add storage space for larger collections of historic data, it is necessary to buy not just the storage capacity itself, but the entire server as a unit, with added cost for compute that might not be needed, as well as increased datacenter resources to support it. The inefficiency associated with that requirement multiplies as the log analytics undertaking grows.

Virtualized Storage

Storage virtualization arose in part as a means of improving resource efficiency. This model continues to use DAS at the physical level, with a virtualization layer that abstracts storage from the underlying hardware. Even though the physical storage is dispersed, applications see it in aggregate, as a single, coherent entity of which they can be assigned a share as needed. This approach helps eliminate unused headroom in any given server by making it available to the rest of the network, as well.

The software-defined nature of the virtualized storage resource builds further upon the inherent rise in efficiency from this method of sharing local storage. Orchestration software dynamically creates a discrete logical storage resource for a given workload when it is needed and eliminates it when it is no longer needed, returning the storage capacity to the generalized resource pool.

Virtualization of compute and networking extends this software-defined approach further, by creating on-demand instances of those resources, as well. Together, these elements enable a software-defined infrastructure that can help optimize efficient use of capital equipment and take advantage of public, private, and hybrid cloud. Because the storage topology for the network is continually redefined in response to the needs of the moment, the infrastructure theoretically reflects a best-state topology at all times.

Software-defined storage was developed for the performance and density requirements of traditional IT applications. Because log analytics has I/O requirements that are dramatically higher than those workloads, it can outpace the capabilities of the storage architecture. In particular, the distributed and intermixed nature of the infrastructure in this model can create performance bottlenecks that strain the ability of the network to keep up.

In addition, because data movement across the network is typically more expensive and slower than computation, it is not unusual to create multiple copies of data at different locations, which increases physical storage requirements.

Physically Disaggregated Storage and Compute

Moving to a more cloud-like model, the storage architecture can physically disaggregate storage and compute. In this model, large numbers of servers are provisioned that don't keep state themselves, but just represent and transform that state. Separate storage devices are deployed as a separate tier that maintain all the states for those servers. The storage devices themselves are purpose-built to be highly efficient at scale, handling data even in the multipetabyte range.

Because the compute and storage elements of the environment are physically disaggregated, they can be scaled out independently of each other. More compute or storage elements can be added in a flexible way, providing the optimal level of resources needed as business requirements grow. In addition, as historical data stores become larger, they benefit from the fact that this physically disaggregated model allows storage elements to be packed very densely together, to improve network efficiency, and to use efficient error encoding to get maximum value out of the storage media. The centralized nature of the storage also helps improve efficiency by avoiding the need to create extraneous copies of the data.

At large scale, physical disaggregation is often superior to compute and storage virtualization at meeting the high IO requirements of log analytics implementations. And while conceptually, this design approach is not entirely new compared to SAN or NAS, it can be built to design criteria that are beyond the performance and throughput capabilities of those traditional architectures. Physically disaggregated architectures for log analytics are, at some level, just new and enhanced manifestations of established concepts.

For example, one critical design goal is to build in extremely high concurrency, parallelizing work at a very fine-grained level that enables it to be spread evenly across the environment. In addition, the representations of file structure and metadata should be thoroughly distributed. These consistency measures help avoid hot spots within the infrastructure. This allows for the optimal use of resources to support real-time operation in the face of the massive ingest rates associated with log analytics, which can commonly exert 10 times the performance and throughput requirements on the infrastructure compared to common enterprise workloads.

TIP

Physically decoupling compute and storage enables each to scale out independent of the other, decreasing total cost of ownership (TCO) by removing the need to purchase extraneous equipment. Following an architectural pattern similar to SAN and NAS but with modern data management and flash-based storage enables the levels of performance and throughput needed for real-time analytics at scale.

The Role of Object Stores for Log Data

Object storage and file storage take different approaches to organizing data, as illustrated in Figure 10.

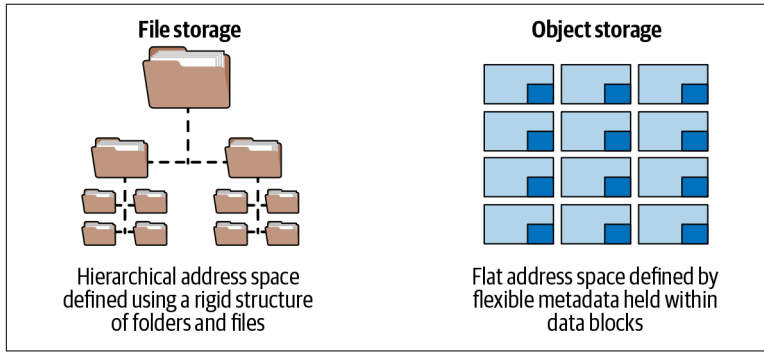

Figure 10. Data structures used by file and object storage

Object storage uses a flat address space and comprehensive metadata to store chunks of data referred to as "blocks" in place of the hierarchical folder structure used in file storage, making it far more scalable. Thus, data retrieval remains fast, even as data stores swell from larger historical datasets to feed increasingly complex analytics. In addition, object storage is well suited to managing unstructured data by means of custom metadata to describe an object's contents. This characteristic makes data self-describing, which provides flexibility for implementations of advanced analytics.

Object stores are designed to scale to hundreds of petabytes in size without degraded performance. They also protect data integrity while maximizing usable storage space. This technique divides data into shards, each of which is a subset of the table's full set of rows. Shards are distributed across multiple database instances to spread load and increase parallelism for large objects.

Multiple copies of each shard can be stored to provide data resiliency, although this approach can become prohibitive as data volumes increase. Erasure encoding can also be employed to efficiently protect data by computing and storing error-correcting shards along with the data shards.

Because it is designed to scale out, including to distributed architectures, object storage is more cost effective than file storage, especially as stores of historical log data grow larger. Moreover, because object storage is cloud native, log analytics systems can make use of either private cloud or public cloud object stores, including Amazon Simple Storage Service (Amazon S3), Microsoft Azure Blob, and Google Cloud Storage. Those environments also provide high availability and useful developer tools and design patterns that streamline the production of microservices to provide custom manipulations on log data.

Although object storage offers tremendous scalability and flexibility, accessing offsite public cloud storage can create unacceptable lags in response time for latency-sensitive log analytics applications. Accordingly, many IT organizations choose to build on-premises private cloud object storage infrastructure This approach also enables them to realize the value of object storage without entrusting their data to a third party.

Conversely, using public cloud can reduce requirements for local storage by placing a portion of the data on remote object stores. Increasing the proportion of data on public cloud infrastructure can reduce the in-house administration burden, freeing personnel from the care and feeding of a growing storage infrastructure.

For example, Splunk SmartStore combines a remote storage tier with an on-premises cache manager that allows data to reside locally. To increase performance, the warmer data in on-premises caches avoids the latency associated with fetching that data remotely.

Pure Storage provides fast on-premises storage for SmartStore to avoid that latency, making it possible to have large volumes of historical data available for fast queries while reducing the dependence on large, fast local storage for the hot tier. It also disaggregates storage from compute without having to take that storage off-premises.

The cache manager controls the flow of data between the cache and remote storage as well as monitoring and adjusting the size of those

caches according to recorded performance and hit rates. Organizations that tend to do frequent long-term searches, for example, might find that they need larger cache capacities to prevent frequent calls to remote storage.

In the present reality where not all applications are designed or rearchitected to use object stores, unified file and object access plays a pivotal role, as illustrated in Figure 11. This architecture allows data to be either ingested or accessed through either a file or an object interface, increasing compatibility with both legacy and cloud-based applications. Applications designed to access file-based data are therefore able to use data stored in the cloud as objects.

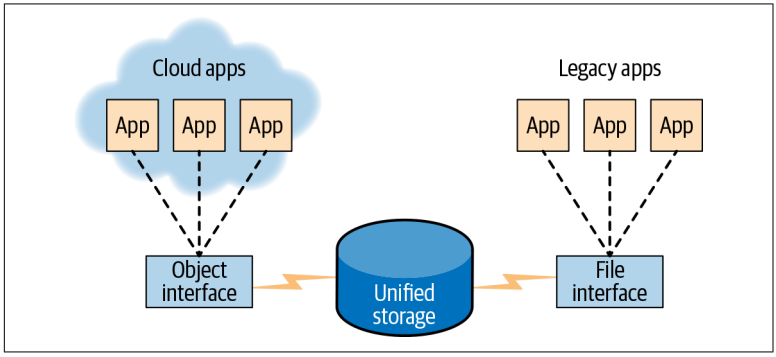

Figure 11. Unified file and object access to storage

The Trade-Offs of Indexing Log Data

Indexing data amounts to creating a map used to quickly locate the specific database record or records, allowing for faster query results compared to scanning the entire table. Determining when to index data is a nice computer science problem. In essence, the conundrum is that indexing data can accelerate queries against it later, but the act of indexing itself is computationally intensive. Particularly when indexing is done at the point of ingress, it can affect throughput and therefore scalability.

One way of preventing impacts on data ingress is to index data at rest as resources allow. It can also often be valuable to index only the highest-value portion of data, storing the remainder in unindexed form. In these usages with partial indexing of log data, algorithms must be capable of determining the priority of indexing individual, specific pieces of data, based on the likelihood of performance

benefit in future queries. This approach targets queries that are known and planned for in advance, so it must be accompanied by the ability to return fast results from unplanned queries on unindexed data, as well.

Indexing is therefore well suited to point queries, such as finding the incidence of a specific phrase or event, a so-called "needle-in-the-haystack" problem within a predictable sphere of data. Each query can consult the index and return fast results, potentially justifying the indexing overhead. On the other hand, for ad hoc queries that consider unbound amounts of data, the computational cost of scanning the world of unindexed data can be justified by the fact that indexing all of that data would be more expensive still.

Both indexed and raw, unindexed data typically cohabitate in enterprise environments, potentially accessed by different applications. A healthy combination of indexing at the point of ingest and reindexing is called for, with reindexing (despite its resource-intensive nature) playing a key role as the projected uses for data change.

Performance Implications of Storage Architecture

Growing IT complexity is a fact of life as more business processes become digitized, higher levels of automation are enabled, and new technologies enter the datacenter. That growing complexity drives increasing volumes of log data that can potentially be used for log analytics; a company of a given size would generate far more logs today than a company of similar size a decade ago.

The availability of more log data generates the potential for more sophisticated log analytics, placing more extensive demands on the underlying systems. Specifically, even as the amounts of data that need to be ingested, handled, and stored rises exponentially, so do the numbers of queries being made against it, both by automated systems running reports and dashboards as well as by human users placing ad hoc queries to generate business insights.

The emergence of flash storage for the enterprise in the past 10 years or so represents a sea change in storage architecture because of its dramatically higher speed and longevity. Notwithstanding those advantages, cost constraints mean that spinning disks still dominate in the datacenter.

Conventional HDDs are appropriate for dumping large amounts of data that won't be searched against often, but the random reads and writes are much slower than flash. Searching a huge dataset to support real-time or near-real-time log analytics requirements can be prohibitively time consuming with spinning disks.

Log analytics operations depend on rapid, dependable access to stored data, which places growing performance and scalability demands on the storage hardware. Fast response rates are critical to business use cases, both to optimize efficiency and to provide a good user experience. These requirements are driving flash adoption in the enterprise; in fact, flash storage has become the standard implementation for many use cases.

Even as the volumes of log data being generated are growing rapidly, many companies are extending their standard retention periods, requiring longer-term preservation of data. A key driver behind this trend is the fact that analytics models can often be made more powerful by making larger sets of historical data available to them. Querying against several years of data allows tracking of long-term trends, and as AI models are increasingly adopted for analytics of all kinds, those large datasets can also be useful for training deep learning models.

Log analytics presents substantial performance challenges. Throughput must be optimized from the point of ingest, through all stages of processing, to outputs such as alerts, dashboard visualizations, or reports. Data pipelines and storage must be consolidated, eliminating data silos and the practice of storing multiple copies of data for different usages.

Tiered storage is a common approach to handling large amounts of historical data, including log data. By providing multiple areas of storage, each with a different balance between cost and

performance, tiered storage allows the medium to be tailored to different needs, as illustrated in Figure 12. In this conception, the performance tier houses the data most likely to be queried frequently and needed for real-time analytics scenarios (i.e., the "hottest" data). The archive tier is for long-term storage of infrequently accessed "cold" data, and the capacity tier in between strikes a balance between the two for "warm" data.

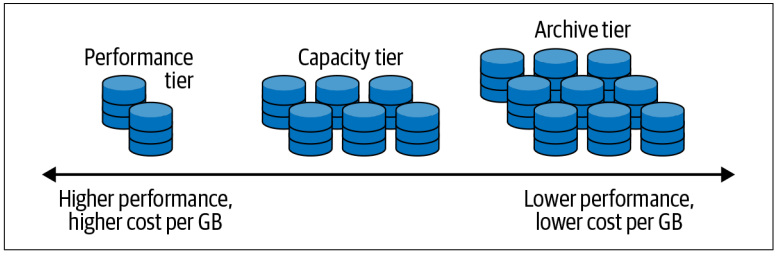

Figure 12. Tiered storage balances cost and performance for different data types

In practical terms, this range might correspond to the age of log data. For example, the most recent 30 days of log data might be stored in the performance tier, data 31 days to a year old in the capacity tier, and data that is older still in the archive tier.

As organizations continue to stretch the limits of what they can monitor, accomplish, and predict with log analytics, query volumes will continue to increase, including searches against older data. Increased requirements to perform analysis and export insights from data stored in the capacity and even archive tiers compels architects to increase the performance of those tiers, which is largely accomplished by the addition of flash storage.

Drive More Value from Storage with Data-Reduction Technologies

Because flash is an order of magnitude faster than spinning disks, it has become all but essential to data-intensive workloads such as log analytics. Particularly for real-time results that draw on historical data, the storage tier requires the speed that only flash can offer. At the same time, the relatively high cost per gigabyte compared to conventional hard disks makes getting maximum value from the available capacity of flash storage a first-order concern.

Storage vendors have devised a range of data reduction features that reduce the amount of capacity needed to store a given body of data. While not all are available from all providers, the following capabilities have been developed by the industry:

- *Pattern removal* detects and consolidates simple binary patterns within datasets (e.g., summarizing a string as "1,000 zeroes" instead of actually encoding the 1,000 identical values). These measures can reduce storage requirements as well as processing for other data-reduction measures such as deduplication and compression.
- *Deduplication* ensures that only unique blocks of data are committed to flash storage. Ideally, deduplication works globally (rather than within a volume or a pool) and on variable block sizes for maximum efficiency.
- *Inline compression* reduces the number of bits needed to represent a given piece of data. The use of multiple algorithms is desirable, to optimize compression ratios among different types of data that have different requirements.
- *Post-process compression* applies additional compression algorithms to data after the process is complete, increasing the data reduction result achieved with inline compression.
- *Copy reduction* handles data-copy processes within the flash storage medium using only metadata, producing snapshots and clones that offer greater efficiency than actual copies of the data.

To get the full value of these capabilities in the datacenter, they should be available right out of the box, without requiring extensive configuration or tuning. High-performance data-reduction measures should be always-on and suitable for use across workloads, regardless of size, type, or criticality.

Enabling Log Data's Strategic Value with Data Hub Architecture

Even though data is widely regarded today as every business's most important asset, it tends to be isolated in silos across the enterprise that are each developed to meet the needs of a discrete set of applications and workloads. This fragmentation of data limits the degree

to which it can be accessed in real time, hampering the ability to perform flexible analytics on it. This limitation runs counter to the modern perspective of data as a primary strategic differentiator, unleashed by the power of analytics.

Data lakes provide a single, unified store of structured, semi-structured, and unstructured data in its raw form as well as various transformed versions, using a flat architecture. They make all data available to any application that needs it, overcoming the data isolation that is inherent in an environment that has developed an array of data silos.

At the same time, data lakes are essentially uncurated masses of data, dwelling in a storage architecture designed to store its contents as efficiently as possible, rather than with speed of access, sharing, and delivery. This property makes the data-lake approach to storage limited for log analytics. In addition, data lakes lack the ability to tailor data delivery to the specific latency, throughput, and I/O requirements of individual applications and usages.

Data hub arose as a data-centric storage architecture conceived specifically for data sharing as efficiently as possible, overcoming limitations of data silos and data lakes for enterprise-wide log analytics, as illustrated in Figure 13. Key requirements for a data hub implementation include the ability to deliver very high throughput for both file and object storage, scale-out performance for support of growing workloads, and investment protection.

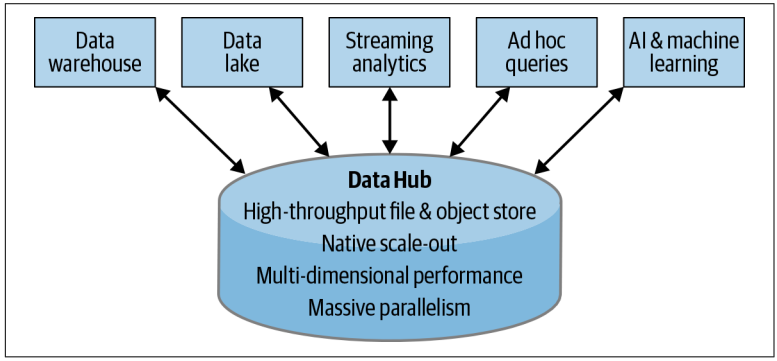

Figure 13. Unification of data with data-hub architecture

This approach specifically satisfies the different requirements for data delivery by multiple applications, allowing individual partitions to be tuned for specific workloads in a dynamic fashion that is

managed in software. This capability requires massive hardware and software parallelism to provide real-time results for demanding workloads.

Storage based on a data hub architectural approach represents software-defined, on-demand infrastructure built to accommodate requirements that are constantly changing and must be responded to in real time. It embraces virtualized constructs, including virtual machines and containers, as well as bare-metal physical hardware. The data hub also depends on having a modern storage medium such as flash technology, which is superior to older spinning disks at handling multiple simultaneous demands. This medium dramatically improves performance, particularly in the small data accesses and random reads and writes that are prevalent in analytics workloads, in contrast to the focus on large, sequential accesses in data lake architectures.

Storage Platforms to Enable Data-Hub Architecture

The specific storage systems that enterprises use when building a data-hub architecture are the foundations for next-generation data analytics in general and log analytics in particular. Because these systems must operate at petabyte scale, architects should choose storage platforms that provide high performance out of the box and that can adapt to requirements dynamically. Efficient operation across the spectrum of requirements demands that ongoing manual tuning or configuration are not required for different workloads.

Hot-pluggable blades are often the form factor of choice, allowing capacity and performance to be easily and instantly scaled out as needed. That simplicity can be critical to agility when accommodating tens of thousands of clients simultaneously as well as tens of billions of data objects. The platform should support both file and object storage so that the architecture can flexibly adapt to varying requirements. Key design criteria to consider for data hubs to provide on-demand access to enterprise data for log analytics include the following:

- *Native-built file and object protocols* to maximize throughput
- *Scale-out architecture* to increase both storage and performance linearly
- *Multidimensional performance* to eliminate bottlenecks for various workloads

- *Massive parallelism* to accommodate open-ended scale of clients and data

As nonvolatile storage has come down in price, all-flash architectures have become viable for mainstream implementations. These devices provide order-of-magnitude improvements in performance and latency over mechanical HDDs, especially with the small packets that are common with log analytics workloads. Accordingly, flash has become the gold standard for storage systems used in data hubs.

The provider of the storage systems is every bit as important as the storage itself. There is simply no substitute for their experience working with end-customer architects, system administrators, database administrators, and others to identify and resolve the diverse challenges in implementing data storage topologies for log analytics. That expertise can be critical to working through issues related to I/O and storage bottlenecks, scaling and concurrency, and high availability, among others. Top-tier providers can also provide assistance around integration with enterprise software platforms such as analytics engines, databases, ERP systems, and virtualization platforms, across operating environments, to reduce risk and achieve the best results possible.

Nine Guideposts for Log Analytics Planning

The benefits that log analytics can provide vary dramatically among different organizations, as do the infrastructure and techniques best suited to enabling those benefits. Nevertheless, the common set of best practices and considerations described here can help guide architects during the planning process.

Guidepost 1: What are the trends for ingest rates?
: Accommodate future needs for performance and capacity.

Guidepost 2: How long does log data need to be retained?
: Fine-tune data retention policies to optimize costs and minimize liability.

Guidepost 3: How will regulatory issues affect log analytics?
: Provide verifiable measures to govern data usage, transport, and storage.

Guidepost 4: What data sources and formats are involved?
: Forecast and prepare for upcoming requirements for changes in data-transformation pipelines.

Guidepost 5: What role will changing business realities have?
: Align infrastructure planning for log analytics with broader corporate strategy.

Guidepost 6: What are the ongoing query requirements?
: Identify future query volumes among different types such as ad hoc versus point queries.

Guidepost 7: How are data-management challenges addressed?
: Plan for impacts on log data formatting and delivery from changes in the environment.

Guidepost 8: How are data transformations handled?
: Ensure that tools and applications in place to transform data are sufficient for the future.

Guidepost 9: What about data protection and high availability?
: Designate log data's criticality and sensitivity, reflected in security and backup/restore policies.

In particular, it is important to keep in mind that key considerations and concerns that bear on planning infrastructure for log analytics will intensify as log data continues to grow in volume, velocity, and variety. Architects must therefore plan for flexible scalability of capacity and performance in their storage systems to support the log analytics function as it continues to become more demanding as well as more valuable to the enterprise as a whole.

Guidepost 1: What Are the Trends for Ingest Rates?

Log data originates all over the environment, and its volume grows continually as the environment becomes more complex over time. Even as multiple terabytes per day inundate the log analytics platform initially, the sheer scale of the data is unbound in the future. Determining how those data volumes are likely to grow over time is critical to understanding the future state of the environment.

In particular, the storage infrastructure must be designed to accommodate future needs from both the capacity and performance perspectives. The capacity aspects of this requirement speak to the value of decoupling compute and storage so that the latter can scale

independently of the former. The storage infrastructure must be able to scale in terms of performance to support the larger numbers of more complex queries against ever-growing log data volumes.

Guidepost 2: How Long Does Log Data Need to be Retained?

Corporate standards, audit provisions, and regulatory requirements can all affect the required retention period for log data. As volumes continue to grow exponentially, the cost and complexity associated with storing it can become burdensome or even untenable. At the same time, growing data stores make a structured approach to tracking the data life cycle more vital so that it is not retained longer than necessary.

Best practices in this area include assessment and tuning of data-retention policies to ensure that they are appropriate both to meet requirements and to ensure that retention periods are maintained at the shortest appropriate level. If possible, retention requirements should be projected forward to discern whether they are likely to increase or decrease in the future. Nearer-term requirements include data reduction techniques such as deduplication and compression to reduce the burden on storage system capacities as much as possible.

Architects should also consider options for cost-effectively storing archival data. For example, if performance requirements associated with older data stored as Amazon S3 objects are lower than for current operating data, it might be desirable to push them out to Amazon Glacier or a similar cost-optimized service.

Guidepost 3: How Will Regulatory Issues Affect Log Analytics?

Setting data-retention standards is a clear issue associated with meeting regulatory requirements, but the full scope of considerations in this area is far broader. Personally Identifiable Information (PII) and other sensitive data must be controlled and protected with verifiable measures that govern how it is used, transported, and stored. This set of concerns can affect issues such as how specific data can be used in public cloud infrastructures or shared with partners, for example.

Particularly for organizations that operate in multiple geographic areas, data sovereignty can be a complex issue. Because data is governed by the laws of the jurisdiction where it is located, organizations must be concerned with the physical locations of their data, particularly in cases for which public cloud resources are used. For example, data that was collected through perfectly legitimate means in one country can be in violation of the privacy laws in another.

As a related matter, confidential data might be subject to subpoena or other unwanted inspection by government or legal entities in the jurisdiction where it is stored. Response times potentially required by subpoena actions can be a challenge in the common case where it requires weeks to restore older data from backup and then days to query against the associated large volumes of data. For organizations that must regularly respond to law-enforcement requests for archival data, architects might need to accommodate streamlined access.

In a world in which regulations over data privacy and usage are evolving, forecasting legal requirements can be problematic. One approach is to consider the measures taken to date by government entities that have provided early leadership. Frameworks such as the European Union's General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA) suggest the types of regulatory patterns that might later be adopted elsewhere.

Guidepost 4: What Data Sources and Formats Are Involved?

The semi-structured nature of log data means that the sources of those logs play an important role in determining how to handle the data. As IT complexity and the scope of sources grow and change over time, the associated challenges can become more complex. Particularly as IoT topologies are built out over the next several years, many organizations will find themselves needing to accommodate a vast assortment of new sensors and other endpoints, and that variety will also usher in an expanded diversity of log types and formats.

As the universe of data sources and log types becomes broader and more complex, novel transformation pipelines will be needed to create a coherent whole from that data. Forecasting those changes in

advance and accommodating them in how the infrastructure scales is an important set of requirements for architects. The ability to cost-effectively scale out compute and storage, independently of each other, is central to successfully meeting this set of evolving requirements.

Guidepost 5: What Role Will Changing Business Realities Have?

The changing needs of a business and its use of log analytics to guide business change are deeply intertwined. Planning should include considering what business questions log analytics can answer, such as how to allocate resources for maximum efficiency or whether to launch a marketing initiative geared toward increasing revenue. Architects must consider how intelligence generated through log analytics can inform business strategy.

Just about any major business event can have an impact on infrastructure planning for log analytics. That reality makes it valuable for infrastructure planning to draw on the business's larger corporate strategy. For example, architects should consider the potential impacts if the company were to enter into new market segments or geographies. Either type of change would be likely to increase the volume and variety of log data. Likewise, mergers and acquisitions could introduce new and unpredictable infrastructures alongside what the company already operates. Because every organization is subject to unforeseen circumstances, architects must design flexible, scalable frameworks that can accommodate uncertain future needs.

Guidepost 6: What Are the Ongoing Query Requirements?

The central issues around query requirements are what types of searches are being done against the data and how many. In addition to searches by human users and machine-to-machine systems, planning must take into consideration factors such as executive dashboards, data visualization systems, reporting, and real-time alerting. Architects must account for potential addition of such new loads on the log analytics infrastructure as well as the expected growth in workloads generated by existing systems.

The nature of the queries being made against the log data store is a key design factor for log analytics. In practical terms, for example,

architects should identify what data is most likely to have ad hoc queries made against it, as opposed to point queries. They also need to consider how often each type is likely to occur as well as how much historical data is likely to be searched and how frequently. These capabilities should be tuned to avoid slow or cumbersome search, avoiding lost opportunities or missed deadlines.

Likely usages such as ad hoc queries being done in conjunction with threat hunting activity should be considered to help guide this process. The answers to those questions can help guide design decisions about factors such as how to implement tiered storage or which data to index at the point of ingress.

Guidepost 7: How Are Data-Management Challenges Addressed?

Effectively managing data is a cornerstone of driving value from the massive volumes of log data constantly being generated by hardware, software, and processes of every description. In particular, the inherent variety within log data adds complexity to matters of schema evolution and back-compatibility as log data sources change and multiply. Solution architects should identify strategy for maintaining coherence and functionality in the face of such changes.

Data-management concerns associated with log analytics extend to changes in the operating environments in which log data is generated. For example, the firmware or software running on sensors, systems, and other entities that are transmitting logs might be upgraded or reconfigured. Those changes may alter the formatting of log files, requiring adaptation by the log analytics platform. Proper planning requires establishing processes to anticipate such changes in advance and setting standards for addressing them when they arise.

Guidepost 8: How Are Data Transformations Handled?

Transforming log data as it is collected from a wide variety of sources places significant demands on the underlying systems that must typically be satisfied in real time on steadily growing data streams. The alternative of simply transmitting and storing logs as flat files puts untenable processing burdens on analytics processes. The transformation workload itself can be highly compute intensive, and

it takes place over a complex and varied compute layer. It also places significant demands on the storage layer.

The tools and applications in place to perform those transformations must be capable of handling any foreseeable data requirements as well as integrating and interoperating effectively with the other components of the log analytics pipeline. Likewise, the underlying infrastructure must be designed to provide storage and compute architectures that can accommodate the associated performance and scalability requirements.

Guidepost 9: What About Data Protection and High Availability?

As log analytics evolves within an organization, it enables increasingly sophisticated usages that deliver increasingly significant value. Over time, those usages can become business critical or even mission critical, elevating the value of the underlying log data as well as the requirements for assuring its accessibility. Foreseeing and planning for that transition involves providing high availability for a subset of log data, without interfering with the smooth operation of analytics based on mixed data streams.

Designating specific bodies of log data as critical is also tied into other aspects of IT planning. From a security perspective, this status must be considered when identifying requirements for how it should be protected and how sensitive information in the log data should be masked as well as its recoverability after tampering or other interference as the result of a breach. Backup and restore processes for protecting that data should also reflect its potentially changing value to the organization.

Conclusion

To deliver on the potential for log analytics to improve operations across the business, architects are challenged to adapt their existing storage infrastructures to real-time needs for access to diverse log data at scale. Done right, legacy architectures based on spinning disks can be replaced with all-flash solutions at similar or lower cost. Data-hub architectures, for example, can dramatically increase throughput and scale while tailoring data access to the needs of

specific applications and workloads, so that log analytics can better meet the needs of the business.

Disaggregating storage and compute facilitates that scalability by enabling each to be built out and added to independently of the other, using a cost-effective "pay as you grow" scale-out approach. At the same time, a well-designed log analytics architecture can seamlessly scale capacity and performance together, helping to ensure that query and response performance continues to meet SLAs and provide excellent end-user experience as the volume of log data and the purposes it serves continue to grow exponentially.

About the Author

Matt Gillespie is an independent technical author and editor working out of the Chicago area and specializing in emerging hardware and software technologies. You can find him at *http://www.linkedin.com/in/mgillespie*1.