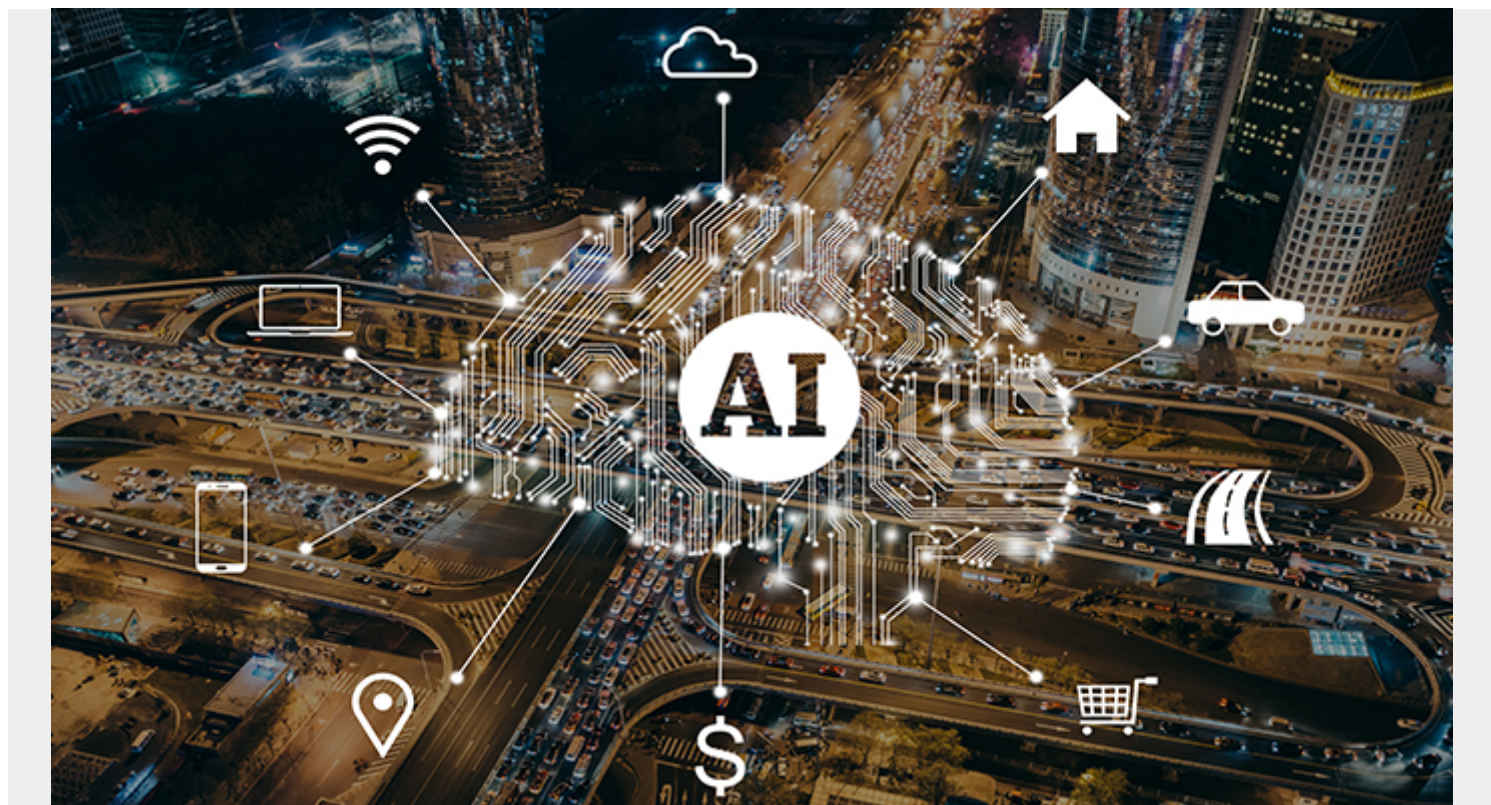


A ROADMAP TO AIOps



In my conversations with customers about AIOps, I frequently hear concerns about maturity. Customers may believe, for example, they aren't mature enough to implement analytics, or that there is a linear progression for AIOps capabilities and they must start from a certain point corresponding to their own maturity self-assessment. Oftentimes they say something like 'I have to get X in place first before I can even think about Y'. Usually the "X" they are talking about is getting a handle on exploding amounts of events and alerts or unifying dispersed monitoring.

I understand and empathize with their concerns. At the same time, I think that decades of ITIL training, with its rigid and regimented processes – reinforced by analysts and vendors – has made it difficult for all of us to see the possible or envision alternative solutions to our long-standing problems. AIOps holds the promise of step-function improvement without the strictures of ITIL but there is very little practical guidance about what that might look like.

In this post I want to propose some concrete steps that I believe are required or highly desirable to build an AIOps practice. I will then offer a 'roadmap' for taking these steps in an AIOps implementation, indicating which are prerequisites for others, which can be pursued simultaneously and which have dependencies.

A Quick AIOps refresher

Gartner has identified an emerging IT market trend: traditional IT processes and tools are not suited to dealing with the challenges of modern digital business. (More information [here](#)) This has to do

with the velocity, variety and volume of digital data; the distribution of responsibility and budget in the broader organization outside of IT; and the need to move from offline, historical analysis to real-time analytics.

Gartner's response to this trend is AIOps: the merging of IT Service Management (ITSM), IT Operations Management (ITOM) and IT Automation at the data layer. That data must reside in a big data platform that supports the application of real-time analytics as well as deep historical queries. The analytics must be managed by machine learning that supports both supervised and unsupervised processing as the data streams in.

The idea is that tools in the IT silos remain sovereign, e.g. Service Management still handles requests, incidents, etc. and Performance Management still monitors metrics, events and logs, but that their data is joined and subjected to machine-driven analysis for the purposes of enabling a) better, faster decisions and b) process as well as task automation.

Keep the End State in Mind

Remember that the end state is a system where data streams freely from multiple IT data sources into a big data platform; that data is analyzed upon ingestion and post-processed with data from other sources and types; machine learning is used to manage and modify the analytics and algorithms; and automated workflows are triggered, whose output also becomes a data feed into the system. The system adapts and responds as data volumes, types and sources change, automatically adjusting response and informing administrators as needed.

Early stage: Identify your current use cases

In a situation of change, transformation and fluidity, the best place to start is with what you know. Most customers have initiatives around solving for use cases that they can't currently accommodate, or adapting how they are currently solving for a use case to be more responsive, scalable, accommodate new technologies, etc.

I always encourage customers to enumerate the list of use cases that they currently address or want to address. Having disclosure and transparency around current 'desired' state opens the dialogue to:

- Questioning the 'why' of those desired outcomes
- Assessing the priority of specific use cases
- Highlighting gaps in capability, tools, skills or process

This is a terrific starting point for developing an AIOps strategy that will be successful. Emphasis on "starting". We don't know what we don't know – new use cases will come up, new desired outcomes will emerge and priorities will shift as your business and technologies change. New AIOps approaches will open new possibilities and pose new challenges.

The important thing is to start down a path with a purpose that bridges where you are to where you want to be. If where you want to be changes, no problem, you can course correct. But if you don't know where you are and have a realistic understanding of what is needed to get to desired state, you will end up unfocused and likely unsuccessful.

Early stage: Assess your data freedom

The foundational element for AIOps is the free flow of data from disparate tools into the big data repository. Accordingly, you must assess the ease and frequency with which you can get data out of your IT systems. The optimal model is streaming – being able to send data continuously in real-time.

Few IT monitoring and service desk tools support streaming of outbound data. They may support programmatic interaction via REST API in more current versions or iterations. However, if they are based on traditional relational data bases like Oracle or SQL, even having a programmatic interface doesn't mean that they will be able to support streaming. The performance impact to production systems using relational data bases may be too great as they are not designed to support the continuous outflow of data.

Getting clear on your data streaming capabilities is an early and high-priority activity in developing an AIOps strategy. Answer these questions for each data source:

- How do I get data out of my current IT tools?
- What data can I get?
- Can I do it programmatically?
- How frequently can I do it?

The constraints you discover may cause you to change your data consolidation strategy (e.g. start with batch uploads vs streaming) or consider replacing your IT tools with ones that will support real-time data streaming.

Early stage: Agree on a system of record

A second foundational element for AIOps is organizational alignment and communication. Suggesting that IT Operations and IT Service Management come together to review joint data requires that the teams agree on a 'source of truth' and establish a regular cadence of interaction with clear roles and responsibilities. The latter is a larger topic that requires a longer conversation I will pursue at a later date. Here I want to focus on making joint decisions based on shared data.

The data I'm speaking of here is not all the data that might flow into the AIOps big data store for analysis. It is the data required for IT leaders and practitioners to understand what is happening in their environment, understand what actions have been or can be taken, make decisions, and ultimately track their effectiveness. With respect to agreement on data, teams must determine:

- A minimum set of data that is required to overcome the limitations of the status quo
- Where the data is to reside
- The joint view/access that teams will share

In many mature IT organizations, that system is the Service Desk because in the traditional ITIL model, the Service Desk is where request, incident and change data was expected to co-exist. This model gets challenged, however, when [DevOps](#) teams use Jira to log defects and enhancements, use APM tools whose events and telemetry aren't captured by IT Operations or Security teams are working independently to identify threats.

Preparing to implement AIOps means identifying all of the effective causes and resultant indicators in your application, service or business value chain and putting a plan in place to bring that data together. You may leverage the big data platform if you can build meaningful dashboards on top of

it that filter the mass aggregate of data for the specific uses of different IT audiences. Single data source – multiple views. However, it may make more sense in your environment to select a subset of data and feed it into (e.g. Jira tickets, APM events, etc.) your established system of record.

Early stage: Determine success criteria and begin tracking them

Successful management of any business and certainly IT, begins with an understanding what key performance indicators (KPIs) or metrics best indicate success or failure. It seems facile to say but is worth repeating that:

- Understanding what to measure
- Implementing consistent and robust measurement
- Regularly reporting out or providing visibility to the performance measures and
- Holding responsible parties accountable

is required for actionable understanding of your business.

Most organizations measure lots of things. Most IT tools come with lots of measurement tools and templates. But frequently, an understanding of the business needed to identify which of the things is important is missing. I have been in many situations where teams report out to me on 'performance', but when I ask why such a measure is important or what is driving it, the response is a blank stare or 'I'll get back to you'.

Quantity doesn't trump quality in measurement. It may be that there is one thing that needs to be measured – assuming you know what drives that measure up or down. Those things too may need to be measured but without understanding causal relationships, simply throwing graphs on a chart is unhelpful and more often detrimental. Understanding your KPIs is understanding your business.

Also often neglected is a comprehensive process for sharing information, engaging stakeholders, determining actions and holding people accountable. Visibility is primary, but visibility without action or response is empty. When action is required, people and teams need make commitments with timelines and execute against them. These need to be documented and measured as well to ensure that the business, and hence the KPIs, move in the right direction.

Mid stage: Assess current and future state data models

This is one that is critical, but which few customers understand or feel comfortable addressing. Essentially, you must take stock of the data model for each of the data sources you want to use for your AIOps solution and the data model that is required to realize the AIOps use cases and determine how the data from different sources will interact to deliver the desired results.

The reason this is challenging is that the data model in most IT tools is hidden from the user, few organizations have an idea about how big data platforms (NoSQL) differ from traditional data bases (SQL) and fewer still have data analyst/science expertise. I have written a separate blog post [here](#) on big data for AIOps that gives some background and context. Here I want to address the idea of data 'relationships' for the purposes of analytics.

The AIOps approach is to join data from different IT (and non-IT sources) in a single big data repository. The idea is then to make that data 'talk to each other'; to find relationships in the data that will yield insights unattainable when the data sits separately in different silos. But what are those

relationships? How can diverse data from different sources with different structures be brought together for analysis? And who can do it?

There are a number of shared data structures that can be processed by an AIOps system without additional modification from AIOps practitioners:

- Timestamps – events, logs and metrics all have time signatures that can be used to bring them together around a point in time or a time window. Timestamps can be used to correlate events with each other and with time-series data for causal analysis.
- Properties – using the term loosely for key pairs (key : value) of information associated with an event, log or metric such as 'status', 'source', 'submitter', etc. Properties can be used to create relationship models between different data sets.
- Historicity – the past performance of time-series or event activity data. This can be used to forecast future performance or predict future threshold achievement (e.g. saturation, degradation, etc.)
- Seasonality – the shape or regularity of time-series data over a day, week, month, etc. Seasonality can be used to correlate multiple data sets or anticipate resource requirements for scalability, e.g.
- Application, service and business models – if you have a robust and regular discovery and configuration management practice, you can leverage these to inform an AIOps platform with asset relationship information for grouping, correlation, suppression, de-duplication, etc.

In general, IT time-series data is well formed and structured. Correlating, analyzing and forecasting time-series data is a fairly well-established practice in IT Operations monitoring and management tools. What changes for AIOps implementation is the need to bring together IT and non-IT data (e.g. user counts + performance, latency + conversions, etc.); increase the granularity of data e.g. from five minutes to sub-one minute; and the application of analytics on streaming data – in 'real-time' or on ingestion – vs. ad-hoc historical queries.

For IT events that have structured, semi-structured or unstructured properties, AIOps represents a paradigm shift. To begin with, most IT event data is not well formed. Human generated events are inconsistent, with large amounts of missing or unstructured data. Machine generated events have more consistency, but are often incomplete and have large amounts of repetitive, semi-structured data. They also come in at an order of magnitude in volume larger than human generated events. Machine logs, seen as events, are essentially blobs of semi-structured data. For AIOps analysis of events to be effective, AIOps systems must overcome the challenges of poor, missing, incomplete, incorrect and unstructured data.

This is why much of the current activity in the AIOps space is centered on event management, analysis and correlation. Once data begins to flow into an AIOps platform, customers must consider how they will approach data structure and integrity to support machine analysis. One strategy is to perform 'ETL' (Extract, Transform, Load) on incoming data. Specifically, normalizing and transforming data as it flows in, to adhere to centralized standards so the data can be correlated and analyzed.

This approach suffers from limitations that will likely make it untenable for many enterprises. First, the amount of processing required to transform the data on ingestion but before analysis will likely either render the system not real-time or be cost prohibitive. Second, any centralized standard that is manually managed will require constant maintenance that will not be able to keep up with changes and will only comprehend the known, not the unknown or new.

A more promising strategy is "tagging", which is what is employed as a best-practice in most cloud services. Tagging allows the hashing of variable attributes of different types of objects, which can then be referenced, sorted, correlated and analyzed using the tags – regardless of what the object is or how it is tagged. Instead of requiring mapping of pre-defined properties with common values, tags are fluid and can change with the data. Tagging is how NoSQL databases handle attributes and how hyper-scale analytics tools like Elasticsearch are enabled. Additionally, tagging can be done in real-time by machines as data flows in, which overcomes blindness to the unknown and human-scale limitations.

For customers looking to adopt an AIOps strategy, understanding current and desired data structures is a critical but secondary consideration. First you need to get the data flowing together. Any big data platform that supports an AIOps practice will have the capability to support the ETL or tagging approach. After data is flowing, you can determine which one works best for your business needs and budget.

Mid stage: Implement Existing Analytics Workflows

It is likely that when you begin your AIOps journey, you will already have certain analytics in place. I do not mean here the analytics that are embedded in your IT tools. I mean offline, mostly manual analytics that you do regularly, irregularly or periodically to identify areas for process improvement, reduce costs, improve performance, etc.

These manual efforts are precisely what your AIOps solution should address and automate in its first iteration. Once the data you use to do these investigations is flowing into your data platform, you should seek to recreate and automate the analyses. The initial value you will generate is reduction of manual effort spent on analysis, but you should also immediately be able to increase the frequency and perhaps the scope (data points, systems, permutations, etc.) of the analysis.

Remember that AIOps is intended to put you into a position of doing real-time analysis on data sets beyond human scale. The easiest way to move in this direction while simultaneously realizing immediate value is to reduce the time/effort and increase the speed/frequency with which you do analyses that are already part of your operational process.

Mid stage: Begin Implementation of Automation

Ah, automation. Everyone knows its value. Everyone knows they need it (or at least could use it). Few organizations put it into practice. Fewer still approach it as a practice with discipline. There used to be a mantra in performance management – 'Monitor all the things!' The mantra in the digital era is 'Automate all the things!'

It should be sufficient to say that in a digital enterprise, data grows and moves at speeds beyond human scale. To address this you need to turn to machines to perform analysis and execute automation. There are, however, other process factors that impact the desperate need for IT operations to automate. Prominent among them is the rise of the developer and DevOps, more specifically "continuous" integration and delivery (CI/CD).

Let's clarify something first: you **automate** tasks; you **orchestrate** processes. Task automation in IT Operations typically has been and remains segregated by tools. Your service desk has some automation, you have automated patching for your servers, you may automate some remediations from your monitoring tools. Orchestration across these tools is more difficult to achieve and rarely

fully accomplished.

DevOps is essentially the automation of development tasks and their orchestration – to eliminate the bottlenecks caused by phased review processes in waterfall developments, segregated test and compliance activities and operational, pre-production interlocks. What this means for IT is that DevOps application teams creating the innovative cloud services impacting the business are now moving at lightning speed compared to the traditional application teams of the past.

For IT Operations to keep up, they must not only 'automate all the things', they must orchestrate them and also plug into the CI/CD tool chain. If you don't know when things move from test to staging to production; if you don't know who owns the code or what impact it has on production; if you can't measure and identify developer backlog/productivity on business services, you can't effectively manage your environment.

That is the situation that modern IT Ops finds itself in. They need to match the speed and agility of the DevOps teams spread throughout their organization while simultaneously adding visibility to those teams' activities into their value chain. This begins by automating and orchestrating the things they already do – across siloed tools – and finding ways to connect, share information and communicate with the DevOps teams in their enterprises.

Late stage: Develop New Analytics Workflows

Above I talked about implementing existing, manual analytics workflows into your AIOps solution to automate and scale them. Once this is accomplished, you should have the bandwidth to:

- Assess the value of those workflows
- Modify and improve those workflows
- Develop new workflows based on the existing or to address gaps

Part of the problem with the 'brute-force spreadsheet' approach to doing analysis with disparate data sets is that the energy and focus it requires oftentimes exhausts the capacity for the practitioner to assess the value of what is being delivered. Reports have been promised, meetings are scheduled and expectations have been set. Unless a leader calls for a re-evaluation of the approach, rarely is the process questioned.

Once the existing process has been automated in the AIOps platform, the practitioner can step back and evaluate whether the necessary information is being analyzed, insights are being gained and results are actionable. Having done so, s/he can make improvements using the AIOps platform – which should be an order of magnitude easier than doing so in the spreadsheet(s) – and evaluate the impact of those changes.

Simultaneously, s/he can determine where information/insight gaps exist and envision higher-levels of analysis that leverage the outcomes of existing workflows. Again, the promise of AIOps is the ability not only to execute what heretofore wasn't practically feasible; it's doing it at a scale and speed that makes previously unrealized analytics opportunities possible.

Late stage: Adapt Organization to New Skill Sets

It should be obvious by now that if the AIOps platform is taking the analysis and response activities off of the plate of the IT Ops practitioner, the role of the IT practitioner will evolve. You will transition

out of the need to have someone who has domain knowledge for the purposes of tactically addressing issues to one who can put that knowledge to use training the system.

This is not a simple semantic distinction. The ability to know when something is wrong, determine how to tell a system to alert about that fact and then fix it is fundamentally different from the ability to understand how systems are operating well or poorly, how the system is reading and reacting and then adjust the system accordingly (or give appropriate guidance thereto).

IT Ops will move from a 'practitioner' to an 'auditor' role. This doesn't require in-depth, data-science level understanding of machine analytics. It does require understanding how systems are processing data and whether the desired business outcomes are being achieved. Of all of the changes AIOps will bring to IT Operations, I believe this will be the most disruptive.

IT Operations has long had a bunker, hero mentality, particularly with monitoring teams. Giving up control to a machine will be one of the most difficult transitions those who have been steeped in the practice for decades will experience. Many will not be successful. This is an inevitable result of market trends as they exist now. The move to business beyond human scale will have significant consequences for the humans who have been used to managing it.

Organizations will have to cultivate this new skill in their existing – reduced – workforce or bring in talent that either has the skill or can adapt to the change. This will be challenging in two ways: the scarcity of such skills and the fact that the market may take a while to respond with the education, certification and practical opportunities necessary to build a robust AIOps labor force. It will take time for these changes to have noticeable impact and it may be that only the highest-performing organizations understand and realize it. But it will happen and will be a tectonic shift in the discipline of IT Operations.

Late stage: Customize Analytic Techniques

The last activity I will discuss is both the most speculative and the most contentious. It is the question of whether IT Operations organizations will need to develop a mature data science practice or not. Some analysts believe you do. I disagree. I believe in the segregation between domain and data science knowledge.

I have two preceding paradigms in mind: the scientist-analyst and the developer-analyst. Scientists have long been executing complex, data-intensive analyses. With the rise of machine computation, scientists had to develop, at least, the ability to craft the mathematical algorithms that they wanted to run on their data sets. At first, when computational resources were shared, scientists built their own analyses to be run on systems maintained by computer experts. The languages, parameters and constraints were dictated by the systems and scientists had to work within them.

In that paradigm, scientist developed specialized knowledge that allowed them to leverage the computational systems. Once computational resources and analytic languages became less expensive, more powerful and more accessible, scientists had to develop not only the domain knowledge in their fields, but also data science and computational knowledge sufficient to execute their desired analyses on contemporary computing platforms.

They were able to do this because:

- Their programs were research, not commerce and hence weren't subject to market or business pressures (at least not immediately like IT)

- They were self-selected for the education, drive and acumen to learn and master both types of knowledge (Ph.D.)
- They were afforded the time in an academic setting to acquire the skills and knowledge necessary
- Failure to do so would be fatal to their careers – labor competition

Let us contrast this with the programmer-analyst. Currently the market stands in critical need of data science practitioners who can also implement their data science knowledge in code. In spite of the ubiquity of data science jobs and data science education (both formal and informal), the market is bereft of people who have M.A. or Ph.D. level knowledge of statistical modeling (e.g.) and are at least adequate Python or R programmers.

This may change but I do not foresee that happening soon, if ever. It is simply the case that it is too hard for most people to learn the math required and too easy to make very good money with just the coding to incent them take on more than that. And even if they did, they still would need the domain knowledge required for a particular industry or problem area.

Asking IT Operations practitioners to know math, IT and coding to manage infrastructure, applications and services is, I think, too much. In my vision of the future, IT Operations would be the stewards of semi-intelligent, semi-autonomous systems with deep knowledge of the domain, sufficient knowledge of the math to understand what the system is doing and no knowledge (or no need for knowledge of) coding.

In this paradigm, AIOps vendors provide systems that offer multiple analytics options from which practitioners select combinations that best fit their environments. Ideally this would require less knowledge of the math than of the business outcomes. Also ideally, the AIOps platforms would provide regression analysis that would suggest 'best-fit' options from which practitioners could make informed decisions.

This is how I see new and customized analytics coming out of AIOps. Some organizations may have the wherewithal and will to have domain, data science and programmatic implementation expertise. Teams of people. For revenue generating activities, this may make sense. I don't see a future where such an approach will be feasible for IT Operations.

Concluding Thoughts

I have offered 9 steps for an AIOps roadmap.

1. Identify current use cases
2. Agree on a system of record
3. Determine success criteria and begin tracking them
4. Assess current and future state data models
5. Implement existing analytics workflows
6. Begin implementation of automation
7. Develop new analytics workflows
8. Adapt organization to new skill sets
9. Customize analytic techniques

#1 and #3 are table stakes for IT operations in its current state and so certainly for AIOps. If you don't know what you are currently trying to accomplish and/or you can't measure it, you can't hope to

manage it even with existing tools. #2 can be done with existing tools or it may be an assessment that current tools are unsatisfactory. If the latter, building out requirements for how different organizations will share a view of what is happening is the logical response. These are all early stage activities on your AIOps roadmap.

#4 is a requirement for any activities that follow. As I mentioned in that section, understanding your current and future data needs is paramount to a successful AIOps implementation. It can be done piecemeal, but it must be done. #5 depends on #4. #6 depends on #5 for the analytics portion of the AIOps process but automation of tasks and orchestration between tools can and should be pursued at whatever stage of maturity an IT organization finds itself.

#s 7, 8 and 9 are more intertwined and likely to evolve organically in tandem, taking different courses in different organizations. It may be impossible to forecast or plan at early- or even mid-stages for their eventualities but the highest performing organizations will comprehend them in their strategic horizons.

To paraphrase [Peter Drucker](#), the future has "already happened". The only IT organizations that aren't thinking about how AIOps leveraging machine learning, big data and analytics will radically alter the way they function are those that haven't realized it yet. And they are the ones likely to miss the almost limitless opportunities that [digital transformation](#) presents.

To learn more about BMC's approach to AIOps and how we can help you on your journey, read more [here](#) or [contact](#) us.