

Succeeding in Economies of Speed

By Gregor Hohpe, AWS Senior Principal Evangelist

Economies of speed

As we look out our windshields, the future is fast approaching. Our technical capabilities are improving rapidly, and the pace of innovation is increasing. Customer expectations and competitive pressures also are growing at top speed, putting more demands on businesses amidst trade tensions and supply chain disruptions. These demands are bound to affect your organization's technology stack, processes, and structure, and what we expect from leaders.

Many organizations are transforming to succeed in these new economies of speed. At AWS, we have supported many customers on this journey and have observed that the cloud is a critical asset for this transformation — as long as organizations don't view it as a mere tech upgrade. To fully understand the potential of cloud computing, it's helpful to examine how the role of technology in organizations has fundamentally changed. And for that, we look back at an earlier "digital transformation."



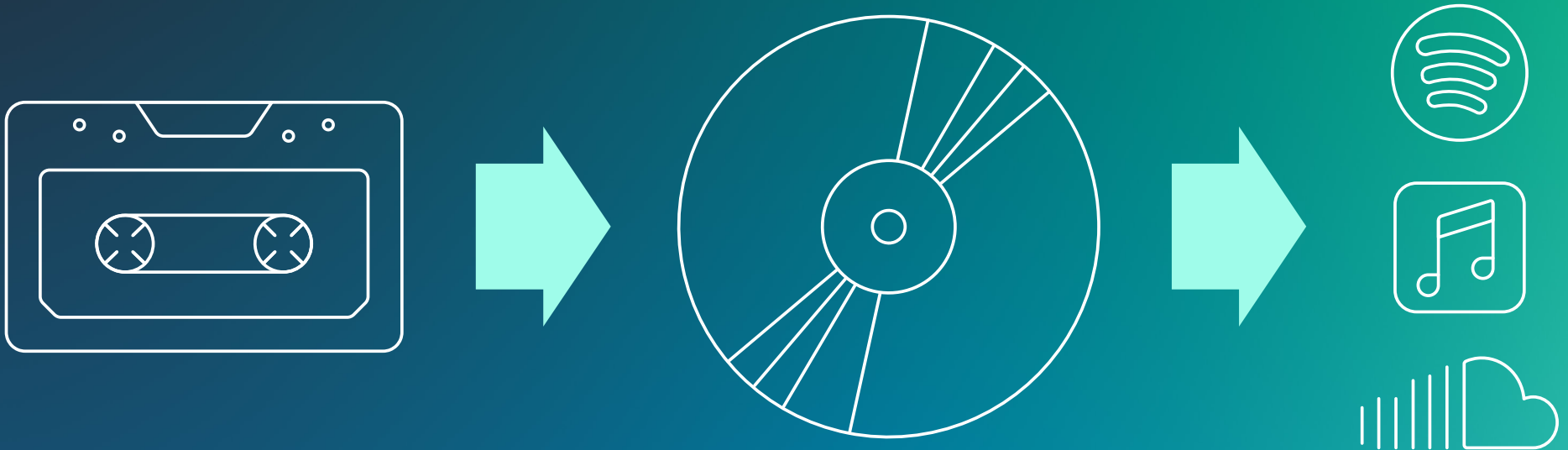
Beyond digital copies

Years ago, music came on cassette tapes, which we dutifully inserted into a player to enjoy 45 minutes of our favorite songs. (Some of us still remember vinyl and marveled at tapes' ability to play both sides in one go.) When the compact disc, a digital medium, appeared, it improved virtually every aspect of the cassette tape. It held more music, had better sound quality, and could immediately play any song. Today, though, the CD has also become a relic of the past, made obsolete by music streaming.

The CD improved an existing product with better digital technology, but it was not a transformation because it did not change how users interacted with the music. We still had to buy a physical product and insert it into a player to listen to a limited collection of songs. Streaming, however, did change the

model. Now we have a hundred thousand songs at our fingertips, paid via subscription. There's no player to buy, and machine-learning algorithms can suggest new songs based on our listening history. Streaming, in a word, was transformational.

Corporate IT has traditionally automated manual processes. Although those improvements made corporate processes more reliable, accurate, efficient, and faster, they were just digital copies of the originals. In other words, IT made enterprise-wide compact discs. Meanwhile, digital-native competitors were redefining how customers interact with a business and its products; they were streaming.



From economies of scale to Economies of speed

Building digital copies is baked deeply into most IT departments' operating models. In a typical IT environment, work begins with a business case that justifies a project. When the project finishes, we return to business as usual while the business case unfolds.

This approach is based on two foundational assumptions. First, the business needs to have a good idea of what it will build and the benefit it will bring. That's the basis of the business case calculation. Second, it's assumed that the change can be contained within the project. The business itself does not change; it is stable. In this model, change is business as *unusual*.

While this model works well when creating digital replacements for existing processes, organizations looking to create something novel (like streaming) cannot calculate a predetermined return or build a business case upfront. They can only test their ideas and hypotheses by running experiments, creating solutions, and observing how users react. And, because it's unreasonable to expect the first experiment to be a home run, the business must experiment continuously. To drive innovation and achieve differentiation, change, not stability, must become business-as-usual.

This new operating model cannot rely on economies of scale; its success depends on agility and fast learning — economies of speed.



The vocabulary of economies of speed

Economies of scale

Absolutes
Investment
Budget
Timeline
Capacity
Use Cases

Economies of speed

Relatives / Rates
Price per unit
Burn Rate
Velocity
Consumption
Cost of experimentation

*Organizations embracing economies of speed
think and speak in the first derivative*

The shift to economies of speed requires a different way of thinking — and a new vocabulary.

Organizations that leverage economies of scale describe their projects in absolute values: a specific investment made over a concrete timeline, with predetermined headcount and hardware investments.

For organizations that have embraced economies of speed, such absolute values are less meaningful. These companies, acknowledging the reality of constant change, speak in terms on the right side of the chart. They discuss the burn rate and the project cost per month. They track velocity, how many features the team can deliver per agile sprint. They aren't looking for the perfect idea; instead, they try to minimize the cost of experimentation. And because one can't predict an experiment's outcome, procuring servers for tens of thousands of dollars would seem foreign to these organizations. Instead, they consume computing capacity by the hour or minute from the cloud. Mathematically speaking, they think in the first derivative; they're less interested in where they are than how fast they are going.

The cloud's essential nature encourages economies of speed.

Transformation: Rethinking your levers for change

CIOs are used to operating between a rock and a hard place. Cost and compliance pressures dictate harmonization and standardization, while the business demands innovation and agility. CIOs had to choose between doing things quick and dirty or slow and right, it seems.

Today's technology has reduced or eliminated many historical constraints: harmonized cloud platforms boost innovation, automation speeds up software delivery while improving quality, and machine learning creates customized experiences for minimal cost. Because those past constraints are baked into an organization's existing operating model, a company can't transform by fine-tuning the way it currently works. It must define a new model that acknowledges a world with fewer constraints. That means, it has to rethink its levers for change.

Traditional change programs adjust the org chart, perhaps setting up new teams or departments, design new processes, and establish guardrails to ensure compliance and reliable operation. Meanwhile, they grant exceptions from existing processes to an experimental unit.

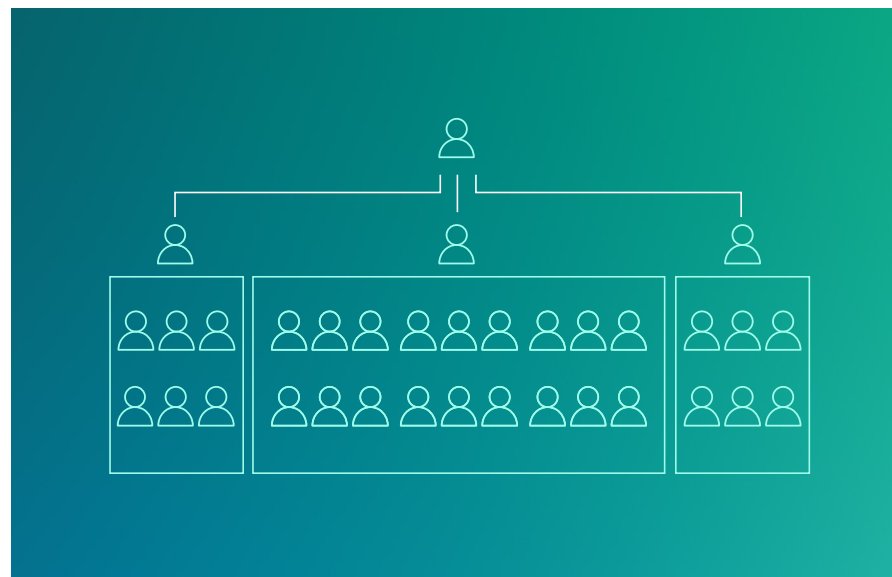
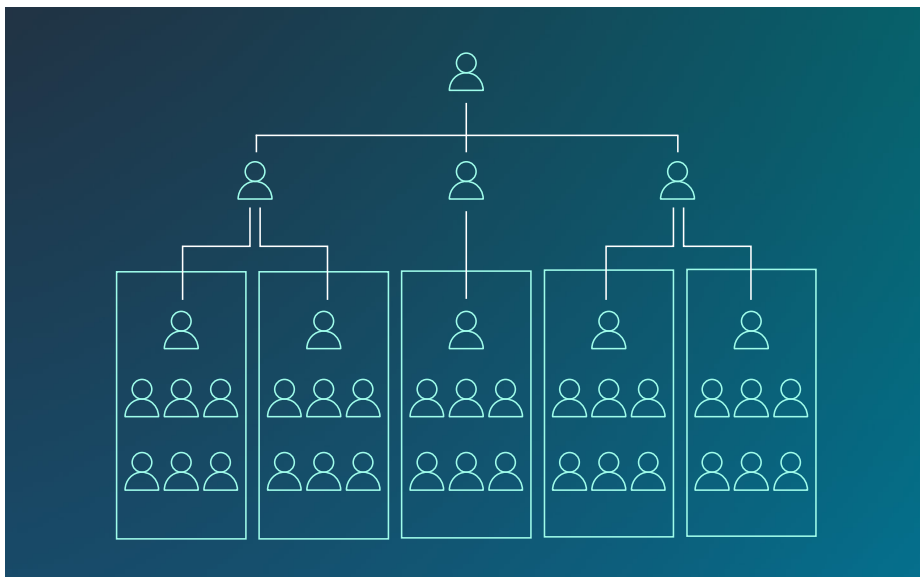
Although those are sensible actions, they are unlikely to effect a lasting transformation. AWS has helped many customers rethink and revamp their operating models to realize cloud technology's full potential. We have learned that businesses who transform their operating model have found four new levers for change.



LEVER #1

From org charts to flywheels

Many transformation initiatives begin by adjusting the organization's structure, using ever-popular org charts. But looking at an organizational hierarchy alone might not tell the whole story. For example, which of the following organizations will perform better?



The structure on the left may appear to have too many layers and too much middle management, which might make it rigid and fraught with overhead. But what if these teams are product-centric and cross-functional agile groups that each own one product and comprise all the skills they need?

And what if the structure on the right, which appears flatter, operates in stages, with one department defining products, another implementing them, and a third testing? Such a setup will cause enormous friction, often leading to the creation of yet another department called “project coordination.” To make matters worse, the development team might be a third party in a different time zone with a fixed scope three-year agreement.

In the abstract, we can't judge the effectiveness of an organization in economies of speed based on static charts because effectiveness depends on *how* change happens. Because change rarely occurs in giant leaps, but through small improvements that reinforce each other, companies need a dynamic view of their organization, one that depicts learning and improvement cycles. At Amazon, we call such a cycle a flywheel.

For example, migrating applications to the cloud provides immediate benefits like improved uptime and lower costs. But viewing those benefits as the sole goal of a cloud journey would mean falling back into the project-oriented mindset of past economies of scale.

Using a dynamic view of the organization, leaders can discover the continuous improvement cycle enabled by cloud technology. Applications running in the cloud yield greater visibility into which parts are most resource-hungry or least stable. Such insights create more opportunities for modernization, for example, breaking up a monolithic application into independent services. These, in turn, give more transparency.

That's the flywheel effect: each revolution produces new insights and opportunities.

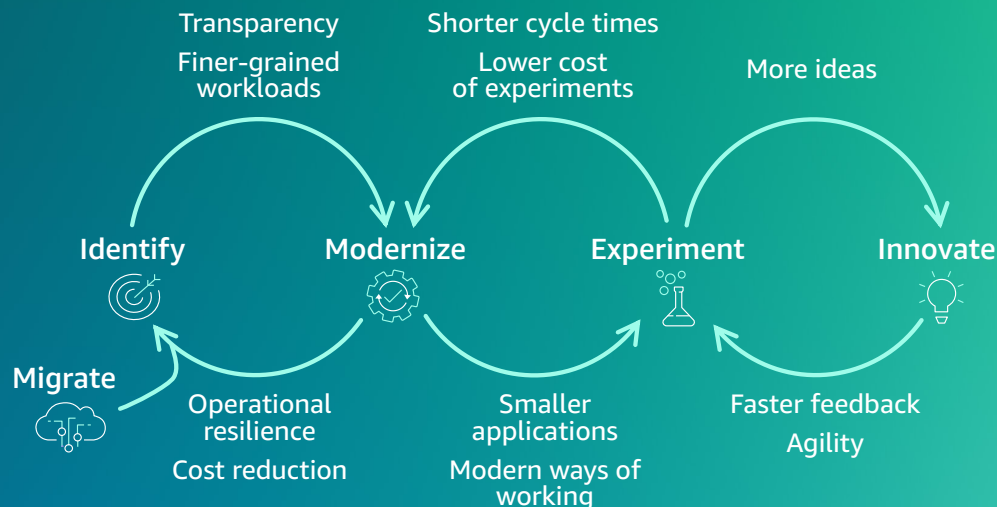
Modern applications using microservices or serverless architectures also have shorter development cycle times and lower experimentation costs, which support economies of speed and, in turn, lead to additional, smaller, lighter-weight applications. Using the flywheel, teams can conduct agile development with smaller experiments and faster feedback, generating additional and more effective ideas.

Once you view your organization as a combination of flywheels, you can identify the organizational misalignments created by traditional static structures. In many cloud journeys, infrastructure teams prepare the initial cloud migrations, while a cloud center

of excellence might identify the workloads to be migrated and modernized by separate application teams. Meanwhile, product teams create new experiments, and an innovation center might be charged with taking advantage of those modernized applications.

Splitting the flywheels across team boundaries demands an enormous amount of coordination. That can cause friction within flywheels. The second major lever an organization that wants to transform must pull is to eliminate this friction.

Cloud migration kicks off continuous improvement flywheels



LEVER #2

From processes to reducing friction

Friction can come from misaligned incentives. For example, the infrastructure teams likely want to lift and shift applications quickly to reduce operational costs, while the applications team prefers to re-architect them for agility. So, if the infrastructure team, citing the business case, pushes back against the applications team, one or the other may look bad. Meanwhile, the product teams' experiments stall.

In rare instances, friction is good; for example, in the brakes on a car. Surprisingly, the primary purpose of brakes isn't to slow down but to allow cars to go faster without crashing in the turns. That's why better brakes can reduce a race car's lap times as much as a more powerful engine.

But racing with the brakes on all the time will damage the car (and lose the race). Unfortunately, that's what too many organizations do. They define processes assuming they will lead to good results, but those processes cause friction, which is the enemy of economies of speed. In a process-heavy environment, every project requires a detailed plan, a business case, a steering committee, weekly updates, review meetings, approvals, and so on. Those processes generate friction and overhead.



Friction: The enemy of economies of speed



The primary goal of reducing friction isn't to become more efficient; it's to enable different ways of working.

In process-heavy, friction-filled environments, project managers believe that by making projects bigger, they will save the company money because overhead doesn't grow as fast as the project — economies of scale thinking. But big projects entail more risk, requiring more controls and thus causing even more friction. This flywheel spins in the wrong direction. As leaders of different parts of the organization optimize locally, excessive friction builds up between them, causing damage globally.

So, instead of just creating more processes, we must lower the friction; for example, through automation and by aligning incentives across the cycle so that every team — infrastructure, applications, and product — can work toward common goals and reap the benefits of a frictionless flywheel.

Less friction leads to smaller projects, fewer risks, and less stress. It also leads to greater agility and helps attract much-needed talent. The primary goal isn't to become more efficient; it's to enable different ways of working.

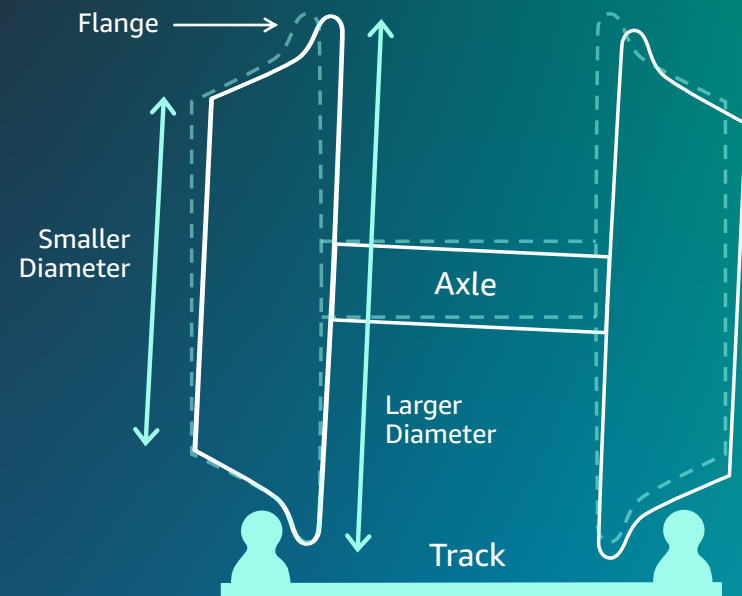
LEVER #3

From guardrails to steering mechanisms

With fewer processes and controls, how can organizations ensure that projects don't derail? The common approach is to put guardrails in place. These keep project teams safe and compliant, just like they keep cars from going down a cliff. However, neither drivers nor projects relish hitting a guardrail — they are for emergencies.

The third lever at leaders' disposal is a different mechanism to keep projects on track. And once again, leaders can learn from the physical world, this time by looking at train wheels. Trains stay on the track, even at high speed. Why? It's not because of the wheels' large inner flange (those are for emergencies or very tight turns). It's due to the shape of the wheels. Train wheels are smaller on the outside than on the inside. If the axle goes off-center, it tilts, and gravity brings it back to the horizontal, centered on the track.

Trains stay on track thanks to a steering mechanism



Cloud computing supports several steering mechanisms that help keep IT projects on track. Increased cost transparency allows teams to cost-optimize and course correct en route to avoid hitting a budget limit that might trigger alarms or shut down services.

Modern application delivery approaches like DevSecOps, supported by cloud services such as AWS Inspector, provide teams with rapid feedback about their security posture, allowing them to make corrections early in the delivery cycle. Encoded engineering principles and architecture frameworks help

teams make consistent decisions, for instance, to balance scalability and complexity.

With these feedback mechanisms, teams don't need quality gates, steering committees, or managers and architects in every decision loop, causing bottlenecks and rework. Failing a final security or architecture review is like hitting a guard rail; better than driving off a cliff but still not pleasant. The feedback mechanisms, however, work like train wheels to keep teams on track — without hitting any guardrails.

Cloud steering mechanisms reduce bottlenecks and rework



Cost transparency

Share resource cost allocation and elasticity



DevSecOps

Automate enforcement and awareness of security controls



Engineering principles

Encode preferred trade-offs for decision consistency



Well-architected frameworks

Benefit from vast experience in six best practices pillars



LEVER #4

From granting exceptions to autonomy

A question that reveals a lot about an organization is how long it takes to procure a server. In traditional on-premises computing environments, the answer is often “Much too long.” But the honest answer is usually, “Well, it depends on who’s asking.”

Although this way of working is occasionally characterized as collaborative (“you need to know whom to talk to”), I consider it a black market. Black markets that rely on relationships hinder broad access to resources. That degrades efficiency, impairs productivity, damages morale, and imposes unnecessary costs. Despite good intentions, granting exceptions to some teams, for example, via an innovation center, is a slippery slope towards a black market where different rules apply to different people.

The fourth lever a company should pull is to democratize access to resources and enable all teams to innovate and make decisions autonomously. The best ideas might come from unexpected places, and if those teams don’t have access to the resources needed for an experiment, those ideas might be lost forever. The cloud democratizes access with an abundance of tools that everyone can use.

It’s easy to mistake autonomy for everyone doing what they think is best. Sadly, though, that isn’t autonomy; that’s anarchy. Accordingly, autonomy needs an architecture to guide teams while they are moving fast.

To begin, teams share a vision to achieve alignment on common goals. That's why autonomous teams require more leadership, not less, so they have a clearly articulated strategy and principles of engagement. Then, autonomous teams can choose their tactics, but they work on a platform that avoids duplication. Cloud platforms are ideal for this as they provide everything needed for the undifferentiated heavy lifting, allowing teams to focus on generating business value. Last, rapid feedback tells teams when they're heading in the right direction and allows them to correct course when they aren't.

The good news: the cloud provides the controls and transparency for autonomy, not anarchy.

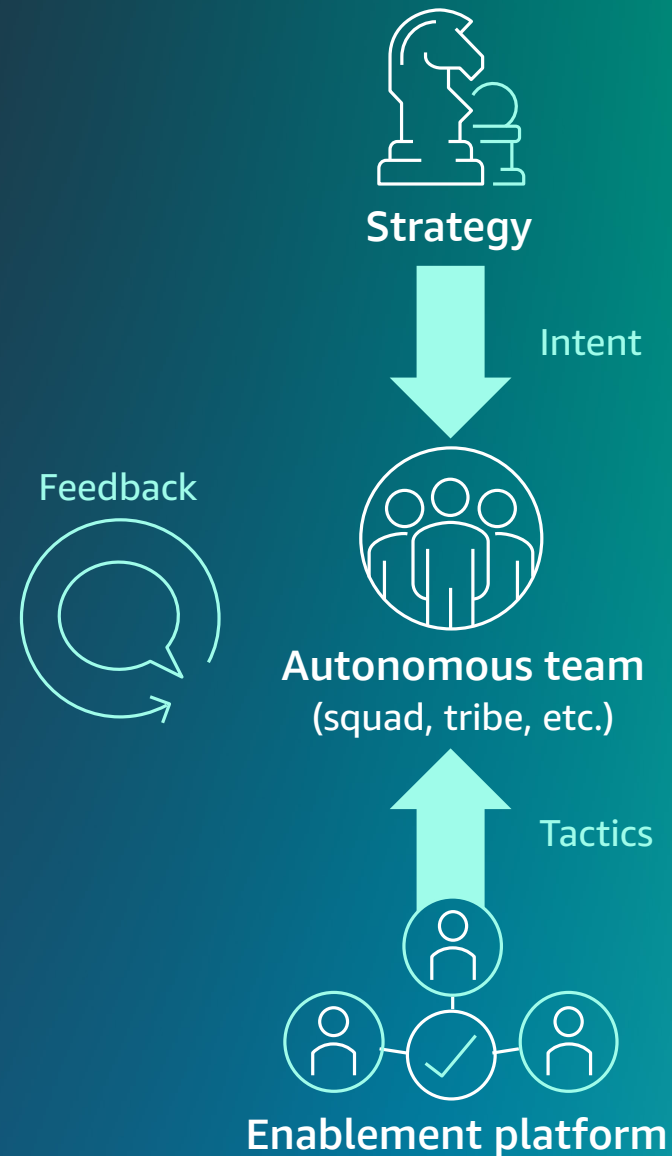


Everyone doing what they think is best isn't autonomy. That's anarchy."



Autonomous teams require more leadership."

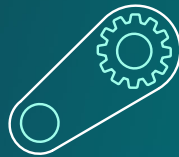
Autonomy has an architecture



The journey to economies of speed



Define
structure



Create
flywheels



Improve
continuously



Follow
process



Reduce
friction



Move
faster



Impose
constraints



Deploy
mechanisms



Stay on
track



Grant
exceptions



Foster
autonomy



Decide
locally



Stay in touch with the IT engine room

New levers also mean rethinking leadership. Leading autonomous teams toward continuous improvement flywheels is a relatively novel skill. In the past, a good leader defined a goal, set a direction, and tracked progress with high-level KPIs. In economies of speed, static goals and rigid long-term plans are no longer meaningful and might equate to flying blind.

Remember the architecture team that wanted to reduce costs by lifting and shifting their data center, including all its problems, to the cloud? That enterprise may get a better data center but miss out on the cloud's benefits like agility or low cost of experimentation.

In economies of speed, technology architecture can fuel a company's success, or present a major hindrance. That's why the IT engine room has to be closely connected to the business and vice versa. Technical staff learn from their leader how the business's systems function and which business priorities drive technical design decisions. Meantime, business leaders can no longer treat the engine room like a black box but must understand IT architecture decisions that will determine the business's ability to maneuver.

Lead with purpose and empathy

To change employees' behavior, people-management systems must change. Employers don't have to throw money after everyone, but they do have to improve performance management so that high performers are recognized and rewarded. Reducing friction and bias allows people to realize their true potential. It's bad not to recognize high performers, but it's worse to hold back a potential high performer by internal friction.

These factors contributed to the Great Resignation — employees changing jobs en masse to acquire the employers and systems they want to work with, those with purpose and empathy. Modern leaders need to create role models and reward desired behavior, not paste posters with motivational slogans around the office. They must instigate genuine change, not relabel existing roles. Calling project managers SCRUM masters won't do much to transform an organization.

Leaders must understand the engine room's complex systems while being sensitive to those who toil in them. Specializing in one while passing on the other will no longer suffice. Instead, leaders must stretch in both directions.



Leaders must understand the engine room's complex systems while being sensitive to those who toil in them. Specializing in one while passing on the other will no longer suffice. Instead, leaders must stretch in both directions."

Living in economies of speed

What have we learned?

The cloud matches the mental model of economies of speed: thinking in rates and consumption, not in long-term absolutes. Transformation can't be achieved by setting existing levers to new positions; you have to rethink them, moving from org structures to positive flywheels, designing processes to reduce friction, and avoiding close encounters with guard rails by implementing self-centering mechanisms that keep operations on track with minimal intervention. And as tempting as it is to create exceptions, avoid creating black

markets by democratizing access to technology. That fosters autonomy and allows decisions to be made rapidly by the people who know best.

These prescriptions sound straightforward, but they aren't easy to implement. Using new levers to steer an organization to work differently is tremendously hard, making strong leadership more crucial than ever.

When you have all that, you can box up your CDs and start streaming.

To continue the conversation, connect with AWS



Follow AWS Executive Connection on LinkedIn for more insights ›



Listen to the AWS Conversations with Leaders Podcast ›



Read the latest perspectives on AWS Executive Insights ›

About the author

Gregor Hohpe, AWS Senior Principal Evangelist

As an AWS Senior Principal Evangelist, Gregor helps IT and business leaders connect their technical decisions to their business strategy. Fully harnessing the potential of new technology, he believes, requires unlearning current assumptions and rethinking levers for change.

A large part of Gregor's career has been spent on large-scale IT initiatives, including migrating on-premises data centers to the cloud, at Allianz, the Government of Singapore, and Google. His past roles — start-up co-founder, system integrator, consultant, software engineer, trainer, chief architect, and CxO advisor — have allowed him to see IT from almost every angle. He often draws on that background when talking with customers, as he's faced many of the same challenges that they're dealing with. Outside of work, Gregor likes to be outdoors and in motion, practicing a variety of sports, from snowboarding and mountain biking to sailing or hiking. As a former IT architect, he also appreciates learning and going out to see "real" architecture and design.

About AWS

For over 15 years, Amazon Web Services has been the world's most comprehensive and broadly adopted cloud offering. Today, we serve millions of customers, from the fastest-growing startups to the largest enterprises, across a myriad of industries in practically every corner of the globe. We've had the opportunity to help these customers grow their businesses through digital transformation efforts enabled by the cloud. In doing so, we have worked closely with the C-suite, providing a unique vantage point to see the diverse ways executives approach digital transformation—the distinct thought processes across C-suite roles, their attitudes and priorities, obstacles to progress, and best practices that have resulted in the most success.

