InfoWorld DeepDive

Moving legacy applications to the cloud

Making the most of existing assets in a cloud world

Deep Dive

Moving legacy applications to the cloud

Enterprises accumulate hundreds of applications essential to the business over the years. Many can benefit from a move to the cloud

BY DAVID S. LINTHICUM

Like all things that are cheap and easy, "lift and shift" **comes with trade-offs.** Applications that are not designed or refactored specifically for the cloud cannot take advantage of cloud-native features. Lift-and-shift applications may not perform as well **and could actually cost more to operate over time.**

Much of the conversation about commercial use of the cloud is focused on the doings of large, cloud-native corporations. Witness, for example, the recent headlines generated after online gaming giant Zynga announced that it would abandon a $100 million data center it stood up just four years ago to move its infrastructure back to Amazon's public cloud.

However, an even more momentous transition is happening below the radar, as rank-and-file enterprises begin to move millions of legacy applications from self-managed, on-premises deployments to public, private, and hybrid clouds. How big will this trend be? According to 451 Research, within two years, 34 percent of enterprises will have 60 percent or more of their applications on a cloud platform. To be sure, some of those are new "cloud-native" applications, but most are legacy, with the majority being more than 10 years old.

Enterprises migrate applications to cloud-based platforms for several reasons:

- The perception that public cloud platforms are mature enough to provide a stable and secure platform for enterprise applications.
- The desire to reduce costs, which includes reduced requirements to purchase more hardware, software, and data center space.
- Public clouds' ability to provide a platform analog for the applications that doesn't require changes to the code or data.
- The need to support centralized application deployment to better support the emerging DevOps models.

Migration to the cloud is basically a platform change. In many instances, the applications are simply ported directly to public clouds. This is known as "lift and shift," and it is the least expensive path to move an application to the cloud. Like all things that are cheap and easy, this approach comes with trade-offs. Applications that are not designed or refactored specifically for the cloud cannot take advantage of cloud-native features. Lift-and-shift applications may not perform as well and could actually cost more to operate over time.

However, most enterprises are not looking for the best way to run their applications in the cloud, but the quickest (and cheapest) path to doing so. Thus, massive "lift and shift" migrations are happening today. It is not unusual to have more than 1,000 applications and associated data move in a single year.

The data on lessons learned is beginning to come in. As it turns out, legacy application migration to the cloud is not as straightforward as the cloud providers would lead you to believe. There are many mistakes that can be made, and in many instances, the migration to the public cloud could actually cost you more money and increase your risk of failure.

In this report, we'll take a deep dive into the major aspects of legacy application migration to the cloud. We'll provide you with the processes and details you'll need to select the right applications to migrate, including how to create a business case tailored for your business, and advice about how to deal with the important technical issues, such as performance,

Deep Dive

security, governance, and business continuity and disaster recovery.

Why the cloud is an option

As shown in Figure 1, the cloud is a platform to host existing applications, and it brings new advantages as well. For instance, applications have access to resources on demand, thus they can scale out to the capacity that is needed. Forget about waiting weeks or months for hardware or software upgrades. Other advantages include the ability to support stateless applications, distributed data, self-healing applications, and the ability to pay only for the resources you use.

These new architectures do require some level of change to capture their benefits. A core decision is what changes you should make to your legacy application to ensure it lives well in the cloud.

Figure 2 lists some of the choices that you have when migrating a legacy application to the cloud. They include replacing the application with a third-party SaaS provider, such as Salesforce.com, or changing the application from the ground up to leverage the cloud more effectively, a process known as "refactoring."

Less aggressive approaches include replatforming, which does not require as many changes as refactoring, and rehosting (basically "lift and shift"). Finally, it is always possible that a move to the cloud isn't advisable. In that case organizations can choose to leave the legacy application alone ("retain") or discontinue it ("retire").

The process requires evaluation of each legacy application: assessing its breadth and depth as well as its value to the business. Once you determine that the application is a good candidate for the cloud, you must then select the best approach from those presented in Figure 2. Then, consider other items, from security to operations.

Generally speaking, the older the legacy

Figure 1: Cloud brings new advantages to traditional applications

THE 'OLD WORLD'		THE 'NEW WORLD'		THE TARGETS
Traditional Application Architecture	**Refactor**	**Cloud-aligned Application Architectures**	**Continuous delivery**	
• Scale up	→	• Scale out	→	amazon web services™
• Monolithic	→	• Distributed	→	Windows Azure
• Stateful	→	• Stateless	→	Google Compute Engine
• Infra dependent	→	• Infra agnostic	→	openstack
• Fixed capacity	→	• Elastic capacity	→	rackspace
• LAN, SAN	→	• WAN, location transparency	→	SaaS
• Latency intolerant	→	• Latency tolerant	→	terremark
• Tightly coupled	→	• Loosely coupled	→	vmware
• Consolidated/clusterd DB	→	• Shared/replicated/distributed DB	→	
• Rich/chatty client	→	• Mobile/thin client	→	
• Commercial licenses	→	• Cloud PaaS/open source	→	
• Infr supported availability	→	• App supported availability	→	
• Manual build/deploy	→	• Automation	→	
• Manual fault recovery	→	• Self-healing	→	
• Active/passive/DR	→	• Active/active	→	
• Perimtere security	→	• Defense in depth	→	
• Allocated costs	→	• Metered cost	→	

Deep Dive

Figure 2: There are a number of ways to migrate legacy applications, ranging from replacing the application to refactoring.

Replace	• Replace application with third-party SaaS or outsourced service • Build cloud-native apps with similar/improved characteristics and features
Reuse	• Develop and productize common business and technical services • Consolidate similar applications and services
Refactor	• Target changes to the application to address issues; leverage the cloud • Revise applications to leverage common services
Replatform	• Move to more cloud-aligned technology and platform services • Integration with cloud operations and monitoring
Rehost	• Lift and shift, minimizing changes, virtualization replatforming • Physical to virtual to cloud (P2V2C), Virtual to Cloud (V2C)
Retain	• Leave application as is; do not move to cloud
Retire	• Application end of life

- Determining the right migration strategy for your app depends on its level of cloud alignment, cloud readiness, potential benefits achieved from migrating, and risks
- Not everything will migrate
- Migration strategy selection also depends on the target cloud endpoint
- Public and other external cloud endpoints may require 2x-10x effort
- Rehost and replatform strategies minimize change for applications that are either cloud ready, or have higher risks or lower sloud benefits
- Refactor and reuse strategies accept the cost of change in return for business benefits
- There are patterns, processes, teams and tools to support each strategy and each cloud endpoint

application, the more difficult it will be to move it to the cloud. Moreover, the more the application is tightly coupled to the data or other application components, the more difficulty is introduced. Thus, good candidates are applications that are less complex, loosely coupled, and less than 10 years old.

Selecting the right applications

To evaluate a portfolio of legacy applications to move to the cloud, you should develop a process to properly consider each application. Figure 3 depicts a basic process: First, create a business case for each application, consider the funding available, determine the breadth (the amount of functions that the application performs) of the application, opportunities for modernization, the right migration approach to properly port the application, and then approaches to operations.

For example, let's take an inventory control application that is currently operating on an IBM mainframe. We would first determine what value would be brought to the business, in cost savings and productivity, by migrating that application to the cloud. We would consider how the application is architected and built, any areas of the application that need to be improved, and the best approach to migrating the application to the cloud.

The idea is to arrange the legacy applications by their importance to the business, ease

Figure 3: When looking at your application portfolio, you need to consider several steps.

Deep Dive

of moving to the cloud, and the cost of any changes that would occur if the application were moved. Using this kind of a ranking system, you can determine which applications should take priority over others and even which applications should not move or be replaced.

It is important that those who choose the applications to move to the cloud properly analyze the applications with the goal of understanding enough about the application to make a proper assessment about whether and how the application should move to the cloud. Typically, mistakes occur when the enterprise does not take the time to understand the breadth and depth of each legacy application, leading them to underestimate the complexity and cost of cloud migration or conversely, to overestimate the value of leaving an application in its existing, on-premises deployment.

Applications must be studied closely to identify their key function and architecture, the technology they rely on, current configuration, and future development plans.

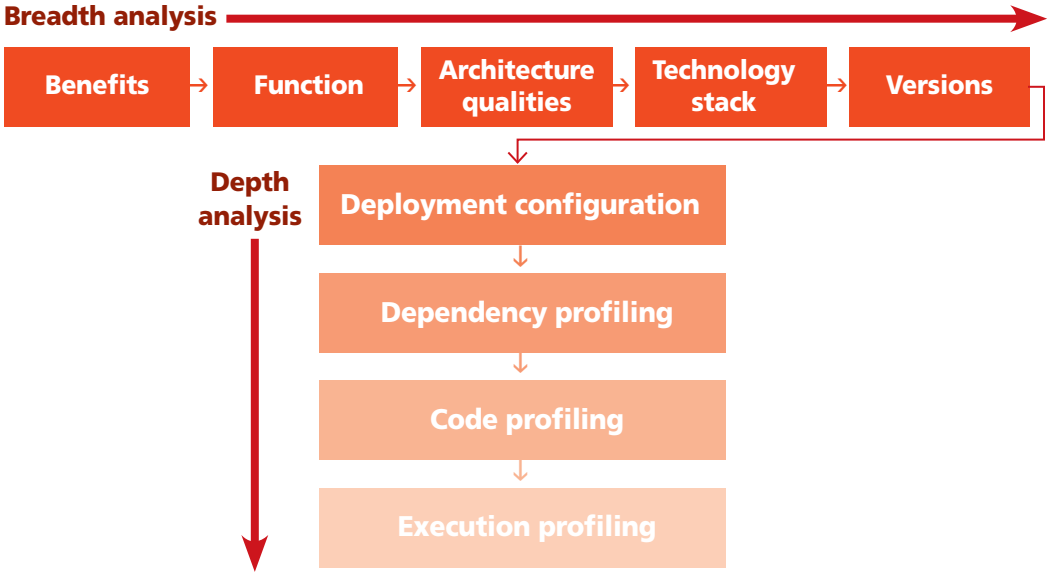
To understand both the breadth and depth of each app (see Figure 4), you must determine the benefits, function, configuration, and other technical and non-technical details that make up the essence of the application. For example, determine the number of function or object points that exist in the application. An application's complexity is a good indication of the amount of work required to migrate the application to the cloud. Skipping this step, which is a common mistake, can result in missing aspects about the application that may make it a good candidate for the cloud or not.

Considering cost

It goes without saying that "cloud" is not an unmitigated good. Cloud migrations must be justified in the same manner as any other technology investments or business decisions. Specifically, the benefits of moving any application to a private cloud must be weighed against the costs of doing so.

How does an organization determine what those costs are? Figure 5 depicts a simple model that looks at the existing costs of maintaining legacy applications and the cost of maintaining the same capacity in the cloud. This model provides a baseline of "savings" over a 3-year period against which we can balance the cost of migration, including any changes to the application that need to be made. Again, these numbers will vary greatly from application to application.

Figure 4: To understand the breadth and depth of each application, you need to become an expert in what the application is and does.



Deep Dive

Figure 5: A simple cost analysis allows you to quickly determine the amount of money that the cloud migration could save.

	Average migration	Legacy	Cloud	3-yr per-unit savings	Total 3-yr	3-yr savings
Total servers	Cost/server	Cost/server/ month	Cost/server/ month	Cloud vs. legacy	Operational savings	Less migration costs
10,000	$623	$300	$250	$1,800	$18,000,000	$11,765,798
10,000	$623	$400	$250	$5,400	$54,000,000	$47,765,798
10,000	$623	$500	$250	$9,000	$90,000,000	$83,765,798

Considering performance

Many consider the cloud a fix for performance problems since you can provision resources on demand and gain access to a massive amount of compute and storage services to support the application.

The truth is that application performance is more a product of the architecture of the application than the resources available to run it. The effective use of native platform resources is an important consideration when weighing whether to migrate a legacy application to the cloud.

Unfortunately, many application performance issues are the product of inefficient application architecture or design, not coding errors that are easy to fix. A legacy application that performs well on a legacy platform, may not perform any better—or even as well when moved to the cloud because of dependencies that are designed into the applications.

Common performance issues with legacy applications include:

- Data that is tightly coupled with the application, making it difficult to separate the workloads to balance performance and scale the application.
- Applications that are "chatty" when communicating with the user or other processes or data. These applications are vulnerable to network or Internet latency issues.
- The application uses platform-specific and raw I/O, and may not perform well on cloud systems that have well-defined APIs to manage access to resources.
- The language processing system may not meet performance expectations, such as may be the case with languages such as Cobol.

The trick to managing performance in a cloud migration is in really identifying and understanding the root cause of application performance issues. As noted, these often lie in the architecture of the application itself, rather than in discrete flaws in the application logic.

By understanding the root of application performance issues, you'll be standing on solid ground when you estimate the level of effort to correct those performance issues in concert with a move to the cloud. In some cases, the level of effort and the cost to repair those issues will be too high and you will conclude that a legacy application is not a good candidate for migration to the cloud. But making that decision up front, rather than after the fact, will save you both time and money.

Modifying the legacy applications to take advantage of native features of the hosted platform is another way to increase performance. Typically, you can access these features via layers, including the topmost virtual platform or operating system and underlying resources accessible to the virtual machine your application is running on, such as storage and data. Finally, cloud-native services, such as provisioning and tenant management, will vary from provider to provider.

Deep Dive

Typically, cloud-native applications leverage the cloud-native features and APIs of the environment in which they run. These tightly coupled integrations make better use of cloud resources, resulting in an application that runs more efficiently and uses less of the underlying hardware and software. Because resource use is a major factor in the cost of application hosting, better-managed hosted applications are also less expensive hosted applications.

When it comes to migrating legacy applications to the cloud, it is often the case that cloud-based instances of an application can also access the cloud-native features of the public cloud services to provide better performance than would be possible with non-native features. For example, in migrating your application, you might want to find a way to leverage a native I/O system that works with an auto-scaling and load-balancing feature in your hosted environment.

It is important to **understand the specifics of your application's security and authentication scheme** and to make informed decisions about how migration to the cloud will impact those.

Considering security

Application security is an area that warrants special attention as you weigh whether to migrate your application to the cloud. In fact, uncertainty about the security of deployed applications and data—for example, as a result of a data breach - is among the most oft-cited concerns of cloud migration.

The truth is that cloud deployments are no less secure than traditional, on-premises application deployments and may even offer substantial benefits over self-hosted applications. Still, as with other challenges that stand between a legacy application and the cloud, it is important to understand the specifics of your application's security and authentication scheme and to make informed decisions about how migration to the cloud will impact those.

In general, you have two choices for data security when moving legacy applications to the cloud. First, you can leave the application as is and retrofit cloud security services to the application and its data. Unfortunately, this is a blunt instrument that typically means restricting access at the level of the virtual machine- or virtual storage instance. Although such restrictions are effective in securing access, they can be ham-fisted, creating a "you're in" or "you're out" type of security solution that doesn't allow you to set granular access policies, and that isn't likely to scale to support scores, hundreds, or thousands of users.

The second approach is to modify your application to leverage more fine-grained, application-level security services, such as those that are entity based. This allows you to manage access to specific application services and data in ways that give you greater control over how the application is used by individual users. However, this typically requires modifications to the application to leverage this type of security, such as SAML (Security Assertion Markup Language). SAML is an open standard used for federated identity.

A move to the cloud won't be a magic elixir that makes insecure applications suddenly bullet-proof. It goes without saying that applications that were built without much thought or rigor around security won't be as secure, no matter where they are deployed.

It is possible that the act of migrating an application to the cloud could make both the application and the data it handles more vulnerable to compromise through modifications to the existing application and the risks that go along with deployment on a public, private, or hybrid cloud environment.

The best practice is to create a well-defined and executed security strategy that takes into account both existing risks to your application and additional risks that accompany deployment in the cloud. Be careful to choose the right enabling technology (encryption, HIDS [host-based intrusion detection system], security testing, etc.) and leverage that security plan across your legacy applications, modified or not.

Consider access control and entitlement management

Extending our security discussion to access management services, you need to consider two core concepts: entitlement management and user provisioning and de-provisioning.

When considering entitlement management, it's a matter of allowing control to ensure that systems and applications have the ability to leverage externalized entitlement management services. In general, this means designing your

Deep Dive

application and systems to provide and consume XACML (eXtensible Access Control Markup Language) type information. XACML is a general-purpose standard for describing policy management and access decisions and may be the best path for providing access control to your migrated legacy application.

When leveraging XACML you're implementing a common authorization standard across all applications by providing a standardized language, a method of access control, and policy enforcement. This is in contrast to SAML, which is an open standard used for federated identity.

User provisioning and de-provisioning are key elements of identity and access management services. This is the process of allocating users and applications and granting identities access to applications and application components. The process of provisioning and de-provisioning will set the initial entitlements and link them to an identity.

Some suggested courses of action:

- Understand your security and governance requirements for each legacy application and data store. Many of those who deploy security around cloud or traditional systems don't understand what problems they are attempting to solve. You need to define those up front, beyond the application.
- Consider the challenges and opportunities in migrating from on-premises (LDAP) identity and entitlement management and the many options for doing so in the cloud. What needs to change, and how do you morph over to the new technology?
- Understand that controlling access is much more important than the location of the data. Look at how the data is accessed, and look specifically at opportunities to breach. Again, most data breaches occur around finding vulnerability, no matter if it's cloud-based or on-premises.
- Finally, vulnerability testing is an absolute necessity, no matter if you're testing the security of cloud-based or traditional systems. Untested systems are unsecured systems.
- Data should be encrypted in flight and at rest. If you leverage encryption, understand the impact it will have on the performance of the application.
- Make sure to consider compliance or other legal issues around how data is managed. This will determine how much work you'll need to do to relocate the legacy application to the cloud, thus the cost and risk. Make sure your target public cloud providers support the required compliance standards.

At every stage, the goal of information governance policies should be to maximize the value of enterprise data and properly manage the risks associated with using it.

Considering information governance

Stepping back, information governance policies should also be an area that warrants analysis and planning ahead of any cloud migration. Information governance is a discipline that encompasses security but also comprises related disciplines such as records management, compliance, risk management, and so on.

At every stage, the goal of information governance policies should be to maximize the value of enterprise data and properly manage the risks associated with using it. As with security, the approach to managing information governance in the cloud should be to create a common governance strategy that encompasses all your cloud-based applications and data. This is not about creating an on-off governance system from your migrated legacy system, but a governance approach and strategy for all cloud-based applications. Thus, you're not just solving the governance problem for a single application, but for the new public cloud platform which will be the host of many applications.

In the context of the cloud, it is important to realize that the impact of information governance activities might be felt at different layers of the cloud "stack": from the level of individual data to the application or service level down to the machine level. For the purposes of migrating applications to the cloud, it is important to realize that legacy applications that do governance at the service level may need to be modified to work with cloud-native resources. This doesn't need to be a "deal breaker," but it could affect the cost and risk associated with a move to the cloud and change the business case for making such a move. In considering a move to the cloud, you need to select technology that can manage security and governance policies that meet the

Deep Dive

requirements of your application.

In cloud-based deployments, governance breaks down to service-based governance, or managing access to services or APIs (see Figure 6). Resource-level governance manages resources such as cloud brokers and manages the provisioning and de-provisioning of those resources. Finally, there are governance systems that are native to the cloud provider, either service- or resource-based.

Considering business continuity and disaster recovery

Migrating to the cloud offers many advantages in the areas of business continuity and disaster recovery. After all, an important component of the value proposition of cloud providers is the centralization of otherwise resource-intensive activities such as 24/7 database and server management, disaster planning, backup and recovery, and so on.

Cloud deployments also introduce new risks that on-premises deployments do not. Among these are a reliance on third-party infrastructure, the performance impact of adjacent applications in hybrid or public cloud environments, and other "externalities" such as weather-related disruptions, denial-of-service attacks, hardware failure, or law enforcement actions.

As with applications deployed within your own data center, you must ensure that your cloud-hosted legacy applications can survive outages and other unanticipated events.

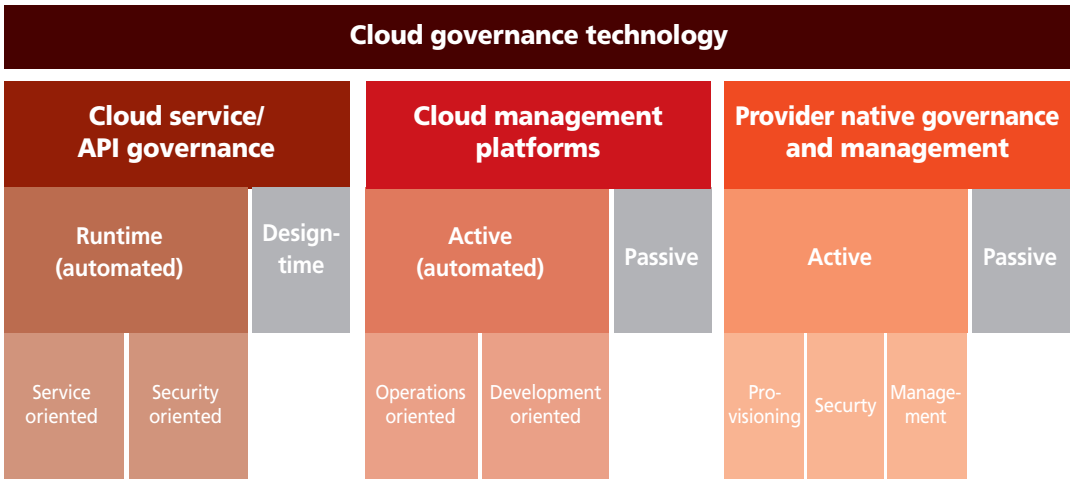
Most public and hybrid clouds support redundant services, either as a core element of the cloud service itself or as an add-on feature. Redundancy services keep your application and data operating instantiated in at least two different points of presence, with one location available as a hot fail-over. In the case of an interruption at a primary cloud center, your application will fail over to the secondary location transparently, without causing any interruption in application services for users.

In considering migrating legacy applications to a cloud provider, you can rely on the cloud service provider to handle resiliency services as a paid service. Alternatively, you can build business continuity and disaster recovery services directly into your application. For example, you could design your application so that data is written to redundant locations by the application itself or to fail over to backup instances of the application that are kept running at all times.

Taking the plunge

In any business, the least complicated path is always to change nothing and maintain the status quo. But choosing to do nothing isn't without costs. That's especially true when the question is

Figure 6: Cloud governance around legacy application migration requires consideration of service and resource governance.



Deep Dive

whether or not to embrace cloud computing.

As we've noted, the case for moving your legacy enterprise applications from your data center to the cloud is persuasive. Private-, hybrid-, and even public-cloud deployments can dramatically lower the cost to operate your legacy applications, while improving the availability and reliability of those applications. It also provides an opportunity for your organization to leverage the resources and infrastructure of larger providers at a price that is both reasonable and predictable.

Still, migrating any application to the cloud isn't a decision that should be made cavalierly. Although many enterprises are quick to embrace the cloud, the reality is that the mileage you get out of a cloud migration will vary a great deal depending on what your application does, how it was designed, and how well your organization is able to identify and address issues that are likely to arise as a result of the cloud migration.

In this paper, we've outlined some key areas to consider as you assess your application infrastructure with an eye to leveraging cloud deployments. We believe the best approach is to create a uniform cloud strategy for application cloud migration that starts with the business case for relocating to the cloud, moves on to examine the specific technical aspects of the application, and then weighs the probable costs of addressing any problems that are likely to arise in making your legacy application cloud-ready.

Among other things, we note that it is crucial to understand the ways in which cloud migration will save your organization money, but also to weigh that against less easily quantified factors such as the design and function of your legacy applications, performance issues, security and authentication, data governance, as well as the likely costs of migration before making a final decision.

The truth of the matter is that at the end of this process you may find that many legacy applications will not be cost effective to move to the cloud. But you may be surprised to realize that applications you might assume should not move will actually add more value deployed on the cloud.

While it is true that there are often considerable development costs that must be borne by organizations that want to (properly) migrate legacy applications to the cloud, it is also true that cloud providers are becoming better at supporting legacy application migration, lowering the cost for individual customers. You can be confident that this trend will continue and accelerate.

Migrating any application to the cloud isn't a decision that should be made cavalierly. Although enterprises are quick to embrace the cloud, the reality is that the mileage you get out of a cloud migration will vary a great deal.

ABOUT THE AUTHOR

Leading technology publications frequently name David S. Linthicum among the top 10 enterprise technologists in the world. He is a true thought leader in the industry and an expert in complex distributed systems, including cloud computing, data integration, SOA (service-oriented architecture), and big data systems. As the author of more than 13 books on computing with over 3,000 published articles, as well as radio and TV appearances as a computing expert, he is often quoted in major business and technology publications. In addition, David is a frequent keynote presenter at industry conferences, with more than 500 presentations given in the past 20 years.

David's industry experience includes tenures as CTO and CEO of several successful public and private software companies, and upper-level management positions in Fortune 100 companies. He has delivered over $2 billion in value by transforming companies from traditional to innovative technologies, and moving them to lucrative exits that benefitted investors, employees, and customers.