

Application Modernization Should Be Business-Centric, Continuous and Multiplatform

IS THIS CONTENT HELPFUL TO YOU?

YES

NO

FOUNDATIONAL Refreshed 11 December 2020, Published 15 August 2019 - ID G00430084 - 11 min read

By Thomas Klinect, Stefan Van Der Zijden



Modernization of complex, aging application portfolios tends to focus on the oldest platforms rather than on the most valuable business capabilities. This research will help application leaders adopt a business-focused, multiplatform modernization strategy to maximize value.

Overview

Key Challenges

- Whereas IT sees applications as the unit of management, business stakeholders value capabilities that often span multiple applications, platforms and ecosystems, which are often laden with technical debt.
- Enterprise portfolios are far too large for a cost-effective and comprehensive inventory of all applications, in all locations, and how the applications and database are intertwined during a workflow.
- Huge modernization programs are hard to greenlight and often fail to meet the planned outcomes with more than half never reaching the first significant program milestone.

Recommendations

Application leaders responsible for application and product portfolio governance should:

- Modernize valuable business capabilities through an end-to-end product life cycle, across all applications, platforms and ecosystems that comprise the complete capability. Avoid focusing modernization efforts on a single application, platform or technology.
- Perform a targeted but deep assessment of the applications, components, services, data and APIs that comprise these valuable business capabilities, to set modernization priorities to align with and support digital business strategy.
- Budget and plan modernization work for the entire product life cycle and not as an occasional exception, by performing continuous modernization which will proactively reduce technical debt.

Strategic Planning Assumptions

By 2025, 90% of current applications will still be in use, and most will continue to receive insufficient modernization investment.

By 2025, technical debt will continue to compound on top of existing technical debt consuming more than 40% of the current IT budget

Introduction

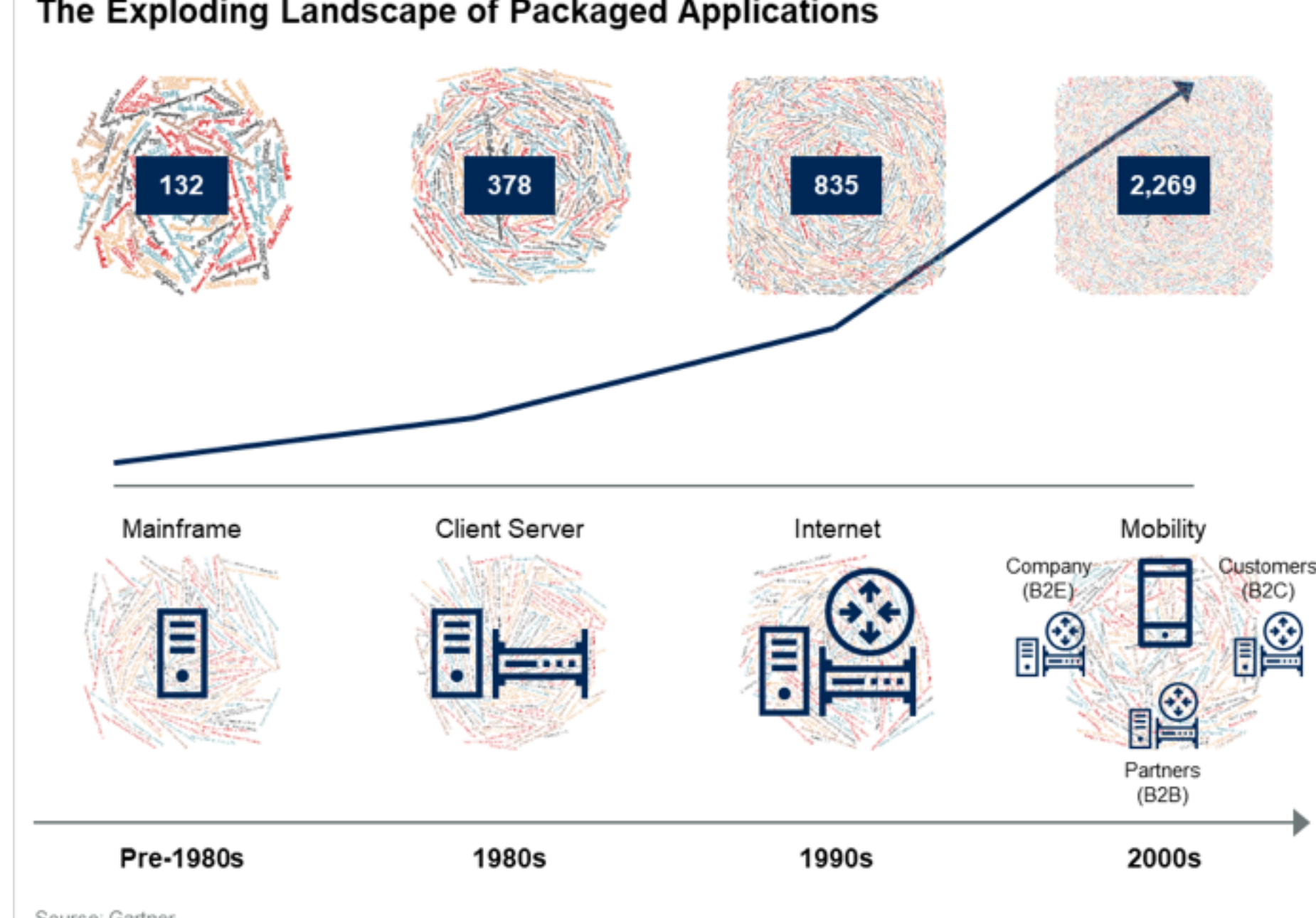
Application leaders are struggling to manage an increasingly complex landscape of applications, components, services, data and APIs across a wide range of platforms and ecosystems, including:

- Software or hardware platforms nearing the end of their support life – for example, older Java application servers, custom-made applications running on mature platforms like the mainframes or minicomputers.
- Packaged business applications (COTS) which have reached the limits of their business value, or need an expensive upgrade (such as EBS to Oracle Cloud) which is unlikely to produce a return on the required investment.
- Older core applications supported by teams of aging developers with hard-to-find skills.
- Line-of-business applications (including SaaS) bought or built by "shadow IT," (better known as "business IT") but turned over to the application group for ongoing support and maintenance.
- Websites and mobile apps built on 10- to 15-year-old technology, delivering user experiences need a design refresh.
- Large volumes of data which is stored on a wide range of file systems, in multiple databases, managed by multiple data warehouses or data marts and cloud storage; several of which are no longer considered strategic for the firm.

Most application leaders do not have a complete inventory of all application assets for the enterprise, making it impossible for them to manage this aging estate that is loaded with technical debt (see "A Primer on Technical Debt"). And even among those lucky few organizations that have an inventory, precious few know the business and technical fitness of these assets (see "How to Assess Your Current Application Portfolio Using Fitness and Value Review Processes").

Past efforts at compiling an application inventory have often floundered or are partially completed because of the daunting size of the task. The proliferation of software like Excel allows business units to create an application which is not accounted for in the applications inventory as Excel hides the application from view and software scanning will not find such applications. The complexity of corporate systems continues to increase over many generations of application and platform technology (see Figure 1). The software landscape of packaged applications exploded from approximately 132 applications (pre-1980) to over 2,269 applications by 2017 – and this doesn't even include the equivalent rise in the numbers and complexity of custom-developed software.

Figure 1. Application Leaders Manage Complex Application Estates



One key reason for past failures is the IT-centric nature of this approach to modernization. IT tends to focus more on costs whereas business stakeholders are more focused on value. Modernization efforts focused on cost reduction for specific platforms, technology trends or applications, inevitably disappoint due to the high cost of any modernization effort, coupled with the lack of focus on value to the business.

Faced with this situation, business and application leaders often prefer to ignore the mounting debt and instead focus on adding new applications and platforms to the mix, further exacerbating the problem.

The application organization is ripe for a new solution to this conundrum.

Analysis

Modernize Business Capabilities Through the Entire Product Life Cycle

Application leaders have grown accustomed to managing applications that only loosely relate to the value streams and business capabilities they support. Figure 2 shows this IT-centric approach, which may suffice when the mix of work is mainly implementation or basic maintenance, but tends to be wasteful when it comes to major upgrades or modernization. Figure 3 depicts a more business-centric approach.

To maximize the business impact of their efforts, application leaders should:

- Modernize critical business capabilities, not only specific applications. Analyze how well the application portfolio currently supports these capabilities across all platforms, pace layers, APIs, services, ecosystems and data resources. Avoid focusing on applications and platforms due to their IT characteristics, such as age or cost.

See "What Is Gartner's Pace-Layered Application Strategy and Why Should You Use It?"

- Manage collections of business capabilities as products or services. Once you have embraced this outside-in, business-centric approach for prioritizing your modernization focus, the product-centric model becomes the most natural way to organize your modernization efforts.

See "Reinventing Applications as Products for the Digital World."

- Manage products across their full life cycle rather than only during related projects. A full life cycle perspective enables IT and business leaders to take all relevant factors into account, such as technical debt or product obsolescence, when laying out the product investment roadmap. This avoids the blinkered thinking that leads to applications with insufficient maintenance budgets, or wasted spending on applications that are well past their sell-by date.

Figure 2. An IT-Centric Approach May Suffice for Implementation or Basic Maintenance

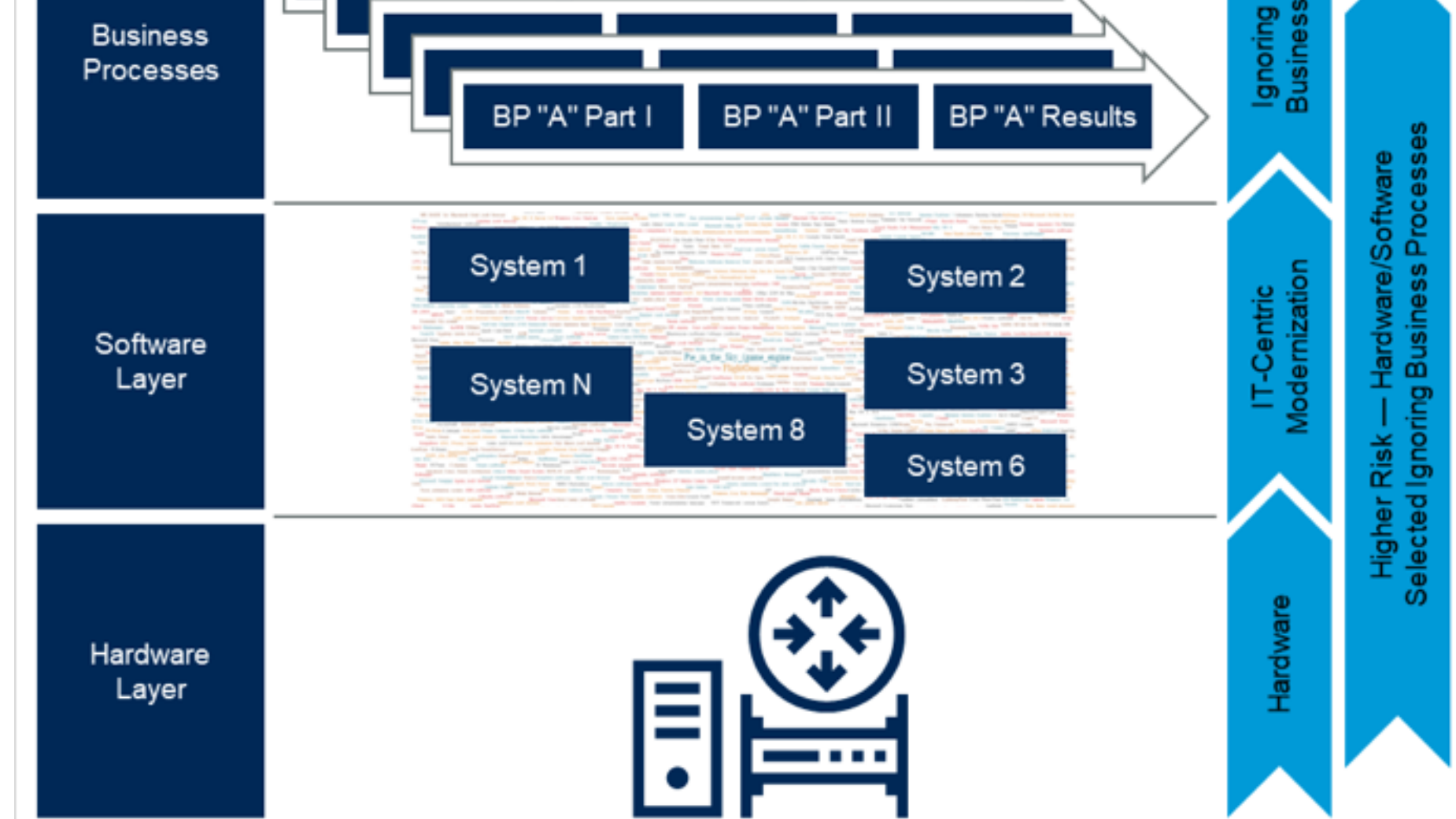
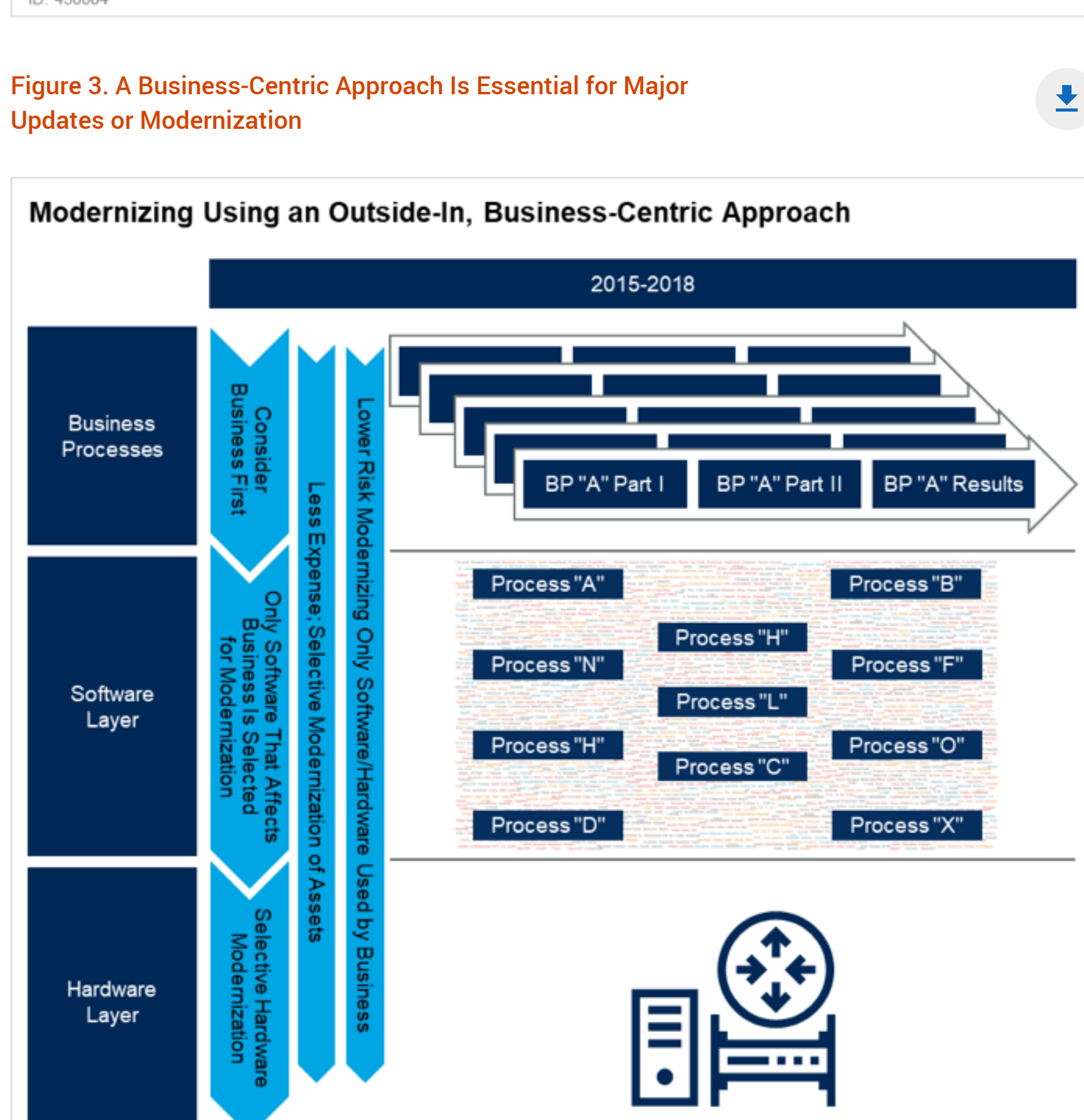


Figure 3. A Business-Centric Approach Is Essential for Major Updates or Modernization



Taking the business-centric approach to prioritizing and managing modernization work – top-down analysis where business processes drive modernization – helps to place that work in the context of the business and digital initiatives it serves. It also tends to focus that work more efficiently on just the application components or services that people actively use, rather than a broad-brush approach.

Perform a Targeted but Deep Assessment of Your Application Estate

The application estate is far too large for it to be cost- and time-effective to analyze in depth, in its entirety by humans. Enterprise Complexity Analysis (ECA) can be used, however for most enterprises the costs are prohibitive (see "Complexity Analysis Tools Are Worth Their Price for Application Modernization and Business Alignment"). Instead, application leaders should:

- Focus modernization on the most important business capabilities. Begin with a simple, high-level business capability model. Drill down to more detail only where the business needs to invest to significantly improve these capabilities – for example, where current capabilities are inadequate to support the firm's digital ambitions.
- Deeply analyze these critical business capabilities in all their pieces and parts. Perform a business-centric, time-boxed assessment of the applications, components, services, data and APIs that comprise these business capabilities. Set modernization priorities to align with and support digital business strategy.

Collect the business capabilities under examination to avoid this kind of failure.

Case Example: Scoping Failure

A large bank discovered – only after it had already begun its modernization efforts – that one application slated for modernization depended on third-party products that were proprietary, and not available in the new target environment. This discovery brought the effort to a screeching halt. Had these dependencies been part of the original analysis, the money wasted on this abortive effort could have been minimized.

To avoid this kind of failure, be sure to conduct an examination on the following:

- All the applications, components, services, APIs and data assets used to deliver the capability.
- How this estate supports the business, at a business process level, and using what features.
- The interdependencies between applications, components, services, APIs and data assets.
- The interdependencies between platforms.
- The full complement of target-state technologies that will be required to run this workload.
- The capabilities of all staff who will be required to operate and maintain this workload.
- The current capabilities of the target environment.
- The business agility, competitive capabilities and innovation competency for the target environment.

It is usually not enough for the modernized capabilities to simply run in the new environment – even if at reduced cost and improved performance. They must also support the firm's requirements for rapid change and innovation, to keep pace with today's competitive digital landscape.

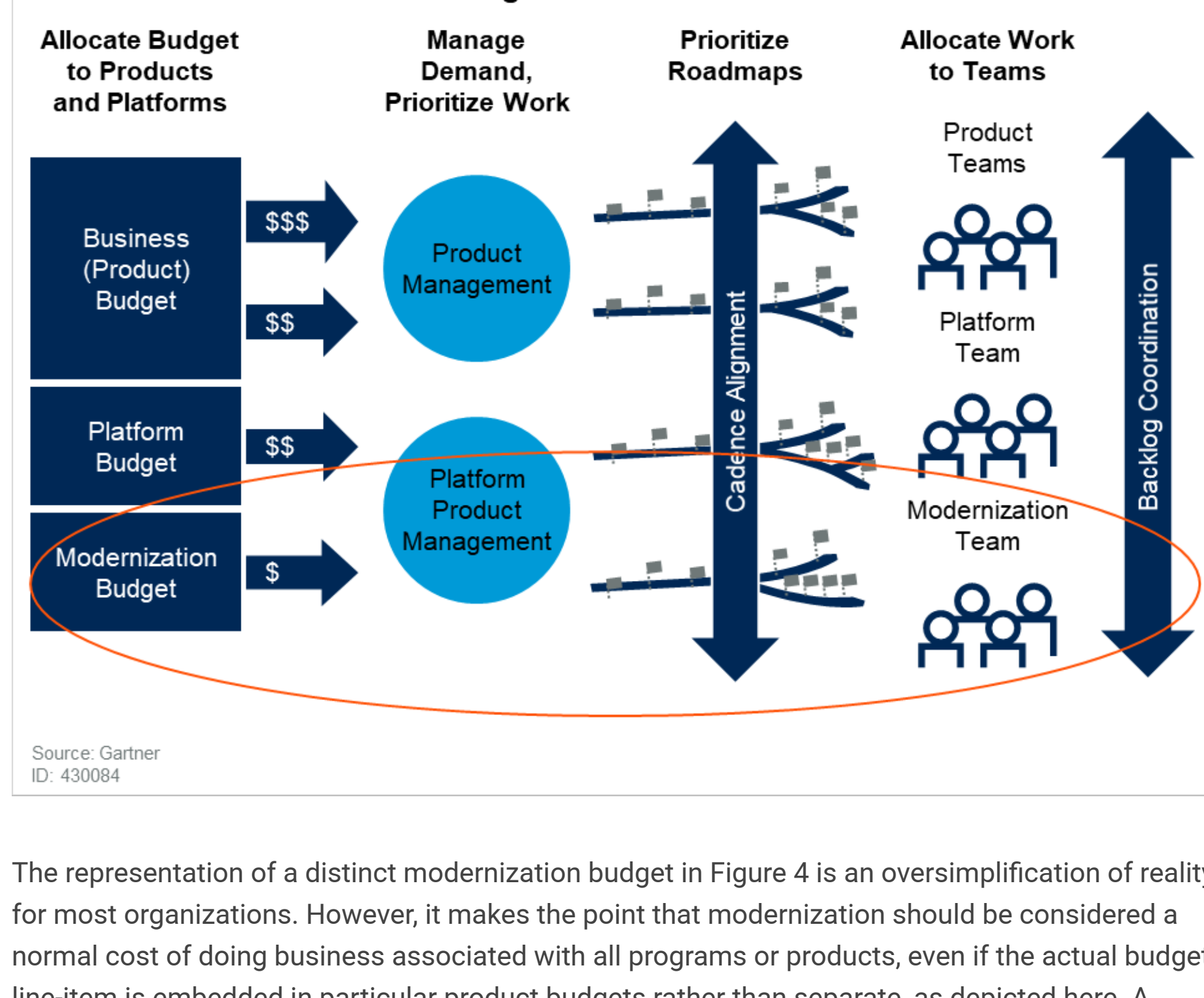
Budget and Plan Modernization Work for the Entire Product Life Cycle, Performing Continuous Modernization

Once the application leader has laid out the prioritized roadmap of smaller chunks of modernization work for individual business capabilities, it might be tempting to toss all these in the familiar "big bucket" of a large modernization project or program. But that approach significantly raises the risk of failure.

Instead, application leaders should:

- Use a more agile, "continuous modernization" approach to allocate the work. What this means in practice will vary by organization (see one example depicted in Figure 4). For those with product teams, allocate work to teams from the overall modernization roadmap, and synchronize their regular agile delivery cadence to produce repeat deliverables. This significantly lowers risk while also enabling a wider range of replacement strategies, including incremental replacement of individual features or services (see "Use Continuous Modernization to Build Digital Platforms From Legacy Applications").
- Allocate work to technology-centric teams where necessary. Where teams are still organized by application area or technology platform, this creates a more complex program management challenge. Program managers must actively manage a more complex web of dependencies, allocating work across teams in a way that minimizes disruption. This often requires compromises such as teams often have slower delivery cadences (monthly, quarterly, etc.).
- Release software for use as early and often as possible. Your ability to chunk releases will likely be constrained by the need to avoid disrupting current users, perhaps even requiring training in the use of new or updated software capabilities. Still, once delivery teams are producing frequent deliverables, your decisions about when to compose a release – even if only for beta testing – will have much more latitude to reduce the risk and speed up delivery of the most valuable features.
- Use the "strangler pattern" to incrementally update platform services. Where applications provide platform services exposed via APIs, this pattern enables teams to isolate other parts of the system from the areas being modernized, via the API. Once the new or rehosted implementations of those services are available, the old services can be turned off, or "strangled," once the new service proves its mettle. Cutover can also be staged and gradual, exposing only a few consumers to the new service at first, then more later (see "How to Design Microservices for Agile Architecture").

Figure 4. How to Implement Continuous Modernization Will Vary by Organization



The representation of a distinct modernization budget in Figure 4 is an oversimplification of reality for most organizations. However, it makes the point that modernization should be considered a normal cost of doing business associated with all programs or products, even if the actual budget line-item is embedded in particular product budgets rather than separate, as depicted here. A distinct modernization budget is more likely to happen when a large modernization program is required, but aims to deliver via the continuous approach.

This continuous approach enables application leaders to stop treating modernization as an occasional exception that requires a big project, and instead more proactively manage technical debt throughout the product life cycle. Don't allow technical debt to accumulate to the point of business pain before breaking loose more funding to address it. This only perpetuates a vicious circle of failed expectations and periodic binges of modernization spending, which are much riskier and more disruptive.

© 2021 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner is a registered trademark of Gartner, Inc. and its affiliates. This publication may not be reproduced or distributed in any form without Gartner's prior written permission. It consists of the opinions of Gartner's research organization, which should not be construed as statements of fact. While the information contained in this publication has been obtained from sources believed to be reliable, Gartner disclaims all warranties as to the accuracy, completeness or adequacy of such information. Although Gartner research may address legal and financial issues, Gartner does not provide legal or investment advice and its research should not be construed or used as such. Your access and use of this publication are governed by Gartner's Usage Policy. Gartner prides itself on its reputation for independence and objectivity; its research is produced independently by its research organization without input or influence from any third party. For further information, see "Guiding Principles on Independence and Objectivity."