

Build a standardized data architecture

(or: How financial services organizations
can achieve Master Integrator status)





Contents

Introduction	3 →
Entering the cloud race	6 →
Importance of integration	11 →
NetApp's approach	15 →

The cloud was supposed to fix everything

Perhaps no other industry sector has been subjected to as many economic and political challenges over the past decade than the financial services and insurance market. The 2007-2008 global financial crisis presaged tougher market conditions and tighter oversight for banks and insurers while paving the way for a new breed of insurgent providers threatening the incumbents. In parallel, an era of restructuring, retrenchment and realignment coincided with dramatic changes in customer expectations, a generational shift and rapid technological advances.

The cloud was supposed to fix everything—making your entire operation better, faster, and stronger to meet ever-changing market and customer demands. And in many ways, it has. But chances are, your data lake is starting to look like the [Great Pacific Garbage Patch](#). Instead of trash, it's “cruft” data—code that is poorly designed, unnecessarily complicated, and generally unwanted.

No one wants a buildup of cruft code. And there are best practices for sweeping

the cruft out from the dark corners of your infrastructure. When you clean up technical debt, you're presented with a rare opportunity to reorganize the whole house: connecting, integrating, and generally improving your company's data and applications. The data can live anywhere—on premises, in the cloud, in between—but the business processes for harnessing and managing that data must be apparent. (And they're rarely apparent from the get-go, which is why there's so much cruft cluttering your attic.)



The cloud was supposed to fix everything (cont.)

Meanwhile, the speed of change is sobering. We know the data universe has expanded. The fact is, physical channels are dying as human behavior adapts and our daily lives move online. This has resulted in tougher data privacy and security regulations. In addition, specialized startups such as fintechs and insuretechs who can quickly spin up applications and services in the cloud to meet customer demands pose strong competition for incumbent financial institutions. Want a single-screen mortgage app? Want to buy insurance on-demand by monthly cycles? Done. To come out ahead, financial services organizations must support new market requirements and customer expectations by anticipating change rather than reacting to it. That starts with data and IT teams. Your team.

Financial services organizations have demonstrated the resilience to respond to the biggest financial crisis in a generation while newcomers have yet to be stress-tested. And they have access to an enormous amount of valuable data capable of providing unparalleled understanding of customer behavior and needs. But the financial incumbents must make much better use of the data they possess. They must turn that data into knowledge, that knowledge into insights, and those insights into actions.





Why this e-book?

This e-book aims to help you understand how and why to build a standardized data architecture—a set of unified standards that govern which data is collected and how it's stored, integrated, and used. It's not about drawing a fancy reference architecture and leaving you to implement sans context. Ultimately, we want to help you take ownership of your data—in clouds, data centers, on the edge. To do this, build a sound data services stack that creates cohesion in your infrastructure and, in turn, in your day-to-day operations.

Luckily, your IT teams already have the skills to facilitate a connected and integrated storage infrastructure so that your company can succeed amid the tech boom. Let's take a closer look at three obstacles to building a standardized data architecture and how to address them.

Entering the space (well, the cloud) race

To succeed, it's vital that you accelerate your digital efforts, not slow them down. Even if you've already gone fully digital first, you might find your efforts to build a streamlined stack confounded by disorganized data or elusive processes. This friction and its attendant hurdles are what we call the change-stability paradox.

The change-stability paradox

Digitalization pulls application leaders in two opposing directions: adapting to the new competitive and regulatory climate without losing out on existing stability.

Change is already here, both organizationally and technologically speaking, and it demands corresponding advances in applications, storage architecture, and myriad other adjacent technologies. Meanwhile, firms shouldn't introduce too much risk. They need to maintain capabilities and operational

stability, particularly in the current economic climate. This push-pull leaves organizations underprepared for change, because in this environment of conflicting demands, they can't properly develop the tools and architectures needed to tackle future problems.

But this problem isn't unconquerable; you can resolve it by taking inventory, identifying your weak spots, and developing a clear plan that balances risk with reward.



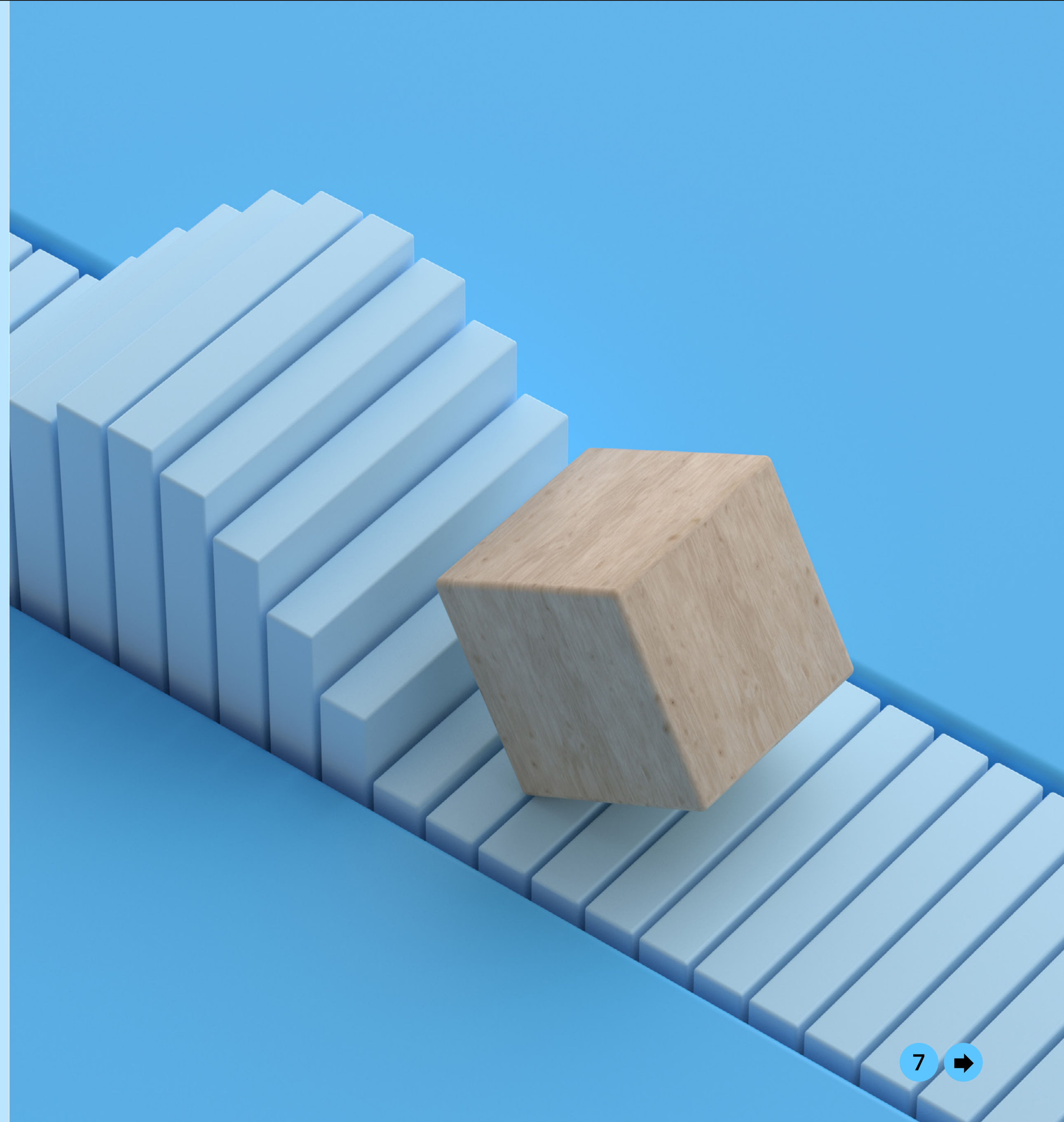
The three impediments of standardized architecture

In our experience, you might encounter three primary impediments as you set out on your quest to create a standardized architecture:

1. Your data exists in a parallel universe →

2. The purse strings are tight when it comes to infrastructure →

3. There's a philosophical data dilemma →



1. Your data exists in a parallel universe

2. The purse strings are tight when it comes to infrastructure



3. There's a philosophical data dilemma



Dark data, cold storage, floppy disks—we've seen it all. In essence, your unruly data has contributed to a buildup of inefficiencies that've barnacled to your systems. On top of that, your organization might have technical debt (or "code debt"). Technical debt results from hurrying to code a solution in the most expeditious manner, rather than a more efficient and more expensive way.

For example, suppose developers inherit the legacy code of a hastily programmed feature; after a couple of years, that code will probably be incomprehensible. If the thinking behind the original code isn't clear or the code isn't sufficiently clean, the move toward a more functional, standardized data management model is an uphill battle. When the code gets messy, refactoring is often involved. These attributes make it more expensive and difficult (the "debt" in "technical debt") to move to a standardized data architecture.

1. Your data exists in a parallel universe →

2. The purse strings are tight when it comes to infrastructure

3. There's a philosophical data dilemma →



Unfortunately, it's not always easy to get infrastructure funding. Why? "Boring" back-end issues don't have an equal seat at the table, and other business initiatives often take precedence over projects to optimize storage. As a result, misguided funding can stymie cloud adoption, critical upgrades, and your long-term competitive edge. Rather than receiving infrastructure investments that will make future improvements quick and easy to implement, IT departments get the equivalent of more cat videos. Each new tactical improvement built without best practice cloud architectures adds to technical debt, which means extraneousness and delays in building anything cool in the future.

1. Your data exists in a parallel universe



2. The purse strings are tight when it comes to infrastructure



3. There's a philosophical data dilemma



Do we put “cloud” in the data center or the “data center” in the cloud? As infrastructure becomes programmable, IT and operations leaders are ceding responsibility for key aspects of infrastructure to software developers; organizational fissures inevitably result from this move. Why? Because software developers are, well, singular—in a good way. A stronger developer role in infrastructure decisions leads to choices that might seem unorthodox, and IT leaders might not immediately understand those choices.

We’ve witnessed the rise of the “infrastructure developer,” a new superhero. She is a code creator, automation engineer, and overall build manager. She’s a tool maker who disrupts outdated notions of occupational silos, pushing companies to rethink infrastructure, data, cloud, and code. So, in a changing world, you must rethink roles, constantly. And as for the philosophical data dilemma, it depends. As we’ll discuss, “cloud” and “data center” don’t have to be mutually exclusive.

The importance of integration:

How to overcome the three impediments and emerge from “chaotic inertia”

To arrive at a unified stack, you'll need to connect and integrate. When you talk about data architecture, you must talk about storage. You already know from a quick glance around your data center that you're living with a hodgepodge of different arrays and heavily customized integration code that break down data silos to aid information sharing. Although public clouds seem to hold the answer, they're prone to data silos, too. This is where a unified data services stack can solve several problems.

Here's how to build it

1

Prioritize the systems that can be retired or rearchitected then decide which of those should be moved to public cloud.

Nailing down and documenting your multi-cloud strategy is a first and essential step to laying out your common data architecture. Assess the technology that exists that is squarely aimed at solving the problem of multiple data subsystems. Your goal is to standardize on a solution that provides predictability and common management, independent of the data center or public cloud it runs on.

2

Select new cloud-based solutions.

Think about where you have opportunities for automation, then you can plan for debt repayment throughout the lifecycle of the system (in other words, continuous improvement of deployed cloud solutions).

3

Reduce compound technical debt by partnering with stakeholders.

Make sure to include all domains of IT and beyond (especially enterprise architecture and application development teams) so that you can manage debt that's outside of their direct purview.

You can integrate your applications with new services in the cloud and pay off technical debt

After you've taken these steps, you can extend data centers into the cloud. Freed from the complexity of the different data control planes, managing and administering many workloads across clouds becomes drastically simpler. Enterprises can benefit from each cloud provider's features without sacrificing operational predictability. This means that firms can now confidently move business-critical applications and enterprise-grade storage services—which formerly would have required rearchitecting—to public clouds. Remember that the leading cloud platforms not only support huge infrastructure requirements for applications (including virtual machines and container management systems), but they also provide global autoscaling. This capability has enabled organizations to migrate their data center applications to cloud platforms. We repeat: You can integrate your applications with new services in the cloud and pay off technical debt.



Good integration is about making your applications and data structures work together, whether they're inside or outside your organization. This means IT teams have the latitude to create a robust, common, and high-performing storage architecture that demonstrates credible, quantifiable business value. In turn, you can define common guidelines for the entire organization, accounting for business objectives, benefits, risks, and key adoption criteria. In other words, less stress, better business outcomes, and free espresso shots in the break room.

Now, about the people who are actually going to build this...

Make life easier for your IT teams

Consider the reality of your IT teams' day-to-day challenges:

1. Development teams outside the IT department are working on custom software applications or corporate chatbots. Each homegrown application has its own set of requirements for living in the cloud. It won't come with an instruction manual the way SAP and Oracle would.
2. Some line-of-business employees need to quickly integrate newly acquired applications with existing systems, and in fact do this already using third-party integration services.
3. Staff members who don't specialize in application and system integration are required to do so within the scope of their own projects—for example, developing a custom mobile application or an internal database.

All of these tasks burden IT staff members by adding superfluous complexity to their jobs.



Unlocking the Master Integrator achievement

Modernizing your infrastructure might be more costly at the start because of technical debt and cruft. The simple solution? Deploy and build in a standard data storage environment.

Modernizing allows you to guard against a state of entropy. You're planning for your company's future, no longer just fixing things as they break or burning copious amounts of time in frequent refactoring. This unified environment connects a wide cross-section of cloud and on-premises applications, systems, and databases, and can be deployed both in the cloud and on premises. This architecture, when deployed in a public cloud, typically offers predefined APIs for applications and standardized tools for building custom connectors. In other words, that achievement? You've unlocked it.

Why NetApp

This is the part where we'd generally tout NetApp's approach to solving all of these problems. But we're not going to do that. We don't tout. We gently whisper.

But, in case you're interested:

- We've developed a unified data services stack that runs on almost anything.
- We offer a comprehensive and unified set of storage and data management services.
- Those offerings work on premises, in the cloud, and in or on all variations in between.
- Our products offer standardization that's guaranteed long after you've left your job for a better gig. (So, there's no need to worry about pesky legacy code or contributing to your company's technical debt burden.)

No matter where your data lives today, now is the time to master it, connect it, and integrate it.



For a look at the specifics, check out:

- NetApp Cloud Central >
- NetApp solutions for financial services >
- Our reference architecture kit >