Mature architecture capabilities

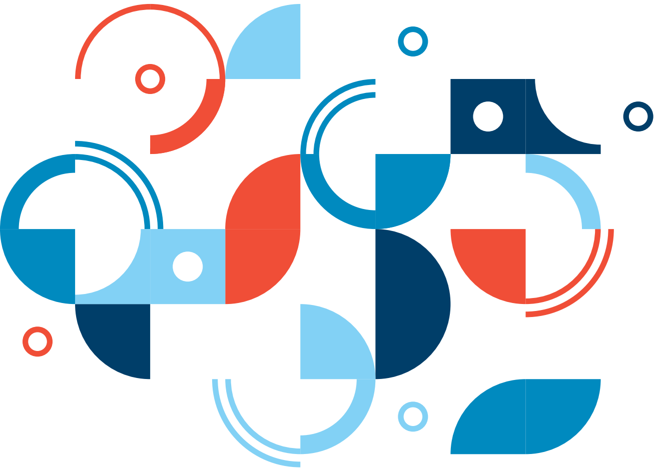
An indispensable foundation for cloud initiatives

Contents

The critical role of architecture in an automated, integrated cloud environment

The benefits of cloud are widely touted: cost savings, increased business agility and improved, efficient service quality. In response, more than 80 percent of enterprise IT organizations will commit to hybrid cloud architectures by 2017.[1] Companies are using cloud to move from a largely resource-based approach to an integrated, automated service model with the increased flexibility needed to meet business goals. (See Figure 1.) **The importance of architecture to this new paradigm cannot be overstated.**

IT architectural guidance and decisions are pivotal to the success of cloud initiatives. They impact cost, speed of adoption and risk. To best utilize architecture in business environments and industries that are often in the throes of disruption, you need capable architects, an effective management and governance structure, and an organizational culture that values architecture. **Business leaders, project and program managers, IT specialists and others are certainly important. Yet, it is the architecture function that aligns stakeholders and leads content direction and structure strategies.**



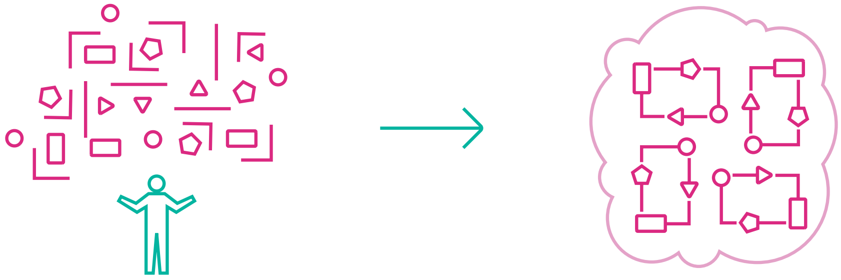
IT automation is a critical component as well. In fact, it's the very foundation of cloud (as well as DevOps, Continuous Delivery, Agile[2], and Software Defined Environments) and enables many of the related benefits. IT automation is used not only to deploy specific business functions but also to provide *all related lifecycle aspects*, such as security, manageability, accountability, provisioning and sunsetting. Given these diverse and often competing objectives, building and maintaining IT automation is complex, especially in medium- or large-sized organizations. The structured approach of IT architecture is indispensable to this effort, allowing you to balance innovation and efficiency. However, not all stakeholders understand or recognize this need, so it is essential to create and maintain an IT architecture in a balanced, diplomatic manner.

Architecture is also essential to the formulation and execution of a cloud adoption strategy. Organizations typically have hundreds or even thousands of existing business applications. Moving all applications to the cloud in a short time frame based on proper business cases is impossible. But, a cloud adoption strategy that uses architecture as a foundational element can provide guidance and direction over an extended period.

In this paper, we'll explore architecture's role in the IT automation of cloud, the components of a mature architecture capability, the need to balance agility with architectural compliance, and architecture's role as organizations increasingly move to environments that blend traditional IT with the external provisioning and global multisourcing of cloud.

Old Thinking
IT maintains IT **resources** that support the business.

New Thinking
IT delivers **services** designed to meet business **goals**.



From silos…

…to integrated flexible building blocks

Figure 1. Organizations are increasingly using cloud to move from a disconnected, largely resource-based approach to an integrated, automated service model with increased flexibility.

What is architecture? The Open Group Architecture Framework perspective

The Open Group is a global consortium that seeks to fulfill business objectives through IT standards. The Open Group Architecture Forum (TOGAF), with more than 200 members spanning all sectors of the IT community, develops and maintains the TOGAF standard (https://www.opengroup.org/togaf/). TOGAF defines architecture as follows:

The structure of components, their inter-relationships, and the principles and guidelines that govern their design and evolution over time.[3]

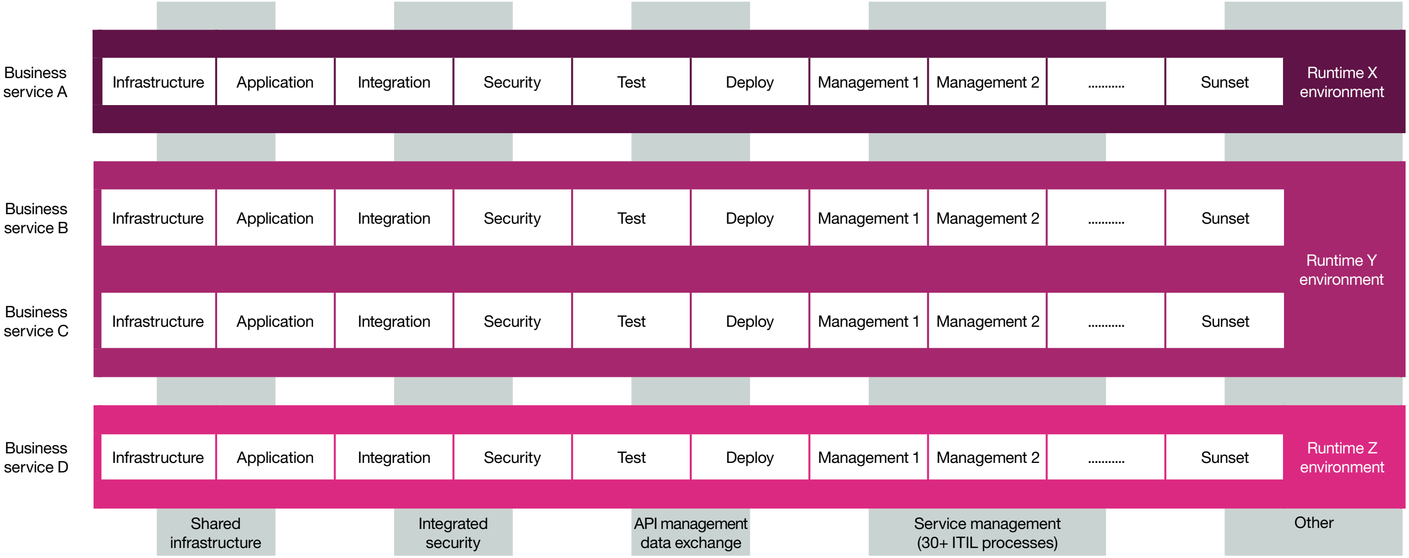
TOGAF 9.1 describes the core elements of the architecture capability, including governance. The Architecture Development Method (ADM) itself can also be used to build that capability. The TOGAF framework is regarded as the most widely accepted resource. A good analogy is that TOGAF is to architecture what ITIL is to operations.

See the Frequently Asked Questions section of the TOGAF website for more details at http://www.opengroup.org/public/arch/p1/togaf_faq.htm

The essence of cloud: IT automation

Cloud has been front and center on the IT stage for several years. Yet, not everyone has recognized the role of IT automation (see sidebar "What is IT automation and how has it evolved?" on page 4) as cloud's foundation and enabler. In fact, other approaches such as DevOps, Agile, Continuous Delivery and Software Defined Environments are also based on full IT automation. These paradigms are based on the assumption that all business service lifecycle activities are captured in programs. This includes not only the initial deployment of the application or business function, but also all activities related to infrastructure provisioning, test execution, security implementation, accountability, preparation for future updates, maintenance and eventually sunset.

It's helpful to visualize IT automation as a long script[4] containing multiple blocks of code, with each addressing part of the business service. These are the horizontal strings shown in Figure 2.



Business service A	Infrastructure	Application	Integration	Security	Test	Deploy	Management 1	Management 2	....	Sunset	Runtime X environment
Business service B	Infrastructure	Application	Integration	Security	Test	Deploy	Management 1	Management 2	....	Sunset	Runtime Y environment
Business service C	Infrastructure	Application	Integration	Security	Test	Deploy	Management 1	Management 2	....	Sunset	
Business service D	Infrastructure	Application	Integration	Security	Test	Deploy	Management 1	Management 2	....	Sunset	Runtime Z environment

Shared infrastructure
Integrated security
API management data exchange
Service management (30+ ITIL processes)
Other

Figure 2. Business services can be visualized as long scripts containing multiple blocks of code.

By using these code “strings,” you can instantly deploy a full business service either from a local environment (private cloud) or from the Internet (public cloud) in a repeatable, consistent format. Automation can also facilitate application or infrastructure updates in short daily or weekly cycles. This includes automated, reliable testing and helps facilitate the iterative nature of Agile, Continuous Delivery and DevOps.

However, many dependencies exist between the individual business service deployment scripts:

- Every script will need a runtime environment. Obviously, an organization wants only a limited number of those.
- Business services need to exchange data through secure, limited integrations.
- Management is also needed. ITIL recognizes more than 30 management processes.

These relationships are represented in the vertical layers in Figure 2. Each must be covered, at least minimally.

For example, the financial reporting requirements of pay-for-use business services illustrate these dependencies. Users interact with multiple disparate business services, and the incurred cost will be summarized and most likely charged to a specific business department. A unified, enterprise-level approach is needed to capture usage—and this process must run in an automated fashion as well.

The need to support these relationships in a fully automated enterprise setting creates considerable challenges for both infrastructure- and application-related initiatives. And, the increasing prevalence of hybrid environments means that you will need to consider traditional IT as well. Starting from scratch is not an option.

Architectural decisions and guidance are needed to structure these environments and enable an effective and efficient operation. Architects and their management will require new skills and knowledge to deal with these complexities—expertise around orchestration, APIs and all other emerging techniques.

They will also need to embrace Agile and its corresponding challenges. Scrum and similar teams (a scrum team provides incremental deliverables as required in the Agile method) should have the freedom to operate in short effective cycles. However, architecture should provide a framework that balances developer freedom with collaboration, reuse and sharing. We delve into this more in “Architectural compliance and the flexibility of Agile: Striking a balance” on page 8.

What is IT automation and how has it evolved?

In this paper, we define IT automation as the technique of operating or controlling a process by an IT program or an integrated set of IT programs, reducing human intervention to a minimum. We define a program as a series of instructions that can be entered into a computer to make it perform an operation.

Automating IT is nothing new. For decades, programmers and system operators have simplified their lives by writing software that helps them execute repetitive tasks, such as installing a server, combining various program components in an install version or applying a specific fix.

While distributed IT automation as described makes sense, problems and issues can arise in several scenarios:

- The implementation of automation in a disparate way by individuals operating in silos
- The lack of proper documentation or quality assurance
- A simple automation script that is complicated by the addition of functionality over time
- Scripts that are not user-friendly and require specialist expertise
- Scripts that do not cover a complete process and require decisions or actions from users

The cloud can be more conducive to IT automation because of the integrated, specifically designed orchestration solutions used in that environment. These enable a cohesive, secure and maintainable approach that covers complete business services without the intervention or expertise of a specialist. For example, an end user can order, deploy and set up a new test environment with just a simple, portal-based action.

Even though hybrid models include traditional IT, both in-house and external, organizations are trending toward external provision of services in a cloud-based environment to take advantage of cloud's flexibility and scalability. Ironically, that very flexibility and scalability is derived from IT automation, a functionality that depends on the control of a mature IT architecture and standardization.

The four components of a mature architecture capability

In IBM's experience, a strong architecture that supports standardization and automation will depend on solid capabilities in four complementary areas: organization, process, technology and information. (See Figure 3.)

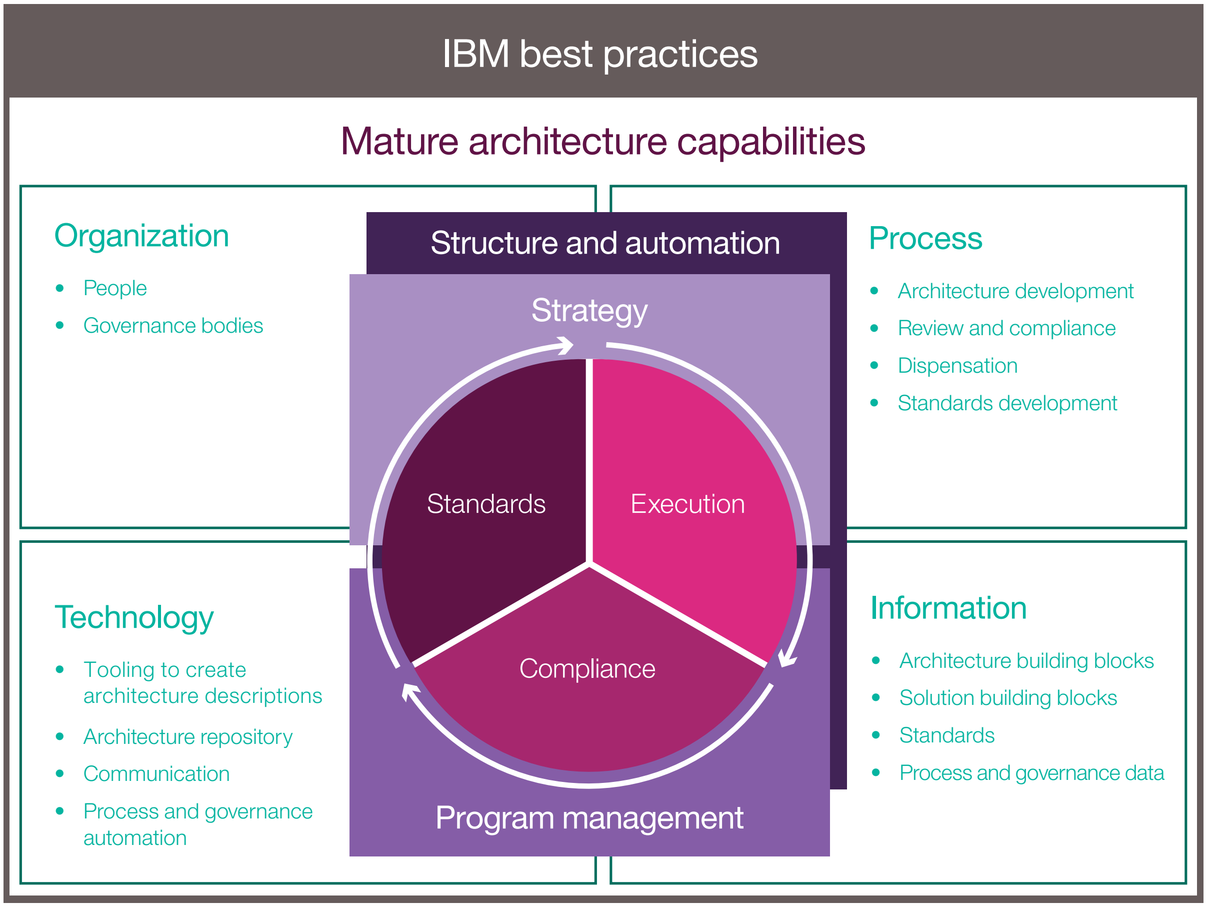


Figure 3. Architecture best practices encompass organization, process, technology and information.

Organization

Your architects should balance competencies and skills that both include and extend beyond the technical. For example, architects must align and rally diverse stakeholders around a common framework and strategy, so communication, tact, collaboration and leadership are important.

Additionally, you will need to implement an architectural authority structure, such as a review and governance board or design authority. This entity will supervise, make or refine major architectural decisions to facilitate alignment with organizational strategy. What services will you migrate to cloud? Which cloud will be used? What are the priorities?

You should avoid strictly cloud-centric authority structures, because decisions must include perspectives across all of IT. In fact, any decision board should include representatives of all stakeholders, with architects holding an advisory role.

Note that you can create a Cloud Competence Center as a dedicated support group to concentrate knowledge and expertise, but this group should not hold exclusive governance authority.

Process

Architecture management relies on four processes:

1. **Architecture development.** This process organizes the architecture development itself by creating and adjusting the actual architectural assets and building blocks. How will your architects embark on projects or activities? How will they collaborate with others? How will you manage their output to facilitate cyclical, iterative activity?

2. **Review and compliance.** Architecture requires regular assessments to monitor and validate major decisions for quality and standards compliance. This is essential for risk avoidance, consistency and optimal re-use. Ideally this process should be automated, with validation and explicit approvals embedded into appropriate project steps.

3. **Dispensation.** At times, you might encounter circumstances that seem to justify and require deviation from standards. Once an architecture is found to be noncompliant, you can choose from two options:

   a. Adjust the architecture to comply with the standards.
   b. Give a dispensation.

   In option "b," you will need evidence that the additional cost or risk is **justified by potential additional business value**. In IBM's experience, organizations need a strict governance process to help support dispensations.

4. **Standards development.** You will need to create and maintain policies, guidelines and rules. Selecting the right granularity and cycles for maintenance is critical, and it may take some trial and error to establish the best choices. These maintenance updates will improve service quality and alignment with external developments.

Information

Architecture and architecture governance require a typical knowledge management process. The data they generate will need advanced organization and categorization, and a simple repository won't suffice. The following categories typically apply:

- **Architecture building blocks.** Architectures described in logical terms. For example, the components of a private cloud or hybrid cloud management use cases.
- **Solution building blocks.** The actual technical solutions. Choices for specific systems or approaches, such as execution patterns for IBM Cloud Orchestrator.
- **Standards.** Guidelines, policies, principles and rules.
- **Process and governance requirements, documentation, reports and so forth.**

Technology

Like any other business process, architecture development and governance depends upon technology for administrative and functional support. The listing here is just a sampling, and functions can be combined. For example, an enterprise architecture tool can be used to create architecture descriptions but can also function as a communication platform or repository. Supporting technology includes:

- **Tooling to create architecture descriptions.** Technology used to create and maintain the architectures. You could use specific enterprise architecture tooling or more generic Microsoft Office business services.
- **Architecture repository.** The place or places where the architectural artifacts are stored.
- **Communication.** A platform that promotes the architecture and helps make it more easily accessible and easier to use.
- **Process and governance automation.** Includes workflow, registration, trailing and reporting to support effective, efficient execution. **Your organization should strive for as much automation as possible.**

Standards, execution and compliance in an iterative cycle

Standards are typically foundational and enable re-use, which is essential to the business case for automated IT. They curtail unlimited growth of unique, one-off business services that could lead to increased risk and cost. Standards are also an essential instrument in integrating cloud into existing traditional IT environments (creating a hybrid cloud model) while presenting a unified interface to users—as well as preventing vendor lock-in.

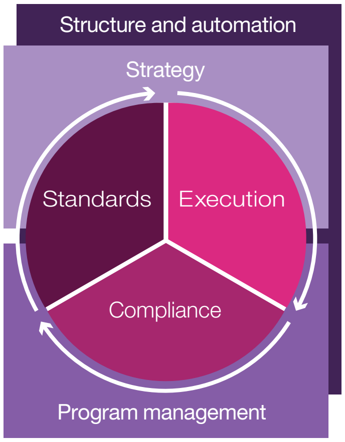


Figure 4. IBM best practices have shown that organizations benefit from a structured approach to standards handling, execution and compliance, ideally in an automated form.

Standards are never entirely static. They evolve over time, for example, in yearly cycles, to accommodate new insights, environmental or strategy changes or other factors. As illustrated in Figure 4, IBM best practices include a structured approach to standards handling, execution and compliance, ideally in an automated form.

However, the most astute policies, guidance and instructions provide no benefit if users don't adhere to them. Automated governance is, of course, the best way to avoid this problem. For example, if a rule states that all services should be described in the service catalog, then it should be impossible to bypass the service catalog when ordering a service.

Still, you do not need to rely solely on a restrictive authority-based system. The use of "carrots," or positive stimulation, can complement your governance process. In one scenario, you could highlight the rapid delivery advantages of an automated cloud environment. A traditional virtual server can take three months to deploy, but you could be operating a standard cloud-based server offering tomorrow.

Architecture standards and guidance play a role at several levels. At a high level, an organization could implement rules that guide the business application transitions from traditional IT to a cloud infrastructure. In 2015, one large global company created a rule that development and test environments should run in a cloud Infrastructure as a Service (IaaS) environment, while critical business applications should remain on a traditionally sourced environment.

At project-specific levels, policies can encourage the use of standardized building blocks, patterns and other detailed solutions as outlined in "The essence of cloud: IT automation" on page 3. The same standards-execution-compliance cycle could apply to failover solutions or security instruments.

Architectural compliance and the flexibility of Agile: Striking a balance

The ability to deliver business services faster in a culture of collaboration and empowerment is not only highly attractive, it's also becoming "business as usual" for IT development. As a result, Agile is here to stay, and its adoption will inevitably impact architecture.

Balancing competing priorities

Any specific approach faces limits to what it can accomplish on its own. One analogy is the choice between communism and capitalism. Living, breathing economies are a mixture of ideas from many philosophies, including communism and capitalism. Correspondingly, good architectural governance is not about implementing one true answer in its purest form. Good architectural governance is about finding a reasonable parity between many competing priorities and strengths.

Clearly, balance is key. While self-steering teams need the autonomy and flexibility provided by an Agile approach, they should also adhere to the checks and balances provided by the architectural structure and plan. The market has responded to this with multiple methods that take control and limits into account, like Scaled Agile Framework (SAFe)[5], Disciplined Agile Delivery (DAD)[6] and Agile@Scale.[7] You may also reference The Open Group's informative white paper, "World-Class EA: The Agile Enterprise," which describes the effects of Agile on enterprise architecture.[8]

Facilitating agility with a solid architecture

Agile should not be confused with *agility*. *Agile* as a development method focuses on fast delivery in short iterative steps with small self-steering teams. *Agility* is an organization's ability to react to evolving requirements in a short time frame. The Agile development method helps facilitate agility, but the two are distinct entities. Both Agile and agility share a need for mature IT standardization and automation.

Essentially, a need for agility should be embedded into all IT solutions or outputs, whether developed through the Agile methodology or not. All solutions should ideally be nimble enough to deal with relentless change, and this agility requires the support of flexible, well-architected, integrated designs. Architecture and architecture governance provide the up-front thinking, framework approach, development and guidelines to help manage compliance.

Migrating to cloud: Adopting an iterative approach

Most organizations have hundreds or even thousands of active business applications. Migrating them all to cloud within a short time frame requires too many resources for even well-capitalized companies. Organizations must prioritize based on strategic importance, "quick hit" results and strong business cases. They should thus adopt an iterative approach as depicted in Figure 5.

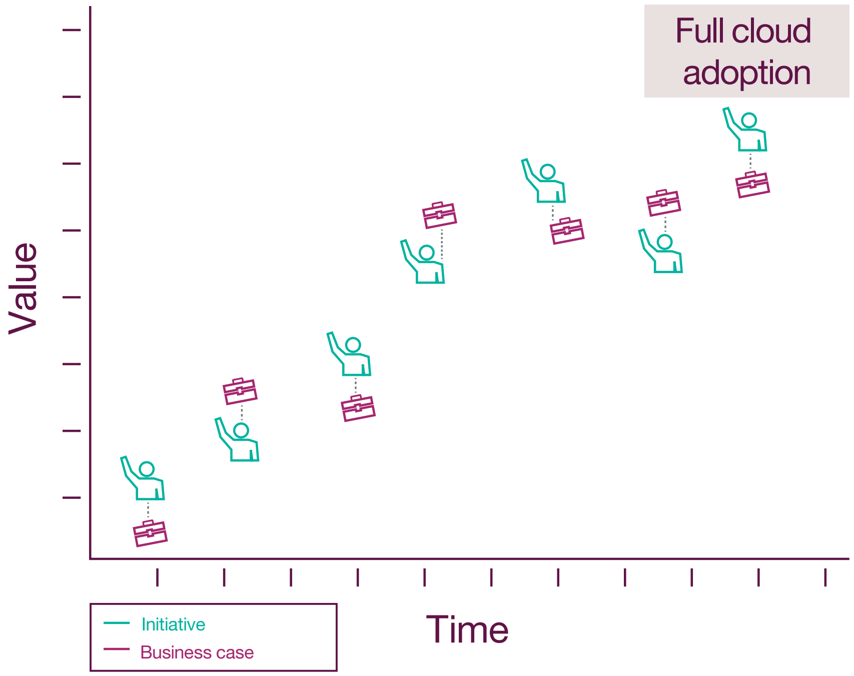


Figure 5. Many individual projects (and business cases) are needed to realize the benefits of a cloud environment and full cloud adoption.

Logical opportunities will become apparent—such as when current infrastructure is due for replacement, or the timing of a major functional release. Mergers or other disruptions also provide opportune times to move to the cloud. Decisions to migrate specific business services to the cloud or to invest in cloud infrastructure will be done in individual projects or activities, with each requiring a sufficient business case. A large waterfall-like approach will not work. However you should also avoid sub-optimization at a project level. You will need to strike a balance. When the decision points are less obvious, a solid, strategic architecture provides essential guidelines for formulating and executing your cloud adoption strategy.

Three case studies: Architecture in action

Case study 1: Deploying more than 5,000 applications to a standardized cloud infrastructure

Motivated by cost savings, this energy sector client started a strategic initiative to deploy more than 5,000 applications on standardized (cloud) infrastructure environments. Additionally, this organization wanted a permanent framework to ensure a continuous approach to lifecycle management and innovation.

Strategic standard environment metamodel

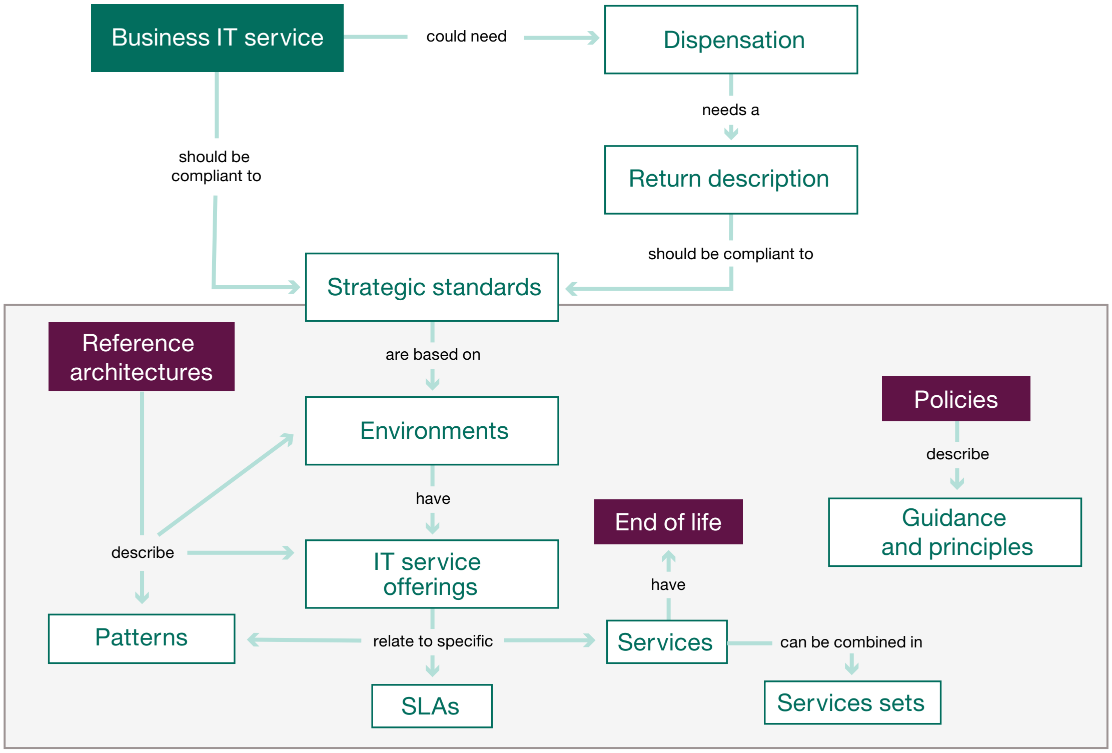


Figure 6. Standards are not always simple concepts, but often involve an extended metamodel with a multilevel structure containing different entities.

In short, this company realized that the fluid, complex nature of standards and innovations requires structure, as shown in Figure 6. Its ambitious objective was not only to drive the business applications towards the standard, but also to drive a permanent approach to adapting and following future standards.

The company embarked upon an extensive campaign to communicate its vision and empower influencers. Still, they underestimated resistance within the organization, and initial results were not impressive. The company realized that articulating this business case was not as straightforward as expected. "Carrots" (for example, cloud can provide a more rapidly available server) and "sticks" (such as the displeasure of senior management) were both needed.

It also became clear that developing concise, unambiguous standard descriptions often requires several iterations. And the more detailed the standards, the more difficult they can be to enforce. Balance is the key and ideally, standards should be automated as well.

The same balanced, iterative approach was required for related processes, such as validation and dispensation. These processes should be workable, streamlined and acceptable to all stakeholders, especially business sponsors.

A solid architecture governance coverage of this initiative proved essential. This included key measurements to gauge progress and aid in identifying required adjustments to the adoption program. IBM was instrumental in building and implementing this governance framework. As a result of IBM's contribution, the client:

- Significantly increased the maturity of their technical architecture
- Saved millions of dollars
- Embedded a process of enterprise-wide standardization, cost saving, measurement and planning

Case study 2: A wholesaler learns the importance of architecture when engaging with an external supplier

A wholesale organization engaged an external supplier to help build a private platform-as-a-service (PaaS) cloud. The wholesaler expected to go live with their first business application by a specific date. Unfortunately, a number of architecture issues cropped up, with security and network being the most significant obstacles to the go-live date. Additionally, it became clear that future consumption would require architectural changes to enable new design areas such as failover and scalability. The organization asked IBM to review the situation and advise how to proceed.

The review team discovered:

- Satisfactory basic project management at the workstream level
- Strong technical skills in infrastructure, networking, security and DevOps
- A reliance on individual "heroics" and long hours to resolve challenges and meet target dates
- A supplier contract that provided basic "building blocks," not the overall cloud transformation
- Continually changing requirements
- No established design authority to drive key architectural decisions and governance

As a result, the IBM review team, understanding the need for heightened architectural governance and structure around this project, made these recommendations:

- Continue to use existing technical resources
- Create a war room to address key issues
- Implement a Project Management Office (PMO) layer to provide coordination, vision, issue resolution and so forth
- Enforce program-level management discipline

- Establish a design authority for architectural governance, and integrate with the wholesaler's architecture governance capability
- Build a business plan to help drive cloud consumption
- Create a plan to ensure that consistent delivery and further optimization and demand alignment are virtually guaranteed post go-live date (after which the supplier withdraws resources)

The wholesaler followed IBM's recommendations. While the go-live date did move out by a few weeks, the end result satisfied all parties. Thanks to collaborating on future architectural visions, the supplier and the wholesaler have partnered to develop additional functionality and applications for the cloud environment.

Case study 3: Creating and implementing an architecture capability from the ground up

A medium-sized retail and wholesale client in the food sector asked IBM to support its implementation of an architecture function that would improve efficiency and effectiveness of its IT function and business support. The client employed five architects.

Seven areas of focus were identified, with a senior IBM architect assigned for a total of 40 days. He teamed with one of the client architects, who also spent about 40 days on this implementation. They used a time-boxed approach for each topic, as shown in Figure 7.

Given the organization's size, the team decided to focus on only two processes: a project architecture development process and an enterprise architecture process. Each architect was asked to fulfill a double role: project architect AND enterprise architect for a specific domain. The Architecture Board function merged with a pre-existing monthly governance meeting.

Step	Days
0. Introduction and getting started	2
1. Why architecture?	5
2. Architecture processes	7
3. Architecture standards (models, template, viewpoints, etc.)	12
4. Architecture compliance	3
5. Architecture governance	3
6. Repository and communication	4
7. Roles and responsibilities	4
	40

Figure 7. IBM and a client architect used a time-boxed approach to implement an architecture capability in 40 days.

The project deliverables were portrayed in an architecture framework that will facilitate future overview, understanding and communication. The client was highly satisfied with the outcome.

Summary: Architecture on the cloud should combine stability with agility

The Internet of Things, big data, cloud, mobility and possible new paradigms will provide many opportunities for innovation in the coming decade. **IT will be a key competitive element of every business or activity.** Global multisourcing will be a reality, offering a nearly endless supply of potential IT solutions, each with more possibilities and advantages than the one preceding it. By its very nature, this new environment will be dynamic, yet volatile.

Even with incessant (and desirable) change shaping your business and technical environments, your users mustn't be frequently distracted by shifts in IT provisioning, disconnects in interfaces, service disruptions or security issues. In other words, your users should experience enhanced business capabilities—not the iterations and adjustments behind cloud and external provisioning.

Therefore, agility must be balanced with stability, integration and control. Organizations will need a mature framework to structure brokerage, management and integration of the disparate services they develop and consume, as shown in Figure 8.

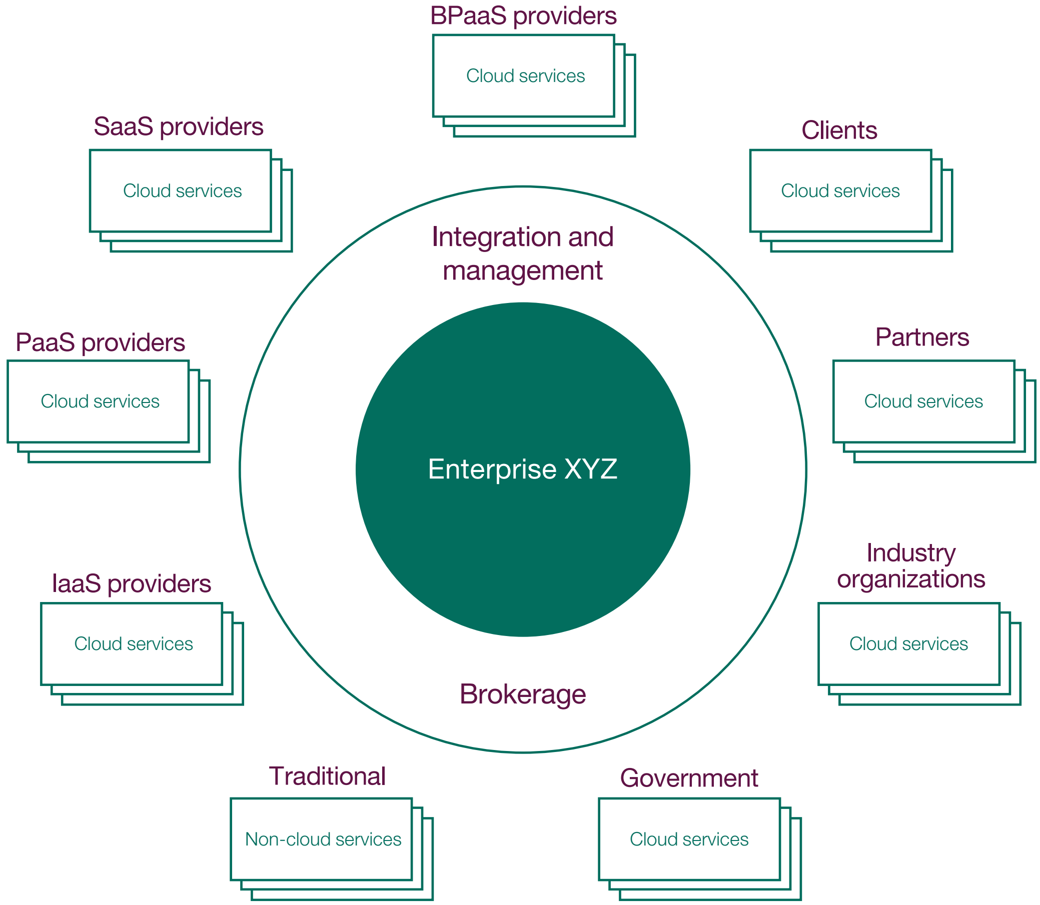


Figure 8. Organizations will need a mature, stable architecture framework to manage disparate, evolving services in an increasingly automated cloud environment. (Note: BPaaS = Business Process as a Service, SaaS = Software as a Service, PaaS = Platform as a Service, and IaaS = Infrastructure as a Service.)

Ultimately, the responsibility to maintain control and provide your users with stable, structured services rests within your organization. You will need to retain architecture, management, direction and governance—in short, oversight—of an increasingly complex IT environment. You will need to design and maintain a services brokerage, integration and management layer with a structured approach based upon core principles and directional choices. Security, API policies, service management and governance will need detailed architectural definitions that are partially supplier- and vendor-agnostic. **This mature, stable architectural framework is a core strategic asset for an organization.**

For this reason, you will need to employ internal professionals, or engage external consultants, who can help you develop, maintain and govern this framework that balances the need for agility with the need for control. As your business services become more adaptive, the IT automation that underlies that flexibility requires structure.

We face a dizzying array of business opportunities around the Internet of Things, mobile, social and analytics—all inspired by cloud-based technologies. But without the grounding of a solid architecture, you won't get the traction you need—or the transformation you're looking for. It's only with a solid architectural foundation in place that you can fully realize your business potential on the cloud.

Why IBM?

Whether envisioning your opportunities on a hybrid cloud, analyzing your gaps, building your hybrid cloud implementation roadmap or strengthening your architecture capability for cloud and IT automation, IBM can help. We can assist you with any or all of the following:

- Define or optimize your architecture management framework and governance model
- Assess your current architecture capability and its readiness for cloud, IT automation and Agile programming
- Train and educate your architects on cloud, IT automation, Agile programming, architecture governance and IT architecture, including The Open Group certification programs
- Develop a cloud or IT automation strategy or adoption program
- Build your business case

Early on, IBM recognized the importance of a strong solution and enterprise architect professional program, aligned with external bodies such as The Open Group. As a result, thousands of our IT architects have been trained to build systems that address your increasingly complex business challenges.

Overall, IBM was positioned as a leader in the IDC MarketScape: Worldwide Cloud Professional Services, 2014 Vendor Analysis. According to IDC's 2014 *Global Cloud Professional Services Buyer Perception Survey*,[9] clients highlighted IBM as strongest in providing industry insights and competence, creating a more effective business and optimizing ratio of onshore and offshore efforts on a project. And Synergy Research ranked IBM as the number one hybrid cloud provider for the enterprise.[10]

With in-depth consulting and implementation expertise and a wide portfolio of cloud services, IBM helps you take the next step with strategy, design, implementation and migration services. We have expertise in 17 industries and global capabilities that span 170 countries, and we help clients around the world benefit from opportunities available on the cloud.

For more information

To learn more about IBM Cloud Professional Services, please contact your IBM representative or visit the following website: **ibm.com**/cloud

About the author

Rik Lammers is a certified senior IT architect based in Amsterdam. He has an extensive background in IT service management and architecture governance. For the past several years, he has exclusively worked on the impact of cloud and DevOps on IT service management, architecture and IT governance practices, and he has co-developed the IBM internal TOGAF classroom certification training. He served as the lead architect for one of IBM's first extended Strategic Outsourcing multitenant ITIL-based service management solutions. He was also one of IBM's global leading professionals on IBM i (AS/400) systems management solutions. Lammers has been with IBM for more than 30 years.

IBM Cloud
Route 100
Somers, NY 10589

Produced in the United States of America
March 2016

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

THE INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT. IBM products are warranted according to the terms and conditions of the agreements under which they are provided.

The client is responsible for ensuring compliance with laws and regulations applicable to it. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the client is in compliance with any law or regulation.

[1] IDC. Worldwide IDC Cloud FutureScape 2016: Mastering the Raw Materials of Digital Transformation. IDC #259840, Nov. 2015. https://www.idc.com/getdoc.jsp?containerId=259840

[2] Agile refers to iterative, short-cycled and incremental development methods. Largely based on the Agile manifesto at http://www.agilemanifesto.org/

[3] TOGAF® Version 9.1, *an Open Group Standard*. http://pubs.opengroup.org/architecture/togaf9-doc/arch/chap02.html. Accessed Jan. 6, 2015.

[4] Strictly speaking, script is not a correct word in most cases. In practice, there are many different options, depending on the technique offered.

[5] Scaled Agile Framework (SAFe). http://www.scaledagileframework.com/. Accessed Dec. 29, 2015.

[6] Disciplined Agile Delivery (DAD). http://www.disciplinedagiledelivery.com/. Accessed Dec. 29, 2015.

[7] "Agility@Scale: Strategies for Scaling Agile Software Development." IBM developerWorks®. https://www.ibm.com/developerworks/community/blogs/ambler?lang=en. Accessed Dec. 29, 2015.

[8] Adams, Mick, Tim Davey, Peter Haviland and Jamie West. "World-Class EA: The Agile Enterprise." The Open Group. April 2012. https://www.opengroup.org/ogsys/catalog/W123

[9] IDC. IDC MarketScape: Worldwide Cloud Professional Services, 2014 Vendor Analysis. IDC #250238. Aug. 2014. http://www.idc.com/getdoc.jsp?containerId=250238

[10] IBM press release: IBM Ranked #1 Hybrid Cloud Provider by Synergy Research, Again. Jan. 15, 2016. http://www-03.ibm.com/press/us/en/pressrelease/48829.wss