Microsoft

Modern Service Management For Azure

Published Version 1.1 January 2017.

Contributors

Chris Bolash, Sander Brokke, John Clark, Thomas Ellermann, Paul Fijnvandraat, Edwin Griffioen, Tim Hoogerwerf, Alex Lee, Samantha Marsh, Carroll Moon, Niels Nijweide, Ryan Schmierer, Kathleen Wilson

Sponsors

Eric Swift, Simon Boothroyd, Travis Gerber, Adam Fazio, Alexandre Pombo, Christian Linacre, Eduardo Kassner, Mark Jewett, Venkat Gattamneni

Credits

Jim Dial, Thomas Shinder – Cloud Services Foundation Reference Model 2013

Reviewers

Name
Marty Larsen
Eduardo Kassner
Conrad Sidey
Joseph Sgandurra
Onat Atayer
Amine Rahmouni
Henrik Savia
Kaisa Selkokari

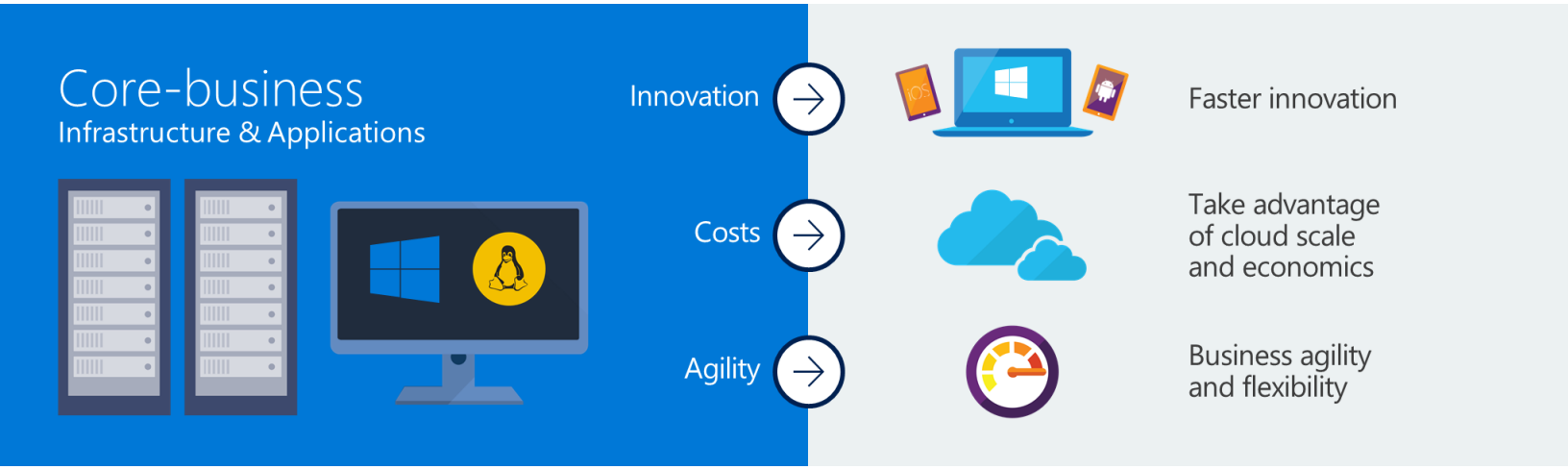
To find this document online go to https://azure.microsoft.com/en-us/resources/msm-for-azure/

Contents

Cloud Computing & Digital Transformation – What Changes

Traditionally, infrastructure and applications were acquired, managed, optimized and supported by centralized IT groups. This model led to the development of service management methodologies that solved for on-premises problems. In the age of Cloud, the business is often able to acquire their own cloud services, without IT oversight, creating new challenges for effective service management. Cloud can quickly streamline and shift operational demands and increase agility, is more cost efficient, and enables tremendous capabilities not available from on-premises data centers. IT infrastructure has become commoditized, and the need for specialized roles focused on computing in datacenters is dwindling.

To stay relevant in this rapidly changing ecosystem, IT must expand their service management methodologies to include managing these new cloud services. Cloud adoption opens a new opportunity for IT to play a strategic role in the future of the business to ensure new cloud services are acquired, integrated and managed effectively. Some of the questions that can arise from cloud adoption include: What does an IT organization look like when they utilize cloud services (IaaS, Software as a Service (SaaS), and Platform as a Service (PaaS))? What changes when they broker services from the cloud, from many cloud providers and from on premises? Modern Service Management to help IT remain relevant, while meeting the demands of today's digitally transforming business.



Modern Service Management (MSM) for Microsoft Azure

Microsoft's Modern Service Management is not a new framework, a set of books, or intellectual property. It is simply an evolved perspective that adapts service management to enable IT to evolve in a changing ecosystem and better meet the transformation and optimization needs of today's businesses. It can be stated as;

> *"A lens, intended to focus our Service Management experts around the globe, on the most important outcomes that evolve our customers from legacy, traditional IT models to easier, more efficient, cost effective and agile service structures" - Microsoft Services 2016*

Why adopt a Modern Service Management approach?

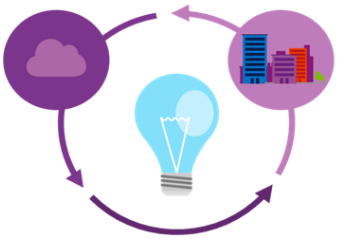
Legacy practices and approaches often failed to achieve outcomes that demanded by the business. Often the business wanted "SM" (service management), but the IT organization still delivered IT (information technology), never achieving some of the main goals of Information Technology Service Management (ITSM). The speed of business demands agility, innovation, expediency, quality and impact. Often IT cannot support this velocity of change and level of demand.

It is time to revisit the role of IT in the enterprise and how this role is transformed with the use of public cloud. Cloud represents a chance for IT to finally establish an agile and cost effective way of delivering IT services while providing the advantages of public cloud services to business units.

What is Modern Service Management?

Modern Service Management is an approach that Microsoft has adopted to make ITSM relevant in a cloud world. MSM will not change current operations and install state, MSM is information on how to utilize cloud capabilities for deployments into Azure. Before applying the MSM lens to individual service management areas it is important to understand the design principles of this vision. Design principles are supported by a rationale and a description of the implications for that rationale.

Add Customer Value



Design Led
(replace, don't repair)

Minimize Manual Activities

Add Customer Value

Rationale

Customers want to pay for activities that create business value, the public cloud is attractive to the business because of its agility, flexibility its innovative platform which on-premises IT struggles to provide.

Implications

IT needs to transform to take advantage of what public cloud offers; automation, self-service and rapid deployment, yet existing operational practices from on-premises are not agile enough to deliver the business value that Azure and other public clouds provides.

Design led (replace, don't repair)

Rationale

Prevent service disruptions and unpredictable outcomes, any updates to the production environment must be tested as a new release.

Implications

Instead of designing for failure prevention, a cloud design accepts and expects that components will fail and focuses instead on mitigating the impact of failure and rapidly restoring service when the failure occurs.

Zero Touch

Rationale

To maximize cloud agility, predictability and minimize failure, manual activities and interventions must be eliminated.

Implications

The resiliency required to manage and run cloud operations requires that organizations invest and implement automation. There should be no manual involvement from detection to response, from release approval to deployment.

Cloud Services Foundation Reference Model & Modern Service Management for the Cloud

Cloud Services Foundation Reference Model (CSFRM), published in 2013 by Microsoft, forms the foundation of our framework for the management of Private and Hybrid cloud scenarios.

Modern Service Management (MSM) for the Cloud modifies existing CSFRM processes and capabilities, and introduces new processes and capabilities. These changes are required for IT organizations to evolve their service management practices (people, process, technology) for today's rapidly evolving mobile first, cloud first businesses.

The diagram below illustrates how to apply MSM to the Cloud subdomains, components, and relationships. It is modeled on the original CSFRM diagram to ease comparisons between the Traditional Way (CSFRM) and the Modern Way (MSM for the Azure).

Subdomains: There are four subdomains, represented by large blue and green boxes. Each subdomain includes sets of underlying components, allowing a collection of components to be referred to collectively

Components: Represented by The small boxes inside the subdomains, components fall under two categories Green boxes contain process capabilities. Blue boxes contain technical capabilities, (functionality provided by hardware, software, or services).

Relationships: The arrows illustrate the relationship between subdomains and how the subdomains impact each other.

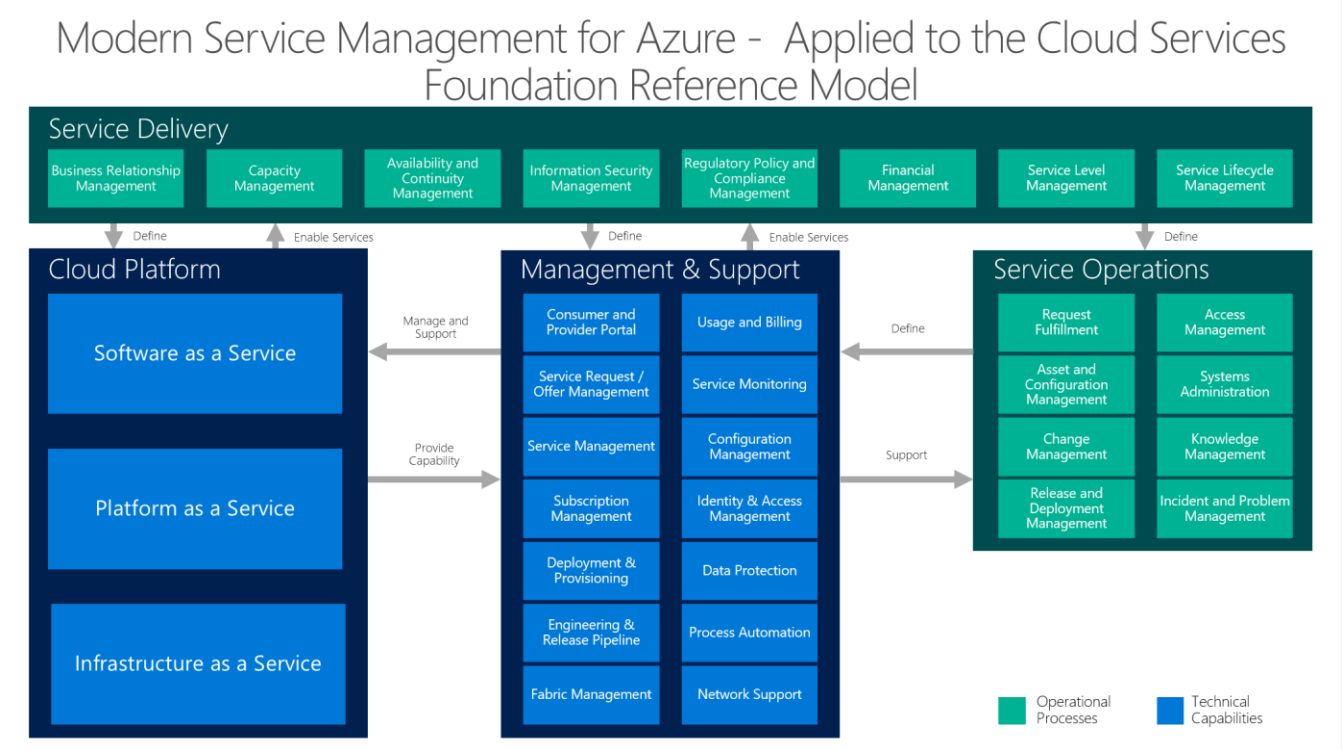


Figure 1: Cloud Services Foundation Reference Model Platform updated with Modern Service Management for Azure

Modern Service Management Applied to Cloud Services Foundation Reference Model

In this section, we will define the processes aligned to the Cloud Services Foundation Reference Model, sharing the traditional approach, and define what changes are needed when you move to MSM utilizing principles, rationales and implications.

Note - Principles are composed of: statement, rationale, implication, risk, actions. Principles focus on the future state.

This approach is the recommended way to start the discussion with IT on what needs to transform to adopt Hybrid Cloud. This approach expands on the Cloud Services Foundation Reference Model to provide guidance on how to manage Hybrid Cloud environments

Principle:	General guidelines that requires judgment and informs decisions
Rationale:	Highlights the business benefits of adhering to the principle, using business terminology
Implication:	Highlights the requirements in terms of resources, costs and activities/tasks for both business and IT to carry out the principle

	Traditional IT Environment	Modern IT Environment
Service Management DNA	Intermediation: Enforce / Control / Prevent failure	Disintermediation: Enable / Support / Accept failure
Business Relationship Management	IT Service Catalogue	Market Trend and Business development
Demand/Capacity Management	IT Infrastructure Capacity	Business demand forecasting and financial pre-commit
Availability/Continuity Management	Service Level Agreements	Design and Quality of Experience
Information Security Management	Network Centric	Identity Centric
Regulatory Policy and Compliance Management	Compliance Requirements	Risk and Threat Modelling
Financial Management	Budget Allocation/Usage	Forecasting/Consumption
Service Level Management	Service Level KPI's	End-User Experience
Service Life Cycle Management	Service Requirements	Service adoption/change
Service Request Fulfillment	Mandating IT	End-User Empowerment
Service Asset & Configuration Management	Lagging, Discovered, and Audited	Leading, Declarative and Monitored
Change Management	Changes reflected in documents	Actual change visible and available in code
Release & Deployment Management	Releases reflected in documents	Actual release (incl. test results) visible and available in code
Access Management	Central	Delegated using a delegation model
Knowledge management	Centralized aggregation of knowledge	Knowledge shared using social collaboration
Incident Management	Modify configuration, process etc..	Register as bug, modify code and redeploy
Problem Management	Root cause analysis across siloes	Bug hunting within BizDevOps team (backlog)

Figure 1 Modern Service Management applied to the Cloud Services Foundation Reference Model

Service Delivery

This subdomain focuses on the translation of Customer requirements into cloud based services and describes how to manage the delivery of these services throughout the lifecycle. The focus is on preventing IT from becoming a middleman for the service-strategy and service design, to instead becoming a service broker or partner to the business, utilizing pre-built Azure capabilities.

Business Relationship Management

Maintain a positive relationship with customers, identify needs of existing and potential customers and help make sure that appropriate services are developed to meet those needs

Traditional Way

IT services are custom-built by IT, based on business needs using wave based service delivery and delegated service operations.

Modern Way

Business and IT collaborate in teams to develop new services that deliver value that the business needs to keep up with the market trends, competition and customer demand.

Principle

Add the value that business units' want

Rationale

Business units will only pay for activities that change the state of a product or service to make it worth more than before

Implications

If IT does not become the Service Provider towards the business demands, business units will bypass IT and source it outside IT (shadow IT, the Business unit will become IT).

Capacity Management

Capacity management's goal is to help make sure that resources are right-sized to meet current and future demand and that resources are used as effective as possible.

Traditional Way

Traditionally characterized by what the current IT infrastructure is capable of scaling to support the business demand, which requires forecasting for short/long term with manual activities to support the process.

Modern Way

Focused on business demand forecasting and utilizing the elasticity of cloud resources to grow and shrink to meet the business demand, this is incurred with charges as demand grows. Proactively monitoring current resources in cloud to help make sure that allocated cloud resources are right sized to control costs.

Principle

Capacity management for cloud must incorporate the elasticity of cloud, drive for resource optimization and forecast business demand while minimizing manual effort.

Rationale

Handling cloud capacity in the traditional way will result in financial and availability implications and will not enable business velocity. Cloud capacity management moves to a business demand forecasting and financial pre-commit exercise frees up expensive unused capacity and allows for rapid expansion.

Implications

- Cost of Cloud are perceived as too high
- Low customer satisfaction due to lack of usage of cloud capacity options
- Wrong sizing of the cloud resources

Availability & Continuity Management

Availability Management defines the availability of a service under normal conditions and Continuity Management defines how risk will be managed in a disaster scenario to make sure minimum service-levels are maintained. Service Providers define Service Level Agreements (SLAs) in the terms of availability under normal conditions or during certain times of the day/week/year.

Traditional Way

The goal of the Availability Management process is to help make sure that the level of service availability delivered in all services is matched to or exceeds the current and future agreed needs of the business, in a cost-effective manner and manage continuity using the reduction measures and recovery plans based on agreed service levels.

Modern Way

In the Cloud, BizDevOps teams are the Service Consumers, who focus on availability and continuity by utilizing highly available Azure solutions (who previously built on-premises solutions based on a set of IT components) and service continuity accomplished through resiliency and advised deployment patterns in Azure (previously redundancy). Cloud providers won't negotiate customer specific individual Service Level Agreement (SLA)'s but measure service availability and continuity using monitoring and reporting functionalities.

Principle

The modern service management approach uses these principles:

- Perception of Continuous Service Availability
- Take a Holistic Approach to Availability Design
- Minimize Human Involvement

Rationale

Azure services provide a cost-effective way of maintaining high availability utilizing its resiliency therefore removing the complexity and cost of redundancy. Business demand forecasting and financial pre-commit exercise frees up expensive unused capacity and allows for rapid expansion.

Implications

Organizations must transform their Architects into Azure Cloud Architects to enable them to deploy updated services focused on business outcomes. Failure to do so could hamper organizations' ability to adopt agile Azure Services and could cause migrated workloads to perform poorly.

Information Security Management

Information security management (ISM) defines the policies, processes to minimize risk associated with security breaches. ISM must help make sure confidentiality, integrity and availability of an organization's information, data and IT services.

Traditional Way

ISM is network centric and focused on managing access control, confidentiality and availability by monitoring and securing IT assets used for service delivery. Traditional ISM involves reviewing logs and data to monitor for events and conduct the appropriate incident response.

Modern Way

ISM becomes Identity centric. Monitoring environment and assets are important, but ISM will focus on identity, applications and data. Emphasis will be placed on:

- Identity Management
- Confidentiality Controls
- Access Controls
- Proactive Controls
- Automated Corrective Controls
- Secure Development Lifecycle.

Principle

The principles for ISM are the same for both the traditional and modern way. The focus on what to manage changes when moving to Azure. The focus should no longer be network centric, but identity and data centric. Automated detection and monitoring is mandatory.

Rationale

When moving to Azure, organizations reduce their on-premises footprint in their datacenters and use public cloud services which utilize virtualized datacenters, where tenants are isolated using Identity and Access Controls and zero standing access.

Implications

ISM focusing on the data, identity, and confidentiality controls of the applications and services introduces the need to mitigate risk and protecting services from malicious attacks by managing end to end services and not physical components.

Regulatory Policy and Compliance Management

Compliance is a process that makes sure individuals are aware of and adhere to regulations, policies, and procedures outlined for their industry or organization. Regular reviews evaluate the activities within the organization vs. the intended results laid out by management's objectives, policies, and regulatory requirements.

Traditional Way

Regulatory policies are applied within industry and governments, such as SOX or HIPPA. Organizations need to manage these regulatory requirements and apply them to services and components. These controls are translated into procedures, settings and technical implementations. Compliance management governs the operational procedures and processes applied to specific components or services at a tactical level (hardening, settings, control) to help make sure regulatory requirements are met. This creates the need for ongoing resource heavy auditing across services and underlying infrastructure.

Modern Way

The operational procedures for compliance will change in Azure. IT organizations will no longer have to manage many of the compliance controls if they use Azure, and IT can focus on managing how their services comply to Regulatory requirements, using the native audit capabilities in Azure.

Azure has met many compliance requirements and they can be found here

This will reduce the risk for the management of the platform for many IT organizations.

Principle

The principle of 'Regulatory Policy and Compliance Management' stays the same and will not change moving into cloud. How it is performed changes as IT organizations will assume the compliance coverage of Azure.

Rationale

Azure undergoes rigorous third-party audits, such as by the British Standards Institute, verify Azure's adherence to the strict security controls these standards mandate. IT organizations can benefit from this built-in capability to reduce overhead and drives down overall service cost.

Implications

Organizations need to re-evaluate the translation of their controls and what it means to purchase the base capabilities that Azure provides. Otherwise they will need to invest in standard audit capabilities that Azure offers and enable customers to automate and centralize their auditing needs.

Financial Management

Financial Management incorporates the functions and processes used to meet a Service Provider's budgeting, accounting, metering, and charging needs. Financial Management in Azure provides cost transparency to the business by structuring a usage-based cost model for the consumer that includes management fees.

Traditional Way

Financial management focuses on IT components that are owned by organizations and allocates budget to manage IT services across an organization.

- IT is responsible for delivering financial information to the business.
- Focus on CAPEX (larger budget) and OPEX (smaller budget).

Modern Way

With Azure, Financial Management is focused on the costs of consuming Azure services. Azure consumers can use cloud services and determine costs without IT involvement. The cloud platform directly provides the financial data to the service consumer (in BizDevOps teams preferably the product owner). Budgeting moves from Capex to Opex, considers forecasted Azure usage and helps manage current usage appropriately.

Principle

A Financial Management process for cloud services takes the following principles into account:

- Driving predictability
- Incentivizing desired behavior
- Minimizing human involvement

Rationale

Organizations need to manage costs to be able to properly manage budget and not overconsume or under consume Azure resources.

Implications

Not adjusting the Financial Management processes and procedures for consumption of cloud services can lead to:

- Cloud costs are perceived as too high as there is no direct linkage to business value
- Inability to do chargeback or show back
- Lack of financial control.

Service Level Management

Service Level Management provides continual identification, monitoring and review of the levels of IT services specified in service level agreements (SLAs). Service Level Management helps make sure that arrangements are in place with internal IT support-providers and external suppliers in the form of operational level agreements (OLAs) and underpinning contracts (UCs), respectively.

Traditional Way

The service provider and service consumer determine the conditions and service levels while creating a contract for an IT service. These service levels are in line with the service consumer's requirements and represent the essence of the service in concrete metrics. The agreed metrics can be backed by financial penalties if the service provider fails to deliver. Once the service is delivered, the service provider will monitor and report these metrics to service consumer as part of overall service.

Modern Way

Azure offers highly standardized services, SLA's are predefined and are the same for all consumers. Within organizations, BizDevOps teams own the end to end service delivered to the Service Consumer (business).

SLA's become XLA's (experience level agreements) aligned to what business users value (the experience), traditional SLA metrics clarify/underpin the business user experience of the delivered service. For example, Skype for Business measures these "experiences" from each

customer call and SLA type metrics and measurements are used to clarify the score afterwards.

Principle

Instead of building fully customized solutions, IT organizations need to utilize existing building blocks with predefined SLA's to achieve the same goal.

Rationale

With the right patterns and guidance, the same service levels can be met using standardized offerings to meet the service consumer requirements. With Azure the focus will shift from service level agreements to business user experience level agreements.

Implications

Not delegating service level management of applications deployed on cloud services to application owners, will result in the business not perceiving value in the service. In Azure, service level management is more lightweight, as the interfaces and functionality are predefined and metrics are predefined by Microsoft. Service consumers will need to manage their service delivery expectations when utilizing cloud services.

Service Lifecycle Management

The act of managing the end to end of a service from inception through retirement, including but not limited to optimization, maintenance, and retirement. All the processes listed within the CSFRM for Modern Service Management help to support, enable, and implement the Service Lifecycle Management.

Traditional Way

The traditional way was based on principles such as Plan, Do, Check, Act or Plan, Deliver, Operate, and Manage. Businesses, would engage with IT with a product/Solution or need, and IT would interpret business requirements to develop Solutions or implement Solutions identified by the business.

Determining hardware, capabilities, etc. were driven by the business, typically not consistently. In many cases Service Lifecycle Management was focused on a specific product or tool, rather than the overall service and its ability to deliver capabilities to those consuming it. The principles are more about controlling the processes leveraged to help make sure products or tools remained up, running, and relevant to the business needs.

Modern Way

Service Lifecycle Management takes on a new meaning in the future. Plan, Do, Check, Act or Plan, Deliver, Operate, and Manage will be focused more on the service and less on Solution components.

Business units will be able to use available services in Azure depending on their business needs. Service development will be driven by initial and subsequent usage of the service. Resulting compliant service design, deployment and configuration patterns will be published in a repository for re-use and versioning over time. Usage of already pre-approved compliant service design, deployment and configuration patterns will result in compliant solutions. Use of these patterns is fully traceable in the release pipeline (which uses principles like: no standing rights in target environment, straight through processing, automated mandatory testing, mandatory approvals) auditing can take place at any time without interfering service delivery or operations.

Teams will be aligned to their business counterparts so that they better understand how and why development is done to support the business outcomes the service needs to adhere to, instead of what powers the service.

Principle

The key principles will be focused on the service itself.

- Performance
- Optimization of features and functionality
- Alignment to Business strategies and Objectives

Rationale

The rationale for the shift in how we approach Service Lifecycle Management is a direct result of businesses wanting to move faster, understand the service (they never cared about the hardware), and helping make sure business objectives are being met.

The end to end understanding of a service, those things that enable a service to be consumed, and why it is being provided will be key. It is no longer about what enables the service, but the complete picture.

Companies will only enable Service Lifecycle Management effectively if they consider organizational and operational change to support the what and why of a service.

Implications

Organizations that continue with the traditional way of managing a service will experience continued incident handling, scaling by adding and never removing, inability to innovate and adopt new trends, and have misalignment to business objectives.

Financial implications might be seen by the continued need for physical data centers, personnel focused on hardware, and the inability to scale long term.

Additionally, Business and IT will not be partnered, leading to businesses taking IT into their own hands. Shadow IT becomes a normal activity, introducing a new level of risk to the organization, which IT might not be prepared to manage.

Service Operations

Service Operations help make sure that each cloud service continuously meets requirements defined from Service Delivery . Organizations define each of these components as standardized processes, but specific application of the processes often varies across services. Management and Support components support the components of this subdomain. Automation is recommended for many of these processes to drive agility and quality while minimizing human error and cost.

Request Fulfillment

Request Fulfillment exists to capture non-incident support and inquiry requests. This includes requests for administration, information, support, and various services.

Traditional Way

Semi-automated request submissions through a service portal or other associated IT Service Management tool. Manual activities may be employed where automated processes do not exist. Requests are often blended with incidents as "tickets", consisting of manual intake and processing with limited service alignment. Requests are often completed through manual determination by the fulfillment resource.

Modern Way

Highly automated with limited human involvement for most requests. Human involvement is required when automated decision processing cannot be utilized or does not exist. Fulfillment resources are focused on assisting service consumers use portal and self-service functionality.

Principle

Request Fulfillment utilizes automation and self-service to limit and reduce manual interactions.

Rationale

Cloud offers many highly standardized services for service consumers that do not require manual resource determination resulting in: reduced cost, reduced human error, increased speed, increased consistency, and higher consumer satisfaction.

Implications

A flexible, role based, and easily accessed service portal exists. Information within the service portal is current and focused on the individual service consumer making the request.

Automation is easily integrated with the service portal to allow simplified triggering and monitoring of automation tasks.

Automation tasks can perform easily and consistently against numerous technologies leveraged by the service organization and any associated cloud service providers.

Asset & Configuration Management

Asset & Configuration Management exists to manage service related configuration items (CI's) and assets. Financial IT Asset Management is an overlapping but separate discipline as the focus here is on service dependency.

Traditional Way

Manual and semi-automated data entry with limited reconciliation, limited service focus, or service relationships and automation. Reactive configuration management based on auditing and considerable manual intervention to maintain quality and accuracy.

Modern Way

Automated population and reconciliation with inherent service dependency which incorporates both configuration items (logical, virtual, non-financial) and assets. Proactive configuration is based on dynamic discovery utilizing a cloud service based approach, rather than building services based on components.

Principle

Asset and configuration management is the true blueprint for services defined prior to service actualization, or discovered from instrumentation in applications and technology providing automated service definition and mapping.

Rationale

Service dependency mapping and CMDB/CMS have never been successful without considerable manual effort which is impacted by latent accuracy due to traditional discovery models. Future asset and configuration management must be self-defined from instrumentation or easily assembled using meta-model components that are easily identifiable in cloud infrastructure.

Implications

Configuration Management is declarative based on service dependency hierarchies.

Configuration Items are self-identifying for attribute and relationship information.

Configuration Management and Asset Management are Automated processes.

Change Management

MSM Change Management is more about monitoring, communicating and reacting to continuous changes, resulting from Release Pipelines. Rather than focusing on change control and delay, as changes are smaller in scope and impact but much greater in number.

Traditional Way

Large changes are managed and approved through a Change Management process with Change Advisory Boards managing the risk of change to the production environment. Change Management processes are often skipped and unenforced due to bureaucracy that results from poorly implemented change management.

Modern Way

Smaller changes driven by Release Pipeline where change schedules are known, mitigation and risk controls (e.g. no standing rights in target environment, automated deployments, automated mandatory testing, mandatory approvals) are engineered into the change operations and support is provided by the same team involved in engineering the change.

Principle	Rationale	Implications
Change Management exists to communicate, monitor and react to a continuous change schedule. May still manage business impacting changes from a risk perspective.	Change Management can no longer limit agility and expediency due to perceived risks. Modern Service Management principles and practices can help mitigate perceived risk. Communication and awareness are transformed into collaboration in a Modern Service Management environment.	Changes are constantly communicated from Cloud Service Providers providing ample time to react and prepare for change. ITSM systems are "Receptive" of Change notifications and notifications are identified from service maps. Service maps should exist for all services.

Release & Deployment Management

Release and Deployment models and patterns enabled by cloud change from heavily manual, heavily tested and slow releases.

Traditional Way

Manual promotion from Dev to Test to Prod

Manual testing efforts

Discrete and planned schedule with larger release packages.

Modern Way

Automated promotion from Dev to Test to Prod

Automated and integrated testing that occurs as prior to and after promotion

Continuous on regular schedule with smaller release packages.

Principle	Rationale	Implications
Principles from Release Pipeline paper Continuous release, deployment, testing, monitoring.	Drives value of agile/DevOps.	DevOps tool chain Cloud infrastructure to support continuous release (VIP swap, virtual).

Access Management

Customers should adopt recommended Microsoft Trustworthy computing policies and principles including "No standing access" to production systems and user data

Traditional Way

Administrators typically have full admin rights to domains and systems allowing easy access to virtually all resources in an environment. Often these rights are extended to non-administrative roles to enable them to update and administer security roles, often exposing domain security to vulnerabilities.

Modern Way

Access is assigned real-time and on-demand. There should be no standing access to production systems or user data. Humans should not add (or configure) capacity or deployments—it should all be automated. If there is no standing access and if all capacity and deployments/releases occur through code, the complexity is minimized and the human factor is minimized. Reference the Release Pipeline Model.

Principle

Elimination of standing access, introduction of real-time assignment of access with automated workflow, and automation of deployments leads to modern access management that will align with modern security management requirements.

Rationale

Humans make mistakes accidentally and or take advantage of access intentionally, so there should be no standing access to production and all deployments should be automated.

Implications

Organizations need to invest heavily in automation and identity management to eliminate standing access to cloud based services and data.

Systems Administration

Systems Administration in this context represents the daily, weekly, monthly, and as-needed tasks that are required for maintaining and supporting any running environment.

Traditional Way

Supporting a private cloud or self-hosted servers requires many different operations teams performing often independent operational and manual tasks. Systems administration is often an undocumented and unmonitored process..

Modern Way

Cloud management platforms (CMP) provide management capabilities of public, private and hybrid cloud environments. Scenarios include scheduling, monitor and performing needed operational tasks in an automated, repeatable, reportable and controlled manner. CMP also provides governance and approval mechanisms for management oversight. BizDevOps teams will only use release management tooling (TFS/VSTS) to target deployment of services on multiple cloud environments.

Principle

A single CMP to manage various cloud environments, including provisioning, reporting and management.

Rationale

Automate to simplify and report on output

Remove repeating tasks from the system administrators

Leave more time to develop and deploy for the IT organization

Implications

If not performing proper systems administration, eventually the platform will become unstable, unreliable and business units will lose its confidence in the Service and IT.

Knowledge Management

Knowledge Management is the management of current, legacy and preview knowledge and driving the organization to become more "knowledge based"

Traditional Way

Operating Instructions, FAQs, Known Issues and Work Arounds based on system design intent and observed issues from incident management processes. Built on a content publishing process model, content is created, aggregated, curated, published through portal and integrated social and email communications to the consuming audience and periodically reviewed for continued relevance.

Modern Way

Modern knowledge management is role and context aware filtered and sorted based on relevance to the consuming user with embedded opportunities to provide feedback on relevance. Modern knowledge management also merges and integrates multiple information sources together providing a curated experience across external (public) content, internal organizational content (provisioning processes, environmental outage notices, etc.), tool / component specific content (such as known defects) and community based resources (discussion forums, social media, etc.). This curated information is available across multiple consumption methods (device types, embedded within UIs, in a Knowledge Management tool, search, etc.)

BizDevOps team will imbed knowledge in code (app/infra) and in the release pipeline for a certain project.

Principle

Aggregate and curate from multiple sources, filter and sort based on user behavior, and make available across a wide variety of consumption methods.

Rationale

Users have a shrinking tolerance for things not working properly and/or being confusing. When they encounter a question or work stoppage, they want to get the information to resolve the issue and return to productivity quickly, without having to leave the context of the activity they were performing. The expectation is that the knowledge management system is aware of the user's context and can provide relevant targeted information based on the environment.

Implications

Integrated Service Knowledge Management System provides multi-source curated knowledge, so knowledge creation is simplified.

Incident Management

How normal incidents are handled at varying levels of technical support workflow

Traditional Way

The traditional approach to Incident Management is to handle incidents as tickets. The user feels impact and takes action to contact support. Depending on the issue, the appropriate level of help desk agent engages the end user to work the issue. Some customers move towards a Tier 0 approach to help desk where end users can run automation for common issues.

Modern Way

The modern approach to recovery drives resolution to the end user so that the incident can be recovered without extending time-to-resolution and without engaging IT human resources needlessly. Every call and intake transaction to support is treated as a bug that should have a corresponding code fix or automation. With respect to Incidents, root causes should be eliminated by code/bug fixes.

Principle

Service disruption interactions to support are treated as bugs with subsequent code-level resolution. If human involvement takes place, resolution should be driven towards first touch.

Rationale

The modern approach to incident management recognizes that the traditional incident and problem process is inefficient both in human resource engagement and speed to resolution for the end user.

DevOps is about increased speed and reduced cost thru software. The incident aspects of providing a service should be no different. Incidents should be eliminated through code just like deployment waste and inefficiencies should be eliminated with automation.

Implications

Root-cause determination is incorporated into bug fixing.

Major Incident Management

Major Incident Management (MIM) dictates how major incidents (including security events) are managed.

Traditional Way

The traditional approach to Major Incident Management (MIM) is that the helpdesk or operations center triggers and establishes a bridge, and everyone who needs to be involved from IT joins the bridge. Often, the operations center resources staff and coordinate the bridge including manually paging the needed resources.

Modern Way

The modern approach to MIM takes a balanced, metrics-driven approach. The desired outcome is to balance speed of resolution with efficiency of the bridge. The process itself should output the metrics therein. For example:

- Balancing reduction of major incidents with time-to-bridge-impacting events
- Balancing time-to-resolution with number-of-people-on-the-bridge
- Balancing time-to-join-bridge with number-of-non-critical-people-on-the-bridge

For every application or service, the following metrics (at a minimum) should be tracked.

- Availability. # Major Incidents. Bridge Duration.
- Time to Detect. Time to Communicate. Time to Restore
- Customer Calls due to MIs. # people paged per MI. # people on bridge per MI
- Approximate Bridge Minutes (bridge duration * people on bridge)
- Support Requests for ANB (including non-MI SRs)
- Total Alerts. Paged Events (even if not MI)

The modern "operations center" drive the metrics and improvements across different workload teams. Workload teams carry accountability for metric target achievement. Drive the cadence and reporting centrally. Decentralize achievement of the desired outcomes and metrics. The balanced metric approach will lead to requirements for automation including automatic bridge establishment, automatic paging and phone calls of needed bridge attendees, automated dependency determination, etc.

Principle

The reality of delivering a service is that incidents will happen. The question is in most cases is how we respond to the incidents for all constituents. For example, for the users and customers, how we handle the incident in terms of speed to resolution, in open communication, and in kindness in discussion are very important. For the service delivery team(s), cost is a factor as is customer satisfaction. We need to acknowledge the needs of each "master" and evolve our process and tooling for MIM in support of achievement of those metrics.

Rationale

Most IT organizations miss opportunities to build trust during incidents. We need to capture that opportunity.

Many IT organizations have no idea how much each bridge costs them and they are doing nothing to improve the cost or the outcome.

Implications

John F. Kennedy once said, "When written in Chinese, the word 'crisis' is composed of two characters. One represents danger and the other represents opportunity". That quotation captures the modern approach to modern Incident Management. The modern approach is to measure and to be intentional. The modern approach is to see incidents to differentiate their service.

Problem Management

Problem Management provides proactive and reactive analysis from both Incidents and Release information (release notes), industry intelligence and telemetry from service monitoring.

Traditional Way

Problem Management is often either not implemented or implemented as an additional support tier and is often understaffed and under-appreciated for its value removing defects. When implemented, Problem Management most often focuses on incidents and may not incorporate application development teams or events from monitoring.

Modern Way

Modern Problem Management takes a collaborative approach involving multiple teams responsible for operations, infrastructure, applications and the business. These teams work together to identify primary issues from a service, collect and measure data to drive relevance and analysis and propose team recommended changes to improve overall service applicability and quality. This is an untapped area where Machine Learning and predictive analytics can augment and automate this process.

Principle

Problem Management provides collaborative reactive and proactive analysis, review, data collection and recommendations across all service components.

Rationale

Proper Problem Management reduces incident volumes.

Implications

Failure to implement Problem Management prevents IT organizations from using the knowledge of what is happening in their environments to improve the business experience and outcomes.

Management and Support Capabilities

Management and Support is concerned with the application of technical capabilities to support the requirements defined by the components of the Service Delivery and Service Operations sub-domains (above). When selecting, and implementing technical capabilities keep in mind a technical capability:

- *Can satisfy the requirements of multiple components*
- *Might only satisfy some of the requirements of a component*
- *May satisfy some or all requirements in different ways*

Consumer & Provider Portal

The Consumer and Provider Portal is the self-service consumption capability for end user individuals and organizations to engage with the IT service provider organization

Traditional Way

Traditional user portals focused on incident and knowledge management capabilities (searching knowledge bases and requesting support) and aligned to a traditional view of service management

Modern Way

The modern Consumer and Provider portal is expanded to include a catalog of business and technical services available to provision, request management and approval workflows, account and subscription management, chargeback/show back of service utilization costs and the ability to capture demand/sentiment feedback - in addition to the traditional incident and knowledge management capabilities. Modern consumer and provider portals also capture CRM (customer relationship management) information from the user about their role, their needs, the relationships between business processes and technology, behaviors/preferences, and indicators of future technology needs.

Principle

Enable greater self service capabilities supported by deep automation, and API integration.

Rationale

Provide standard, consistent methods to interact with cloud services/resources.

Implications

Reduce friction by automating the processes used to interact with cloud, and with IT.

Requires a catalog of pre-approved services/operations, their quotas/limits, and in some cases their costs.

The portal is a method to request new services/operations, not new engineering.

The catalog contains the pre-approved manifestation of cloud resources/services. Pre-approved means the appropriate capacity, identity, engineering, configuration, and automation controls have been implemented by/with the appropriate teams (IT, business, application owner, etc.).

Usage & Billing

The Usage and Billing capability is responsible for tracking the provisioning of technical services to specific users and organizations along with usage of those services. The usage information is combined with service and component cost data to determine the cost of services consumed. The total cost of services may be shared directly with the users or translated into pricing if a mark-up is desired. Show back of cost and/or price may be presented to the user/organization through the Consumer and Provider Portal. Additionally, chargeback of costs/price may be fed into financial management systems.

Traditional Way

Traditional us age and billing systems provide a picture of the services, applications and/or capacity that was ordered/provisioned to the user or organization along with directly attributable costs of each provisioned component. Most indirect costs (shared components, administrative, infrastructure, etc.) lumped into an overhead "tax" that is either expressed separately from the components or applied as a mark-up to component cost.

Modern Way

Modern usage and billing systems seek to show the cost of services consumed (which may be different from what was ordered/provisioned). Costs are usage based (actual utilization of services and capacity consumed) and configuration context aware (built on more complete configuration management information about the components that make up a service). Transparency is provided to both direct and indirect costs (including allocated component and infrastructure costs) and may be articulated as either price or cost.

Usage and billing data are an input to metering/throttling controls (cost or capacity based usage quotas) to enable organizations to prevent over consumption and the incurrence of un-forecasted costs.

To incentivize desired behavior a recommended practice is to move usage and billing insights and budget as close as possible to where the generated business value is received/perceived (e.g. product owner).

Principle	**Rationale**	**Implications**
Provide users with cost transparency associated with their consumption of technology resources. The cloud can help reduce capital expenditures for IT, and optimize operational spending.	Enables deeper understanding of the total cost of a solution/application/service and leads to more efficient use of resources.	Analysis of usage and billing data inform architectural patterns and decisions. Understanding costs leads to better informed decisions about hosting and rightsizing. Organizations can choose the cloud provider, resource size, and cloud model (IaaS, PaaS, etc.) that meet their technical and monetary requirements

Service Request Catalog/Offer Management

Service Catalog Management and Offer Management provide a mechanism for defining and managing the services offered to individuals and organizations including the management of component part lists, service offerings, catalog views, and the configuration templates used to provision services. Depending on the organizational context, this could include a technical service catalog, a business service catalog, or both.

This capability also includes the management of entitlement rules that define what audiences are authorized to request each service along with any associated approval workflows.

Traditional Way

The Service Request Catalog focuses on services built and managed by on-premises IT. You can only make service requests against existing in-production services, anything that is not in the catalog is treated as a change.

Modern Way

The catalog contains the pre-approved manifestation of cloud resources/services. Pre-approved means the appropriate capacity, identity, engineering, configuration, and automation controls have been implemented by/with the appropriate teams (IT, business, application owner, etc.).

Business units will be able to use available services in Azure depending on their business needs. If a service will be used for the first time within the enterprise, service development will be driven by initial and subsequent usage of the service. Resulting compliant service design, deployment and configuration patterns, will be published in a repository for re-use and versioning over time. Usage of already pre-approved compliant service design, deployment and configuration patterns will result in compliant solutions. Given the fact that use of these patterns is fully traceable in the release pipeline (which uses principles like: no standing rights in target environment, straight through processing, automated mandatory testing, mandatory approvals) auditing can take place at any time without interfering service delivery or operations.

Principle

Service Providers offers a curated catalog of applications and services available to end users.

Rationale

Reduces friction for end users by moving the integration and engineering work to a pre-request (pre-catalog availability) phase.

Implications

Requires a separate process to handle non-standard requests. How does a user make a request that starts the engineering effort? Engineering effort (requirements, development, testing) occur up front, before the offer is placed in the catalog.

Provider works in coordination/collaboration with the Offering/application/service/ owners.

Catalog management includes defining the quotas/limits, target audiences (identity & RBAC). Who can request/order, how much. Catalog management is the means to pre-approve what is allowed.

Service Monitoring

Service Monitoring is a very important part of any organization. The monitoring provides a view of the current health and performance of all relevant services in the IT Organization. If thresholds are reached or services become unavailable, it is the monitoring tool that initiates a series of events like state changes in views, email notifications, automated incident creation etc.

Traditional Way

Monitoring of component availability and responsiveness/performance is done on a node by node basis. Busines systems are not monitored from a service map perspective and traditional monitoring tools do not offer transparency into dependencies across servers and services. Many traditional monitoring solutions create false alerts, leading to the perception that the solution is unreliable and untrustworthy.

Modern Way

Modern Service Monitoring provides end-to-end service availability and performance monitoring with abstraction of individual components. The dependencies across servers and services are automatically updated (dynamic service maps), and previously identified threshold breaches are solved automatically by initiating automation workflows. Modern Service Monitoring is integrated with the service management system to drive awareness in IT-operations teams and accelerate response times. Any new service being deployed, whether IaaS or PaaS will be automatically added to the Service Monitoring solution. The service owners in the business unit will also have access to view the health and performance of their services, to drive availability and performance transparency and to enable a self-service approach for root cause identification.

Principle

Monitoring at the service boundary (technical services or business services) provides the business units (customer) and IT organization a common view of the state of the service.

Rationale

Provides a near real time view of owned services and enables a common view on dependencies across both on-premises and public clouds.

Implications

Without proper service monitoring the business impact can be enormous, as unavailable services can greatly reduce revenue and reputation in the market.

Service Management

The Service Management capability supports many of the capabilities described under the ITIL© service operations process area and represented in legacy IT Service Management software packages. Specific focus areas are features related to Service Level Management, Incident Management, Problem Management, and Change Management processes

Traditional Way

Traditional Service Management capabilities have focused on enabling delivery of service management (ITIL©) processes within a provider organization.

Modern Way

Modern Service Management capabilities are focused on coordination of service management processes across a service provided to the business. BizDevOps teams responsible for the end-to-end services delivered to the business primarily rely on application architecture and design patterns to achieve the KPI's in the agreed XLA/SLA. For on-premises and/or hybrid environments these teams will rely on the service integrator role within the central IT organization. This role will provide service integration across a supplier ecosystem and will be responsible for brokerage and routing functions, maintaining workflow connectivity across delivery organizations and aggregating incident, problem and availability/performance SLAs.

Many organizations will require both a legacy service management system and a service brokerage system to support bi-modal service operations.

Principle

IT organizations shift from a design/build shop to a broker of services from a diverse supplier ecosystem.

Rationale

In public cloud the traditional demand/supply relationship between service consumer and service provider (incl. hosting) shifts towards a pull/push relationship between service consumer and cloud provider (hosting) where the service provider only provides functions (procurement/contracting, billing, compliance etc..) that the service consumer (or the organization that the service consumer is residing) is willing to pay for (added/required value).

Implications

Not becoming an IT broker of Services puts on-premises IT in jeopardy of being bypassed by Shadow IT and the business units who will directly consume cloud resources to get the agility they need.

Subscription Management

The fundamental organizational construct within Azure is the subscription as all resources are contained and transacted within a subscription. Subscription Management is a new capability required to manage cloud services/resources. Subscription management enables IT to measure SLAs, control costs and access (security, compliance). Fine-grained control is available at the service/resource level within a subscription.

All subscriptions have resource limits imposed by the cloud provider. Resource limits allow the cloud provider to manage their capacity and define their SLAs. The resource limits placed on subscriptions are also a determining factor when designing one's subscription management strategy.

Traditional Way

Subscription Management does not exist for on-premises datacenters. The closest analogy is a shared services resource model for the entire datacenter including the hardware, network and all the IT components.

Modern Way

A Subscription Management strategy is required to define the subscriptions required to satisfy the IT and business control boundaries within the contexts of the cost structure of the business, the security requirements, the application/service requirements, and the subscription limits.

Principle

A Subscription Management strategy is required.

Rationale

Subscription Management ensures that IT and the business can deliver the right resources/services to the right audiences, at the right time, and with the proper controls.

Implications

Without a Subscription Management Strategy, organizations run the risk of unchecked, unmanageable consumption and increased costs.

The application / service XLA (eXperience Level Agreement) is an amalgamation of the resource level and subscription level SLAs.

A strategy enables purposeful and responsible consumption of resources.

Configuration Management

The Configuration Management capability is focused on understanding the component elements of the technology ecosystem and the connective tissue/dependencies amongst them. This includes capabilities for discovery, inventory, relationship mapping and lifecycle management (planned/deployed/in-use/retired).

Traditional Way

Traditional Configuration Management is focused on the "as-designed" or "as-intended" ecosystem configuration, defined either as a part of system design/deployment and/or through inventory/audit of operating environments and a few key data facets.

Modern Way

Modern Configuration Management acknowledges a highly dynamic set of relationships between people, technology, data, infrastructure, costs and geographies to create an "as-operating" picture of the ecosystem. Modern Configuration Management offers a high degree of automation for discovery, correlation and visualization of real-time relationships.

When using principles like: no standing rights in target environment, straight through processing, automated mandatory testing, mandatory approvals in the release pipeline, the resulting configuration is dictated by the release pipeline and documented in the release pipeline. Automated discovery provides oversight from an operational perspective.

Principle

The technology environment is too complex and changing too quickly for legacy human based configuration management approaches. Informed management decisions require accurate, real-time insights into how things connect.

Rationale

IT defines both the "as-intended" state and know/discover the "as-operating" state.

Implications

Automated processes and systems define the "as-intended" state and record the "as-operating" state. Automation constructs such as ARM templates and DSC (Desired State Configuration) define the "as intended" state, and constrain the possible "as-operating" states. The required definitions and constraints are pre-approved, and pre-engineered, enabling automation and eliminating manual processes that are unreliable and reduce agility.

Identity and Access Management

The Identity and Access Management capability consists of several components: A store of entities (users, groups, computers, applications). Each entity's entitlements, relationships, and memberships. The ability to authenticate that an entity is who it claims to be. Methods to synchronize and federate entities and attributes across different stores and providers.

Traditional Way

Authentication. Authorization. Directory.

Modern Way

Identity is the new control plane.

Principle

Identity secures the Enterprise, empowers users, and facilitates agile response to on/off-boarding of users, applications, services, data access.

Rationale

Unified/Common identity between on-premises and cloud across cloud services enables a good cloud experience. Managing disparate identities is arduous and time consuming.

Implications

Identity is another perimeter. Access to everything is controlled by one's identity.

Identity is an enabler for users:

- Self-service identity management unblocks productivity stoppages.
- Single sign on minimizes password prompts.
- Strong and multi-factor authentication reduces reliance on weak passwords.

Identity is an enabler for security:

- Another perimeter.
- On/Off-boarding of users to/from Enterprise assets (data, applications, services).
- Selective use of strong and multi-factor authentication by entity (user, data, application)
- Detailed reporting and machine learning detects who's accessing what.

Engineering System / Release Pipeline Mgmt / DevOps Tool Chain

Includes the automation of some or all: engineering requirements, backlog, source control, peer review, testing, staged release (staging environments, A/B testing), and feedback mechanisms. The engineering systems define how the work is planned, executed, and released. Automating these systems through a release pipeline reduces friction in the systems and enables rapid/agile response to business needs.

Traditional Way

Changes are identified, engineered, tested, approved, and deployed. Configuration drift and other unforeseen environmental differences have significant impact on production state. Fallback is often not fully possible and there is a point of no return. it may be impossible to understand the full extent and implications of the change. Significant effort is spent toward understanding the implications and mitigating risk. Changes may lead to reduced capacity and availability of the production environment..

Modern Way

There is a tight integration between tools & process, creating a pipeline, or several to release into production. The goal is to reduce the friction in engineering effort and allow for an agile response to business priorities.

Production environments in most cases are deployed new, not upgraded, modified or changed. The Product owner approves release and deployment into production. Leveraging straight through processing with zero touch deployment, continuous deployment and release.

BizDevOpsSec members have no "edit" rights in target production environment, where Engineering effort only performed in dev/test/staging. Testing is mandatory and automated covering static analysis and unit testing.

Once the release is live, Customer traffic routed from old to the new or updated environment. If there is failure, the resolution is to route back to the previous environment. For decommissioning of services, the approach is to route users to the new environment and delete the old environment.

Principle

Extreme automation applied to all Engineering, DevOps, Release Pipeline phases and processes.

Rationale

Manual intervention and processes slow the agile response to business need.

Implications

A pipeline encapsulates a process or set of processes. Pipelines increase agility by applying extreme automation.

A pipeline removes opportunities for human error and helps drive consistent repeatability.

A pipeline does not have to encompass the end to end process. A pipeline can automate a subset of processes.

Effort is required to build the pipeline, incorporate feedback, manage and prioritize the backlog, build the pipeline, and build out the automation of the pipeline

Data Protection

The goal is to protect data in the event of data corruption or loss or underlying storage capability failures, or any combination of these events. Data Protection directly supports data retention policies that support the Information Security Management, Availability and Continuity Management, and Regulatory Policy and Compliance Management components.

Traditional Way

Data plans are created to determine how on-premises data will be kept confidential while still maintaining integrity and availability. Access to data is managed by least privileged access

Modern Way

Classify your data, its sensitivity / risk horizon, what the damage would be if compromised, and categorize it relative to an overall information security management policy. Understand and document data flow requirements and processes to identify risks and necessary points of (protection) enforcement. Such activities are also core to standards compliance practices. For more information on how Azure helps help make sure internal integrity, refer to the Compliance section of the Microsoft Azure Trust Center.

Principle

Maintaining information security and privacy is a continuous process that spans both your on-premises datacenter and your Azure environment.

Rationale

Failure to classify and understand different categories of data leads to organizations applying overly-strict policies to data that is at high risk of being compromised.

Implications

Leverage the capabilities of Azure to keep your data at rest, in transit and protected, minimizing the risk to your IT organization without negatively affecting availability.

Deployment & Provisioning

When it comes to deployment and provisioning of servers and services, it is a recommended practice to use a standardized and repeatable way of performing such tasks. With the introduction of Azure, the possibilities to provision a service end-to-end has significantly increased.

Traditional Way

Many companies are using an image deployment method for server deployment, often leaving many manual tasks to complete afterwards. These tasks include: applying latest updates, mounting and formatting storage, modifying permissions, etc.

Modern Way

In a modern approach, there is zero IT-admin involvement in provisioning. With Azure comes a variety of options for provisioning services, including ARM templates, desired state configuration, Infrastructure as Code etc. Using these automation tools, provisioning time decreases and frees up IT to focus on other, higher priority tasks. Anything that needs to be repeated more the once should be automated.

After provisioning, it's also beneficial to introduce a "Rightsizing of Compute in Azure" program. In Azure, you pay for the size of VM you have deployed, not how it's utilized. If over a period for days, weeks, months monitoring shows a small utilization, it will be beneficial to downsize that VM. The same goes for premium storage, where you pay for allocation and not consumption. In the world of PaaS there is often several tiers to choose from, from the rightsizing point of view it's also important to consider the value for money and eventually downsize or upscale a service.

Principle

With the correct usage of deployment and automation tools, you will get a standardized and reliable platform, deliver business requests faster and release IT staff to other tasks.

Rationale

Manual intervention and processes slow down response times. Cloud can provide rapid provisioning and decommissioning.

Implications

Modern Deployment and Provisioning introduces DevOps capabilities to move away from manual interactions and focus on release templates or ARM templates to deploy updated resources. Automation can increase agility, accelerate time to market and reduce human error.

Process Automation

Process Automation coordinates automated processes across multiple Management and Support and Infrastructure components. It helps make sure that processes are completed in accordance with their defined tasks. This component directly supports automation of many of the Service Delivery and Service Operations components.

Traditional Way

Automation is done in pockets, it has been adopted in functional silos and leadership has not been aggressive in driving automation integrated across the organization.

Modern Way

Automation is a byproduct of all the work that IT does in the cloud, with the requirements to have zero standing access, bimodal mode 2 operations, IT organizations aggressively automate incident remediation, deployment, decommission, patching and operations.

Principle

Automation is key to organizational IT maturity and business velocity in the cloud.

Rationale

You should not be manually intervening and manage cloud resources, the expectation is that all routine and repeatable patterns and processes are automated.

Implications

Roles must be assigned to focus and help make sure that extreme automation is applied, managed and implemented to manage end to end services.

Fabric Management

Fabric management is the management of the on-premises hypervisor, e.g. Hyper-V and VMWare. With the usage of Azure, the fabric is being maintained by Microsoft and the Customer's IT department does not perform Fabric Management for resources in Azure. When using a hybrid cloud approach, The Customer's IT department is responsible to manage from the operating system and up (for IaaS) and it it's a PaaS deployment the IT department manages even less, for example if using Azure SQL, no more database platform management is necessary.

Traditional Way

The management of the network, storage and compute in the traditional data center. Managing new deployments, failovers, firmware upgrades, physical cabling etc. is time consuming and the skilled IT staff are becoming harder to find.

Modern Way

Moving to a public cloud eliminates the need for the traditional fabric management, but introduces new areas instead. In a public cloud the IT staff still needs to plan and manage storage and networking, or the service level and not on the fabric level.

Microsoft Global Foundation runs all Microsoft's data centers; you can read more about it here: https://www.microsoft.com/en-us/cloud-platform/global-datacenters

Network Support

Enables the use of network protocols used by Infrastructure component to communicate with each other and other devices. Typically includes functionality such as dynamic host configuration protocol for internet protocol (IP) address assignment and management, domain name system for IP name and address resolution, and pre-boot execution environment to enable a network interface-based boot of the Compute component without direct-attached storage (DAS) or operating system. This capability directly supports the Infrastructure components.

Traditional Way

Focus of networking covers both virtual networks and physical networks.

Modern Way

Cloud migration changes the volume and nature of traffic flows within and outside a corporate network. It also affects approaches to mitigating security risk. You will still manage on-premises networks, but will integrate your network into Azure.

Roles

To cover the broad impact of Cloud to the organization, we also added role guidance. While processes describe how activities are handled, the roles describe the people responsible for managing and completing these activities. This section outlines the accountabilities and requirements for various roles.

Don't over complicate

Keeping the cloud IT organizational role model simple and sharing the accountabilities of the roles early in the process of your Azure migration will reduce the IT organization's fear of losing their job when moving to the cloud. Set up a new organization to manage the new Azure based service and have them only work on the new service, leaving the on-premise support to the remainder of IT. Avoid building a heavy weight organization to maximize the utilization of the flexibility and time to market capabilities of cloud.

Roles do not necessary relate one on one to dedicated functions. Functions are more related to the Functional Hierarchies of an organization, and the required amount of resources in terms of Full Time Employees (FTE's).

While the following may make, it seem a huge team is required for proper cloud management, this is not the case. Roles can be assigned to existing functions and sometimes combined into one function. However, to encourage advocacy and remain focus, this assignment should be taken with care.

Topics for guidance

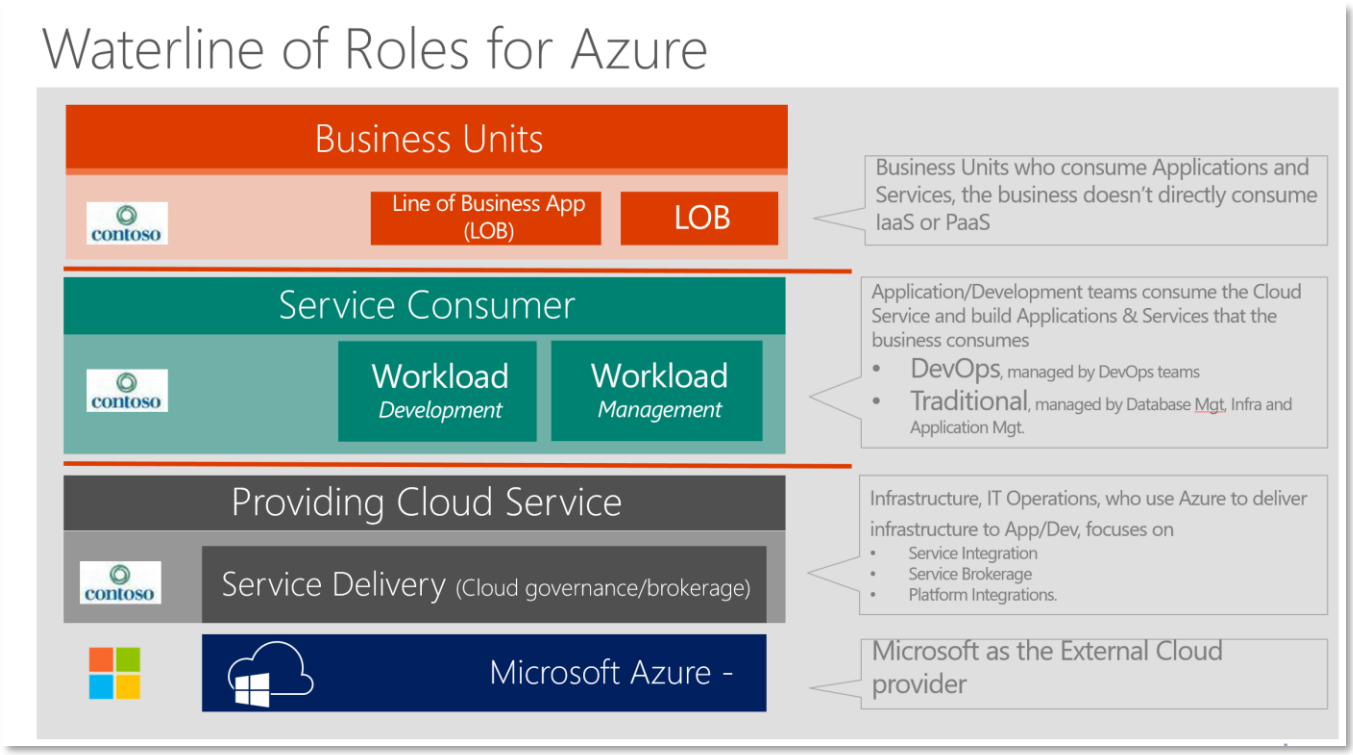
For each role, we provide guidance on some key areas:

- Description of Roles, to define the role
- Process areas of accountability, to highlight accountability and address process focus
- Required skillset, for providing technical and soft skills guidance
- Critical success factors, to set goals, ambitions and CSF's for the role

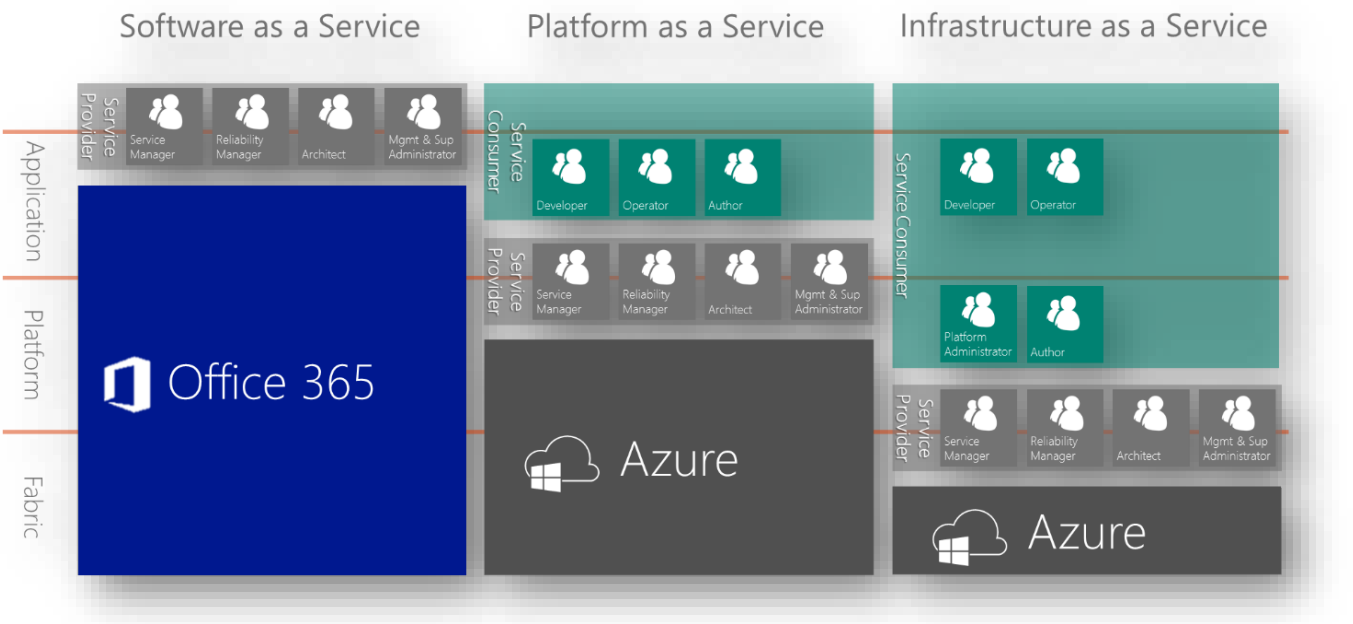
The tables on the neck page outline the roles necessary to support Cloud Services, this is just a start of the role types and descriptions of what they do. Realize that more than one person can be assigned to a role and that a person can have multiple roles.

Identifying the organizational areas

Cloud has similarities to outsourcing, where governance needs to be established within a demand supply situation. Within the cloud environment we can identify the Service Consumer organization and the Cloud Provider Organization.



Cloud presents itself in several models like SaaS, PaaS and IaaS, which does impact the accountability levels, and the related roles. The graphical representation (often referred as the Cloud Accountability Waterline) provides an overview of the roles applied to several cloud models.



In the following sections, list roles and associated guidance. This list is not a conclusive list but defines the minimum roles needed to effectively manage hybrid cloud services.

Service Consumer Roles – Ownership

These roles represent the ownership of the key cloud elements and do not include the comprehensive set of operational activities. The daily activities are run by the other roles within the Service Consumer organization in Azure.

Tenant Owner – Owns the Azure Subscription(s)

The tenant owner manages the all the tenants and subscriptions in Azure. The tenant owner is an important partner of the workload owner(s) and stakeholder on the service consumer side.

Process Areas of Accountability

Business Relationship Management

Financial Management

Capacity Management

Subscription Management

Service Lifecycle Management

Service Level Management

Skill Sets required (Keywords)

Business IT alignment

Financial (Capital and non-capital)

Business Demand and patterns

Critical Success Factors of Role

This role must be able to identify business demands and patterns and have demand conversations with the service provider about the required capacity and service roadmap, while helping make sure financial implications are understood and accepted.

Workload Owner

Responsible for a (set of) workload(s) running on or consuming the cloud services. In many cases the workload owner role is represented by a service owner or application owner within the service consumer organization.

Process Areas of Accountability

Financial management

Capacity Management

Service Level Management

Service Lifecycle Management

Identity and Access Management Management

Skill Sets required (Keywords)

Business IT alignment

Business Demand and patterns

Critical Success Factors of Role

This role is responsible for offering the required functionalities to business users.

Service Consumer roles - SLM/DevOps roles

Cloud and DevOps are often mentioned in the same sentence. It is true that cloud and particularly with PaaS services are suited very well to the DevOps philosophy and are aligned on how services are managed, operated and updated with the Development team and Operations team work with each other. Service Lifecycle management is important in Azure, and to drive that level of Service Level Management, the DevOps philosophy is an excellent approach to drive agility.

Within the Service Consumer organization, we identify the Development and the Operations role. This role can be applied to a traditional (development and administration) or a DevOps scenario. For DevOps, we recommend having a product owner and scrum master/architect leading and guiding the team using business value as a catalyst. Essential in DevOps is a culture where change, collaboration and recognition is the new normal. The breakdown between Development and Operations for deployment and management in the cloud will vary from customer to customer depending on where they are at in their journey bimodal mode 2 operations. There is no single role model for DevOps teams as it relies heavily on collaboration. The roles provided are an example of roles encountered in such teams (not conclusive).

Developer

This role develops solutions or code on infrastructure, platform or software level and is therefore responsible for:

- *Assessing (business and operating) requirements*
- *Creating technical design specifications (with acceptance criteria) in close cooperation with architects and estimates workload*
- *Development of service workload(s) (solutions) to meet business requirements*
- *Integrating solutions with other workloads, platforms or services*
- *Deployment of solutions, between environment*
- *Serving as liaison between partners, IT and vendors*
- *Participating in project planning processes*
- *Creating test case, scenario's and scripts based on business and technical requirements*
- *Creating and maintaining coding and unit testing*
- *Creating and maintaining functional and non-functional automated testing*
- *Running error detection and resiliency tests*
- *Providing information to perform Solution troubleshooting*
- *Share, test & deploy (custom) resources within development environment.*
- *Adopting and applying standards, policies and procedures during development and in the solutions.*
- *Staging artifacts deployment*
- *Providing feedback and requirements to run automated acceptance test*
- *Providing feedback and requirements to perform exploratory tests manually*
- *Providing feedback and requirements to run automated performance and load tests*

Process Areas of Accountability	Skill Sets required (Keywords)
Change Management	Team player
Release and deployment management	Analytical
Development processes	Business oriented
Knowledge Management	Flexible / adaptable
(Major) Incident and Problem Management	

Critical Success Factors of Role

Agility

Speed of development

Managing the Service Lifecycle management of applications and services

Smooth deployment

Operator

This role is responsible for the day to day operations of service workloads in the Cloud Platform.

- *Runs error detection and resiliency tests*
- *Performs Solution telemetry, performance monitoring and troubleshooting*
- *Manages monitoring for service workload(s)*
- *Performs day-to-day operations for service workload(s)*
- *Package software builds from the developers into one or multiple service templates*
- *Manages Service Accounts for solutions*
- *Creates and maintains release workflow and tools (development to QA to UAT to Prod with checkpoints)*
- *Production support*

Process Areas of Accountability

Request fulfilment

System Administration

Change Management

Access Management

(Major) Incident and problem management

Knowledge Management

Skill Sets required (Keywords)

Problem solving

Analytical

Communication skills

Critical Success Factors of Role

Manageability

Well tested

Stable and secure operations

Service Consumer roles – Platform Administration (IaaS only)

In the IaaS cloud service model the infrastructure level is applied in the cloud. The Operating Systems and the Server role related software is deployed and needs to be managed. The platform administrator is responsible for managing the software and settings, where the author is managing the templates to provide a standardized approach.

Platform Administrator

This role is responsible for administration of the infrastructure workloads in the cloud platform, for example SQL server within Azure (IaaS).

Process Areas of Accountability

Operations

Lifecycle Management

Software Update Management

Incident and Problem Management

Skill Sets required (Keywords)

Problem solving

Analytical

Communication skills

Critical Success Factors of Role

Platform stability and lifecycle management are managed and perform well

Author

This role is designing, building and maintaining the infrastructure templates and runbooks for Azure. This role can either have a developer background or an operations background. Where the role resides in the organizational chart will depend on where the organization is on its journey to bimodal mode 2 operations

Process Areas of Accountability

Problem Management

Configuration Management

Deployment

Skill Sets required (Keywords)

Analytical

Automation

Scripting, PowerShell DSC, ARM Templates

Standardization approach

Critical Success Factors of Role

Standardized environment, the creation of runbooks and templates and the automation of manual tasks to increase efficiency and standardization.

Service Provider roles – Service Delivery

The following roles are applicable for an organization who is responsible for delivering cloud services towards the consumer.

Service Manager

Represents all Cloud based services and its alignment and integration as consumed by the Service Consumer Organizations within the Company. Accountable and Responsible for:

- *Overall Customer Satisfaction related to Cloud Consumption*
- *All Service Support and Service Delivery actions taken to help make sure aligned Cloud Service Delivery*

In a DevOps environment, this central Service Manager, manages the relationship between Microsoft and their organization but does not manage the cloud specific services being consumed.

Process Areas of Accountability

Business Relationship Management

Financial Management

Service Level Management

Demand Management

All processes (internal process integration)

Skill Sets required (Keywords)

Ability to communicate with Senior business leaders

Negotiator

Leadership and Change Management (Lead Change and Adoption processes)

Overview of the Microsoft Cloud portfolio

Ability to define the critical success factors for Azure

Critical Success Factors of Role

Cloud Consumption and Cloud effectiveness

Cloud awareness and removal of Cloud blockers

Customer Satisfaction and Business alignment

Inter-process awareness and drives for continual service improvement

Drives innovation and adoption of new Azure services to enable business velocity and lifecycle management

Reliability Manager

This role is responsible for planning and maintaining the availability and reliability of IT services to help make sure that IT can effectively meet service targets in to support business objectives.

The scope of Reliability includes Confidentiality, Integrity, Availability, Continuity and Capacity. To help make sure reliability this requires:

- *Planning. Gathering and translating business requirements into IT measures*
- *Implementation. Building the various plans and helping make sure that they can meet expectations*
- *Monitoring and Improvement. Proactively monitoring and managing the plans and making necessary adjustments*

The reliability manager is responsible for managing the different reliability aspects of the cloud service: capacity, availability, continuity and security (confidentiality and integrity) management in relation to the service level targets, policies and compliancy regulations.

Process Areas of Accountability

- Availability & Continuity Management
- Capacity Management
- Information Security Management
- Regulatory Policy and Compliancy Management

In a DevOps environment, the central Reliability Manager, manages the relationship between Microsoft and the organization not the individual services in Azure. Policy and compliance is governed by using straight through processing in the release pipeline and use of pre-approved compliant design, deployment and configuration patterns.

Skill Sets required (Keywords)

Compliancy regulations

Metrics

DR concepts

Security concepts

Auditing

Critical Success Factors of Role

Maximize the use of cloud capabilities leveraging Azure reliability and Azure security to meet business requirements

Availability of actively managed Reliability and Security related planning processes on a business level (e.g. Business Capacity Management)

Data integrity and confidentiality maintained

Critical business services available during significant failures

Services available to users when needed and IT capacity aligned to business needs

Cloud Architect

Oversees the Business and Cloud roadmap to build the company IT Cloud roadmap. Advises and validates on the applied Cloud Principles and Concepts. Drives Cloud innovation.

Process Areas of Accountability

Reliability Management

Security and Information Management

Lifecycle Management

Demand Management

Skill Sets required (Keywords)

- Innovator
- Business systems
- IT systems
- Cloud principles, concepts and solutions

In a DevOps environment, the Chief Cloud Architect, manages the Microsoft Azure Service overall, but does not manage the services that make use of Azure, these are managed by the DevOps teams

Critical Success Factors of Role

Define and publish roadmaps to support the service design process and cloud deployment patterns

Ensure current state is reliable

Facilitate Business/IT alignment

Develop long-term possible solutions and choices

Describe future consequences and possibilities

Operations Manager

Responsible for the management and operations of the hybrid Azure environment. Defines, oversees and provides the manageability of the workloads in Azure by providing the manageability platform that enables:

- *Monitoring*
- *Identity Management*
- *Automation workflows*
- *Testing facilities*
- *Reporting (and Business Intelligence)*
- *Backup and Disaster recovery*

In a DevOps environment, the "central" Operations Manager manages the Cloud Provider Services, but does not execute the services that make use of Azure services, these are managed by the DevOps teams

.

Process Areas of Accountability

Service Operations processes, including but not limited to Incident(Major), Problem, Change, Request and Release Management and System management

Skill Sets required (Keywords)

Cloud principles, concepts and solutions

System Management Software concepts and products, such as: OMS, EMS and software management

Critical Success Factors of Role

Stability and Manageability of the system, and helps make sure that all applications are monitored, managed and operate predictably.

Service Provider Roles - Cloud Service Provider

In the world of Azure, the IT department is a partner to the BizDevOps teams within the enterprise. Being a Service Provider requires IT to work as a Partner, building a Cloud competency center and to enable BizDevOps on Azure. IT is responsible for brokering new requirements and managing updated services. Instead of building services from the ground up, IT utilizes Azure capabilities as the building blocks for updated business services

The role below is not relevant for IT organizations utilizing Azure, though if IT organizations have Private clouds they will require the Fabric Administration roles below

Fabric and Fabric Management Administrators (private cloud)

The Fabric Administrator manages the Compute, Network and Storage and virtualization layer of the private cloud platform.

The Fabric Management Administrator is responsible for managing and operating the monitoring and automation components of the private cloud platform. Based on these systems the cloud services are provided towards the service consumer including request portals, patching options, dashboards and request automation

.

Process Areas of Accountability

Request fulfilment

Availability and continuity management

Asset and configuration management

Change management

System administration

Incident and problem management

Service Monitoring

Capacity Management

Skill Sets required (Keywords)

Deep technical knowledge of the platform and automation possibilities for manageability and consumption

Critical Success Factors of Role

Level of automation

Time needed for request fulfillment

Effectiveness of the platform: ratio between Datacenter Capacity and used private cloud capacity