Business

Technical appendix to Managed Applications Service Description

Managed Applications on Azure

Table des matières

1 Overview of the Service

1.1 Overall description

The document is an appendix to the Managed Application Service Description. It provides service description and further details for the

- MANAGED BUSINESS APPLICATION ON AZURE
- MANAGED CLOUD NATIVE SERVICES ON AZURE

2 Managed Cloud Native Services on Azure

Customer's business application deployed on Azure are dependent on Azure Cloud Native Services (IaaS, PaaS). The Provider provides the managed services necessary to ensure service assurance and change management for those dependences, as well as the configuration and deployment for building and recovering them.

2.1.1 The cloud native services

One can typically distinguish 3 categories of services:

- The user plane services: if a business application depends on it, the business application is likely to be affected by a defect of it. The service does not have persistent data, therefore the recovery does not necessitate data restore.
- The data services: if a business application depends on a data service, the business application is likely to be affected by a defect of it. The service has persistent data, therefore a recovery may necessitate data restore. Data loss, data corruption may affect the business application as well.
- The other services: the business application does not depend on them. Most of those services are used for automation, observation, migration. The loss of the service is not likely to affect the business application. Some of the services are used for managing the user plane and data plane services of the business application, some others have specific usage for which a scope of work shall be established would the customer requires The provider to leverage them as part of the managed service provided.

User plane services	Data services	Other services	
Compute □ App Service □ App Service (Linux) □ Azure Functions □ Container Instances □ Dedicated Host □ Kubernetes Service □ Service Fabric □ Virtual Machines □ VM Scale Sets **Networking** □ Application Gateway □ Azure Bastion □ Azure DNS □ Azure Firewall □ Azure Front Door □ Express Route □ Load Balancer □ Network Watcher □ Private Link □ Traffic Manager □ Virtual Network □ VPN Gateway **Integration** □ API Management □ Logic Apps □ Notification Hubs □ Service Bus **Media** □ Azure CDN □ Media Services **Automation** □ Automation □ Site Recovery	**Storage** □ Azure Storage □ Managed Disks □ StorSimple **Databases** □ Cosmos DB □ Database for MariaDB □ Database for MySQL □ Database for PostgreSQL □ Redis Cache □ SQL Database □ SQL Server Stretch **Identity & Security** □ Azure Active Directory □ Azure AD B2C □ Azure AD DS □ Azure Key Vault □ Azure Lighthouse	**Management & Governance** □ Azure Advisor □ Azure Arc □ Azure Backup □ Azure Batch □ Azure Blueprints □ Azure Monitor □ Azure Policy □ Azure Portal □ Cloud Shell □ Container Registry □ Cost Management □ Scheduler **Security management** □ Azure Sentinel □ Security Center **Integration** □ Event Grid	**Development** □ App configuration □ Azure DevOps □ DevTest Labs □ Lab Services □ Visual Studio App Center **Migration** □ Azure Migrate □ Data box □ DB Migration Service

Azure Cloud Native services by category

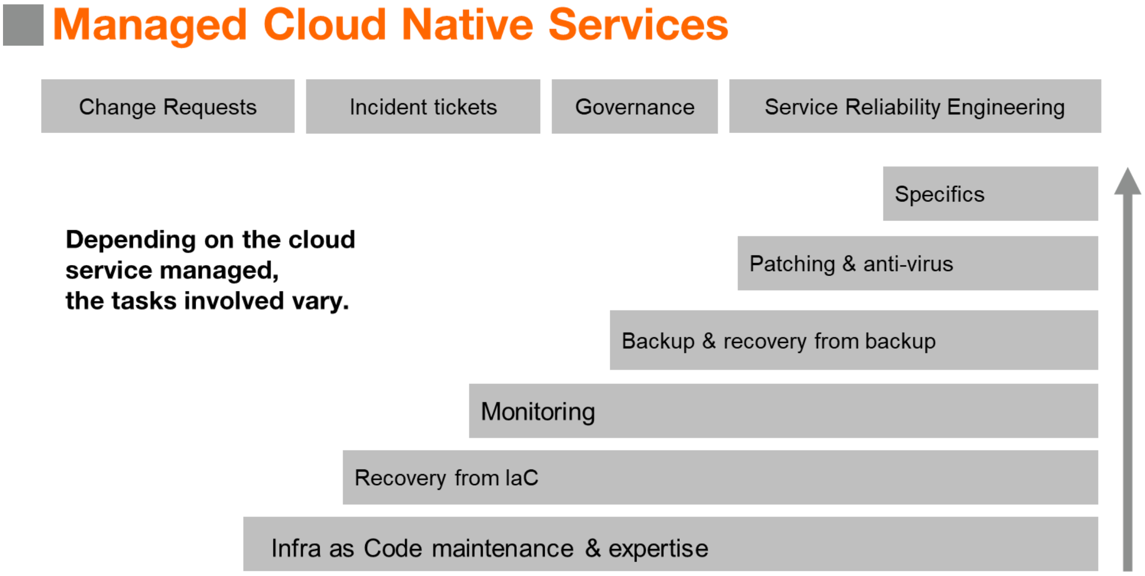
2.1.2 Tasks involved Cloud Native service management.

The tasks involved for the management of a cloud native service depends on the service. They consist in:

- Configuring and deploying the service: Infrastructure as Code is leveraged in order to configure the service, the observability, the backup. Level 3 expertise on the service is leveraged for proper implementation thanks to the scope of work (refer to detailed description of build and SRE services)
- Applying the security group and access control policy defined by the customer.
- Service recovery thanks to Infrastructure as Code: in case of failure, most of the services requires to be recovered thanks to a redeployment. Re-configuring the service manually from scratch is not an efficient option: it takes time and is error prone. This is why recovery / redeployment from Infrastructure as Code is preferred.
- Supervision and remedial consists in watching for alarms raised on the service during the monitoring range (typically: 8x5 or 24x7). When an alarm occurs, an incident ticket is raised, a

priority is assigned, the customer is notified. Then remedial action is taken thanks to the procedures made available to Level 2 / 1 by the Level 3. The remedial on a cloud native service may be necessary to restore the service of the business application. Would the procedure not remedy to the incident, then the incident is escaladed to the Level 3. Would the root cause be the CSP itself, then the incident is raised to the CSP by the Level 3.

- Backup and restore: Depending on the service (if the service has persistence), it is necessary to backup the service data. The management service consists in configuring the backup solution and monitoring the proper run of it. Note: the backup solution must be subscribed separately e.g. Azure backup. Restoring the service on incident may involve restoring the data from a backup.
- OS patching and anti-virus: keeping OS up to date and virus free is a managed service for Managed Virtual Machine / Managed OS. Please refer to the detailed description.
- Specifics: some cloud native services may have specific configuration or management tasks.
- Business application specifics: by default, standard alerts are watched. The configuration of alerts, logs on a cloud native service which are specific to a business application is subject to a specific scope of work.



Managed Cloud Native Services

Change Requests

Incident tickets

Governance

Service Reliability Engineering

Depending on the cloud service managed, the tasks involved vary.

Specifics

Patching & anti-virus

Backup & recovery from backup

Monitoring

Recovery from IaC

Infra as Code maintenance & expertise

Tasks involved in managed services for cloud native service

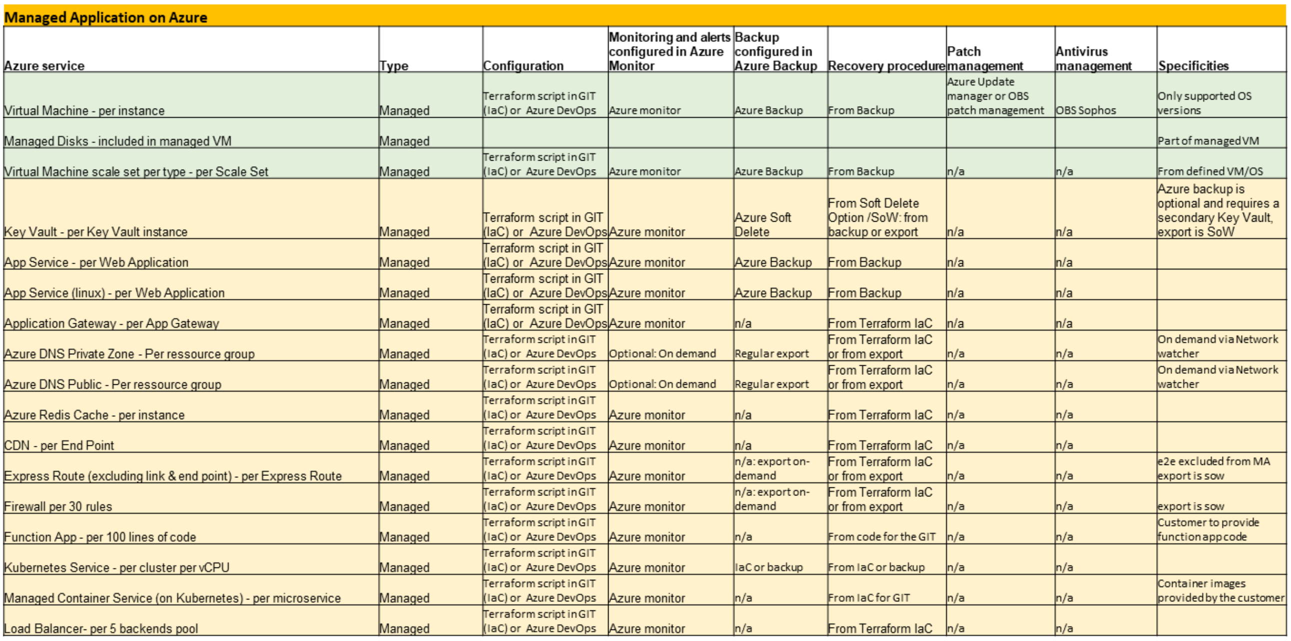
Depending on the cloud native service managed, more or less management tasks are necessary and included in the managed service. This drives the complexity of the managed service.

The tasks involved typically depends on the category of the cloud native service, whether user plane, data plane on which the business application depends, or other services upon which the business application does not depend.

	Charging model	User plane services	Data plane services	Other services
Purpose		Used to support customer application	Used to support customer application	Used to operate user plane or data plane
Build	One-time charge	IaC in Git, pushed via CI / CD	IaC in Git, pushed via CI / CD	IaC in Git, pushed via CI / CD

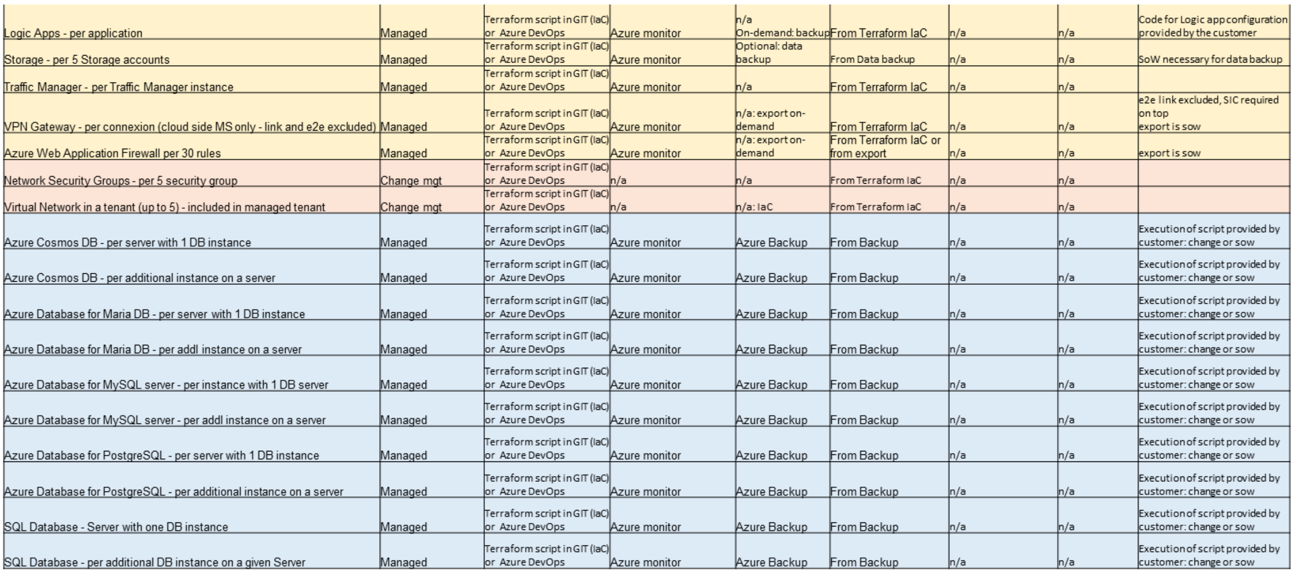
	based on SoW			
Maintaining IaC without changes	Monthly recurring charge	Yes	Yes	Yes
Monitoring & alerts	Monthly recurring charge	Yes	Yes	
Configuration restore on incident	Included in MRC	Yes, from IaC or export	Yes, from IaC or backup	Yes, from IaC when applicable
Data backup and restore on incident	Included in MRC		Yes	
Network and Security Management	Based on SoW	Optional: Based on SoW	Optional: Based on SoW	
Service Desk	Per incident ticket or percentage	Yes	Yes	Yes
Change Management	Per change, in Tokens vs complexity	Via IaC in Git, pushed via CI / CD.	Via IaC in Git, pushed via CI / CD.	Via IaC in Git, pushed via CI / CD
Disaster recovery	Specific design and quote	Optional: Based on SoW	Optional: Based on SoW	

2.1.1 Table of tasks involved in the management a Cloud Native service



Managed Application on Azure

Azure service	Type	Configuration	Monitoring and alerts configured in Azure Monitor	Backup configured in Azure Backup	Recovery procedure	Patch management	Antivirus management	Specificities
Virtual Machine - per instance	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	Azure Update manager or OBS patch management	OBS Sophos	Only supported OS versions
Managed Disks - included in managed VM	Managed							Part of managed VM
Virtual Machine scale set per type - per Scale Set	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	From defined VM/OS
Key Vault - per Key Vault instance	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Soft Delete	From Soft Delete Option /SoW: from backup or export	n/a	n/a	Azure backup is optional and requires a secondary Key Vault, export is SoW
App Service - per Web Application	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	
App Service (linux) - per Web Application	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	
Application Gateway - per App Gateway	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a	From Terraform IaC	n/a	n/a	
Azure DNS Private Zone - Per ressource group	Managed	Terraform script in GIT (IaC) or Azure DevOps	Optional: On demand	Regular export	From Terraform IaC or from export	n/a	n/a	On demand via Network watcher
Azure DNS Public - Per ressource group	Managed	Terraform script in GIT (IaC) or Azure DevOps	Optional: On demand	Regular export	From Terraform IaC or from export	n/a	n/a	On demand via Network watcher
Azure Redis Cache - per instance	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a	From Terraform IaC	n/a	n/a	
CDN - per End Point	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a	From Terraform IaC	n/a	n/a	
Express Route (excluding link & end point) - per Express Route	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a: export on-demand	From Terraform IaC or from export	n/a	n/a	e2e excluded from MA export is sow
Firewall per 30 rules	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a: export on-demand	From Terraform IaC or from export	n/a	n/a	export is sow
Function App - per 100 lines of code	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a	From code for the GIT	n/a	n/a	Customer to provide function app code
Kubernetes Service - per cluster per vCPU	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	IaC or backup	From IaC or backup	n/a	n/a	
Managed Container Service (on Kubernetes) - per microservice	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a	From IaC for GIT	n/a	n/a	Container images provided by the customer
Load Balancer- per 5 backends pool	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a	From Terraform IaC	n/a	n/a	



Logic Apps - per application	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a On-demand: backup	From Terraform IaC	n/a	n/a	Code for Logic app configuration provided by the customer
Storage - per 5 Storage accounts	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Optional: data backup	From Data backup	n/a	n/a	SoW necessary for data backup
Traffic Manager - per Traffic Manager instance	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a	From Terraform IaC	n/a	n/a	
VPN Gateway - per connexion (cloud side MS only - link and e2e excluded)	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a: export on-demand	From Terraform IaC	n/a	n/a	e2e link excluded, SIC required on top export is sow
Azure Web Application Firewall per 30 rules	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	n/a: export on-demand	From Terraform IaC or from export	n/a	n/a	export is sow
Network Security Groups - per 5 security group	Change mgt	Terraform script in GIT (IaC) or Azure DevOps	n/a	n/a	From Terraform IaC	n/a	n/a	
Virtual Network in a tenant (up to 5) - included in managed tenant	Change mgt	Terraform script in GIT (IaC) or Azure DevOps	n/a	n/a: IaC	From Terraform IaC	n/a	n/a	
Azure Cosmos DB - per server with 1 DB instance	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	Execution of script provided by customer: change or sow
Azure Cosmos DB - per additional instance on a server	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	Execution of script provided by customer: change or sow
Azure Database for Maria DB - per server with 1 DB instance	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	Execution of script provided by customer: change or sow
Azure Database for Maria DB - per addl instance on a server	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	Execution of script provided by customer: change or sow
Azure Database for MySQL server - per instance with 1 DB server	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	Execution of script provided by customer: change or sow
Azure Database for MySQL server - per addl instance on a server	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	Execution of script provided by customer: change or sow
Azure Database for PostgreSQL - per server with 1 DB instance	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	Execution of script provided by customer: change or sow
Azure Database for PostgreSQL - per additional instance on a server	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	Execution of script provided by customer: change or sow
SQL Database - Server with one DB instance	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	Execution of script provided by customer: change or sow
SQL Database - per additional DB instance on a given Server	Managed	Terraform script in GIT (IaC) or Azure DevOps	Azure monitor	Azure Backup	From Backup	n/a	n/a	Execution of script provided by customer: change or sow

Table of tasks involved in the management of cloud services (extract of services)

2.1.2 Tooling used for cloud native managed services

Azure tooling and The Provider backend operations tooling are leveraged to deliver the managed services. Would the customer require the use of a different tooling, the feasibility shall be confirmed with The Provider and the RACI and work-units may be revised.

Process	Tool used by The Provider MA delivery
Configuration of the infrastructure	Terraform script (The Provider Clouds / multicloud) Azure DevOps GIT referential CI / CD
Supervision solution	Azure Monitor with connector to The Provider supervision
Backup	Azure Backup (incl snapshots)
OS patching solution	Azure Update Manager The Provider MA Patching Tool (The Provider Cloud platforms and Multicloud) The Provider OS Factory
Antivirus solution	The Provider MA Antivirus Tool Sophos (then TrendMicro)
Logging solution	Azure Insight (on demand based on Scope of Work) Azure Log Analytics (on demand based on Scope of Work)
Recovery	From backup when applicable From Terraform script in GIT when applicable Ideally from up to date Infra as code with CI/CD
Admin connectivity	VPN to The Provider CASA Zone
Portal for access to MA contract, incident & change ITSM	The Provider CloudStore

2.1.3 General pre-requisites to the run of managed services

The following pre-requisites are necessary to all managed services:

- The Customer shall have defined a valid architecture. (The Provider can optionally provide Professional Services for architecture definition).

- The Customer shall have **a valid subscription to Azure including subscription to Azure Support plan and procure the Azure resources and Azure support plan. The Provider can optionally supply this subscription inclusive of Azure support (ref to Multi-Cloud Ready offer for Azure), however, the subscription, the IaaS resources, the Azure support are not part of the Managed Services.** The Managed Services will leverage this support contract to escalades incident to Azure CSP.
- Azure platform for the Customer shall be urbanized alongside best practices of Azure's landing zone or shall offer comparable services.
- The Provider proposes a default RACI depending on the class of transition and the resource managed. As a pre-requisite to the project, The Provider and the Customer shall agree on the RACI.
- Agreement on the tooling used for GIT, CI / CD chain, Monitoring, Logging and Alerting solution.
- Additional pre-requisites are required when transition is not the entire responsibility of The Provider (e.g. required for partial build like "Operations Build" or "Backend Build" models, refer to chapter 8 of the document: Build Scope of Work)

In the case of Fully Managed service, The Provider is using its own Git, CI / CD chain, Monitoring, Logging and Alerting solution.

In the case of a Co-managed service, The Provider and the Customer agree on the Git, CI / CD chain, Monitoring, Logging and Alerting solution to be used. By default, the tooling is

- Either based on Azure tools i.e. Azure DevOps, Azure Monitoring
- Or based on generic multi-cloud tooling proposed by The Provider e.g CaasCad (Prometheus, Grafana,…)

This tooling not included in the Managed Applications work units and can be purchased separately as part of Azure Subscription or as a multi-cloud tooling proposal made by OBS.

2.1.4 Criteria for the run of a managed cloud native service component

Criteria shall be met with an approval by Level 2 before turning a cloud native component to an active manage service (i.e. Run) by the Level 2 / Level 1 operations. The owner of the Build and of the Level 3 support owns the responsibility of making sure that the criteria are met:

- The architecture and deployment of the service shall be defined.
- The service shall be deployed thanks to Infrastructure-as-Code and tested prior to transitioning to the run team. Typically, successful testing in pre-production, with a pre-production environment iso-production. Note: IaC is necessary to recover the services in case of major failure.
- The use of the service shall be explained to the operation team.
- The security policies and access control shall have been configured.
- The access shall have been configured allowing The Provider Level 2 teams access.
- The service shall export the necessary metrics towards Azure Monitor.
- The data backup shall be configured in Azure Backup when backup is applicable.
- The disaster recovery shall be configured when applicable.
- The troubleshooting and service restoration procedures shall be provided to Level 2.

- Whereas a procedure requires logs or dashboard those shall have been developed and deployed prior to transferring to run phase.
- A remedial procedure on incident shall not last more than 15 minutes. Beyond, that time amount, the effort would be charged on time base.

3 The build of services & managed services on Azure

When the build effort is uncertain from pre-sales documentation, an assessment is proposed at the beginning of the build project by The Provider Cloud Expert Services. During this assessment, the following tasks are performed:

- Collection of the architecture diagrams with dependences, HLD, LLD of applications, and infrastructure to be managed and any other useful information.
- Check of the inventory of resources to be deployed and managed.
- Review for each of the dependence the remaining work requested to The Provider for completing the build to reach readiness for the run. Review the criteria for a resource build to qualify to a given model of build. Hence determining for each resource which build model applies: No build, Backend build, Operations Build or Full Build.
- Confirmation that the pre-required tools for operations are in place (or alternatively agreeing on a specific scope of work for different tooling if agreeable).
- Establishing requested responsibilities defined between the customer and The Provider (RACI) for build and for the run.
- Identifying potential limitations on the managed application service if criteria are not met.

3.1 Criteria for qualifying as “backend build” model a.k.a class 2 SoW for a resource:

The “backend build” scope of work model for a resource is used for:

- a resource/service in scope for managed service for which the infrastructure is already built and deployed by the customer leveraging Infrastructure-as-Code.
- And, for which Azure tooling is fully configured and operational prior to transition under customer’s responsibility. The tooling used shall be:
  - Azure Monitor for supervision with proper alerts defined.
  - Azure Backup properly configured and functional
  - Update Manager configured for VM patching.
  - Remedial and troubleshooting procedures on known incident are defined and provided.
  - Recovery procedures to be used are defined and provided by the customer.
- And customer provides documentation i.e. schema, HLD and DAT/LLD, architecture explaining how availability & HA, monitoring, security policies and access control, backup, disaster recovery, baseline security, SLA are achieved.

The build effort provided by The Provider in the “backend build” includes integrating the alarms from Azure Monitoring to The Provider backend systems, capturing the procedural guides provided by the customer into The Provider knowledge repository of operations, and operations readiness. It includes as well getting the administrative backend, The Provider ITSM, the portal and billing readiness for operations.

3.2 Criteria for qualifying as “operations build” model a.k.a class 4 SoW for a resource:

The “operations build” scope of work model for a resource is used for:

- a resource/service in scope for managed service for which the infrastructure is already built and deployed by the customer leveraging Infrastructure-as-Code.
- And, customer provides documentation i.e schema, HLD and DAT/LLD, architecture explaining how availability & HA, monitoring, backup, disaster recovery, baseline security, SLA are achieved.
- And, agreement reached between the customer and The Provider to use the Azure and The Provider backend tooling.

The build effort provided by The Provider in the “operations build” includes that of the “backend build” plus the configuration and deployment of Azure tooling thanks to Infrastructure as Code and of The Provider backend i.e:

- Azure Monitor for supervision with alerts
- Azure Backup configuration and deployment
- Update Manager configuration for VM patching
- Anti-virus configuration for VM
- Use of standard remedial and troubleshooting procedures on known incident for the cloud native service.
- Use of standard recovery procedures for the cloud native service.

For further details on the operations per service, please refer to **Chapter 9: detailed description per cloud service.**

3.3 Criteria for qualifying as “full build” model a.k.a class 5 SoW for a resource:

The “full build” scope of work model for a resource is used for:

- a resource/service in scope for managed service not yet built and deployed.
- And, customer provides documentation i.e schema, HLD and DAT/LLD, architecture explaining how availability & HA, monitoring, backup, disaster recovery, baseline security, SLA are achieved.
- And, agreement reached between the customer and The Provider to use the Azure and The Provider backend tooling.

The build effort provided by The Provider in the “full build” includes that of the “backend build” plus that of the “operational build” plus

- The configuration of the Landing Zone and the infrastructure of the resource leveraging Infrastructure as Code.

For further details on the operations per service, please refer to **Chapter 5: detailed description per cloud service.**

For further details of Infrastructure as Code for full build model, **please refer to chapter Infrastructure as code methodology.**

3.4 Mitigation in case of pre-requisites or criteria not met:

The assessment may reveal that criteria are not met for qualifying to a given build model. Then 3 options are possible:

- the scope of work shall be revisited with a more appropriate build model. This may affect the duration of the project, efforts, quote and price.
- the customer may remedy to the missing criteria. This may affect the duration of the project and project management and coordination efforts.
- the customer and The Provider may agree to live with some limitations in the management capabilities and responsibilities due to the missing criteria.

Would the project be delayed and would resources effort be overspent by The Provider as result of pre-requisites and criteria under customer's responsibility not being met, then The Provider would be entitled to charge the overspent effort based on time and material.

3.5 Charging model for build

Service	Work Unit
Project management	Time and material
Service Implementation Coordination	Time and material
Service Reliability Engineer	Time and material
Technical Architect	Time and material (when necessary for documentation)
Full build model - 1st Resource Unit*	One Time Charge per resource
Full build model - subsequent Resource Unit of same type*	OTC per resource
Operations build model - 1st Resource Unit*	OTC per resource
Operations build model - subsequent Resource Unit same type*	OTC per resource
Backend build model - 1st Resource Unit *	OTC per resource
Backend build model - subsequent Resource Unit same type*	OTC per resource

Resource unit*: please refer to Chapter 5: detailed description per cloud service for the definition of the Resource Unit per cloud native service.

4 Detailed responsibilities and accountabilities

The following tables describe the standard default responsibilities between The Provider and the customer depending **on the build model**.

The following tables describe the standard default responsibilities between The Provider and the customer depending on classes of service. Those may be amended with mutuel consent depending on project.

- R stands for responsible
- A stands for Accountable
- C stands for Contributor
- I stands for Informed

4.1.1.1 RACI for Managed OS

Service Implementation	The Provider	Customer	The Provider	Customer	The Provider	Customer
OS Server infrastructure implementation	Full build		Operations build		Backend build	
Deployment of the infrastructure	R, A	I	I	R, A	I	R, A

Deployment of LAN components	R, A	I	I	R, A	I	R, A
Deployment of DNS and NTP services	R, A	I	R, A	I	I	R, A
Backup tools for operations (Azure backup & Azure Snapshots)	R, A	I	R, A	I	I	R, A
Deployment of the OS patching solution (Azure Update Mgt)	R, A	I	R, A	I	I	R, A
Deployment of the Antivirus solution	R, A	I	R, A	I	SoW	SoW
Deployment of the supervision solution (Azure Monitor)	R, A	I	R, A	I	I	R, A
Deployment of the logging solution (Azure Insight)	R, A	I	R, A	I	I	R, A
Deployment of security groups and firewall rules	R, A	I	SoW	SoW	I	R, A
Recovery procedure (Infra as Code, restore, other…)	R, A	I	I	R, A	I	R, A
Testing and validation of infrastructure implementation	R	A	I	R, A	I	R, A
Testing and validation of Azure tooling implementation and lifecycle management	R	A	R	A	I	R, A
OS Server Implementation						
Evaluation or deployment of the operating system	R, A	I	R, A	I	I	R, A
Deployment of new packages	R, A	I	R, A	I	R, A	I
Test and validation of operating system implementation for new packages	R, A	I	R, A	I	R, A	I
Service implementation documentation						
Conception, architecture and low-level design for infrastructure	I	R, A	I	R, A	I	R, A
Implementation and operation documentation for infrastructure	R, A	I	I	R, A	I	R, A
Conception and low-level design for tooling (Azure)	R, A	I	R, A	I	I	R, A
Implementation & operation documentation for tooling (Azure)	R, A	I	R, A	I	I	R, A

4.1.1.2 RACI for Database as a Service

Service Implementation	The Provider	Customer	The Provider	Customer	The Provider	Customer
Database aaS services conception and implementation	Full build		Operations build		Backend build	
Maintenance of Infrastructure architecture referential	R, A	I	I	R,A	I	R,A
Maintenance of tooling configuration referential	R, A	I	R, A	I	I	R,A
Deployment of the infrastructure	R, A	I	I	R,A	I	R,A
Deployment of the supervision solution (Azure Monitor)	R, A	I	R, A	I	I	R,A
Deployment of the logging solution (Azure Insight) (optional)	R, A	I	R, A	I	I	R,A
Deployment of the backup solution (Azure Backup, Snapshot)	R, A	C, I	R, A	C, I	I	R,A
Recovery procedure for infrastructure from referential (Infra as code, restore from backup, other…)	R, A	C, I	I	R, A	I	R,A

Recovery procedure for tooling from referential (Infra as code, restore, other…)	R, A	C, I	R, A	C, I	I	R,A
Testing and validation of infrastructure implementation	R, A	I	I	R,A	I	R,A
Testing and validation of tooling implementation and lifecycle management	R, A	I	R, A	C, I	I	R,A
Customer provided script execution on DB instance	R	A, I	R	A, I	R	A, I
OBS script execution on DB instance	R, A	C, I	R, A	C, I	R, A	C, I
Service implementation documentation						
Conception, architecture and low-level design for infrastructure	C, I	R, A	I	R,A	I	R,A
Implementation and operation documentation for infra	R, A	C, I	I	R,A	I	R,A
Conception and low-level design for tooling (Azure)	R, A	C, I	R, A	C, I	I	R,A
Implementation & operation documentation for tooling (Azure)	R, A	C, I	R, A	C, I	I	R,A

Service Operation	**The Provider**	**Customer**	**The Provider**	**Customer**	**The Provider**	**Customer**
Database aaS services operations	Full build		Operations build		Backend build	
Monitoring through Azure Monitor	R	I	R	I	R*	I
Investigation through Azure Insights	R, A	C,I	R, A	C,I	R*	A
Restore from Infra as Code and backup	R, A	C,I	R, A	C,I	R*	A
Changing capacity of database instance	R, A	C,I	C, I	R, A	C, I	R, A
ITSM operations						
Change Management	R	A	R	A	R	A
Incident Management	R, A	R**,I	R, A	R**,I	R, A	R**,I
Event management	R, A	I	R, A	I	R, A	I
Baseline security management	R	A	SoW	SoW	SoW	SoW
Configuration management	R, A	C, I	R	A	R	A
Report management via SDM service	R, A	C, I	R, A	C, I	R, A	C, I
Invoicing management	R, A	I	R, A	I	R, A	I

R*: within the limitations of tooling provided by the Customer
R**: in co-management model, customer may have joint responsibilities related to the activity & incident

4.1.1.3 RACI for other Native Services managed

Service Implementation	**The Provider**	**Customer**	**The Provider**	**Customer**	**The Provider**	**Customer**
Native service infrastructure implementation	Full build		Operations build		Backend build	
Deployment of the infrastructure	R, A	I	I	R, A	I	R, A
Backup tools for operations (Azure backup)(1)	R, A	I	R, A	I	I	R, A
Deployment of the supervision solution (Azure Monitor)(1)	R, A	I	R, A	I	I	R, A
Deployment of the logging solution (Azure Insight) optional (1)	R, A	I	R, A	I	I	R, A
Deployment of security groups and firewall rules	R, A	I	SoW	SoW	I	R, A

Recovery procedure (Infra as Code, restore, other…)	R, A	I	I	R, A	I	R, A
Testing and validation of infrastructure implementation	R	A	I	R, A	I	R, A
Testing and validation of Azure tooling implementation	R	A	R	A	I	R, A
Packages						
Deployment of new packages (1)	I	R, A	I	R, A	I	R, A
Service implementation documentation						
Conception, architecture and low-level design for infrastructure	C, I	R, A	I	R,A	I	R,A
Implementation and operation documentation for infrastructure	R,A	I	I	R,A	I	R,A
Conception and low-level design for tooling (Azure)	R,A	I	R,A	I	I	R,A
Implementation & operation documentation for tooling (Azure)	R,A	I	R,A	I	I	R,A

Service Operation	**The Provider**	**Customer**	**The Provider**	**Customer**	**The Provider**	**Customer**
Native service operations	Full Build		Operations build		Backend build	
Monitoring (1)	R, A	I	R, A	I	R*	A
Backup (1)	R	A	R	A	R*	A
Restore from Infra as Code and backup (1)	R, A	C,I	R, A	C,I	R*	A
Security groups, Firewall rules setting	R	A	SoW	SoW	I	R,A
ITSM operations						
Change Management	R	A	R	A	R*	A
Incident Management	R, A	R**,I	R, A	R**,I	R*, A	R**,I
Event management	R, A	I	R, A	I	R*	A
Baseline security management	R	A	SoW	SoW	SoW	SoW
Report management via SDM service	R, A	I	R, A	I	R, A	I
Invoicing management	R, A	I	R, A	I	R, A	I

R*: within the limitations of tooling provided by the Customer
R**: in co-management model, customer may have joint responsibilities related to the activity & incident
(1) When applicable as per detailed description per service

5 Detailed description of the run tasks per cloud service (Extract)

5.1 API Management

5.1.1 Description

Azure API management allows the secured publication of APIs at scale to developers, partners and employees.

5.1.2 Build to run service included in the OTC

5.1.2.1 Build service pre-requisite

- Refer to generic description.

5.1.2.2 Build to run service

- Refer to generic description.

5.1.3 RUN services included in the MRC

5.1.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the API management.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.1.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Metrics supported for API Management service:

- BackendDuration
- Capacity
- ConnectionAttempts
- Duration
- EventHubDroppedEvents
- EventHubRejectedEvents
- EventHubSuccessfulEvents
- EventHubThrottledEvents
- EventHubTimedoutEvents
- EventHubTotalBytesSent
- EventHubTotalEvents
- EventHubTotalFailedEvents
- FailedRequests
- NetworkConnectivity

- OtherRequests
- Requests
- SuccessfulRequests
- TotalRequests
- UnauthorizedRequests
- WebSocketMessages

Alerts observed

- FailedRequests
- UnauthorizedRequests

5.1.3.3 Backup and restore for the Site Recovery configuration

Service restore: The Continuous Deployment chain is used to redeploy the same configuration of the Site Recovery from the reference Git.

5.1.3.4 Azure SLA High Availability and Disaster Recovery inter-region

The service can be deployed in multi-region by design.

5.1.3.5 Limitations & pre-requisite

Whenever the API is customized, there should be procedures provided by the customer describing how to monitor and troubleshoot the API.

5.1.4 Charging model

Work Unit
Per API

5.1.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Modify API behavior	On quote
Other changes	Estimation in tokens based on time spent

5.2 Application Gateway

5.2.1 Description

Azure Application Gateway is a web traffic load balancer that enables you to manage traffic to your web applications..

5.2.2 Build to run service included in the OTC

5.2.2.1 Build service pre-requisite

- Refer to generic description.

5.2.2.2 Build to run service

- Refer to generic description.

5.2.3 RUN services included in the MRC

5.2.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the CDN.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.2.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Metrics supported by Application Gateway V2 SKU:

- Backend connect time
- Backend first byte response time
- Backend last byte response time
- Application gateway total time
- Client RTT
- Bytes received
- Bytes sent
- Client TLS protocol
- Current capacity units
- Current compute units
- Current connections
- Estimated Billed Capacity units
- Failed Requests
- Fixed Billable Capacity Units
- New connections per second
- Response Status
- Throughput
- Total Requests
- Backend response status
- Healthy host count
- Unhealthy host count
- Requests per minute per Healthy Host

Metrics supported by Application Gateway V1 SKU

- CPU Utilization

- Current connections
- Failed Requests
- Response Status
- Throughput
- Total Requests
- Healthy host count
- Unhealthy host count

Alerts observed

- Backend connect time (V2)
- Backend response status (V2)
- Application Gateway Total Time (V2)
- Throughput (V1, V2)
- Client RTT (V2)
- Failed Requests (V2)
- Custom: %age of failed request (Failed Requests / Total Requests) (V2)
- Unhealthy Host Count (V2)
- CPU Utilization (V1)
- Failed Requests (V1)
- Response Status (V1)

5.2.3.3 Backup and restore

Data backup and restore

Can be exported from CI/CD Pipeline.

Service restore

The Continuous Deployment chain is used to redeploy the Application Gateway from the configuration file of reference for production environment committed in the Git.

5.2.3.4 Azure SLA High Availability and Disaster Recovery inter-region

For Application Gateway V2, the service HA is managed by Microsoft.
The DR can be customized by design.

5.2.4 Charging model

Work Unit
Per Instance

5.2.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Add/modify Backend	1 Token
Certificate Installation	1 Token
Other changes	Estimation in tokens based on time spent

5.3 Application Insights – basic monitoring with class 2 transition

5.3.1 Description

Application Insights, a feature of Azure Monitor, is an extensible Application Performance Management (APM) service for developers and DevOps professionals. Use it to monitor your live applications. It will automatically detect performance anomalies, and includes powerful analytics tools to help you diagnose issues and to understand what users actually do with your app. It's designed to help you continuously improve performance and usability.

The basic monitoring excludes the middleware and application management as well as remedial actions.

5.3.2 Build to run service included in the OTC

5.3.2.1 Build to run service pre-requisite

The pre-requisite to Application Insights basic monitoring with class 2 transition is that Application Insights has been configured by the Customer including

- Resources monitored
- SDK deployed on the resources when applicable
- Metrics and alerts forwarded to Azure Monitor
- Performance dashboards

5.3.2.2 Build to run service

For Application Insight basic monitoring with class 2 transition, the build to run service included in the OTC consists in integrating the alerts from Azure Monitor configured in Application Insights into The Provider supervision backend.

5.3.3 RUN services included in the MRC

5.3.3.1 Run service pre-requisite

- The resource monitored is in the inventory Scope of Work of managed service : infrastructure resource, middleware resource, application resource, database resource, Kubernetes cluster resource, microservice resource, etc…
- A referential file exists in the Git including the reference configuration of Application Insights.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.3.3.2 KPI & alerts

Monitoring

Yes

Alerts observed

- Alerts defined in Application Insights and forwarded to Azure Monitor for resources in the Scope of Work of managed services.

5.3.3.3 Monitoring service

As part of the Application Insights basic monitoring service, The Provider operations will monitor the alerts, raise tickets and inform the Customer on incident. The basic service excludes remedial of incident.

5.3.3.4 Backup and restore

Backup and restore of Application Insights: N/A

Service restore of Application Insights: The configuration of Azure Application Insight can be recovered from Infrastructure-as-code if its configuration has been done through infrastructure as code.

Backup and restore of resources monitored by Application Insights: N/A

Restore from IaC for resources monitored by Application Insights: N/A

5.3.3.5 Limitations & pre-requisite

the Application Insights basic monitoring service is monitoring only.

5.3.4 Charging model

Work Unit
Per managed resource

5.3.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Other changes	Estimation in tokens based on time spent

5.4 Application service

5.4.1 Description

Azure App Service is an HTTP-based service for hosting web applications, REST APIs, and mobile back ends. You can develop in your favorite language, be it .NET, .NET Core, Java, Ruby, Node.js, PHP, or Python. Applications run and scale with ease on both Windows and Linux-based environments.

App Service not only adds the power of Microsoft Azure to your application, such as security, load balancing, autoscaling, and automated management. You can also take advantage of its DevOps capabilities, such as continuous deployment from Azure DevOps, GitHub, Docker Hub, and other sources, package management, staging environments, custom domain, and TLS/SSL certificates.

With App Service, you pay for the Azure compute resources you use. The compute resources you use are determined by the App Service plan that you run your apps on

5.4.2 Build to run service included in the OTC

5.4.2.1 Build service pre-requisite

- Refer to generic description.

5.4.2.2 Build to run service

- Refer to generic description.

5.4.3 RUN services included in the MRC

5.4.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.4.3.2 Co-manage option

To be defined

5.4.3.3 KPI & alerts

Monitoring

Yes

KPI monitored

Response Time
Average memory working set
CPU Time
Data In
Data Out
Health Check Status
Requests
Thread Count
Other azure metrics on demand

Alerts observed

Health Check Status
Others will be identified with the customer

5.4.3.4 Backup and restore

Data backup and restore

Provided by Azure Backup depending on customer's design and build.

Service restore

On-demand from Azure Backup.

5.4.3.5 Azure SLA High Availability

HA and non HA are provided by Azure depending on the design and service parameter configuration

5.4.3.6 Recovery for region failure

Optional with charge: based on regular snapshot and recovery from this snapshot.

5.4.4 Charging model

Work Unit
Per Web Application

5.4.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Deploy a new version of an existing webapp	1 token
Add a custom domain on an Azure App Service	2 tokens
Configure a connection string to access another resource	1 token

Add CORS functionality	Estimation in tokens based on time spent
Add a SSL certificate	1 token
Enable authentication for front-end application	1 token
Enable authorization for front-end application	Estimation in tokens based on time spent
Move an App Service in another region	1 token
Other changes	Estimation in tokens based on time spent

5.5 Azure DNS

5.5.1 Description

Azure DNS host your Domain Name System (DNS) domains in Azure.
Azure DNS Private Zones provides a simple, reliable, secure DNS service to manage and resolve names in a VNET without the need for you to create and manage custom DNS solution. This capability allows you to use your own domain names, rather than the Azure-provided names available today. It provides name registration in VNet and also resolution for VNets that does not need registration.
Additionally, you can configure zones names with a split-horizon view allowing a private and a public DNS zone to share the same name.

5.5.2 Build to run service included in the OTC

5.5.2.1 Build service pre-requisite

- Refer to generic description.

5.5.2.2 Build to run service

- Refer to generic description.

5.5.3 RUN services included in the MRC

5.5.3.1 Run service pre-requisite

- A referential file exists in the Git used by The Provider which includes the reference configuration of the DNS.
- This file can be executed with a CI/CD used by The Provider and the execution has been tested successfully.

5.5.3.2 Co-manage option

For the Public part, The Provider work with the customer for the publics domain naming context.
For the private Part, a RACI must be done.

5.5.3.3 KPI & alerts

Monitoring

Yes, On demand by Network watcher

KPI monitored

Number of changes in the DNS database.

Alerts observed

Number of changes in the DNS rules

5.5.3.4 Backup and restore

Data backup and restore

Yes. Backup is proposed based on regular export.

Service restore

The CI/CD chain is used to redeploy the records from a backup zone into the native DNS service or from an export

5.5.3.5 Azure SLA High Availability and Disaster Recovery inter-region

Microsoft global network of name servers has the scale and redundancy to give you ultra-high availability for your domains.

5.5.4 Charging model

Work Unit
Per resource group

5.5.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Create / update/ delete zone (one zone including reverse)	1 token
Create / update/ delete record (up to 10 records)	1 token
Zone delegation*	1 token
Configure Firewall DNS	2 tokens
Other changes	Estimation in tokens based on time spent

Zone Delegation*: Specification should be received as a prerequisite.

5.6 Azure Firewall

5.6.1 Description

Azure Firewall is a managed, cloud-based network security service that protects your Azure Virtual Network resources. It's a fully stateful firewall as a service with built-in high availability and unrestricted cloud scalability.

5.6.2 Build to run service included in the OTC

5.6.2.1 Build service pre-requisite

- Refer to generic description.

5.6.2.2 Build to run service

- Refer to generic description.

5.6.3 RUN services included in the MRC

5.6.3.1 Run service pre-requisite

- A referential file exists in the Git used by The Provider which includes the reference configuration of the service.

- This file can be executed with a CI/CD used by The Provider and the execution has been tested successfully.

5.6.3.2 Co-manage option

No, The Provider manages the Firewall

5.6.3.3 KPI & alerts

Monitoring

Yes

KPI monitored

- Application rules hit count
- Network rules hit count
- Data processed
- Throughput
- Firewall health state
- SNAT port utilization

Alerts observed

Default

Firewall health state

Optional

Application rules hit count
Network rules hit count
Data processed
Throughput
SNAT port utilization

5.6.3.4 Backup and restore

Data backup and restore

On demand export of rules in JSON format file

Service restore

The Continuous Deployment chain is used to redeploy the Firewall from the configuration file of reference for production environment committed in the Git.

5.6.3.5 Azure SLA High Availability and Disaster Recovery inter-region

Azure Firewall can be configured during deployment to span multiple Availability Zones for increased Availability depending on design Scope of Work.

5.6.3.6 Network and security managed services

Additional Network and Security Managed services might be added optionally depending on Scope of Work.

5.6.4 Charging model

Work Unit
Per pack of 30 rules in Azure Firewall

5.6.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Add / modify / delete rules or NAT (up to 5 rules)	1 token
Other changes	Estimation in tokens based on time spent

5.7 Azure Function

5.7.1 Description

Azure Function processes events with serverless code

5.7.2 Build to run service included in the OTC

5.7.2.1 Build service pre-requisite

- Refer to generic description.

5.7.2.2 Build to run service

- Refer to generic description.

5.7.3 RUN services included in the MRC

5.7.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference code of the Function.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.7.3.2 Co-manage option

TBD

5.7.3.3 KPI

KPIs:

- Response Time
- Average memory working set
- Function execution count
- Function execution unit
- Data In
- Data Out
- Health Check Status
- Requests
- Thread Count
- Other azure metrics on demand

5.7.3.4 Alerts

By default no, customized alerting can be added as an option based on customer needs.

5.7.3.5 Backup and restore

Data backup and restore

Backup is not used by default.

Service restore

By default, the Function source code in the GIT is the referential and the Continuous Deployment chain workflow is used to deploy it. Shall a problem occur on a Function, the Continuous Deployment chain is used to redeploy the Function from the version of reference in the GIT.

5.7.3.6 Azure SLA High Availability

HA and non HA are provided by Azure depending on the design and service parameter configuration as per design Scope Of Work.

5.7.3.7 Disaster Recovery inter-region

In the design Scope Of Work, customer can request HA inter-region to be configured to protect against region loss.

5.7.4 Charging model

Work Unit
Per package of 100 lines of Function code

5.7.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Activate / deactivate a function	1 token
Connect a function app to a Virtual Network	1 token
Configure a connection string to access another resource	1 token
Add a customer domain on Function App	2 tokens
Add CORS functionality	Estimation in tokens based on time spent
Add SSL certificate	1 token

5.8 Azure storage

5.8.1 Description

The description of Managed Services for storage exclude that Managed Services for Disks which is included in the Managed OS for Virtual Machines.

5.8.2 Build to run service included in the OTC

5.8.2.1 Build service pre-requisite

- Please refer to generic description

5.8.2.2 Build to run service

- Build to run service for Storage are necessary. They encompass the parameters setting for the storage e.g Tiering. Optionally, if an optional recurring managed service has been requested, build to run task will include the selection of Kpis to be observed and alerts to be set up based on KPI thresholds, or external calls to test the availability of the storage. Please refer to generic build to run description.

5.8.3 RUN services included in the MRC

Recurring run managed services for Azure Storage are optional. Depending on Customer's interest in monitoring the storage KPIs, in alerting based on KPIs, in backup / restore, the Customer may request the service. By default, there is no recurring task proposed on storage, but on demand changes and on demand investigations.

5.8.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the storage and of the metrics and alerts observed for the storage.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.8.3.2 Co-manage option

Yes

5.8.3.3 KPI & alerts

Monitoring

Optional: Metrics and health probes

Alerts observed

- API not reachable
- Transactions failure rates
- Size threshold

5.8.3.4 Backup and restore

Data backup and restore

Optional: storage can be highly available. Whether the customer wants a versioning of backup for storage, it is provided has part of a recurring proposal

Service restore

Optional: subject to customer having ordered backup and restore for storage.

5.8.3.5 Azure SLA High Availability and Disaster Recovery inter-region

Multiple available options are proposed by Azure depending on the class of service.

5.8.4 Charging model

Work Unit
Per 5 storage accounts

5.8.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Configure access policy for storage account	1 token
Changes	Estimation in tokens based on time spent

5.9 Content Delivery Network (CDN)

5.9.1 Description

Azure Content Delivery Network (CDN) is a global CDN solution for delivering high-bandwidth content. It can be hosted in Azure or any other location. With Azure CDN, you can cache static objects loaded from Azure Blob storage, a web application, or any publicly accessible web server, by using the closest point of presence (POP) server. Azure CDN can also accelerate dynamic content, which cannot be cached, by leveraging various network and routing optimizations.

5.9.2 Build to run service included in the OTC

5.9.2.1 Build service pre-requisite

- Refer to generic description.

5.9.2.2 Build to run service

- Refer to generic description.

5.9.3 RUN services included in the MRC

5.9.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the CDN.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.9.3.2 Co-manage option

Yes based on RACI determined during pre-sales or build.

5.9.3.3 KPI & alerts

Monitoring

Yes: Metrics and diagnostic logs

KPI monitored

- Byte Hit Ratio
- Request Count
- Response Size
- Total Latency
- Customized ping page per zone

Alerts observed

- Customized ping page per zone
- Latency

5.9.3.4 Backup and restore

Data backup and restore

Can be exported from CI/CD Pipeline.

Service restore

The Continuous Deployment chain is used to redeploy the CDN from the configuration file of reference for production environment committed in the Git.

5.9.3.5 Azure SLA High Availability and Disaster Recovery inter-region

Based on design SOW, the service can be built in multiple regions.

5.9.4 Charging model

Work Unit
Per Endpoint

5.9.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Purge CDN	1 Token
Add URL	1 Token
Other changes	Estimation in tokens based on time spent

5.10 Event Hubs

5.10.1 Description

Azure Event Hubs is a big data streaming platform and event ingestion service. It can receive and process millions of events per second. Data sent to an event hub can be transformed and stored by using any real-time analytics provider or batching/storage adapters.

The following scenarios are some of the scenarios where you can use Event Hubs:

- Anomaly detection (fraud/outliers)
- Application logging
- Analytics pipelines, such as clickstreams
- Live dashboarding
- Archiving data
- Transaction processing
- User telemetry processing
- Device telemetry streaming.

5.10.2 Build to run service included in the OTC

5.10.2.1 Build service pre-requisite

- Refer to generic description.

5.10.2.2 Build to run service

- Refer to generic description.

5.10.3 RUN services included in the MRC

5.10.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the Event Hubs.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.10.3.2 Co-manage option

No, The Provider manages the Load Balancer

5.10.3.3 KPI & alerts

Monitoring

Yes

KPI monitored

- Incoming Requests
- Successful Requests
- Throttled Requests

Alerts observed

- Throttled Requests

5.10.3.4 Backup and restore

Data backup and restore

Not applicable. Event Hubs does not store data persistently.
Datastore is excluded from the scope of work of the work unit. It is a separate work unit.
Note: as a chargeable separate service the datastore where the data has been injected can be backed up and restored.

Service restore

The Continuous Deployment chain is used to redeploy the Event Hubs from the configuration file of reference for production environment committed in the Git.
Restore of the datastore is a separate work Unit.

5.10.3.5 Azure SLA High Availability and Disaster Recovery inter-region

Azure ensures High Availability of the Event Hubs with standard SKU.

Maintaining a cross region Disaster Recovery requires specific design and subject to a specific additional charging.

5.10.4 Charging model

Work Unit
Per source type

5.10.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Add a new source into Event Hubs	On quote
Other changes	Estimation in tokens based on time spent

5.11 Express Route

5.11.1 Description

Express Route allow to extend on-premises networks into Microsoft Cloud (Azure, Microsoft 365) over a private connection. Express Route connections offer more reliability, faster speeds, consistent latencies and higher security than connections over Internet.

Managed service for express route only covers the Azure End point. It does not cover the distant end point nor the end-to-end link. Managing end to end networking can be proposed by The Provider additionally, based on Scope Of Work and RACI.

5.11.2 Build to run service included in the OTC

5.11.2.1 Build service pre-requisite

- Refer to generic description.

5.11.2.2 Build to run service

- Refer to generic description.

5.11.3 RUN services included in the MRC

5.11.3.1 Run service pre-requisite

- A referential file exists in the Git including a partial configuration of the connectivity.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.11.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

- Availability
- Bandwidth

Alerts observed

- Availability
- Bandwidth
- Custom status metric

5.11.3.3 Backup and restore

Data backup and restore

On demand. Backup is proposed based on export template.

Service restore

The Continuous Deployment chain is used to redeploy the initial configuration or from an export.

5.11.3.4 Azure SLA High Availability and Disaster Recovery inter-region

Azure ensures High Availability of the Express Route and can be maximize by design.

Cross region Disaster Recovery based on WAN Architecture requirements.

5.11.3.5 Network and security managed services

No by default.

5.11.4 Charging model

Work Unit
Per peering

5.11.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Modify bandwidth	1 token

Other changes	Estimation in tokens based on time spent

5.12 Front door

5.12.1 Description

Azure Front Door is a global, scalable entry-point that uses the Microsoft global edge network to create fast, secure, and widely scalable web applications. With Front Door, you can transform your global consumer and enterprise applications into robust, high-performing personalized modern applications with contents that reach a global audience through Azure.

5.12.2 Build to run service included in the OTC

5.12.2.1 Build service pre-requisite

- Refer to generic description.

5.12.2.2 Build to run service

- Refer to generic description.

5.12.3 RUN services included in the MRC

5.12.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the Azure Front Door.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.12.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Metrics supported for Front Door service:

- RequestCount
- RequestSize
- ResponseSize
- TotalLatency
- BackendRequestCount
- BackendRequestLatency
- BackendHealthPercentage
- WebApplicationFirewallRequestCount

Alerts observed

- BackendRequestLatency(CDN)
- BackendHealthPercentage(CDN)
- WebApplicationFirewallRequestCount(WAF)

5.12.3.3 Backup and restore

Data backup and restore: N/A

On-demand export template

Service restore

The Continuous Deployment chain is used to redeploy the Front Door from the configuration file of reference for production environment committed in the Git.

5.12.3.4 Azure SLA High Availability and Disaster Recovery inter-region

The service is in high-availability pattern by default in Azure.

5.12.4 Charging model

Work Unit
Per Instance

5.12.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Add / modify /delete a rule	1 Token
Add / modify a backend	2 Tokens
Add a new CDN configuration	3 Tokens
Other changes	Estimation in tokens based on time spent

5.13 Key vault

5.13.1 Description

Azure Key Vault is a cloud service for securely storing and accessing secrets. Key Vault has two service tiers: Standard, which encrypts with a software key, and a Premium tier, which includes HSM-protected keys.

5.13.2 Build to run service included in the OTC

5.13.2.1 Build service pre-requisite

- Refer to generic description.

5.13.2.2 Build to run service

- Refer to generic description.

5.13.3 RUN services included in the MRC

5.13.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the KeyVault.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.13.3.2 KPI & alerts

KPIs:

- Vault availability
- Vault saturation
- Service API Latency
- Total Service API Hits by Activity Type
- Total Service API Hits by HTTP Status Code

Alert	Description	Severity	Source
Overall Vault Availability	Alert if vault is unavailable (less than 100%)	P1	Metrics
Overall Vault Saturation	Alert if vault capacity is exceeded (greater than 75%)	P3	Metrics
Overall Service API Latency	Alerts if average latency is above 500 ms	P3	Metrics
Count Total Service API Hits By Status Code	Alert if the total of error code exceed the standard value for the customer context (dynamic value)	P1	Metrics
Vault Deleted	Alert if key vault is deleted	P1	Activity Log

5.13.3.3 Backup and restore

Data backup and restore

By default The Provider enables soft delete option on Azure KeyVault which preserves the data for 90 days.

Backup is an optional task based on scope of work as it requires either a secured storage or a secondary KeyVault as a target. By setting-up backup to a secondary KeyVault, one protects against disaster on the KeyVault, see below.

5.13.3.4 Azure SLA High Availability and Disaster Recovery inter-region

Supported by Microsoft. The Key Vault content of one region is automatically replicated in its paired region except in the case of the Brazil South region.

The rare times an entire Azure region is unavailable, the requests that you make of Azure Key Vault in that region are automatically routed (failed over) to a secondary region except in the case of the Brazil South region.

5.13.3.5 Security

Security recommendations can be part of an optional security scope of work based on customer request.
By default, the MRC does not cover security recommendations

5.13.4 Information on Azure service

5.13.4.1 SKU

- Standard: Software encrypted keys
- Premium: Hardware encrypted keys (HSM-protected keys)

5.13.4.2 Service Limits

https://docs.microsoft.com/en-us/azure/key-vault/general/service-limits

5.13.5 Charging model

Work Unit
Per Key Vault instance

5.13.6 Changes catalogue – in Tokens, per act

Changes examples	Effort
Add/remove key	1 token
Configure access policy	2 tokens
Add/Remove/Configure end of life for Certificate	1 token
Configure Azure native services to use key vault	Estimation in tokens based on time spent
Other changes	Estimation in tokens based on time spent

5.14 Load Balancer

5.14.1 Description

Azure Load Balancer operates at layer 4 of the Open Systems Interconnection (OSI) model. It's the single point of contact for clients. Load balancer distributes inbound flows that arrive at the load balancer's front end to backend pool instances. These flows are according to configured load-balancing rules and health probes. The backend pool instances can be Azure Virtual Machines or instances in a virtual machine scale set.

A public load balancer can provide outbound connections for virtual machines (VMs) inside your virtual network. These connections are accomplished by translating their private IP addresses to public IP addresses. Public Load Balancers are used to load balance internet traffic to your VMs.

An internal (or private) load balancer is used where private IPs are needed at the frontend only. Internal load balancers are used to load balance traffic inside a virtual network. A load balancer frontend can be accessed from an on-premises network in a hybrid scenario.

5.14.2 Build to run service included in the OTC

5.14.2.1 Build service pre-requisite

- Refer to generic description.

5.14.2.2 Build to run service

- Refer to generic description.

5.14.3 RUN services included in the MRC

5.14.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the load balancer.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.14.3.2 Co-manage option

No, The Provider manages the Load Balancer

5.14.3.3 KPI & alerts

Monitoring

Yes: insights, Metrics and health probes

KPI monitored

- Data path availability
- Health probe status
- SYN (synchronize) count
- SNAT connection count
- Allocated SNAT ports
- Used SNAT ports
- Used SNAT ports
- Bytecount
- Packet count

Alerts observed

- Data path availability
- Health probestatus

5.14.3.4 Backup and restore

Data backup and restore

Not applicable. Load balancer does not store data persistently.

Service restore

The Continuous Deployment chain is used to redeploy the Load Balancer from the configuration file of reference for production environment committed in the Git.

5.14.3.5 Azure SLA High Availability and Disaster Recovery inter-region

Azure ensures High Availability of the Load Balancer with standard SKU.

Maintaining a cross region Disaster Recovery requires specific design and subject to a specific additional charging.

5.14.4 Charging model

Work Unit
Per Load Balancer instance

5.14.5 Changes catalogue – in Tokens, per act

Changes examples	Effort	Impact on MRC
Setup / modify / delete URI	1 token	
Change health probes / Add new backend	2 tokens	
Other changes	Estimation in tokens based on time spent	

5.15 Log Analytics – basic monitoring with class 2 transition

5.15.1 Description

Log Analytics is a tool in the Azure portal to edit and run log queries from data collected by Azure Monitor Logs and interactively analyze their results. You can use Log Analytics queries to retrieve records that match particular criteria, identify trends, analyze patterns, and provide a variety of insights into your data.

The basic monitoring excludes the middleware and application management as well as remedial actions.

5.15.2 Build to run service included in the OTC

5.15.2.1 Build to run service pre-requisite

The pre-requisite to Log Analytics basic monitoring with class 2 transition is that Log Analytics has been configured by the Customer including

- Log collection for the resources
- Metrics and alerts forwarded to Azure Monitor

5.15.2.2 Build to run service

For Log Analytics basic monitoring with class 2 transition, the build to run service included in the OTC consists in integrating the alerts from Azure Monitor configured in Log Analytics into The Provider supervision backend.

5.15.3 RUN services included in the MRC

5.15.3.1 Run service pre-requisite

- The resource configured is in the inventory Scope of Work of managed service: infrastructure resource, middleware resource, application resource, database resource, Kubernetes cluster resource, microservice resource, etc…
- A referential file exists in the Git including the reference configuration of Log Analytics.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.15.3.2 KPI & alerts

Monitoring

Yes

Alerts observed

- Alerts defined in Log Analytics and forwarded to Azure Monitor for resources in the Scope of Work of managed services.

5.15.3.3 Monitoring service

As part of the Log Analytics basic monitoring service, The Provider operations will monitor the alerts, raise tickets and inform the Customer on incident. The basic service excludes troubleshooting or remedial of incident.

5.15.3.4 Backup and restore

Backup and restore of Log Analytics: N/A

Service restore of Log Analytics: The configuration of Azure Log Analytics can be recovered from Infrastructure-as-code if its configuration has been done through infrastructure as code.

Backup and restore of resources monitored by Log Analytics: N/A

Restore from IaC for resources monitored by Log Analytics: N/A

5.15.3.5 Limitations & pre-requisite

The Log Analytics basic monitoring service is monitoring only.

5.15.4 Charging model

Work Unit
Per managed resource

5.15.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Filter and send logs for a specific resource	Estimation in tokens based on time spent
Other changes	Estimation in tokens based on time spent

5.16 Logic App

5.16.1 Description

Automate the access and use of data across clouds without writing code

5.16.2 Build to run service included in the OTC

5.16.2.1 Build service pre-requisite

- Refer to generic description.

5.16.2.2 Build to run service

- Refer to generic description.

5.16.3 RUN services included in the MRC

5.16.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.16.3.2 Co-manage option

Yes

5.16.3.3 KPI

- ActionLatency
- ActionsFailed
- ActionThrottledEvents
- RunFailurePercentage
- RunLatency
- RunsCancelled

- RunsCompleted
- RunsFailed
- RunsStarted
- RunsSucceeded

5.16.3.4 Alerts

Optional to be discussed with customer based on case by case.

5.16.3.5 Backup and restore

Data backup and restore

Not in place by default.

Service restore

The Continuous Deployment chain is used to redeploy the Logic App from the configuration file of reference for production environment committed in the Git.

5.16.3.6 Azure SLA High Availability

Depends on design Scope Of Work.

5.16.4 Charging model

Work Unit
Per Application

5.16.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Changes on demand	Estimation in tokens based on time spent

5.17 Network Security Group - Network and Security management services

5.17.1 Description

Azure network security group used to filter network traffic to\from Azure resources in an Azure virtual network. It contains security rules that allow or deny inbound and outbound network traffic.

At the basic level, managing Network Security group consists in building, deploying and maintaining the IaC for it and managing the changes.

The management of Network Security Groups is included as part of a larger bundle of Network and Security Managed services which provides network and security design, maintain, network watching, intrusion detection, troubleshooting depending on an agreed Scope of Work.

5.17.2 Charging model

Work Unit	OTC & MRC

Network and security management services	Custom, depending on agreed Scope of Work

5.17.3 Changes catalogue – in Tokens, per act

Changes examples	Effort
Add / modify / delete Security rules (up to 5 rules) excluding dependencies*	1 token
Add / modify / delete Security group (up to 5 rules) excluding dependencies*	1 token
Other changes	Estimation in tokens based on time spent

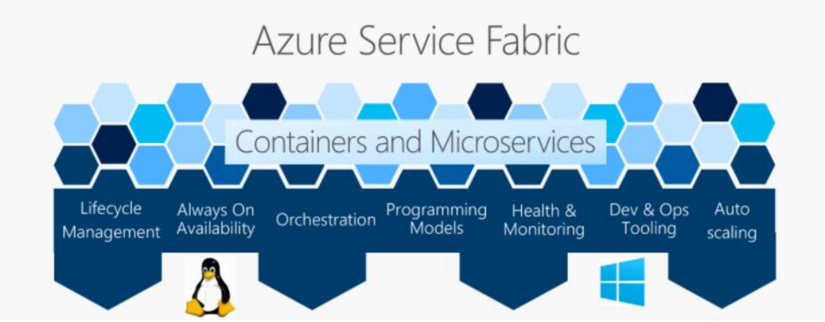
*Dependencies include all triggered applications like Azure Sentinel, Log Analytics, Azure Firewall, Logic App Security, Azure DB services and other native services.

5.18 Service Fabric

5.18.1 Description

Azure Service Fabric is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices and Azure Service Fabric is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices and containers. Service Fabric also addresses the significant challenges in developing and managing cloud native applications.

A key differentiator of Service Fabric is its strong focus on building stateful services. You can use the Service Fabric programming model or run containerized stateful services written in any language or code. You can create Service Fabric clusters anywhere, including Windows Server and Linux on premises and other public clouds, in addition to Azure.



Service Fabric powers many Microsoft services today, including Azure SQL Database, Azure Cosmos DB, Cortana, Microsoft Power BI, Microsoft Intune, Azure Event Hubs, Azure IoT Hub, Dynamics 365, Skype for Business, and many core Azure services.

5.18.2 Build to run service included in the OTC

5.18.2.1 Build service pre-requisite

- Refer to generic description.

5.18.2.2 Build to run service

- Refer to generic description.

5.18.3 RUN services included in the MRC

5.18.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the Service Fabric.

- This file can be executed with a CI/CD and the execution has been tested successfully.

5.18.3.2 KPI & alerts

Monitoring

Yes – cluster level. KPI and Alerts for the micro-services layer is handled as part of the additional service called Managed Container.

KPI monitored

Metrics supported for Service Fabric:

- PrimaryCount
- ReplicaCount

Alerts observed

- Idem

5.18.3.3 Backup and restore for the Site Recovery configuration

Service backup and restore: The native Azure backup for Service Fabric is used.

5.18.3.4 Azure SLA High Availability and Disaster Recovery inter-region

A service fabric multi-node cluster delivers high-availability by design.
Deployed on multi-region, a multi-region availability can be achieved.

5.18.3.5 Limitations & pre-requisite

Managing the microservice layer is an additional managed service called Managed Container charged per microservices. Please refer to Managed Application main service description document.

5.18.4 Charging model

Work Unit
Per cluster Node

5.18.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Scale application / services in a cluster	1 Token
Deploy containers to Service Fabric cluster	1 Token
Delete a cluster	1 Token
Upgrade the runtime of a Service Fabric Cluster	1 Token
Create a Service Fabric Cluster	On quote
Deploy an Azure Service Fabric cluster across Availability Zones	On quote
Other changes	Estimation in tokens based on time spent

5.19 Site Recovery

5.19.1 Description

Azure Site Recovery is delivering built-in disaster recovery service for Virtual Machines.

5.19.2 Build to run service included in the OTC

5.19.2.1 Build service pre-requisite

- Refer to generic description.

5.19.2.2 Build to run service

- Refer to generic description.

5.19.3 RUN services included in the MRC

5.19.3.1 Run service pre-requisite

- List of Virtual Machines should be provided.
- Compatibility of workload Architecture with Site Recovery protection mechanism.

5.19.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Metrics supported for Site Recovery service:

- Azure Site recovery replication stats
- Azure Site recovery replication dataupload rate
- Azure Site recovery replication recovery points
- Azure Site recovery replication replicated items

Alerts observed

- Azure Site recovery replication stats
- Azure Site recovery replication recovery points

5.19.3.3 Backup and restore for the Site Recovery configuration

Data backup and restore: N/A

Service restore: Restore will be recreating the Site Recovery Plan using the same configurations.

5.19.3.4 Testing the Site Recovery configuration (Failover/Failback Simulation)

Testing the site recovery can be handled as a complex change request. The time spent will be estimated in a number of Tokens.

5.19.3.5 Recovery of Virtual Machines with Site Recovery

The Virtual Machines protected by the Site Recovery can be recovered thanks to Azure mechanism. Implementing the recovery of Virtual Machines can be handled as a complex change request. The time spent will be estimated in a number of Tokens

5.19.3.6 Azure SLA High Availability and Disaster Recovery inter-region

The service purpose is to implement Disaster Recovery.

5.19.3.7 Limitations

Azure monitor is only used for replication monitoring within same region which might cause some limitations. A discussion is necessary for each customer case by case to discuss the monitoring.

5.19.4 Charging model

Work Unit
Per Virtual Machine protected

5.19.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Adding a VM to an already defined Site Recovery	2 Tokens
Testing the failover& failback Site Recovery	Estimation in tokens based on time spent
Failover for a site	Estimation in tokens based on time spent
Other changes	Estimation in tokens based on time spent

5.20 Traffic Manager

5.20.1 Description

Azure Traffic Manager is a DNS-based traffic load balancer. This service allows you to distribute traffic to your public facing applications across the global Azure regions. Traffic Manager also provides your public endpoints with high availability and quick responsiveness.

5.20.2 Build to run service included in the OTC

5.20.2.1 Build service pre-requisite

- Refer to generic description.

5.20.2.2 Build to run service

- Refer to generic description.

5.20.3 RUN services included in the MRC

5.20.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the CDN.

- This file can be executed with a CI/CD and the execution has been tested successfully.

5.20.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

- Endpoint status by endpoint
- Queries by endpoint returned

Alerts observed

- Endpoint status by endpoint

5.20.3.3 Backup and restore

Data backup and restore

Can be exported from CI/CD Pipeline.

Service restore

The Continuous Deployment chain is used to redeploy the Traffic Manager from the configuration file of reference for production environment committed in the Git.

5.20.3.4 Azure SLA High Availability and Disaster Recovery inter-region

The service is globally managed by Microsoft

5.20.4 Charging model

Work Unit
Per Profile

5.20.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Add/modify Backend	1 Token
Other changes	Estimation in tokens based on time spent

5.21 Virtual Machines and OS

5.21.1 Description

The Managed Service for Virtual Machines is called Managed OS. The Provider manages both the OS and the Virtual Machine.

5.21.2 Build to run service included in the OTC

5.21.2.1 Build service pre-requisite

- Refer to generic description.

5.21.2.2 Build to run service

- Refer to generic description.

5.21.3 RUN services included in the MRC

5.21.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the Virtual Machines.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.21.3.2 KPI & alerts

Monitoring is performed through configuration and activation of Azure Monitoring.

The Provider backend supervision system is collecting alerts from Azure Monitoring.

Alerts for VM with Linux OS:

Alert
Filesystem Usage
Filesystem Inode Usage
Agent Status
Load Usage
Memory Usage
Network Status
Zombie Detected
Ping

Alerts for VM with Windows OS:

Alert
Ping
Agent_Status
CPU
Disks
Pagefile
Physical Memory
Windows Services

5.21.3.3 OS patching

Azure Update Manager

For managed OS, The Provider leverages Azure Update Manager for the patching of the Operating System (OS). It presents the advantage to provide real-time status of patching level, to be consistent with Azure security and visible through Azure Advisor.

Behavior: With Azure Update Manager, patches are decided by Microsoft and all patches are to be applied if mandatory for the Virtual Machine for Windows and Linux.

The Provider Managed Application patching system

As an alternative when patches shall be chosen, The Provider leverages its own central patching system whereby all patches have been validated and tested by The Provider image factory. The Provider patching system allows for central reporting to The Provider operations teams of the proper patching level of each VM managed.

VPN connectivity to The Provider CASA zone is a pre-requisite.

The Provider shall not take responsibility of managed OS and its risks avoidance (security, defect) based on a Customer specific patching system.

5.21.3.4 Antivirus

For managed OS, The Provider leverages its central anti-virus system based on Sophos. This requires the installation of the anti-virus agent on the VM OS for each VM as well as the VPN connectivity to The Provider CASA zone. The Provider systems allows for central reporting on Malware from its backend console system.

Would the Customer desire to keep its own Antivirus system, then The Provider shall not be taken responsible for protection against viruses.

5.21.3.5 Backup and restore

Data backup and restore

By default, The Provider leverages Azure Backup on the Virtual Machines for Managed OS. The configuration of Azure Backup pattern and well as retention period shall be agreed with the Customer prior to the RUN. As example: 1 x backup per week, 1x incremental backup per day per VM. The retention period depends on customer request.

Restore of VM are performed from the backup.

- In case of incident, latest version of backup can be restored
- Upon change request, a previous version of backup can be restored.

5.21.3.6 Azure SLA High Availability and Disaster Recovery inter-region

By default, a Virtual Machine is not highly available.

The Customer shall leverage Azure VM Availability Set to expect High availability for the Availability Set of VMs (design requirement)

The Customer shall leverage Azure Site Recovery to allow protection from disaster (optional).

5.21.3.7 Administration tasks tracing

Actions performed by The Provider managed teams on the managed OS are done from The Provider CASA zone through an access controlled by a CyberArk bastion. The Provider CyberArk bastion protects the access and keep trace of the actions performed by the maintenance team allowing for audit.

The VPN connectivity to the CASA zone necessary for the management.

5.21.3.8 Login on to the Virtual Machine

For Windows OS based VM, access shall be granted by the Customer to The Provider managed application operations staff through a domain account configured with proper privilege groups.

For Linux OS based VM, an encrypted key is created and provided to The Provider managed application operations staff to log onto the VM. The key itself is stored in a safe i.e Azure KeyVault.

For Applications, in case of managed application: a secret stored in a safe.

5.21.3.9 Logs

Log management is not included in the managed OS / managed virtual machine service.
Optionally it can be activated through Azure Log Analytics through Change Request process.

5.21.3.10 Security

By default, the MRC includes the use of security policies and groups as per customer's configuration request.
The MRC does not cover security recommendations. Security recommendations can be part of an optional security scope of work based on customer request.

5.21.3.11 Limitations

Managed Application services is provided only for OS versions supported by the CSP vendor.

5.21.4 Charging model

Work Unit
Per Virtual Machine instance

5.21.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Create Application Security Groups	4 Tokens
Modify/delete Application Security Groups	2 Tokens
Create Virtual Machines, Availability Sets, VMSS, disk and image	8 Tokens
Modify Virtual Machines, Availability Sets	4 Tokens
Modify VMSS, disk and image [4tk]	4 Tokens
Delete Virtual Machines, Availability Sets	4 Tokens
Delete VMSS, disk and image	
Start/Stop/Restart Virtual Machines	2 Tokens
Create/modify/delete Storage Accounts	2 Tokens

5.22 VPN Gateway

5.22.1 Description

A VPN gateway is a specific type of virtual network gateway that is used to send encrypted traffic between an Azure virtual network and an on-premises location over the public Internet or Microsoft backbone network.

Additional managed services ca n be added optionally based on Scope of Work, refer to network and security services.

5.22.2 Build to run service included in the OTC

5.22.2.1 Build service pre-requisite

- Refer to generic description.

5.22.2.2 Build to run service

- Refer to generic description.

5.22.3 RUN services included in the MRC

5.22.3.1 Run service pre-requisite

- A referential file exists in the Git used by The Provider which includes the reference configuration of the VPN Gateway.
- This file can be executed with a CI/CD used by The Provider and the execution has been tested successfully.

5.22.3.2 Co-manage option

No, The Provider manages the VPN Gateway

5.22.3.3 KPI & alerts

Monitoring

This service can be monitored by Azure Monitor using Alerts and Metrics

KPI monitored

- AverageBandwidth
- P2SBandwidth
- P2SConnectionCount
- TunnelAverageBandwidth
- TunnelEgressBytes
- TunnelEgressPackets
- TunnelEgressPacketDropTSMismatch
- TunnelIngressBytes
- TunnelIngressPackets
- TunnelIngressPacketDropTSMismatch

Alerts observed

- AverageBandwidth
- P2SBandwidth
- P2SConnectionCount
- TunnelAverageBandwidth
- TunnelEgressBytes
- TunnelEgressPackets
- TunnelEgressPacketDropTSMismatch
- TunnelIngressBytes

- TunnelIngressPackets
- TunnelIngressPacketDropTSMismatch

5.22.3.4 Backup and restore

Data backup and restore

The Backup is N/A for VPN Gateway, but the deployment template can be exported on-demand before any configuration change.

Service restore

The Continuous Deployment chain is used to redeploy the VPN Gateway from the configuration file of reference for production environment committed in the Git.

5.22.3.5 Azure SLA High Availability and Disaster Recovery inter-region

Azure proposed availability for the VPN Gateway:

- 99.9% availability for each Basic Gateway for VPN or Basic Gateway for ExpressRoute.
- 99.95% availability for all Gateway for VPN SKUs excluding Basic.
- 99.95% availability for all Gateway for ExpressRoute

Availability is ensured by Azure and depends on design.

5.22.3.6 Network and security managed services

Additional Network and Security Managed services might be added optionally depending on Scope of Work.

5.22.4 Charging model

Work Unit
Per VPN Gateway

5.22.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Route modification	1 token
Create IPSec site-to-site Tunnel	2 tokens
Configure Network Gateway	Estimation in tokens based on time spent
Other changes	Estimation in tokens based on time spent

5.23 Web Application Firewall

5.23.1 Description

Azure Web Application Firewall (WAF) provides centralized protection of your web applications from common exploits and vulnerabilities. Web applications are increasingly targeted by malicious attacks that exploit commonly known vulnerabilities. SQL injection and cross-site scripting are among the most common attacks.

5.23.2 Build to run service included in the OTC

5.23.2.1 Build service pre-requisite

- Refer to generic description.

5.23.2.2 Build to run service

- Refer to generic description.

5.23.3 RUN services included in the MRC

5.23.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.23.3.2 Co-manage option

No, The Provider manages the WAF
OR, can be done with RACI determined during pre-sales or project build.

5.23.3.3 KPI & alerts

Monitoring

Yes

KPI monitored

- Data Processed
- Firewall Health
- SNAT Port Utilization
- Application Rule Hit
- Network Rule Hit

Alerts observed

- Firewall Health

5.23.3.4 Backup and restore

Data backup and restore

By default, N/A.

Service restore

The Continuous Deployment chain is used to redeploy the rules from the configuration file of reference for production environment committed in the Git.

5.23.3.5 Azure SLA High Availability and Disaster Recovery inter-region

Based on design Scope of Work, to be confirmed during presales phase.

5.23.3.6 Network and security managed services

Additional Network and Security Managed services might be added optionally depending on Scope of Work.

5.23.4 Charging model

Work Unit
Per IP of protected asset

5.23.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Add already existing rule	1 token
modify/delete rule/rules (up to 5)	1 token
Create a simple rule	1 token
Other changes	Estimation in tokens based on time spent

5.24 Azure Database for MySQL

5.24.1 Description

Azure Database for MySQL is a relational database service powered by the MySQL community edition. You can use either Single Server or Flexible Server to host a MySQL database in Azure. It's a fully managed database as a service offering that can handle mission-critical workloads with predictable performance and dynamic scalability.

5.24.2 Build to run service included in the OTC

5.24.2.1 Build service pre-requisite

- Refer to generic description.

5.24.2.2 Build to run service

- Refer to generic description.

5.24.3 RUN services included in the MRC

5.24.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.24.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Azure Monitor supported metrics for Azure Database for MySQL are available at:
Azure Monitor supported metrics by resource type - Azure Monitor | Microsoft Docs

Alerts observed

- active_connections
- aborted_connections
- total_connections
- cpu_credits_consumed
- cpu_credits_remaining
- io_consumption_percent
- cpu_percent
- memory_percent
- storage_percent
- network_bytes_egress
- replication_lag

5.24.3.3 Backup and restore

- **Data backup and restore**

Azure Database for MySQL servers are backed up periodically to enable Restore features. Using this feature, you may restore the server and all its databases to an earlier point-in-time, on a new server. The backup retention period governs how far back in time a point-in-time restore can be retrieved, since it's based on backups available. It could be set between 7 and 35 days.

- **Service restore**

The Continuous Deployment chain is used to redeploy the rules from the configuration file of reference for production environment committed in the Git.

5.24.3.4 Azure SLA High Availability and Disaster Recovery inter-region

Azure Database for MySQL provides fast restart capability of database servers, redundant storage, and efficient routing from the Gateway. For additional data protection, you can configure backups to be geo-replicated, and also deploy one or more read replicas in other regions. The estimation will be based on design Scope of Work, to be confirmed during presales phase.

5.24.3.5 Minor Version patching

Azure Database for MySQL automatically patches servers with minor releases (within maintenance window).

5.24.3.6 Major Version patching

Automatic in-place upgrades for major versions from 5.6 to 5.7 is supported.

Automatic in-place upgrades for major versions from 5.7 to 8.0 is not supported.
It could be done using either one of the following:

- Use mysqldump to move a database to a server created with the new engine version.
- Use Azure Database Migration service for doing online upgrades.

The estimation will be based on the database size.

5.24.4 Charging model

Work Unit
Per Database Instance

5.24.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Provision database	2 tokens
Restart server	1 token
Azure Database for MySQL failover	1 token
Stop & start database	1 token
Delete database	1 token
Move an Azure Database for MySQL Flexible server from one Azure region to another using the Azure portal	2 tokens
Create a clone	Estimation in tokens based on the database size
Restore a server to point-in-time and into a new copy of the server	Estimation in tokens based on the database size
Modify the service parameters configuration	1 token
Major version upgrade in Azure Database for MySQL	Estimation in tokens based on time spent
Other changes	Estimation in tokens based on time spent

5.25 Azure Database for PostgreSQL

5.25.1 Description

Azure Database for PostgreSQL is a relational database service based on the open-source Postgres database engine. It's a fully managed database-as-a-service that can handle mission-critical workloads with predictable performance, security, high availability, and dynamic scalability.

5.25.2 Build to run service included in the OTC

5.25.2.1 Build service pre-requisite

- Refer to generic description.

5.25.2.2 Build to run service

- Refer to generic description.

5.25.3 RUN services included in the MRC

5.25.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.25.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Azure Monitor supported metrics for Azure Database for PostgreSQL are available at:
Azure Monitor supported metrics by resource type - Azure Monitor | Microsoft Docs

Alerts observed

- active_connections
- connections_failed
- connections_succeeded
- cpu_percent
- memory_percent
- storage_percent
- read_throughput
- write_throughput

5.25.3.3 Backup and restore

Data backup and restore

Azure Database for PostgreSQL takes backups of the data files and the transaction log. Depending on the supported maximum storage size, we either take full and differential backups (4-TB max storage servers) or snapshot backups (up to 16-TB max storage servers). These backups allow you to restore a server to any point-in-time within your configured backup retention period. The default backup retention period is seven days. You can optionally configure it up to 35 days. All backups are encrypted using AES 256-bit encryption.

Azure Database for PostgreSQL provides the flexibility to choose between locally redundant or geo-redundant backup storage in the General Purpose and Memory Optimized tiers.

Service restore

Recovery will be from Infra as Code.

5.25.3.4 Azure SLA High Availability and Disaster Recovery inter-region

Built on Azure architecture, the service has inherent high availability, redundancy, and resiliency capabilities to mitigate database downtime from planned and unplanned outages, without requiring you to configure any additional components.

5.25.3.5 Minor Version patching

Azure Database for PostgreSQL automatically patches servers with minor releases (within maintenance window).

5.25.3.6 Major Version patching

Automatic in-place upgrades for major versions are not supported. It could be done using either one of the following:

- Use pg_dump and pg_restore to move a database to a server created with the new engine version.
- Use Azure Database Migration service for doing online upgrades.

5.25.4 Charging model

Work Unit
Per Database Instance

5.25.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Provision database	2 Tokens
Restart instance	1 Token
Delete instance	1 Token
Modify compute/storage	2 Tokens
Modify High availability	1 Token
Modify Server parameters	1 Token
Restore point-in-time to a new server	Estimation in tokens based on database size
Modify the server parameters	1 Token
Other changes	Estimation in tokens based on time spent

5.26 Azure SQL Database

5.26.1 Description

Azure SQL Database is a fully managed platform as a service (PaaS) database engine that handles most of the database management functions such as upgrading, patching, backups, and monitoring without user involvement.

5.26.2 Build to run service included in the OTC

5.26.2.1 Build service pre-requisite

- Refer to generic description.

5.26.2.2 Build to run service

- Refer to generic description.

5.26.3 RUN services included in the MRC

5.26.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.26.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Azure Monitor supported metrics for Azure SQL Database are available at:
Azure Monitor supported metrics by resource type - Azure Monitor | Microsoft Docs

Alerts observed

- Metric alert
- Log alert
- Activity log alert

5.26.3.3 Backup and restore

Data backup and restore

Azure SQL Database creates:

- Full backups every week.
- Differential backups every 12 or 24 hours.
- Transaction log backups approximately every 10 minutes.

The exact frequency of transaction log backups is based on the compute size and the amount of database activity. When you restore a database, the service determines which full, differential, and transaction log backups need to be restored.

By default, Azure SQL Database stores data in geo-redundant storage The Provider that are replicated to a paired region. Geo-redundancy helps protect against outages that affect backup storage in the primary region. It also allows you to restore your databases in a different region in the event of a regional outage.

This table summarizes the capabilities and features of point-in-time restore (PITR), geo-restore, and long-term retention.

Backup property	PITR	Geo-restore	LTR
Types of SQL backup	Full, differential, log.	Replicated copies of PITR backups.	Only the
Recovery point objective (RPO)	10 minutes, based on compute size and amount of database activity.	Up to 1 hour, based on geo-replication. *	One we policy).
Recovery time objective (RTO)	Restore usually takes less than 12 hours but could take longer, depending on size and activity.	Restore usually takes less than 12 hours but could take longer, depending on size and activity.	Restore than 12 take lon size anc
Retention	7 days by default, configurable up to 35 days.	Enabled by default, same as source. **	Not ena Retentic years.
Azure Storage	Geo-redundant by default. You can optionally configure zone-redundant or locally redundant storage.	Available when PITR backup storage redundancy is set to geo-redundant. Not available when PITR backup storage is zone-redundant or locally redundant.	Geo-re default. zone-re redunda

* For business-critical applications that require large databases and must ensure business continuity, use auto-failover groups.

** All PITR backups are stored on geo-redundant storage by default, so geo-restore is enabled by default.

Service restore

Recovery will be from Infra as Code.

5.26.3.4 Azure SLA High Availability and Disaster Recovery inter-region

Azure SQL Database and Azure SQL Managed Instance feature a built-in high availability solution, that is deeply integrated with the Azure platform. It is dependent on Service Fabric for failure detection and recovery, on Azure Blob storage for data protection, and on Availability Zones for higher fault tolerance (as mentioned earlier in document not applicable to Azure SQL Managed Instance yet). In addition, SQL Database and SQL Managed Instance use the Always On availability group technology from the SQL Server instance for replication and failover. The combination of these technologies enables applications to fully realize the benefits of a mixed storage model and support the most demanding SLAs.

5.26.4 Charging model

Work Unit
Per Database Instance

5.26.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
provision database	2 tokens
delete database	2 tokens
Restore a server to point-in-time	Estimation in tokens based on the database size
Modify the service parameters configuration	1 token
Other changes	Estimation in tokens based on time spent
Changes examples	Effort
Other changes	Estimation in tokens based on time spent

5.27 Azure Cosmos DB

5.27.1 Description

Azure Cosmos DB is a fully managed NoSQL database. Cosmos DB handles most of the database management functions with automatic management, updates and patching. It also handles capacity management with cost-effective serverless and automatic scaling options that respond to application needs to match capacity with demand.

5.27.2 Build to run service included in the OTC

5.27.2.1 Build service pre-requisite

- Refer to generic description.

5.27.2.2 Build to run service

- Refer to generic description.

5.27.3 RUN services included in the MRC

5.27.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.27.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Azure Monitor supported metrics for Azure Cosmos DB are available at:
Azure Monitor supported metrics by resource type - Azure Monitor | Microsoft Docs

Alerts observed

- ServiceAvailability
- TotalRequests
- DataUsage
- IndexUsage
- ReplicationLatency
- ServerSideLatency
- CosmosDbRequests

5.27.3.3 Backup and restore

Data backup and restore

There are two backup modes:

- **Continuous backup mode** – This mode has two tiers. One tier includes 7-day retention and the second includes 30-day retention. Continuous backup allows you to restore to any point of time within either 7 or 30 days.

- **Periodic backup mode** - This mode is the default backup mode for all existing accounts. In this mode, you configure a backup interval and retention for your account. The maximum retention period extends to a month. The minimum backup interval can be one hour.

Data restore will be done from backup.

Service restore

Recovery will be from Infra as Code.

5.27.4 Charging model

Work Unit
Per Database Instance

5.27.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
provision database	2 tokens
delete database	2 tokens
Restore a server to point-in-time	Estimation in tokens based on the database size
Modify the service parameters configuration	1 token
Other changes	Estimation in tokens based on time spent

5.28 Azure Database for MariaDB

5.28.1 Description

Azure Database for MariaDB is a managed service you can use to run, manage, and scale highly available MySQL databases in the cloud.

5.28.2 Build to run service included in the OTC

5.28.2.1 Build service pre-requisite

- Refer to generic description.

5.28.2.2 Build to run service

- Refer to generic description.

5.28.3 RUN services included in the MRC

5.28.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.28.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Azure Monitor supported metrics for Azure Database for Maria DB are available at:
Azure Monitor supported metrics by resource type - Azure Monitor | Microsoft Docs

Alerts observed

- active_connections
- connections_failed
- cpu_percent
- memory_percent
- storage_percent
- serverlog_storage_percent
- io_consumption_percent
- seconds_behind_master
- network_bytes_egress

5.28.3.3 Backup and restore

Data backup and restore

Azure Database for MariaDB takes backups of the data files and the transaction log. These backups allow you to restore a server to any point-in-time within your configured backup retention period. The default backup retention period is seven days. You can optionally configure it up to 35 days. All backups are encrypted using AES 256-bit encryption.

These backup files aren't user-exposed and can't be exported. These backups can only be used for restore operations in Azure Database for MariaDB.

Long-term retention of backups beyond 35 days is currently not natively supported by the service yet.

Azure Database for MariaDB provides the flexibility to choose between locally redundant or geo-redundant backup storage in the General Purpose and Memory Optimized tiers.

Data restore

In Azure Database for MariaDB, performing a restore creates a new server from the original server's backups and restores all databases contained in the server.

There are two types of restore available:

- **Point-in-time restore** is available with either backup redundancy option and creates a new server in the same region as your original server utilizing the combination of full and transaction log backups.
- **Geo-restore** is available only if you configured your server for geo-redundant storage and it allows you to restore your server to a different region utilizing the most recent backup taken.

Service restore

Recovery will be from Infra as Code.

5.28.3.4 Azure SLA High Availability and Disaster Recovery inter-region

Azure Database for MariaDB provides built-in high availability.

5.28.4 Charging model

Work Unit
Per Database Instance

5.28.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Provision a database	2 Token
Restart instance	1 Token
Delete instance	1 Token
Restore point-in-time to a new server	Estimation in tokens based on database size
Modify the server parameters	1 Token

Other changes	Estimation in tokens based on time spent

5.29 Azure Managed Instance for Apache Cassandra

5.29.1 Description

Azure Managed Instance for Apache Cassandra provides automated deployment and scaling operations for managed open-source Apache Cassandra datacenters.

5.29.2 Build to run service included in the OTC

5.29.2.1 Build service pre-requisite

- Refer to generic description.

5.29.2.2 Build to run service

- Refer to generic description.

5.29.3 RUN services included in the MRC

5.29.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.29.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Azure Monitor supported metrics for Apache Cassandra are available at:
Azure Monitor supported metrics by resource type - Azure Monitor | Microsoft Docs

Alerts observed

- cassandra_cache_hit_rate
- cassandra_cache_size
- cassandra_table_row_cache_hit
- cassandra_client_request_failures
- cassandra_client_request_timeouts
- cassandra_client_request_contention_histogram
- cassandra_table_bloom_filter_false_ratio
- cassandra_table_bloom_filter_false_positives
- cassandra_table_bloom_filter_disk_space_used
- cassandra_table_write_latency
- cassandra_thread_pools_currently_blocked_tasks

5.29.3.3 Backup and restore

Data backup and restore

Snapshot backups are enabled by default and taken every 4 hours with Medusa. Backups are stored in an internal Azure Blob Storage account and are retained for up to 2 days (48 hours).

Data restore

Backups can be restored to the same VNet/subnet as your existing cluster, but they cannot be restored to the same cluster. Backups can only be restored to new clusters. Backups are intended for accidental deletion scenarios and are not geo-redundant. They are therefore not recommended for use as a disaster recovery (DR) strategy in case of a total regional outage. To safeguard against region-wide outages, we recommend a multi-region deployment.

Service restore

Recovery will be from Infra as Code.

5.29.3.4 Azure SLA High Availability and Disaster Recovery inter-region

5.29.4 Charging model

Work Unit
Per Database Instance

5.29.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
provision cluster	3 tokens
scale cluster	3 tokens
delete cluster	3 tokens
update Cassandra configuration	1 token
Other changes	Estimation in tokens based on time spent

5.30 Azure Cache For Redis

5.30.1 Description

Azure Cache for Redis provides an in-memory data store based on the Redis software. Redis improves the performance and scalability of an application that uses backend data stores heavily. It's able to process large volumes of application requests by keeping frequently accessed data in the server memory, which can be written to and read from quickly.

5.30.2 Service Tiers

Azure Cache for Redis is available in these tiers:

Tier	Description
Basic	An OSS Redis cache running on a single VM. This tier has no service-level agreement (SLA) and is ideal for development/test and non-critical workloads.
Standard	An OSS Redis cache running on two VMs in a replicated configuration.

Premium	High-performance OSS Redis caches. This tier offers higher throughput, lower latency, better availability, and more features. Premium caches are deployed on more powerful VMs compared to the VMs for Basic or Standard caches.
Enterprise	High-performance caches powered by Redis Inc.'s Redis Enterprise software. This tier supports Redis modules including RediSearch, RedisBloom, and RedisTimeSeries. Also, it offers even higher availability than the Premium tier.
Enterprise Flash	Cost-effective large caches powered by Redis Inc.'s Redis Enterprise software. This tier extends Redis data storage to non-volatile memory, which is cheaper than DRAM, on a VM. It reduces the overall per-GB memory cost.

5.30.3 Build to run service included in the OTC

5.30.3.1 Build service pre-requisite

- Refer to generic description.

5.30.3.2 Build to run service

- Refer to generic description.

5.30.4 RUN services included in the MRC

5.30.4.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.30.4.2 KPI & alerts

Monitoring

Yes

KPI monitored

Azure Monitor supported metrics for Azure Cache for Redis are available at:
Azure Monitor supported metrics by resource type - Azure Monitor | Microsoft Docs

Alerts observed

- Cache Latency (preview)
- Cache Misses
- Cache Miss Rate
- Cache Read
- Cache Write
- Connected Clients
- Connections Created Per Second
- Connections Closed Per Second
- CPU
- Errors
- Evicted Keys
- Expired Keys

5.30.4.3 Backup and restore

Data backup and restore

In Standard and Basic tiers, all data is stored in the memory of the service, meaning that data loss is possible if a failure occurs where Cache nodes are down. For the Premium tier, Redis persistence can be configured using either Redis Database (RDB) or Append Only File (AOF):

- RDB persistence - When you use RDB persistence, Azure Cache for Redis persists a snapshot of your cache in a binary format. The snapshot is saved in an Azure Storage account. The configurable backup frequency determines how often to persist the snapshot. If a catastrophic event occurs that disables both the primary and replica cache, the cache is reconstructed using the most recent snapshot.

- AOF persistence - When you use AOF persistence, Azure Cache for Redis saves every write operation to a log. The log is saved at least once per second into an Azure Storage account. If a catastrophic event occurs that disables both the primary and replica cache, the cache is reconstructed using the stored write operations.

If data persistence is enabled, geo-replication can't be enabled for the same cache.

Service restore

Service restore will be from Infra as Code.

5.30.4.4 Azure SLA High Availability and Disaster Recovery inter-region

Azure Cache for Redis provides built-in redundancy by hosting each cache on two dedicated virtual machines stored in separate update and fault domains. This applies to Standard, Premium and Enterprise tiers. To avoid datacenter level failures, zone redundancy is also supported for the Premium and Enterprise tiers and can be configured during the deployment process. With zone redundancy enabled, the cache runs on VMs spread across multiple availability zones, which provides higher resilience and availability with this configuration enabled, the data transfer between Azure Availability Zones will be charged at Microsoft's standard bandwidth rates.

Additionally, the service supports geo-replication for Premium tier only. Geo-replication is designed as a disaster-recovery solution and links together two Premium Azure Cache for Redis instances as well as creates a data replication relationship. The two instances can be hosted in the same region or in two different regions, with one instance acting as primary and the other as secondary. The primary handles read and write requests and propagate changes to the secondary.

Automatic failover across Azure regions isn't supported for geo-replicated caches, meaning that a manual failover has to be performed during a disaster recovery scenario. To avoid performance issues, Microsoft recommends bringing up the entire application stack in a coordinated manner in the backup region.

Various high availability options are available in the Standard, Premium, and Enterprise tiers:

Option	Description	Availability	Standard	Premium	Enterprise
Standard replication	Dual-node replicated configuration in a single data center with automatic failover	99.9%	✓	✓	✓
Zone redundancy	Multi-node replicated configuration across Availability Zones, with	99.9% in Premium; 99.99% in Enterprise	-	✓	✓

	automatic failover				
Geo-replication	Linked cache instances in two regions, with user-controlled failover	Premium; Enterprise	-	Passive	Active
Import/Export	Point-in-time snapshot of data in cache.	99.9%	-	✓	✓
Persistence	Periodic data saving to storage account.	99.9%	-	✓	Preview

5.30.5 Charging model

Work Unit
Per Redis cache

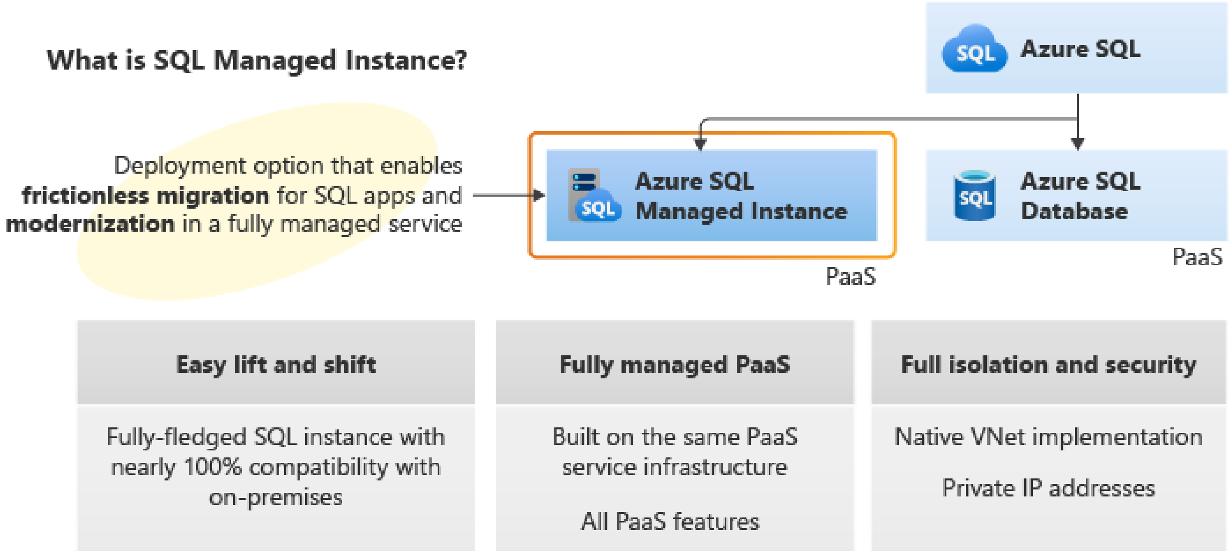
5.30.6 Changes catalogue – in Tokens, per act

Changes examples	Effort
Reboot Redis services	I token (not available for enterprise tire)
Scaling the resource	2 Token
Changing the service Model	1 Token
Other changes	Estimation in tokens based on time spent

5.31 Azure SQL Managed Instance

5.31.1 Description

Azure SQL Managed Instance is the intelligent, scalable cloud database service that combines the broadest SQL Server database engine compatibility with all the benefits of a fully managed and evergreen platform as a service. SQL Managed Instance has near 100% compatibility with the latest SQL Server (Enterprise Edition) database engine.


What is SQL Managed Instance?
Deployment option that enables **frictionless migration** for SQL apps and **modernization** in a fully managed service
Azure SQL
Azure SQL Managed Instance
PaaS
Azure SQL Database
PaaS
Easy lift and shift
Fully-fledged SQL instance with nearly 100% compatibility with on-premises
Fully managed PaaS
Built on the same PaaS service infrastructure
All PaaS features
Full isolation and security
Native VNet implementation
Private IP addresses

5.31.2 Build to run service included in the OTC

5.31.2.1 Build service pre-requisite

- Refer to generic description.

5.31.2.2 Build to run service

- Refer to generic description.

5.31.3 RUN services included in the MRC

5.31.3.1 Run service pre-requisite

- A referential file exists in the Git including the reference configuration of the service.
- This file can be executed with a CI/CD and the execution has been tested successfully.

5.31.3.2 KPI & alerts

Monitoring

Yes

KPI monitored

Azure Monitor supported metrics for Azure Database for MySQL are available at:
Azure Monitor supported metrics by resource type - Azure Monitor | Microsoft Docs

Alerts observed

Built-in monitoring of basic MI telemetry (CPU, storage, IOPS).

5.31.3.3 Backup and restore

- **Data backup and restore**

Automated Backups: Full backups are taken every 7 days, differential 12 hours, and log backups every 5-10 min

- **Service restore**

Point-in-time Recover: It is possible to restore any database to an earlier point in time within its retention period.

5.31.3.4 Azure SLA High Availability and Disaster Recovery inter-region

The auto-failover groups feature allows you to manage the replication and failover of some or all databases on a logical server to another region.

5.31.4 Charging model

Work Unit
Per Database Instance

5.31.5 Changes catalogue – in Tokens, per act

Changes examples	Effort
Provision Managed Instance	2 tokens
Instance property change (admin password, Azure AD login, Azure Hybrid Benefit flag	1 token
Instance storage scaling up/down	1 token
Instance compute (vCores) scaling up and down	1 token
Other changes	Estimation in tokens based on time spent

6 End of the document