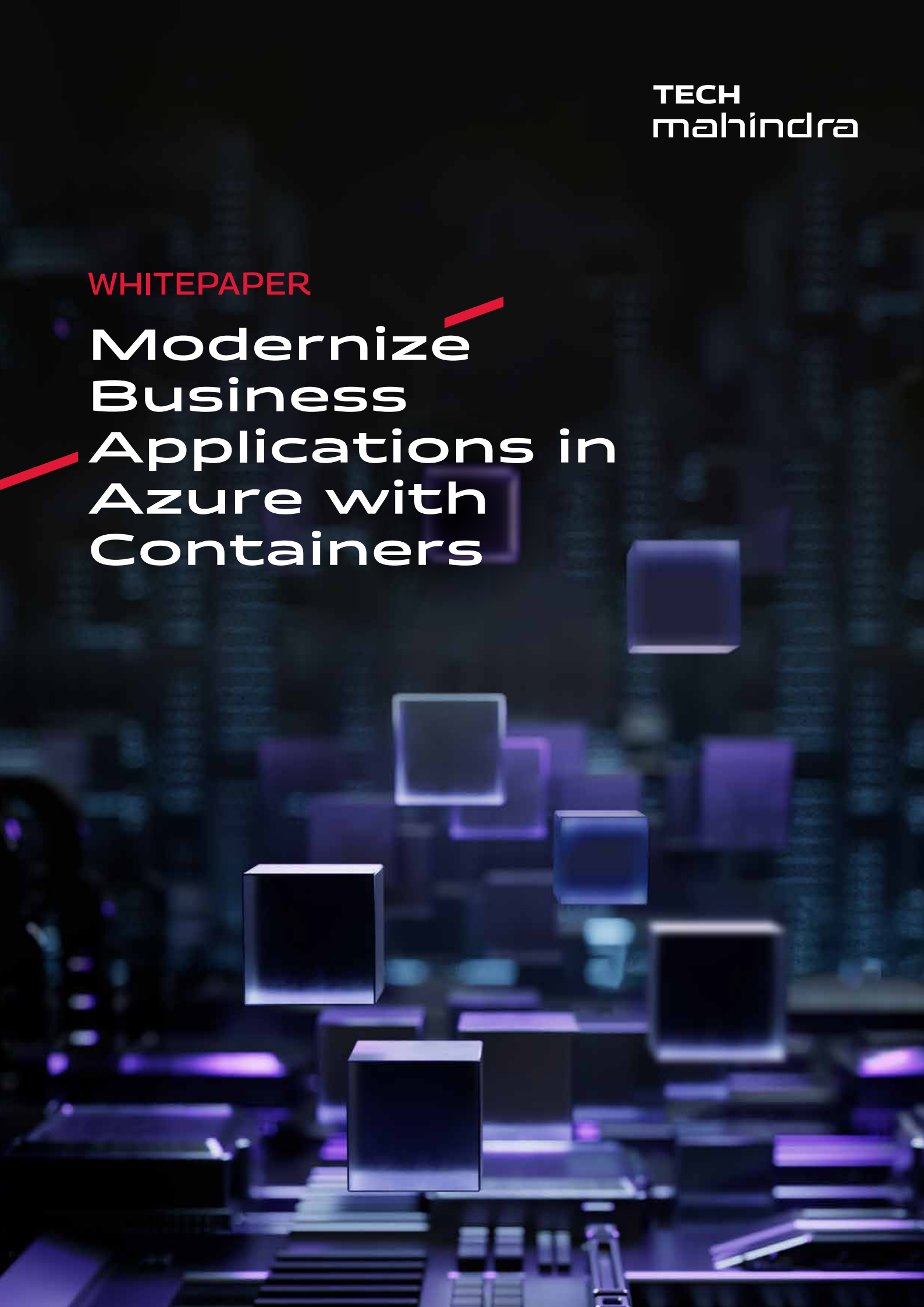
TECH mahindra

WHITEPAPER

Modernize Business Applications in Azure with Containers



Abstract

This whitepaper is intended for architects and developers who want to deploy containerized applications on Microsoft Azure. The paper covers details about monolithic architecture and microservices and why should we adopt containers to modernize business applications by going towards microservices. In this whitepaper, we have shared details of several Azure services with their respective implementation scenarios to provide insights around the starting point for building microservices with containers on Azure.

Key Takeaways

Application Modernization on Azure

Adoption of Containers for Microservices

Use Cases for Containers

Azure Offerings for Containers

Guidance and Recommendation to Choose the Correct Azure Services

A case study around one of the world's biggest logistics company

TechM's capability on Azure

Introduction

Enterprises are now building a set of small focused microservices to deliver their features faster rather than building a large singular tightly coupled application. Whether you are considering a legacy application or a greenfield application for containers, it always provides you an agile, scalable, and efficient microservices.

By using Azure's services for containers, enterprises are getting maximum flexibility and scalability while deploying their application into cloud. In this whitepaper, we will go through different container services that are currently offered by Microsoft Azure and how these services help you to modernize your applications.

Industry Landscape

Nowadays, every enterprise is heading towards microservice based architectures so that their application can become an agile and scalable product. To respond to the rapid changes in today's world, an application must be quick to deploy, always available, and easy to maintain.

Microservices with containers are emerging as the best way to develop today's modern applications. You can run containers on any on-prem hardware, in the cloud, or in a hybrid environment.

About the Solution

Before starting with containers and the different services for containers in Azure, let's begin with the architectural approaches. First, we will discuss about monolithic and microservices along with their pros and cons, and then gradually we will introduce the different Azure services which you can use to build your microservices with containers.

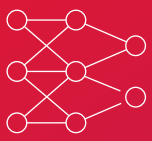
Monolithic and Microservices

A monolithic architecture is a traditional model of an application, which is built as a single unit that is self-contained and independent from other applications. Building and managing applications can often be complex in monolithic architecture.

It offers you a single business application with a large source code base that tightly couples all the business functions together. To roll out a new feature, you need to build, test, and deploy the entire code stack every time. Hence there is a huge restriction in faster release and the whole process of release deployment is time-consuming.

On the other hand, adopting a microservices approach to your application development can improve the way you build, test, and deploy your application which eventually leads to deliver features faster.

Microservices are a set of small, independent, and scalable applications that communicate to each other over standard APIs.



Benefits of Microservices over Monolithic Applications

- Microservices are small, independent, and loosely coupled applications. Unlike a large team needed in monolithic application development, a small team of developers can write and maintain a fully functional service which can give you a complete application functionality. For example, you can have different small teams for different set of application tier like web, app, and data.
- Unlike a huge codebase used for monolithic app, every microservice has a separate smaller codebase. Hence the codes are easier to track, understand, troubleshoot, update, and manage for the developers.
- Microservices can be built, deployed, tested, scaled independently. You can update a specific microservice without rebuilding and redeploying the whole application.
- Microservices enable a DevOps culture in your team which leverage the benefits of the Agile methodology of continuous integration and continuous delivery (CI/CD) for software development.
- Microservices are responsible for persisting their own data. This differs from the traditional monolithic model, where a separate data layer handles the data persistence for the entire application.
- Microservices support polyglot programming for your entire application stack. That means, you can use different programming language and framework for different services, taking most out of these tech stacks. As these microservices will communicate to each other over an API, your entire application does not depend on a specific programming language, library, or framework.

Adoption of Containers

Adoption of containers has soared in recent years and containerizing business-critical application is a fast-moving trend.

Containers help you to innovate, perform rapid changes in your code and scale the application in no time without effecting other application components.

Here in this whitepaper, we will discuss about the containers and different Azure services for container which will help you to modernize your applications as microservices and to go cloud native.

A container consists of source codes and all its dependencies. Application can be deployed and run quickly and reliably in any containerized platform as you don't need any separate application runtime to deploy your application.

Container images which are immutable and lightweight in nature, include all the dependent libraries, configurations, runtimes, packages which are needed to run the application. Containers are portable and platform agnostic. Hence this eliminates the application portability problem i.e., getting errors while deploying the same application across different platform/environment.

Containerised application can be tested as a single unit and deployed in any containerized environment like Docker or Kubernetes. Due to its seamless portability functionality, Containers enable developers to deploy their applications across environments with little or no modification.

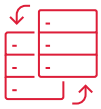
The emergence of containers, with their very fast start-up, application packaging, and isolation model, is further contributing to efficiency and agility to frequent application releases.

Use Cases - Containers

Application Modernization - Cloud native apps
Containers are used in cloud-native applications to provide a uniform operating paradigm across different environments. You can modernize your application by containerizing them so that they can be migrated to cloud.

Lift and shift existing applications into Cloud
Although shifting to the cloud might provide considerable benefits, an organisation may not want to update an existing application. You can shift your apps to the cloud using containers without modifying any code.

Multi Cloud Deployment
Containers help you deploy your applications across the different cloud environments due to its portability feature. You can migrate them from one cloud to another on-premise, to the cloud, or vice versa.



Batch
Containers make it simple to perform batch processes without needing to maintain an environment and dependencies. Dynamic computing solutions, such as Azure container instances (ACI), may be utilised to effectively ingest source data, analyse it, and store it in a long-term store, such as Azure Blob storage.

Machine learning
Machine learning applies algorithms to data to create predictions based on patterns discovered in the data. With containers, Machine learning programmes can be self-contained and easily scalable in any environment.

DevOps Adoption
It is very easy to configure containers as you only need to put those configurations in your source code. Containers will include all these configurations in its image build stage. You can adopt DevOps with containers by integrating them with CI/CD pipelines.

Azure Service Offerings for Containers

If you want to	Use this
Deploy and scale containerized application on managed Kubernetes cluster	Azure Kubernetes Service (AKS)
Build, deploy and scale microservices using serverless containers in Azure	Azure Container Apps
Execute event-driven, serverless code with an end-to-end development experience	Azure Functions
Run containerised web apps on Windows and Linux	Web App for Containers
Deploy containers with hypervisor isolation and without managing servers	Container Instances
Build, store, secure, scan, manage and replicate container images, and artifacts in Azure	Container Registry
Deploy and operate always-on, scalable, distributed apps in Azure, on-premises, or other clouds	Service Fabric
Deploy and scale containers on highly available, fully managed RedHat OpenShift clusters	Azure Red Hat OpenShift

As you can see, there are several options for you to build and deploy containerized workloads on Azure. Here in the next section, we have tried to give you some guidance and recommendation so that you can choose the best fit for your requirement.

Guidance and Recommendations to Choose the Correct Azure Services for Containers based on TechM's success stories with customers

Azure Kubernetes Service

Azure Kubernetes Service (AKS) is a fully managed Kubernetes solution offered by Azure. Azure Kubernetes service is the best choice for teams that are searching for a fully managed version of Kubernetes in Azure to deploy, scale and manage containers and container-based applications. You can integrate AKS with Azure DevOps or GitHub action to perform CICD operations. It also supports open sources like Helm, Istio, and Consul.

Azure Container Apps

Azure Container Apps (ACA) allows you to build serverless microservices with containers. It is best suited for a particular application that span several microservices deployed in containers. Direct access to the underlying Kubernetes APIs is not made available by Azure Container Apps. Use Azure Kubernetes Service (AKS) if you need access to the Kubernetes APIs and control plane. But if you want to create Kubernetes-inspired apps but don't need direct access to all the native Kubernetes APIs and cluster administration, container apps offers a fully managed experience based on industry best practises. And for this, many teams prefer to start their journey to containers with Azure container apps and over the period, they move to Azure Kubernetes Service.

Azure Functions

Azure Functions is a serverless solution that allows you to write less code, maintain less infrastructure. It is a serverless functions-as-a-service (FaaS) solution from Azure. It is intended to run event-driven apps that use the functions programming model. For teams wishing to initiate the execution of functions on events and bind to other data sources, the Azure functions programming architecture delivers productivity benefits. The Azure functions programming paradigm is accessible as a basic container image, allowing teams to migrate it to other container-based computing platforms and reuse code as environment needs evolve. It can be used functions to build web APIs, respond to database changes, process IoT streams, and manage message queues.

Azure App Service (Web App for Containers)

Azure App Service offers fully managed online application hosting solution that includes websites and web APIs. You can use Azure app service to deploy your web application as it is designed specifically for web apps and it can be integrated easily with other Azure services like Azure container app and Azure functions. Azure app service is an excellent choice for developing online applications and APIs.

Azure Container Instances

Azure Container Instances (ACI) is the fastest and simplest way to run containers in Azure, without managing any virtual machines. It provides a single pod of Hyper-V isolated containers on demand. You can run your docker containers in a managed, serverless Azure environment. ACI enables you to operate in isolated containers, without any container orchestration functionality. When compared to container apps, it may be regarded of as a lower-level "building block" alternative. ACI containers do not include concepts like as scalability, load balancing, or certifications. Azure container instances is an excellent choice for any situation that requires isolated containers, such as simple web apps/APIs, task automation, and build tasks.

Azure Container Registry

Azure Container Registry allows you to build, store, and manage container images and artifacts in a private registry for all types of containerized deployments. You can use Azure container registry tasks to build container images in Azure on-demand, or automate builds triggered by source code updates, updates to a container's base image, or timers.

Azure Service Fabric

Azure Service Fabric is Microsoft's container orchestrator tool, a distributed systems platform, that makes it easy to package, deploy, and manage scalable and reliable microservices and containers. You can use the Service Fabric programming model or run containerized stateful services written in any language or code. You can run Service fabric both on premises and in other public clouds, in addition to Azure.

Azure Red Hat OpenShift

Azure Red Hat OpenShift provides single-tenant, high-availability Kubernetes clusters on Azure, supported by both Red Hat and Microsoft. It enables you to deploy fully managed OpenShift clusters on Azure. Your team can pick their own registry, networking, storage, CI/CD solutions, or they can use OpenShift's built-in solutions for automated source code management, container and application builds, deployments, scaling, health management, and more from OpenShift. If your organisation already uses OpenShift, Azure Red Hat OpenShift is the ideal option.

Case Study - Application Modernization for one of the World's Biggest Logistics Company

The client is a world leader in the logistics business. It has delivered over a billion items globally with over 300,000 employees in over 200 countries. This company needed to transform some of their traditional enterprise programmes into microservice-based apps and transfer them to the cloud.

The key focus areas include, application modernization strategies, planning on application migration to cloud and breaking monolithic applications to microservice based applications assisting in containerizing these microservice based applications. Delivering apps and features faster with containers and container orchestration, implementing DevOps lifecycle and providing a stable environment and seamless operations support. It also aims at enhancing security and enabling intelligent security analytics and threat intelligence along with reducing the overall total cost of ownership (TCO).

Challenges:

Aging Infrastructure

- Low efficiency and reliability.
- High operational costs and capital expenditure.
- Growing security, audit, and compliance requirements.
- Inflexible and unable to keep up with business growth.

Stagnant Architecture

- Legacy stack and code.
- Long deployment times and release cycles.
- Incompatibilities with modern software systems.
- It's hard or impossible to add new functionality.

Solution:

TechM modernized the existing tightly coupled monolithic applications to loosely coupled micro services and deployed to Azure Kubernetes Services. TechM has designed and built the solution with various Microsoft Azure services like Azure Kubernetes Service (AKS), Azure DevOps, Azure databases, Azure policy, Azure sentinel. This architecture also incorporates third-party technologies like Splunk and Helm. The complete landing zone was built using terraform, an Infra-as-Code (IaC) tool.

The services which are offered are:

- Establishment of the right cloud strategy as per the business needs and evaluating the impact of issues related to security, governance, risk, and compliance.
- Defining the cloud-based application modernization strategy
- Implementing the project
- Provided Managed Operation Services like, 24/7 operations and monitoring of the platform by our support engineers, providing L1, L2 and L3 supports, escalation to Microsoft Support team, managed blue/green deployment process and updating and maintaining the workload templates

Services Used:

Microsoft Services:	Third Party Services
Microsoft Azure Kubernetes Service (AKS)	Terraform
Azure DevOps	Splunk
Azure Sentinel	Helm
Azure Policy	
Azure Container Registry	

Benefits:

Our solution helps the client with:

- Reduced total cost of ownership (TCO) with 10x cost reduction
- Faster time to market with 65% faster developments and 13x more releases
- Better performance and less downtime from highly available applications with 62% better availability
- Better agility, scalability, and portability due to adoption of containers
- Improved security with Azure's distributed Denial-of-services(DDoS) and threat protection services
- Reduction in IT admin cost due to the self-service module implementation to provision for the resources
- Improved productivity due to less operational activities and more focus on the app deployments rather than managing the cluster

The customer now ships new features faster, achieve scalability with confidence, increased operational efficiency by reducing the time and budget spent on infrastructure management.

Whitepaper: https://files.techmahindra.com/static/img/pdf/Enterprise-application-modernization-azure-kubernetes.pdf

Conclusion

Microsoft Azure provides a wide range of services that can be used to deploy your containerized workloads. These Azure services, with their comprehensive orchestration, security, and DevOps features, assist us in achieving agility, security, and scale with ease.

In this whitepaper, we have addressed how customers can get the benefits by adopting the microservices with several Azure services which are present for all the needs for containerized workloads.Azure services help enterprises to modernize their business applications and transform them into microservices and deploy those application into Azure's managed Kubernetes cluster.

Customers can automate the whole release lifecycle management with the help of DevOps CICD and Helm.

How TechM can Help You to Expedite Your Application Modernization Journey

Our vast experience in containers, Kubernetes, DevOps and experienced SMEs from Microsoft business unit will help you to assess and modernize your application workloads and also help them to get deployed in different Azure services for containers with respect to their best fit.

With the help of TechM's Azure competency team, you can scale and secure your applications rapidly in Azure with containers and build microservices to deliver features faster. Our delivery team will assist you to implement CI/CD to develop, deploy and update your containerized applications with either Azure's CICD platform Azure DevOps or Microsoft's GitHub Actions.

Tech Mahindra Microsoft Azure Competency

Tech Mahindra is a Gold Certified Microsoft partner with an association spanning over 15 years. As a Gold level, GSI partner for Microsoft, Tech Mahindra helps enterprises accelerate business transformation with a full spectrum of cloud services leveraging our proven accelerators, frameworks and solutions powered by Azure with a dedicated centre of excellence.

Tech Mahindra has been recognized as a Microsoft Azure expert managed service provider (MSP), enables us to provide a comprehensive Azure cloud service portfolio, to support the journey of digital transformation for global customers. We have dedicated Microsoft business unit with more than 8000+ FTEs and dedicated regional delivery centres.

We currently have 4 dedicated Azure labs, having more than 3900 Certified Azure architects and 2500+ accredited associates.

The customer now ships new features faster, achieve scalability with confidence, increased operational efficiency by reducing the time and budget spent on infrastructure management.

Some of the advance specialization that we have completed.

- Kubernetes on Microsoft Azure
- Modernization of Web App to Microsoft Azure
- Linux and Open-Source Database Migration to Microsoft Azure
- Cloud Security
- Network Services on Azure

From Microsoft Azure Competency, we offer,

- Cloud strategy consulting
- Cloud transformation and migration
- Cloud managed services
- Cloud application services
- Industry and domain specific solutions

About the Authors

Mr. Guru Prasad C P

Group Practice Head, Azure Cloud Services

Guru Prasad C P has an experience of over 22 years with over 8 years specifically in the public cloud working in Asia, ANZ, Europe and the US. His experience includes, setting up practice teams aligned to industry verticals and horizontals, analyst interactions for positioning the offerings, hiring the right talent, involving in strategic exercise mergers and acquisitions, organization building, and creating frameworks and IP's. At Tech Mahindra he is responsible for practice and competency development which includes alignment with OEMs for solutions, offerings and adoption of new technologies, customer interfacing where he acts as a trusted advisor in providing unbiased views/opinions and aligning with organization goals at the same time, value creation, developing practice areas deal making, solution support for large deals, and carve out deals from azure and hybrid cloud perspective

Mr. Arunava Basu

Solution Architect (Application Modernization/ SRE/ DevOps/ Automation)

Arunava Basu is a Solutions Architect at Tech Mahindra. He is a seasoned professional with 13 years of experience in architecting cloud native applications, migrations, and administration. He also has experience as a cloud DevOps and automation architect to automate tools infrastructure, CI-CD Platform, application provisioning, deployment management with deep understanding and scaling of DevOps process and tools to build stable products. Arunava has a keen interest in automating things.

Mr. M Rajashekar Reddy

Solution Architect (Application Modernization/ Integration/ Analytics)

M Rajashekar Reddy is a Multi-Cloud Architect with 15 years of experience, leveraging proven product, program management, pre-sales and technical architecture skills. His experience spans the aerospace, healthcare, insurance, power and energy, oil and gas, semiconductors, geo-spatial, and transportation industries. As part of his expertise, he is responsible for Cloud Roadmaps and Transition, Digital Innovation and Strategy, Cloud Native Transformations, DevSecOps Governance, Application Modernization and Migration, Data Governance, Microservices, API Integrations, API Ops, Data Ops, MLOps, Product Evaluations, SAP, Salesforce, Pro-Active Pitches, Pursuit Ownership and Contribution to the sales cycle, Proposal Management, NPD, GTM Strategy, and Roadshows.