Tech Mahindra Response to ‘BT’

for Creation of platform to centralise the delivery of machine learning capabilities

Statement of Confidentiality

The information contained herein is proprietary to Tech Mahindra Limited and may not be used, reproduced or disclosed to others except as specifically permitted in writing by Tech Mahindra. It has been made available to BT solely for an objective, evaluation of Tech Mahindra's solutions and services, and such information may be disclosed only to those employees of BT, who have the need to know such information in order to perform the evaluation.

Document Ownership

Department	Date
Tech Mahindra	

Document Revision History

Author	Version	Date	Reviewed by	Review Comments	Approved by & Version Approved
Manish	Draft	23-Feb-21			

Document Validity

This document shall remain valid and open for acceptance for a period of 60 days from the above date of submission to BT.

Business Primary Contact

Name	V.A Ramkumar
Title	Client Delivery Head
Address	RMZ Eco World, Tower 4A&4B, 4th Floor, Outer Ring Road, Bangalore – 560103
Mobile	+91 9820765659 / UK Dial: 01252696431
E-Mail	ramk@TechMahindra.com

Copyright © 2020, Tech Mahindra. All rights reserved.

Table of Contents

Copyright © 2020, Tech Mahindra. All rights reserved.

1. Background

Telecom industry is investing significantly in artificial intelligence (AI) and machine learning (ML) applications to monetize data assets, improve customer experience, customize product and service offerings, drive business growth, and enhance operational efficiencies. As a Tier 1 Telco in UK, British Telecom plans to create world class platform to centralise the delivery of machine learning capabilities pan BT ensuring standards of build, deploy and integration allowing different business units to benefit from reuse and faster time to market thus enabling competitive edge.

The platform should have following capabilities:

- Confirm to Industry Standards
- Deliver Infrastructure Services
- Extensible to deliver customized solutions
- Capability oriented
- Ease of integration and catalog driven
- Leverage crowd sourcing
- Support edge compute and Integration

2. Scope

Based on our understanding of BT's vision and needs to implement a world class BT platform, this document focuses on defining a high-level platform architecture strategy and approach, which can help implement, operate and accelerate the adoption of AI/ML across British Telecom (BT). The strategy takes into consideration the governance, environment, underlying business strategy and the operational needs of a robust AI/ML platform. The architecture development processes should help envision, define, articulate, build, validate and rollout the AI/ML Enterprise Architecture, Design, Implementation. It should align to BT's EA Standards, Policies, Processes and technology landscape.

3. Industry View and Context

AI/ML projects have evolved their own development methodologies including CRISP-DM i.e. Cross-industry standard process for data mining just like DevOps, these methodologies are grounded in principles and practices learned from real-world projects. AI/ML teams use an approach unique to data science projects where there are frequent, small iterations to refine the data features, the model, and the analytics question. It's a process intended to align a business problem with AI/ML model development nuances.

CRISP-DM breaks the process of data mining into six major phases:

- Business Understanding
- Data Understanding
- Data Preparation
- Modeling
- Evaluation
- Deployment

Copyright © 2020, Tech Mahindra. All rights reserved.

The sequence of the phases is not strict and moving back and forth between different phases as it is always required. The arrows in the process diagram indicate the most important and frequent dependencies between phases. The outer circle in the diagram symbolizes the cyclic nature of data mining itself. A data mining process continues after a solution has been deployed. The lessons learned during the process can trigger new, often more focused business questions, and subsequent data mining processes will benefit from the experiences of previous ones.

An open, adaptable AI/ML architecture will help an enterprise execute this process more effectively. This architecture requires several key technologies and capabilities to support it viz.:

• **AI/ML and DevOps tools** that allow data scientists, ML engineers, and application developers to create, deploy, and manage ML/DL models and applications.

• **Data pipelines** provide cleansed data to data scientists for creating, training, and testing ML/DL models and to application developers for data management needs.

• **A cloud platform** gives data engineers, data scientists, ML engineers, and application developers access to the resources they need to work rapidly.

• **Compute, storage, and network accelerators** speed data preparation, model development, and inferencing tasks.

• **Infrastructure endpoints** provide resources across on-site, virtual, edge, and private, public, and hybrid cloud environments for all stages of AI/ML operations.DevOps and AI/ML development are two independent methodologies with a common goal: to put an AI application into production. Today it takes the effort to bridge the gaps between the two approaches. DevOps for AI/ML is increasingly being refined as being referred to as MLOPS by the industry.

AI/ML projects need to incorporate some of the operational and deployment practices that make DevOps effective and DevOps projects need to accommodate the AI/ML development process to automate the deployment and release process for AI/ML models.

4. Key Architectural Considerations and Guiding Principles

Following are some of the key architectural considerations and guiding principles in envisaging, building and rolling out an AI/ML platform. This is based on our own learning and experience with similar exercises with our other customers

I. The architectural building blocks for AI/ML processing should support decoupling all the phases of the machine learning lifecycle from each other for maximizing flexibility.
II. The architectural processes and their building blocks should be repeatable, replicable and reproducible so that they lend to easier build, scale, monitoring, governance and help avoid reinventing the wheel.
III. The Catalog/marketplace of existing Assets should support collaboration and model sharing, its code and artifacts sharing and experimentation from different teams for each other's shared models to sponsor/promote reuse and cross pollination of innovations/capabilities across the organization.
IV. The MLOPs operations should abstract the ML lifecycle phases and should be able to orchestrate all of the phases seamlessly on a diverse set of infrastructure.

Copyright © 2020, Tech Mahindra. All rights reserved.

V. The governance and audit functions should be overarching across the ML infrastructure, that help capture the selected metrics that drive the ongoing future strategy, roadmap.

VI. Automation should be augmented in all the phases wherever possible to deliver efficiencies and scale to the ML processes and operations.

VII. Newer specific roles should be created that map the MLOPS personas and processes and their governance requirements in alignment with the EA function across BT.

VIII. The downstream application processes, use cases that consume the ML predictions and outcomes should be analyzed for impact and appropriate instrumentation be enabled that feeds back into the ML infrastructure for continuous learning and reinforcement.

IX. Ability to offer and consume infrastructure-as-a-service across different cloud platforms for model development and deployment.

X. ML solutions should be deployed and offered as a service or integrated within applications. Both lightweight and heavy weight versions should be created for a ML solution as per use case and requirement. That is ML solution should be deployed as a Docker image and micro-service or it should be deployed as a serialized object with consuming applications.

XI. Choice of ML solution development toolkit and infra is use case and data dependent. It is generally advisable to have the ML model training done nearer to the training data source. This enables better and faster and cheaper access to data and hence model training. Model deployment in turn may depend upon consuming applications and production data. Platform should support both these scenarios and hence not prescribe a particular ML development toolkit or environment

XII. Last, but not the least is the recommendation for a pilot use case development. A pilot use case should be identified whose execution and outcomes will help learn, establish and seed the principles and processes that will drive the eventual adoption of it's successful patterns across the organization. This pilot can be architected, scoped and executed quickly with well-defined success criteria that will mitigate and pre-empt hidden complexities, gray areas and surprises that may lurk around. The pilot will also help identify and leverage current best practices and processes that exist across the organization.

5. Design Considerations:

Based on guiding principles and lessons learned from several TECHM implementations, we recommend following design considerations for bringing AI/ML teams, processes and tools together:

a) Integrated DevOps for AI/ML

The AI/ML process relies on experimentation and iteration of models and it can take hours, days or weeks for a model to train and test. It makes sense to

- Carve out separate workflows to accommodate the timelines and artifacts for a model build and test cycles.
- For AI/ML teams, to think about models as having an expectation to deliver value over time rather than over an one-time construction of the model. Adopt practices and processes that plan for and allow a model lifecycle and evolution.
- Eensure that AI/ML is represented on feature teams and is included throughout the design, development, and operational sessions since DevOps is often characterized as bringing together business, development, release, and operational expertise to deliver a solution

b) Establish performance metrics and operational telemetry for AI/ML

Use metrics and telemetry to inform what models will be deployed and updated. Metrics can be standard performance measures like precision, recall, or F1 scores. Or they can be

Copyright © 2020, Tech Mahindra. All rights reserved.

scenario specific measures like the industry-standard fraud metrics developed to inform Fraud management about a fraud detection model's performance. Here are some ways to integrate AI/ML metrics into an application solution:

- Define model accuracy metrics and track them through model training, validation, testing, and deployment.
- Define business metrics to capture the business impact of the model in operations.
- Capture data metrics, like dataset sizes, volumes, update frequencies, distributions, categories, and data types. Model performance can change unexpectedly for many reasons and it's expedient to know if changes are due to data.
- Track operational telemetry about the model: how often is it invoked? By which applications or gateways? Are there any problems? What are the accuracy and usage trends? How much compute or memory/storage does the model consume?
- Create a model performance dashboard that tracks model versions, performance metrics, and data sets.

c) *Automate the end-to-end data and model pipeline*

The AI/ML pipeline is an important concept because it connects the necessary tools, processes, and data elements to produce and operationalize an AI/ML model. It also introduces another dimension of complexity for a DevOps process. One of the foundational pillars of DevOps is automation, but automating an end-to-end data and model pipeline is a complex integration challenge in itself.

Work streams in an AI/ML pipeline are typically divided between different teams of experts where each step in the process can be very detailed and intricate. It may not be practical to automate across the entire pipeline because of the difference in requirements, tools, and languages. Identify the steps in the process that can be easily automated like the data transformation scripts, or data and model quality checks.

Consider the following work streams:

Work stream	Description	Automation
Data Analysis	Includes data acquisition and focusing on exploring, profiling, cleaning, and transforming. Also includes enriching, and staging data for modeling.	Develop scripts and tests to move and validate the data. Also create scripts to report on the data quality, changes, volume, and consistencies.
Experimentation	Includes feature engineering, model fitting, and model evaluation.	Develop scripts, tests, and documentation to reproduce the steps and capture model outputs and performance.
Release Process	Includes the process for deploying a model and data pipeline into production.	Integrate the AI/ML pipeline into the release process
Operationalization	Includes capturing operational and performance metrics.	Create operational instrumentation for the AI/ML pipeline. For subsequent model retraining cycles, capture and store model inputs, and outputs.

Copyright © 2020, Tech Mahindra. All rights reserved.

Model Re-training and Refinement	Determine a cadence for model re-training.	Instrument the AI/ML pipeline with alerts and notifications to trigger retraining.
Visualization	Develop an AI/ML dashboard to centralize information and metrics related to the model and data. Include accuracy, operational characteristics, business impact, history, and versions.	n/a

An automated end-to-end process for the AI/ML pipeline can accelerate development and drive reproducibility, consistency, and efficiency across AI/ML projects.

d) Artifacts Versioning

. Versioning is about keeping track of an application's artifacts and the changes to those artifacts.

In software development projects this includes code, scripts, documentation, and files. A similar practice is just as important for AI/ML projects because—typically—there are multiple components, each with separate release and versioning cycles. For AI/ML projects, the artifacts could include:

- Data: training data, inference data, data metrics, graphs, plots, data structures, schemas
- Models: trained models, scoring models, A/B testing models
- Model outputs: predictions, model metrics, business metrics
- Algorithms, code, notebooks
- Versioning can help provide:
- Traceability for model changes from multiple collaborators
- Audit trails for project artifacts
- Information about which models predictions are consumed from which applications

A practical example of the importance of versioning for the AI/ML team happens when the performance of a model changes unexpectedly, and the change has nothing to do with the model itself. The ability to easily trace back inputs, dependencies, model, and data set versions could save days or weeks of effort. At a minimum, decide on a consistent naming convention and use it for the data files, folders, and AI/ML models. Several different teams will be involved in the modeling process and without naming conventions, there will be confusion over which data sets or model versions to use.

e) Consider container architectures

Data scientists, ML engineers, and application developers need access to their preferred tools and resources to be most productive. At the same time, IT operations teams need to ensure that resources are up to date, in compliance, and used in a secure manner. Containers let you quickly and easily deploy a broad selection of AI/ML tools across hybrid environments in a consistent way. Teams can iteratively modify and share container images with versioning capabilities that track changes for transparency. Meanwhile, process isolation and resource control improve protection from threats. Look for a robust, highly available container platform that includes integrated security features and makes it easy to deploy, manage, and move containers across your environment. Container architectures have the potential to streamline and simplify model development, test, and deployment. And as a package-based interface, containers make it easy for software applications to connect. Containers create an abstraction layer between models and the underlying

Copyright © 2020, Tech Mahindra. All rights reserved.

infrastructure. This lets the AI/ML team focus on model development and not worry about the platform. Containers can easily enable:

- A/B testing
- Deployment to multiple environments (IoT edge, local desktop, Cloud infrastructure)
- Consistent environment configuration and setup to drive faster model development, test, and release cycles
- Model portability and scalability

Choose an open source platform that integrates with a broad set of technologies to gain more flexibility and choice as AI and ML become increasingly more important components for applications, more pressure will exist to ensure they are part of an organization's DevOps model.

f) *Cloud vs In-premise dilemma.*

Another area of decision making will be where to carry training & inference pipeline. How much of the work load needs to stay in premise and how much can be pushed onto the cloud.

Most of the enterprise follows a hybrid cloud approach, where they traditionally have had data centers. TECHM has several customers who usually use at least a one- to two-year plan. Sometimes they might make that a status quo as well. That is, they would decide to have some part of their data and software in the cloud, and leave some part of it on-prem. That is also a strategy that most customers like to use.

AI/ML models, software, and applications require infrastructure for development and deployment. A consistent hybrid cloud platform allows you to develop, test, deploy, and manage AI/ML models and applications in the same manner across all parts of your infrastructure, giving you more flexibility. It can also provide self-service capabilities to speed resource delivery while maintaining IT control. Finally, a consistent platform supplies a foundation for technology integrations from third-party vendors, open source communities, and any custom-developed tools you may use. Look for technologies that connect to your existing databases, data lakes, and other repositories. Standardized application programming interfaces (APIs) and high-bandwidth, low-latency networking will make it easier to access data throughout the AI/ML life cycle. Integration with open source data streaming, manipulation, and analytics tools like Apache Spark, Kafka, and Presto can help you manage your data more efficiently. You should also select technologies that provide data governance
capabilities and integrated security features to protect your business.

g) *Security*

Security is another obligatory need, i.e. compliance ecosystems like GDPR, or HIPAA, PCI compliance. It is tricky to say that on-prem is all GDPR-compliant. One can be on-prem and still not be GDPR-compliant, and one can be in the cloud and still be GDPR-compliant. One has to re-analyze what to see or run an audit to see whether your new service that is going to be running on the cloud, is going to be compliant with your existing certifications. That is very important.

Security is very important, especially data security. So this is something that needs to be considered really well before you move any customer data or critical data to the cloud

h) *Data Location & Availability*

Copyright © 2020, Tech Mahindra. All rights reserved.

Another area to consider is the machine learning workload, or data science workload. The first point there is, for users to be able to do machine learning, they need the data. Data is what can run your machine learning algorithms on, which is what we call data gravity. One can run machine learning algorithms only in the location where they have the data. That means its physical location. If data is on-prem, or let's say, if your data is on cloud storage, on a specific region, that is the region in which you can run your machine learning workload and train your models.

One cannot run these machine learning models somewhere else because they need to access the data. There might be even heavy access. So proximity to data is probably very important. What that means is, before one can say that they can start to run machine learning workloads on the cloud, one need to make sure data is up, and ready to be used on the cloud, in the location that one is interested in. This also means that enterprise might have to work with data lake operators and administrators and see if they can move the data to that location, to the cloud. That could also means moving the data, or actually collecting the data in the cloud itself, or have a secure connection from on-prem data storage to the cloud, on wherever one is going to be running the models. That is probably the most important aspect of running machine learning workload on the cloud. Think about where the data is.

i) Work Load

Another important thing with machine learning workloads, especially during the training phase, is a classic example of a bursty workload. It's impossible to accurately predict how much resources one would need. Also, when one is running the machine learning or training, one might be using eight GPUs for two hours, and then could be shutting it down. This is a very bursty workload, and it does not make a lot of sense to purchase new hardware and have it in your on-prem data center, just for this bursty workload, because requirement could change very rapidly. It's best if it's in the cloud. It's what I would call a "perfect synergy."

6. PLATFORMS Capabilities

Looking at the above design considerations, an enterprise needs to invest in an AI Platforms and Framework which inherently has:

- Ease of Use,
- Meets Requirements,
- Ease of Setup,
- Quality of Support,
- Ease of Administration,
- Ease of customization,
- Supports Accessibility

Data Accessibility

Researchers ingest data from different sources. It's a dirty job and never been easy. There is no such place for researchers to get a glance at all of the data, to help them to understand relationships between the data. The importance of data for AI is undeniable. AI receives inputs, or requests, for a certain function or task, and in order to output a solution, it requires access to data. The more data AI has access to the better. Researchers have to get fully understand our data in a better way. That's why to improve data accessibility is not to be ignored.

Copyright © 2020, Tech Mahindra. All rights reserved.

Workflow Accessibility

From data ingestion, data labeling, data analysis, data transformation, model training, data validation, refine, re-training, deploy, evolution and serving. It's a very common workflow for machine learning operationalization. researchers don't share such a tool to simplify their workflow. Researchers require inevitable effort on the workflow, but most of their work is duplicate and difficult to reuse. We must offer a tool to deploy, monitor, and manage machine learning models. Track the health and performance of deployed machine learning models. Provide a holistic management tool to better understand all models deployed across a product.

Knowledge Accessibility

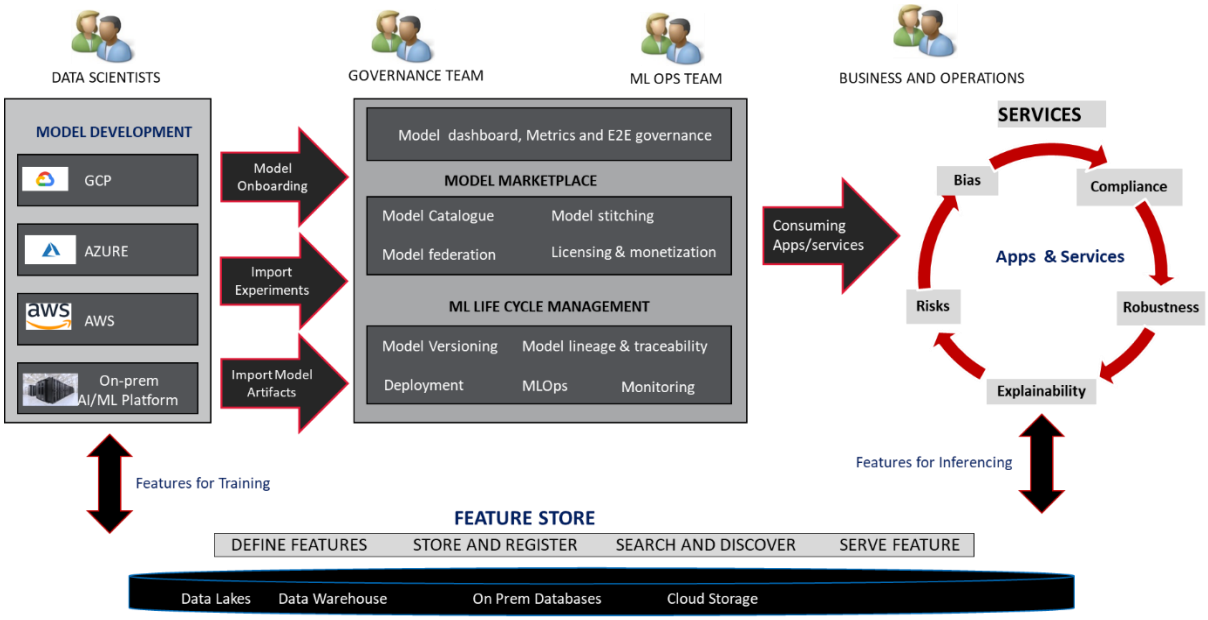
AI is becoming a staple of all business software and will continue to for the foreseeable future. We don't actually know much about AI, and how it can lead us to the coming few years. Organizational users write many a documentation, guide, specifications, design, standard, solution, etc. And these become their AI knowledge center. Everyone can access it and learn from it so they know how to apply AI to their business and develop new domains by AI. They become an AI-driven company eventually. build an AI knowledge center is the top priority for AI Platform.

Service Accessibility

AI infrastructure layer, ML services layer, AI services layer, they are services of AI Platform. These services should design for each other, evolve for each other. So the AI Platform could move quickly. Especially AI services layer, we define what we should serve in this layer so we know what we need to concentrate on. AI service should be the product ready layer.

7. Capability View

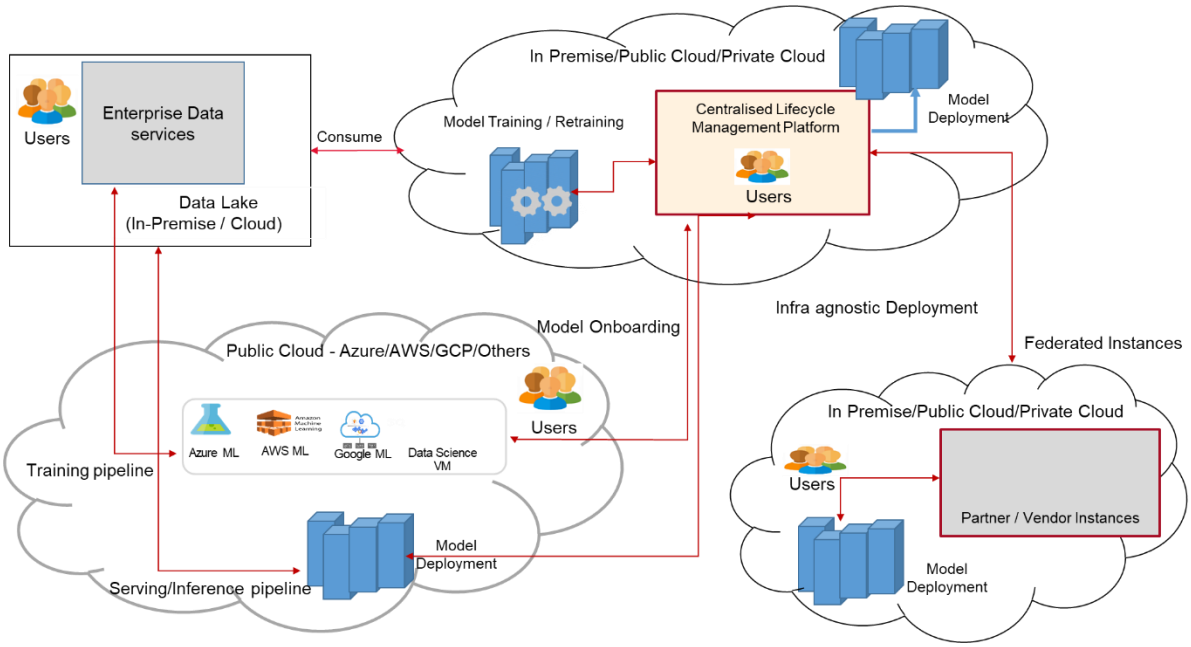
A high level capability view for such a platform will look like below:



Copyright © 2020, Tech Mahindra. All rights reserved.

8. Infrastructure view – Hybrid Environment

Since the model development, model lifecycle management and deployment infrastructure are going to stay hybrid for a long time in enterprises like BT, it is suggested to have a centralized Lifecycle management with integration to both north bound and south bound platforms, IDEs and applications. A generic infrastructure based view should look like below:



9. Case Studies:

Above architecture and design considerations are proposed based on various case studies and similar engagements of TECHM. Details on some of the case studies are given below:

Customer	Client Need	Architecture summary
Tier 1 CSP - North America	• How to reuse AI/ML assets fragmented across organization? • Platform for easy deployment of assets to production. • Governance and standard based development.	• Open Source based IDEs used for development • Deployment in Azure and in premise • Centralized Lifecycle management • Integration with existing ML platforms • Data availability in in Premise • Collaborative Catalog based ecosystem for collaboration among functions, partners and vendors • Integration with existing CI/CD pipeline

Copyright © 2020, Tech Mahindra. All rights reserved.

Tier 1 CSP – North America	• Cross functional COE to leverage organizational Assets • Standardization of AI / ML development effort & assets. • Integration of capabilities for business value generation.	• Open Source based IDEs used for development • Model development in Azure, GCP and In-Premise • Deployment in Azure, GCP and in premise • Centralized Lifecycle management • Integration with existing CI/CD • Data availability in In-Premise, GCP and Azure • Collaborative Catalog based ecosystem for collaboration among functions, partners and vendors
Tier 1 Transportation Company- North America (Pilot)	• Standardization of AI / ML development effort & ML Ops	• Platform deployment, support and Platform customization through GAiA, integration with AzureML
Tier 1 – FMCG Giant (Pilot)	• Standardization of AI / ML development effort & assets – integrated with Automation framework • ML Ops and E2E ML lifecycle management to streamline data science work going on in different functions	• Model development in Azure – Data Bricks • Model deployment in Azure • Collaborative Catalog based ecosystem for collaboration among functions, partners and vendors • Integration with RPA systems – Uipath • Integration with Azure /Databricks for centralized AI/ML lifecycle management

Copyright © 2020, Tech Mahindra. All rights reserved.

Disclaimer

Tech Mahindra Limited herein referred to as TechM provide a wide array of presentations and reports, with the contributions of various professionals. These presentations and reports are for information purposes and private circulation only and do not constitute an offer to buy or sell any services mentioned therein. They do not purport to be a complete description of the market conditions or developments referred to in the material. While utmost care has been taken in preparing the above, we claim no responsibility for their accuracy. We shall not be liable for any direct or indirect losses arising from the use thereof and the viewers are requested to use the information contained herein at their own risk. These presentations and reports should not be reproduced, re-circulated, published in any media, website or otherwise, in any form or manner, in part or as a whole, without the express consent in writing of TechM or its subsidiaries. Any unauthorized use, disclosure or public dissemination of information contained herein is prohibited. Individual situations and local practices and standards may vary, so viewers and others utilizing information contained within a presentation are free to adopt differing standards and approaches as they see fit. You may not repackage or sell the presentation. Products and notes mentioned in materials or presentations are the property of their respective owners and the mention of them does not constitute an endorsement by TechM. Information contained in a presentation hosted or promoted by TechM is provided “as is” without warranty of any kind, either expressed or implied, including any warranty of merchantability or fitness for a particular purpose. All expressions of opinion are subject to change without notice.

Thank You

Visit us at techmahindra.com

Copyright © 2020, Tech Mahindra. All rights reserved.