MX.3 on the Cloud
Deployment Strategy on Azure

MUREX™

TABLE OF CONTENTS

Contents

INTRODUCTION

DOCUMENT SCOPE

The document provides a guide to the full and hybrid deployments of MX.3 environments on the cloud. Recommended settings and deployment setups are presented to accompany clients intending to “lift-and-shift” their MX.3 environments to the cloud or deploying MX.3 for the first time on the cloud. The document also provides the best practices to leverage high-availability features of cloud vendors and to prepare the disaster recovery site lined up with the production site.

SUPPORTED OPERATING SYSTEM

Cloud infrastructure is supported as an operating environment for MX.3 application running over Red Hat Enterprise Linux.
Other Operating Systems are not supported by Murex on the Cloud.

ARCHITECTURE

OVERVIEW OF MUREX ARCHITECTURE

Murex is based on a three tier architecture structured as follows:

- The Client Tier which communicates through compressed incremental messages with the application tier. It provides access to all the application functionalities.

- The Application Tier which hosts all the business logic of the application, is based on three distinct layers:

  - The Business Layer exposes the business engines including among others aggregation engines, workflow engines, pricing engines, risk engines, accounting engines, deal life cycle engines, liquidation engines…

  - The Orchestration Layer is composed of vertically scalable servers, such as workflow server, position server, aggregation and compliance server… Each server orchestrating multiple engines hosted on the Business Layer.

  - The Monitoring Layer is composed of JMX monitoring federation services collecting all the information published by the application services.

- The RDBMS Tier which focuses on transactional storage of all business objects.

DEPLOYMENT STRATEGY ON THE CLOUD

INFRASTRUCTURE RECOMMENDATIONS

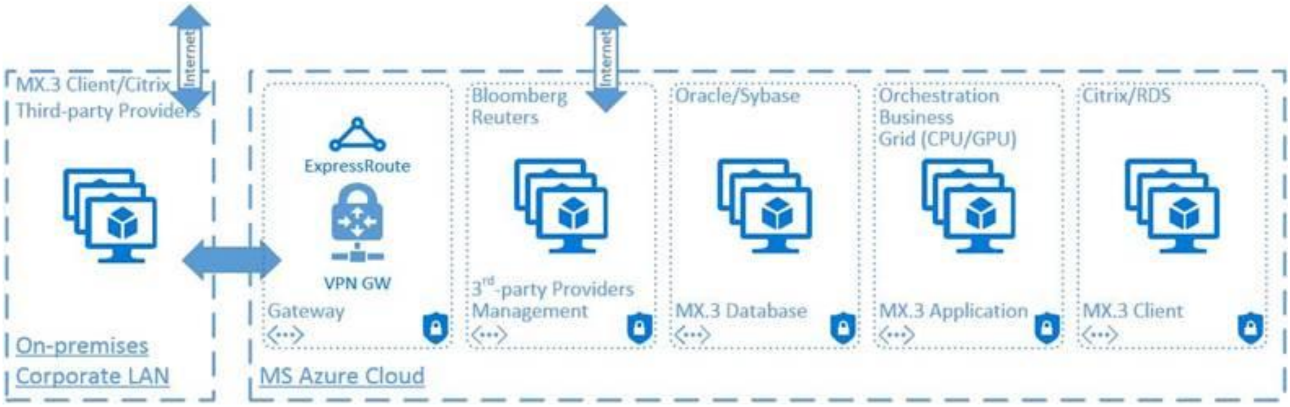
Deploying and operating MX.3 on the cloud requires extending the client’s network to access the infrastructure which will host the client, application and database tiers. This access is possible either through:

- VPN IPsec, or
- Microsoft Azure Express Route with a dedicated connection (recommended for better network performance, and increased reliability and data security)

Murex recommends deploying the MX.3 Production site on an infrastructure provisioned within the same Availability Zone. The infrastructure must conform to the Operating Environment requirements of the MX.3 platform documented in the file **fs/docs/MxOperatingEnvironmentRequirements.pdf** shipped with the Murex setups. This document contains information about the supported Operating System, Java version, Database client/server…

Deploying within the same Availability Zone, will ensure physical proximity among servers and therefore a reduced Network latency. Murex application requires low latency connection within the application layer. By design, the services on different application servers communicate constantly and heavily and therefore a very low sub-millisecond latency is required for a smooth operation of the system.

Azure Reserved instance can be used to guarantee the same compute capacity at all time. It is defined at a Region Level.



The table below provides additional recommendations to improve and stabilize the overall infrastructure performance and data security.

Recommendation	Rationale
Provision Linux VMs in the same Availability Zone	Reduce network latency
Select Linux VMs having at least 1 Gbps network bandwidth	Improve network throughput
Activate “Azure Accelerated Networking” on Linux VMs	Reduce network latency, jitter and CPU utilization
Store the MX.3 application directory and databases on “Azure Managed Disks” with Premium Storage option (SSD)	No data loss in case of VM failure Data encrypted at rest Protect Murex Intellectual Property
Disable “host caching on all Azure Managed Disks	Remove risk on Data Availability and Consistency
Oracle ASM preferred on cooked File System	Optimize Oracle Server performance

DEPLOYMENT RECOMMENDATIONS

Murex supports the deployment of MX.3 tiers (Client, Application and Database tiers) on the cloud in one of two possible deployment models: Full or Hybrid deployment.

Resource Tagging

Murex recommends following Microsoft best practices and tag all Murex IT assets on the Cloud to facilitate inventory keeping, security and access control, automation, and cost monitoring of all MX.3 infrastructure elements.
Please refer to the Azure official documentation on resource Tagging strategies and governance.

Full Deployment on the Cloud

In this deployment model, the Application and Database layers are hosted on the Cloud:

- The Client Tier is deployed either on the users' workstations or in the cloud on Citrix XenApp instances

- The Application Tier is deployed on a centralized application directory stored on Azure Managed Disks mounted on all application server machines. The table below maps the recommended virtual machine type to the different Application Tier layers for Production and Performance environments.

Application Tier Layer	Workload Type	Recommended VM Family
Orchestration	Memory-intensive	Memory Optimized
Business	CPU-intensive	Compute Optimized
Grid	- CPU-intensive - GPU-intensive	- High Performance Compute Optimized - GPU
RDBMS	IO-intensive	IO Optimized

- The Database Tier supports both Oracle and SAP ASE as database servers. As in on-premise deployments, Murex recommends hosting the Financial Database and Datamart Database on different VMs for optimal performance. Murex also recommends separating data and log devices on at least two Azure Managed Disks to optimize database performance. Managed Databases are not supported yet by Murex.

Test and Development environments can be hosted on "General Purpose" virtual machines.

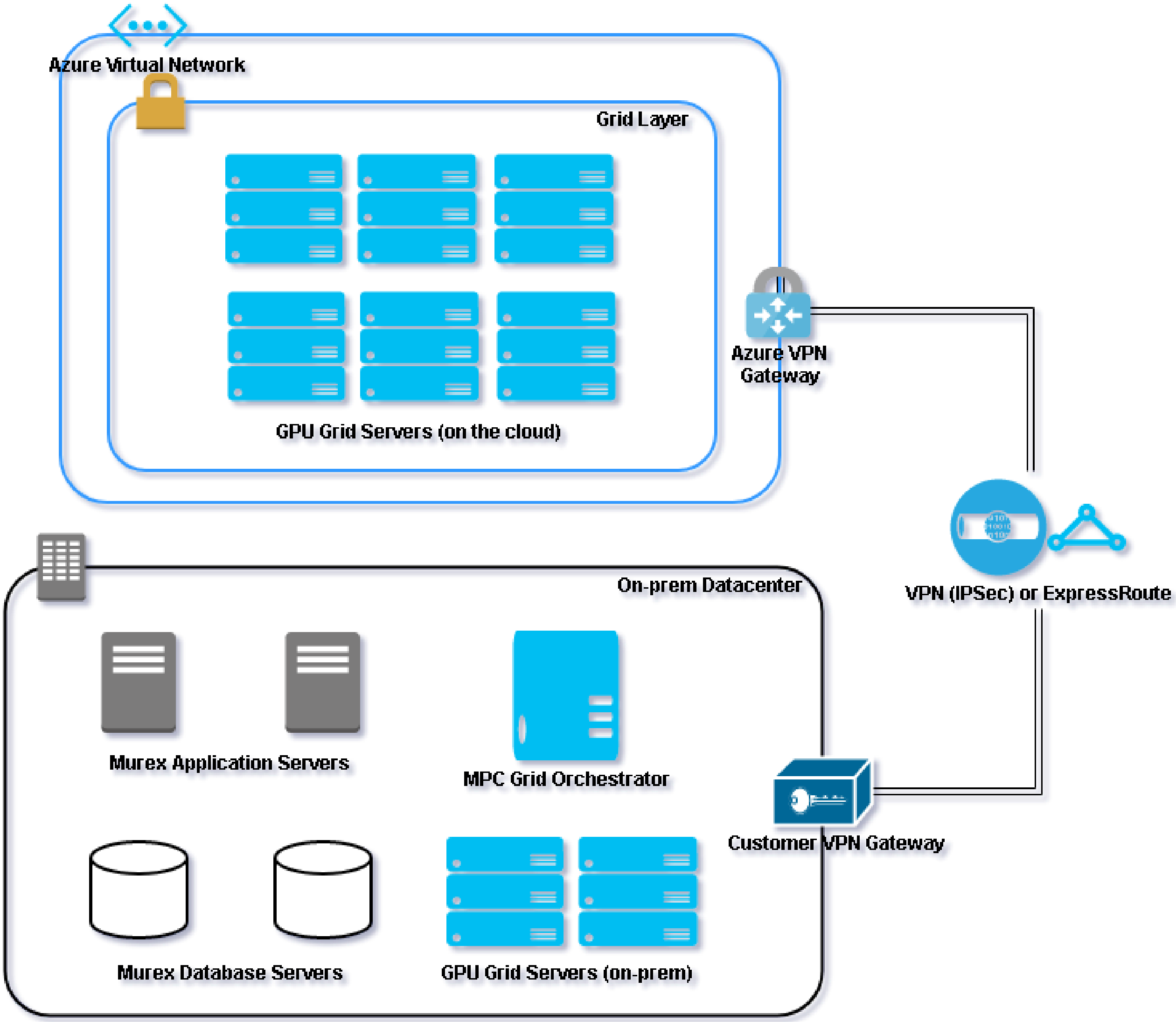
Note: Murex PAC contacts should be consulted for sizing the Hardware Infrastructure of the MX.3 environments. Note that performance is not strictly comparable between cloud infrastructure and on-premise physical infrastructure. Cloud infrastructure for MX.3 should be designed in consultation with Murex PAC contacts when specific benchmarks or KPIs (Key Performance Indicators) are targeted.

Hybrid Deployment

As of today, Murex has tested the following Hybrid architecture where only a part of the Grid Layer engines resides on the Cloud and the rest of the application (including the MPC orchestrator, and other Grid engines) is hosted on-premise in the Customer's data center.
Other Hybrid deployment architectures are currently being considered and should be tested for technical feasibility and Performance.

As shown in the below diagram:

- The MPC Grid orchestrator server is deployed on-premise
- The Grid engines servers are distributed between the local on-premise Datacenter and the Cloud
- The application servers and Database servers all reside in the on-premise Datacenter



» Connectivity to the Cloud

The connection between the on-premise Datacenter and the Azure cloud could be established in 2 possible ways:

- VPN IPsec tunnel where a VPN Gateway exists on each side of the VPN tunnel
- Azure Express Route service with a dedicated connection (recommended for better performance and increased reliability)

It's recommended to choose the connectivity option that ensure the lowest possible network latency to the Cloud to support the deployment of a performant and well responsive Grid layer.

Each Business case is different in terms of requirements on network latency depending on the required response time, amount of workload that needs to be processed on the Grid, and other factors. Therefore, it's advisable to collaborate with your PAC contact to define the required QoS on your connection to the Cloud for your particular use-case.

» Required Firewall configurations

In the above diagram, a Firewall typically exists behind the "Customer VPN Gateway", and it's required to have the following configurations to ensure the correct functioning of the MX.3 application in this architecture:

1. The firewall should allow **bi-directional** connectivity (as opposed to uni-directional) between the servers residing on-premises from one side to the servers on the cloud infrastructure from the other side
2. The firewall should **allow connections on Any port**: The Grid calculation sessions communicate over RMI protocol with the rest of the application, and therefore the ports used are random and cannot be predicted beforehand
3. The Firewall should enforce a **TCP Idle Timeout of no less than 30 minutes**
4. **TCP Keepalive** should be correctly configured on the Operation Systems of all participating application layer servers, Database layer servers, and Grid layer servers, as well as in the configuration files of MX.3 application. This will prevent the firewalls from taking down long living TCP sessions established between Grid servers and Application servers. Please check with your PAC team contact for more information on configuring TCP Keepalive for MX.3 application
5. The Firewall should allow connections to and from whitelisted IPs only for increased security. These are the list of Application Server IPs both on-premises and in the Cloud
6. MSS enforcement with **Max Segment Size set to 1350** as recommended by Microsoft (please check with the Cloud vendor for the latest recommended value)

LIFT-AND-SHIFT MX.3 ENVIRONMENTS

This section describes the “Lift-and-Shift” procedure for clients intending to move their MX.3 environments to the cloud. Murex expects clients to have personnel trained on the target cloud vendor in order to setup the infrastructure underlying MX.3 environments.

The steps below are applicable for both Production and Test/Dev environments:

- Audit the current production Hardware utilization (CPU, Memory, Disk IO, Network IO, …)
- Identify the corresponding Virtual Machines types and sizes on the cloud provider
- Setup the infrastructure in line with Murex Security Guidelines
- Restore the latest MX.3 backup (application directory and dump) on the cloud infrastructure
- Execute a test campaign with a primary focus on Performance
- Go-live on the cloud
- Monitor the infrastructure utilization and re-size (upsize/downsize) if needed

HIGH AVAILABILITY

FAULT TOLERANT PLATFORM

Similar to the on-premise deployments, each layer of the Application tier is deployed on, at least, two machines in order to ensure the resilience of the environment. In case of a general failure of one of the virtual machines, Microsoft Azure automatically spins a replacement machine having the same IP as the failed one. Azure offers the possibility to configure custom scripts to be executed upon the start-up of the replacement machine. Hence, lost services can be restarted on that machine via those scripts.

AUTOMATIC RECOVERY

To achieve automatic recovery from VM failures, it's possible to leverage the Azure monitoring features combined with Azure automation tools in order to detect failures and trigger an automatic provisioning of replacement servers and restart of failed services.

» Business Layer

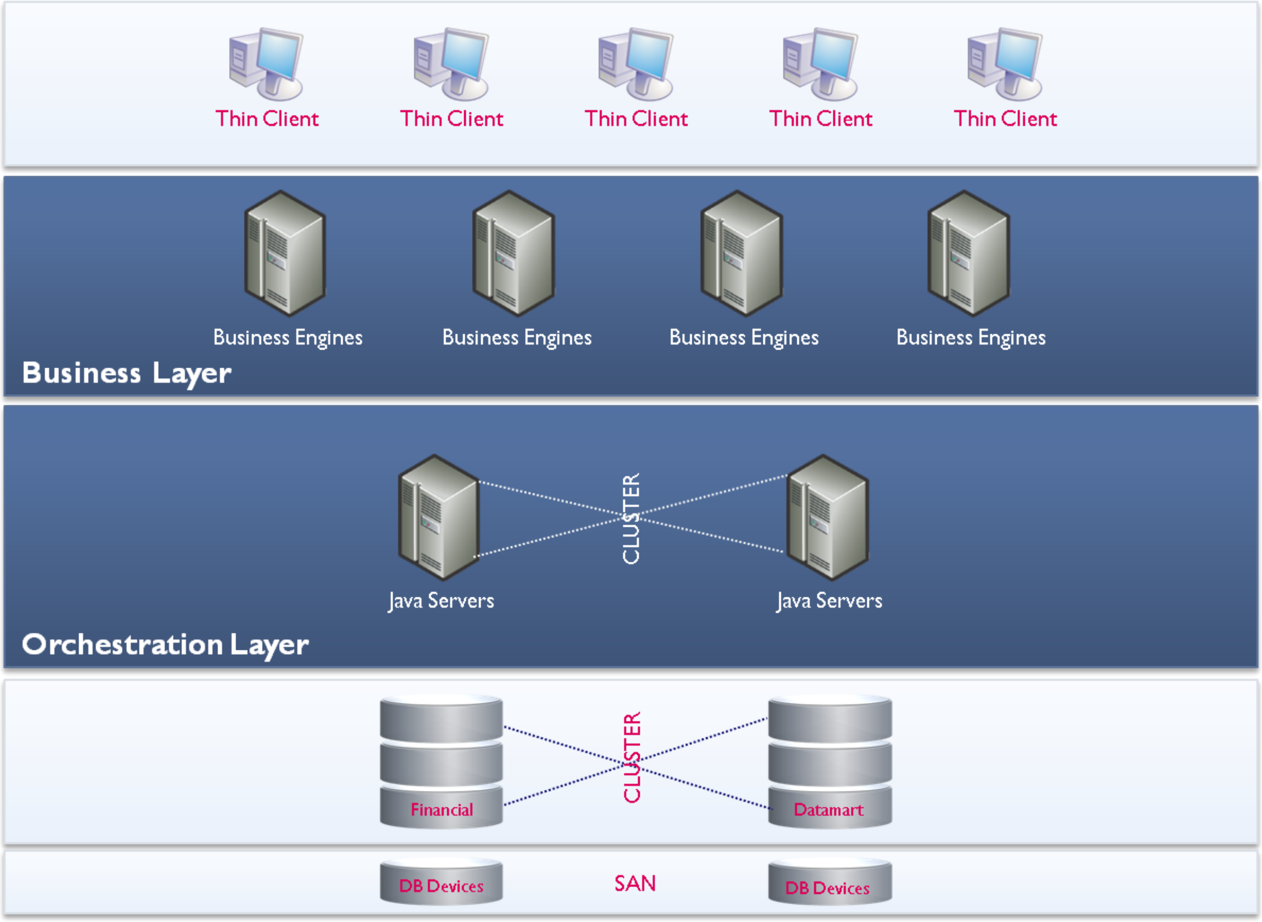
The calculation engines are restarted automatically or on-demand, based on the type of the engine. Thus, there are no specific actions required on this layer.

» Orchestration Layer

Murex recommends the usage of the logical naming for the hostnames of these machines. In case of a general failure of one of these machines, the Java servers can be immediately switched to the automatically spawned virtual machine. The Orchestration Layer can be restarted through custom scripts configured at the level of each machine, based on a certain dependency.

» RDBMS Tier

The same applies to the Database Tier. When a database server is down, a replacement machine is spawned automatically by Microsoft Azure. Custom scripts can be used to restart the database server.
It is worth noting that Oracle RAC is not supported by Microsoft Azure.



Any infrastructure level failure on the Orchestration Layer or on the RDBMS, requires a complete restart of the application.
If all the recommendations above are implemented, a system failure can be recovered within few minutes.

DISASTER RECOVERY & BACKUP STRATEGIES

BACKUP & RESTORE

There are two main tiers implied in the backup and restore mechanism. The application tier and the RDBMS tier. To backup the application:

- On the application tier, only the centralized application directory needs to be archived on weekly or daily basis.
- On the RDBMS tier, the database schemas need to be copied. This can be done either through:
  - An ordinary backup of the schema using standard RDBMS tools
  - Or, a data replication strategy such as: Oracle Data Guard, etc…

Azure provides multiple types of storage solutions:

- Locally redundant storage (within the same datacenter)
- Zone-redundant storage (asynchronous replication across datacenters within one or two regions)
- Geo-redundant storage (asynchronous replication to a secondary region that is hundreds of miles away)

To restore the application:

- Shut down the application.
- On the application tier, only the centralized application directory needs to be restored.
- On the RDBMS tier, the database schemas need to be reloaded.
- Backup and remove the logs folder, and restart the application.

DISASTER RECOVERY

Murex recommends provisioning the Disaster Recovery site on a different region. The virtual machines intended to run this environment are placed within the same Availability Zone. The MX.3 Disaster Recovery site can be equipped either by exactly the same infrastructure as the production environment, or a lighter infrastructure. Lighter infrastructure means less number of machines, cores, memory, therefore reduced performance, volume and activity.

In order to have a disaster recovery site lined up with the production site, two main actions are required:

- On the Application Tier: a weekly or daily refresh of the centralized application directory of the Disaster Recovery site.
- On the RDBMS Tier: a data replication strategy thanks to third party data replication methods such as: Oracle Data Guard, Log backup, etc…