



SAP on Azure Enablement

Monday, Oct 26, 2020

Cameron Gardiner
Ravi Gangampalli
APAC, Singapore

Module Two – Week Three

Day 1 – Monday, Oct 26, 2020

IMPT NOTICE:

- If you choose to participate in this session using Microsoft Teams, your name, email address, phone number, and/or title may be viewable by other session participants.
- **Please note that the training will not and cannot be recorded in alignment with Microsoft's policies**



SAP on Azure Partner Enablement

Module Two – Week Three

Day 1 – Migration & Architecture best practices



Cameron Gardiner

Principal Architect – SAP on Azure
Azure Engineering



Ravi Gangampalli

Cloud Solution Architect– SAP on Azure
One Commercial Partner

Check-in

We are happy to host you 😊

<https://aka.ms/apac-enablement-check-in>

<https://aka.ms/SAPAPAC-POE-FEEDBACK>

Feedback for previous session



Check-In Form



Feedback Form

Agenda

Key design principles for SAP workloads on Azure

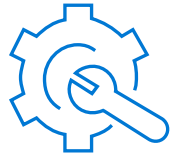
Migration Strategies for SAP on Azure

Q &A

Security



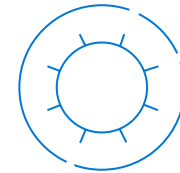
Performance



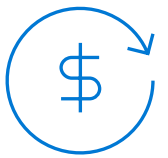
Availability



Efficiency

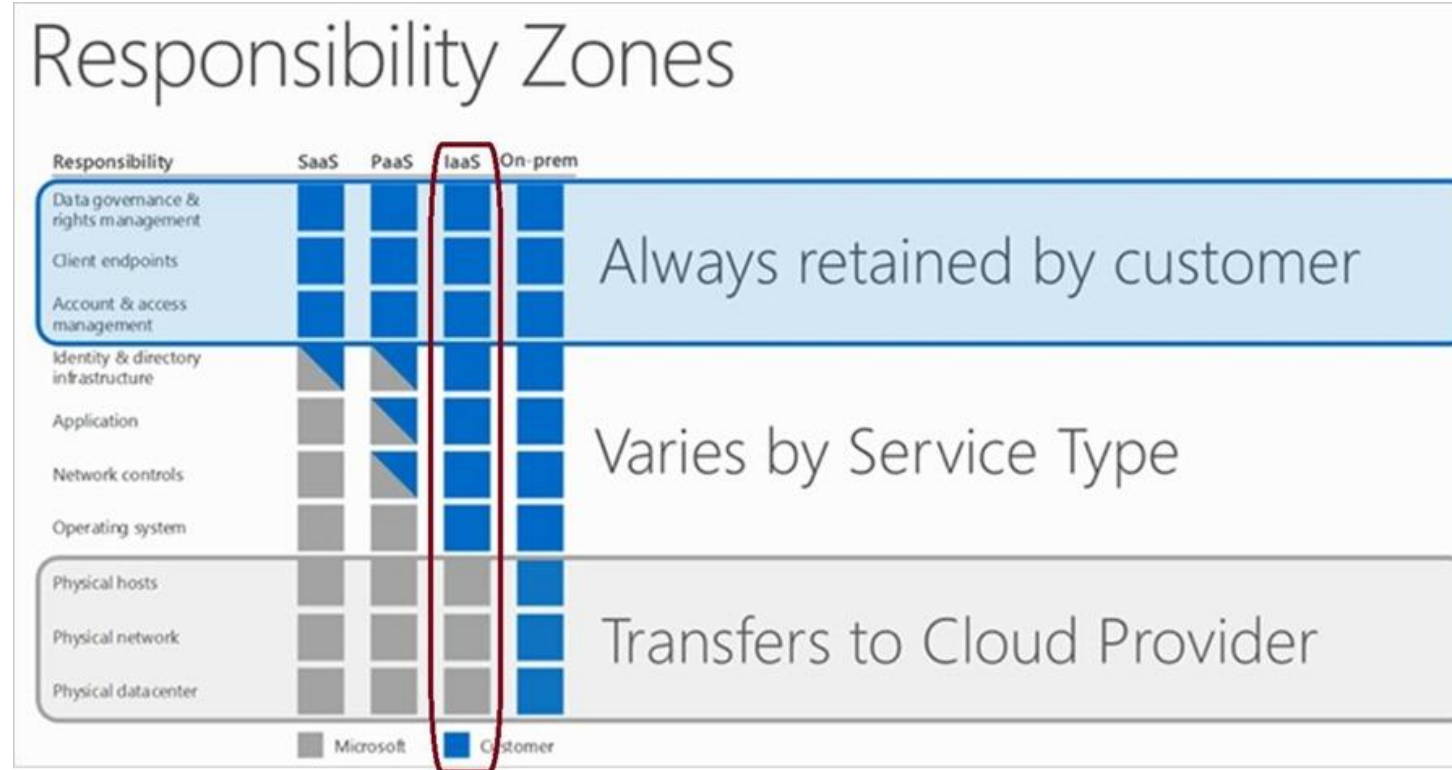


Cost
Optimised



Great SAP on Azure Architectures are built on the pillars of security, performance and scalability, availability and recoverability, and efficiency and operations.

Design for Security



Security is a shared responsibility

Key Security Considerations for SAP on Azure



**RESOURCE BASED ACCESS
CONTROL & RESOURCE
LOCKING**



**BUILD SECURE
AUTHENTICATION**



**PROTECT APPLICATION
AND DATA FROM
NETWORK
VULNERABILITIES**



**MAINTAIN DATA
INTEGRITY THROUGH
ENCRYPTION METHODS**



**HARDEN THE OPERATING
SYSTEMS**

Design for availability and recoverability

Protect Single Point of Failures

Leverage Azure SLA offerings

Leverage first party solutions

Leverage Operating System Clusters

Integrate Third-party Solutions

Architecting to Protect Single Points of Failure



1. Protect SAP's Single Points of Failures

SPOF's	HA Protection Mechanisms	DR Protection Mechanisms
Application Servers	Multiple App Servers + Availability Sets / Zones	Azure Site Replication (ASR)
ABAP SAP Central Services	SAP HA + Clustering + Availability Sets / Zones	Azure Site Replication (ASR)
Database Instance	Clustering + Availability Sets / Zones	DB Replication or Storage Replication

SLA'S, RPO, RTO dictate the architecture

- Design to cover the customers Service Level Agreement 99.9%, 99.95% or 99.99%?
- How fast to recover the landscape in a DR scenario and how much data can they afford to lose dictates the DR design.

Architecting to Protect Single Points of Failure



2. High Availability (HA)

Service Level Agreement	HA Protection Mechanisms
99.9%	Single VM's with Premium Storage
99.95%	SAP HA + Clustering + Availability Sets
99.99%	SAP HA+ Clustering + Availability Zones



3. Disaster Recovery (DR)

Recovery Point Objective	DR Protection Mechanisms
0 mins	Availability Zones with DB Replication Synchronous Mode
0>30 mins	Azure Region Pair with DB Replication Asynchronous
Recovery Time Objective	DR Protection Mechanisms
4 Hours	This is procedure based and vary depending on disaster recovery and business continuity plans in place and automation.

Architecting to Protect Single Points of Failure



Azure Site Recovery (ASR)

ASR is a protection mechanism for replicating the VM's from one location to another.

Couple of things to consider:

- Some limitations on VM's Types, OS versions, Regions Availability. All details: <https://docs.microsoft.com/en-us/azure/site-recovery/azure-to-azure-support-matrix>
- Suitable for App and ASCS servers. Not suitable for DB nodes. (Churn rate limitations)



Database Replication Options (DR)

There are multiple options for replicating the DB. Specifically:

Replication Type	Example
DBMS Replication	HANA System Replication or SQL Always On
Storage Replication	Available on SAP Hana Large Instances
Azure NetApp Files Replication	Cloud Sync Replication using ANF

Design for Performance

1

Design compute for performance

Quick Sizer or Reference Sizing

2

Design highly performant storage

IOPS and throughput requirements

3

Design network for performance

Proximity placement groups

4

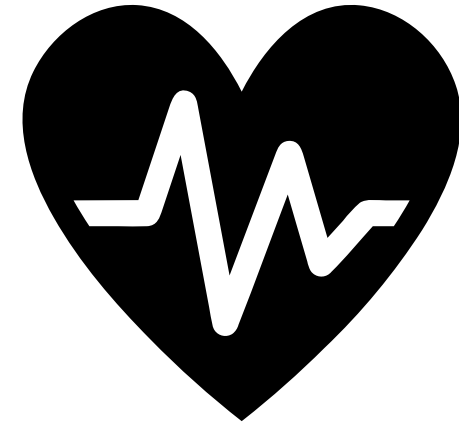
Design for scalability

E and M series virtual machines

5

Azure global regions at Hyperscale

Azure regions are growing



Design for Efficient Operations

1

Simplify the architecture

Do not create complex architectures

2

Optimise Dev, Test and Sandbox deployments

Use SAP Snooze, LaMa etc

3

Automate deployments

Use landing zones and ARM scripts

4

Leverage Azure monitor

Azure monitor for SAP Solutions

5

Leverage native Azure tools

Azure site recovery, Azure backup for HANA etc



Design for Cost Optimisation

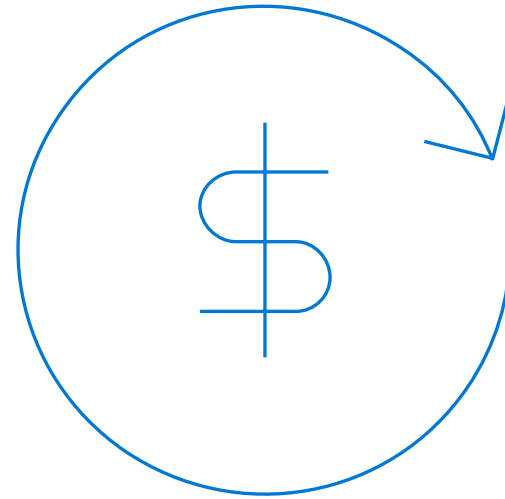
Start small, Do not oversize

Scale up as needed

Use Azure Reserved Instances

Use SAP Snooze Power app

Configure Cost Optimisation



Accelerate | Connect | Excite



Migration Methodologies

Homogenous migrations (low downtime)

- Backup/Restore
- Database log shipping
- HANA system replication

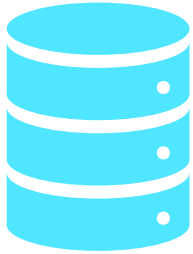
Heterogenous migrations

- Classical Migration via Software Provisioning Manager (SWPM)
- Database Migration Option (DMO) to HANA
 - Can facilitate SAP upgrade & DB migration to HANA (one-step-migration)

Whitepaper: <https://azure.microsoft.com/en-us/resources/migration-methodologies-for-sap-on-azure/>

Migrating SAP Workloads:

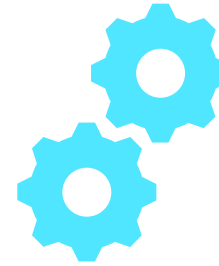
Classic vs. SAP Database Migration Option (DMO)



Classical Migration

SWPM is used exclusively for database migrations. SWPM exports data from a source system and imports it to AnyDB target system. **This method uses a file-based approach.**

Classical Migration uses a heterogeneous system copy approach (**a two-step migration**) with the first step being that of a **migration followed by** a second step facilitating an **SAP upgrade**.

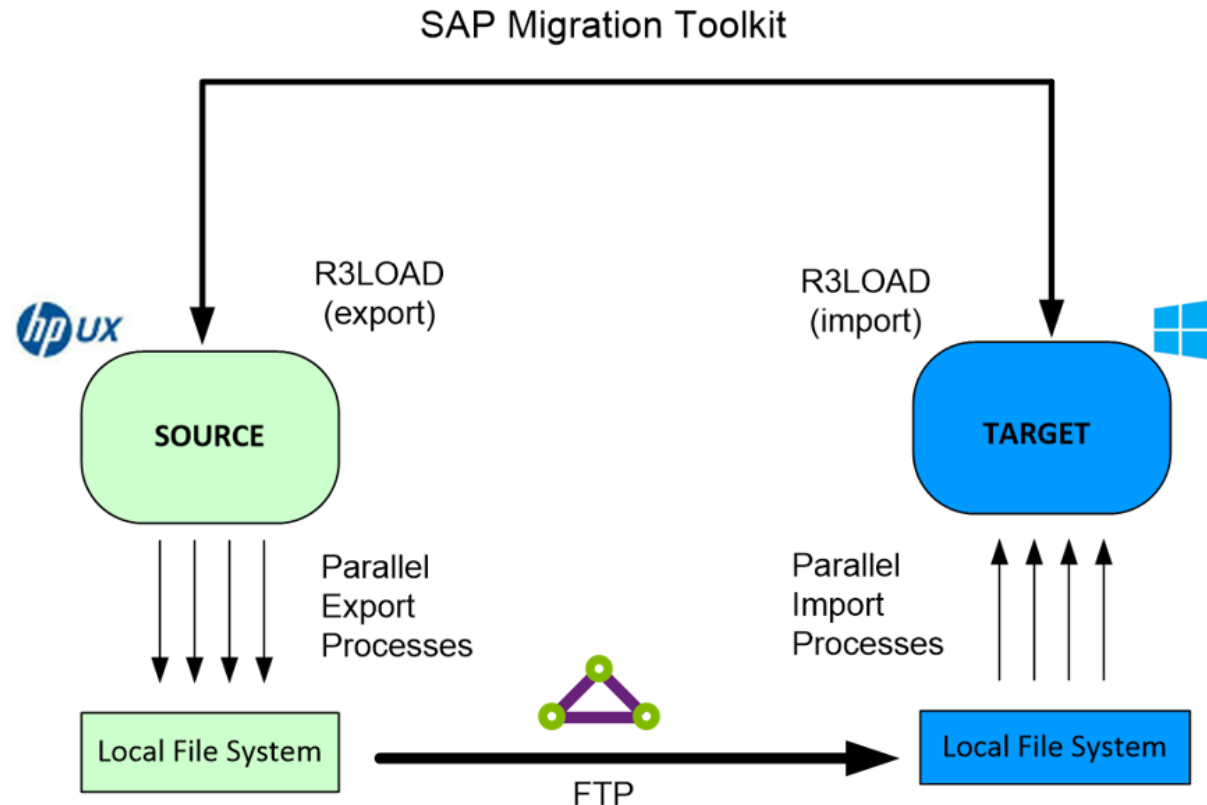


SAP Database Migration Option (DMO)

DMO facilitates **both an SAP upgrade and a database migration to the SAP HANA database via one tool**. Both steps are handled at once, so DMO is referred to as a **one-step migration**.

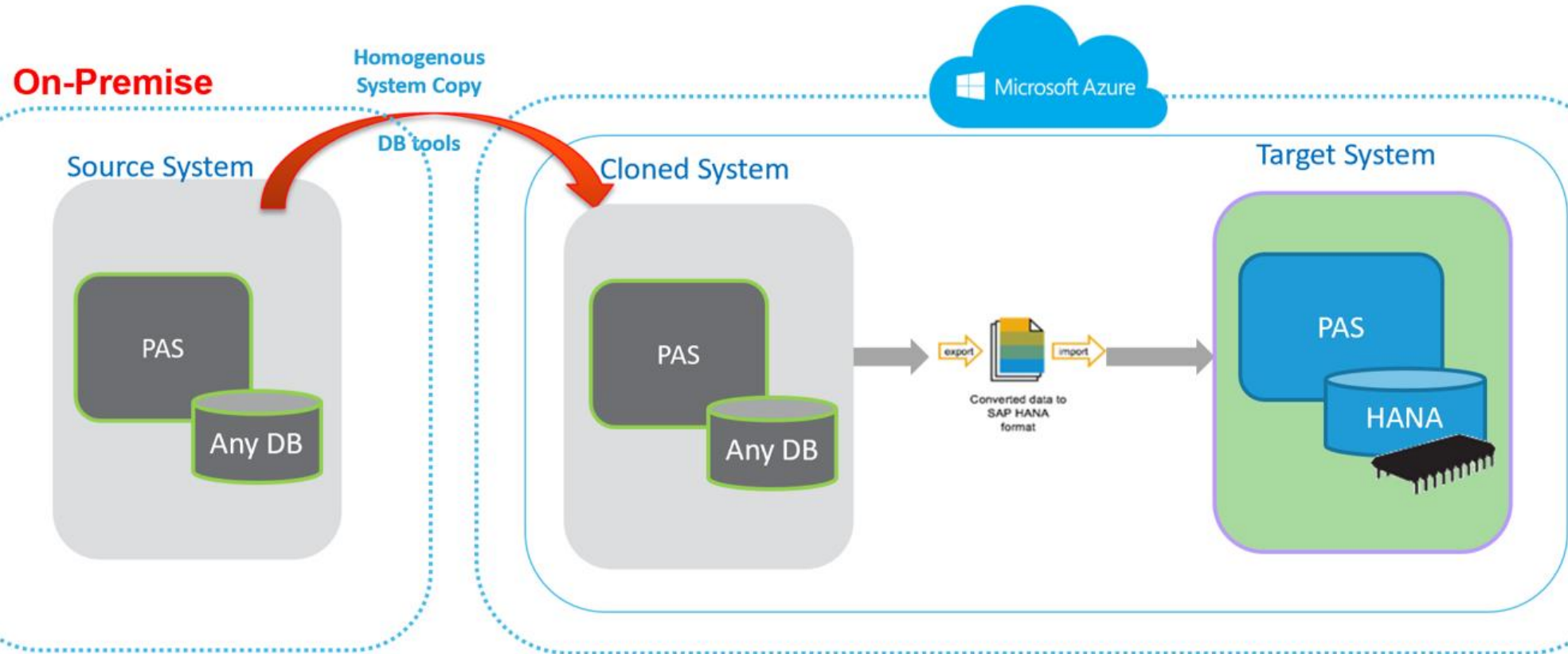
White paper: <https://azure.microsoft.com/en-gb/resources/migrating-sap-applications-to-azure/>

Classical – OSDB Migration



1. Table Splitting of largest tables
2. Optimize order of tables exported (order_by)
3. DBMS Tuning for R3LOAD (export/import)
4. Dedicated SSD for TEMP Tablespace (source cluster table sorting, target index build)
5. Export/Import packages land on Premium SSD (striped)
6. Parallelization of INDEX creation.
7. Target tablespaces created in advance with Transparent Data Encryption (TDE)
8. Azure disk cache (db_data) = NONE (push max throughput i.e. ~960MB/s, DS15_v2)

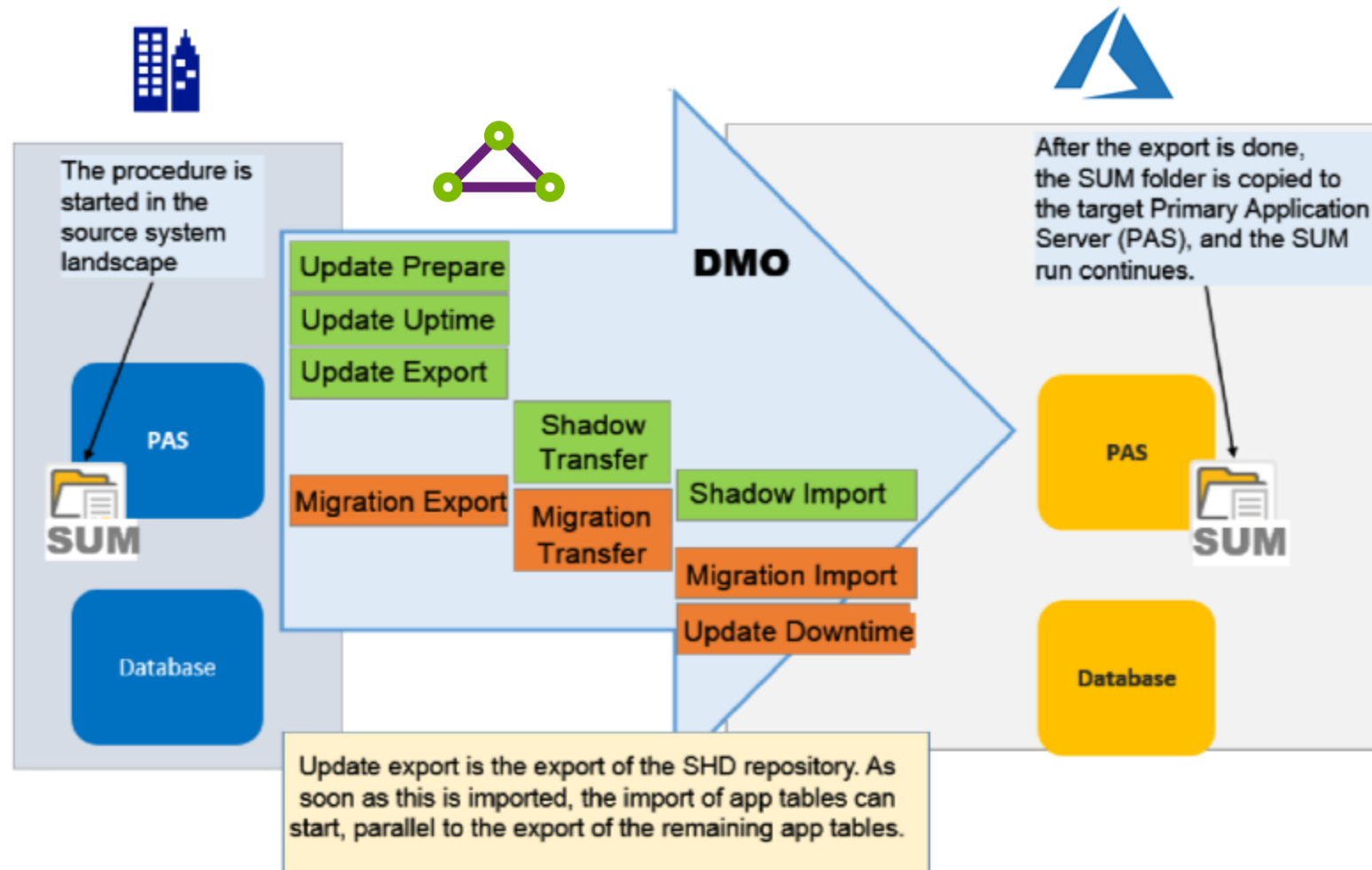
Two Step Migration: Lift and Shift followed by DMO



One Step: Downtime Minimized Migration to Azure

DMO with
System Move
– change PAS
host during DMO

LINK



Source SAP: DMO/SUM 1.0 SP21 (source PAS must be UNIX/LINUX based)

Migration Performance Optimization

Optimize Export, File Transfer & Import

Migration:

- [Parallel Export/Import](#) with Classical Migration or DMO/System Move (dmotocloud.sh - RSYNC)

On-Premise:

- Source DB - [Archive/Delete](#) data in advance of migration to reduce database footprint
- Source DB – index rebuild, DB stats, [DB Parameters \(R3Load\)](#), etc
- Deploy [Dedicated Migration Server](#) (PAS, AAS) with extensive compute capacity
- Implement [SSD on source DB and Migration Server](#) (PAS, AAS)

Network/Connectivity:

- [ExpressRoute recommended](#) with maximum bandwidth (up to 10 Gbit/sec)

Azure:

- [Select VM with horse-power](#) to drive the import e.g. M128ms (128 vCPU, 4TB, D 2GBps, N 30Gbit/s)
- Post migration scale-back VM to 'run' size

Post Migration Steps

- SAP
 - SM21, ST22, ST04, RZ03, SMLG, RZ12, SM61, SM59
- DB
 - Check database logs for errors
 - Check to space issues
 - Check DB memory usage
- Infrastructure
 - Check for network latencies/throughput (niping, qperf)
 - Monitor CPU and Memory Usage (top, iostat, etc)
- Backup the System
- HA Testing (failover/fallback)
- Smoke Testing – Functional, Integration and Acceptance Testing
- BTCTNRS2 – Reschedule Batch jobs
- Establish + Monitor Data Sync to DR location
- Enable Monitoring – Infrastructure and Application
- Update Service marketplace definitions (SAP Router, hostnames, IPs)

Migration - Lessons Learned

- Execute multiple migration **mock test cycles** – iterate and refine runbook.
- **Performance Testing** (compare DIAG, BTC runtimes vs on-prem)
- **Avoid NVA between APP & DB layer** & between Pacemaker Cluster and SBD
- Individual SAP Systems need to **move as one (xSCS, DB, APP,)**
- Pay attention to **latency sensitive interfaces** between SAP Systems and chain applications – define 'move groups' to move as one.
- **Archive** and compress databases.
- **Test HA/DR** capabilities
- **System Rationalization** (decommission before migration)
- Decide on **cloud region strategy**, not all services are available in all regions

Azure Site Recovery

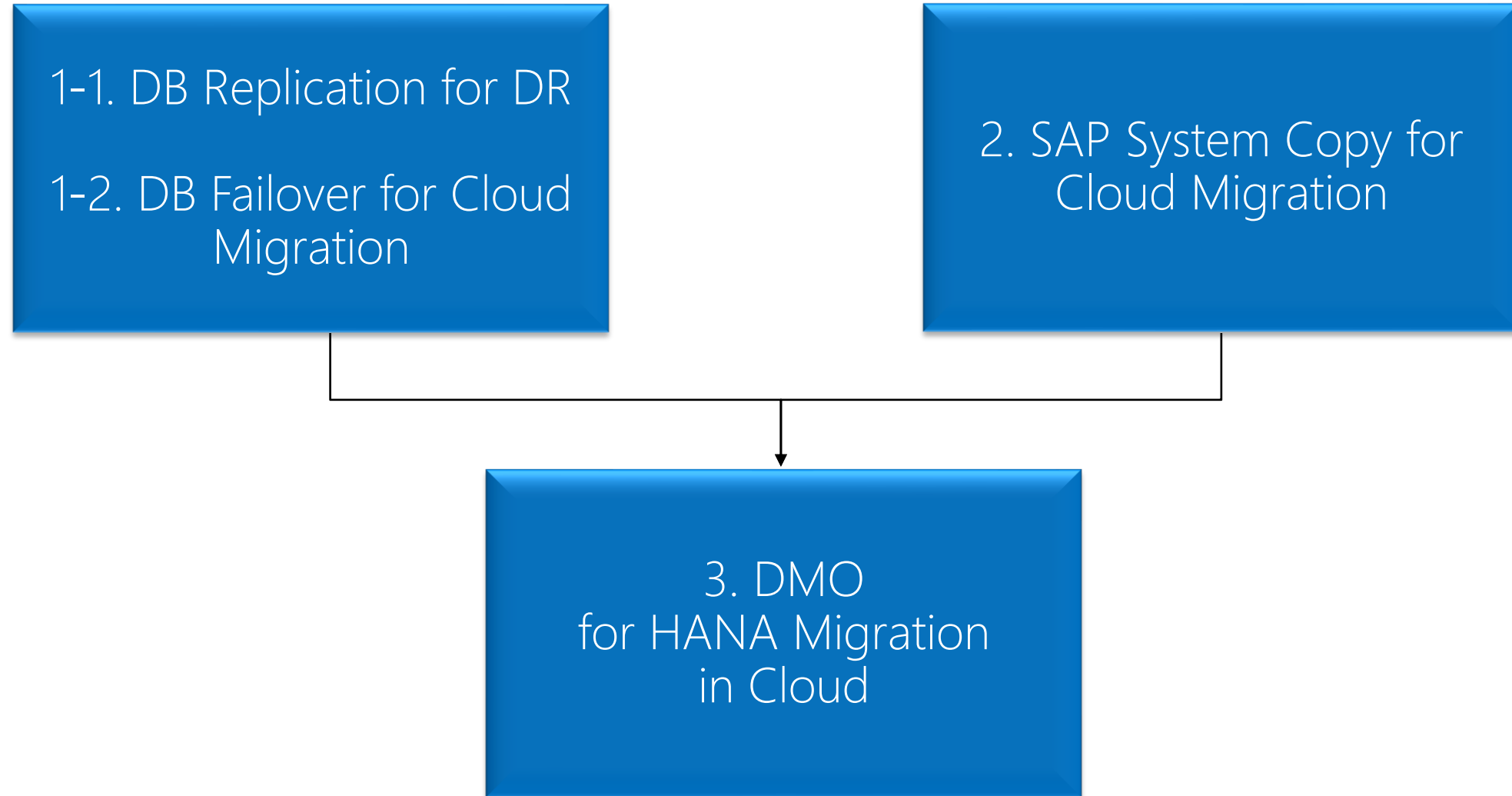
- Configure replication of Azure VMs
- Configure networking:
 - Assign static IP addresses to VM replicas
- Create a recovery plan:
 - Supports **sequencing of various tiers** in a multi-tier application during a failover.
- Add virtual machines to failover groups:
 - **VMs in the same failover group fail over together**
- Add scripts to the recovery plan:
 - Implement **pre-action and post-action scripts** by using **Azure Automation**
- Run a test failover
- Run planned failover

Cloud Migration option

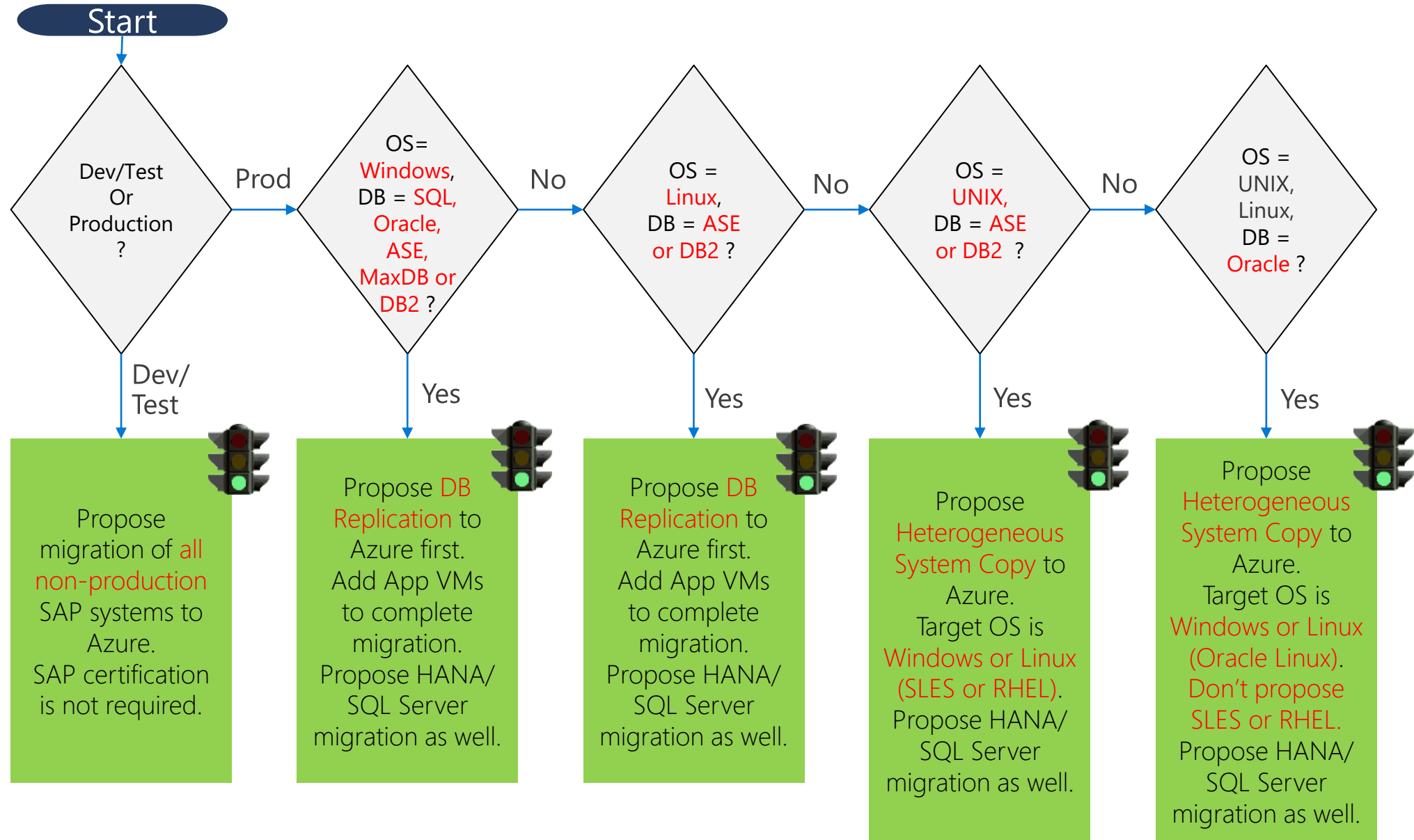
Accelerate | Connect | Excite



Typical Cloud/HANA Migration Methodologies



Qualifying SAP on Azure Migration Methodology



Q&A

Reach out to the team

sap-on-azure-pe-apac@microsoft.com



SAP on Azure Enablement

Next Session – Building High Availability Systems for SAP on Azure

Tuesday, Oct 27, 2020, 10am SGT

Reach out to the team

sap-on-azure-pe-apac@microsoft.com

