Dynatrace Training - Day 1 Notes

# 1. Dynatrace Overview & Value Proposition

What is Dynatrace?  
Dynatrace is a comprehensive full-stack observability platform that provides monitoring, analytics, and automation for applications, infrastructure, and user experiences. It is powered by artificial intelligence (Davis AI) and automation capabilities that simplify cloud complexity and optimize performance.

Capabilities Overview:  
- APM (Application Performance Monitoring): Deep code-level visibility for Java, .NET, PHP, Node.js, etc.  
- Infrastructure Monitoring: Tracks servers, VMs, containers, and cloud environments.  
- RUM (Real User Monitoring): Tracks actual user behavior, performance metrics, and interactions.  
- Synthetic Monitoring: Simulates user journeys and API availability.  
- Log Monitoring: Centralized log ingestion and analytics.  
- Davis AI: Automatic root cause detection and anomaly analysis.

Monitoring Modern Apps:  
Dynatrace excels at monitoring cloud-native architectures, microservices, Kubernetes clusters, containerized workloads, and multi-cloud environments. It ensures observability in dynamic environments using OneAgent and AI-driven insights.

# 2. Dynatrace Architecture

Key Architectural Components:  
- OneAgent: Installed on each host; collects metrics, traces, logs, and topology.  
- ActiveGate: Acts as a communication proxy between agents and the cluster; used for DMZ, cloud integrations, and remote traffic.  
- Cluster Node (SaaS/Managed): Processes data and hosts Dynatrace UI/API.

Communication Flow:  
OneAgent -> ActiveGate (optional) -> Cluster -> Dynatrace UI -> Davis AI Engine

Deployment Models:  
- SaaS: Managed by Dynatrace, hosted in the cloud.  
- Managed: Self-hosted; requires on-prem or private cloud infrastructure.

Cluster Architecture:  
- Cluster nodes operate in high-availability mode.  
- Scale-out horizontally by adding nodes.  
- Davis AI and Smartscape modules are distributed across nodes.

Sizing and Availability:  
- Cluster sizing is based on host units, environment size, and data volume.  
- Managed deployments should plan for backup, failover, and data redundancy.

# 3. Dynatrace Deployment on Azure

Azure-specific Considerations:  
- Enable Azure Monitor integration for resource metrics.  
- Use Network Zones to segregate monitoring domains.

Integration with Azure Monitor and AKS:  
- Connect Azure subscriptions via Dynatrace Hub.  
- Enable AKS integration using Helm and DaemonSets.  
- Collect container, node, and workload metrics from AKS.

Cloud-native Monitoring Practices:  
- Use tagging to auto-detect resource groups.  
- Enable auto-discovery for scalable, ephemeral environments.  
- Leverage OneAgent on Azure VMs and App Services.

# 4. Cluster Management Console (CMC)

Overview:  
The CMC is used in Managed deployments to manage and monitor cluster nodes and configurations.

Capabilities:  
- Add/remove cluster nodes.  
- Monitor cluster health and performance.  
- Configure and view licensing usage.  
- Manage updates, maintenance windows, and troubleshooting.

# 5. ActiveGates

Definition:  
An ActiveGate is a software component that routes traffic, enables remote monitoring, and supports integrations.

Use Cases:  
- Secure routing in DMZ environments.  
- Cloud integrations with Azure, AWS, GCP.  
- Synthetic Monitoring execution nodes.  
- Monitoring remote or untrusted zones.

Deployment Scenarios:  
- Installed on Linux/Windows VMs.  
- Can be used in private, public, or hybrid cloud setups.  
- Configurable to support specific roles like synthetic, routing, or extensions.

# 6. OneAgents

Capabilities:  
- Deep instrumentation of apps, services, OS, containers.  
- Auto-detects running processes and technologies.  
- Sends data to Dynatrace cluster in near real-time.

Supported Technologies:  
- Java, .NET, Node.js, Python, PHP, Go, Kubernetes, Docker, etc.  
- Web servers, messaging systems, databases.

Installation & Updates:  
- Install using GUI, CLI, or scripts.  
- Self-updating mechanism configurable via UI.  
- Easily deployed using automation tools (Ansible, Terraform, Helm).

# 7. User Management – Managed

Access via CMC:  
- Add users manually or import from LDAP/SAML.  
- Configure roles and permissions per group or zone.

RBAC:  
- Define roles: Viewer, Admin, Environment Admin, etc.  
- Assign access to specific apps, dashboards, or zones.

Federation/SSO Options:  
- SAML-based authentication.  
- Integration with enterprise IAM solutions.  
- OAuth and OpenID Connect support.

# 8. Organizing Your Environment

Entity Hierarchy:  
- Hosts -> Processes -> Services -> Transactions  
- Visualized in Smartscape topology.

Tagging:  
- Auto-tags based on metadata (cloud tags, service name).  
- Manual tags for fine-grained control.

Management Zones:  
- Logical partitioning of the environment.  
- Helps segregate access (e.g., Dev vs. Prod).  
- Enables dashboard scoping, alert rules, and filtering.