How Microsoft Built MySQL, PostgreSQL and MariaDB for the Cloud

Santa Clara, California | April 23th – 25th, 2018
Azure Data Service Architecture

• Share Cluster with SQL DB

• Cluster is decomposed into Azure Service Fabric applications (10+ applications)

• All applications and all tenants are individually deployable

• DB engine Instances are “services” managed by Azure Service Fabric
Azure Relational Database Services Platform

**SQL Database**
- Azure Resource Manager APIs, Client Tools, Portal integration
- Orchestration of Management Workflows
- Connection Proxy and Connection Redirection
- Resource Governance and Resource Isolation per Server/Database
- Database Jobs
- Continuous Delivery through Deployment Automation
- Smart Monitoring
- Auto-mitigation of LiveSite incidents
- Cross-region and in-region data migration
- A/B Testing

**SQL DW**
- SQL Server 2017 + PDW
- Backup Manager & Backups retention, PITR and Geo-Restore
- Local Replication for HA
- Workload Insights
- Data Security & Compliance
- Proactive Analytics and Alerting
- A/B Testing

**MySQL/MariaDB**
- MySQL/MariaDB Server
- Automated Rollback
- Rolling Upgrades
- Placement Services
- Load balancing
- Self-healing
- Failure Detection & Failover

**PostgreSQL**
- PostgreSQL Server
- Automated Rollback
- Continuous Delivery through Deployment Automation
- Data Security & Compliance
- Proactive Analytics and Alerting
- A/B Testing

**Database Services Platform**
- Local Replication for HA
- Active Geo-Replication
- Database Jobs
- Continuous Delivery through Deployment Automation
- SMART Monitoring
- Proactive Analytics and Alerting
- A/B Testing

**Service Fabric**
- High Availability
- Stateful services
- Fast startup & shutdown
- Low Latency Cluster messaging
- Location Services & Routing
- Health Monitoring
- Load balancing
- Self-healing
- Failure Detection & Failover

**Smart Monitoring**
- Placement Services
- Load balancing
- Self-healing
- Failure Detection & Failover

**A/B Testing**
- Continuous Delivery through Deployment Automation
- Smart Monitoring
- Proactive Analytics and Alerting
- A/B Testing

**Azure Storage**
- Azure Storage
- Global Azure with 38 Regions

**Azure Networking**
- Azure Networking

**Azure Compute**
- Azure Compute

**Azure Monitoring**
- Azure Monitoring
Microsoft Azure Service Fabric
A platform for reliable, hyperscale, microservice-based applications

- High Availability services
- Fast startup & shutdown
- Low Latency Cluster messaging
- Location Services & Routing
- Health Monitoring
- Container Orchestration & lifecycle management
- Failure Detection & Failover

- Hyper-Scale
- High Density
- Automated Rollback
- Rolling Upgrades
- Resizing Service
- Placement Services
- Load balancing
- Self-healing

- Management
- Communication
- Reliability
- Hosting Subsystem
- Testability
Cluster: A federation of machines

A set of machines that Service Fabric stitches together to form a cluster

One cluster can scale to 1000+ machines
Relational data services – Control Plane

One cluster per region managed by service fabric

Provides front-end and cluster control services.
Relational data services – Data Plane

One to many clusters per region managed by service fabric

Each node has application services (MySQL server) and platform services
Multi-tenancy

- Multi-tenancy is really hard
  - Noisy neighbors; accidental or intentional abuse
- Different levels of multi-tenancy have different tradeoffs in cost, capacity and density
  - More sharing leads to greater efficiencies but adds more points of contention
- Expectations on performance predictability need to be managed via min guarantee and max caps across different hardware SKUs
Our Solution

- Running a strip down version of latest Windows in a security container (SQLPAL)
  - Strong Security Isolation
  - Strong Resource Isolation
  - Less memory footprint (compare to a Full OS)
  - Less attack surface (Lock down to bare minimum for engine)
  - Leverage Microsoft SQL Server schedulers and memory management

- Resource Governance combined with Native Windows and SQLPAL
  - CPU
  - Memory
  - Disk
  - Network
SQL Platform Abstraction Layer (SQLPAL)

- Windows Host Extension has a driver for creating the Pico process and a monitor process (user mode) that implements non-perf related ABIs.
- ABI calls are handled by the driver and are either handled directly (Like File IO) or are marshalled to the monitor process for handling (Like File Open)
Decoupled Compute and Storage

• Remote Storage built on top of commodity hardware
• Different optimizations for I/O Path of Log and Data files
  • Log files require low latency write, sequence read when crash recovery
  • Data files require high throughput, random read/write
• Snapshot based backup
  • Never possible for huge amount of data through other ways (TB+)
  • Snapshot Support
  • PITR support
Security Enhancement

• Network Security
  • VNET
  • Firewall Support
  • Both Inbound and outbound lock down

• Port Sharing Service (One per Node)
  • One port listen for each server
  • Duplicate the socket and SSL security context to the real instance

• Encrypt-At-Rest

• Threat Detection
Introducing gh-ost: triggerless, painless, trusted online schema migrations
11:20 → 12:10, Matterhorn 2
Thank You!