

@MaxtaInc

Carolyn Crandall  
Agenda

## **Company Overview**

Yoram Novick, CEO

Founded 2009

Malta maximises the promise of hyper-convergence

- choice
- simplicity
- scalability
- cost

SDS: MxSP

Hyper-converged appliances: MaxDeploy

Funded by Andreessen Horowitz, Intel Capital, Tenaya Capital

Align the storage construct with the abstraction layer

Dramatically simplified mgmt  
went class vm-level data services  
eliminate storage arrays and storage networking  
leverage flash/disk + capacity optimization

Solutions

MaxDeploy

Hyper-Converged Appliance

- preconfigured servers and Maxta software
- 15 minutes from zero to VMs

Peace-of-mind

- interoperability
- ease of ordering and deployment
- predictability of performance

MxSP

Software-Defined Storage

- Software only
- perpetual and subscription

Flexibility

- DIY - your choice in hardware
- works with existing infrastructure - no forklift upgrades

Full-featured

- enterprise class data services
- support latest and greatest technologies

Customize configuration to User

- major server vendors supported
- proposed configuration validated

- fulfilled by partners

Use cases

- Business Apps
- ROBO
- VDI
- DR Site
- Test and Dev
- MSPs

## **Maxta Solutions Overview**

Kiran Sreenivasamurthy, Director of Product Management

Architecture Overview

[diagram]

Data Services

- data integrity
- data protection / snapshots / clones
- high availability
- capacity optimisation (thin / dedupe / compression)
- linear scalability

Simplicity

- VM-centric
- tight integration with orchestration software / tools
- policy based management

Multi-hypervisor support (VMware, KVM, OpenStack integration)

Value Proposition

Maximise choice - any server, hypervisor, storage, workload

Maximise IT simplicity - manage VMs, not storage

Maximise Cost Savings - standard components and capacity optimisation

Provide high levels of data resiliency, availability and protection

## **Maxta Design Principles**

Christian Czeatke - Lead Engineer

Raghu Shastry, Lead Engineer

Design Points

Scale out, shared nothing

leverage industry standard hardware

leverage heterogeneity of vendors and components

enable server / storage convergence

- support for DAS (internal drives / JBOD)

Purpose built for virtualised environments - hypervisor agnostic

enterprise class storage services

independent scalability of both compute and storage

Configuration Details

one maxta instance per server that is part of the cluster

Maxta stack runs inside a VM (linux based on ESXi), Bare metal on KVM

Maxta stack runs in user space

each maxta instance provides an entry point to the global namespace / filesystem for the server it is running on either NFS or VFS layer

maxta instance best practice: 12GB (VM Factor), 4 cores

- improved performance with increased memory and CPU allocation

Maxta Internode configuration

- shared or dedicated 10GigE or

- dedicated 1GigE

Server attached Storage

- leverage pass through RDM on ESXi, minimises storage controller issues

- direct access to block devices on KVM

HDD - leverage low RPM, High capacity drives

- write optimised layout

Flash Best Practice

- 10% of HDD capacity in flash

- eMLC SSD or PCIe/NVMe flash

- spread out across 2 devices per node

Key Components

- Clustering / cluster membership

- unified namespace service

- distributed file system

- management and user interface

Clustering

- cluster membership / quorum membership

- resource discovery

- recommended to keep it contained within vSphere cluster

Unified Namespace Service

- strong consistency: single system image, cluster-wide ordered updates

- global namespace replicated on several nodes, accessible from every node

- replace a namespace bearing node with a non-namespace bearing node dynamically

Maxta Distributed File System

I/O

Writes are synchronously replicated to multiple nodes, acknowledged only after writing to all replicas

each node persists writes to an intent log backed by flash

data is striped / mirrored across multiple drives / nodes

reads: augmented by in-memory caching and host caching

flash accelerated metadata

Data Layout

Cluster-wide capacity balancing

Favours placement of new data on new / under-utilised disks / nodes

periodic rebalancing across disks / nodes

proactive data relocation

Policies

Policies based on VMs and / or virtual disk

- number of replicas, affinity, metro cluster
  - page size
  - read cache, rebuild priority
- Policies can be changed dynamically, after VM creation
- Snapshot policies for periodic check pointing of VMs
- first line of data protection, rapid recovery

Data Services

Time / performance / capacity efficient snapshots and clone

- zero-copy read-only snapshots
- zero-copy read-write clones
- hierarchy of snapshots and clones
- sub-second snapshot / clone creation
- no data movement when deleting snapshots
- number of snapshots / clones does not affect performance

Data Integrity

Data is check summed on media (Flash / HDD) and in transit

Ability to detect UBER and misdirected writes

ability to recover from checksum mismatches by obtaining copy from another replica

Fault Mgmt

Transient node failures

- incremental data resync

Disk Failure

- “treat cluster as spare”, rebuild onto one or more other disks / nodes
- avoid single choke point during rebuild
- no dedicated spares

Flash Media Failure

- intent log, vs. read cache vs metadata storage
- flash media failure does not require full node rebuild

Node Failure

- treat cluster as spare

Network Failure

- leverage NIC learning

Software Failures

- “Never give up”
- NFS Server IP Failover (ESXi)

Management and User Interface

Management Server is stateless

Well documented REST APIs exposed by mgmt server

Proactive Monitoring / Alerting for HW, Space utilisation, critical events

- on-premises
- cloud-based

HTML5-based UI

Upgrade and Patching

Non disruptively upgrade Maxta Software

- minor and major versions

Rolling Upgrade of Maxta Software on each server  
ESXi and Maxta can be upgraded in a rolling manner

**Demo**