

A photograph of a server rack with a black metal mesh front. A red rectangular highlight is drawn around a control panel on the right side of the rack. The panel contains several buttons and indicator lights. The text 'Hewlett Packard Enterprise' is overlaid on the left side of the image.

**Hewlett Packard**  
Enterprise

# DMFUG 2018

DMF7 Architectural Overview and Design  
Introduction

Zsolt Ferenczy

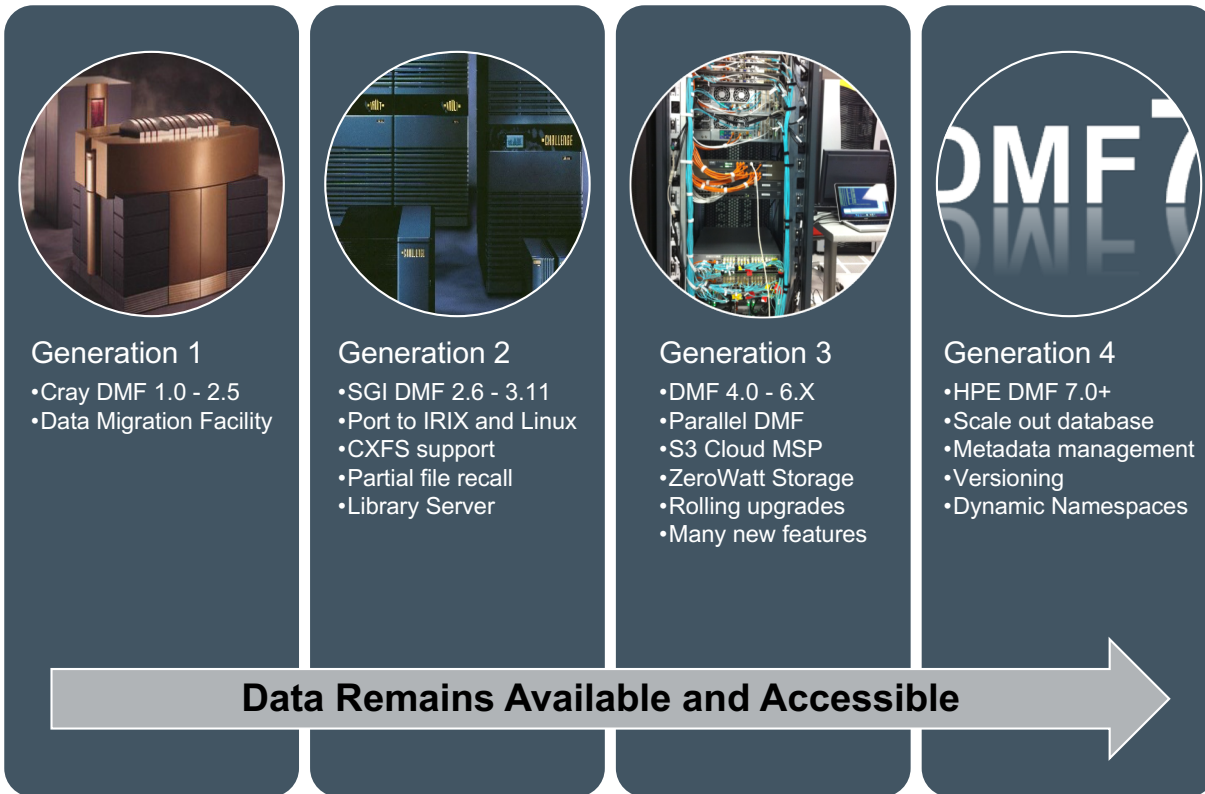
# Confidentiality Notice

- **The information contained in this presentation** is proprietary to Hewlett Packard Enterprise (HPE) Company and is offered in confidence, subject to the terms and conditions of a Confidential Disclosure Agreement
- **HPE makes no warranties regarding the accuracy of this information.** This document contains forward looking statements regarding future operations, product development, product capabilities and availability dates. This information is subject to substantial uncertainties and is subject to change at any time without prior notification. Statements contained in this document concerning these matters only reflect Hewlett-Packard Enterprise's predictions and / or expectations as of the date of this document and actual results and future plans of Hewlett-Packard Enterprise may differ significantly as a result of, among other things, changes in product strategy resulting from technological, internal corporate, market and other changes. This is not a commitment to deliver any material, code or functionality and should not be relied upon in making purchasing decisions.





# Data Management Framework | DMF Evolution



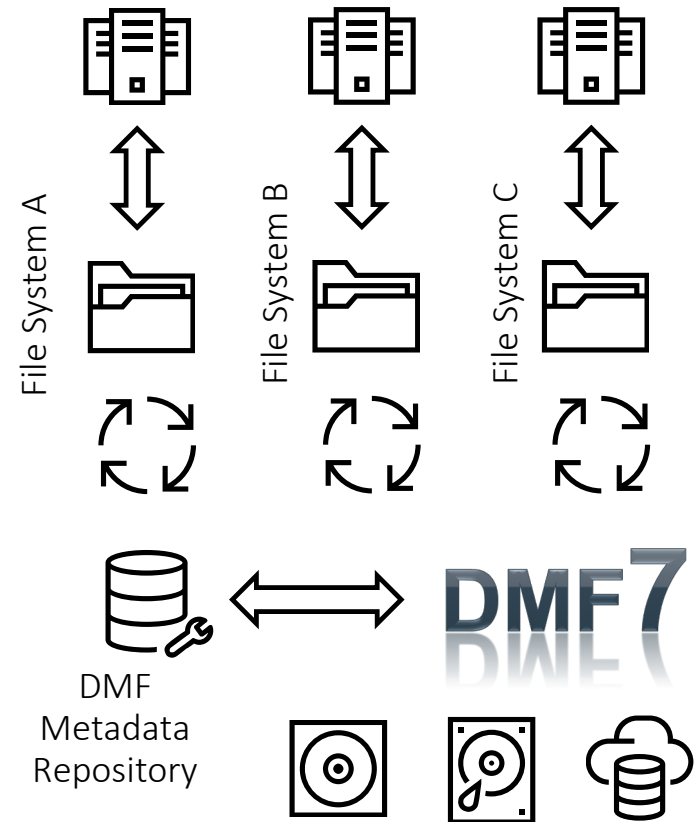
---

## Data Management Framework | **DMF 7** Introduction and Summary

- DMF7 is an ambitious Data Management Framework with many advanced capabilities
- 2 years in development, we are now close to customer release
- DMF7 development has been challenging, but the team never wavered from chosen architecture and still believes in it
- Would NOT be possible without support from management. This generally reflects the value that HPE places on this project
- While innovating – we care a lot about current DMF6 customers, and will continue to support them in their DMF6 days and in transition

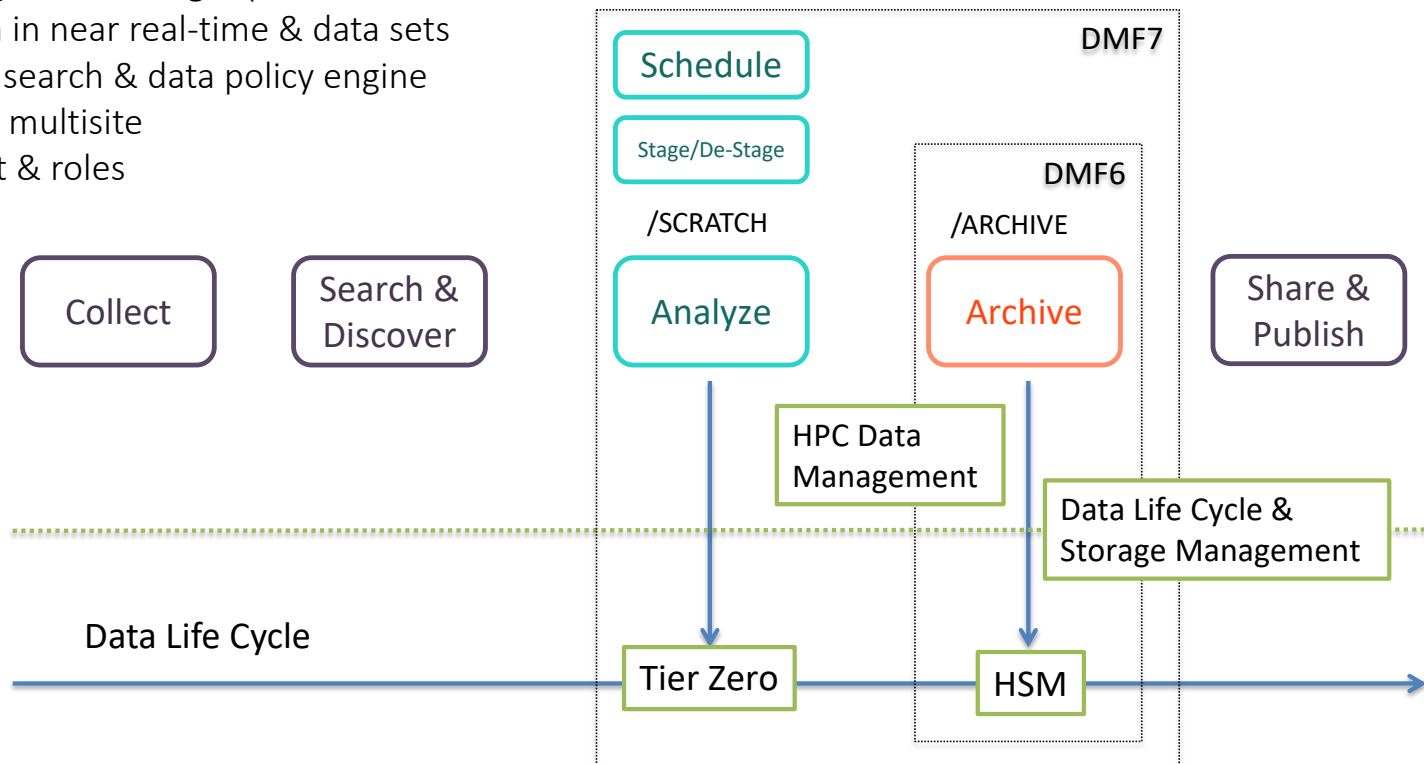
# Data Management Framework | DMF 7 Concepts & Vision

- 1 Integrate with High Performance Flash Filesystem
- 2 Offload Metadata Management to a Scalable Framework
- 3 Capture Filesystem Events in Real-time Changelog to avoid scans
- 4 Provision Namespaces On Demand by HPC Scheduler in addition to Static Filesystems
- 5 Optimize Data Transfer by Chunking and Parallel Data Movers
- 6 Keep Transparency. Replace Traditional Backup



# Data Management Framework | DMF 7 Accelerate HPC Workloads

- Integration with Tier Zero
- Up-tiering, staging & offloading capabilities
- Scalable metadata in near real-time & data sets
- Flexible metadata search & data policy engine
- Data replication & multisite
- User management & roles



---

## Data Management Framework | **DMF 7** Conceptual Definitions

- DMF7 is all of these:
  - HSM System
  - Data Management Hub (HPC workflow aware)
  - Posix-to-Object Bridge (versioned metadata map)
  - Data Transfer Engine (optimized for specific backends)
  - Data Policy Engine (space management or workflow driven)

---

## Data Management Framework| **DMF7** A New Architecture

### **Replace the DMF database**

- Use a hyper-scalable DB engine capable of handling 100s of billions of objects
- Bring the namespace into the DMF database

### **Move from a scan based model to an event and DB driven model**

- Listen to event stream from the managed filesystem
- Policy decisions made in the database not in the filesystem
- Make DMF independent from front end filesystem
- Unified DMF policy engine for all filesystem types

### **Move to a metadata centric data model**

- DMF becomes the authoritative metadata repository for both migrated and non-migrated files
- Enable searchable and user extendable metadata from POSIX
- Enable versioning
- Enable dormant data with no POSIX representation



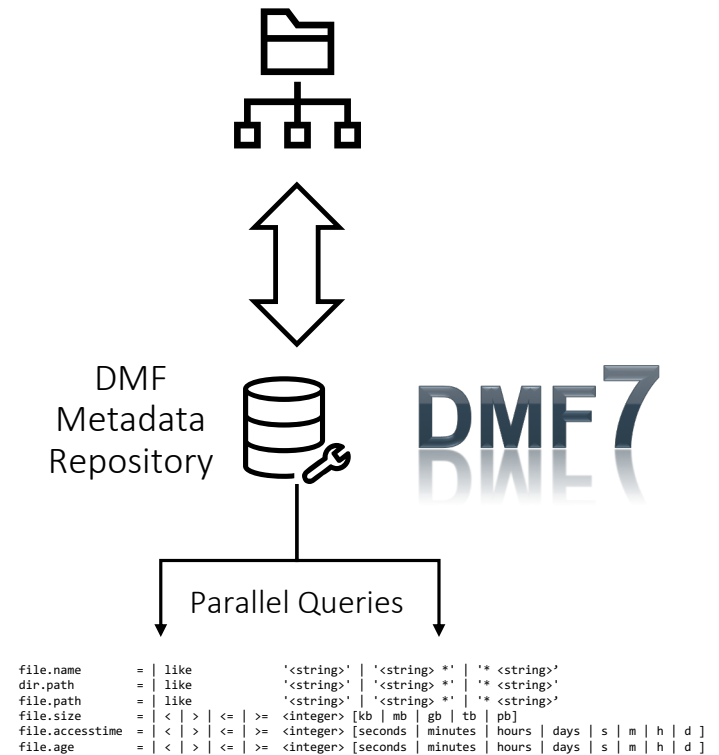
# Data Management | DMF 7 Global Namespace Reflection

## Key Takeaways

### Global Namespace Reflection

DMF 7 captures file-system metadata changes via the changelog stream and stores them as a set of tables in a highly scalable database.

- **Metadata Protection:** DMF migration preserves the namespace alongside the data.
- **Parallel Metadata Queries:** Using big data tools, such as Apache Spark, enables parallel metadata queries on extremely large data sets.
- **Filesystem Offload:** Allows for indexed metadata searches, data versioning, and policy execution without putting additional strain on the system.



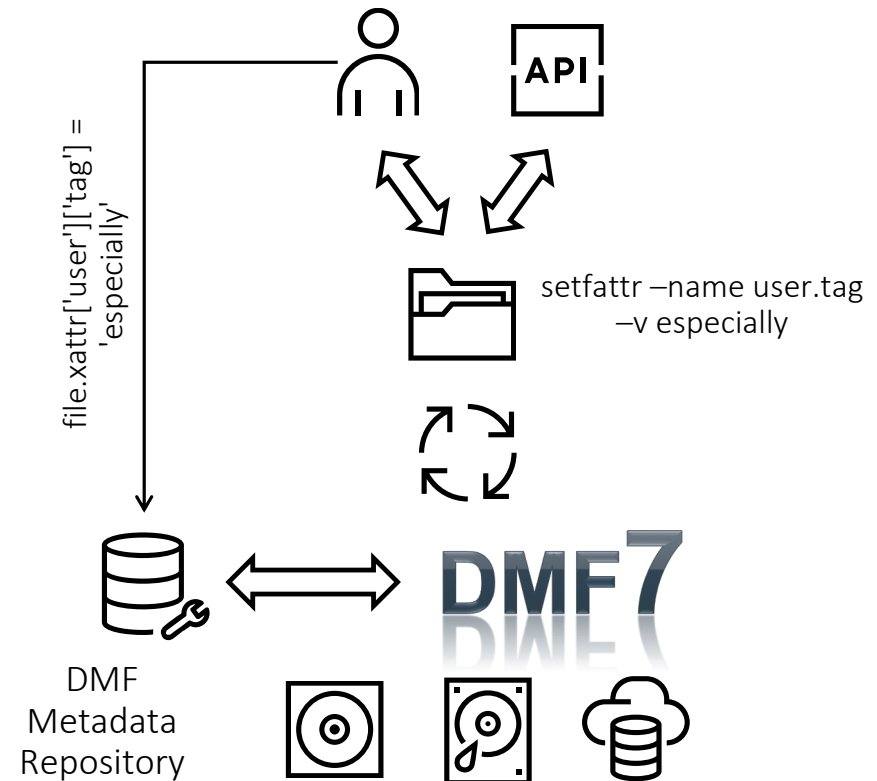
# Data Management | DMF 7 Metadata Search & Extended Attributes

## Key Takeaways

### Searchable Metadata with Standard POSIX

DMF 7 captures and stores all file and directory metadata, including POSIX extended attributes. Users can add extended attributes directly to the files and directories and include them into metadata search queries.

- **Standard POSIX API:** Attributes are added as simple key-value pairs using standard POSIX utilities such as **setfattr** or **setfacl**. No external databases are necessary. Multiple attributes with key names up to 250 characters and values up to 64KB are supported.
- **All Attribute Namespaces Supported:** User-defined extended attributes, SELinux labels and capabilities (security namespace), POSIX ACLs (system namespace) and trusted attributes are captured in metadata repository, and can be specified in queries.
- **Metadata Restore on Staging:** All extended attributes are recovered when DMF 7 object is staged into filesystem and then tracked for changes.



# Data Management Framework | DMF 7 Filesystem Queries

## Filesystem Reflection

File Field	Operators	Values	Usage
file.name	=   like	'[*]<string>[*]<string>']	file.name like '*.csv'
dir.path	=   like	'[*]<string>[*]<string>']	dir.path like '/dataset/*'
file.path	=   like	'[*]<string>[*]<string>']	file.path = '/dataset/run.csv'
file.size	=   <   >   <=   >=	<integer> [ kb   mb   gb   tb   pb ]	file.size > 10gb
file.accesstime	=   <   >   <=   >=	<integer>	file.accesstime > 0
file.modifytime	=   <   >   <=   >=	<integer>	file.modifytime > 0
file.changetime	=   <   >   <=   >=	<integer>	file.changetime > 0
file.createtime	=   <   >   <=   >=	<integer>	file.createtime > 0
file.age	=   <   >   <=   >=	<integer> [ s[econd[s]]   m[inute[s]]   h[our[s]]   d[ay[s]] ]	file.age > 2 hours
file.userid	=   <   >   <=   >=	<integer>	file.userid = 0
file.groupid	=   <   >   <=   >=	<integer>	file.groupid = 0
file.projectid	=   <   >   <=   >=	<integer>	file.projectid = 1
file.state	=	REG   DUL   OFL   MIG   UNM   MOD   INV	file.state = DUL
file.type	=	REG   DIR   SYM   PIPE   SOCK   BDEV   CDEV   DDEV	file.type = DIR
file.id	=	<uuid>	file.id = 000e0000-37b7-0592-0000-000014320594
file.xattr	=	'<string>'	file.xattr['namespace']['name'] = 'value', file.xattr['namespace'] = 'name', file.xattr = 'namespac

# Data Management Framework | DMF 7 Object Queries

## Object

Object Field	Operators	Values	Usage
object.path	=   like	'[*]<string>[*]<string>']	object.path like '/dataset/*'
object.size	=   <   >   <=   >=	<integer> [ kb   mb   gb   tb   pb ]	object.size <= 1mb
object.userid	=   <   >   <=   >=	<integer>	object.userid = 0
object.groupid	=   <   >   <=   >=	<integer>	object.groupid = 0
object.projectid	=   <   >   <=   >=	<integer>	object.projectid = 1
object.accesstime	=   <   >   <=   >=	<integer>	object.accesstime > 0
object.modifytime	=   <   >   <=   >=	<integer>	object.modifytime > 0
object.changetime	=   <   >   <=   >=	<integer>	object.changetime > 0
object.createtime	=   <   >   <=   >=	<integer>	object.createtime > 0
object.tags	contains	'<string>'	object.tags contains 'beta'
object.type	=	REG   DIR   SYM   PIPE   SOCK   BDEV   CDEV   DDEV	object.type = REG
object.id	=	<uuid>	object.id = 3b26d16a-a473-411f-bb4e-c790042223aa
object.fsid	=	<uuid>	object.fsid = 3b26d16a-a473-411f-bb4e-c790042223aa
object.xattr	=	'<string>'	object.xattr['user']['project'] = 'omega'

Complex query usage: file.state = REG and (file.name like '\*.csv' or file.name like '\*.tsv') or (file.size > 1gb and file.state = DUL)

---

## Data Management Framework | **DMF 7** Query Examples

### Filesystem Queries:

```
find -q "file.age > 1 day" --fs static100
find -q "file.type = DIR" --fs static100
find -q "file.size < 1 mb" --fs static100
```

### Object Queries:

```
find -q "object.path like '*daily*'"
find -q "object.tags contains 'sloth'"
```

### “Action” Queries:

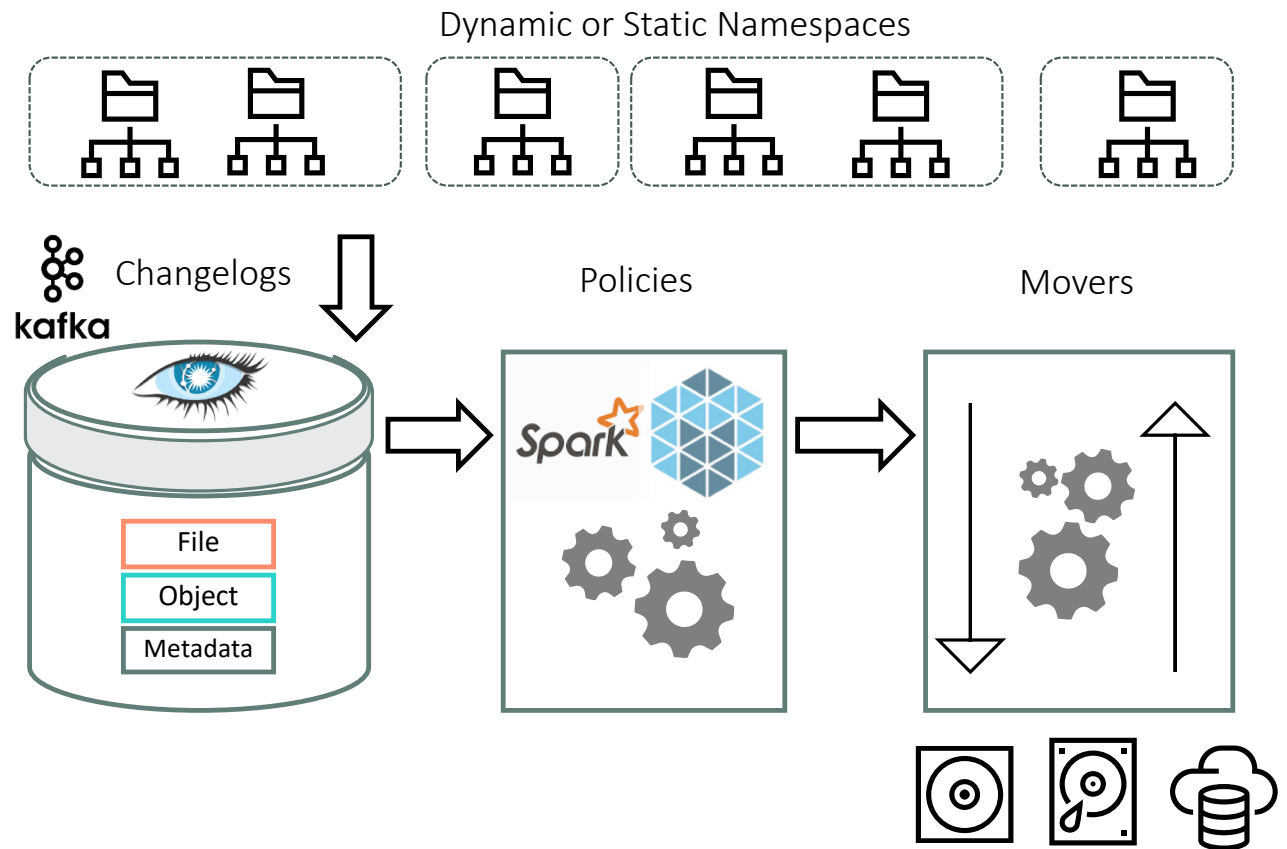
```
put -q "file.xattr['user']['tag'] = 'sloth'" --fs static100 --set sloth
stage --query "object.tags contains 'sloth'" /static100/stage
```



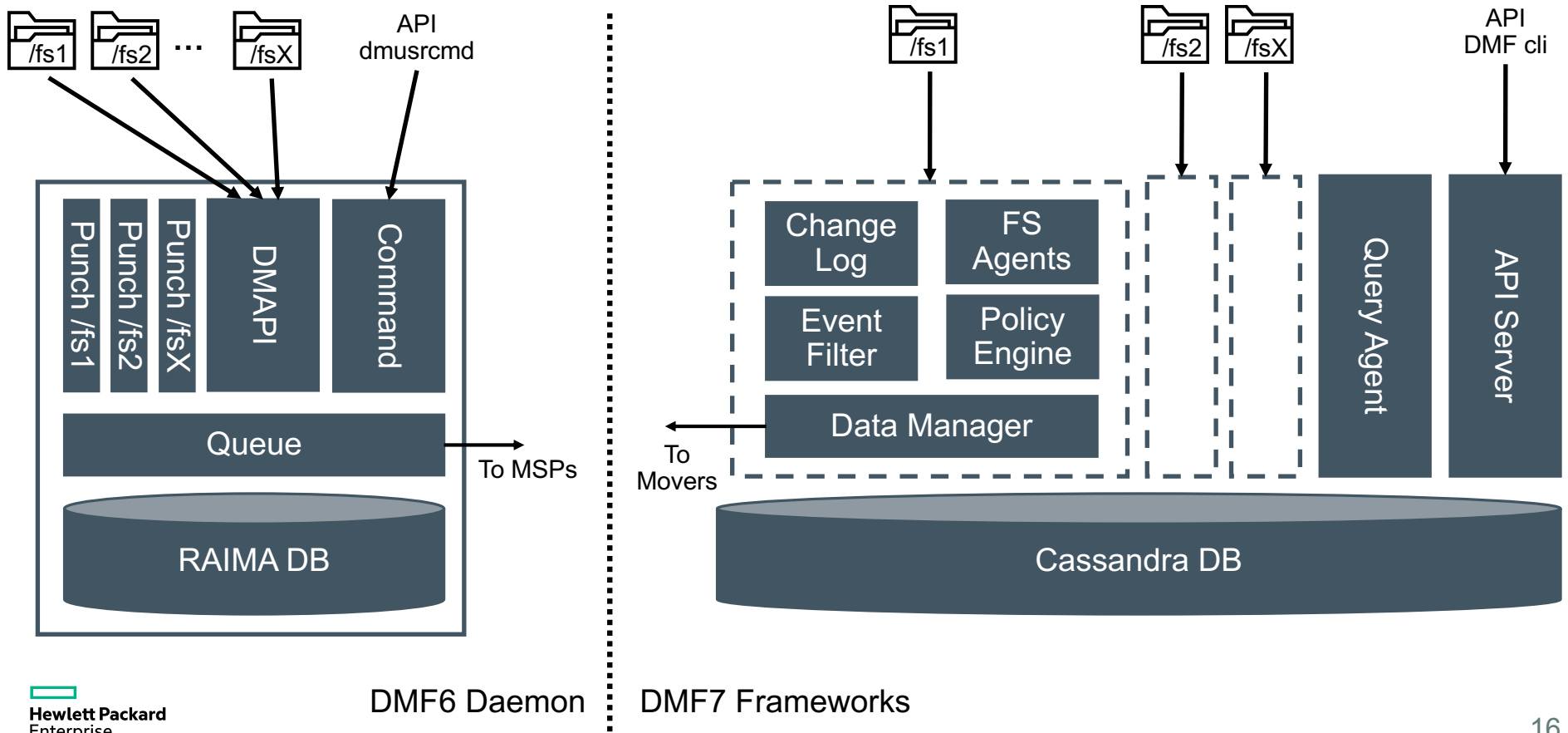


# Data Management Fabric | DMF 7 Architecture

- Kafka for Changelog processing
- Cassandra for Scalable Metadata
- Mesos for Task Scheduling
- Spark for Query Engine
- Zookeeper for Configuration
- Containerized Components
- Dedicated Components per Filesystem
- Component level HA

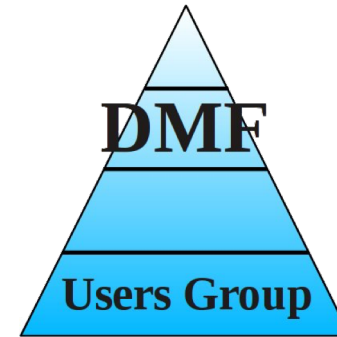


# Data Management Framework | DMF 7 Scalable Front End





**Hewlett Packard**  
Enterprise



**Thank You**