the SDSys Framework

Building Multi-Purpose (Storage) Infrastructure Using A Software-Defined System

SDMaster  SDWorkflow  SDData
SDCompute  SDStorage  SDNetwork
AGENDA

TOPIC

✴ Review SDSys (Storage) Architecture and Components
✴ Discuss protocols used for SDSys Communications
✴ Compare SDSys to Infrastructure (IaaS) and Platform (PaaS) models
✴ Detail use cases for SDSys in Production Environments
Processing ‘Domains’...

...SO...WHY IS IT SO COMPLEX?
What we (seem to) have...
What we (should) have...

Management Plane

Control Plane

Processing Plane

INFRA

DATA

APPS
A **Software-Defined System** (SDSys) creates a unified **Control-Plane** that is...

Simple, Scalable & Automated

...allowing Management to synchronize the Processing-Plane domains of **Applications, Data & Infrastructure** into a highly-productive system capable of complex work, with minimal effort.
SDSys Control-Plane

**MASTER Controller**
Protocol, Directory & Scheduling Services

**WORKFLOW Controller**
Intelligent Application & Workflow Management

**APPLICATION Controller**
Applications, Services & Licensing Management

**SECURITY Controller**
System Authentication & Policy Management

**COMPUTE Controller**
CPU/GPU/Memory Provisioning & Management

**NETWORK Controller**
Secure Connections & Bandwidth Management

**STORAGE Controller**
Storage Provisioning & Service Optimization

**DATA Controller**
Global Data Catalog & “Presentation” Management
Building SDSys

Pre-Built Workflows
Real-time Requests

-- DISTRIBUTED SDSys CONTROLLERS --

Applications | Data Sets | Compute | Network | Storage

allocate
configure
present
run
pause
de-allocate
DATA is abstracted from the underlying storage via the SDSys DATA PRESENTATION model, as enabled by the DATA & WORKFLOW Controllers and Presentation GATEWAYS.
TOPIC

- Review SDSys Architecture and Components
- Discuss protocols used for SDSys Communications
- Compare SDSys to Infrastructure (IaaS) and Platform (PaaS) models
- Detail use cases for SDSys in Production Environments
SDSys Communications

MASTER Controller

WORKFLOW

APPs

COMPUTE

NET

DATA

STORAGE

SECURITY

Current Southbound Comm
- Python
- Java
- C
- SMI-S

Northbound Comm
- NbC TLS
- SOAP [WSS]

Southbound Comm
- SbC [TLS] [As Needed]
How do SDSys controllers negotiate for resources?

1. **Request**: Workflow Controller queries the SMC for System Resources for a given Site ID and Time Period.

2. **Review**: SMC Determines Availability of Requested System Resources from other Controllers and Master Schedule.

3. **Approve**: SMC responds with SR Reservation ID and SR Details.

4. **Execute**: SMC instantiates SR and WC executes WORKFLOW as per Reservation.

5. **Complete**: WC completes designated WORKFLOW / SR are deallocated.
Topics:

- Review SDSys Architecture and Components
- Discuss protocols used for SDSys Communications
- Compare SDSys to Infrastructure (IaaS) & Platform (PaaS) models
- Detail use cases for SDSys in Production Environments
Review SDSys Architecture and Components

Discuss protocols used for SDSys Communications

Compare SDSys to Infrastructure (IaaS) and Platform (PaaS) models

Detail use cases for SDSys in Production Environments
  - 4K Resolution Video Post-Production
  - Genome Sequencing Pipeline
  - Trusted Digital Repository
4K Post-Production

-- DISTRIBUTED SDSys CONTROLLERS --
4K Post (ALT View)

Control Plane

Data Plane

Data Controller

Data Cataloging

Metadata Extraction

Data QC

Virus Check

Upload Utility

Asset Storage

MD Store

Workflow Controller

Automated QC

Operator Review

Metadata Packaging

Deliver

Archive (to Cloud)

Metadata Packaging

Operator Review

Metadata Packaging

Asset Storage

MD Store

Wednesday, April 23, 14
Genome Sequencing Pipeline

SDSys Controller-Based Automation

Applications  Data Sets  Compute  Network  Storage

-- DISTRIBUTED SDSys CONTROLLERS --

Workflow Designer

alignment  QC  annotate  CNV  visualize

ingest  assembly

archive
A Trusted Digital Repository is...

A TDR definition must start with ‘a mission to provide reliable, long-term access to managed digital resources for its Designated Community, now and into the future.’

• TDR is an extension of Open Archival Information System (OAIS) Reference Model (ISO 14721)

• Formalized for Auditing & Certification under ISO 16363
The Repository Shall . . .

...assign and maintain **persistent identifiers** of the [archival] Information Package and its components so as to be unique within the context of the repository.

... **automate repair** to make it is as fast, cheap, and reliable as possible

...have adequate specifications enabling **recognition & parsing of content**

...commit to a regular schedule of self-assessment and external certification.

...provide a mechanism for **auditing & integrity verification** the integrity of the collection/content, working to detect latent faults as early as possible

...have mechanisms in place for monitoring the preservation environment detect latent faults as soon as possible

...define, collect, track, and appropriately provide its information integrity measurements.

...strive for **independence of replicas**
SDSys Archive?

- persistent identifiers
- recognition & parsing content
- auditing & integrity verification
- automated repair
- independence of replicas
Software-Defined Archive?

What inherent functionality of a SDSys can we apply to build a TDR with...

- persistent identifiers
- recognition & parsing content
- auditing & integrity verification
- automated repair
- independence of replicas
✓ **persistent identifiers**

Every object/file in a SDSytem is given a unique identifier, which is tracked within the object itself and within the SDS Data Controller(s)
What inherent functionality of a SDSys can we apply to build a TDR with...

✓ recognition & parsing of content (essence)

Workflow and Data Controllers typically include content introspection capabilities, which is then passed into the metadata tags of the SDSys objects. Additionally, Workflow steps can include automated QC process using 3rd-party solutions to provide bit-level introspection.
What inherent functionality of a SDSys can we apply to build a TDR with...

✓ auditing & integrity verification

Content Integrity Checking can be maintained through comparison of the metadata, (including checksums) held on the Data Controllers and in the respective content objects.
What inherent functionality of a SDSys can we apply to build a TDR with...

✓ **automated repair**

Through the use of multiple repositories, detected faults can be automatically and repaired without administrator intervention - even if the underlying medium are completely different (i.e. - disk & tape)
What inherent functionality of a SDSys can we apply to build a TDR with...

✓ independence of replicas

The inherent behavior of ObjectStore Technology is to provide multiple copies of data. When combined with Write-Once-Read-Many capabilities, it becomes possible to build independent content replicas immune from centralized administrative destruction and correlated faults from Hardware, Software, or Location.
SDSystems are an agile collection of applications, data and infrastructure working together to optimize data-intensive processing. Many of the built-in features of a Software-Defined System support the simplified construction and low-cost operation of a Trustworthy Digital Repository.
Thank you for your Attention!

paul.evans@daystrom.com
www.daystrom.com