

A wide-angle photograph of the New York City skyline, featuring the Empire State Building as the central focus. The sky is a clear, pale blue. In the foreground, a body of water is visible. A large red rectangular box is overlaid on the lower left portion of the image, containing white text.

Ingesting and Managing Digital Content at Scale

Features, Process, and Tips

Rosetta Advisory Group 2016 – New York City
Josh Weisman, Development Director

Agenda



Introduction

Scalability Considerations

Hardware Planning

Scalability Features

Optimizing Rosetta

Case Studies

Summary

Agenda



Introduction

Scalability Considerations

Hardware Planning

Scalability Features

Optimizing Rosetta

Case Studies

Summary

Introduction

- Digital content is proliferating at unprecedented scale
 - Widespread digitization
 - Fast-paced generation of born digital content
- Rosetta has been designed with high throughput ingest scenarios in mind
- Improvements are consistently being made



Introduction

Rosetta is a highly scalable, enterprise digital asset management system.

Like any high-performance system, Rosetta needs to be tuned and throughput must be taken into consideration when configuring workflows.

This presentation describes some considerations when designing a high throughput workflow and highlights some of the features Rosetta supports to optimize such flows.

- Rosetta has a dedicated performance lab on which each version is tested
 - Full regression suite
 - Ingest benchmark
- 3 all-in-one servers
 - 24 CPUs, 32G RAM each
- Current environment has
 - 240K IEs
 - 4M Files
 - 0.5M – Images
 - 2M – PDF
 - 1.3M – Office (doc,xls,ppt)
 - 0.2M – Other

Improvements

- Improving performance is never “done”
 - Delivering high performant enterprise systems is a complex endeavor which involves the software, infrastructure vendors, internal IT staff
- Rosetta improvements based on:
 - Internal testing
 - Tuning projects/efforts
 - Reports from customers
- From the field...

Recently implemented new features in the Rosetta search engine to significantly improve publishing times in large repositories (5.1)

Linear Scalability

- Rosetta has proven near-linear scalability
- Rosetta can “scale wide”- adding additional Rosetta instances will increase total ingest rates
 - Can be impacted by infrastructure bottlenecks such as network, storage, etc.
- Rosetta allows for a “go and grow” approach- start small and add servers according to requirements

Agenda



Introduction

Scalability Considerations

Hardware Planning

Scalability Features

Optimizing Rosetta

Case Studies

Summary

Parallelization

- To increase throughput in a system, it is necessary to examine how long an individual process takes and identify opportunities to parallelize that processing.
- When ingesting digital content, Rosetta parallelizes at the level of the SIP (submission information package).
- An individual worker thread operates on the package and moves it through the various processing stages.
- Depending on system resources, the number of worker threads can be increased to achieve higher levels of parallel processing.

File Handling

- Rosetta maintains individual storage locations for 3 different modules of the system- Deposit, Operational, and Permanent.
- Combine those modules with the original staging location of the digital content and there are three distinct file copy operations which Rosetta performs as it moves the files through the stages of processing.
- As moving large numbers of files around the file system can have a big impact on processing time, how the copy operations are handled becomes significant in high throughput scenarios.

File Operations (I/O)

- As part of ingesting digital content into the repository, Rosetta performs several operations on the files themselves.
- In the validation stack phase, Rosetta runs fixity (check sum), virus check, format identification, and metadata extraction operations.
- In the enrichment phase, Rosetta can be configured to create access copies of the ingested content.
- Since these activities must be performed for each file, how they are configured can have a great impact on the overall system throughput.

Agenda



Introduction

Scalability Considerations

Hardware Planning

Scalability Features

Optimizing Rosetta

Case Studies

Summary

- Database storage requirements can be affected by usage patterns and configuration of Rosetta features. Some of the considerations which affect database storage include:
 - Descriptive metadata per intellectual entity (IE), representation, and file
 - Events
 - Publishing
 - Technical metadata extraction

Database

- Database storage benchmarks based on typical configuration

Number of Files	DB Storage
0.5M	175 GB
1M	275 GB
2.5M	570 GB
5M	1050 GB
10M	2000 GB

Storage

- The biggest factor affecting required storage size for the Rosetta digital repository is ingest workflow file handling.
- If files are moved rather than copied, the storage requirements for the deposit and operational storage are reduced considerably.

Storage

- Storage benchmarks based on typical Rosetta configuration

Storage	Factor
Deposit	1 x monthly ingest rate
Operational	2 x monthly ingest rate
Permanent	1.2 x repository size

Compute

- In lab and customer scenarios, Rosetta has comfortably ingested 50 GB of files per hour per machine meeting the minimum system requirements.
- Some customers have successfully ingested at sustained rates significantly higher than this benchmark.
- There are many factors which influence ingest rates, the most significant of which being network bandwidth and disk IOps.

Agenda



Introduction

Scalability Considerations

Hardware Planning

Scalability Features

Optimizing Rosetta

Case Studies

Summary

Scalability Features

- Rosetta boasts many features which can be leveraged to increase the rate at which content can be ingested.
- Since there are trade-offs when implementing these features, institutions should consider the factors outlined below when making decisions about how to configure Rosetta.

Parallel SIP Processing

- Rosetta relies on a system of queues and worker threads to parallelize activities.
- Rosetta maintains queues for several discreet areas of system processing. For the purposes of optimizing ingest levels, the SIP processing queue is of greatest interest.
- Out of the box, Rosetta is configured to allocate 5 worker threads to the SIP processing queue.

Parallel SIP Processing

- Rosetta can perform parallel processing at the level of an individual SIP.
- In order to ensure maximum utilization of resources, it is important to plan the ingest workflow so that there are enough individual SIPs to occupy all of the allocated worker threads simultaneously.
- Factors affecting proper worker levels:
 - Server's resources (CPU cores, RAM, network bandwidth, storage IO rates)
 - Other work being performed by the server (maintenance jobs, user interface/API requests)

Parallel SIP Processing

ExLibris Rosetta Administration

User: John Smith Help Logout

HomeAdvanced Configuration

Quick LaunchSite Map

Home > Advanced Configuration > General > SIP Processing Workers

Waiting SIPs 0

In Processing SIPs 0

Default Level 4

	Server Name	Level	Server Role
1		0	DEP,REP,DEL,PER
2		0	DEP,REP,DEL,PER
3		4	DEP,REP,DEL,PER
4		4	DEP,REP,DEL,PER
5		4	DEP,REP,DEL,PER
6		4	DEP,REP,DEL,PER
7		2	DEP,REP,DEL,PER,IDX
8		2	DEP,REP,DEL,PER,IDX

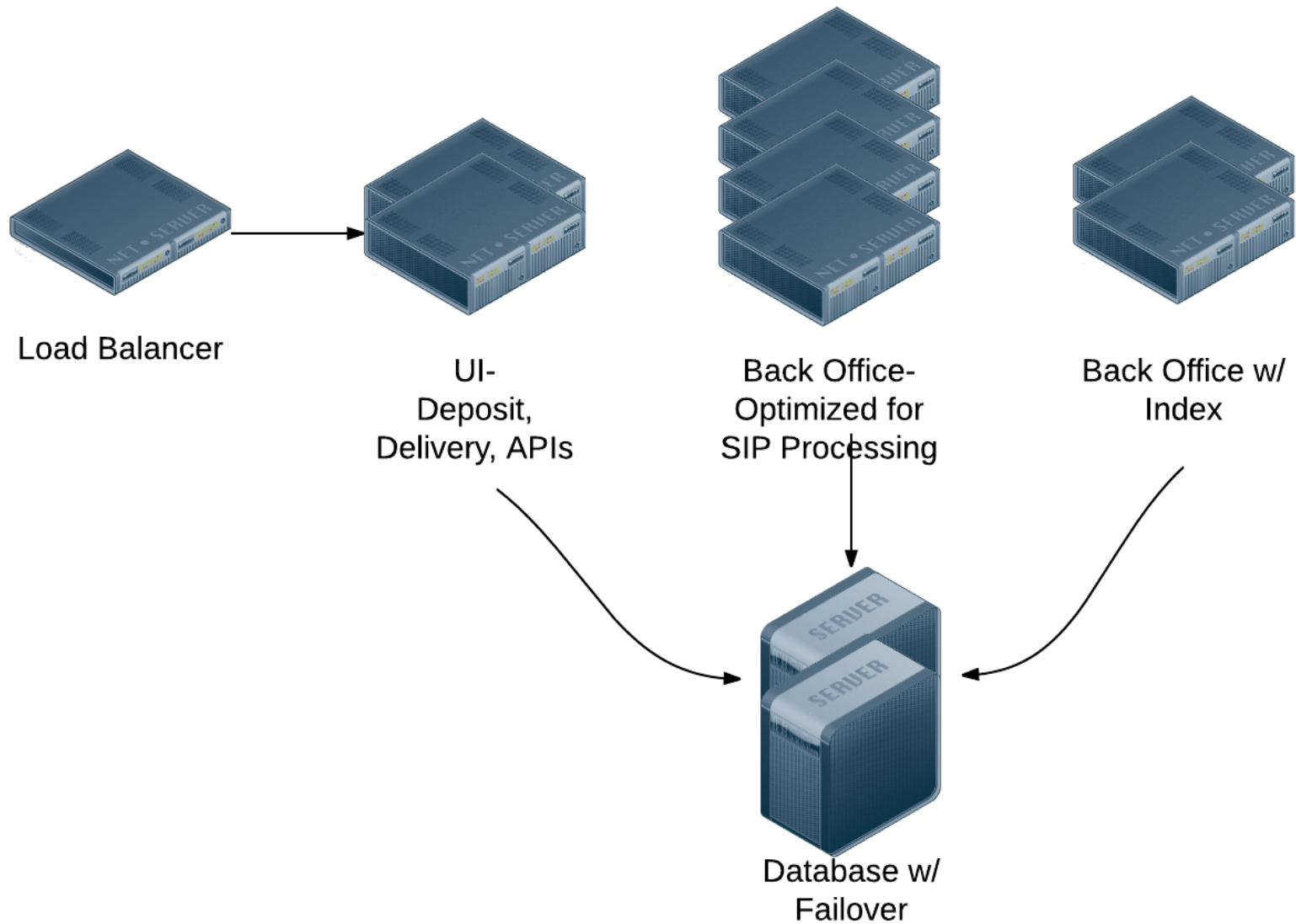
Cancel

Refresh

Commit Changes

© Ex Libris Ltd., 2016

Sample Topology



File Handling

- When ingesting digital content into Rosetta, the files are logically moved through four “stations”:
 - from its original storage location,
 - to the deposit storage,
 - through operational storage
 - into the permanent repository.
- Rosetta can be configured to move the files through those stations in different ways:
 - Copy
 - Move
 - Link

File Handling - Copy

- Default file handling method is to copy the file from each location.
- Provides the maximum security and flexibility for the files
 - individual copies are maintained throughout the entire workflow
 - can always be restored in case of error.
- Requires the most time, network, and storage resources.

File Handling - Move

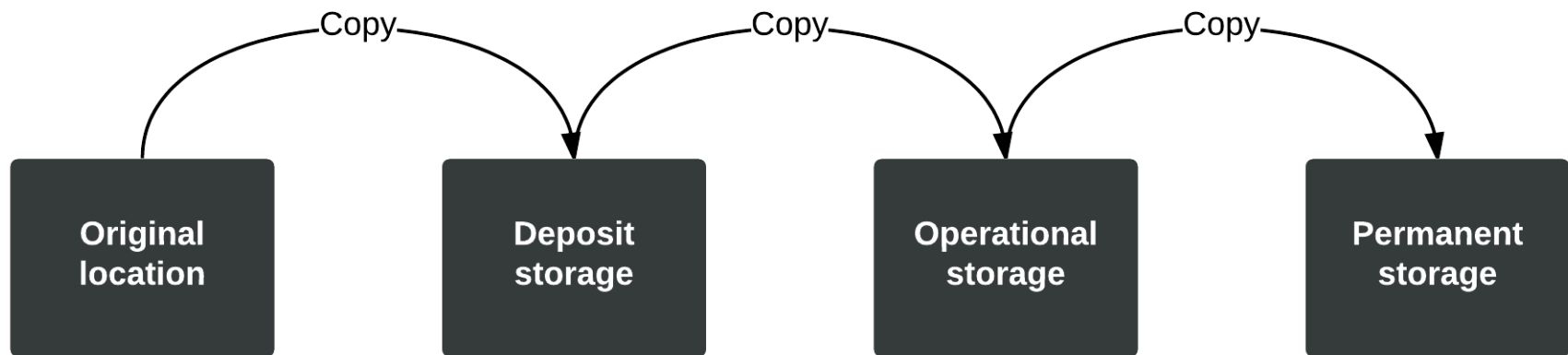
- If storage locations are on the same physical mount point, it is more efficient to move the file to each location,
 - Requires only a logical update in the filesystem
- File cannot be rolled back in case of corruption as the original file is being operated on

File Handling - Link

- Most efficient file handling method is to simply link to the original location of the file.
- Requires no updates to the storage location.
- Limits the ability to rollback to re-process a file in case something goes wrong.

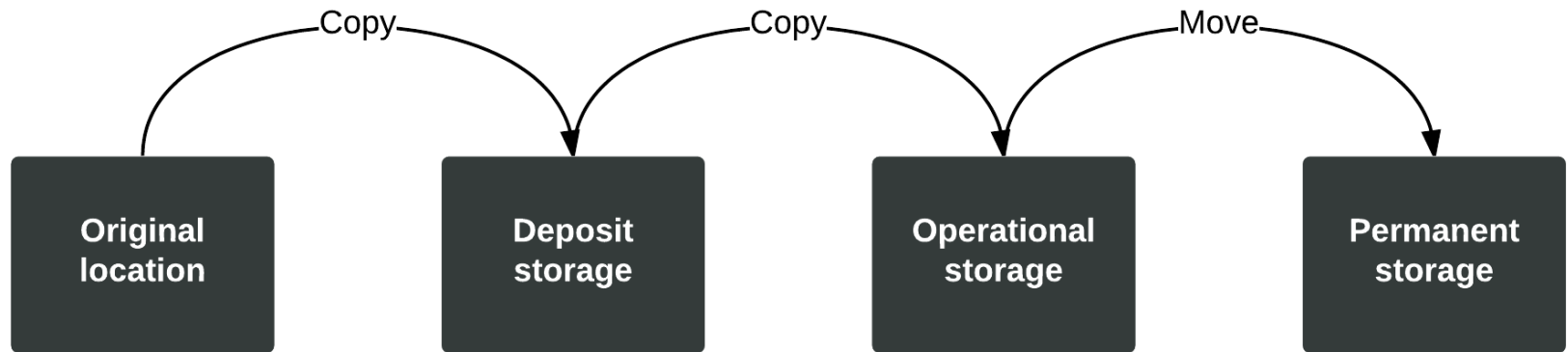
File Handling – Safe and Sound

- Pros:
 - Files can be rolled back or re-processed at any time
- Cons:
 - Requires more storage space
 - Takes time to copy file



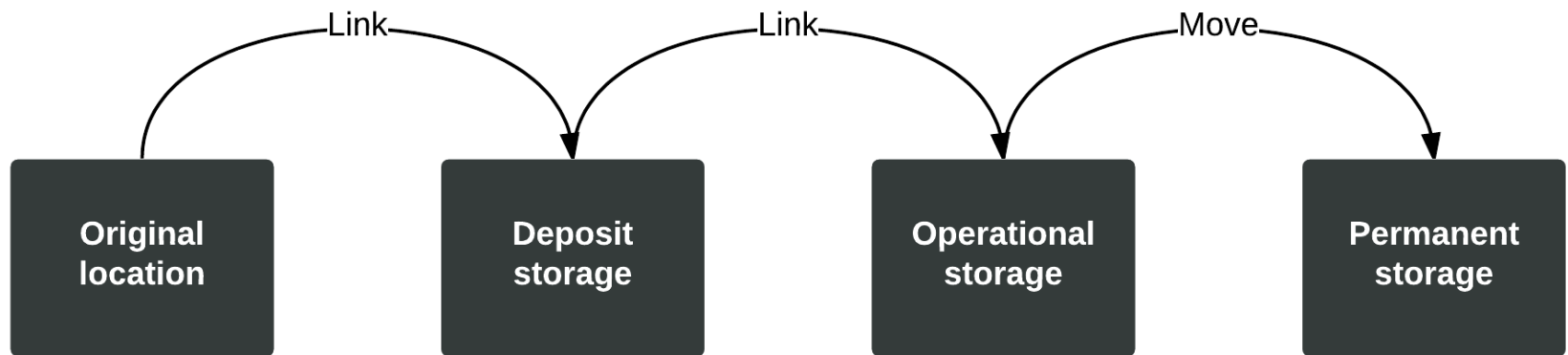
File Handling - Balanced

- Pros:
 - Optimizes moving between operational and permanent
 - Maintains a copy on the deposit storage for rollback purposes
- Cons:
 - Additional space in deposit storage
 - Time to copy file to operational



File Handling – Throughput Optimized

- Pros:
 - Requires least amount of storage space
 - Minimal time required for IO
- Cons:
 - Limited options for rollback



Format Identification

- As part of the validation stack, Rosetta runs a format identification task to evaluate the file and determine its format.
- The amount of the file which is scanned by the format identification task and provided to the tool can be configured.
- Out of the box, Rosetta scans the first 64Kb of the file as a part of this task.
- This number can be reduced for high throughput workflows
 - May reduce the effectiveness of the format ID process.

Checksum

- As part of the validation stack, Rosetta performs a checksum on the file
 - Uses the check sum to ensure bit-level consistency of the file
 - Validated throughout the file processing workflow and into the permanent repository.
- Out of the box, Rosetta is configured to calculate the checksum using three algorithms- MD5, SHA1 and CRC32
- The fixity algorithms can be limited in high throughput workflows
 - May slightly reduce the overall confidence in the consistency of the repository.

Workflow Configuration

- The SIP processing workflow in Rosetta is configured out of the box to leverage all of the features of Rosetta.
- In some cases, an institution may want to perform some of the tasks pre- or post-ingest in order to optimize the workflow for higher ingest rates.
- Examples of tasks which are candidates to be removed from the ingest workflow include:
 - Virus check
 - Derivative copies
 - Metadata extraction

Workflow Configuration – Virus Check

- Files are often scanned for viruses at an institution's edge
- This may make an additional virus check in Rosetta redundant.
- In this case, it is recommended that the virus check in Rosetta be removed from the workflow.

Workflow Configuration – Derivative Copies

- In some cases, derivative copies are created as a pre-ingest step
- Provided to Rosetta along with the original.
- This reduces the amount of resources required during the ingest flow.

Workflow Configuration – Metadata Extraction

- Some institutions for whom ingest rates are of primary importance are willing to forgo the metadata extraction step in the ingest workflow.
- Files are stored in the permanent repository along with full provenance and descriptive metadata.
- Technical metadata can be extracted at a later time in a maintenance process.

Agenda



Introduction

Scalability Considerations

Hardware Planning

Scalability Features

Optimizing Rosetta

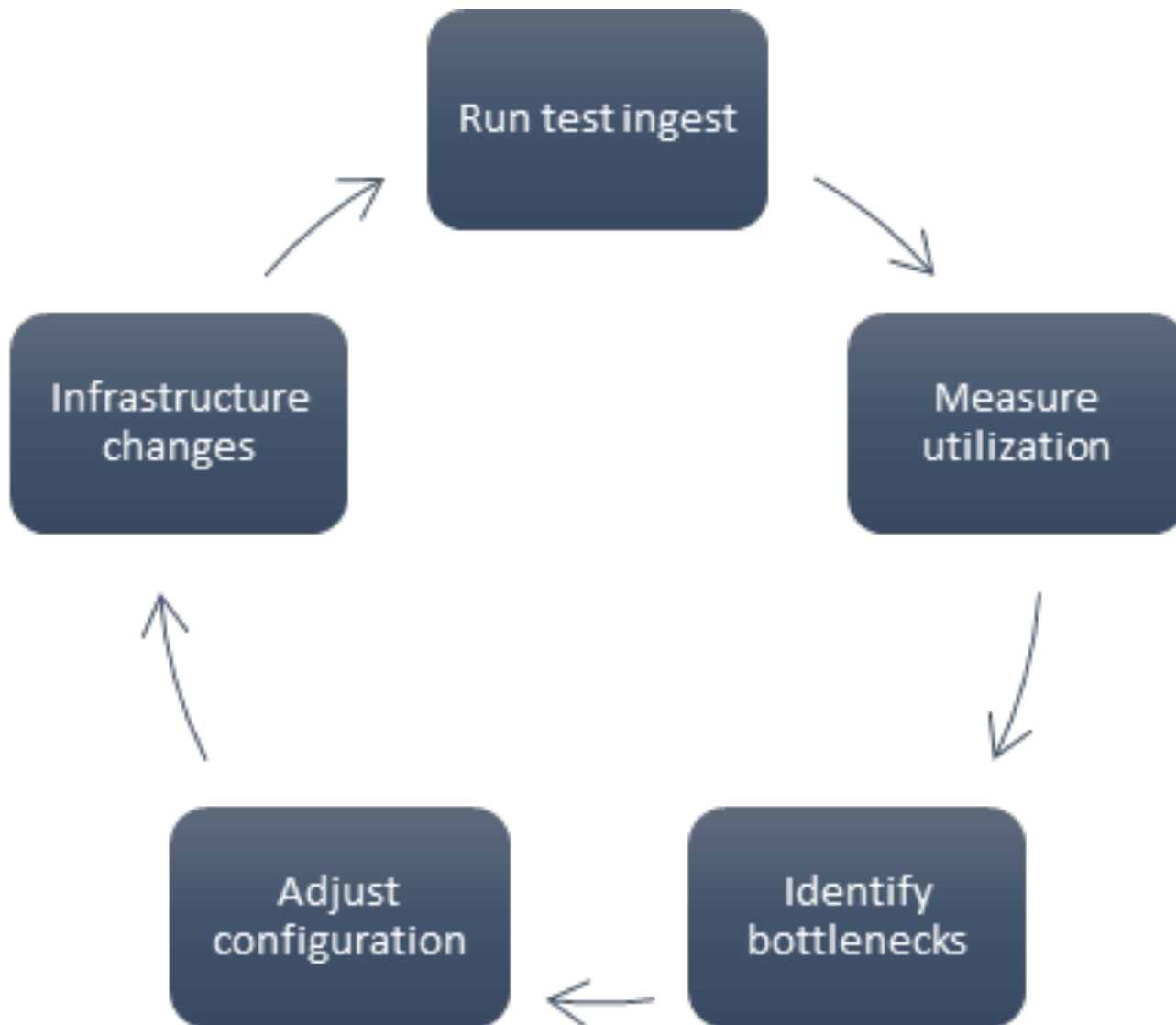
Case Studies

Summary

Optimizing Rosetta

- A tuning exercise can help achieve desired throughput levels from an end-to-end ingest workflow
- Will help determine the optimized workflow, configuration, and performance benchmark
- The process for the tuning effort includes:
 - Setting resource utilization goals,
 - Running test ingests flows,
 - Monitoring results,
 - Analyzing for bottlenecks,
 - Making configuration changes,
 - Rerunning the workflow.

Optimization Process



Optimization Process

- Prepare representative content (or actual content if in production)
- Run an initial ingest cycle is executed with the out-of-the-box configuration while monitoring CPU utilization, memory usage, network saturation, and disk latency.
- Assuming resources are not fully utilized, the number of SIP processing workers is increased and another test run is executed.
- This is repeated until utilization no longer increases, indicating a resource bottleneck.
 - If the desired throughput levels have been reached, the process can end there.
 - If not, the bottleneck must be analyzed and the appropriate steps taken to resolve the bottleneck revealed in the infrastructure.

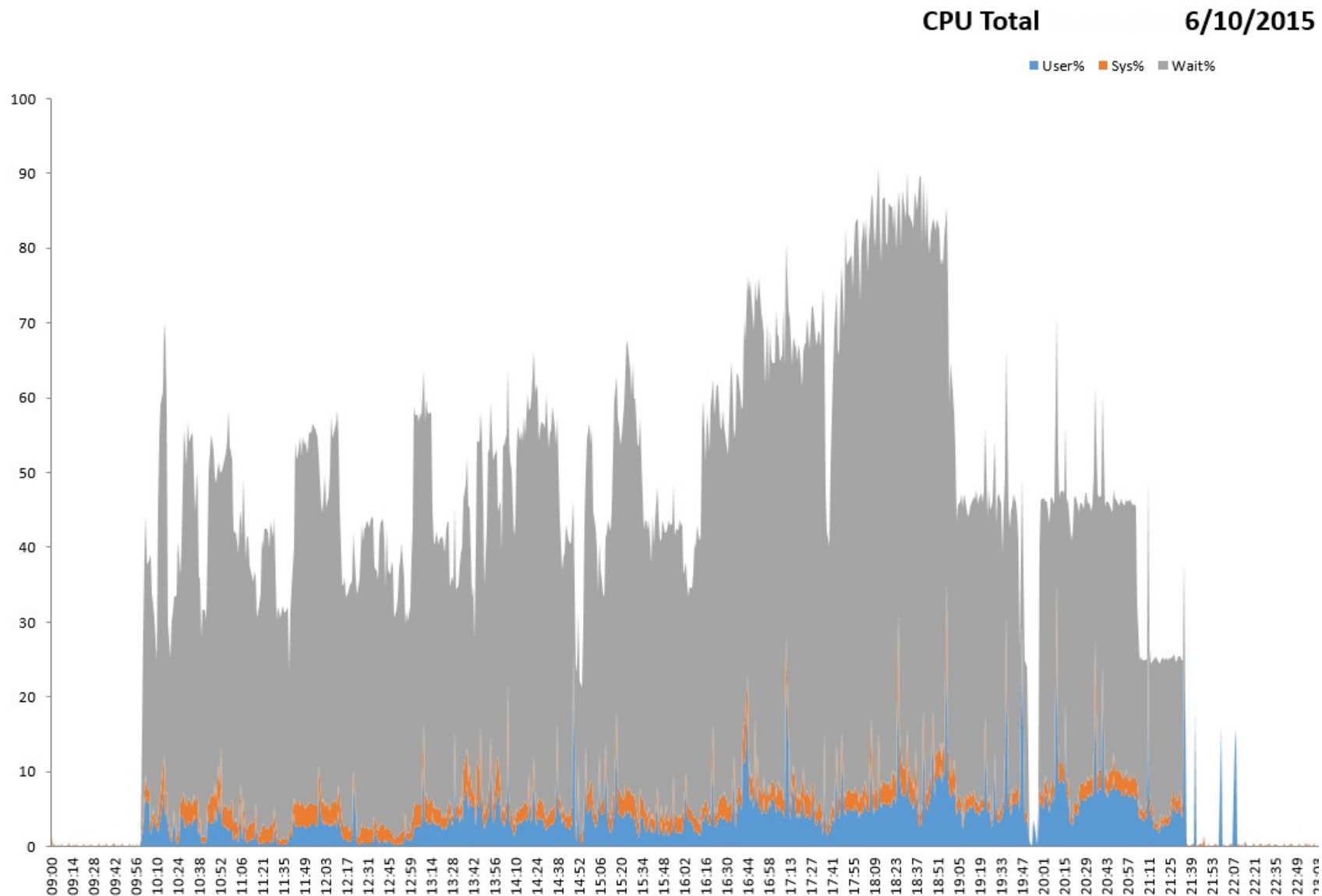
Infrastructure Review

- Infrastructure should be reviewed to ensure it complies with best practices.
- Confirm that the environment complies with the current Rosetta system requirements for a new installation.
- Validate that all of the required ports are open and that the servers can communicate among themselves.
- Use the top command to check that the system is allocated with the proper resources.
- Virtual machines must have dedicated resource allocation.
- Verify NFS mount points configuration
- Validate heap size configuration

Tools - nmon

- Use a script to execute monitoring of system resources via nmon
- For regular monitoring, add the following to `crontab` The nmon log files can be loaded into Excel for analysis.

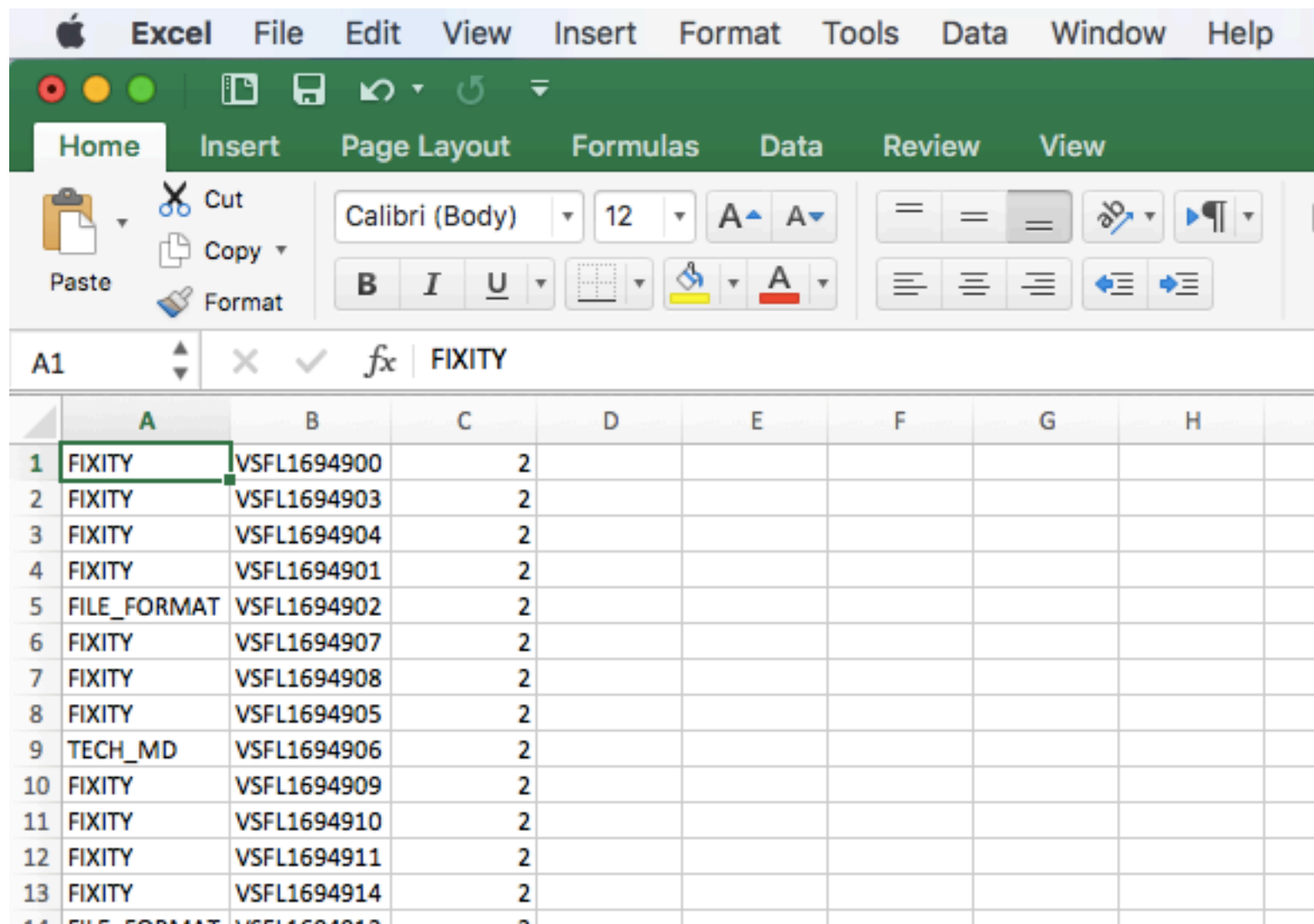
Tools - nmon



Tools – Log mining

- Rosetta prints the time it takes to execute each task in the SIP processing workflow to the log file.
- It can be helpful to analyze the log file to look for long running validation stack tasks.
- This often can point to infrastructure performance problems such as slow disk speeds or saturated network.

Tools – Log mining



The screenshot shows the Microsoft Excel interface. The menu bar includes Apple, Excel, File, Edit, View, Insert, Format, Tools, Data, Window, and Help. The ribbon is set to 'Home', showing options for Cut, Copy, Paste, Format, and various text and alignment tools. The formula bar displays 'FIXITY' for cell A1. The spreadsheet contains the following data:

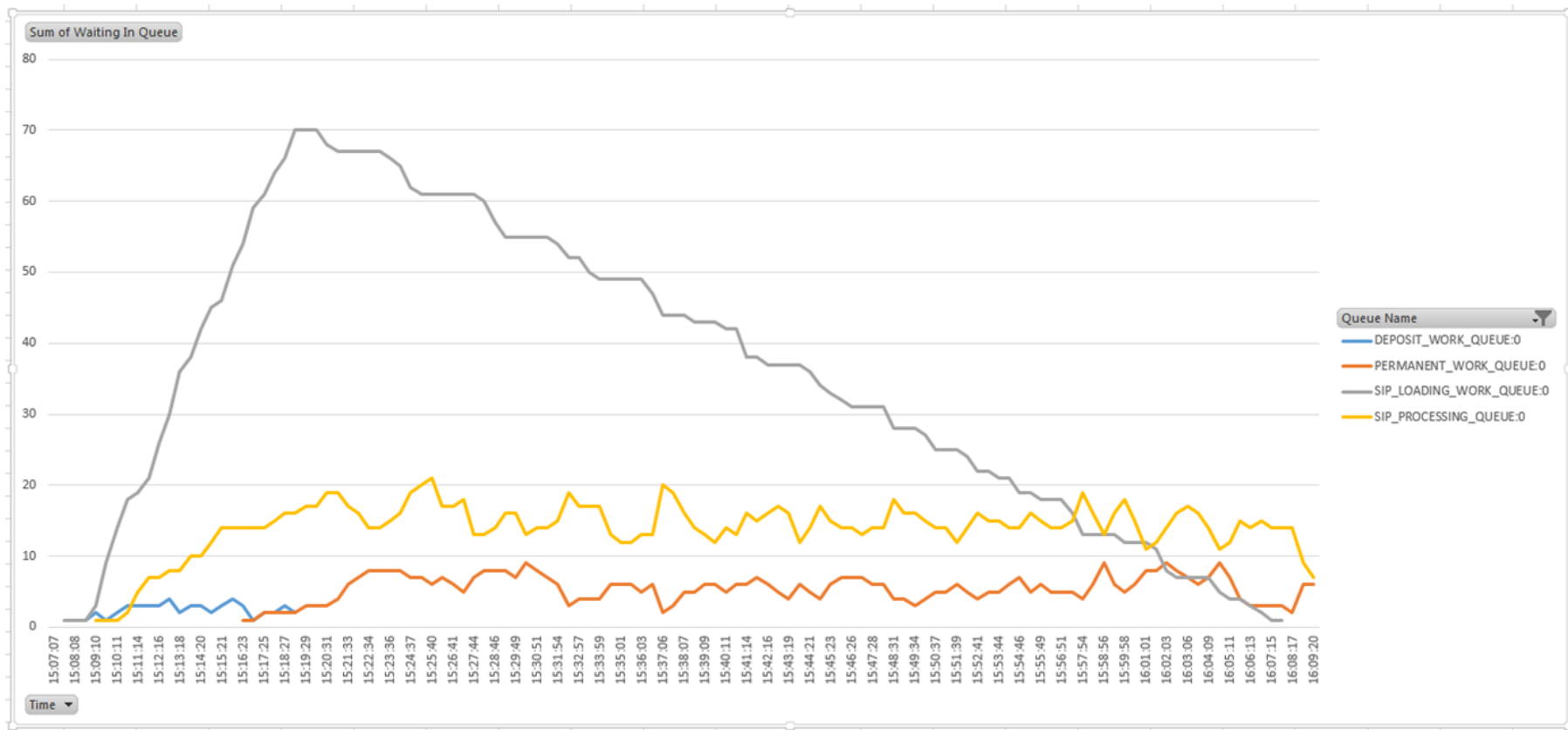
	A	B	C	D	E	F	G	H
1	FIXITY	VSFL1694900	2					
2	FIXITY	VSFL1694903	2					
3	FIXITY	VSFL1694904	2					
4	FIXITY	VSFL1694901	2					
5	FILE_FORMAT	VSFL1694902	2					
6	FIXITY	VSFL1694907	2					
7	FIXITY	VSFL1694908	2					
8	FIXITY	VSFL1694905	2					
9	TECH_MD	VSFL1694906	2					
10	FIXITY	VSFL1694909	2					
11	FIXITY	VSFL1694910	2					
12	FIXITY	VSFL1694911	2					
13	FIXITY	VSFL1694914	2					
14	FILE_FORMAT	VSFL1694913	2					

Tools – Queue Monitoring

- Rosetta utilizes several worker queues each with its own pool of worker threads.
- The default settings for the number of workers in each pool is sufficient in most cases.
- Some customers may have content or workflows which require customization of the default settings.
 - For example, IEs with an especially large number of files.
- Queue backlog can be monitored during an ingest run to determine if there are sufficient worker threads allocated to handle the workload.
 - A monitor queues script can be run which writes the number of SIPs waiting in each queue at that time.

Tools – Queue monitoring

The output is written to a file and can be loaded into Excel for analysis. A “healthy” system is represented below, with a bell curve for the SIP Processing queue and a relatively flat backlog for the other queues:



Tuning

- As a result of the findings of the optimization cycles, several adjustments can be made.
- After each adjustment, the cycle should be rerun to measure the impact of the change.
- This process is repeated until an acceptable ingest rate has been achieved.
- Changes in infrastructure, workflows, or content will require a new optimization process.

Tuning

- *Heap size* can be changed based on the garbage collection logs and the memory utilization indicated by the nmon results
- The *work level* may be adjusted based on the CPU, memory, and network utilization shown in the nmon output
- Several *environment-related issues* may be revealed, including:
 - connection timeout
 - I/O latency or NFS configuration (such as no-lock)
 - network connectivity
- *Worker tuning*- If the system resources are optimally utilized, and the environment is configured correctly, the backlog in the different worker queues can be monitored with different ingest flows

Agenda



Introduction

Scalability Considerations

Hardware Planning

Scalability Features

Optimizing Rosetta

Case Studies

Summary

Case Studies

- Case studies from institutions for which throughput rates were critical
- Each institution underwent efforts to optimize their workflow, configuration, and infrastructure.
- Every institution has unique needs and capabilities; but these case studies are helpful as examples of the optimization process and its results.

Topology	<ul style="list-style-type: none">▪ 3 REP servers▪ Additional 2 DEL servers▪ Each with 8 cores and 64 GB RAM
Content Profile	<ul style="list-style-type: none">▪ Scanned daily newspapers (10%)▪ Images (90%)▪ Other: Audio files, digitized books, audio/video, etc.
Performance Goals	<ul style="list-style-type: none">▪ Maximize the system and keep up with incoming digitization workload
Maximum Sustained Throughput Rate	<ul style="list-style-type: none">▪ 3 TB and 180,000 files per day
Current Repository Size	<ul style="list-style-type: none">▪ 15.5M files▪ 2.8M IEs

Topology	<ul style="list-style-type: none">■ Dark Archive<ul style="list-style-type: none">■ 7 all-in-one servers■ Each with 8 cores / 32 GB■ DAM<ul style="list-style-type: none">■ 6 servers
Content Profile	<ul style="list-style-type: none">■ Dark Archive –images and audio/visual materials■ DAM- mixed- PDFs, images, etc.
Performance Goals	<ul style="list-style-type: none">■ 10 TB per day
Maximum Sustained Throughput Rate	<ul style="list-style-type: none">■ 5.2 TB and 25,000 files in 10 hours.
Current Repository Size	<ul style="list-style-type: none">■ 1.2 PB

Topology	<ul style="list-style-type: none">▪ 8 all-in-one servers▪ 2 UI, 6 back office (2 with index)▪ 16 GB RAM each
Content Profile	<ul style="list-style-type: none">▪ Digitized books▪ Variety of digitized content and PDF files
Performance Goals	<ul style="list-style-type: none">▪ 200 books per day (2TB, ~ 200,000 files)
Maximum Sustained Throughput Rate	<ul style="list-style-type: none">▪ 2 TB and 200,000 files in 10 hours.
Current Repository Size	<ul style="list-style-type: none">▪ 160,000 IEs▪ 900,000 files

Agenda



Introduction

Scalability Considerations

Hardware Planning

Scalability Features

Optimizing Rosetta

Case Studies

Summary

Announcing

- Github- scripts & templates
- Whitepaper – will be available on the Knowledge Center

ExLibrisGroup / **Rosetta.Optimization** Watch 3 Star 0 Fork 0

Code Issues 0 Pull requests 0 Pulse Graphs

Scripts for use in Rosetta optimization

1 commit 1 branch 0 releases 1 contributor

Branch: master New pull request Find file Clone or download

jweisman Initial commit		Latest commit 3593f0a 35 minutes ago
README.md	Initial commit	35 minutes ago
analyze-vs-log	Initial commit	35 minutes ago
monitor_queues.csh	Initial commit	35 minutes ago
nmon analyser v34a.xls	Initial commit	35 minutes ago
nmon-monitor	Initial commit	35 minutes ago

README.md

Rosetta Optimization Scripts

Set of scripts and templates for use during Rosetta optimization projects.

Summary

- Rosetta is a proven solution for the large digital repositories with significant throughput requirements.
- Rosetta has proven ingest rates which would allow the creation of a **4 petabyte repository within a year**
- Your institution can optimize its workflows and infrastructure to take advantage of the scaling features provided by Rosetta.
- For consultation regarding optimizing your institution's digital ingest workflows, you can speak to your implementation project manager or open a support case.



QUESTIONS?

An aerial photograph of the New York City skyline at dusk. The sky is a mix of orange, yellow, and blue. The city lights are on, and the Empire State Building is prominent in the center. A semi-transparent red banner covers the lower half of the image, containing the text.

THANK YOU
josh.weisman@exlibrisgroup.com