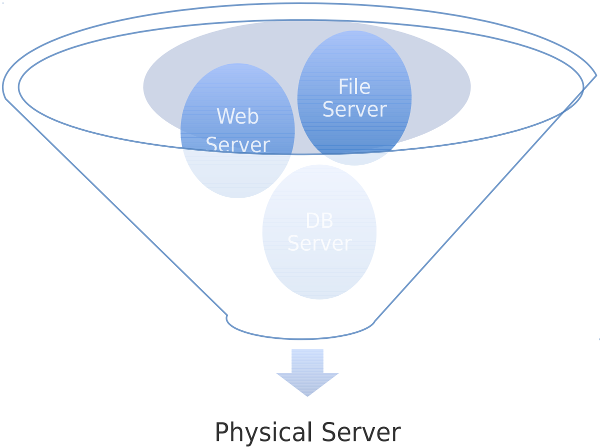
Coriolis: Scalable VM Clustering in Clouds

Daniel Campello [1] <dcamp020@fiu.edu> Carlos Crespo [1]
Akshat Verma [2] Raju Rangaswami [1] Praveen Jayachandran [2]

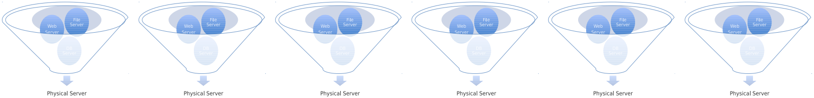
[1]School of Computing and Information Sciences
Florida International University

[2]IBM Research - India

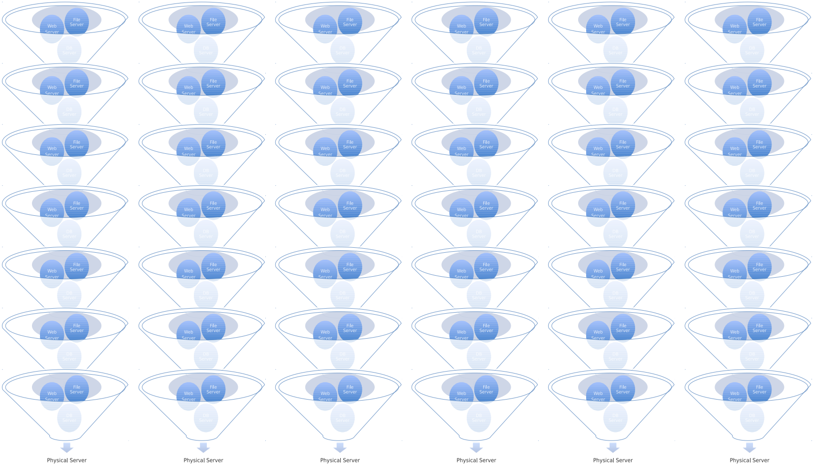
The Benefits of Virtualization



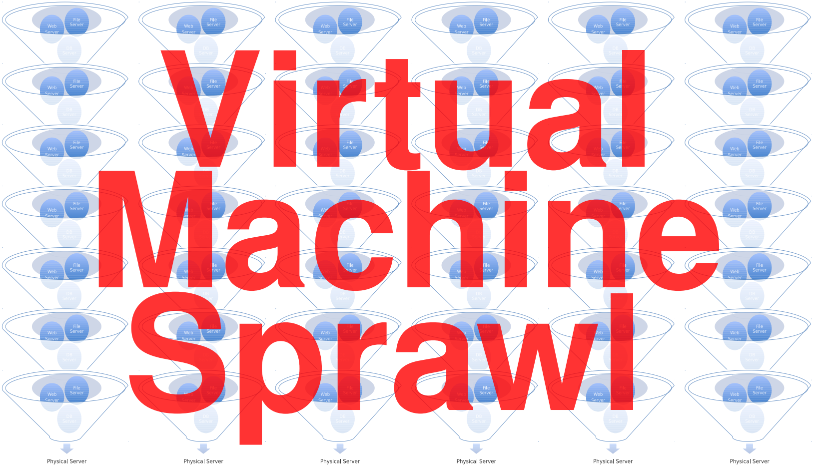
Virtualization in Data Centers



Virtualization in Data Centers



Virtualization in Data Centers



Cloud Computing

Age of the Cloud

- IT is no longer capital-intensive
- Commodity acquired on-demand
- Paid as per usage

Cloud Computing

Age of the Cloud

- IT is no longer capital-intensive
- Commodity acquired on-demand
- Paid as per usage

Emerging problem

- Virtual machine sprawl

Cloud Computing

Age of the Cloud

- IT is no longer capital-intensive
- Commodity acquired on-demand
- Paid as per usage

Emerging problem

- Virtual machine sprawl

Standardization is the key

- Allow services on-demand
- Reduce system management costs in the software stack

Motivating Virtual Machine Clustering

Classifying (possibly) diverse virtualized servers in a cloud into clusters of *similar* virtual machines (VMs) can improve the planning of many system management activities

Outline

Content Similarity

- Refers to data similarity in the raw files
  - ✓ Subset of bytes contained within images are identical

Content Similarity

- Refers to data similarity in the raw files
  - ✓ Subset of bytes contained within images are identical
- Extensively studied in the context of data deduplication

Content Similarity

- Refers to data similarity in the raw files
  - ✓ Subset of bytes contained within images are identical
- Extensively studied in the context of data deduplication

Recent Findings

- Large-scale study of VM in a production IaaS cloud:

Content Similarity

- Refers to data similarity in the raw files
  - ✓ Subset of bytes contained within images are identical
- Extensively studied in the context of data deduplication

Recent Findings

- Large-scale study of VM in a production IaaS cloud:
  - ✓ Images tend to be similar to a small subset of collection.

Content Similarity

- Refers to data similarity in the raw files
  - ✓ Subset of bytes contained within images are identical
- Extensively studied in the context of data deduplication

Recent Findings

- Large-scale study of VM in a production IaaS cloud:
  - ✓ Images tend to be similar to a small subset of collection.
  - ✓ Computing pair-wise similarity is very expensive

Semantic Similarity

- Characterizes the similarity of the software functionality.

Semantic Similarity

- Characterizes the similarity of the software functionality.
- Some examples:
  - ✓ Instances of same application
  - ✓ Different versions of the same application
  - ✓ Different applications with same goal (i.e MySQL and DB2)

Harnessing Image Similarity

System Management Scenarios

- Allocation of servers to system administrators
  - ✓ Administrators can manage up to 80% more servers

Harnessing Image Similarity

System Management Scenarios

- Allocation of servers to system administrators
  - ✓ Administrators can manage up to 80% more servers
- Troubleshooting
  - ✓ Identify servers with similar software stack that respond differently to an update to find and fix possible issues

Harnessing Image Similarity

System Management Scenarios

- Allocation of servers to system administrators
  - ✓ Administrators can manage up to 80% more servers
- Troubleshooting
  - ✓ Identify servers with similar software stack that respond differently to an update to find and fix possible issues
- Placement of virtual machines to hosts
  - ✓ In-memory and storage deduplication

Harnessing Image Similarity

System Management Scenarios

- Allocation of servers to system administrators
  - ✓ Administrators can manage up to 80% more servers
- Troubleshooting
  - ✓ Identify servers with similar software stack that respond differently to an update to find and fix possible issues
- Placement of virtual machines to hosts
  - ✓ In-memory and storage deduplication
- Migration of enterprise applications across data centers
  - ✓ Migration performed in batches or waves
  - ✓ Minimize network transfer and re-configuration costs

Virtual Machine Clustering

Clustering is NP-hard. Various heuristic exist.

Virtual Machine Clustering

Clustering is NP-hard. Various heuristic exist.

k-means

- Popular technique employed in the real world

Virtual Machine Clustering

Clustering is NP-hard. Various heuristic exist.

k-means

- Popular technique employed in the real world
- Each iteration:
  - ✓ Assignment Step - Distance operation (kN)
  - ✓ Update Step - Merge operation ($N - 1$)

Virtual Machine Clustering

Clustering is NP-hard. Various heuristic exist.

k-means

- Popular technique employed in the real world
- Each iteration:
  - ✓ Assignment Step - Distance operation (kN)
  - ✓ Update Step - Merge operation ($N - 1$)
- In practice *Distance* and *Merge* operations are usually very small

Virtual Machine Clustering

Clustering is NP-hard. Various heuristic exist.

k-means

- Popular technique employed in the real world
- Each iteration:
  - ✓ Assignment Step - Distance operation (kN)
  - ✓ Update Step - Merge operation ($N-1$)
- In practice *Distance* and *Merge* operations are usually very small
  - ✓ Problems with 100 dimensions require only about 100 addition and division operations

Virtual Machine Clustering

- *Distance* can be calculated as $1 - SIM(I_i, I_j)$

$$SIM(I_i, I_j) = \frac{wt(I_i \cap I_j)}{wt(I_i \cup I_j)}$$

Virtual Machine Clustering

- *Distance* can be calculated as $1 - SIM(I_i, I_j)$

$$SIM(I_i, I_j) = \frac{wt(I_i \cap I_j)}{wt(I_i \cup I_j)}$$

- *Merge* can be calculated as $(I_i \cup I_j)$

Virtual Machine Clustering

- *Distance* can be calculated as $1 - SIM(I_i, I_j)$

$$SIM(I_i, I_j) = \frac{wt(I_i \cap I_j)}{wt(I_i \cup I_j)}$$

- *Merge* can be calculated as $(I_i \cup I_j)$

Image Size	Similarity (Content)	Merge
8.8 GB	45.5 sec	14.7 sec
12.3 GB	75.2 sec	24.1 sec
13.6 GB	98.5 sec	31.2 sec
16.3 GB	142.3 sec	44.2 sec
19.7 GB	172.2 sec	53.5 sec
22.1 GB	232.7 sec	64.9 sec

Virtual Machine Clustering

- *Distance* can be calculated as $1 - SIM(I_i, I_j)$

$$SIM(I_i, I_j) = \frac{wt(I_i \cap I_j)}{wt(I_i \cup I_j)}$$

- *Merge* can be calculated as $(I_i \cup I_j)$

Image Size	Similarity (Content)	Merge
8.8 GB	45.5 sec	14.7 sec
12.3 GB	75.2 sec	24.1 sec
13.6 GB	98.5 sec	31.2 sec
16.3 GB	142.3 sec	44.2 sec
19.7 GB	172.2 sec	53.5 sec
22.1 GB	232.7 sec	64.9 sec

- A data center with 1000 images would have to perform 1000^3 similarity operations, about 2000 years
- By using in-memory data structures, about 40 years

Approximate Clustering

k-medoids

- k-medoids is a variant of k-means
  - ✓ Restricts the cluster center to be one of the existing points (images)
  - ✓ Pair-wise similarity can be computed in advance
  - ✓ Similarity computation required for all images (N^2)

Approximate Clustering

k-medoids

- k-medoids is a variant of k-means
  - ✓ Restricts the cluster center to be one of the existing points (images)
  - ✓ Pair-wise similarity can be computed in advance
  - ✓ Similarity computation required for all images (N^2)

- A data center with 1000 images would have to perform 1000^2 similarity operations, about 2 years
- By using in-memory data structures, about 15 days

Solution Idea: Asymmetric Clustering

Coriolis' tree-based clustering

- *Coriolis*' clustering approach involves constructing a tree

Solution Idea: Asymmetric Clustering

Coriolis' tree-based clustering

- *Coriolis*' clustering approach involves constructing a tree
  - ✓ The tree is constructed by adding images to it one by one

Solution Idea: Asymmetric Clustering

Coriolis' tree-based clustering

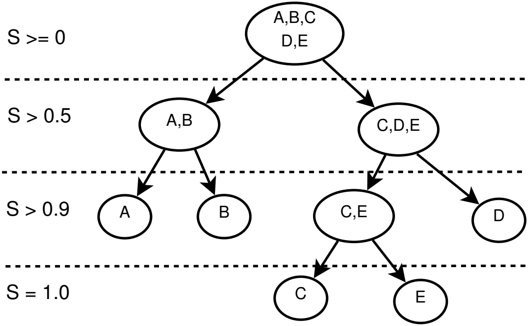
- *Coriolis*' clustering approach involves constructing a tree
  - ✓ The tree is constructed by adding images to it one by one
  - ✓ Each node of the tree is either a cluster of images or a single image

Solution Idea: Asymmetric Clustering

Coriolis' tree-based clustering

- *Coriolis*' clustering approach involves constructing a tree
  - ✓ The tree is constructed by adding images to it one by one
  - ✓ Each node of the tree is either a cluster of images or a single image
  - ✓ Each level in the tree represents a minimum extent of similarity within a node

Coriolis' Tree-based Clustering



Coriolis' tree-based clustering

Two key ideas

- Speed up similarity computation

Coriolis' tree-based clustering

Two key ideas

- Speed up similarity computation
  - ✓ An asymmetric similarity function S: Coverage offered by a larger node B (typically a cluster) to a new node A

Coriolis' tree-based clustering

Two key ideas

- Speed up similarity computation
  - ✓ An asymmetric similarity function S: Coverage offered by a larger node B (typically a cluster) to a new node A

$$S = \frac{wt(A \cap B)}{min(wt(A), wt(B))}$$

Coriolis' tree-based clustering

Two key ideas

- Speed up similarity computation
  - ✓ An asymmetric similarity function S: Coverage offered by a larger node B (typically a cluster) to a new node A

$$S = \frac{wt(A \cap B)}{min(wt(A), wt(B))}$$

- Reuse similarity computations

Coriolis' tree-based clustering

Two key ideas

- Speed up similarity computation
  - ✓ An asymmetric similarity function S: Coverage offered by a larger node B (typically a cluster) to a new node A

$$S = \frac{wt(A \cap B)}{min(wt(A), wt(B))}$$

- Reuse similarity computations
  - ✓ We only compute the similarity of the new image to each children of the nodes where the image has been merged

Coriolis' tree-based clustering

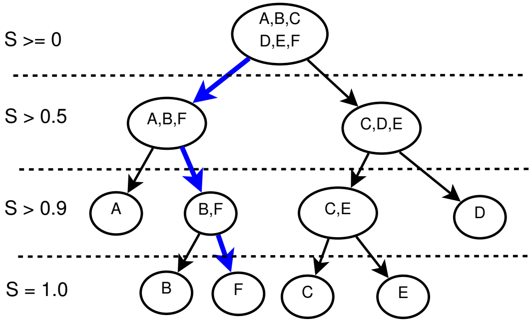
Two key ideas

- Speed up similarity computation
  - ✓ An asymmetric similarity function S: Coverage offered by a larger node B (typically a cluster) to a new node A

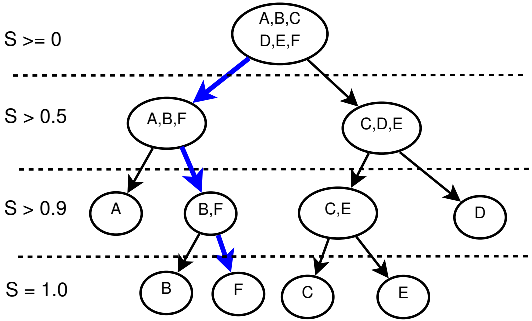
$$S = \frac{wt(A \cap B)}{min(wt(A), wt(B))}$$

- Reuse similarity computations
  - ✓ We only compute the similarity of the new image to each children of the nodes where the image has been merged
  - ✓ *Similarity* and *Merge* operations are proportional to the depth of the tree

Clustering a new Image F



Clustering a new Image F



Which clusters are formed with similarity greater than 0.5?

Scalability Evaluation

Experimental Setup

- We used VM images from 2 production data centers

Scalability Evaluation

Experimental Setup

- We used VM images from 2 production data centers
  - ✓ 9 images from a large-scale enterprise data center at IBM

Scalability Evaluation

Experimental Setup

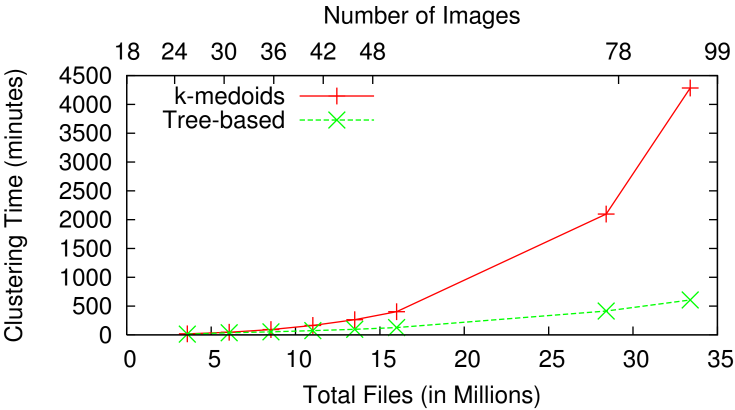
- We used VM images from 2 production data centers
  - ✓ 9 images from a large-scale enterprise data center at IBM
  - ✓ 12 images from the Computer Science department's small scale data center at FIU

Scalability Evaluation

Experimental Setup

- We used VM images from 2 production data centers
  - ✓ 9 images from a large-scale enterprise data center at IBM
  - ✓ 12 images from the Computer Science department's small scale data center at FIU
  - ✓ We randomly sampled files contained in 3 of the 21 images and generated new synthetic images

Scalability of k-medoids and Tree-based Clustering



Related Work

Finding Similar Clusters

- VMFlock: Virtual Machine Co-migration for the Cloud[IEEE/ACM HPDC'11]
  - ✓ Applies standard de-duplication techniques for images
  - ✓ Eliminate raw data duplicates across a given set of VM images
  - ✓ It does not tackle identifying images with high redundancy or leveraging semantic similarity

Conclusions and Future Work

Conclusions

- We described different types of similarity metrics for VMs and their use to aid administrators in their management activities

Conclusions and Future Work

Conclusions

- We described different types of similarity metrics for VMs and their use to aid administrators in their management activities
- We argued that state-of-the-art k-medoids clustering algorithm incurs quadratic complexity infeasible for cloud scale data centers

Conclusions and Future Work

Conclusions

- We described different types of similarity metrics for VMs and their use to aid administrators in their management activities
- We argued that state-of-the-art k-medoids clustering algorithm incurs quadratic complexity infeasible for cloud scale data centers
- We described the *Coriolis* framework and system specifically designed for scalable clustering of VM images while supporting arbitrary similarity metrics

Conclusions and Future Work

Conclusions

- We described different types of similarity metrics for VMs and their use to aid administrators in their management activities
- We argued that state-of-the-art k-medoids clustering algorithm incurs quadratic complexity infeasible for cloud scale data centers
- We described the *Coriolis* framework and system specifically designed for scalable clustering of VM images while supporting arbitrary similarity metrics

Future Work

- Our future work will explore the utility of *Coriolis* for data center administrator allocation, troubleshooting, and large-scale VM migration

Thank you!

(I'll be happy to take questions)