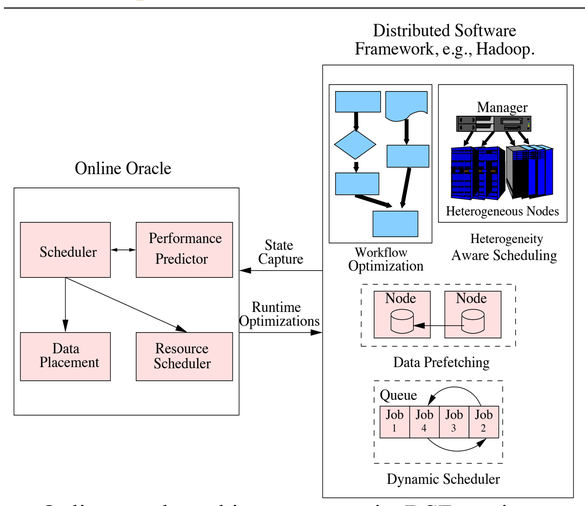
Simulation Driven Tools for Designing Efficient Big Data Systems

PI: Ali R. Butt Graduate Researchers: Y Cheng, Krish KR, H Sim, A Anwar, L Xu, B Wadhwa, A Paul, N Zhao, J Khan

VirginiaTech



Online oracle architecture to assist DSF runtime

Motivation:

• Relates to Computer Systems Research because the new compile-time and runtime optimizations will enable warehouse-scale Distributed Software Frameworks (DSFs).
•Research is motivated by the need to handle the increasing heterogeneity in DSF infrastructure and emerging multi-faceted applications.
•Critical gap to be addressed is making DSFs aware of heterogeneity.
•Vertically advances the field by designing Pythia that integrates compiler-provided information into holistic simulations and drives efficient DSF resource scheduling and management.
•Transformative because Pythia will enable DSFs to efficiently handle heterogeneity in datacenters and support complex emerging applications.

Objective:

•Design a compiler-based application behavior analyzer and optimizer.
•Design an accurate heuristics based DSF performance predictor.
•Design an online oracle to guide efficient resource management in DSFs.

Technical Approach:

Pythia exploits compile time User-Defined Functions analysis and integrated online simulations of DSF runtimes to provide a comprehensive solution for efficient DSF resource management and scheduling.

Outreach and Broader Impacts Plan:

•Pythia will achieve significantly reduced time-to-solutions for mordern data-intensive applications.
•Both graduate and undergraduate students will participate in the project, with special focus on recruiting and mentoring women and minorities.
•Research and Education are integrated by enhancing/creating courses, publishing source codes, and providing online materials for K-12 education.

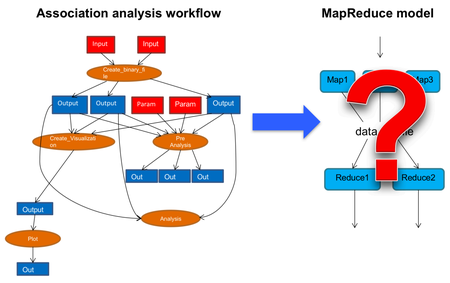
Prior Results, Deliverables:

•Conducted a simulation-based study that identifies the characteristics of different hardware-software configurations in DSFs (CLUSTER 2014).
•Developed prediction-based application placement using software-hardware profiling and characteristics simulations (MASCOTS 2014).

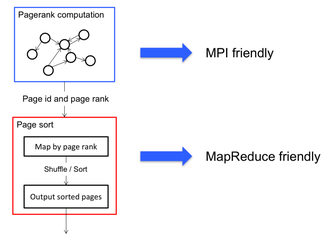
Schedule:

•Y1: Develop compiler-based analysis and optimization techniques for workflow analysis.
•Y2: Extend heuristics-based performance oracle to other state of the art DSFs.
•Y3: Implement and evaluate all components of Pythia.

GERBIL: MPI+YARN [CCGrid'15]



MapReduce programming paradigm is widely adopted due to its ease of use. However, the simplicity of MapReduce is not able to capture complex communications.

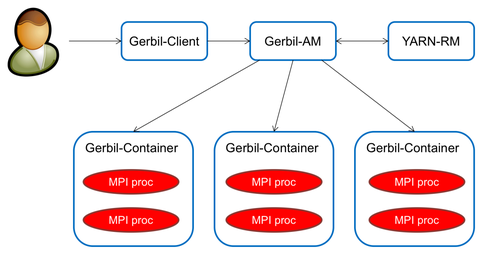


Current setup:
•One cluster per model
•Clusters are connected in a sequence

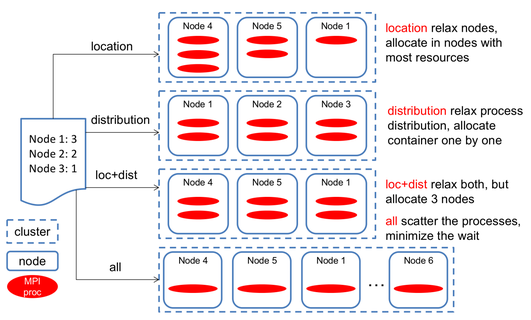
Problems:
•Data transfer overhead
•High maintenance cost
•Low cluster utilization

Emerging complex workflows contain both MPI friendly and MapReduce friendly applications

Gerbil architecture



Resource allocation strategies



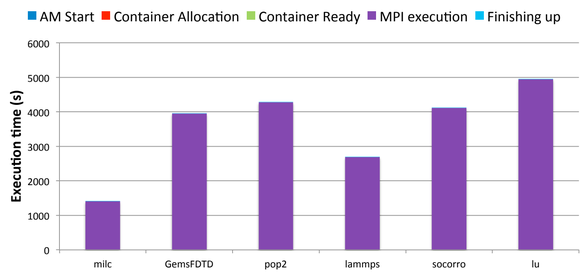
location relax nodes, allocate in nodes with most resources

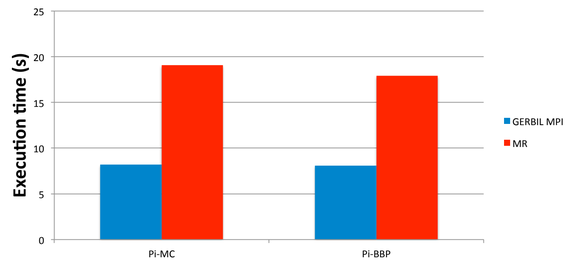
distribution relax process distribution, allocate container one by one

loc+dist relax both, but allocate 3 nodes

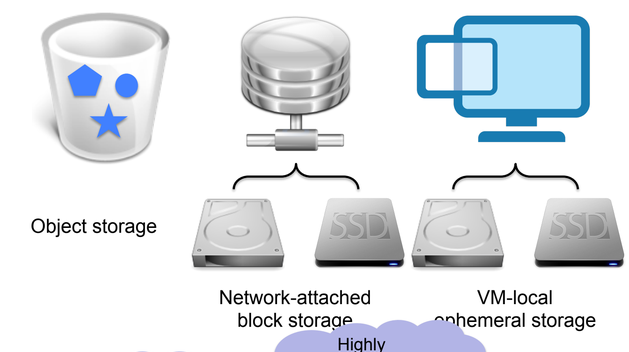
all scatter the processes, minimize the wait

Evaluation



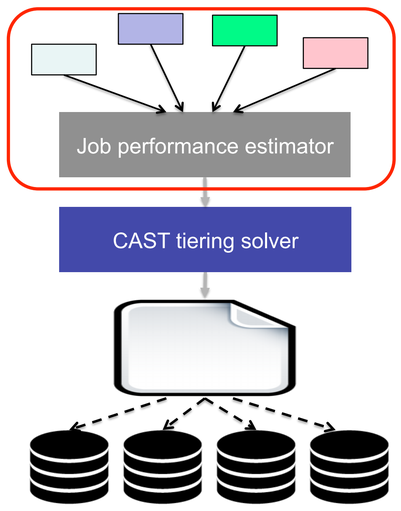


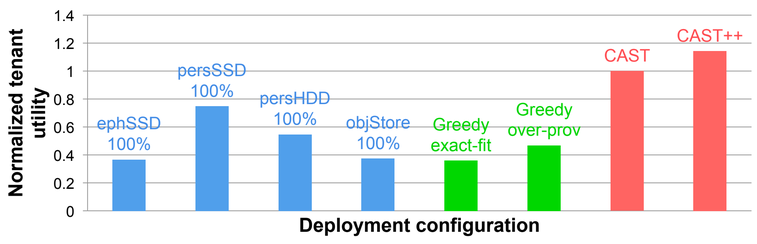
CAST: Cloud Data Analytics Storage Tiering [HPDC'15]



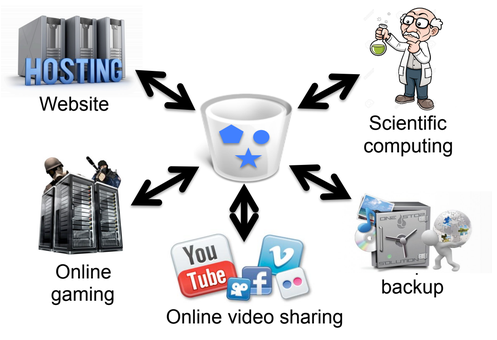
Storage type	Capacity (GB/ volume)	Throughput (MB/sec)	IOPS (4KB)	Cost ($/month)
ephSSD	375	733	100000	0.218×375
persSSD	100	48	3000	0.17×100
	250	118	7500	0.17×250
	500	234	15000	0.17×500
persHDD	100	20	150	0.04×100
	250	45	375	0.04×250
	500	97	750	0.04×500
objStore	N/A	265	550	0.026/GB

ephSSD: VM-local ephemeral SSD, **persSSD**: Network-attached persistent SSD,
persHDD: Network-attached persistent HDD, **objStore**: Google cloud object storage





MOS: Workload-aware Elasticity for Cloud Object Stores [HPDC'16]



Using monolithic object store to serve variety of workloads is not the right approach!

Key Insights

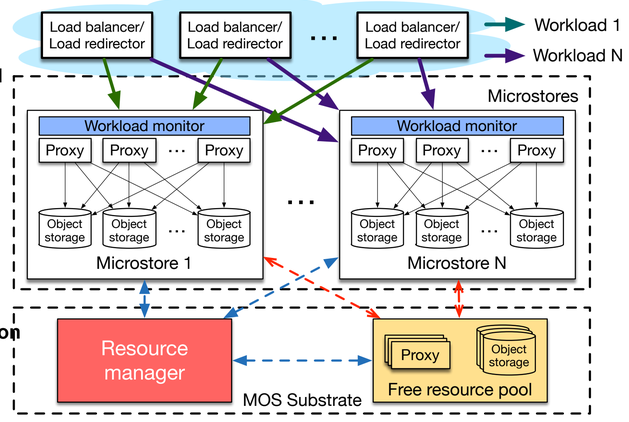
- Cloud object store workloads can be classified based on the size of the objects in their workloads
- When multiple tenants run workloads with drastically different behaviors, they compete for the object store resources with each other

Applications have different service level requirements, e.g., average latency per request, queries per second (QPS), and effective data transfer bandwidth (MB/s)

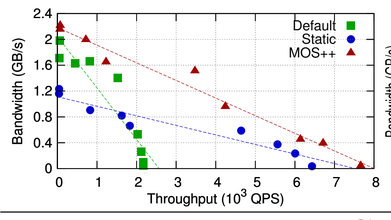
Observations

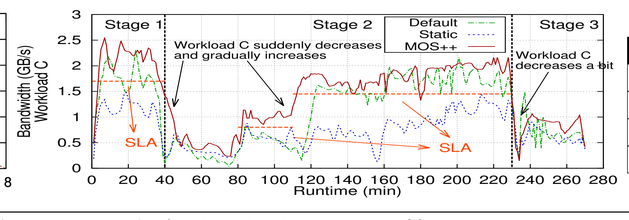
O1: Object size distribution is a key factor for classifying workload characteristics
O2: CPU capacity of proxy servers is the first-priority resource for small-object intensive workloads
O3: proxyCores = storageNodes ∗ coresPerStorageNode
O4: BWproxies = storageNodes ∗ BWstorageNode
O5: Network bandwidth plays a critical role in the performance of large-object intensive workloads
O6: A faster network cannot effectively improve QPS for small-object intensive workload
O7: For large-object intensive workloads, we have to collectively consider the network bandwidth limits and the storage configuration

MOS Design



Evaluation





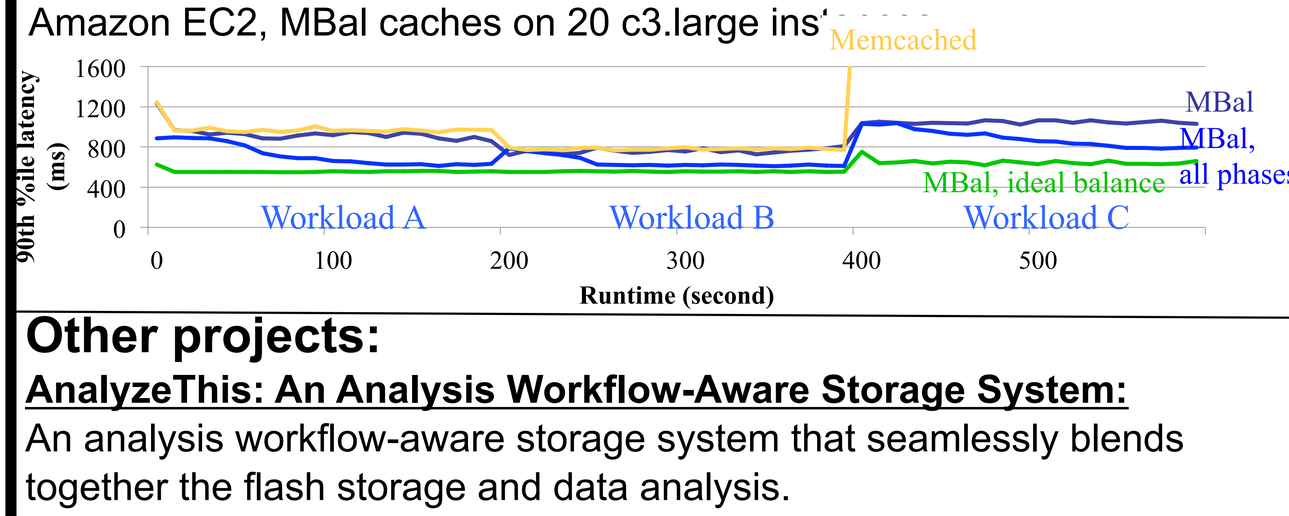
Summary

Workload	MOS	MOS++
A	4444 QPS	4994 QPS
B	3828 QPS	4429 QPS
C	2 GB/s	2.3 GB/s
D	1.6 GB/s	1.9 GB/s

MBal: A Load Balanced Memory Cache Tier [EuroSys'15]

MBal synthesizes different techniques and combines them into **a novel holistic system** that improves in-memory caching performance.

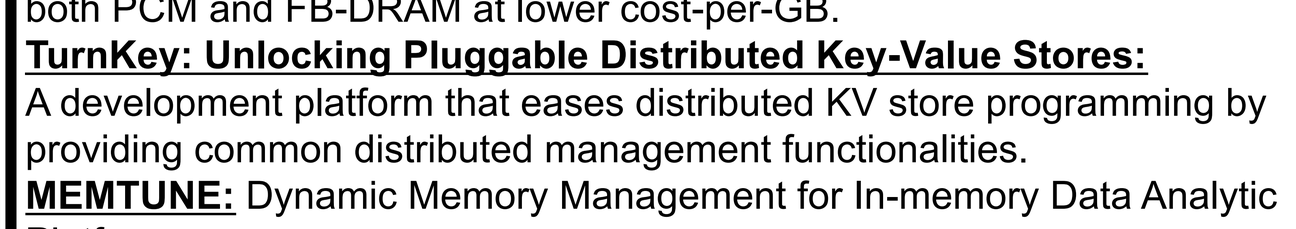
MBal cache



Cachelet design

•Encapsulates/isolates resources
•Avoids lock contention
•Enables fine-grained control of flexible load balancing

MBal performance



Other projects:

AnalyzeThis: An Analysis Workflow-Aware Storage System:
An analysis workflow-aware storage system that seamlessly blends together the flash storage and data analysis.

Multi-tiered Buffer Cache for Persistent Memory Devices:
A tiered caching system for combining PM devices to achieve the best of both PCM and FB-DRAM at lower cost-per-GB.

TurnKey: Unlocking Pluggable Distributed Key-Value Stores:
A development platform that eases distributed KV store programming by providing common distributed management functionalities.

MEMTUNE: Dynamic Memory Management for In-memory Data Analytic Platforms

DUX: an application-attuned dynamic data management system for data processing frameworks

Contact: Ali R. Butt, http://research.cs.vt.edu/dssl/, butta@cs.vt.edu

This research is supported in part by NSF awards CNS-1405697, CNS-1615411, and CNS-1565314.