BlueSky: A Cloud-Backed File System for the Enterprise

Michael Vrable Stefan Savage Geoffrey M. Voelker

University of California, San Diego
Computer Science and Engineering Department

February 16, 2012

Computing Services for the Enterprise

- Our work is focused primarily on small/medium-sized organizations
- These organizations run a number of computing services, such as e-mail and shared file systems
- Often brings significant cost:
  - Purchasing hardware
  - Operating hardware
  - Managing services
- Outsourcing these services to the cloud offers the possibility to lower costs

. . . Migrated to the Cloud

Some services are already migrating to the cloud. . .

Gmail™

. . . Migrated to the Cloud

Some services are already migrating to the cloud. . .

Network file systems have not yet migrated, but still have potential benefits:

- File system size entirely elastic: simpler provisioning
- Cloud provides durability for file system data
- Hardware reliability less important
- Integration with cloud backup

We build and analyze a prototype system, **BlueSky**, to investigate how to do so

Cloud Computing Offerings

Spectrum of service models:

- Software-as-a-Service: Complete integrated service from a provider

salesforce SOFTWARE Gmail™

Cloud Computing Offerings

Spectrum of service models:

- Software-as-a-Service: Complete integrated service from a provider

  salesforce SOFTWARE   Gmail™

- Platform/Infrastructure-as-a-Service: Building blocks for custom applications

  amazon web services™   Windows Azure   the rackspace cloud

Cloud Computing Offerings

Spectrum of service models:

- Software-as-a-Service: Complete integrated service from a provider

  salesforce SOFTWARE Gmail™

- Platform/Infrastructure-as-a-Service: Building blocks for custom applications

  amazon web services™ Windows Azure™ the rackspace cloud

In both cases:

- Infrastructure moved within network
- Reduce/eliminate need for hardware maintenance
- Reduce need for ahead-of-time capacity planning

Cloud Computing Offerings

Spectrum of service models:

- Software-as-a-Service: Complete integrated service from a provider

salesforce Gmail

- Platform/Infrastructure-as-a-Service: Building blocks for custom applications

amazon web services Windows Azure the rackspace cloud

In both cases:

- Infrastructure moved within network
- Reduce/eliminate need for hardware maintenance
- Reduce need for ahead-of-time capacity planning

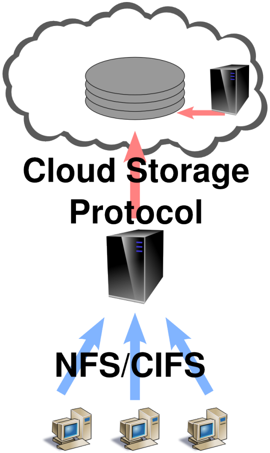
SaaS: Easy to set up
PaaS/IaaS: More choice among service providers, potentially lower cost

Challenges

Cloud storage (e.g., Amazon S3) acts much like another level in the storage hierarchy but brings new design constraints:

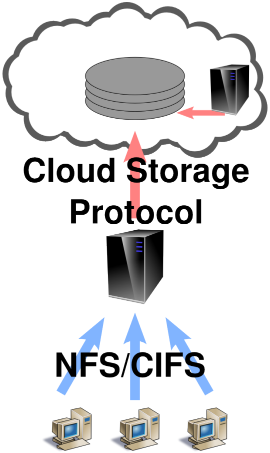
- New interface
  - Only supports writing complete objects
  - Does support random read access
- Performance
  - High latency from network round trips
  - Random access adds little penalty
- Security
  - Data privacy is a concern
- Cost
  - Cost is very explicit
  - Unlimited capacity, but need to delete to save money

BlueSky: Approach



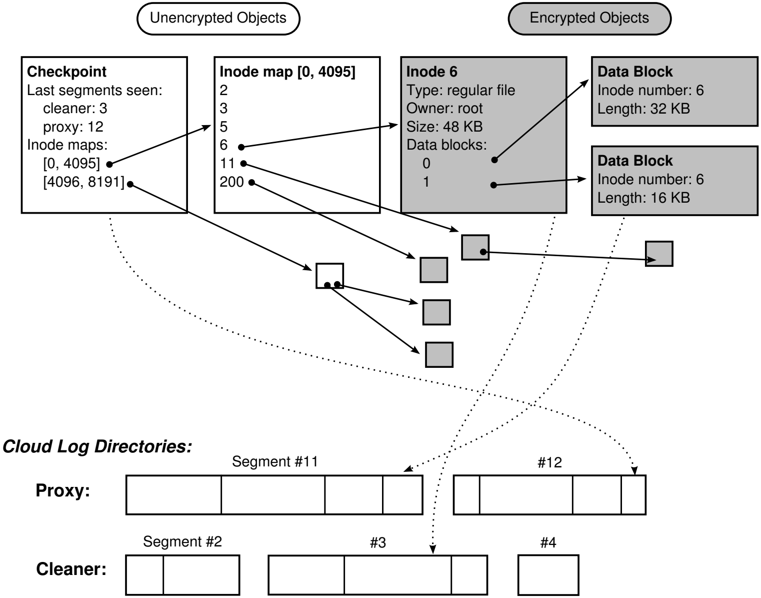
- For ease of deployment, do not change software stack on clients
  - Clients simply pointed at a new server, continue to speak NFS/CIFS
- Deploy a local proxy to translate requests before sending to the cloud
  - Provides lower-latency responses to clients when possible by caching data
  - Implements write-back caching
  - Encrypts data before storage to cloud for confidentiality

BlueSky: Approach

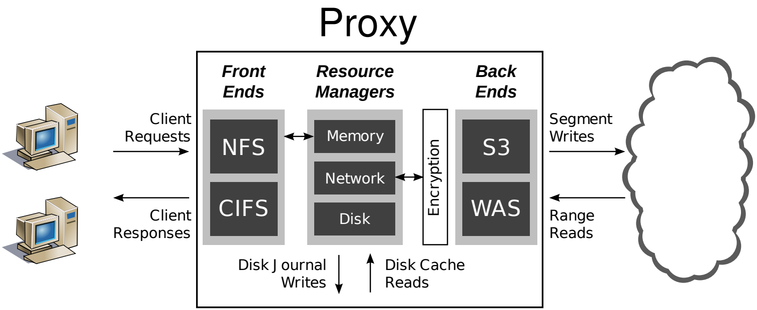


- BlueSky adopts a log-structured design
  - Each log segment uploaded all at once
  - Random access allowed for downloads
- Log cleaner can be run in the cloud (e.g., on Amazon EC2) for faster, cheaper access to storage
  - Log cleaner can run concurrently with active proxy
  - Cleaner not given full access to file system data

File System Design



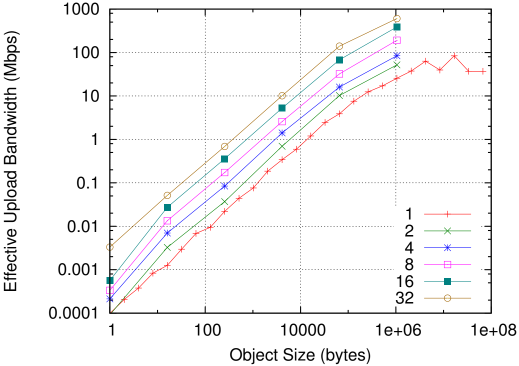
Architecture



- Proxy internally buffers updates briefly in memory
- File system updates are serialized and journaled to local disk
- File system is periodically checkpointed: log items are aggregated into segments and stored to cloud
- On cache miss, log items fetched back from cloud and stored on local disk

Cloud Storage Performance

- We are assuming that users will have fast connectivity to cloud providers (if not now, then in the near future)
- Latency is a fundamental problem (unless cloud data centers built near to customers)



- Network RTT: 30 ms to standard (US-East) S3 region, 12 ms to US-West region
- Proxy can fully utilize bandwidth to cloud
- Results argue for larger objects, parallel uploads

Application Performance

Simple benchmark: unpack Linux kernel sources, checksum kernel sources, compile a kernel

	Unpack	*Check*	*Compile*
	(write)	(read)	(R/W)
Local NFS server	10:50	0:26	4:23
NFS server in EC2			
BlueSky/S3-West			
warm proxy cache			
cold proxy cache			
full segment prefetch			
BlueSky/S3-East			
warm proxy			
cold proxy cache			
full segment prefetch			

Application Performance

Simple benchmark: unpack Linux kernel sources, checksum kernel sources, compile a kernel

	Unpack	*Check*	*Compile*
	(write)	(read)	(R/W)
Local NFS server	10:50	0:26	4:23
NFS server in EC2	65:39	26:26	74:11
BlueSky/S3-West			
warm proxy cache			
cold proxy cache			
full segment prefetch			
BlueSky/S3-East			
warm proxy			
cold proxy cache			
full segment prefetch			

Application Performance

Simple benchmark: unpack Linux kernel sources, checksum kernel sources, compile a kernel

	Unpack	*Check*	*Compile*
	(write)	(read)	(R/W)
Local NFS server	10:50	0:26	4:23
NFS server in EC2	65:39	26:26	74:11
BlueSky/S3-West			
warm proxy cache	5:10	0:33	5:50
cold proxy cache			
full segment prefetch			
BlueSky/S3-East			
warm proxy			
cold proxy cache			
full segment prefetch			

Application Performance

Simple benchmark: unpack Linux kernel sources, checksum kernel sources, compile a kernel

	Unpack	*Check*	*Compile*
	(write)	(read)	(R/W)
Local NFS server	10:50	0:26	4:23
NFS server in EC2	65:39	26:26	74:11
BlueSky/S3-West			
warm proxy cache	5:10	0:33	5:50
cold proxy cache		26:12	7:10
full segment prefetch			
BlueSky/S3-East			
warm proxy			
cold proxy cache			
full segment prefetch			

Application Performance

Simple benchmark: unpack Linux kernel sources, checksum kernel sources, compile a kernel

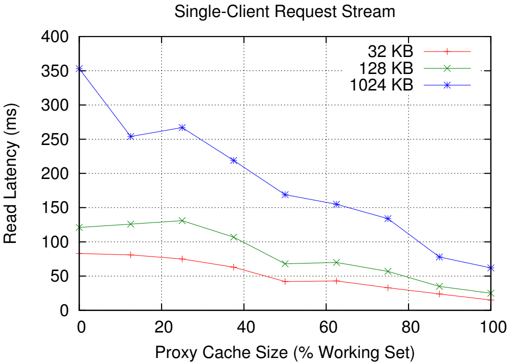
	Unpack (write)	*Check* (read)	*Compile* (R/W)
Local NFS server	10:50	0:26	4:23
NFS server in EC2	65:39	26:26	74:11
BlueSky/S3-West			
warm proxy cache	5:10	0:33	5:50
cold proxy cache		26:12	7:10
full segment prefetch		1:49	6:45
BlueSky/S3-East			
warm proxy			
cold proxy cache			
full segment prefetch			

Application Performance

Simple benchmark: unpack Linux kernel sources, checksum kernel sources, compile a kernel

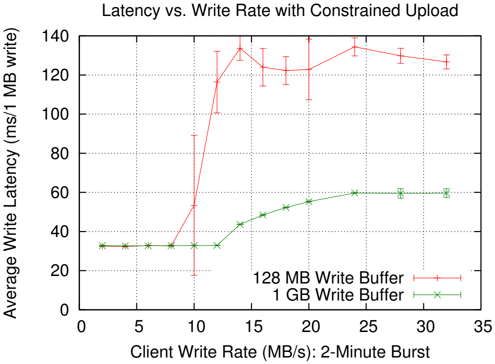
	Unpack	*Check*	*Compile*
	(write)	(read)	(R/W)
Local NFS server	10:50	0:26	4:23
NFS server in EC2	65:39	26:26	74:11
BlueSky/S3-West			
warm proxy cache	5:10	0:33	5:50
cold proxy cache		26:12	7:10
full segment prefetch		1:49	6:45
BlueSky/S3-East			
warm proxy	5:08	0:35	5:53
cold proxy cache		57:26	8:35
full segment prefetch		3:50	8:07

Read Performance Microbenchmark



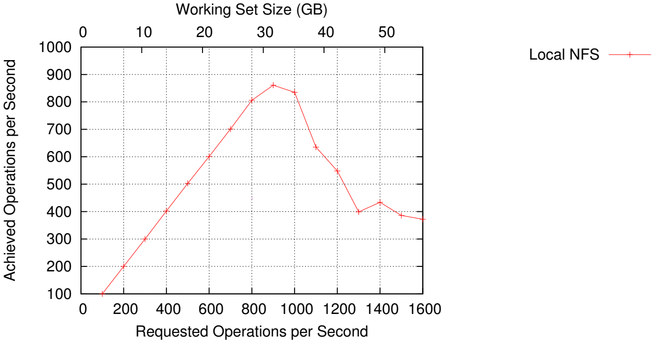
- Read performance depends on working set/cache size ratio
- At 100% hit rate, comparable to local NFS server
- Even at 50% hit rate, latency within about 2× to 3× of local case

Write Performance Microbenchmark



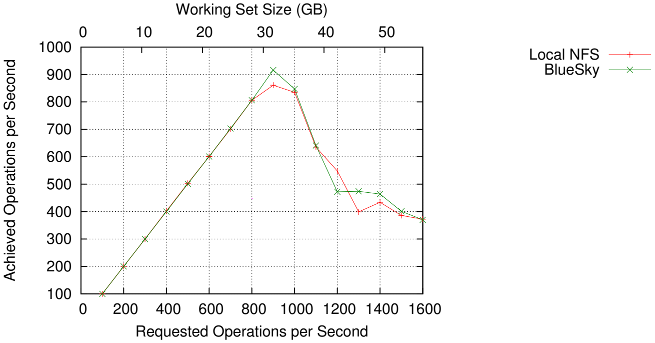
- Configure network to constrain bandwidth to cloud at 100 Mbps
- Write performance: similar to local disk, unless write rate exceeds cloud bandwidth and write-back cache fills

Aggregate Performance: SPECsfs2008



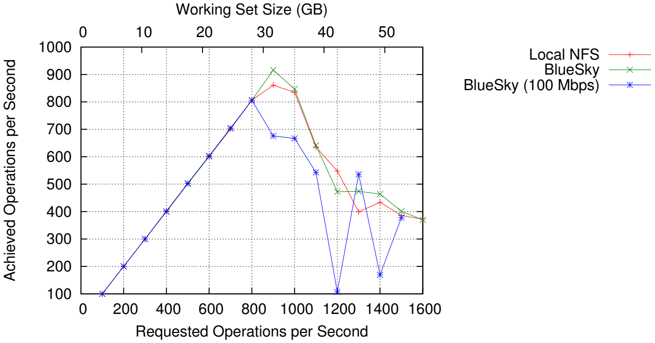
- Models a richer workload mix

Aggregate Performance: SPECsfs2008



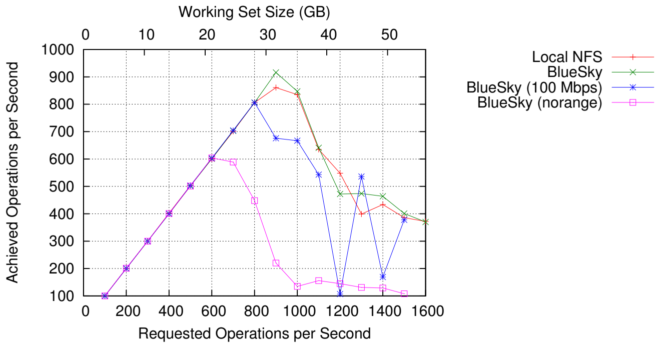
- Models a richer workload mix
- BlueSky is comparable to local NFS (as before, slight advantage on writes from log-structured design)

Aggregate Performance: SPECsfs2008



- Models a richer workload mix
- BlueSky is comparable to local NFS (as before, slight advantage on writes from log-structured design)
- Performance is less predictable with a constrained network link

Aggregate Performance: SPECsfs2008



- Models a richer workload mix
- BlueSky is comparable to local NFS (as before, slight advantage on writes from log-structured design)
- Performance is less predictable with a constrained network link
- Fetching full segments is a big loss with mostly random access

Monetary Cost: SPECsfs2008

Normalized cost: cost per million SPECsfs operations
(for S3 prices: $0.12/GB download, $0.01/1000–10000 ops)

	Down	*Op*	*Total*	*(Up)*
Log-structured baseline	$0.18	$0.09	$0.27	$0.56
No aggregation	0.17	2.91	3.08	0.56
Full segment downloads	25.11	0.09	25.20	1.00

- Log-structured design minimizes cost for cloud storage operations
- Support for random access on reads (byte-range request) needed for low cost
- Storage cost also an important consideration, but less sensitive to system design

Conclusions

- BlueSky is a prototype file server backed by cloud storage
- Prototype supports multiple client protocols (NFS, CIFS) and storage backends (Amazon S3, Windows Azure)
- Allows clients to transparently move to cloud-backed storage
- Performance comparable to local storage when most access hits in cache
- Design is informed by cost models of current cloud providers