High-Performance Big Data Management Across Cloud Data Centers

Radu Tudoran

PhD Advisors

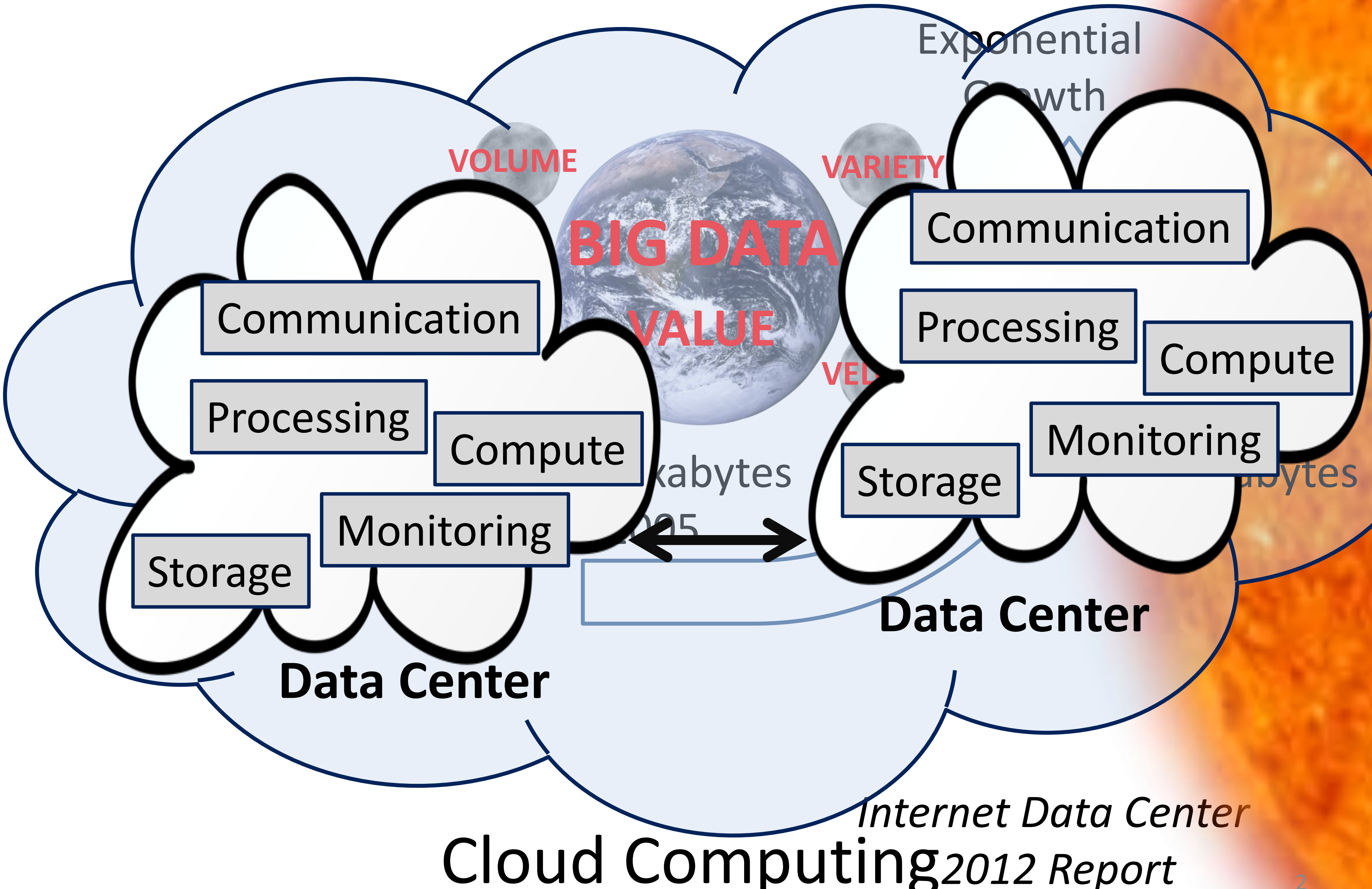
Gabriel Antoniu INRIA

Luc Bougé ENS Rennes

KerData research team

IRISA/INRIA Rennes

Doctoral Work: Context



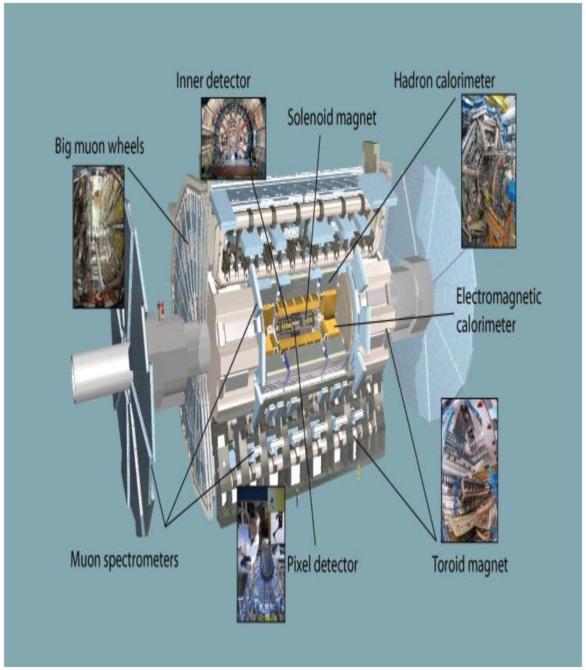
Internet Data Center 2012 Report

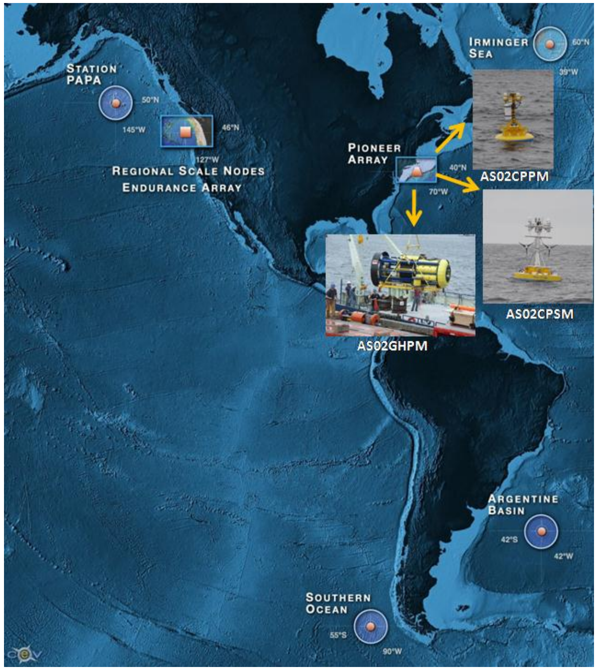
Geographically-Distributed Processing

- CERN ATLAS
- PB of data distributed for storage across multiple institutions

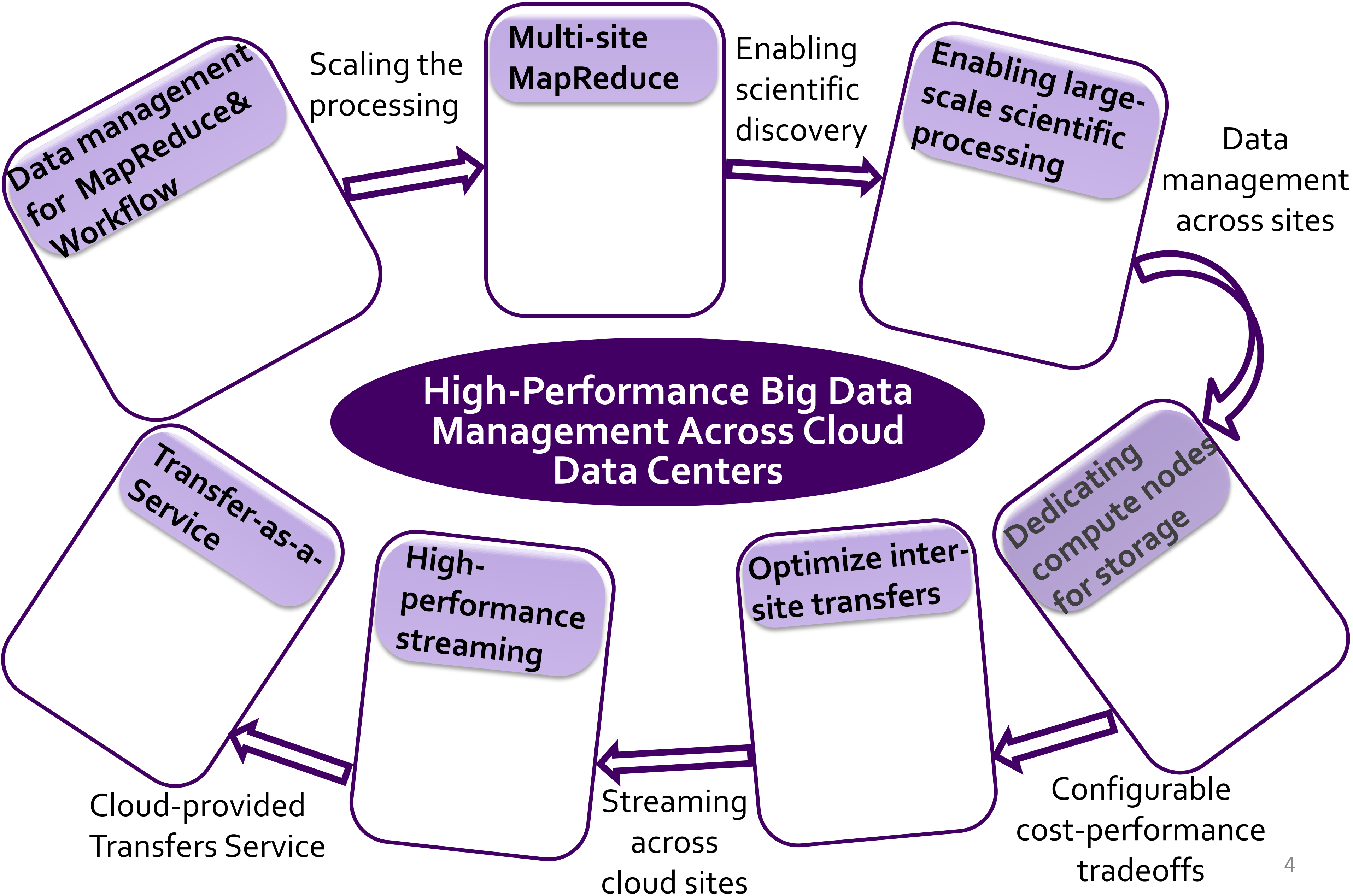
- Ocean Observatory
- Data sources located in geographically distant regions

- Large IT Web-Services
- Data processing exceeds site limits

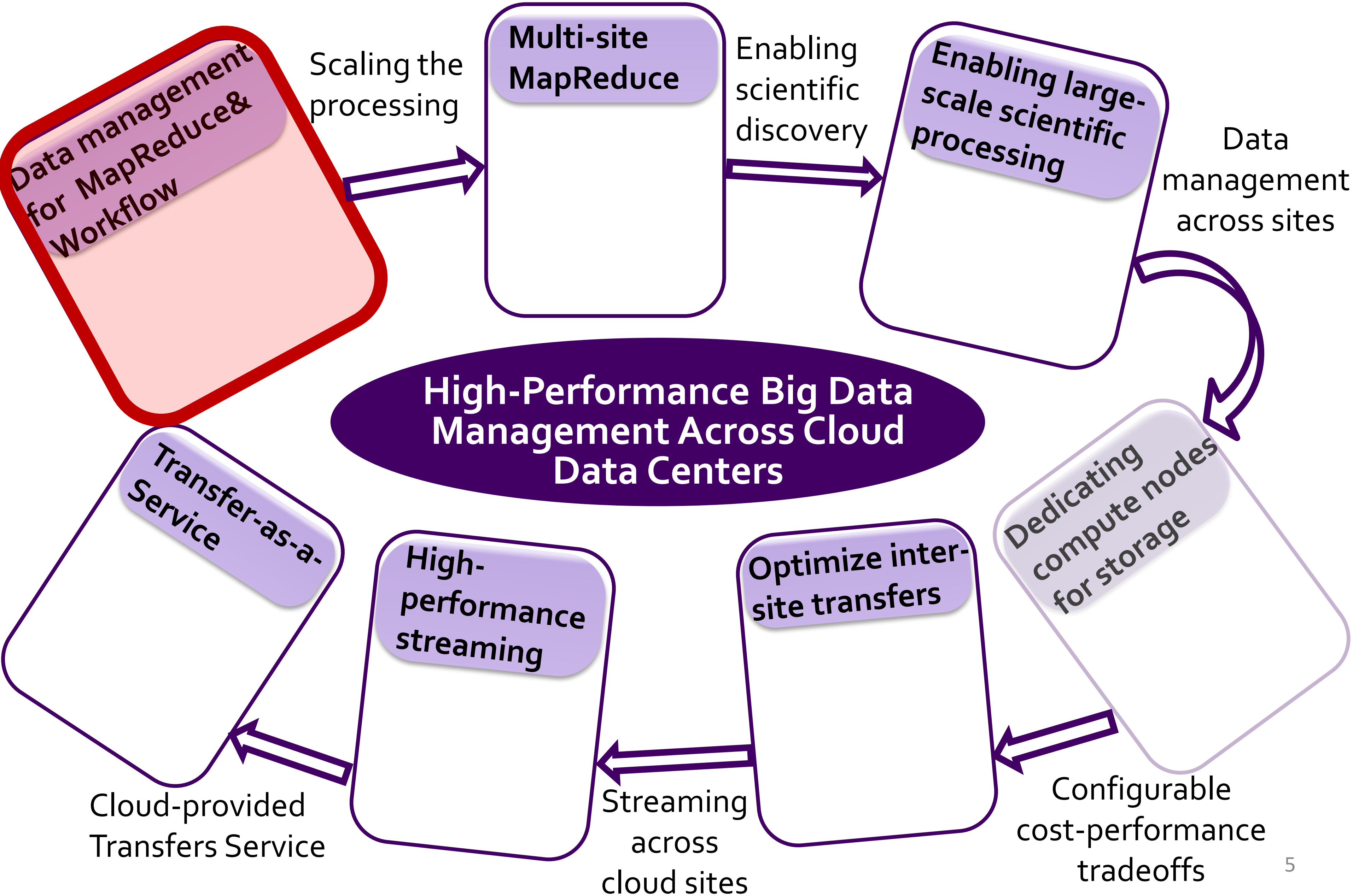




Doctoral Work in a Nutshell



Doctoral Work in a Nutshell



A Big Data Case Study: The A-Brain Application

Image data → dimension $n_{voxels} \times n_{subjects}$

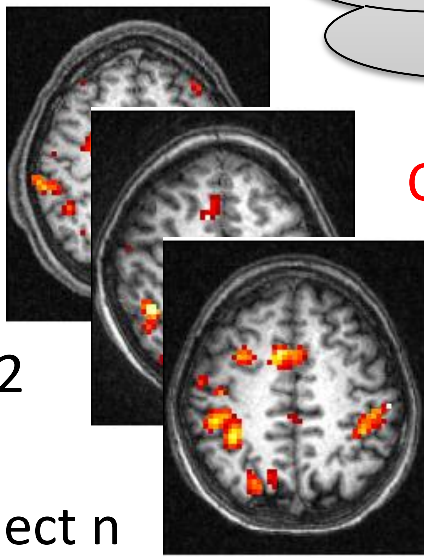
Value: find the correlation between brain markers and genetic data in order to understand the behavioral variability and diseases

Subject 1

Subject 2

…

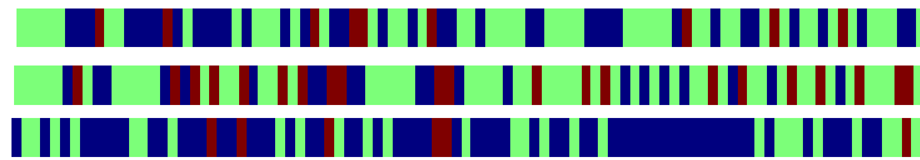
Subject n



Correlations ?

Genetic data → dimension $n_{snps} \times n_{subjects}$

SNP data



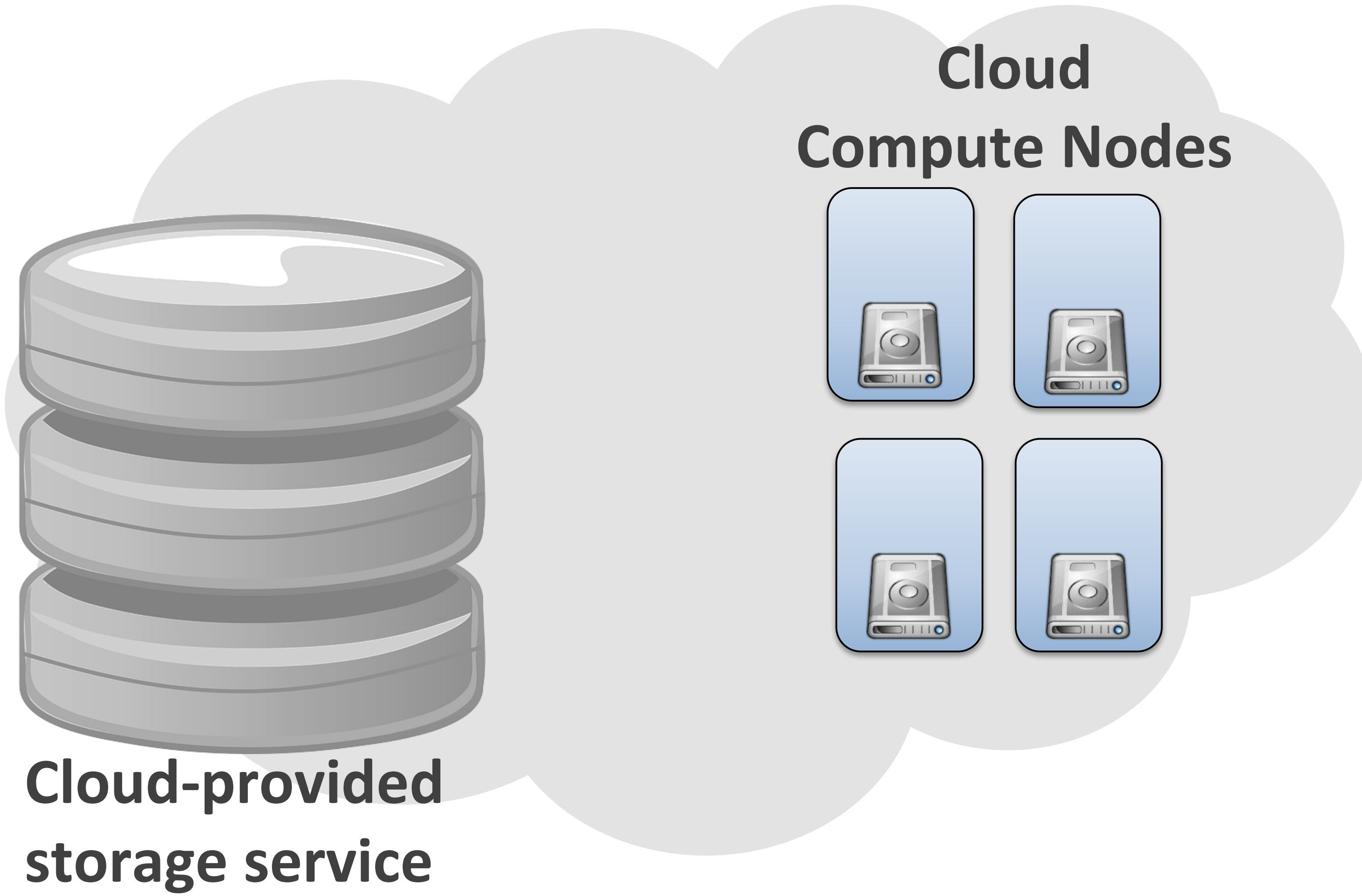
Variety	Veracity
Multi-modal joint analysis Data contains outliers from acquisition process	Biologically significant results and false detection control requires 10^4 permutations

Volume	Velocity
$n_{voxels} = 10^6$ $n_{snps} = 10^6$ $n_{subjects} = 10^3$ Data space potentially reaches TB to PB level	…not the case … But other examples are coming in a few slides

Microsoft Research - Inria
JOINT CENTRE

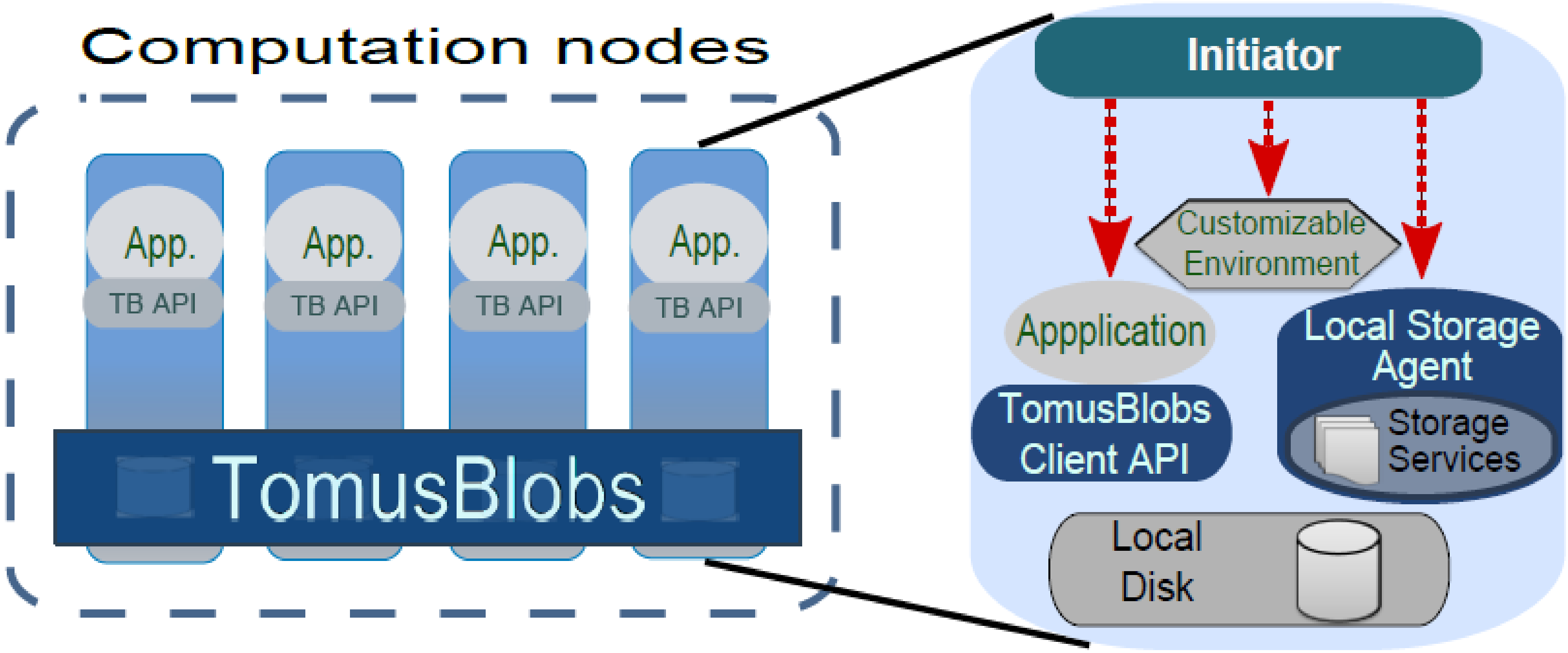
PARIETAL

Data Management on Public Clouds



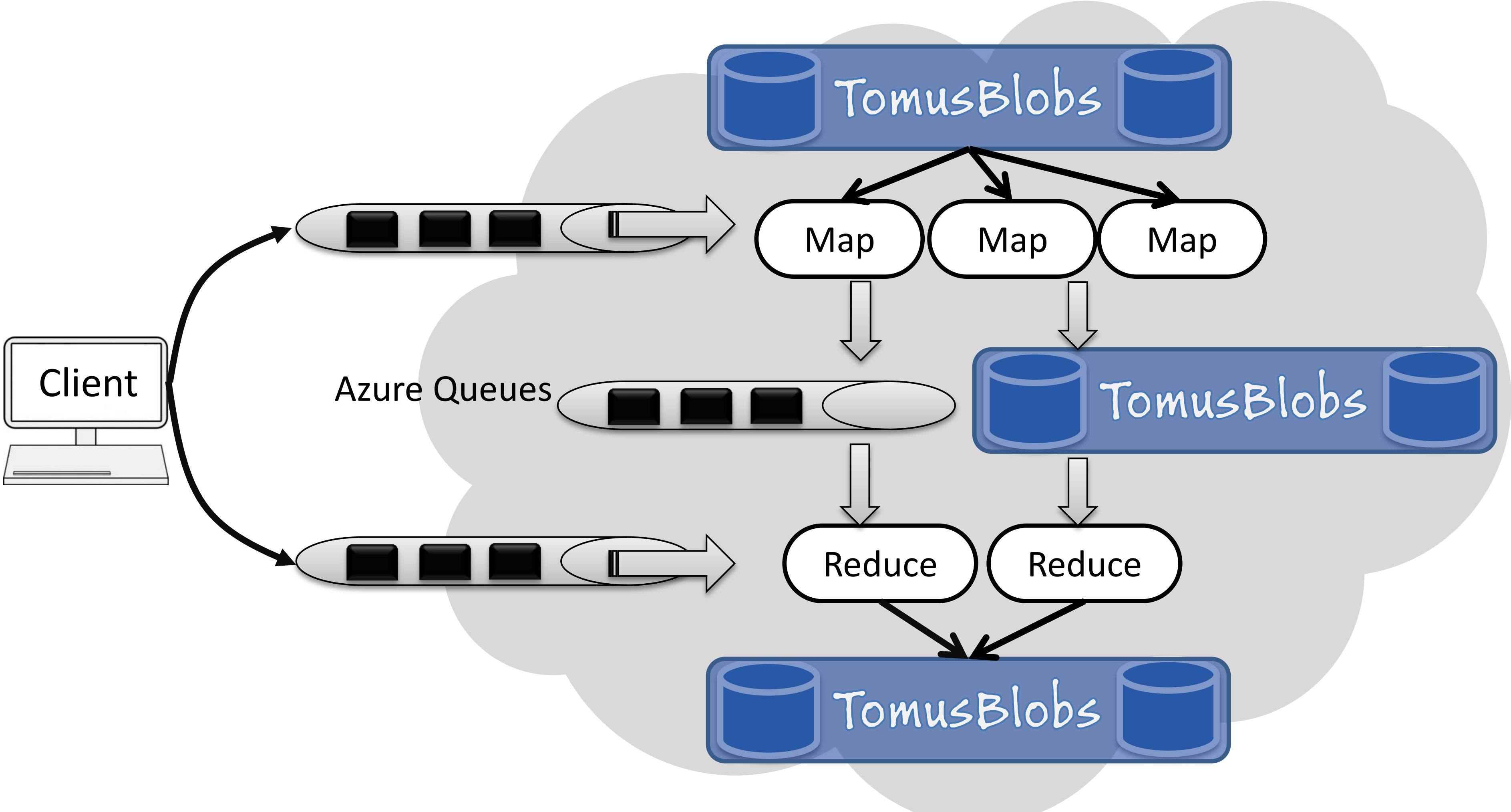
How about **data locality**?

Our approach: TomusBlobs



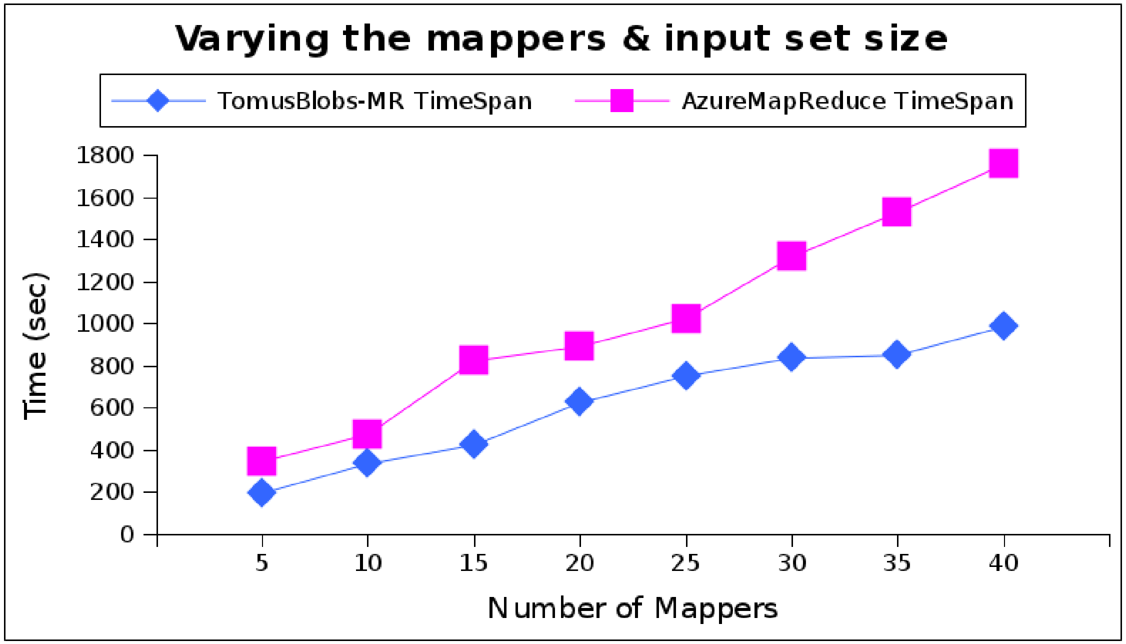
- Collocate computation and data in PaaS clouds by federating the virtual disk of compute nodes
- Self-configuration, automatic deployment and scaling of the data management system
- Apply to **MapReduce** and **Workflow** processing

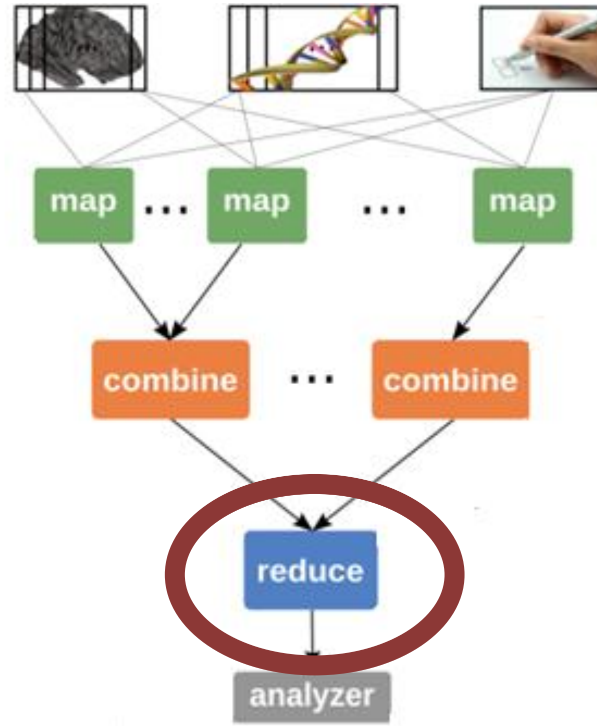
Leveraging TomusBlobs for MapReduce Processing



- New MapReduce prototype (no Hadoop at that point on Azure)
- Adopt BlobSeer as storage backend

Initial A-Brain Experimentation

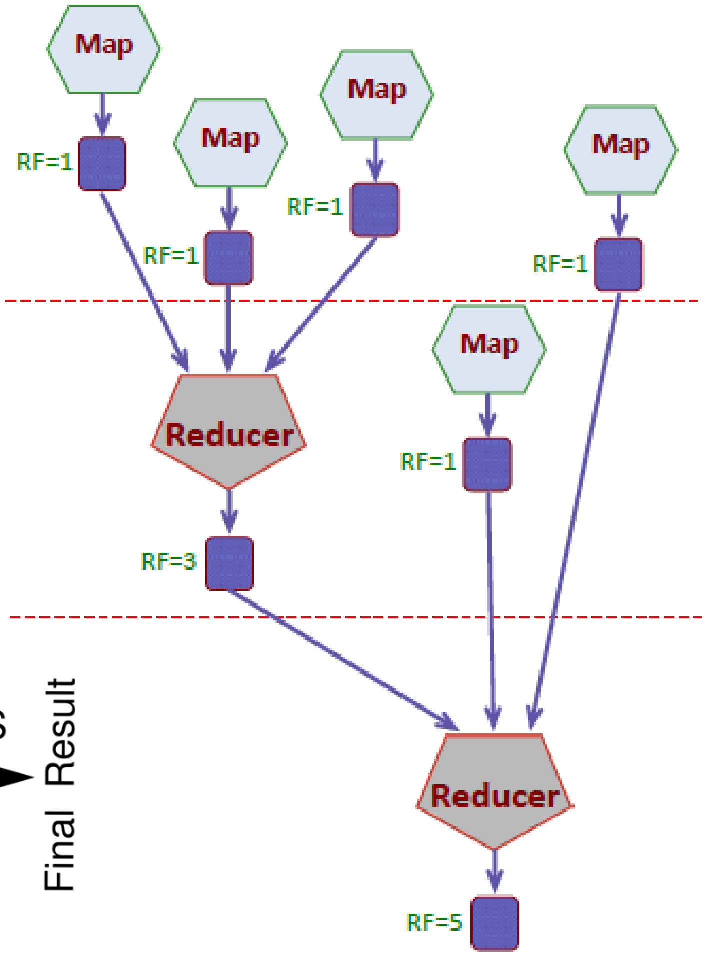


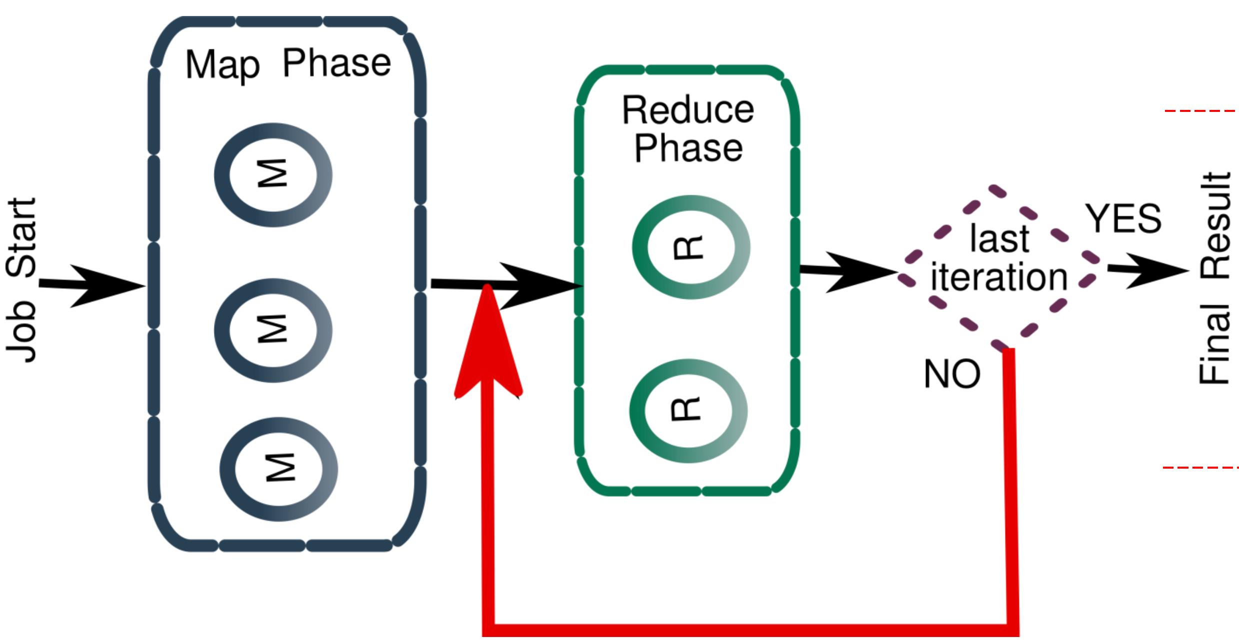


- **Scenario**: 100 nodes deployment on Azure
- Comparison with an Azure Blobs based MapReduce
- TomusBlobs is 3x-4x faster than the cloud remote storage

Beyond MapReduce: Map-IterativeReduce

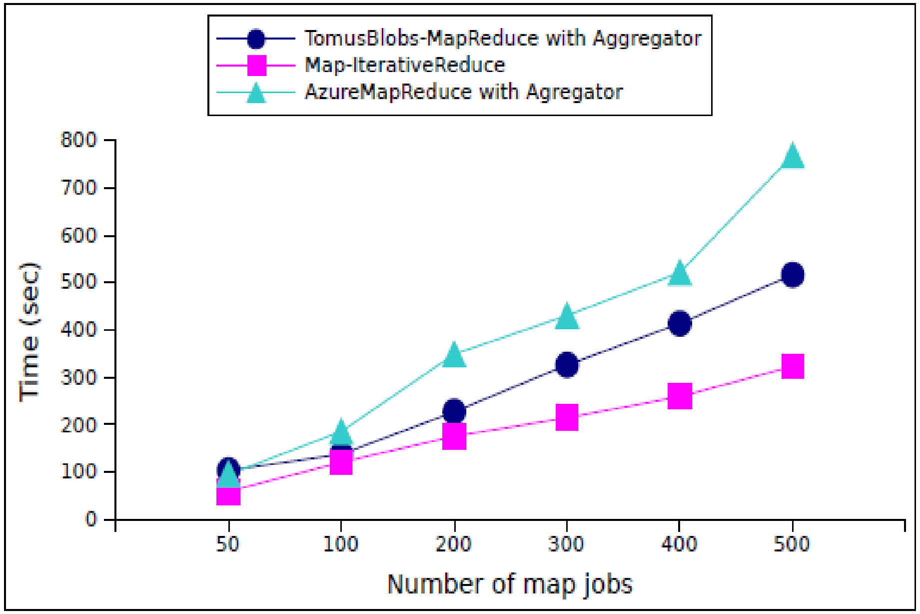
- Unique result with parallel reduction
- No central control entity
- No synchronization barrier





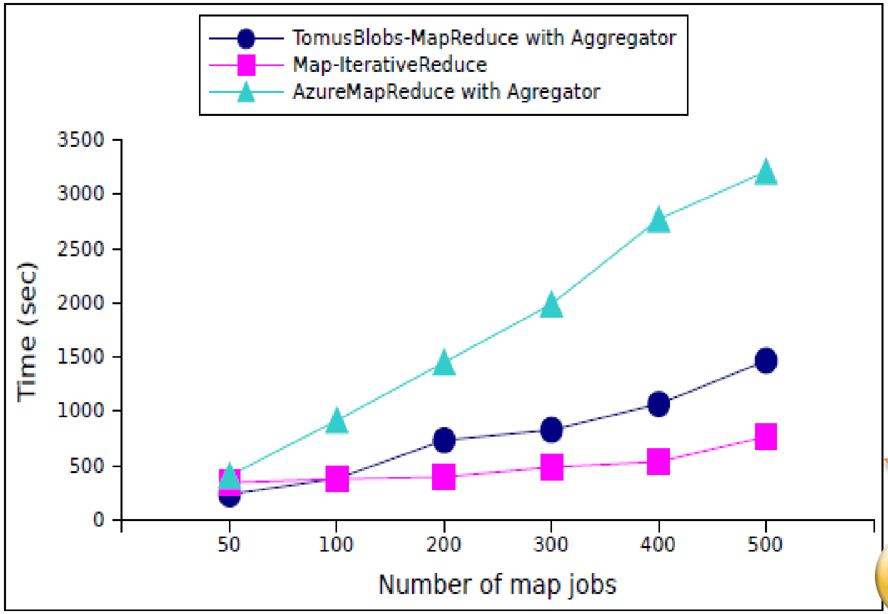
The Efficiency of Full-Reduction

The Most Frequent Words benchmark



Data set 3.2 GB to 32 GB

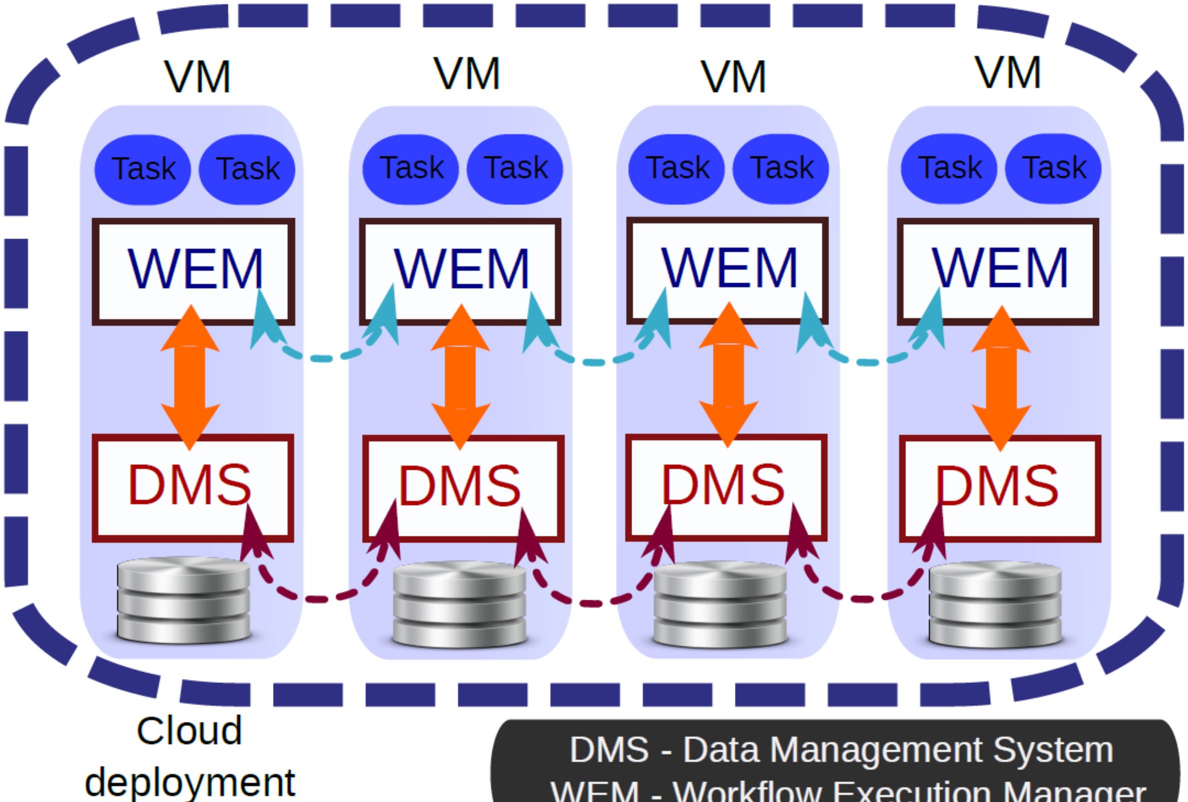
A-Brain initial experimentation



Data set 5 GB to 50 GB

- **Experimental Setup:** 200 nodes deployment on Azure
- Map-IterativeReduce reduces the execution timespan to half

TomusBlobs for Workflow Processing

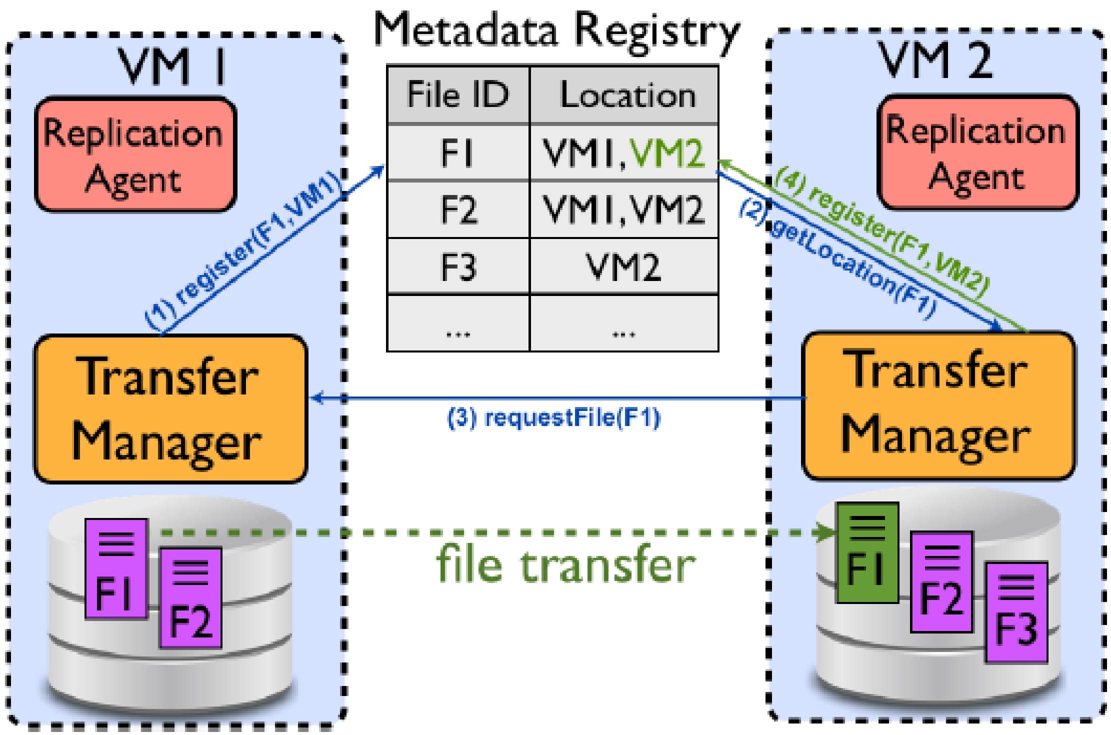


DMS - Data Management System
WEM - Workflow Execution Manager

Exploit workflow specificities:

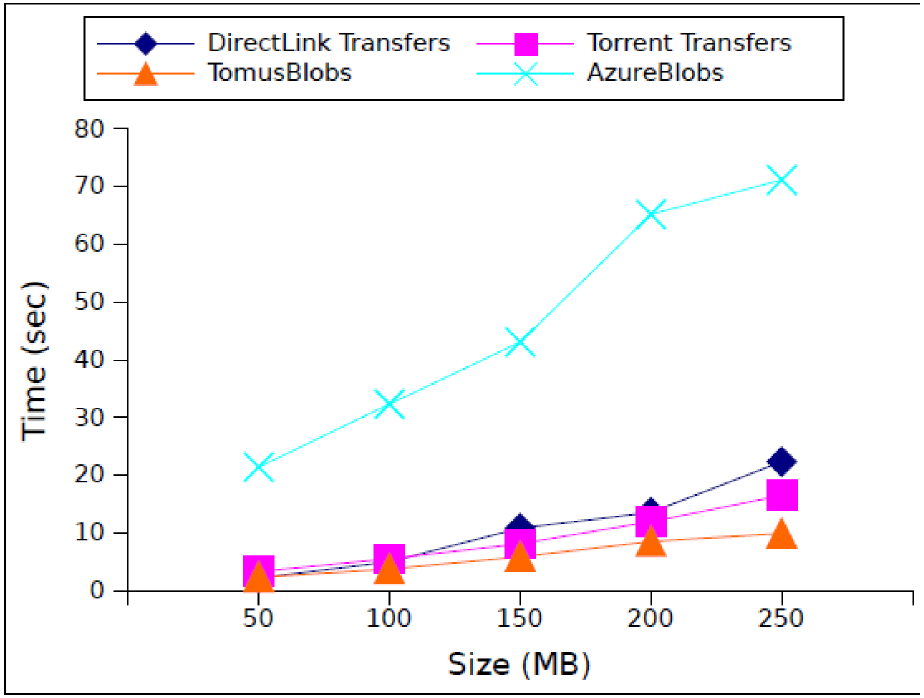
- Data access patterns
- File manipulation
- Batch processing

TomusBlobs for Workflow Processing

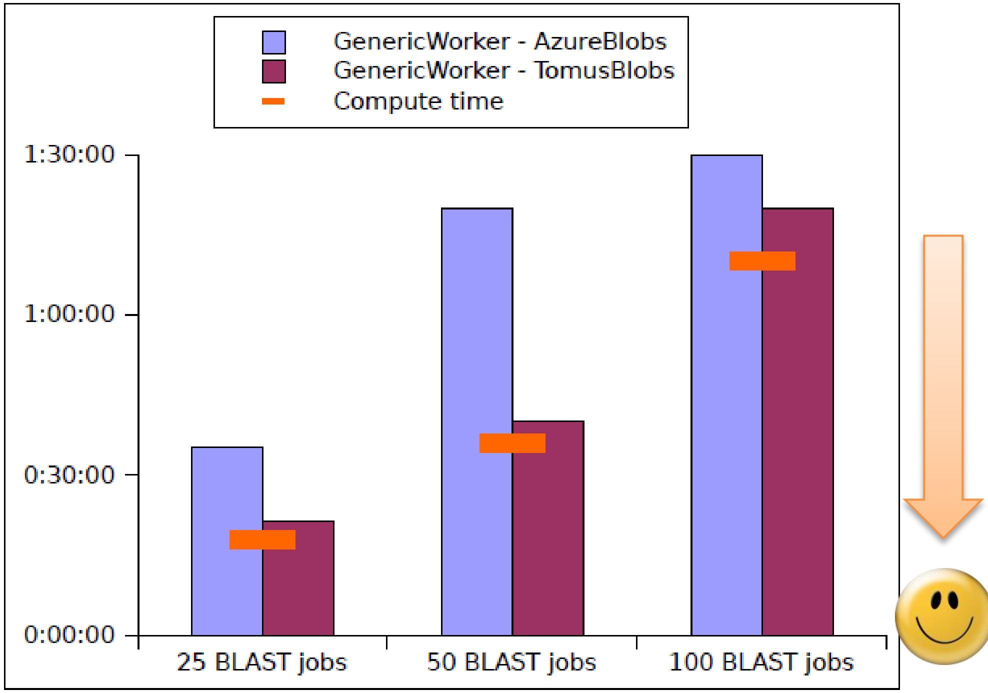


- **Multiple transfer** solutions: FTP, In-Memory, BitTorrent
- Adapt the transfer to the **data access pattern**
- Adaptive **replication** strategies **for higher performance**
- Integration with Microsoft Generic Worker

Workflow Processing on Cloud



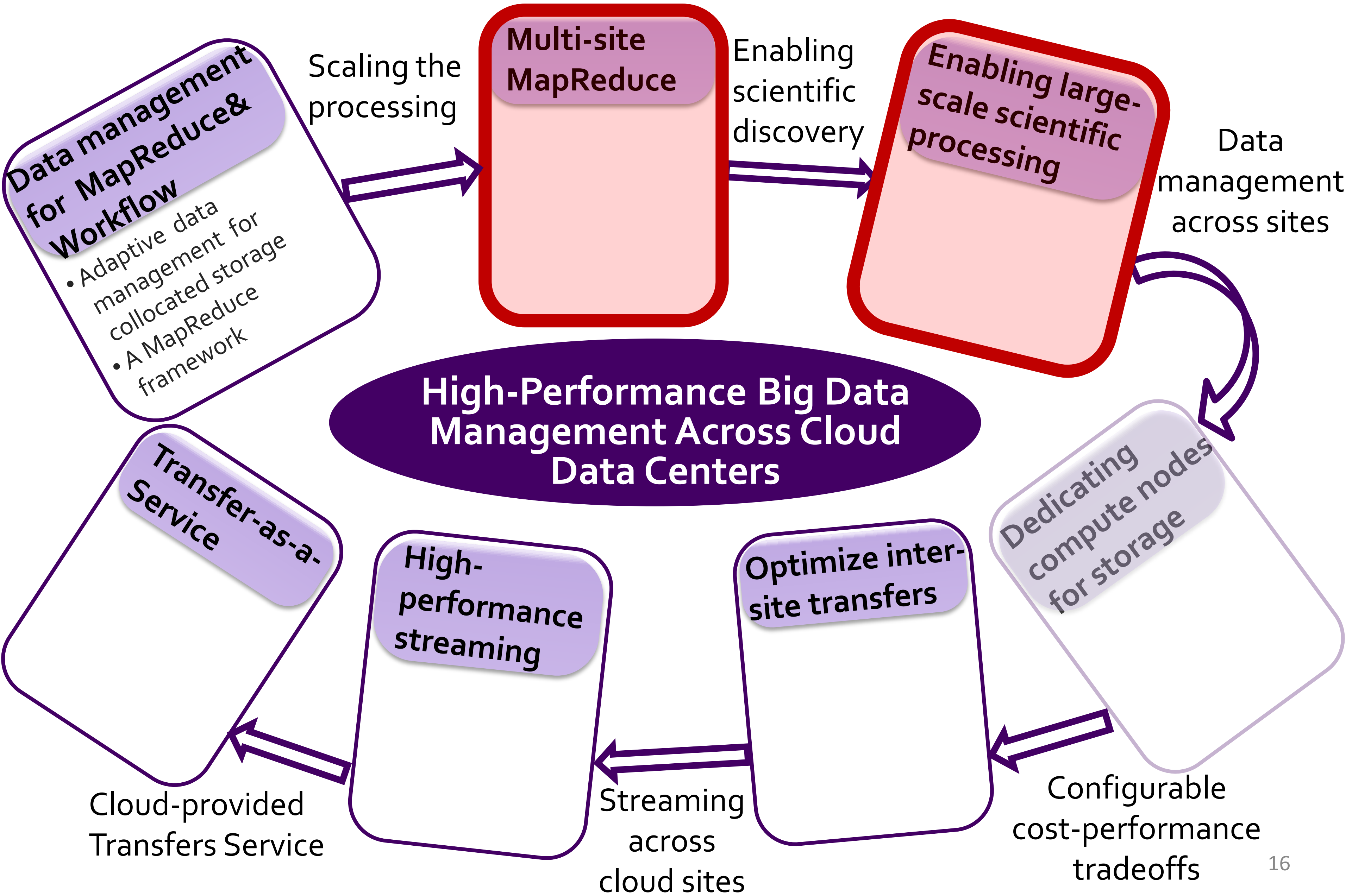
Synthetic workflow



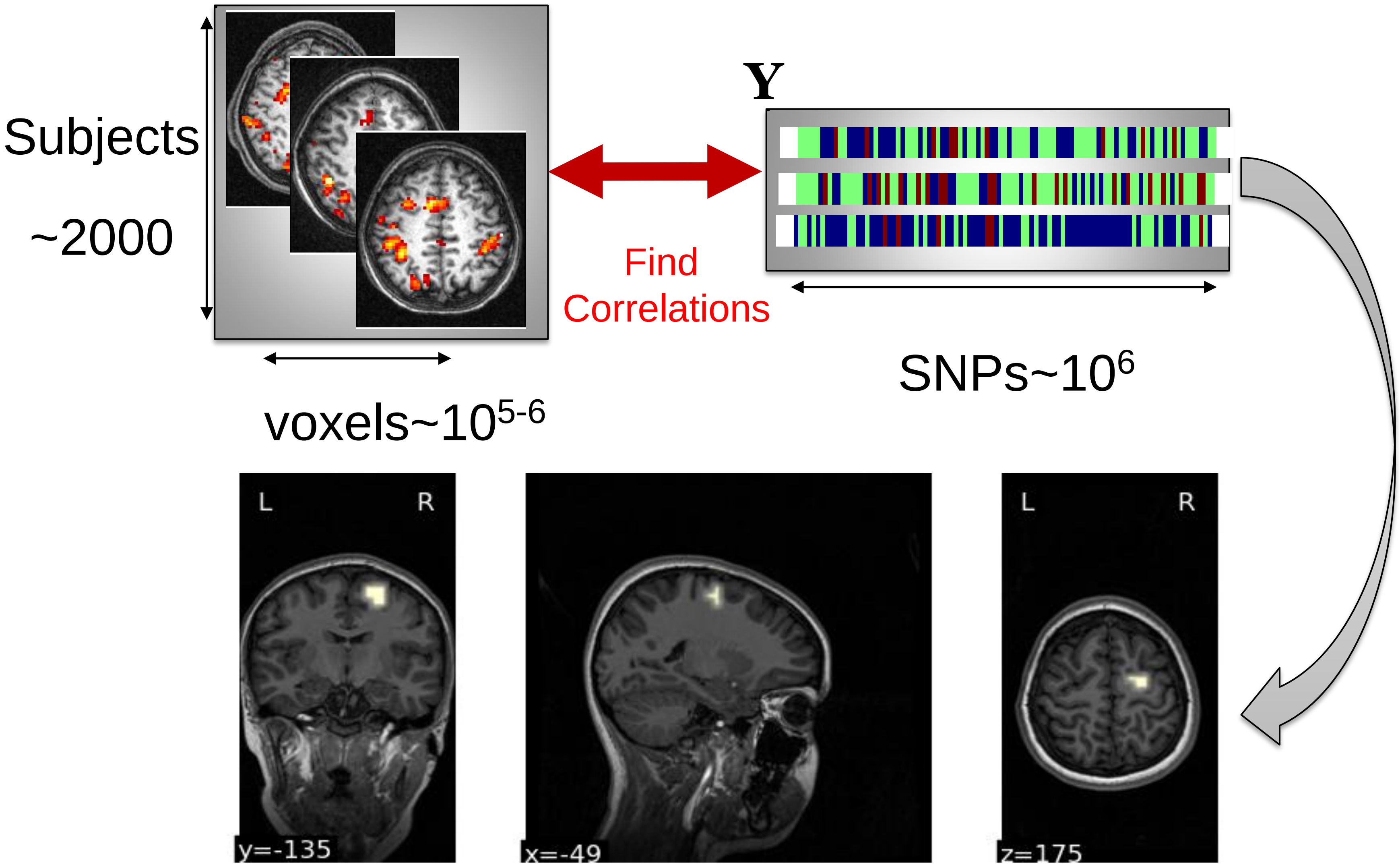
BLAST scientific workflow

- **Experimental Setup:** 100 Azure nodes, Generic Worker engine
- TomusBlobs adaptively chooses each time the best strategy

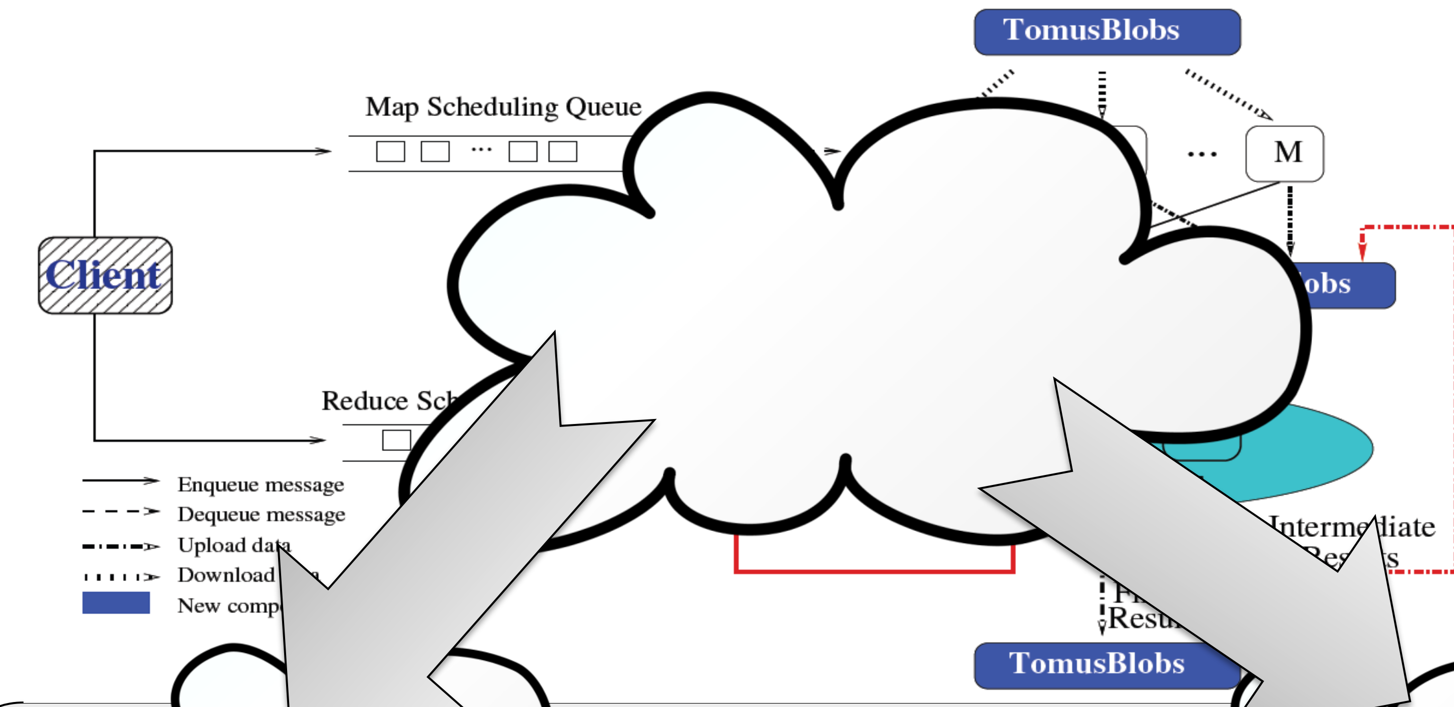
Doctoral Work in a Nutshell



A-Brain


Subjects
~2000
voxels~10^5-6
Find
Correlations
Y
SNPs~10^6
L
R
y=-135
x=-49
L
R
z=175

Single-Site Computation on the Cloud



P… Azure cloud across 350 cores …

A…

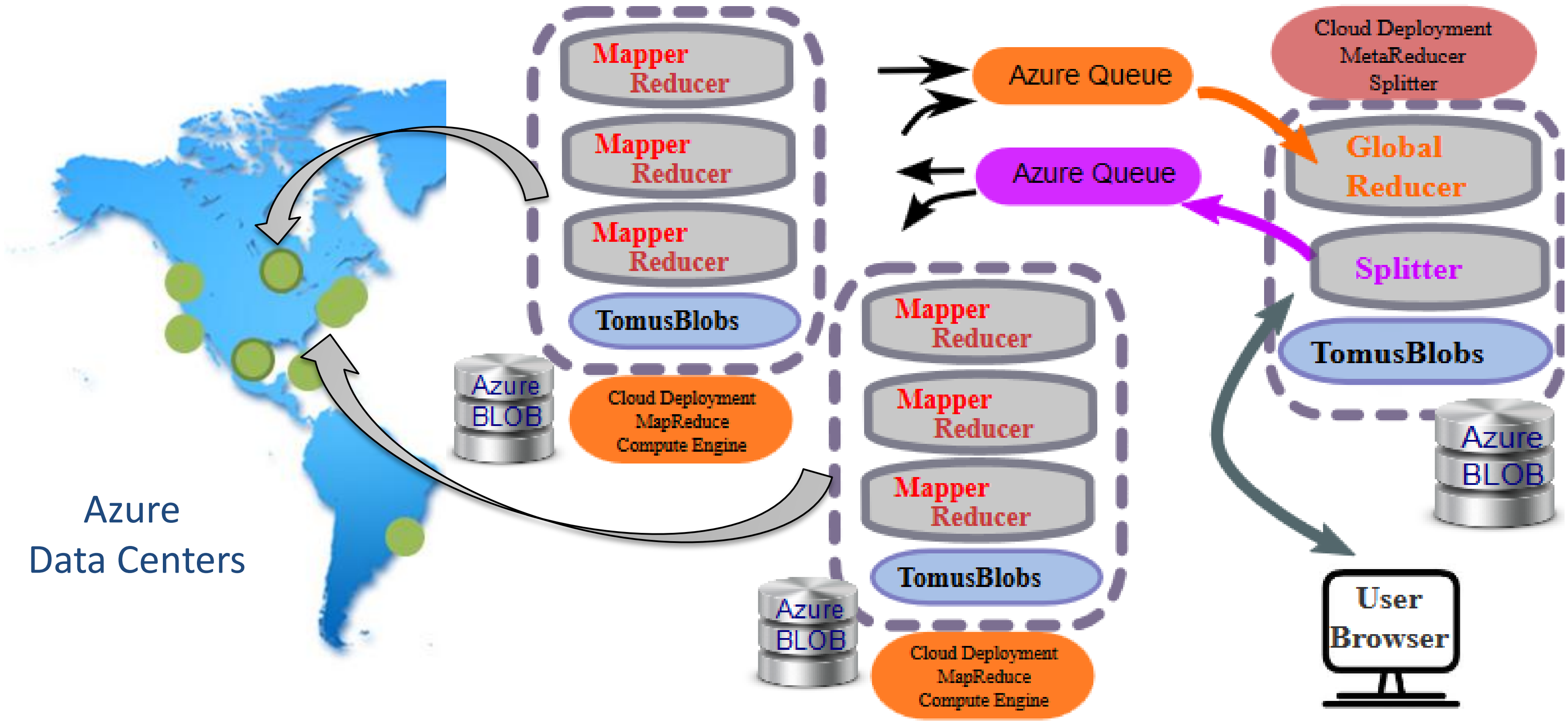
… to 5.6 days

…emonstrat…

Geographically distributed processing

- New cha… computation requirements to 86 years (for single core)

Going Geo-distributed

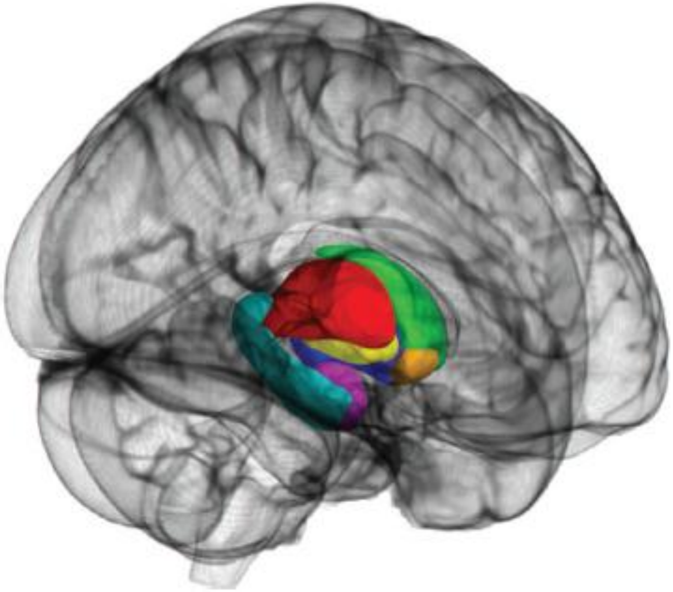


- Hierarchical multi-site MapReduce: Map-IterativeReduce, Global Reduce
- Data management: TomusBlobs (intra-site), Cloud Storage (inter-site)
- Iterative-Reduce technique for minimizing transfers of partial results
- Balance the network bottleneck from single data center

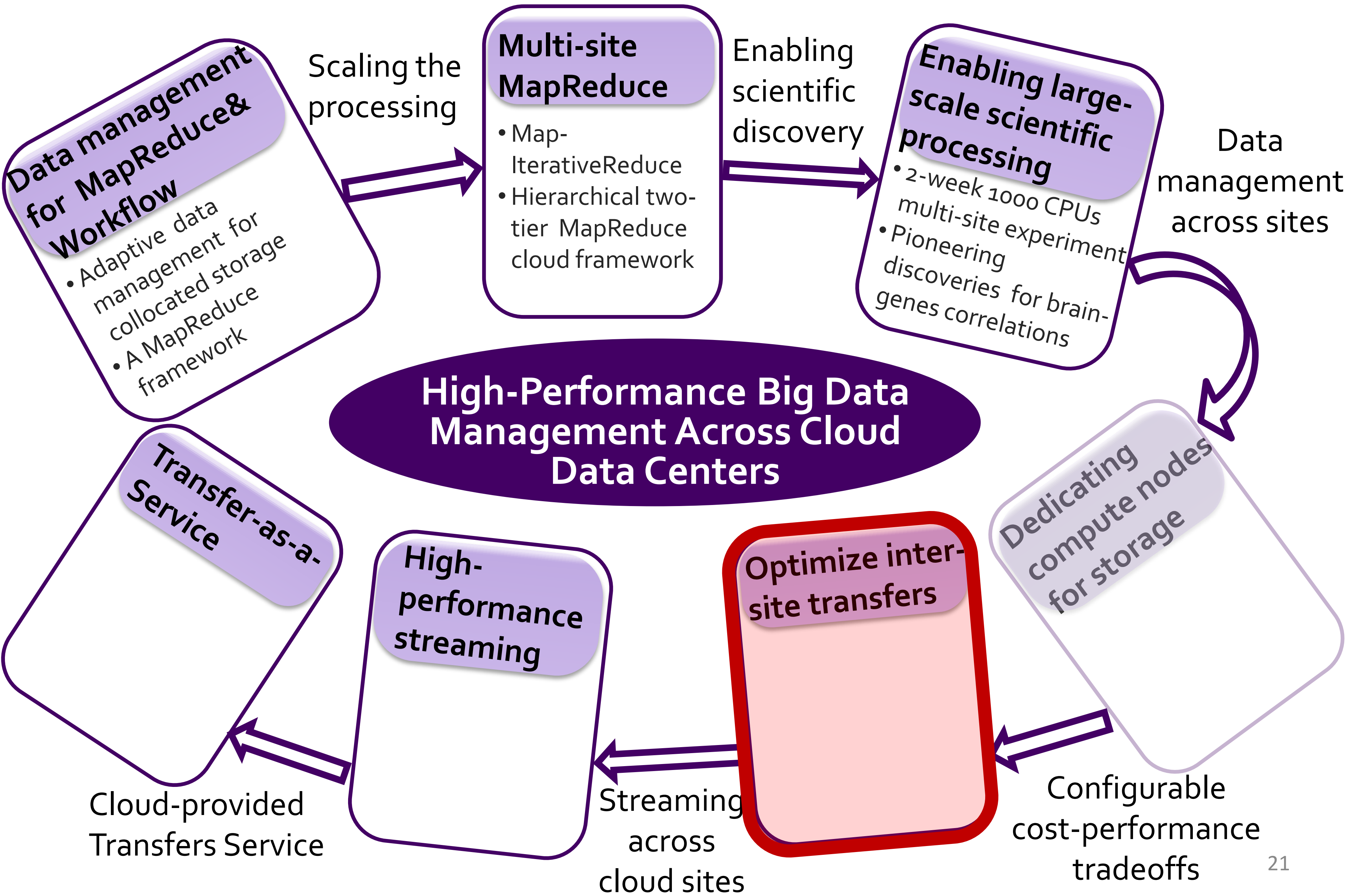
Executing the A-Brain Application at Large-Scale

- Multi-site processing: East US, North US, North EU Azure Data Centers
- Experiments performed on 1000 cores
- Experiment duration: ~ 14 days
- More than 210.000 hours of computation used
- Cost of the experiments: 20000 euros (VM price, storage, outbound traffic)
- 28000 map jobs (each lasting about 2 hours) and ~600 reduce jobs
- Data transfers more than 1 TB

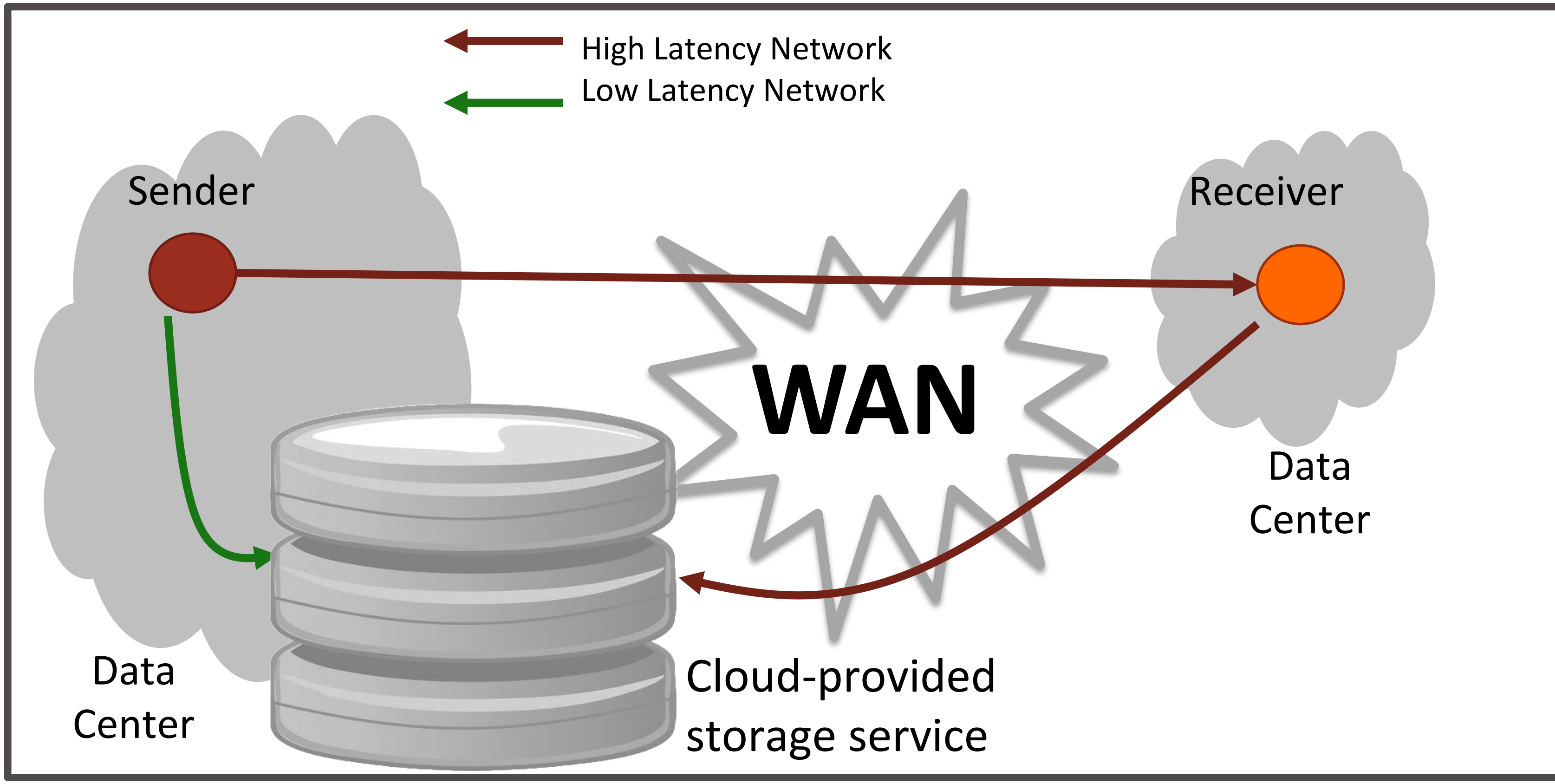
Scientific Discovery: Provided the first statistical evidence of the heritability of functional signals in a failed stop task in basal ganglia



Doctoral Work in a Nutshell



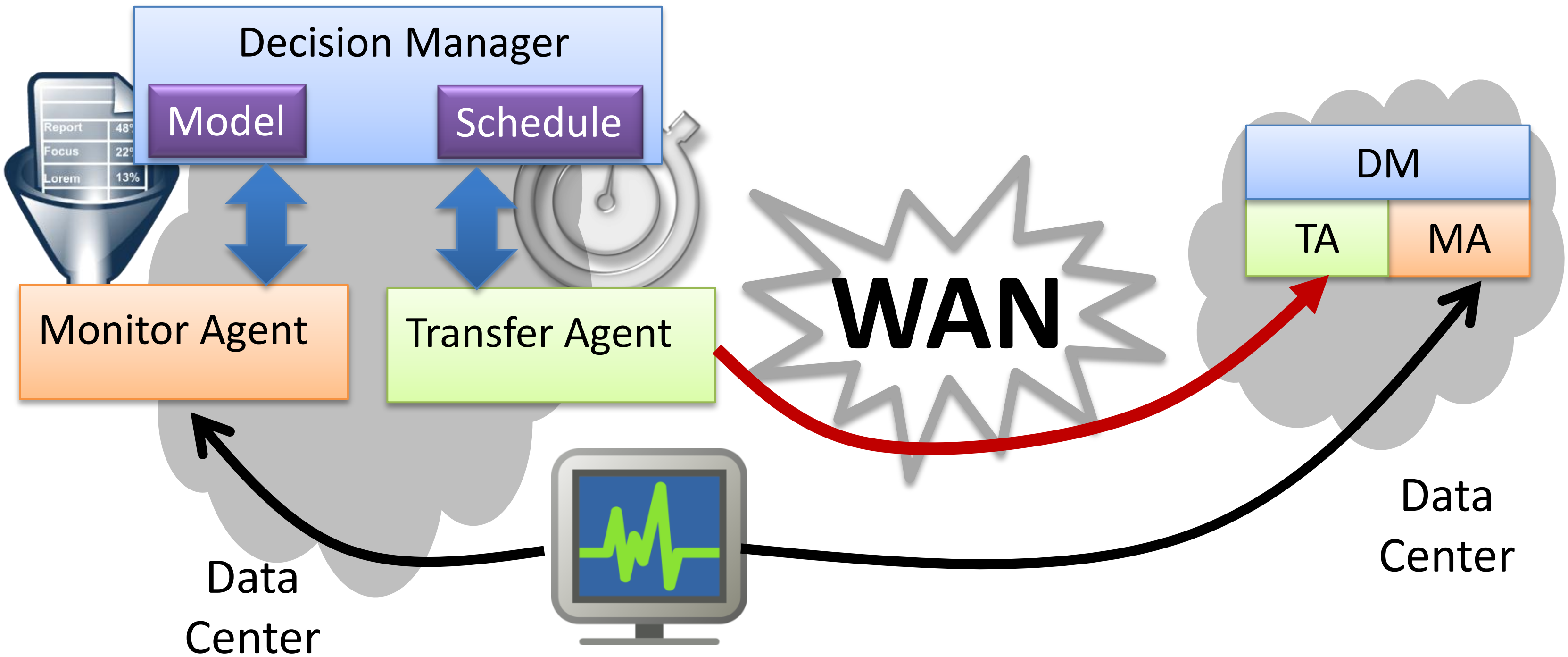
To Cloud or Not to Cloud Data?



Limitations:

- No (or weak) SLA guarantees
- High-latency and low throughput transfer

Addressing the SLA Issues for Inter-Site Transfers



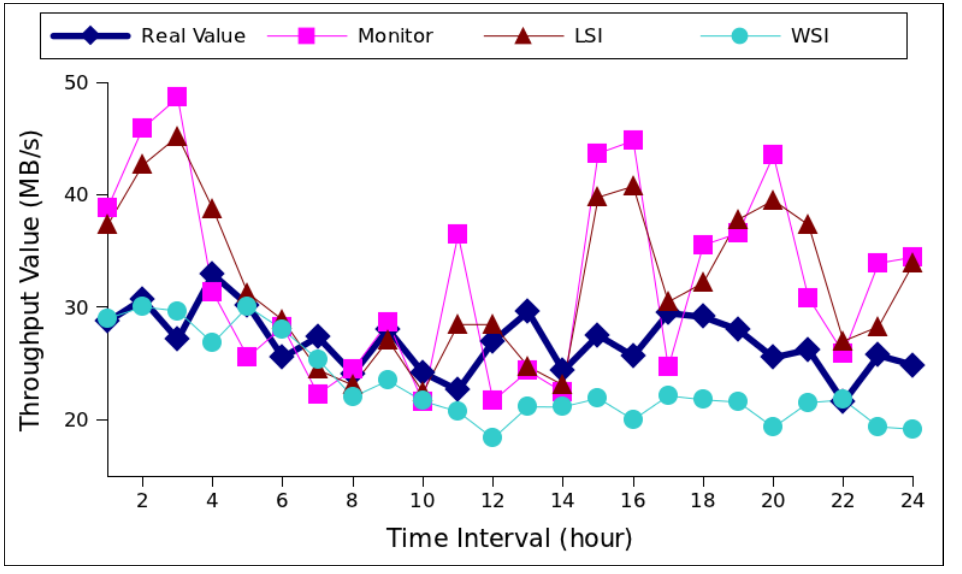
Three design principles:

- Environment awareness: model the cloud performance
- Real-time adaptation for data transfers
- Cost effectiveness: maximize throughput or minimize costs

Modeling Cloud Data Transfer

Sampling method

– Estimate the cloud performance based on the monitoring trace

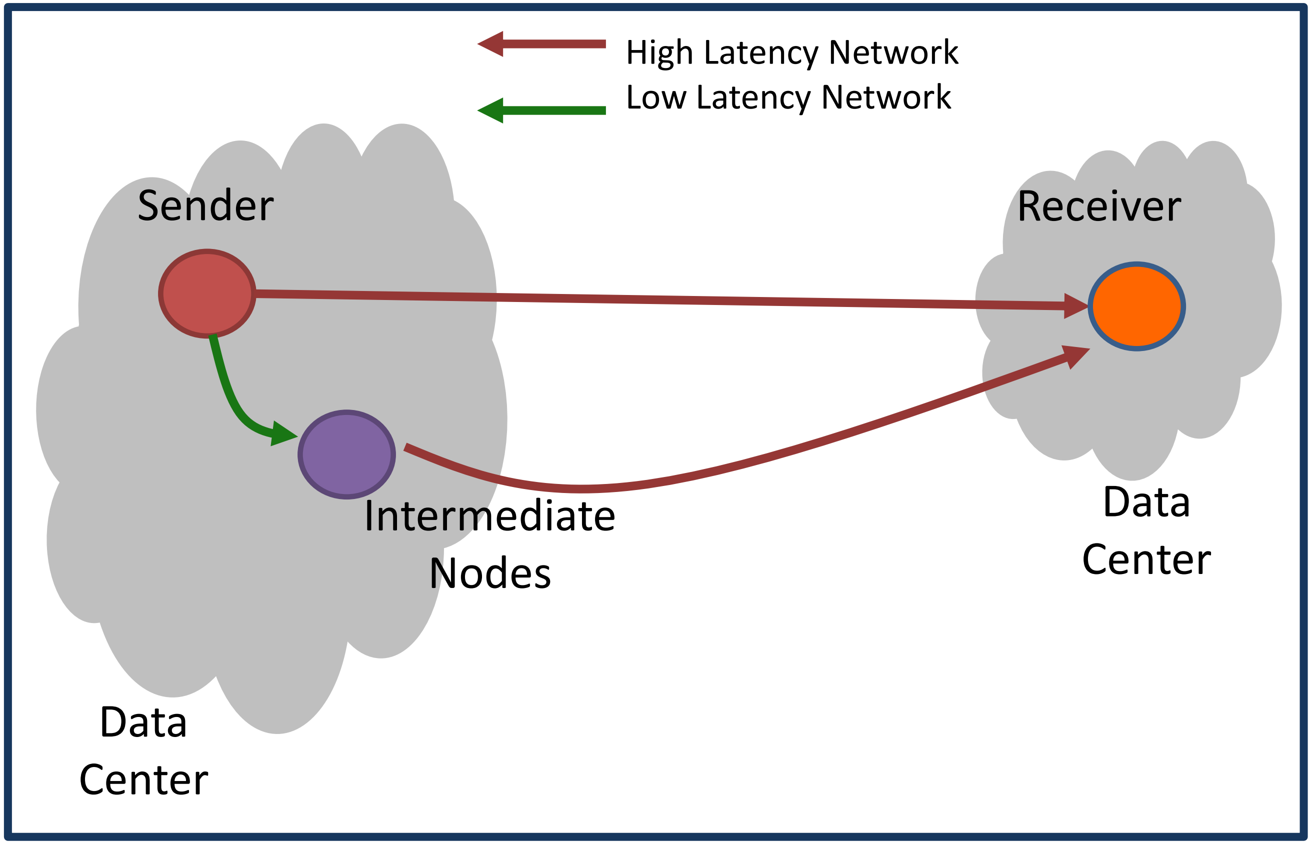


Average and variability estimated for each metric:

- Updated based on weights given to fresh samples: from 0 (no trust) to 1 (full trust)

- Predictive transfers: express transfer time and cost
- Dynamically adjust the transfer quotas across routes

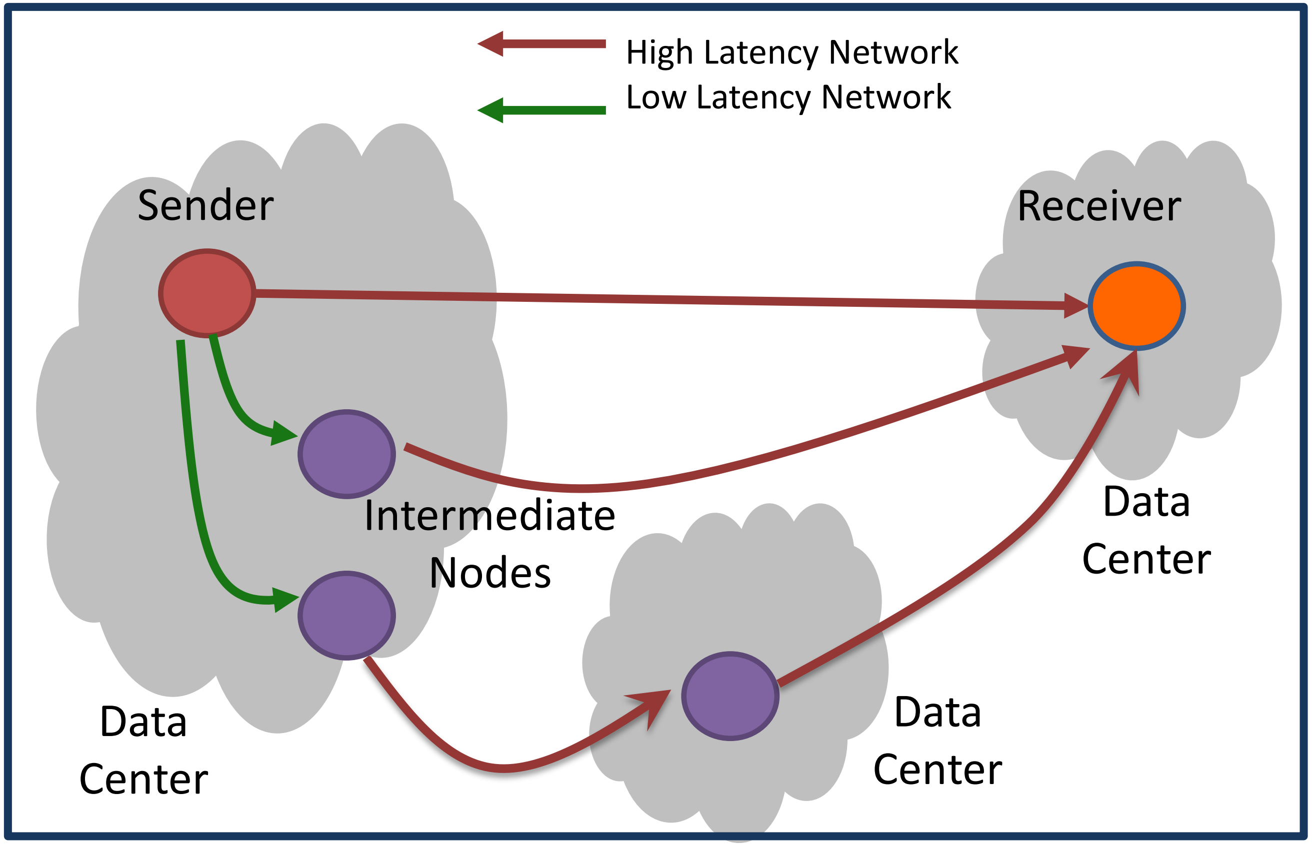
Addressing Inter-Site Transfer Performance: Multi-Path Transfers



Leverage network parallelism:

Aggregate inter-site bandwidth through multi-path transfers

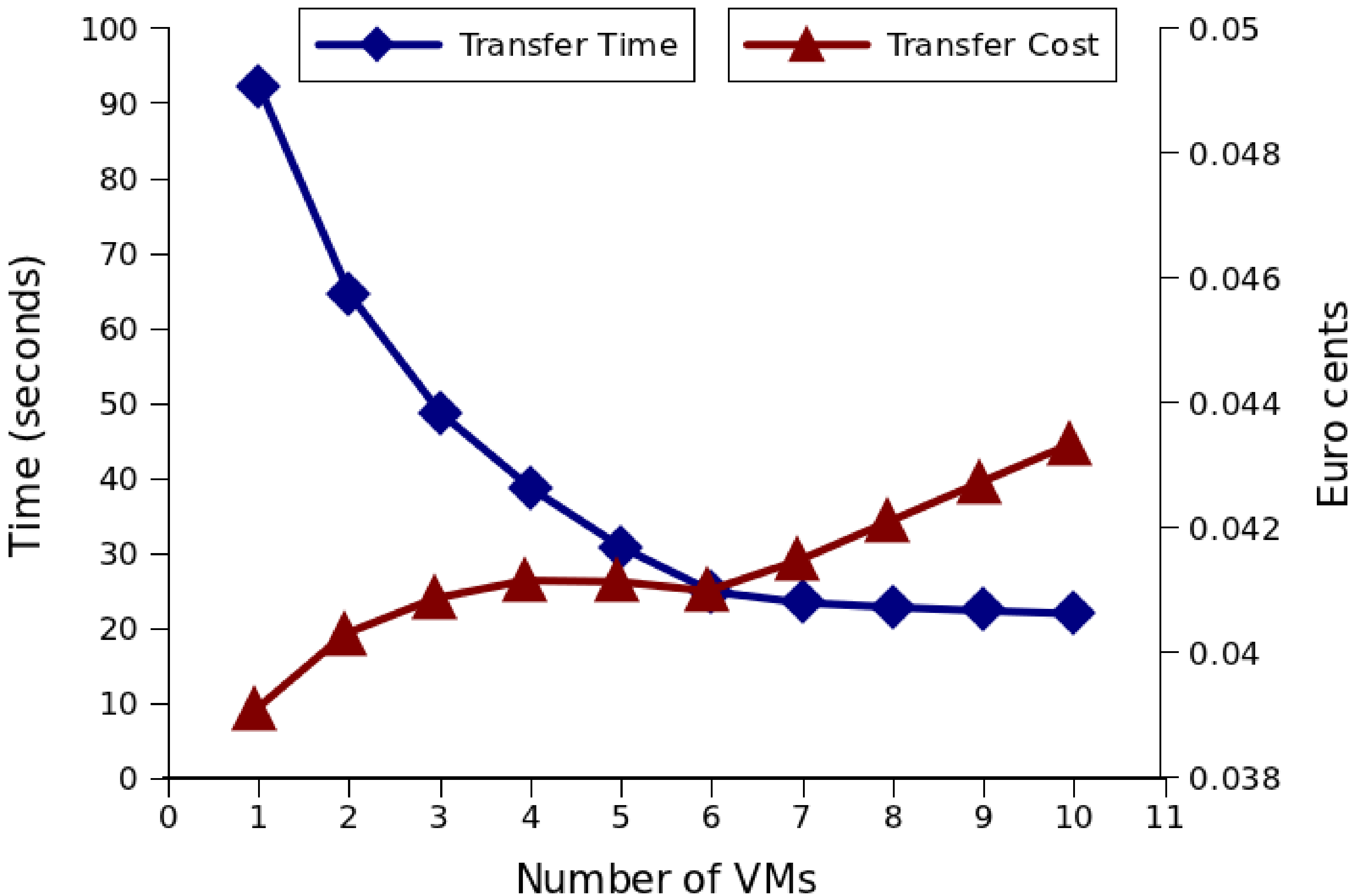
Addressing Inter-Site Transfer Performance: Multi-Hop Transfers



Further increase network parallelism:

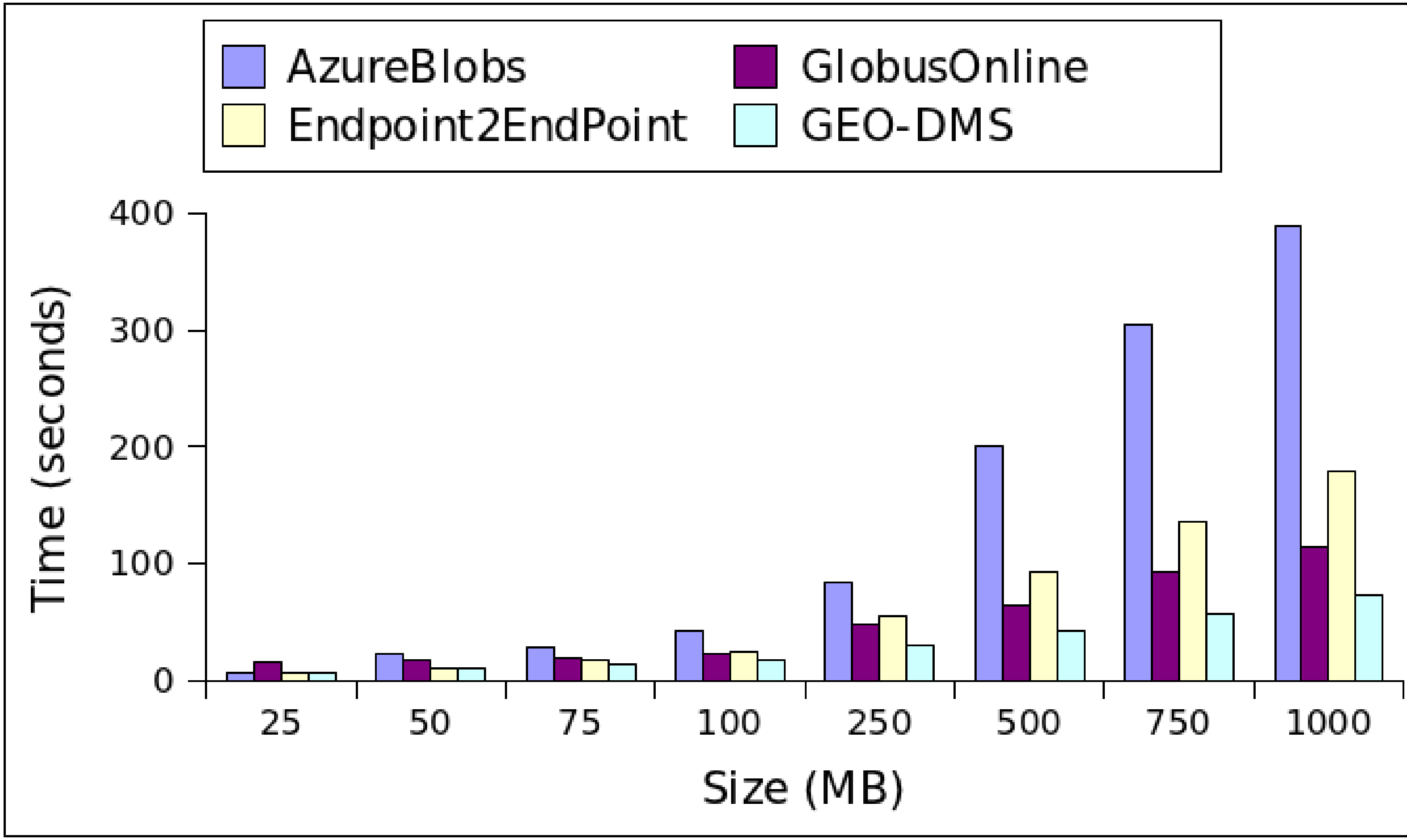
Avoid network throttling by considering alternative routes through other data centers

When Money Meets Performance



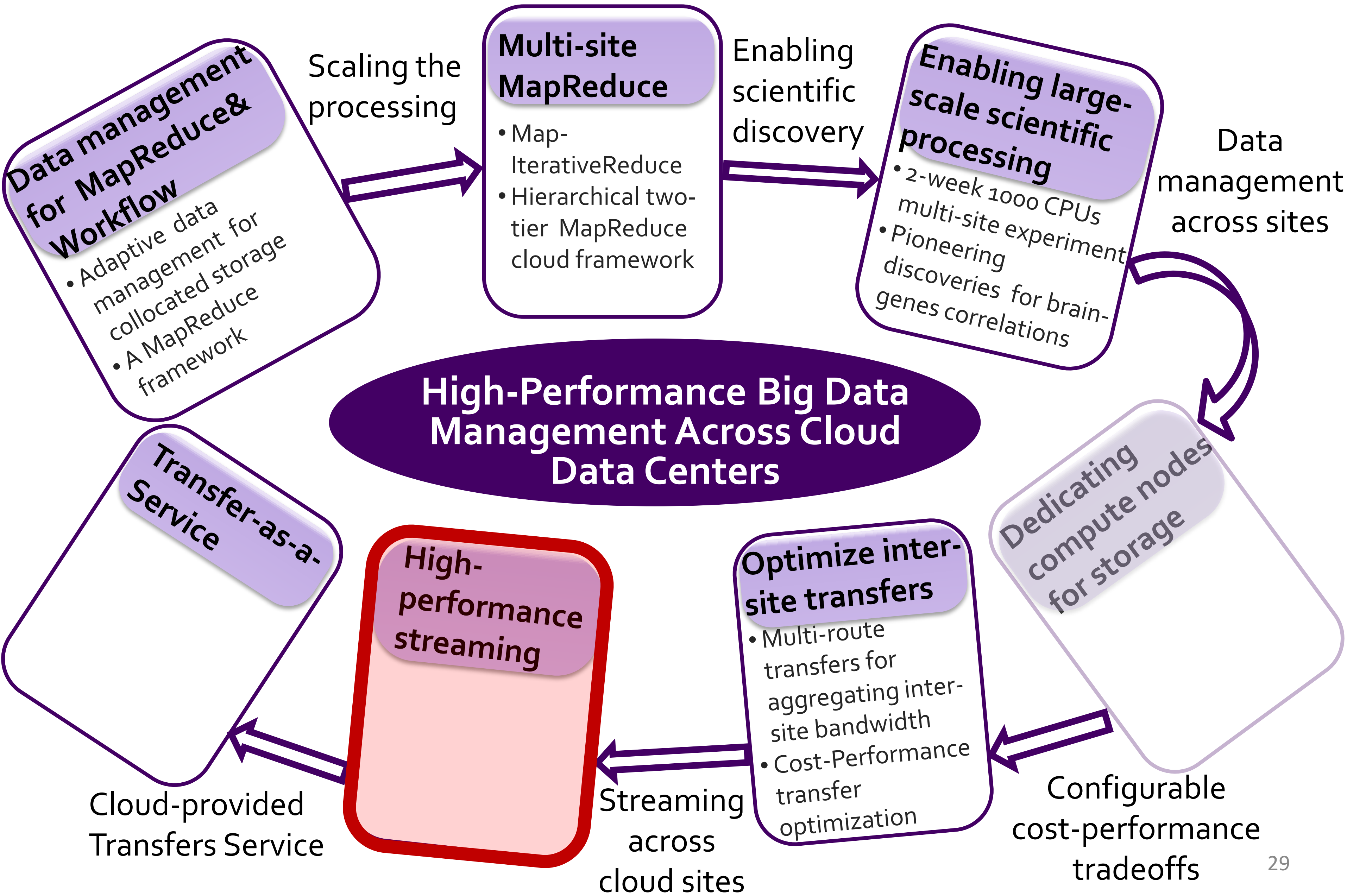
How much are you willing to pay for performance?
How much is it actually worth paying?

Comparing to Existing Solutions



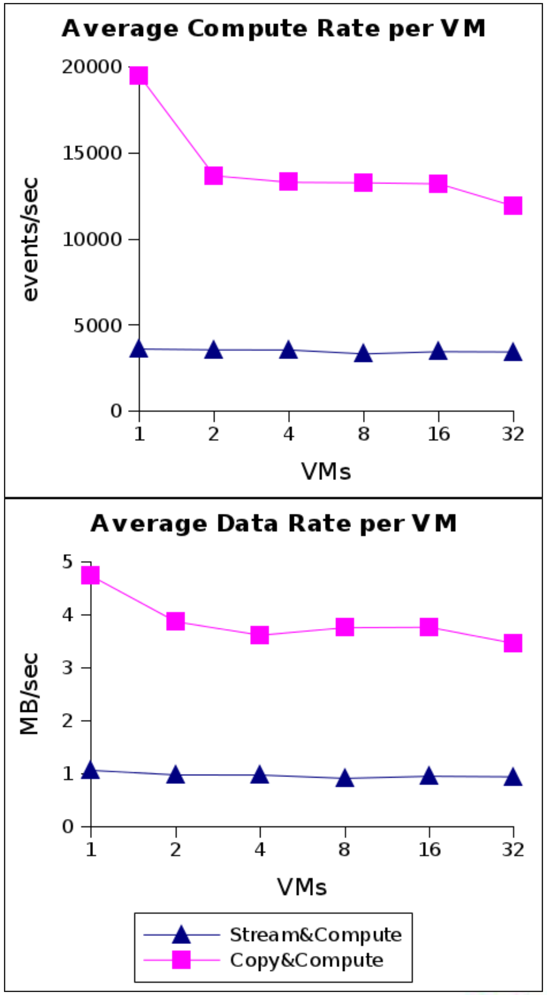
- **Experimental setup:** up to 10 nodes, Azure Cloud
- Transfers between North Central US to North EU Azure data centers

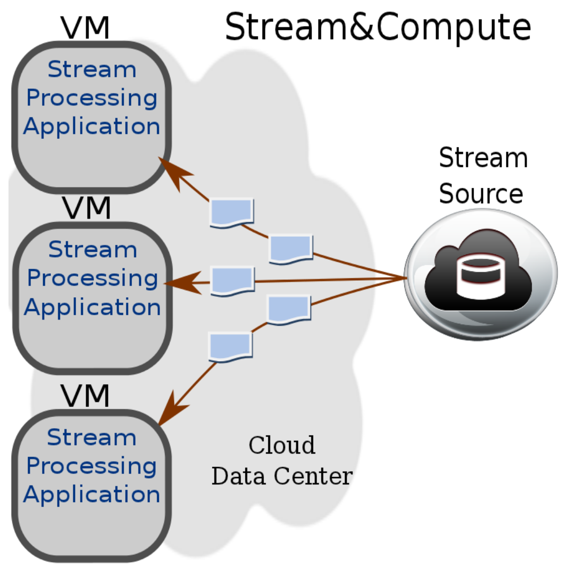
Doctoral Work in a Nutshell

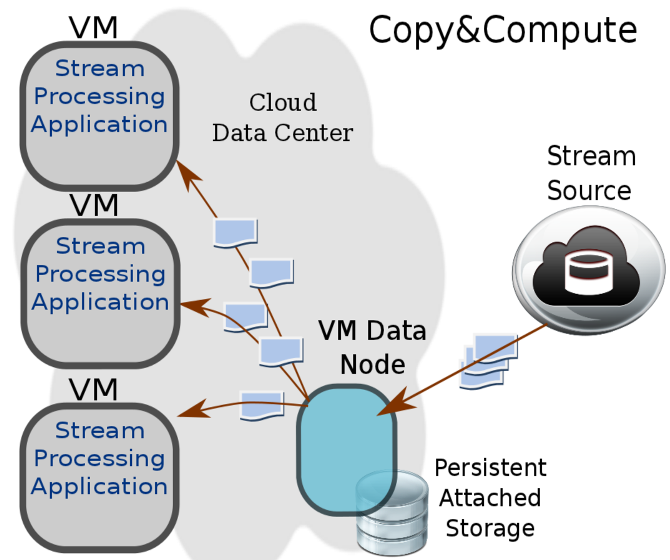


To Stream or Not to Stream?

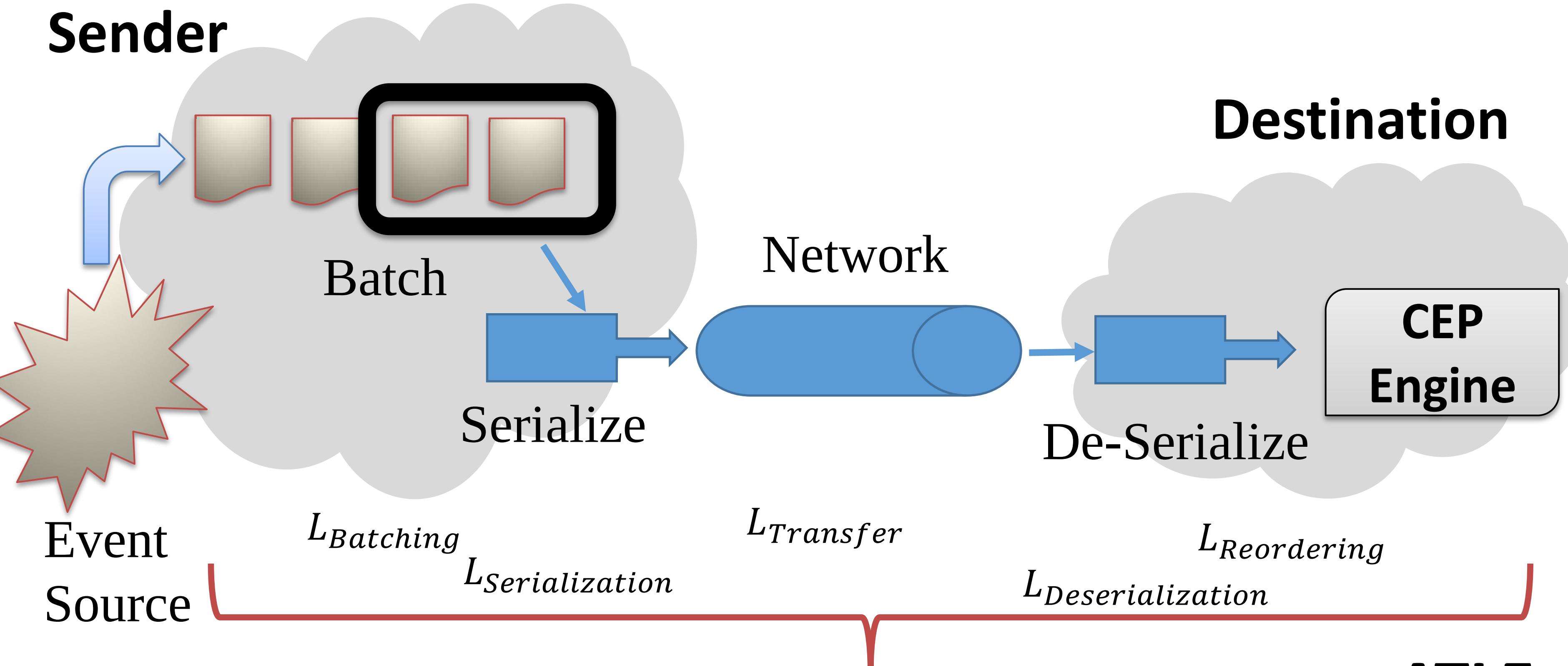
CERN Atlas







Towards Dynamic Batch-based Streaming

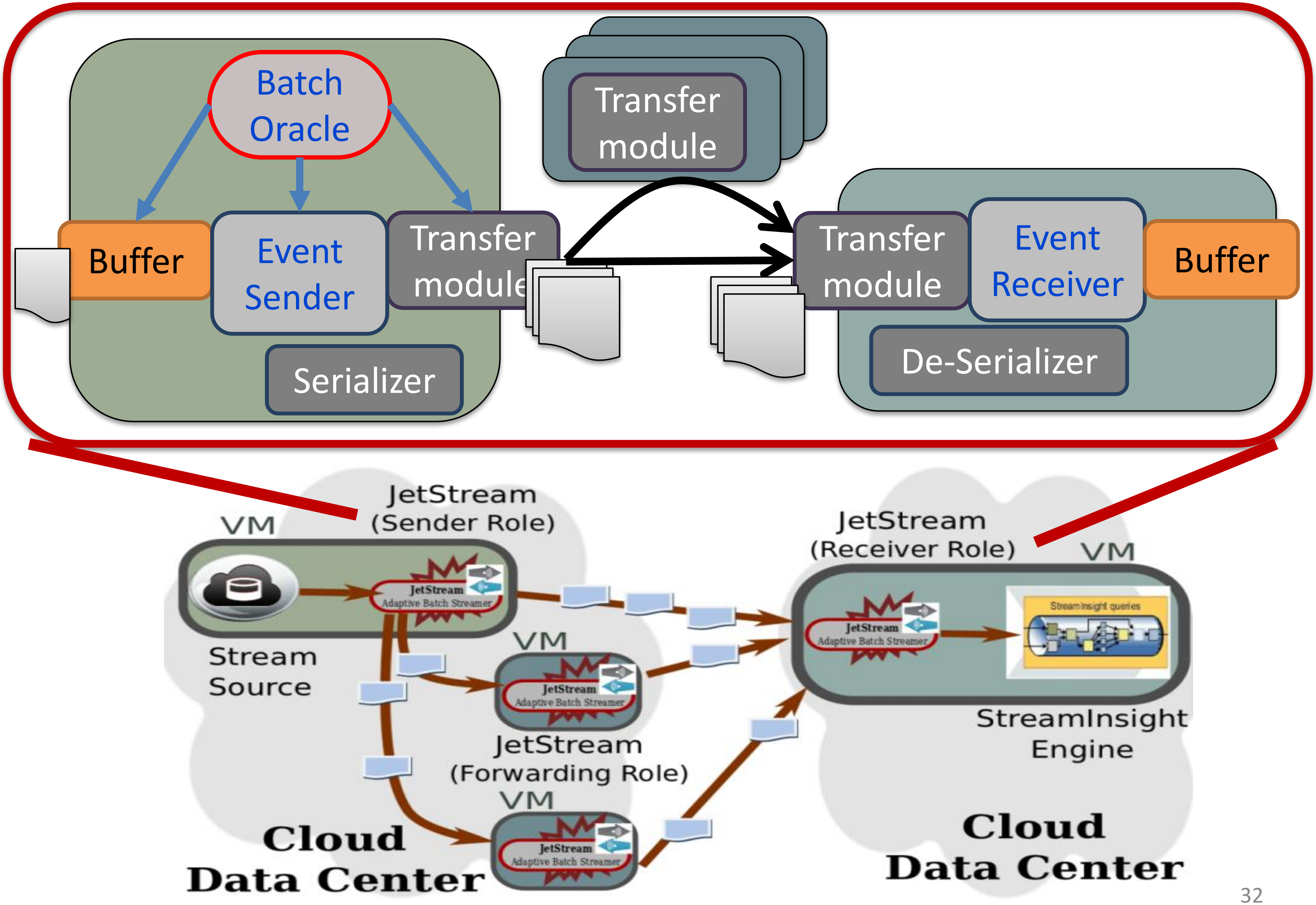


$$\underset{Nb}{\arg\min}\, L_{Event}$$

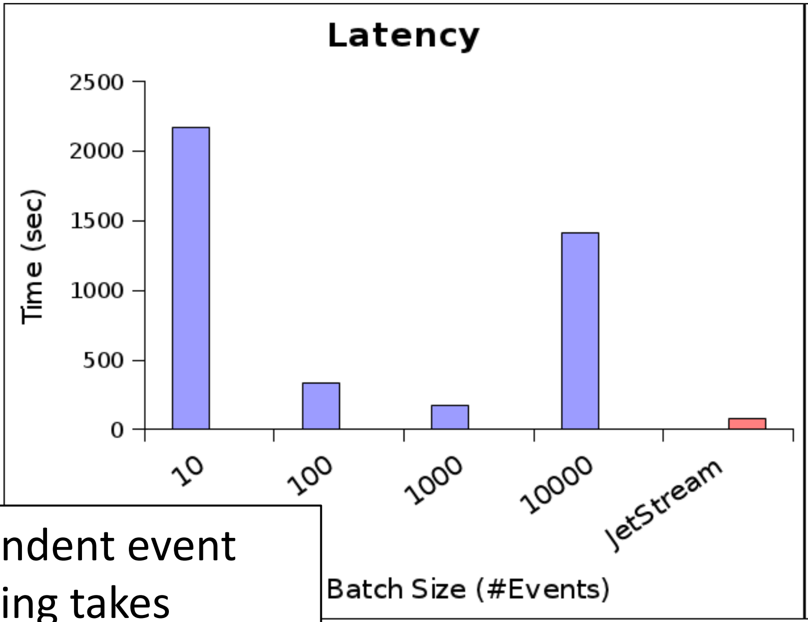
Microsoft Research ATLE

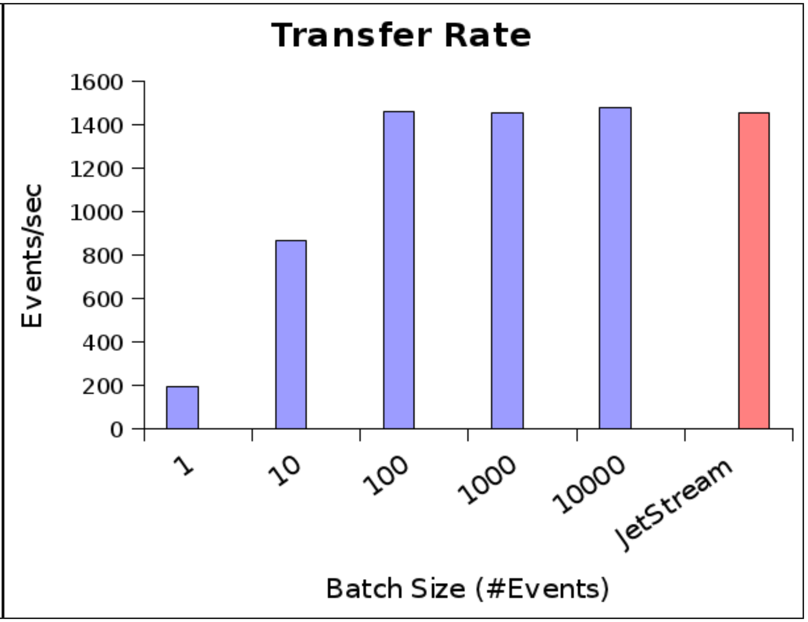
Latency (L) modeled based on stream context:
event size, throughput, arrival rate, routes, serialization/de-serialization technology, *batch size*

JetStream


Batch Oracle
Buffer
Event Sender
Transfer module
Serializer
Transfer module
Transfer module
Event Receiver
Buffer
De-Serializer
JetStream (Sender Role)
VM
Stream Source
JetStream Adaptive Batch Streamer
VM
JetStream (Forwarding Role)
VM
Cloud Data Center
JetStream (Receiver Role)
VM
StreamInsight queries
StreamInsight Engine
Cloud Data Center

JetStream for MonALISA

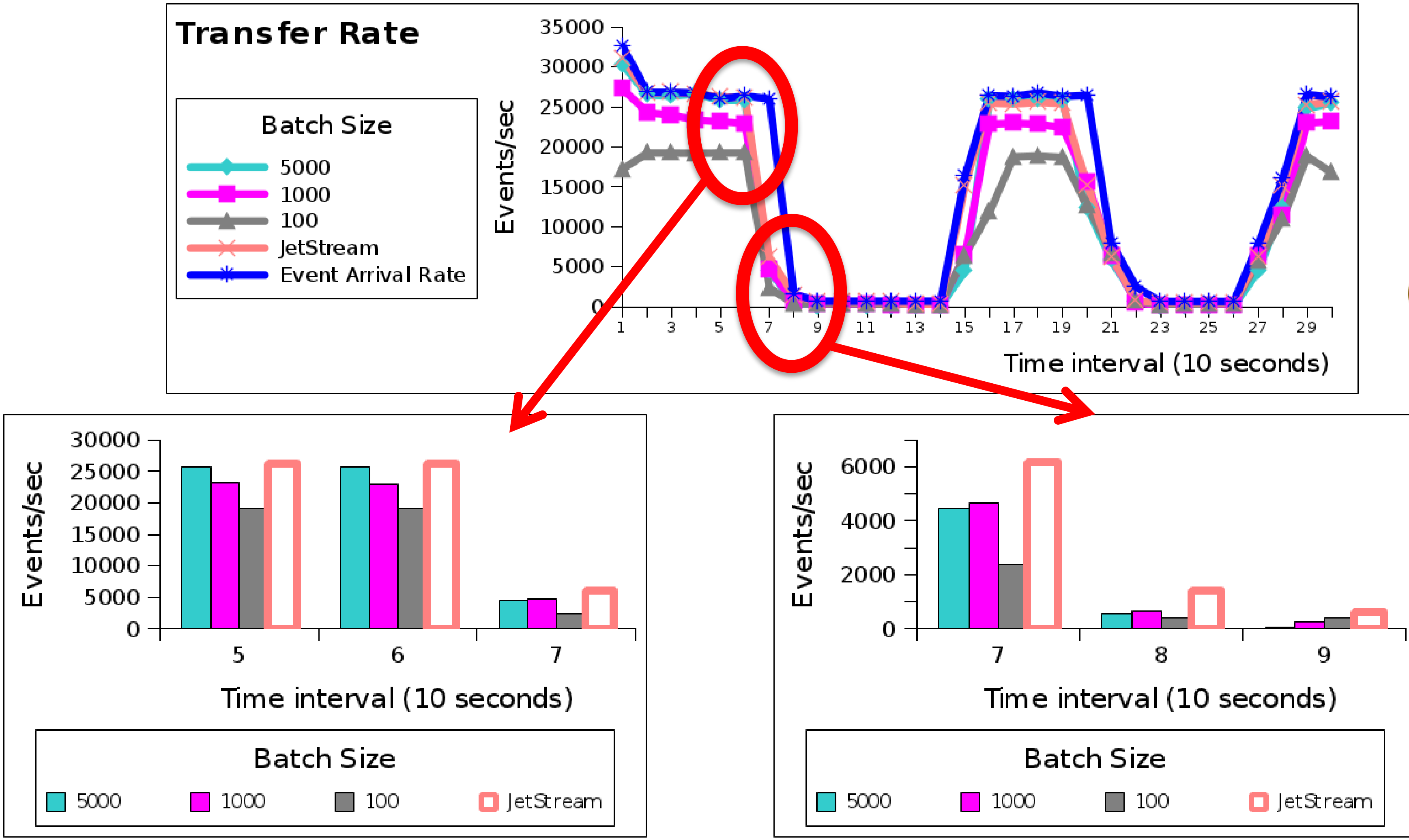


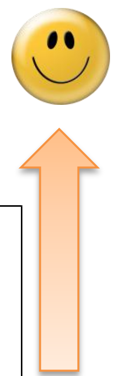


Independent event streaming takes 30,900 seconds compared to 80 seconds for JetStream

- 1.1 million events; North US to North EU Azure data centers
- Automatically resource optimization
- Optimizing the latency and transfer rate tradeoff

Variable Streaming Rates

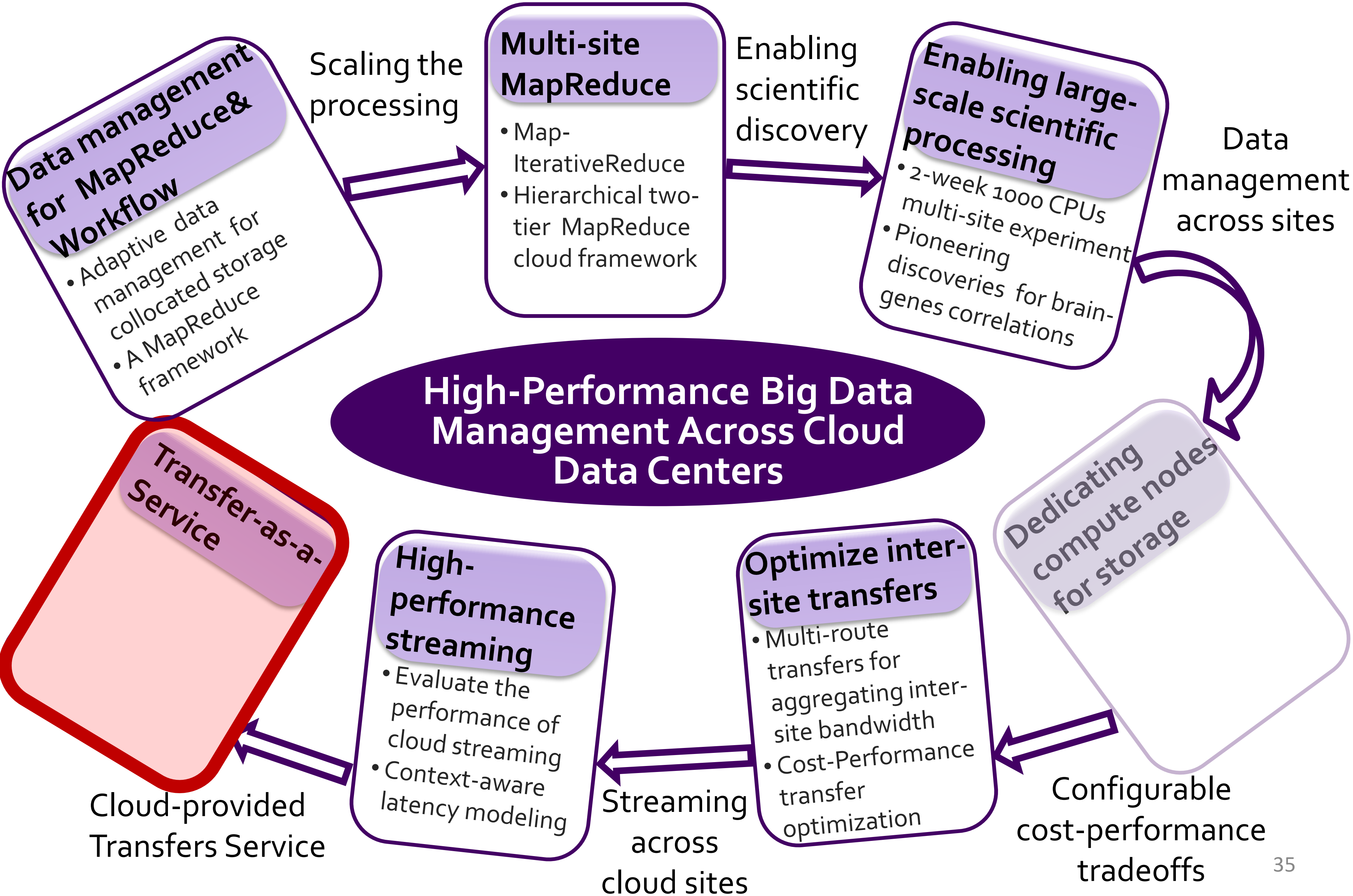




Elastic scaling of the resource based on load

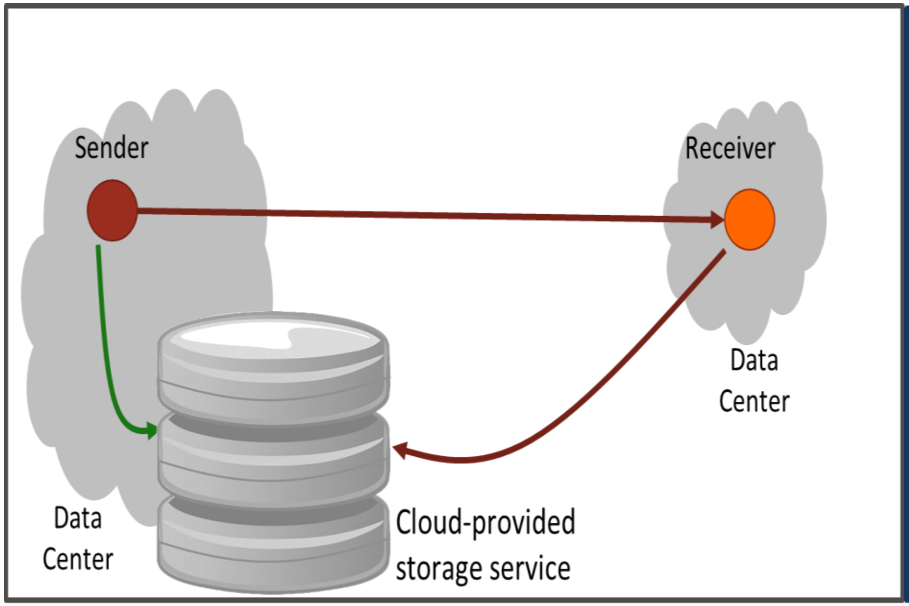
Environment-aware → self-optimization

Doctoral Work in a Nutshell



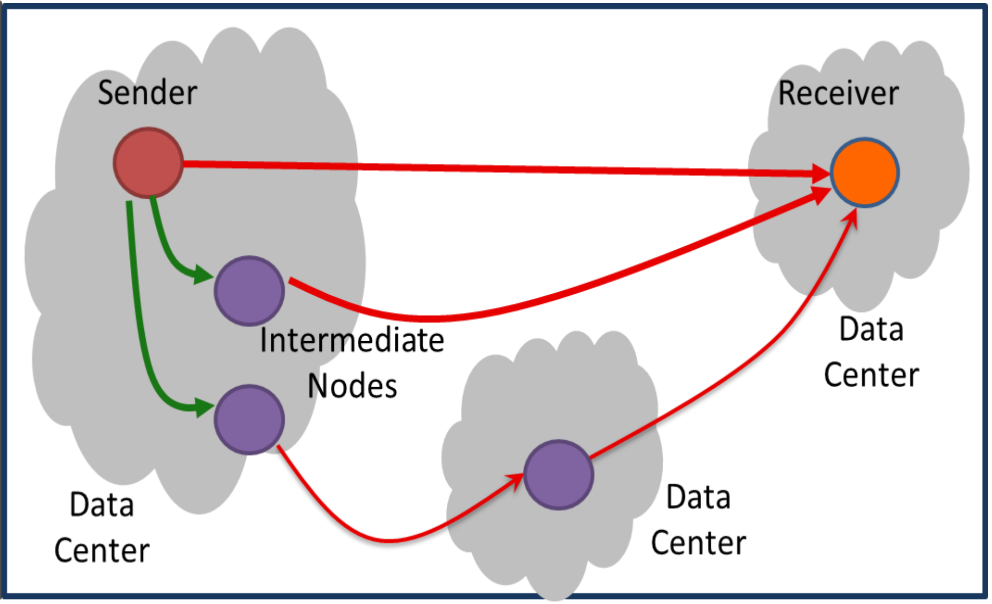
Transfer Options on the Cloud

The Default Option



- No (or minimal) configuration
- High-latency and low throughput transfer
- Fixed price scheme

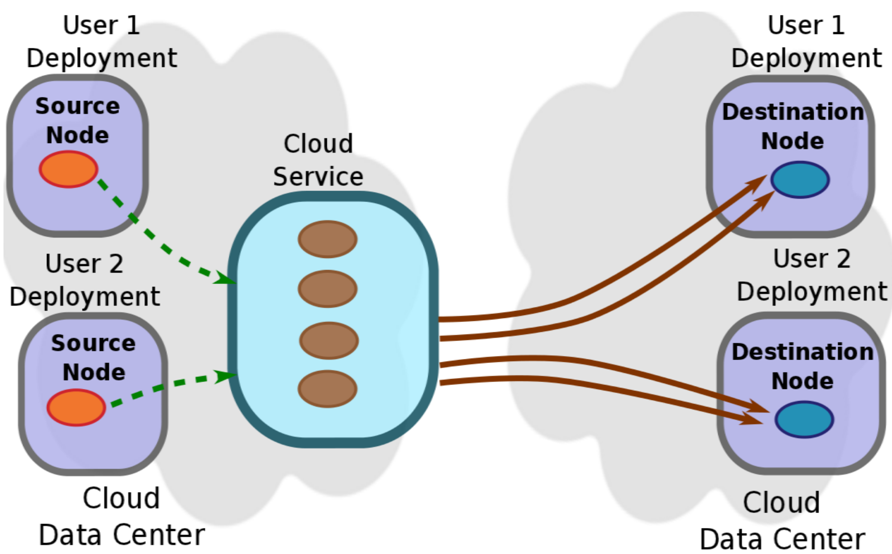
Multi-path transfers



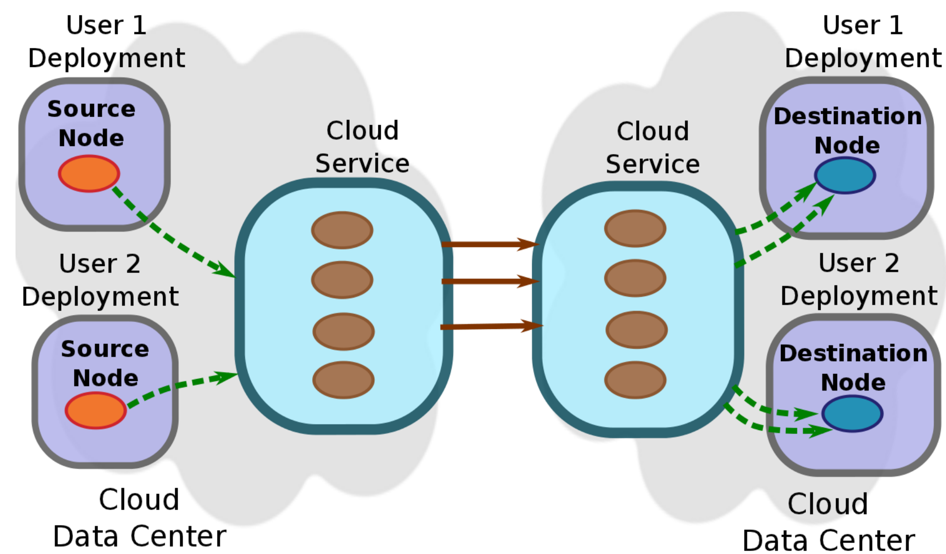
- Aggregate inter-site throughput
- Fixed price scheme
- Managed, configured and administrated by users

How About a Transfer as a Service?

Asymmetric cloud service



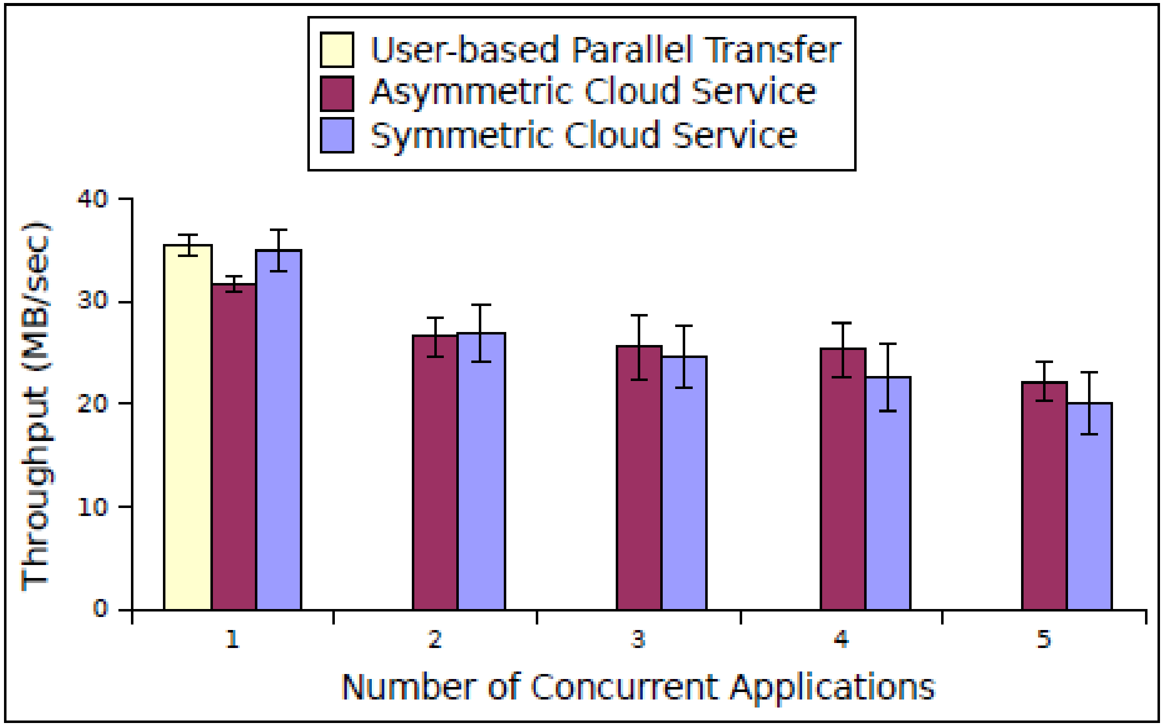
Symmetric cloud service



- **Federated clouds**
- No transparent communication optimizations

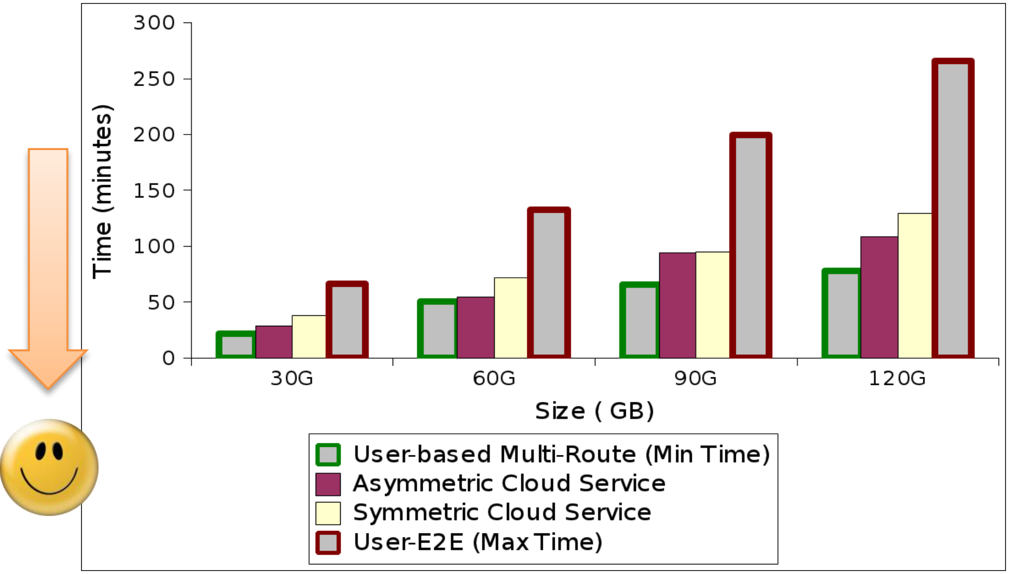
- **Same cloud vendor**
- Allows any number of communication optimizations

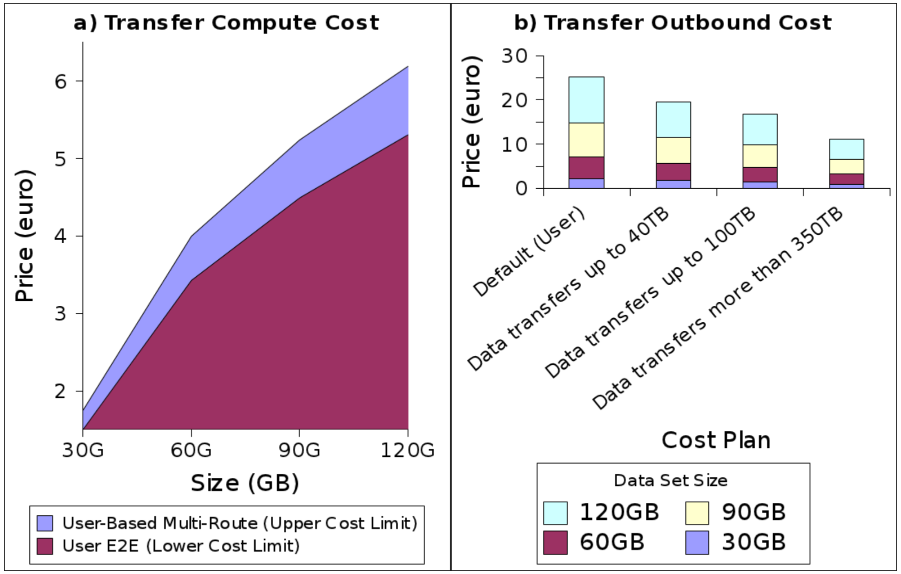
Is TaaS Feasible Performance-wise?



Multi-tenant service usage: performance degradations of 20%
… while the number of service nodes per app is decreased from 5:1 to 1:1

Performance-Based Cost Models?

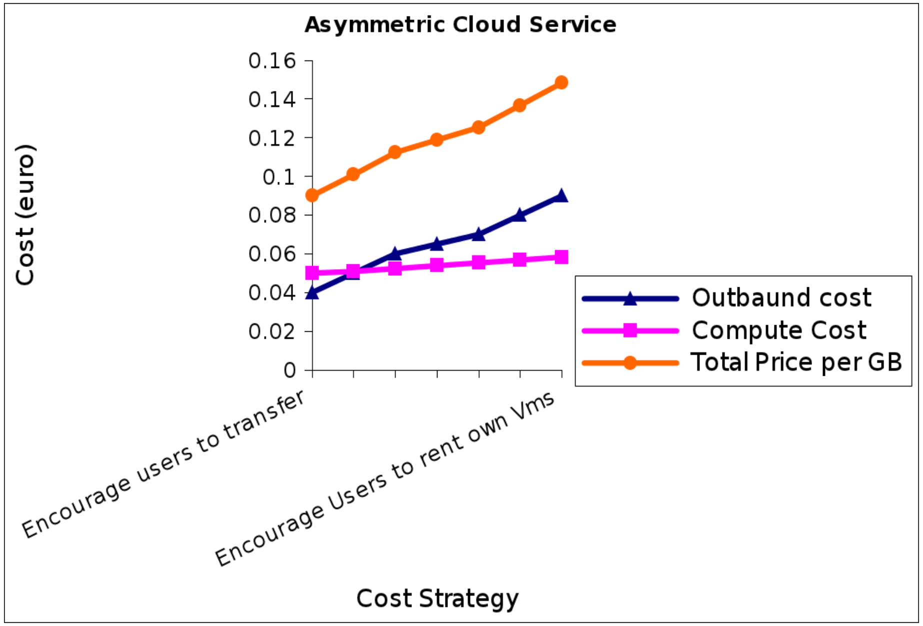


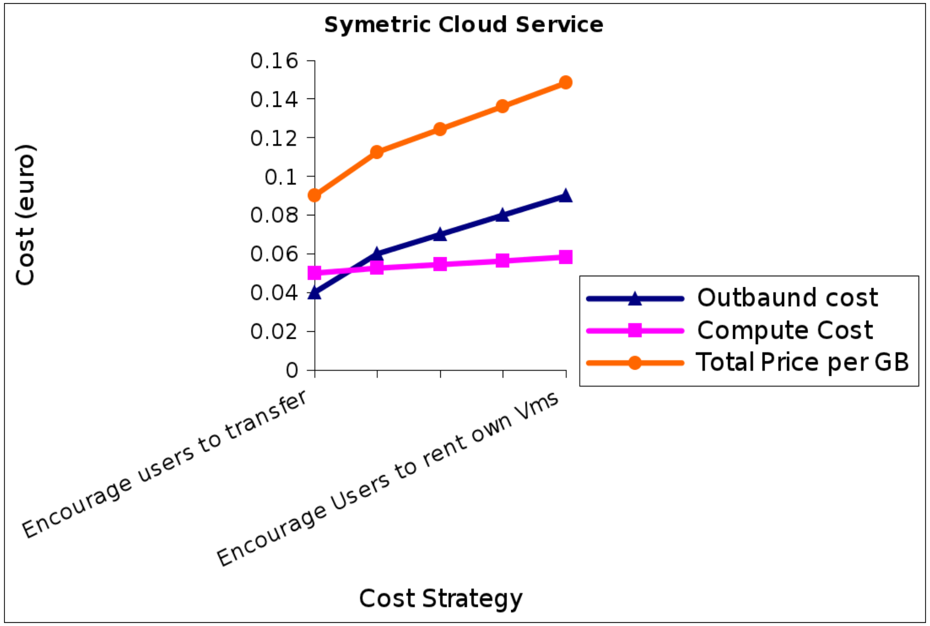


Scenario: Transfer large volumes of data across Azure sites

Cost: Cost margins for the service usage can be defined based on performance

Data Transfer Market





Data transfer market: Flexible and dynamic pricing

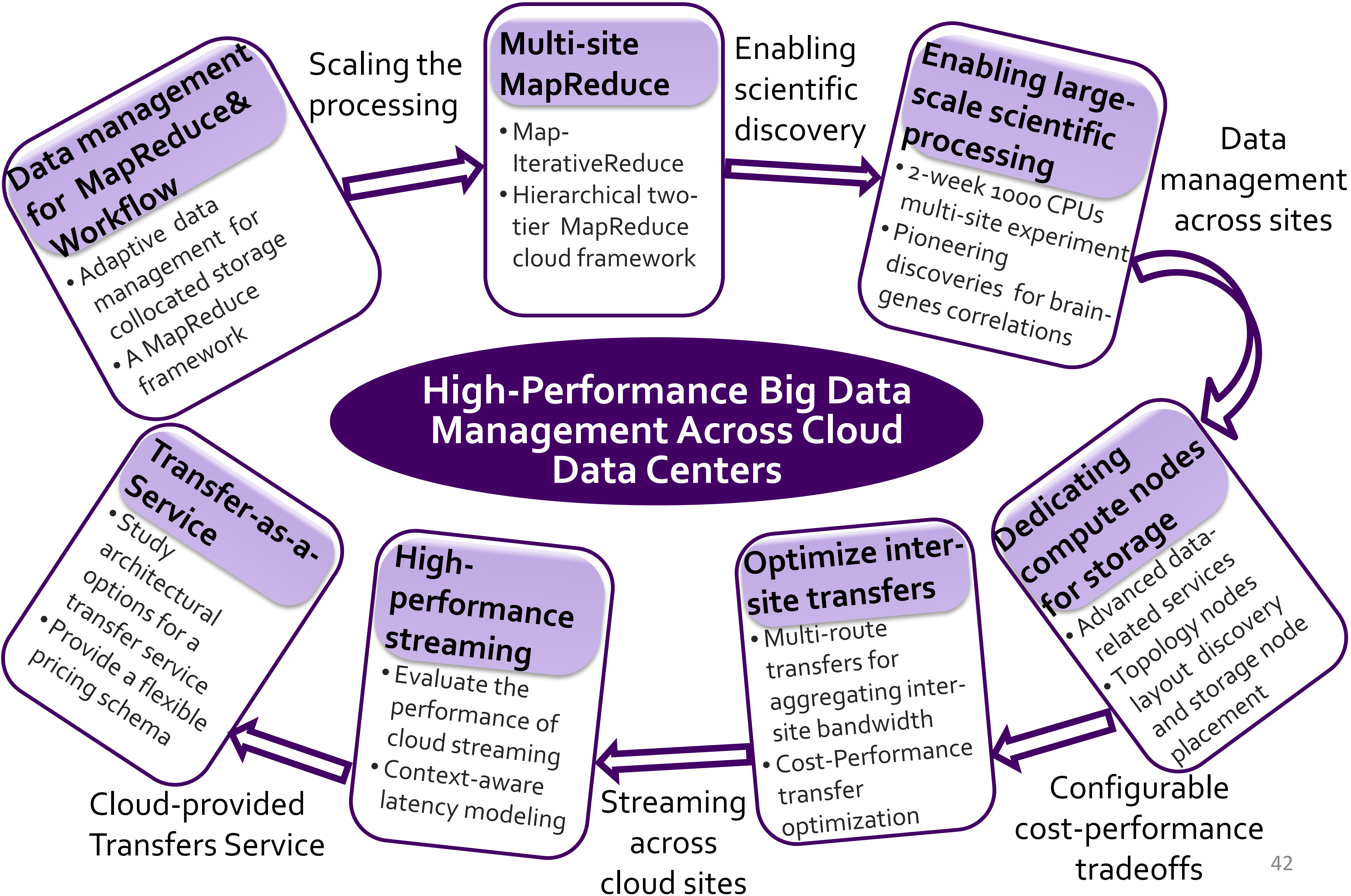
=> **win-win** situation for cloud vendor and users

Why? Decrease price => to reduce idle bandwidth

Increase price => to decrease network congestion

Conclusions & Perspectives

Doctoral Work in a Nutshell



Achievements

Publications

- 1 Book Chapter
  - In Cloud Computing for Data-Intensive Applications, Springer 2015
- 3 Journal articles
  - Frontiers in Neuroinformatics 2014
  - Concurrency and Computation Practice and Experience 2013
  - ERCIM Electonic Journal 2012
- 7 International Conferences publications
  - 3 papers at IEEE/ACM CCGrid 2012 and 2014 (Cloud Cluster and Grid, **rank A**), **Acceptance rates: 26%, 19%**
  - IEEE SRDS 2014 (Symposium on Reliable Distributed Systems, **rank A**)
  - IEEE Big Data 2013, **Acceptance rate 17%**
  - ACM DEBS 2014 (Distributed Event Based Systems), **Acceptance rate 9%**
  - IEEE Trustcom/ISPA 2013 (**rank A**)
- 7 Workshops papers, Posters and Demos
  - MapReduce in conjuction with ACM **HPDC (rank A)**
  - CloudCP in conjuction with ACM **EuroSys (rank A)**
  - IPDPSW in conjuction with IEEE **IPDPS (rank A)**
  - Microsoft: CloudFutures, ResearchNext, PhD Summer School
  - DEBS Demo in conjunction with ACM **DEBS**

Software

TomusBlobs

PaaS data management middleware
- Available with Microsoft GenericWorker

MapReduce engine for the Azure cloud
- Cloud service for bio-informatics

Cloud Benchmark Service

SaaS for benchmarking the performance of data stage-in to cloud data centers
- Available on Azure Cloud

JetStream

Middleware for batch-based, high-performance streaming across cloud sites
- Binding with Microsoft StreamInsight

External Collaborators

- Microsoft Research ATLE, Cambridge
- Argonne National Laboratory
- Inria Saclay
- Inria Sophia Antipolis

Perspectives

- Multi-site workflow across geographically distributed sites

  Worflow data access patterns, self-* processing, cost/performance tradeoffs

Z-CloudFlow

Microsoft Research - Inria JOINT CENTRE

- Cloud stream processing

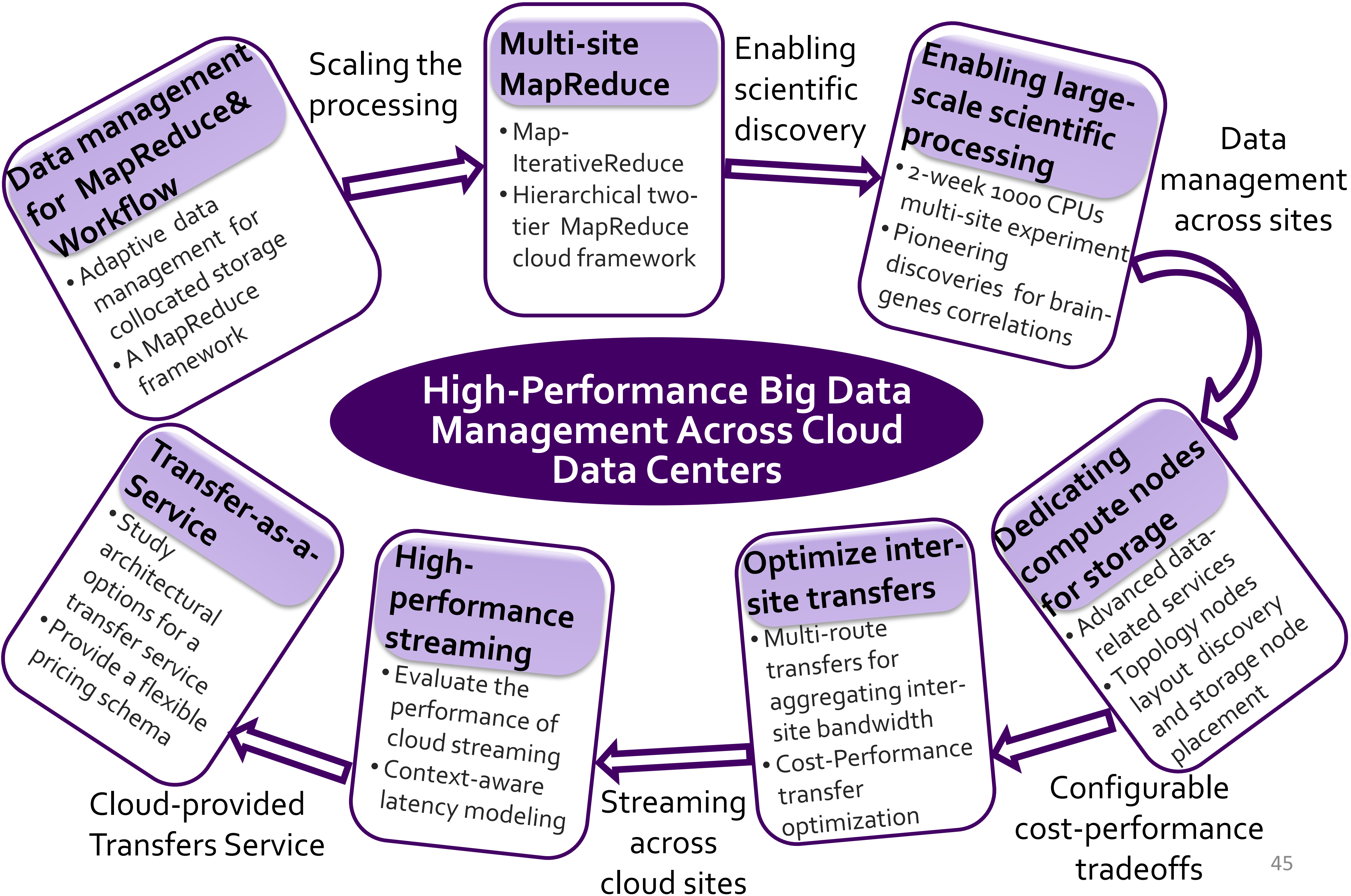
  Management of many small events, latency constraints for distributed queries

- Diversification of the cloud data management ecosystem

  One size does not fit all!

  X-as-a-Service, uniform storage across sites, API for task orchestration

Doctoral Work in a Nutshell



Backup slides

Data Centers

From few-large DCs → tomorrow → To many small DCs

http://www.extremetech.com/wp-content/uploads/2013/07/microsoft-data-center.jpg

http://www.datacenterdynamics.com/focus/archive/2014/04/huawei-launches-40ft-and-20ft-data-center-containers

Multi-site processing

- Integrated MapReduce processes across sites
- Workflow orchestration
- Site cross-scheduling of tasks

Multi-site data management

- Uniform storage across data centers
- High-performance transfer tools – Transfer as a Service
- Usage and data access patterns

Service Diversification

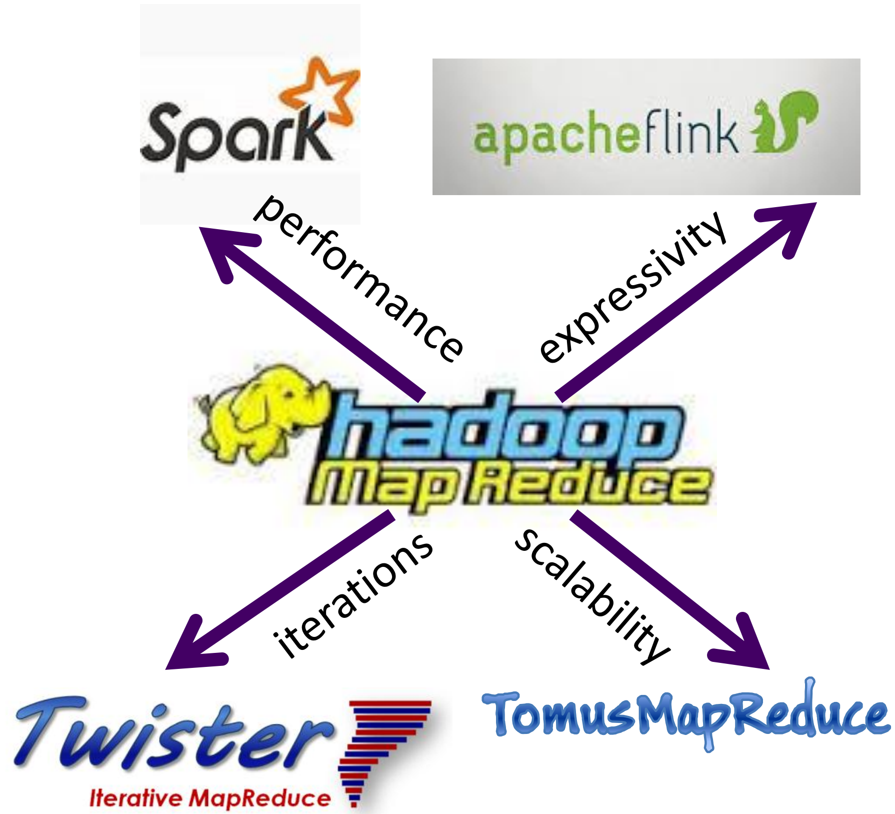
Handling Big Data grows in complexity

- Architectural design options for many items storage
- Enriched and diversified data-oriented services
- Smart replication strategies

Diversification of processing

- Customizable-user API: towards business workflows
- Solutions for providing the versatility of workflows and simplicity of MapReduce

One size does not fit all!



Lessons learned: Starting an analysis in the cloud

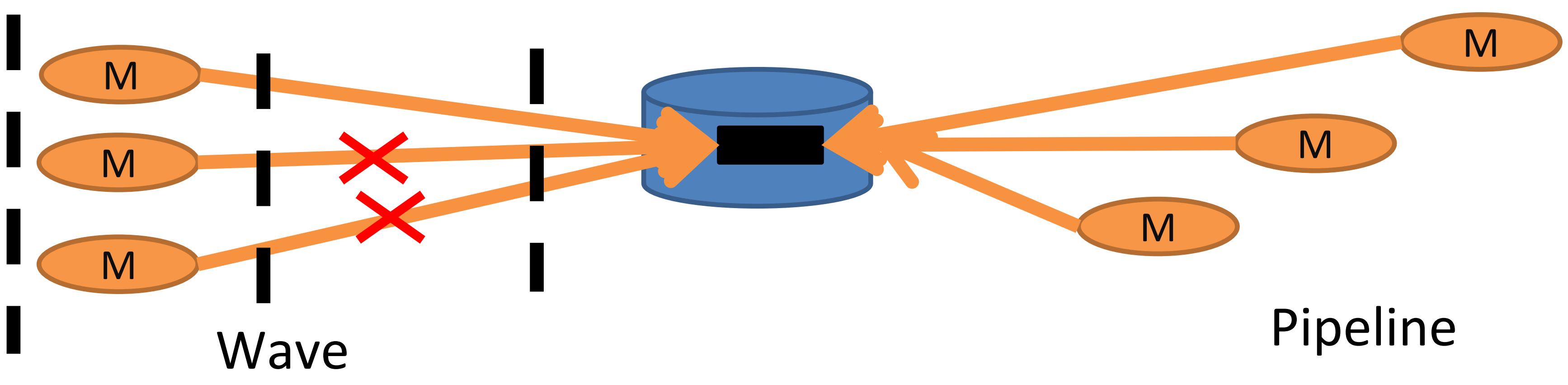
Deployment start times

- For each new or updated deployment on Azure, the fabric controller prepares the nodes → High deployment times (though better after the update from Nov. '12)
- Bigger problems reported for Amazon EC2:

  "The most common failure is an inability to acquire all of the virtual machine images you requested because insufficient resources are available. When attempting to allocate 80 cores at once, this happens fairly frequently."

  Keith R. Jackson, Lavanya Ramakrishnan, Karl J. Runge, and Rollin C. Thomas. 2010. Seeking supernovae in the clouds: a performance study. In Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing (HPDC '10).

Scheduling mechanisms for efficient data access.

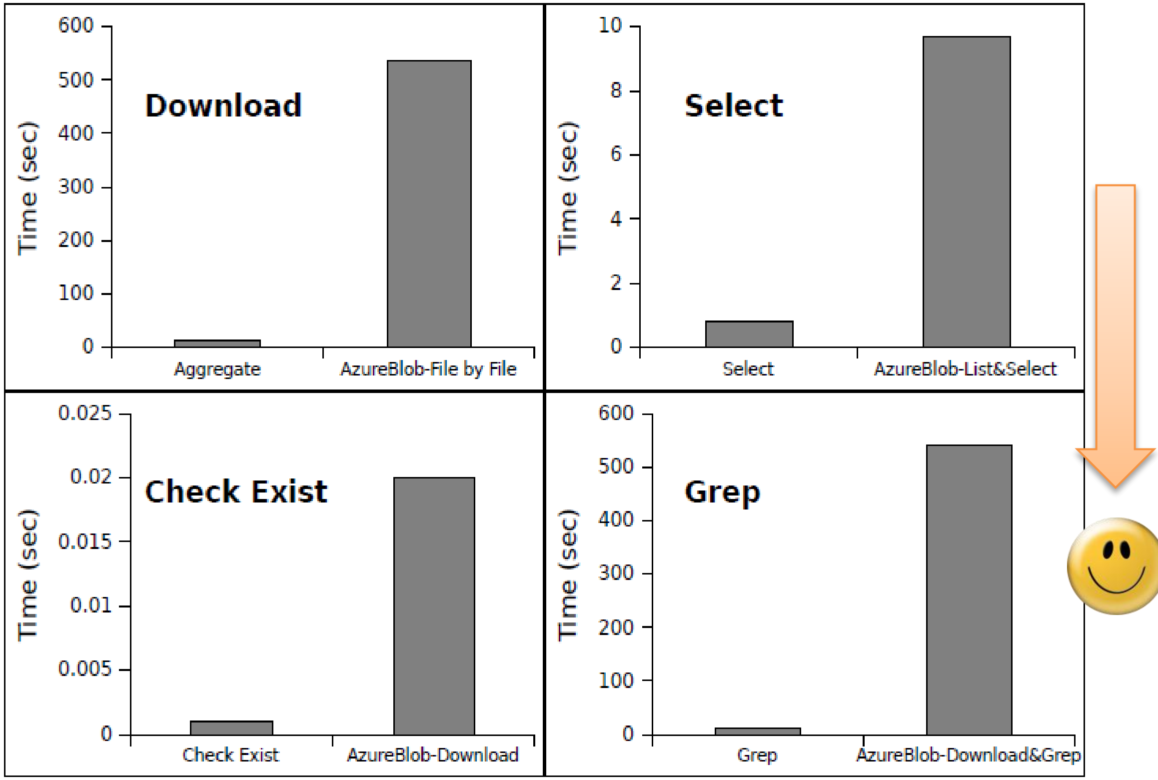


Lessons learned: running BigData applications

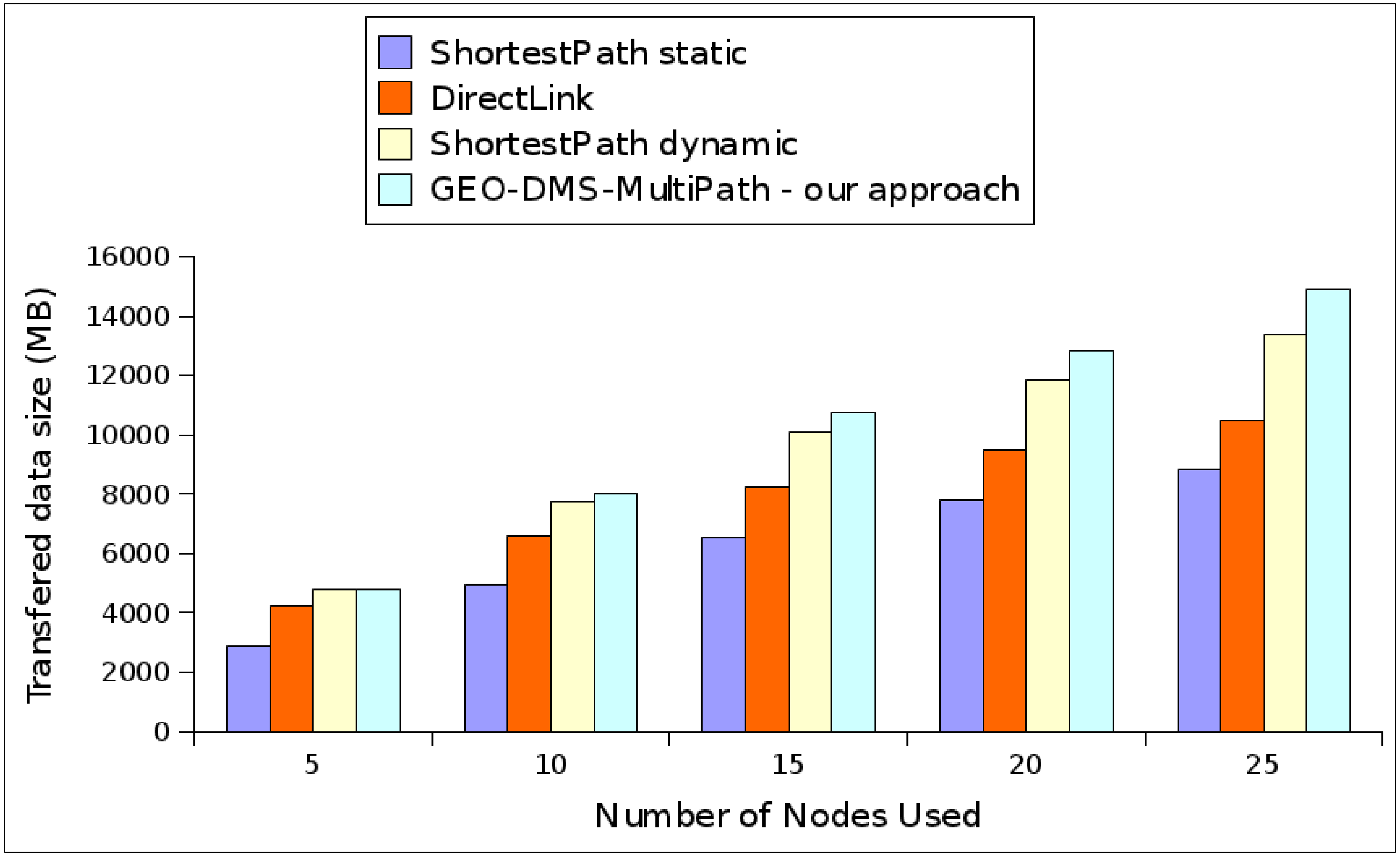
A real need for advanced data management functionality for running scientific Big Data processing in the clouds

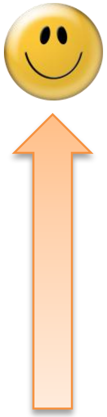
- Monitoring API
  - ➢ Monitoring and logging services for Big Data
  - ➢ Current cloud storage APIs do not support even simple operations on multiple files/blobs (e.g. grep, select/filter, compress, aggregate)

- **Data management for geo-distributed processing**
  - ➢ Cloud storage delivers poor performances → High performance alternatives
  - ➢ Inter-site data transfer is not supported → Transfer as a service



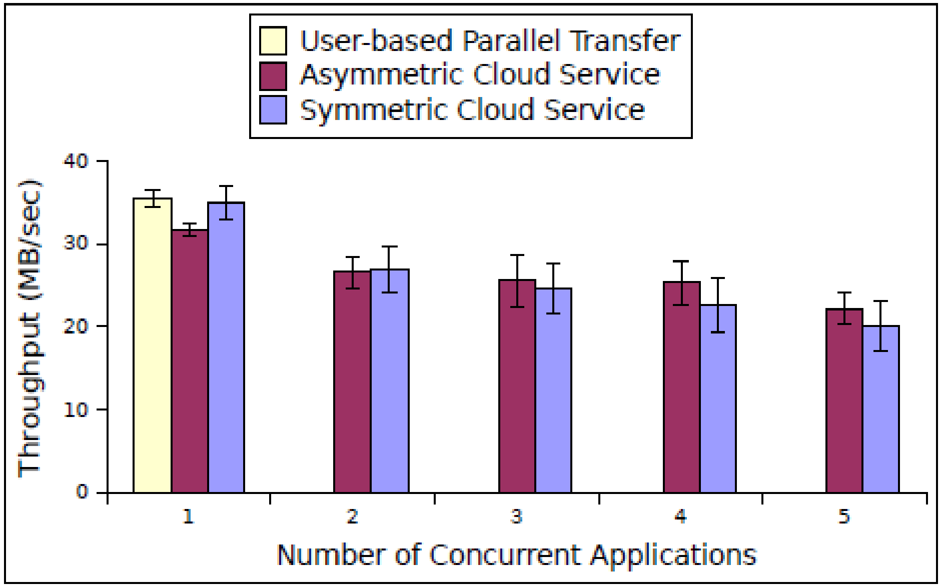
How much data can I transfer using 25 VMs for 10 minutes?



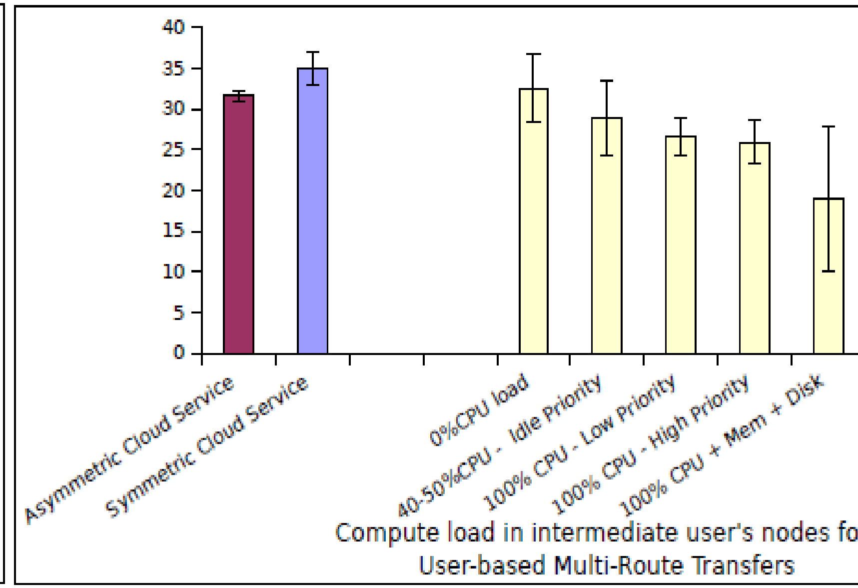


- **Experimental setup:** up to 25 nodes, Azure Cloud
- Transfers between North Central US to North EU Azure data centers

Is it feasible performance-wise?



Multi-tenancy

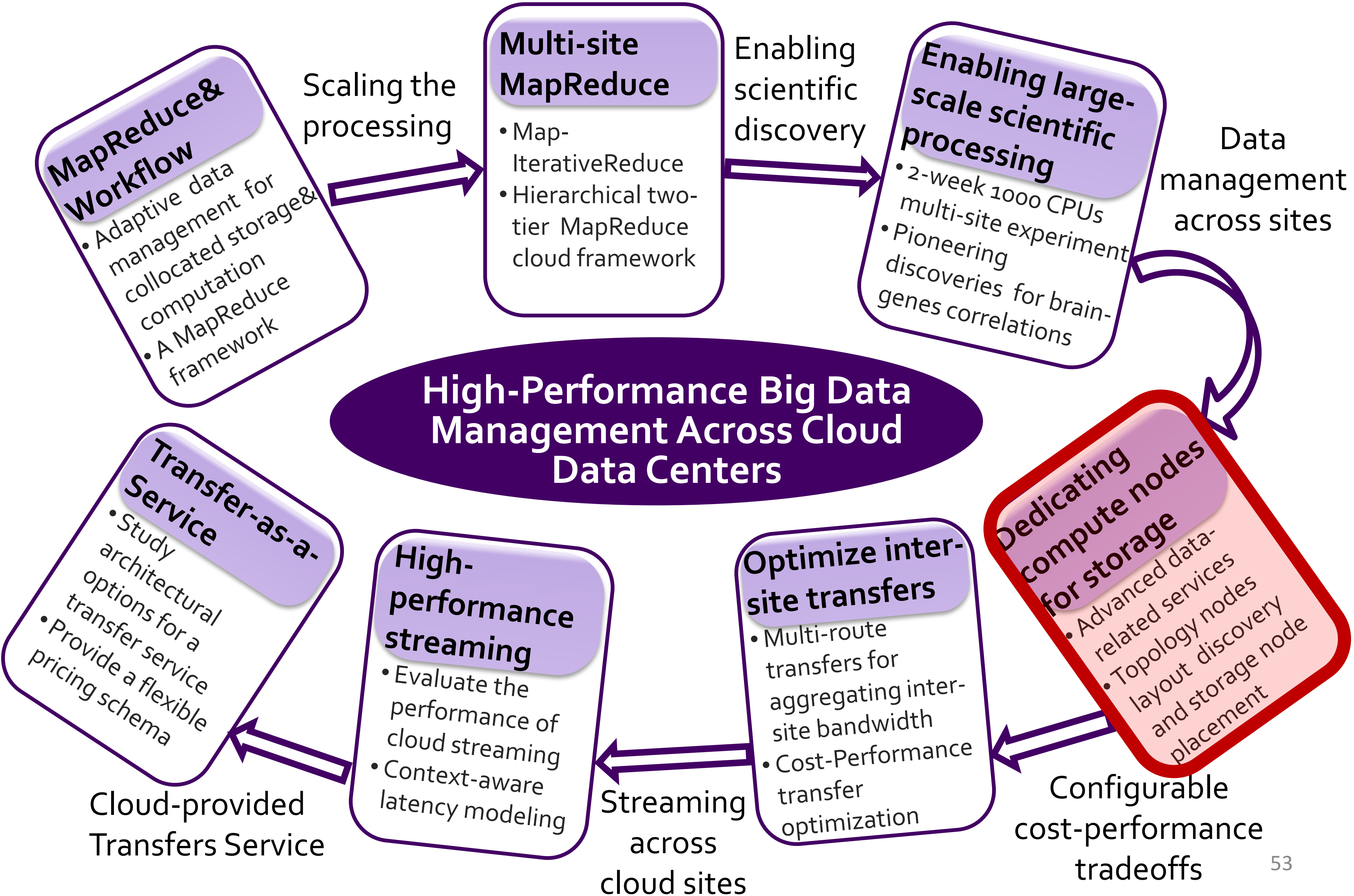


Impact of CPU Load on I/O

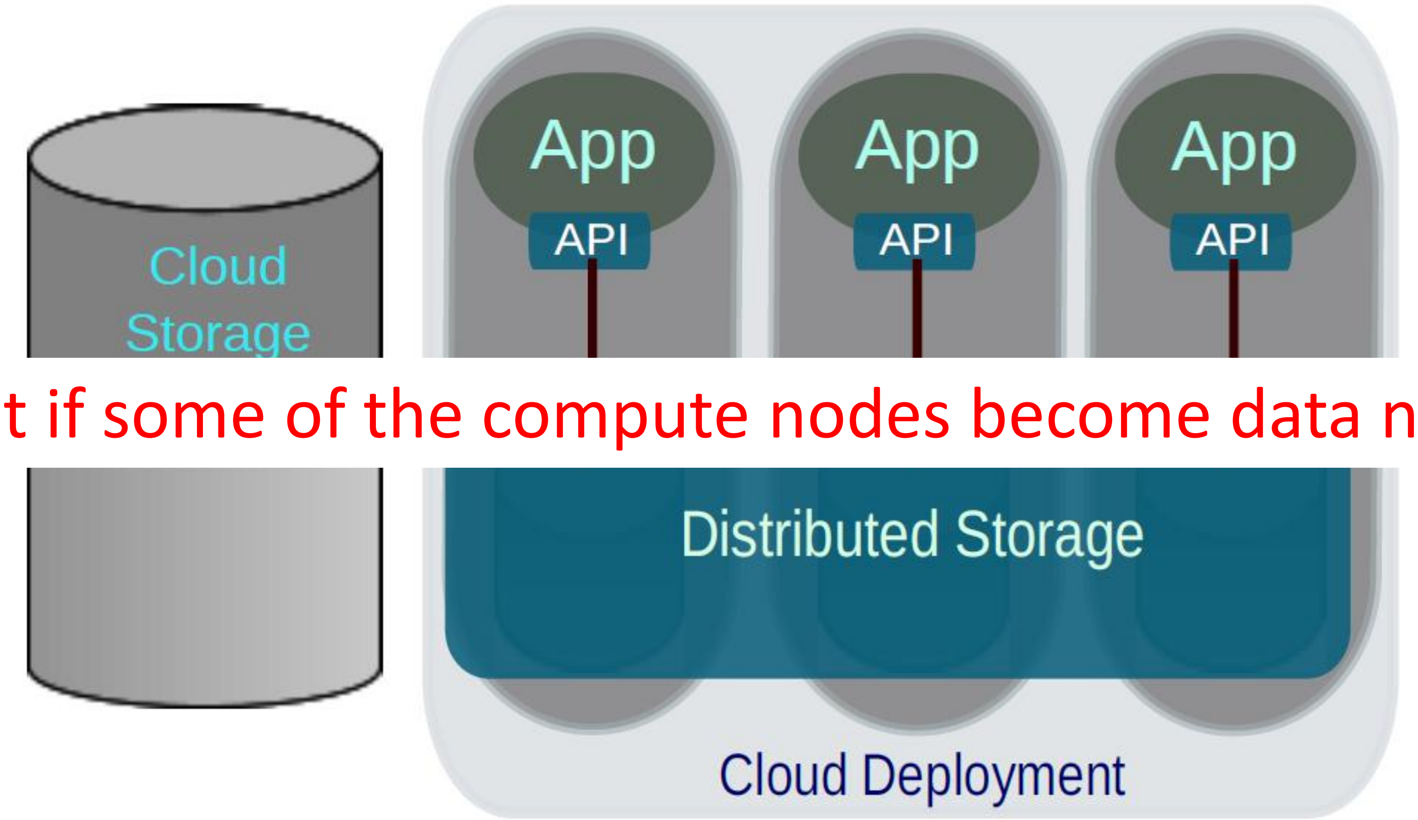
Service Access Concurrency: performance degradations of ~20% when reducing the service nodes per application from 5:1 to 1:1

CPU load on user transfer nodes: performance degradation up to 40%

Doctoral Work in a Nutshell



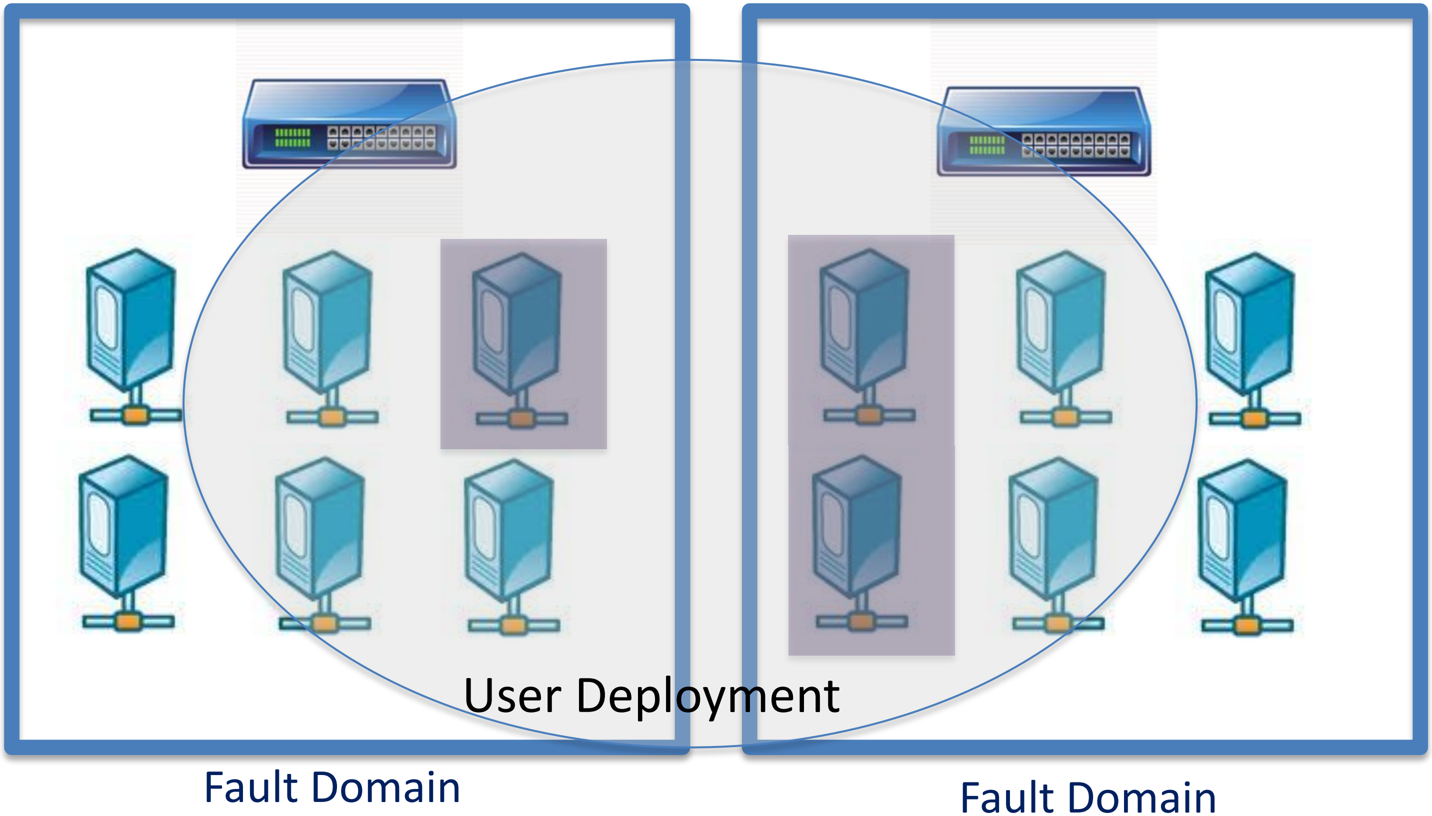
Collocating data and computation



What if some of the compute nodes become data nodes?

Beyond the Put/Get data management systems:
What is the good option to build advanced data management functionality?

Which Nodes to Dedicate?

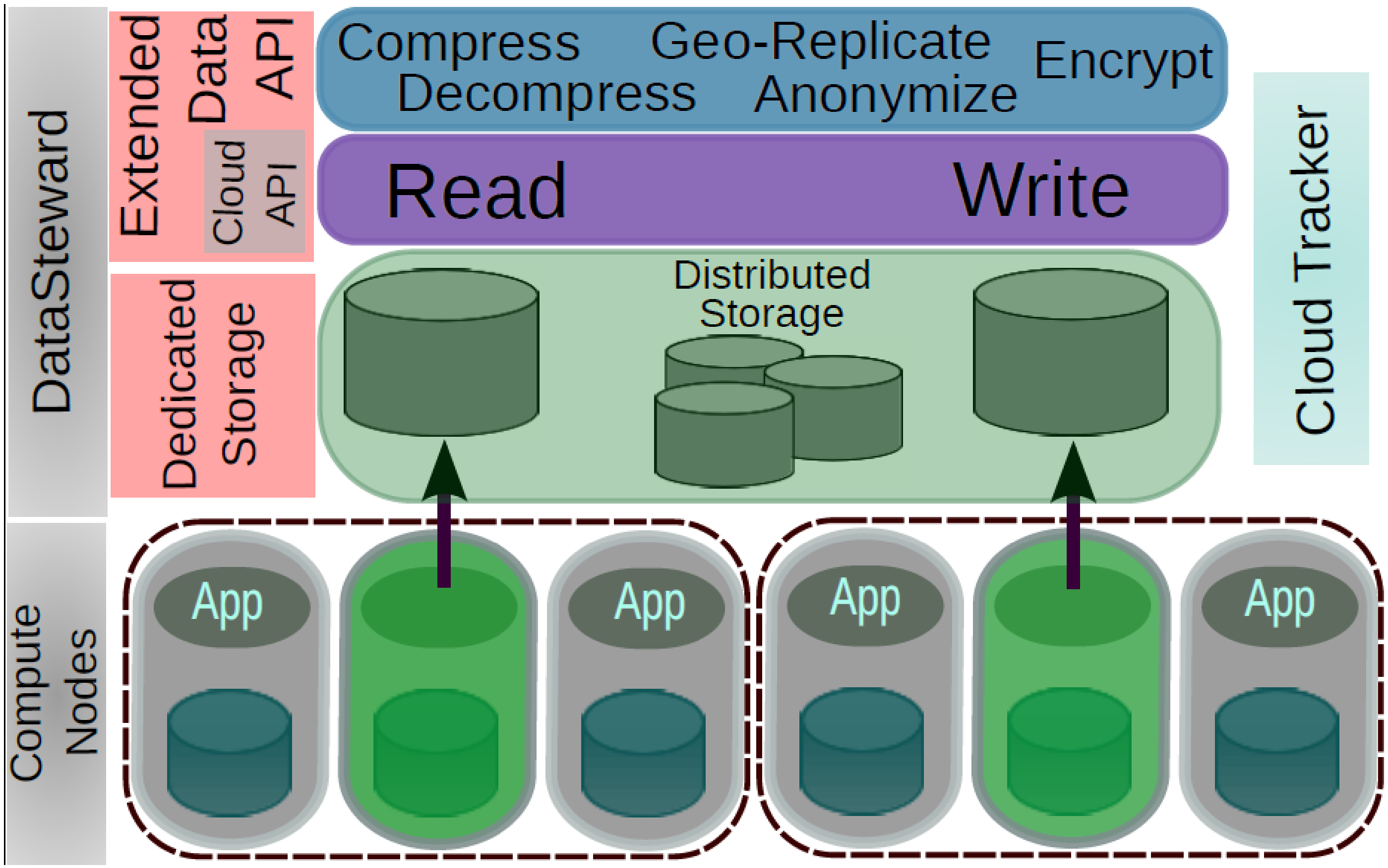


Design Principles

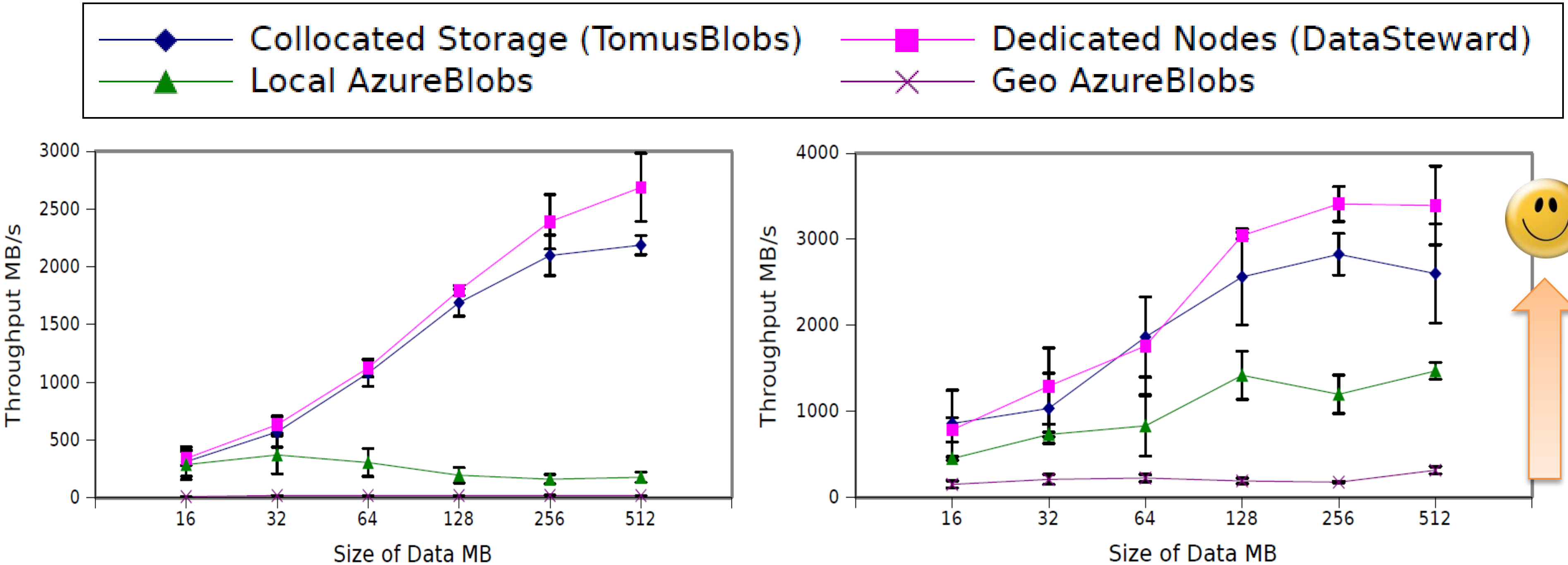
- Dedicate compute nodes for managing and storing data
- Topology awareness
- No modification to the cloud middleware

A topology-aware selection

- Discover the virtualized topology → **Clustering approach**
  - Throughput measurements between VMs
  - Asserting the performance
- Maximize throughput between application nodes and storage nodes



Assessing the storage throughput



Read throughput

Write throughput

- **Scenario:** Cumulative throughput
- **Experimental setup:** 50 client nodes, 50 storage nodes
- Transfer improvement due to CPU and network management