AWS re:Invent

A N T 3 1 2

Migrate Your Hadoop/Spark Workload to Amazon EMR and Architect It for Security and Governance on AWS

Abhishek Sinha
Principal Product Manger
Amazon Web Services

Jian Chen & Guang Yang
Software Engineer
Airbnb

Wang Cheug
Director Data Platform Architecture
Guardian Life Insurance

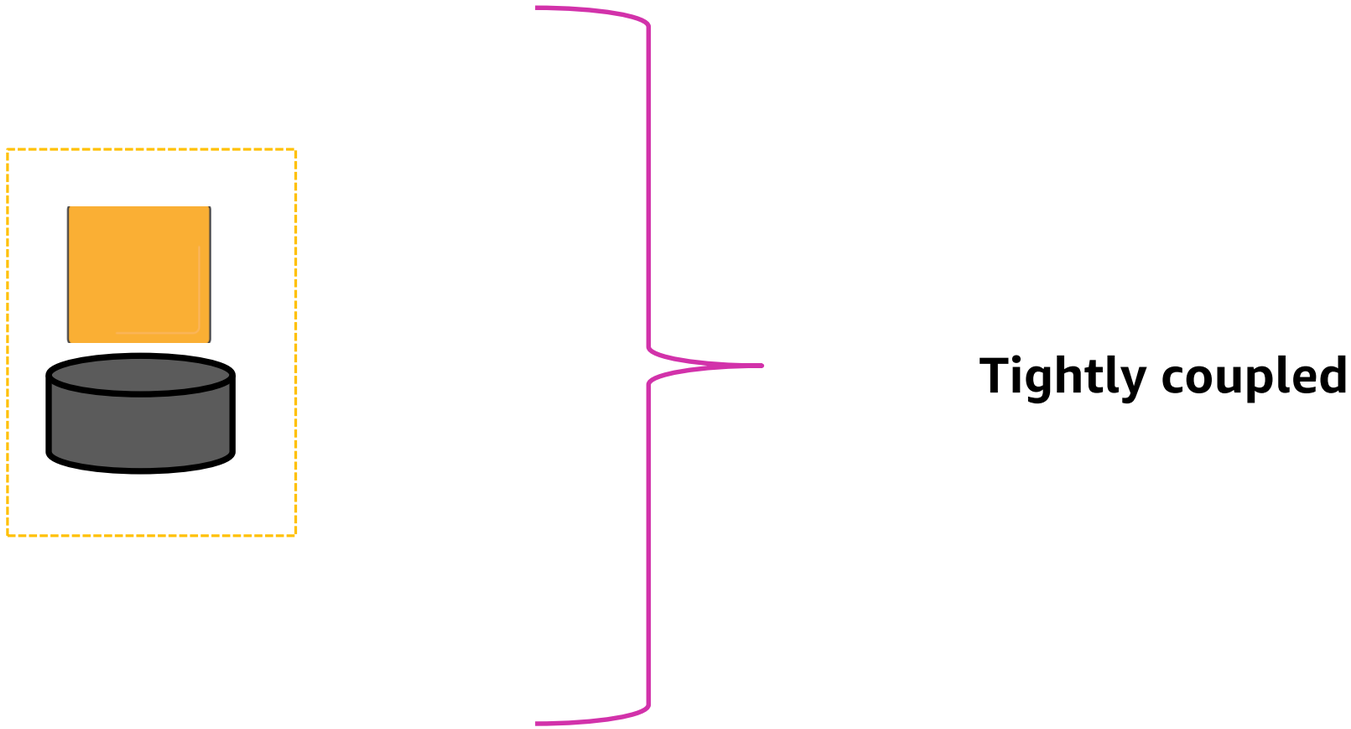
AWS re:Invent

aws

Agenda

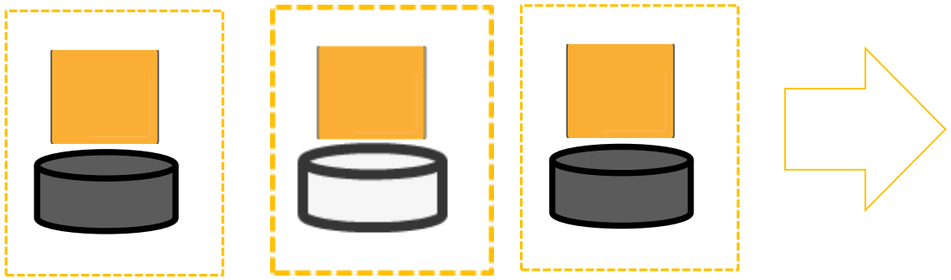
1. What are the major reasons customers migrate their on-premises Hadoop/Spark environment to Amazon EMR
2. Airbnb's story
3. Guardian Life's story

Hadoop/Spark deployments on-premises



Tightly coupled

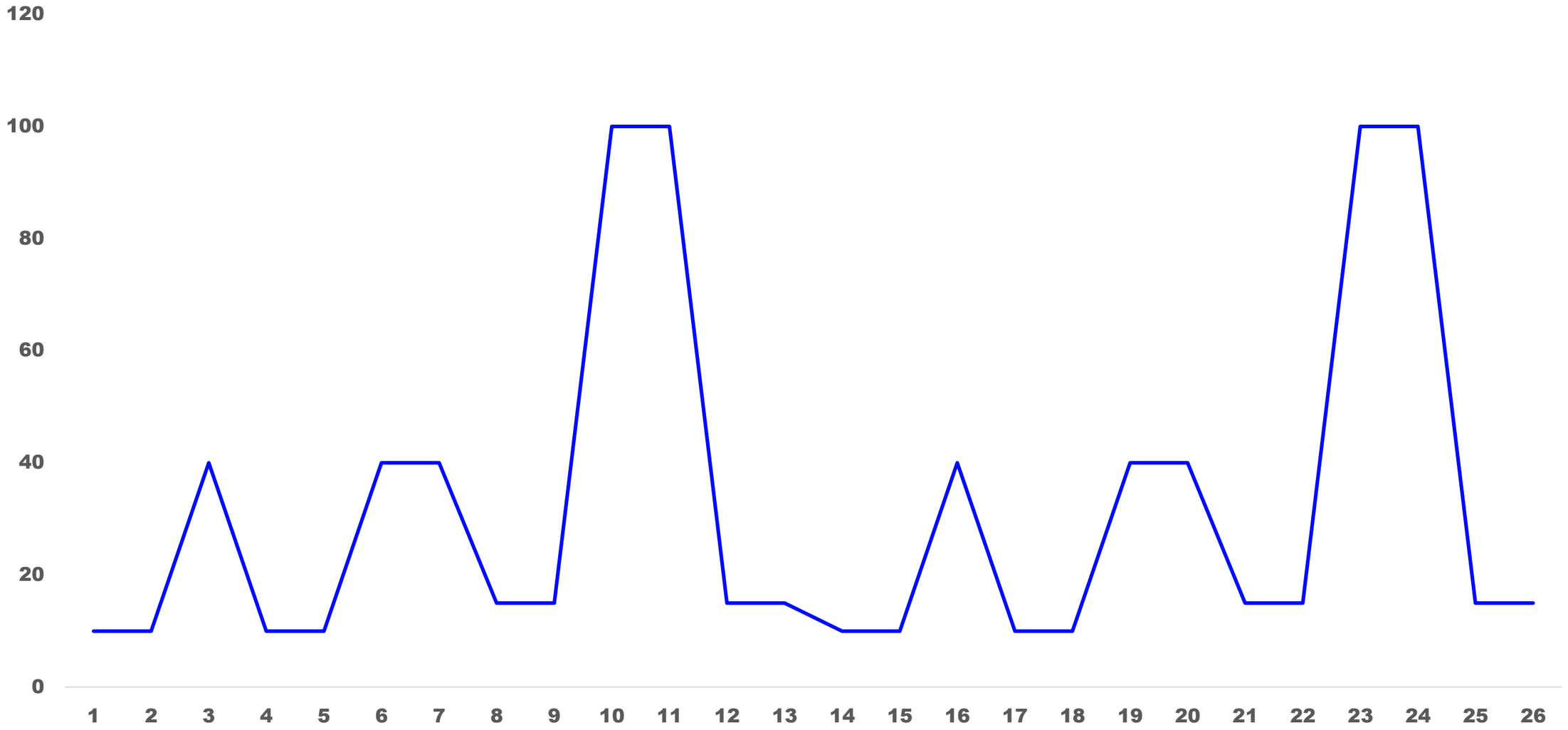
Compute and storage grow together



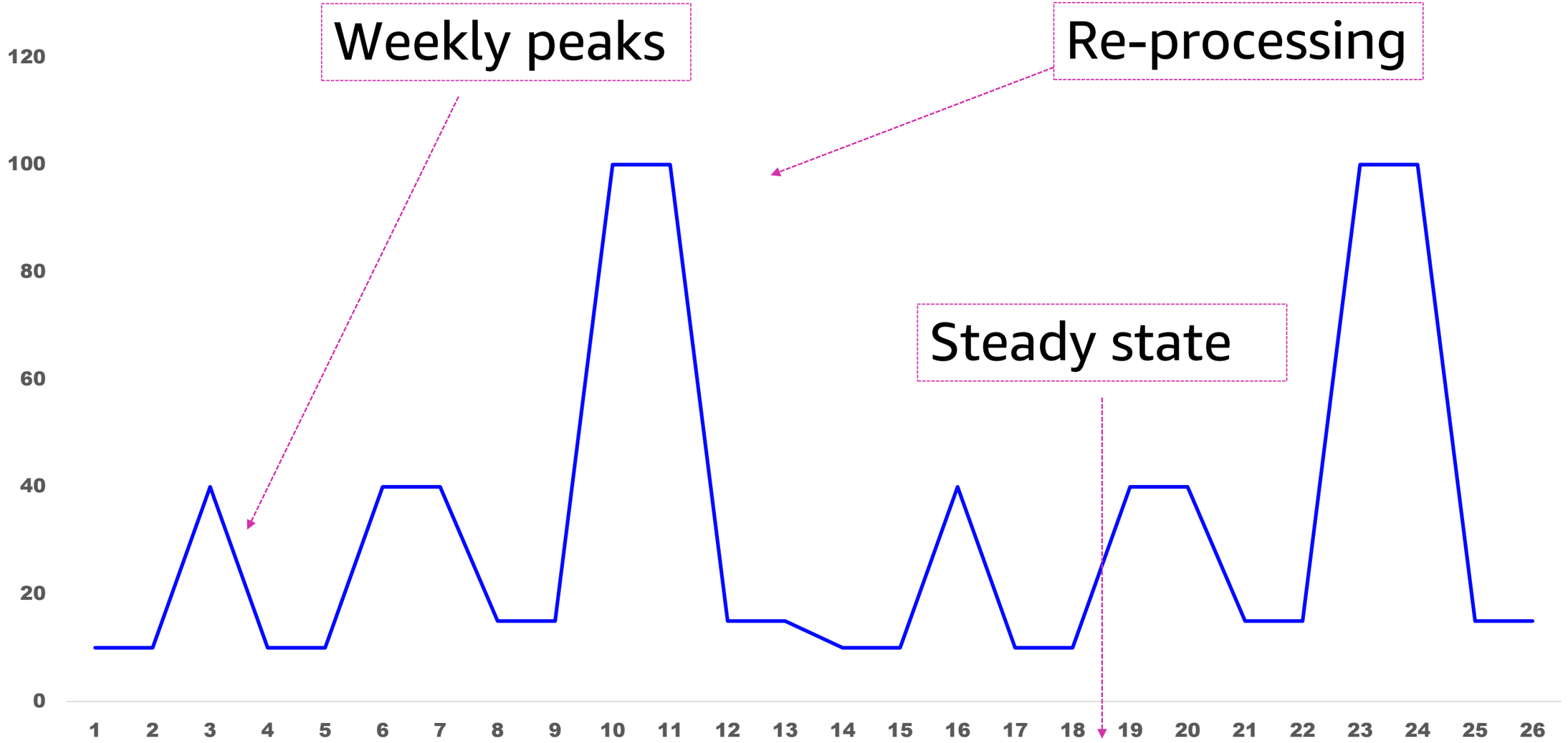
Storage grows along with compute

Compute requirements vary

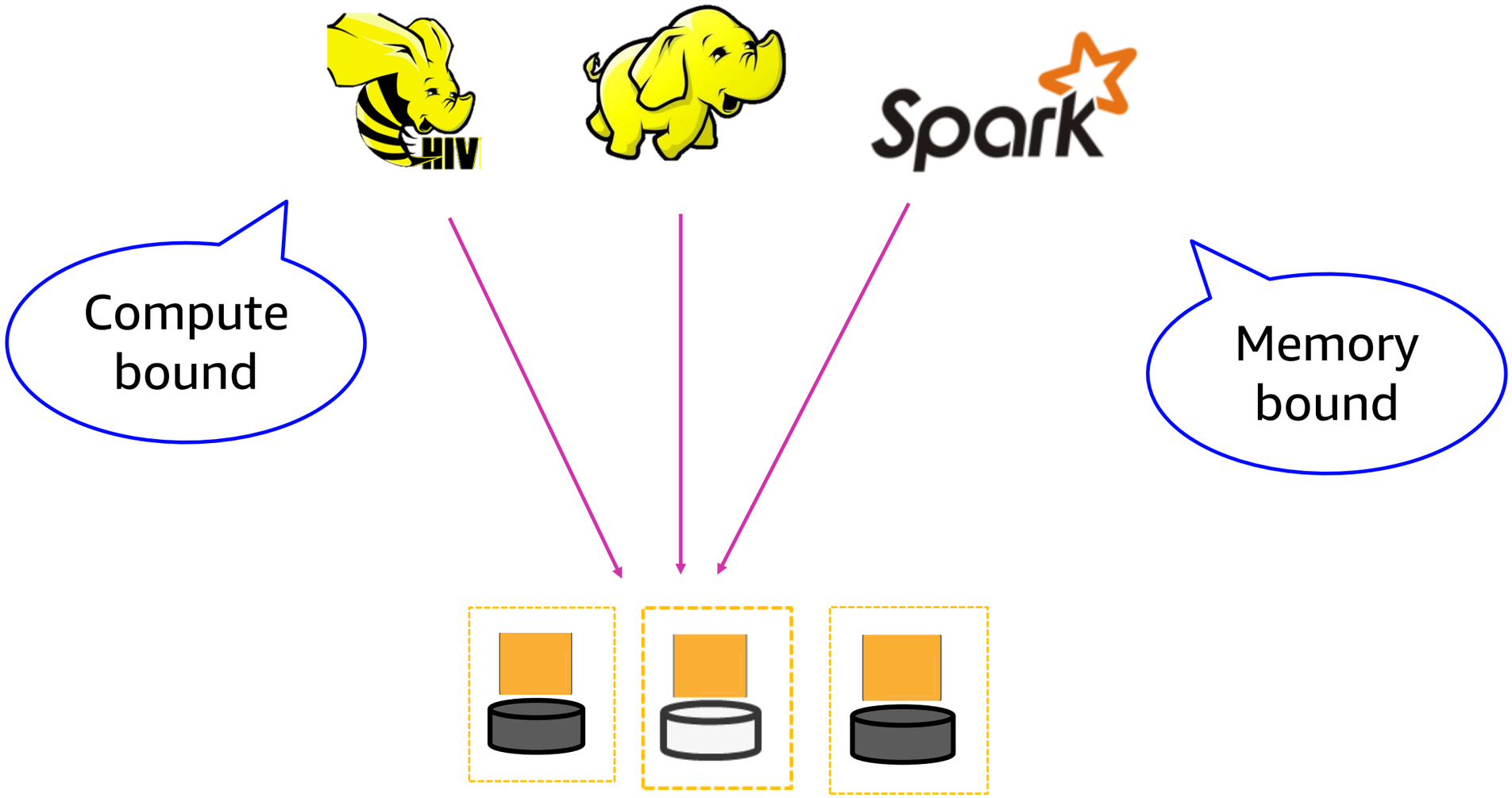
Underutilized or scarce resources



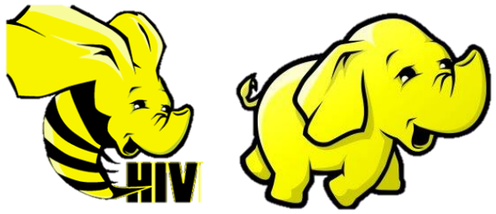
Underutilized or scarce resources



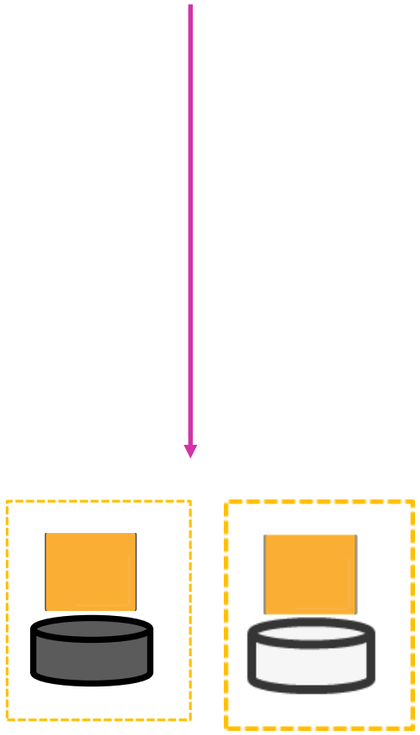
Contention for same resources

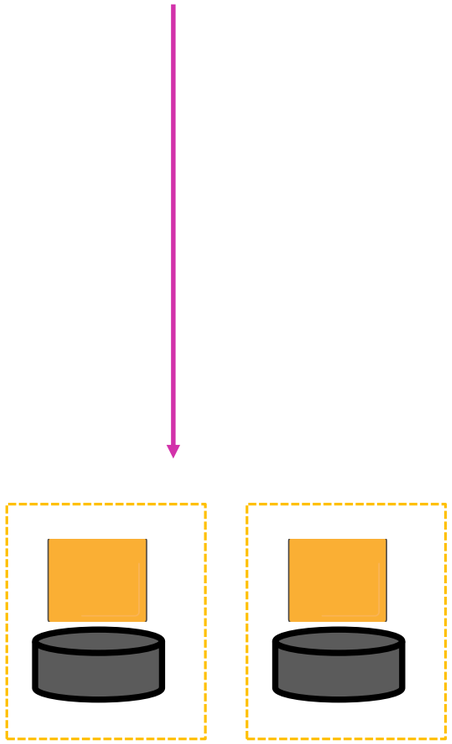


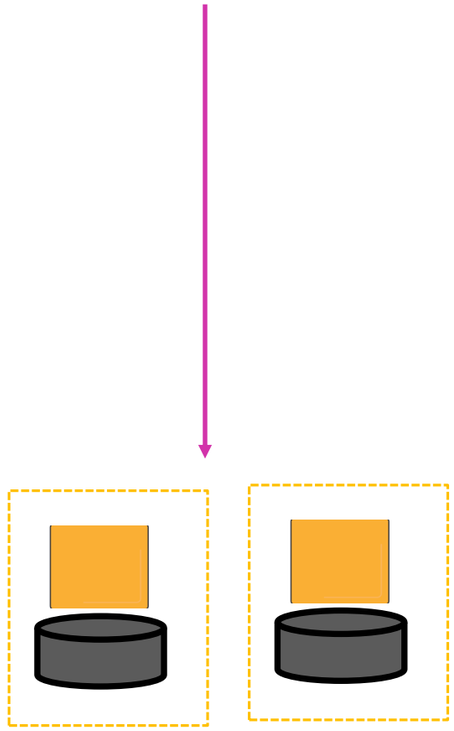
Separation of resources creates data silos



Team A







Replication adds to cost

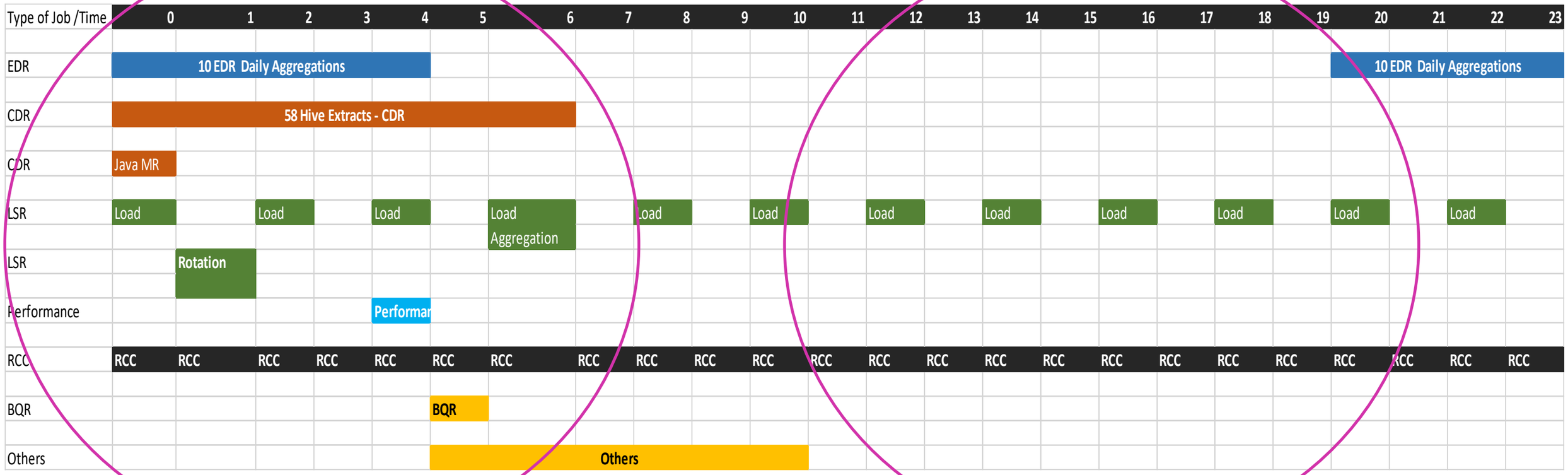
3x

Single data center

Application point of view

Large scale transformation: Map/Reduce, Hive, Pig, Spark
Interactive queries: Impala, Spark SQL, Presto
Machine learning: Spark ML, MxNet, TensorFlow
Interactive notebooks: Jupyter, Zeppelin
NoSQL: HBase

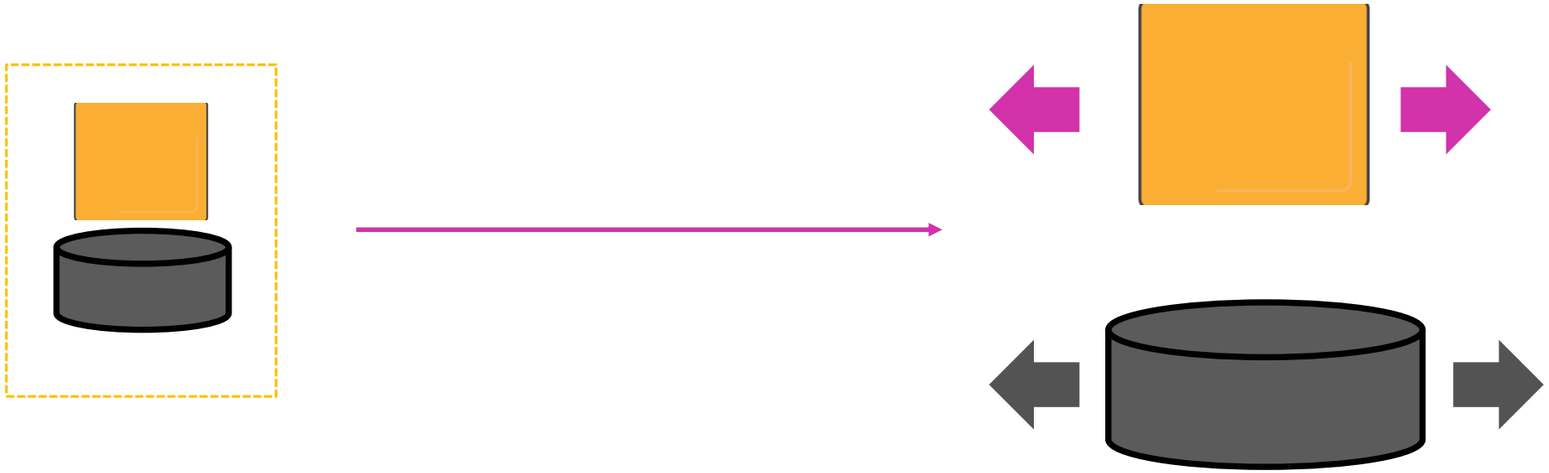
Swim lane of jobs



Over utilized

Under utilized

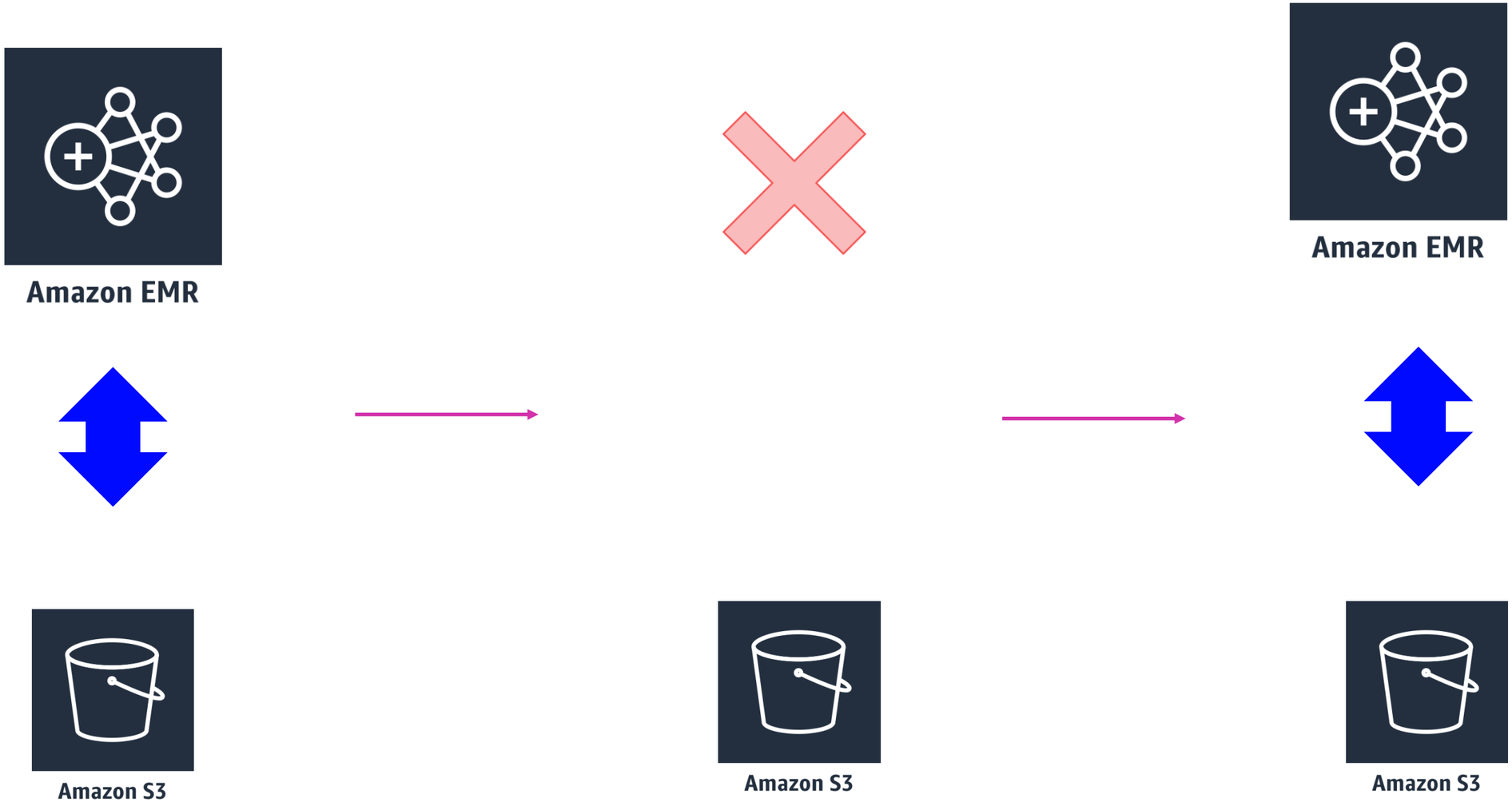
Decouple storage and compute



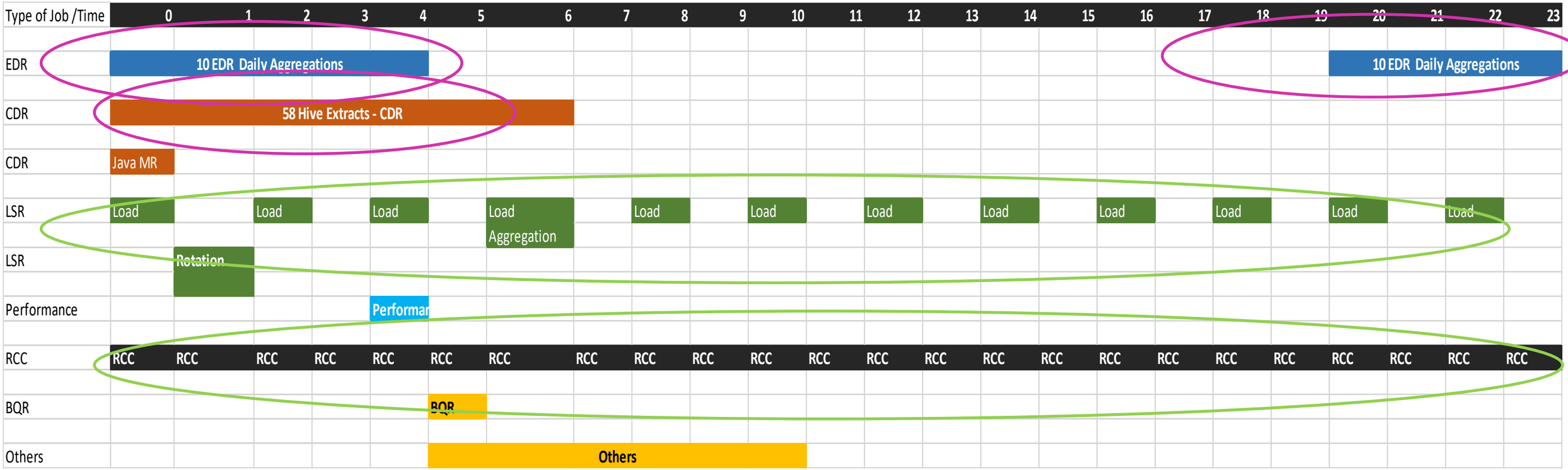
Amazon Simple Storage Service (Amazon S3) is your persistent data store

11 nines of durability
Low cost
Life cycle policies
Versioning
Distributed by default
EMR FS

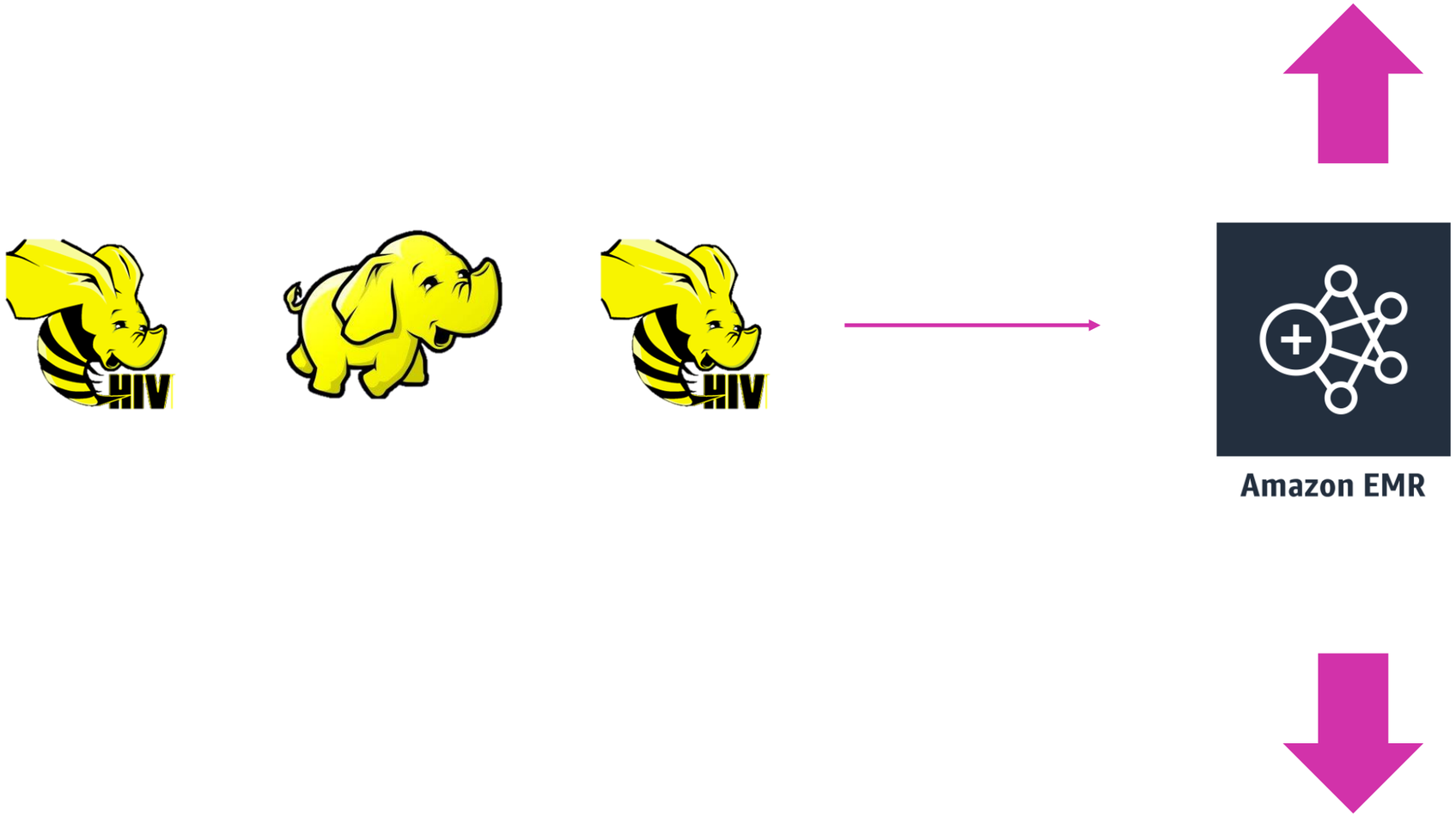
Benefit 1: Switch off clusters



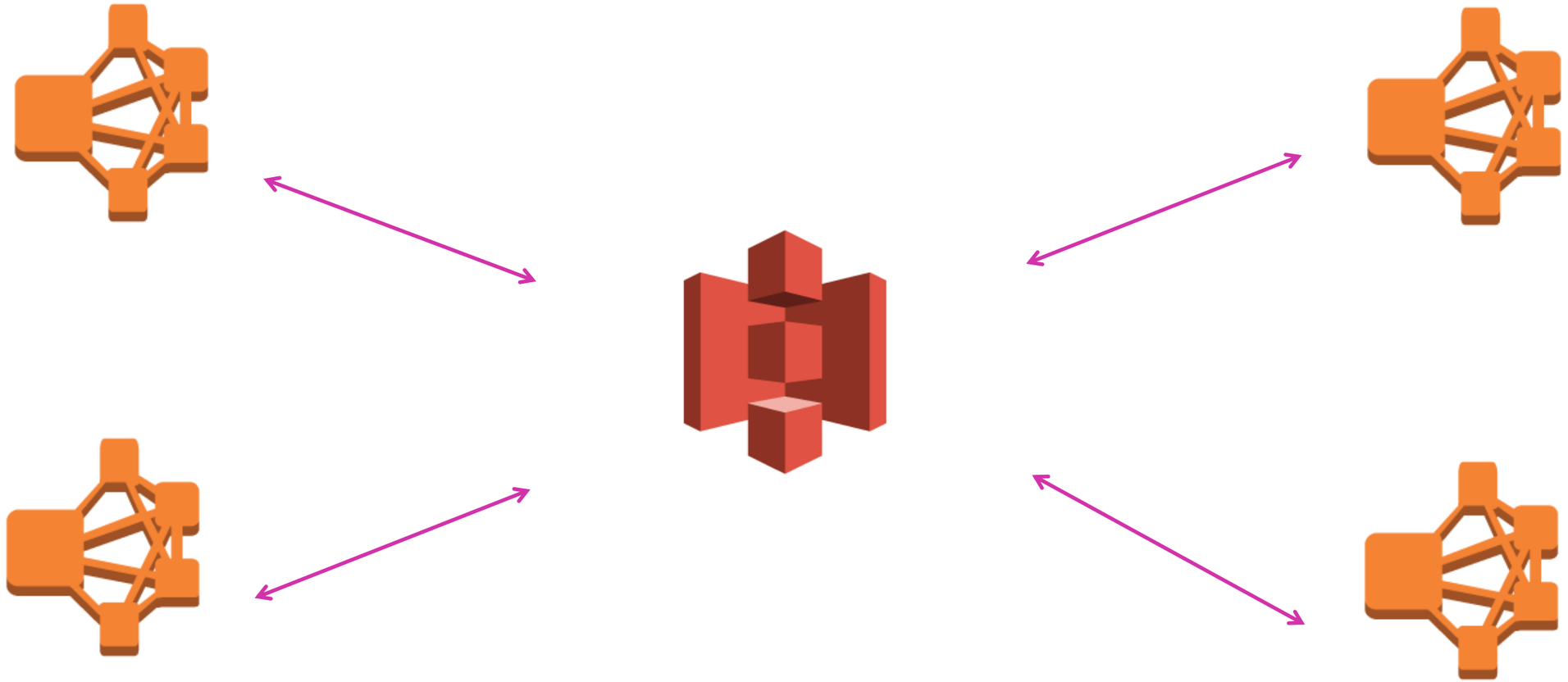
Separate them to run on persistent or long-running clusters



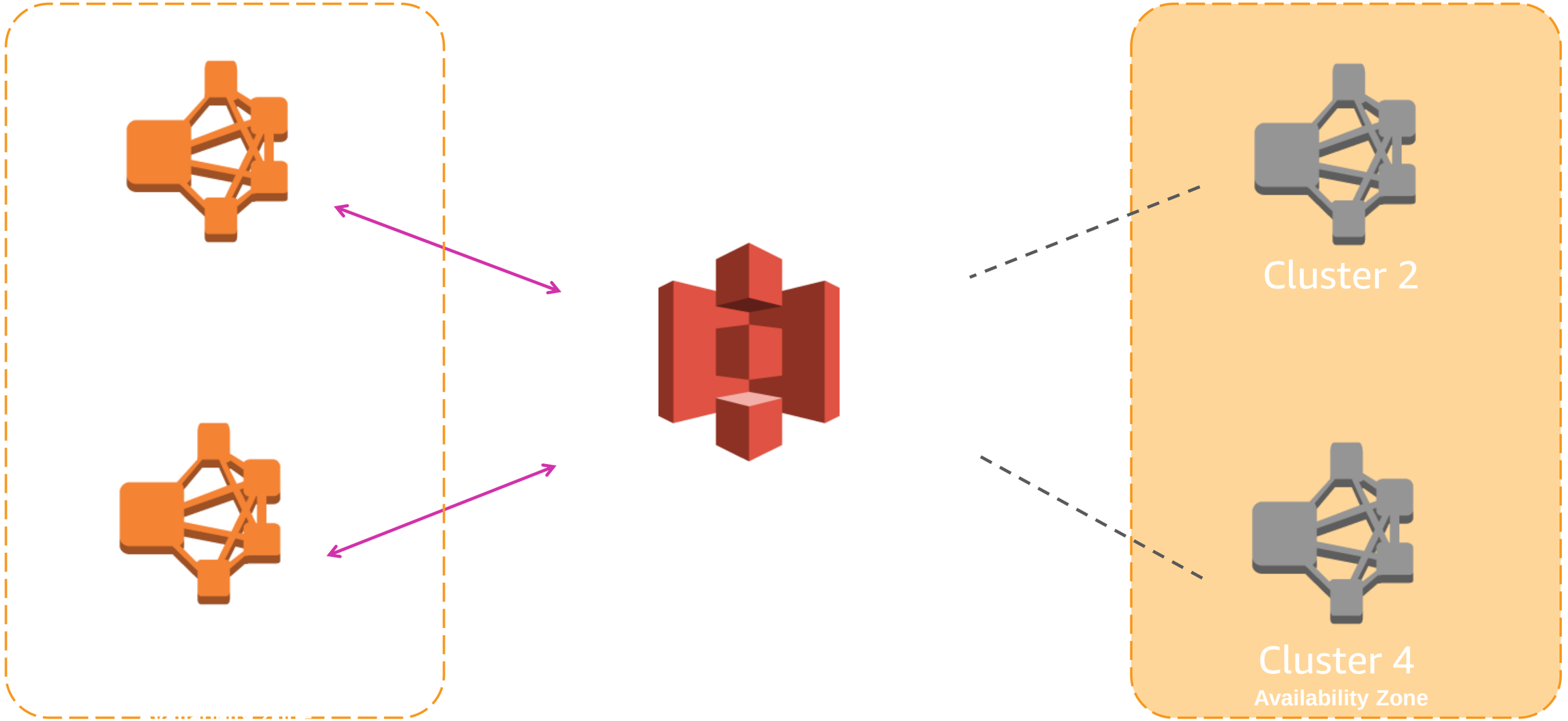
Benefit 2: Autoscale persistent clusters to save costs



Benefit 3: Logical separation of jobs



Benefit 4: Disaster recovery built in



Migrating from Amazon Elastic Compute Cloud (Amazon EC2) to Amazon EMR & Amazon S3

Jian Chen, Software Engineer
Guang Yang, Software Engineer
Airbnb

AWS re:Invent

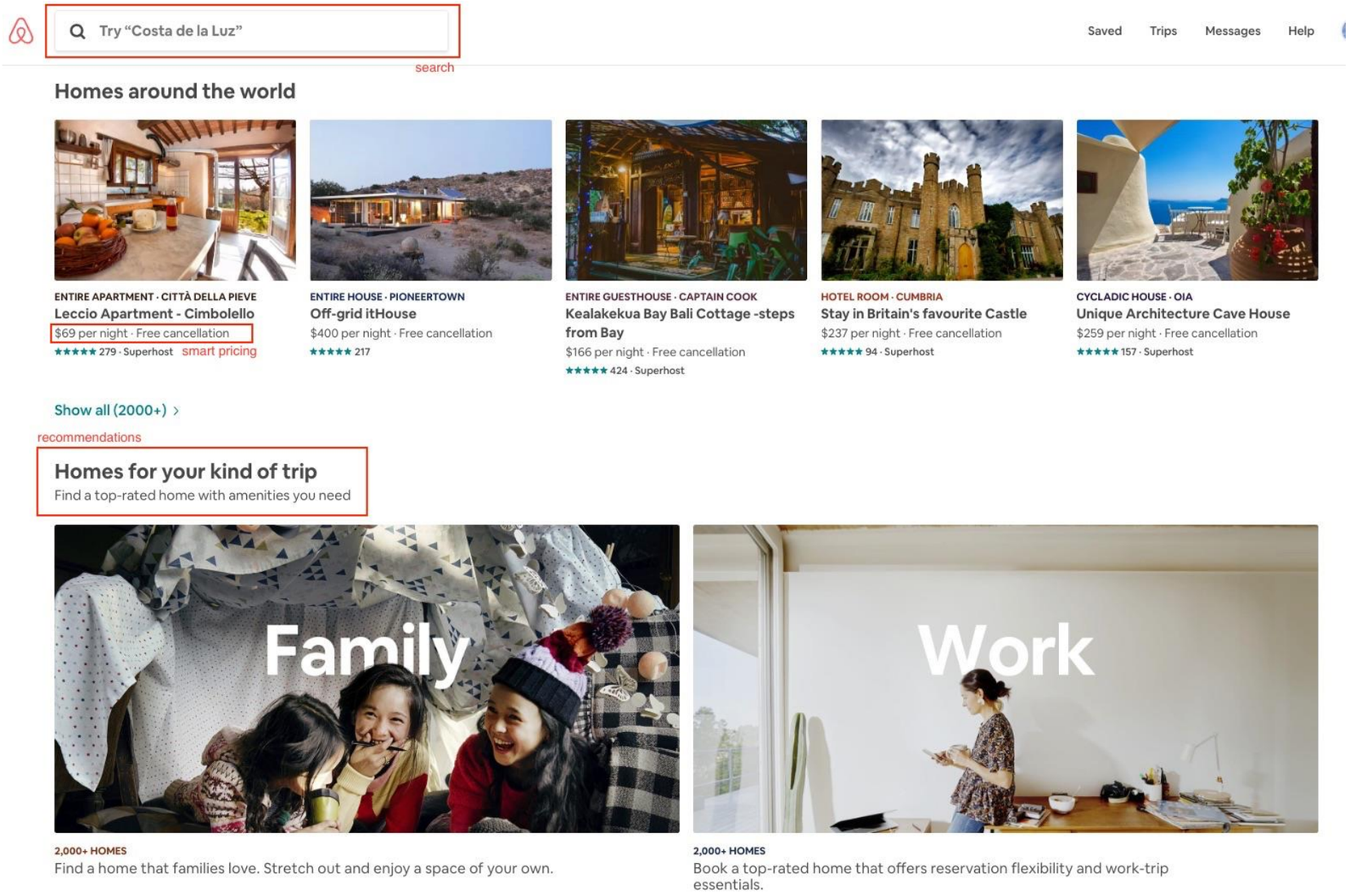
aws

Agenda

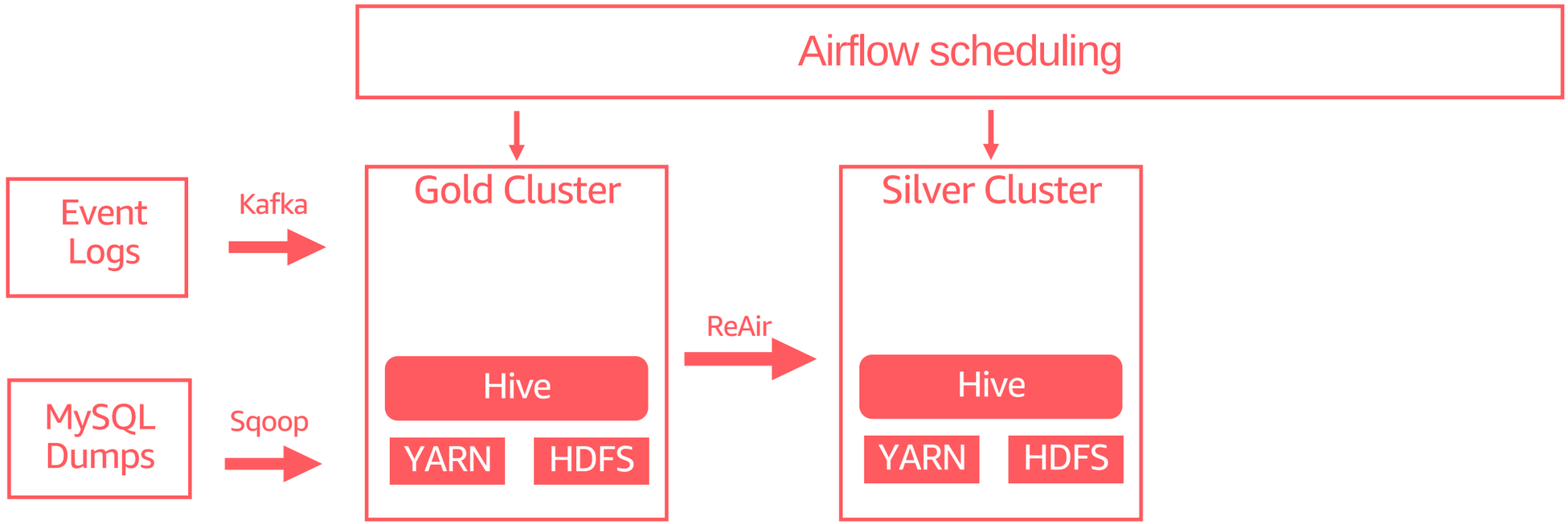
- Snapshot of data infrastructure at Airbnb
- Challenges
- Why Amazon EMR/Amazon S3
- Migration and lessons learned

Airbnb is a data driven company

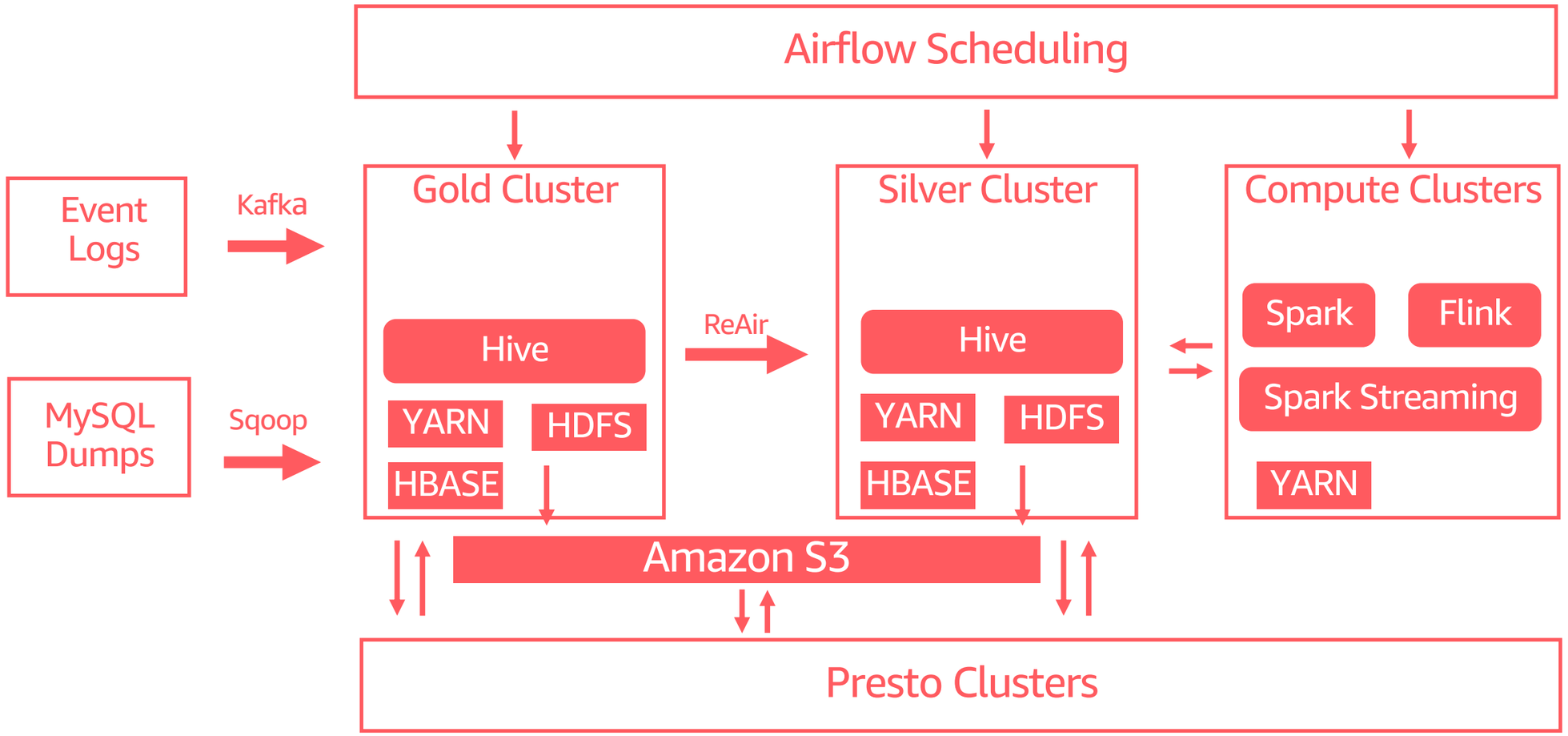
1. Recommendations
2. Search relevance
3. Smart pricing
4. Company metrics
5. Financial reports
6. Experimentation framework
7. And many more …



Data infrastructure (year 2014 - 2015)



Data infrastructure (year 2015 -)



Data infrastructure—Trending

- More Spark workloads
  - Hive to Spark, better performance, testability, and maintainability
  - Machine learning
- More streaming workloads
  - Airstream, in-house streaming framework on top of Spark streaming
  - Apache Flink

Growth

- Business
- Employees (2x YoY)
- Datasets (3x YoY)
- Number of jobs (2x YoY)

Compute and storage are tightly coupled

Scalability issues

- 1000 x d2.8xlarge instances
  - YARN does not schedule jobs even there are available resources
- HDFS: 150 MM blocks
  - Long GC pauses causing fail-over
  - Takes more than 10 hours to rolling restart the cluster
  - Several major outages

Lack of elasticity

Old Hadoop hardware stack—Hard to upgrade

- Hadoop 2.5
- Hive 0.13
- OS upgrade for security patches

High CapEx and OpEx

- High CapEx
  - Provision for peak load
- Hard to maintain
  - Hard to add instances, even harder to remove
- Very hard to allocate costs across the organization with multi-tenanted clusters

Why Amazon EMR/Amazon S3?

decouple compute and storage

- Amazon S3 as the data lake
- Stateless compute infrastructure using Amazon EMR
  - Easy to set up, rotate, upgrade, scale out/in
- Better AWS integration
  - EMRFs, spot instances, connectors with various AWS services
- EMR clusters for each business unit
  - Better isolation
  - Cost attribution
  - Customized software/hardware

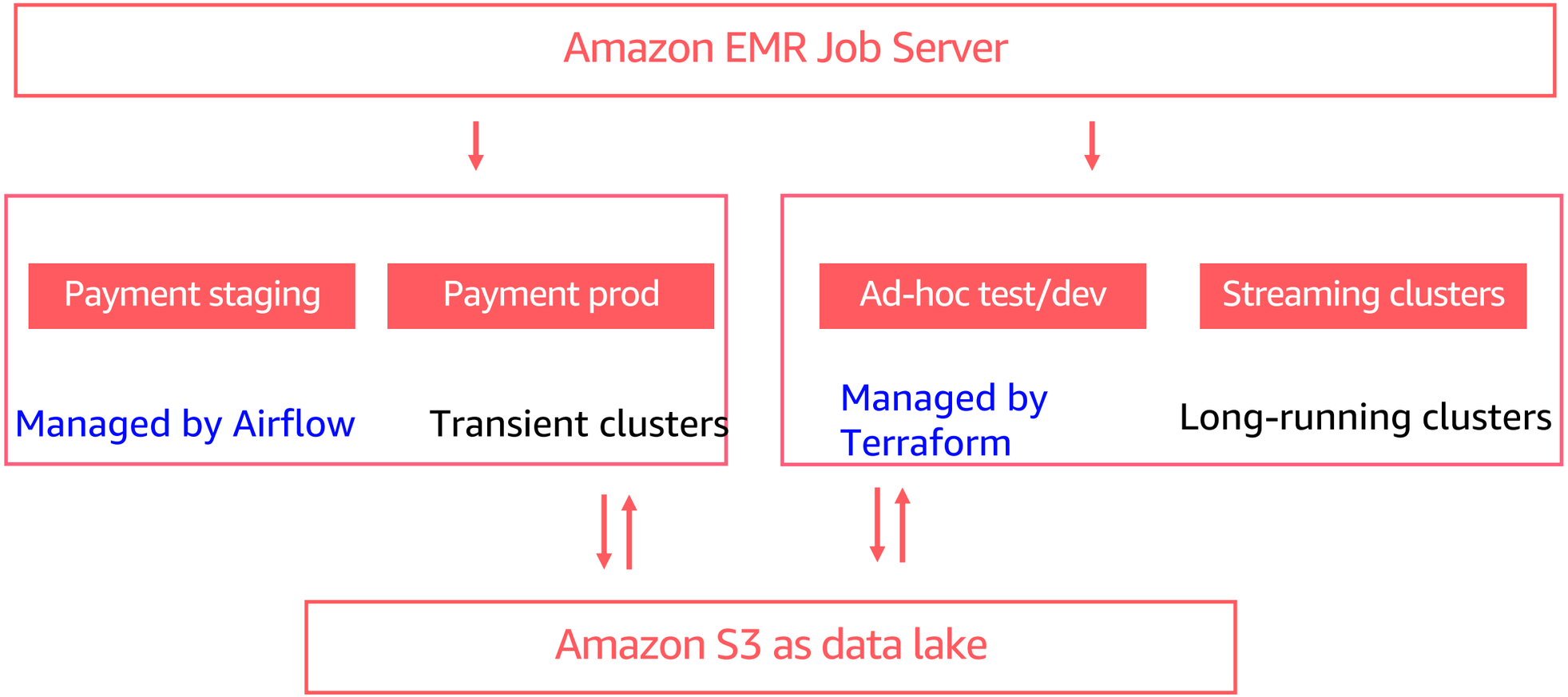
Migration path

Sounds great … but how do we get there?

- User only writes to HDFS, we archive the data to Amazon S3 later
- Started Amazon EMR migration conservatively
  - Non-critical jobs as pilot use case
  - Long-running clusters with auto-scaling
  - Spark first with Hive-0.13 client, to learn/tune the system
- Four production long running clusters for Spark (with auto-scaling)
  - Jobs run 3x faster

- Hive migration
- Amazon EMR job server
- Transient clusters

New architecture



Lesson learned

- Get it working first
  - r/w to existing HDFS, Hive Metastore, setup gateways
- Start with less hardware variations
  - r4.8xl with different size of EBS volume
- Pick representative use cases, make them happy
- Default configuration may not work well
  - Heap size, yarn conf
- Auto-scaling is awesome
  - Long-running cluster with auto-scaling

Guardian Life's migration to Amazon EMR

Wang Cheung
Director, Data Platform Architecture

AWS re:Invent

aws

About Guardian

158-year-old mutual company
Fortune 239 ranking

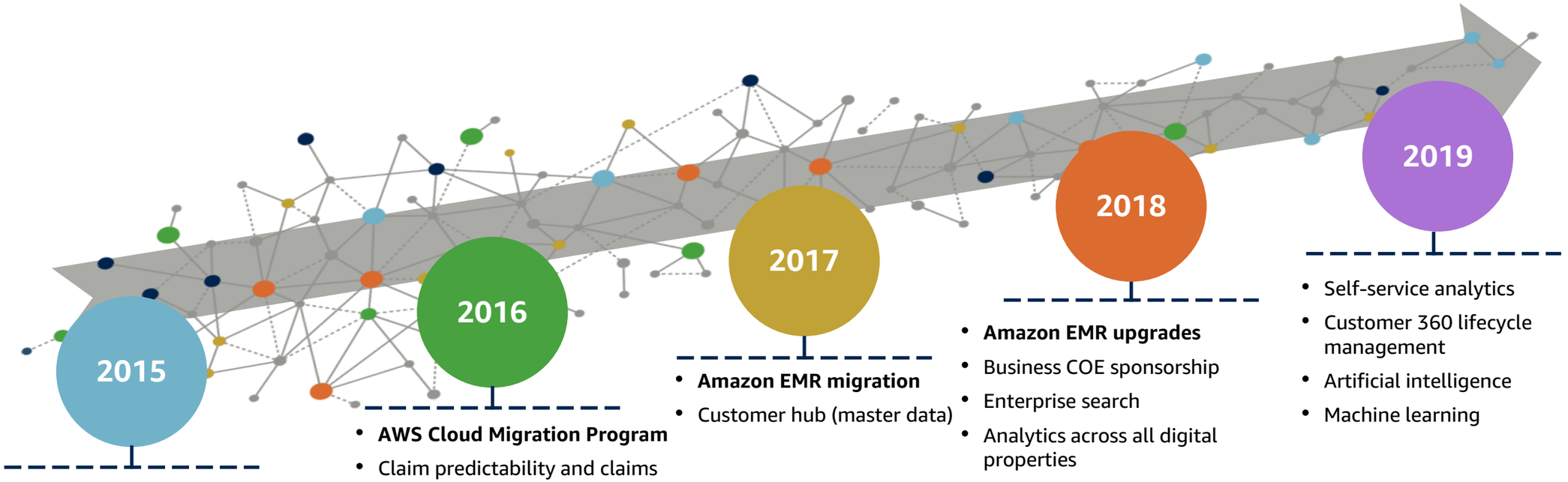
9000 Employees
Over2,750 financial representatives and more than 55 agencies

Annuities
Investment
Life insurance
Dental insurance
Employee benefits
Disability income
Insurance

For more information, visit Guardian's website: www.GuardianLife.com

History lesson—No enterprise data

Guardian big data journey



- **Launched big data platform**
- Guardian Data Council established
- Built Enterprise Data Marketplace
- Ingested group data
- LOB Data Summits
- Initiated advanced analytics primers

- **AWS Cloud Migration Program**
- Claim predictability and claims termination advanced analytics
- Client profiling & segmentation

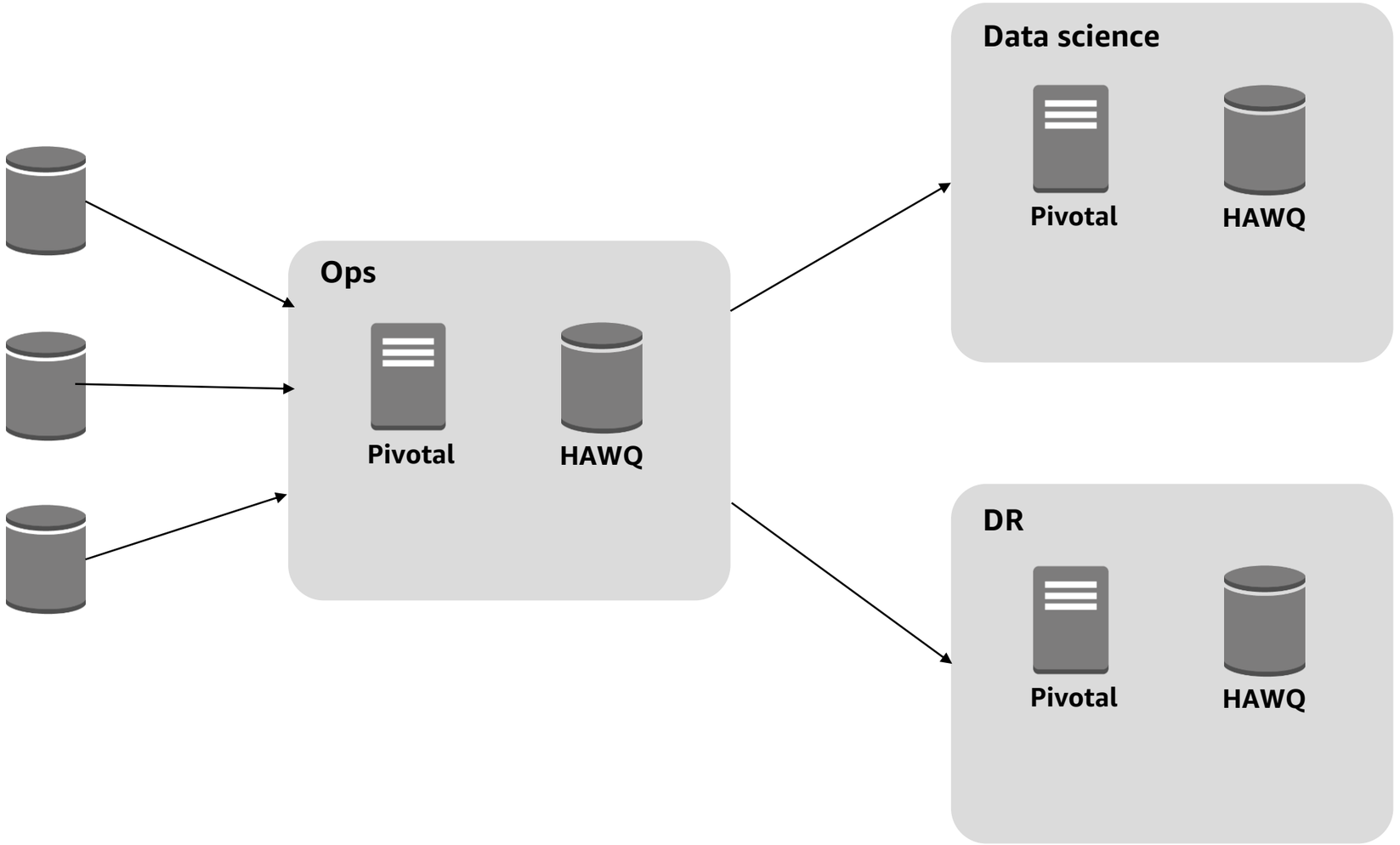
- **Amazon EMR migration**
- Customer hub (master data)

- **Amazon EMR upgrades**
- Business COE sponsorship
- Enterprise search
- Analytics across all digital properties
- Prototyping artificial intelligence
- Prototyping machine learning

- Self-service analytics
- Customer 360 lifecycle management
- Artificial intelligence
- Machine learning

Why did we migrate to Amazon EMR?

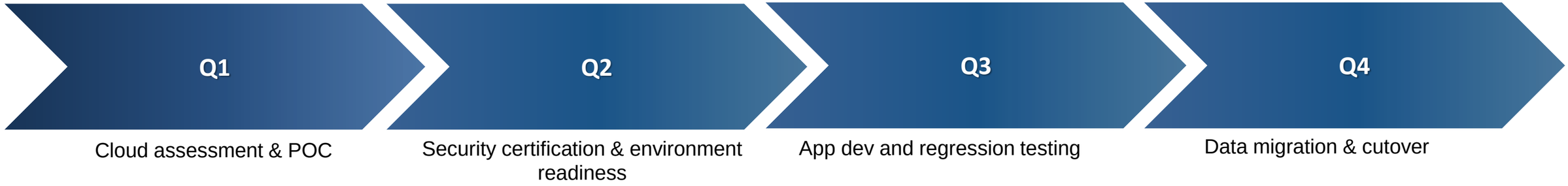
On-premises Hadoop architecture

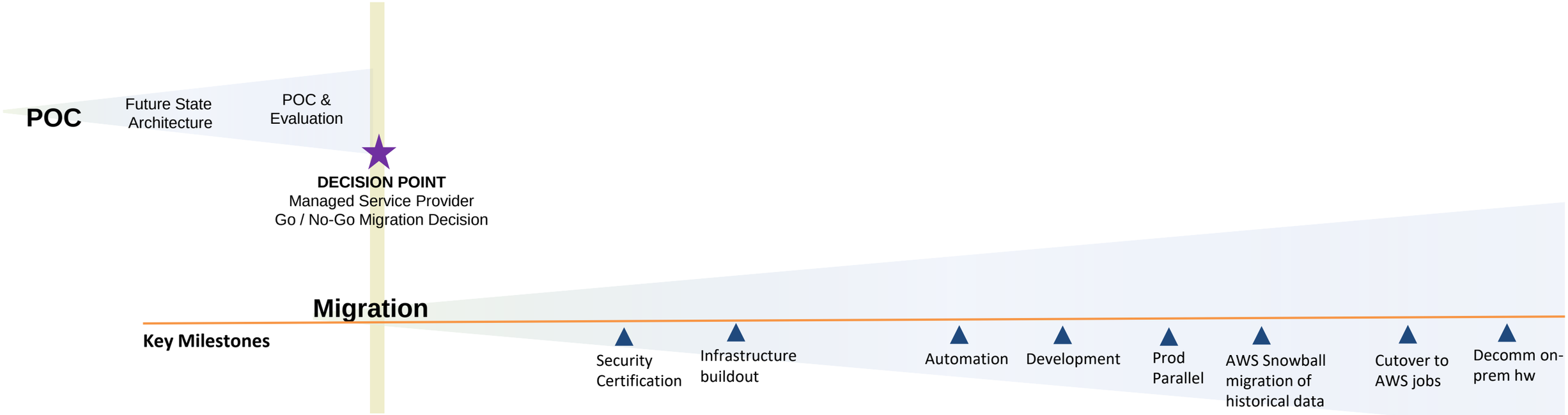


On-prem Hadoop challenges

- Combined storage and compute on shared servers
- Fixed storage
- Disk failures
- Inability to quickly scale
- Costly TCO – multiple clusters
- Costly DR – third-party software
- Unused capacity during off-peak periods
- Team of dedicated operators to maintain hardware
- Slow adoption to address changing business needs

Amazon EMR migration strategy

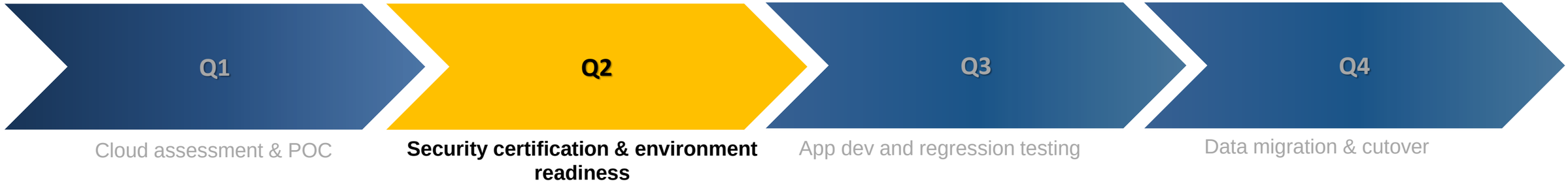




AWS EMR Migration Strategy

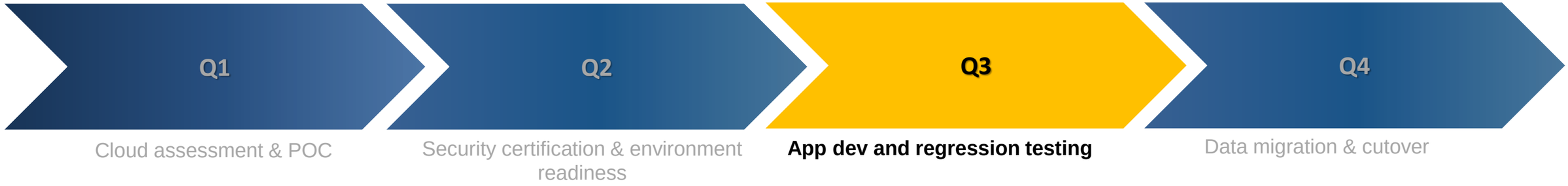
Category	EMR
Functional capabilities	Meets data processing and analytics requirements
Infrastructure and software costs	Medium
Ease of AD and Kerberos Integration	Achieved using custom solutions – not available out of the box
Amazon EC2 Image used	Amazon Linux – new minimum baseline security image required to meet Guardian security & compliance standards
DNS	Internal DNS – Amazon EMR is very sensitive to DNS and does not work with Guardian's DNS
Familiarity to the team	Medium
Ease of complete product installation	High
Ease of deployment automation	High
Ease of DR setup	High
Risks for unknown factors	Medium

Amazon EMR migration strategy



- Partner with IT security team
- Amazon Linux - require new minimum baseline security standard
- Obtain security exceptions – Kerberos, CIS benchmarks
- Third-party software changes required for integration
- Edge node setup - security hardening by shutting off SSH access on the cluster
- Custom DNS
- Data protection and controls – Amazon S3 encryption; SSL / HTTPS
- Multi-region DR requirements – Amazon S3; automation

Amazon EMR migration strategy

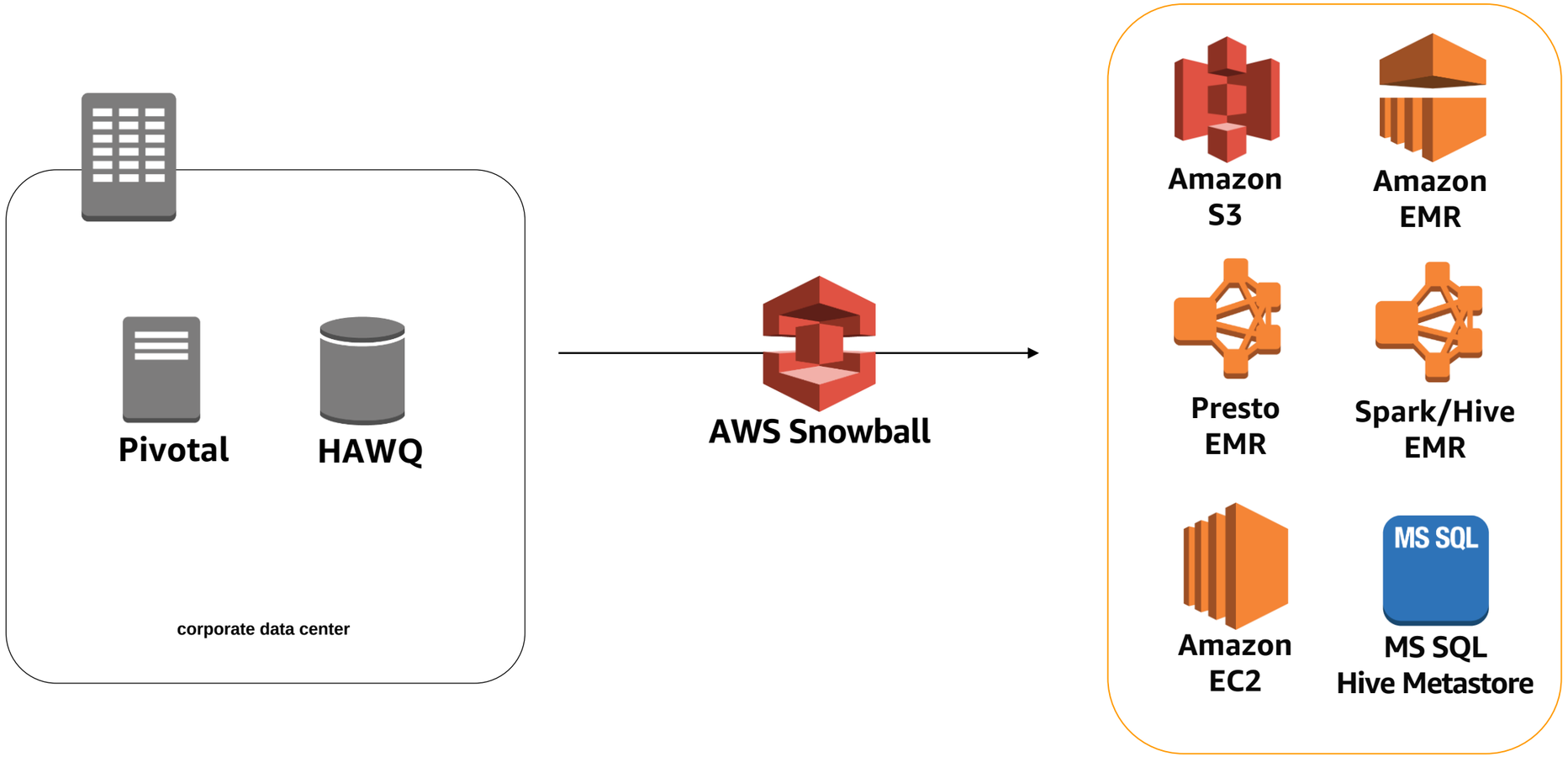


- Automation – Terraform, Puppet
- CI/CD integration – Bitbucket, Jenkins
- Refactor and test 300+ workloads
- Scope of code changes – Syncsort, Pig, Python, R, Shell Scripts
- Code migration plan
  - Accommodate in-flight AppDev projects (Dev & UAT)
  - Migrate to AWS dev and promote up
- Set code freeze for on-prem AppDev

Amazon EMR migration strategy

- Conduct parallel production testing between on-prem and AWS
- Determine up front the data set to snapshot for parallel production runs
- Historical data reserved for Snowball
- Utilize multiple Snowball Edge – migrate 350 TB
- Archive operations – migrate data to lower tier Amazon S3 (for example, S3-IA, Amazon Glacier)
- Shutdown on-prem workloads and repurpose hardware

Amazon EMR migration



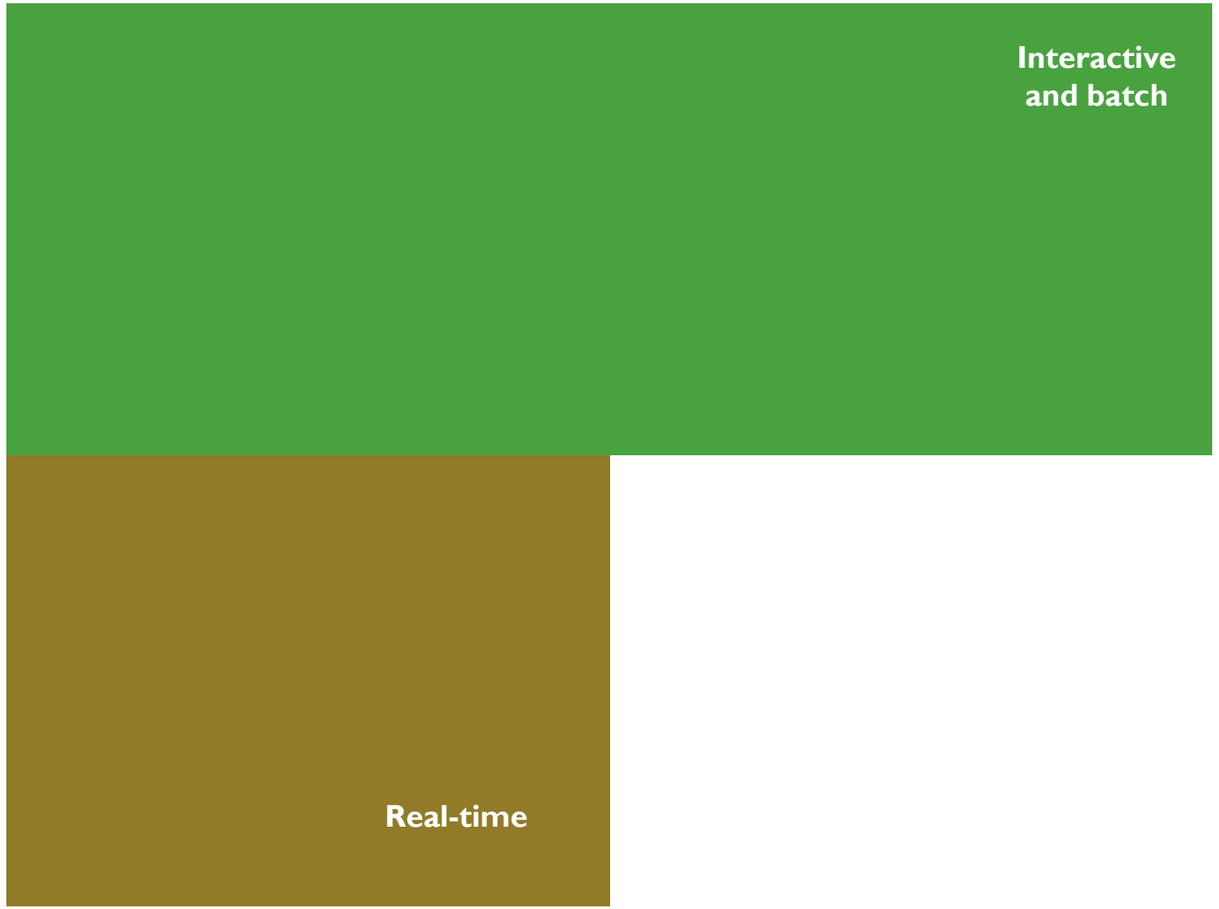
Overall architectural design pattern

Data lake architectural patterns

Process

Store

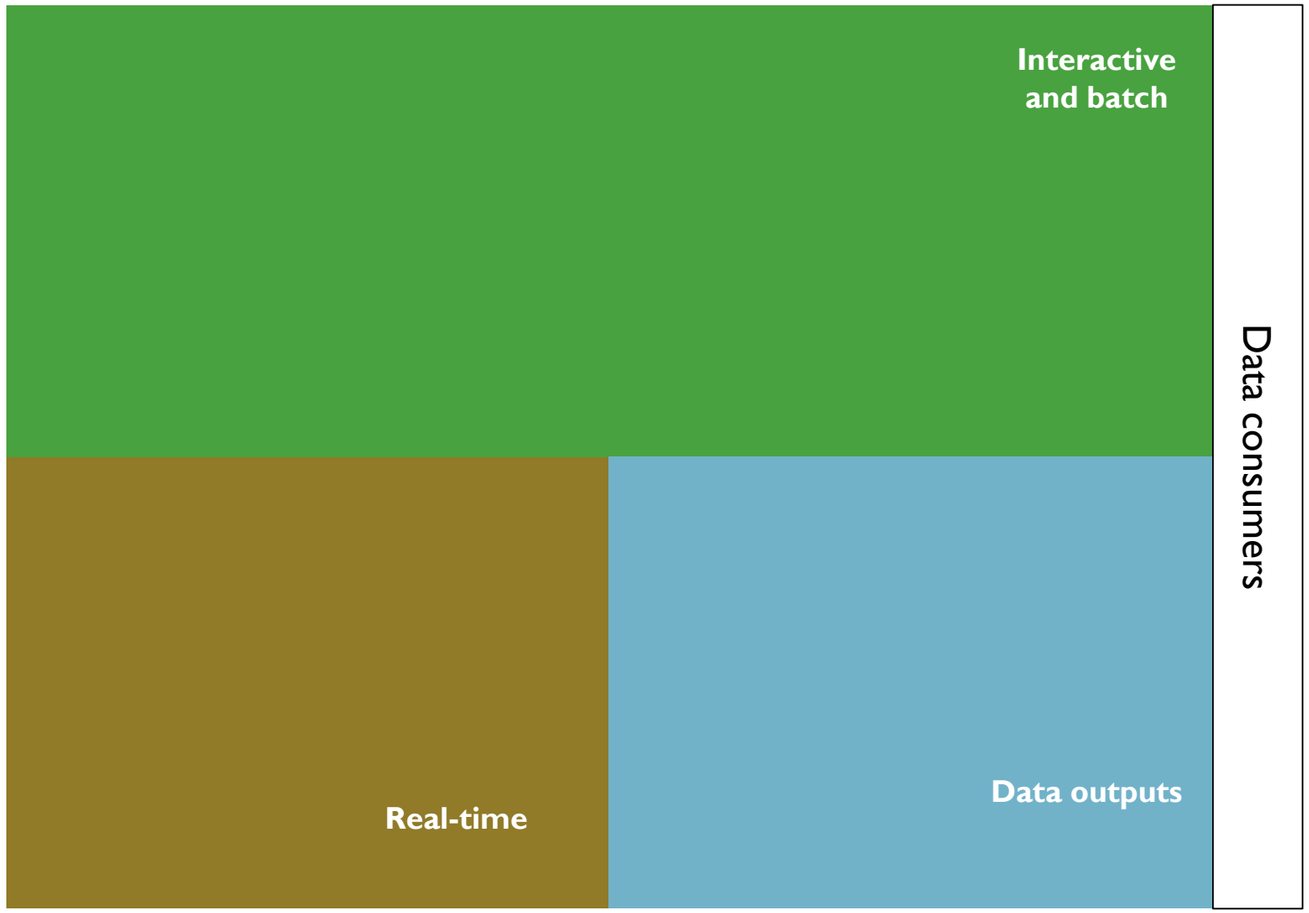
Data lake architectural patterns



Process

Store

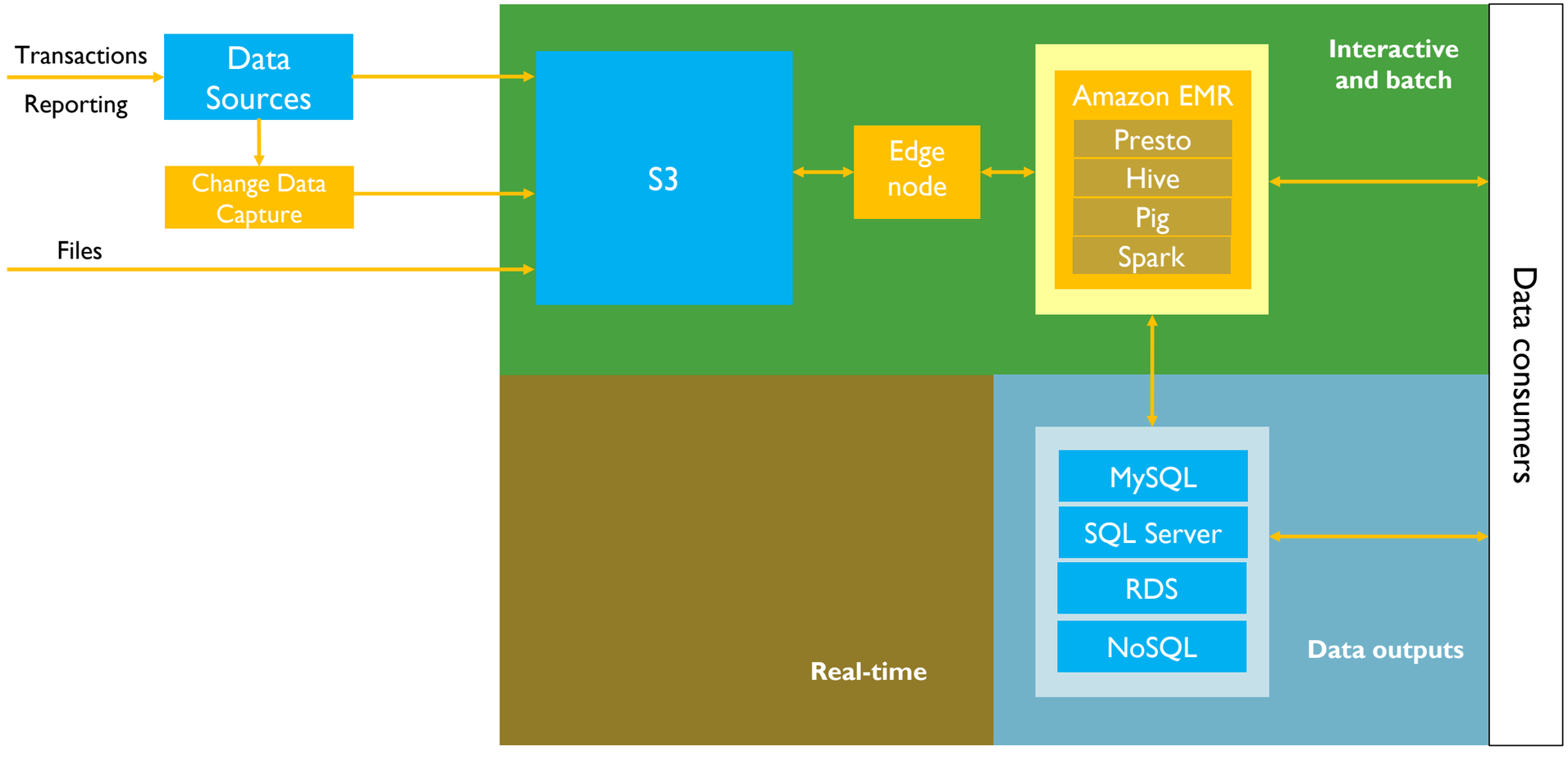
Data lake architectural patterns



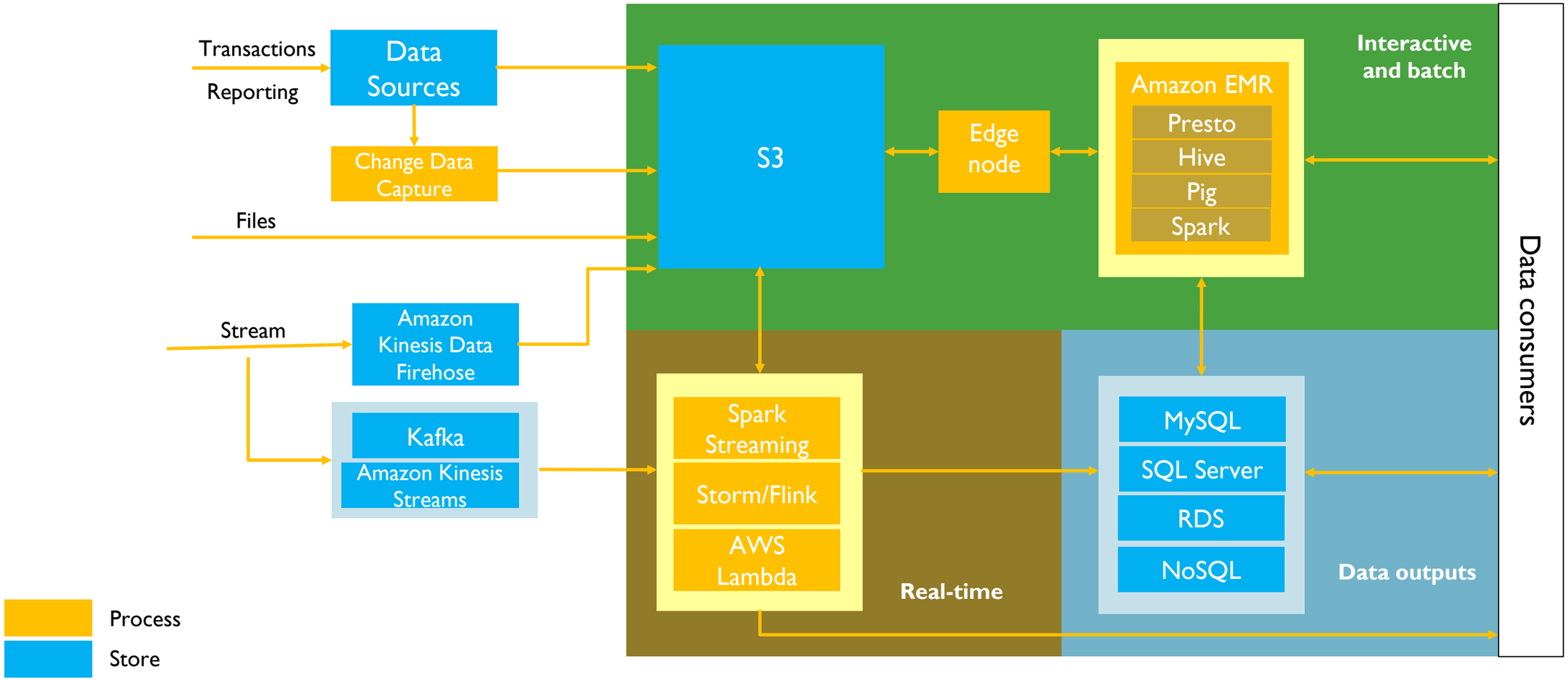
Process

Store

Data lake architectural patterns



Data lake architectural patterns



AWS EMR & Amazon S3

Key takeaways for adoption

- Cost-effective
- Decoupling storage from compute
- Security
- Automation
- Open source integrations
- Resilient and scalability

Guardian®

Thank you.

Park Avenue Securities LLC (PAS) is an indirect, wholly-owned subsidiary of The Guardian Life Insurance Company of America (Guardian). PAS is a registered broker-dealer offering investment products, as well as a registered investment advisor offering financial planning and investment advisory services. PAS is a member of FINRA and SIPC.

Individual disability income products underwritten and issued by Berkshire Life Insurance Company of America (BLICOA), Pittsfield, MA. BLICOA is a wholly owned stock subsidiary of and administrator for The Guardian Life Insurance Company of America (Guardian), New York, NY or provided by Guardian. Product provisions and availability may vary by state.
2018-66009 Exp. 09/2020

Please complete the session survey in the mobile app.