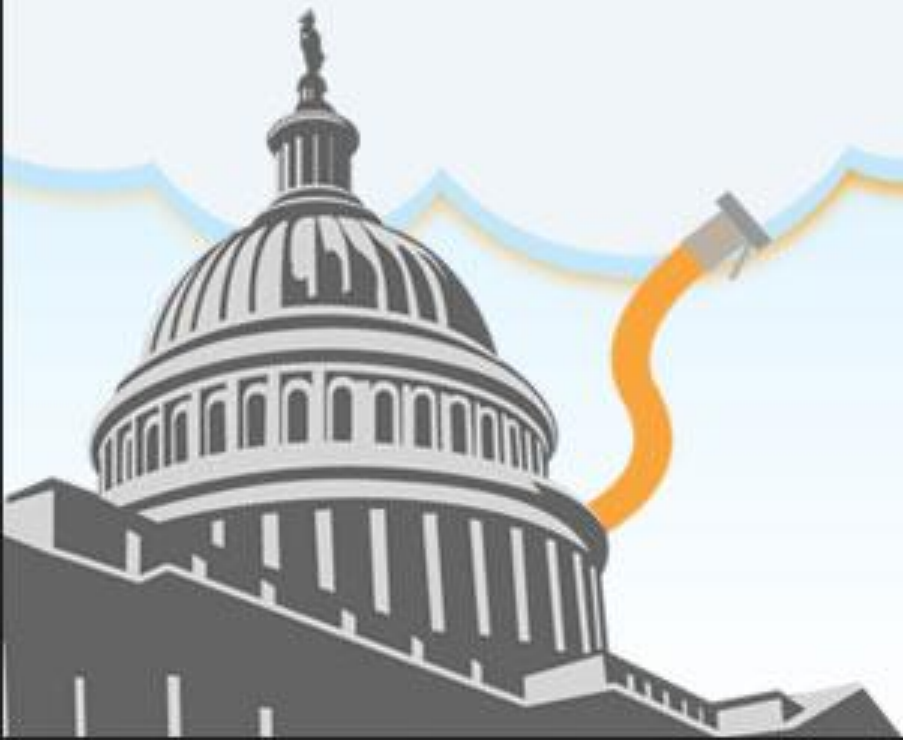
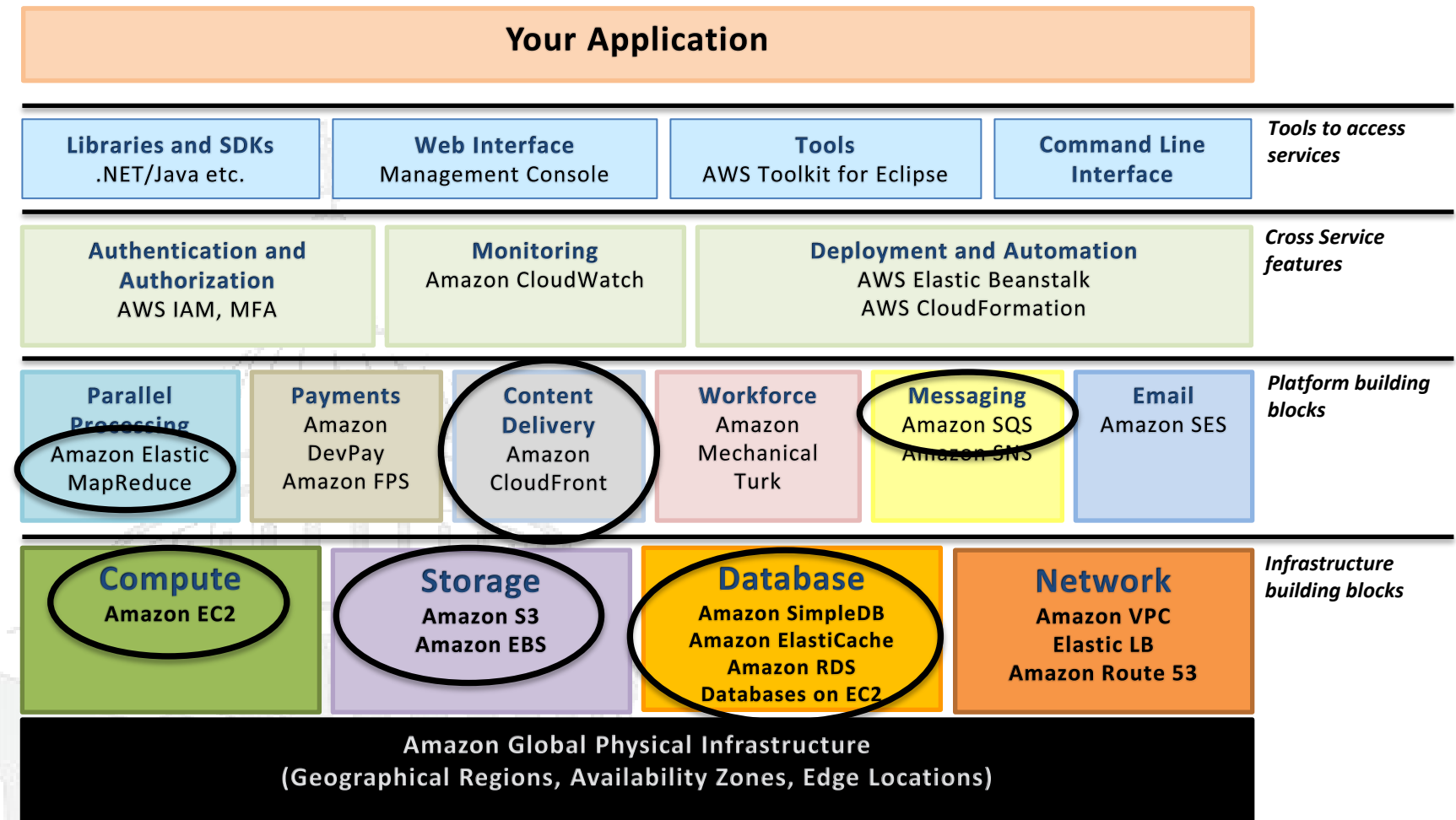


Storage Options in the AWS Cloud

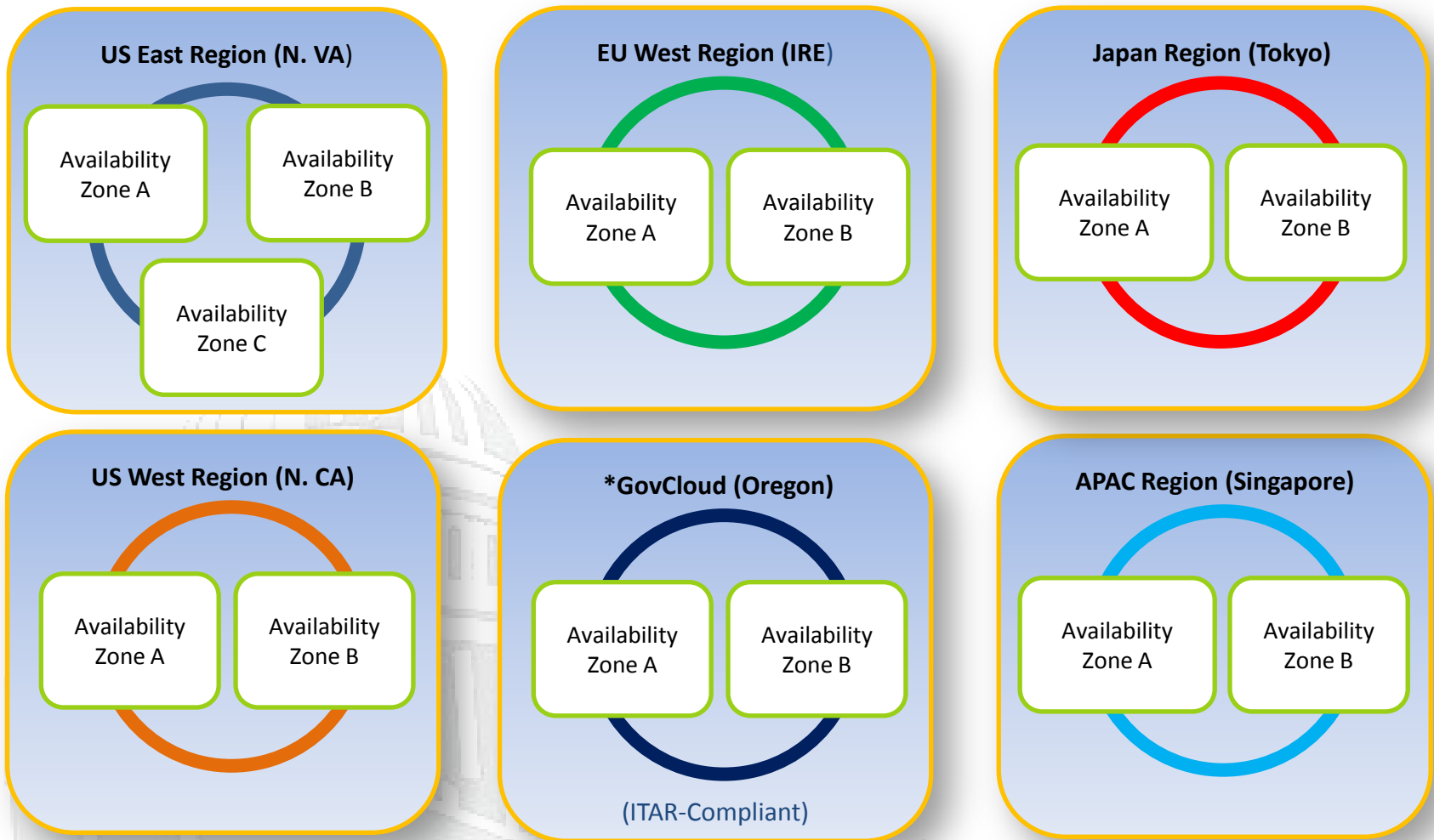
Joseph G. Baron, Principal Solutions Architect



Storage in the Amazon Cloud









Amazon EC2 Regions and Availability Zones






Note: Conceptual drawing only. The number of AZs may vary

Traditional Storage

Multiple Storage Options

-  Memory – caches, RAM disks
-  DAS – local block devices (disks)
-  SAN / NAS – network-attached block devices (LUNs) / file systems (NFS & CIFS file servers)
-  Structured Storage – databases on SAN, DAS or NAS
-  Message Queues – FIFOs, MOM
-  Backup / Archival Storage – off-line backup and DR

Differ in

-  Performance
-  Durability
-  Cost

Usually form a Storage Hierarchy

AWS Cloud Storage

❏ Multiple Storage Options

- ❏ Memory – Amazon ElastiCache
- ❏ DAS – EC2 Instance Store
- ❏ SAN / NAS – Elastic Block Storage (EBS), Simple Storage Service (S3)
- ❏ Structured Storage – EC2 Database AMIs, Amazon Relational Database Service (RDS), Amazon SimpleDB
- ❏ Message Queues – Amazon Simple Queue Service (SQS)
- ❏ Backup / Archival Storage – EBS Snapshots, S3

❏ Also differ in

- ❏ Performance
- ❏ Durability
- ❏ Cost

❏ Also usually form a Storage Hierarchy

Choosing AWS Cloud Storage Options

Examine each

-  Description

-  Ideal use cases

-  Characteristics (durability, scalability, etc.)

-  Anti-Patterns



In-Memory Storage

Amazon ElastiCache

- ❏ Fully-managed, distributed, in-memory cache
- ❏ Wire-protocol compliant with de facto standard Memcached
- ❏ “Pushbutton” management
 - ❏ Easily scale memory by add / deleting nodes
 - ❏ Automatically detects and replaces failed nodes
- ❏ Cost
 - ❏ Priced per Cache-Node-hour based on type
 - ❏ Cache nodes from \$.095 to \$2.24 per hour
- ❏ Ideal use cases
 - ❏ Reduce database load
 - ❏ Improve performance of
 - ❏ Database-centric applications – query cache
 - ❏ Dynamic web-apps – page / session cache
 - ❏ Content management systems – object cache for Drupal, Joomla, WordPress etc.



Direct-Attached Storage

Amazon Local Instance Storage

- ❏ Block-level *temporary* storage for EC2 instances
 - ❏ Also called “Ephemeral drive” or “Local drives”
- ❏ “In the box” storage
 - ❏ No network variability, but...
 - ❏ No persistence – *All data is automatically deleted when an instance stops, fails, or is terminated!*
- ❏ Cost: No additional charge
- ❏ Ideal use cases:
 - ❏ Scratch disks
 - ❏ Temporary files, Buffers, Caches
 - ❏ Easily-replicated data – web server’s DocumentRoot or equivalent

Instance Storage Characteristics

- ❏ Number and size of instance store devices varies by EC2 instance type:
 - ❏ Larger instances have larger / more volumes
 - ❏ c1.xlarge has four 450GB drives
 - ❏ Micro instances have none
 - ❏ Local storage may be *available* but not *exposed*
 - ❏ See addendum slide on S3-backed vs. EBS-backed AMIs
- ❏ Not optimized for random I/O
 - ❏ EBS generally better for random I/O
 - ❏ Instance stores generally better for sequential I/O
- ❏ Can be striped using RAID 0
 - ❏ Aggregate IOPS
 - ❏ Aggregate throughput (bandwidth)

Amazon Instance Store Anti-Patterns

Persistent storage

-  Consider EBS

Database / Structured storage

-  Generally need persistence beyond lifecycle of single EC2 instance
-  Consider EBS, SimpleDB or RDS

Shareable storage

-  Local instance stores cannot be moved / shared
-  Consider EBS

Backups

-  Need easy point-in-time backups, shareability
-  Consider EBS and EBS Snapshots



Network-Attached Storage

Amazon Elastic Block (EBS) Storage

- ❏ Block-level *persistent* storage for EC2 instances
 - ❏ Data lifetime independent of EC2 instance lifetime
 - ❏ Each EBS volume is like a hard drive on a physical server
- ❏ Off-instance, network-attached storage
 - ❏ 1GB to 1TB per volume
 - ❏ Can attach multiple EBS volumes to an EC2 instance
 - ❏ Can only attach any single EBS volume to one EC2 instance at a time
- ❏ Cost: \$.10 per GB-month (provisioned storage)
 - ❏ 1 x 500 GB volume same cost as 10 x 50 GB volumes
- ❏ Ideal use case -- primary EC2 storage for:
 - ❏ OS Boot device / root filesystem
 - ❏ File systems
 - ❏ Databases
 - ❏ Raw block devices

Amazon EBS Characteristics




- ❏ Durable off-instance storage
 - ❏ Mirrored (replicated) within a single Availability Zone
 - ❏ EBS Snapshots provide enhanced durability, other benefits
 - ❏ Expect .1% - .5% annual failure rate (using snapshots)
- ❏ Optimized for random I/O, expect:
 - ❏ About 100 IOPS for 8K random reads
 - ❏ Up to 80 MB/sec sequential access
 - ❏ Some level of variability – network-attached, shared resource
- ❏ Can be striped using RAID 0 or LVM
 - ❏ Can aggregate IOPS
 - ❏ Generally cannot aggregate throughput

Amazon EBS Anti-Patterns

Temporary storage

-  Consider EC2 Instance Storage

Very high-durability storage

-  Consider Amazon S3 or Snapshots
 -  S3 design point eleven 9's annual durability (per object)
 -  EBS design point 99.5 to 99.9% annual durability (per volume)

Storing static web content

-  Consider Amazon S3

Storing structured data or Key-Value pairs

-  Consider Amazon SimpleDB or
-  Amazon RDS

Short Detour: Two Types of AMIs / EC2 Instances

- ❏ Amazon Machine Image (AMI) used to launch EC2 instances
- ❏ Instance store root ("S3-backed")
 - ❏ Original AMI type, boots from ephemeral storage
 - ❏ Can Start and Terminate only
 - ❏ All data is ephemeral (unless separate EBS volume attached)
- ❏ EBS root ("Boot from EBS")
 - ❏ Newer AMI type, boots from an EBS volume
 - ❏ Boots quickly, can Start, Stop, Create Image, and Terminate
 - ❏ Ephemeral storage still available, but not exposed by default
 - ❏ Use instance Block Device Mapping (command line and API only)
- ❏ Additional EBS volumes can be attached to instances started from either AMI type

Amazon Simple Storage Service (S3)




- ❏ Scalable and durable data storage in the cloud
 - ❏ Read / write / delete objects (files) from 1 Byte up to 5TBs
 - ❏ Concurrent reads / writes to single S3 bucket or object
- ❏ Store nearly any kind of data
- ❏ Pay-as-you-go tiered pricing:
 - ❏ \$0.14/GB (first TB) to \$0.055/GB (over 5PB)
 - ❏ Plus data transfer and requests
- ❏ Ideal use cases
 - ❏ Static web content – often used with CloudFront CDN
 - ❏ Source and working storage for large-scale “Big Data” computation or analytics
 - ❏ Backup, archival, and DR storage that is always “live”

Amazon S3 Characteristics




- 📦 Two layer hierarchy: Buckets and Objects
- 📦 Every object has a unique URL
- 📦 Simple Get, Put, Delete API using HTTP
- 📦 Extreme durability
 - 📦 Automatic replication to multiple locations in a Region
 - 📦 Design point is eleven 9's durability and four 9's availability
 - 📦 Reduced Redundancy Storage offers lower durability at lower cost
- 📦 Extreme Scale
 - 📦 Unlimited number of objects per bucket
 - 📦 Web-scale concurrent read / writes
- 📦 Supports
 - 📦 Multiple access control mechanisms
 - 📦 Encryption – both in-transit and at-rest
 - 📦 Versioning

Amazon S3 Anti-Patterns




File systems

-  S3 is an object store, not a POSIX file system
 -  Can *emulate* a folder/file hierarchy
-  For a true filesystem, use EBS storage

Structured data with query

-  S3 does not support query
 -  Must know the bucket name and the key
-  Use in conjunction with SimpleDB or database

Rapidly-changing, fine-grained changes

-  S3 generally reads / writes whole files
 -  “Web-like” rather than “disk-like” latencies
-  Use EBS for fine-grained changes, lower latencies

Amazon CloudFront

- ❏ Easy-to-use Content Delivery Network (CDN)
- ❏ Often uses Amazon S3 as the origin store
 - ❏ Also can use non-S3 origins, such as web server on EC2
- ❏ Worldwide network of 20 edge locations
 - ❏ US, Europe, Asia, South America
- ❏ Pay-as-you-go tiered pricing on Data Transfer Out:
 - ❏ \$0.12/GB (first TB) to \$0.02/GB (over 5PB)
 - ❏ Plus origin fetch requests
- ❏ Ideal use case
 - ❏ Static web content that must be delivered to global user base at
 - ❏ Highest bandwidth / Lowest latency / Lowest cost



Structured Storage

Amazon SimpleDB

- ❏ Core database functionality for storage and querying of text data
 - ❏ Non-relational offering – no joins or complex transactions
 - ❏ Supports SQL-like queries with SELECT statement
- ❏ Low-Touch: no schema, no data modeling, no DBA needed
 - ❏ Eliminates the overhead of managing a relational database
- ❏ Cost: First 1GB-Month and ~2M queries free
- ❏ Ideal use cases:
 - ❏ Metadata storage -- often used in conjunction with S3
 - ❏ Structured, fine-grained data needing query
 - ❏ Data needing flexible schema

Amazon SimpleDB Characteristics

SimpleDB		RDBMS
Domains	↔	Table
Item	↔	Row
Attributes	↔	Column *
Values	↔	Values

- ❏ Fast, flexible, and highly durable
 - ❏ Near-LAN latencies for queries from EC2
 - ❏ Everything is indexed, can add attributes as needed
 - ❏ Data automatically replicated in multiple locations
- ❏ 10GB per domain
 - ❏ Can “shard” larger data sets over multiple domains
- ❏ Supports both:
 - ❏ Eventually-consistent reads (for speed & scalability)
 - ❏ Consistent reads and conditional Put/Delete (for transactional updates)

Amazon SimpleDB Anti-Patterns

- ❏ If your application:
 - ❏ Is tied to a specific traditional relational database
 - ❏ Needs Joins and Complex transactions
 - ❏ Needs BLObs (Binary Large Objects) support
 - ❏ Needs Typed or Numeric data
 - ❏ Needs very large data (>>10GB)
- ❏ Then consider Amazon RDS or a traditional relational database

Amazon Relational Database Service (RDS)

- ❏ Fully-functional MySQL or Oracle relational database provided as a managed, cloud-based service
- ❏ Automates:
 - ❏ Provisioning
 - ❏ Patching
 - ❏ Backups
 - ❏ Replication (MySQL only)
- ❏ Cost: \$.11 to \$2.60 per hour (based on instance size/speed), plus storage and data transfer costs
- ❏ Ideal use cases:
 - ❏ Any application that needs a full native capabilities of traditional relational database
 - ❏ With minimal administrative overhead

Amazon RDS Characteristics

- ❏ Fully-managed, tuned MySQL or Oracle 11g database
 - ❏ Compatible with all normal applications, tools, & drivers
- ❏ Simple to deploy
 - ❏ Make a few clicks or API calls, get a connection string
- ❏ Scalable
 - ❏ Scale vertically (increase / decrease compute, increase storage 5GB to 1TB)
 - ❏ Scale horizontally using read replicas * (MySQL only)
- ❏ Reliable
 - ❏ Automated backups of DB and logs, point-in-time restore
 - ❏ User-initiated DB Snapshots
 - ❏ Multi-AZ deployments – synchronous replication / automated failover * (MySQL only)

Amazon RDS Anti-Patterns

- ❏ Simple index-and-query focused data
 - ❏ If you don't need joins and complex transactions, consider using SimpleDB
- ❏ Heavy use of BLOBs, or very large BLOBs
 - ❏ Consider storing the BLOBs in S3, with pointers and metadata in RDS
- ❏ Scaling beyond RDS vertical / horizontal scaling limits
 - ❏ Consider using S3 and SimpleDB together
- ❏ Your application requires a specific RDBMS not supported by RDS, or
- ❏ You need complete administrative control
 - ❏ Consider using EC2 Relational Database AMIs

Amazon Relational Database AMIs

- ❏ EC2 instances and EBS storage provide a platform to run many relational databases
- ❏ Ready-to-use AMIs for
 - ❏ Oracle, SQLServer, DB2, Informix, PostgreSQL, Sybase, Vertica, etc.
- ❏ Costs and license terms vary by vendor
- ❏ Ideal use cases:
 - ❏ Need the full native capabilities of one particular relational database
 - ❏ With full administrative control

Amazon Relational Database AMIs

- ❏ Performance characteristics depend on
 - ❏ Database software
 - ❏ Number / size of EBS storage volumes
 - ❏ Configuration of DB
 - ❏ ...
- ❏ Benchmark your application

Relational Database AMIs – Anti-Patterns

- ❏ Index-and-query focused data
 - ❏ Consider SimpleDB
- ❏ Have BLObs
 - ❏ Consider moving BLObs to S3
- ❏ Need more Automation, Scaling etc.
 - ❏ Consider RDS, SimpleDB

Choosing an AWS Database Solution

	Amazon SimpleDB	Amazon Relational Database Service	Amazon EC2 Relational Database AMIs
Features	Schema-less data store	Native access to a relational database engine, with key features of a relational database, such as joins or complex transactions	Choose your own database server (IBM DB2, Microsoft SQL Server, MySQL, Oracle, and more)
Administration	Zero administrative overhead (automatic handling of geo-redundant replication, index creation, database tuning)	Gain a managed experience and offload common administrative tasks, such as provisioning, backup	Exert complete administrative control over your database
Flexibility	Easy to use web service APIs	Easy migration path (existing code, tools, applications are compatible)	Easy migration path (existing code, tools, applications are compatible)
Scalability	Automatic and elastic scaling of resources to meet request load	Quickly scale up resources with only an API call	Employ many of the scalability features of Amazon EC2
Backup	High availability (multiple copies of data for reliability and failover)	Use automatic backup service at an interval you specify	Store snapshots to Amazon S3



Message Queues

Amazon Simple Queue Service (SQS)

- ❏ Reliable, highly-scalable hosted message queuing service
- ❏ *SQS is storage? I thought this was an asynchronous communication protocol...*
 - ❏ SQS provides temporary storage (and reliable delivery) of short messages
 - ❏ Avoids use of other storage, such as temp files
- ❏ Cost: First 100K requests free, \$.01 per 10K
- ❏ Ideal use case:
 - ❏ "Software glue" to enable loose coupling
 - ❏ Store and move data between servers or application components
 - ❏ Often used with data stored in S3

Amazon SQS Characteristics

- ❏ SQS provides *Durable* but *Temporary* storage
 - ❏ Messages must be text-only, up to 64KB
 - ❏ Messages stay in queue for up to 14 days
- ❏ Highly Reliable
 - ❏ Messages are stored redundantly on multiple servers and data centers in a Region
- ❏ Highly Scalable
 - ❏ Unlimited number of clients reading / writing an
 - ❏ Unlimited number of messages
- ❏ Not designed to maximize single-thread performance
 - ❏ 5 - 50 messages per second per thread
 - ❏ Higher performance with multiple messages per call

Amazon SQS Anti-Patterns

- ❏ Binary data or
- ❏ Large data (>64KB)
 - ❏ Store the data in S3 or RDS
 - ❏ Store a pointer to the data in SQS
- ❏ Long-term storage – over 14 days
 - ❏ Consider S3 or other storage



Backup / Archival Storage

Amazon EBS Snapshots

- ❏ Snapshots provide point-in-time incremental backups for EBS volumes, stored in S3
 - ❏ Saves only the data changed since the last snapshot
 - ❏ Any single snapshot represents a complete backup
 - ❏ Create (restore) one or more new EBS volumes from a snapshot
- ❏ Snapshots also enhance durability of EBS
 - ❏ EBS volumes “live” (are mirrored) in a single Availability Zone (AZ)
 - ❏ Snapshots “live” in S3, are replicated within a Region
 - ❏ Durability of EBS volume is proportional to amount changed data – Snapshot frequently!
- ❏ Snapshots also allow you to easily:
 - ❏ Clone an EBS volume in different AZ
 - ❏ Resize an EBS volume
 - ❏ Share an EBS volume across accounts

Backup to Amazon Simple Storage Service (S3)

- ❏ S3 provides highly-durable and highly-secure backup, archival, and DR storage
- ❏ Highly-Durable
 - ❏ Data stored on multiple copies / locations in Region
 - ❏ S3 automatically detects and repairs any lost redundancy
 - ❏ Versioning and MFA Delete capability protect against human error
- ❏ Highly-Secure
 - ❏ Encryption in-flight: SSL (HTTPS)
 - ❏ Encryption at-rest: Server Side Encryption
 - ❏ Encryption at-rest: Client Side Encryption
 - ❏ Rich access control: ACLs, IAM policies, Bucket Policies, Query-string Authentication
 - ❏ Logging: all S3 request activities can be logged
- ❏ Ideal use case
 - ❏ Backup, archival, and DR storage that is offsite, but always “live”

AWS Import / Export Service

- ❏ Accelerates moving large data in or out of AWS using portable storage devices
 - ❏ Bypass the Internet
 - ❏ Target can be S3 bucket or EBS snapshots
- ❏ Cost
 - ❏ \$80 per device and \$2.49 per hour data loading time
 - ❏ Plus normal storage costs for target
- ❏ Ideal use cases
 - ❏ Offsite backup
 - ❏ Disaster recovery
 - ❏ Data migration or distribution

If You Only Remember Three Things....

- 📦 **EBS** – primary storage for filesystems and databases
- 📦 **Snapshots** – point-in-time backups for EBS volumes
- 📦 **S3** – highly-durable, unlimited file storage on the web

Additional Resources

- 📦 AWS Architecture Center
<http://aws.amazon.com/architecture>
- 📦 AWS Security and Compliance
<http://aws.amazon.com/security>
- 📦 AWS Cloud Computing Whitepapers
<http://aws.amazon.com/whitepapers>
- 📦 AWS Economics Center
<http://aws.amazon.com/economics>
<http://calculator.s3.amazonaws.com/calc5.html>
- 📦 Contact me:
joebaron@amazon.com

Thank You!!

