

The Double Booking, Peak Traffic, and Cloud Migration Problem

QuickStart Cloud Engineering Bootcamp

Project 1 – Case Study 1

Eddie Leach

Introduction:

In this case study, I solved Cross Country Trains' problem including handling peak hour traffic, double booking through google.com, and cost of service and maintenance of on-prem servers and databases. I offer two solutions, compared them, gave a cost analysis, and a final recommendation.

In this document you will find the following:

- Introduction
- Deliverables
- Solution 1 – Full Cloud Migration
- Solution 2 – Hybrid Cloud (Burst)
- Index A: Cost Comparison
- Index B: Compare and Contrast Solutions
- Index C: Final Recommendations

I was not able to offer a single solution due to needing more information about Cross Country Trains regulations, investments, and end goals. Alternatively, I gave two recommendations and explained the benefits of each in specific situations.

Deliverables:

Below I reference different parts of this document to answer the questions in the "Deliverables:" section of the original project document.

1. Solutions 1 and 2 both address the issue of availability, peak hours, and double booking. They have different advantages and disadvantages that can be found in the comparison in Appendix B. Final Recommendation can be found in Appendix C.
The main reasons these solutions are effective are scalability, no dropped or double-booked requests, latency, disaster recovery, cost, redundancy, backups, maintenance, and on-prem costs. (Appendix A, B, and C)
2. Solution 1 is the suggested full cloud migration option. This solution should be implemented before the next scheduled 4-year period of maintenance and refresh or adding any extra servers. The reasoning is that the biggest cost benefit is from saving money from adding new servers or performing maintenance on current on-prem hardware. The migration cost is easily offset by the cost of adding and maintaining new servers. Reference Solution1 and Appendix A.
3. Migration to Cloud will provide availability and redundancy that is all built into the services. Reference Solution 2, and Appendix B, and C.
4. Reference the cost analysis in Appendix C. The tool used for my cost analysis is AWS Pricing Calculator along with average industry costs.

Peak Hours and Double-Booking Solution 1

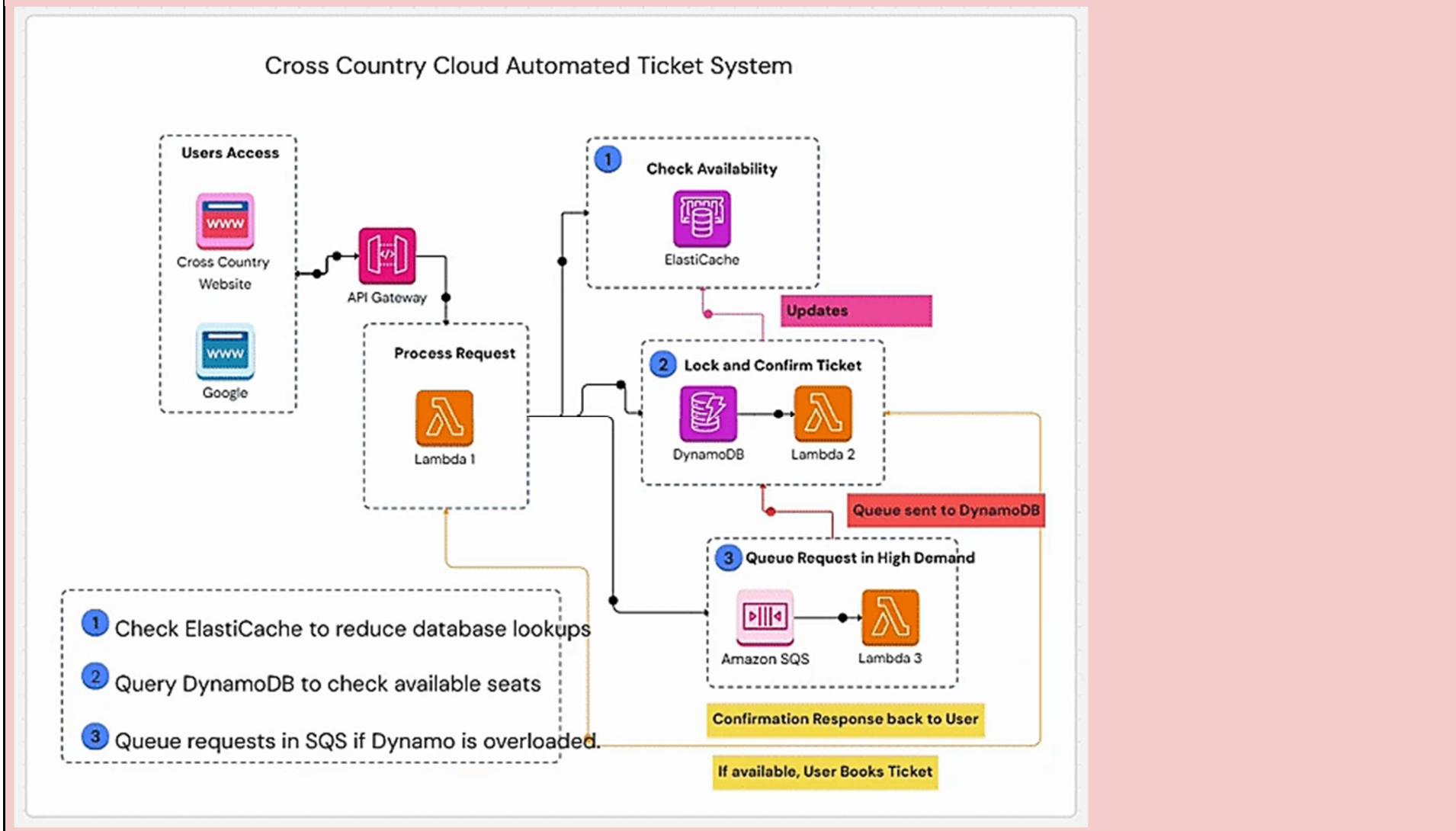
(Full Cloud Migration)

Summary of Services

- **API Gateway:** Entry point for request.
- **Lambda #1:** Booking Processor.
- **DynamoDB:** Manage ticket availability and prevent double booking.
- **ElastiCache:** Caches request to alleviate requests and lower database lookups.
- **Amazon SQS:** Queues request when DynamoDB to ensure no bookings are dropped.
- **Lambda #2 and #3:** Paired with services to process requests and send and receive information.

Service	Purpose	Functions	When
API Gateway	Handles High-Traffic Requests	<ul style="list-style-type: none"> -Serves as the entry point for ticket booking requests. -Can handle hundreds of thousands of concurrent API calls. -Can handle 42,000+ Transactions per second. -Routes requests to AWS Lambda for processing. 	Always
AWS Lambda Function #1	Cloud Autoscaling Serverless Booking Processor	<ul style="list-style-type: none"> -Executes when a booking request is received. -Process request from user -Check seat availability with ElastiCache and DynamoDB -Send requests to Amazon SQS if Dynamo overloaded -Confirm ticket with user if seat available 	Always
Amazon ElastiCache	Speeds Up Ticket Availability Checks	<ul style="list-style-type: none"> -Stores cached seat availability to reduce database lookups. -Ensures real-time response times by keeping popular ticket data in memory. -DynamoDB Streams keep the cache updated with the latest bookings. 	Always
Amazon DynamoDB	Manages Ticket Inventory & Prevents Double Booking	<ul style="list-style-type: none"> - Optimistic locking (Conditional Writes) ensures that a seat cannot be double booked. - Lock seat and confirm with user - Supports millions of transactions per second. - DynamoDB Streams syncs data with Amazon ElastiCache - Sends Confirmation to customer - Paired with Lambda Function #2 to process request, update cache, confirm with user, optimize performance, and auto scale to meet demand 	When ElastiCache not updated
Amazon SQS	Queue High-Traffic Requests During Peaks	<ul style="list-style-type: none"> - Ensures no request is dropped when traffic spikes. - If DynamoDB is overwhelmed, Lambda places booking requests into an SQS queue. - Send Requests to DynamoDB when available - Paired with Lambda Function #3 to process queued requests, optimize performance, and auto scale to meet demand 	High Demand if DynamoDB is overwhelmed

Diagram



Peak Hours and Double-Booking Solution 2

(Cloud Bursting “Hybrid”)

Summary of Services

Normal Load: Route 53 prioritizes on-prem resources first. On prem Databases are used to process requests

Peak Load: When capacity is reached, Route 53 directs traffic to Cloud Resources.

**Minimal cloud resources can be used concurrently with on-prem to minimize cloud costs and utilize on-prem resources at all times.

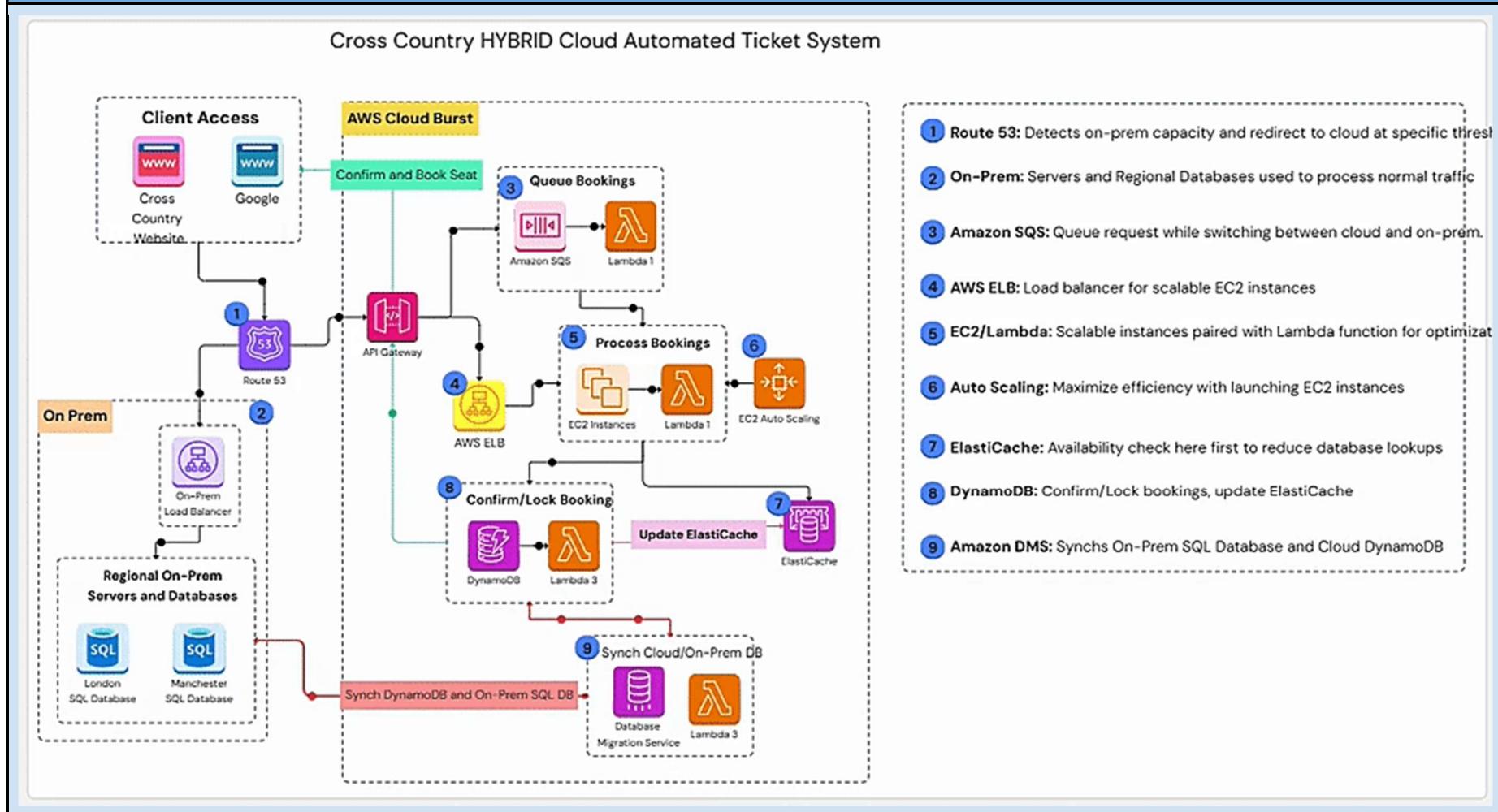
Service	Purpose	Functions	When it is used
AWS Route 53	Scalable Domain Name Service	<ul style="list-style-type: none"> - Routes service between On-Prem and AWS cloud resources - Detects On-Prem server at capacity and redirects to AWS ELB - Gradually shift back to on-prem when resources available 	Always
On-Prem Load Balancer	Balance traffic across on-prem servers	<ul style="list-style-type: none"> - Receive Traffic - Monitor on-prem resource usage and request rate - Marks servers as overloaded - Balance requests between servers - Detects Overload (%CPU or TPS) - Update Route 53 to redirect traffic to AWS 	Normal Demand
On-Prem Servers and Data Center	SQL Databases and Servers (London and Manchester)	<ul style="list-style-type: none"> - Process Ticket Sales at normal traffic levels - Track Available Seats - Store Data - Return Booking Confirmation to user 	Normal Demand

Cloud Burst during high Demand

API Gateway	Handles High-Traffic Requests	<ul style="list-style-type: none"> - Serves as the entry point for ticket booking requests in the cloud. - Can handle hundreds of thousands of concurrent API calls. - Can handle 42,000+ Transactions per second. - Routes requests to AWS ELB. 	High Demand
Amazon SQS + Lambda	Queues Requests	<ul style="list-style-type: none"> - Queues requests when traffic switches between on-prem and cloud - Lambda function picks up and processes queued requests without blocking users. 	While switching between on-prem and cloud
AWS Elastic Load Balancer	Balance traffic across AWS	<ul style="list-style-type: none"> - Distribute requests evenly across healthy EC2 instances - Prevent overload on any single server 	High Demand

	servers	<ul style="list-style-type: none"> - Primary trafficker if on-prem fails - Works together with AWS Auto Scaling to maximize performance and cost 	
AWS Auto Scaling	Adjust number of EC2 instances	<ul style="list-style-type: none"> - Launches with AWS ELB during high demand - Works together with AWS ELB to maximize performance and cost - Ensures there are enough EC2 instances to handle traffic - Dynamically increases and decreases EC2 instances based on demand 	High Demand
EC2 Instances	Scalable On-Demand Backup Servers	<ul style="list-style-type: none"> - Primary cloud compute resource to handle application logic and requests - Handle overflow traffic when on-prem resources are overloaded - Check ElastiCache for seat availability - Query DynamoDB (if ElastiCache not updated) - Increase and decrease (scale) based on Demand 	High Demand
EC2 Lambda Functions	Offload compute Functions	<ul style="list-style-type: none"> - Improve efficiency - Process queued request - Enhance EC2 functionality <ul style="list-style-type: none"> - Handle background tasks - Optimize database lookups - Real-time updates 	High Demand
Amazon DynamoDB + Lambda	Manages Ticket Inventory & Prevents Double Booking	<ul style="list-style-type: none"> - Optimistic locking (Conditional Writes) ensures that a seat cannot be double booked. - Lock seat and confirm with user - Supports millions of transactions per second. - DynamoDB Streams syncs data with Amazon ElastiCache for fast availability checks. - Sends Confirmation to customer - Process queued requests from Amazon SQS - Amazon DMS synchs On-Prem DB with DynamoDB 	High Demand
Amazon ElastiCache + Lambda	Speeds Up Ticket Availability Checks	<ul style="list-style-type: none"> - Stores cached seat availability to reduce database lookups. - Ensures real-time response times by keeping popular ticket data in memory. - DynamoDB Streams keep the cache updated with the latest bookings. - If cache is not updated, DynamoDB is accessed 	High Demand
AWS Database Migration Service (DMS)	Synchronize data between On-Prem and Cloud	<ul style="list-style-type: none"> - Continuous synchronization so On-Prem and Cloud data match - Crucial for avoiding double booking while switching to and from on-prem and cloud 	Normal and High Demand

Diagram:



Appendix A: Cost Comparison

I attempted to use the AWS Pricing Calculator but there are too many unknowns to do a full cost analysis. Data storage size, average item size, write and rewrite settings are some examples. Alternatively, I have created a general cost analysis and made an estimate based on average industry cost, this will be a rough estimate.

Category	On-Prem Cost	Hybrid Cost	Full Cloud Cost
Infrastructure & Hardware Costs	High (Buying & upgrading servers)	Medium (On-prem + cloud integration)	Low (AWS owns infrastructure)
Operational & Maintenance Costs	High (IT staff, power, cooling)	Medium (Some IT staff still needed)	Low (AWS manages infrastructure)
Operational & Maintenance Costs	High (IT staff, power, cooling)	Medium (Some IT staff still needed)	Low (AWS manages infrastructure)
Scalability & Flexibility Costs	High (Requires new hardware purchases)	Medium (Cloud scales, on-prem does not)	Low (Auto Scaling & pay-as-you-go)
Software Licensing & Database Costs	High (SQL Server, VMware, etc.)	Medium (Some software moves to cloud)	Low (AWS RDS, DynamoDB reduce licensing needs)
Network & Connectivity Costs	High (Data center bandwidth, VPN)	Medium (AWS Direct Connect/VPN fees)	Low (AWS handles global networking)
Storage Costs	High (SAN/NAS storage purchase)	Medium (AWS S3 + on-prem storage)	Low (AWS S3, EBS, Glacier)
Security & Compliance Costs	High (Firewalls, compliance audits)	Medium (Some AWS security tools used)	Low (AWS handles compliance & security)
Disaster Recovery & High Availability Costs	High (Multiple data centers required)	Medium (AWS provides failover)	Low (AWS offers global failover options)
IT Staffing Costs	High (24/7 on-site engineers)	Medium (Some engineers needed)	Low (AWS handles maintenance)
Electricity & Facility Costs	High (Power, cooling, real estate)	Medium (Smaller on-prem footprint)	Low (AWS data centers are optimized)
Total Cost of Ownership (TCO)	High	Medium	Low

Category Break Down				
1. Infrastructure & Hardware Costs	2. Operational & Maintenance Costs	3. Scalability & Flexibility Costs	4. Software Licensing & Database Costs	5. Network & Connectivity Costs
<p>On-Prem: Organizations must buy and replace physical servers, storage, networking gear every 3-5 years.</p> <p>Hybrid Cloud: Reduces some infrastructure costs, but on-prem hardware is still required.</p> <p>Full Cloud: No upfront hardware purchases—AWS manages infrastructure.</p> <p>Cloud is more cost-effective for businesses that need to scale quickly.</p>	<p>On-Prem: IT teams must manage, update, and troubleshoot physical hardware 24/7.</p> <p>Hybrid Cloud: Some maintenance is offloaded to AWS, but on-prem systems still require management.</p> <p>Full Cloud: AWS handles maintenance, patching, and monitoring at no extra cost.</p> <p>Cloud eliminates hardware maintenance and IT overhead.</p>	<p>On-Prem: Scaling requires buying more servers and waiting weeks/months for delivery.</p> <p>Hybrid Cloud: AWS provides on-demand Auto Scaling, but on-prem systems remain fixed.</p> <p>Full Cloud: Auto Scaling instantly adds or removes resources, reducing cost waste.</p> <p>AWS Auto Scaling ensures businesses only pay for what they use.</p>	<p>On-Prem: SQL Server, Windows Server, VMware, and other licenses must be purchased.</p> <p>Hybrid Cloud: Some workloads can migrate to AWS RDS (managed SQL/PostgreSQL), reducing licensing needs.</p> <p>Full Cloud: AWS provides DynamoDB, Aurora, RDS with built-in licensing, reducing costs.</p> <p>Cloud reduces software licensing costs by offering managed services.</p>	<p>On-Prem: High networking costs (bandwidth, VPNs, data center connectivity).</p> <p>Hybrid Cloud: AWS Direct Connect or VPNs are required to link on-prem to AWS.</p> <p>Full Cloud: Cloud-native networking (AWS VPC, Transit Gateway) is fully managed.</p> <p>Cloud networking is cheaper for global deployments.</p>
<p>6. Storage Costs</p> <p>On-Prem: Expensive SAN/NAS storage upgrades every few years.</p> <p>Hybrid Cloud: Some data remains on-prem, but AWS S3 & EBS reduce costs.</p> <p>Full Cloud: AWS S3, EBS, and Glacier provide cost-effective, scalable storage.</p> <p>AWS storage eliminates upfront hardware costs and scales automatically.</p>	<p>7. Security & Compliance Costs</p> <p>On-Prem: Businesses must pay for firewalls, intrusion detection, compliance audits.</p> <p>Hybrid Cloud: AWS security tools (IAM, AWS Shield, GuardDuty) reduce risks.</p> <p>Full Cloud: AWS handles security, compliance certifications, and audits.</p> <p>AWS reduces compliance and security costs significantly.</p>	<p>8. Disaster Recovery & High Availability</p> <p>On-Prem: Requires secondary data centers for failover, increasing costs.</p> <p>Hybrid Cloud: AWS can provide DR solutions (multi-region backups, failover).</p> <p>Full Cloud: AWS offers automated backups, global failover, and disaster recovery (AWS Backup, Route 53 Failover).</p> <p>Cloud reduces disaster recovery costs by providing built-in failover solutions.</p>	<p>9. IT Staffing Costs</p> <p>On-Prem: Requires 24/7 engineers, security, and networking teams.</p> <p>Hybrid Cloud: AWS reduces staffing needs, but some on-prem IT staff is still required.</p> <p>Full Cloud: AWS handles most infrastructure maintenance, reducing IT staff costs.</p> <p>Cloud enables businesses to operate with fewer in-house IT staff.</p>	<p>10. Electricity & Facility Costs</p> <p>On-Prem: High power and cooling costs for data centers.</p> <p>Hybrid Cloud: On-prem footprint is reduced, lowering electricity bills.</p> <p>Full Cloud: No power costs—AWS optimizes energy efficiency.</p> <p>Cloud eliminates data center electricity costs.</p>

Cost Comparison Across			
Cost Category	On-Prem	Hybrid Cloud	Full Cloud
Hardware & Refresh (Every 5 Years)	\$7,500,000	\$2,500,000	\$0
Software Licensing	\$150,000	\$75,000	\$0
IT Staffing (24/7 Support, Engineers)	\$350,000	\$200,000	\$100,000
Power, Cooling, and Real Estate	\$1,962,360	\$981,180	\$0
AWS Compute (EC2, Lambda, Auto Scaling)	\$0	\$24,000	\$300,000
AWS Storage (S3, EBS, Glacier, DynamoDB)	\$0	\$75,000	\$150,000
AWS Networking (Route 53, Direct Connect, Transit Gateway)	\$0	\$50,000	\$100,000
Security & Compliance	\$75,000	\$30,000	\$20,000
Disaster Recovery & Backup	\$100,000	\$50,000	\$25,000
API Gateway Costs (Monthly)	\$60,000	\$210	\$420
*Total Annual Cost	*\$2,697,360	*\$1,485,230	\$995,420
*Total 5-Year Cost (Including Hardware Refresh)	*\$20,986,800	*\$9,926,150	\$4,977,600
*Note: Annual costs do not include hardware and refresh but 5 year costs do include maintenance and refresh.			
AWS Cost Estimate			
I estimated a few services to compare with the above calculations.			
Google Drive Link			

Appendix B: Compare and Contrast Solutions

Pros and Cons

Hybrid Cloud (On-Prem + AWS Cloud)

Pros of Hybrid Cloud

- ✓ **Leverages Existing Investments** – Uses already owned data centers (London & Manchester), reducing cloud costs.
- ✓ **Regulatory Compliance** – Some industries require local data storage, which can be handled on-prem.
- ✓ **Latency Optimization** – Keeps critical workloads close to users in specific regions (e.g., UK customers get local response times).
- ✓ **Gradual Cloud Transition** – Allows a step-by-step migration to AWS without full commitment.
- ✓ **Failover & Redundancy** – If on-prem goes down, AWS provides high availability.

Cons of Hybrid Cloud

- ✗ **Complexity & Management Overhead** – Requires managing both on-prem and cloud resources, increasing operational workload.
- ✗ **Higher Infrastructure Costs** – Need to maintain on-prem hardware, staffing, and maintenance costs alongside cloud expenses.
- ✗ **Latency Between On-Prem & Cloud** – Data synchronization between on-prem and AWS introduces potential latency.
- ✗ **Security & Compliance Risks** – Data consistency and security policies need to be enforced across both environments.

Full Cloud Solution (AWS-Only)

Pros of Full Cloud

- ✓ **Scalability** – AWS Auto Scaling handles traffic spikes instantly with no physical hardware limits.
- ✓ **Cost Efficiency** – No upfront investment in hardware; pay only for what is used.
- ✓ **Simplified Management** – AWS manages hardware, networking, redundancy, and disaster recovery automatically.
- ✓ **High Availability & Disaster Recovery** – Multi-AZ and Multi-Region deployment ensure near 100% uptime.
- ✓ **Better Performance** – Global AWS infrastructure ensures low latency worldwide without physical limitations.

Cons of Full Cloud

- ✗ **Migration Costs & Effort** – A full shift to the cloud requires refactoring legacy applications and may involve downtime.
- ✗ **Vendor Lock-in** – Fully relying on AWS means potential cost increases over time with limited exit strategies.
- ✗ **Data Transfer Costs** – Moving large amounts of data to/from AWS can be expensive (AWS egress charges).
- ✗ **Latency for On-Prem Dependent Systems** – If some workloads still rely on on-prem services, network delays might occur.

Cloud Servers

Hybrid: EC2 Instances

Hybrid Cloud Uses Auto Scaling EC2 Because It Mimics On-Prem Servers

- Hybrid Cloud is designed to extend On-Prem infrastructure, so AWS needs to work similarly to on-prem servers.
- EC2 instances allow the same software and applications that run on-prem to run in AWS without modification.
- Many legacy applications (monolithic apps) are not serverless-compatible, so EC2 provides a direct lift-and-shift option. Auto Scaling ensures AWS can handle peak load dynamically while still working with on-prem components.

Full Cloud: Serverless

Full Cloud Uses AWS Lambda Because It's Fully Serverless

- When fully in AWS, there's no need for traditional servers (EC2), so everything can be event-driven and serverless.
- AWS Lambda scales automatically and handles high volumes (42,000+ transactions per second) without infrastructure management.
- No need to maintain or configure servers, reducing costs and complexity.
- Ideal for microservices-based applications where each function runs independently.

Factor	Hybrid (Auto Scaling EC2)	Full Cloud (AWS Lambda)
Compute Model	Uses EC2 instances (traditional servers)	Uses AWS Lambda (serverless functions)
Scaling	Auto Scaling launches more EC2 instances as needed	Lambda scales automatically, no servers needed
Cost Efficiency	Pay for running EC2 instances, even when idle	Pay only for actual function execution
Compatibility	Best for applications that require traditional infrastructure	Best for fully cloud-native applications
Management Overhead	Requires managing EC2 instances, OS updates, security patches	AWS fully manages Lambda, no OS maintenance
Startup Time	Slower (EC2 needs to boot up new instances)	Faster (Lambda executes instantly)
Use Case	Extends on-prem systems to AWS without major changes	Built for cloud-first, event-driven architectures

Appendix C: Final Recommendations

Hybrid Cloud is best if:

- You must keep some workloads on-prem (e.g., compliance, local storage).
- You already invested heavily in on-prem data centers and want to extend capacity with AWS.
- You need a gradual transition to the cloud.

Full Cloud (AWS-Only) is best if:

- You want full scalability, automation, and reduced maintenance costs.
- You are ready to migrate everything to AWS without on-prem dependencies.
- Your goal is high availability, performance, and cost savings long-term.