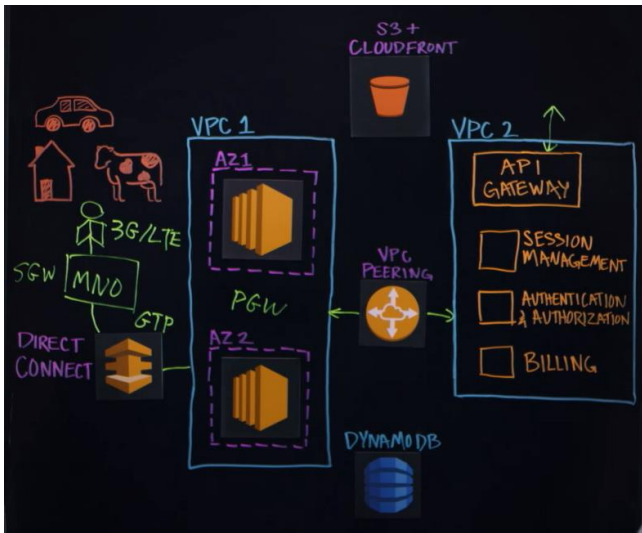
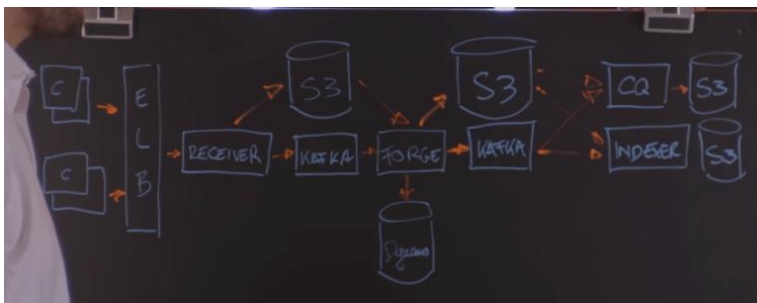


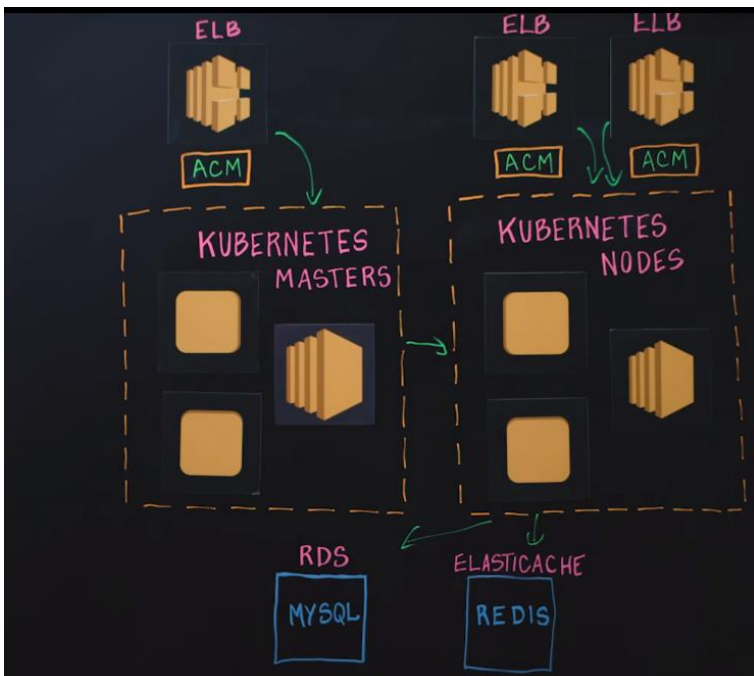
This Is My Architecture



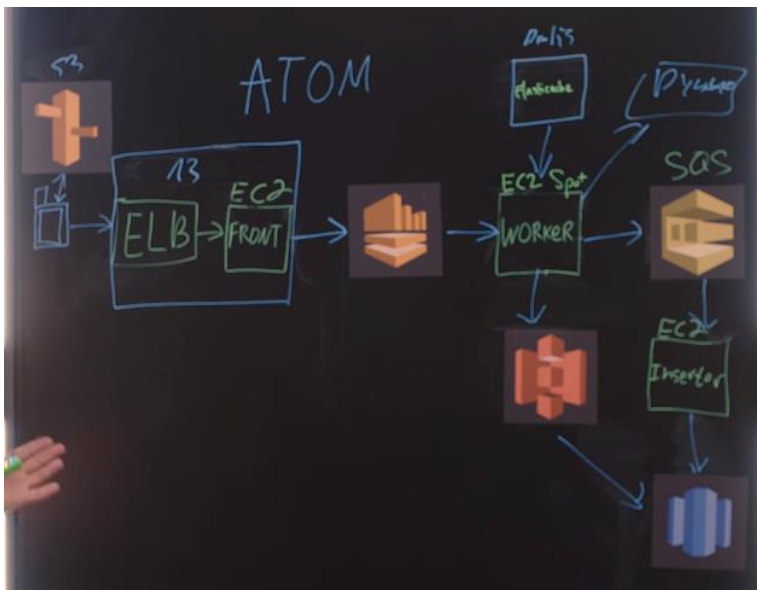
SORACOM – IOT Scale Telecom Infrastructure Built on Top of AWS



Sumo Logic – Ingesting Massive Amounts of Logs

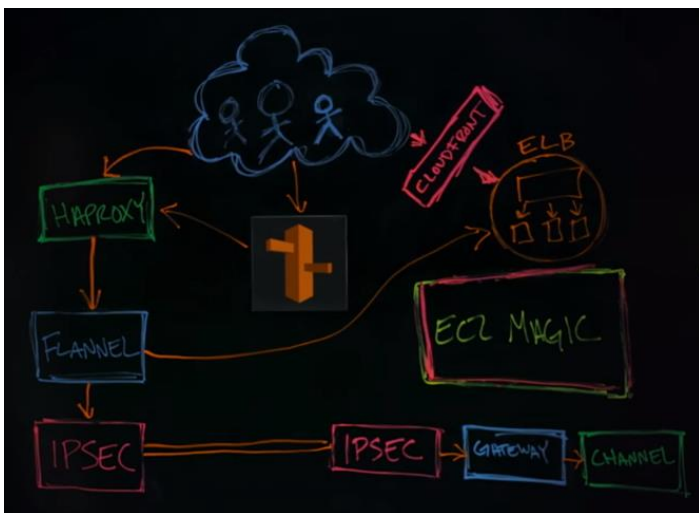


Spire Labs – Fault-Tolerance with Kubernetes on AWS. ACM is the Amazon Certificate Manager.



IronSource – Low Latency Stream Processing 100s of Billions of Events in 13 AWS Regions

The **EC2** workers are running the **Kinesis Client Library** in Python to get data/work off the **Kinesis stream**. We are using **Redis Elasticache** for doing de-duplication of the data by storing completed work Jod IDs in there and not allowing a worker to pick an already completed work. The workers (using EC2 spot instances) put completed work into **S3** and register completed **events** into **SQS**, the Inserter pulls events of **SQS** and knows which S3 files are ready to be inserted into **RedShift** tables for each customer. We have a Kinesis table in **DynamoDB** that every worker process checkpoints their state into in case they died and gets replaced.



Slack – Real-Time Communication with HAProxy and Route53 on AWS



Bulletproof – Slaying Monoliths One API Service at a Time

Bulletproof is an AWS premier consulting partner working with payment outdoor terminals. We have broken down a single huge Java application into smaller NodeJS apps with RESTful endpoints behind Nginx LB. **NodeJS apps** sit behind an **Nginx** layer receives telemetry data and sends them to the **SQS** queue, **CloudWatch Alarms** go off depending on things in the queue to trigger relevant **Lambdas** for getting processing done. Lambda is scheduled by **CloudWatch** for triggering actions that need to be performed.



ZocDoc – From Monolith to Less Than 100 milliseconds with Amazon ECS and Kinesis

ZocDoc is a patient portal for booking doctor appointments, reading reviews, information about profiles, doctor appointments and insurance details in a very fast online patient experience. The monolith publishes change feeds into Kinesis that ends up feeding it into an **ECS container** running Scala to clean up and orders the data and dump into a 2nd Kinesis stream that feeds a **DynamoDB** table that is a source of truth similar to the data in the monolith. The 2nd **Kinesis stream** also feeds into some Lambda functions that do relevant business logic and calculate projections and put the projections data into **Elasticache** for fast sub 100 millisecond search responses.



4000 Apps Later – **Expedia's** Automated CI/CD Platform with Github, Jenkins and Amazon ECS

This CI/CD platform makes it easier to let developers focus on their apps and be able to deploy on AWS. Using an application called Primer that runs on ECS, Devs can go to a page and enter details on the what the app name and other things needed. Then they click on a button to create the environment on AWS. This will create a Github helloworld repo for them that they can run locally also, it also creates Jenkins Jobs to allow them do the CI/CD for the new app, they get an email that the app has been created and they can start working and deploying it.

We use the AWS Jenkins plugins to scale our Jenkins Jobs, the Docker registry is also run on AWS and it is where we push the images. All deployments are triggered from Jenkins slaves running in AWS into ECS. Jenkins and ECS triggers SNS events and lambdas store data in DynamoDB.

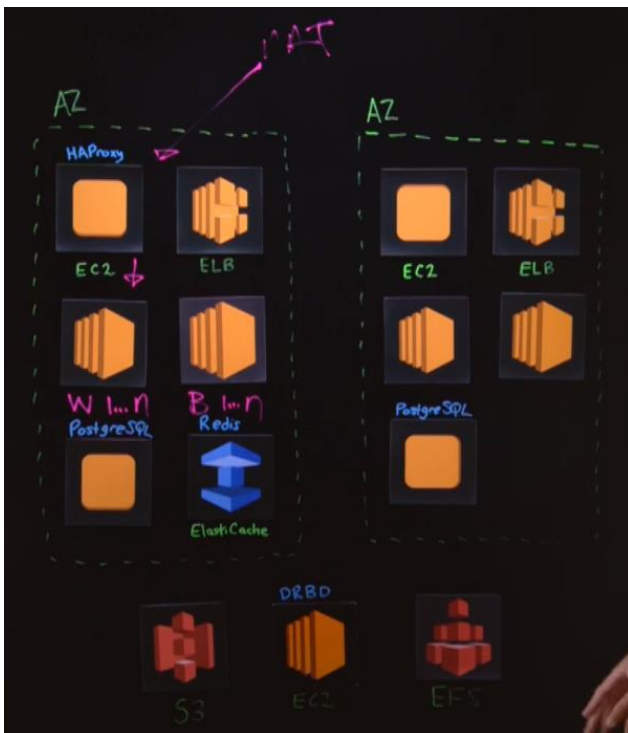
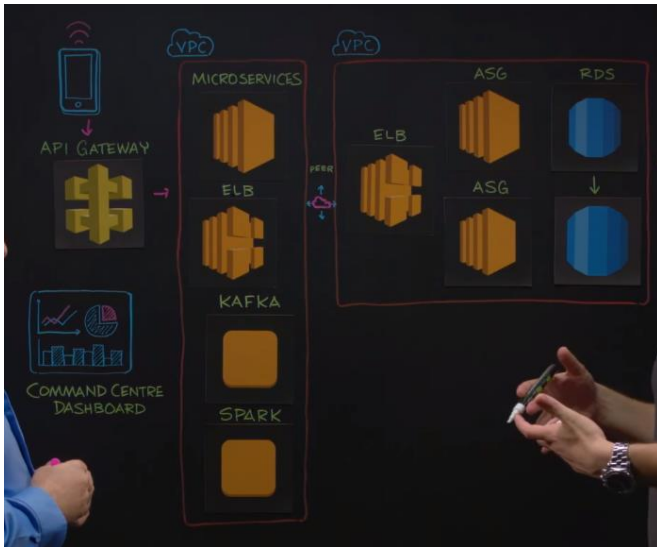


Tableau – Online on AWS

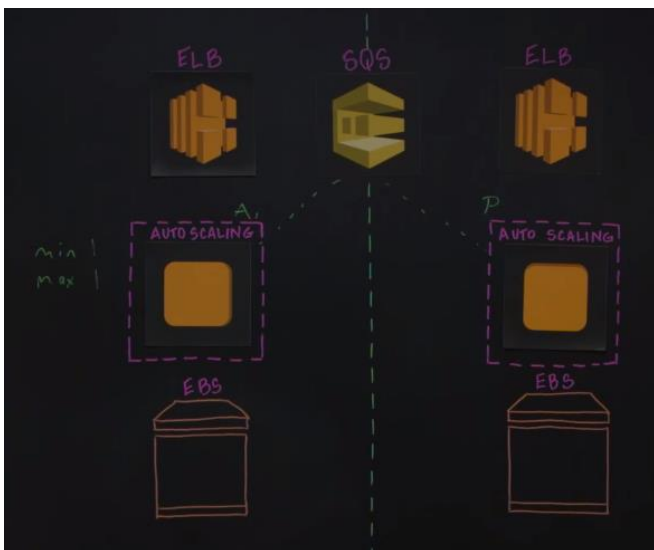
This is a cloud application to help users analyze their data in a multi-tenant fashion for each customer. We have a NAT that uses one of the 2 AZs. In each AZ, there are pools of EC2 instances that take on different roles to do needed jobs like worker roles that are used to render the visualizations to the user. Another role is the Backgrounder role that ingests data on a scheduled basis. We can scale the worker or backgrounder instances depending on the load.

Using HAProxy, we can route or not route traffic to each of the machines in an AZ user pool. PostgreSQL is how we store a lot of our state like username, it is how we achieve multi-tenancy and do reporting per user. Redis Elasticache is our cache layer/store for reuse. S3 is used for storing static content fronted by a CDN like CloudFront. EFS is used to store all the Tableau data extracts and data.



Scale360 – Core Banking in the Cloud

Scale360 build apps for digital banks. Spark jobs are being run for analytics, microservices are running for things like billing and checking account balance all within the 1st VPC while keeping the data and system of records secure in the 2nd VPC



Qantas – Building a Highly-Available, Multi-AZ CIFS Cluster on AWS

Qantas has a rostering application that helps to coordinate ground staff, this needed a HA storage layer solution for their CIFS platform. We have an Active-Passive design with synchronous replication with 2 ASG with a min of 1 and a max of 1 for auto-healing. We also use auto-scaling lifecycle hooks that drops a message to a queue that gets picked up by the passive AZ using long polling.

