Deploying R(eplicable) Data Projects & Systems Like the Iowa Caucus

With the help of AWS, Docker, and Terraform

Tyler Sanders
Clawed Z. Eagle 2016-2019
Head of Engineering, Red Oak Strategic

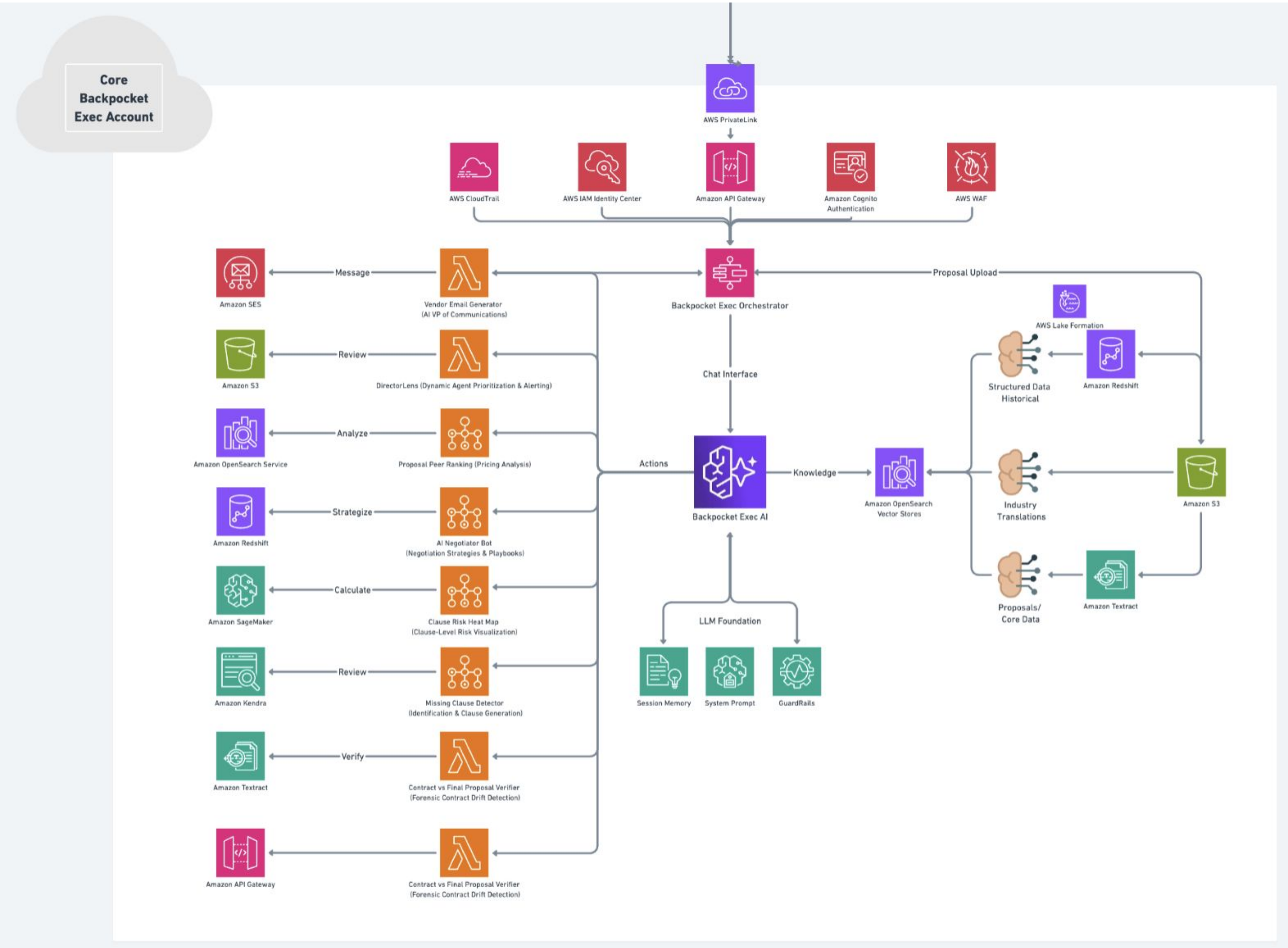
Github Repo: https://github.com/ty-sanders/Deploying-R-Based-Data-Solutions-on-AWS
LinkedIn: https://www.linkedin.com/in/tyler-sanders-8275b5150/
Company Profile: https://www.redoakstrategic.com/about

My Data Journey

- Introduced to R by Professor Moore 2017
- Took 2 additional R-focused courses during my undergrad
- Participated in the Community-Engaged Research Program with Professor Jane Palmer
- Short intern stints in policy areas of interest, Coalition for Smarter Growth (transit+urbanism) and Urban Institute (Housing and youth development in DC)
- Was very uncertain and uncomfortable about leaving AU with a job that would help me start a career and start to address looming student loans, gravitated towards the "hard" skill of coding and data analysis
- Blind applied to an Indeed job with Red Oak Strategic which mentioned political data AND R Tidyverse(!!!)
- Interviewed with my mentor Jake and even though I'd never heard of SQL he realized I was passionate about data and R
- Team of just the 2 of us built and improved together our core political data modeling pipeline, built automations with Amazon Web Services
- Company wanted to start selling AWS Cloud Consulting and as the most junior employee I had the most time to take the certifications and learn the process and eventually got to own that part of our business, worked hard but very lucky
- Now have the chance to lead our engineering teams and largely no longer work with politics, but I get the chance

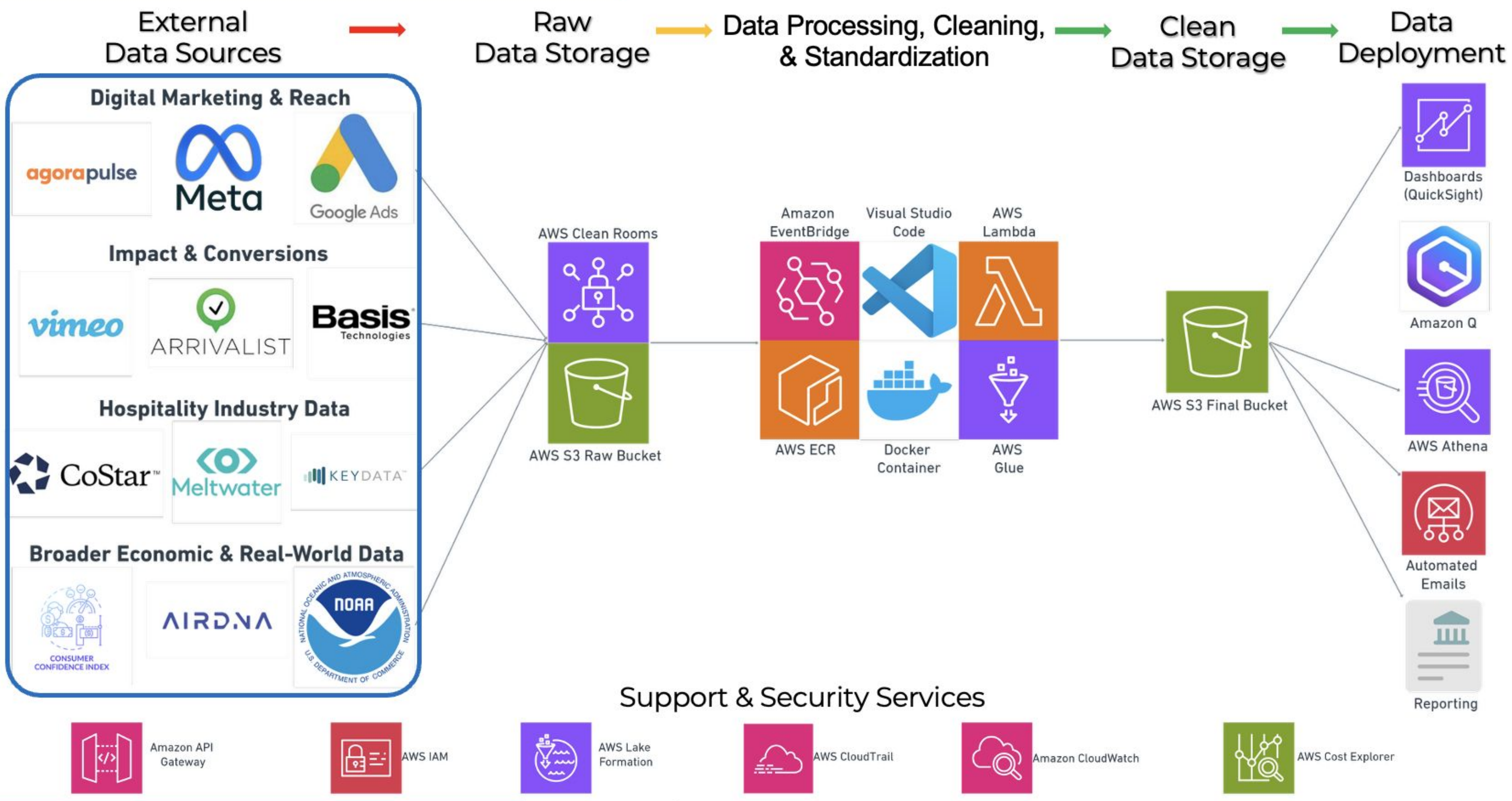
What is System Design (or Solutions Architecture)

- Mostly drawing Boxes and Lines :)
- How to bring everything together to solve a problem - usually a business problem - with data best practices
- Data Scientists care deeply about process - the same practices that make good scripts make great systems



Core Backpocket Exec Account
AWS PrivateLink
AWS CloudTrail
AWS IAM Identity Center
Amazon API Gateway
Amazon Cognito Authentication
AWS WAF
Amazon SES
Message
Vendor Email Generator (AI VP of Communications)
Backpocket Exec Orchestrator
Proposal Upload
AWS Lake Formation
Amazon S3
Review
DirectorLens (Dynamic Agent Prioritization & Alerting)
Chat Interface
Structured Data Historical
Amazon Redshift
Amazon OpenSearch Service
Analyze
Proposal Peer Ranking (Pricing Analysis)
Actions
Backpocket Exec AI
Knowledge
Amazon OpenSearch Vector Stores
Industry Translations
Amazon S3
Amazon Redshift
Strategize
AI Negotiator Bot (Negotiation Strategies & Playbooks)
Amazon SageMaker
Calculate
Clause Risk Heat Map (Clause-Level Risk Visualization)
LLM Foundation
Proposals/ Core Data
Amazon Textract
Amazon Kendra
Review
Missing Clause Detector (Identification & Clause Generation)
Session Memory
System Prompt
GuardRails
Amazon Textract
Verify
Contract vs Final Proposal Verifier (Forensic Contract Drift Detection)
Amazon API Gateway
Contract vs Final Proposal Verifier (Forensic Contract Drift Detection)

Data Lake Architecture & Roadmap



External Data Sources → Raw Data Storage → Data Processing, Cleaning, & Standardization → Clean Data Storage → Data Deployment

Digital Marketing & Reach

agorapulse

Meta

Google Ads

Impact & Conversions

vimeo

ARRIVALIST

Basis Technologies

Hospitality Industry Data

CoStar

Meltwater

KEYDATA

Broader Economic & Real-World Data

CONSUMER CONFIDENCE INDEX

AIRDNA

NOAA

AWS Clean Rooms

AWS S3 Raw Bucket

Amazon EventBridge

Visual Studio Code

AWS Lambda

AWS ECR

Docker Container

AWS Glue

AWS S3 Final Bucket

Dashboards (QuickSight)

Amazon Q

AWS Athena

Automated Emails

Reporting

Support & Security Services

Amazon API Gateway

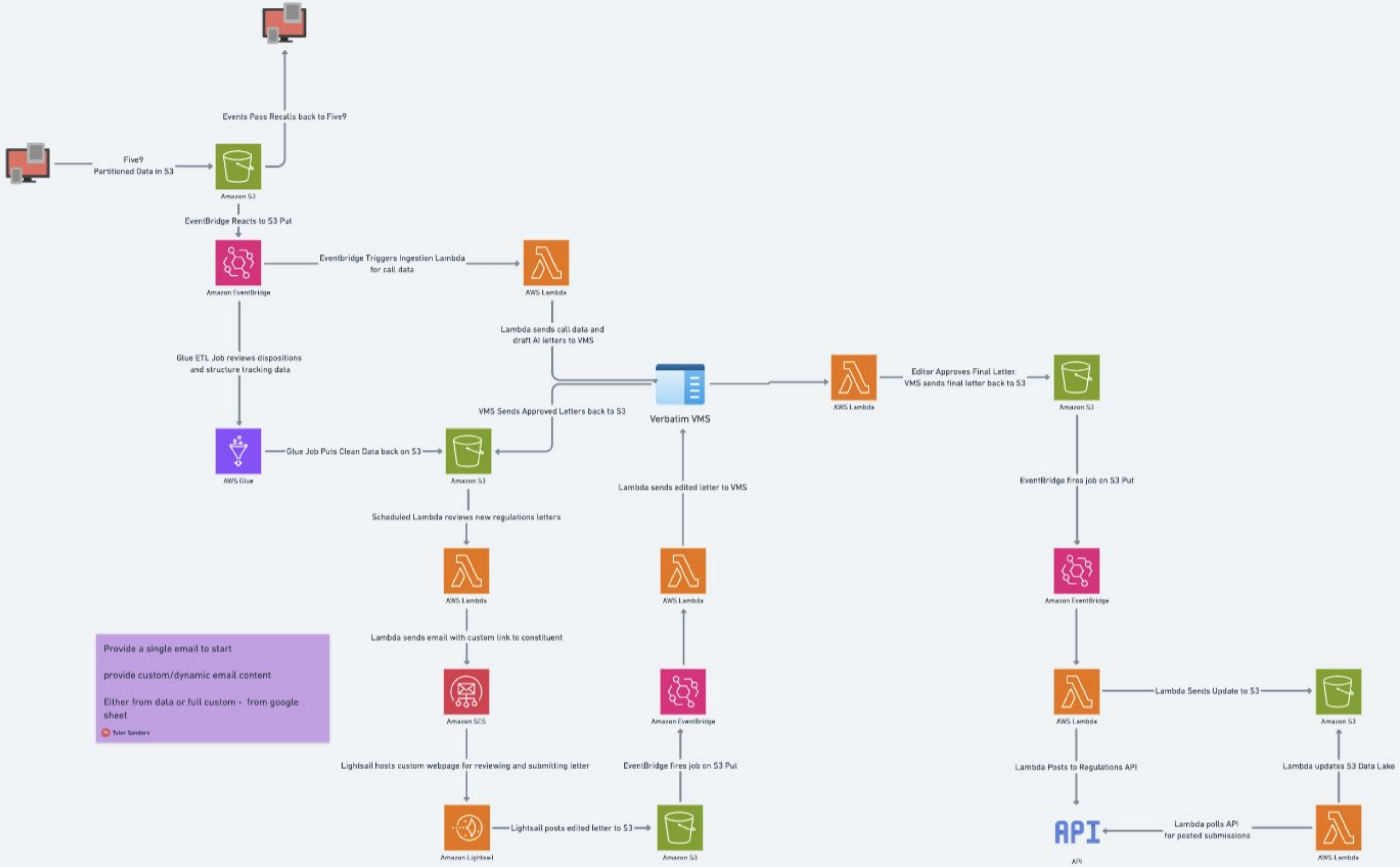
AWS IAM

AWS Lake Formation

AWS CloudTrail

Amazon CloudWatch

AWS Cost Explorer


Events Pass Recalls back to Five9
Five9
Partitioned Data in S3
Amazon S3
EventBridge Reacts to S3 Put
Amazon EventBridge
Eventbridge Triggers Ingestion Lambda for call data
AWS Lambda
Lambda sends call data and draft AI letters to VMS
Glue ETL Job reviews dispositions and structure tracking data
AWS Glue
Glue Job Puts Clean Data back on S3
Amazon S3
VMS Sends Approved Letters back to S3
Verbatim VMS
AWS Lambda
Editor Approves Final Letter. VMS sends final letter back to S3
Amazon S3
Scheduled Lambda reviews new regulations letters
AWS Lambda
Lambda sends email with custom link to constituent
Amazon SES
Lightsail hosts custom webpage for reviewing and submitting letter
Amazon Lightsail
Lightsail posts edited letter to S3
Amazon S3
EventBridge fires job on S3 Put
Amazon EventBridge
AWS Lambda
Lambda sends edited letter to VMS
EventBridge fires job on S3 Put
Amazon EventBridge
AWS Lambda
Lambda Sends Update to S3
Amazon S3
Lambda Posts to Regulations API
API
API
Lambda polls API for posted submissions
AWS Lambda
Lambda updates S3 Data Lake

Provide a single email to start

provide custom/dynamic email content

Either from data or full custom - from google sheet

Tyler Sanders

System Design: Iowa Caucus

Goals:

-Design and build from scratch a system for vote collection, validation, and reporting for the 2024 Iowa Caucus (GOP)
-Do No Harm to American Democracy
-Facilitate accurate data collection from 1,687 caucus sites
-Validate results BEFORE public receives them
-100% reported on Caucus Night

Tyler Sanders

Challenges:
-You've never been to Iowa or run an election

-The last company that tried to run this failed and no longer exists :)

-1,200 volunteer caucus chairs collect the votes and on average they are > 65 years old

-Your a serviceable Data Scientist in R and an experienced AWS dev, but never used Terraform or built a front-end application

Tyler Sanders

Assets:
-8 AWS Engineers and Project Managers (incredibly helpful, cannot contribute production code)

-3 Frontend engineers from a partner firm (very strong)

-Access to some historical data from 2016 at the precinct level

-Roughly $80,000 in credits and budget for resource allocation

-A dedicated War Room in Des Moine for caucus night and 3 days around the event

-Client-side support from 2 caucus/party heads that have been involved with past 4 caucuses

Tyler Sanders

KPIs and SLAs:
-Support a result page with up to 100million visits on caucus night

-Anticipate both domestic and international interference attempts, specifically to network and public endpoints, keep 100% uptime

-Publish first results within 45 minutes of vote closure, complete by 5am

-Final audit should find less than 10% of caucus site discrepancies and a total vote count discrepancy of less than 2%

-All resources deployed using Infrastructure as Code, deployed in Zero-Trust networking space

Tyler Sanders

Stage 1: Questions for client team and engineering team, what else do we need to know to build a valid system?

Stage 2: Overall architecture. How does everything connect at a high level, how do we secure it, what happens if something breaks?

Stage 3: Validation App Data Science - what is the best validation we can manage given the available data and timeline?

Stage 4: 3 real surprises that changed the requirements in the final 2 months of the project - how can we adjust to handle them...

-Worst blizzard in about 30 years hit Iowa 2 days prior (-30 degree, weren't sure we were going to be able to fly in) (less volunteers, more Super Sites (4 or more precincts vote in the same location/ room)

-While the number of precincts were the same, there was a redististricting that shifted the names of the precincts within a county, and they did know or have a re-mapping

-We signed a deal with the Associated Press to create a results API that the news feeds, CNN etc, could use to feed their results (this also include NYT and WashPo)

Stage 5: Compare to the real deal - how does our system stack up against the real Caucus Night

Tyler Sanders

Why did last one fail?!
-They ran everything on a single Application vote collection, (maybe validation), and the front end website. That server ended up crashing and it actually blocked the volunteers from submitting their new votes
-Microsoft Azure (was cloud-based) but a different cloud

-2016 a different company accidently had the wrong list of precinct names, so their system crashed after they received mis-matched entries, took 2 days to sort that out, lawyers get involved, that blocks/slows things down

Are their successes?:

-2012, both parties. They used google sheets. It was far less technical, it was not in the cloud, built on laptops, and it took about 16 hours in total, had about 15% inaccuracy, they collected vote tallies via Phone Tree, manual.

-Not secure, not modern, doesn't scale, far too much manual risk

How do people even get internet connection out there:
-We can't trust the internet outthere
-About 35 hotspots delivered to the hardest to reach spots
-There was a training packet that said - if you can't do the app - just go ahead and call

Tyler Sanders

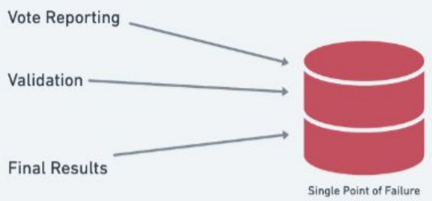
Learnings:

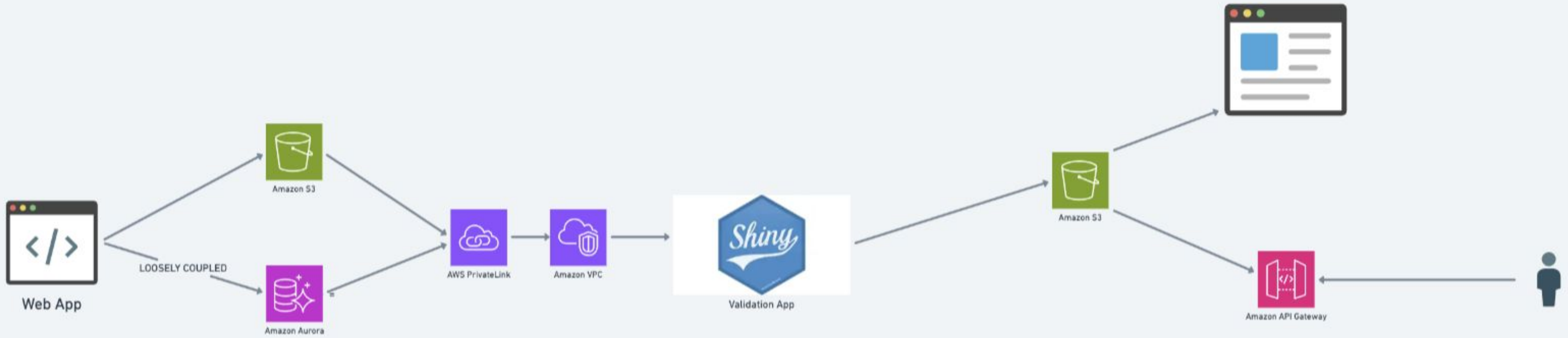
-Single-point of failure, not good, seperate into different components

-They like the phone tree as a backup. If app doesn't work or a volunteer loses their sign-in → they can fallback to calling a hotline number

-We knew from a security perspective, that all of the volunteers HAD TO BE IN IOWA, unlike the reporting app which was global, we geo-restricted sign-in to the app to just IOWA (Iowa shared ISP locations with neighboring states, had to open up to just USA

Tyler Sanders





How to validate:

-We had 6 humans-in-the-loop

What can we do?
-Total % change from 2016
-Total% change from 2020 (1,200, other had 8)
-Total Raw # change from 2020 (8 → 16)
-Caucus → GOP population est
-Trump % change
-Trump total winshare
-The N of votes per time period
-I wanted to wait until 10% of a counties precincts were submitted, and then to look at county-based vote volume
-trends, before we submitted any of those precincts, state-wide
-
-

Tyler Sanders