How Billion Dollar Startups Use a Cloud-Native Approach to Accelerate Growth

A Solution Architecture Guide to Building Data-First Platforms (In the Cloud)



Contents

A Solution Architecture Guide to Building Data-First Platforms (In the Cloud)

Why Successful Startups Are Using a Data-First Mindset	4
The Next Generation of Applications Will Be Built Natively in the Cloud (And It has everything to do with data)	4
Core Technology Components of a Cloud-First, Data Focused Model	5
Components of Data-First Organization	6
Cloud Operations (Infrastructure Automation)	7
The cloud fixes two of the most time consuming, frustrating things: infrastructure & code deployment	8
Infrastructure as Code — Deploy reliable infrastructure in minutes (not months)	8
CI/CD Pipelines on the cloud — Save your developers time & frustration with CI/CD Automation	9
Modern Data Platform Architecture (Data Lakes)	11
The cloud puts the power of data front and center	12
Building a data-first solution architecture with Data Pipelines and Data Lakes	13
Next Generation Applications	15
Let the cloud take care of it for you with Serverless	16
Functions (AWS Lambda, Azure Functions, Google Cloud Functions)	15
Containers	17
API Gateway	
File Storage	18
The future is Serverless	18
Predictive Analytics Using Machine Learning and AI (Only Possible with a Data-First Mindset)	19
Predictive Analytics Using Machine Learning	20
How to Choose the Right Cloud Vendor to Start Your Data-First Journey	
So, what's your cloud strategy?	23
CASE STUDY	24
Digital Citations-A Paperless Parking Citation System	25
It Starts With Connecting Data Points	25
Digital Notification Citation System	26



Twilio

You might have never heard of this cloud-first company, Twilio, but there's a 100% chance you've used them.

If you've ever received an SMS that your Uber driver has arrived, notifications on AirBnB about your reservation, or a 2-factor authentication message via text message on any major platform, you have used Twilio.

Twilio, founded in 2008, has been using AWS cloud since day 1 and has powered many popular services that you use daily. Twilio currently processes 100B API requests annually, all going through Amazon's cloud infrastructure. They were an early adopter to the cloud, and it has paid off in dividends. With an IPO in 2016 and a market cap of \$14.6B, they wouldn't have a business if it wasn't for becoming a cloud-first organization.



Zoom

"Someday someone is going to build something on the cloud, and it's going to kill me,"

> - Zoom CEO, Eric Yuan, prior to leaving WebEx and starting Zoom

Epic Games

Epic Games, creator of the world's most popular video game, Fortnite, uses a cloud-first mentality to deliver Fortnite to more than 200 million players around the world, support growth of more than 100 times in just 12 months, and <u>perform analysis</u> <u>that helps it improve the game</u>¹. Fortnite runs nearly entirely on AWS, including its worldwide game-server fleet, backend services, databases, websites, analytics pipeline, and processing systems.





Uber

Then there's another cloud-first startup who needs no real introduction, Uber.

Halloween and New Year's Eve are the two busiests nights of the year for Uber. Those days can see an increase in 50-100% extra trip volume compared to the rest of the year. Their developers are able to test the scalability of their code in AWS, many weeks before the actual night, all on the fly. Developers spin it up when they need it, and they shut it down when they're done testing. Instead of an all hands on deck control room for both nights, <u>the engineers now monitor these nights from their home.</u>²

Why Successful Startups Are Using a Data-First Mindset

The reason you see so many articles about the application of Artificial Intelligence and Machine Learning, but not that many real-life examples, is because the reality is not many companies have utilized them.

Even though AWS, Azure, and Google Cloud Platform, make it easier for companies to build and scale ML models, many enterprises have many data issues that prevent them from feeding the models. Most of the issues stem from companies not having their data infrastructure setup properly. Whether it's accessibility issues, or data-integrity problems, data has always been an afterthought.

This is where data-first comes in and why so many of the successful startups have been able to thrive: **They** prioritize the structure and access to data.

So, instead of focusing on the applications you want to build first, many companies are focusing on what data they need to power these applications first. **The data powers long-term innovation**. The data powers analytics and insights. The data enables companies to utilize ML and Artificial Intelligence as part of their daily process.

The Next Generation of Applications Will Be Built Natively in the Cloud (And it has everything to do with data)

You can't talk about the benefits of the cloud without talking about Amazon's ground-breaking success and their inspirations.

Amazon is on a mission to bring unbelievable computing power to companies and individuals at the lowest cost possible, even if it means competing with themselves directly. They are using an open data architecture to successively turn every single piece of the company into a separate platform—and thus opening each piece to outside competition.

As developers choose what platform to build their next enterprise application in, it makes sense to build everything within Amazon's cloud ecosystem (or any of the major cloud providers). Wordpress currently powers one-third of all websites on the internet, and for the same reasons why businesses build with Wordpress, it makes sense to build your next startup or enterprise application entirely within Amazon's cloud.

They are building the operating system of the internet, and it will disrupt every vertical in a big way. As enterprises start their cloud journeys, many startups have taken this opportunity to use the cloud to grow and beat their competitors. Amazon pioneered the creation of the cloud, but Azure and Google Cloud Platform are starting to compete in big ways. The best part about the cloud is that there isn't just one single cloud vendor to choose from, and that's potentially the biggest benefit of a cloud-first approach.

The cloud has evolved so fast that technology strategies which worked 3 years ago are completely extinct today. Projects that took 15 engineers to build and maintain are now being replaced by cloud native functions developed by a single developer. So, when we started creating this white paper, we knew we couldn't fit a cloud strategy packaged in one nice piece, and it was disingenuous for us to tell you that there was a one size fits all package.

Let's start with the core technology and business reasons why fast-growing companies start with the cloud.

Core Technology Components of a Cloud-First, Data Focused Model

There are three major reasons why startups and innovative companies choose cloud-first models.

First, scaling applications without the headache of an infrastructure and beneficial pricing models. This is the most obvious and biggest use case for going to the cloud. The major reason Snapchat signed a \$2B deal with the Google Cloud team is for Google App Engine which allows them to scale without worrying about the underlying infrastructure. It's also why companies like Uber use AWS to make sure the service survives Halloween and New Year's Eve, the two busiest nights of the year.

Second, with the creation of Data Pipelines, Data Lakes, and many other data-focused services native to the cloud, **data is treated as a first-class citizen**. You now have the ability to ingest any type of structured and unstructured data across your organization and create insights from disparate data sources, all performed in real-time. This is important as it becomes critical for organizations to make business and operational decisions based on large amounts of unstructured data. Peloton, founded in 2012 and already raised nearly \$1B, places a huge focus on data. They rely on AWS to power its on-demand, live and customizable leaderboard.

Lastly, the big three cloud providers are using a data-first mentality to provide functional APIs that allow you to utilize Machine Learning and Artificial Intelligence computing in a single API call. **This puts unbelievable computing power in the hands of any company, big or small**. Coinbase, the leader in digital cryptocurrency exchanges, uses Artificial Intelligence in the cloud to prevent fraud. And Zocdoc, a company that connects patients with doctors, <u>uses Google's TensorFlow</u>² in the cloud to reduce the wait time for appointments. Without data being treated as a first class citizen, you cannot feed the machines the information it needs.

In the not so distant future, AWS envisions a new generation of developers that won't think about instances, servers, and clusters. Developers will focus on writing software, or possibly purchasing Lambda functions, and it will automatically be available and connected across every imaginable service in the infrastructure. "We are expecting a totally different programming model." says Jassy, CEO of AWS.

The good news is that when it comes to data, the cloud has a little something for every occasion. Each one of these tools serves its purpose to make sure you can process, store, and analyze data natively in the cloud.

Let's talk about our data-first solution architecture in the cloud.

COMPONENTS OF DATA-FIRST ORGANIZATION

Cloud Operations

Modern Data Platform

Next-Gen Applications

THE RESULT? BUSINESS DRIVING INSIGHTS

Cloud Operations (Infrastructure Automation)



The cloud fixes two of the most time consuming, frustrating things: **infrastructure & code deployment**

If you've ever tried to configure & secure servers in preparation for the launch of a new application, you understand the difficulty and frustration of doing this activity.

When we build in the cloud, we have two important values:

- 1) Build reliable and secure infrastructures quickly
- 2) Develop, test & deploy code quickly and automatically (without issue)

These two values have single handedly allowed companies to get better, data-focused applications in production faster and more reliably than ever. The reason that CloudOps automation works so well is that it's **reliable, predictable and scalable.**

CloudOps promises one thing: Less manual configuration and technical setup, and more focus on continuous integration and delivery & infrastructure architecture automation.

There are of course many services in the cloud in terms of infrastructure and DevOps, but there are two areas that stand out the most to us: **Infrastructure as Code & Continuous Integration / Delivery Pipelines.**

Infrastructure as Code – Deploy reliable infrastructure in minutes (*not months*)

A large component on building and supporting a data-first organization on the cloud, is the ability to deploy infrastructure easily and reliably.

That's where Infrastructure as Code (IaC) comes in to play.

Traditional approach

Over the years, IT teams used to provision a physical servers and racks IT infrastructure through a painful and time-consuming manual process. After the hardware was configured, which could easily take months, they had to manually configure the hardware to the proper operating system and application settings. THEN you can get the green light to launch the application.

Of course, after launching your application, it was still a wish and a prayer if the application worked on the live server configuration, even when it worked properly on DEV and QA environments.

Luckily, setting up and configuring your IT infrastructure manually is now considered old school, and you'll understand once you understand the benefits of Infrastructure as code (IaC)

Cloud-first approach

Now, with IaC, you can define what you want the infrastructure to look like using configuration scripts and through a provider like Terraform, re-produce that environment **within minutes.** The best part is that you can define multiple services on different cloud environments. So, you can create an AWS infrastructure with servers, storage & databases, and a separate infrastructure in Azure, **all using similar configuration scripts**.

Core	
Technical	
Differences	

10 Years Ago*	Today
Unpredictable errors: New applications were deployed to old servers, leading to unpredictable errors. You would upgrade servers in-place.	Standard, Predictable and repeatable: New servers are spun up for each new application deployment within minutes for all environments You roll-out brand new infrastructure for new upgraded servers!
Mutable: Each deployment would overwrite the old code, making rolling back challenging to impossible. Version control becomes extremely difficult to manage.	Immutable: The previous "prod" application servers can be kept around for a while until you're confident the new application (running on the new servers) is completed, making roll backs easy and version control much easier. (This is also known as "blue/green" deployments)
Complex phased deployments: Doing phased deployments is hard, because managing the traffic allocations to specific servers required a high level of sophistication.	Automated advanced phased deployment: The load balancers live in the cloud, making it easy to deploy your new application (on new servers) to, say, 5% of your audience for testing. All completely automated, and reliable.

CI/CD Pipelines on the cloud

Save your developers time & frustration with CI/CD Automation.

The first part of CloudOps is building reliable infrastructures quickly and that's what IaC accomplishes. The second aspect of CloudOps is developing and deploying reliable code to production environments and that's what where CI/CD automation comes into play.

Now, continuous integration and continuous delivery have always been a thing. However, with cloud-native services, it became much more feasible.

The cloud goal: Automate delivery and deployment (using cloud-native services)

Here is an overview of a CI/CD pipeline.



Retrieved from: https://aws.amazon.com/devops/continuous-delivery/

The best part about building a CI/CD pipeline is that it can all be done in a serverless fashion.

Core benefits of CI/CD in the Cloud

Significant reduction in configuration

Noticing a trend here?

One thing that can't be removed by hosting a CI solution in the cloud is the need for configuration. However, it can be significantly reduced. By using a cloud solution, the configuration requirements are reduced just to those required for your CI pipeline(s).

All you have to do is choose the one that best fits your software and make supplementary changes. As an example, Azure provides basic commands for such languages and frameworks as Go, Java, PHP and Python

Dynamic Scalability

The greatest advantage of using the cloud is its elastic nature, which means that you can scale your infrastructure both up and down, as you see fit. No more guessing on how much hardware you need to acquire, and definitely not having to fight with the finance team on what's needed. Get what you need. No more, no less.

Flexibility

In an Azure pipeline, you can continuously build, test, and deploy to any platform and cloud.



Retrieved from: https://azure.microsoft.com/en-us/services/devops/pipelines/

So, this means if you started with two core technologies and decided to add or integrate different technologies, the cloud pipeline will almost always support this new configuration.

Compare that to a homegrown CI solution, where you would have to re-build and configure again to support your new technical environment. If you want to build data-focused applications quickly and reliably, start with a CloudOps mindset.

Modern Data Platform Architecture (Data Lakes)



The cloud puts the power of data **front and** center

It comes down to this:

The easier it is to access your data, the easier is to transform your business. A modern data structure allows you to create insights to enhance business applications and enable new digital services.

Here's an example:

Let's say you wanted to know more about customers who complained about your brand on social media.

At a fundamental level, you can gather insights about the positive and negative sentiment of the social media posts for your brand. But, what if you wanted to go deeper? What if you wanted to tie these complaints to actual customers and understand the impact of your customer service team across social and telephone?

You will want to know the answers to questions like this:

- What specific incident triggers customers to complain the most on social media?
- Are the customers who complain on social media more likely to cancel the service than the baseline customer? (Social Media Feed + Billing Data)
- What regions/service locations complained the most on social media?
- What types of customers complained the most?
- Do the customers who complained on social media try to resolve their issues on the website first before going to customer support?
- How many channels did the customers use before complaining on social media?

This data empowers businesses to understand the most profitable customer cohort, the cause of customer churn, and the promotions or rewards that will increase loyalty.

This above scenario is very difficult to solve if you don't have a modern data structure. And that's what a data-first architecture solves.

Building a data-first solution architecture with **Data Pipelines and Data Lakes**



A modern data architecture in the cloud has **3 main** components

The ability to collect and ingest data from multiple data sources (structured and unstructured)

IT teams have usually been restricted by the ability to only store structured data. This has now changed with the creation of Data Lakes and Data Pipelines in the cloud. **Data lakes allow you collect any type of data, store it in a single, accessible location (at scale).** Financially, cloud platforms have made this type of service a no-brainer.

Now, instead of having to only store and utilize structured data from specific systems, you can collect unstructured data from ERP Systems, Social Media, Network Traffic Logs and CRMs with buyer data, as an example. Cloud native architecture doesn't discriminate against the type of data.

The ability to natively process and move structured and unstructured data between compute and storage services.

You'll often hear this referred to as a data pipeline. All of the major cloud providers provide cloud native services that allow you to store and process data. For example, when you have your data stored on Amazon S3, you're able to utilize cloud-native serverless services that can access and manipulate data with zero configuration.

AWS Glue is a fully-managed, pay-as-you-go, extract, transform, and load (ETL) service that automates the timeconsuming steps of data preparation for analytics. An ETL utility that is native to the cloud and integrates with several other services is a very powerful tool to transform and process data. A game-changer for developers in many ways.

The ability to easily produce analytics and insights

In cloud-native environments, data is not locked into and by default is easily accessible internally as well as externally. In many instances, you have the ability to integrate with leading external big data analytical tools like Tableau and Looker.

Now that your data is all in one location, you can start with producing analytics and explore how you can feed this data into Machine Learning applications.



Retrieved from: https://aws.amazon.com/big-data/datalakes-and-analytics/what-is-a-data-lake/

AWS Athena is an interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL. Simply point to your data in Amazon S3, define the schema, and start querying using standard SQL.

Athena helps query data from S3 files without loading them into a costly storage like Data Warehouse or Relational Databases. This cuts out many other IT processes needed by data analysts to get the data they want. The actual speed of data access using S3 & Athena is slower than a traditional cloud database, but engineers can achieve results much quicker since they don't need to set up the databases. Slower speed, but much faster for business goals.

Next Generation Applications (Serverless)



Let the cloud take care of it for you with **Serverless**

Many companies, large and small are in the beginning stages of utilizing serverless functions that are being offered by all of the major cloud providers.

As an example, Netflix uses a serverless architecture to give real-time recommendations based on petabytes of data and Clash of Clans uses a serverless architecture to predict when a user is going to exit the game.

So, what is serverless?

Here is how Amazon describes serverless:

Serverless is the native architecture of the cloud that enables you to shift more of your operational responsibilities to your cloud provider, increasing your agility and innovation. Serverless allows you to build and run applications and services without thinking about servers. It eliminates infrastructure management tasks such as server or cluster provisioning, patching, operating system maintenance, and capacity provisioning.

Let me summarize for you: All services are native to the cloud and completely managed by your cloud provider. You never have to worry about your infrastructure failing or not scaling.

In the grand scheme of things, when it comes to building a data-first approach, building applications without worrying about servers is a massive benefit. There are many serverless applications, but let's focus on a few that can make a major impact on how you deploy applications. **Functions, Containers, API Gateways and File Storage**

Functions (AWS Lambda, Azure Functions, Google Cloud Functions)

Since there are many serverless components, let's dive deep into one powerful service. When it comes to computing, one of the biggest serverless services that all three cloud providers offer is an ability to run Functions. AWS Lambda lets you run an application without rigging or managing servers. It is an event-driven computing platform. The code is executed in Lambda upon configured event trigger. You pay only for the compute time you consume—there is no charge when your code is not running.



Lambda Use-Case

A simple use case could be that when an image is uploaded into Amazon Simple Storage Service (S3), a Lambda function could automatically resize the image for mobile, tablet, and desktop devices. The event that triggers the Lambda function is the file being uploaded to S3. Lambda then executes the function of resizing the image.

This is all done without managing an infrastructure.



Retrieved from: https://aws.amazon.com/serverless/

Lambda functions can be used for:

- 1) RESTful API
- 2) Any transformation/conversion of the input message
- Any scheduled or event-based triggers of other AWS services
- 4) Infrastructure security checks
- 5) Rules engine
- 6) Alerting triggers
- 7) Low latency applications
- 8) Arbitrary script that can be executed on a data set

Containers

Containers provide a standard way to package your application's code, configurations, and dependencies into a single object. Containers share an operating system installed on the server and run as resource-isolated processes, ensuring quick, reliable, and consistent deployments, regardless of environment.

The AWS Cloud offers infrastructure resources optimized for running containers, as well as a set of orchestration services that make it easy for you to build and run containerized applications in production.

Benefits:

1) **Run anywhere** - Containers package your code with the configuration files and dependencies it needs to consistently run in any environment.

2) Improve resource utilization - Containers provide process isolation that lets you granularly set CPU and memory utilization for better use of compute resources.

3) Scale quickly - Each container runs as a separate process that shares the resources of the underlying operating system. This enables containers to start and stop quickly.

API Gateway

API Gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale. API gateways are critical to data-first organizations as it saves them significant time in configuration and testing.

File Storage

From storing user generated content (S3), to replacing tape-backups (Glacier), Hybrid onpremise, cloud (AWS Storage Gateway) and Big Data (Amazon EBS) solutions, there are no limitations of what types of data can't be stored in the cloud.

The future is serverless

The reasons to not use an all-native serverless cloud approach when building applications are dwindling. There are three main reasons we chose this approach:

Ability to scale up or down with almost zero DevOps (aka NoOps):

All of the core functions/services are managed completely by the cloud provider. It's the responsibility of the cloud provider to scale the services up or down based on the performance metrics.

Automation of deployments and change management:

With proper tools like AWS CloudFormation or Terraform, the Ops team can focus on following Infrastructure as Code (IaC) and Immutable Infrastructure patterns and not waste time with scaling or maintenance of servers. The ratio of infrastructure to Ops team headcount can be significantly improved.

Engineering Alignment from Top to Bottom:

If the organization decides to use a cloud-first approach within the entire engineering department and every team is given autonomy over selection of services, many architectural complexities can be simply reduced to serverless services price analysis and performance optimization. Many engineering hours can be saved by utilizing pre-built cloud services instead of building our own. There are few exceptions to this, but will cover those in later blogs

"Although EC2 instances will be around forever, containers and serverless computing are completely rethinking the basic unit of compute which is getting much smaller, and the new kinds of workloads benefit from different types of compute power." ~ Andy Jassy, CEO, AWS Predictive Analytics Using Machine Learning and AI (Only Possible with a Data-First Mindset)



Predictive Analytics Using Machine Learning and Al (Only Possible with a Data-First Mindset)

Machine Learning and AI are all the rage of every technology trend report. What's not being covered often is how companies like Amazon are making these technologies available to use by offering APIs.

"If you look at the continued pace of innovation in AWS this year, we'll launch a little over 1,800 significant services and features in 2018 up from 1,400 a year ago. And a thousand the year before that. The pace of innovation is getting faster and faster," says Jassy.

I agree that not everyone is ready to implement ML and AI into their business. In most cases, organizations aren't ready because they don't have their data set up properly, which as discussed, Amazon has the infrastructure to support.

When you combine your data lake, data pipelines, and event-based pipeline with ML and Analytics you can make predictions and smart recommendations. It opens up many possibilities.



Machine Learning Services from Leading Cloud Providers

Predictive Analytics Using Machine Learning

It's everyone's dream to predict what's going to happen next, and with predictive analytics, organizations are getting better at it. Again, this is only possible if your confident in your data governance and stewardship. **Here are three common uses for predictive analytics:**

Optimize a marketing campaign for when someone is most likely to buy – Once you identify the characteristics of someone who is near the finish line of buying your product, then you can start targeted campaign messages or discounts to finish the deal.

Personalized user experiences – As mentioned before, Clash of Clans uses data powered by their serverless architecture to predict when a user is going to exit the game. Their able to customize the next sequence of events to keep them engaged in the game.

Detecting fraud – By accessing a variety of unstructured data, you can build analytics models that can detect fraudulent customer activity, in real-time.

And many more ...

Most common Machine Learning services

The next benefit of having a data-first approach is that you can utilize popular ML models in the cloud, some even being vertical specific.

Here are a few that stand out to us when we're using a data-first model.

Natural Language Processing (NLP) – You can use a service like AWS Comprehend to find and understand relationships in your unstructured data. A typical use case is uncovering the sentiment of your customers through customer support emails, social feeds, and app store reviews. In AWS Comprehend Medical, they take a step further where you can take unstructured data such as a doctors note or invoice, and identify medical information, such as medical conditions, medications, dosages and strengths.

You can use tools like custom classification and custom entity recognition to group certain types of text that you're evaluating. An example of this is being able to auto-route a customer support email to the right team based on the content alone (and no prior classification).

Object Detection (Images or Videos) – API's like Amazon Rekognition allow you to add image and video analysis to your applications. The service can identify the objects, people, text, scenes, and activities in real-time.



Retrieved from: <u>https://aws.amazon.com/rekognition/</u>

As you can imagine, the possibilities are endless with this type of technology.

Text Extraction (OCR on steroids) – A service like AWS Textract allows you to extract text and data from any document. It's essentially OCR, but with applied machine learning, which means its able to extract data without hardcoded rules or formatting issues (which OCR needs).

Topic Modeling - Topic modelling gives you the ability to auto-detect what "topics" occur in a collection of documents. This used to be a manual, time-consuming process where you had to structure and organize the information first before evaluating. Now, it can be done automatically through code.

And then for any other modelling, you can use a service like AWS Sagemaker. Sagemaker provides every developer and data scientist with the ability to build, train, and deploy machine learning models quickly.

How to Choose the Right Cloud Vendor to Start Your Data-First Journey



How to Choose the Right Cloud Vendor to Start Your Data-First Journey

We found that for the most part, when comparing the big 3 cloud vendors (AWS, Azure, and Google Cloud Platform), they had nearly identical features/services across the board, but several key differentiators.

Amazon is by far the leader in the creation of new services, and for many reasons, it makes sense to choose Amazon as a fast-mover.

If you're going with a hybrid cloud approach (on-premise and private/public cloud) or are already using Microsoft enterprise products such as SQL Server, Windows Servers and Powershell scripts then Azure is the easy choice. If you're going big on AI and Machine Learning, then we highly recommend using GCP for a majority of these services.

The best part about all of these choices is that you don't have to be tied into a single vendor. Use Amazon for storage and compute, Azure for the hybrid approach, and GCP for AI and Machine learning.

What's more important, however, is understanding how others are using the cloud to transform their business to help you get a clear picture of why a cloud-first strategy is your best approach to building a next generation business.

So, what's your cloud strategy?

If someone asked you what your cloud strategy was, you might well have been asked what the meaning of life was.

The solution architecture decisions we made 3 years ago are now completely different thanks to infrastructure automation, modern data platforms and serverless architecture. The cloud is moving the power of technology into the hands of everyday company's, not just behemoth Fortune 500 organizations.

CASE STUDY



Digital Citations—A Paperless Parking Citation System (An All-Amazon Serverless Cloud Solution)



The worst part about parking in Los Angeles isn't trying to find an actual parking spot. It's finding out you have a \$75 parking ticket hanging on your windshield when you come back to get your car.

And to be honest, you'll be lucky if the piece of paper with your heavy fine is still there. Often times, the citation falls off the windshield wiper and you'll never even know you received a ticket in the first place. And let's be real, sometimes you lose the paper citation or just rip it up because you don't need that type of negativity in your life right now.

So, we thought this all-paper experience was horrible for LA citizens, and wanted to do something about.

Introducing: Digital Citations Finder-Get notified digitally when you receive a ticket.

It Starts With Connecting Data Points

As part of an open data initiative, the city of Los Angeles <u>provides daily file uploads</u> containing parking citations in the LA region. It's a pretty impressive data file that gives you details on every single citation issued, the location of the ticket, and a timestamp (among other things).

The data pipeline starts with downloading and copying the data file from the original Amazon S3 bucket, and placing it on our own S3 bucket.

[CODE SNIPPET]

s3 = boto3.resource('s3')
s3.Object(dest_bucket.name, filekey).copy_from(CopySource = {'Bucket': source_bucket.name, 'Key': filekey})

[/CODE SNIPPET]

This is a really informative data set.

CITY OF LOS ANGELES, CA

03/18/2019 | 04:14 PM

LICENSE PLATE #

CITATION NUMBER 706063466

W. Washington Blvd & S Flower St.

By 03/30/2019

After 03/30/2019

60

S100

Payment Options

(866)-561-9742

www.lacity.parking.org

Parking Violations Bureau Los Angeles CA 80030-0247

WHEN

WHERE

LR9689

VEHICLE Silver Toyota Camry

REASON

Meter Violation

Parking Violation

Digital Citations—A Paperless Parking Citation System

We have two powerful use cases from this data:

1) Get a digital notification of your citation. This notification includes all of your parking details as well as information on how to pay the ticket. If we have access to DMV records of vehicle registrations, we can retrieve the contact details of the car owners and send push notifications automatically.

2) Visualize historical citation data so you determine which regions/zip codes you're most likely to get a ticket in.

To take this further, you can even set up push notifications to let the user know they are parking where many people received parking tickets. Since we have the exact latitude and longitude, this is possible.

Great things happen when you can connect multiple disparate data sets.

Digital Notification Citation System To create this experience, we need two things to make it work:

1) Original data from the City of Los Angeles (we have that).

2) Matching license plate to email address (for this POC, we created our own database).

It works like this:

You park > You get ticket > You get notified via email and SMS the next morning with the ticket details.

[CODE SNIPPET]

```
from botocore.exceptions import ClientError
        client = boto3.client('ses',region_name=AWS_REGION)
try:
response = client.send_email(
        Destination={
          'ToAddresses': [
                         RECIPIENT ],},
                 Message={
                         'Body':{
                         'Html': {...},
                         'Text': {...},
                         }.
                         'Subject': {...},
                         }.
                 Source=SENDER,
        except ClientError as e:
                 print(e.response['Error']['Message'])
```

[CODE SNIPPET]

This makes sure every citizen receives their citation information and has easy next steps. This experience would actually make it easier for cities to get paid faster.

Visualizing Historical Data (Insights)

As part of our technical architecture, and all-Amazon cloud infrastructure setup, we made it easy to visualize the data over a period of time (not just the daily file). **To make the visualizations, first we need to put all data into AWS Redshift.**

[CODE SNIPPET]

con = psycopg2.connect(dbname=*, user=*,password=*, port=*, host=*)
 MessageBody = json.loads(record['body'])
 citation =json.loads(MessageBody['Message'])

cursor = con.cursor()
 update_query = """INSERT INTO public."ticketinfo"
 (ticketid, tickettime, zipcode)
 VALUES (%s, %s, %s);"""
 cursor.execute(update_query, (ticketid,tickettime,zipcode))
 con.commit()

[CODE SNIPPET]

Top 29 places in Los Angeles with the most parking violations:



All-Amazon Cloud Native Approach

At Egen, we truly believe startups and corporations will be creating these types of experiences all within the Amazon cloud ecosystem.

Using an Amazon Native approach, we built a fully automated solution to ingest and transform the Los Angeles Parking CSV data into a data warehouse (AWS Redshift) that makes violation visualizations for each zip code region dynamically, as well as generate citation emails.

The beautiful part of serverless architecture is that you can build applications without worrying about setting up servers (EC2 instances), scaling, and other DevOps-related maintenance. AWS takes care of all of these thing so you can focus on the use case.

Technology Stack:

- 1) Amazon S3 (to store the original CSV files)
- 2) AWS Redshift (for running analytical queries)
- 3) AWS QuickSight (for visualization)
- 4) AWS SES (to send email)
- 5) AWS SQS (to queue messages)
- 6) AWS SNS (Amazon's simple notification service)
- 7) Lambda functions (for individual use case code execution)
- 8) CloudWatch Scheduler (to schedule the data refresh)
- 9) Python (for any scripting)

The architecture looks like this:



Engineering break down of the architecture: https://docs.google.com/document/d/1DVaCSCrx9ITQk93BFZAuLuOMIvGA9zGAPbrEgOAYPXU/edit#

Do you have the right cloud partner to start your data-first journey?

For the past 18 years, Egen Solutions has helped companies build and scale smarter, more efficient customer focused solutions. Today, we help companies build cloud-native solutions in AWS, Azure and Google Cloud Platform. We don't believe "the cloud" is a buzzword. It's the foundation for how next generation data-first companies are built.

Whether it's building a brand new product, or building a product that integrates with your existing IT systems, we got you covered. We help our clients get to market faster and smarter.

To start a conversation about your data-first journey, contact digital@egen.solutions

