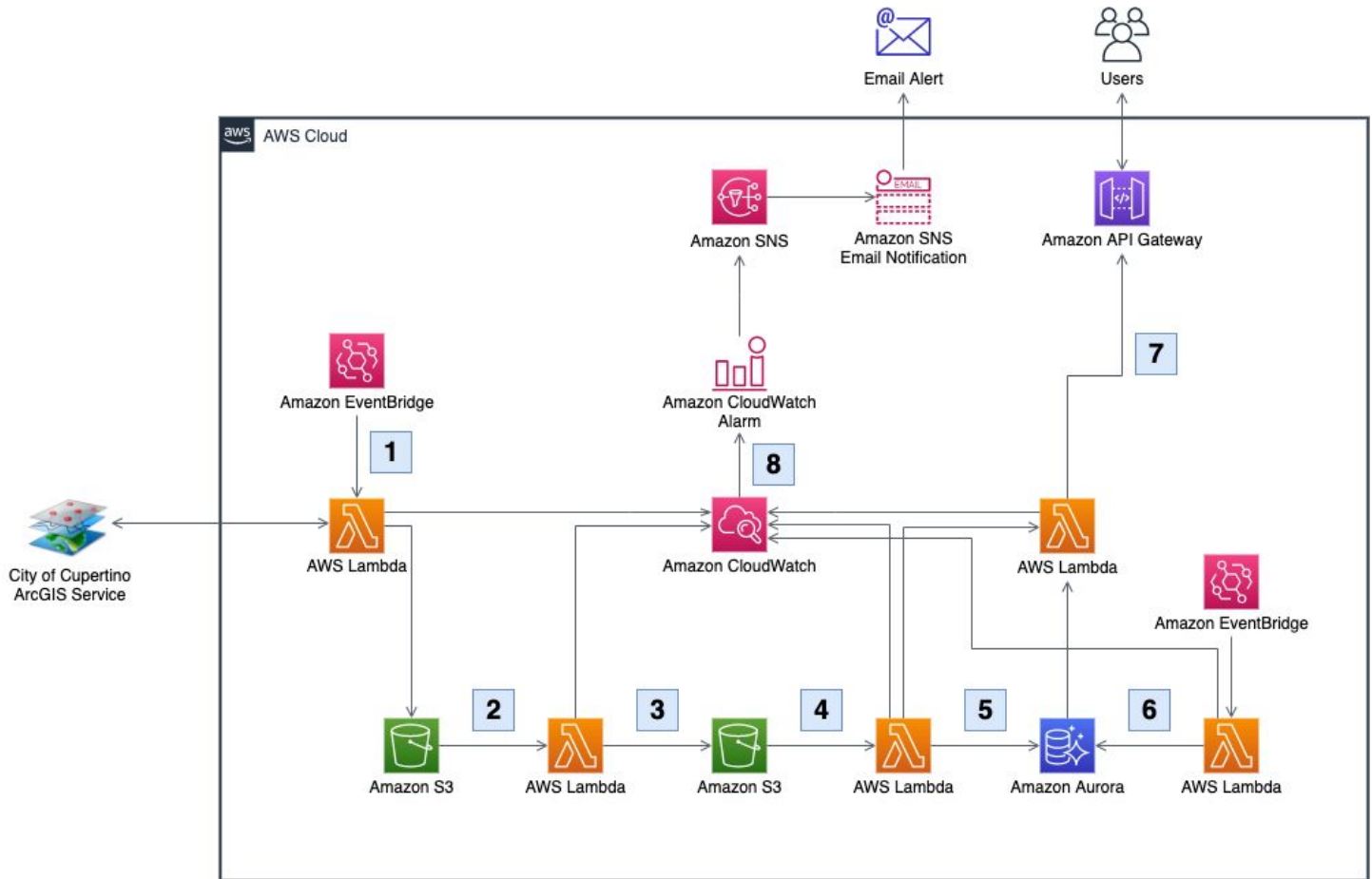


Gridics Architectural Exercise v2 (Implemented in AWS)  
Brandon Worthington 5/17/2020



The diagram above shows a system for ingesting data using resources from Amazon Web Service that has been fully implemented. These are the general steps involved in ingesting, normalizing, storing, and exposing data.

1. Amazon EventBridge schedules an AWS Lambda function to run on a regular basis (e.g., every night) and retrieve data from ArcGIS services for the city of Cupertino. The raw JSON data is stored in an Amazon S3 bucket. Lambda is an event-driven, serverless, scalable computing service.
2. Once a new raw data file is added to the S3 bucket, a different Lambda function is invoked.
3. The Lambda function reads the new file from S3 and normalizes the data to match the format of a table in a database. The normalized data is saved to a different S3 bucket.
4. Once a new normalized data file is added to the S3 bucket, a different Lambda function is invoked.
5. The next Lambda function saves the normalized data to an Amazon Aurora database. Aurora is scalable, high-performance, MySQL-compatible database service.
6. Another EventBridge schedule triggers a Lambda function to run on a regular basis to refresh tables in the database that contain the results of time-consuming queries. The SQL queries that are run are stored on S3.
7. Users or other services can request data from an API defined by Amazon API Gateway. API Gateway sends the request to a Lambda function, which can query the database. API Gateway then returns the response from Lambda to the user or service that made the request.

8. Amazon CloudWatch monitors the health and status of the various services used in this system. It tracks key metrics, which can be viewed in the CloudWatch dashboard. An alarm is set to trigger if an error ever occurs while a Lambda function is running. The alarm causes Amazon Simple Notification Service (SNS) to send an email notification.