Data Engineering and Analytics on AWS

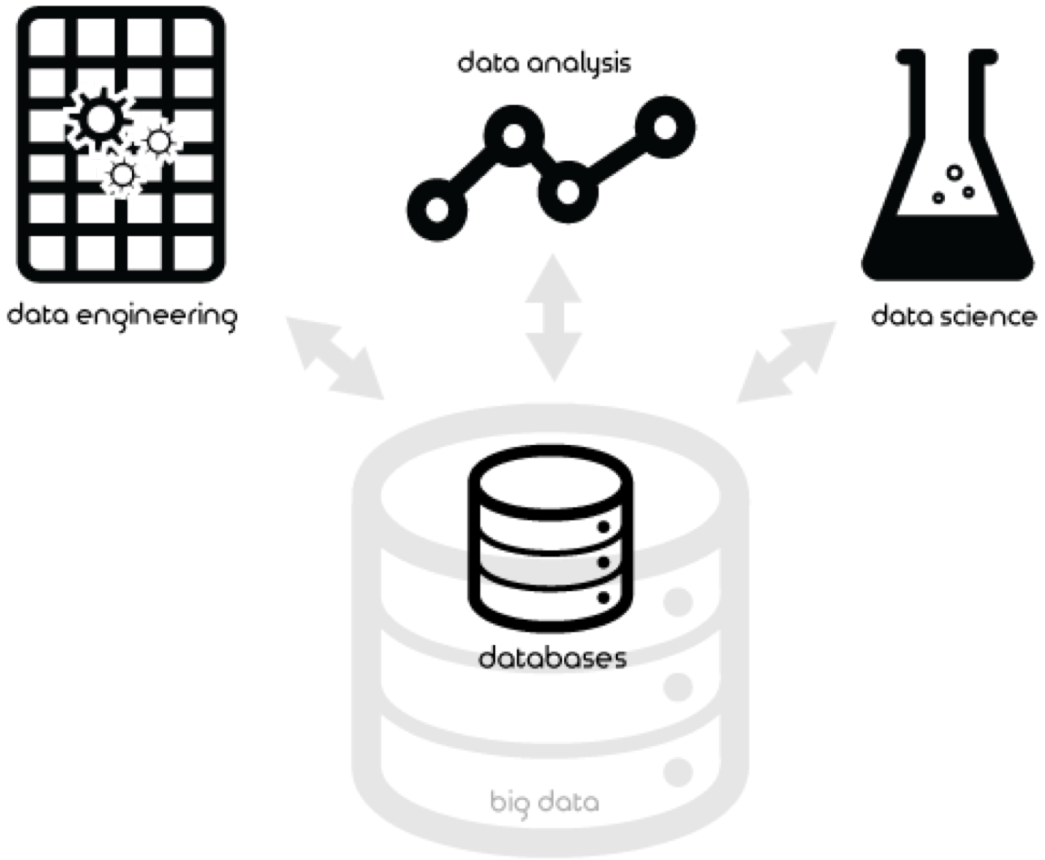
Collect data, visualize, and share insight

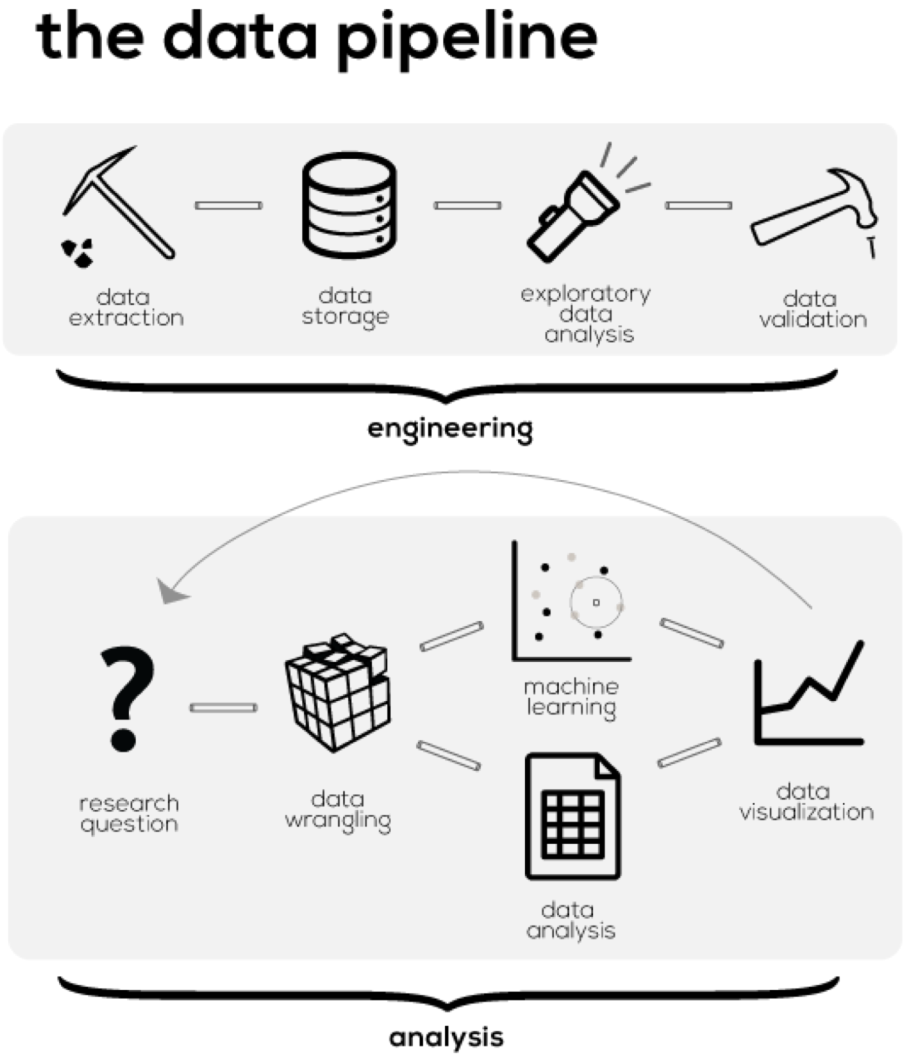
Jung, SeUng, Big Data & Analytics SA, AWS

Data Engineering

Data Engineering

Prepare data pipeline, data catalog, and ETL for data analysis





Catalog

상품 관련 추천

상품 리스트 / 상세 정보

상품 카테고리

AWS Glue – 데이터 카탈로그

데이터를 쉽게 찾고 관리할 수 있게 해주는

Data Source : S3, JDBC 호환 Database, DynamoDB

크롤러는 자동적으로 데이터 스키마를 찾아서 저장

Apache Hive Metastore와 호환

데이터의 검색과 ETL 작업을 가능

테이블 스키마 정보와 컬럼 레벨 통계 정보를 포함

- S3 Data Source: 거의 모든 텍스트 유형의 파일 지원

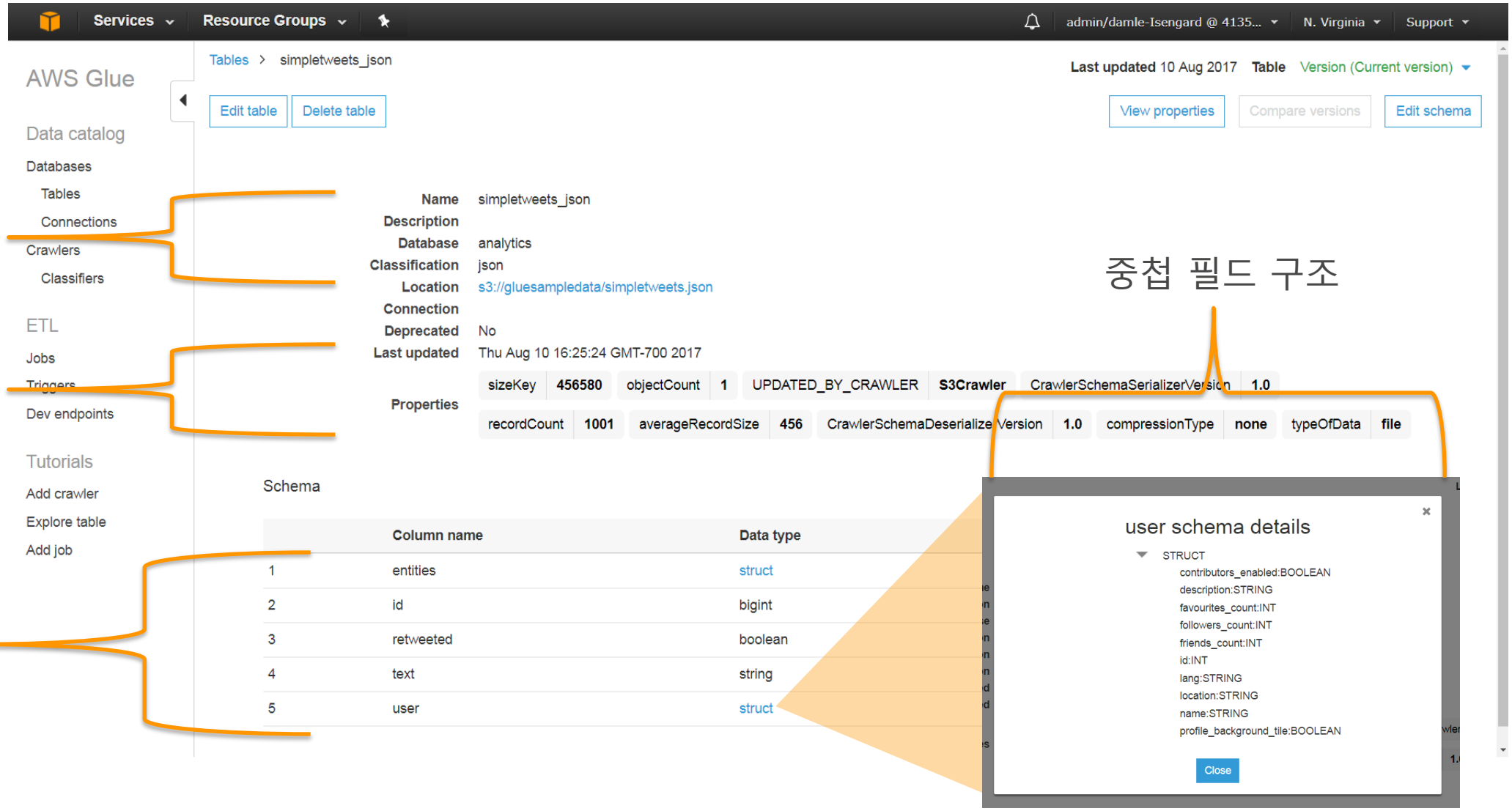
  (Parquet, ORC, Avro, JSON, CSV, TSV, Apache Log)

데이터 카탈로그 - 테이블 상세 정보 포함

테이블 속성

데이터 분포 통계

테이블 스키마



Transforming Data

Over 90% of ETL jobs in the cloud are hand-coded

Which is good …

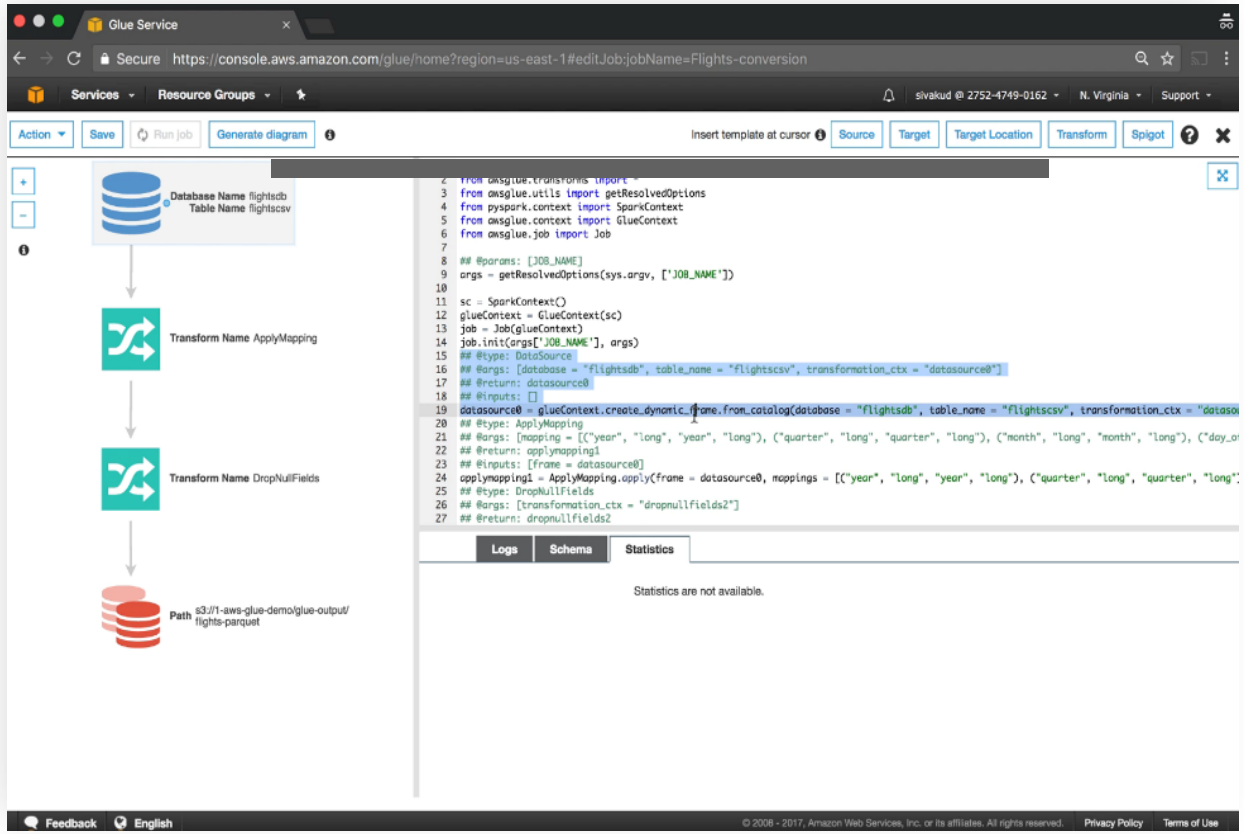
- Flexible
- Powerful
- Unit Tests
- CI/CD
- Developer Tools …

… but also bad!

- Brittle
- Error-Prone
- Laborious
- Sources Change
- Schemas Change
- Volume Changes
- EVERYTHING KEEPS CHANGING !!!

AWS Glue - ETL 서비스

Job 스크립트 작성과 실행을 쉽게 도와주는



서버리스 데이터 변환작업

Apache Spark 기반

클릭 몇번으로 생성되는 ETL code

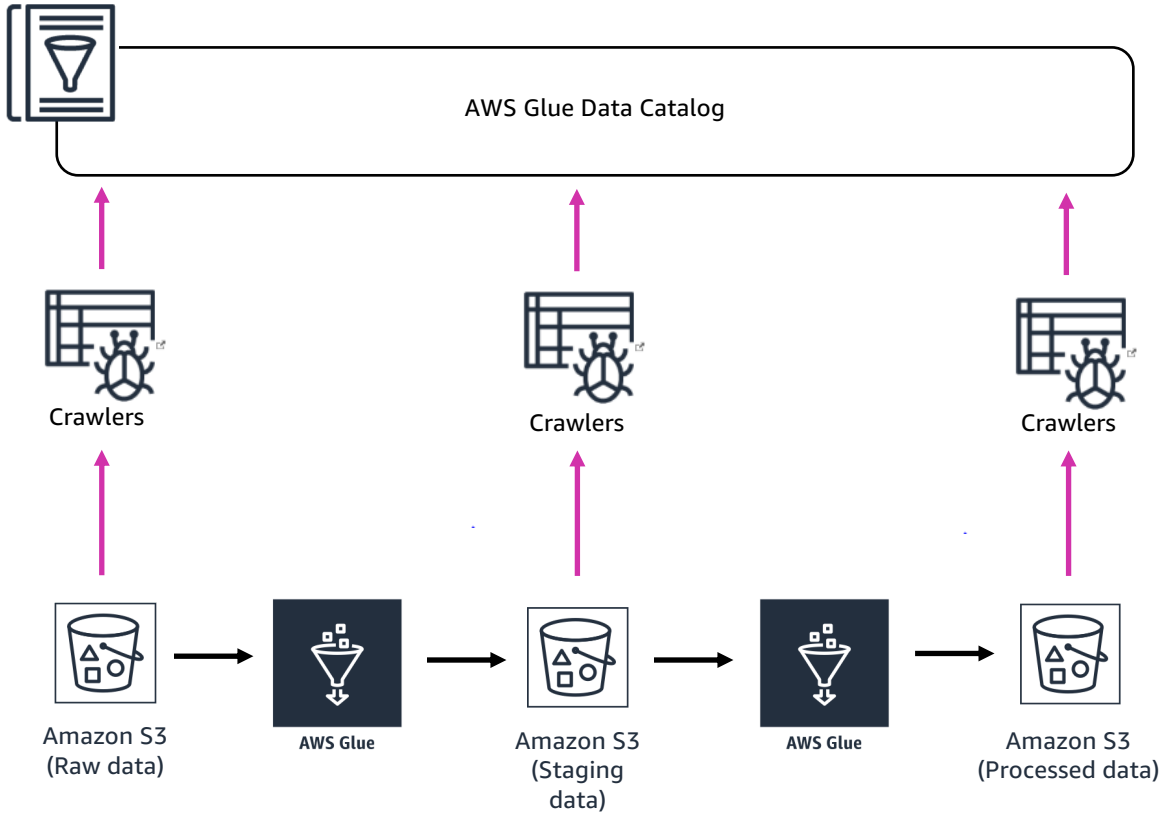
수정 / 추가가 가능한 **PySpark**과 **Scala** 코드

반복 일정과 **이벤트**에 따른 Job 스케줄링

Zeppelin, PyCharm 등 익숙한 환경에서 수정, 디버그, 테스트가 가능하도록 Dev Endpoint 제공 (Python 2.7 지원)

- *Sample ETL Codes : https://github.com/awslabs/aws-glue-samples*
- *About Dev Endpoint : https://docs.aws.amazon.com/ko_kr/glue/latest/dg/dev-endpoint.html*

AWS Glue - 데이터 클랜징, 준비, 카탈로그



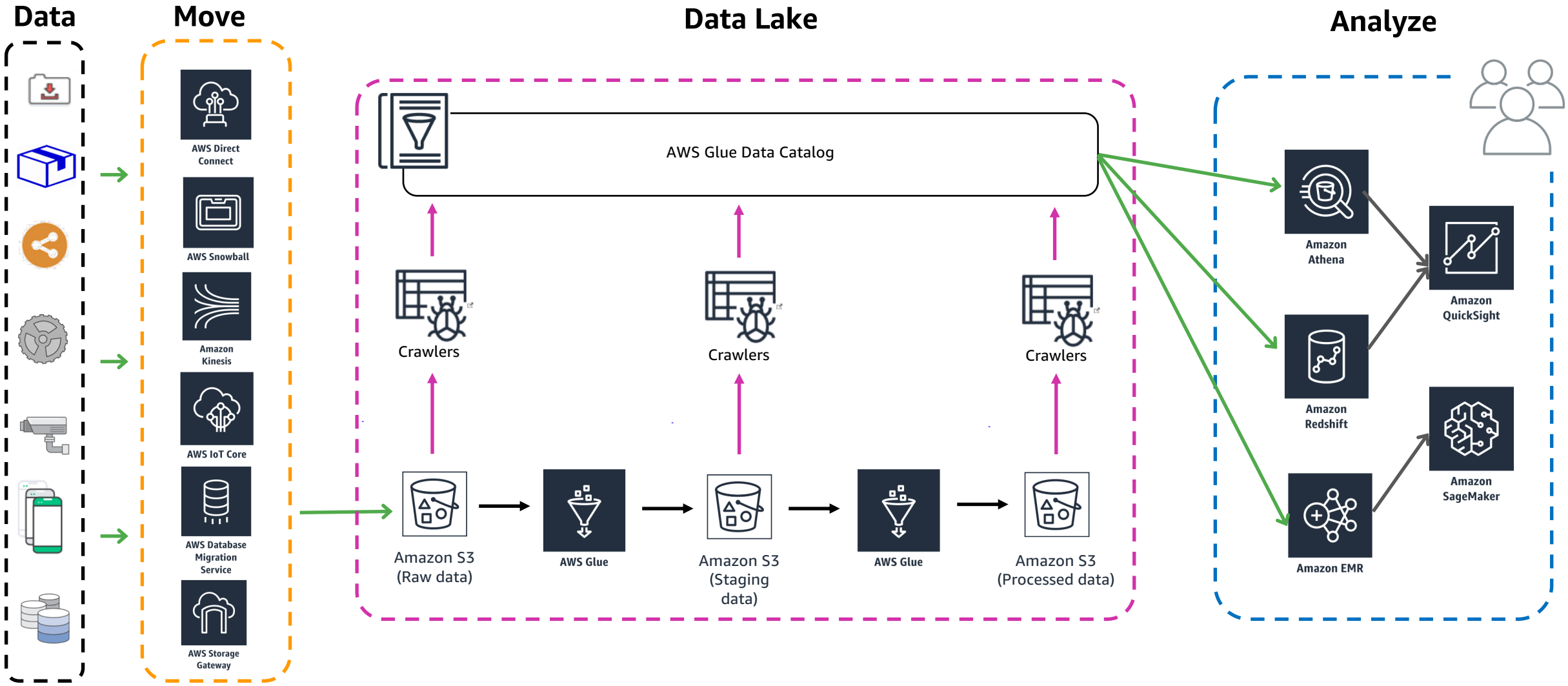
AWS Glue Data Catalog - 데이터 레이크에 대한 단일된 뷰

- Automatically **discovers** data and stores schema
- Makes data **searchable**, and available for ETL
- Contains table definitions and custom metadata

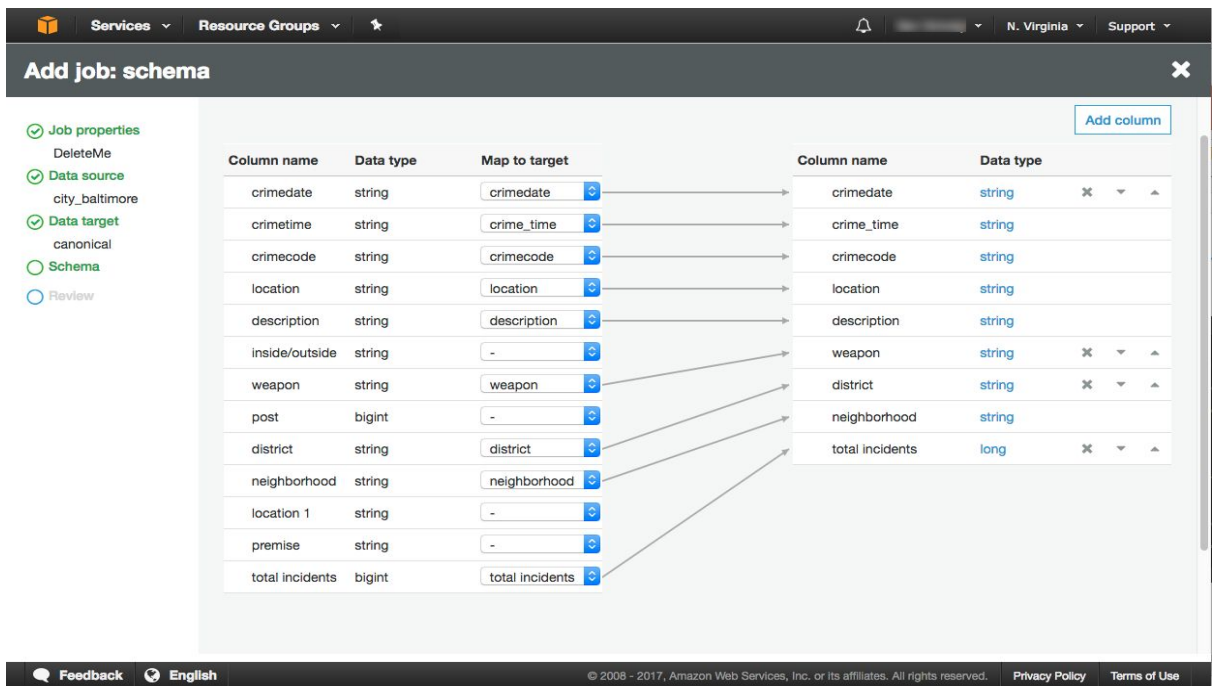
AWS Glue ETL job - 데이터 클랜징, 변환 등의 데이터 프로세싱

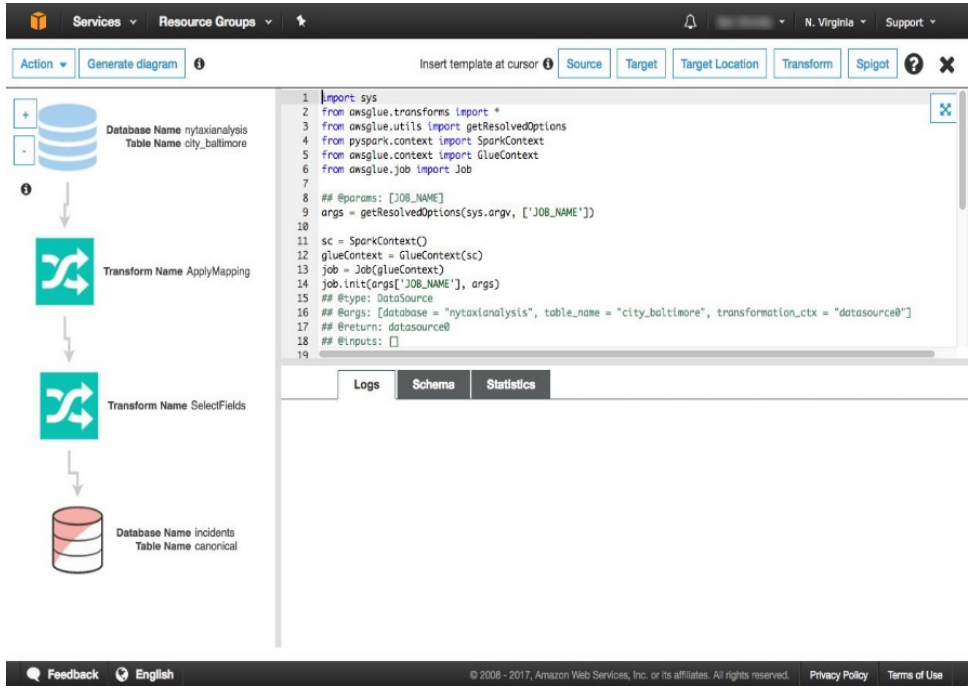
- **Serverless Apache Spark** environment
- Use Glue ETL libraries or bring your own code
- Write code in **Python or Scala**
- **Call any AWS API** using the AWS boto3 SDK

AWS Glue가 데이터 레이크의 중심 역할



Job 생성 - 콘솔에서 코드 생성

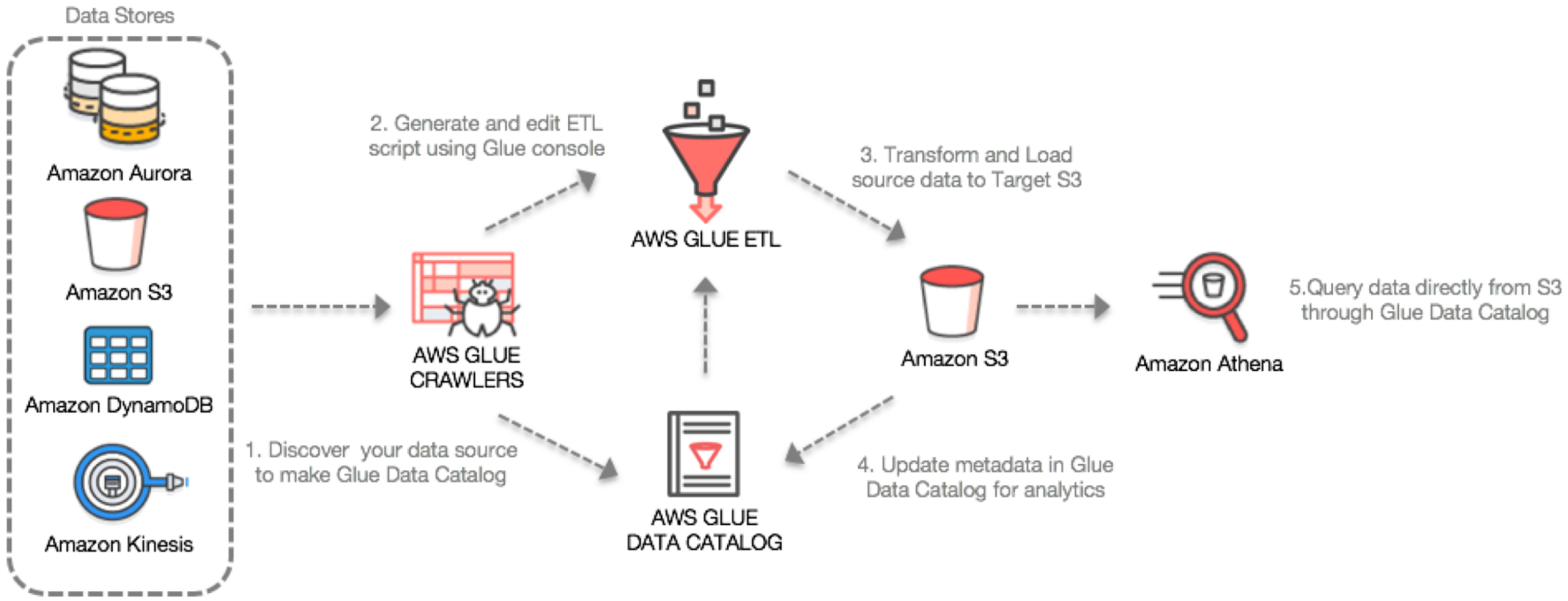




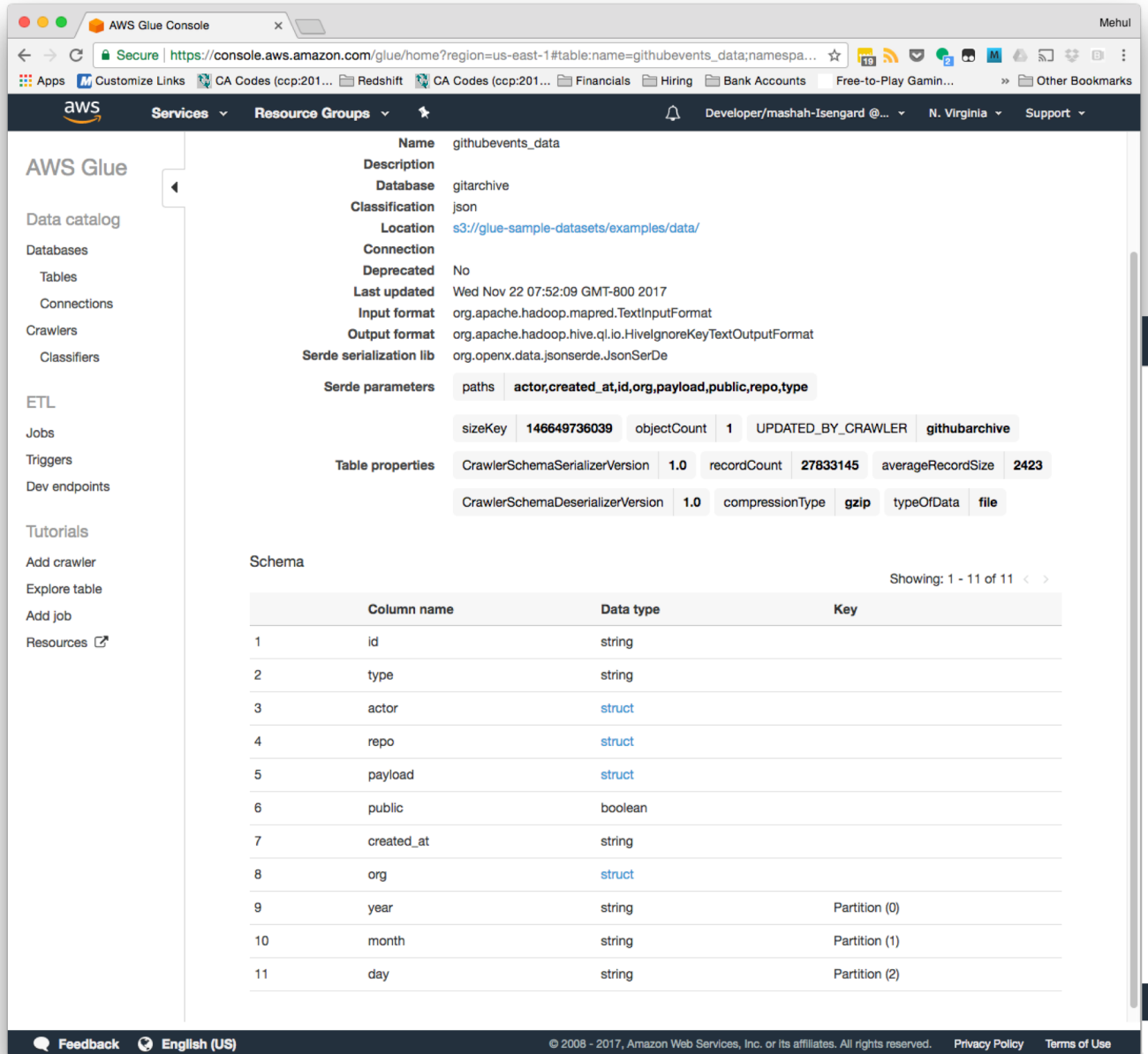
1. 콘솔에서 컬럼 단위의 맵핑을 수정하면
2. Glue에서 자동적으로 데이터 변환 그래프와 **PySpark** *(또는* **Scala)** 코드를 생성, 직접 수정 가능
3. 직접 사용하는 노트북 서비스로 **Dev Endpoint** 이용하여 코딩 가능

AWS Glue 를 활용한 ETL 주요 단계

1. 데이터 소스에 Crawler를 통해 데이터 카탈로그 생성
2. 컬럼 단위의 맵핑을 통해 자동 코드 생성
3. 편리한 환경에서 자유롭게 코드 수정 및 테스트(/w Dev-Endpoint)
4. 실제 운영 환경에서 Job 스케줄링 및 실행

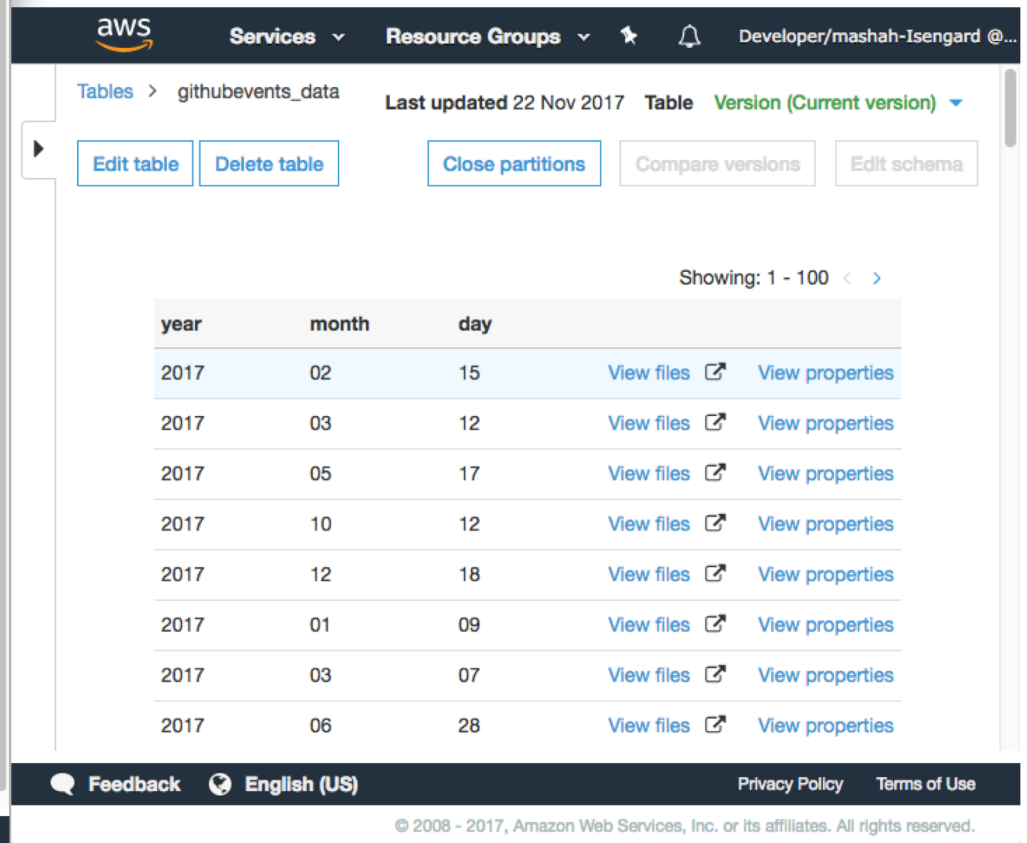


Step 1 : Crawler 실행

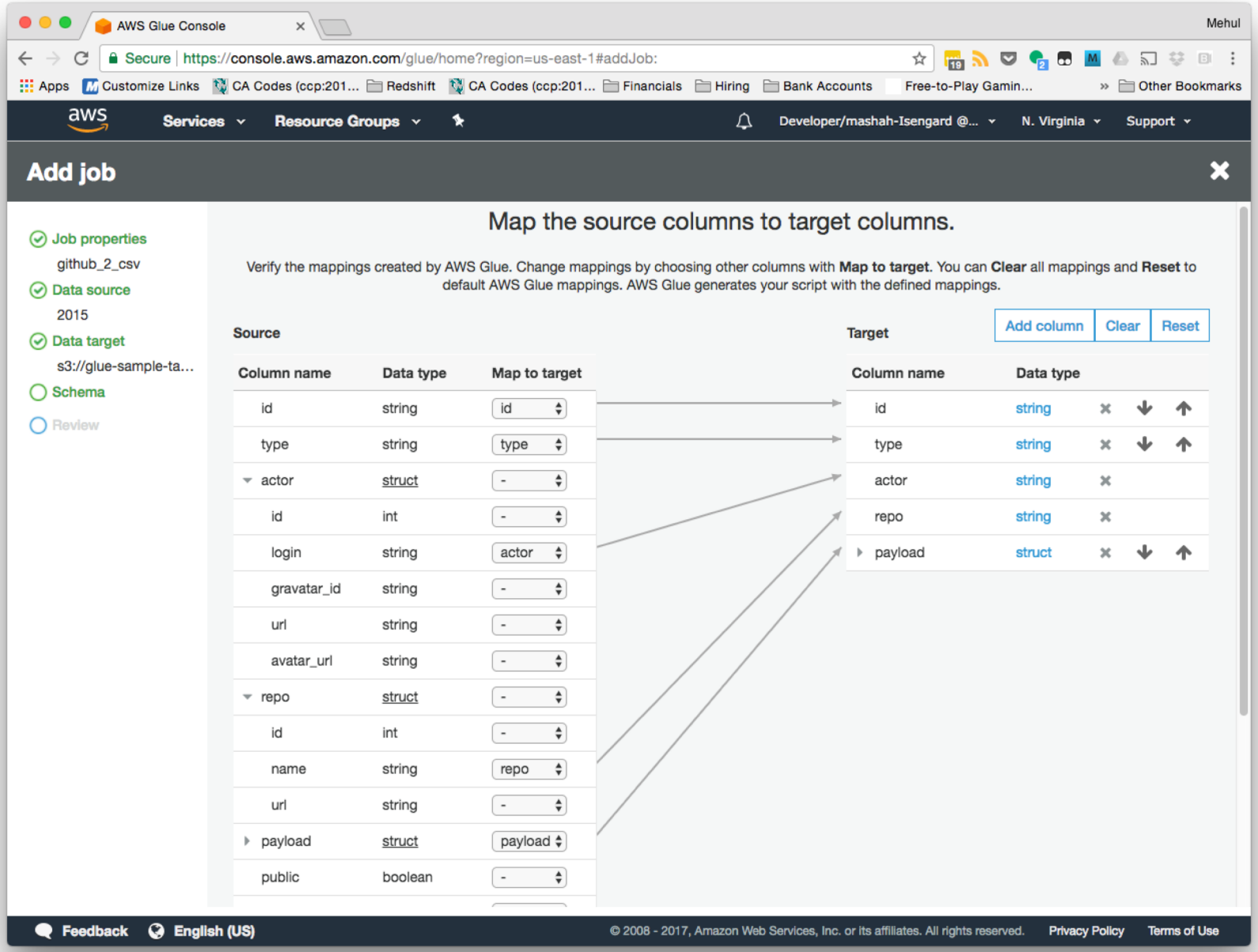


버킷 / 폴더 / 파일유형에 따라
테이블 메타정보 저장

Hive metastore 호환 파티션



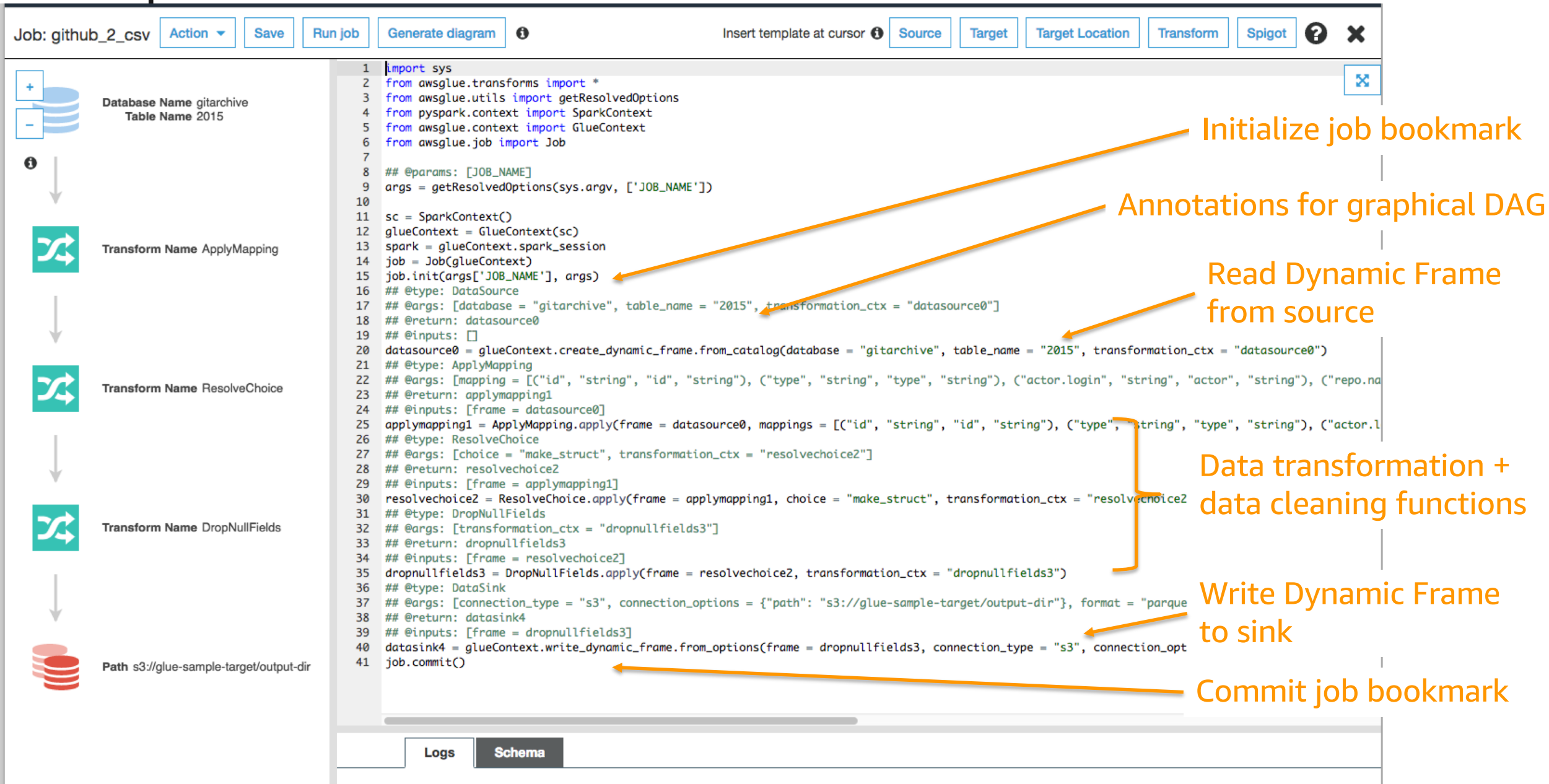
Step 2 : 자동 ETL 코드 생성



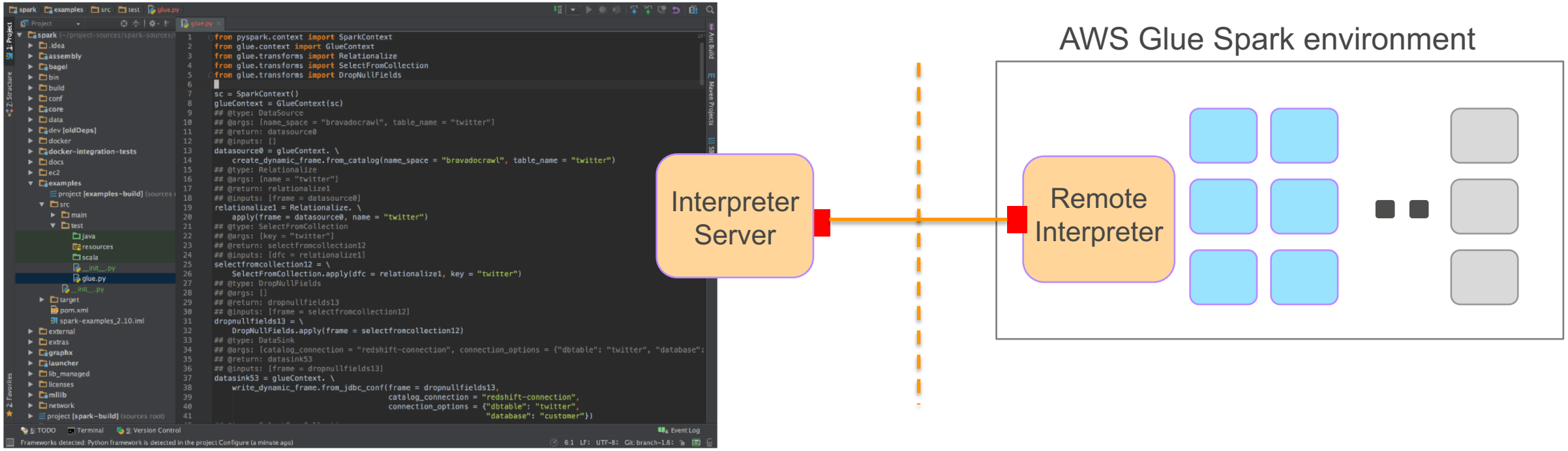
데이터 소스와 타겟 지정

컬럼 레벨의 스키마 변경 가능

Step 2 : 자동 ETL 코드 생성



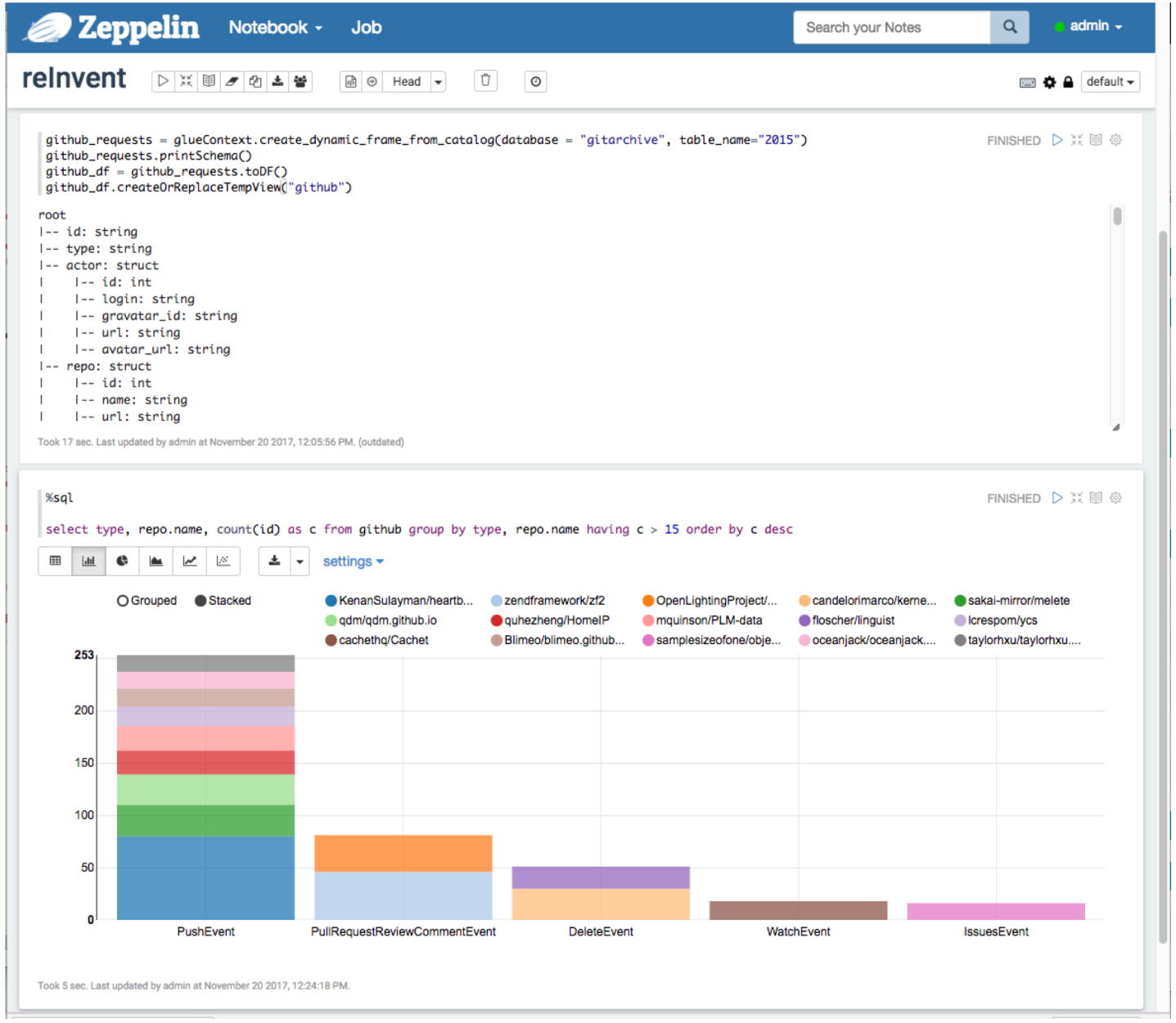
Step 3 : 코드 수정 및 데이터 탐색(/w Dev-Endpoint)



AWS Glue의 Dev-Endpoint를 이용해서 자신의 IDE 환경으로 접속

인터렉티브한 ETL code 개발, 디버그, 테스트 환경 구축

Step 3 : 코드 수정 및 데이터 탐색(/w Dev-Endpoint)

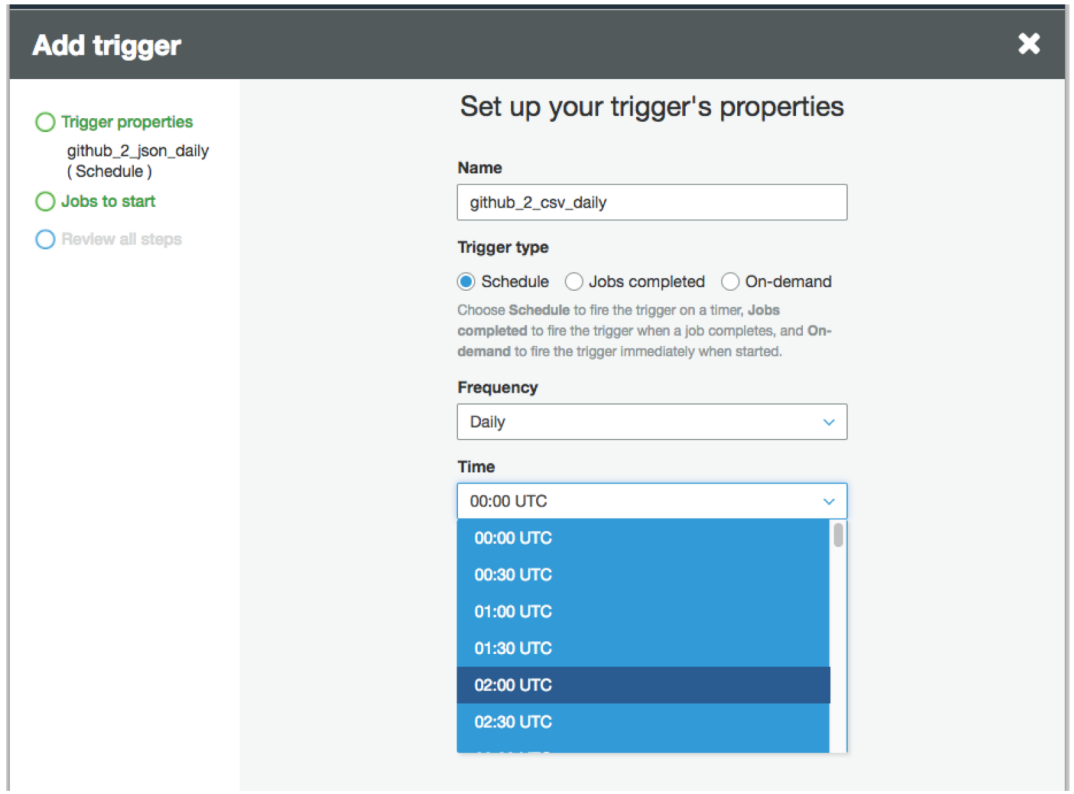


Apache Zeppelin 노트북을 통한 AWS Glue Dev-Endpoint 접속

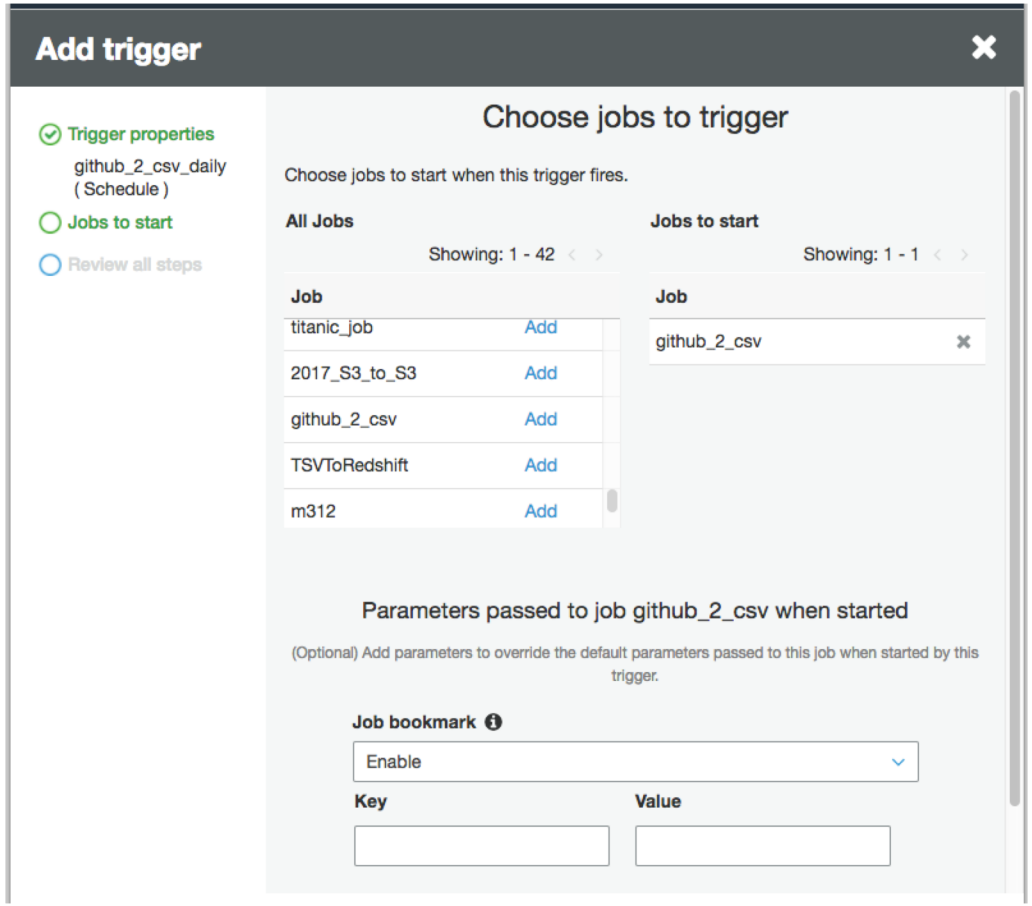
인터렉티브한 데이터 쿼리 및 탐색이 가능

https://docs.aws.amazon.com/ko_kr/glue/latest/dg/dev-endpoint-tutorial-EC2-notebook.html

Step 4 : Job 스케줄링 및 실행

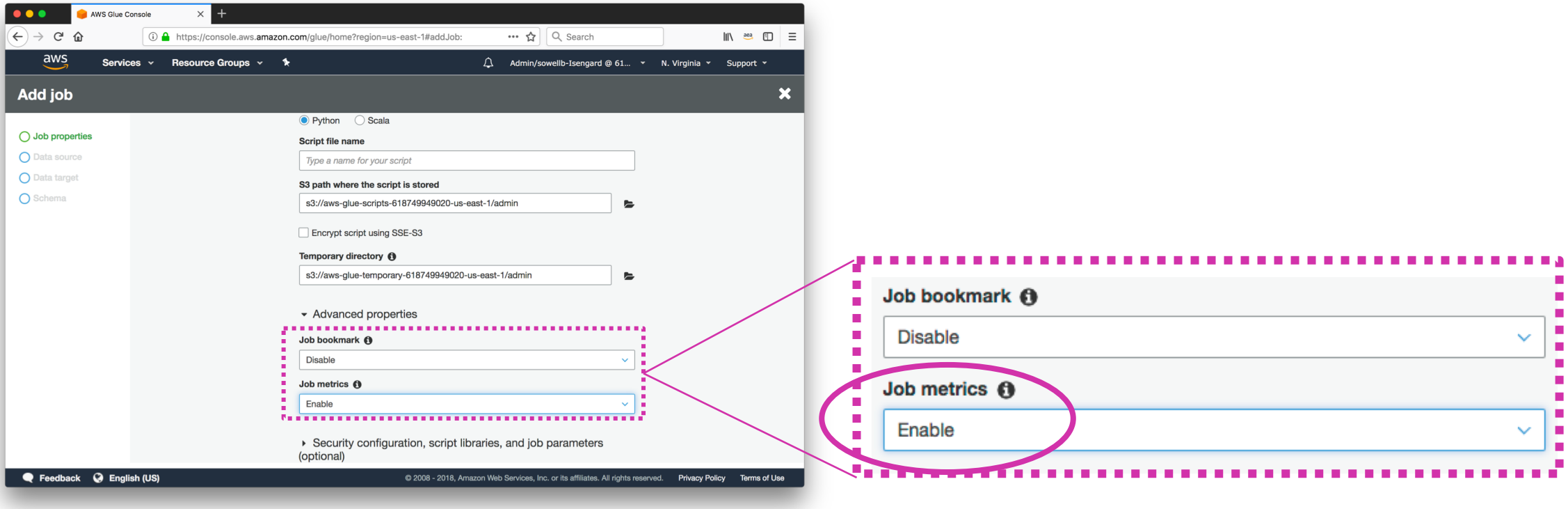


이벤트 유형과 빈도 설정



필요한 파라미터 설정

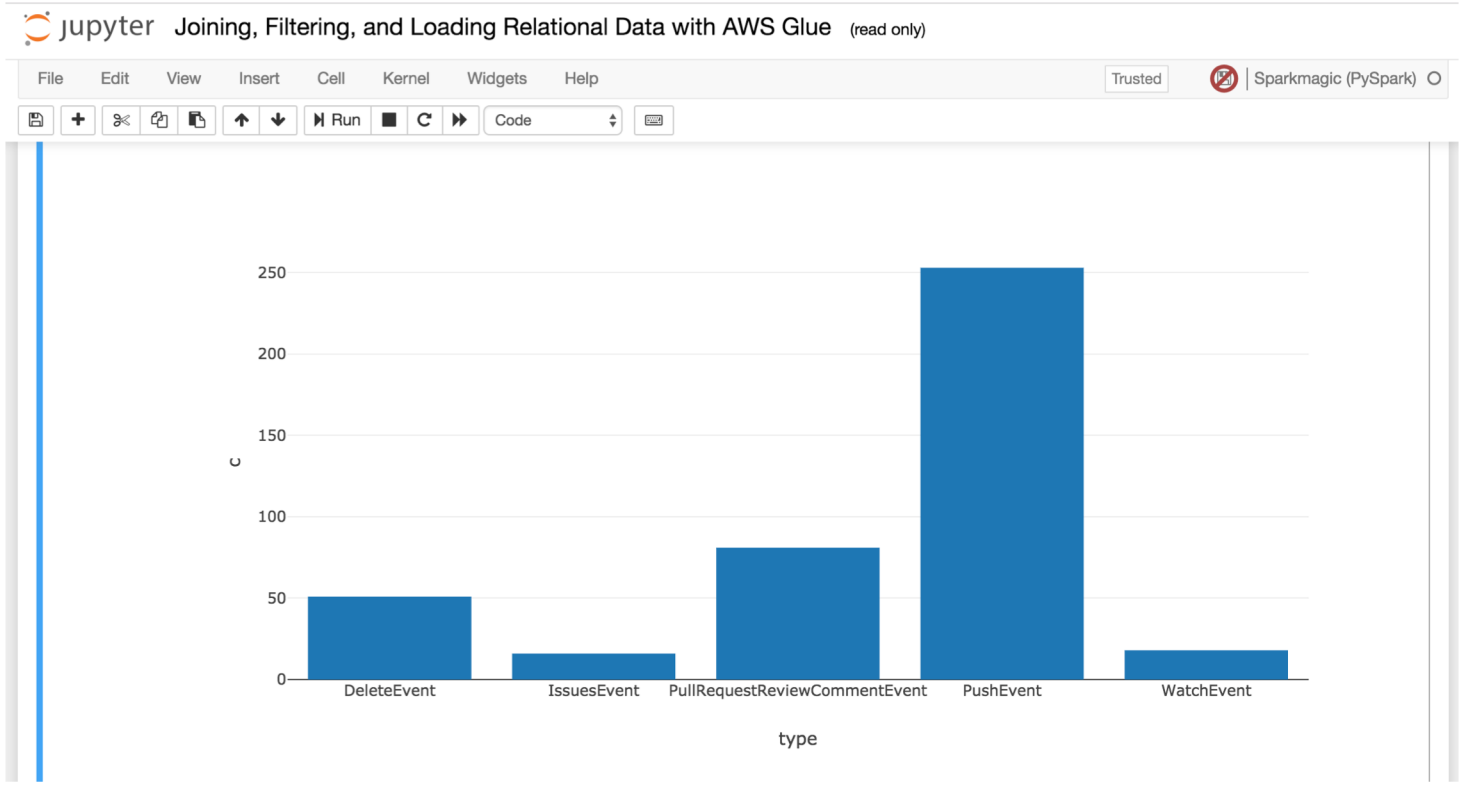
AWS Glue job metrics



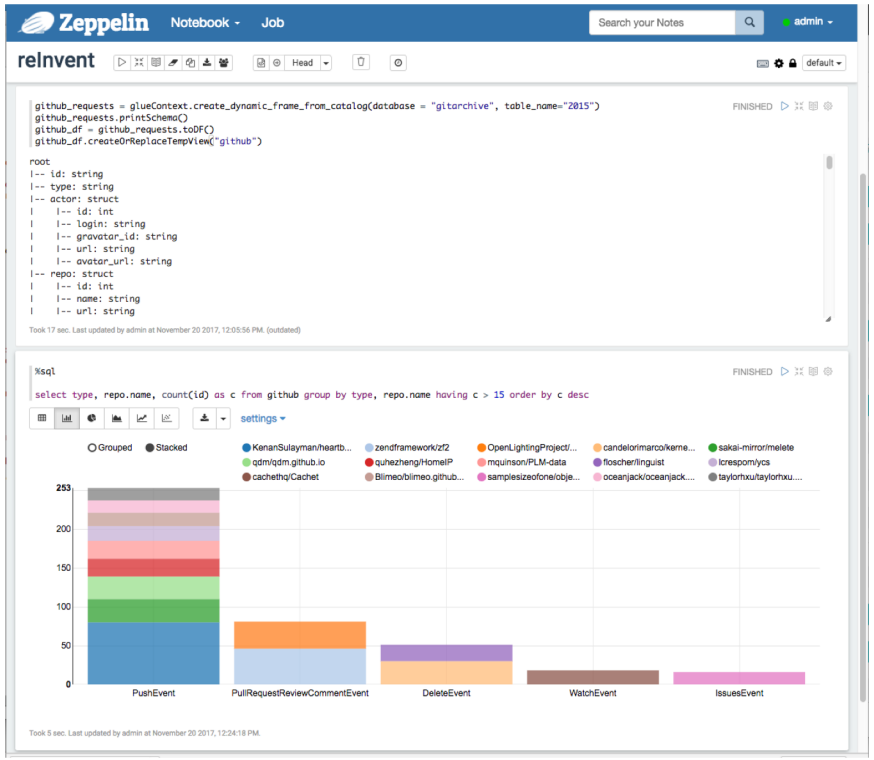
- Metrics can be enabled in the AWS Command Line Interface (AWS CLI) and AWS SDK by passing `--enable-metrics` as a job parameter key.

Beyond ETL: data science and exploration

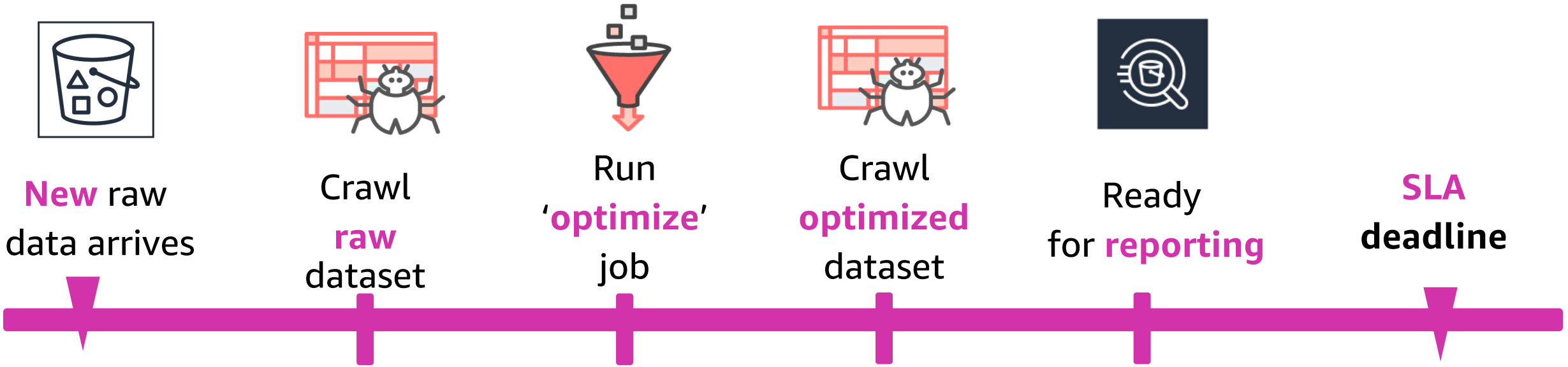
Amazon SageMaker integration



Apache Zeppelin notebook

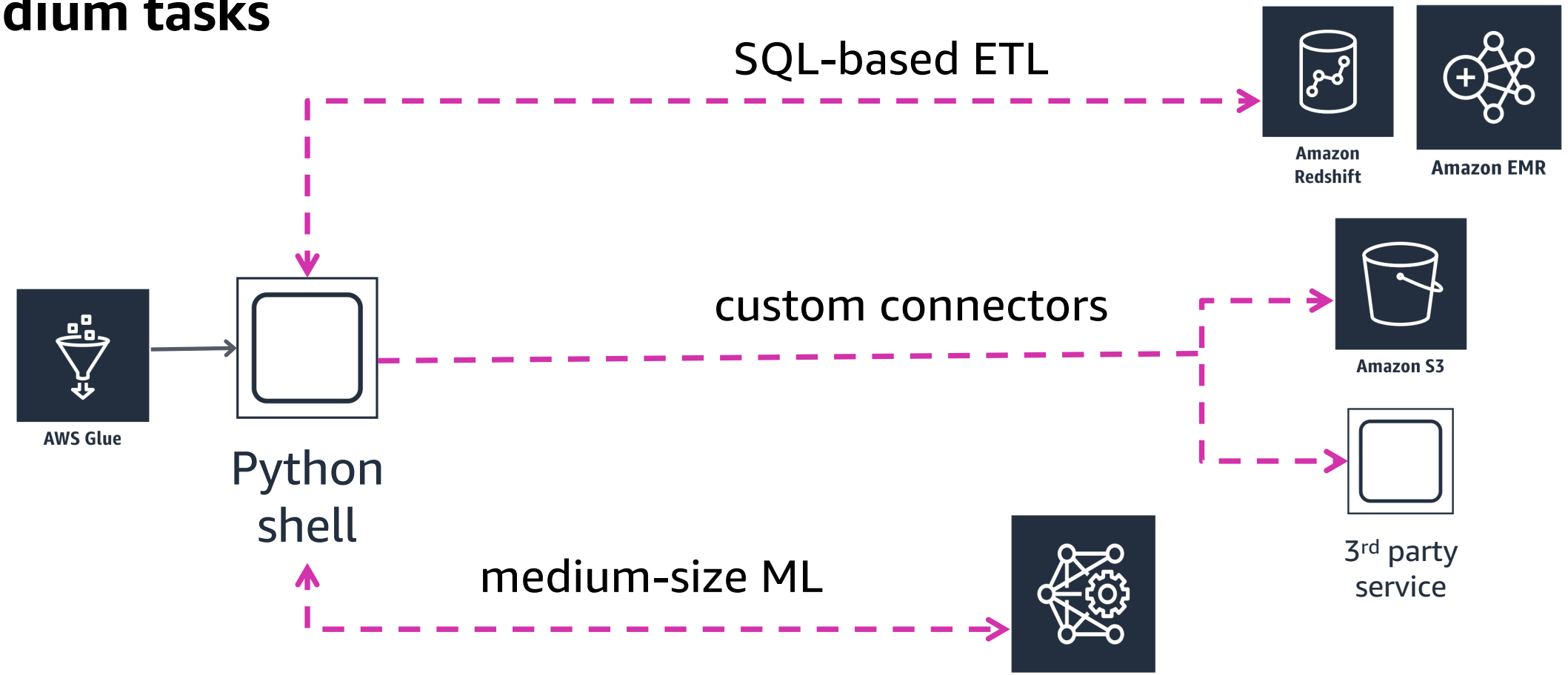


Example event-driven workflow



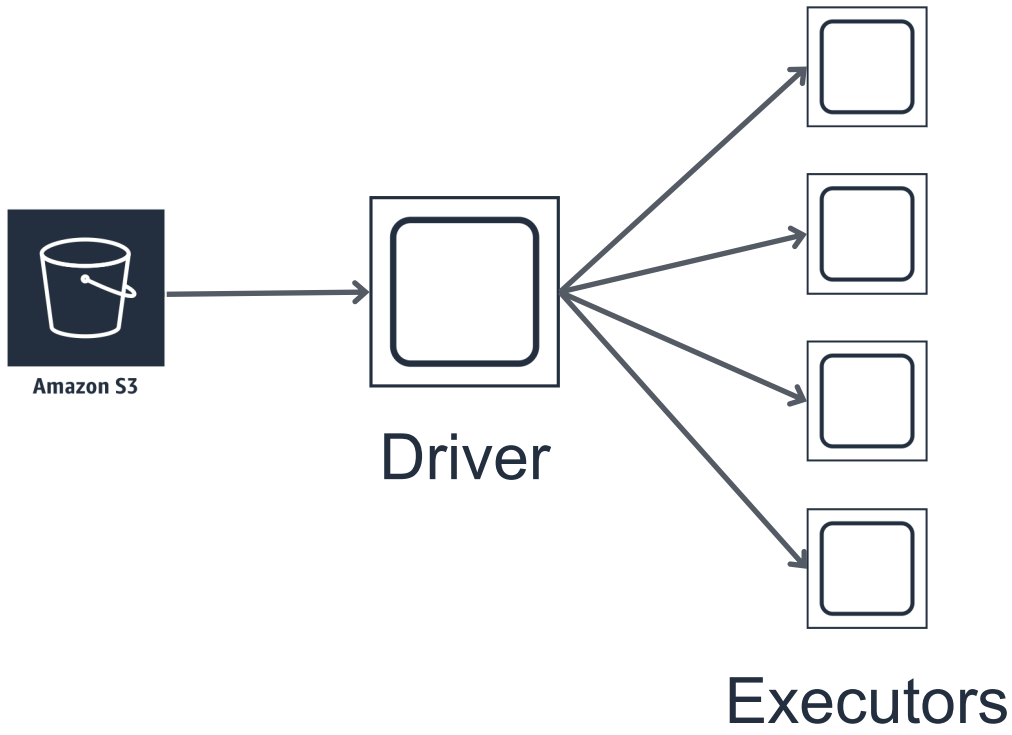
Announcing a new job type: Python shell

A new *cost-effective* ETL primitive for small to medium tasks



AWS Glue ETL

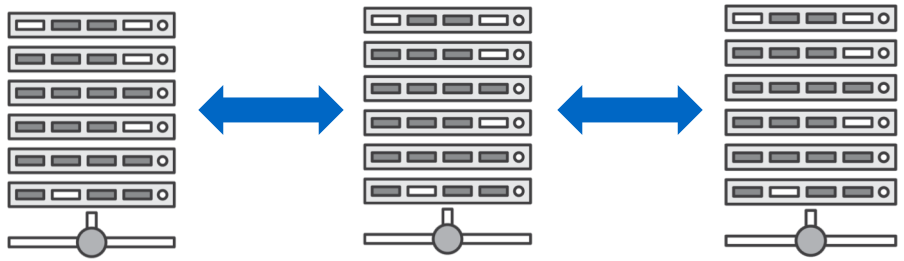
AWS Glue execution model: data partitions



- Apache Spark and AWS Glue are *data parallel*.
- Data is divided into *partitions* that are processed concurrently.
- 1 stage x 1 partition = 1 *task*

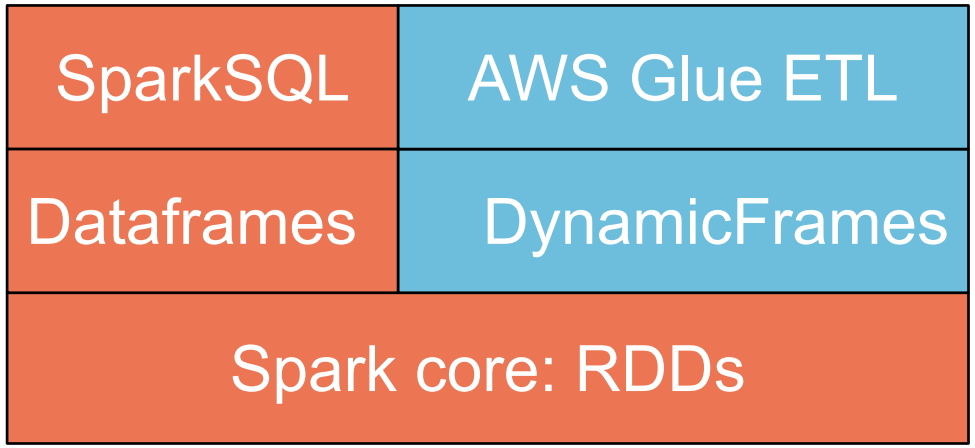
Overall throughput is limited by the number of partitions

Apache Spark and AWS Glue ETL



What is Apache Spark?

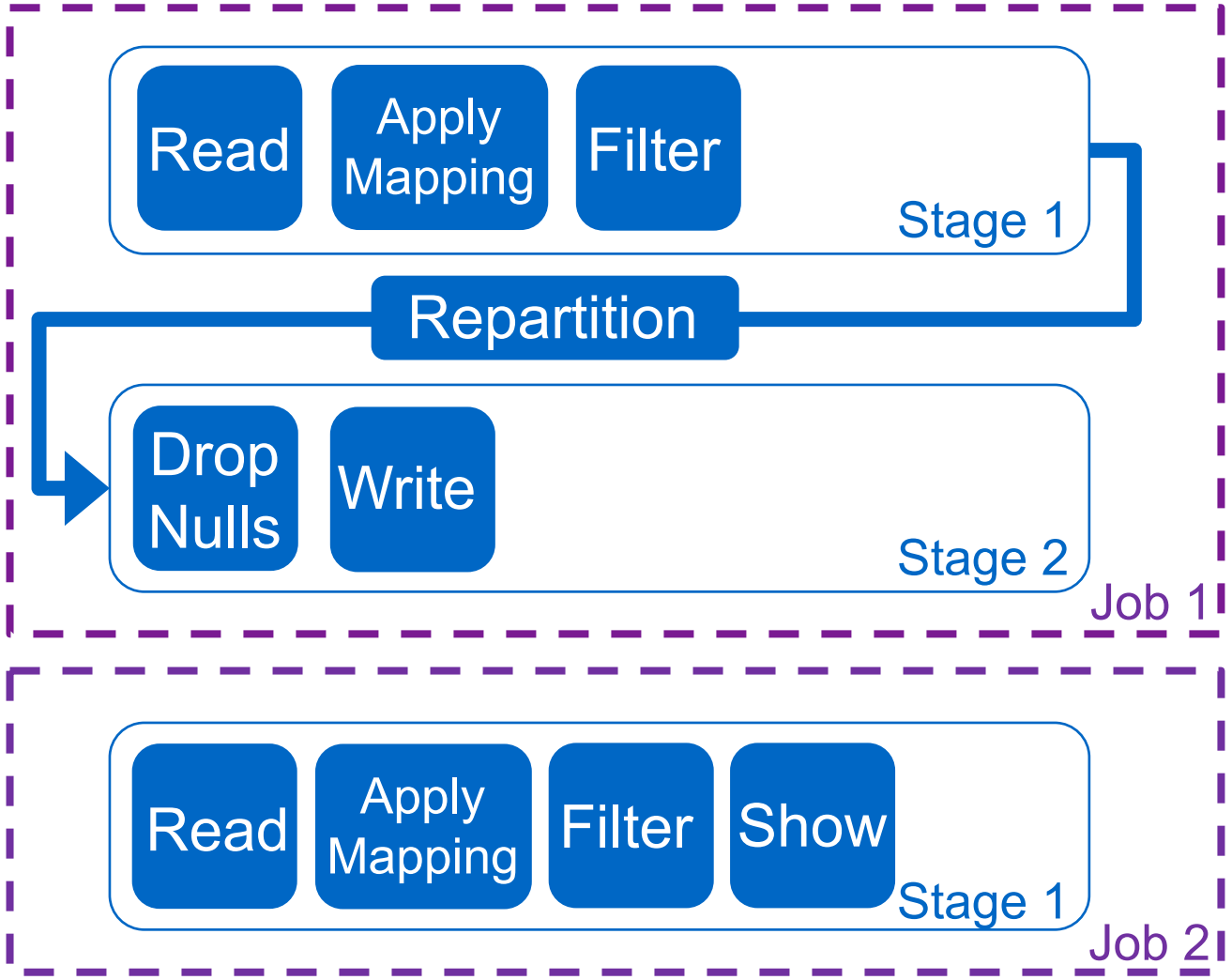
- Parallel, scale-out data processing engine
- Fault-tolerance built-in
- Flexible interface: Python scripting, SQL
- Rich eco-system: ML, Graph, analytics, …



AWS Glue ETL libraries

- Integration: Data Catalog, job orchestration, code-generation, job bookmarks, S3, RDS
- ETL transforms, more connectors & formats
- New data structure: DynamicFrames

AWS Glue execution model: jobs and stages

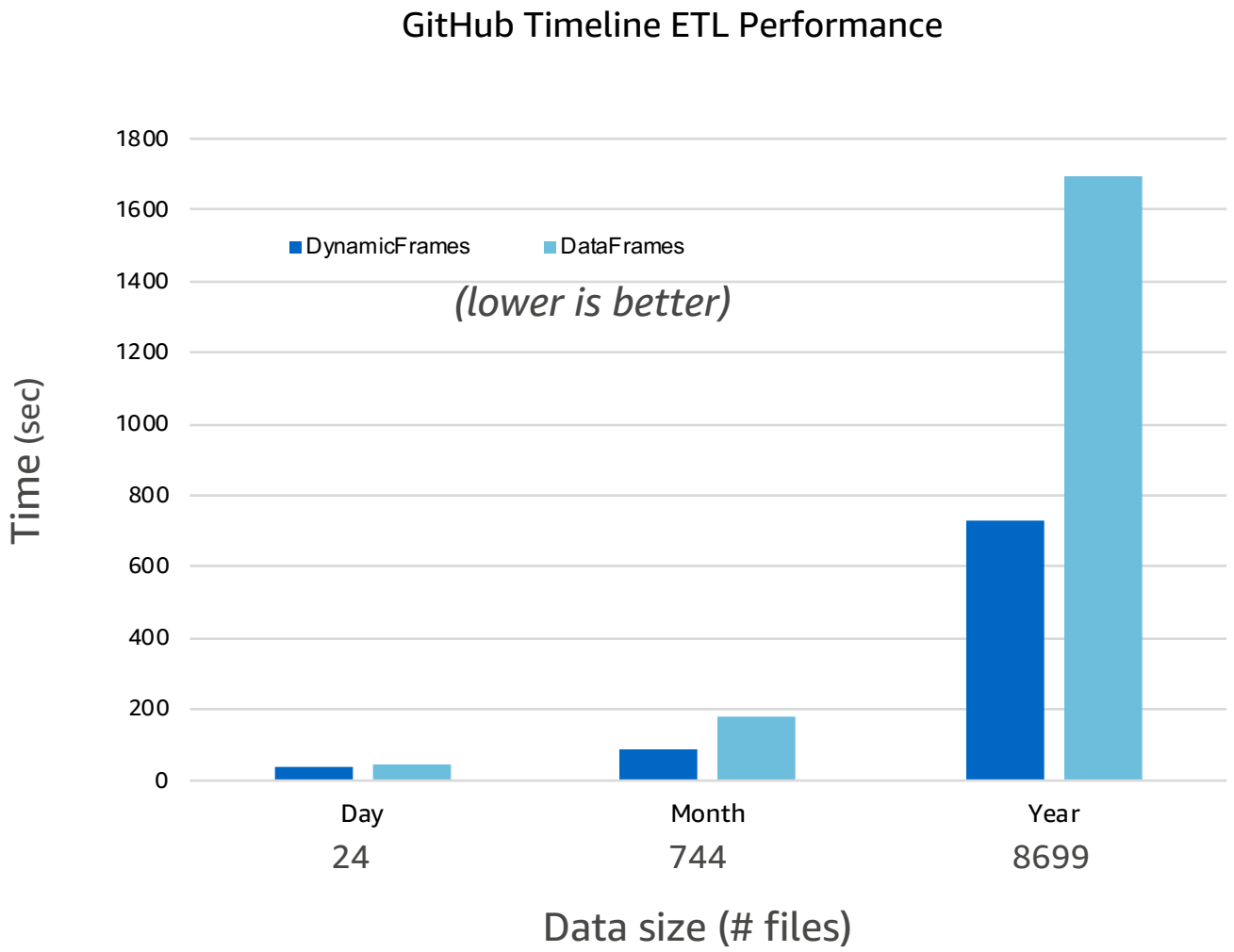



```
df = glueContext.getSource(…)

applyMapping = df.applyMapping(…)

filter = applyMapping.filter(…)

repartition = filter.repartition(10)

dropNulls = repartition.dropNulls()

glueContext.getSink(…)\
   .WriteDynamicFrame(dropNulls)
```

```
filter.show()
```

Jobs

Performance: AWS Glue ETL



Configuration

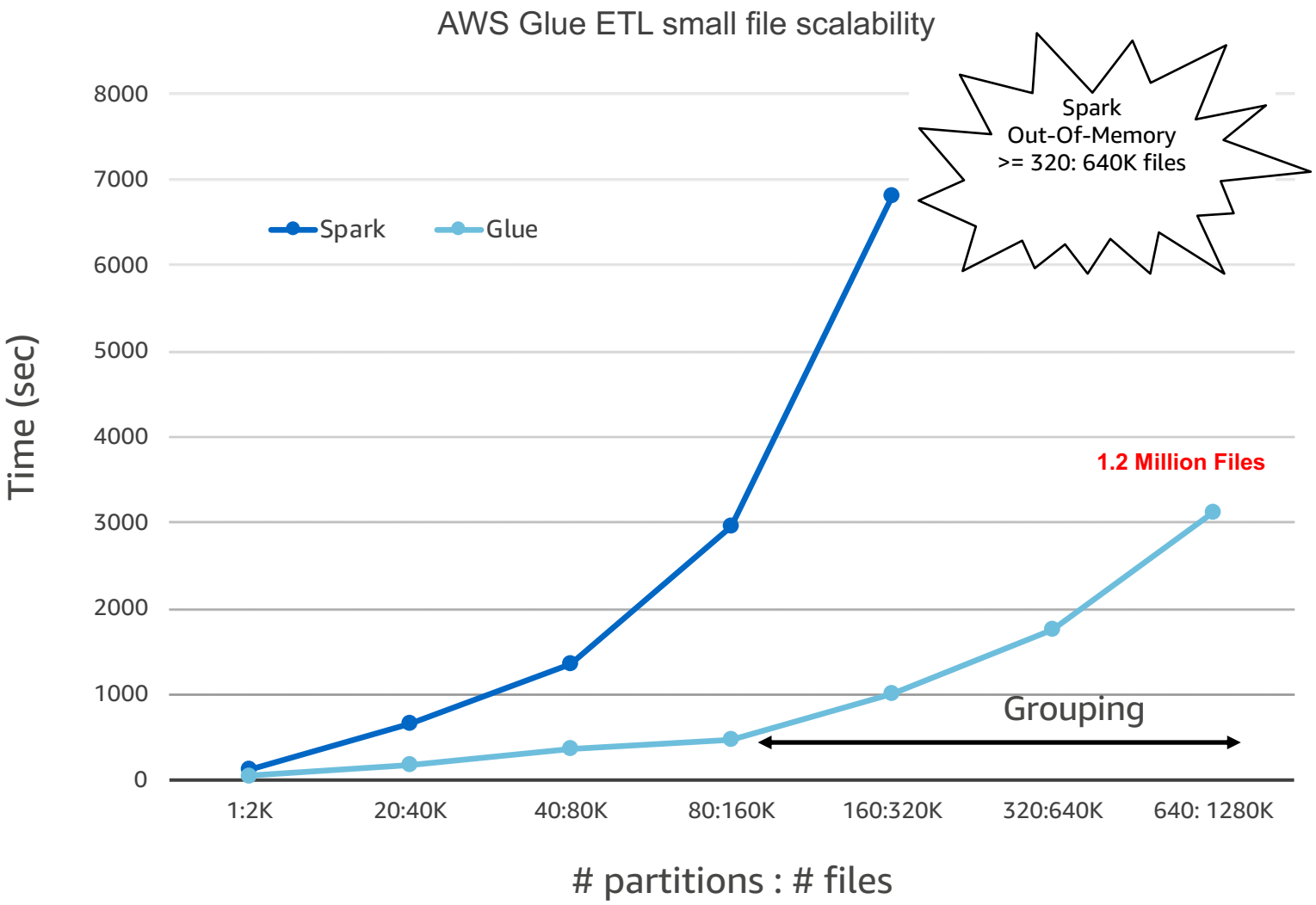
- 10 DPUs
- Apache Spark 2.1.1

Workload

- JSON to CSV
- Filter for Pull events

On average: 2x performance improvement

Performance: Lots of small files



Lots of small files, e.g. Kinesis Firehose

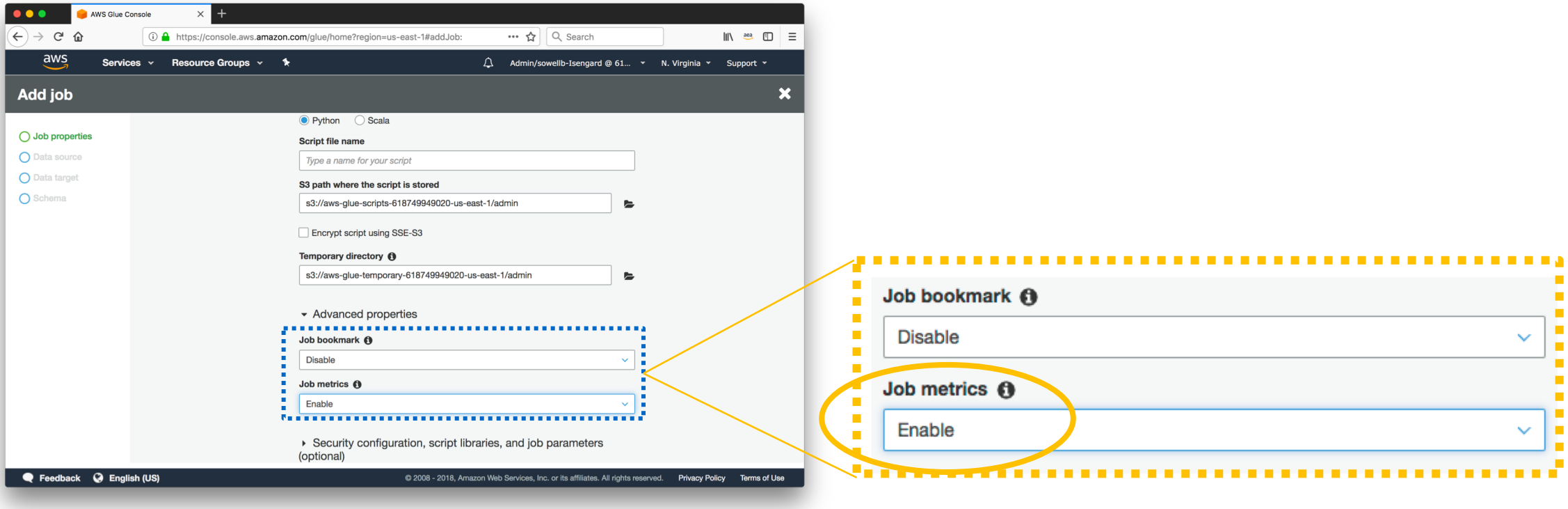
Vanilla Apache Spark (2.1.1) overheads

- Must reconstruct partitions (2-pass)
- Too many tasks: task per file
- Scheduling & memory overheads

AWS Glue Dynamic Frames

- Integration with Data Catalog
- Automatically group files per task
- Rely on crawler statistics

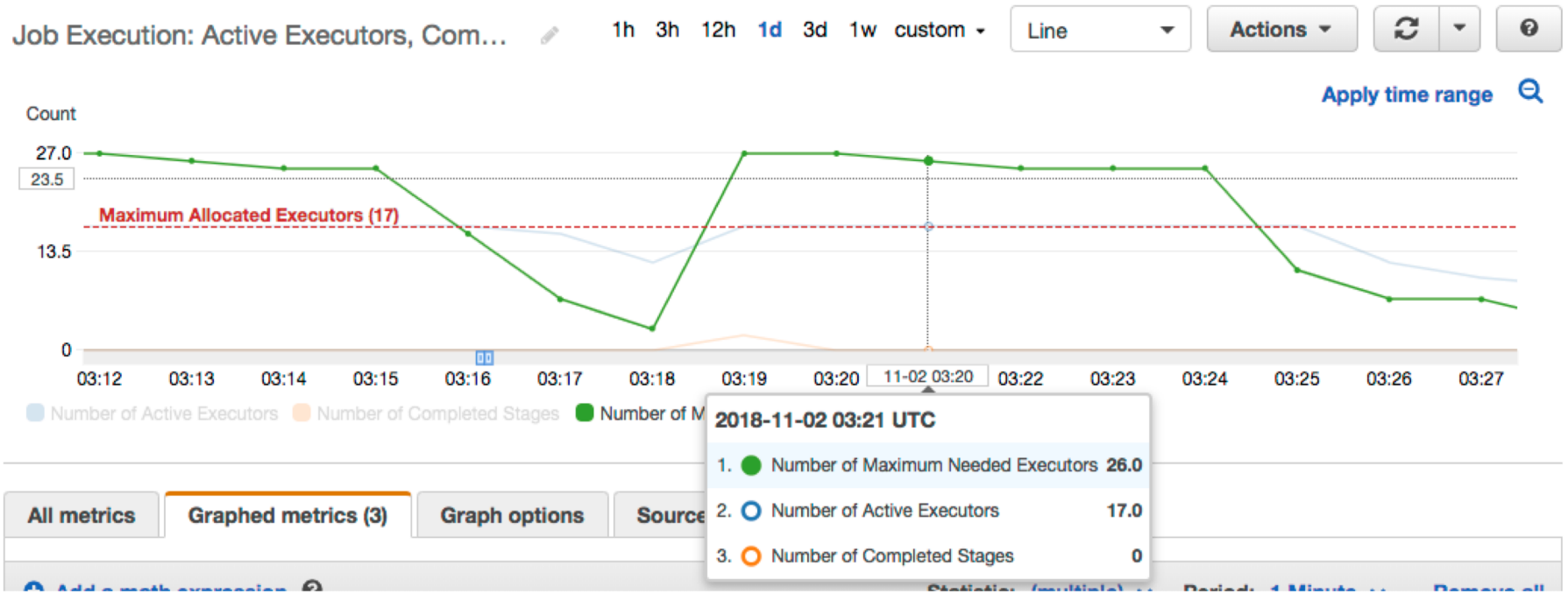
Enabling job metrics



- Metrics can be enabled in the CLI/SDK by passing `--enable-metrics` as a job parameter key.

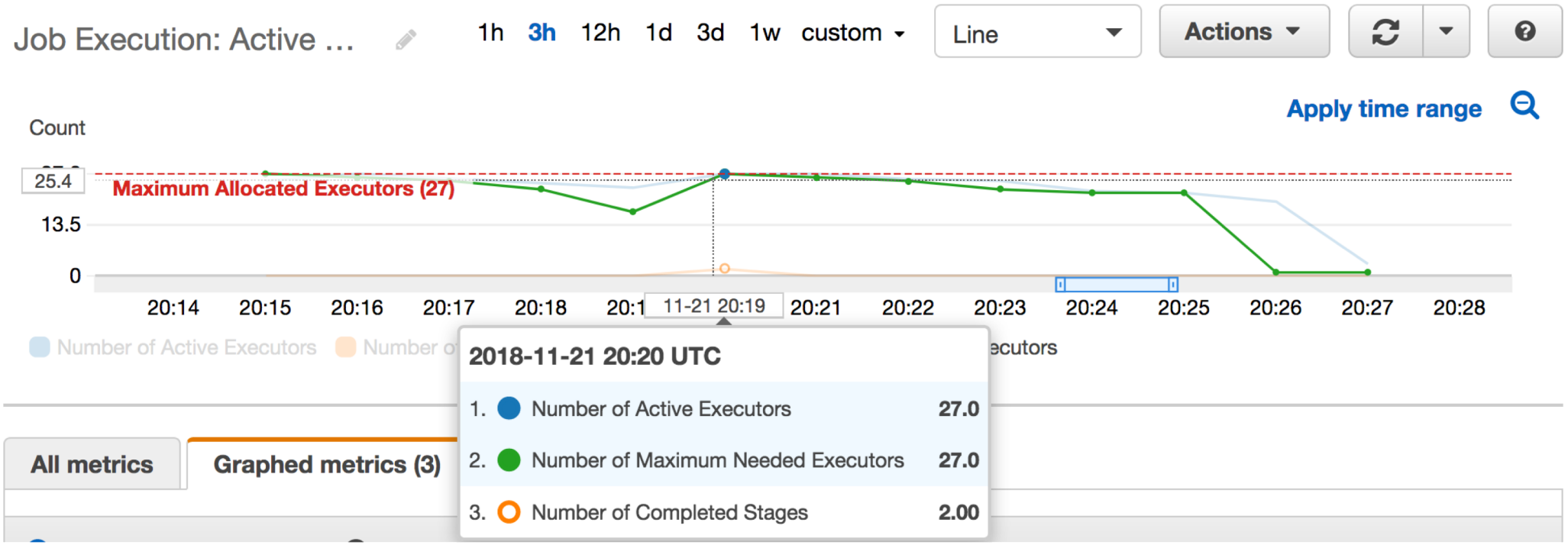
Example: optimizing parallelism

Processing large, split-able bzip2 files.
With 10 DPU, metric *maximum needed executors* shows room for scaling.



Example: optimizing parallelism

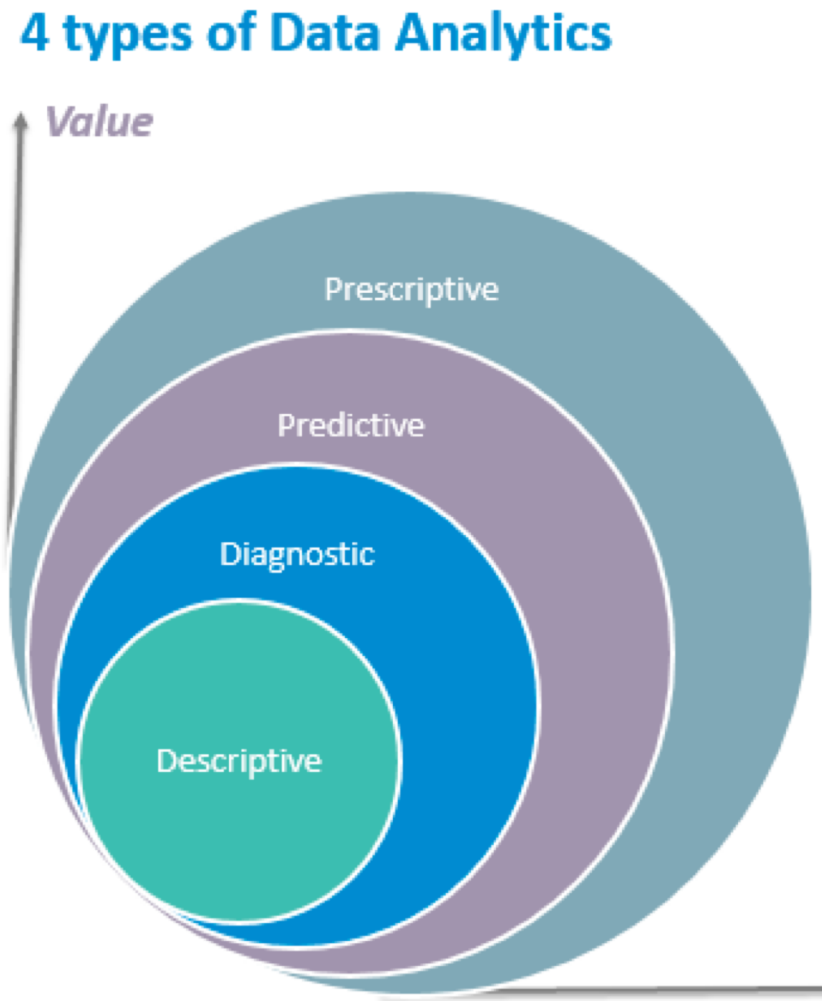
With 15 DPU, *active executors* closely tracks *maximum needed executors*.



Data Analytics

데이터 분석이란?

데이터 분석은 의사결정에 필요한 의미있는 정보와 근거들을 얻어내기 위해 데이터를 정리, 가공, 변환, 조사, 모델링 하는 과정을 뜻한다.(Wikipedia)



What is the data telling you?

Descriptive: ***What's happening in my business?***

- Comprehensive, accurate and live data
- Effective visualisation

Diagnostic: ***Why is it happening?***

- Ability to drill down to the root-cause
- Ability to isolate all confounding information

Predictive: ***What's likely to happen?***

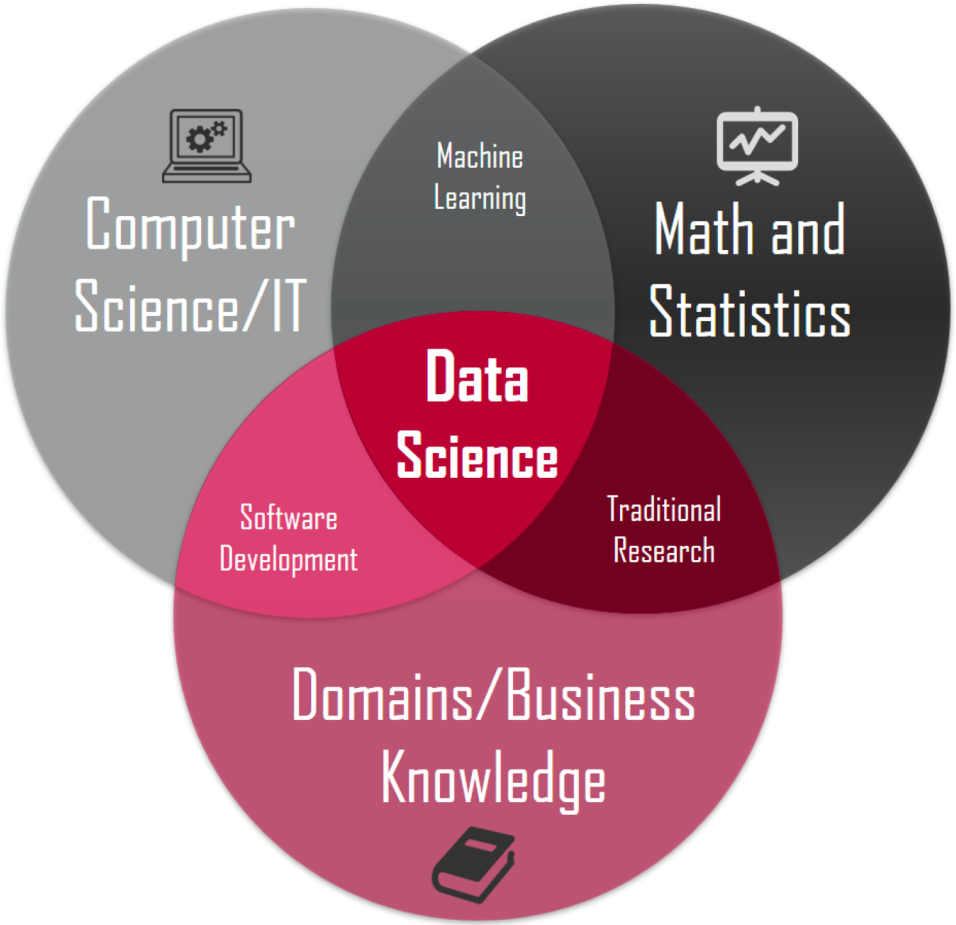
- Business strategies have remained fairly consistent over time
- Historical patterns being used to predict specific outcomes using algorithms
- Decisions are automated using algorithms and technology

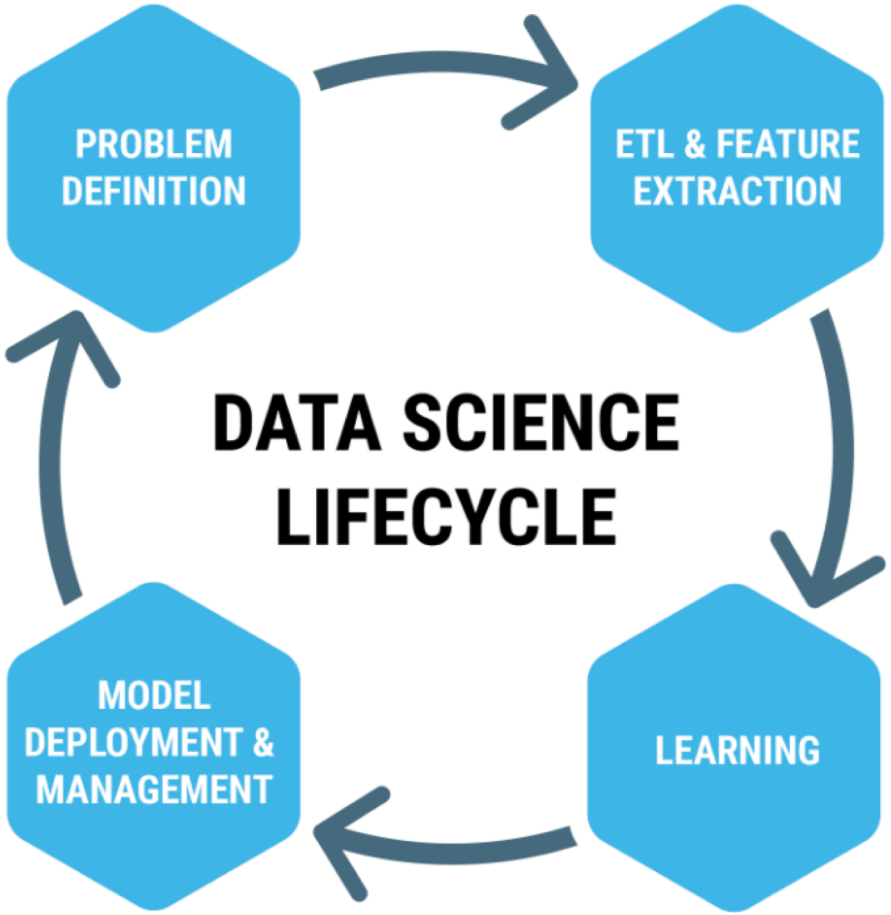
Prescriptive: ***What do I need to do?***

- Recommended actions and strategies based on champion / challenger testing strategy outcomes
- Applying advanced analytical techniques to make specific recommendations

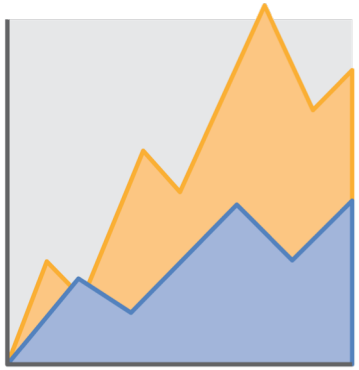
데이터 과학이란?

데이터 과학은 다양한 정형, 비정형 데이터에서 지식과 통찰력을 뽑아내는 행위이고, 통계, 데이터 마이닝, 예측 분석등으로 이어지는 흐름이다.
(Wikipedia)



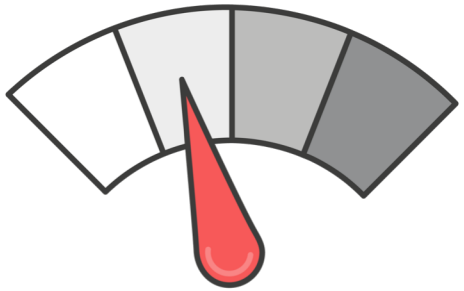


Enabling all Types of Data-Driven Analytics



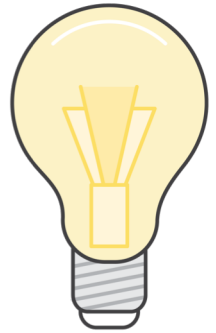
Retrospective
analysis and
reporting

Past Data



Here-and-now
real-time processing
and dashboards

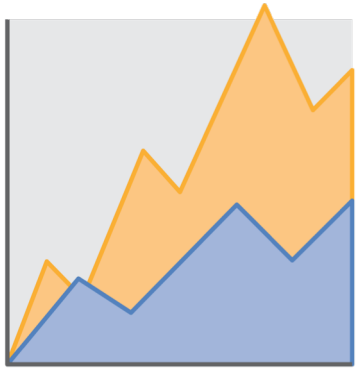
Present Data



Predictions
to enable smart
applications

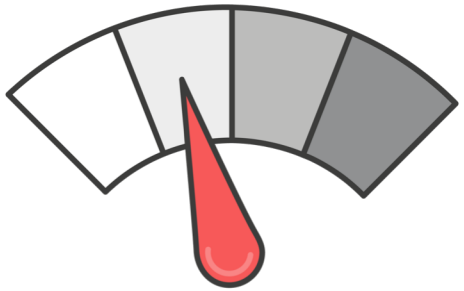
"Future Data"

Enabling all Types of Data-Driven Analytics



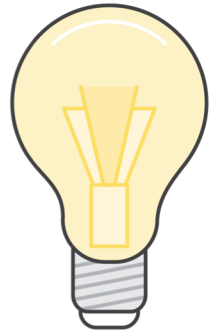
Retrospective
analysis and
reporting

Amazon Redshift
Amazon RDS
Amazon S3
Amazon EMR



Here-and-now
real-time processing
and dashboards

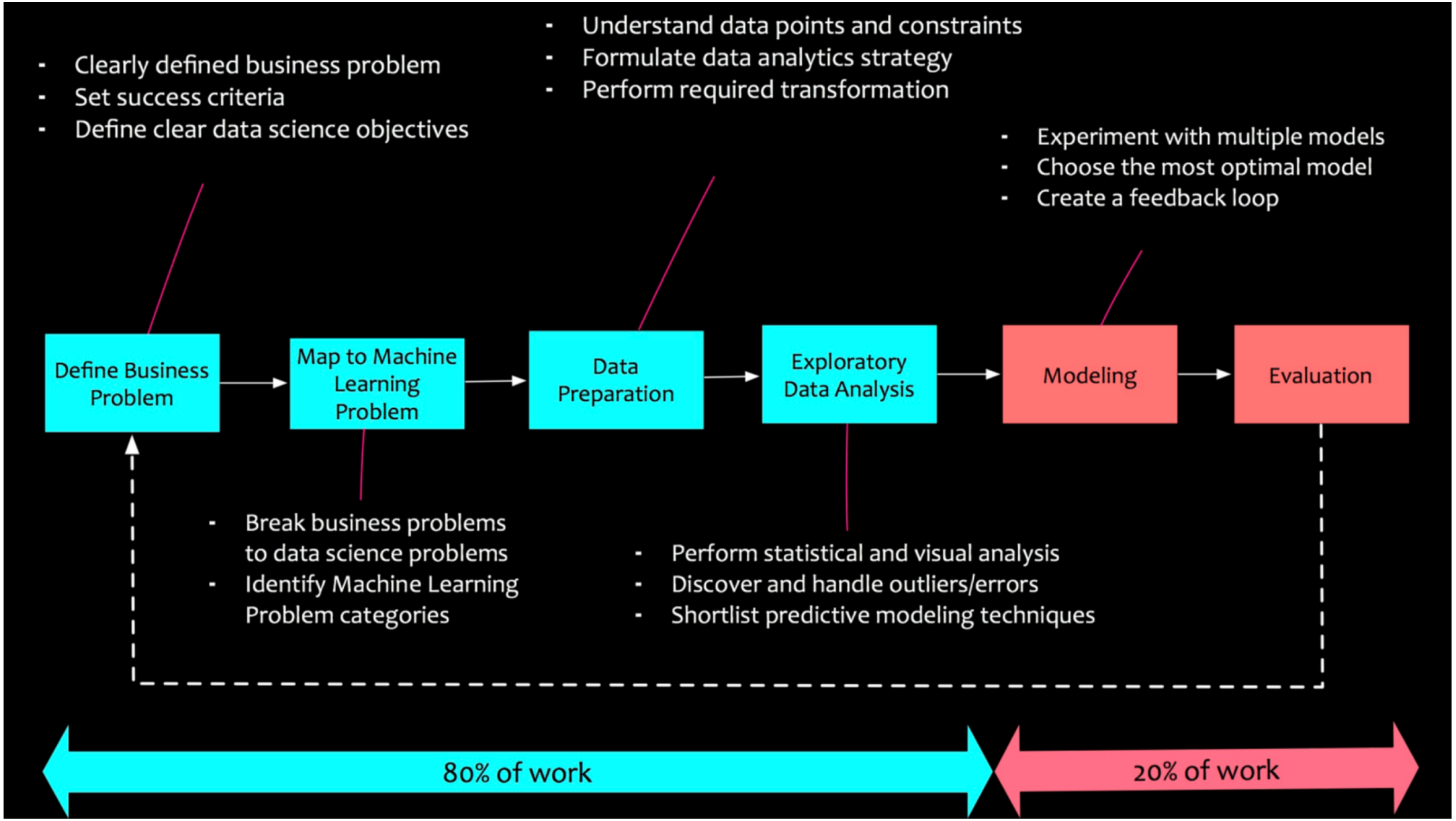
Amazon Kinesis
AWS Lambda
Amazon DynamoDB
Amazon EC2



Predictions
to enable smart
applications

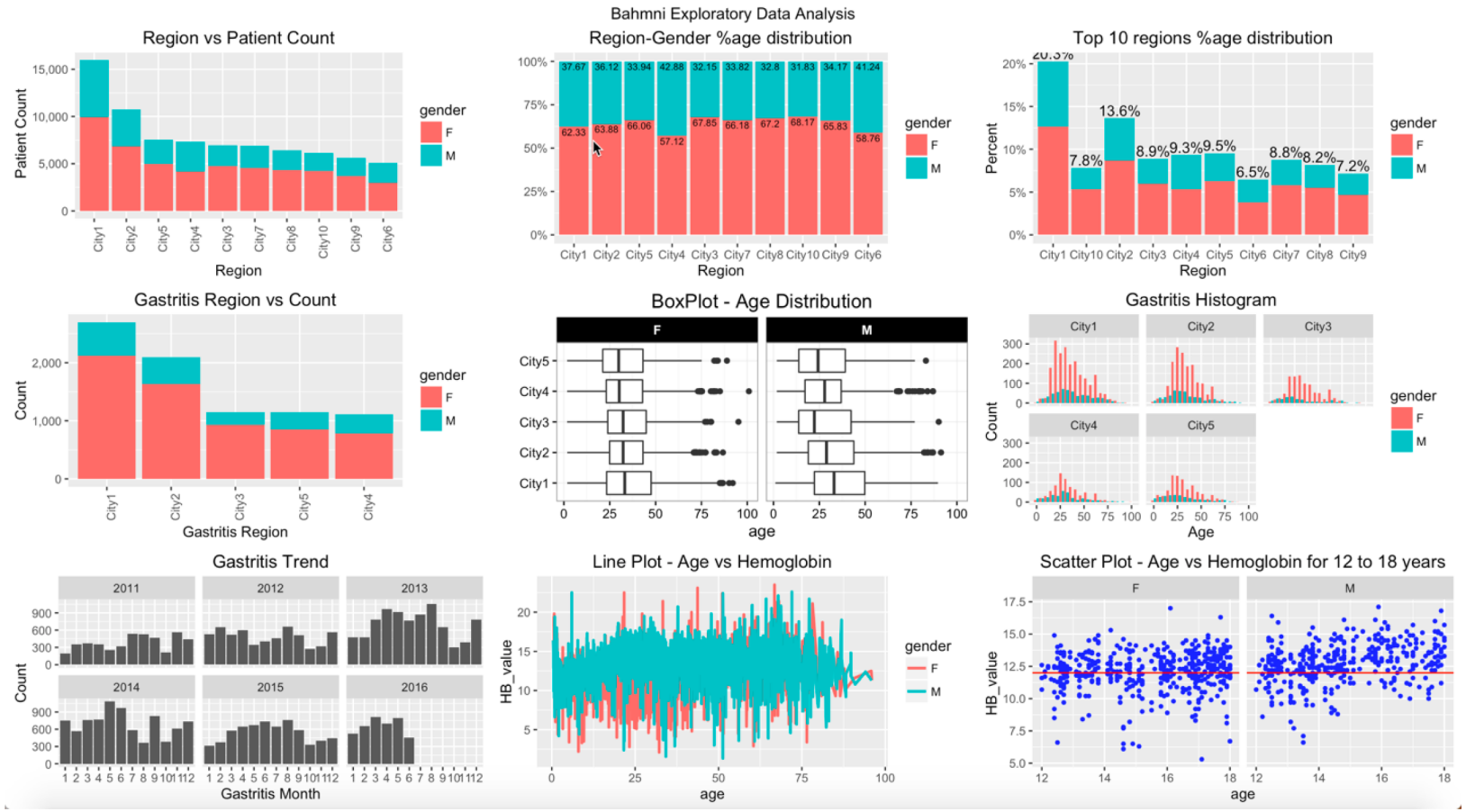
Amazon ML
Amazon Sagemaker

Data Science Process



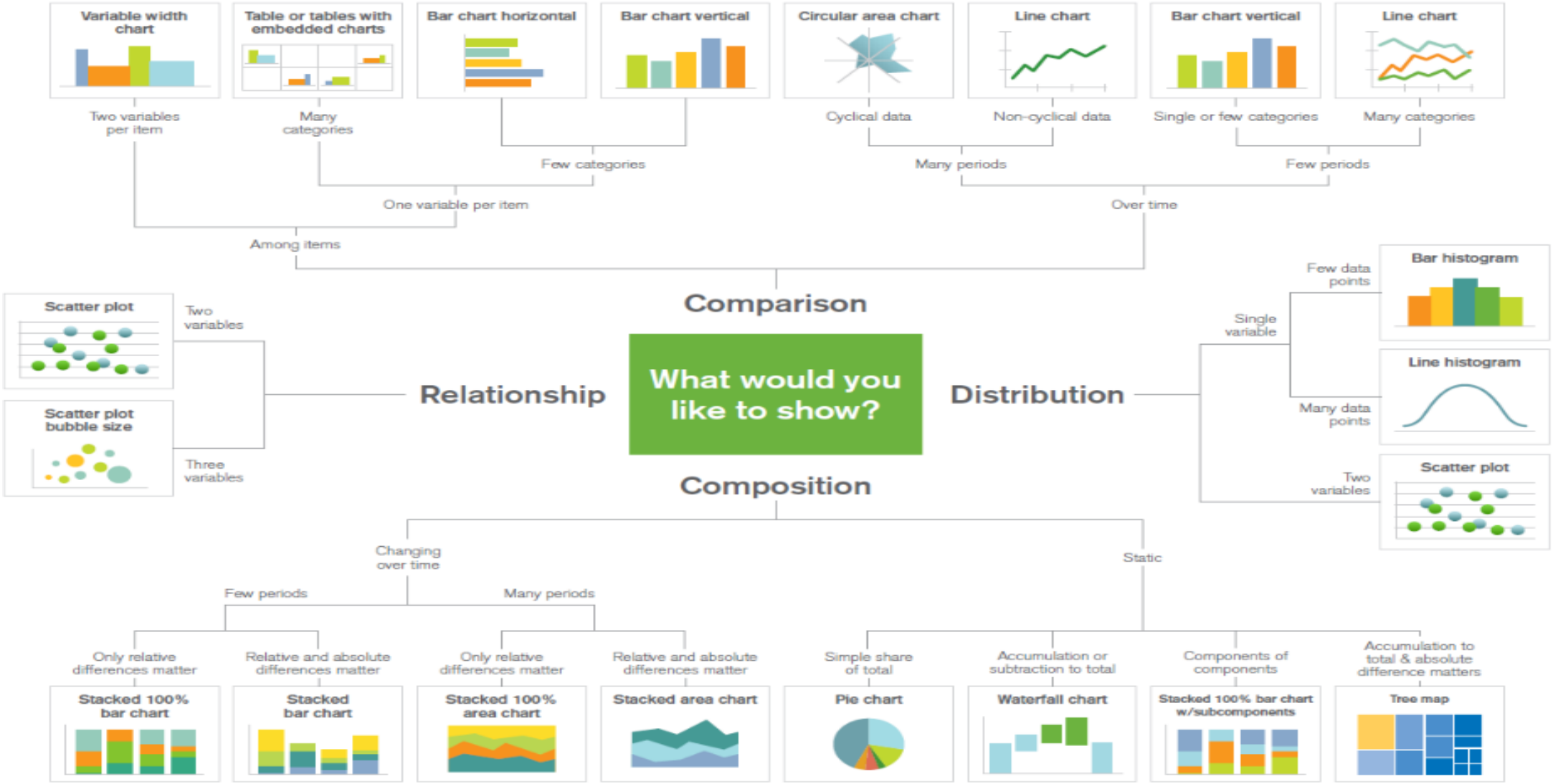
EDA(Exploratory data analysis)

EDA(탐험적 데이터 분석)은 데이터 분석의 초기 단계에 이해하기 위해 다양한 차트를 이용하여 데이터를 표현하고 이해하는 과정이다. 이를 통해 문제 해결에 필요한 데이터를 식별하고, 가설 수립, 알고리즘 선정 등을 할 수 있다.

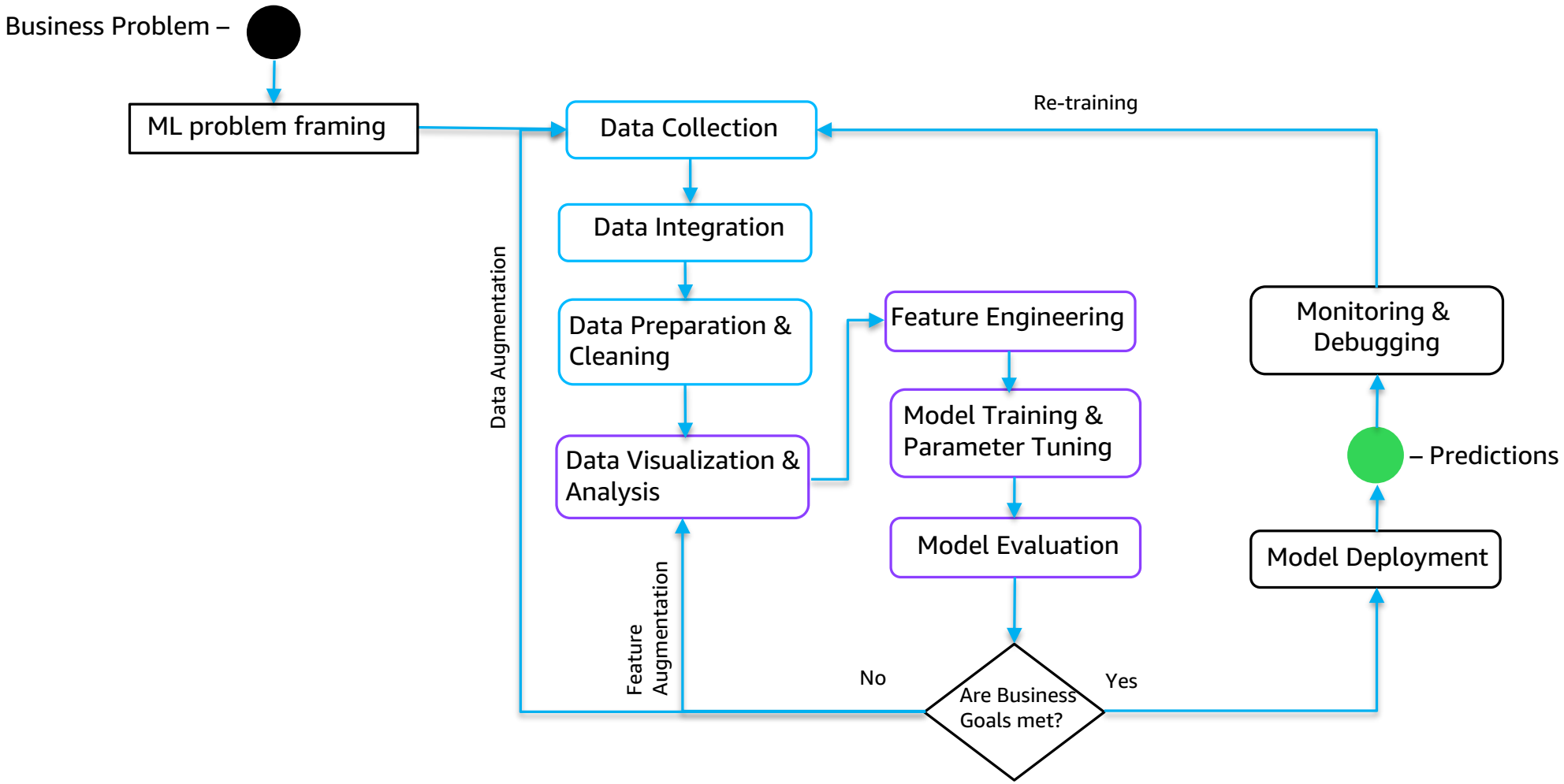


데이터 시각화 방법

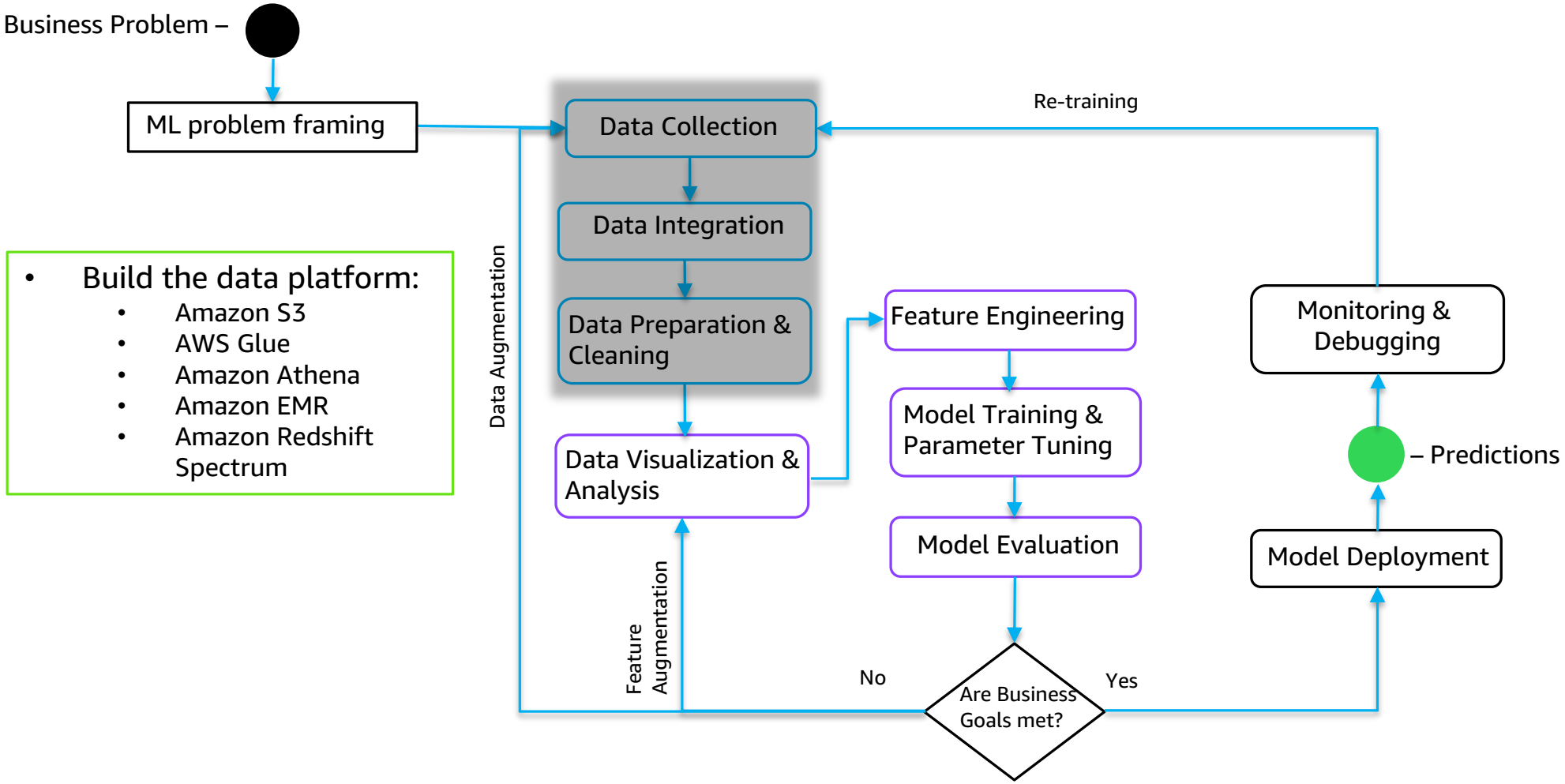
분포, 분류, 관계, 비교, 시간, 공간



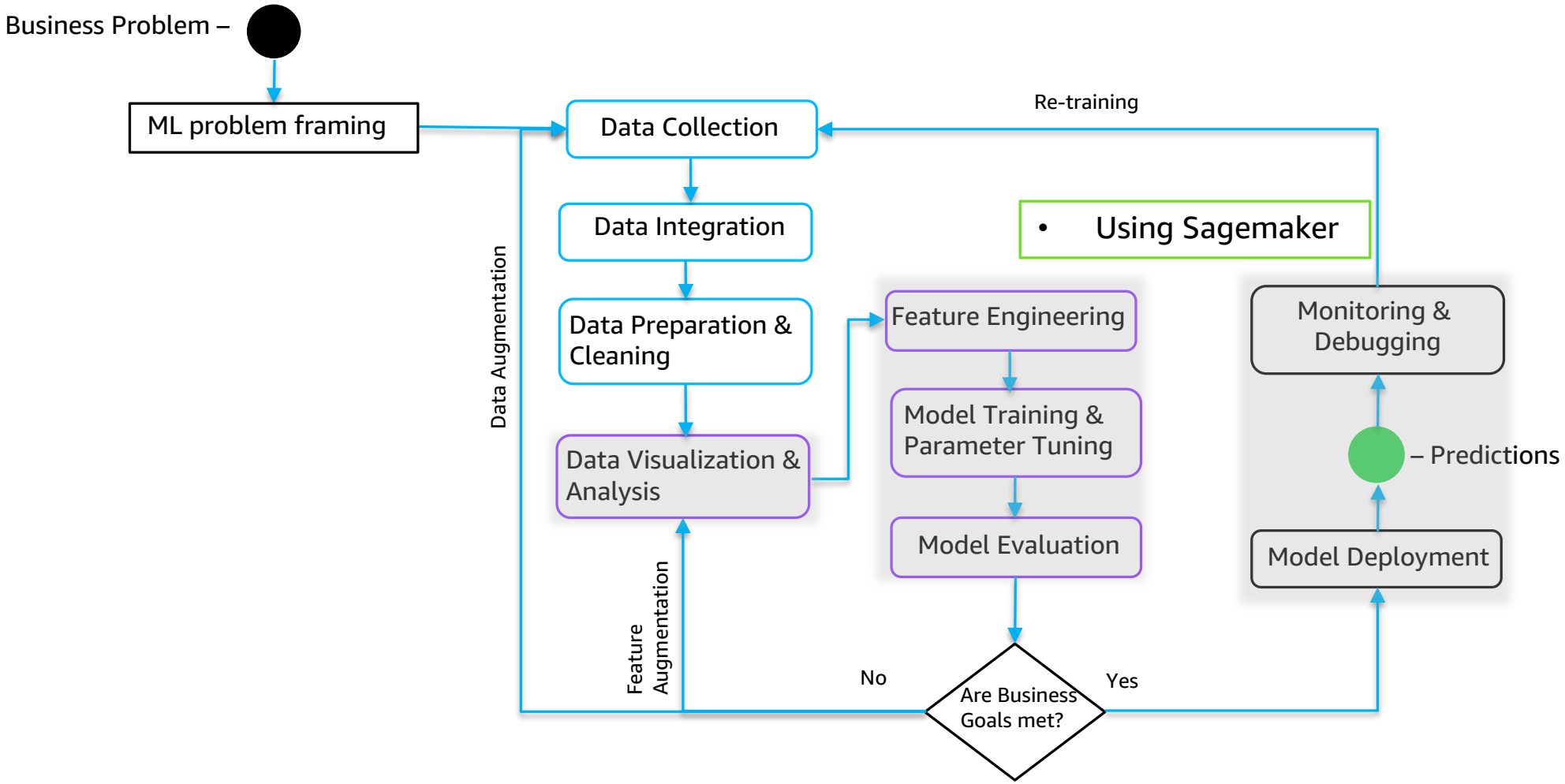
Machine Learning Process



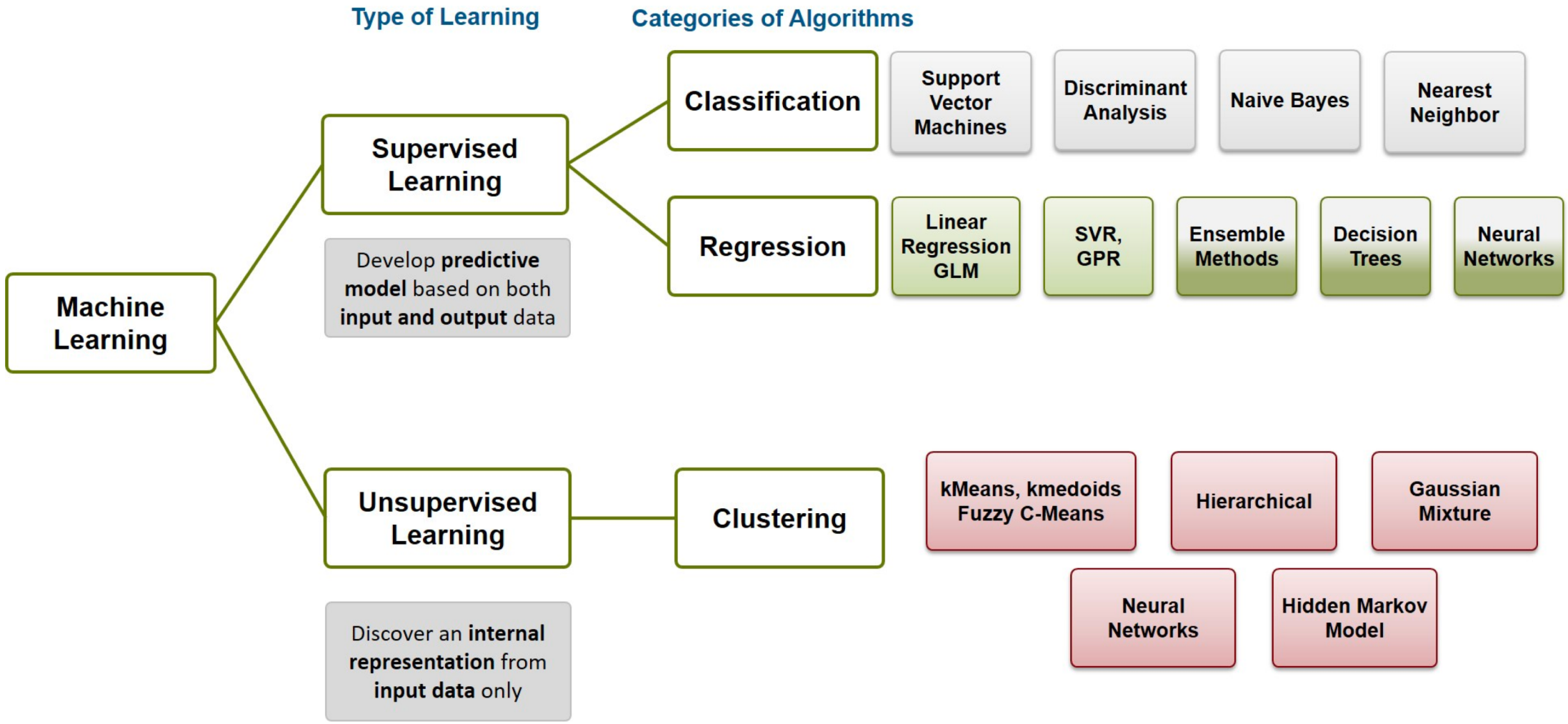
Machine Learning Process



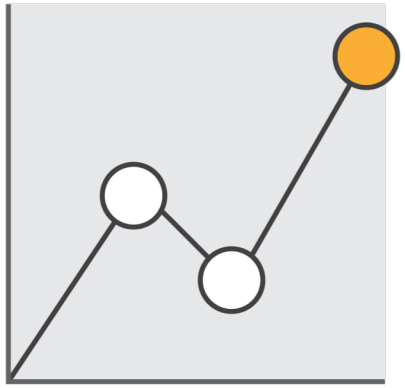
Machine Learning Process



Data Analysis Algorithms

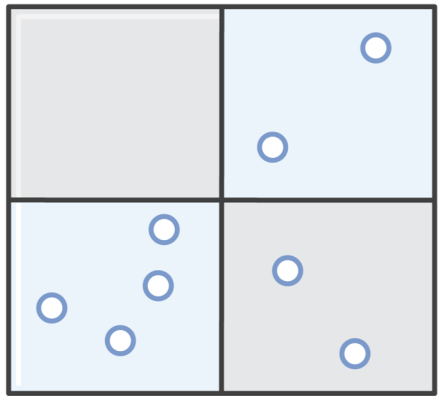


Amazon SageMaker Built-in ML Algorithm



Classification, Regression

- Linear Learner
- XGBoost
- Factorization Machines
- DeepAR



클러스터링, 차원축소

- K-Means
- PCA

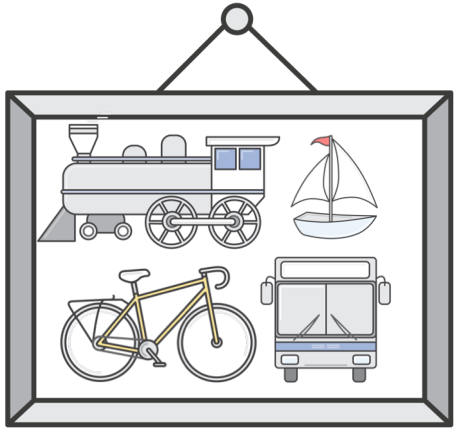
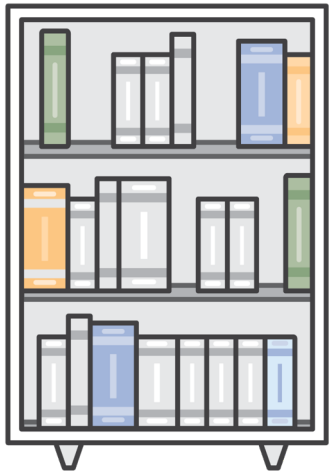


Image Classfication

- Image Classifier (Resnet)



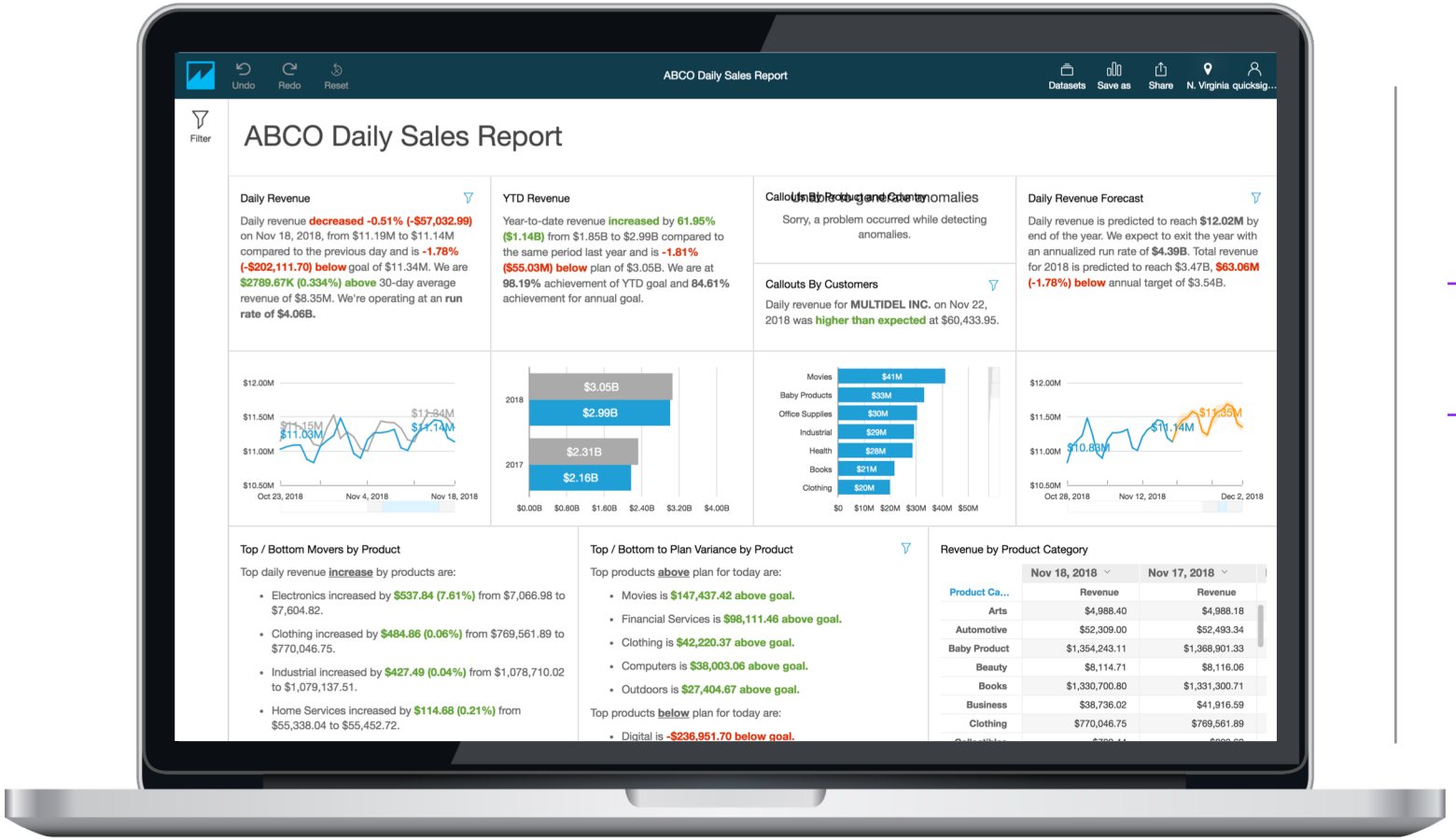
Natural language processing

- LDA, NTM (Topic modeling)
- Seq2Seq (Translation)
- BlazingText (Word2Vec)

Amazon QuickSight—ML Insights (Preview)

Automated business insights powered by ML and natural language

NEW!



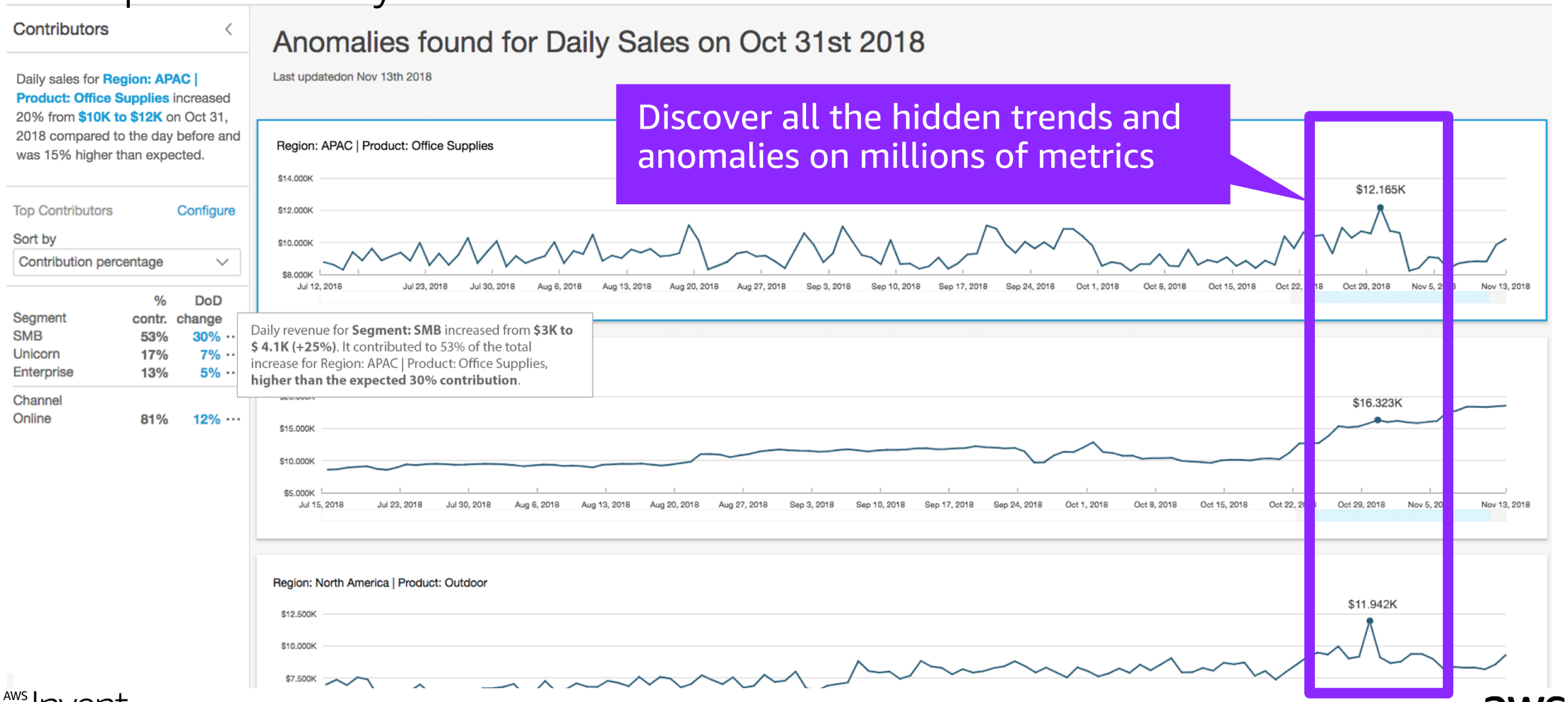
- ML-powered anomaly detection
- ML-powered forecasting
- Auto-narratives

AWS re:Invent

aws

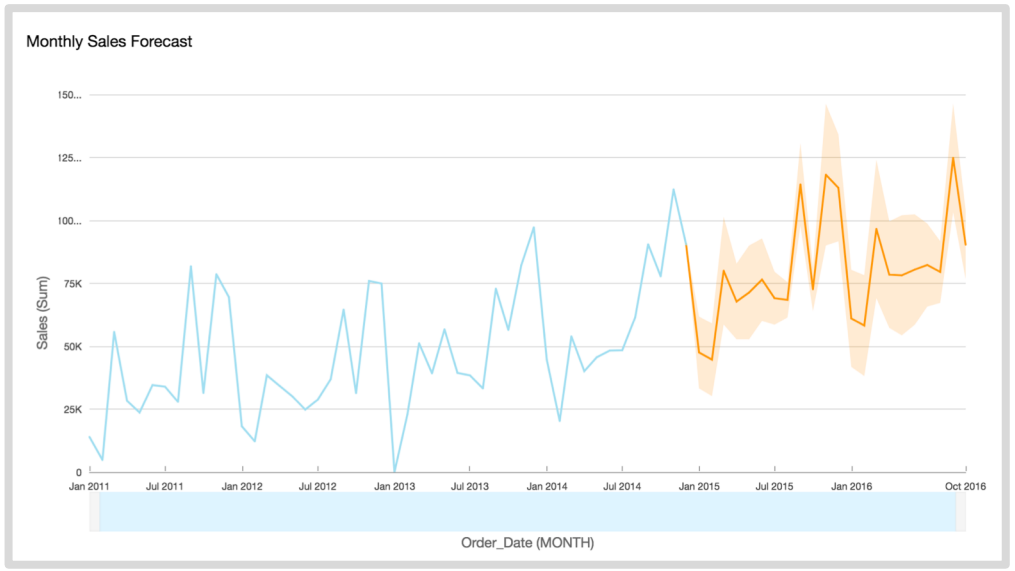
Amazon QuickSight—ML Insights

Example: anomaly detection



QuickSight ML Insights vs. traditional BI forecasting

QuickSight ML-powered forecasting



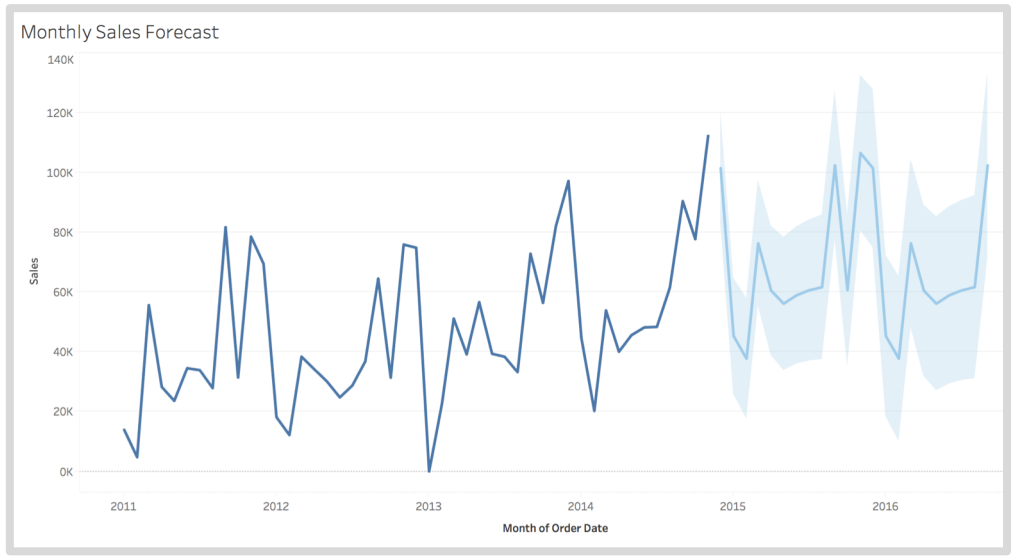
Captures seasonality and upward trends

Automatically excludes bad data

High confidence band

vs.

Traditional BI forecasting

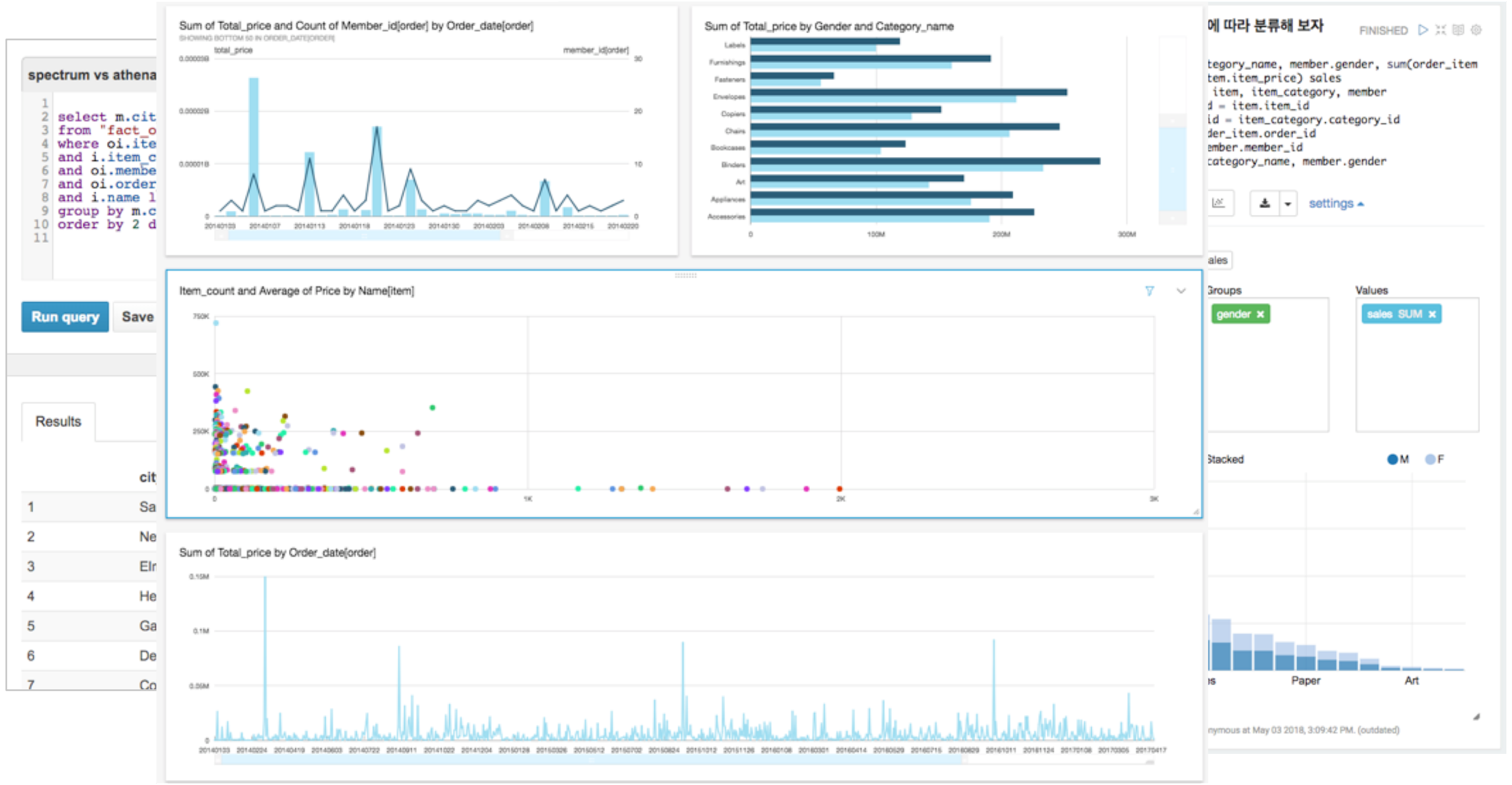


Captures only seasonality

Missing upward trend

Confidence band influenced by bad data

데이터 분석의 결과물 공유



데이터 분석의 결과물 공유