



Glue, Athena, Quicksight를 통한 데이터 분석

데이터 레이크를 중심으로

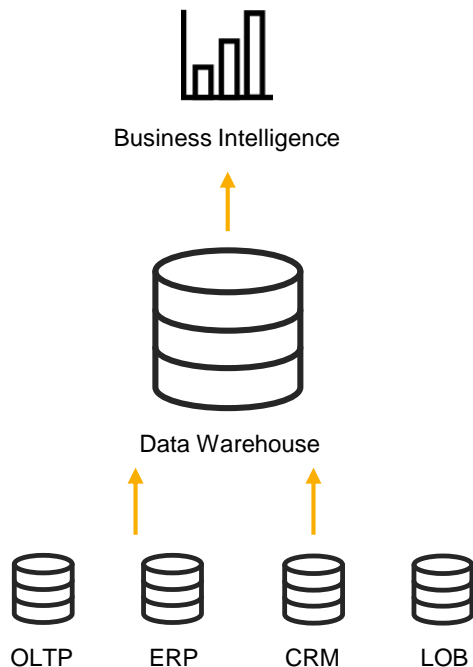
김준형, 솔루션즈 아키텍트



Agenda

- 빅데이터 활용의 장애물과 도전과제
- AWS의 데이터 레이크
- AWS Glue / Amazon Athena / Amazon QuickSight

전통적인 데이터 분석 방식



관계형 데이터베이스에 기반하고

테라바이트에서 페타바이트 규모로 확장

데이터 로딩 전 미리 스키마를 정의 (Schema on Write)

정기적인 리포트의 생성과 간단한 Ad-hoc 분석

대규모의 비용 선투자 + \$10K-\$50K / TB / Year

데이터 팀에 주어진 도전과제들

기하급수적으로 늘어나는 데이터



Transactions



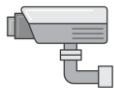
Billing



ERP



Web logs



Sensor Data



Infrastructure logs



Social

Dark Data
복잡한 전처리

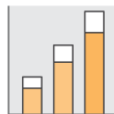
다양한 데이터 소비자들



Data Scientists



Applications



Business Analyst



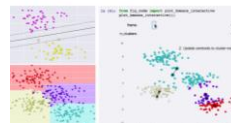
External Consumers

데이터의 중복
원본데이터 관리

많은 접근 방식과 툴들



API Access



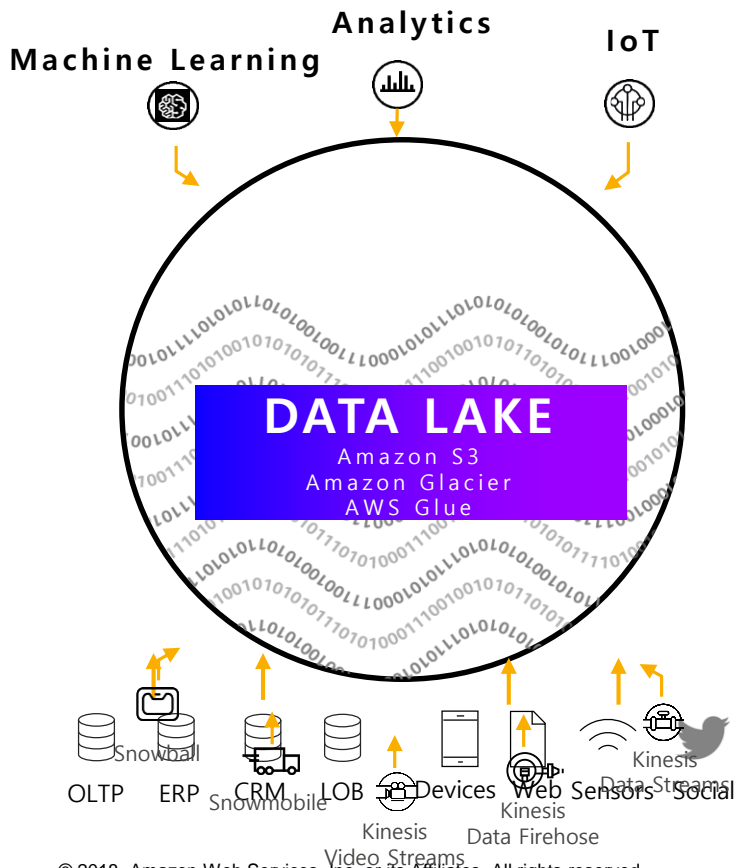
Notebooks



BI Tools

다양한 기술 지원
전문가의 부족

발전된 분석 시스템 - 데이터 레이크



엑사바이트 이상의 규모에 탁월한 내구성과 가용성

우수한 보안, 컴플라이언스, 감사 기능

관계형 데이터와 비정형 데이터를 모두 저장

활용하는 시점에 스키마의 확정 (Schema on Read)

인사이트를 얻기위한 다양한 분석 엔진을 사용

연동 가능한 다양한 데이터 처리 / 분석 에코 시스템

낮은 비용의 스토리지, 적은 분석 비용

데이터 레이크 – 모든 데이터가 한곳에

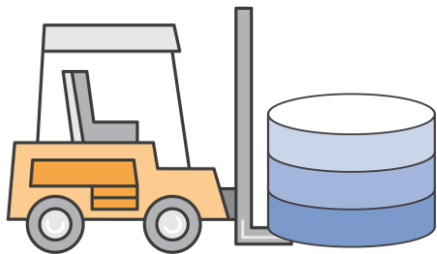


“왜 데이터가 여러 장소에
분산되어 있는가?
어떤 데이터가 정말
원본 데이터 인가?”

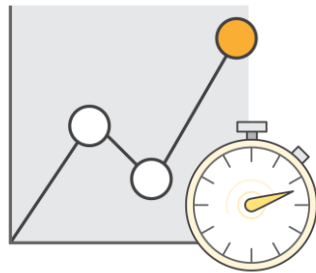


하나의 중앙 저장소에
모든 소스로부터 오는 모든 종류의
데이터를 저장하고 분석

데이터 레이크 – 빠른 데이터 수집

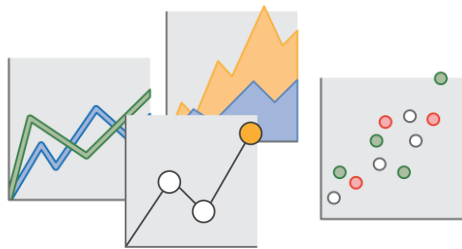


“어떻게 다양한 소스로부터의
데이터를 빠르게 수집하여
효율적으로 저장할 수
있을까?”

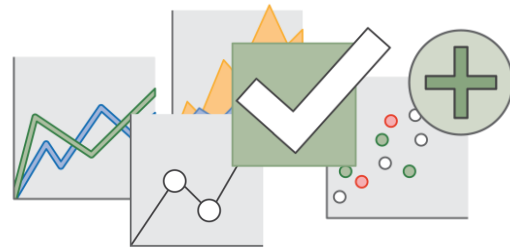


실시간, 배치, IoT등 다양한 수집
도구 활용
별도의 스키마 정의 없이도
빠르게 데이터를 수집

데이터 레이크 – 사용 시점에 스키마 정의

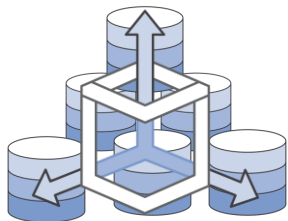


“여러 종류의 분석툴과 프로세싱 엔진에서 같은 데이터를 같이 사용할 수 있는 방법이 있는가?”

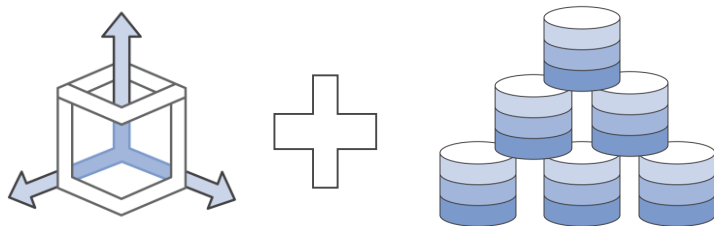


데이터를 저장 시점이 아닌 사용하는 시점에 정의해서 사용함으로써 언제나 Ad-hoc 분석이 가능

데이터 레이크 – 데이터 저장과 처리를 분리

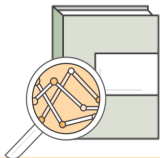


“급격히 늘어나는 데이터에 맞게
어떻게 시스템을 스케일업 할
것인가?”



데이터 저장공간과 분석을 위한 컴퓨팅
리소스를 분리
필요한 리소스만 언제든지 추가 가능

데이터 레이크의 중요한 구성요소



데이터 카탈로그와
검색



수집과 저장



데이터 준비와
변환



데이터 처리와
분석



다양한 접근
방식과 UI



데이터 안정성과
보안

AWS는 데이터 레이크를 위한 모든 서비스를 제공



수집



저장



분석 / 처리



시각화 / 활용



Kinesis
스트리밍 데이터



Direct Connect
데이터 센터와 연결



Snowball
벌크 데이터 로드



Database Migration Service
Oracle, Netezza 등의
데이터 импорт

더 많은 방법들..



Glue
데이터카탈로그와 ETL



Amazon S3
안전하고, 비용
효율적인 스토리지

어디서든 활용 가능한 ..



Redshift
데이터 웨어하우스



EMR
비정형 데이터 처리,
Apache Spark



Athena
ad-hoc 쿼리



QuickSight
시각화, BI



SageMaker
머신러닝 플랫폼

더 많은 방법들..

내부 사용자와 시스템

고객 대상 서비스

다양한 솔루션과 연동

AWS Glue

데이터 분석을 위한 준비 - 데이터 카탈로그와 ETL



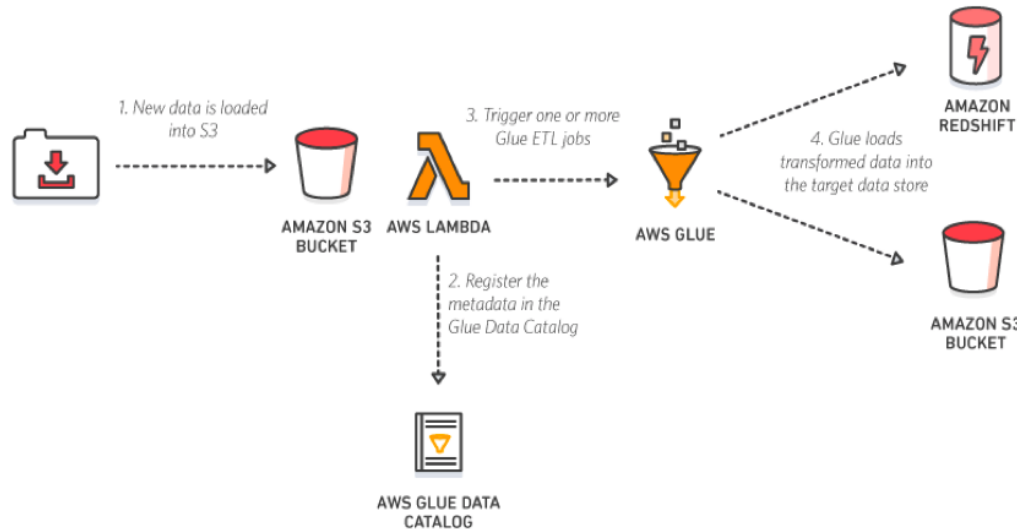
데이터에 대한 하나의 단일
데이터 카탈로그



데이터에 대한 변경과
추가를 관리할 수 있는
메타데이터 관리



데이터의 이동과 변환 작업
(ETL), Job 스케줄링



AWS Glue – 데이터 카탈로그

데이터를 쉽게 찾고 관리할 수 있게 해주는



Data Source : S3, JDBC 호환 Database

크롤러는 자동적으로 데이터 스키마를 찾아서 저장

데이터의 검색과 ETL 작업을 가능

테이블 스키마 정보와 컬럼 레벨 통계 정보를 포함

데이터 분포와 통계정보를 활용하여 쿼리 성능 향상

Glue 데이터 카탈로그



Glue
데이터 카탈로그

데이터 레이크를 위한
중앙 집중 메타데이터

하나의 계정 내 단일된 뷰

하나의 카탈로그를 통해 **Athena, EMR, Redshift Spectrum**에서 모두 공유

몇가지 더 확장된 기능들 :

- **검색** - 메타데이터를 통한 데이터 검색
- **외부 접속 정보** - JDBC URLs, credentials
- **분류기** - 스키마 인식과 통합 (grok 지원)
- **버전 관리** - 스키마 변경과 메타데이터 업데이트에 대한 버전 관리

데이터 카탈로그 - 테이블 상세 정보 포함

테이블 속성

데이터 분포 통계

테이블 스키마

중첩 필드 구조

Table Properties:

- Name: simpletweets_json
- Description: analytics
- Database: analytics
- Classification: json
- Location: s3://gluesampleddata/simpletweets.json
- Connection: No
- Deprecated: No
- Last updated: Thu Aug 10 16:25:24 GMT-700 2017

Properties:

sizeKey	456580	objectCount	1	UPDATED_BY_CRAWLER	S3Crawler	CrawlerSchemaSerializer/Version	1.0
recordCount	1001	averageRecordSize	456	CrawlerSchemaDeserializer/Version	1.0	compressionType	none
						typeOfData	file

Schema:

	Column name	Data type
1	entities	struct
2	id	bigint
3	retweeted	boolean
4	text	string
5	user	struct

user schema details:

```
STRUCT
  contributors_enabled: BOOLEAN
  description: STRING
  favourites_count: INT
  followers_count: INT
  friends_count: INT
  id: INT
  lang: STRING
  location: STRING
  name: STRING
  profile_background_tile: BOOLEAN
```


데이터 카탈로그 - 자동적으로 파티션 구조 파악

AWS Glue Console

Table Details: githubevents_data

- Name: githubevents_data
- Description: gitarchive
- Database: gitarchive
- Classification: json
- Location: s3://glue-sample-datasets/examples/data/
- Connection: No
- Deprecated: No
- Last updated: Wed Nov 22 07:52:09 GMT-000 2017
- Input format: org.apache.hadoop.mapred.TextInputFormat
- Output format: org.apache.hadoop.hive.q1.io.HiveIgnoreKeyTextOutputFormat
- Serde serialization lib: org.openx.data.jsonserde.JsonSerDe
- Serde parameters: paths actor,created_at,org,payload,public,repo,type
- sizeKey: 146649736039, objectCount: 1, UPDATED_BY_CRAWLER: githubarchive
- Table properties: CrawlerSchemaSerializerVersion: 1.0, recordCount: 27833145, averageRecordSize: 2423, CrawlerSchemaDeserializerVersion: 1.0, compressionType: gzip, typeOfData: file

Schema

Column name	Data type	Key
1 id	string	
2 type	string	
3 actor	struct	
4 repo	struct	
5 payload	struct	
6 public	boolean	
7 created_at	string	
8 org	struct	
9 year	string	Partition (0)
10 month	string	Partition (1)
11 day	string	Partition (2)

파티션 구조 탐지

AWS Glue Console

Table Details: githubevents_data

- Name: githubevents_data
- Description: gitarchive
- Database: gitarchive
- Classification: json
- Location: s3://glue-sample-datasets/examples/data/
- Connection: No
- Deprecated: No
- Last updated: Wed Nov 22 07:52:09 GMT-000 2017
- Input format: org.apache.hadoop.mapred.TextInputFormat
- Output format: org.apache.hadoop.hive.q1.io.HiveIgnoreKeyTextOutputFormat
- Serde serialization lib: org.openx.data.jsonserde.JsonSerDe
- Serde parameters: paths actor,created_at,org,payload,public,repo,type
- sizeKey: 146649736039, objectCount: 1, UPDATED_BY_CRAWLER: githubarchive
- Table properties: CrawlerSchemaSerializerVersion: 1.0, recordCount: 27833145, averageRecordSize: 2423, CrawlerSchemaDeserializerVersion: 1.0, compressionType: gzip, typeOfData: file

Schema

Column name	Data type	Key
1 id	string	
2 type	string	
3 actor	struct	
4 repo	struct	
5 payload	struct	
6 public	boolean	
7 created_at	string	
8 org	struct	
9 year	string	Partition (0)
10 month	string	Partition (1)
11 day	string	Partition (2)

데이터 카탈로그 - 스키마 변경 탐지 및 버전 관리

데이터 구조가 변경되면 자동적으로 업데이트 하고 버전 관리 가능

Version 1

Last updated 21 Aug 2017 Table

Name: simpletweets_json

Description: simpletweets_json

Database: simpletweets_json

Classification: json

Location: s3://gluesampleddata/simpletweets.json

Connection: No

Deprecated: No

Last updated: Mon Aug 21 15:23:42 GMT-700 2017

Input format: org.apache.hadoop.mapred.TextInputFormat

Output format: org.apache.hadoop.hive.q1.io.HiveIgnoreKeyTextOutputFormat

Serde serialization lib: org.openx.data.jsonserde.JsonSerDe

Serde parameters: paths entities,id,retweeted,text,user

sizeKey: 456580 objectCount: 1 UPDATED_BY_CRAWLER: TestS3Crawler

mycustom: abc CrawlerSchemaSerializerVersion: 1.0 recordCount: 1001

Table properties

averageRecordSize: 456 CrawlerSchemaDeserializerVersion: 1.0

compressionType: none typeOfData: file

Change	Column name	Data type	Key
	id	bigint	
	retweeted	boolean	
	text	string	
	user	struct	

Version 2

Last updated 25 Nov 2017 Table

Name: simpletweets_json

Description: simpletweets_json

Database: simpletweets_json

Classification: json

Location: s3://gluesampleddata/simpletweets.json

Connection: No

Deprecated: No

Last updated: Sat Nov 25 12:30:28 GMT-800 2017

Input format: org.apache.hadoop.mapred.TextInputFormat

Output format: org.apache.hadoop.hive.q1.io.HiveIgnoreKeyTextOutputFormat

Serde serialization lib: org.openx.data.jsonserde.JsonSerDe

Serde parameters: paths entities,id,retweeted,text,user

sizeKey: 456580 objectCount: 1 UPDATED_BY_CRAWLER: TestS3Crawler

mycustom: abc CrawlerSchemaSerializerVersion: 1.0 recordCount: 1001

Table properties

averageRecordSize: 456 CrawlerSchemaDeserializerVersion: 1.0

compressionType: none typeOfData: file

Change	Column name	Data type	Key
	id	bigint	
	retweeted	boolean	
	text	string	
	user	struct	
Added	url	string	

데이터 카탈로그가 만들어지면..



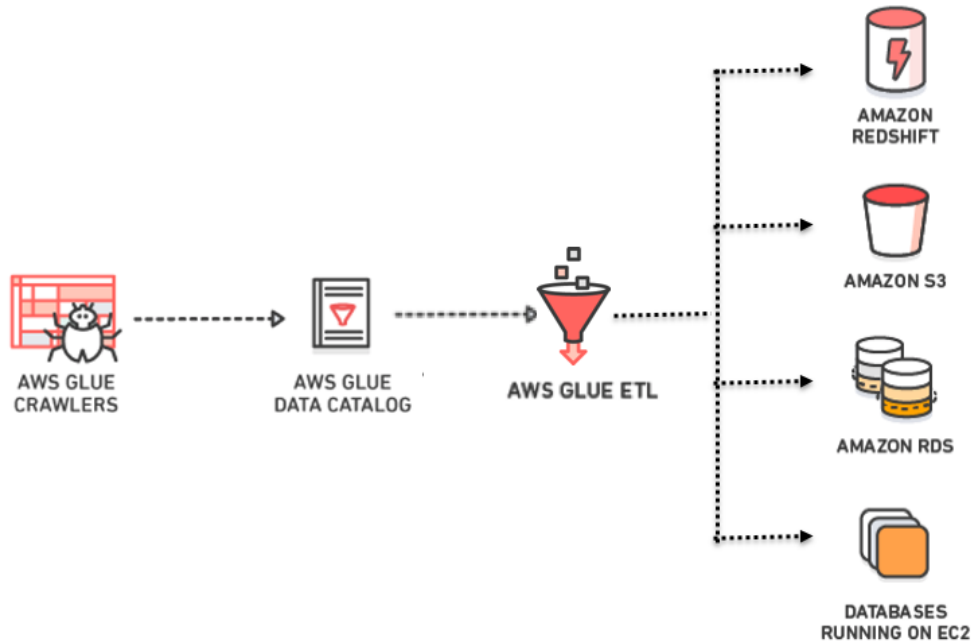
필요한 데이터의 검색 가능



Athena, EMR, Redshift 등에서 단일된 뷰로 동일한 데이터에 접근 / 활용 가능



ETL 작업의 데이터 소스로 즉시 활용 가능



빠르게 필요한 데이터에 대한 검색

테이블 검색

검색 결과 뷰를 저장

Tables A table is the metadata definition that represents your data, including its schema. A table can be used as a source or target in a job definition.

The screenshot shows the Amazon Athena console interface. On the left, a table list is displayed with columns 'Name' and 'Location'. The 'cloudtraildata' table is selected. A context menu is open over the 'cloudtraildata' table, showing options: 'Edit table details', 'View details', 'View data' (highlighted with an orange arrow), and 'Delete table'. The 'View data' option is linked to an external view. The main area shows a search bar with 'log' entered and a 'Filter or search for tables...' dropdown. To the right, a 'Save view' button is visible. Below the table list, a query editor is shown with a SQL query:

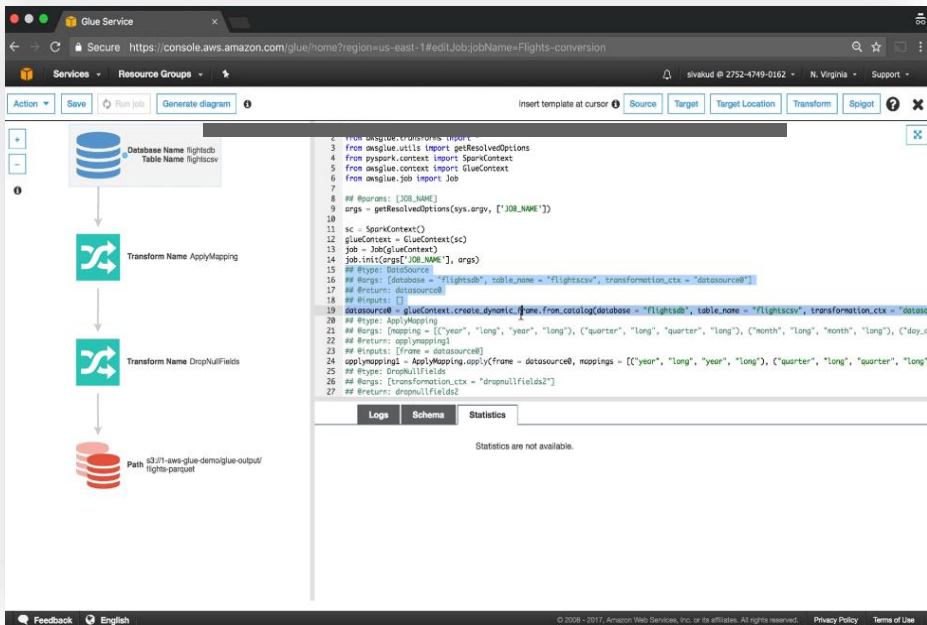
```
select *
from cloudtraildata.parquettrail
where eventtime > '2017-10-23T12:00:00Z' AND eventtime < '2017-10-23T13:00:00Z'
order by eventtime asc;
```

 The query is executed, and the results are displayed in a table with columns: eventversion, eventid, eventtime, sharedeventid, and requestparameters.durationseconds. The results show 10 rows of data.

Amazon Athena를 통한 데이터 쿼리

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 이벤트에 따른 Job 스케줄링

Zeppelin, PyCharm 등 익숙한 환경에서 수정,
디버그, 테스트가 가능하도록 Dev Endpoint
제공

Sample ETL Codes : <https://github.com/aws-labs/aws-glue-samples>

Job 생성 - 콘솔에서 코드 생성

Services

Resource Groups

N. Virginia

Support

Add job: schema

Job properties

DeleteMe

Data source

city_baltimore

Data target

canonical

Schema

Preview

Add column

Column name	Data type	Map to target	Column name	Data type
crimdate	string	crimdate	crimdate	string
crimetime	string	crime_time	crime_time	string
crimecode	string	crimecode	crimecode	string
location	string	location	location	string
description	string	description	description	string
inside/outside	string	-	weapon	string
weapon	string	weapon	district	string
post	bigint	-	neighborhood	string
district	string	district	total incidents	long
neighborhood	string	neighborhood		
location 1	string	-		
premise	string	-		
total incidents	bigint	total incidents		

Services
Resource Groups
N. Virginia
Support

Insert template at cursor
Source
Target
Target Location
Transform
Spigot

```

1 report sys
2 from aws glue.transforms import *
3 from aws glue.utils import getResolvedOptions
4 from pyspark.context import SparkContext
5 from aws glue.context import GlueContext
6 from aws glue.job import Job
7
8 ## @param: [JOB_NAME]
9 args = getResolvedOptions(sys.argv, ['JOB_NAME'])
10
11 sc = SparkContext()
12 glueContext = GlueContext(sc)
13 job = Job(glueContext)
14 job.init(args['JOB_NAME'], args)
15
16 ## @param: database = 'nytaxianalysis', table_name = 'city_baltimore', transformation_ctx = 'datasource'
17 ## @return: datasource
18 ## @inputs: []
19

```

Logs
Schema
Statistics

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

Job 북마크

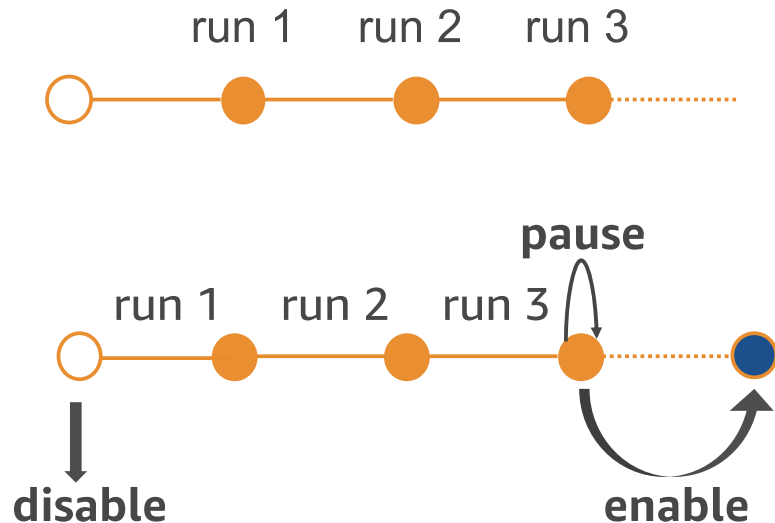
북마크 기능을 통해 지속적으로 추가되는 데이터에 대한 중복 작업 관리가 가능

예제 :

일단위로 증가하는 로그 데이터 처리

시간단위로 Kinesis Firehose 데이터 처리

DB에 저장된 데이터를 시간단위로 처리
(단일 PK 데이터)



옵션	동작 방식
Enable	이전 실행한 이후 데이터만 실행
Disable	이전 단계 무시, 전체 데이터 실행
Pause	필요에 따라 일시적으로 북마크 정지

Job 스케줄링과 모니터링

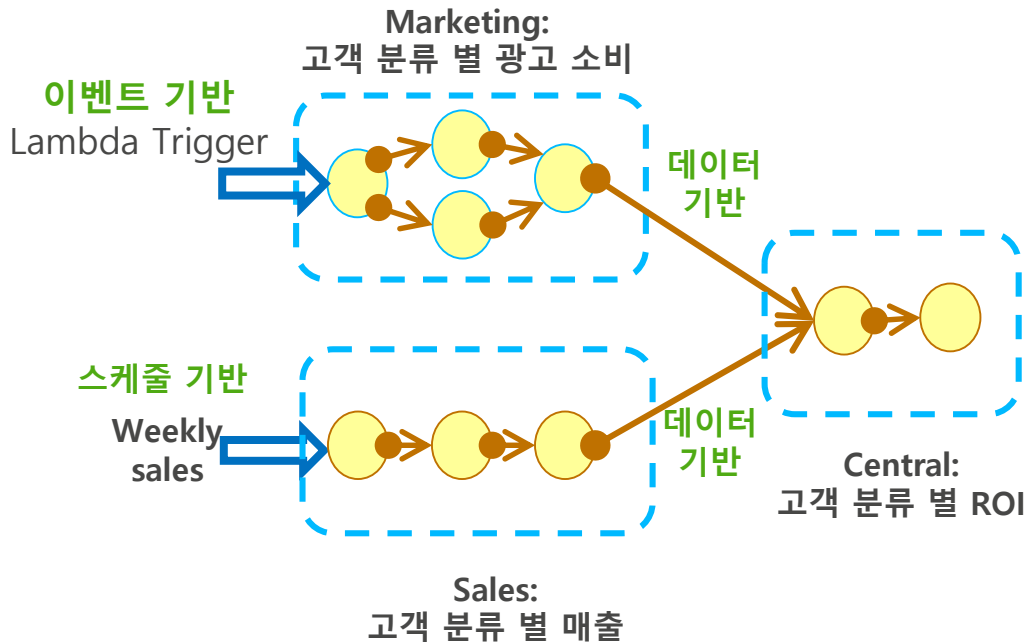
이벤트 기반 Job을 실행 가능하며, 여러 Job 사이에 의존성을 설정이 가능

- 각기 다른 조직에서 업무 연계와 Job의 재사용이 용이

다양한 Job 트리거 방법들

- **스케줄 기반** : 예) 특정 시간, 특정 일
- **이벤트 기반** : 예) Job 종료 / 실패 / 중단
- **On-demand** : 예) AWS Lambda

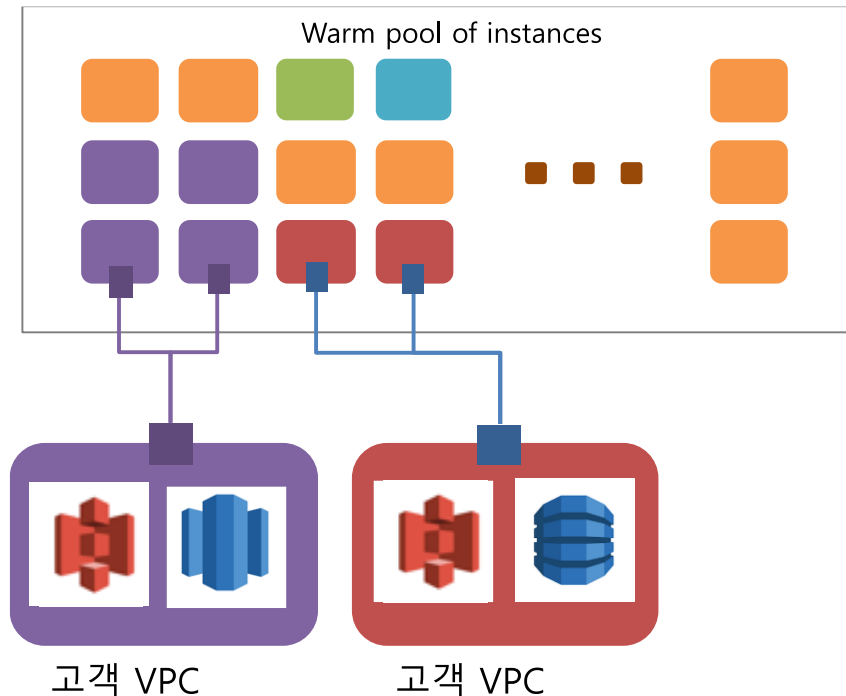
로그와 경고는 **Amazon CloudWatch**를 통해 확인 가능



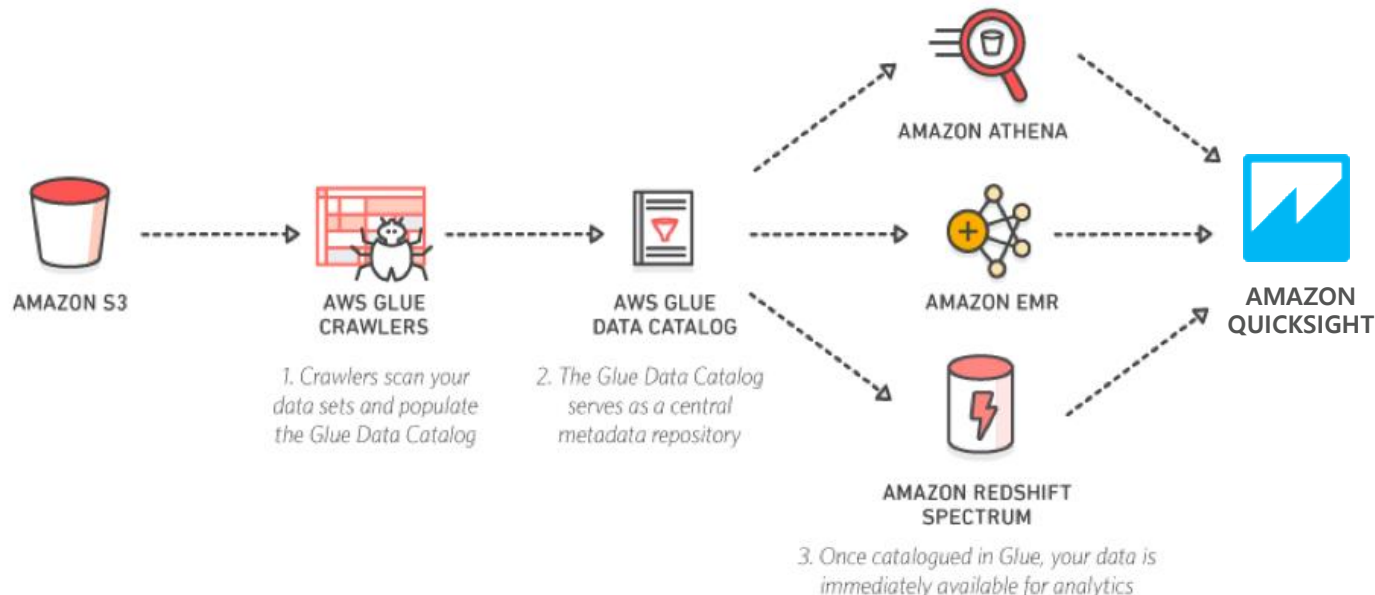
Job 실행 - 서버리스

Job을 실행하기 위해서 자동적으로 인프라를 생성하고 사용한 만큼만 과금

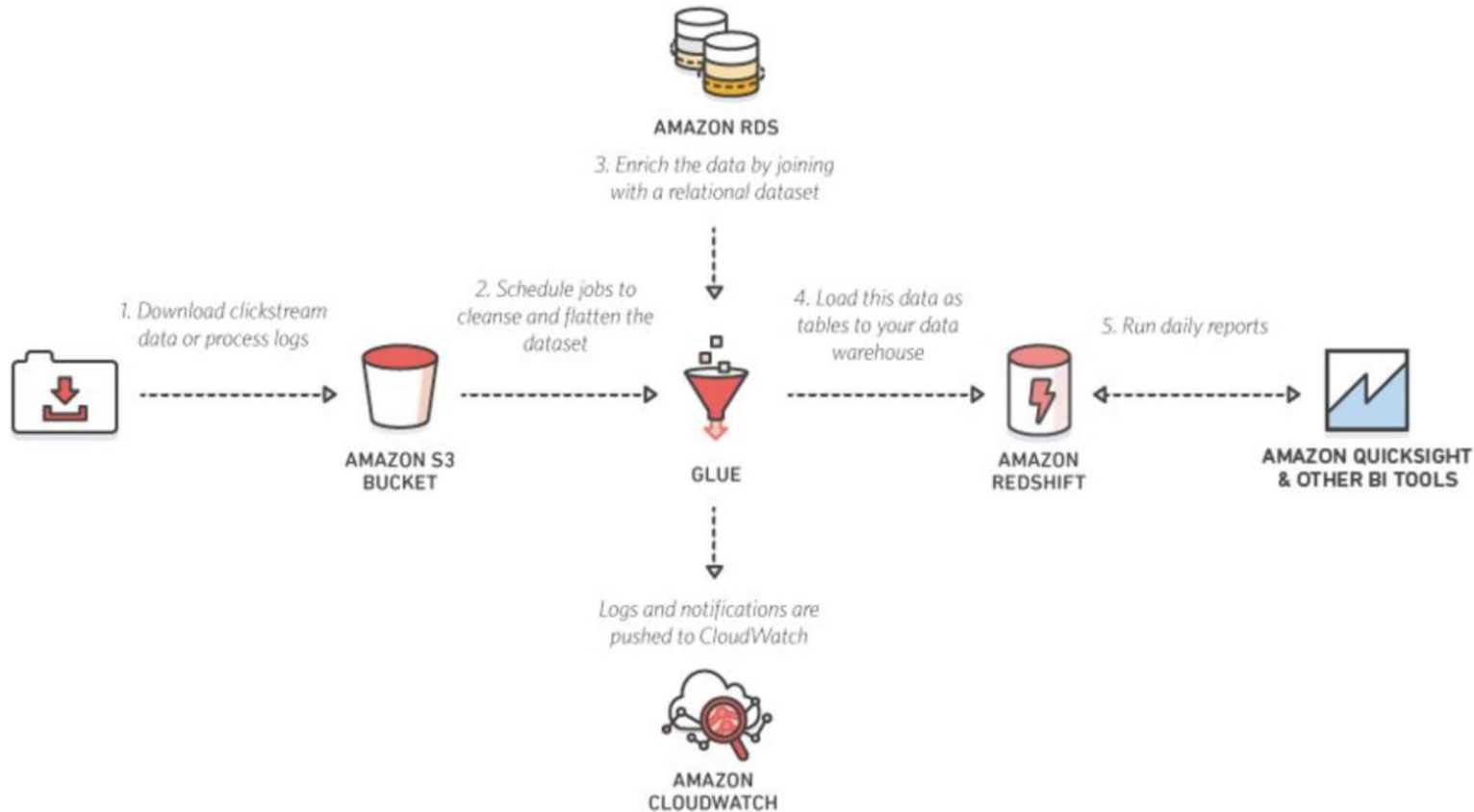
- 서버 풀 : Job 시작시간을 줄이기 위해 미리 설정된 서버 그룹을 운영
- 자동 설정된 VPC와 Role 기반 접근 제어
- 자동적으로 리소스 확장
- 단지, Job 실행을 위해 사용한 리소스에 대해서만 비용 지불



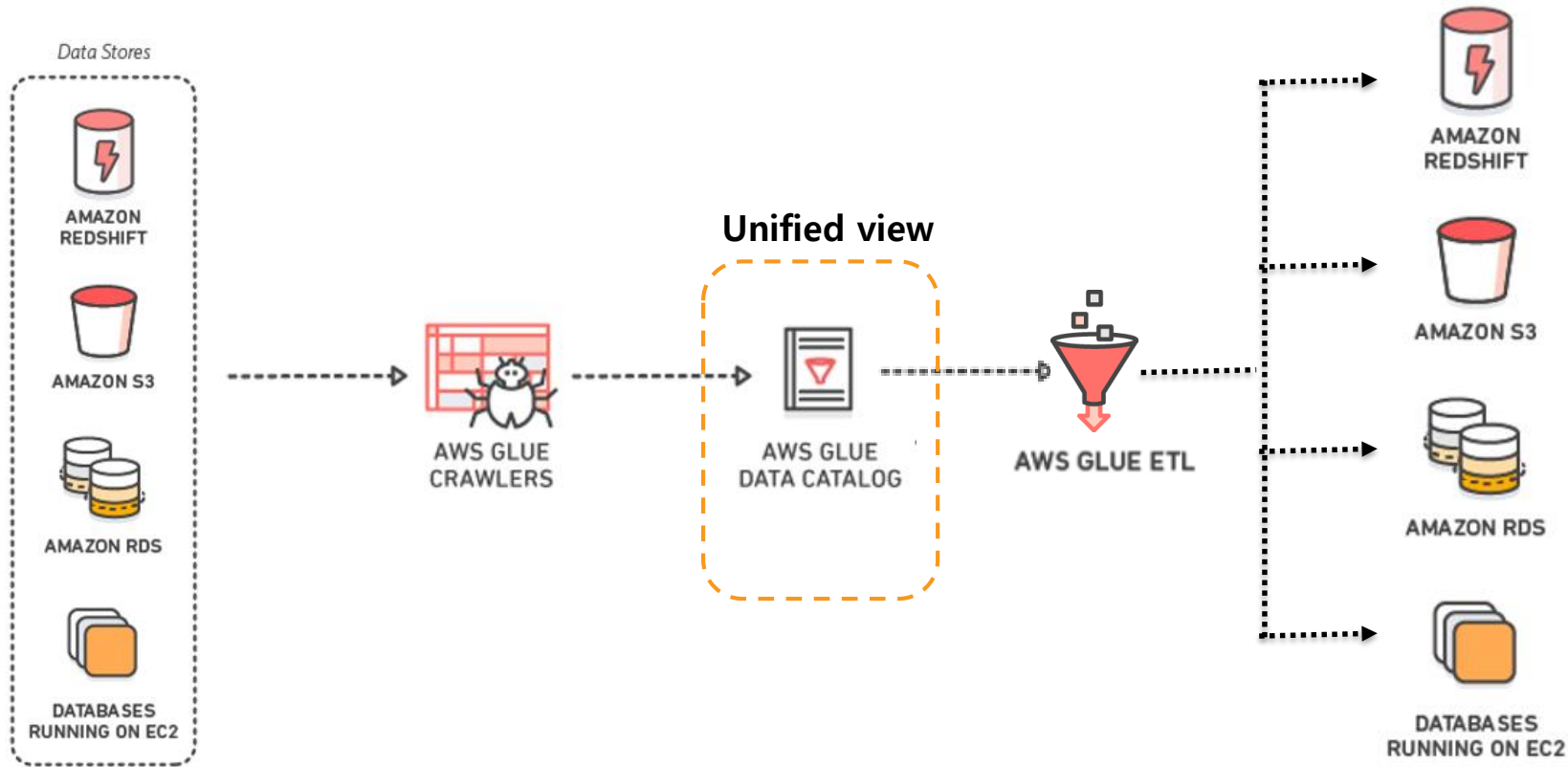
Glue 활용 패턴 - 다양한 방식으로 동일한 데이터 분석



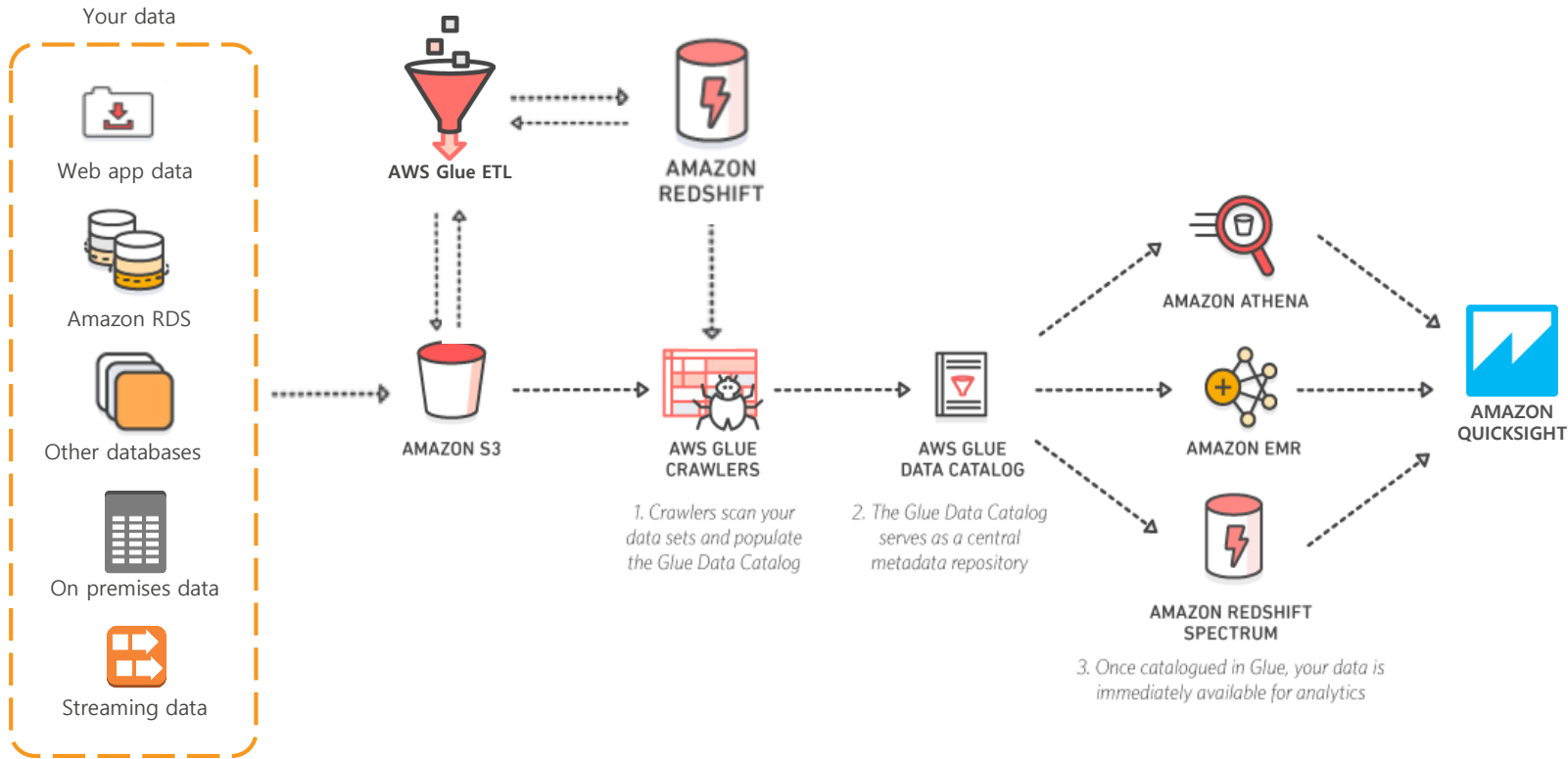
Glue 활용 패턴 - 데이터 웨어하우스로 ETL 작업



Glue 활용 패턴 - 다른 스토리지간의 데이터 이동



Glue 활용 패턴 - 데이터 레이크와 웨어하우스 통합



Amazon Athena

Amazon Athena : 정의

Amazon Athena는 표준 SQL을 사용해
Amazon S3에 저장된 데이터를 간편하게
분석할 수 있는 대화식 쿼리 서비스입니다.

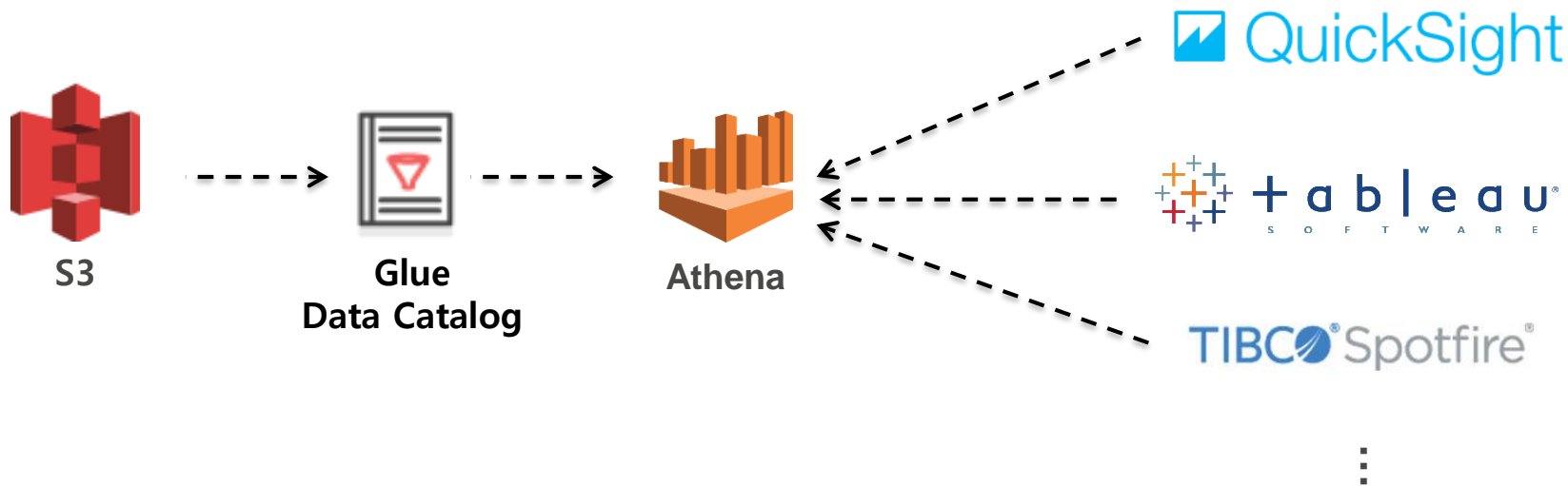
Amazon Athena : 특징

- ✓ 스토리지와 컴퓨팅 노드 분리
- ✓ Schema on Read
- ✓ Query를 위해 Data Loading / ETL 불필요, S3에서 직접 Query 실행
- ✓ Serverless : 인프라 관리 불필요, 자동 확장, Warm Compute Pools (Multi-AZ)
- ✓ 스캔된 데이터 만큼 과금
- ✓ 보안 : IAM을 통한 인증 / 암호화 : 테이블, Query문, Write Output
- ✓ AWS Glue 데이터 카탈로그와 통합

Amazon Athena : Presto SQL

- ✓ ANSI SQL 호환
- ✓ Complex joins, nested queries & window functions
- ✓ Complex data types (arrays, structs, maps)
- ✓ Presto 빌트인 functions
- ✓ 지원 파일 포맷
 - 텍스트 파일 : CSV, JSON, RegEx, Parquet, Avro, ORC, CloudTrail
 - 아파치 웹로그, TSV 파일 포맷
 - JSON (Simple, nested)
 - Columnar format (Apache Parquet, Apache ORC)
 - 압축 파일 : GZIP, Zlib, LZO, Snappy
 - AVRO Support

Amazon Athena : JDBC를 통한 친숙한 BI도구와 연계



JDBC Connection String

```
jdbc:awsathena://athena.{REGION}.amazonaws.com:443
```

Amazon Athena : 비용 최적화

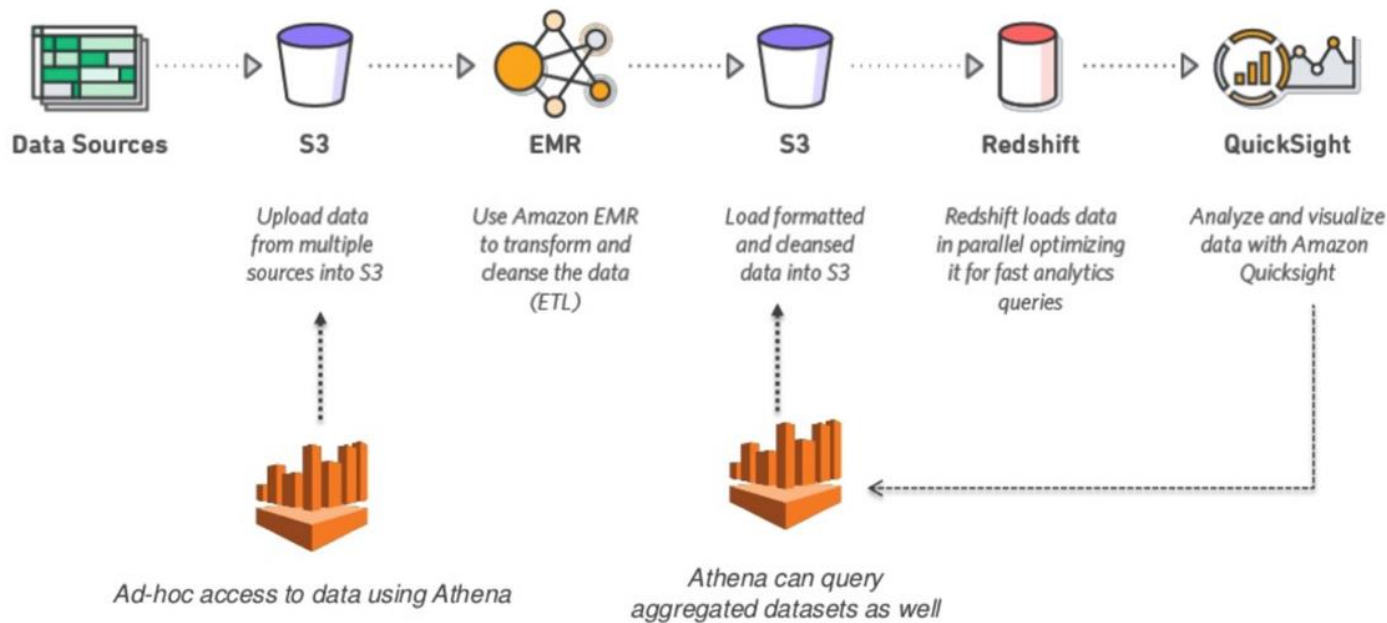
Dataset	Size on Amazon S3	Query Run time	Data Scanned	Cost
Logs stored as Text files	1 TB	237 seconds	1.15TB	\$5.75
Logs stored in Apache Parquet format*	130 GB	5.13 seconds	2.69 GB	\$0.013
Savings	87% less with Parquet	34x faster	99% less data scanned	99.7% cheaper

비용 절감을 위해 Compress/Parquet 등 Columnar format 으로 변경/파티셔닝 고려

Amazon Athena – 10가지 성능 향상 팁

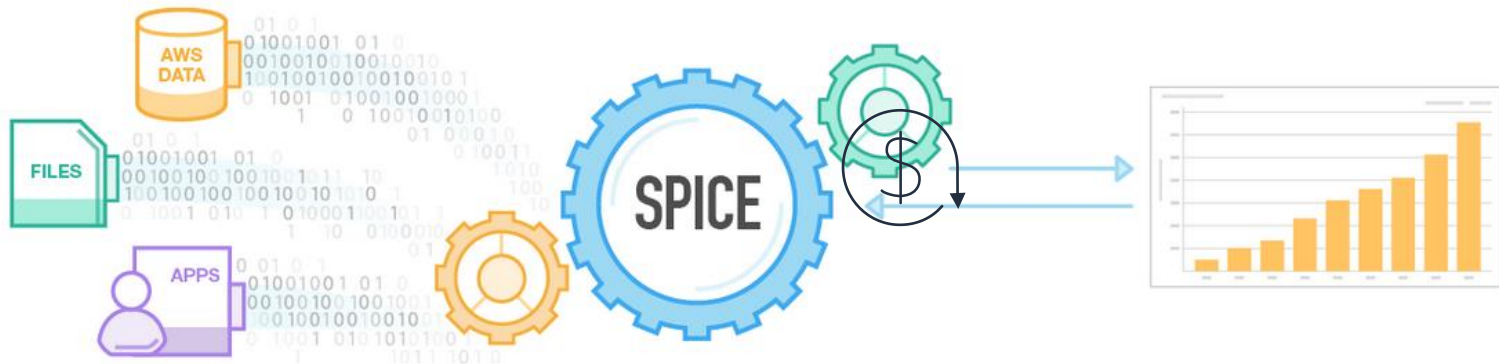
<https://aws.amazon.com/ko/blogs/korea/top-10-performance-tuning-tips-for-amazon-athena/>

Amazon Athena 사용 패턴



Amazon QuickSight

Amazon Quicksight Overview



DATA SOURCES

Connect to AWS data services;
upload files; or connect to apps
such as Salesforce

IN-MEMORY CALCULATION ENGINE

The Super-fast, Parallel, In-memory, Calculation
Engine ("SPICE") generates answers on large
datasets and returns rapid responses

QUICKSIGHT UI

SPICE allows for very fast analysis
and smart visualizations for
sharing and collaboration



자동 확장



사용한 만큼
과금

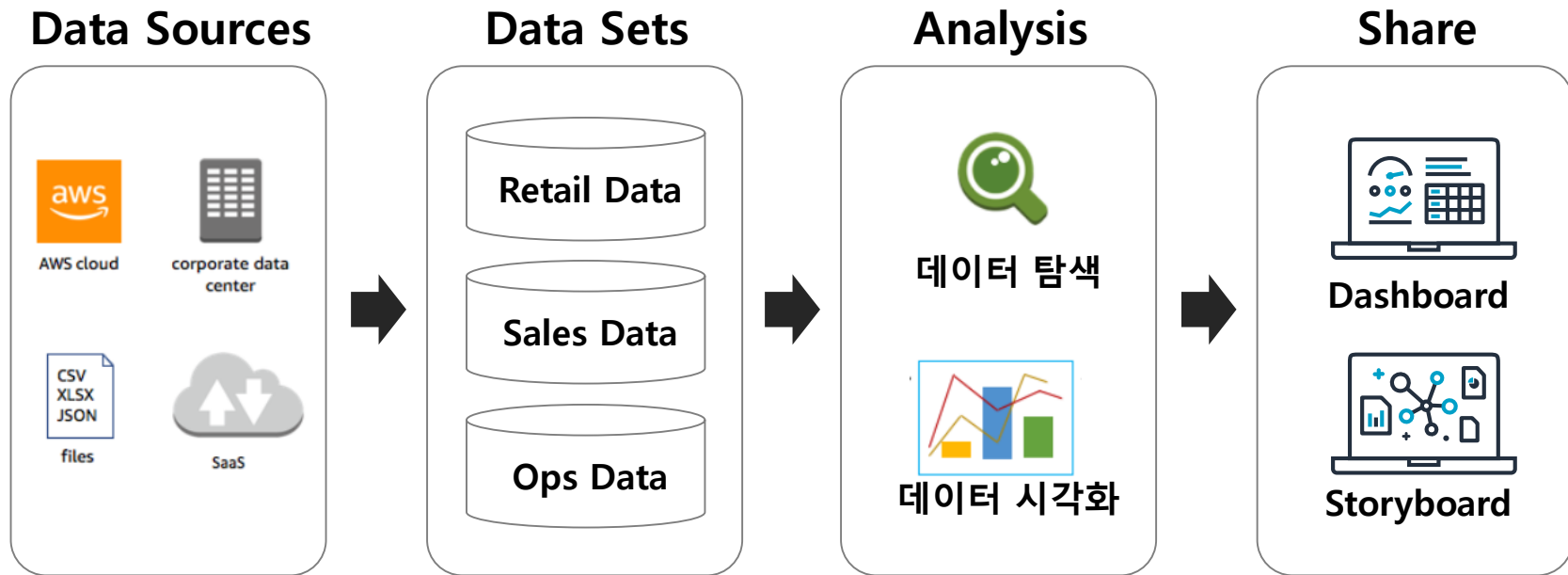


완전 관리형
신속한 배포



AWS서비스와
완벽한 통합

Amazon Quicksight Workflow



Amazon Quicksight SPICE In-Memory 엔진



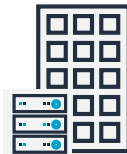
- **S**uper-fast, **P**arallel, **I**n-memory, **C**alculation **E**ngine
- 기계어 코드 생성으로 컴파일 되는 Query
- 풍부한 calculations
- SQL-like syntax
- Very fast response time to queries
- Fully managed – No hardware or software to license
- SPICE 용량은 동일 Region 사용자 간 공유

Amazon Quicksight 데이터 소스

RDB, NoSQL, Amazon EMR, S3, Files, Streaming Data Sources...

On-premises

Securely connect to on-premise databases and flat files like Excel and CSV



- Excel
- CSV
- Teradata
- MySQL
- SQL Server



In the cloud

Connect to hosted database, big data formats, and secure VPCs



- Redshift
- RDS
- S3
- Athena
- Aurora
- Teradata
- MySQL
- Presto
- Spark
- SQL Server
- Postgre SQL
- MariaDB
- Snowflake



Applications
























Connect directly to third party business applications



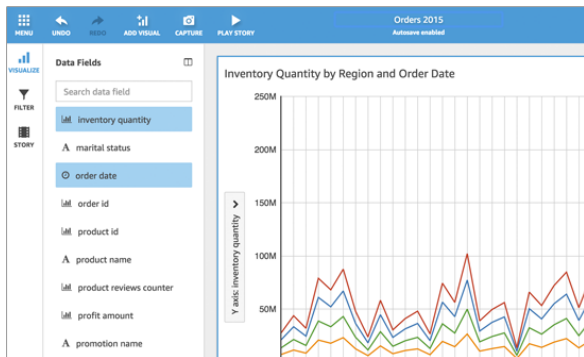
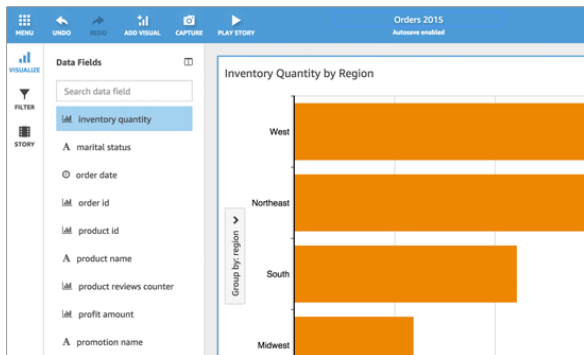
- Salesforce
- Square
- Adobe Analytics
- Jira
- ServiceNow
- Twitter
- Github



Amazon Quicksight 데이터 소스

 Upload a file (.csv, .tsv, .clf, .elf, .xlsx, .json)	 Salesforce Connect to Salesforce	 S3 Analytics	 S3
 Athena	 RDS	 Redshift Auto-discovered	 Redshift Manual connect
 MySQL	 PostgreSQL	 SQL Server	 Aurora
 MariaDB	 Presto	 Spark	 Teradata Provided by Teradata
 Snowflake	 AWS IoT Analytics	 GitHub	 Twitter
 Jira	 ServiceNow	 Adobe Analytics	

직관적인 시각화 및 AutoGraph



- 자동으로 데이터 타입 인지
- 최적 쿼리 생성
- 적절한 그래프 타입 선택
- 그래프 타입 커스터마이징 기능
- 굉장히 빠른 반응

요약

