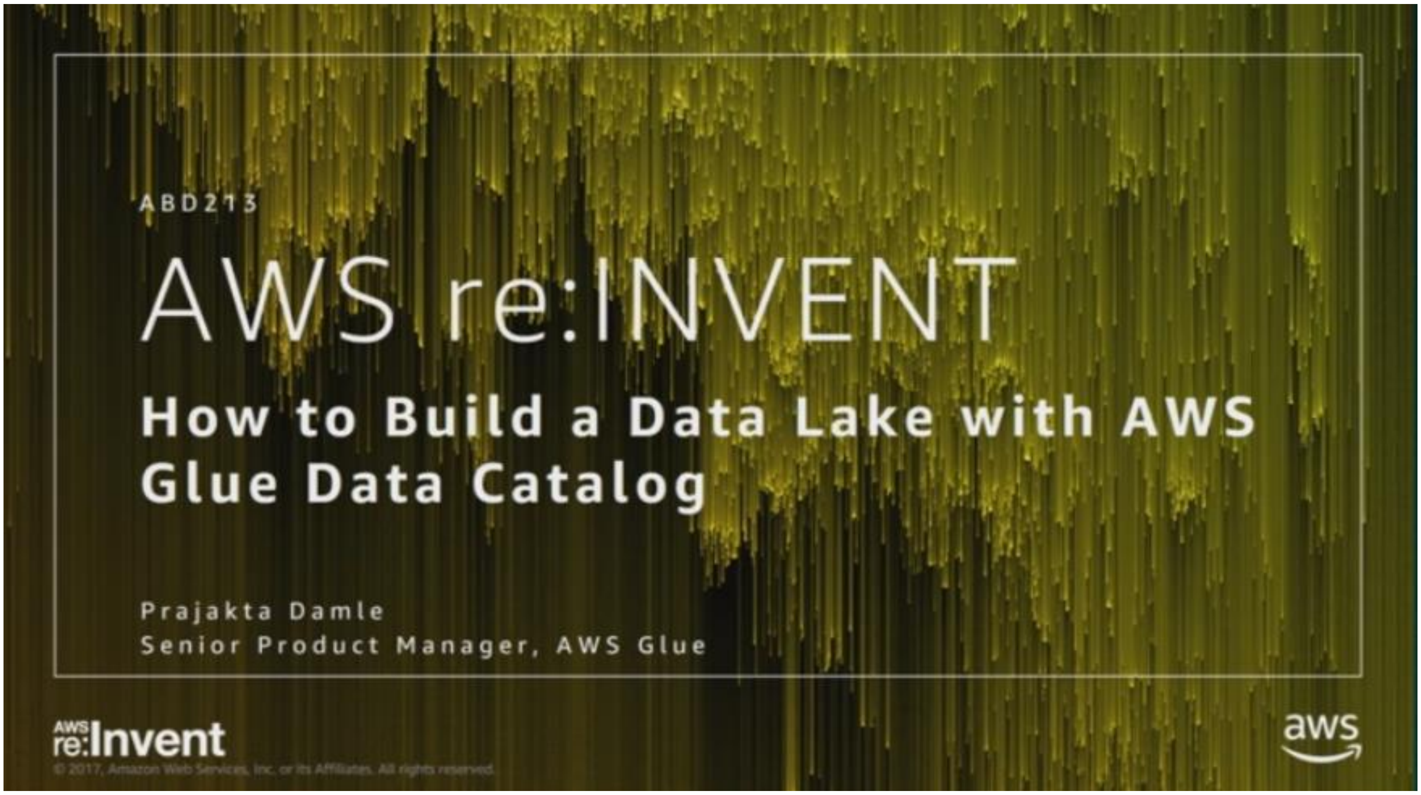


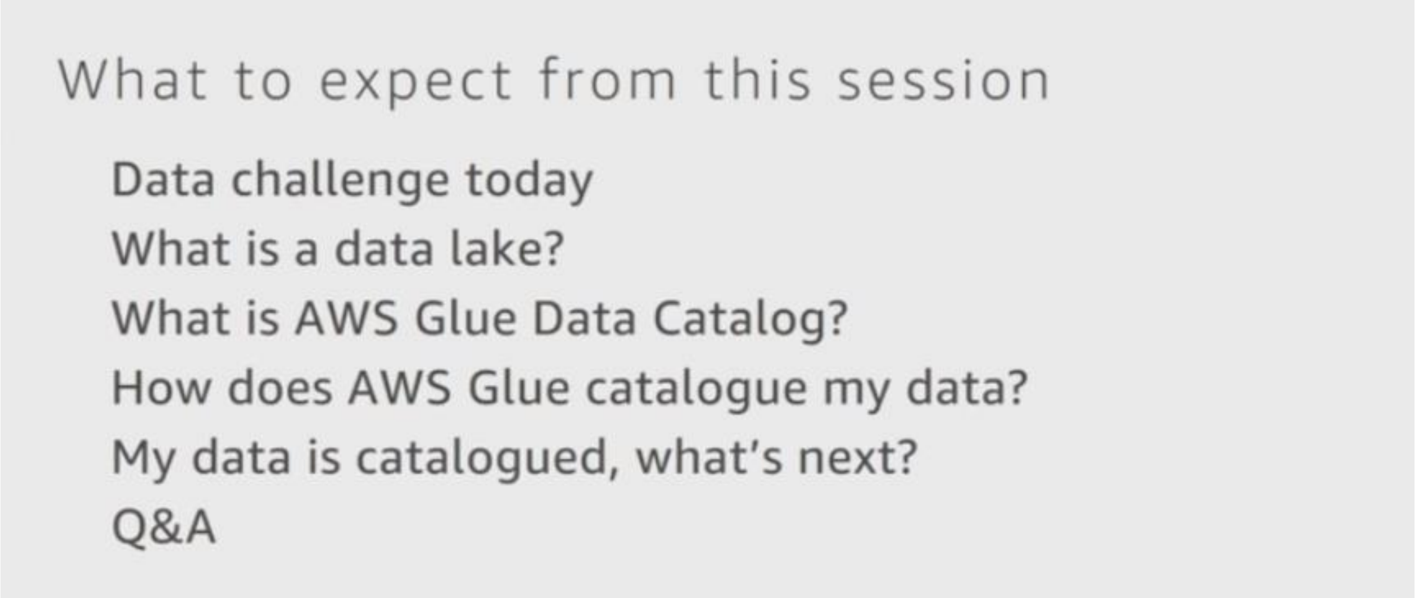
ABD213

AWS re:INVENT

How to Build a Data Lake with AWS Glue Data Catalog

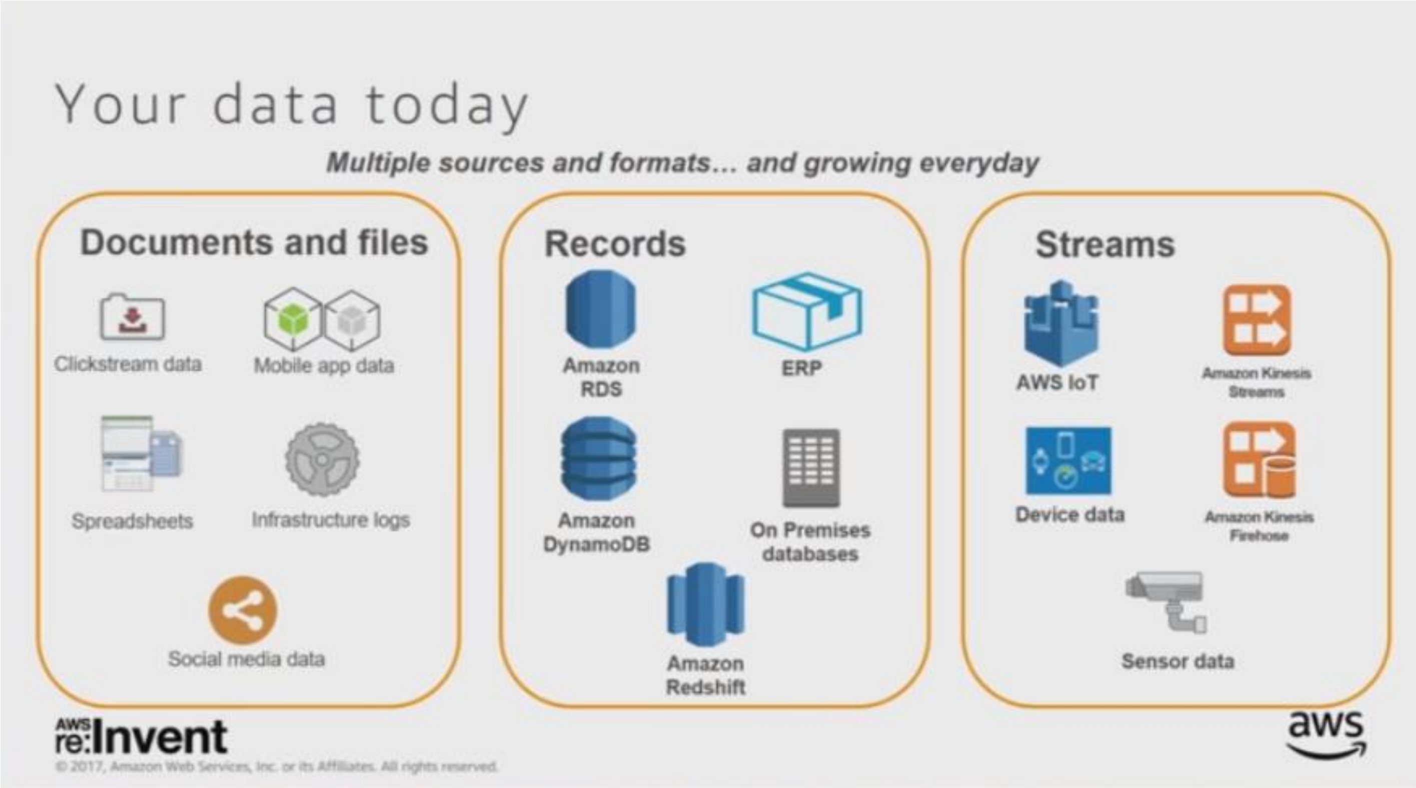
Prajakta Damle
Senior Product Manager, AWS Glue

AWS re:Invent
aws



What to expect from this session

Data challenge today
What is a data lake?
What is AWS Glue Data Catalog?
How does AWS Glue catalogue my data?
My data is catalogued, what's next?
Q&A



Your data today

Multiple sources and formats… and growing everyday

Documents and files

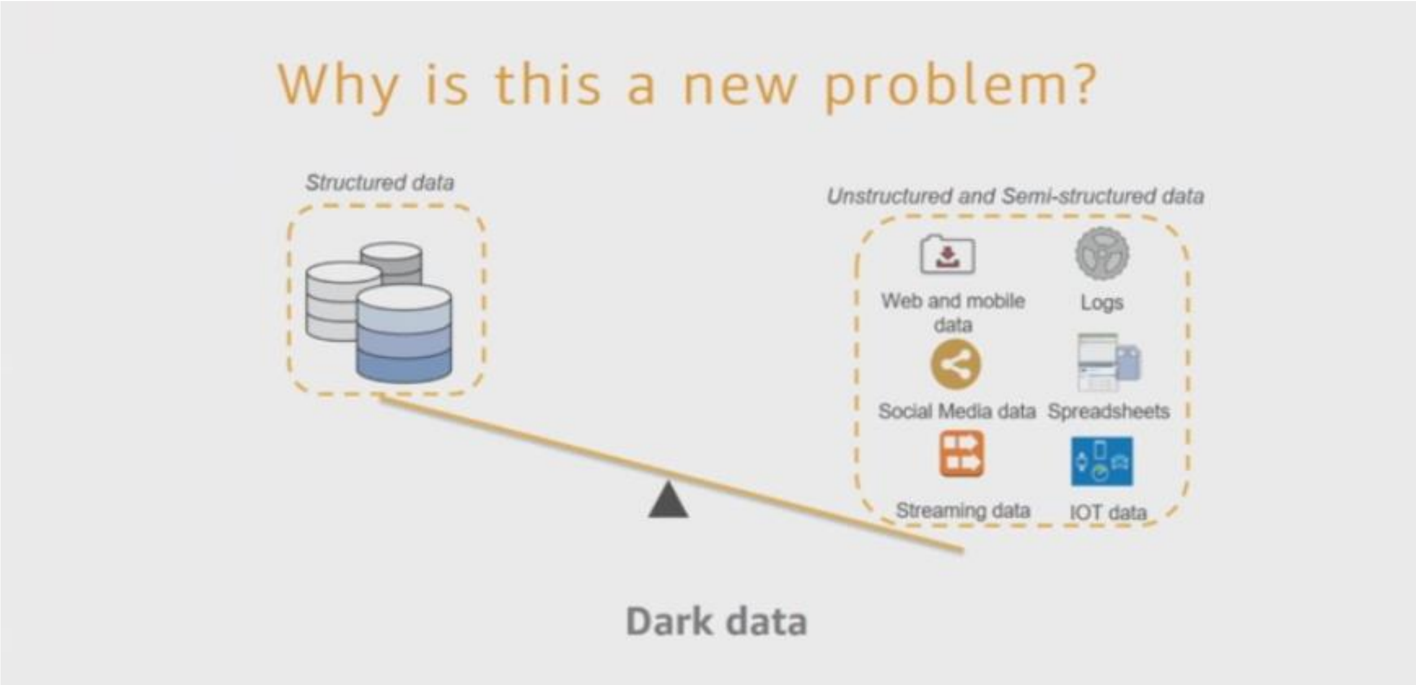
- Clickstream data
- Mobile app data
- Spreadsheets
- Infrastructure logs
- Social media data

Records

- Amazon RDS
- ERP
- Amazon DynamoDB
- On Premises databases
- Amazon Redshift

Streams

- AWS IoT
- Amazon Kinesis Streams
- Device data
- Amazon Kinesis Firehose
- Sensor data



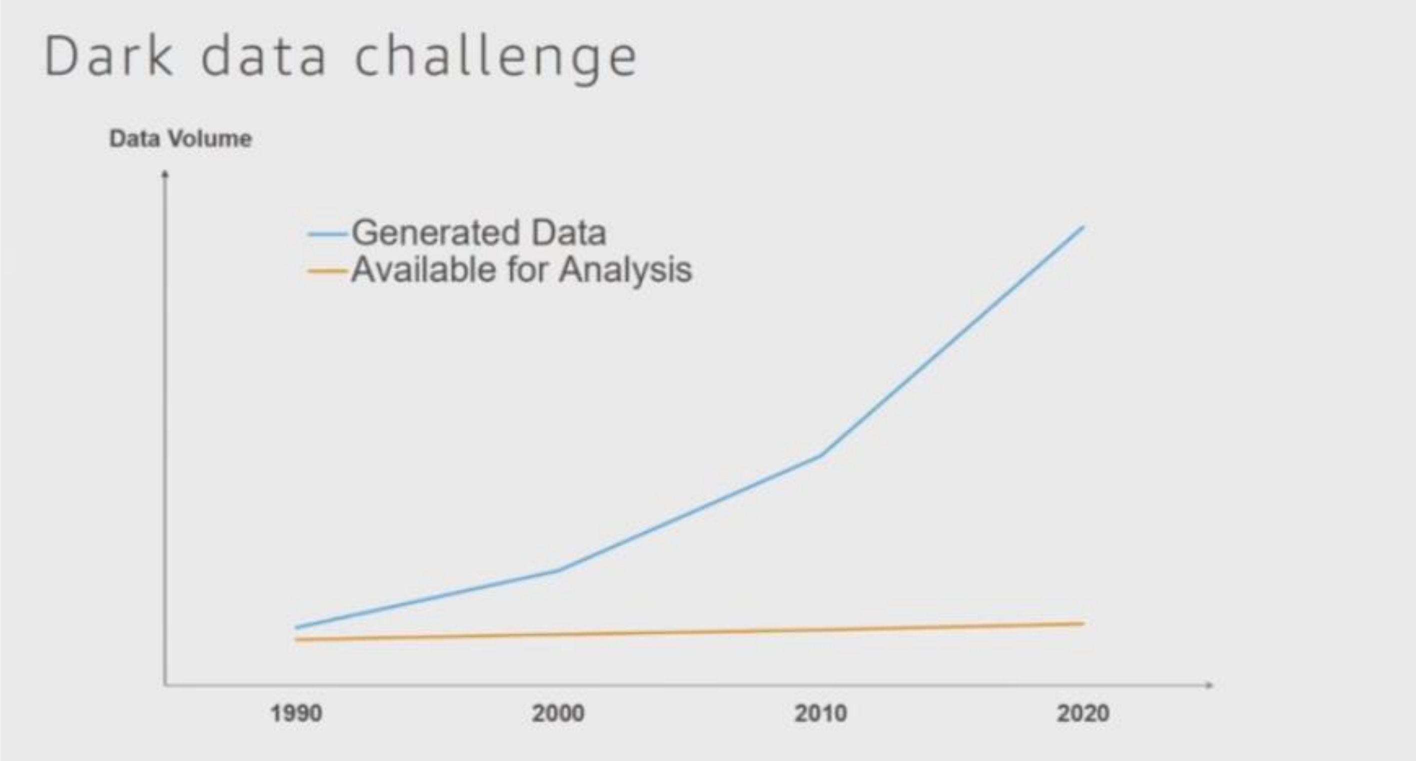
Why is this a new problem?

Structured data

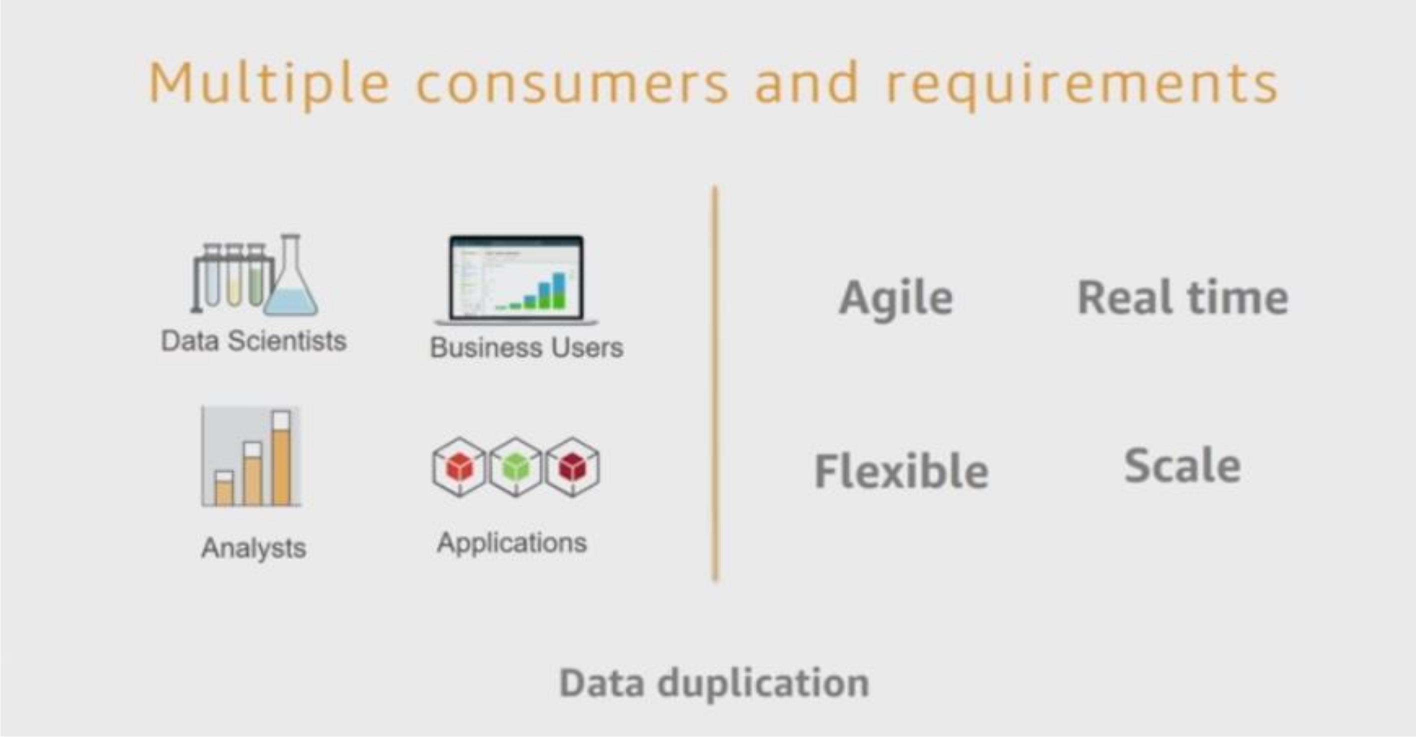
Unstructured and Semi-structured data

- Web and mobile data
- Logs
- Social Media data
- Spreadsheets
- Streaming data
- IOT data

Dark data



Dark data challenge



Multiple consumers and requirements

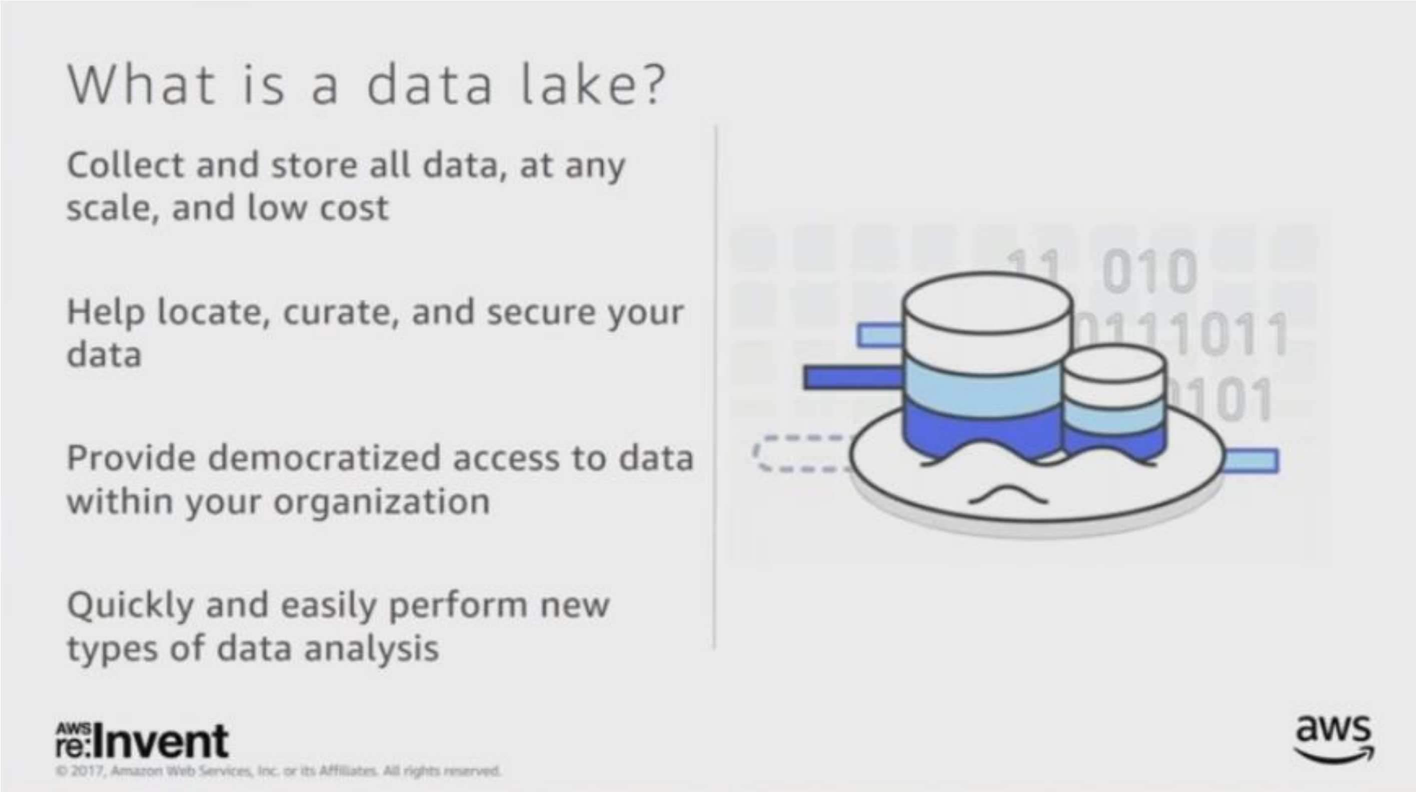
Agile

Real time

Flexible

Scale

Data duplication



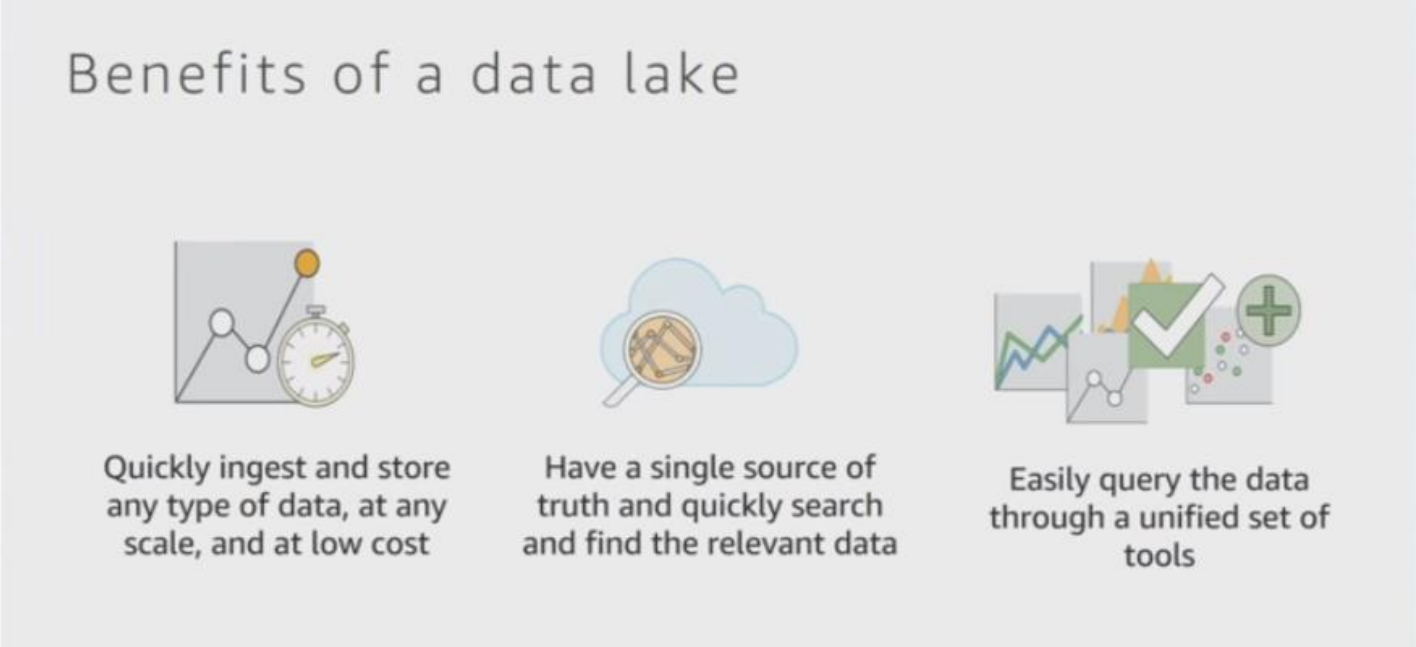
What is a data lake?

Collect and store all data, at any scale, and low cost

Help locate, curate, and secure your data

Provide democratized access to data within your organization

Quickly and easily perform new types of data analysis

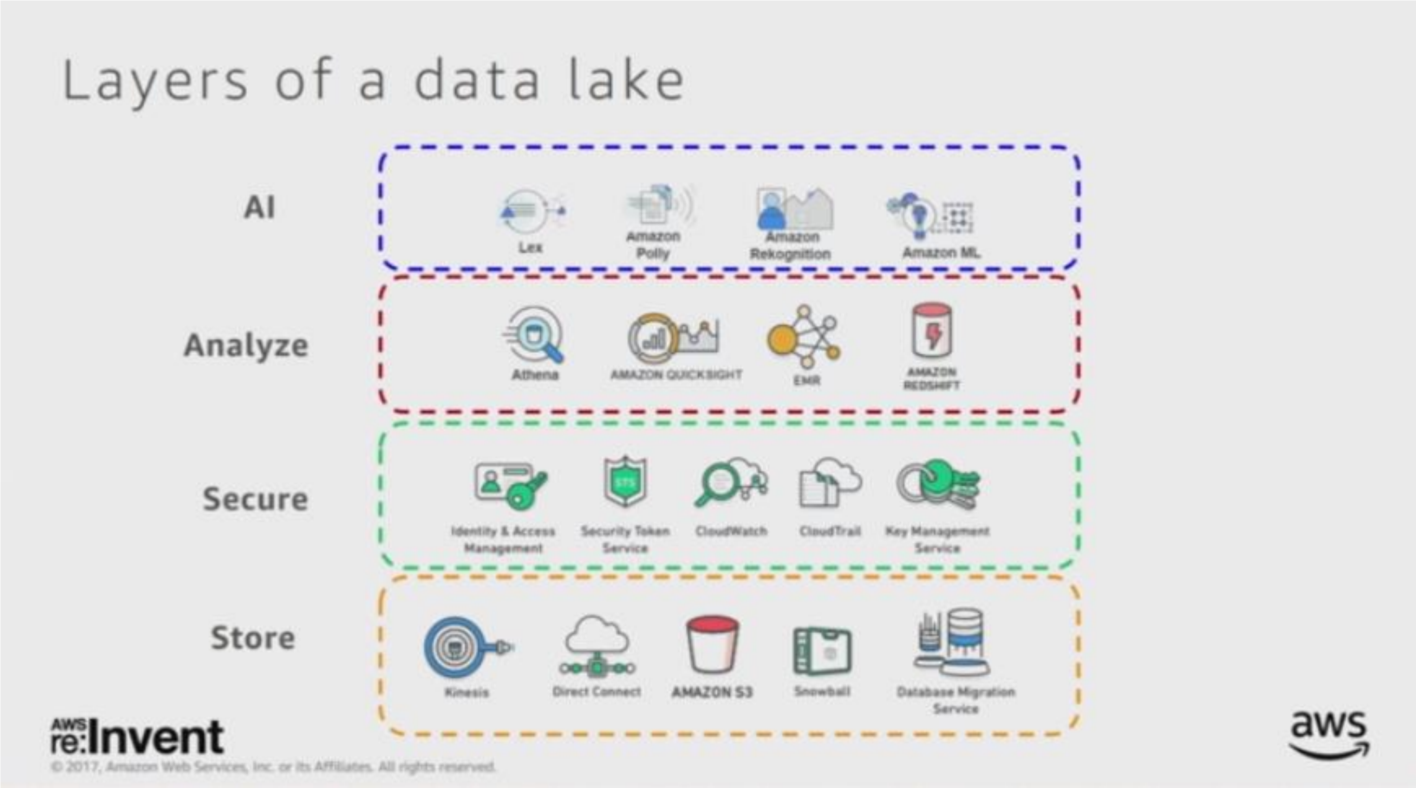


Benefits of a data lake

Quickly ingest and store any type of data, at any scale, and at low cost

Have a single source of truth and quickly search and find the relevant data

Easily query the data through a unified set of tools



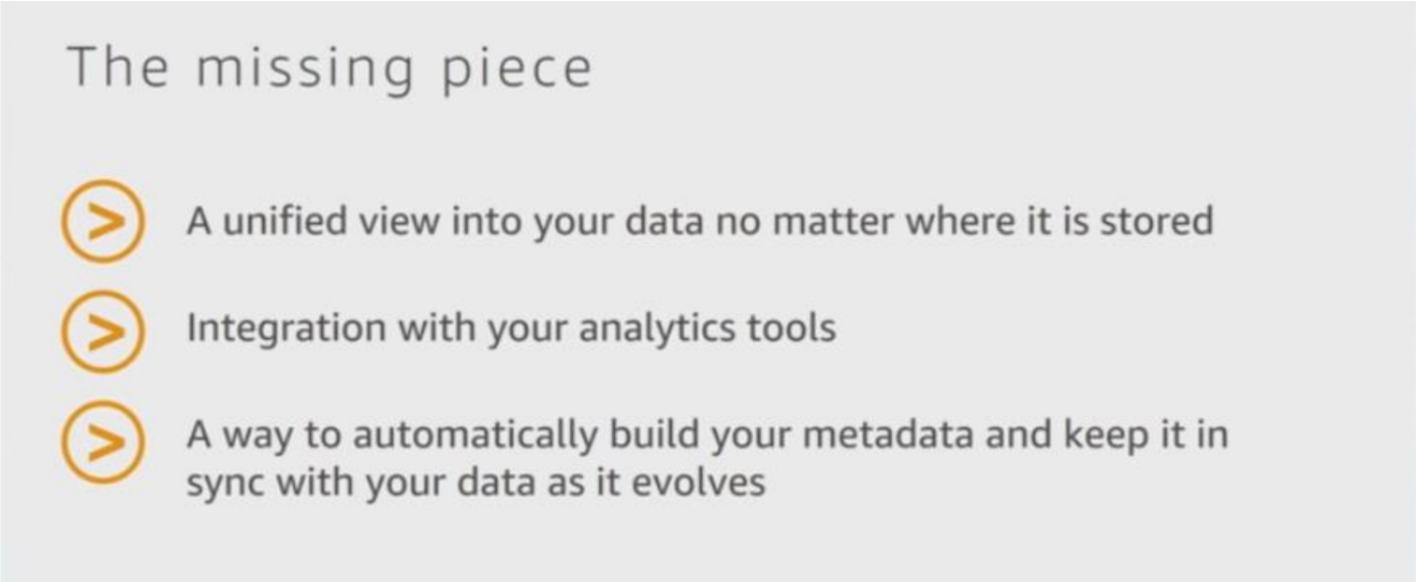
Layers of a data lake

AI: Lex, Amazon Polly, Amazon Rekognition, Amazon ML

Analyze: Athena, AMAZON QUICKSIGHT, EMR, AMAZON REDSHIFT

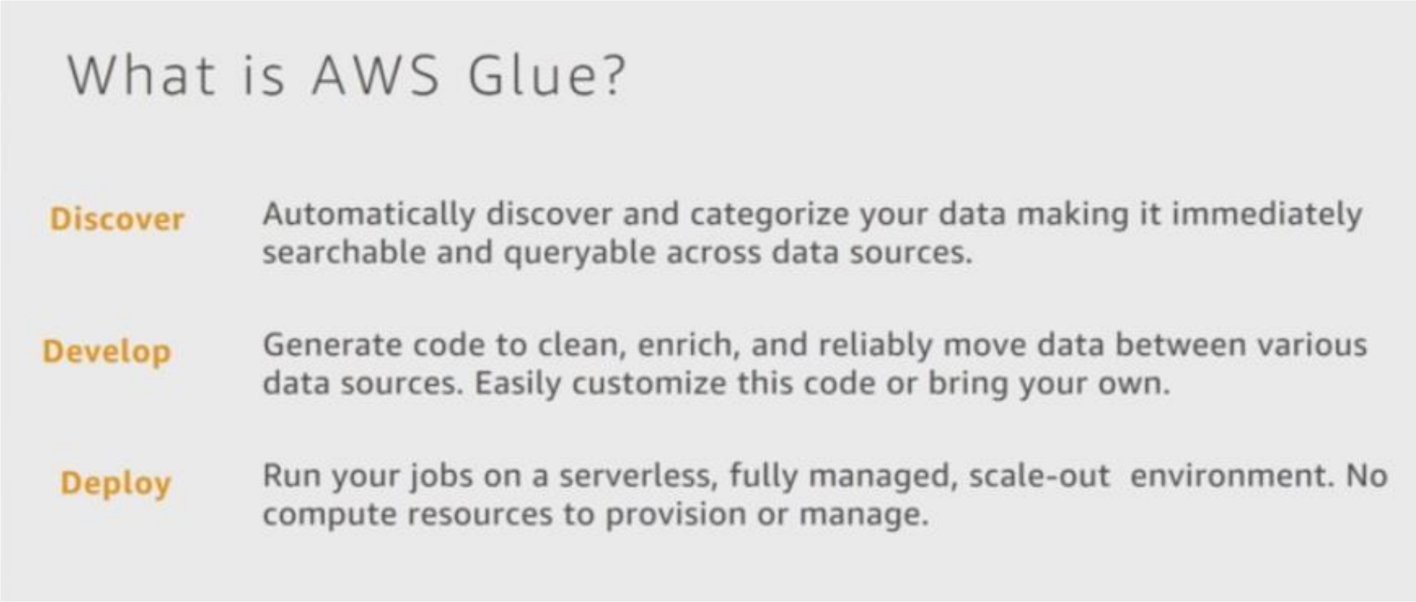
Secure: Identity & Access Management, Security Token Service, CloudWatch, CloudTrail, Key Management Service

Store: Kinesis, Direct Connect, AMAZON S3, Snowball, Database Migration Service



The missing piece

- A unified view into your data no matter where it is stored
- Integration with your analytics tools
- A way to automatically build your metadata and keep it in sync with your data as it evolves



What is AWS Glue?

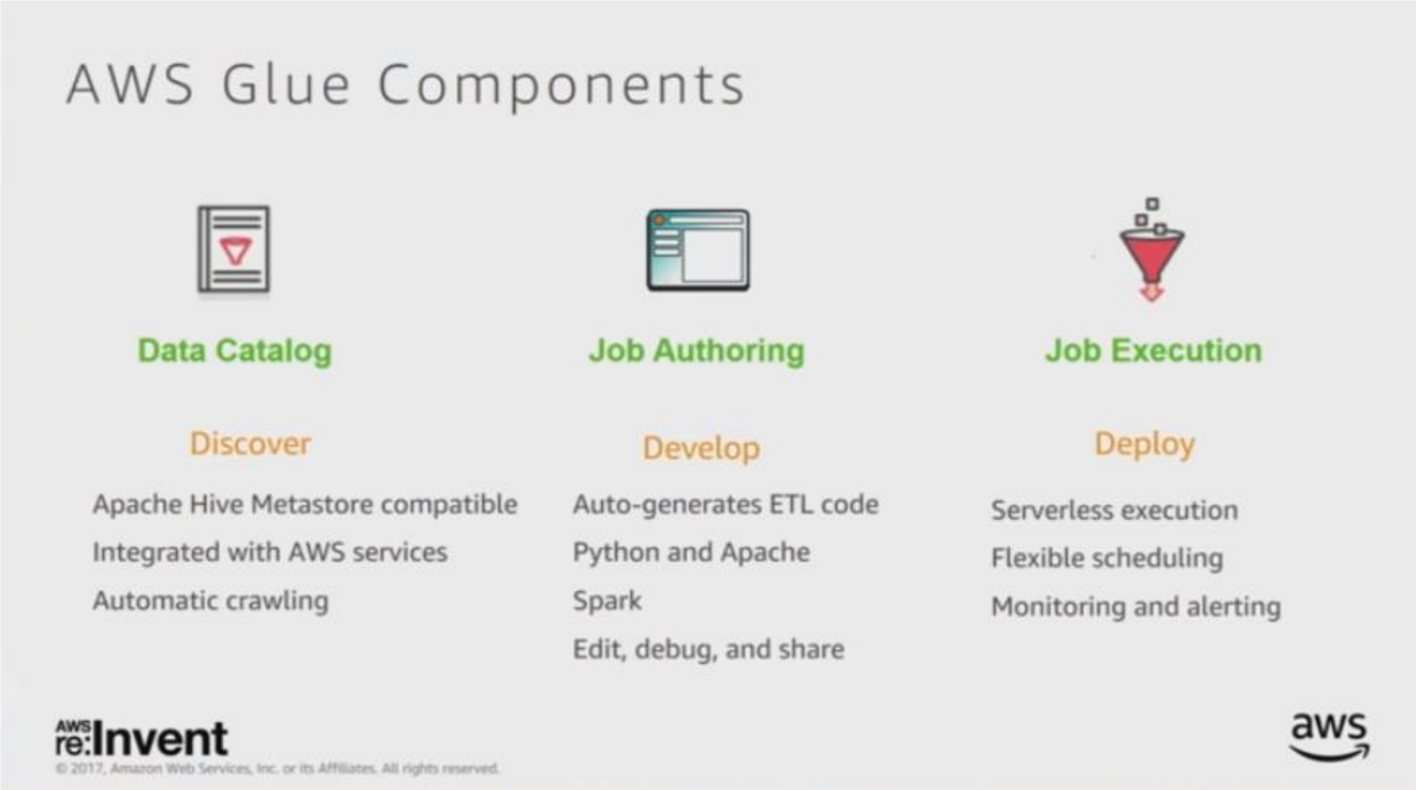
Discover Automatically discover and categorize your data making it immediately searchable and queryable across data sources.

Develop Generate code to clean, enrich, and reliably move data between various data sources. Easily customize this code or bring your own.

Deploy Run your jobs on a serverless, fully managed, scale-out environment. No compute resources to provision or manage.

Select AWS Glue customers

NTT docomo
News Corp
AUTODESK
MediaMath
MERCK
myTomorrows
OLX Group
ost
Expedia
amazon Prime Air
21ST CENTURY FOX
AWS re:Invent
aws



AWS Glue Components

Data Catalog

Discover

- Apache Hive Metastore compatible
- Integrated with AWS services
- Automatic crawling

Job Authoring

Develop

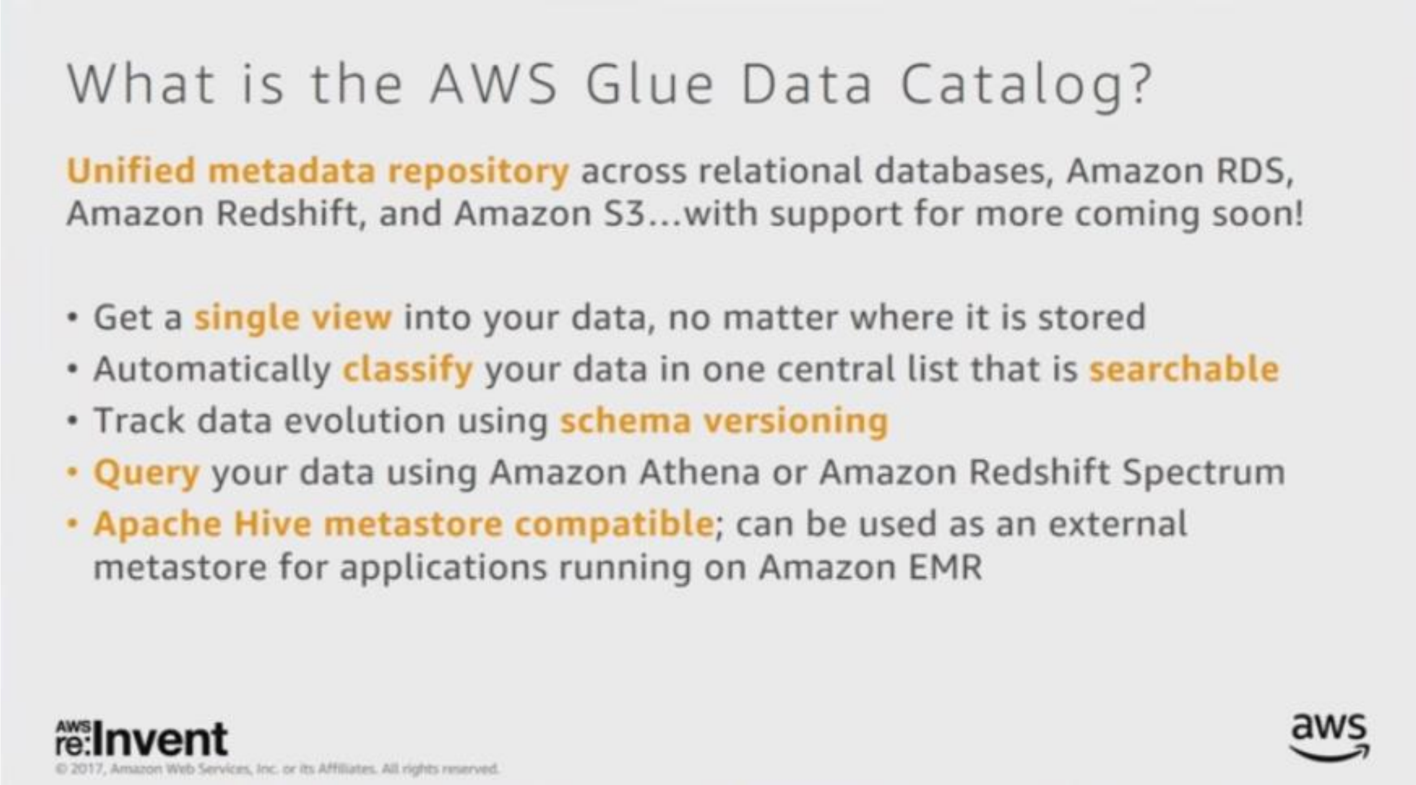
- Auto-generates ETL code
- Python and Apache Spark
- Edit, debug, and share

Job Execution

Deploy

- Serverless execution
- Flexible scheduling
- Monitoring and alerting

re:Invent

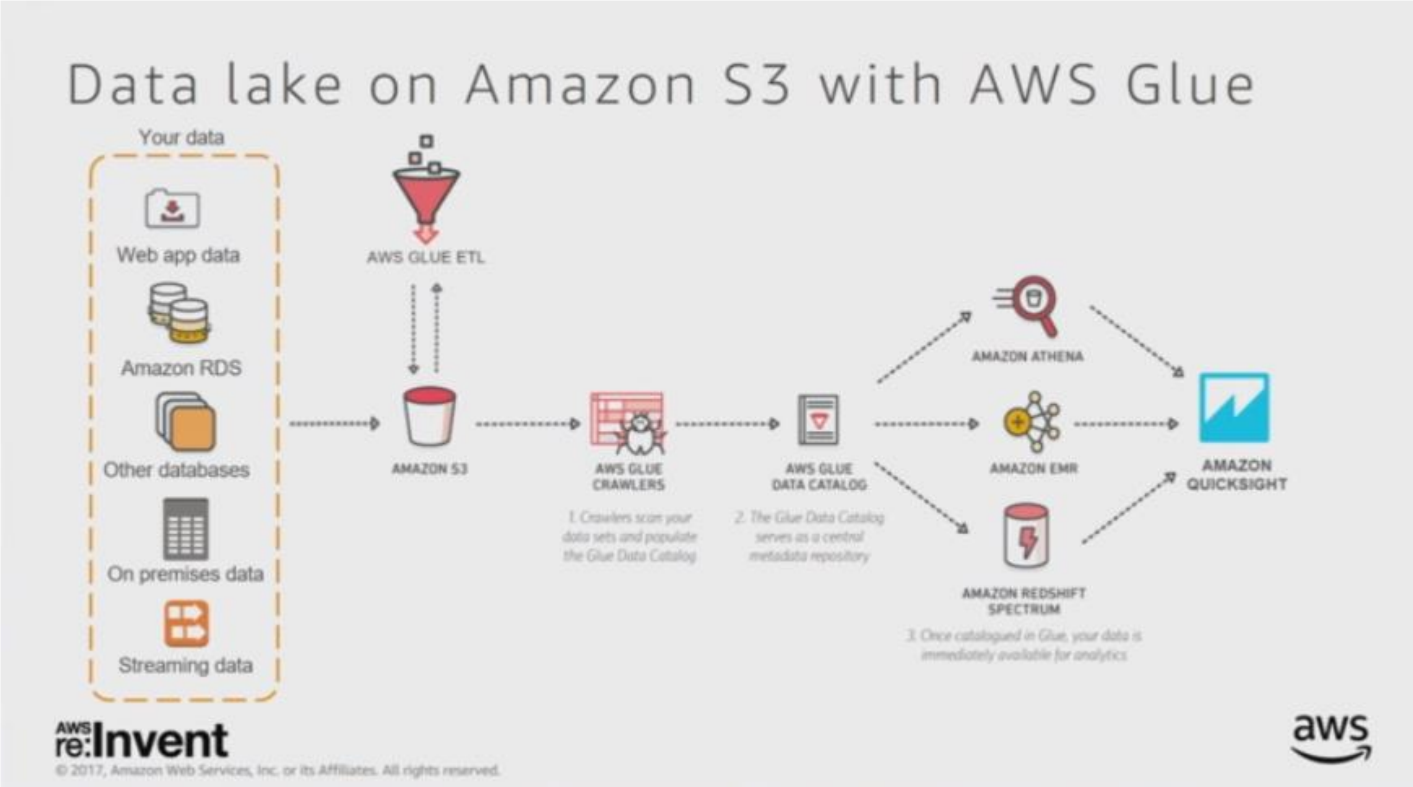


What is the AWS Glue Data Catalog?

Unified metadata repository across relational databases, Amazon RDS, Amazon Redshift, and Amazon S3...with support for more coming soon!

- Get a **single view** into your data, no matter where it is stored
- Automatically **classify** your data in one central list that is **searchable**
- Track data evolution using **schema versioning**
- **Query** your data using Amazon Athena or Amazon Redshift Spectrum
- **Apache Hive metastore compatible**; can be used as an external metastore for applications running on Amazon EMR

re:Invent



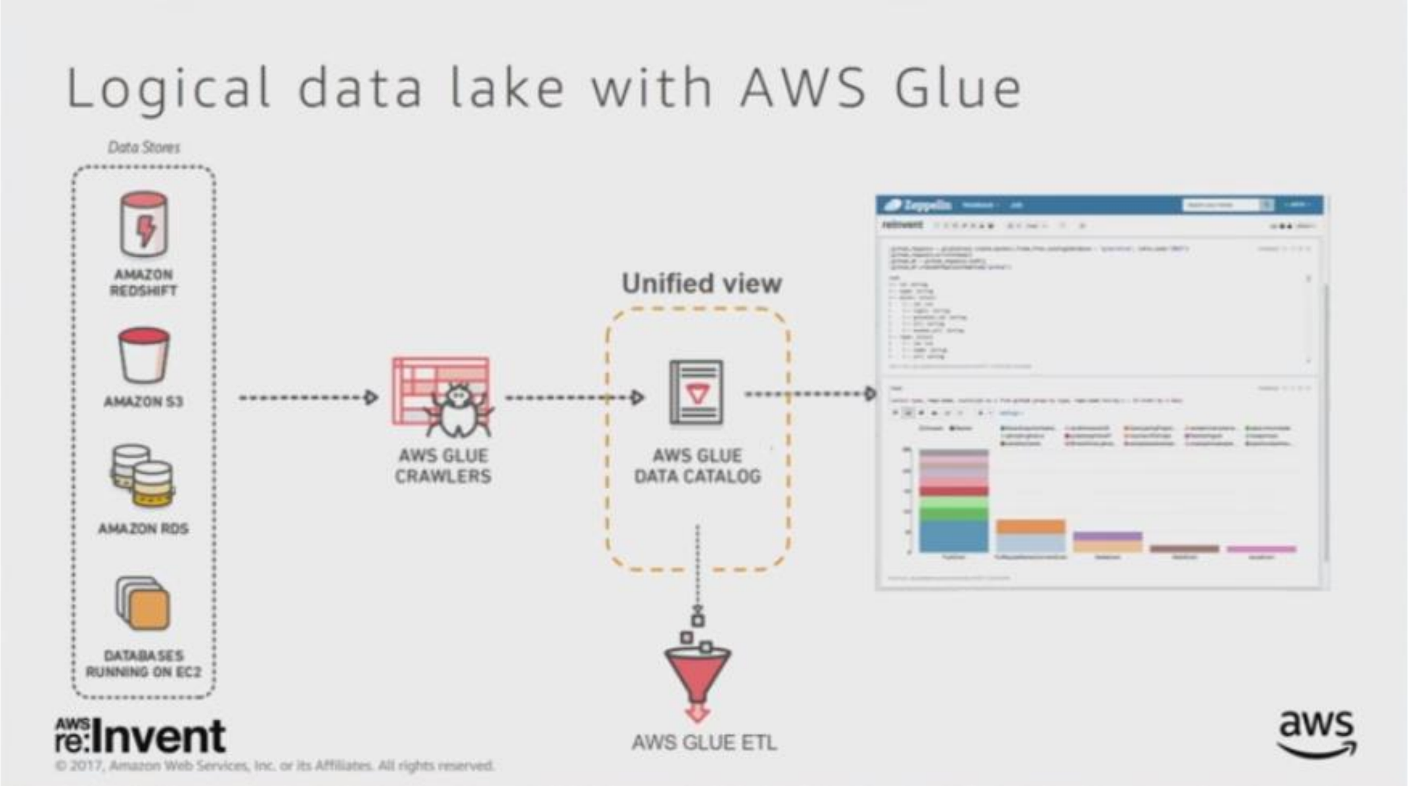
Data lake on Amazon S3 with AWS Glue

1. Crawlers scan your data sets and populate the Glue Data Catalog

2. The Glue Data Catalog serves as a central metadata repository

3. Once catalogued in Glue, your data is immediately available for analytics

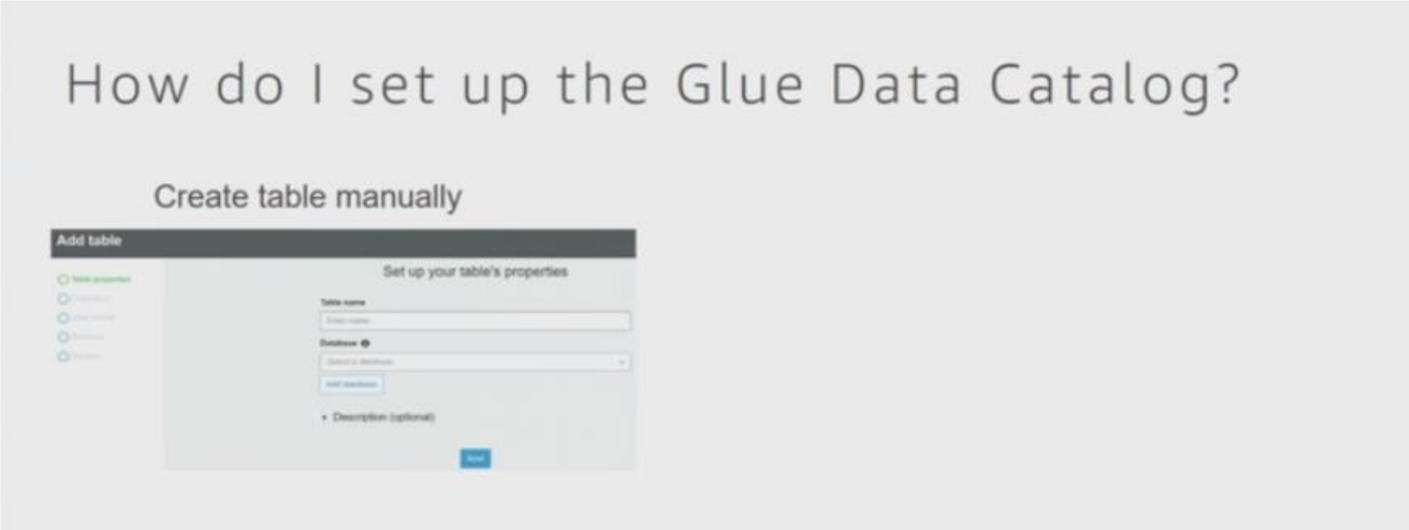
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



Logical data lake with AWS Glue

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

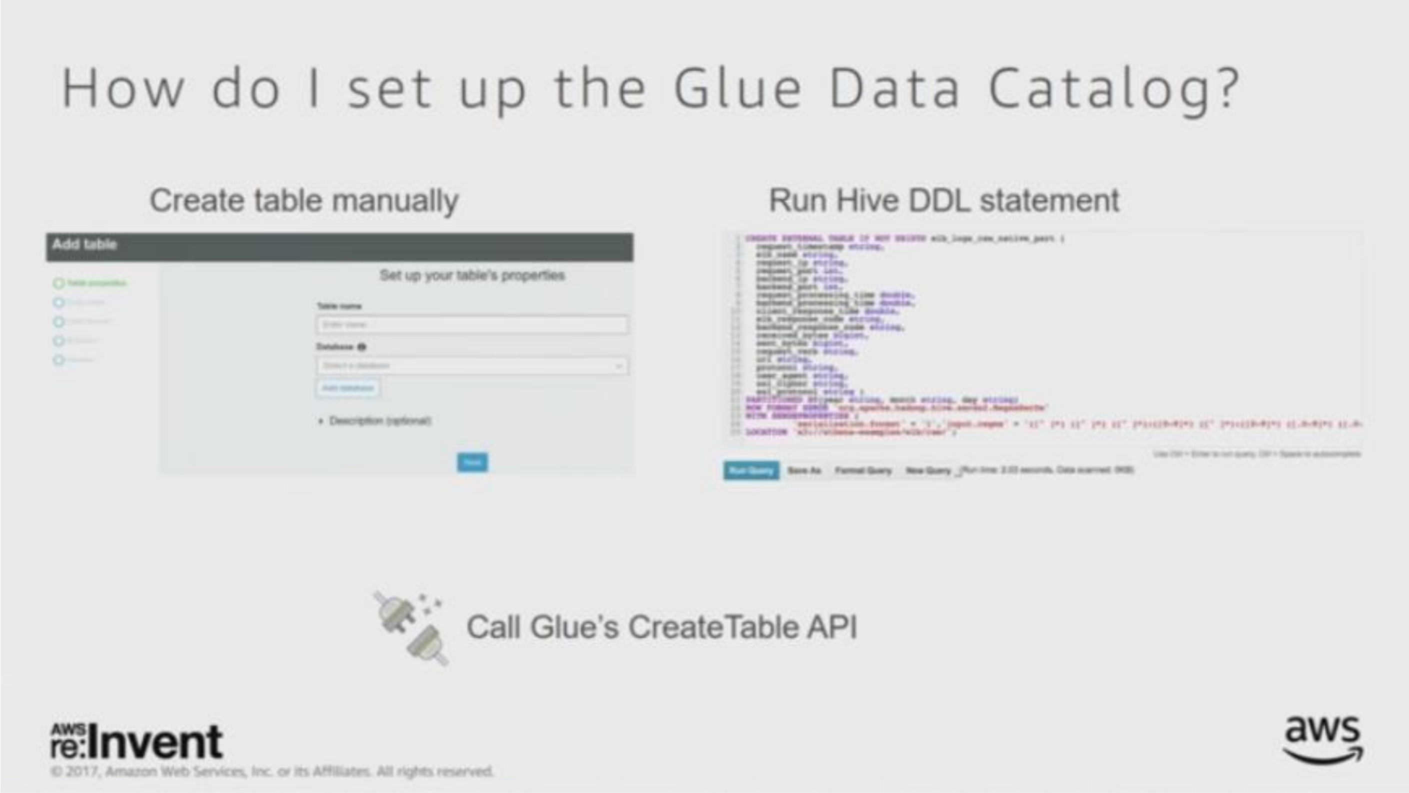
AWS Glue supports using Zepellin Notebook that connects to the Data Catalog. We can spin up a Zepellin Notebook and connect it to the Glue Serverless environment, then read and write data from various data sources by leveraging the metadata that is in the Data Catalog.



How do I set up the Glue Data Catalog?

Create table manually

This is good for quick testing if you know the schema



How do I set up the Glue Data Catalog?

Create table manually

Run Hive DDL statement

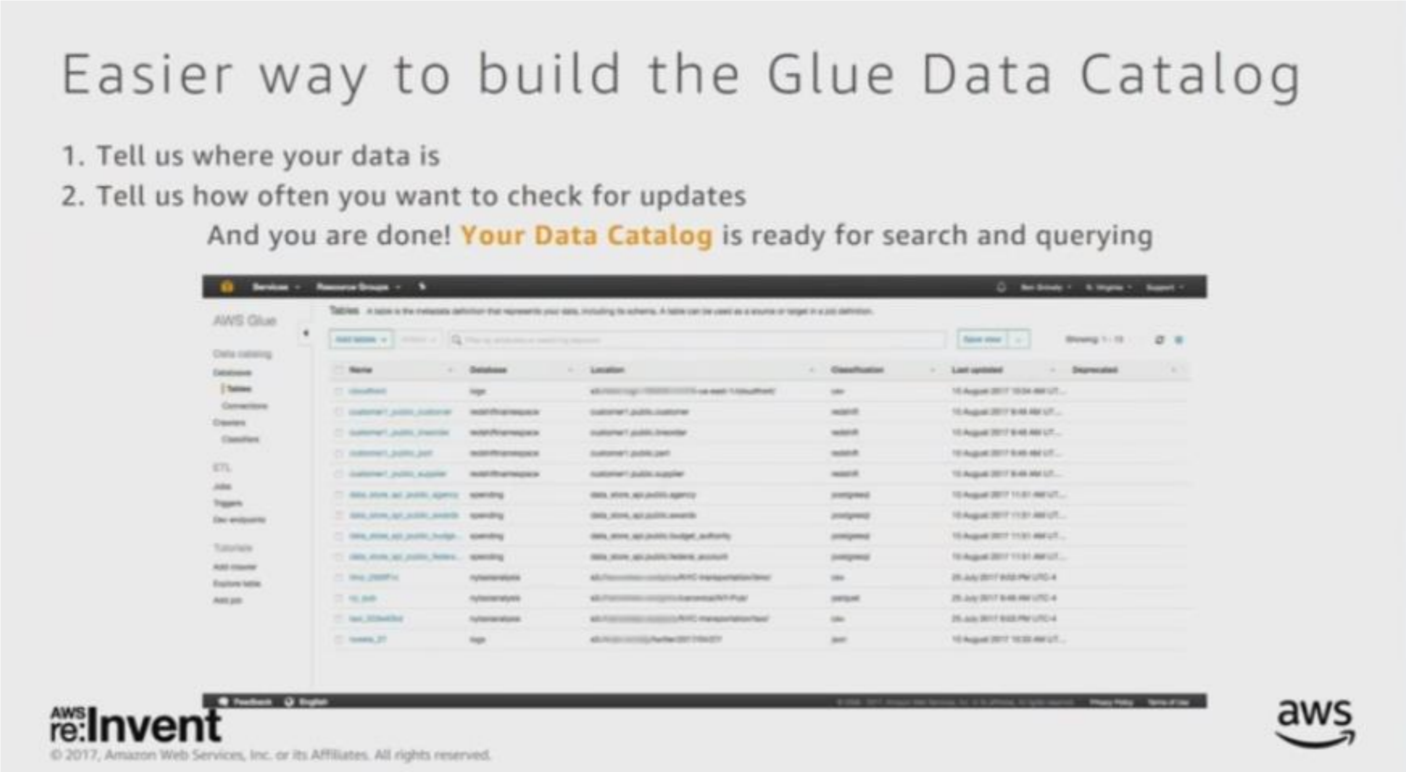
```
CREATE EXTERNAL TABLE IF NOT EXISTS elb_logs_raw_native_part (
  request_timestamp string,
  elb_name string,
  request_ip string,
  request_port int,
  backend_ip string,
  backend_port int,
  request_processing_time double,
  backend_processing_time double,
  client_response_time double,
  elb_response_code string,
  backend_response_code string,
  received_bytes bigint,
  sent_bytes bigint,
  request_verb string,
  url string,
  protocol string,
  user_agent string,
  ssl_cipher string,
  ssl_protocol string )
PARTITIONED BY(year string, month string, day string)
ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.RegexSerDe'
WITH SERDEPROPERTIES (
         'serialization.format' = '1','input.regex' = '([^ ]*) ([^ ]*) ([^ ]*):([0-9]*) ([^ ]*):([0-9]*) ([.0-9]*) ([.0-
LOCATION 's3://athena-examples/elb/raw/';
```

Use Ctrl + Enter to run query, Ctrl + Space to autocomplete

(Run time: 2.03 seconds, Data scanned: 0KB)

Call Glue's CreateTable API

This works great if you have a handful of tables to use.



Easier way to build the Glue Data Catalog

1. Tell us where your data is
2. Tell us how often you want to check for updates

And you are done! **Your Data Catalog** is ready for search and querying

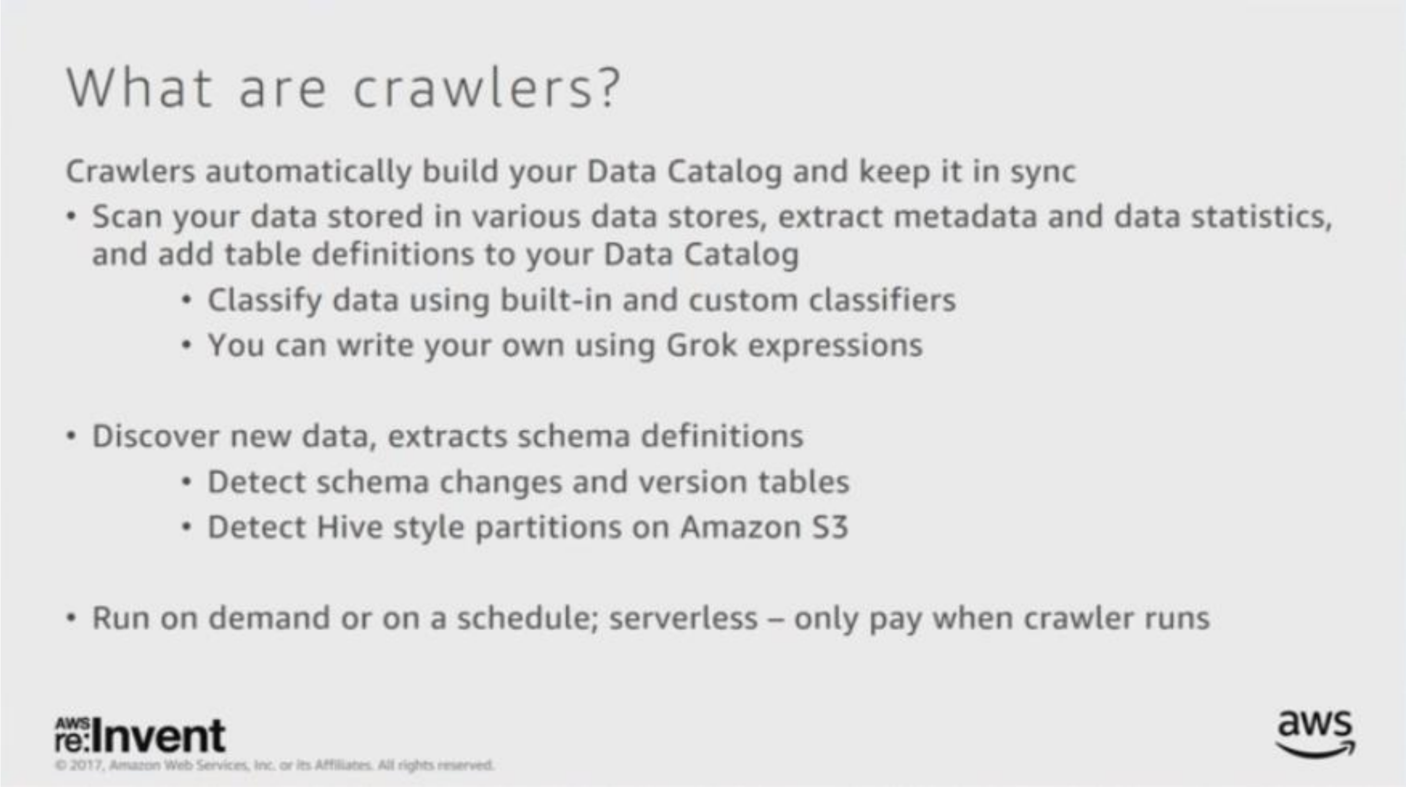
Tables A table is the metadata definition that represents your data, including its schema. A table can be used as a source or target in a job definition.

Name	Database	Location	Classification	Last updated	Deprecated
cloudfront	logs	s3://[illegible]-us-east-1/cloudfront/	csv	10 August 2017 10:04 AM UT...	
customer1_public_customer	redshiftnamespace	customer1.public.customer	redshift	10 August 2017 9:48 AM UT...	
customer1_public_lineorder	redshiftnamespace	customer1.public.lineorder	redshift	10 August 2017 9:48 AM UT...	
customer1_public_part	redshiftnamespace	customer1.public.part	redshift	10 August 2017 9:48 AM UT...	
customer1_public_supplier	redshiftnamespace	customer1.public.supplier	redshift	10 August 2017 9:48 AM UT...	
data_store_api_public_agency	spending	data_store_api.public.agency	postgresql	10 August 2017 11:01 AM UT...	
data_store_api_public_awards	spending	data_store_api.public.awards	postgresql	10 August 2017 11:01 AM UT...	
data_store_api_public_budge...	spending	data_store_api.public.budget_authority	postgresql	10 August 2017 11:01 AM UT...	
data_store_api_public_federa...	spending	data_store_api.public.federal_account	postgresql	10 August 2017 11:01 AM UT...	
[illegible]	nytaxianalysis	s3://[illegible]/NYC-transportation/[illegible]/	csv	25 July 2017 8:02 PM UTC-4	
[illegible]	nytaxianalysis	s3://[illegible]/NY-Pub/	parquet	25 July 2017 9:48 AM UTC-4	
taxi_[illegible]	nytaxianalysis	s3://[illegible]/NYC-transportation/taxi/	csv	25 July 2017 8:03 PM UTC-4	
tweets_[illegible]	logs	s3://[illegible]/twitter/[illegible]/	json	10 August 2017 10:33 AM UT...	

AWS re:Invent

aws

You can simply tell Glue where your data is coming from and it will automatically generate a table for you using that source, then it will keep the table data up to date for you to run jobs on. We can have tables that are mapped to data sources in S3, from Redshift, data warehouse, PostGres DBs.



What are crawlers?

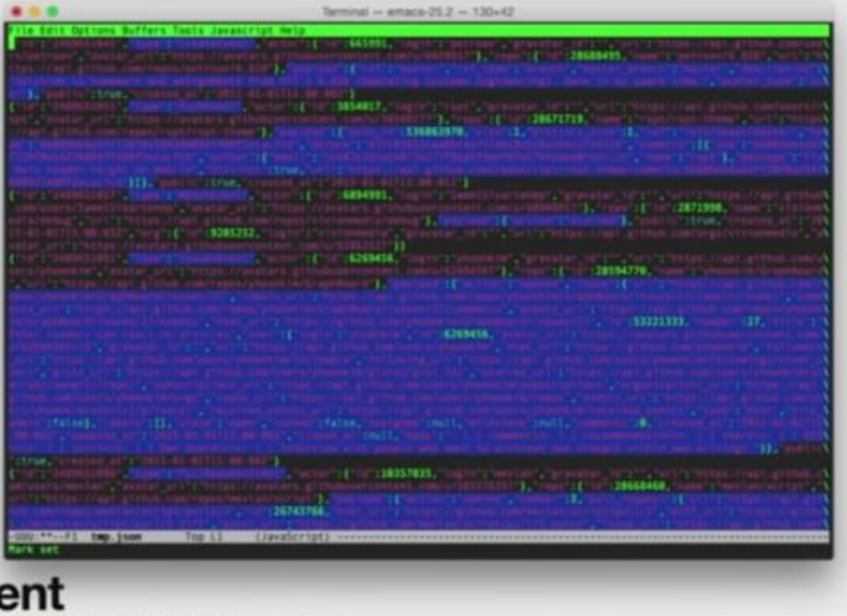
Crawlers automatically build your Data Catalog and keep it in sync

- Scan your data stored in various data stores, extract metadata and data statistics, and add table definitions to your Data Catalog
  - Classify data using built-in and custom classifiers
  - You can write your own using Grok expressions
- Discover new data, extracts schema definitions
  - Detect schema changes and version tables
  - Detect Hive style partitions on Amazon S3
- Run on demand or on a schedule; serverless – only pay when crawler runs

AWS re:Invent

aws

GitHub timeline data


Terminal — emacs-25.2 — 130×42
File Edit Options Buffers Tools Javascript Help
tmp.json
Top L1
(JavaScript)
Mark set

githubarchive.org

20+ event types

unique payload per event type

AWS re:Invent

aws

A table in the Glue Data Catalog

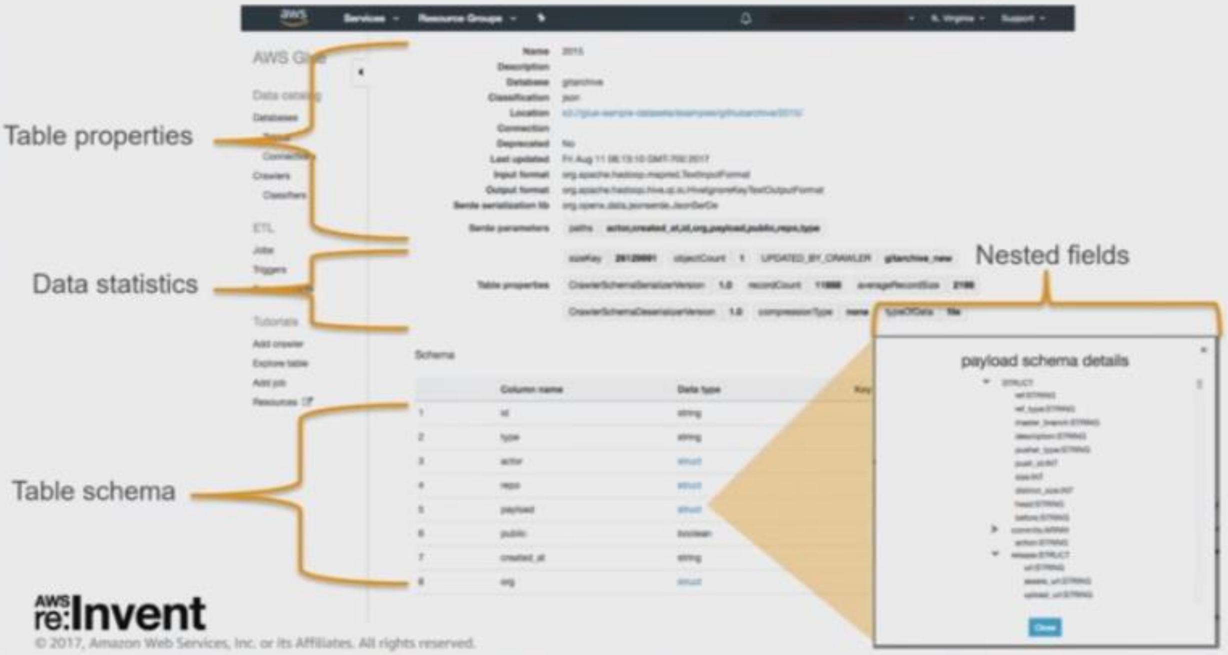


Table properties

Data statistics

Table schema

Nested fields

Name	2015
Description	
Database	githubarchive
Classification	json
Location	s3://glue-sample-datasets/examples/githubarchive/2015/
Connection	
Deprecated	No
Last updated	Fri Aug 11 08:13:10 GMT-700 2017
Input format	org.apache.hadoop.mapred.TextInputFormat
Output format	org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
Serde serialization lib	org.openx.data.jsonserde.JsonSerDe
Serde parameters	paths actor,created_at,id,org,payload,public,repo,type
Table properties	sizeKey 28129991 objectCount 1 UPDATED_BY_CRAWLER githubarchive_new CrawlerSchemaSerializerVersion 1.0 recordCount 11988 averageRecordSize 2198 CrawlerSchemaDeserializerVersion 1.0 compressionType none typeOfData file

Schema

	Column name	Data type
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

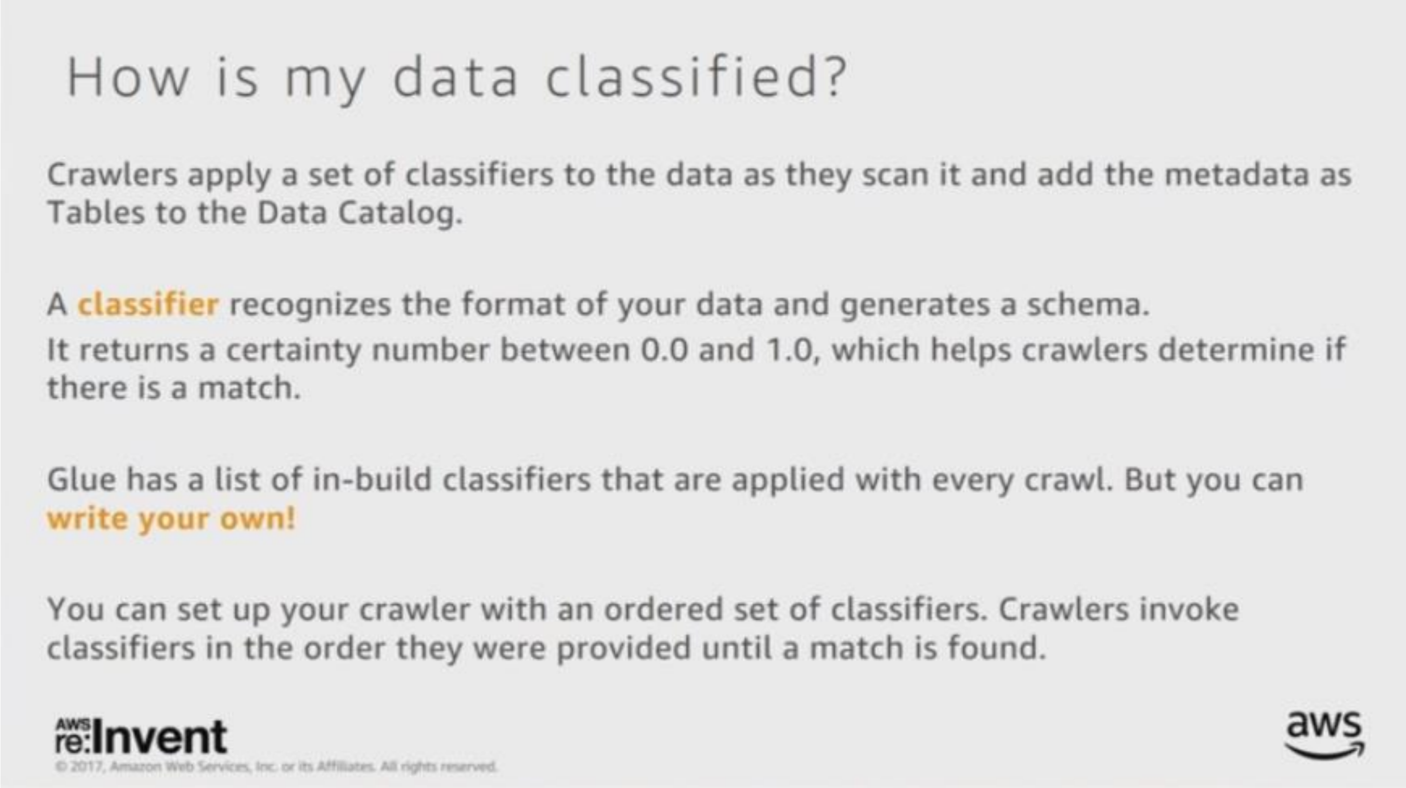
payload schema details

```
▾ STRUCT
    ref:STRING
    ref_type:STRING
    master_branch:STRING
    description:STRING
    pusher_type:STRING
    push_id:INT
    size:INT
    distinct_size:INT
    head:STRING
    before:STRING
  ▸ commits:ARRAY
    action:STRING
  ▾ release:STRUCT
      url:STRING
      assets_url:STRING
      upload_url:STRING
```

AWS re:Invent

aws

When Glue crawls the dataset above, it generates the above table entries in the Data Catalog as above.



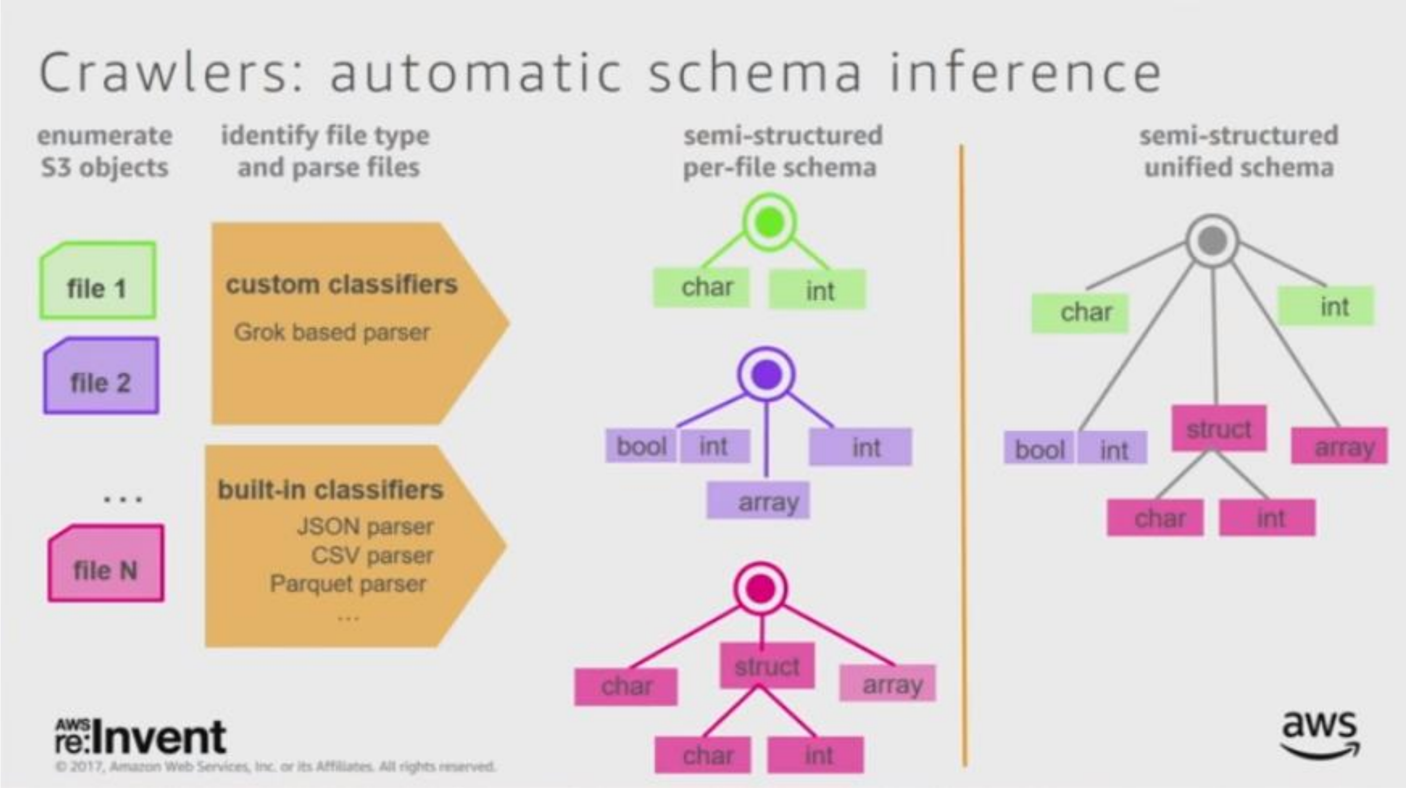
How is my data classified?

Crawlers apply a set of classifiers to the data as they scan it and add the metadata as Tables to the Data Catalog.

A **classifier** recognizes the format of your data and generates a schema.
It returns a certainty number between 0.0 and 1.0, which helps crawlers determine if there is a match.

Glue has a list of in-build classifiers that are applied with every crawl. But you can **write your own!**

You can set up your crawler with an ordered set of classifiers. Crawlers invoke classifiers in the order they were provided until a match is found.



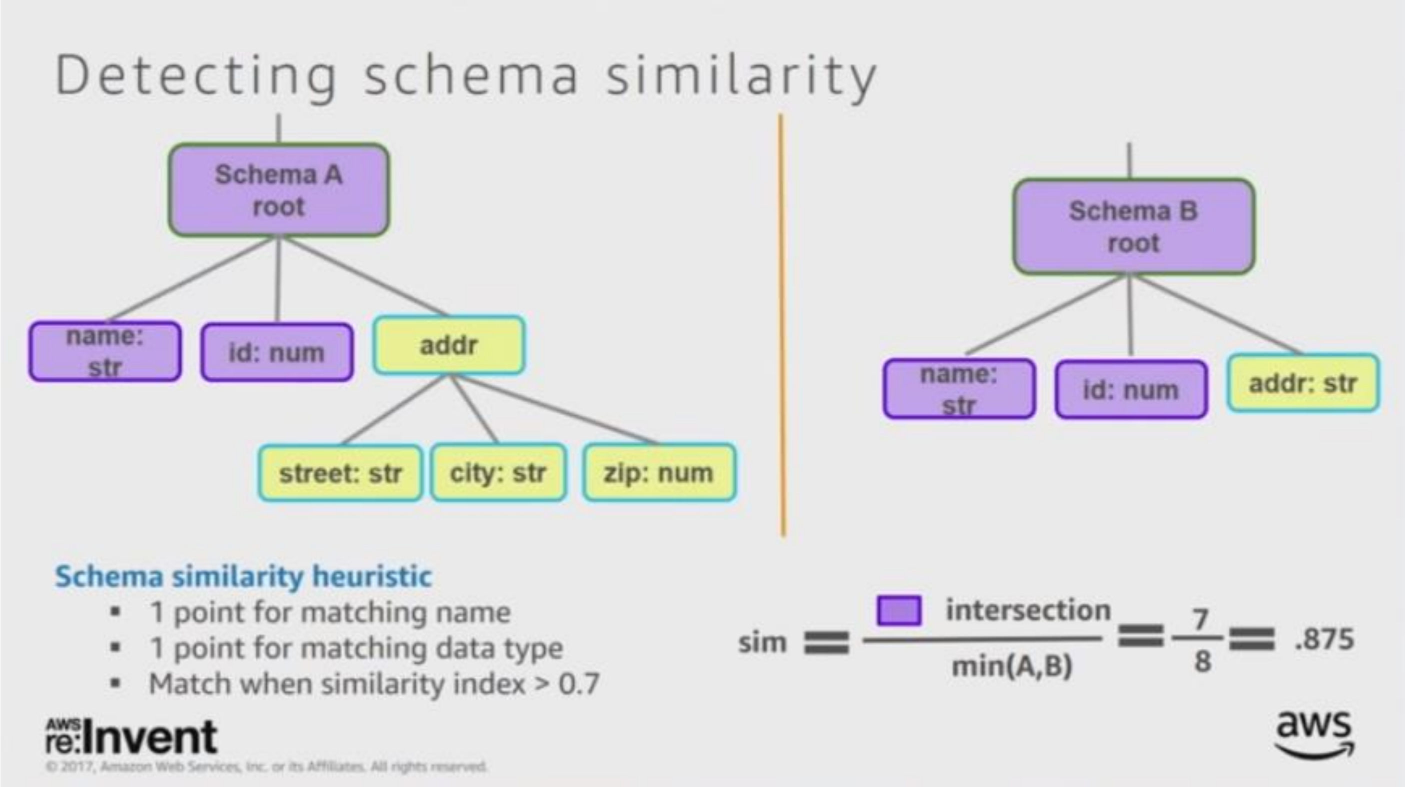
Crawlers: automatic schema inference

enumerate S3 objects

identify file type and parse files

semi-structured per-file schema

semi-structured unified schema



Detecting schema similarity

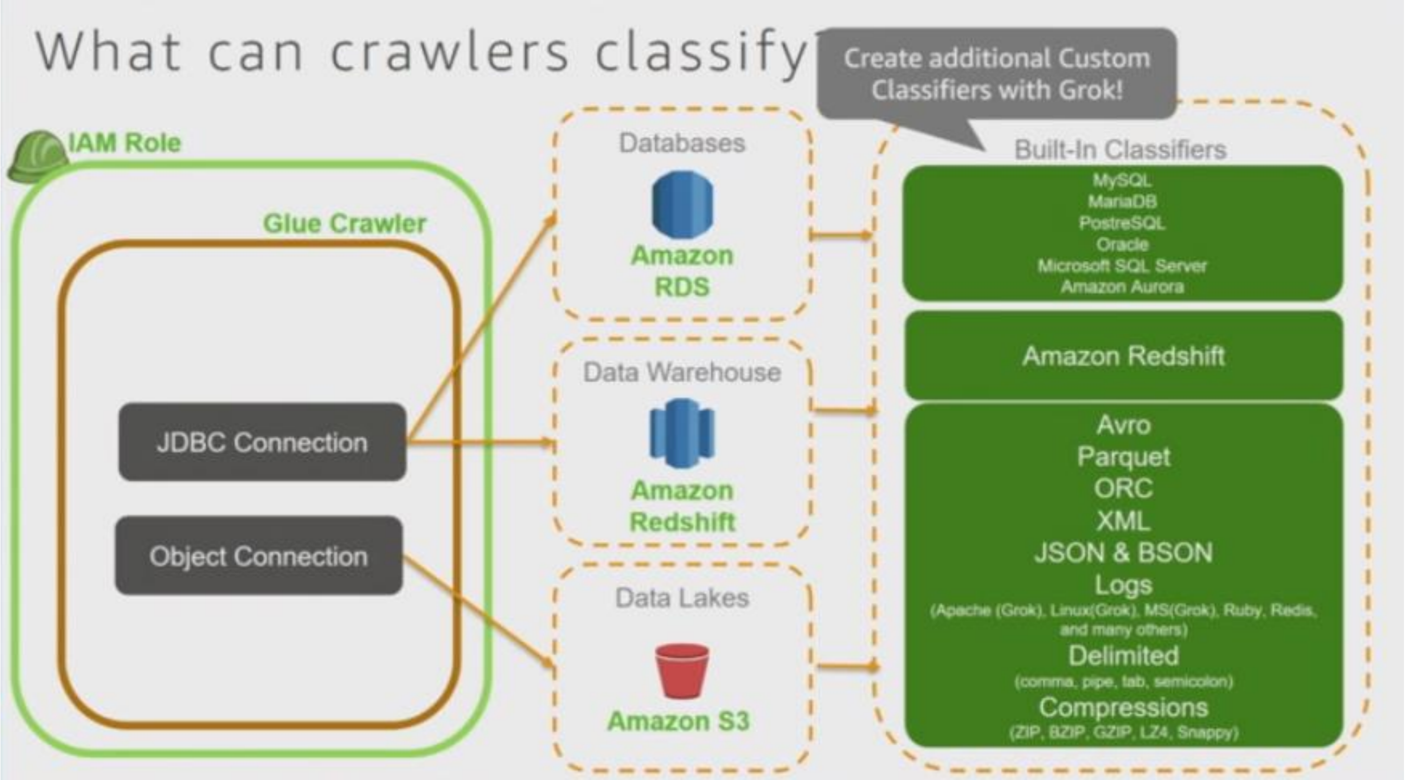
Schema similarity heuristic

- 1 point for matching name
- 1 point for matching data type
- Match when similarity index > 0.7

$$\text{sim} = \frac{■ \text{ intersection}}{\min(A,B)} = \frac{7}{8} = .875$$

AWS re:Invent

aws



What can crawlers classify?

Create additional Custom Classifiers with Grok!

MySQL
MariaDB
PostreSQL
Oracle
Microsoft SQL Server
Amazon Aurora

Amazon Redshift

Avro
Parquet
ORC
XML
JSON & BSON
Logs
(Apache (Grok), Linux(Grok), MS(Grok), Ruby, Redis, and many others)
Delimited
(comma, pipe, tab, semicolon)
Compressions
(ZIP, BZIP, GZIP, LZ4, Snappy)

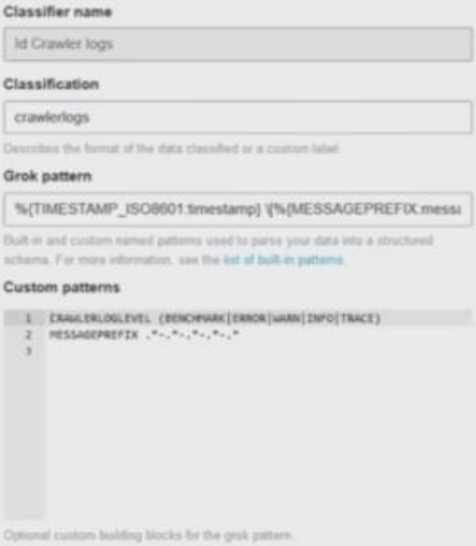
How can I write my own classifiers?

You can write a custom classifier by providing a Grok pattern and a classification string for the matched schema.

A Grok pattern is a named set of regular expressions (regex) that are used to match data one line at a time.

Example:

```
%{TIMESTAMP_ISO8601:timestamp}
\[%{MESSAGEPREFIX:message_prefix}\]
%{CRAWLERLOGLEVEL:loglevel} :
%{GREEDYDATA:message}
```



Classifier name

Id Crawler logs

Classification

crawlerlogs

Describes the format of the data classified or a custom label

Grok pattern

%{TIMESTAMP_ISO8601:timestamp} \[%{MESSAGEPREFIX:messa

Built-in and custom named patterns used to parse your data into a structured schema. For more information, see the list of built-in patterns

Custom patterns

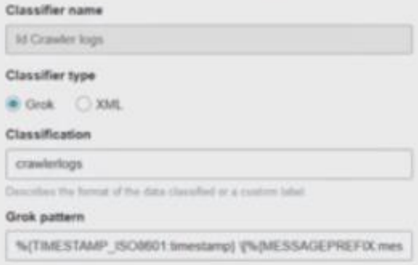
```
CRAWLERLOGLEVEL (BENCHMARK|ERROR|WARN|INFO|TRACE)
MESSAGEPREFIX .*-.*-.*-.*-.*
```

Optional custom building blocks for the grok pattern

AWS re:Invent

Custom classifiers

1. Write a custom classifier



Classifier name

Id Crawler logs

Classifier type

● Grok ○ XML

Classification

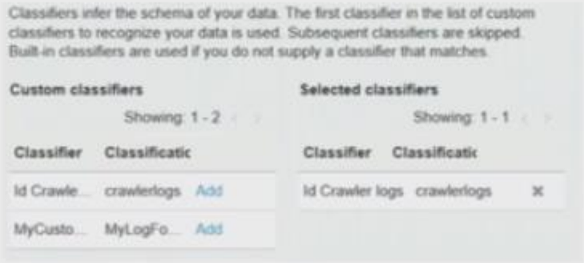
crawlerlogs

Describes the format of the data classified or a custom label

Grok pattern

%{TIMESTAMP_ISO8601:timestamp} \[%{MESSAGEPREFIX:mes

2. Add it to your crawler



Classifiers infer the schema of your data. The first classifier in the list of custom classifiers to recognize your data is used. Subsequent classifiers are skipped. Built-in classifiers are used if you do not supply a classifier that matches.

Custom classifiers

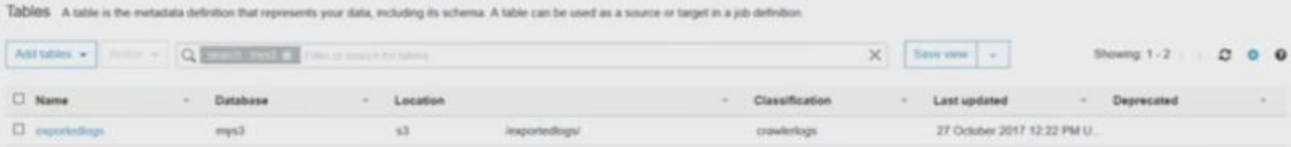
Showing: 1 - 2

Classifier	Classificatic	
Id Crawle...	crawlerlogs	Add
MyCusto...	MyLogFo...	Add

Selected classifiers

Showing: 1 - 1

Classifier	Classificatic	
Id Crawler logs	crawlerlogs	✕

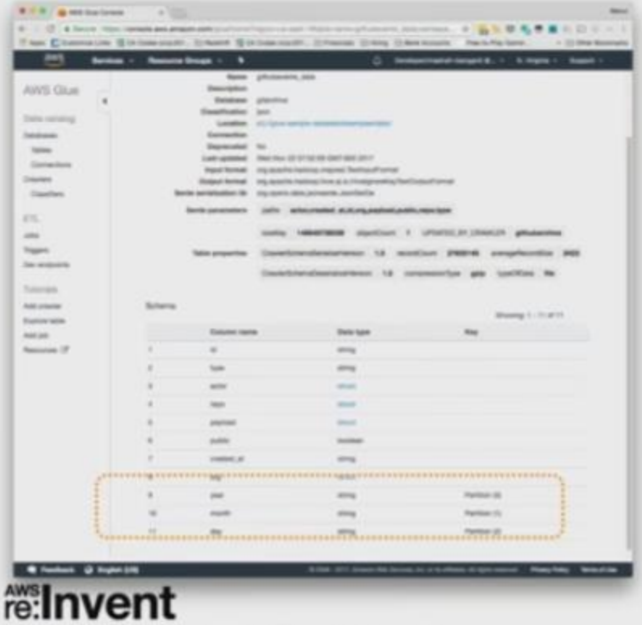


Tables A table is the metadata definition that represents your data, including its schema. A table can be used as a source or target in a job definition.

	Name	Database	Location		Classification	Last updated	Deprecated
□	exportedlogs	mys3	s3	/exportedlogs/	crawlerlogs	27 October 2017 12:22 PM U...	

AWS re:Invent

Automatically detected partitions

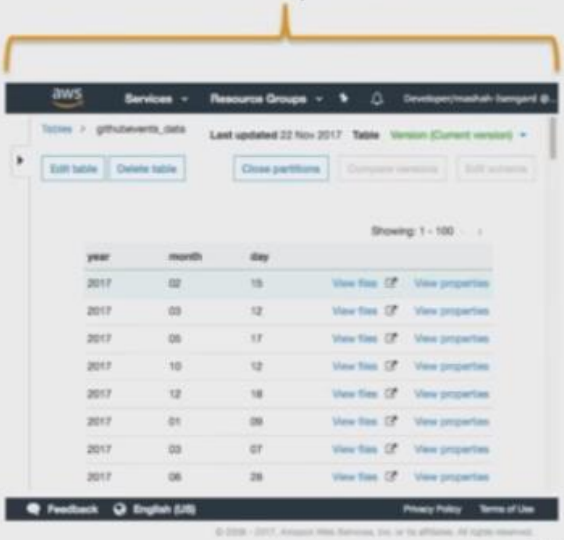


AWS Glue

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	[illegible]	[illegible]	
9	year	string	Partition (0)
10	month	string	Partition (1)
11	day	string	Partition (2)

Available partitions



Last updated 22 Nov 2017 Table Version (Current version)

Showing: 1 - 100

year	month	day		
2017	02	15	View files	View properties
2017	03	12	View files	View properties
2017	05	17	View files	View properties
2017	10	12	View files	View properties
2017	12	18	View files	View properties
2017	01	09	View files	View properties
2017	03	07	View files	View properties
2017	06	28	View files	View properties

How are partitions detected?

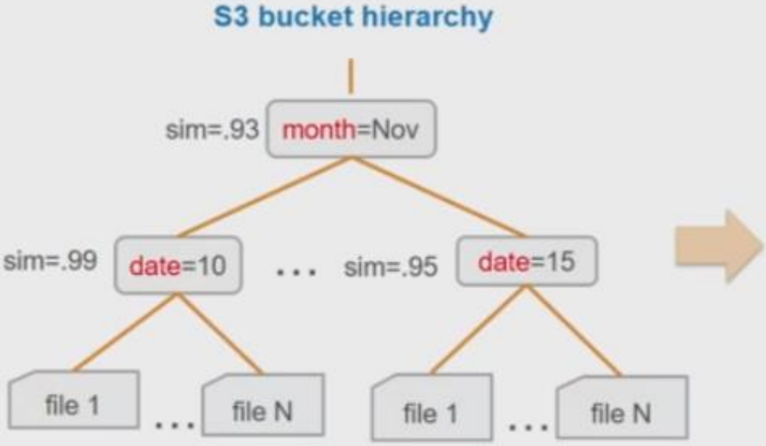


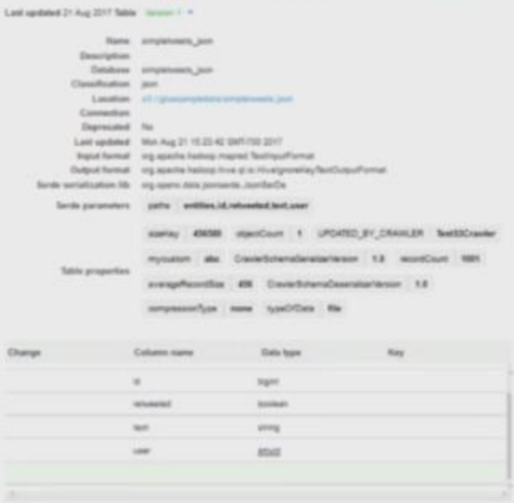
Table definition

Column	Type
month	str
date	str
col 1	int
	float
⋮	⋮

Estimate schema similarity among files at each level to handle semi-structured logs, schema evolution…

Automatic schema versioning

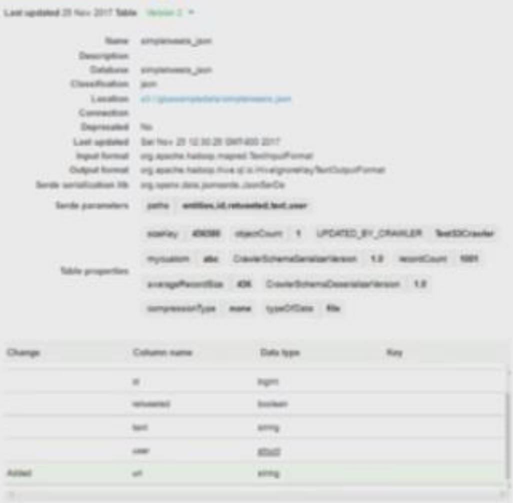
Automatically update table version as data evolves



Last updated 21 Aug 2017 Table Version 1 ▾

Name simpletweets_json
Description
Database simpletweets_json
Classification json
Location s3://gluesampledata/simpletweets_json
Connection
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.ql.io.HiveIgnoreKeyTextOutputFormat
Serde serialization lib org.openx.data.jsonserde.JsonSerDe
Serde parameters paths entities,id,retweeted,text,user
Table properties sizeKey 456580 objectCount 1 UPDATED_BY_CRAWLER TestS3Crawler myCustom abc CrawlerSchemaSerializerVersion 1.0 recordCount 1001 averageRecordSize 456 CrawlerSchemaDeserializerVersion 1.0 compressionType none typeOfData file

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



Last updated 25 Nov 2017 Table Version 2 ▾

Name simpletweets_json
Description
Database simpletweets_json
Classification json
Location s3://gluesampledata/simpletweets_json
Connection
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.ql.io.HiveIgnoreKeyTextOutputFormat
Serde serialization lib org.openx.data.jsonserde.JsonSerDe
Serde parameters paths entities,id,retweeted,text,user
Table properties sizeKey 456580 objectCount 1 UPDATED_BY_CRAWLER TestS3Crawler myCustom abc CrawlerSchemaSerializerVersion 1.0 recordCount 1001 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	

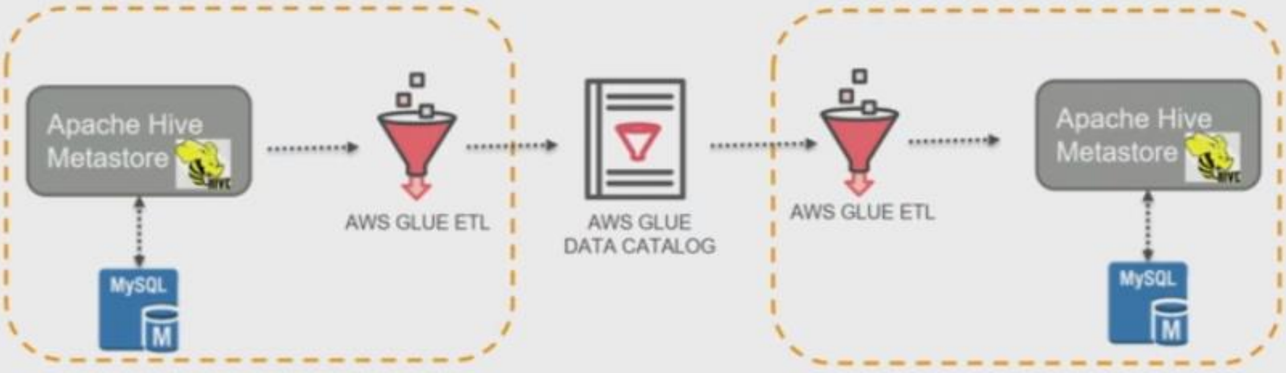
AWS re:Invent

aws

Import/Export your metadata

Import from an external metastore

Export to an external metastore

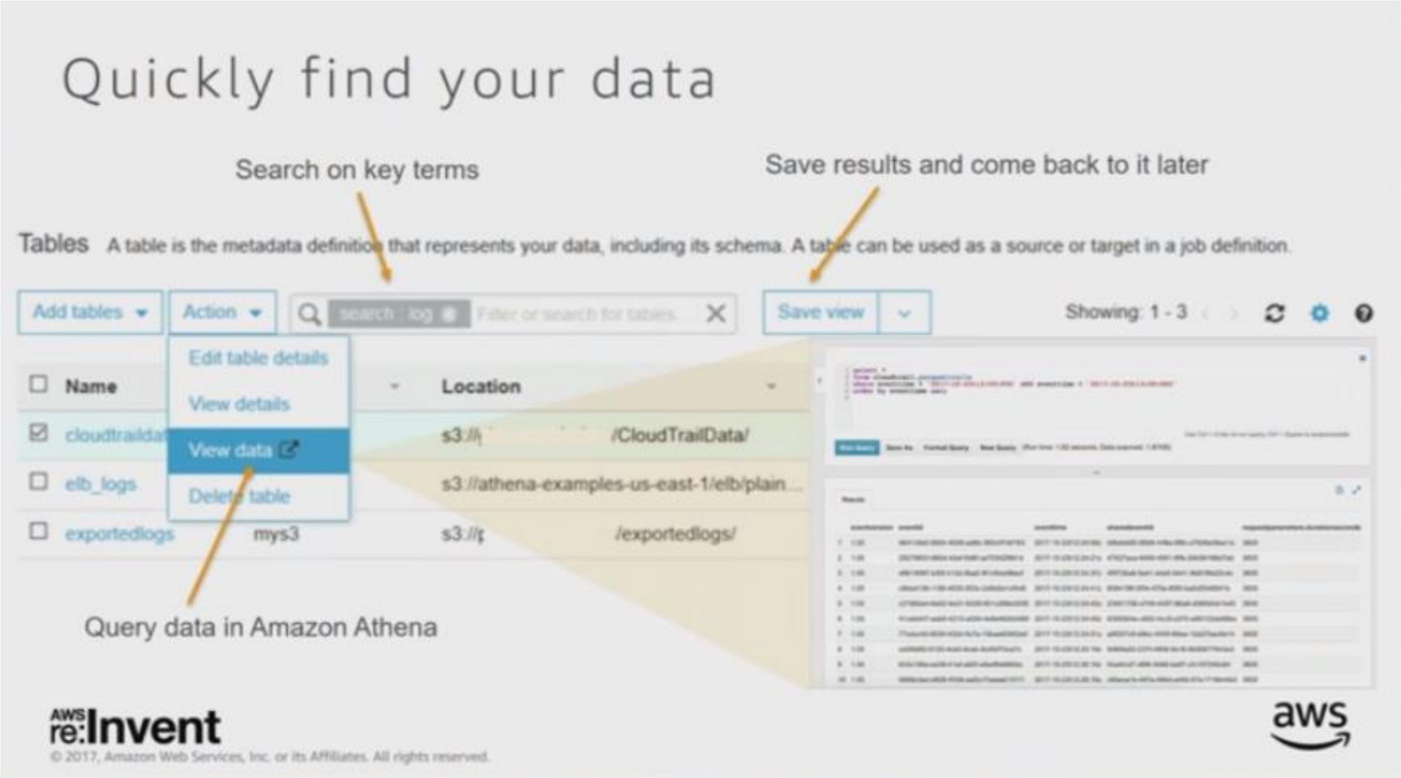


Find the import/export ETL script on Glue's GitHub repository

AWS re:Invent

aws

Your data is catalogued...what's next?



Quickly find your data

Search on key terms

Save results and come back to it later

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.

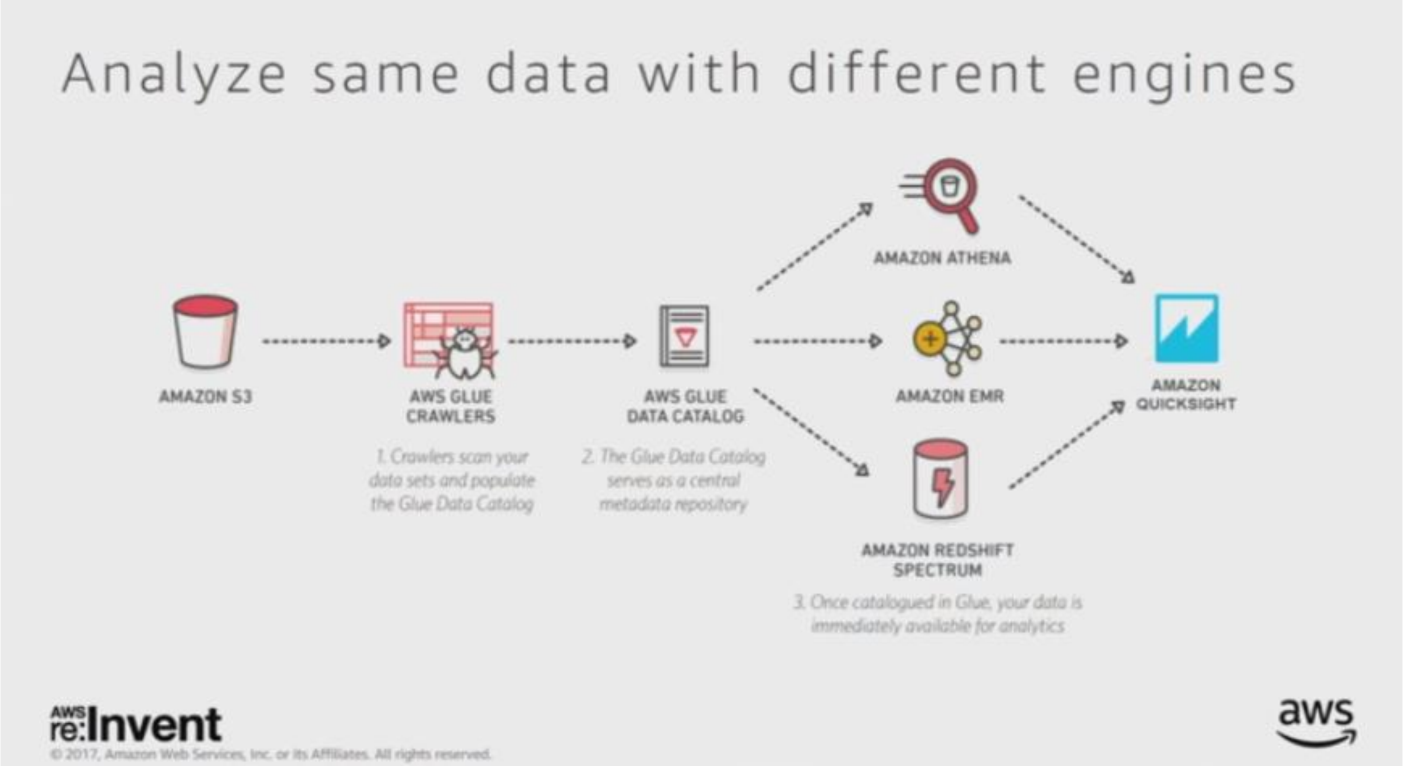
☐ Name		Location
☑ cloudtraildat		s3:// /CloudTrailData/
☐ elb_logs		s3://athena-examples-us-east-1/elb/plain...
☐ exportedlogs	mys3	s3:// /exportedlogs/

Query data in Amazon Athena

AWS re:Invent

aws

You can do text based search and also filter on key attributes from the Glue console to find relevant datasets. You can also start using this in queries with Athena.



Analyze same data with different engines

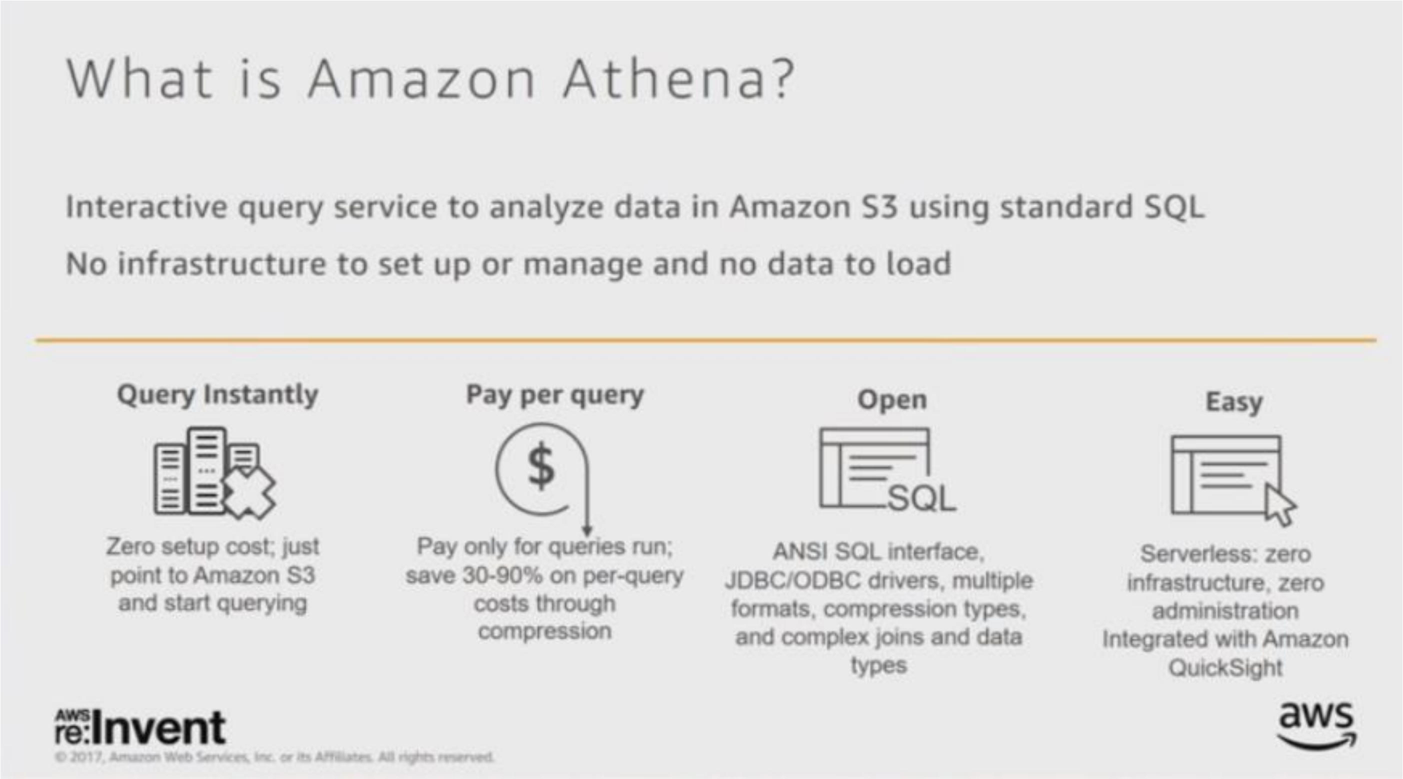
1. Crawlers scan your data sets and populate the Glue Data Catalog

2. The Glue Data Catalog serves as a central metadata repository

3. Once catalogued in Glue, your data is immediately available for analytics

AWS re:Invent

aws



What is Amazon Athena?

Interactive query service to analyze data in Amazon S3 using standard SQL

No infrastructure to set up or manage and no data to load

Query Instantly

Zero setup cost; just point to Amazon S3 and start querying

Pay per query

Pay only for queries run; save 30-90% on per-query costs through compression

Open

ANSI SQL interface, JDBC/ODBC drivers, multiple formats, compression types, and complex joins and data types

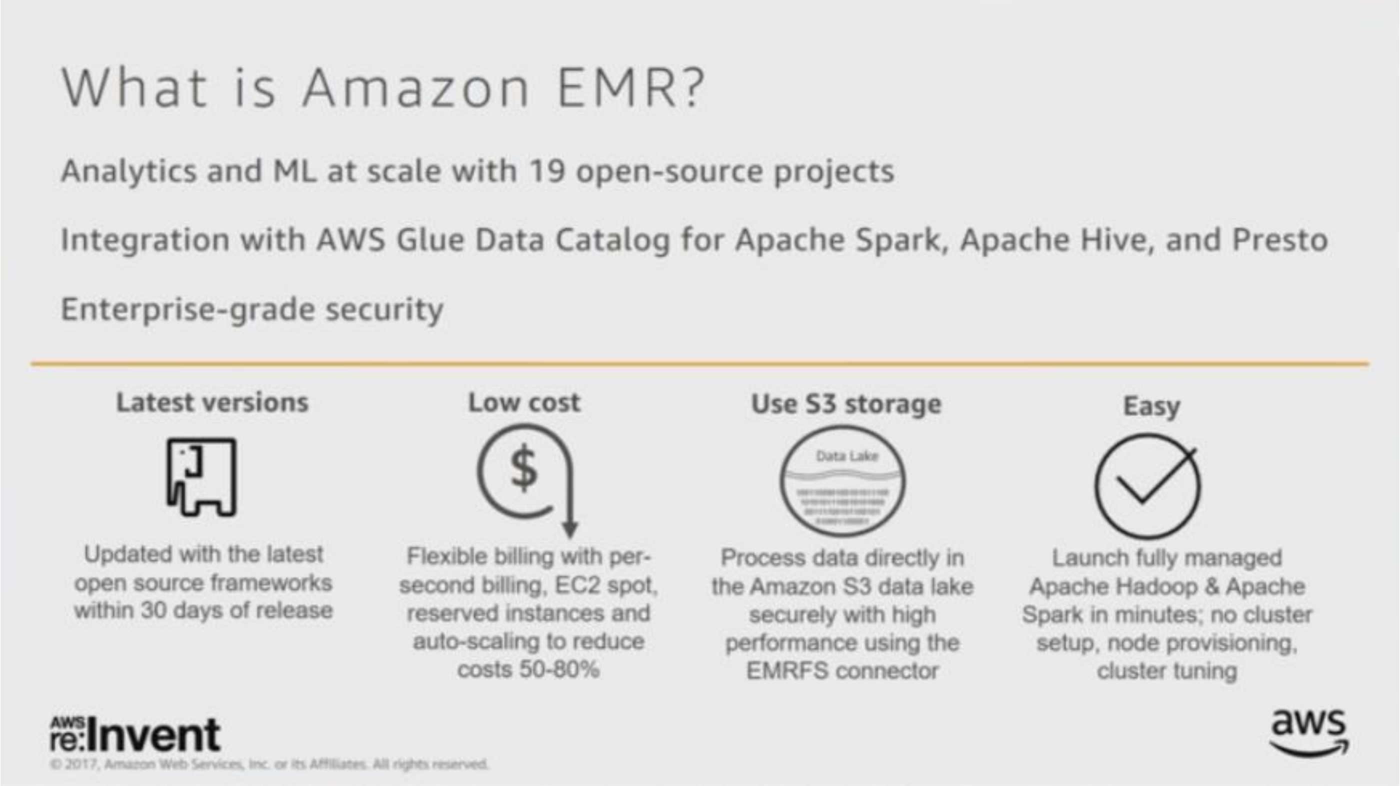
Easy

Serverless: zero infrastructure, zero administration

Integrated with Amazon QuickSight

AWS re:Invent

aws



What is Amazon EMR?

Analytics and ML at scale with 19 open-source projects

Integration with AWS Glue Data Catalog for Apache Spark, Apache Hive, and Presto

Enterprise-grade security

Latest versions

Updated with the latest open source frameworks within 30 days of release

Low cost

Flexible billing with per-second billing, EC2 spot, reserved instances and auto-scaling to reduce costs 50-80%

Use S3 storage

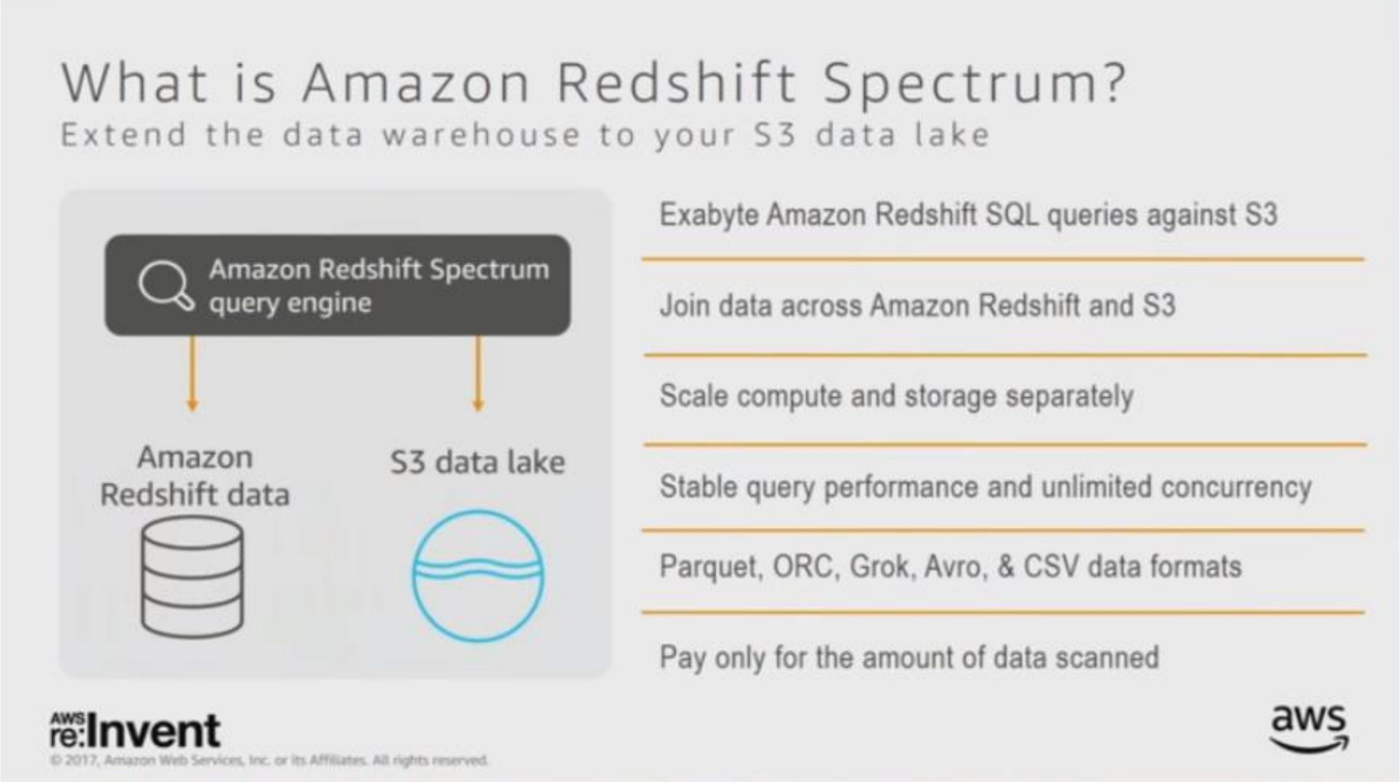
Process data directly in the Amazon S3 data lake securely with high performance using the EMRFS connector

Easy

Launch fully managed Apache Hadoop & Apache Spark in minutes; no cluster setup, node provisioning, cluster tuning

AWS re:Invent

aws



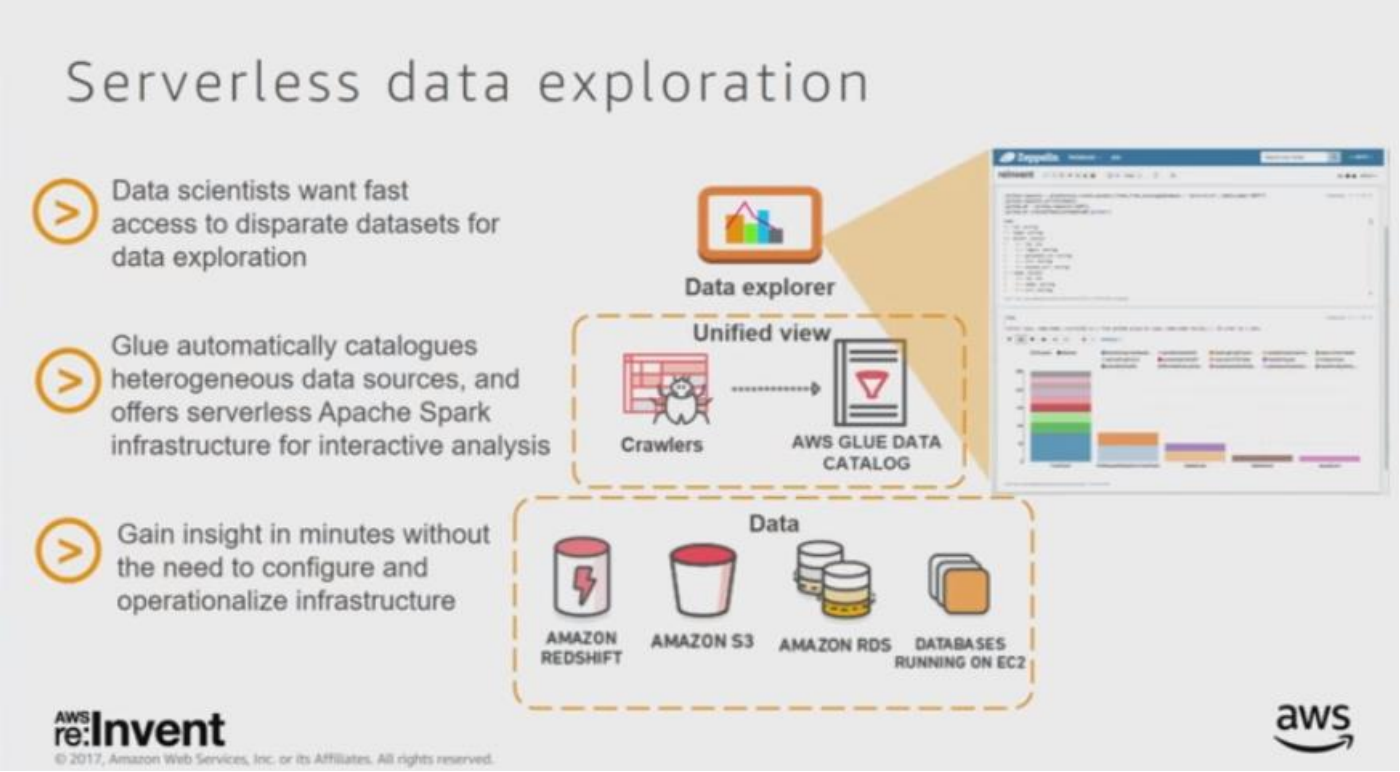
What is Amazon Redshift Spectrum?

Extend the data warehouse to your S3 data lake

- Exabyte Amazon Redshift SQL queries against S3
- Join data across Amazon Redshift and S3
- Scale compute and storage separately
- Stable query performance and unlimited concurrency
- Parquet, ORC, Grok, Avro, & CSV data formats
- Pay only for the amount of data scanned

AWS re:Invent

aws

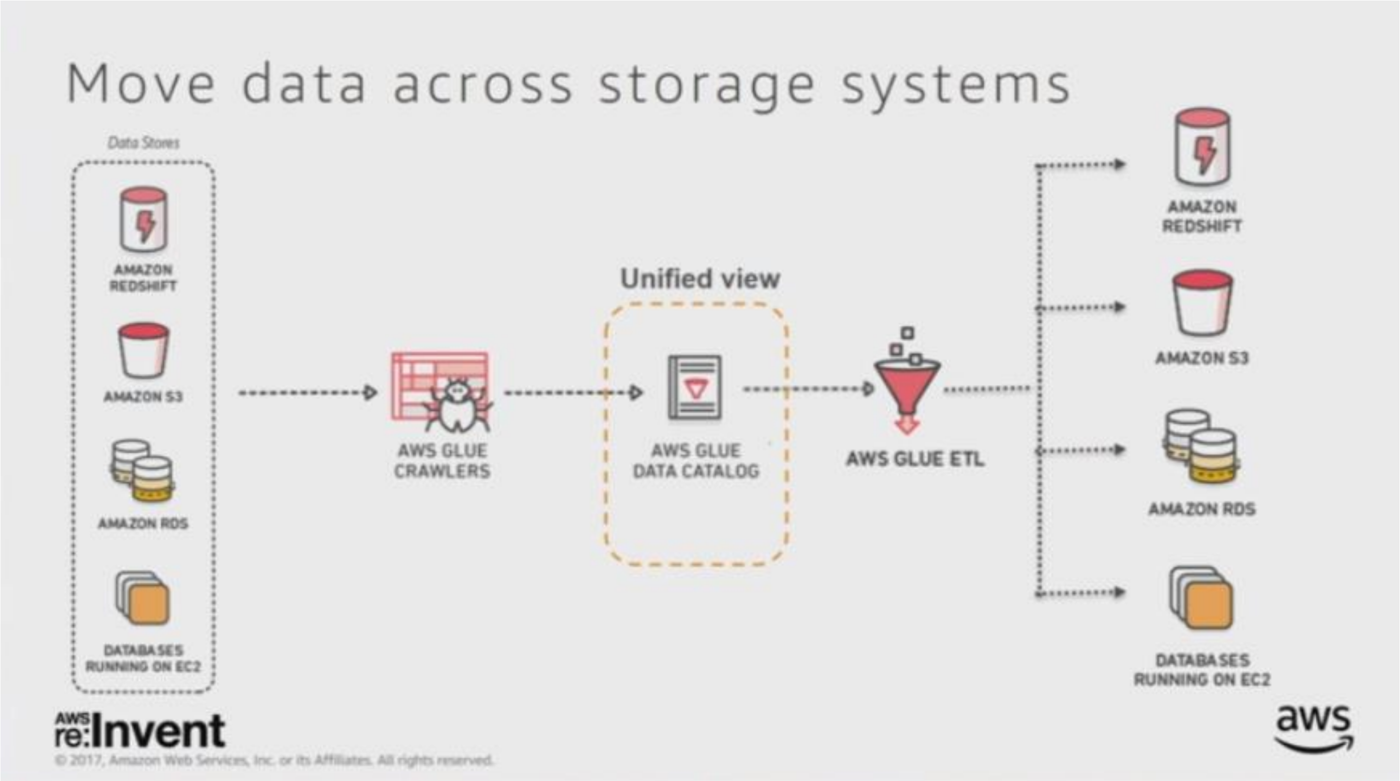


Serverless data exploration

- Data scientists want fast access to disparate datasets for data exploration
- Glue automatically catalogues heterogeneous data sources, and offers serverless Apache Spark infrastructure for interactive analysis
- Gain insight in minutes without the need to configure and operationalize infrastructure

AWS re:Invent

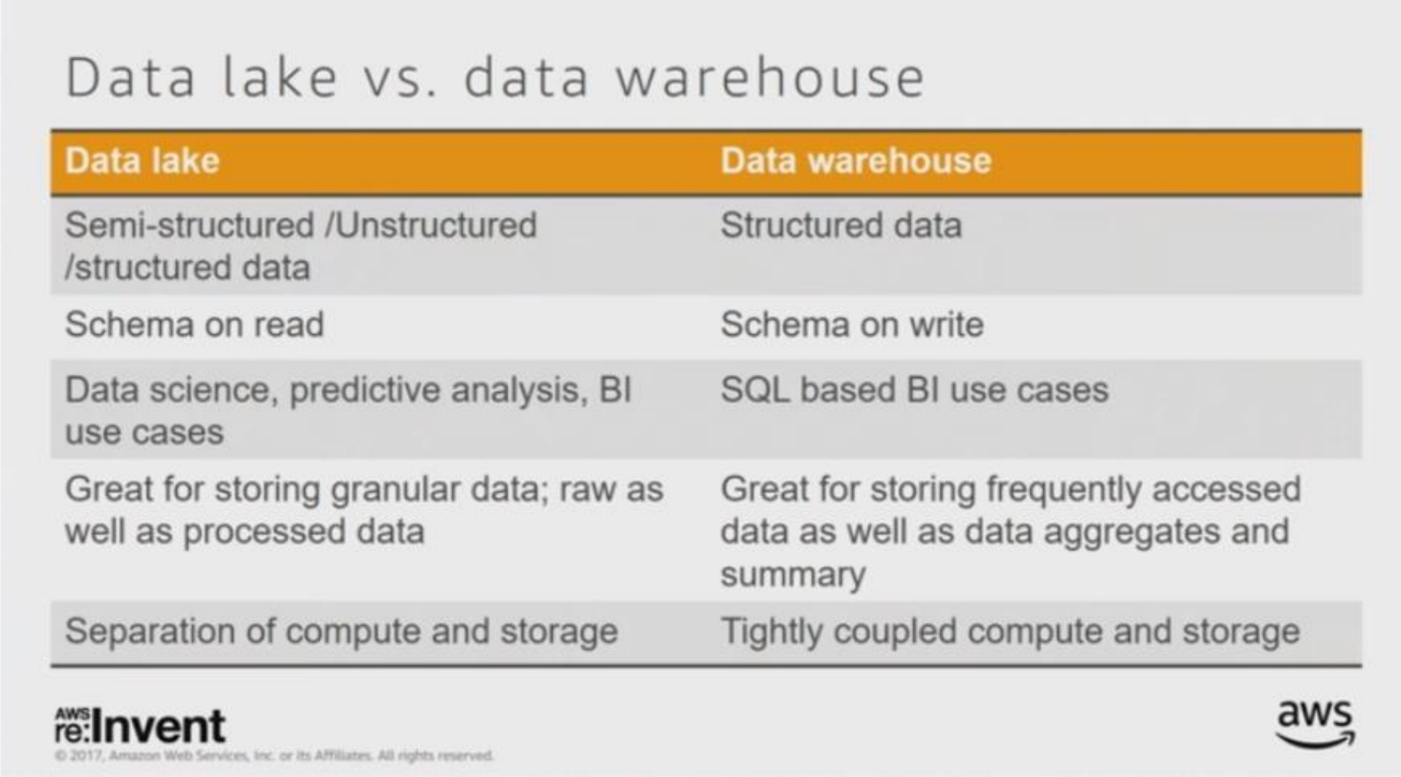
aws



Move data across storage systems

AWS re:Invent

aws

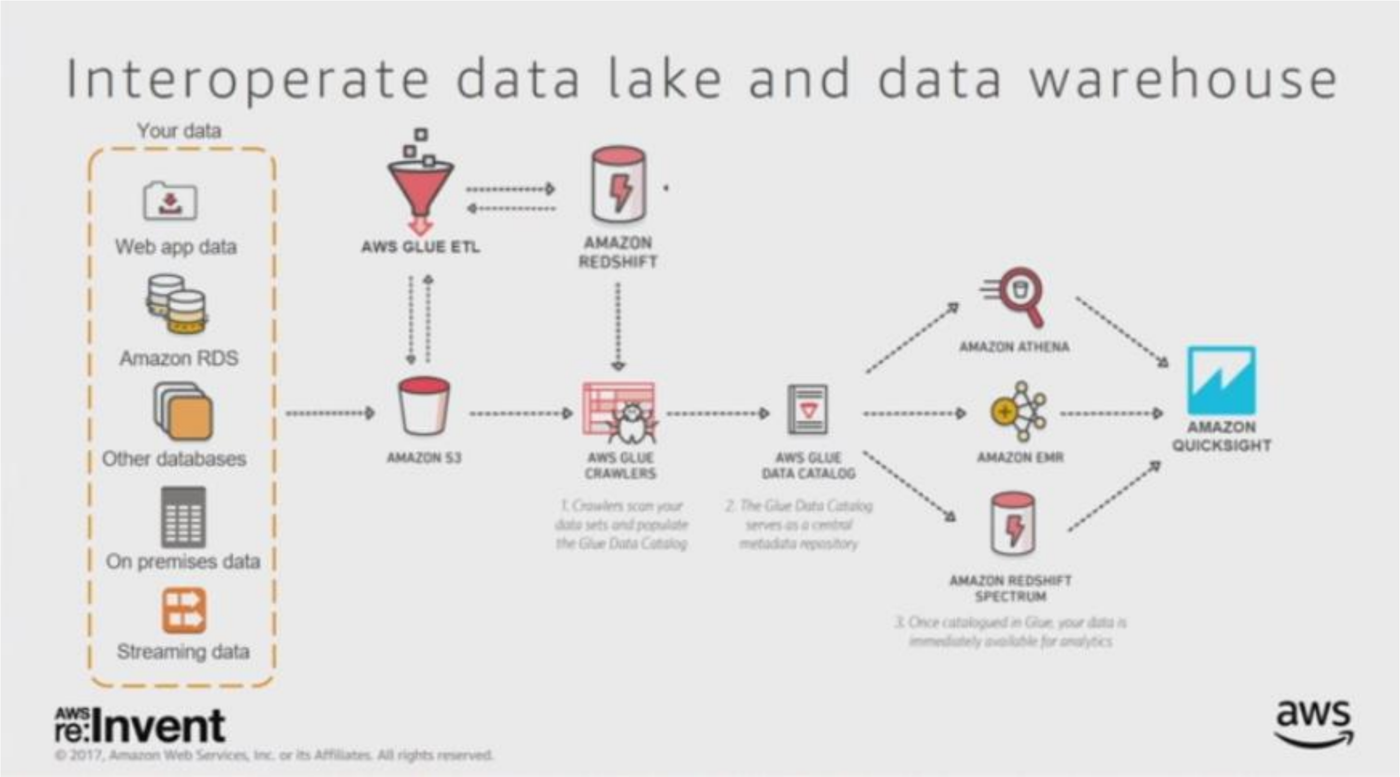


Data lake vs. data warehouse

Data lake	Data warehouse
Semi-structured /Unstructured /structured data	Structured data
Schema on read	Schema on write
Data science, predictive analysis, BI use cases	SQL based BI use cases
Great for storing granular data; raw as well as processed data	Great for storing frequently accessed data as well as data aggregates and summary
Separation of compute and storage	Tightly coupled compute and storage

AWS re:Invent

aws



Interoperate data lake and data warehouse

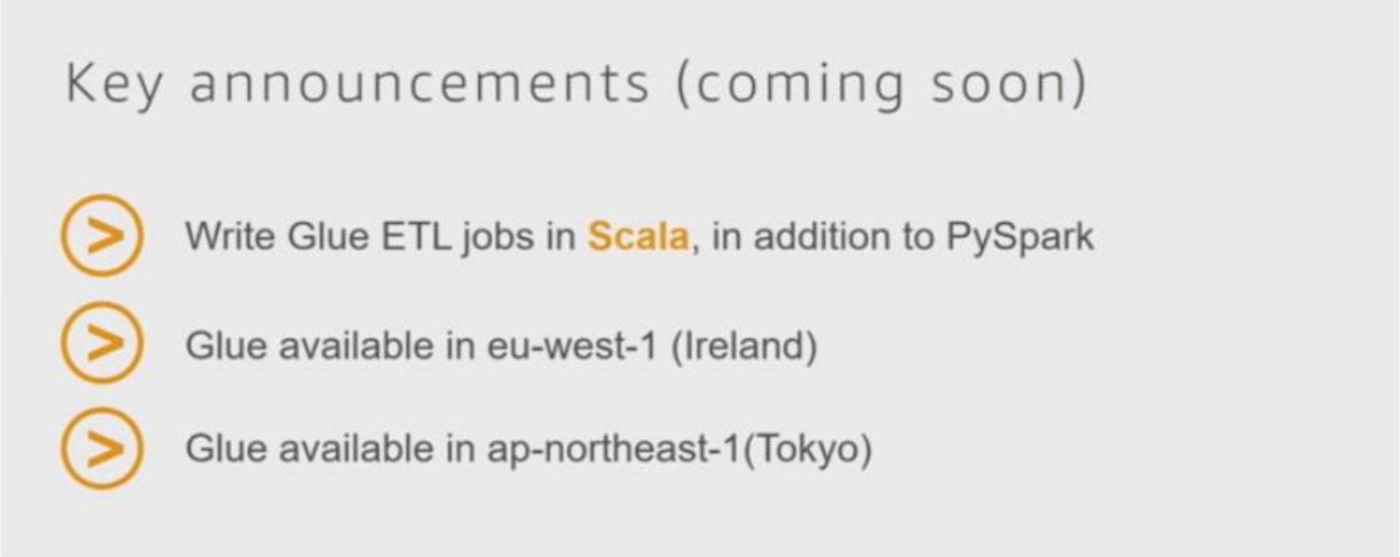
1. Crawlers scan your data sets and populate the Glue Data Catalog

2. The Glue Data Catalog serves as a central metadata repository

3. Once catalogued in Glue, your data is immediately available for analytics

AWS re:Invent

aws



Key announcements (coming soon)

- Write Glue ETL jobs in **Scala**, in addition to PySpark
- Glue available in eu-west-1 (Ireland)
- Glue available in ap-northeast-1(Tokyo)