Tally-demo

Subtitle text

1 AMAZON DATA EXCHANGE

2 S3 Bucket

3

External S3 stages:
tally_s3_stage
tallycovid_s3_stage
tallyevent_s3_stage

Storage Integrations:
tallysnowflake
tallycovidsnowflake
tallyeventsnowflake

COPY INTO

CRYPTOTABLE (CSV)	COVIDTABLE (CSV)	EVENTTABLE (JSON)

DATABSE

Export to Amazon S3

4 CloudTrail

Label

5 CoudWatch / AWS Lambda Function

CloudWatchRuleTally

6 EVENTS

Log Events stored in S3
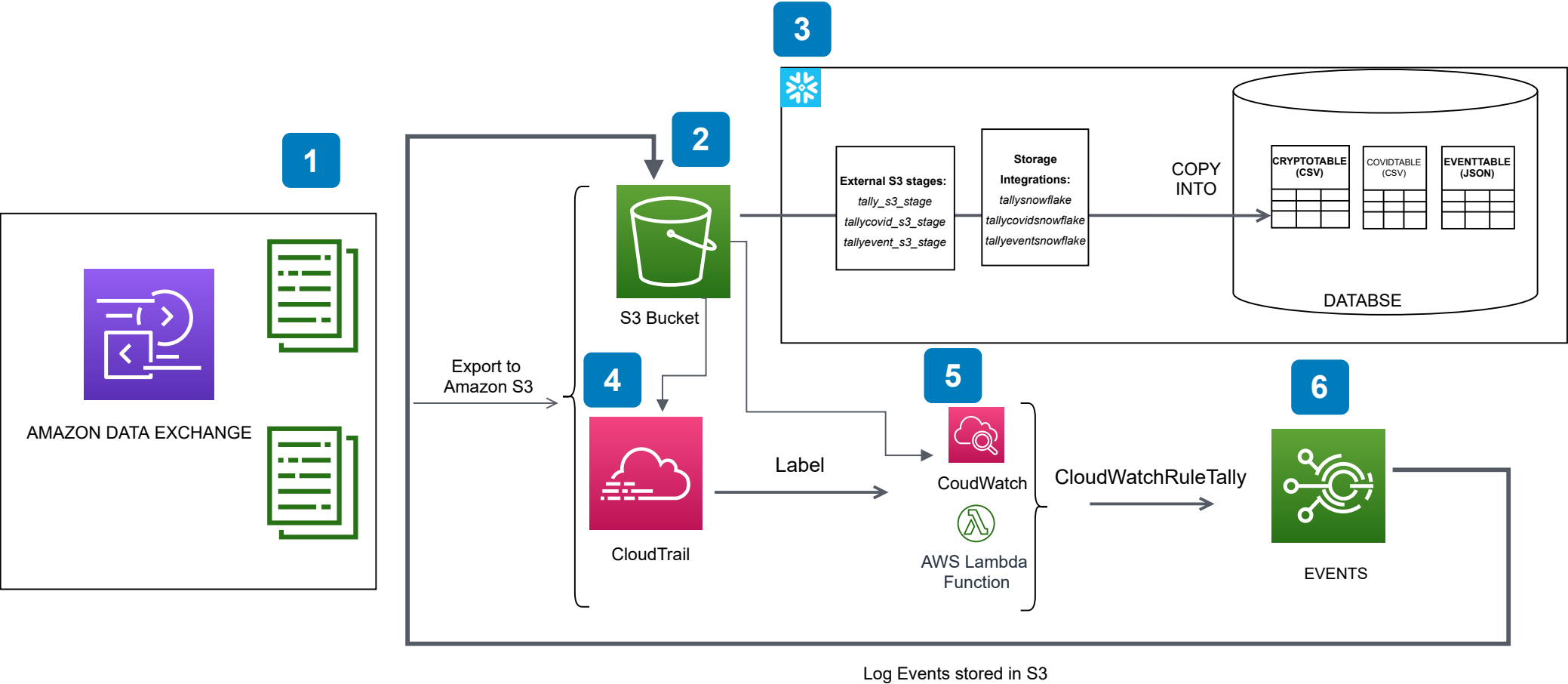
1 Used Amazon Data Exchange, 2 data sets that were freely available so subscribed to:

- **Coinbase Cryptocurrency Exchange | Crypto Data Download:** is archived and will not have updated revisions for the foreseeable future or curently; or
- **Coronavirus (COVID-19) Data Hub:** has updated revisions for the forseeable future.

2 Data sets were exported to S3 into a bucket labelled:

- **tally-demo**

3 Configured Snowflake Storage Integrations to Access Amazon S3, to allow Snowflake to read data from and write data to an Amazon S3 bucket referenced in an external S3 stage.

Loaded Data by executing COPY INTO <table> to load your data into the target table. The COPY command supports column reordering, column omission, and casts using a SELECT statement. Where the file format was JSON the UPPER() function capitilises the data loaded. However, where the file format was CSV, an SQL Compilation error occured: Function 'UPPER' not supported within a COPY. Hence, created a table, EVENTTABLE that received JSON, and capitalised the JSON data that was COPY INTO EVENTTABLE.

EVENTTABLE: The Logs of Amazon S3 Object-Level Operations Using CloudWatch Events.

4 Logged data events for the S3 bucket, **tally-demo**, to AWS CloudTrail and CloudWatch Events, by ensuring an AWS CloudTrail, **tally-management-events**, was created and configured.

5 Created a Lambda function, **LogS3DataEvents**, to log data events for the S3 bucket, **tally-demo**. This function was specified when a rule was created.

6 Created a S3 bucket folder, **AWSLogs**, to store data events for the S3 bucket, **tally-demo**.