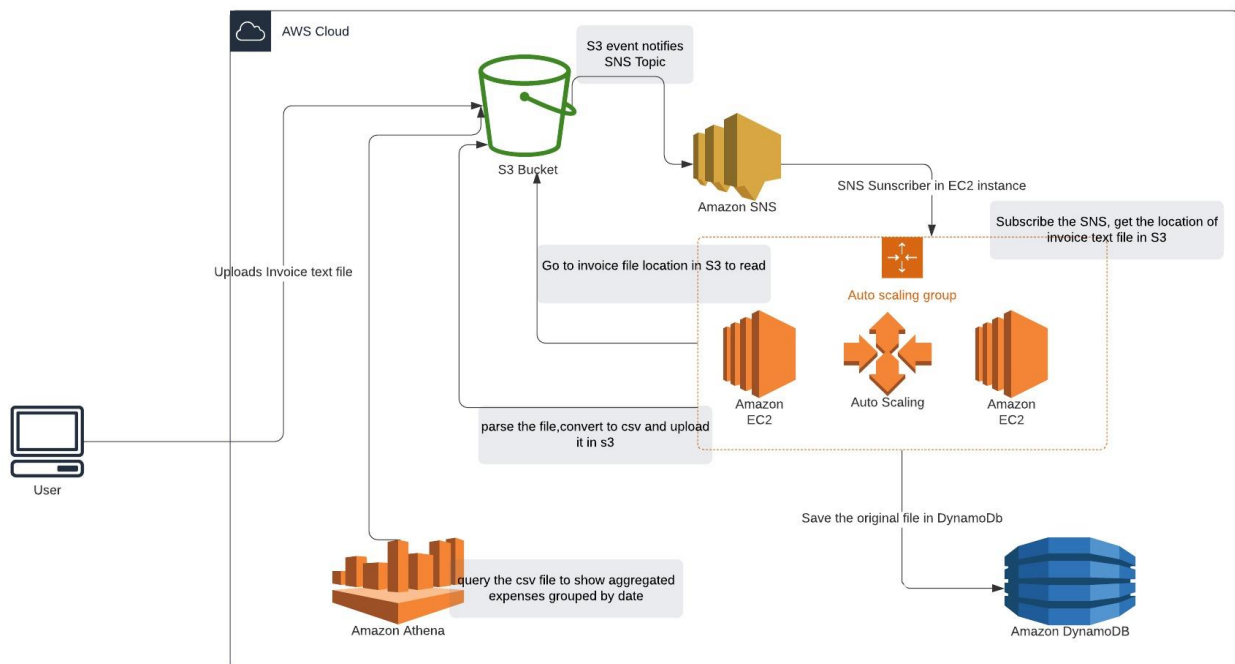


PROJECT 2

Project2_diagram

Janit Sachdeva | February 10, 2020



Steps to be followed:

1. Create a S3 bucket named-**invoice-project2**
2. Upload [docproc-invoice](#) file on s3 bucket.
3. Go to properties—Events
4. Create an SNS topic so that event would be triggered when file is uploaded.

Requester pays

on

ers of

Events

[+ Add notification](#) [Delete](#) [Edit](#)

Name	Events	Filter	Type
OWY5MjE2NjYtOGI2Ny00ZThlLTk0NWltZjhiYTkwY2IxMjQ3			

Name ⓘ

Events ⓘ

☒ PUT
☐ POST
☐ COPY
☐ Multipart upload completed
☐ All object create events
☐ Object in RRS lost
☐ Permanently deleted
☐ Delete marker created

☐ All object delete events
☐ Restore initiated
☐ Restore completed
☐ Replication time missed threshold
☐ Replication time completed after threshold
☐ Replication time not tracked
☐ Replication time failed

Prefix ⓘ

Suffix ⓘ

Send to ⓘ

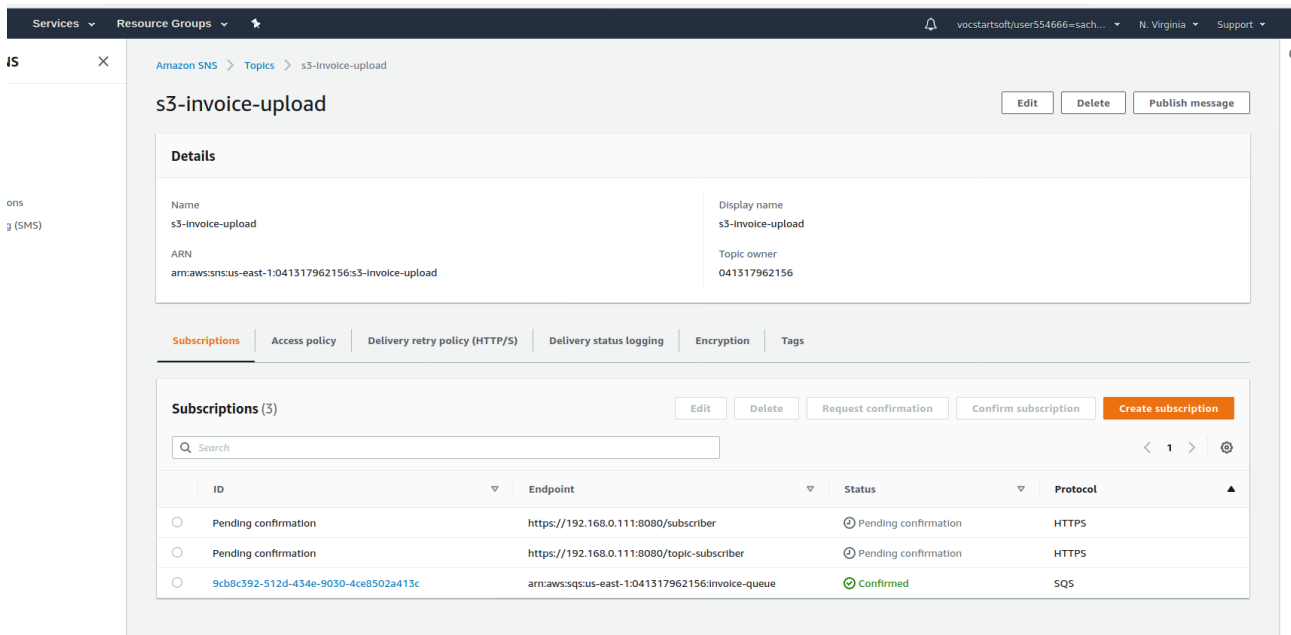
SNS

☒ 1 Active notifications

[Cancel](#) [Save](#)

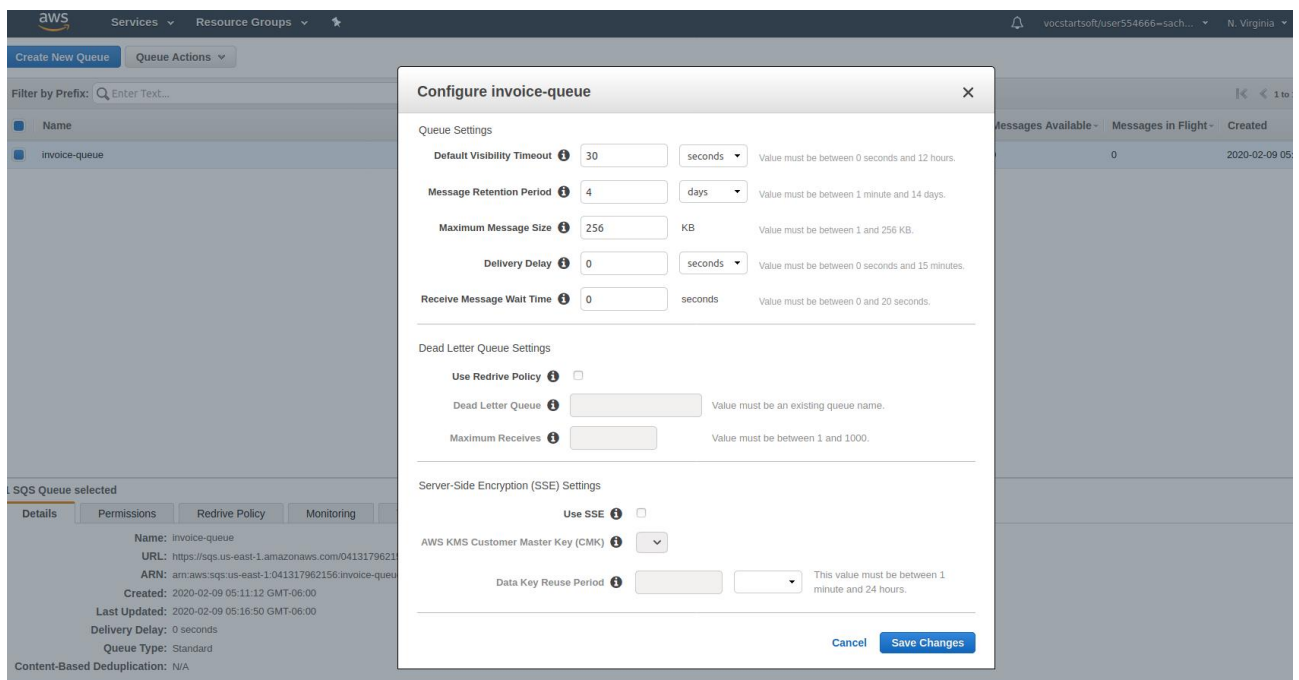
5. SNS topic name- s3-invoice-upload

6. Go to SNS

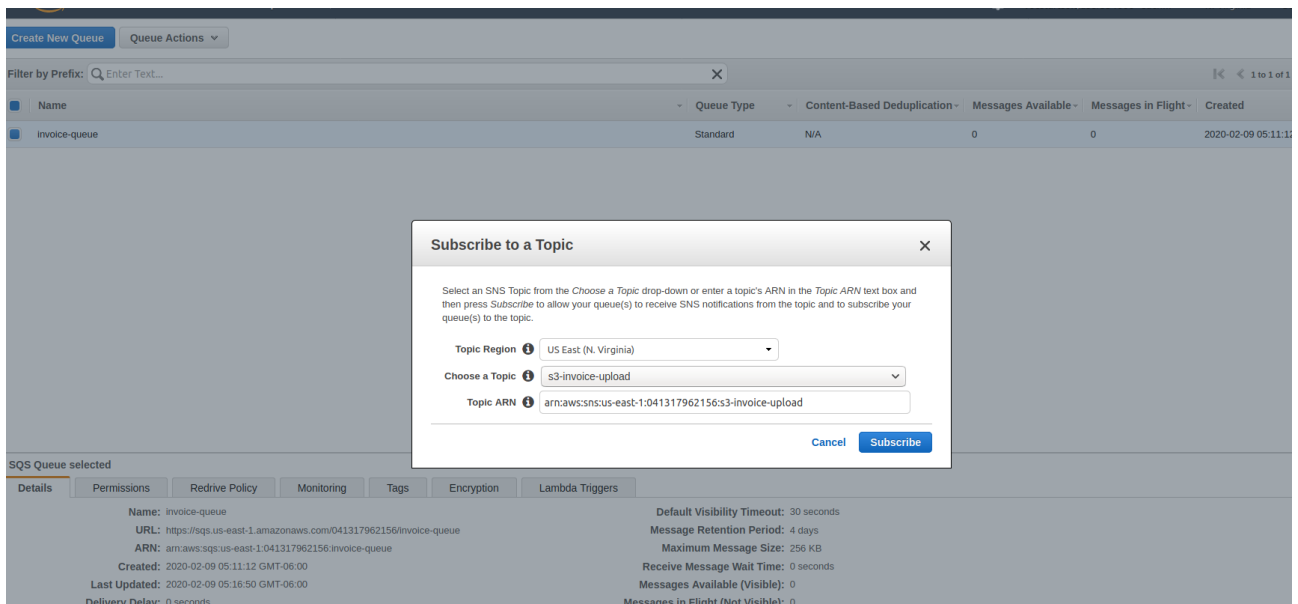


7. Create a HTTP subscription or SQS subscription.

8. Create a SQS queue- **invoice-queue**

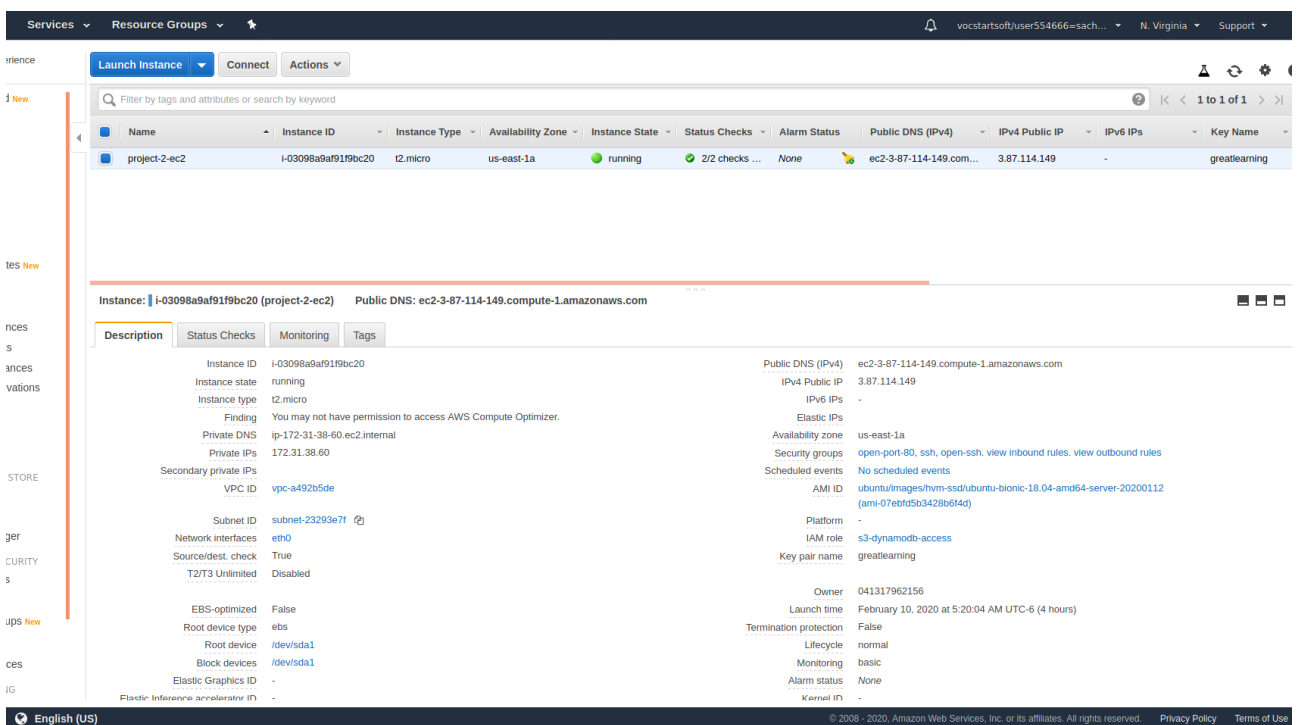


9. Subscribe to the SNS topic



10. An event will trigger in the bucket that will place a message in SNS topic

11. Create an ec2 instance and assign access to cloudwatch, s3, sns, sqs, dynamodb.



12. A custom program running in EC2 will listen to SQS and get the message placed by S3 event to SNS.

13. The program will use S3 API to read from the bucket, parse the content of the file and create a CSV record along with saving the original record in DynamoDB.

14. The program will use S3 API to write CSV record to destination S3 bucket as new S3 object.

15. Code is written in java and has been attached along with the document submission.

16. Create a dynamoDb table-**Invoice**.

Data will be written via java program.

The screenshot shows the AWS DynamoDB console interface. On the left, there's a sidebar with a search bar and a list of tables. The 'Invoice' table is selected. The main panel displays the 'Overview' tab for the 'Invoice' table. It shows recent alerts (none), stream details (stream disabled), and table details. The table details include the table name 'Invoice', primary partition key 'Customer-ID (String)', primary sort key 'Inv-ID (String)', point-in-time recovery status (disabled), encryption type (default), KMS master key ARN (not applicable), CloudWatch contributor insights (disabled), time to live attribute (disabled), table status (active), creation date (February 9, 2020), read/write capacity mode (provisioned), last change to on-demand mode (none), provisioned read/write capacity units (5), last decrease/increase time (none), storage size (199.00 bytes), and item count (1).

Table details	
Table name	Invoice
Primary partition key	Customer-ID (String)
Primary sort key	Inv-ID (String)
Point-in-time recovery	DISABLED Enable
Encryption Type	DEFAULT Manage Encryption
KMS Master Key ARN	Not Applicable
Encryption Status	
CloudWatch Contributor Insights	DISABLED Manage Contributor Insights PREVIEW
Time to live attribute	DISABLED Manage TTL
Table status	Active
Creation date	February 9, 2020 at 10:26:26 AM UTC-6
Read/write capacity mode	Provisioned
Last change to on-demand mode	-
Provisioned read capacity units	5 (Auto Scaling Error)
Provisioned write capacity units	5 (Auto Scaling Error)
Last decrease time	-
Last increase time	-
Storage size (in bytes)	199.00 bytes
Item count	1 Manage live count

17. After running the java program

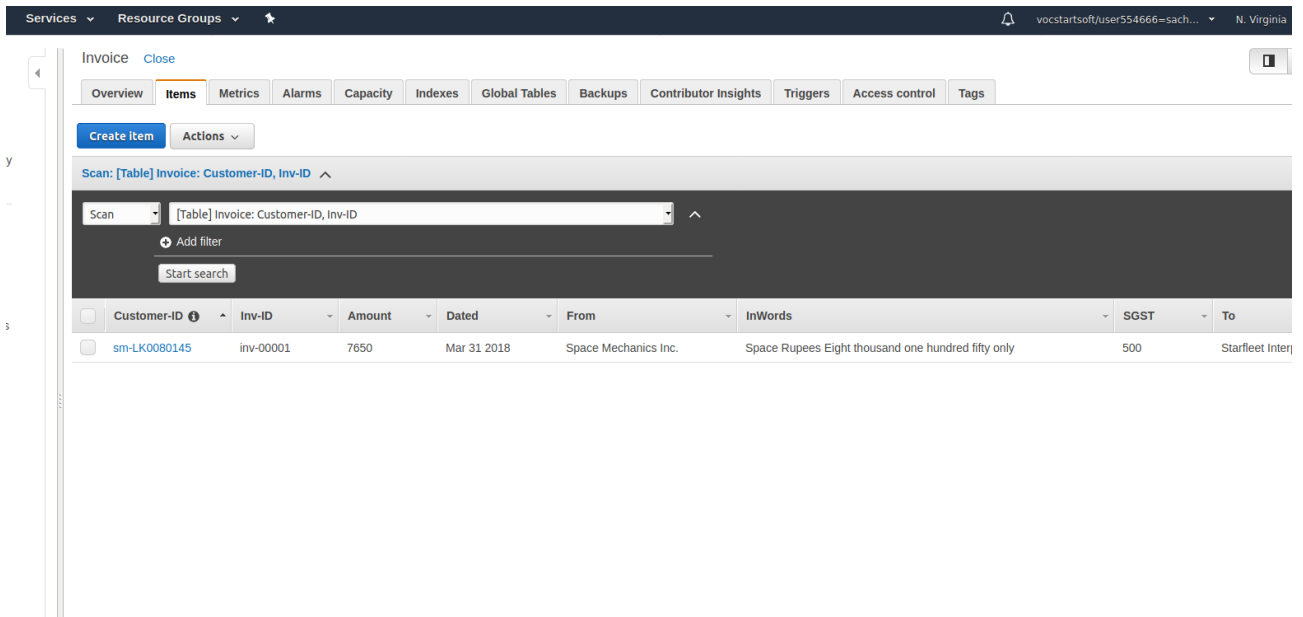
Results:

- Destination bucket is created- **dest-invoice-project2**
- docproc-invoice_new file is uploaded which is in csv format.

The screenshot shows the Amazon S3 console interface. The breadcrumb trail indicates the path: Amazon S3 > dest-invoice-project2 > docproc-invoice_new. The object 'docproc-invoice_new' is selected, and the 'Overview' tab is active. The object details include the owner (aws:labs:0w55355711576272899), last modified date (Feb 10, 2020 2:28:34 AM GMT-0600), etag (83ff50d2b36a513dad45e50807221dec), storage class (Standard), server-side encryption (None), size (218.0 B), key (docproc-invoice_new), and object URL (https://dest-invoice-project2.s3.amazonaws.com/docproc-invoice_new).

Object details	
Owner	aws:labs:0w55355711576272899
Last modified	Feb 10, 2020 2:28:34 AM GMT-0600
Etag	83ff50d2b36a513dad45e50807221dec
Storage class	Standard
Server-side encryption	None
Size	218.0 B
Key	docproc-invoice_new
Object URL	https://dest-invoice-project2.s3.amazonaws.com/docproc-invoice_new

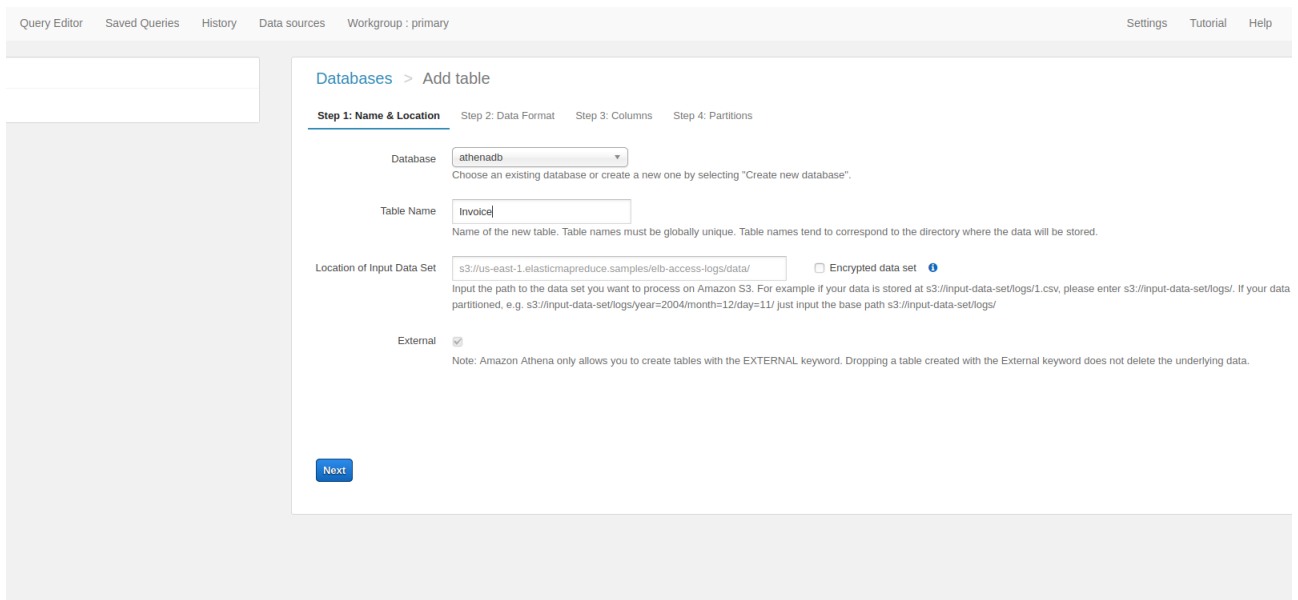
- Items are inserted in Dynamodb table-**Invoice**



18. Athena

Athena is used to query the CSV file (query to show aggregated expenses grouped by date).

Create a table Invoice based on csv file uploaded in S3 bucket.



Athena

Query Editor

Saved Queries

History

Data sources

Workgroup : primary

Settings

Tutorial

Help

What's new

ACTION

Add table

Databases > Add table

Step 1: Name & Location

Step 2: Data Format

Step 3: Columns

Step 4: Partitions

Column Name

Column Name

Column name must be single words that start with a letter or a digit.

Column type

string

Type for this column. Certain advanced types (namely, structs) are not exposed in this interface.

Column Name

Column Name

Column name must be single words that start with a letter or a digit.

Column type

Select a type

Type for this column. Certain advanced types (namely, structs) are not exposed in this interface.

Column Name

Column Name

Column name must be single words that start with a letter or a digit.

Column type

Select a type

Type for this column. Certain advanced types (namely, structs) are not exposed in this interface.

Add a column

Bulk add columns

aws

Services

Resource Groups

Athena

Query Editor

Saved Queries

History

Data sources

Workgroup : primary

Settings

Tutorial

Help

What's new

ta source

Connect data source

rsdatacatalog

tabase

henadb

Iter tables and views...

tables (1)

Create table

voice

fews (0)

Create view

u have not created any views. To create a view, run a query and click "Create view from query"

New query 1

New query 2

New query 3

New query 4

New query 5

+

```
1 select sum(Total) AggregatedTotal from Invoice group by Dated;
```

Run query

Save as

Create

(Run time: 1.85 seconds, Data scanned: 0.16 KB)

Format query

Clear

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

Results

AggregatedTotal

18150.0