

Big Data with `rubygems.org` Download Data

Aja Hammerly

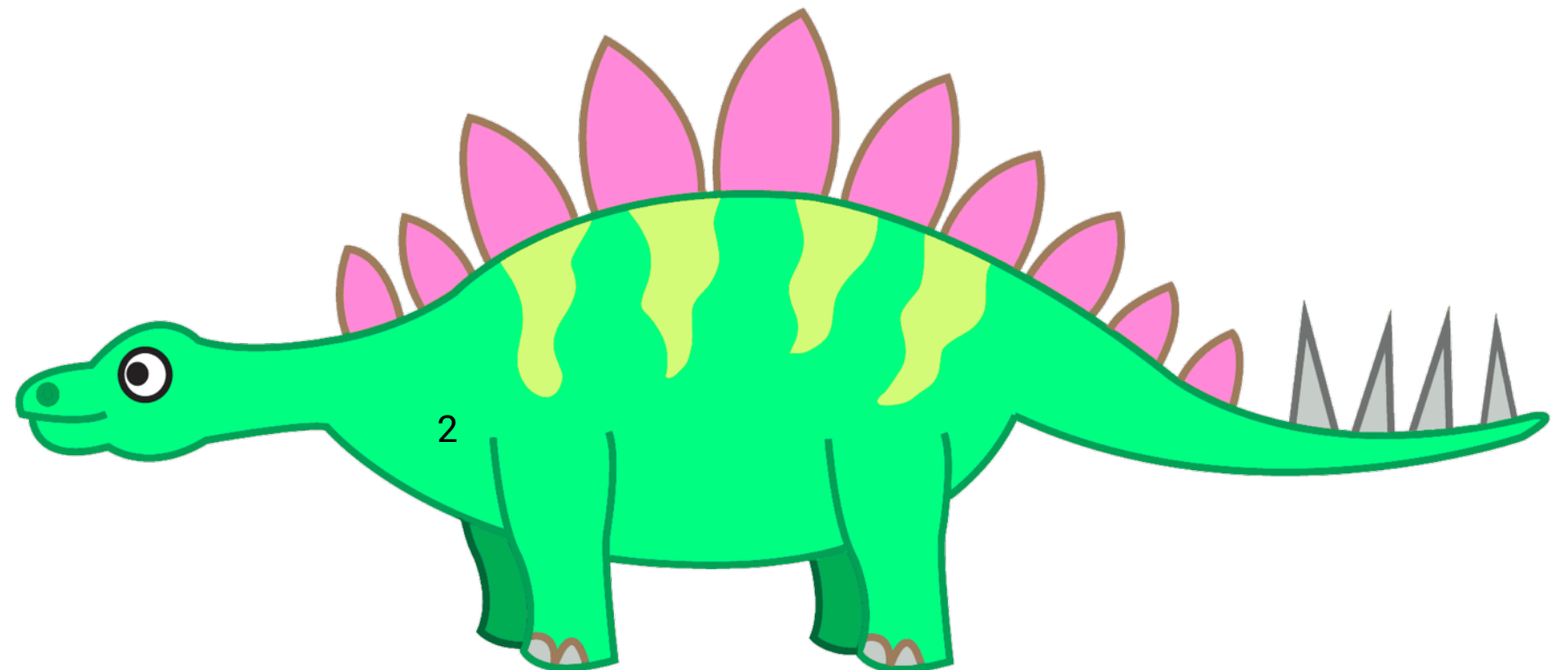
Aja Hammerly

<http://github.com/thagomizer>

@the_thagomizer

<http://www.thagomizer.com>

thagomizer@google.com





Google Cloud Platform



Lawyer Cat Says:
*All code is
copyright Google
and*

Ruby

Questions

Which gems are used
often?

**Is Minitest or Rspec
more popular?**

Do we need to support
Ruby 1.9?

**Is Rails 3, Rails 4, or
Rails 5 more popular**

Guess?

~~Guess?~~

Data

rubygems.org

Github

rubygems.org

Data

Overview

rubygems

Column	Type
id	integer
name	varchar
created_at	datetime
updated_at	datetime
slug	varchar

Column	Type
id	integer
name	varchar
created_at	datetime
updated_at	datetime
slug	varchar

gem_downloads

Column	Type
id	integer
rubygem_id	integer
version_id	integer
count	bigint

dependencies

linksets

versions

Column	Type	Column	Type
id	integer	authors	text
rubygem_id	integer	description	text
size	integer	summary	text
position	integer	requirements	text
number	varchar	platform	varchar
indexed	boolean	full_name	varchar
prerelease	boolean	licenses	varchar
latest	boolean	required_ruby_version	varchar
yanked_at	datetime	required_rubygems_version	varchar
built_at	datetime	info_checksum	varchar
updated_at	datetime	metadata	hstore
created_at	datetime	sha256	varchar

Column	Type	Column	Type
id	integer	authors	text
rubygem_id	integer	description	text
size	integer	summary	text
position	integer	requirements	text
number	varchar	platform	varchar
indexed	boolean	full_name	varchar
prerelease	boolean	licenses	varchar
latest	boolean	required_ruby_version	varchar
yanked_at	datetime	required_rubygems_version	varchar
built_at	datetime	info_checksum	varchar
updated_at	datetime	metadata	hstore
created_at	datetime	sha256	varchar

Column	Type	Column	Type
id	integer	authors	text
rubygem_id	integer	description	text
size	integer	summary	text
position	integer	requirements	text
number	varchar	platform	varchar
indexed	boolean	full_name	varchar
prerelease	boolean	licenses	varchar
latest	boolean	required_ruby_version	varchar
yanked_at	datetime	required_rubygems_version	varchar
built_at	datetime	info_checksum	varchar
updated_at	datetime	metadata	hstore
created_at	datetime	sha256	varchar

Column	Type	Column	Type
id	integer	authors	text
rubygem_id	integer	description	text
size	integer	summary	text
position	integer	requirements	text
number	varchar	platform	varchar
indexed	boolean	full_name	varchar
prerelease	boolean	licenses	varchar
latest	boolean	required_ruby_version	varchar
yanked_at	datetime	required_rubygems_version	varchar
built_at	datetime	info_checksum	varchar
updated_at	datetime	metadata	hstore
created_at	datetime	sha256	varchar

GitHub Data

files

Column	Type
repo_name	string
ref	string
path	string
mode	integer
id	string
symlink_targ	string

Column	Type
repo_name	string
ref	string
path	string
mode	integer
id	string
symlink_targ	string

contents

Column	Type
id	string
size	integer
content	string
binary	boolean
copies	integer

Column	Type
id	string
size	integer
content	string
binary	boolean
copies	integer

commits

languages

Column	Type
repo_name	string
language	record
language.na	string
language.by	integer

Column	Type
repo_name	string
language	record
language.name	string
language.by	integer

licenses

Now What?

BigQuery

What

Why

How

I  BigQuery

SQL

Fast

Scales

Complex

Demo

Vocabulary

Dataset

Table

Import

Streaming

google-cloud

pg

```
require 'pg'  
require 'google/cloud/bigquery'
```

```
ENV["GOOGLE_CLOUD_PROJECT"] =  
"rubygems-bigquery"
```

```
ENV["GOOGLE_CLOUD_KEYFILE"] =  
"#{key_path}"
```

```
bigquery = Google::Cloud.bigquery  
bq_db    = bigquery.dataset "rubygems"
```

```
postgres = PG.connect dbname: "rubygems"
```

```
bq_table ||= bq_db.create_table("gems") do |s|  
  s.integer "id"  
  s.string "name"  
  s.timestamp "created_at"  
  s.timestamp "updated_at"  
end
```

```
columns = %w[id name created_at updated_at]
```

```
postgres.exec("SELECT * FROM rubygems") do |pg_table|  
  pg_table.each do |row|  
    hashed_row = Hash[columns.zip(row.values)]  
    bq_table.insert(data)  
  end  
end
```

```
postgres.exec("SELECT * FROM rubygems") do |pg_table|  
  pg_table.each do |row|  
    hashed_row = Hash[columns.zip(row.values)]  
    bq_table.insert(data)  
  end  
end
```

```
postgres.exec("SELECT * FROM rubygems") do |pg_table|  
  pg_table.each do |row|  
    hashed_row = Hash[columns.zip(row.values)]  
    bq_table.insert(data)  
  end  
end
```

```
postgres.exec("SELECT * FROM rubygems") do |pg_table|  
  pg_table.each do |row|  
    hashed_row = Hash[columns.zip(row.values)]  
    bq_table.insert(hashed_row)  
  end  
end
```

Zip & Hash []

[key1, key2 , key3 , key4]

[val1, val2 , val3 , val4]

zip

[key1 , key2 , key3 , key4]

[val1 , val2 , val3 , val4]

[[,], [,],
[,]]

[key1 , key2 , key3 , key4]

[val1 , val2 , val3 , val4]

[[key1, val1] , [key2, val2] ,
[key3, val3] , [key4, val4]]

```
[ [key1, val1], [key2, val2],  
  [key3, val3], [key4, val4] ]
```

Hash :: []

```
Hash [[key1, val1],  
      [key2, val2],  
      [key3, val3],  
      [key4, val4]]
```

{ key1 => val1,
key2 => val2,
key3 => val3,
key4 => val4 }

Hash[keys.zip(values)]

```
postgres.exec("SELECT * FROM rubygems") do |pg_table|  
  pg_table.each do |row|  
    hashed_row = Hash[columns.zip(row.values)]  
    bq_table.insert(hashed_row)  
  end  
end
```

Batch

Formats

CSV

JSON

Avro

CSV

Import

Google BigQuery

COMPOSE QUERY

Query History

Job History

rubygems-bigquery

▶ rubygems

Public Datasets

▶ bigquery-public-d...

▶ bigquery-public-d...

▶ bigquery-public-d...

▶ bigquery-public-d...

▶ gdelt-bq:hathitrus...

▶ gdelt-bq:internet...

▶ lookerdata:cdc

▶ nyc-tlc:green

▶ nyc-tlc:yellow

Create Table

Source Data

Location: Google Cloud Storage `gs://goruco2016_bq_files/dependencies.csv` ?

File format: CSV [View Files](#)

Destination Table

Table name: rubygems . dependencies ?

Table type: Native table ?

Schema

Name	Type	Mode
id	INTEGER	NULLABLE
requirements	STRING	NULLABLE
created_at	TIMESTAMP	NULLABLE
updated_at	TIMESTAMP	NULLABLE
rubygems_id	INTEGER	NULLABLE
version_id	INTEGER	NULLABLE
scope	STRING	NULLABLE

Add Field

[Edit as Text](#)

Options

Field delimiter: ☒ Comma ☐ Tab ☐ Pipe ☐ Other ?

Header rows to skip: 0 ?

Number of errors allowed: 0 ?

COMPOSE QUERY

Query History

Job History

rubygems-bigquery ▾

▶ rubygems

▼ Public Datasets

▶ bigquery-public-d...

▶ bigquery-public-d...

▶ bigquery-public-d...

▶ bigquery-public-d...

▶ gdelt-bq:hathitrus...

▶ gdelt-bq:internet...

▶ lookerdata:cdc

▶ nyc-tlc:green

▶ nyc-tlc:yellow

Table name rubygems . dependencies ?

Table type Native table ?

Schema

Name	Type	Mode
id	INTEGER ▾	NULLABLE ▾ ×
requirements	STRING ▾	NULLABLE ▾ ×
created_at	TIMESTAMP ▾	NULLABLE ▾ ×
updated_at	TIMESTAMP ▾	NULLABLE ▾ ×
rubygems_id	INTEGER ▾	NULLABLE ▾ ×
version_id	INTEGER ▾	NULLABLE ▾ ×
scope	STRING ▾	NULLABLE ▾ ×

Add Field

[Edit as Text](#)**Options**Field delimiter ☒ Comma ☐ Tab ☐ Pipe ☐ Other ?

Header rows to skip 0 ?

Number of errors allowed 0 ?

Allow quoted newlines ☐ ?Allow jagged rows ☐ ?Ignore unknown values ☐ ?

Write preference Write if empty ▾ ?

Create Table

What Now?

Answer Questions

Which gem has the
most downloads?

```
SELECT name, count
FROM [rubygems.downloads]
JOIN rubygems.gems
ON  rubygems.gems.id =
    rubygems.downloads.rubygem_id
ORDER BY count DESC
LIMIT 5
```

name	count
rake	107,076,261
rack	100,955,906
multi_json	100,171,080
json	95,715,131
bundler	93,085,862

```
SELECT name, sum(count) as total
FROM [rubygems.downloads]
JOIN rubygems.gems
ON rubygems.gems.id =
    rubygems.downloads.rubygem_id
GROUP BY name
ORDER BY total DESC
LIMIT 5
```

name	count
rake	214,152,212
rack	201,911,759
multi_json	200,342,260
json	191,430,173
bundler	186,172,479

How many downloads
does Rails have?

```
SELECT name, sum(count) as total
FROM [rubygems.downloads]
JOIN rubygems.gems
ON  rubygems.gems.id =
    rubygems.downloads.rubygem_id
WHERE name = 'rails'
GROUP BY name
```

name	total
rails	137,635,731

Is Minitest or Rspec
more popular?

```
SELECT name, sum(count) as total
FROM [rubygems.downloads]
JOIN rubygems.gems
ON rubygems.gems.id =
    rubygems.downloads.rubygem_id
GROUP BY name
HAVING name IN ('minitest', 'rspec')
```

name	total
minitest	101,151,246
rspec	77,293,803

**Which version of Rails is
the most popular?**

```
SELECT name,  
       REGEXP_EXTRACT(number, r'(\d*)\.') AS major,  
       sum(rubygems.downloads.count) as total  
FROM [rubygems.downloads]  
JOIN rubygems.gems ON rubygems.gems.id =  
      rubygems.downloads.rubygem_id  
JOIN rubygems.versions ON rubygems.versions.id =  
      rubygems.downloads.version_id  
WHERE name = 'rails'  
GROUP BY major, name  
ORDER BY major
```

```
SELECT name,  
       REGEXP_EXTRACT(number, r'(\d*)\.') AS major,  
       sum(rubygems.downloads.count) as total  
FROM [rubygems.downloads]  
JOIN rubygems.gems ON rubygems.gems.id =  
      rubygems.downloads.rubygem_id  
JOIN rubygems.versions ON rubygems.versions.id =  
      rubygems.downloads.version_id  
WHERE name = 'rails'  
GROUP BY major, name  
ORDER BY major
```

REGEXP_EXTRACT(number, r'(\d*)\.') AS major

version	downloads
0	6,446,448
1	103,236
2	4,627,625
3	28,731,007
4	28,719,391
5	190,789

```
SELECT name,  
       REGEXP_EXTRACT(number, r'(\d*)\.') AS major,  
       sum(rubygems.downloads.count) as total  
FROM [rubygems.downloads]  
JOIN rubygems.gems ON rubygems.gems.id =  
      rubygems.downloads.rubygem_id  
LEFT JOIN rubygems.versions ON  
      rubygems.versions.id =  
      rubygems.downloads.version_id  
WHERE name = 'rails'  
GROUP BY major, name  
ORDER BY major
```

version	downloads
null	68,817,235
0	6,446,448
1	103,236
2	4,627,625
3	28,731,007
4	28,719,391
5	190,789

Do we need to support
Ruby 1.9?

Which version of ruby do gems released in the past year require?

```
SELECT
    required_ruby_version,
    COUNT(*) AS total
FROM
    rubygems.versions
WHERE
    created_at > DATE_ADD(CURRENT_TIMESTAMP(), -1, "YEAR")
GROUP BY
    required_ruby_version
ORDER BY
    total DESC
```

name	total
≥ 0	75,821
$\geq 1.9.3$	6,833
$\geq 2.0.0$	3,829
≥ 2.0	1,428
$\geq 2.3.0$	1,334

name	total
≥ 0	75,821
$\geq 1.9.3$	6,833
$\geq 2.0.0$	3,829
≥ 2.0	1,428
$\geq 2.3.0$	1,334

```
SELECT
  REGEXP_EXTRACT(required_ruby_version,
                  r'(. *? \d \. ?)') AS version,
  COUNT(*) AS total
FROM
  rubygems.versions
WHERE
  created_at > DATE_ADD(CURRENT_TIMESTAMP(), -1, "YEAR")
GROUP BY
  version
ORDER BY
  total DESC
```

name	total
≥ 0	95,851
≥ 1	13,080
≥ 2	12,944
$\sim > 2$	2,040
> 2	49

What gems are most
commonly used?

Which gems are used the
most on GitHub?

```
SELECT REGEXP_EXTRACT(line, r"\s+([\w\-_]*) \(") as gem,  
       count(*) as total  
FROM (  
  SELECT SPLIT(content, '\n') as line  
  FROM github_ruby.gemfilelock  
)  
GROUP BY GEM  
HAVING gem IS NOT NULL  
ORDER BY total DESC
```

name	total
activesuppor	355,791
rack	197,631
railties	179,002
actionpack	167,075
json	132,750

Conclusions



Google Cloud Platform







Thank You