PostHog

Serving 5m analytics queries a month, with Clickhouse!

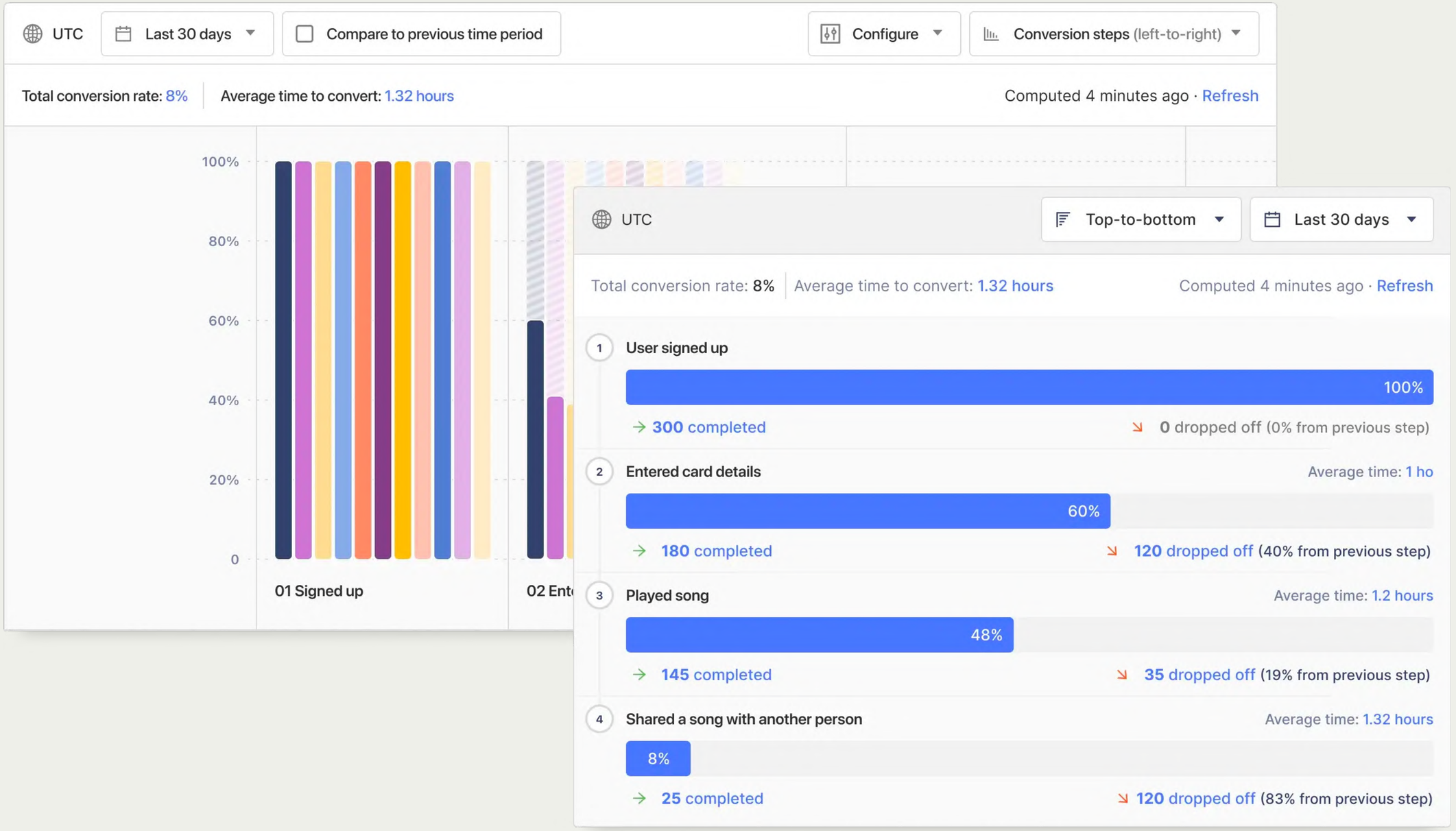
Tim Glaser
Co-founder/CTO

Three parts

What is PostHog

How we scaled to millions of queries/day

Launching HouseWatch


UTC
Last 30 days
Compare to previous time period
Configure
Conversion steps (left-to-right)
Total conversion rate: 8%
Average time to convert: 1.32 hours
Computed 4 minutes ago · Refresh
100%
80%
60%
40%
20%
0
01 Signed up
02 Ent
UTC
Top-to-bottom
Last 30 days
Total conversion rate: 8%
Average time to convert: 1.32 hours
Computed 4 minutes ago · Refresh
1 User signed up
100%
300 completed
0 dropped off (0% from previous step)
2 Entered card details
Average time: 1 ho
60%
180 completed
120 dropped off (40% from previous step)
3 Played song
Average time: 1.2 hours
48%
145 completed
35 dropped off (19% from previous step)
4 Shared a song with another person
Average time: 1.32 hours
8%
25 completed
120 dropped off (83% from previous step)

Series

Enable formula mode

A Pageview | Unique users

where Current URL | = equals | https://www.isgoogleanalyticsillegal.com/

AND Add filter

+ Add graph series

Filters

Filter out internal and test users

+ Add filter group

Breakdown by

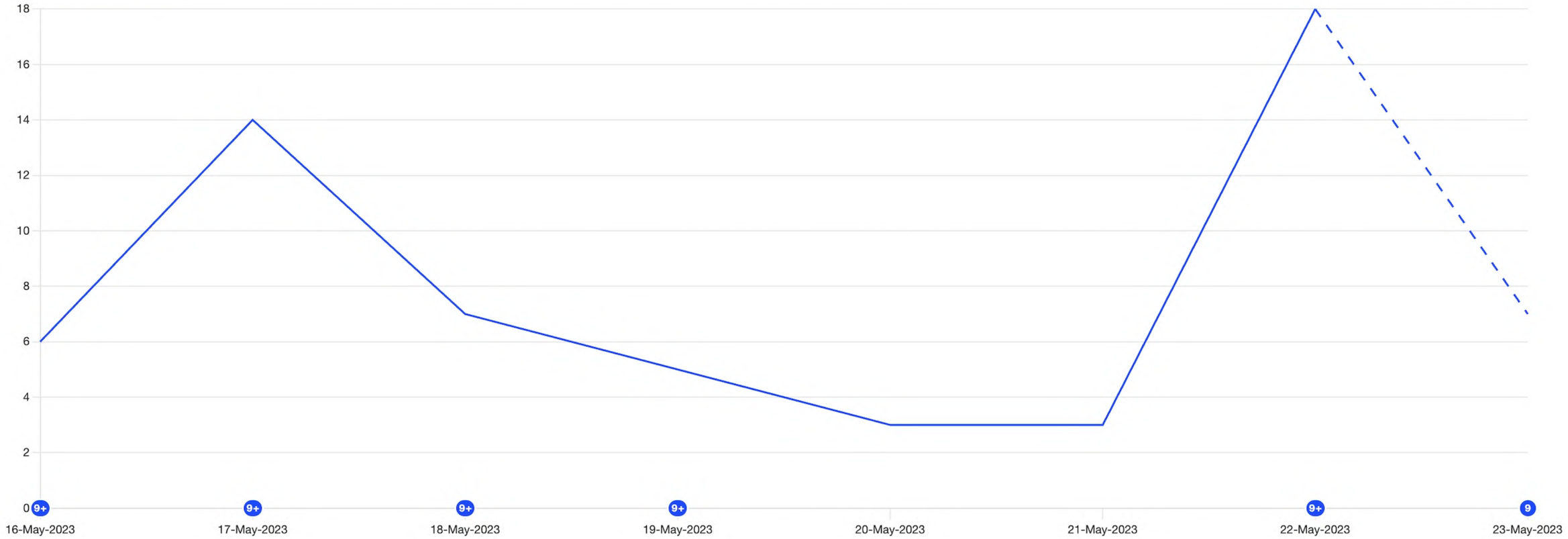
+ Add breakdown

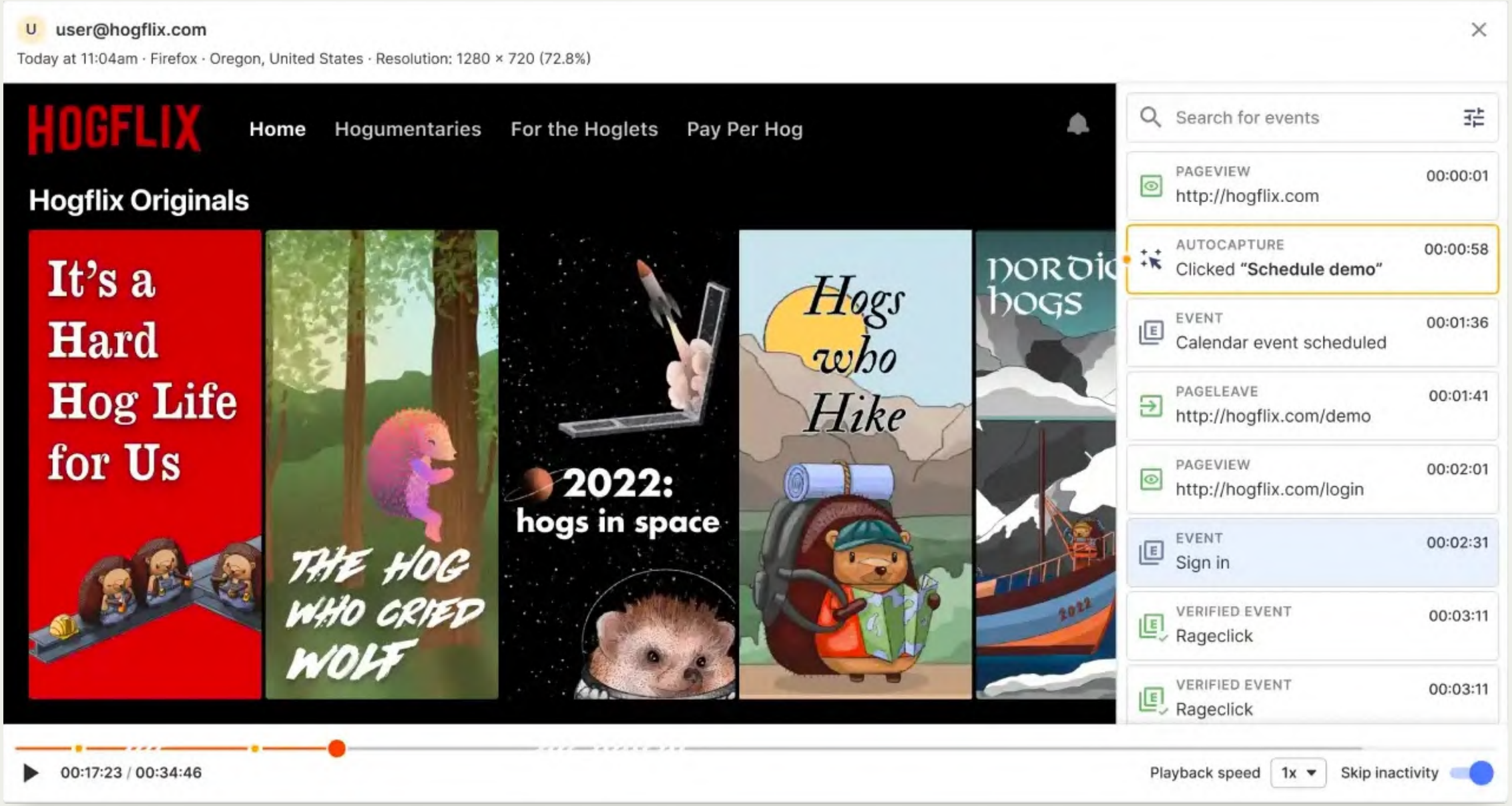
Sampling BETA

Date range Last 7 days | grouped by Day | No smoothing | Compare to previous time period | Show values on series | Y-axis unit None | Chart type Line

Computed a few seconds ago

Show legend




user@hogflix.com
Today at 11:04am · Firefox · Oregon, United States · Resolution: 1280 × 720 (72.8%)
HOGFLIX
Home
Hogumentaries
For the Hoglets
Pay Per Hog
Hogflix Originals
It's a Hard Hog Life for Us
THE HOG WHO CRIED WOLF
2022: hogs in space
Hogs who Hike
NORDIC hogs
Search for events
PAGEVIEW
http://hogflix.com
00:00:01
AUTOCAPTURE
Clicked "Schedule demo"
00:00:58
EVENT
Calendar event scheduled
00:01:36
PAGELEAVE
http://hogflix.com/demo
00:01:41
PAGEVIEW
http://hogflix.com/login
00:02:01
EVENT
Sign in
00:02:31
VERIFIED EVENT
Rageclick
00:03:11
VERIFIED EVENT
Rageclick
00:03:11
00:17:23 / 00:34:46
Playback speed
1x
Skip inactivity

Release conditions

Set 1 Match **users** against **all** criteria

↳ email equals joe@posthog.com luke@posthog.com

& Cohort Feature Flag Creators (Static)

Rolled out to 100% of users in this set.

Experiment progress

Computed a few seconds ago · Refresh

200 participants seen

Goal: 500 participants

Variant results

Test Group 1

Top result · Release to production

Probability that this variant is the best: **65.7%**

Test Group 3

Probability that this variant is the best: **11.4%**

Test Group 2

Probability that this variant is the best: **11.4%**

Control Group

Probability that this variant is the best: **11.4%**

RECOMMENDED RUNNING TIME

~29.7 days

RECOMMENDED SAMPLE SIZE

~500 persons

ELAPSED RUN TIME

5 days

PARTICPANT COUNT

200 persons

CONVERSION GOAL

1. **Pageview**
   - Where Current URL equals **http://hogflix.com**
2. **Viewed pricing**
3. **Signed up**
   - Where Browser equals **Firefox**
   - and Current URL equals **/signup?param=123**

FILTERS

- Where Current URL equals **http://hogflix.com**
- and Current URL equals **http://hogflix.com**
- and Current URL equals **http://hogflix.com**

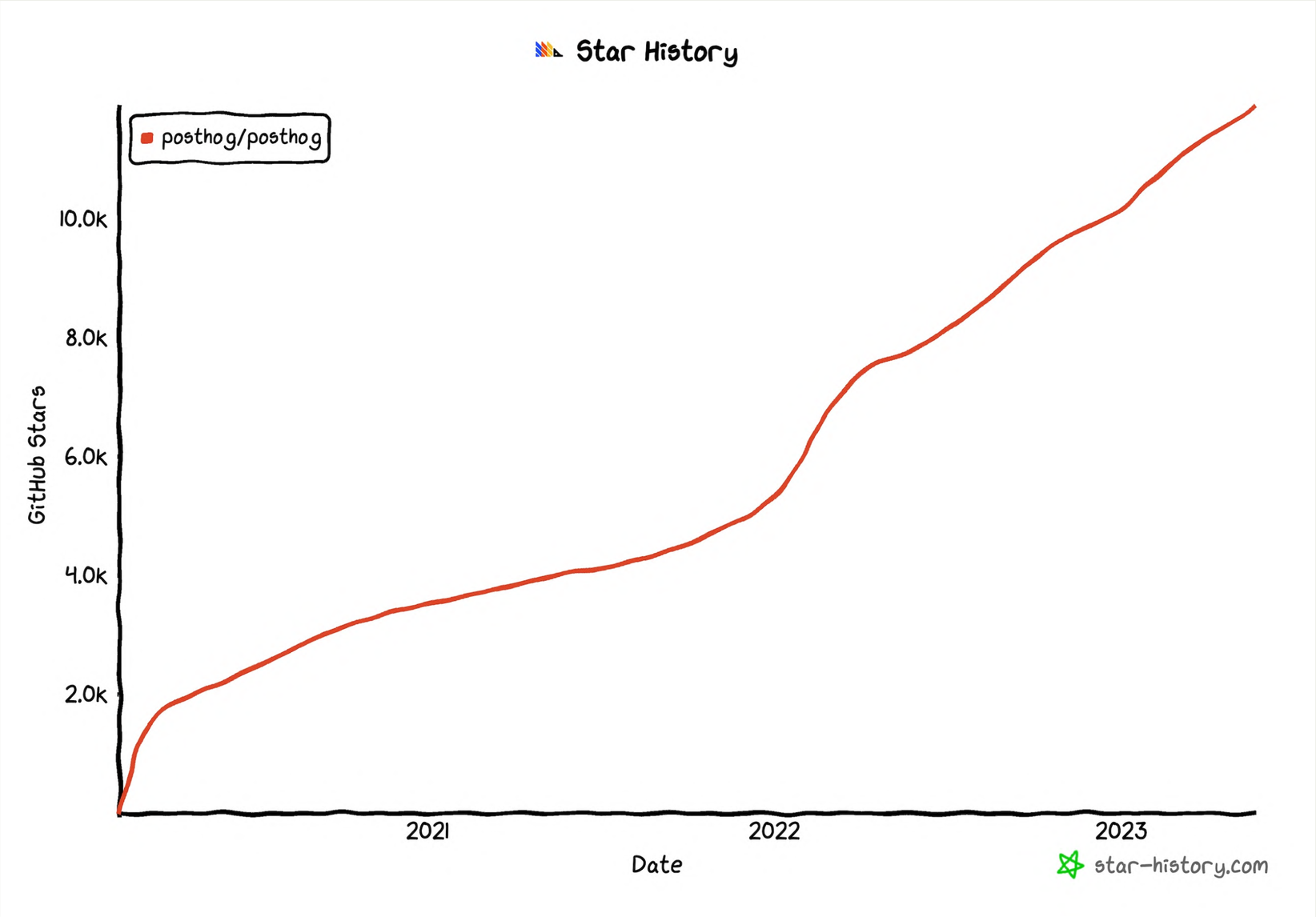
SECONDARY METRICS

	Control	Test 1	Test 2
Subscription started (count)	52	62	61

PostHog

Event pipeline

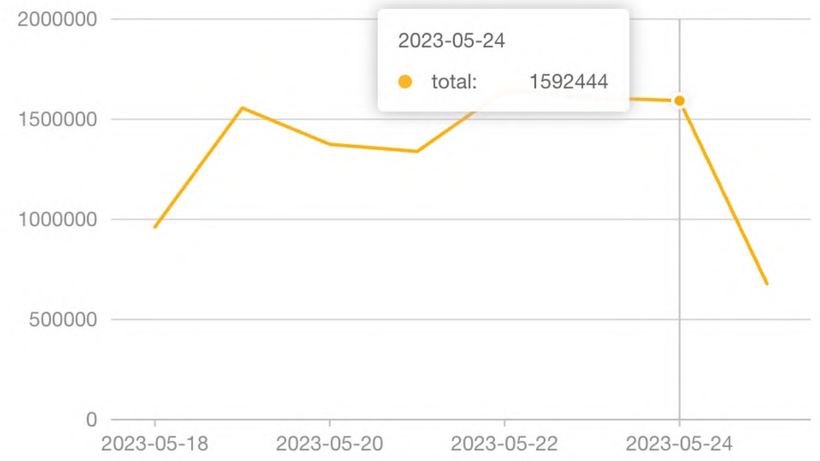
FREE Airbyte Exporter	FREE Avo Inspector	FREE BigQuery Export	FREE Customer.io Connector
FREE Databricks Export	FREE Engage Connector	FREE Google Cloud Storage Export	FREE Google Pub/Sub Connector
FREE Hubspot Connector	FREE Ingestion Alert	FREE Intercom Connector	FREE Laudspeaker Connector
FREE Memphis Exporter	FREE Outfunnel Exporter	FREE Pace Integration	FREE PagerDuty Connector
FREE Patterns Connector	FREE PostgreSQL Export	FREE Redshift Export	FREE PostHog Replicator
FREE RudderStack Export	FREE Amazon S3 Export	FREE Salesforce Connector	FREE Sendgrid Connector
FREE Sentry Connector	FREE Snowflake Export	FREE Twilio Connector	FREE Variance Connector


Star History
posthog/posthog
10.0k
8.0k
6.0k
4.0k
2.0k
GitHub Stars
2021
2022
2023
Date
star-history.com

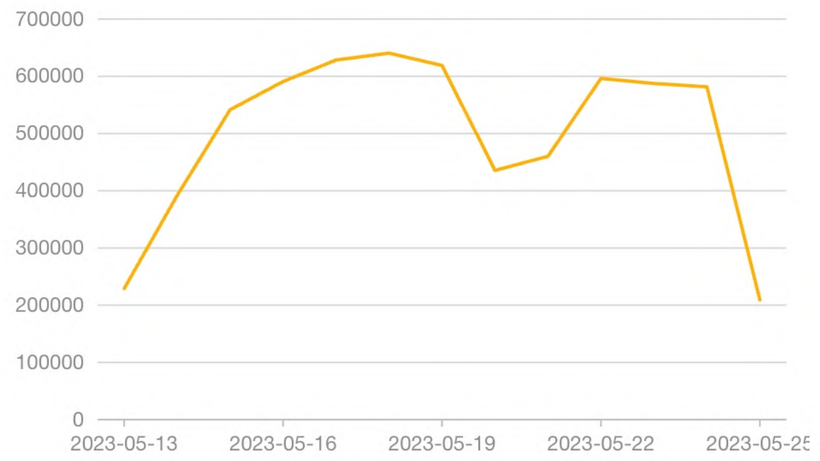
81k+	411+	50b+
Developer community	Contributors	Events tracked

Postl

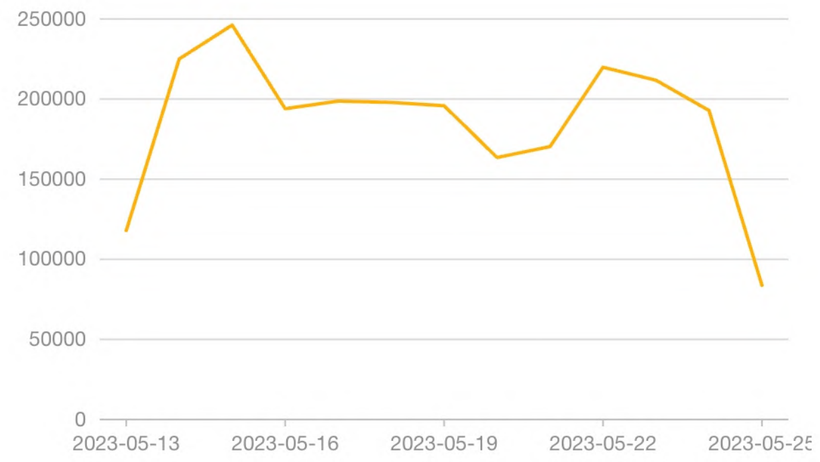
Number of queries



Data read (GB)



Memory usage (GB)



CPU usage (seconds)

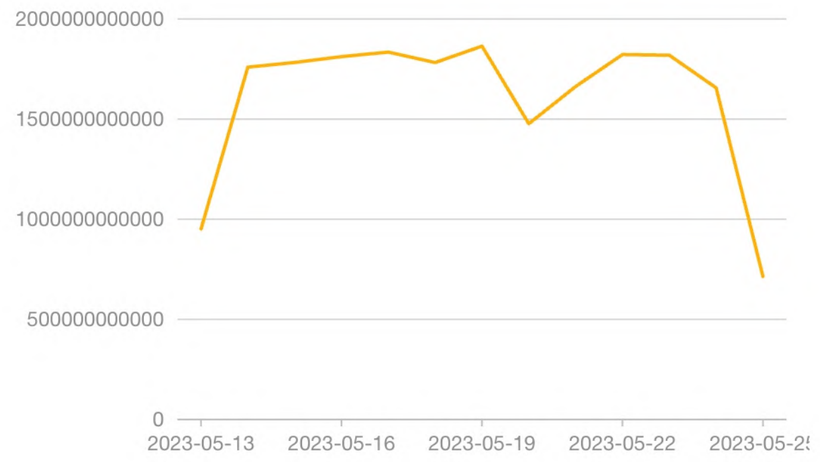
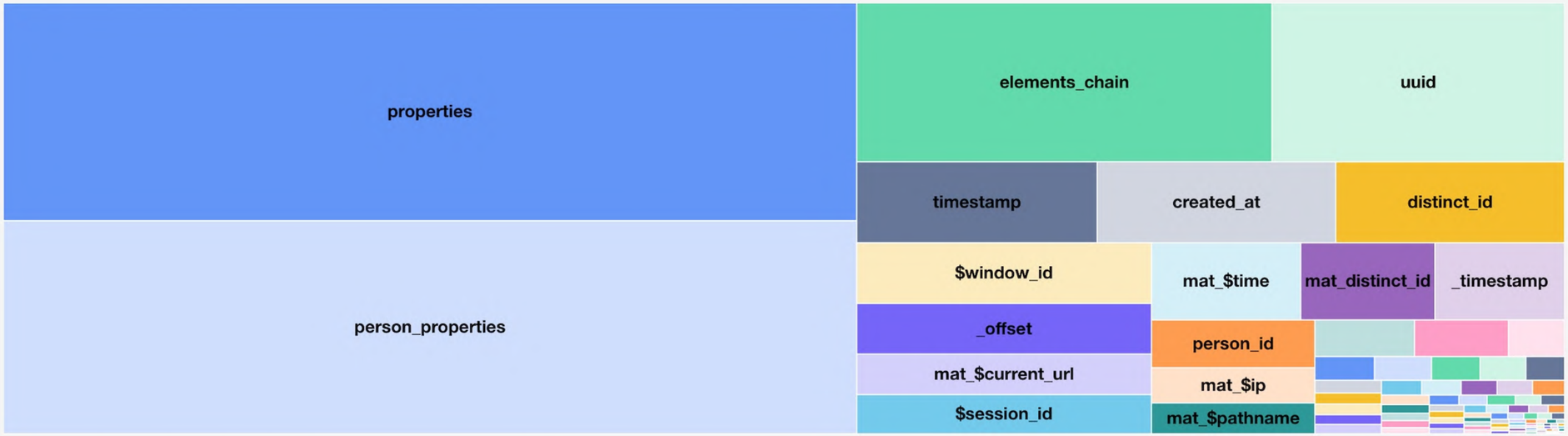


Table: sharded_events

Columns | Parts



Name	type	Compressed	Uncompressed
properties	String	1.89 TiB	62.04 TiB
person_properties	String	1.85 TiB	47.92 TiB
elements_chain	String	687.33 GiB	8.54 TiB
uuid	UUID	484.44 GiB	503.71 GiB
timestamp	DateTime64(6, 'UTC')	203.13 GiB	251.86 GiB
created_at	DateTime64(6, 'UTC')	201.30 GiB	251.86 GiB

Server sizing

Instance Size	vCPU	Memory (GiB)	Instance Storage (GB)	Network Bandwidth (Gbps)***	EBS Bandwidth (Gbps)
r6i.large	2	16	EBS-Only	Up to 12.5	Up to 10
r6i.xlarge	4	32	EBS-Only	Up to 12.5	Up to 10
r6i.2xlarge	8	64	EBS-Only	Up to 12.5	Up to 10
r6i.4xlarge	16	128	EBS-Only	Up to 12.5	Up to 10
r6i.8xlarge	32	256	EBS-Only	12.5	10
r6i.12xlarge	48	384	EBS-Only	18.75	15
r6i.16xlarge	64	512	EBS-Only	25	20
r6i.24xlarge	96	768	EBS-Only	37.5	30
r6i.32xlarge	128	1,024	EBS-Only	50	40
r6i.metal	128	1,024	EBS-Only	50	40
r6id.large	2	16	1x118 NVMe SSD	Up to 12.5	Up to 10
r6id.xlarge	4	32	1x237 NVMe SSD	Up to 12.5	Up to 10
r6id.2xlarge	8	64	1x474 NVMe SSD	Up to 12.5	Up to 10
r6id.4xlarge	16	128	1x950 NVMe SSD	Up to 12.5	Up to 10
r6id.8xlarge	32	256	1x1900 NVMe SSD	12.5	10
r6id.12xlarge	48	384	2x1425 NVMe SSD	18.75	15
r6id.16xlarge	64	512	2x1900 NVMe SSD	25	20
r6id.24xlarge	96	768	4x1425 NVMe SSD	37.5	30
r6id.32xlarge	128	1,024	4x1900 NVMe SSD	50	40
r6id.metal	128	1,024	4x1900 NVMe SSD	50	40

Storage Optimized

Storage optimized instances are designed for workloads that require high, sequential read and write access to very large data sets on local storage. They are optimized to deliver tens of thousands of low-latency, random I/O operations per second (IOPS) to applications.

I3 | I3en | D2 | D3 | D3en | H1

This instance family provides dense Non-Volatile Memory Express (NVMe) SSD instance storage optimized for low latency, high random I/O performance, high sequential disk throughput, and offers the lowest price per GB of SSD instance storage on Amazon EC2. I3en also offers Bare Metal instances (i3en.metal), powered by the Nitro System, for non-virtualized workloads, workloads that benefit from access to physical resources, or workloads that may have license restrictions.

Features:

- Up to 60 TB of NVMe SSD instance storage
- Up to 100 Gbps of network bandwidth using Elastic Network Adapter (ENA)-based Enhanced Networking
- High random I/O performance and high sequential disk throughput
- Up to 3.1 GHz Intel® Xeon® Scalable (Skylake) processors with new Intel Advanced Vector Extension (AVX-512) instruction set
- Powered by the AWS Nitro System, a combination of dedicated hardware and lightweight hypervisor
- Support bare metal instance size for workloads that benefit from direct access to physical processor and memory
- Support for Elastic Fabric Adapter on i3en.24xlarge

Instance	vCPU	Mem (GiB)	Local Storage (GB)	Network Bandwidth
i3en.large	2	16	1 x 1,250 NVMe SSD	Up to 25 Gbps
i3en.xlarge	4	32	1 x 2,500 NVMe SSD	Up to 25 Gbps
i3en.2xlarge	8	64	2 x 2,500 NVMe SSD	Up to 25 Gbps
i3en.3xlarge	12	96	1 x 7,500 NVMe SSD	Up to 25 Gbps
i3en.6xlarge	24	192	2 x 7,500 NVMe SSD	25 Gbps
i3en.12xlarge	48	384	4 x 7,500 NVMe SSD	50 Gbps
i3en.24xlarge	96	768	8 x 7,500 NVMe SSD	100 Gbps
i3en.metal	96	768	8 x 7,500 NVMe SSD	100 Gbps

All instances have the following specs:

- 3.1 GHz all core turbo Intel® Xeon® Scalable (Skylake) processors
- Intel AVX†, Intel AVX2†, Intel AVX-512†, Intel Turbo
- EBS Optimized
- Enhanced Networking

Pageview | T tim@posthog.com | https://app.posthog.com/ | web | a few seconds ago

Properties JSON

Search for property k... | Hide PostHog properties

Timestamp	2023-05-25T13:03:17.290Z DATETIME
OS	Mac OS X STRING
OS Version	10.15.7 STRING
Browser	Chrome STRING
Device Type	Desktop STRING
Current URL	https://app.posthog.com/ STRING
Host	app.posthog.com STRING
Path Name	/ STRING
Browser Version	113 NUMERIC
Browser Language	en-GB STRING
Screen Height	1890 NUMERIC
Screen Width	3360 NUMERIC
Viewport Height	1984 NUMERIC
Viewport Width	1866 NUMERIC

Series

Enable formula mode

A Pageview | Unique users

where Current URL | = equals | https://www.isgoogleanalyticsillegal.com/

AND Add filter

+ Add graph series

Filters

Filter out internal and test users

+ Add filter group

Breakdown by

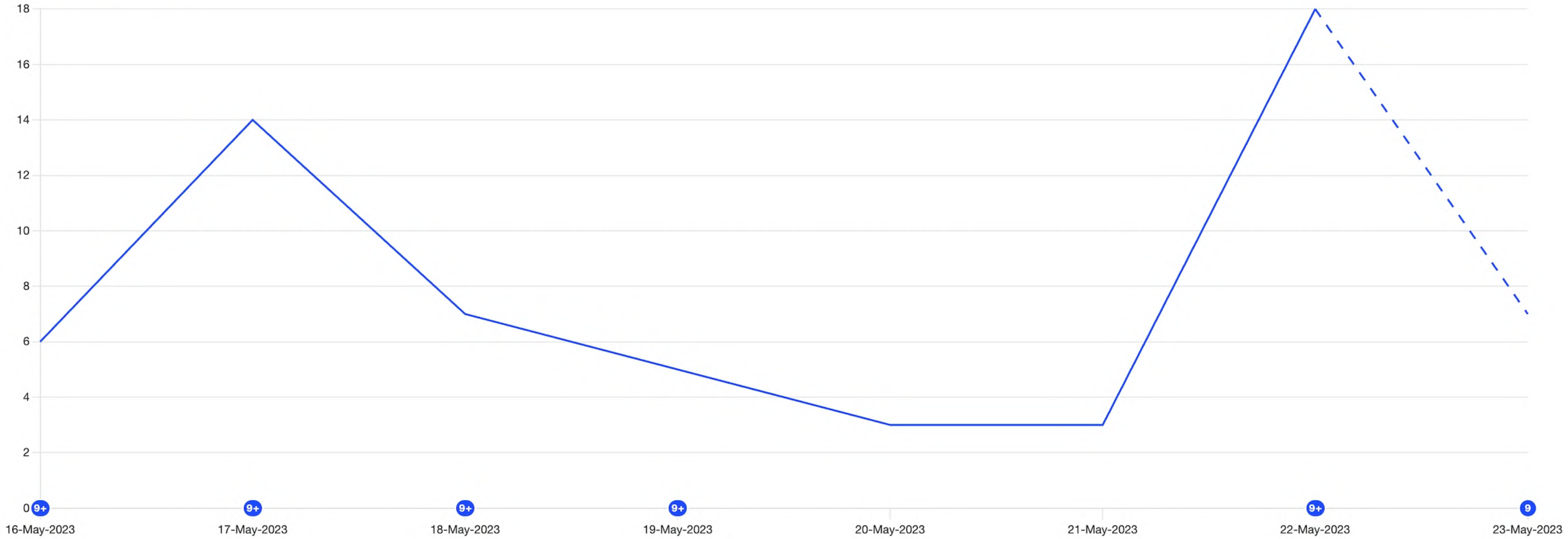
+ Add breakdown

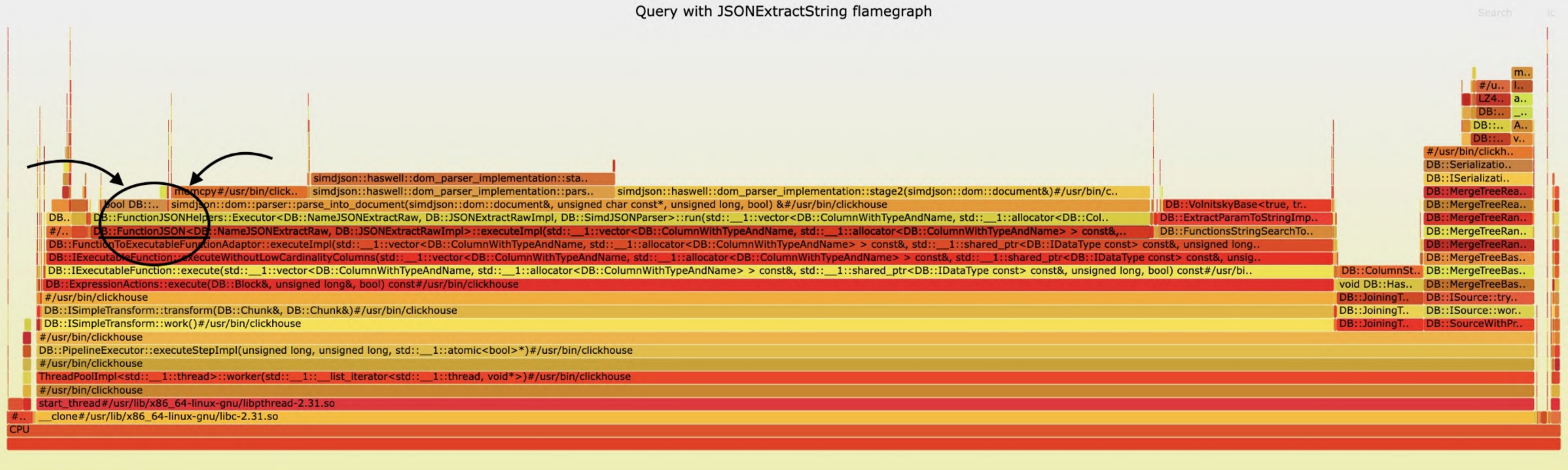
Sampling BETA

Date range Last 7 days | grouped by Day | No smoothing | Compare to previous time period | Show values on series | Y-axis unit None | Chart type Line

Computed a few seconds ago

Show legend




Query with JSONExtractString flamegraph
Search
simdjson::haswell::dom_parser_implementation::sta..
memcpy#/usr/bin/click..
simdjson::haswell::dom_parser_implementation::pars..
simdjson::haswell::dom_parser_implementation::stage2(simdjson::dom::document&)#/usr/bin/c..
bool DB::..
simdjson::dom::parser::parse_into_document(simdjson::dom::document&, unsigned char const*, unsigned long, bool) &#/usr/bin/clickhouse
DB::VolnitskyBase<true, tr..
DB..
DB::FunctionJSONHelpers::Executor<DB::NameJSONExtractRaw, DB::JSONExtractRawImpl, DB::SimdJSONParser>::run(std::__1::vector<DB::ColumnWithTypeAndName, std::__1::allocator<DB::Col..
DB::ExtractParamToStringImp..
#/..
DB::FunctionJSON<DB::NameJSONExtractRaw, DB::JSONExtractRawImpl>::executeImpl(std::__1::vector<DB::ColumnWithTypeAndName, std::__1::allocator<DB::ColumnWithTypeAndName> > const&,..
DB::FunctionsStringSearchTo..
DB::FunctionToExecutableFunctionAdaptor::executeImpl(std::__1::vector<DB::ColumnWithTypeAndName, std::__1::allocator<DB::ColumnWithTypeAndName> > const&, std::__1::shared_ptr<DB::IDataType const> const&, unsigned long..
DB::IExecutableFunction::executeWithoutLowCardinalityColumns(std::__1::vector<DB::ColumnWithTypeAndName, std::__1::allocator<DB::ColumnWithTypeAndName> > const&, std::__1::shared_ptr<DB::IDataType const> const&, unsig..
DB::IExecutableFunction::execute(std::__1::vector<DB::ColumnWithTypeAndName, std::__1::allocator<DB::ColumnWithTypeAndName> > const&, std::__1::shared_ptr<DB::IDataType const> const&, unsigned long, bool) const#/usr/bi..
DB::ColumnSt..
DB::ExpressionActions::execute(DB::Block&, unsigned long&, bool) const#/usr/bin/clickhouse
void DB::Has..
#/usr/bin/clickhouse
DB::JoiningT..
DB::ISimpleTransform::transform(DB::Chunk&, DB::Chunk&)#/usr/bin/clickhouse
DB::ISimpleTransform::work()#/usr/bin/clickhouse
#/usr/bin/clickhouse
DB::PipelineExecutor::executeStepImpl(unsigned long, unsigned long, std::__1::atomic<bool>*)#/usr/bin/clickhouse
#/usr/bin/clickhouse
ThreadPoolImpl<std::__1::thread>::worker(std::__1::__list_iterator<std::__1::thread, void*>)#/usr/bin/clickhouse
#/usr/bin/clickhouse
start_thread#/usr/lib/x86_64-linux-gnu/libpthread-2.31.so
#..
__clone#/usr/lib/x86_64-linux-gnu/libc-2.31.so
CPU
m..
#/u..
l..
LZ4..
a..
DB::..
_..
DB::..
A..
DB::..
v..
#/usr/bin/clickh..
DB::Serializatio..
DB::ISerializati..
DB::MergeTreeRea..
DB::MergeTreeRea..
DB::MergeTreeRan..
DB::MergeTreeRan..
DB::MergeTreeRan..
DB::MergeTreeBas..
DB::MergeTreeBas..
DB::MergeTreeBas..
DB::ISource::try..
DB::ISource::wor..
DB::SourceWithPr..

```
ALTER TABLE events
ADD COLUMN mat_$current_url
VARCHAR MATERIALIZED JSONExtractString(properties_json, '$current_url')
```

Series

Enable formula mode

A Pageview | Unique users

where Current URL | = equals | https://www.isgoogleanalyticsillegal.com/

AND Add filter

+ Add graph series

Filters

Filter out internal and test users

+ Add filter group

Breakdown by

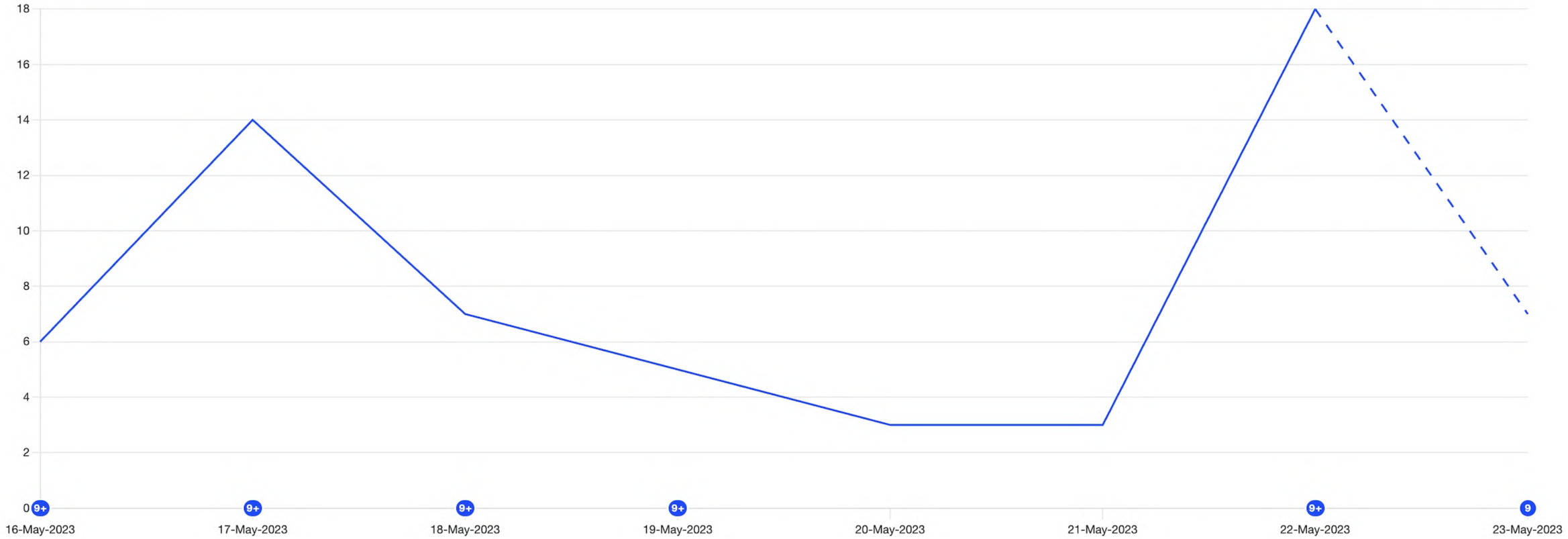
+ Add breakdown

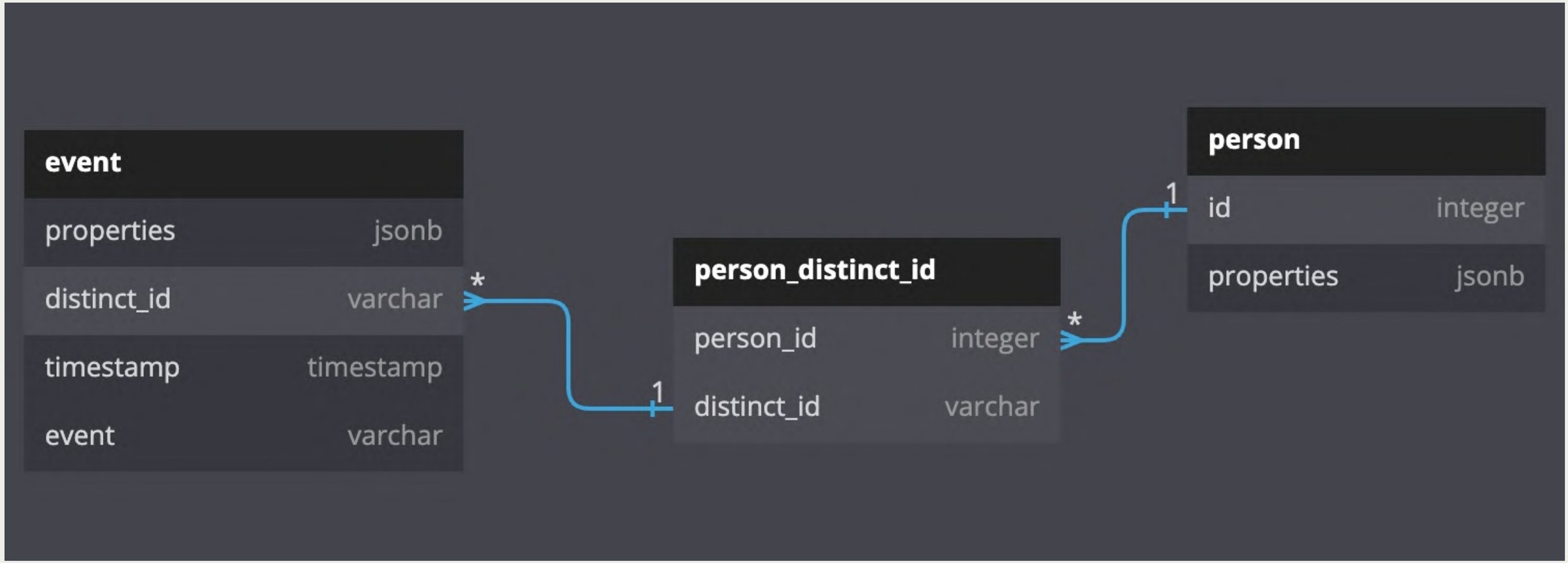
Sampling BETA

Date range Last 7 days | grouped by Day | No smoothing | Compare to previous time period | Show values on series | Y-axis unit None | Chart type Line

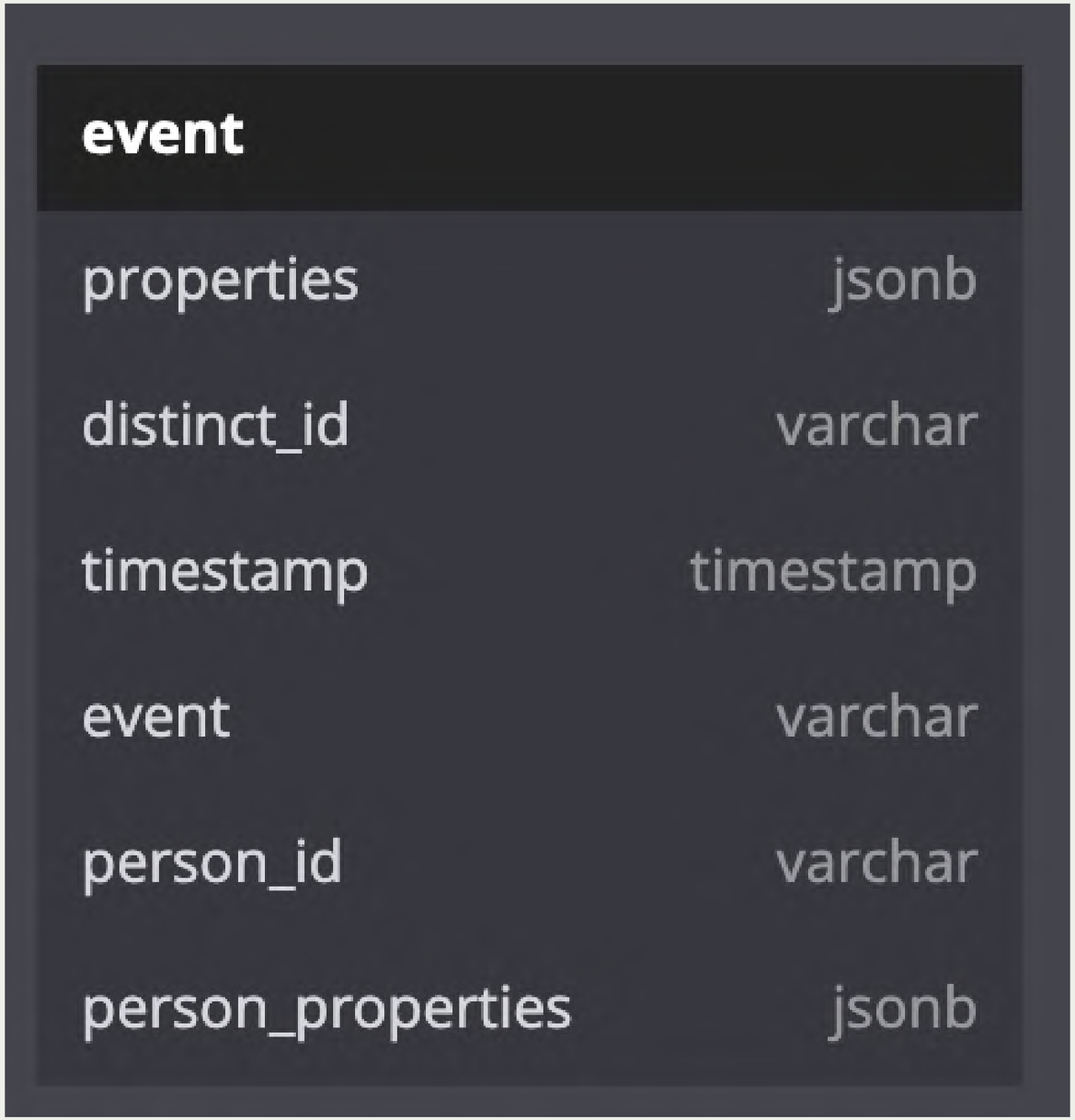
Computed a few seconds ago

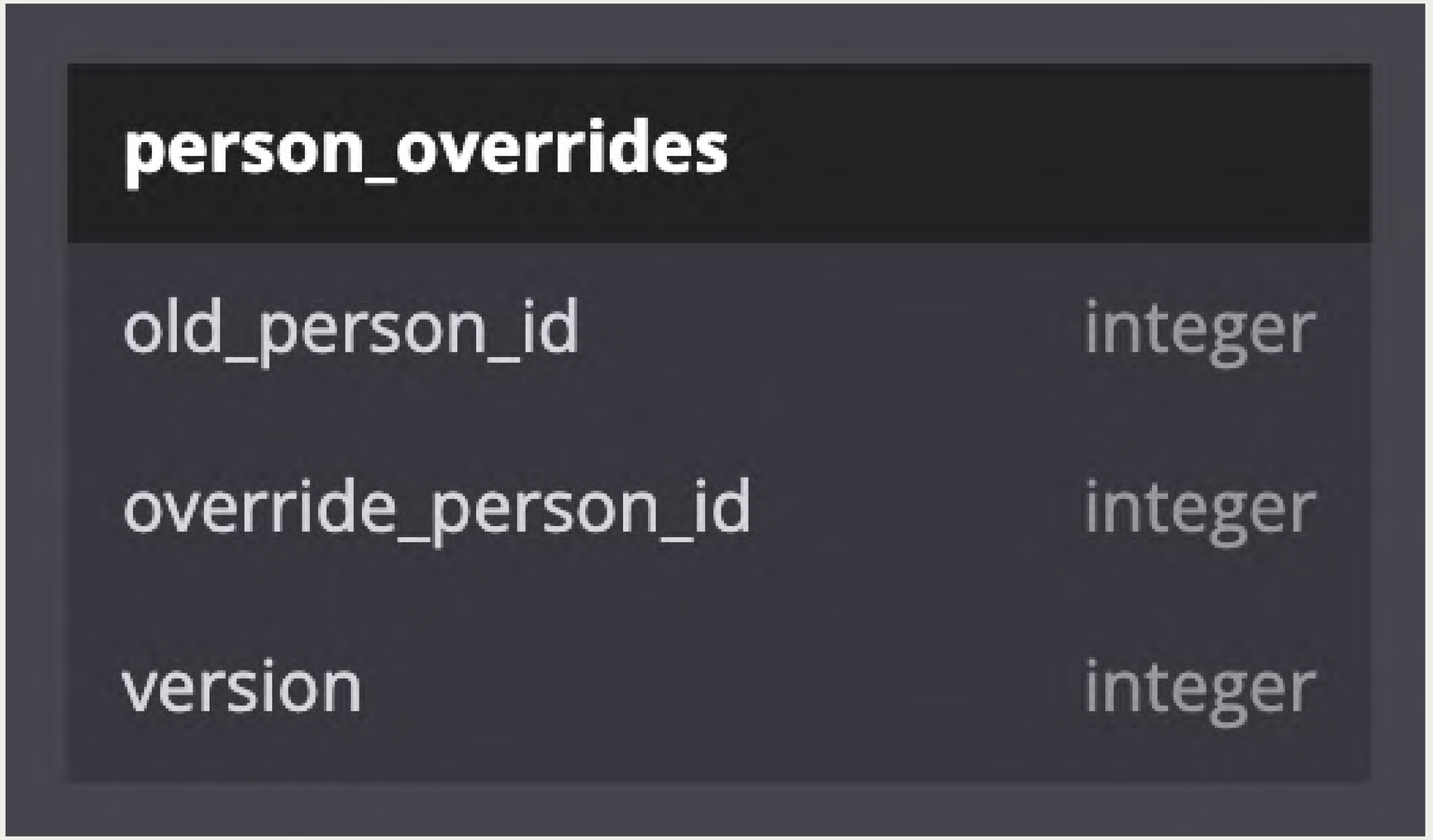
Show legend




event
properties jsonb
distinct_id varchar
timestamp timestamp
event varchar
*
1
person_distinct_id
person_id integer
distinct_id varchar
*
1
person
id integer
properties jsonb

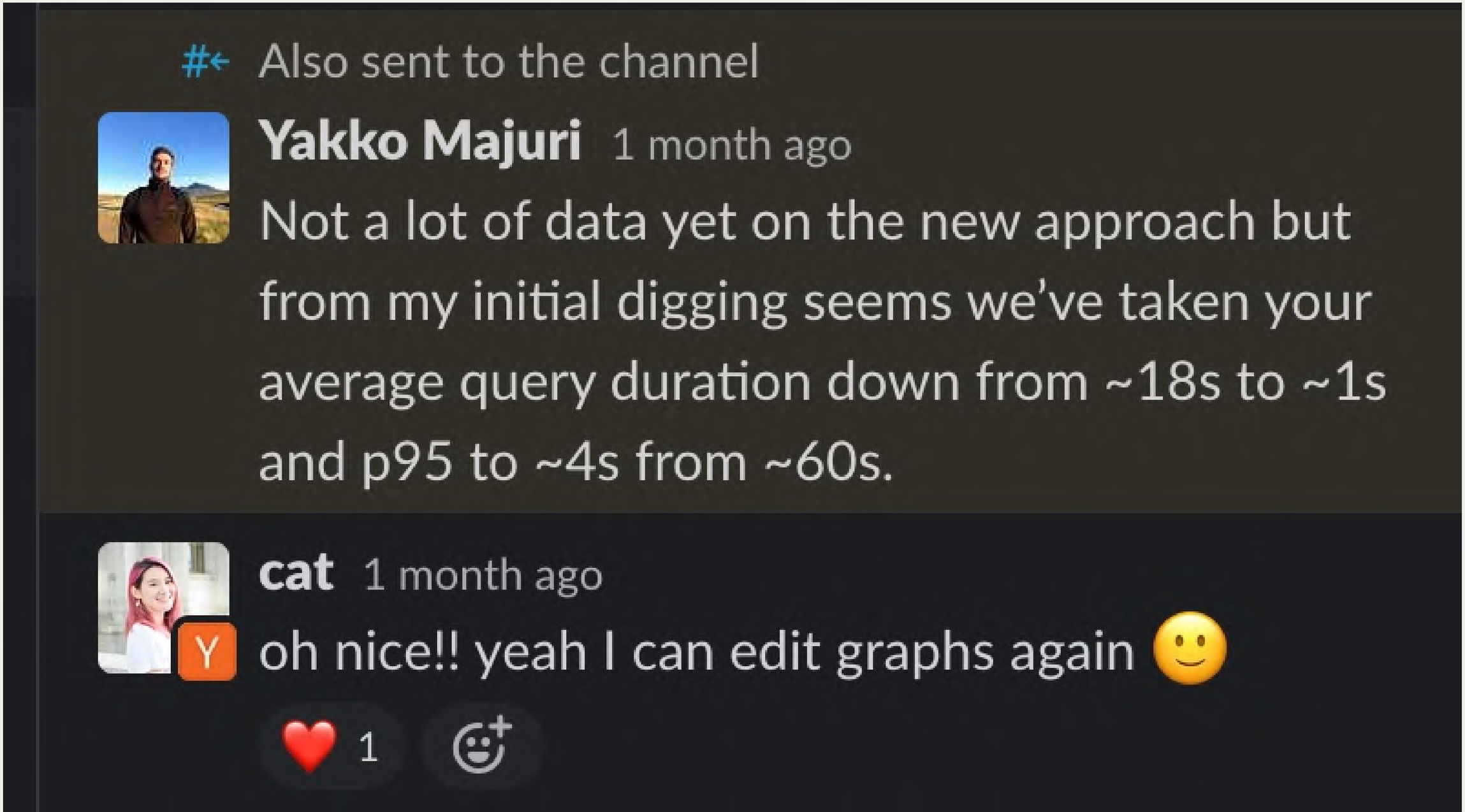
```sql
SELECT
    count(DISTINCT pdi.person_id) AS total,
    toStartOfDay(timestamp) AS date
FROM
    events e
    INNER JOIN (
        SELECT distinct_id, argMax(person_id, version) as person_id
        FROM person_distinct_id2
        GROUP BY distinct_id
        HAVING argMax(is_deleted, version) = 0
    ) AS pdi
    ON e.distinct_id = pdi.distinct_id
WHERE
    event = '$pageview'
```


event
properties jsonb
distinct_id varchar
timestamp timestamp
event varchar
person_id varchar
person_properties jsonb


person_overrides
old_person_id integer
override_person_id integer
version integer

PostH

```sql
SELECT
    count(
        DISTINCT if(
            notEmpty(overrides.person_id),
            overrides.person_id,
            e.person_id
        )
    ) AS total,
    toStartOfDay(timestamp) AS date
FROM
    events e
    LEFT OUTER JOIN (
        SELECT
            argMax(override_person_id, version) as person_id,
            old_person_id
        FROM
            person_overrides
        GROUP BY
            old_person_id
    ) AS overrides ON e.person_id = overrides.old_person_id
WHERE
    event = '$pageview'
```


Also sent to the channel
Yakko Majuri 1 month ago
Not a lot of data yet on the new approach but from my initial digging seems we've taken your average query duration down from ~18s to ~1s and p95 to ~4s from ~60s.
cat 1 month ago
oh nice!! yeah I can edit graphs again
1

```
SQUASH_EVENTS_QUERY = """
ALTER TABLE
    {database}.sharded_events
UPDATE
    person_id = dictGet('{database}.{dictionary_name}', 'override_person_id', (toInt32(team_id), person_id))
IN PARTITION
    %(partition_id)s
WHERE
    dictHas('{database}.{dictionary_name}', (toInt32(team_id), person_id))
    {team_id_filter}
    AND created_at <= %(latest_created_at)s;
"""
```

PostHog

Speeding up queries

Your query is taking a long time to complete. **We're still working on it.**
See below some options to speed things up.

Click here to speed up calculation with 10% sampling

In order to improve the performance of the query, you can also try to reduce the date range of your query, remove breakdowns, or get in touch with us by submitting a bug report.

Query ID: ee21e4a4-47cf-45e5-a0cf-b99b41273661

Your query is taking a long time to complete. **We're still working on it.**
See below some options to speed things up.

Click here to speed up calculation with 1% sampling

In order to improve the performance of the query, you can also try to reduce the date range of your query, remove breakdowns, or get in touch with us by submitting a bug report.

Query ID: 89e395ba-b8ac-4cad-afd3-354d052cd380

Introducing HouseWatch

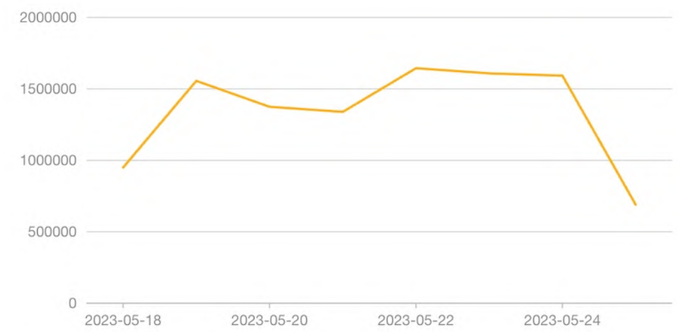
Overview
Query performance
Running queries
Schema stats
Disk usage
Logs
Errors
Query editor
Operations

Overview

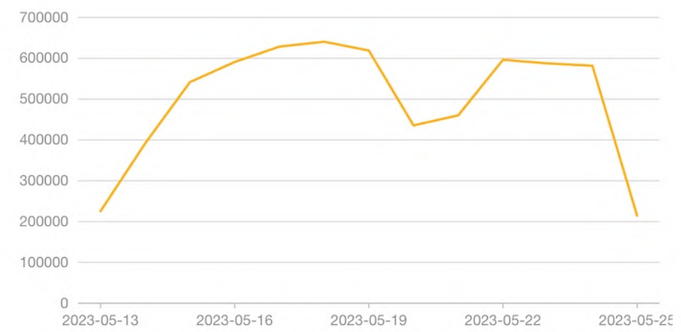
ClickHouse tip of the day

If you store JSON data in a VARCHAR column, consider materializing frequently acessed properties using materialized columns for much faster queries.

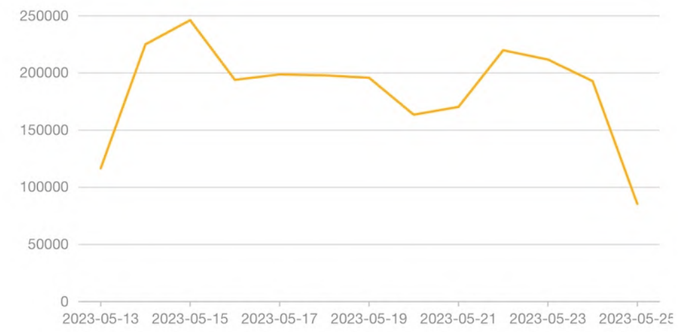
Number of queries



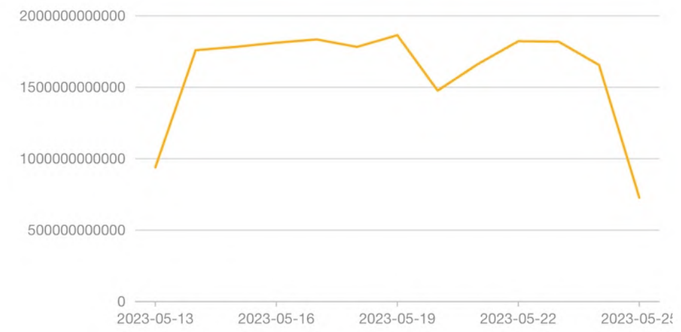
Data read (GB)



Memory usage (GB)



CPU usage (seconds)



Overview
Query performance
Running queries
Schema stats
Disk usage
Logs
Errors
Query editor
Operations

Query performance

Click on queries to display more details.

Query	Avg time (ms)	Calls / min	% of all iops	% of runtime	Total iops
WITH step_runs AS (SELECT ... FROM system.numbers LIMIT ifNull(bin_count, $1) + $2) fill USING (bin_from_seconds) ORDER BY bin_from_seconds	83302ms	0.012	1.5%	0.1%	51.59 TiB
SELECT ... FROM person_distinct_id2 WHERE team_id = $1 GROUP BY distinct_id HAVING argMax(is_deleted, version) = $2) AS pdi ON e.distinct_id = pdi.distinct_id WHERE team_id = $3 AND event IN [$4..] AND…	67510ms	0.001	0.5%	0.0%	16.16 TiB
SELECT ... FROM person_overrides WHERE team_id = $1 GROUP BY old_person_id) AS overrides ON e.person_id = overrides.old_person_id WHERE team_id = $2 AND event = $3 AND toTimeZone(timestamp, $4)…	56684ms	0.001	0.2%	0.0%	6.29 TiB
WITH actor_query AS (WITH $1 as period, NULL as breakdown_values_filter, NULL as selected_interval, returning_event_query as (SELECT ... FROM actor_query AS actor_activity GROUP BY breakdown_values,…	53109ms	0.001	0.2%	0.0%	7.00 TiB
SELECT ... FROM person_overrides WHERE team_id = $1 GROUP BY old_person_id) AS overrides ON e.person_id = overrides.old_person_id WHERE e.team_id = $2 AND event = $3 AND (((NOT ("mat_pp_email"…	52134ms	0.001	0.2%	0.0%	5.45 TiB
INSERT INTO cohortpeople SELECT ... FROM cohortpeople WHERE team_id = $1 AND cohort_id = $2 AND version < $3 AND sign = $4	50269ms	0.001	0.1%	0.0%	4.98 TiB
SELECT ... FROM events e WHERE team_id = $1 AND event IN [$2..] AND toTimeZone(timestamp, $3) >= toDateTime($4..) AND toTimeZone(timestamp, $5) <= toDateTime($6..) AND (((NOT has([$7..],…	42990ms	0.001	0.3%	0.0%	11.23 TiB
SELECT ... FROM events e WHERE team_id = $1 AND ((event = $2) OR (event = $3) OR (event = $4) OR (event = $5) OR (event = $6)) AND toTimeZone(timestamp, $7) >= toDateTime($8..) AND toTimeZone(timestamp, $9)…	42844ms	0.002	0.2%	0.0%	5.42 TiB
SELECT ... FROM person_overrides WHERE team_id = $1 GROUP BY old_person_id) AS overrides ON e.person_id = overrides.old_person_id WHERE team_id = $2 AND event = $3 AND toTimeZone(timestamp, $4)…	40497ms	0.001	0.2%	0.0%	5.88 TiB
SELECT ... FROM events e WHERE team_id = $1 AND ((event = $2) OR (event = $3) OR (event = $4) OR (event = $5) OR (event = $6)) AND toTimeZone(timestamp, $7) >= toDateTime($8..) AND toTimeZone(timestamp, $9)…	39756ms	0.001	0.2%	0.0%	6.05 TiB

HouseWatch

ch8.posthog.net

- Overview
- Query performance
- Running queries
- Schema stats
- Disk usage
- Logs
- Errors
- Query editor
- Operations

Running queries

Query	Elapsed time	Rows read	Memory Usage	Actions
`EXPLAIN header=1, indexes=1 /* user_id:25072 request:_api_projects_19279_insights_959062_ */ WITH step_runs AS ( S...` Expand	23.162638325	~108990275/155010358	1.33 GiB	Kill query

< 1 >

PostHog

HouseWatch

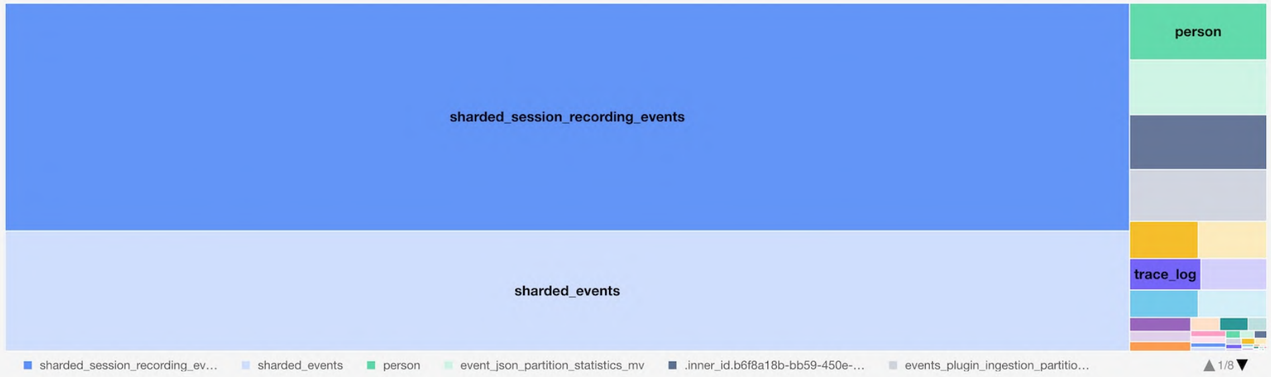
- Overview
- Query performance
- Running queries
- Schema stats
- Disk usage
- Logs
- Errors
- Query editor
- Operations

ch8.posthog.net

Schema stats

Largest tables

Click on the rectangles to get further information about parts and columns for the table. Note that this only covers data stored on the connected node, not the whole cluster.



All tables

Name	Size	Rows	Engine	Partition Key
sharded_session_recording_events	13.12 TiB	657641622	ReplicatedReplacingMergeTree	toYYYYMMDD(timestamp)
sharded_events	6.92 TiB	33804432203	ReplicatedReplacingMergeTree	toYYYYMM(timestamp)
person	409.01 GiB	1703981004	ReplicatedReplacingMergeTree	
event_json_partition_statistics_mv	394.38 GiB	13447450251	MaterializedView	
.inner_id.b6f8a18b-bb59-450e-b198-456ccd524eb6	394.38 GiB	13447450251	AggregatingMergeTree	
events_plugin_ingestion_partition_statistics	372.71 GiB	9366372810	ReplicatedAggregatingMergeTree	
.inner_id.5d8e1c6e-0337-4a0e-bd46-5baa1b136978	134.06 GiB	9947057803	MergeTree	
persons_properties_view	134.06 GiB	9947057803	MaterializedView	
trace_log	117.87 GiB	5646664094	MergeTree	toYYYYMM(event_date)
person_distinct_id2	109.00 GiB	1914693902	ReplicatedReplacingMergeTree	

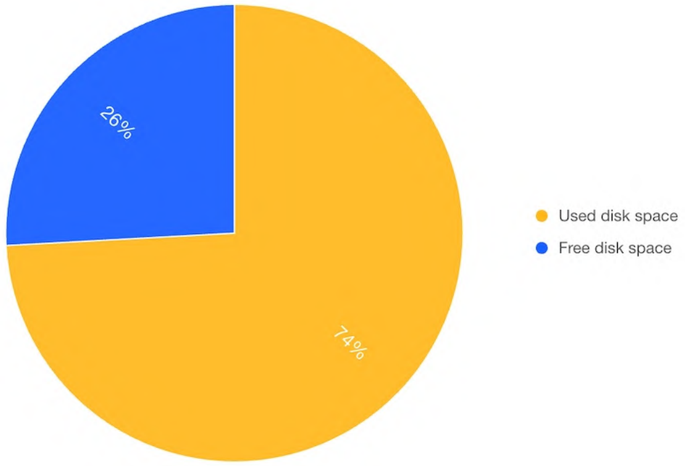
< 1 2 3 4 5 6 > 10 / page

HouseWatch

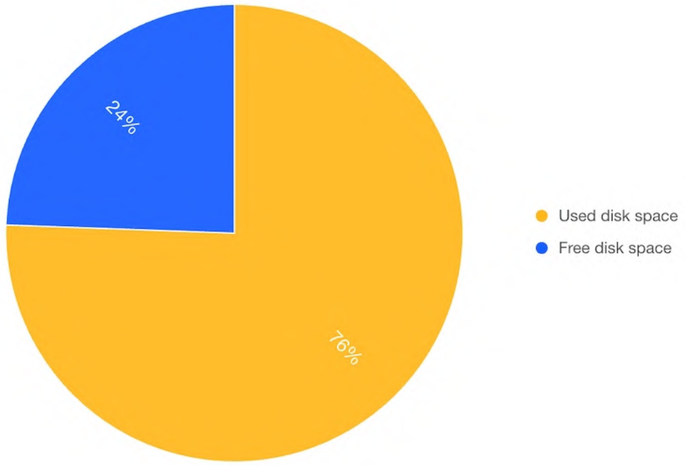
- Overview
- Query performance
- Running queries
- Schema stats
- Disk usage
- Logs
- Errors
- Query editor
- Operations

Disk usage

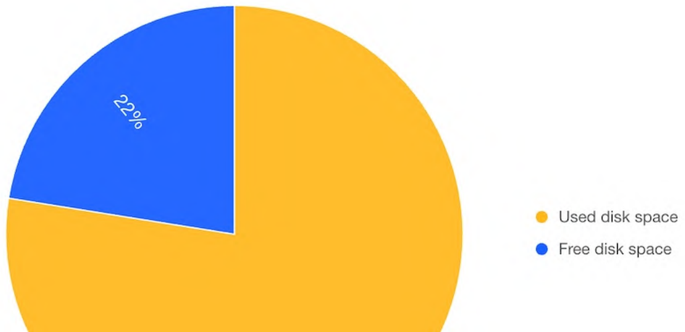
ch9.posthog.net



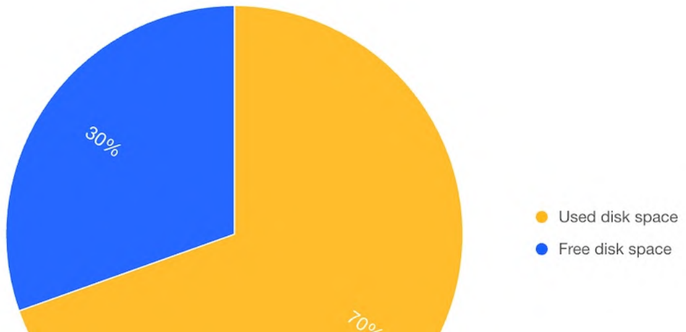
ip-172-31-44-98



ip-172-31-47-136



ip-172-31-65-101

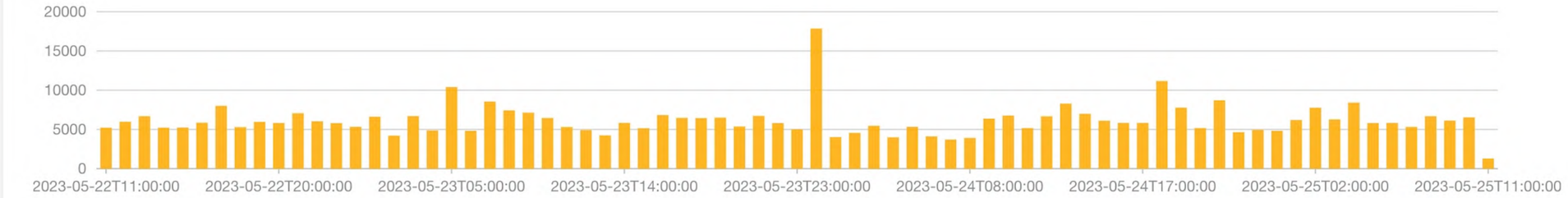


HouseWatch

ch8.posthog.net

- Overview
- Query performance
- Running queries
- Schema stats
- Disk usage
- Logs
- Errors
- Query editor
- Operations

Logs



Time	Level	Host	Message
2023-05-25T11:07:23	Error	ip-172-31-65-101	auto DB::StorageReplicatedMergeTree::processQueueEntry(ReplicatedMergeTreeQueue::SelectedEntryPtr):: (anonymous class)::operator()(DB::StorageReplicatedMergeTree::LogEntryPtr &) const: Poco::Except... Expand
2023-05-25T11:07:20	Error	ip-172-31-65-101	auto DB::StorageReplicatedMergeTree::processQueueEntry(ReplicatedMergeTreeQueue::SelectedEntryPtr):: (anonymous class)::operator()(DB::StorageReplicatedMergeTree::LogEntryPtr &) const: Poco::Except... Expand
2023-05-25T11:07:19	Error	ip-172-31-65-101	auto DB::StorageReplicatedMergeTree::processQueueEntry(ReplicatedMergeTreeQueue::SelectedEntryPtr):: (anonymous class)::operator()(DB::StorageReplicatedMergeTree::LogEntryPtr &) const: Poco::Except... Expand
2023-05-25T11:07:18	Error	ip-172-31-65-101	auto DB::StorageReplicatedMergeTree::processQueueEntry(ReplicatedMergeTreeQueue::SelectedEntryPtr):: (anonymous class)::operator()(DB::StorageReplicatedMergeTree::LogEntryPtr &) const: Poco::Except... Expand

Overview
Query performance
Running queries
Schema stats
Disk usage
Logs
Errors
Query editor
Operations

Operations (Alpha)

Create long-running operations to run in the background in your ClickHouse cluster. Useful for large data migrations, specify SQL commands to run in order with corresponding rollbacks, such that if the operation fails, you rollback to a safe state.

Please exercise caution! This funtionality is still in Alpha.

Operations | Create new operation

Details

Name

Description

Operations

#1

```
CREATE TABLE test_table ( foo String ) Engine=MergeTree()
```

```
DROP TABLE IF EXISTS test_table
```

github.com/posthog/housewatch

Thank you!

We're hiring posthog.com/careers