



# Lightning talk



“My journey as a ClickHouse beginner”

By: Tsvetan Stoychev

# About me / hobbies



I love to walk with Lucky



I do street art

# About me / work

I work at

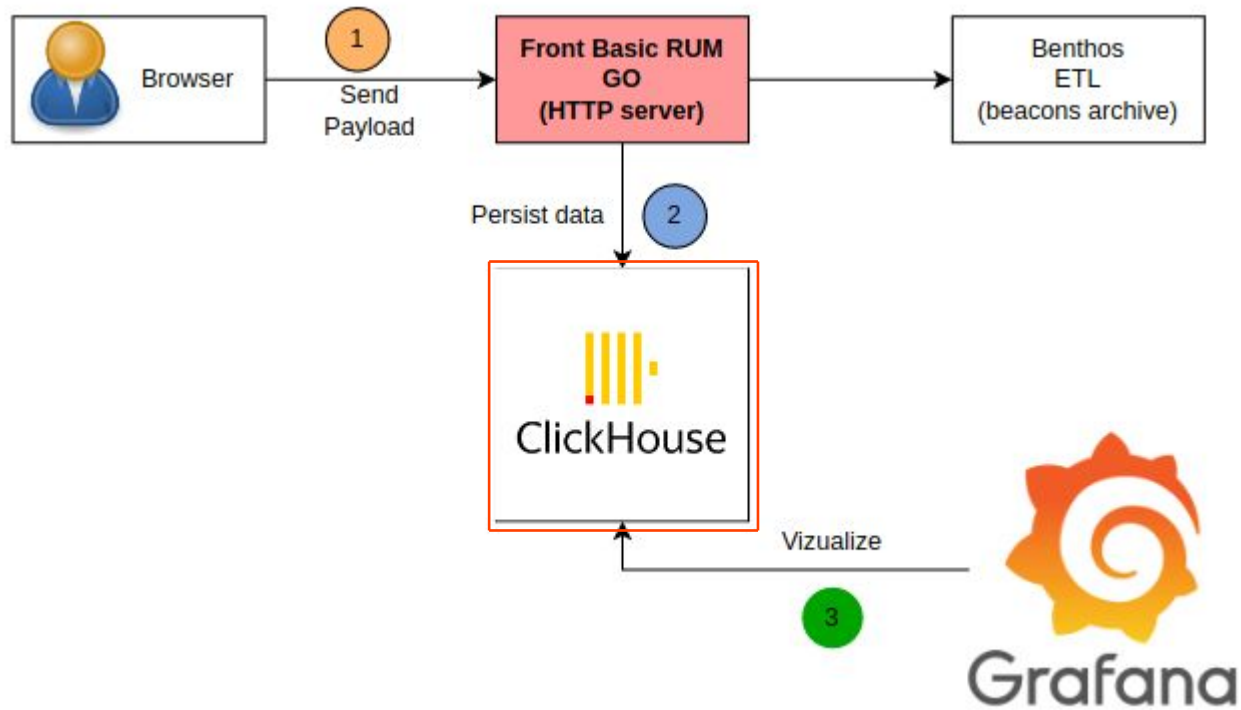


I use ClickHouse for



a hobby project

# Basic RUM in general



# I am a ClickHouse

## Script Kiddie



# Script kiddie

- “A **script kiddie**, skiddie, or skid is a relatively **unskilled** individual who uses scripts or programs, such as a web shell, developed by others to **attack computer systems** and networks and deface websites, according to the programming and hacking cultures.”

Wikipedia





# What I need as ClickHouse beginner?

(1) Playground for experiments



(2) Study/Access to information

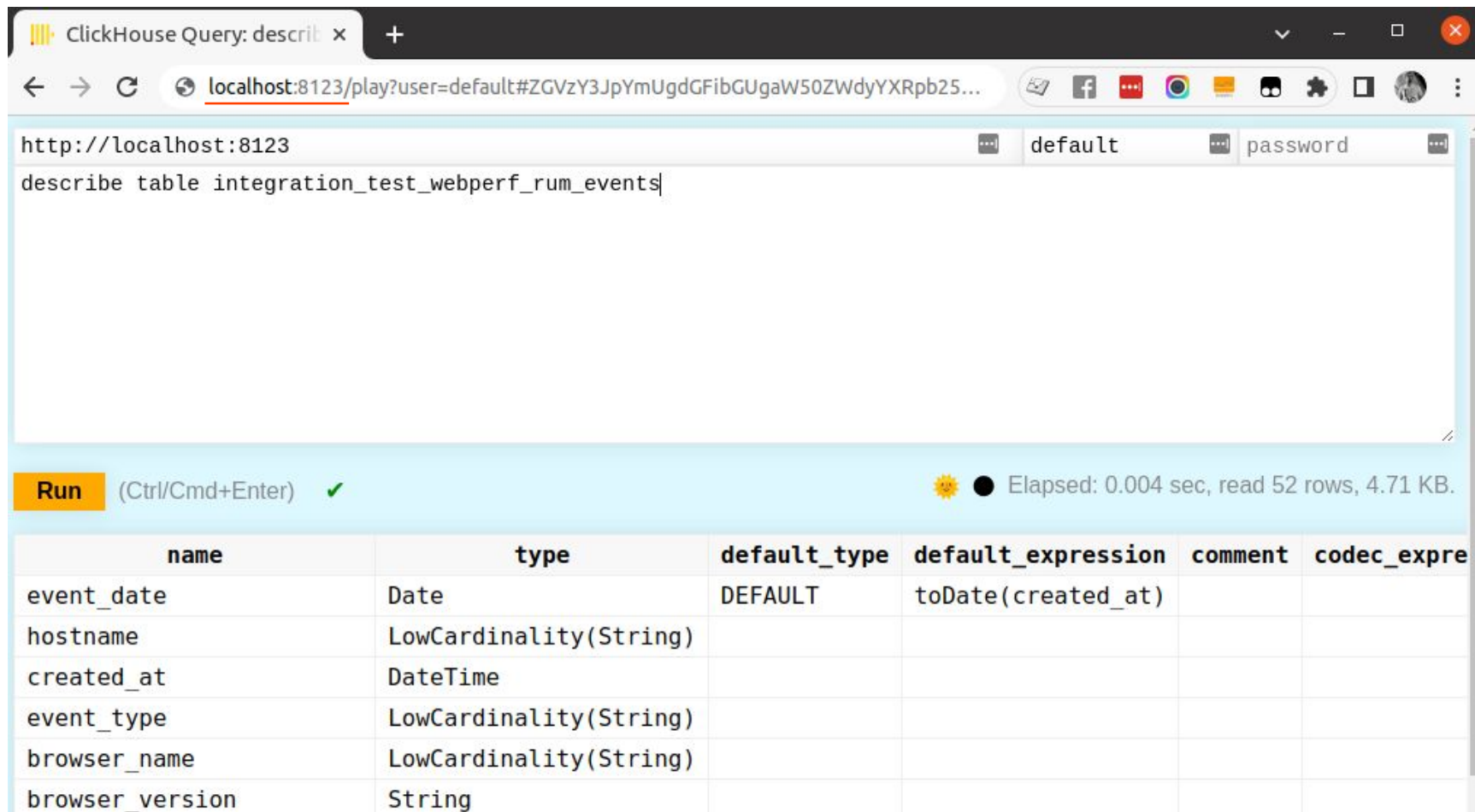


# (1) Playground for experiments





# (1) Playground for experiments



The screenshot shows the ClickHouse Query Playground interface. The browser address bar indicates the URL `localhost:8123/play?user=default#ZGVzY3JpYmUgdGFibGUgaW50ZWdyYXRpb25...`. The query input field contains the SQL command `describe table integration_test_webperf_rum_events`. Below the input field, a light blue bar displays a yellow "Run" button, the keyboard shortcut `(Ctrl/Cmd+Enter)`, a green checkmark, and a status message: "Elapsed: 0.004 sec, read 52 rows, 4.71 KB." Below this bar, a table displays the schema information for the specified table.

name	type	default_type	default_expression	comment	codec_expre
event_date	Date	DEFAULT	toDate(created_at)		
hostname	LowCardinality(String)				
created_at	DateTime				
event_type	LowCardinality(String)				
browser_name	LowCardinality(String)				
browser_version	String				

# (1) Playground for experiments

The screenshot displays the Grafana 'Experimental / Edit Panel' interface. The top navigation bar includes a back arrow, the title 'Experimental / Edit Panel', and buttons for 'Discard', 'Save', and 'Apply'. Below the navigation bar, there are filters for 'First Byte', '100', and 'All'. The main content area is divided into two sections: a table of data and a query editor. The table, titled 'Most popular', shows columns 'cnt' and 'url' with data rows. The query editor shows a SQL query for counting URLs. The right sidebar contains 'Panel options' and 'Table' settings.

1. Query editor

2. SQL query

3. Table data

4. Panel options

cnt	url
8679	https://calendar.perfplanet.com/2021/
5489	https://calendar.perfplanet.com/2021/js-self-profiling-api-in-practice/
4806	https://calendar.perfplanet.com/2021/plugging-memory-leaks-in-your-app/
3816	https://calendar.perfplanet.com/2021/performance-implications-of-javascript-errors/
3769	https://calendar.perfplanet.com/2021/a-unified-theory-of-web-performance/
2784	https://calendar.perfplanet.com/2021/optimizing-state-management-in-react-applications/

```
SELECT
  count(url) as cnt,
  url
FROM $table
WHERE
  $timeFilter
  AND event_type = 'visit_page'
  AND hostname = 'calendar.perfplanet.com'
GROUP BY url
ORDER BY cnt DESC
```

Grafana as a playground

# (1) Playground for experiments

Experimental / Settings

General  
Annotations  
Variables  
Links  
**Versions**  
Permissions  
JSON Model

Save dashboard  
Save As...

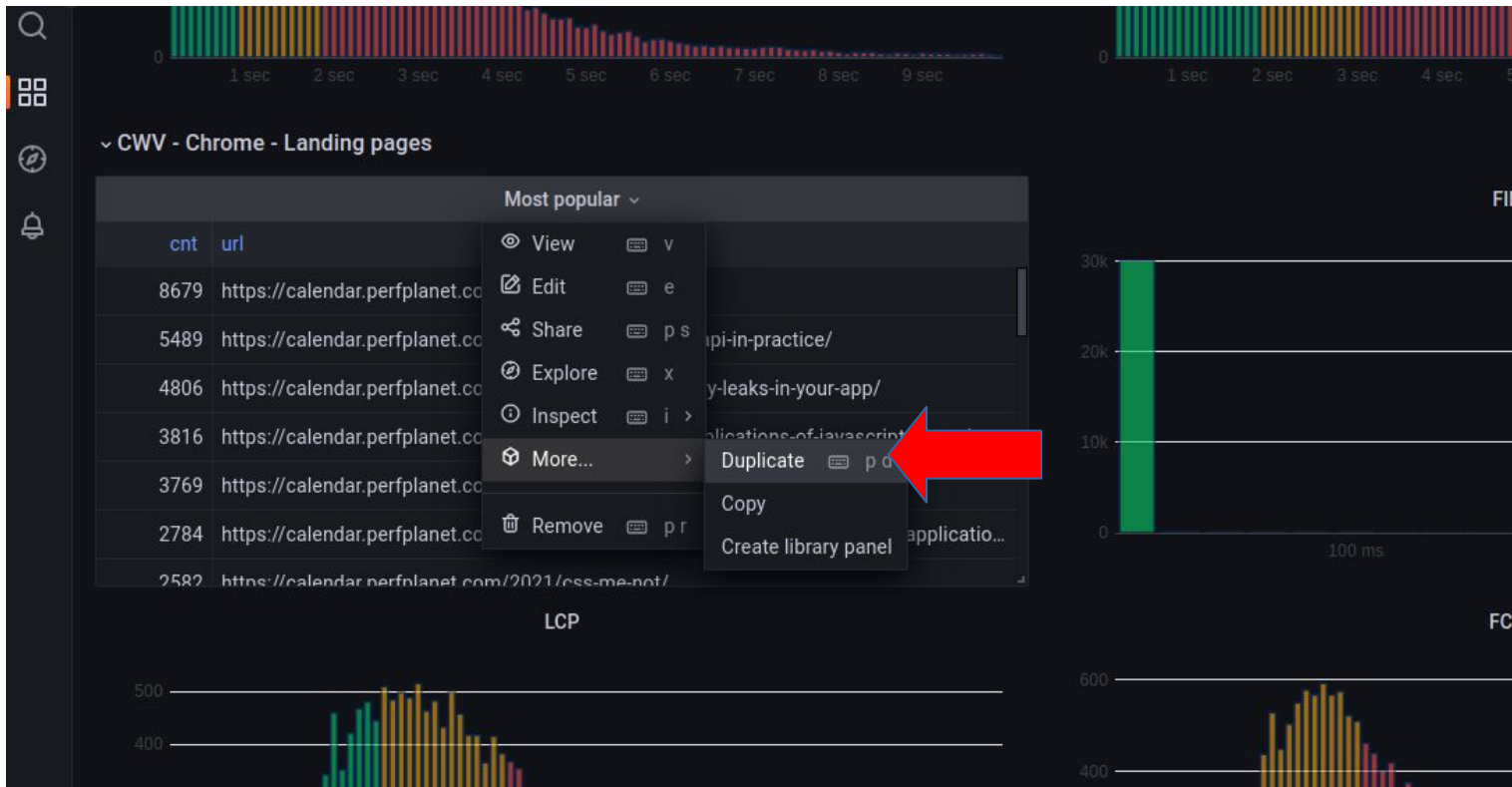
### Versions

	Version	Date	Updated by	Notes
<input type="checkbox"/>	19	2022-07-16 17:01:50		<b>Latest</b>
<input type="checkbox"/>	18	2022-07-16 17:01:19		Restore
<input type="checkbox"/>	17	2022-07-16 17:00:43		Restore
<input type="checkbox"/>	16	2022-07-16 17:00:03		Restore
<input type="checkbox"/>	15	2022-07-16 16:55:59		Restore
<input type="checkbox"/>	14	2022-07-16 16:41:54		Restore
<input type="checkbox"/>	13	2022-07-16 16:35:21		Restore
<input type="checkbox"/>	12	2022-07-16 16:28:19		Restore
<input type="checkbox"/>	11	2022-07-16 16:27:10		Restore
<input type="checkbox"/>	10	2022-07-16 16:16:56		Restore

Show more versions Compare versions

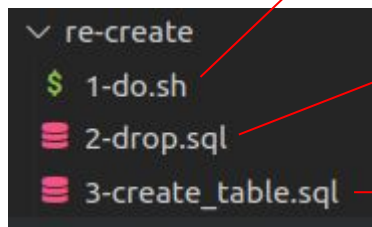
Grafana as a playground

# (1) Playground for experiments



Grafana as a playground

# (1) Playground for experiments



1

```
re-create > $ 1-do.sh
1  #!/bin/bash
2
3  dropSql=$(cat drop.sql)
4
5  createSql=$(cat create_table.sql)
6
7  docker exec benthos_workshop_clickhouse_server_1 clickhouse-client -q "${dropSql}"
8
9  docker exec benthos_workshop_clickhouse_server_1 clickhouse-client -q "${createSql}"
10
```

2

```
re-create > 2-drop.sql
1  DROP TABLE IF EXISTS webperf_rum_events
```

3

```
re-create > 3-create_table.sql
1  CREATE TABLE IF NOT EXISTS webperf_rum_events_dummy (
2      event_date Date DEFAULT toDate(created_at),
3      created_at DateTime,
4      event_type LowCardinality(String),
5      browser_name LowCardinality(String),
6      browser_version String,
7      device_manufacturer LowCardinality(String),
8      device_type LowCardinality(String),
9      user_agent String,
10     next_hop_protocol LowCardinality(String),
11     visibility_state LowCardinality(String),
12
13     session_id FixedString(43),
14     session_length UInt8,
15     url String,
```

# (1) Playground for experiments

```
insert_example.sql
1  ## EXAMPLE INSERT
2
3  INSERT INTO test_table SETTINGS input_format_skip_unknown_fields = true FORMAT JSONEachRow
4  {"created_at":"2022-06-11 13:15:13","c1":"34","c2":"8990","c3":"8990",.....,"c998":"100","c999":"100"}
5
6
7  ## EXAMPLE SCHEMA
8  CREATE TABLE IF NOT EXISTS webperf_rum_events_dummy (
9      event_date Date DEFAULT toDate(created_at),
10     created_at DateTime,
11     c1             Nullable(UInt16),
12     c2             |             Nullable(UInt16)
13 )
14 ENGINE = MergeTree()
15 PARTITION BY toYYYYMMDD(event_date)
16 ORDER BY (c1, event_date)
17 SETTINGS index_granularity = 8192
18
```




## (2) Study/Access to information



## (2) Study/Access to information

# help ▾

October 10th, 2021 ▾

 **Tsvetan Stoychev** 11:50 PM


I am building a Real User Monitoring system that visualizes bunch of interesting browser performance metrics in Grafana. There is a metric called Cumulative Layout Shift the web perf community cares about 3 thresholds that are considered as good, needs improvements and poor. I imported bunch of data and created the following query that shows percentages of each of the 3 thresholds. The query works just fine but I am wondering if it could be written in more elegant and perhaps efficient way.

I hope that I can get an opinion from the experts 😊

Here is what I have so far:


```
SELECT count() AS all_cls,
  (
    (SELECT count()
     FROM rum_data_flat
     WHERE simple_tm_cumulative_layout_shift IS NOT NULL
        AND simple_tm_cumulative_layout_shift > 0.25
        AND derived_bm_browser_name = 'Chrome') / (all_cls)) * 100 AS poor_cls,
  (
    (SELECT count()
     FROM rum_data_flat
     WHERE simple_tm_cumulative_layout_shift IS NOT NULL
        AND simple_tm_cumulative_layout_shift > 0.1
        AND simple_tm_cumulative_layout_shift <= 0.25
        AND derived_bm_browser_name = 'Chrome') / (all_cls)) * 100 AS medium_cls,
  (
    (SELECT count()
```

Thread # help


 **Robert Hodges (Altinity)** 10 months ago

Have a look at countIf(). It's an idiomatic way to count rows that match a particular set of conditions, i.e., a predicate. Here's a trivial example:

```
select count(),
  countIf(number, number > 0 and number <=3)
first_third,
  countIf(number, number > 3 and number <=6)
second_third,
  countIf(number, number > 6 and number <=9)
third_third,
  countIf(number, number <= 0 or number > 9) tails
from
(
  select number from system.numbers limit 10
)
```

 **Robert Hodges (Altinity)** 10 months ago

Your approach will scan the table 4 times if I'm reading the query correctly. The above query works in a single scan.

 **Robert Hodges (Altinity)** 10 months ago

Here's a doc page on aggregate function combinators that may help you further. It's a fun feature of ClickHouse.  
<https://clickhouse.com/docs/en/sql-reference/aggregate-functions/combinators/>




Really good support on the ClickHouse slack space!

## (2) Study/Access to information

The screenshot displays the ClickHouse YouTube channel page. The channel name is "ClickHouse" with 3.82K subscribers. The "SUBSCRIBED" button is highlighted with a red arrow. The page shows a list of video uploads, including:

- ClickHouse v22.07 Release Webinar (900 views • Streamed 5 days ago)
- Alexey's Favorite ClickHouse Features (Chinese...) (824 views • 8 days ago)
- ClickHouse at JD.com (Chinese Community...) (141 views • 9 days ago)
- ClickHouse in YongHui SuperMarket, Retail Use Ca... (121 views • 11 days ago)
- BillBili: from Elasticsearch to ClickHouse, Log Analysis Practice in bilibili (257 views • 12 days ago)
- Data Synchronization Through Apache SeaTunnel... (19:07)
- ClickHouse Realtime Data Warehouse in... (42:08)
- ClickHouse Virtual Meetup May (1:37:04)
- ClickHouse v22.06 Release Webinar (1:09:29)
- ClickHouse Meetup Amsterdam, June 8, 2022 (1:50:09)

# (2) Study/Access to information

 [PRODUCTS](#) [LEARN](#) [DEVELOPERS](#) [BLOG](#) [ABOUT US](#) [CONTACT US](#) [START FOR FREE](#)

## DATA WAREHOUSE BASICS

### Introduction to Data Warehouses

Whether you are a beginner or an experienced user, Altinity's ClickHouse training turns your team into ClickHouse experts.

**Overview**

An introduction to data warehouses for newcomers to SQL. Students will learn how data warehouses work, how to install ClickHouse, and how to issue standard SQL commands to build reports and dashboards in ClickHouse. This class is a recommended prerequisite for people planning to take [ClickHouse 101](#) who have not previously used SQL databases.

The class includes a short lab on a live ClickHouse server to cement understanding of SQL commands. Students can complete exercises using a web browser.

First name\*

Last name\*

Company name\*

Email\*

## (2) Study/Access to information

The screenshot shows the YouTube channel page for Altinity, Inc. The browser address bar displays `youtube.com/c/AltinityLtd/videos`. The search bar contains the text `altinity clickhouse`. The channel banner features a racing car wheel and the text **THE REAL-TIME DATA COMPANY**, with the tagline `Altinity.Cloud Anywhere` in the bottom right corner. The channel name **Altinity, Inc.** is displayed next to its logo. Below the name are navigation tabs: **HOME**, **VIDEOS** (selected), **PLAYLISTS**, **COMMUNITY**, **CHANNELS**, and **ABOUT**. A red arrow points to the **SUBSCRIBED** button, which is accompanied by a notification bell icon. The **Uploads** section is visible, showing a list of videos with their titles and durations. The left sidebar contains the standard YouTube navigation menu, including Home, Explore, Shorts, Subscriptions, Library, History, Your videos, Watch later, Liked videos, and Show more. A **SUBSCRIPTIONS** section is also visible at the bottom of the sidebar.

youtube.com/c/AltinityLtd/videos

altinity clickhouse

YouTube DE

Home Explore Shorts Subscriptions

Library History Your videos Watch later Liked videos Show more

SUBSCRIPTIONS

БЪЛГАРСКА СВО... (x)

Контракоментар ...

Altinity, Inc.

HOME VIDEOS PLAYLISTS COMMUNITY CHANNELS ABOUT

Uploads SORT BY

**{ All About JSON and ClickHouse }**  
TIPS, TRICKS, & NEW FEATURES!  
1:06:57

Moving Big Data from Snowflake to ClickHouse for Fun and Profit  
Robert Hodges - Altinity Engineering  
20 June 2022  
38:29

**PERSONALITY**  
Building Fast ClickHouse Analytics for MySQL Transaction Data Using Kubernetes  
Kurt Stammen - Data Lead, Developer  
Alexander Alexandrov - Security  
54:58

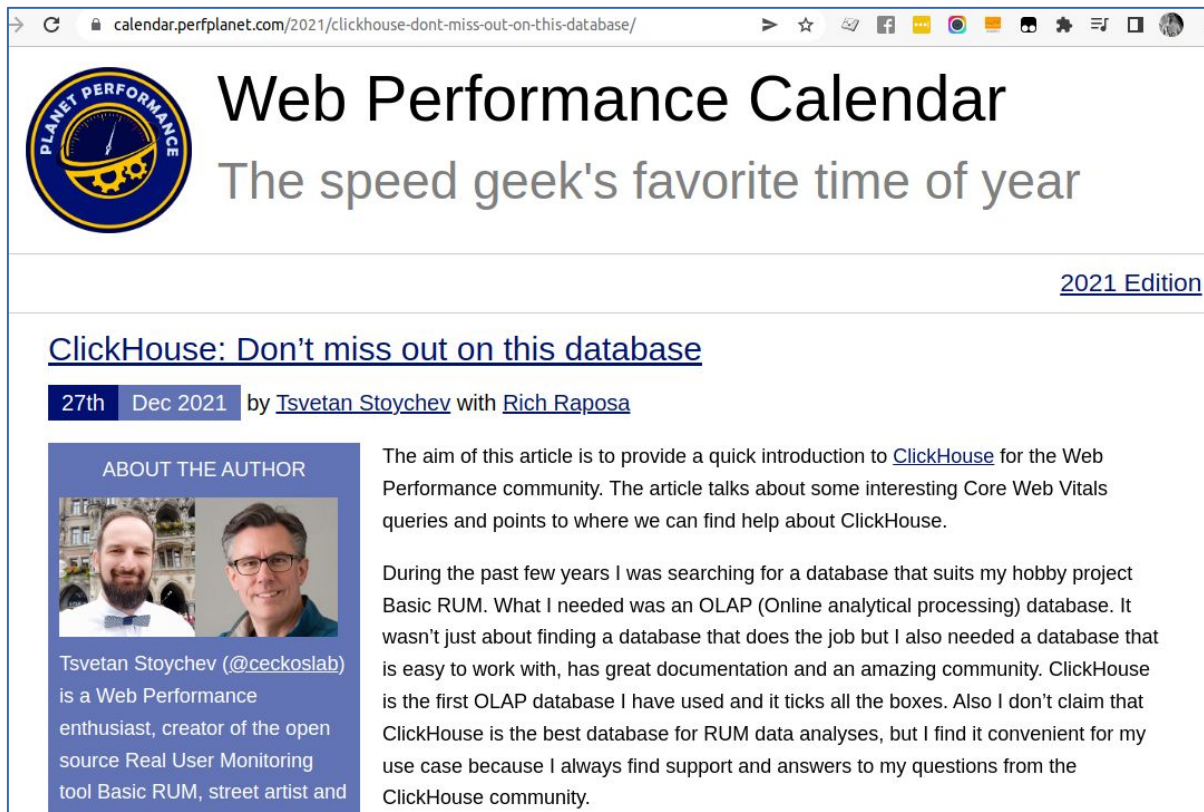
**CLICKHOUSE COMMUNITY MEET**  
1:48:56

**HIGH PERFORMANCE, HIGH RELIABILITY**  
Data Loading on ClickHouse  
Altinity  
1:00:53

**CLICKHOUSE DEFENSE AGAINST THE DARK ARTS**  
Intro to Security and Privacy  
Altinity  
1:03:15




## (2) Study/Access to information



The screenshot shows a web browser window with the address bar displaying "calendar.perfplanet.com/2021/clickhouse-dont-miss-out-on-this-database/". The page features the "Planet Performance" logo, which is a blue circle with a yellow clock face and the text "PLANET PERFORMANCE" around it. The main heading is "Web Performance Calendar" with the subtitle "The speed geek's favorite time of year". A link for the "2021 Edition" is visible on the right. The article title is "ClickHouse: Don't miss out on this database", dated "27th Dec 2021", and written by "Tsvetan Stoychev with Rich Raposa". An "ABOUT THE AUTHOR" section includes a photo of Tsvetan Stoychev and text describing him as a Web Performance enthusiast, creator of the open source Real User Monitoring tool Basic RUM, street artist, and more. The article text discusses the aim of providing a quick introduction to ClickHouse for the Web Performance community and shares the author's personal experience with using ClickHouse as an OLAP database for a hobby project called Basic RUM.

calendar.perfplanet.com/2021/clickhouse-dont-miss-out-on-this-database/


 Web Performance Calendar  
The speed geek's favorite time of year

[2021 Edition](#)

[ClickHouse: Don't miss out on this database](#)

27th Dec 2021 by [Tsvetan Stoychev](#) with [Rich Raposa](#)

**ABOUT THE AUTHOR**

  
Tsvetan Stoychev ([@ceckoslab](#)) is a Web Performance enthusiast, creator of the open source Real User Monitoring tool Basic RUM, street artist and

The aim of this article is to provide a quick introduction to [ClickHouse](#) for the Web Performance community. The article talks about some interesting Core Web Vitals queries and points to where we can find help about ClickHouse.

During the past few years I was searching for a database that suits my hobby project Basic RUM. What I needed was an OLAP (Online analytical processing) database. It wasn't just about finding a database that does the job but I also needed a database that is easy to work with, has great documentation and an amazing community. ClickHouse is the first OLAP database I have used and it ticks all the boxes. Also I don't claim that ClickHouse is the best database for RUM data analyses, but I find it convenient for my use case because I always find support and answers to my questions from the ClickHouse community.

⚡ I learn best when I try to teach someone! ⚡

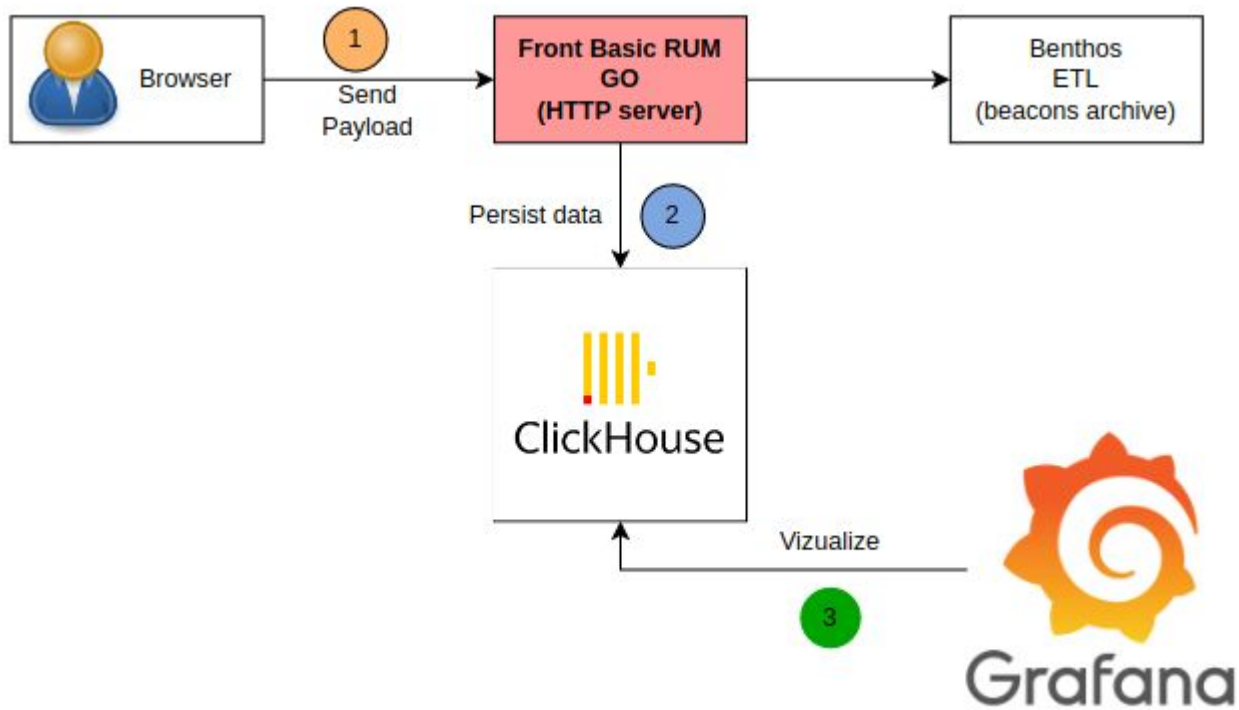


In closing



Kudos for the great ClickHouse community!

# But where to focus first?



 The END 