

Training – Day #3

15 June 2023 04:31

Chart command (continued)

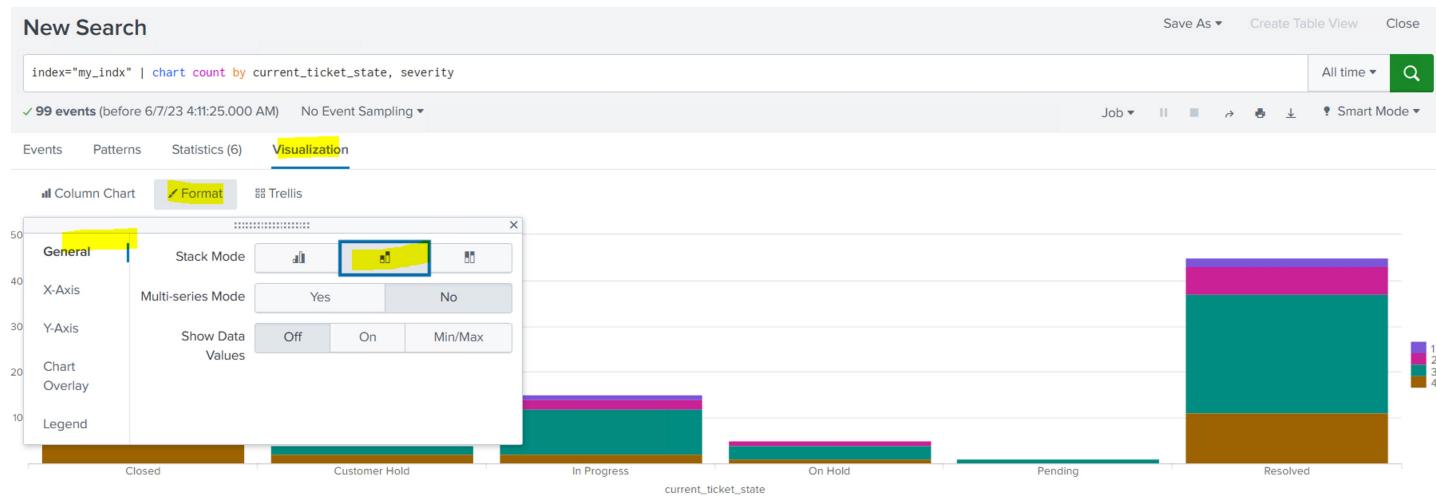
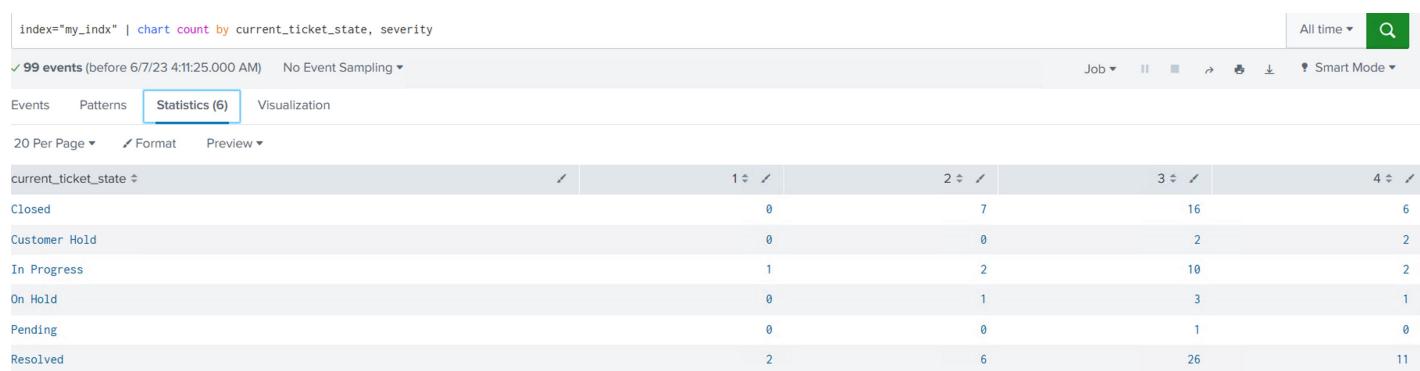
#Chart Overlay

It is used to show threshold line on above the current graph.

#Stack in chart :

Statistics will be same but in visualization, pattern will be different.

index="my_idx" | chart count by current_ticket_state, severity



Note : Log and stacking cannot be enabled at the same time. It will throw error.

Question : Difference between statistical output of below two queries :

Answer :

SPL1 : index="my_idx" | chart count by current_ticket_state, severity

index="my_idx" | chart count by current_ticket_state, severity

✓ 99 events (before 6/7/23 4:17:16.000 AM) No Event Sampling ▾ All time ▾

Events Patterns Statistics (6) Visualization

20 Per Page ▾ Format Preview ▾

current_ticket_state	1	2	3	4
Closed	0	7	16	6
Customer Hold	0	0	2	2
In Progress	1	2	10	2
On Hold	0	1	3	1
Pending	0	0	1	0
Resolved	2	6	26	11

SPL2 : index="my_idx" | stats count by current_ticket_state, severity

index="my_idx" | stats count by current_ticket_state, severity

✓ 99 events (before 6/7/23 4:15:28.000 AM) No Event Sampling ▾ All time ▾

Events Patterns Statistics (17) Visualization

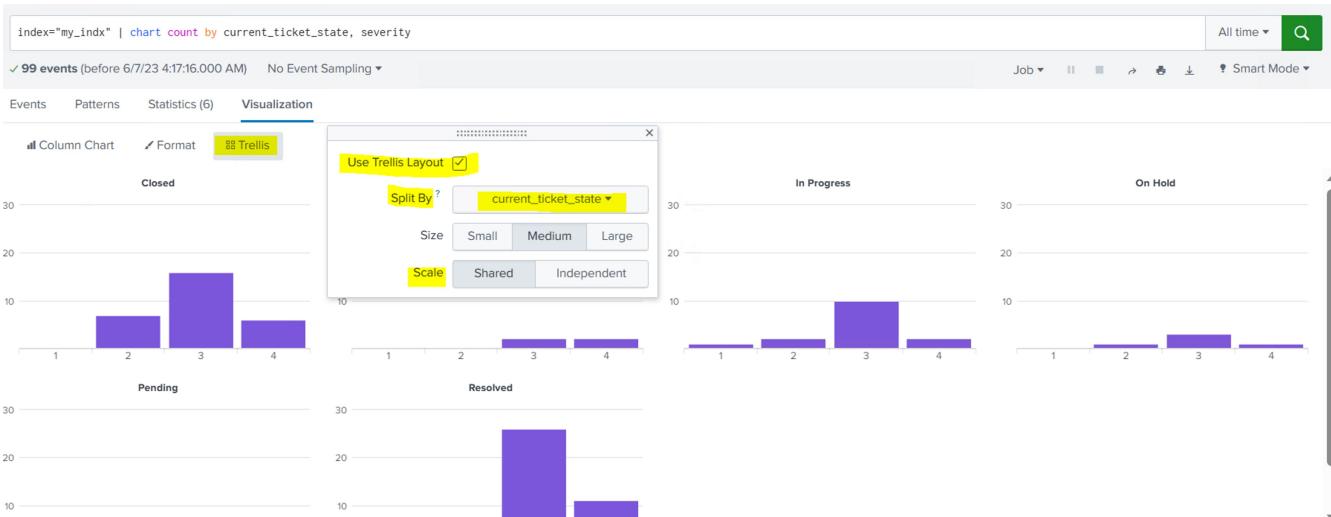
20 Per Page ▾ Format Preview ▾

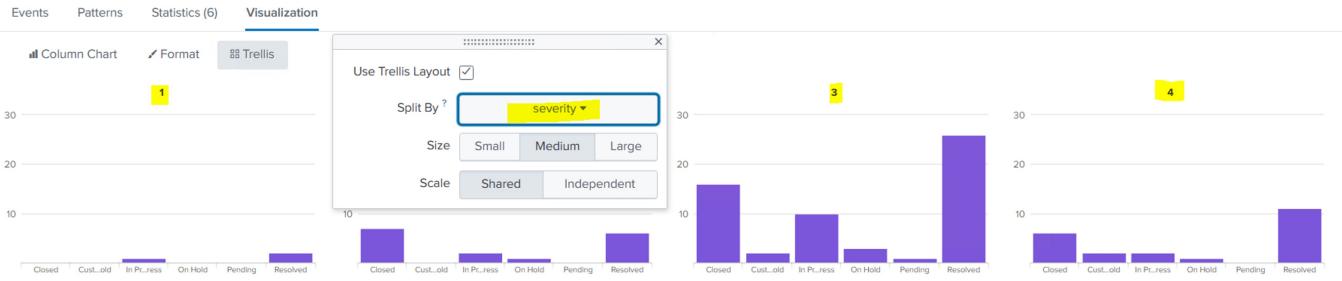
current_ticket_state	severity	count
Closed	2	7
Closed	3	16
Closed	4	6
Customer Hold	3	2
Customer Hold	4	2
In Progress	1	1
In Progress	2	2
In Progress	3	10
In Progress	4	2
On Hold	2	1

#Trellis in chart :

Trellis will convert one chart into multiple chart.

Scale can be "**Shared**" or it can be "**Independent**" for each graph.





#Single Value Visualization :

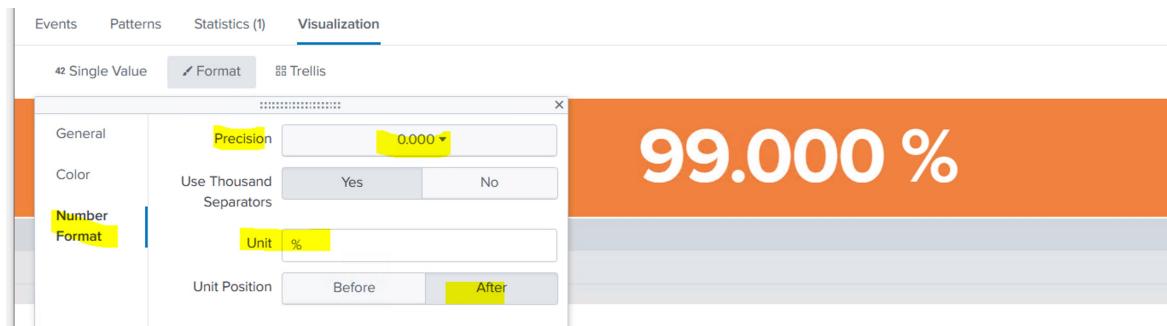
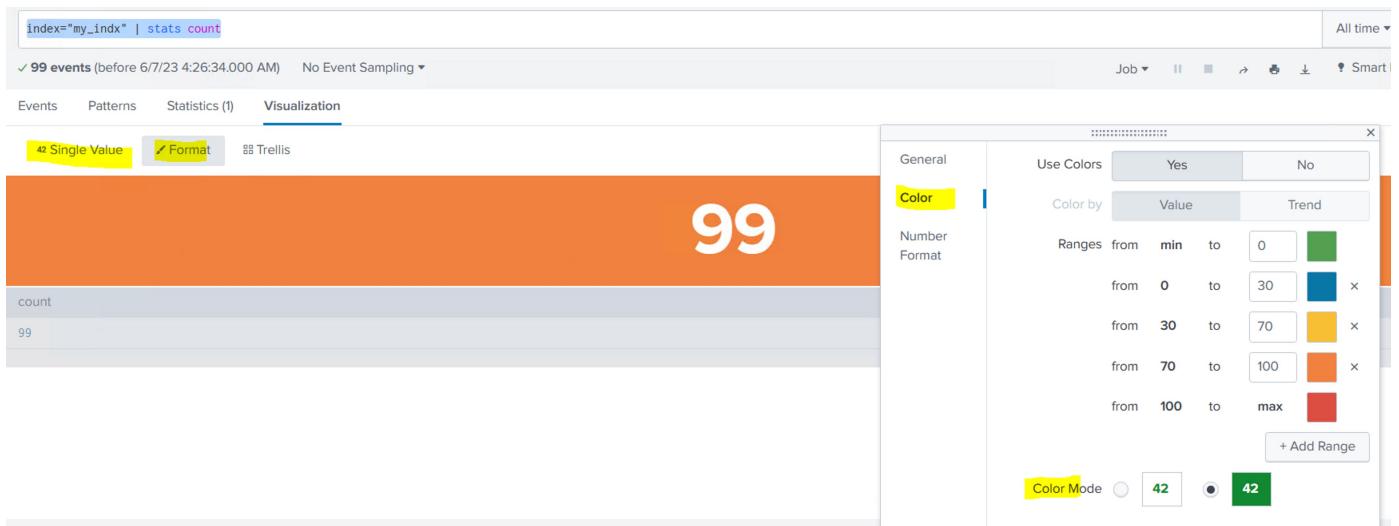
We have **4 type of graph** for Single Value visualization.

- (1) **Single Value**
- (2) **Radial Gauge**
- (3) **Marker Gauge**
- (4) **Filler Gauge**

Note : Make sure "Trellis" is disabled.

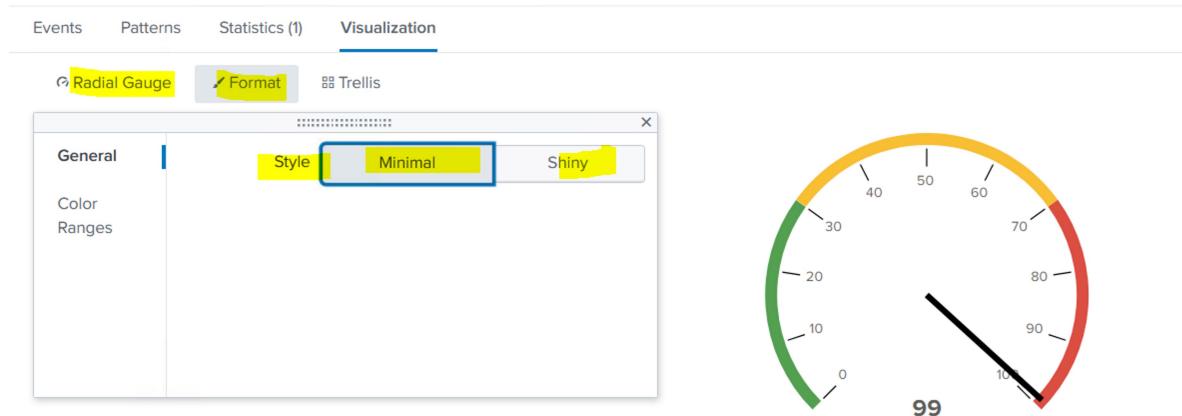
`index="my_idx" | stats count`

1st way : Single Value:



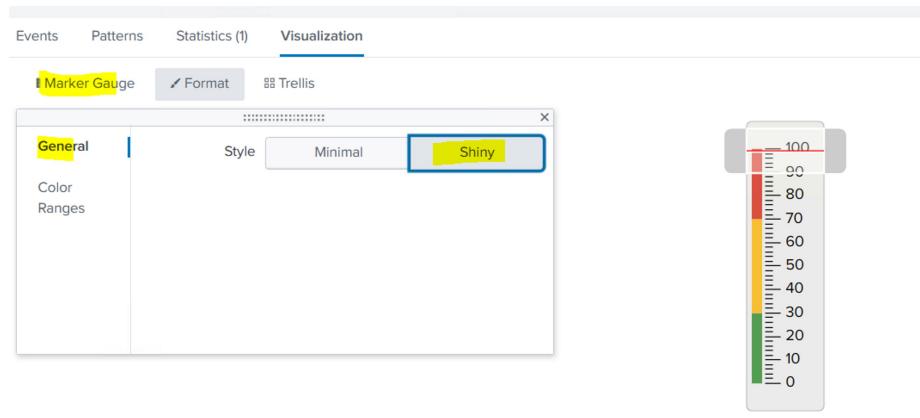
2nd way : Radial Gauge:

Two styles : Minimal and Shiny



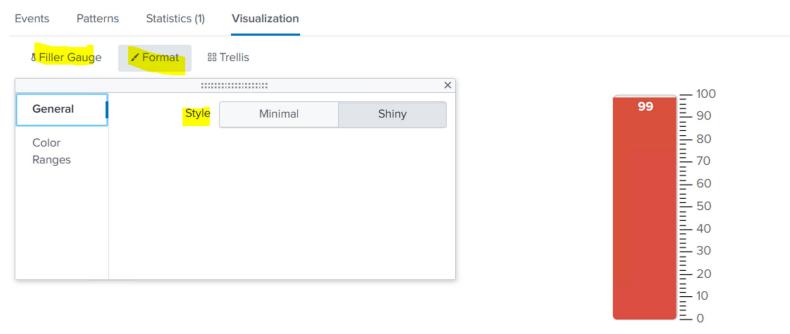
3rd way : Marker Gauge:

Two styles : Minimal and Shiny



4th way : Filler Gauge:

Two styles : Minimal and Shiny



#Timechart :

In timechart command, :

X-axis => Default will be _time

Y-axis => will be count/sum or other stats

A timechart is a statistical aggregation applied to a field to produce a chart, with **time used as the X-axis**.

You can specify a split-by field, where each distinct value of the split-by field becomes a series in the chart.

If you use an eval expression, the split-by clause is required.

With the **limit** and **agg** options, you can specify series filtering. These options are ignored if you specify an explicit where-clause.

If you set **limit=0**, no series filtering occurs.

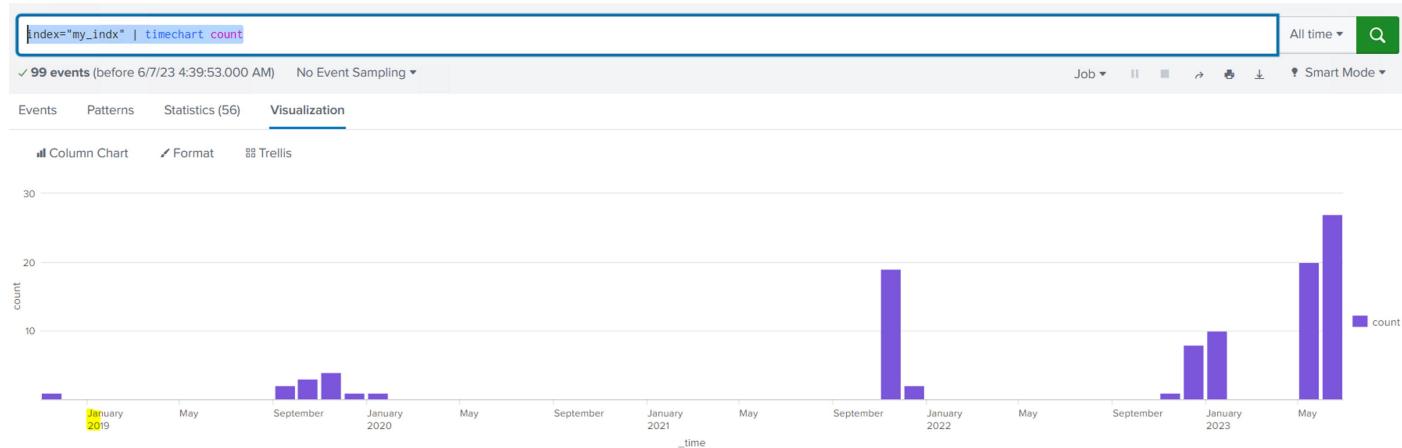
From <<https://docs.splunk.com/Documentation/Splunk/9.0.5/SearchReference/Timechart?ref=hk>>

Default time spans

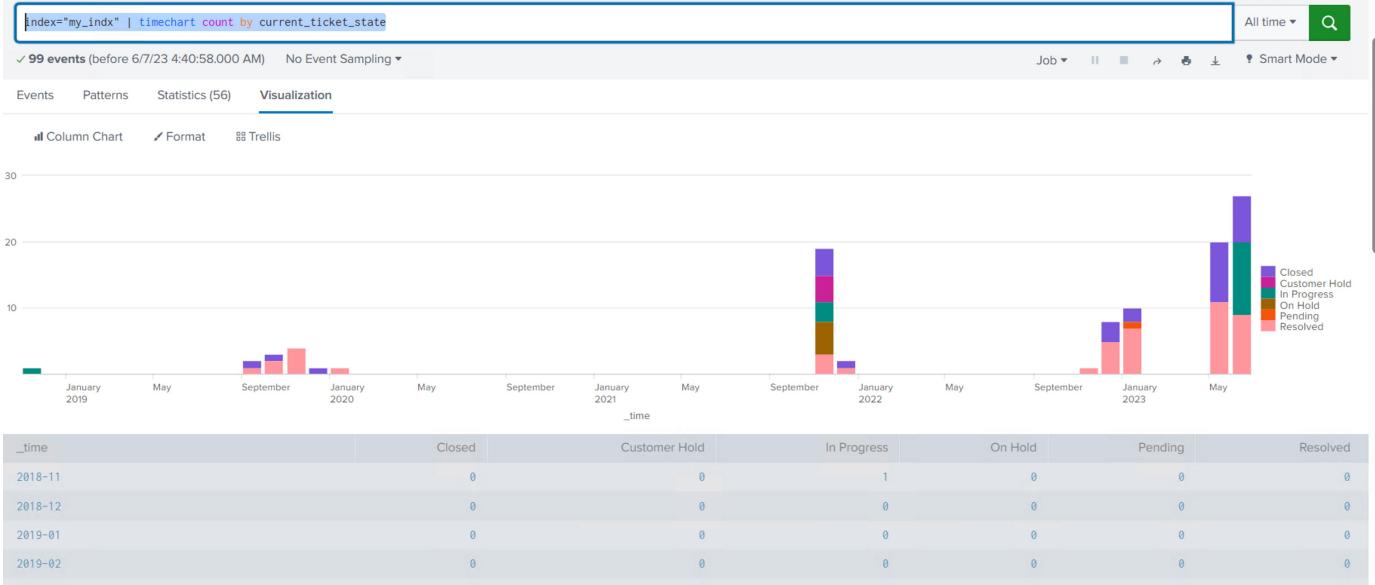
If you use the predefined time ranges in the time range picker, and do not specify the `span` argument, the following table shows the default span that is used.

Time range	Default span
Last 15 minutes	10 seconds
Last 60 minutes	1 minute
Last 4 hours	5 minutes
Last 24 hours	30 minutes
Last 7 days	1 day
Last 30 days	1 day
Previous year	1 month

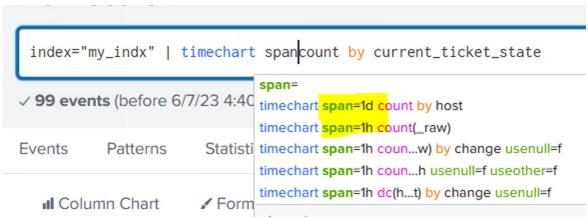
index="my_idx" | timechart count



index="my_idx" | timechart count by current_ticket_state

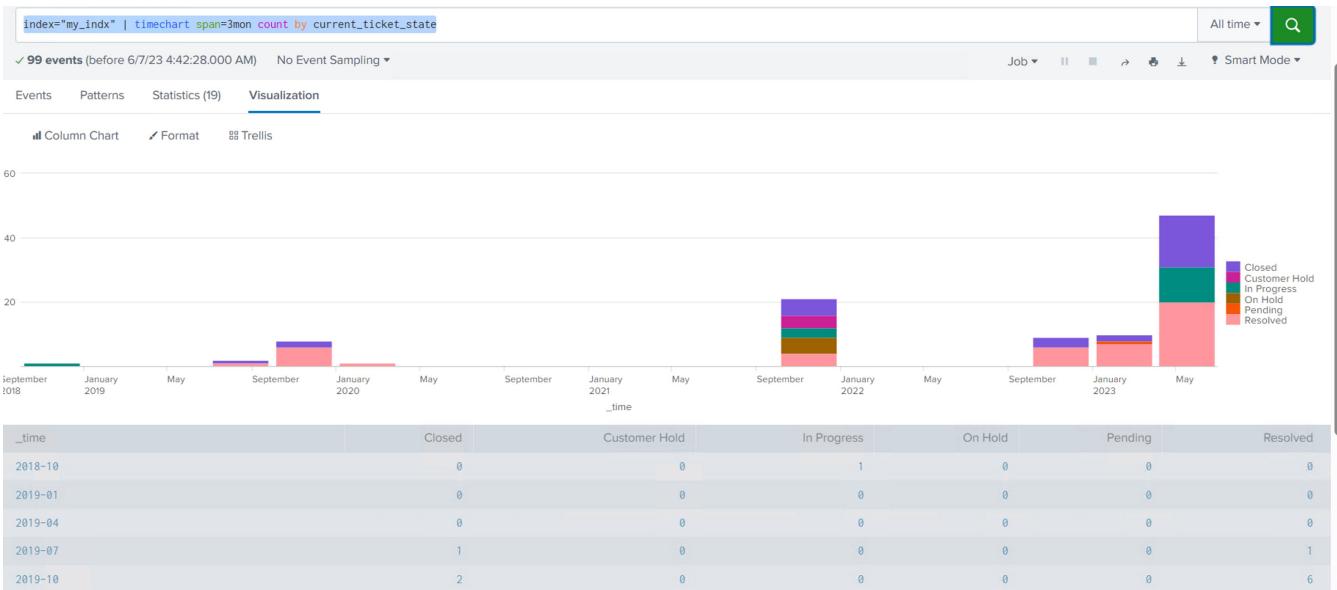


Span



Timescale	Valid syntax	Description
<sec>	s sec secs second seconds	Time scale in seconds.
<min>	m min mins minute minutes	Time scale in minutes.
<hr>	h hr hrs hour hours	Time scale in hours.
<day>	d day days	Time scale in days.
<week>	w week weeks	Time scale in weeks.
<month>	mon month months	Time scale in months.
<subseconds>	us ms cs ds	Time scale in microseconds (us), milliseconds (ms), centiseconds (cs), or decisoseconds (ds)

index="my_idx" | timechart span=3mon count by current_ticket_state



Question: How to use X-axis as other timestamp value (time_submitted) instead of _time(default)

Answer: By using eval to swap _time with time_submitted and then converting _time to EPOCH as after eval conversion splunk will not be able to recognize timestamp.

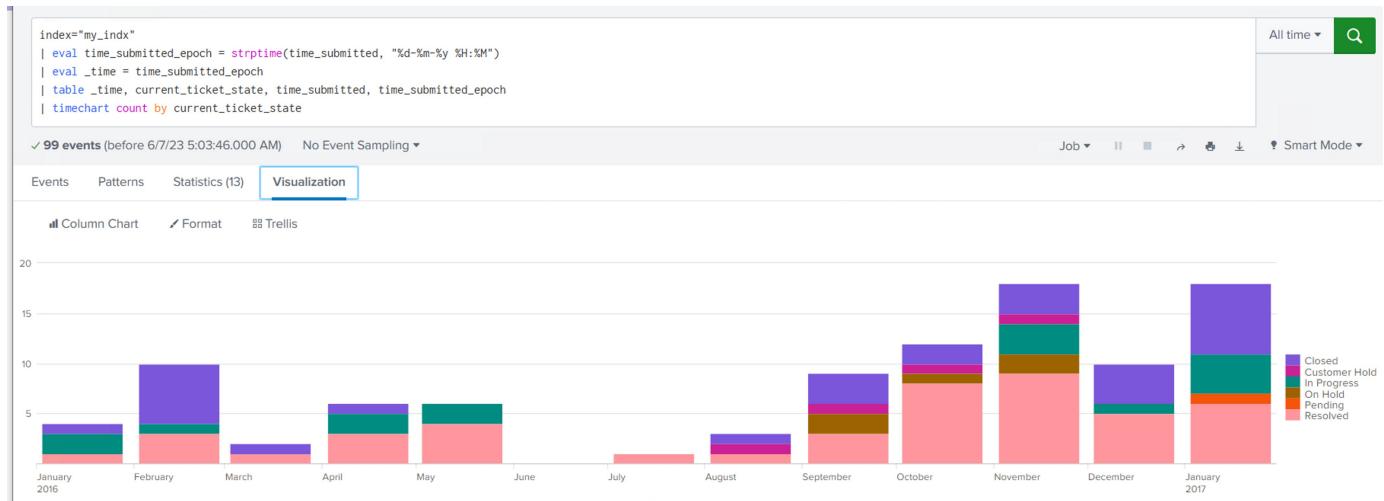
Step #1 : Convert time_submitted (31-08-16 9:44) to epoch by using `strftime()` and `eval()`

Step #2 : Swap _time with time_submitted_epoch y using eval()

Step #3 : Use `table` command to fetch fields

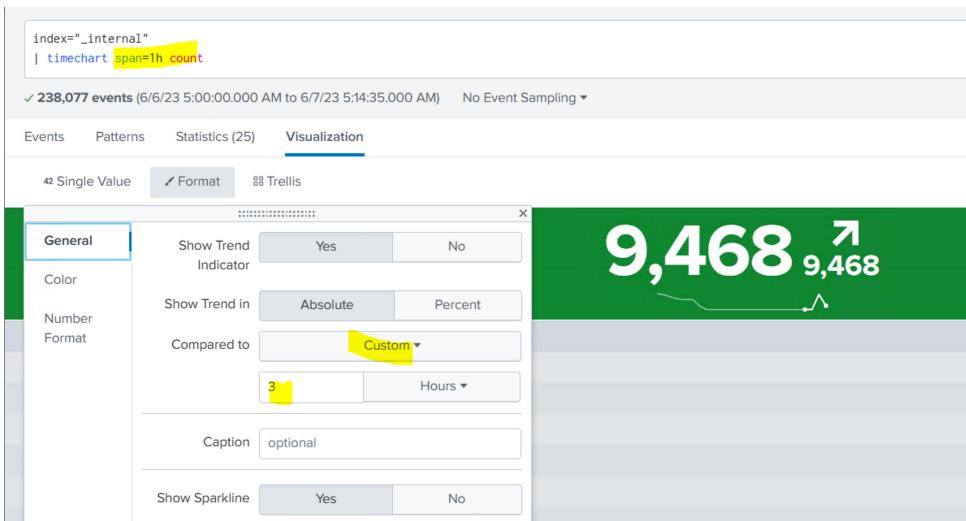
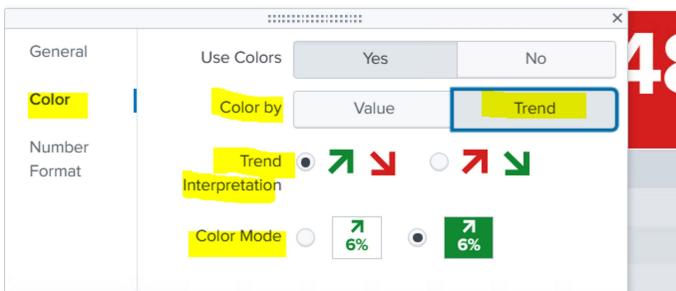
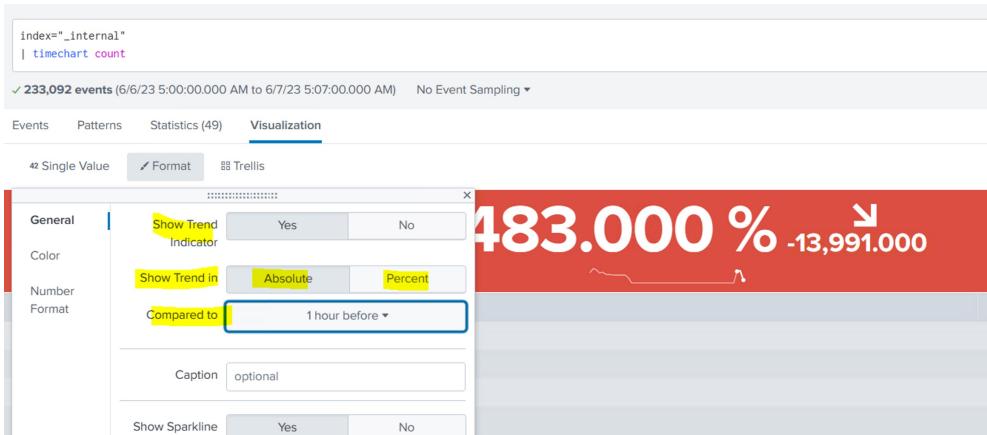
Step #4 : use `timechart` how to print visualization

```
index="my_idx"
| eval time_submitted_epoch = strftime(time_submitted, "%d-%m-%Y %H:%M")
| eval _time = time_submitted_epoch
| table _time, current_ticket_state, time_submitted, time_submitted_epoch
| timechart count by current_ticket_state
```



#Trendline:

Trendline will show trend compared to data of custom/specific time.



#MAP:

We need Latitude and Longitude values to pin any location on Map.

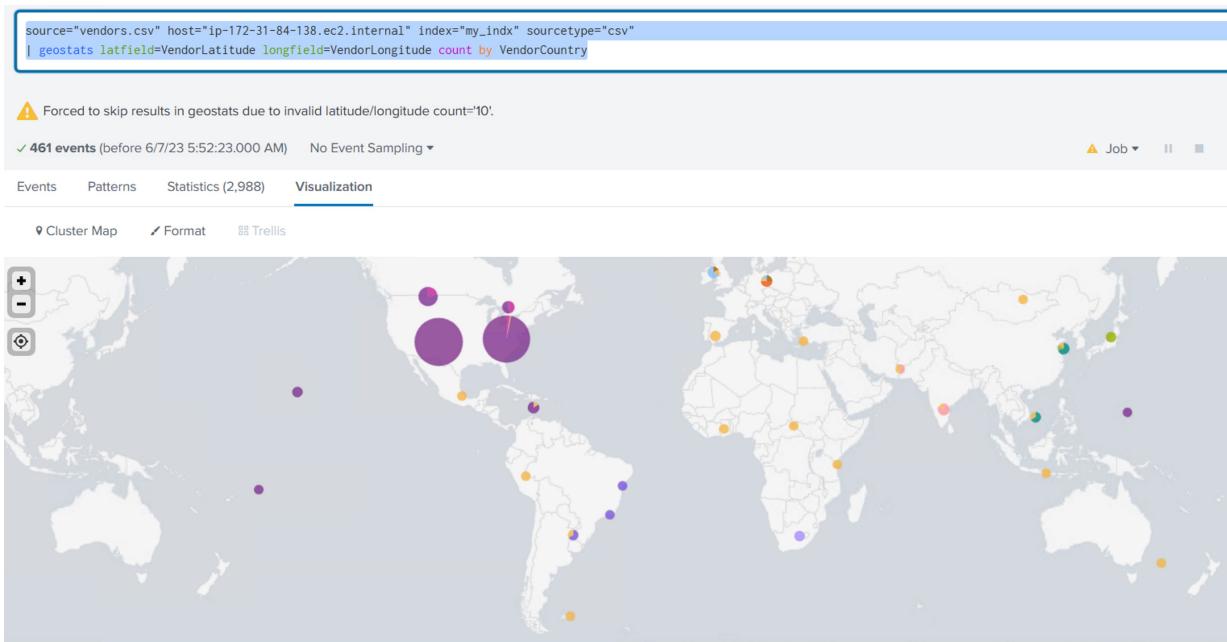
```
source="vendors.csv" host="ip-172-31-84-138.ec2.internal" index="my_idx" sourcetype="csv"
| geostats latfield=VendorLatitude longfield=VendorLongitude count by VendorCountry
```

Events Patterns Statistics (2,988) Visualization

20 Per Page ▾ Format Preview ▾

< Prev 1 2 3 4 5 6 7 8 ... Next >

geobin	Brazil	Canada	China (PRC)	France	Germany	India	Japan	OTHER	South Africa	United Kingdom	United States	latitude	longitude
bin_id_z1_0_y_1_x_2									1			-51.70778	-57.82654
bin_id_z1_0_y_2_x_2									1			-29.57282	-56.64565
bin_id_z1_0_y_2_x_3												-22.88841	-43.21506
bin_id_z1_0_y_2_x_4									3			-29.90986	25.61692



Default Geo Lookups (in .kmz file) comes with Splunk :

- (1) geo_countries
- (2) geo_us_states

Lookup definitions

Lookups > Lookup definitions

Showing 1-4 of 4 items

App Search & Reporting (s... Owner Any Visible in the App geo

Name	Type	Supported fields	Lookup file	Owner	App	Sharing
geo_attr_countries	file	country,region_wb,region_un,subregion,continent,iso2,iso3	geo_attr_countries.csv	No owner	search	Global Permissions
geo_attr_us_states	file	state_name,state_fips,state_code	geo_attr_us_states.csv	No owner	search	Global Permissions
geo_countries	geo	None	geo_countries.kmz	No owner	search	Global Permissions
geo_us_states	geo	None	geo_us_states.kmz	No owner	search	Global Permissions

| inputlookup geo_countries

| inputlookup geo_countries

Last 24 hours ▾

✓ 255 results (6/6/23 6:00:00.000 AM to 6/7/23 6:01:22.000 AM) No Event Sampling ▾

Job ▾ II ⌂ ⌂ ⌂ ⌂ Smart Metrics

Events Patterns Statistics (255) Visualization

20 Per Page ▾ Format Preview ▾

< Prev 1 2 3 4 5 6 7 8 ...

/	count	featureCollection	/	featureId	/	geom
/	0	geo_countries	Afghanistan			{"type": "MultiPolygon", "coordinates": [[[[-71.0498046875, 38.4086470336914], [-71.65302276611328, 36.68701171875], [74.89230346679688, 37.23111343383789], [71.22307586669922, 36.12539291381836], [69.04010772705078, 31.673107147216797], [65.03636932373047, 29.5401611328125], [60.84437942504883, 29.858179092407227], [61.269676208496094, 35.61849975585937], [-71.0498046875, 38.4086470336914]]]]}}
/	0	geo_countries	Akrotiri Sovereign Base Area			{"type": "MultiPolygon", "coordinates": [[[[-32.8408088684082, 34.699466705322266], [32.8408088684082, 34.699466705322266]]]]}}
/	0	geo_countries	Albania			{"type": "MultiPolygon", "coordinates": [[[[-19.747766494750977, 42.57889938354492], [20.18412208557129, 39.63701248168945], [19.747766494750977, 42.57889938354492]]]]}}
/	0	geo_countries	Algeria			{"type": "MultiPolygon", "coordinates": [[[[-8.602510452270508, 36.939510345458984], [7.479832172393799, 33.89390182495117], [9.826148986816406, 29.12853240966797], [9.37780475616455, 26.168947219848633], [11.968860626220703, 23.5173511505126951], [5.794301986694336, 19.449796676635742], [3.3083558082580566, 18.981685638427734], [-8.68238544641113, 27.285415649414062], [-8.250473976135254, 28.994768142700195], [-1.2103348970413208, 32.08967208862305], [-2.222564220428467, 35.08930206298828], [8.602510452270508, 36.939510345458984]]]]}}

| inputlookup geo_us_states

| inputlookup geo_us_states

✓ 51 results (6/6/23 6:00:00.000 AM to 6/7/23 6:02:08.000 AM) No Event Sampling ▾

Events Patterns Statistics (51) Visualization

20 Per Page ▾ Format Preview ▾

/	count	featureCollection	/	featureId	/	geom
/	0	geo_us_states	Alabama			{"type": "MultiPolygon", "coordinates": [[[[-88.31002807617188, 30.2332324981689], [-88.20295715332031, 35.008026123046875], [-85.60516357421875, 34.9846763610839], [-88.47322845458984, 31.893856048583984]]]]}}
/	0	geo_us_states	Alaska			{"type": "MultiPolygon", "coordinates": [[[[-179.4824676513672, 51.98283386230469], [-178.6255340576172, 51.637302398681641711], [-177.44696044921875, 51.97822189331], [-176.20295715332031, 51.893856048583984]]]]}}

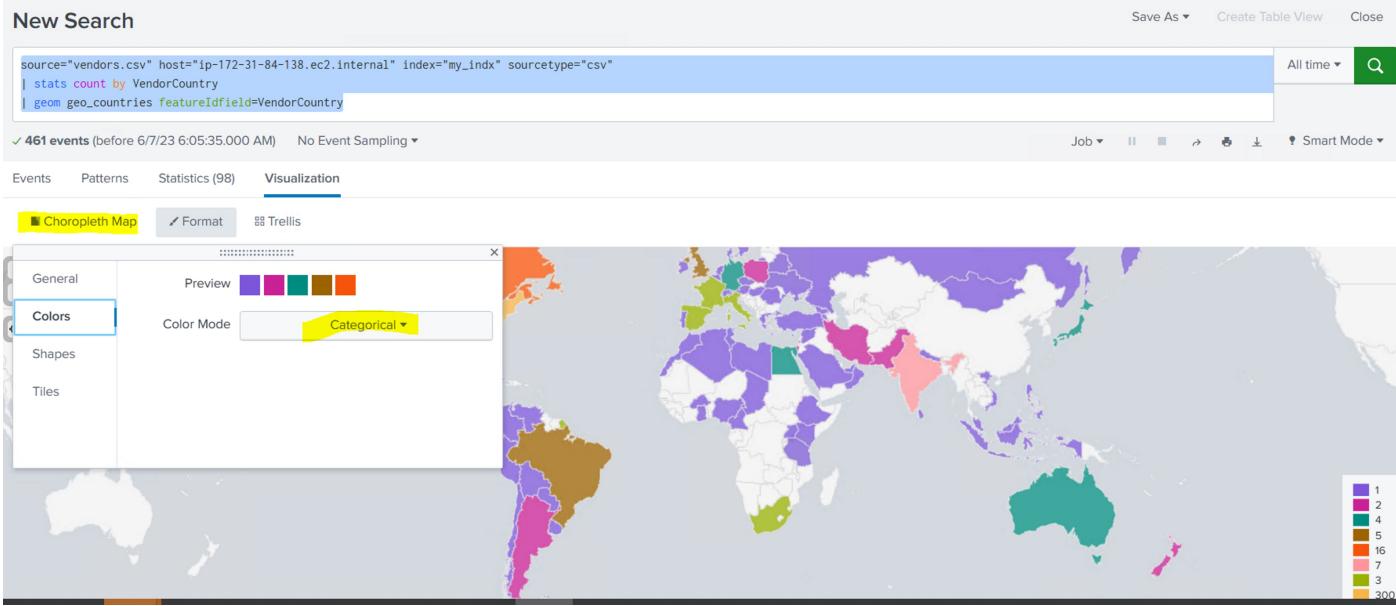
Question : Highlight countries based on their count in map. (**Geospatial data**).

Answer : using **geom** command and **Choropleth** map.

```
source="vendors.csv" host="ip-172-31-84-138.ec2.internal" index="my_idx" sourcetype="csv"
```

```
| stats count by VendorCountry
```

```
| geom geo_countries featureIdfield=VendorCountry
```



Question: How to convert kml file to kmz ?

Answer: Kml file is same as XML. Zip the kml file and give extension to .kmz

#Application:

Precheck before App installation :

- (1) Space
- (2) Sufficient Resources
- (3) Developer/**Author** (Splunk or 3rd party)
- (4) Number of Downloads
- (5) Out of date/Latest/Expiry
- (6) Version of Splunk

The screenshot shows the Splunk App Store interface with the following details:

- Header:** splunk>enterprise Apps ▾ user2 ▾
- Section:** Browse More Apps
- Filters:** Find apps by keyword, technology... (search bar), Newest, Popular, 2022 Apps.
- Category Filter:** CATEGORY (checkboxes: IT Operations, Security, Fraud & Compliance, Business Analytics, Utilities, IoT & Industrial Data, DevOps, Directory Service, Email, Endpoint, Firewall, Generic, Identity Management, Information, Investigative, Network Access Control).
- App Cards:**
 - TrackMe:** Splunk provides visibility and operational excellence to monitor at scale your Splunk data sources availability and quality, and many more. Category: IT Operations, Security, Fraud & Compliance | Author: TrackMe Limited | Downloads: 13264 | Released: 8 hours ago | Last Updated: 8 hours ago | View on Splunkbase
 - Oracle:** A security info... to manage the : includes native leverage to Imp
 - Qualys Technology Add-on (TA) for Splunk:** With a rich set of features and a powerful workflow, TrackMe empowers you day after day to get the most from your Splunk investments and deliver the five stars quality of service your users deserve... More
 - JFrog:** This app helps i

Browse More Apps

sankey

Best Match Newest Popular

6 Apps

CATEGORY

- IT Operations
- Security, Fraud & Compliance
- Business Analytics
- Utilities
- IoT & Industrial Data
- DevOps
- Directory Service
- Email
- Endpoint
- Firewall
- Generic

Splunk Sankey Diagram - Custom Visualization

Install

Custom Visualizations give you new interactive ways to visualize your data during search and investigation, and to better communicate results in dashboards and reports. After installing this app you'll find a Sankey diagram as an additional item in the visualization picker in Search and Dashboard.

Sankey diagrams show metric flows and category rel... [More](#)

Category: IT Operations, Business Analytics, Utilities | Author: Splunk Inc. | Downloads: 113078 | Released: 2 years ago | Last Updated: 7 months ago | [View on Splunkbase](#)

Overview

Details

Custom Visualizations give you new interactive ways to visualize your data during search and investigation, and to better communicate results in dashboards and reports. After installing this app you'll find a Sankey diagram as an additional item in the visualization picker in Search and Dashboard.

Sankey diagrams show metric flows and category relationships. You can use a Sankey diagram to visualize relationship density and trends.

A Sankey diagram shows category nodes on vertical axes. Fluid lines show links between source and target categories. Link width indicates relationship strength between a source and target.

(c) 2016-2020 Splunk Inc. All Rights Reserved.

113,078

Downloads

LOGIN TO DOWNLOAD

VERSION

16.0

BUILT BY

Splunk Inc.

CONTRIBUTORS

John Paul Francisco

Benny Shi

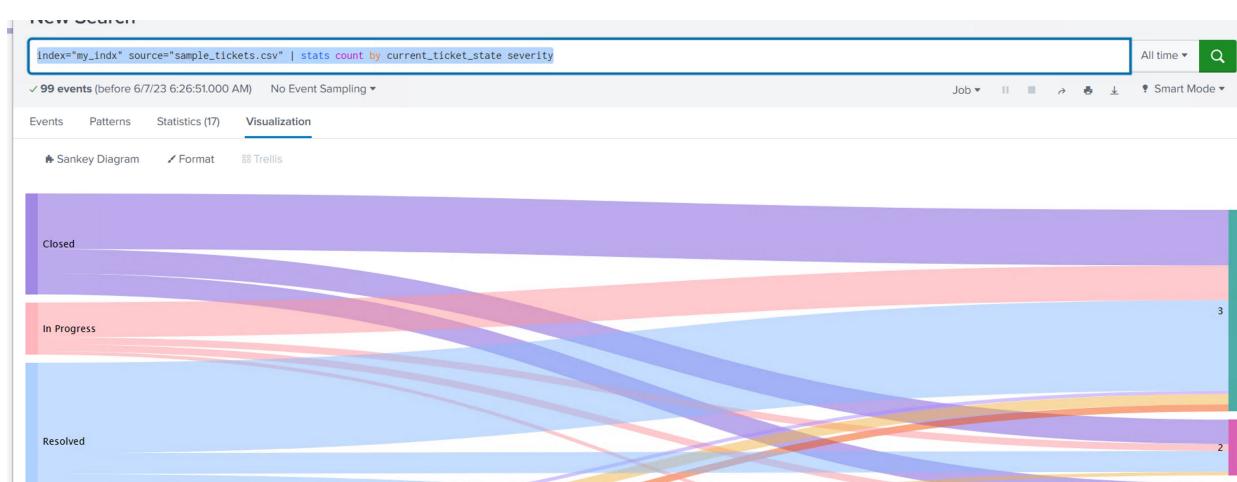
Giulia Mattia

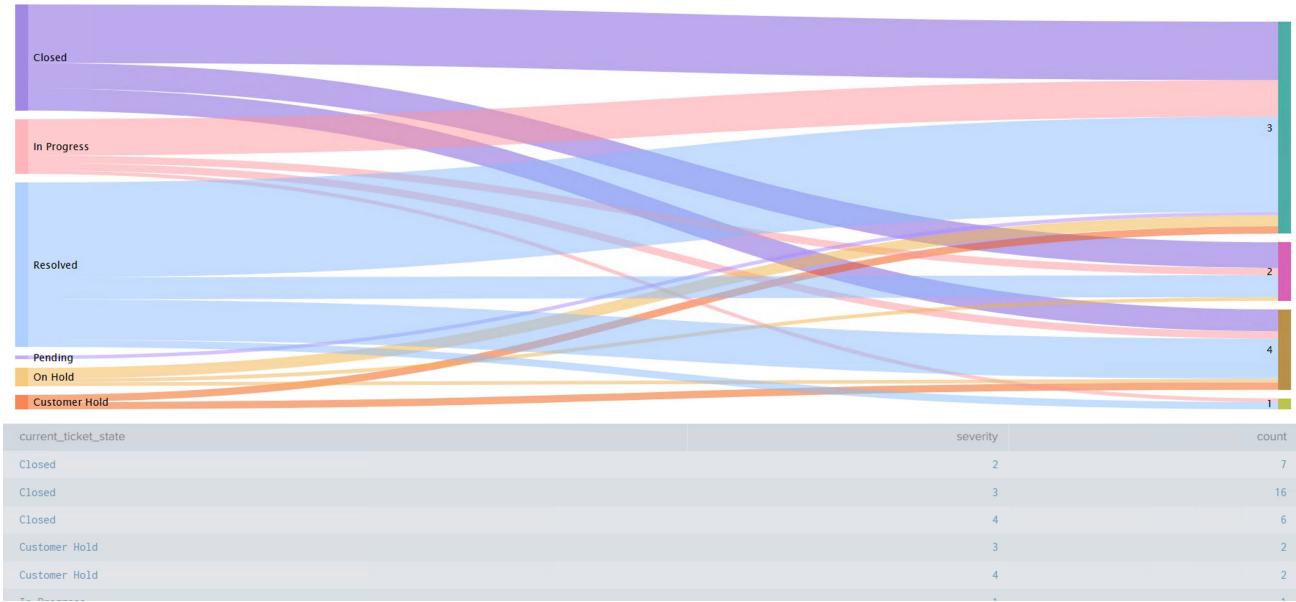
Lizzi Li

#SanKey Diagram:

SanKey diagram needs 2 values.

`index="my_idx" source="sample_tickets.csv" | stats count by current_ticket_state severity`





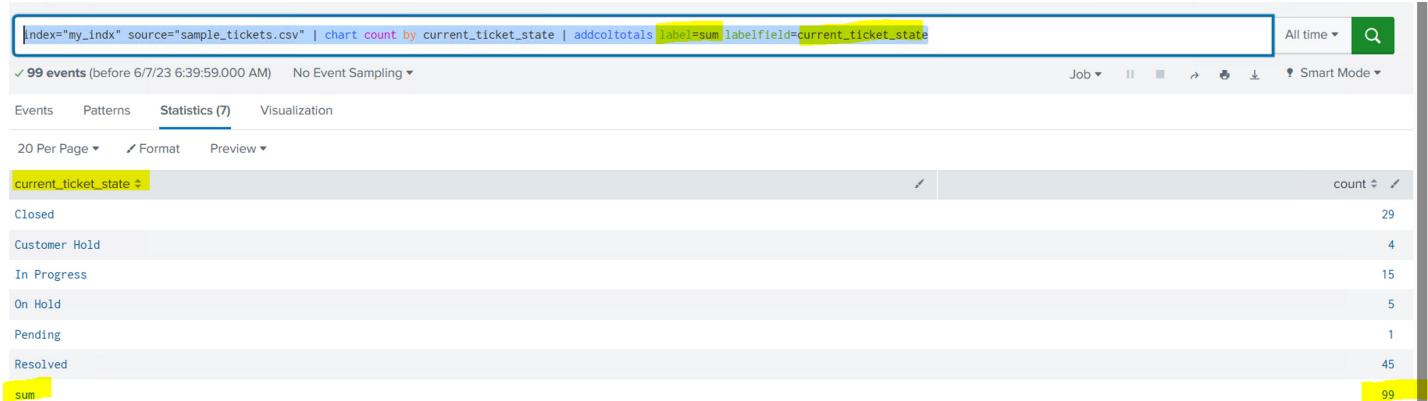
#addcoltotals

It does **Column-wise addition**.

```
index="my_indx" source="sample_tickets.csv" | chart count by current_ticket_state | addcoltotals
label=sum labelfield=current_ticket_state
```

Label => what will be the name of total

Labelfield => after which field/column the label will be visible.



```
index="my_indx" source="sample_tickets.csv" | chart count by current_ticket_state severity |
addcoltotals label=sum labelfield=current_ticket_state
```

index="my_idx" source="sample_tickets.csv" | chart count by current_ticket_state severity | addcoltotals label=sum labelfield=current_ticket_state

✓ 99 events (before 6/7/23 6:44:28.000 AM) No Event Sampling ▾ Job ▾ II III ▾ Smart Mode ▾

Events Patterns Statistics (7) Visualization

20 Per Page ▾ Format Preview ▾

current_ticket_state	1	2	3	4
Closed	0	7	16	6
Customer Hold	0	0	2	2
In Progress	1	2	10	2
On Hold	0	1	3	1
Pending	0	0	1	0
Resolved	2	6	26	11
sum	3	16	58	22

index="my_idx" source="sample_tickets.csv" | chart count by current_ticket_state severity | addcoltotals 1,3 label=sum labelfield=current_ticket_state

index="my_idx" source="sample_tickets.csv" | chart count by current_ticket_state severity | addcoltotals 1,3 label=sum labelfield=current_ticket_state

✓ 99 events (before 6/7/23 6:45:15.000 AM) No Event Sampling ▾ Job ▾ II III ▾ Smart Mode ▾

Events Patterns Statistics (7) Visualization

20 Per Page ▾ Format Preview ▾

current_ticket_state	1	2	3	4
Closed	0	7	16	6
Customer Hold	0	0	2	2
In Progress	1	2	10	2
On Hold	0	1	3	1
Pending	0	0	1	0
Resolved	2	6	26	11
sum	3	16	58	22

#addtotals

It does **Row-wise addition**.

index="my_idx" source="sample_tickets.csv" | chart count by current_ticket_state ,severity | addtotals

index="my_idx" source="sample_tickets.csv" | chart count by current_ticket_state ,severity | addtotals

✓ 99 events (before 6/7/23 6:47:33.000 AM) No Event Sampling ▾ Job ▾ II III ▾ Smart Mode ▾

Events Patterns Statistics (6) Visualization

20 Per Page ▾ Format Preview ▾

current_ticket_state	1	2	3	4	Total
Closed	0	7	16	6	29
Customer Hold	0	0	2	2	4
In Progress	1	2	10	2	15
On Hold	0	1	3	1	5
Pending	0	0	1	0	1
Resolved	2	6	26	11	45

#Rex

Useful websites :

<https://regex101.com/>

<https://www.debuggex.com/cheatsheet/regex/pcre>

2 types of searches:

- (1) **Index-time search** => at time of processing and before ingestion
- (2) **Search-time search** => after ingestion

```
source="data.txt" host="ip-172-31-84-138.ec2.internal" index="my_idx" sourcetype="my_txt"  
| rex field=_raw "From:\s+<(?:From_id>.*>|\s+To"
```

The screenshot shows the Splunk search interface with the following details:

- Search Bar:** source="data.txt" host="ip-172-31-84-138.ec2.internal" index="my_idx" sourcetype="my_txt" | rex field=_raw "From:\s+<(?:From_id>.*>|\s+To"
- Results Summary:** ✓ 4 events (before 6/7/23 7:02:19.000 AM) No Event Sampling ▾
- Event List:** 47 From: <MariaDubois@example.com>
- Field Selection:** From_id
- Reports:** Top values, Top values by time, Events with this field
- Table:** Values

	Count	%
Exit_Desk@example.net	1	25%
Manish_Das@example.com	1	25%
MariaDubois@example.com	1	25%
WeiZhang@example.com	1	25%

- Actions:** (dropdown menu)
- Time Range:** Event, From_id ▾ MariaDubois@example.com

#makeresults

It creates sample results.

_time column will be added by default.

/makeresults

```
| makeresults  
| eval text="user_id=bob;search;my_saved_search"
```

The screenshot shows the Splunk search interface with the following details:

- Search Bar:** | makeresults | eval text="user_id=bob;search;my_saved_search"
- Results Summary:** ✓ 1 result (before 6/7/23 7:06:57.000 AM) No Event Sampling ▾
- Event List:** 2023-06-07 07:06:57 user_id=bob;search;my_saved_search
- Field Selection:** _time
- Text Input:** text
- Text Value:** user_id=bob;search;my_saved_search

/makeresults

```
| eval text="user_id=bob;search;my_saved_search"
| rex field=text "user_id=(?P<user_name>\w+);(?P<app_name>\w+);(?P<serach_query>\w+)"
```

The screenshot shows a Splunk search interface with the following details:

- Search Bar:** Contains the search command: `| makeresults | eval text="user_id=bob;search;my_saved_search" | rex field=text "user_id=(?P<user_name>\w+);(?P<app_name>\w+);(?P<serach_query>\w+)"`
- Results Panel:** Shows 1 result (before 6/7/23 7:29:30.000 AM) with "No Event Sampling".
- Statistics Tab:** Selected, showing the following data:

_time	app_name	serach_query	text	user_name
2023-06-07 07:29:30	search	my_saved_search	user_id=bob;search;my_saved_search	bob

```
|makeresults
| eval credit_card_number = "1234-5678-9109-1234"
| rex field=credit_card_number max_match=0 "(?P<digit>\d{4})"
```

The screenshot shows a Splunk search interface with the following details:

- Search Bar:** Contains the search command: `| makeresults | eval credit_card_number = "1234-5678-9109-1234" | rex field=credit_card_number max_match=0 "(?P<digit>\d{4})"`
- Results Panel:** Shows 1 result (before 6/7/23 7:23:30.000 AM) with "No Event Sampling".
- Statistics Tab:** Selected, showing the following data:

_time	credit_card_number	digit
2023-06-07 07:23:30	1234-5678-9109-1234	1234 5678 9109 1234

```
|makeresults
| eval credit_card_number = "1234-5678-9109-1234"
| rex field=credit_card_number max_match=0 offset_field="new_field" "(?P<digit>\d{4})"
```

The screenshot shows a Splunk search interface with the following details:

- Search Bar:** Contains the search command: `| makeresults | eval credit_card_number = "1234-5678-9109-1234" | rex field=credit_card_number max_match=0 offset_field="new_field" "(?P<digit>\d{4})"`
- Results Panel:** Shows 1 result (before 6/7/23 7:24:29.000 AM) with "No Event Sampling".
- Statistics Tab:** Selected, showing the following data:

_time	credit_card_number	digit	new_field
2023-06-07 07:24:29	1234-5678-9109-1234	1234 5678 9109 1234	digit=0-3&digit=5-8&digit=10-13&digit=15-18

