

Revealing the Magic

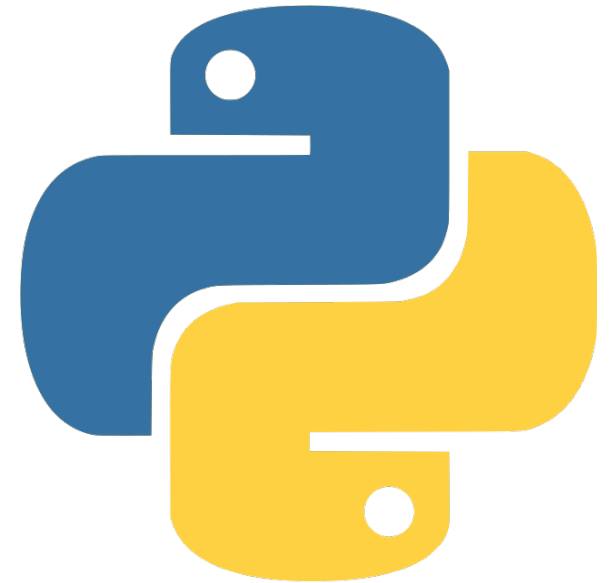
The Lifecycle of a Splunk Search

Kellen Green | Senior Software Engineer

September 27th, 2017 | Washington, DC

About Myself

web developer



splunk>

App: Search & Reporting ▾

Administrator ▾

Search

Datasets

Reports

Alerts

Dashboards

New Search

index="conf2017"

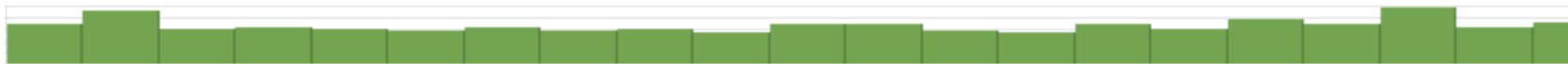
✓ 999 events (before 7/13/17 2:04:47.000 PM) [No Event Sampling ▾](#)

Events (999)

Patterns

Statistics

Visualization

[Format Timeline ▾](#)[— Zoom Out](#)[+ Zoom to Selection](#)[× Deselect](#)[List ▾](#)[Format](#)[20 Per Page ▾](#)[◀ Hide Fields](#)[≡ All Fields](#)

Selected Fields

bar 100

foo 100

a host 1

a source 1

a sourcetype 1

Interesting Fields

i

Time

Event

>

1/31/17
11:24:50.000 PM

```
{ [-]  
  bar: 24  
  foo: 90  
  time: 2017-01-31T23:24:50 +0000  
}
```

[Show as raw text](#)

bar = 24 | foo = 90 | host = workhorse | source = /home/kgreen/conf2017/data.json | sourcetype

>

1/31/17
10:43:30.000 PM

```
{ [-]  
  bar: 2  
  foo: 95  
  time: 2017-01-31T23:43:30 +0000
```


“OUR DEVELOPERS HAVE PRODUCTIVITY SUPERPOWERS.”

Kris Wehner, VP of Engineering, Yelp Reservations



IT'S NOT MAGIC. IT'S SPLUNK™

Magic?

Let's debunk that!

1. Develop a deeper understanding of the core components that make up a Splunk search.
2. Increase performance of your searches through more efficient queries.
3. Obtain stronger grasp of which deployment types are better suited for specific workloads.

Data Set

26 event CSV file

```
time,foo,bar
```

```
2017-09-01T16:00:00 +0000,a,z
2017-09-02T02:00:00 +0000,b,y
2017-09-03T12:00:00 +0000,c,x
2017-09-04T18:00:00 +0000,d,w
2017-09-05T03:00:00 +0000,e,v
2017-09-06T08:00:00 +0000,f,u
2017-09-07T22:00:00 +0000,g,t
```

...

- ▶ One event per day from Sept. 1 - 26
 - Random hour of the day
- ▶ Indexed field `foo`
 - Descending A - Z
- ▶ Unindexed field `bar`
 - Ascending Z - A

Search #1

Indexer Workflow

`index="conf2017" foo="0"`

Sept. 1st to the 27th

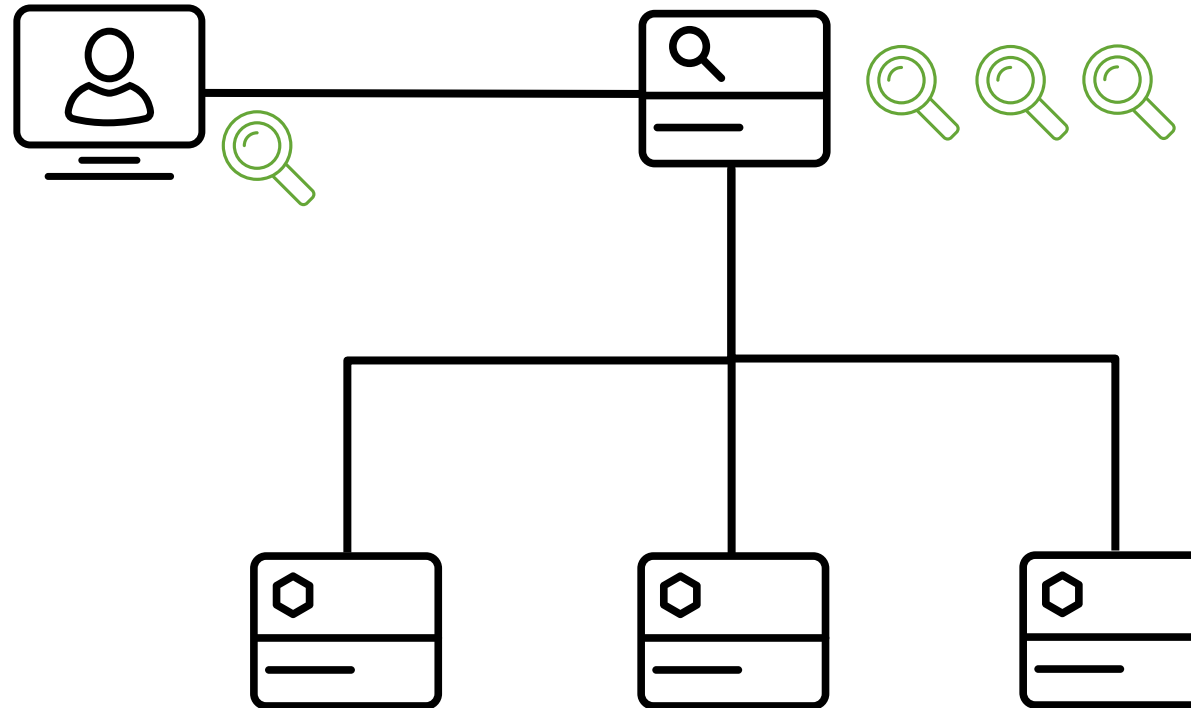
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D1SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03"
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D3SL7FF6ADFF0 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
317 27.160.0.0 - - [07/Jun 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D3SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=5D18SLBFF3ADFF9 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D1SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03"
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D3SL7FF6ADFF0 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
317 27.160.0.0 - - [07/Jun 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D3SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=5D18SLBFF3ADFF9 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D1SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03"
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D3SL7FF6ADFF0 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
317 27.160.0.0 - - [07/Jun 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D3SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=5D18SLBFF3ADFF9 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"

index="conf2017" foo="0"

index="conf2017" foo="0"

Yep, but I promise it's interesting!

```
index="conf2017" foo="0"
```



Indexes Directory

index="conf2017" foo="0"

```
$ cd $SPK_IDX/var/lib/splunk/
```

```
$ ls -l
```

```
audit
```

```
authDb
```

```
→ conf2015
```

```
→ conf2016
```

```
→ conf2017
```

```
→ defaultdb
```

```
historydb
```

```
Kvstore
```

- ▶ Root directory for indexes.
- ▶ Check if queried index directory exists.
- ▶ Specify an index to improve improve search performance.

splunk> **.conf2017**

Index Directory

index="conf2017" foo="0"

```
$ cd conf2017/  
$ ls -l  
→ colddb  
  datamodel_summary  
→ db  
  thaweddb
```

- ▶ colddb houses older searchable data.
 - Implement cheaper storage solutions.
- ▶ db directory for fresh data in high demand.
- ▶ Configurable in `indexes.conf`.

Buckets Directory

index="conf2017" foo="0"

- ▶ Hot buckets are still being written to.
- ▶ Warm buckets are event immutable.
 - Named by time range.
- ▶ Specify strict time range to boost Performance.

```
$ cd db/  
$ ls -l  
    .bucketManifest  
    CreationTime  
→ db_1468867200_1471545599_0  
→ db_1485388800_1493228720_1  
→ hot_v1_2  
    GlobalMetaData
```

Bloom Filter

index="conf2017" foo="0"

- ▶ Scanning buckets can be expensive.
- ▶ Bloom filter provides us with a fast way to determine if a term is **NOT** in a bucket.

```
$ cd db_1485388800_1493228720_1/
$ ls -l
    1485388800-1483228800.tsidx
→ bloomfilter
    bucket_info.csv
    Hosts.data
    optimize.result
    rawdata
    Sources.data
    SourceTypes.data
    Strings.data
```

index="conf2017" foo="0"



index="conf2017" foo="0"

- ▶ For search terms that are common, the bloom filter will do nothing to improve search performance.
- ▶ Huge performance boost for rare and nonexistent events.
 - Speed up on the order of 100x (1-2s to 10ms).

Search #2

Indexing

```
index="conf2017"  foo="a"
```

VS

index="conf2017" **bar="z"**

foo="a" vs bar="z"

2017-09-01T16:00:00 +0000, a, z

TSIDX File

foo="a" vs bar="z"

- ▶ Index file used to reduce the number of matching events.
- ▶ Lexigraphically sorted array of all terms within the bucket.
- ▶ The flag for `|delete` is also set here.

```
$ pwd
    db_1485388800_1493228720_1
$ ls -l
➔ 1485388800-1483228800.tsidx
  bloomfilter
  bucket_info.csv
  Hosts.data
  optimize.result
  rawdata
  Sources.data
  SourceTypes.data
  Strings.data
```

foo="a" vs bar="z"

- ▶ The Lispy query is used to when searching through TSIDX files.
- ▶ Created by the Search Head at search time.
- ▶ `foo="a"` becomes `[foo::a]` in Lispy.
- ▶ This will match all events where `foo` equals exactly `a`.

Lispy for Unindexed Fields

foo="a" vs bar="z"

- ▶ bar="z" becomes [z] in Lispy.
- ▶ This will match all events that contain z anywhere within the event.
- ▶ This might seem counter intuitive, but there is a good reason for this behavior.

```
130.60.4 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03" "Mozilla/4.0" "Opera/9.20"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CW-01" "Mozilla/4.0" "Comodo"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=5D5SL8FF3ADFF9 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-16&product_id=RP-LI-02" "Mozilla/4.0" "Opera/9.20"
130.60.4 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03" "Mozilla/4.0" "Opera/9.20"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CW-01" "Mozilla/4.0" "Comodo"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=5D5SL8FF3ADFF9 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-16&product_id=RP-LI-02" "Mozilla/4.0" "Opera/9.20"
```


foo="a" vs bar="z"

$$[z]$$


Search job inspector

This search has completed and has returned **1** results by scanning **2** events in **0.26** seconds
(SID: 1502366143.92) [search.log](#)

✓ Execution costs

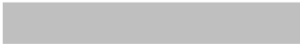
Duration (seconds)	Component	Invocations
	0.00 command.fields	1
	0.00 command.search	1
■	0.01 command.search.expand_search	1
	0.00 command.search.index	2

Search job inspector

This search has completed and has returned **1** results by scanning **1** events in **0.056** seconds

(SID: 1502365983.83) [search.log](#)

✓ Execution costs

Duration (seconds)	Component	Invocations
	0.00 command.fields	1
	0.00 command.search	1
	0.01 command.search.expand_search	1
	0.00 command.search.index	2

Raw Data Extraction

foo="a" vs bar="z"

```
$ cd rawdata/  
$ ls -l  
→ journal.gz  
→ slicemin.dat  
→ slicesv2.dat
```

- ▶ journal.gz
compressed slices of raw events.
- ▶ slices.dat map
from TSIDX to slice.
- ▶ Remaining unwanted events will be filtered during extraction.

Cons of Unindexed Fields

foo="a" vs bar="z"

- ▶ Increased number of potential matching events coming out of TSIDX.
- ▶ This list is kept in memory, leading to increased memory usage.
- ▶ More events, leads to more CPU needed for Journal decompression.

130.60.4 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/category.screen?category_id=GIFTS"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=AV-CB-01&SESSIONID=5D5SL9FF1ADFF3"
130.60.4 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/category.screen?category_id=GIFTS"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=AV-CB-01&SESSIONID=5D5SL9FF1ADFF3"
130.60.4 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/category.screen?category_id=GIFTS"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=AV-CB-01&SESSIONID=5D5SL9FF1ADFF3"

foo="a" vs bar="z"

Walklex Command

foo="a" vs bar="z"

```
$ walklex my.tsidx "foo::a"  
0035130149.tsidx "foo::a"  
my needle: foo::a  
209 1 foo::a
```

```
$ walklex my.tsidx "z"  
0035130149.tsidx "z"  
my needle: z  
287 2 z
```

- ▶ Shows us the number of matching TSIDX events for a given Lispy query.
- ▶ Useful for hunting down field indexing candidates.

Search #3

Wildcards

```
index="conf2017"  foo="*a"
```

VS

```
index="conf2017"  foo="a*"
```


Again Same Result

foo="*a" vs foo="a*"

2017-09-01T16:00:00 +0000,a,z

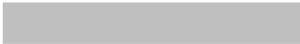
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5015L4FF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)" 468 125.17 14
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5035L7FF6ADFF0 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=AV-CB-01&SESSIONID=50185L8FF3ADFF9 HTTP 1.1" 200 3855 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-16&product_id=RP-LI-02" 468 125.17 14
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5015L4FF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)" 468 125.17 14
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5035L7FF6ADFF0 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=AV-CB-01&SESSIONID=50185L8FF3ADFF9 HTTP 1.1" 200 3855 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-16&product_id=RP-LI-02" 468 125.17 14
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5015L4FF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)" 468 125.17 14
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5035L7FF6ADFF0 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=AV-CB-01&SESSIONID=50185L8FF3ADFF9 HTTP 1.1" 200 3855 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-16&product_id=RP-LI-02" 468 125.17 14

Search job inspector

This search has completed and has returned **1** results by scanning **1** events in **0.056** seconds

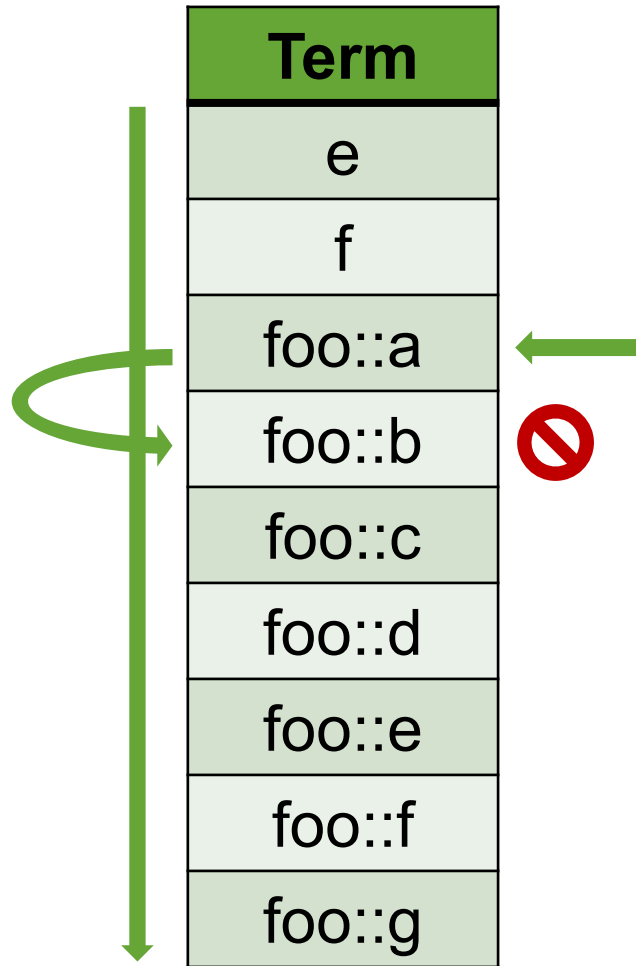
(SID: 1502365983.83) [search.log](#)

✓ Execution costs

Duration (seconds)	Component	Invocations
	0.00 command.fields	1
	0.00 command.search	1
	0.01 command.search.expand_search	1
	0.00 command.search.index	2

foo="*a" vs foo="a*"

foo="*a" vs foo="a*"



- ▶ Terms are sorted lexicographically within the TSIDX file.
- ▶ Binary search the index for the first matching term.
- ▶ For $f_{oo} = "a^*"$, continue downward until we come to the first none matching term.

foo="*a" vs foo="a*"

foo="*a" vs foo="a*"



- ▶ Same as trailing wildcard, start with the first matching term.
- ▶ However this time we must check all events that match our field name.
- ▶ Only when we get to "g", can we stop the search.

Trailing Wildcard + Unindexed

foo="*a" vs foo="a*"

Term
f
foo::
foo::a
foo::c
foo::d
foo::e
foo::f
foo::g
foo::h
foo::i
foo::j
foo::k
foo::l
foo::m
foo::n
foo::o
foo::p
foo::q
foo::r
foo::s
foo::t
foo::u
foo::v
foo::w
foo::x
foo::y
foo::z
g

- ▶ What if we searched for `bar="*z"`?
- ▶ Lispy is for this search is `"[]"`.
- ▶ Skips TSIDX reducing altogether, relying completely on Journal extraction.

Search #4

Transactions

```
index="conf2017"
```

```
| transaction date_hour
```

VS

```
index="conf2017"
```

```
| stats count by date_hour
```

splunk> App: Search & Reporting ▾

Administrator ▾

Messages ▾

Settings ▾

Activity ▾

Help ▾

Find

Search Datasets Reports Alerts Dashboards

Search & Reporting

New Search

Save As ▾ Close

index="conf2017" | stats count by date_hour foo | chart count by date_hour

All time ▾



✓ 26 events (before 8/10/17 3:42:25.000 PM) No Event Sampling ▾

Job ▾



Verbose Mode ▾

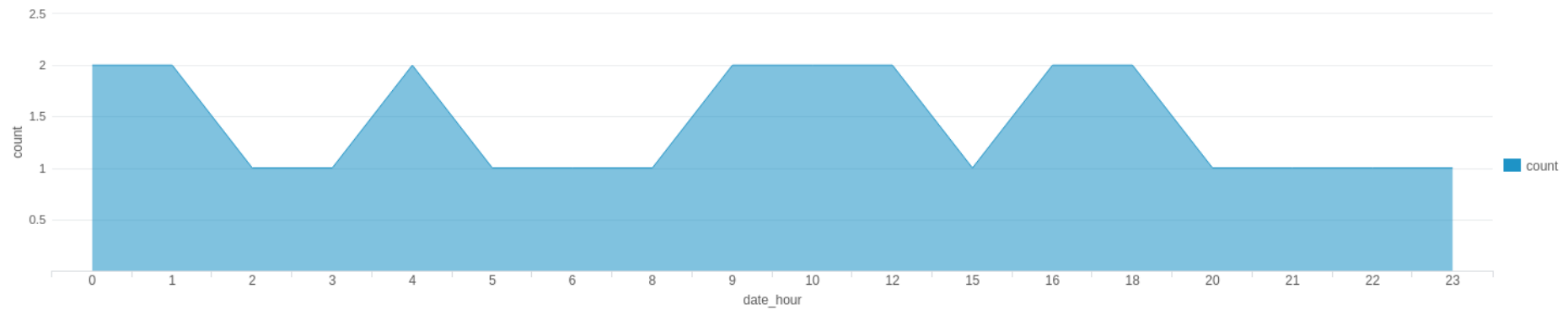
Events (26)

Patterns

Statistics (18)

Visualization

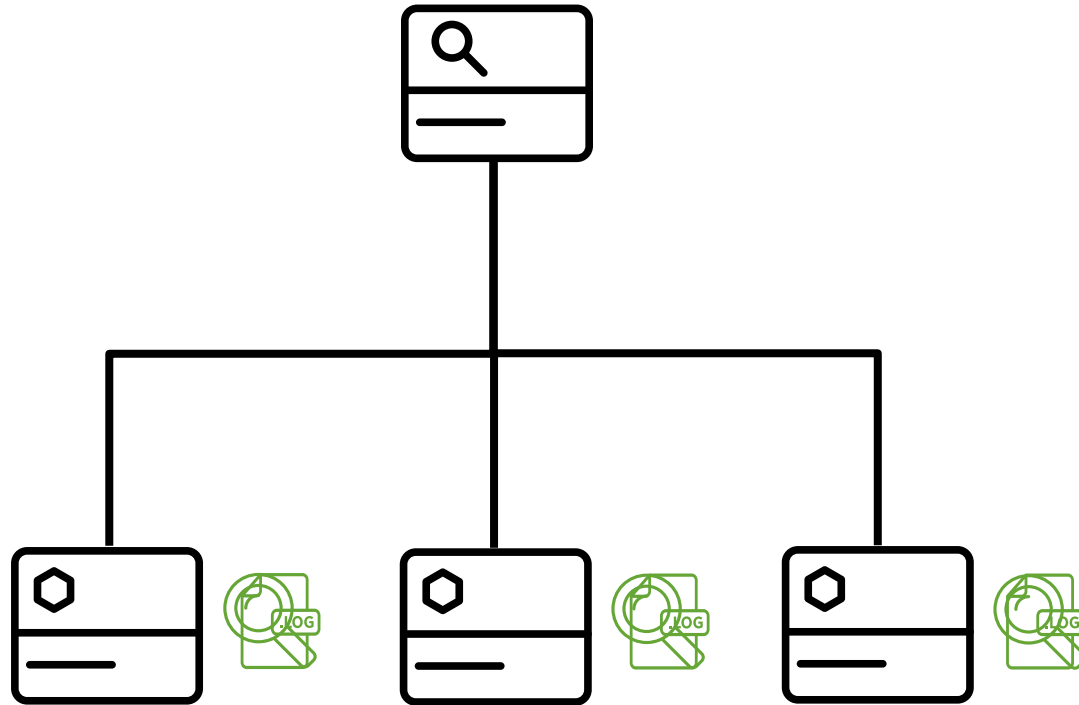
Area Chart Format Trellis



date_hour	count
0	2
1	2
2	1
3	1
4	2
5	1
6	1
8	1
9	2
10	2

Back to the Search Head

transaction vs stats



Dispatch Folder

transaction vs stats

- ▶ Directory of all saved and running searches on the Search Head.
- ▶ `sid` can be obtained in the Job Inspector.

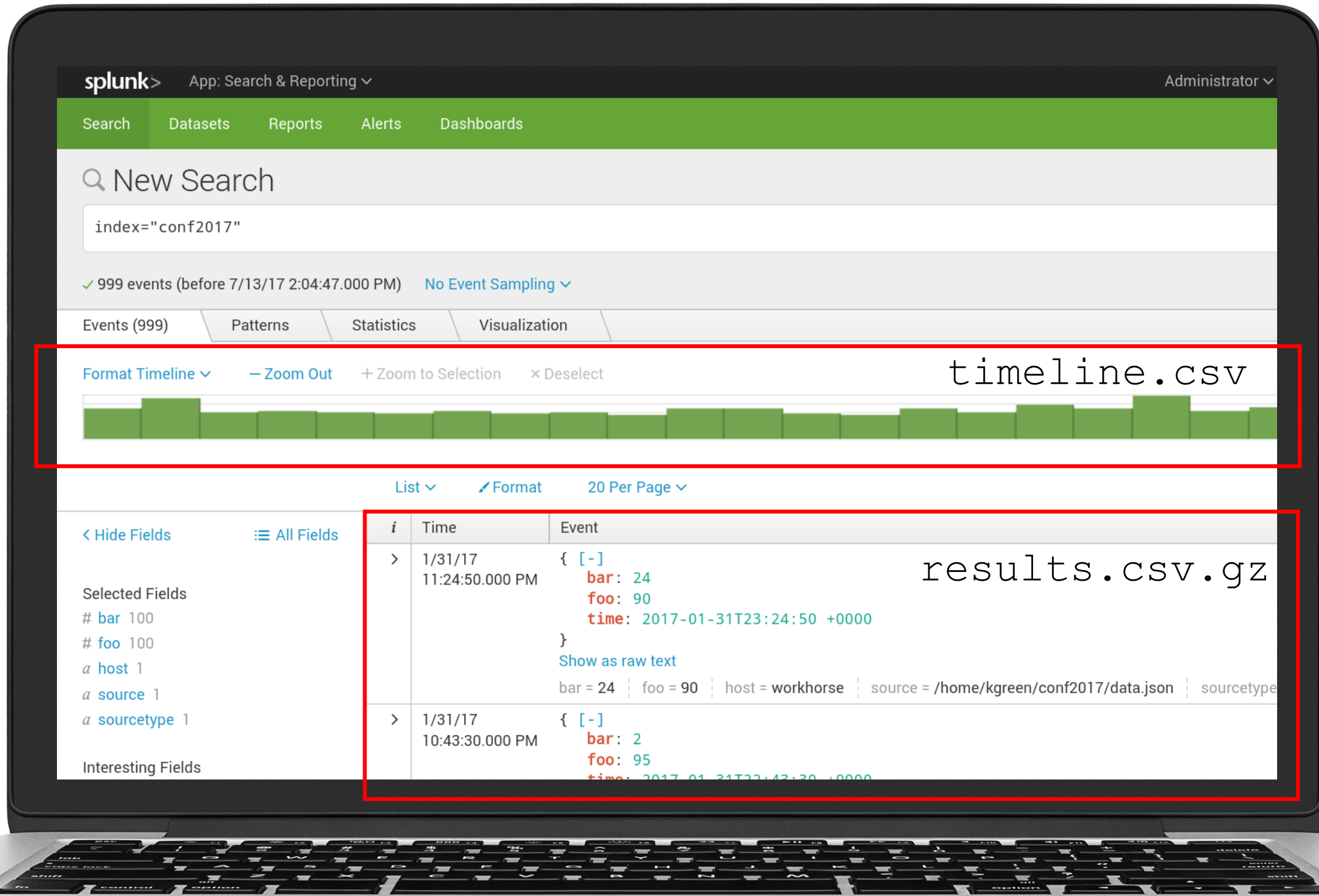
```
$ cd $SPK_SH/var/run/splunk/dispatch
$ ls -l
    1501601198.142
    1501601202.143
    1501601739.144
    1501601740.145
→   1501601741.146
    1501601742.147
```

Search Folder

transaction vs stats

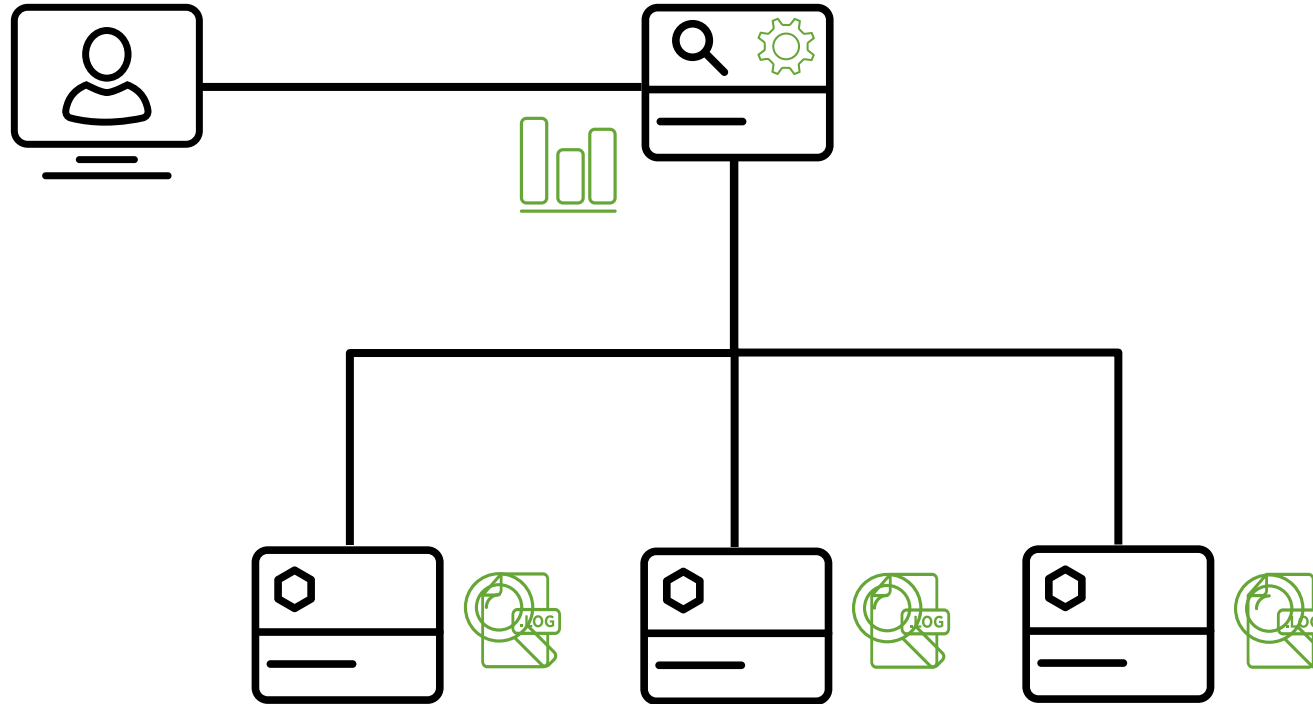
- ▶ Collection of all data being returned from the indexers.
- ▶ `results.csv.gz`
compressed events.
- ▶ `timeline.csv`
UI timeline numbers.

```
$ cd 1501601741.146/  
$ ls -l  
    args.txt  
    buckets  
    custom_prop.csv  
    events  
→ timeline.csv  
    info.csv  
    peers.csv  
→ results.csv.gz  
    search.log
```



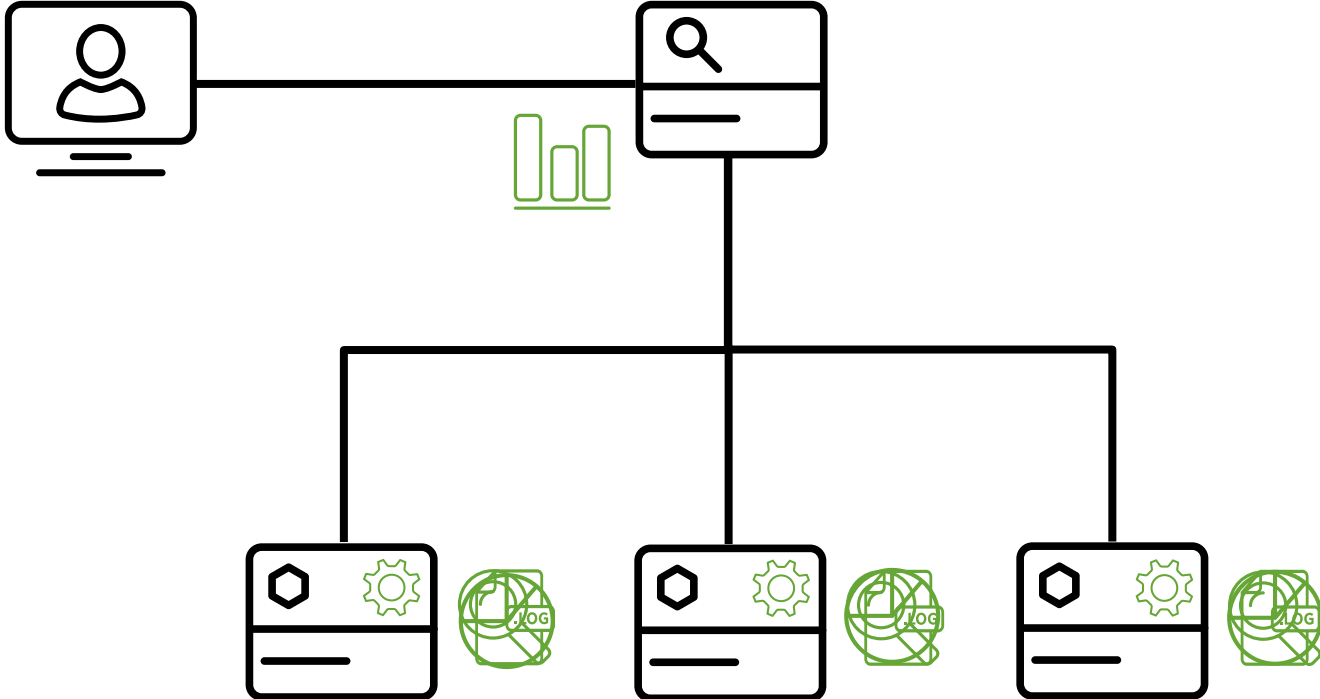
Transaction Workflow

transaction vs stats



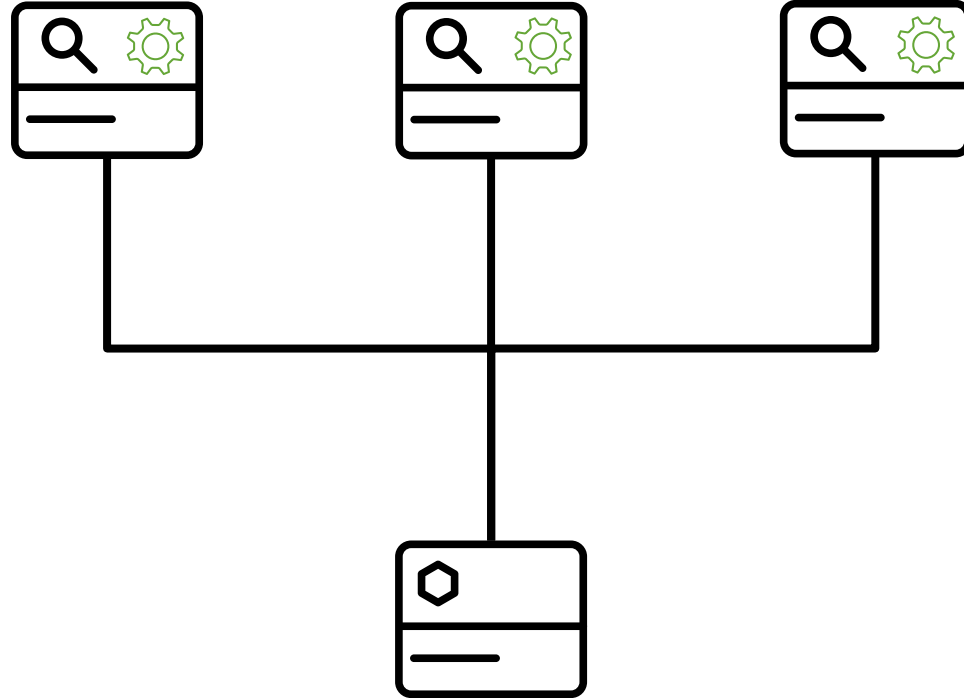
Stats Workflow

transaction vs stats



Search Head Cluster

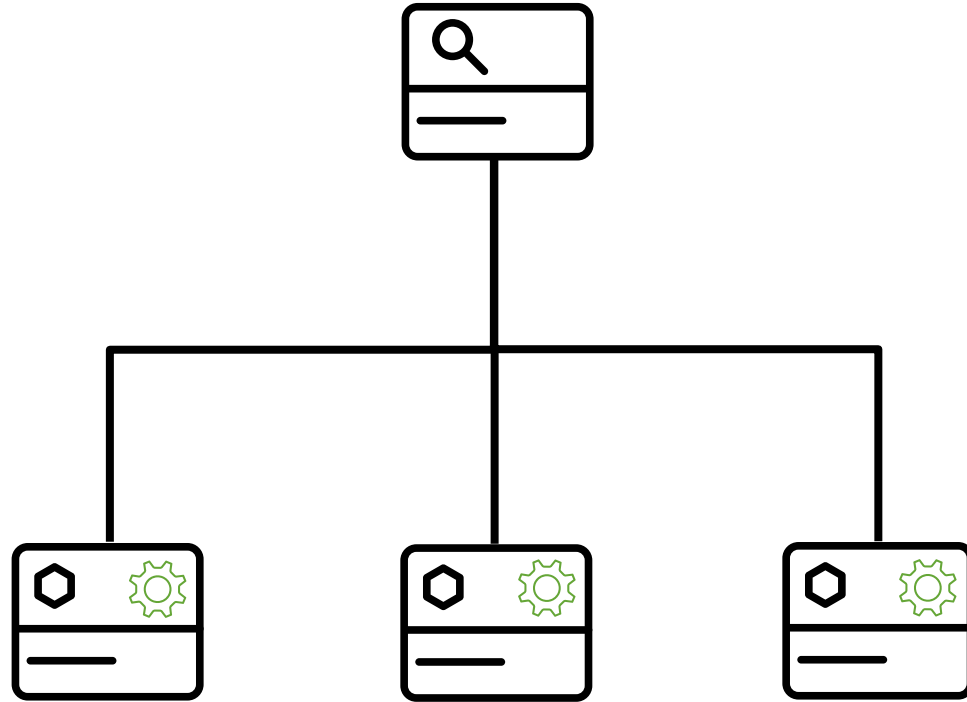
transaction vs stats



Performance boost for transactions running in parallel.

Distributed Search & Index Cluster

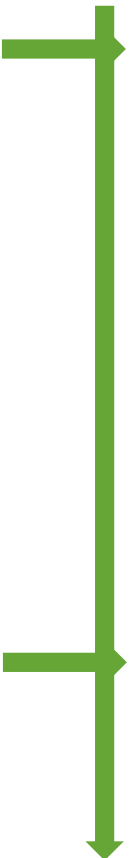
transaction vs stats



Scalable performance boost to stats and eval.

Stats Computation

transaction vs stats

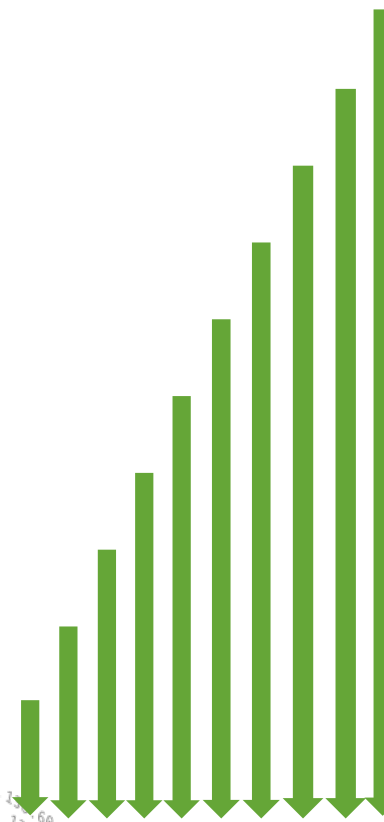


Time	Foo	Bar
2017-09-01T09:00:00	a	z
2017-09-02T07:00:00	b	y
2017-09-03T02:00:00	c	x
2017-09-04T13:00:00	d	w
2017-09-05T23:00:00	e	v
...		
2017-09-21T15:00:00	v	e
2017-09-23T16:00:00	w	d
2017-09-24T09:00:00	x	c
2017-09-25T06:00:00	y	b
2017-09-26T10:00:00	z	a

- ▶ stats only concerns itself with a single event at once.
- ▶ Requires only one pass to complete the computation.
- ▶ For stats count Splunk returns value plus event occurrence count.
 - For example: hour "09" has 2 events.

Transaction Discovery

transaction vs stats



Time	Foo	Bar
2017-09-01T09:00:00	a	z
2017-09-02T07:00:00	b	y
2017-09-03T02:00:00	c	x
2017-09-04T13:00:00	d	w
2017-09-05T23:00:00	e	v
...		
2017-09-21T15:00:00	v	e
2017-09-23T16:00:00	w	d
2017-09-24T09:00:00	x	c
2017-09-25T06:00:00	y	b
2017-09-26T10:00:00	z	a

- ▶ Splunk must iterate over each event for every transaction window.
- ▶ Looking at a time complexity difference between n and n^2 .
- ▶ Running only on a single Search Head doesn't help the situation.

Search #5

transaction plus stats

index=conf2017

```
| transaction foo
```

```
| stats count by foo
```

transaction plus stats

transaction plus stats

- ▶ Splunk runs everything on the Indexer, until the first "slow" command forces otherwise.
- ▶ Everything trailing that command, will be forced to run on the Search Head.
- ▶ `transactions` and `joins` are examples of commands which would trigger this behavior.

```
index=conf2017 ←
| transaction foo ❌
| stats count by foo
```

reduceSearch	transaction foo
remoteSearch	litsearch index=conf2017 fields keepcolororder=t "_txn_ends_with
reportSearch	stats count by foo
request	<pre>{ [-] adhoc_search_level: smart auto_cancel: 30 check_risky_command: false custom.dispatch.earliest_time: 0 custom.dispatch.latest_time: custom.dispatch.sample_ratio: 1 custom.display.general.type: statistics custom.display.page.search.mode: smart custom.display.page.search.tab: statistics custom.search: index=conf2017 transaction foo stats count earliest_time: 0 indexedRealtime: latest_time: preview: 1 provenance: UI:Search rf: * sample_ratio: 1</pre>

Takeaways

You're all wizards now!

1. Leverage `stats` and `eval` over `transactions` whenever possible.
2. Choose trailing wildcards over leading in queries that require such functionality.
3. Look into indexing important fields who shares values with other fields.
4. Move slow commands as far right into the query string as possible.

No Magic

Just Splunk



Don't forget to **rate this session** in the
.conf2017 mobile app

splunk> .conf2017

Q&A
