How Fast Is Fast Enough? Improving Splunk Performance With Batch Mode Search

Becky Burwell

Production Engineer, Flickr/Yahoo

.conf2016

splunk>

Splunk Batch Search Performance Enhancements @Flickr

- The Punch Line: What got better
- Flickr Splunk Architecture Overview
- Upgrade Process
- Analysis Process
- Next Steps

The Punch Line:
Summary of Performance Analysis

- Just upgrading to 6.2.3 -> 6.3.2 was a big performance win.
    - Great news, but it confounded batch search analysis!
- With Batch Search mode parallelization, both scheduled and ad hoc searches got faster
    - Best performance increase seen in long running jobs
    - Batch search mode can help when we have qualified searches

Summary of Performance Gains

	Version 6.2.3	Splunk 6.3.2	Splunk 6.3.3 + batch search mode
Scheduled Jobs: Median Job Length	4.00 sec	2.57s = 1.5x faster	0.793s = **5x faster**
Ad hoc Jobs (> 0): Median Total Job Length	2.93 sec	0.92s = 3x faster	0.83s = **3.5x faster**

SPLUNK @ FLICKR

Design and Architecture

Production Search Head Cluster

4 Search Heads, Splunk 6.3.x

24 Indexers with 64 GB Memory, 12 core, SSD, 6.3.x

3000 Forwarders; Indexing 8TB/day

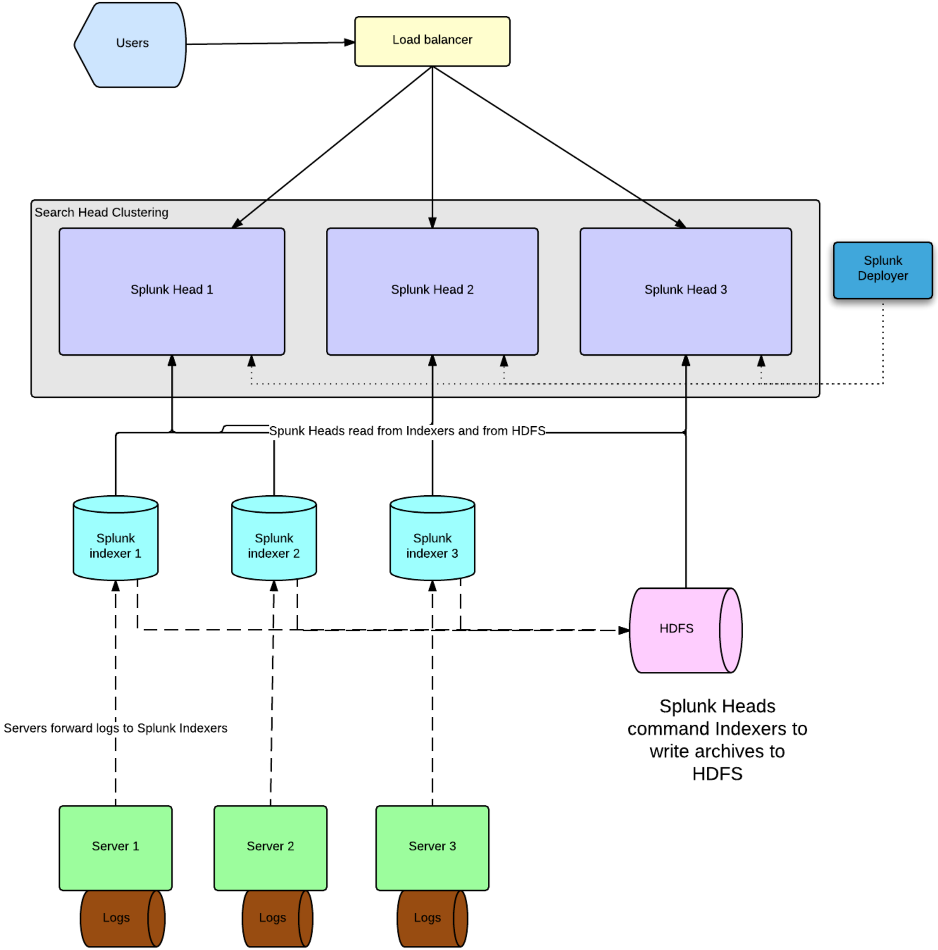
Stage Search Head Cluster

3 Search Heads running Splunk 6.4.2

1 Indexers

2 Forwarders

User data



Splunk 6.3 Searching and Indexing Performance Enhancements

→ Parallelization of indexing

→ Parallel summarization for data models

→ Parallel summarization for report accelerations

→ Batch mode Search Parallelization

Which Enhancements to choose?

Choices that don't fit

- Parallelization of Indexing:
  - Fewer # of cores on Flickr indexers < recommended indexer hardware
- Parallelization of data models:
  - Don't use at all or a lot
- Parallelization of data summary
  - Don't use a lot

Choices that fit

- At Flickr, we chose to implement only Search parallelization:
  - Flickr uses Search a lot
  - Flickr indexer memory exceeds reference hardware

    → A Good fit

Batch Mode Search Parallelization

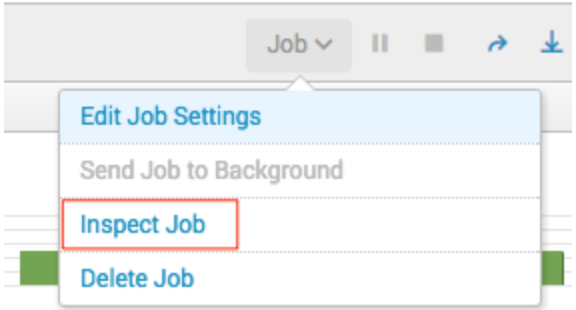
- Process involves opening additional search pipelines on each indexer, processing multiple buckets simultaneously.
- Batch mode searches search and return event data by bucket, instead of by time.
- Adding more batch search pipelines, multiple buckets are processed simultaneously, speeding the return of search results

Not every search can be batched

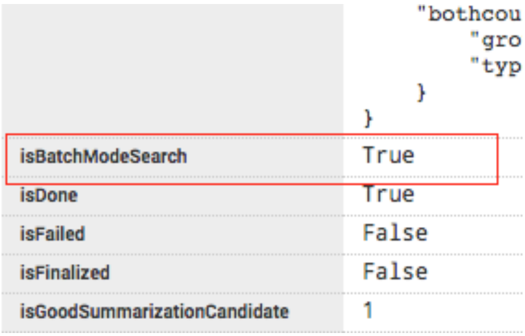
- searches must use a generating command
- may include transformations like chart, stats
- cannot include transaction
- cannot require time ordered events (no tail or head)
- tip: look through your saved searches to see what can be batched

Can this Search be batched?

1. run a search
2. navigate to Job -> Inspect Job



3. look for isBatchModeSearch



```
                   "bothcou
                        "gro
                        "typ
                   }
              }
```

isBatchModeSearch	True
isDone	True
isFailed	False
isFinalized	False
isGoodSummarizationCandidate	1

How to enable batch search mode

Be sure to read the documentation:

http://docs.splunk.com/Documentation/Splunk/6.3.3/Knowledge/Configurebatchmodesearch#Configure_batch_mode_search_parallelization

In limits.conf on each indexer

```
[search]
   batch_search_max_pipeline = <int>
   batch_search_max_results_aggregator_queue_size = <int>
   batch_search_max_serialized_results_queue_size = <int>
```

- The `batch_search_max_results_aggregator_queue_size` parameter controls the size of the results queue. The results queue is where the search pipelines leave processed search results. Default=100MB.
- The `batch_search_max_serialized_results_queue_size` parameter controls the size of the serialized results queue, from which the batch search process transmits serialized search results. Default=100 MB

Flickr Batch Mode Settings

Flickr left the queue size settings alone.

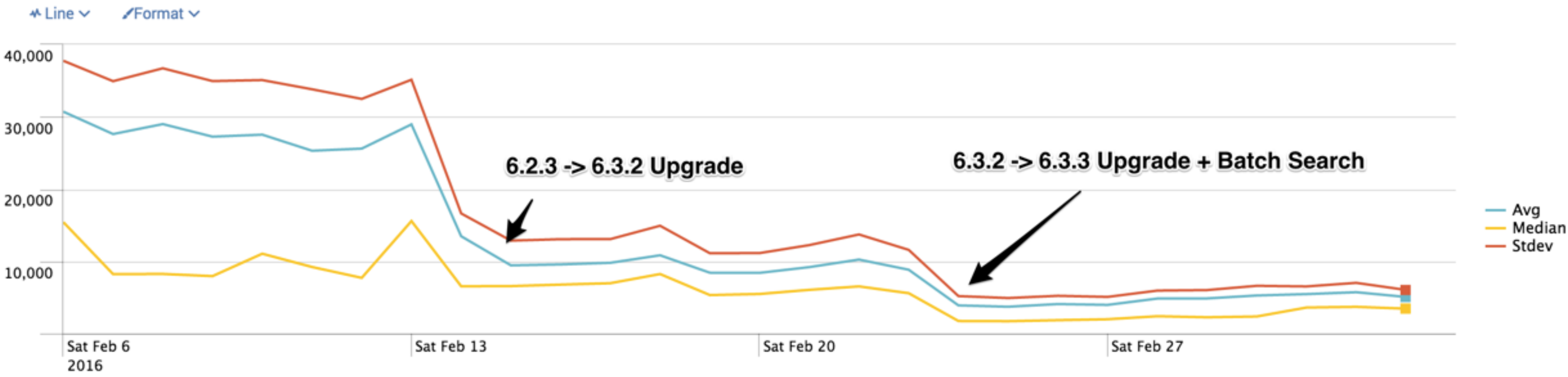
Flickr’s settings:

```
[search]
  batch_search_max_pipeline = 2
```

Upgrade Process

- Flickr had been running Splunk indexer 6.2.3
- Feb 14 upgraded indexers 6.2.3 -> 6.3.2
- Feb 16 enabled batch search mode but hit known bug fixed in 6.3.3
- Upgraded heads, then indexers to 6.3.2 -> 6.3.3 on Feb 23
- **Analysis of 6.2.3 -> 6.3.2 showed huge improvements in search speeds, especially of the longest running jobs.**
    - **Great news, but was it real?**
    - **Confounded analysis of batch searching improvements.**

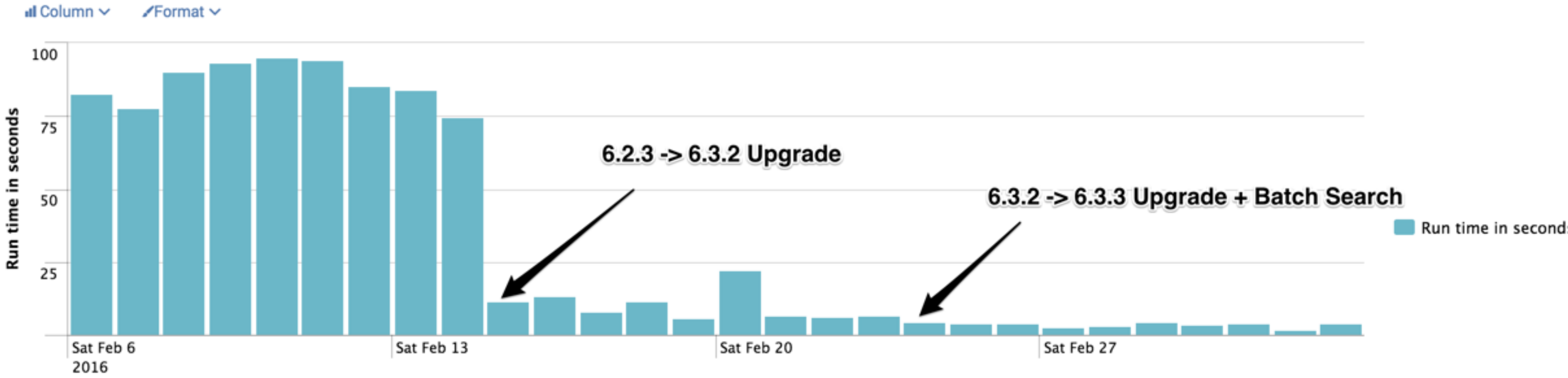
Analysis of Scheduled Searches Total Running Time: Mean, Median, and Standard Deviation



First Upgrade improved performance of slowest jobs.
Second Upgrade and Batch Search improved all jobs.

One example of long running job that got much faster

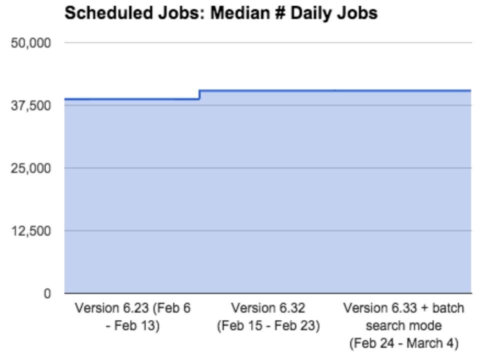
```
sourcetype="flickrecs" "In Network: "
  | rex "(?<incount>\d*)\, Out of Network: (?<outcount>\d*)"
  | eval bothcount=outcount+incount| stats count by bothcount
  | where bothcount < 50
```



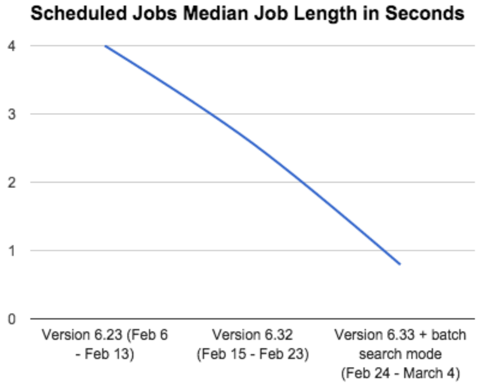
6.2.3 -> 6.3.2 Upgrade

6.3.2 -> 6.3.3 Upgrade + Batch Search

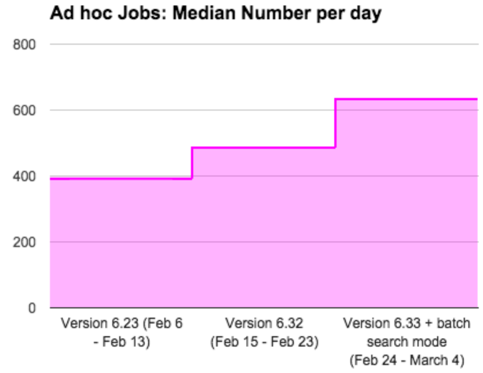
Four Key Job Metrics vs Version: what they showed



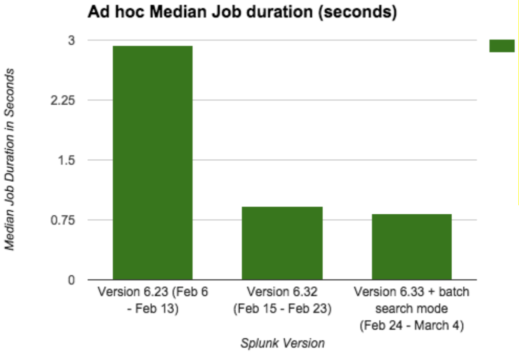
Number of Scheduled Jobs remained steady throughout.



Median Scheduled Jobs got faster and faster.



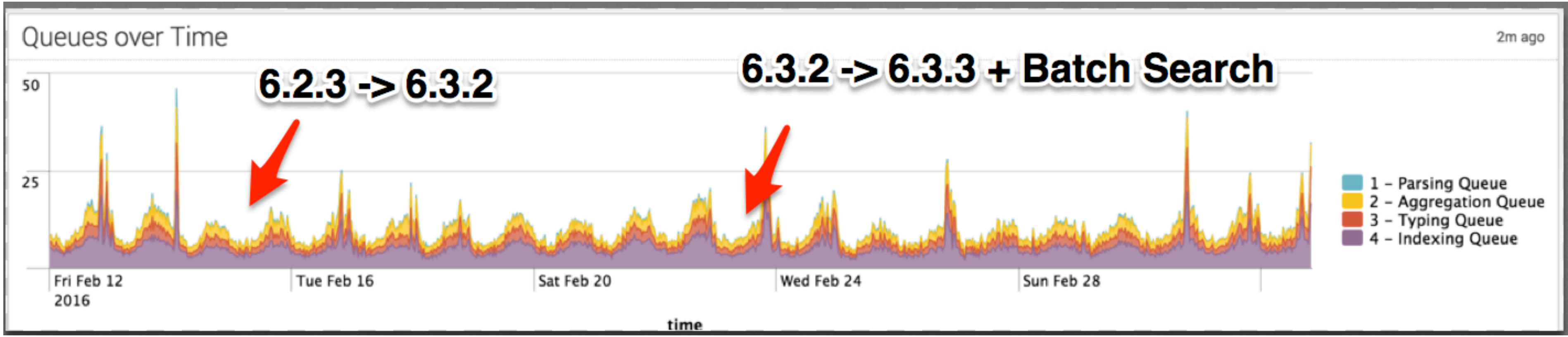
Users ran 50% more queries than before.



User queries tripled in speed.

Indexing did not get worse

Indexing Queues don't seem to be impacted by batch search mode



Summary of Performance Analysis

- Just upgrading to 6.2.3 -> 6.3.2 was a big performance win.
- With batch searching, both scheduled and ad hoc searches got faster
- Bonus: Discovered many failed searches

Next Steps

- We are happy with search speed of batch search mode, feeling of site side seems good
- Leaving batch search mode concurrency at 2
- If time for median saved searches or adhoc searches start to grow, we can try a concurrency of 3
- We will do user education to take advantage of batch search mode searches
- Need an inventory process of saved searches: discovered a number of invalid searches that add to the noise in the error logs

THANK YOU

.conf2016

splunk>

THANK YOU

.conf2016

splunk>

Appendix: Data

.conf2016

splunk>

Key Metrics Data

	Version 6.23 (Feb 6 - Feb 13)	Version 6.32 (Feb 15 - Feb 23)	Version 6.33 + batch search mode (Feb 24 - March 4)
Scheduled Jobs: Median # Daily Jobs	38,735	40,395	40,417
Scheduled Jobs: Median Job Length (secs)	4.00	2.57	0.793
Ad hoc Jobs (> 0 secs): Median # of Daily Jobs	392	486	633
Ad hoc Jobs (> 0): Median Total Job Times (secs)	2.93	0.92	0.83