# Building your app on an **accelerated** data model

Helge Klein

# Disclaimer

During the course of this presentation, we may make forward looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC. The forward-looking statements made in the this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward looking statements we may make.

In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not, be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

# About Helge

- Twitter: **@HelgeKlein**
- Splunk Revolution Award Winner 2014
- Citrix CTP, Microsoft MVP, VMware vExpert
- Founder at vast limits, the **uberAgent** company
- Architect of what later became **Citrix Profile Management**

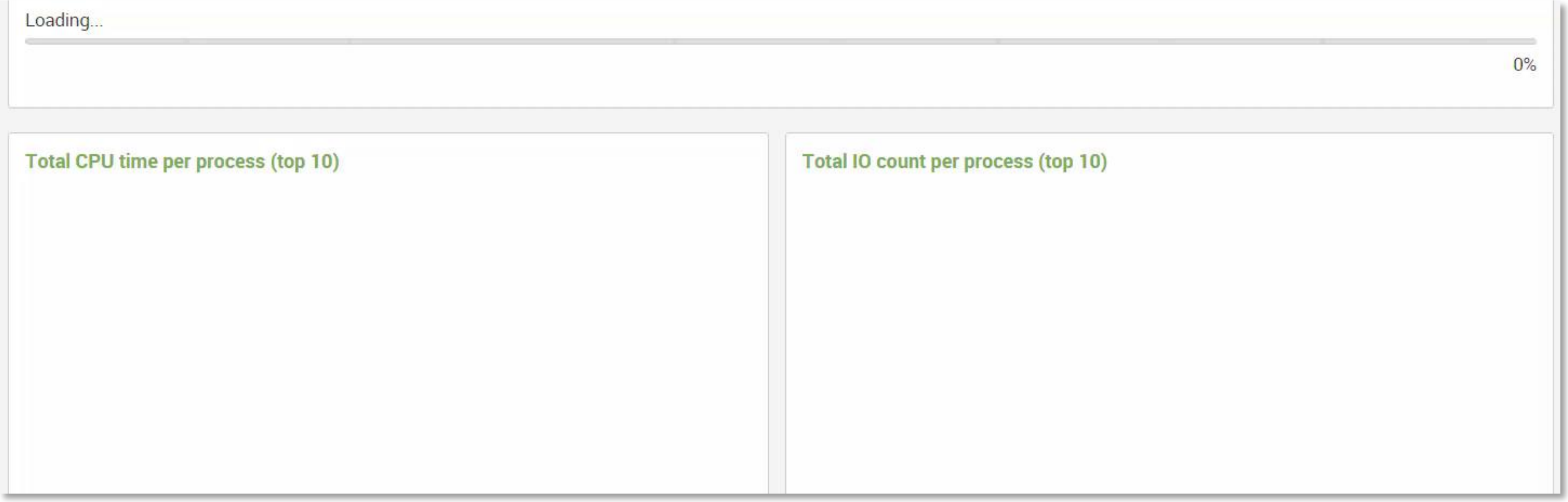
# About uberAgent

- Helge‘s background: **end-user computing**
  - A lot of Citrix and Windows…
- Loved Splunk the minute he saw it
- Why do people only use Splunk for security?
  - Let‘s change that!
- **uberAgent** was born

# Why

## accelerate?

# Ever Seen This?



Loading...

0%

Total CPU time per process (top 10)

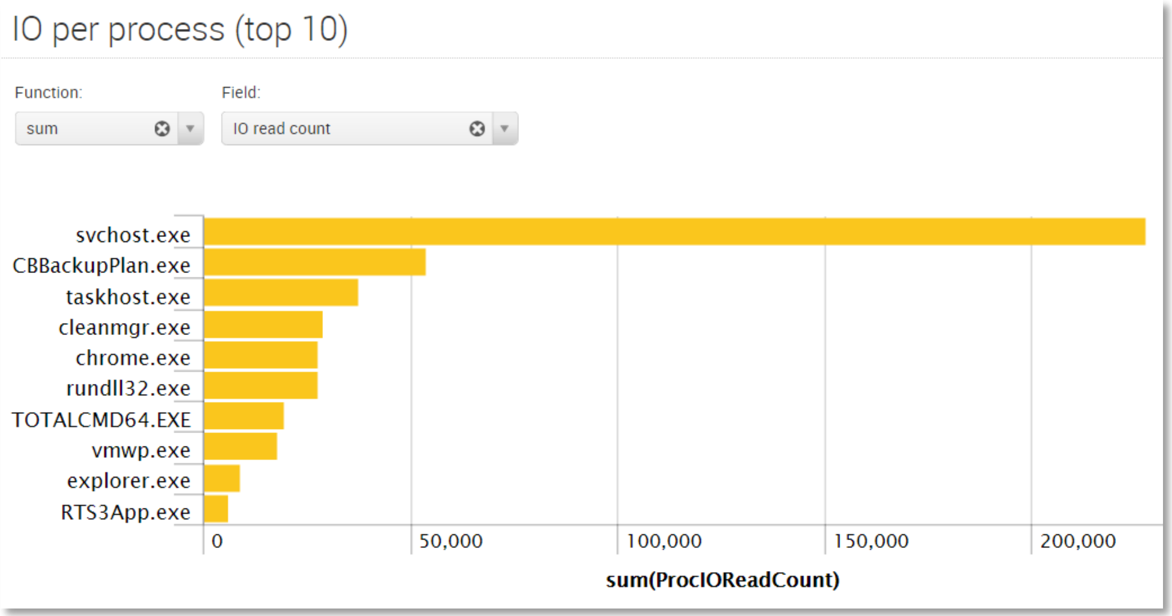
Total IO count per process (top 10)

# Needle in a haystack

- Splunk is **very fast** with needle in a haystack searches
  - E.g. find one keyword in millions of events
- Splunk is **not so fast** with searches that perform calculations on millions of events
  - E.g. calculate the sum or average of fields

# Example: Process IO

- Show 10 processes with highest IO count:

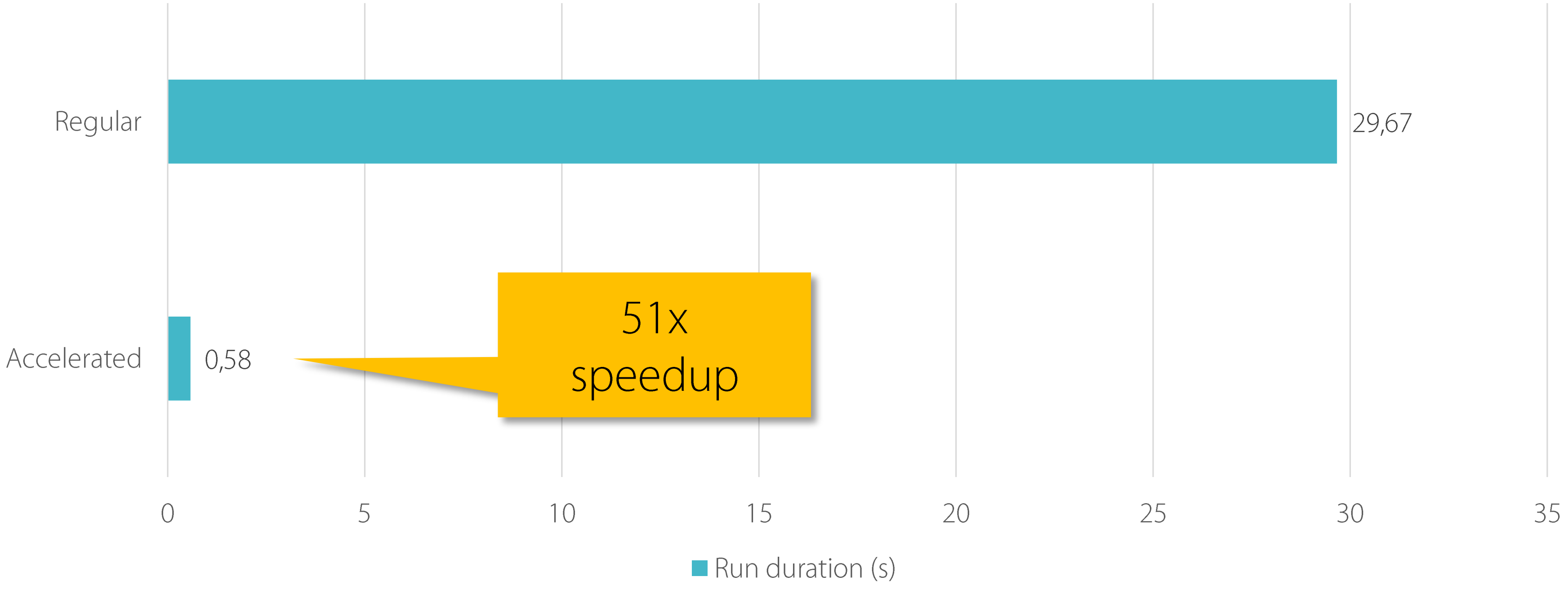


IO per process (top 10)

Function: sum

Field: IO read count

# Run Duration



51x speedup

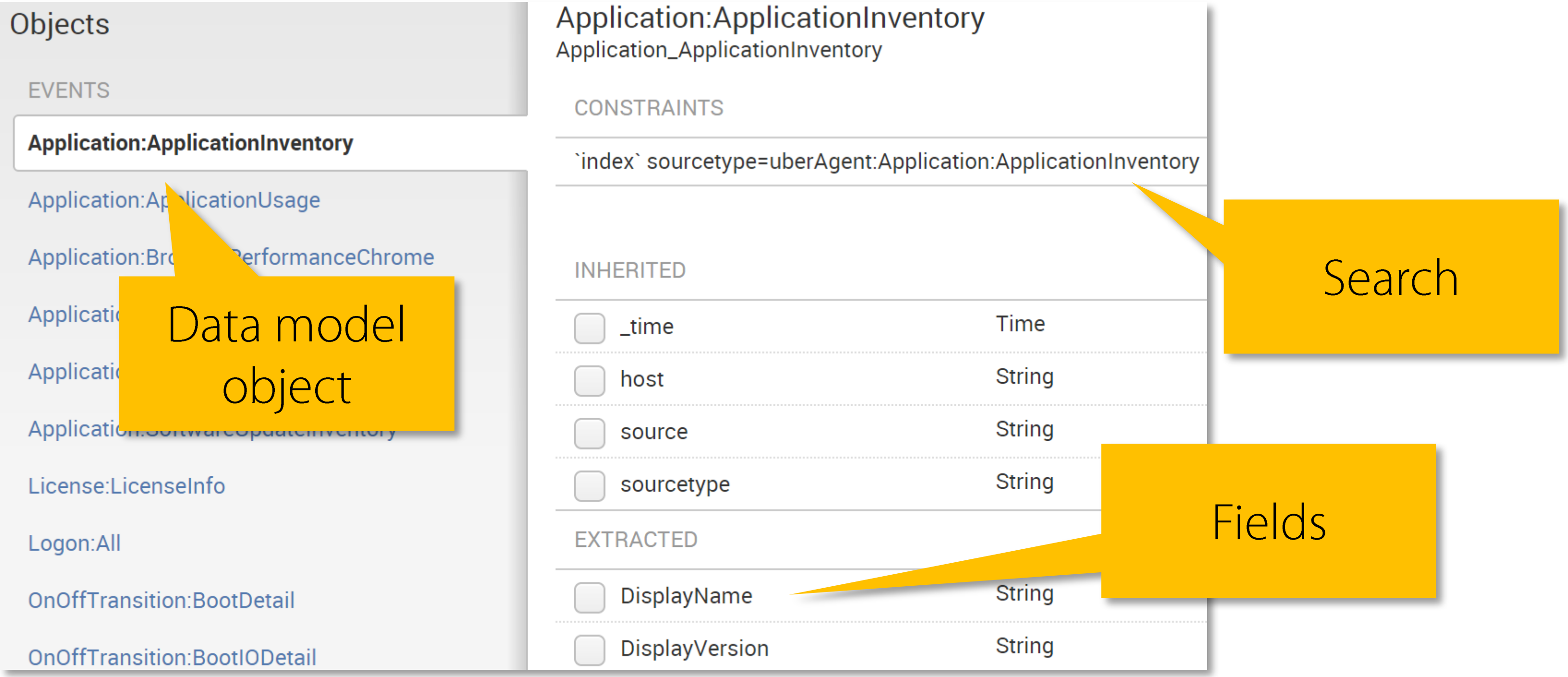
## Data model acceleration

# How it works

# Data Model

- A data model adds a **second layer** to your data
  - Does **not** remove classic Splunk functionality
  - Predefined fields create a **schema**

# Data Model: Example



Objects

EVENTS

**Application:ApplicationInventory**

Application:ApplicationUsage

Application:Br PerformanceChrome

Applicati

Applicati

Application:SoftwareUpdateInventory

License:LicenseInfo

Logon:All

OnOffTransition:BootDetail

OnOffTransition:BootIODetail

Application:ApplicationInventory

Application_ApplicationInventory

CONSTRAINTS

`index` sourcetype=uberAgent:Application:ApplicationInventory

INHERITED

| | | |
|---|---|---|
| □ | _time | Time |
| □ | host | String |
| □ | source | String |
| □ | sourcetype | String |

EXTRACTED

| | | |
|---|---|---|
| □ | DisplayName | String |
| □ | DisplayVersion | String |

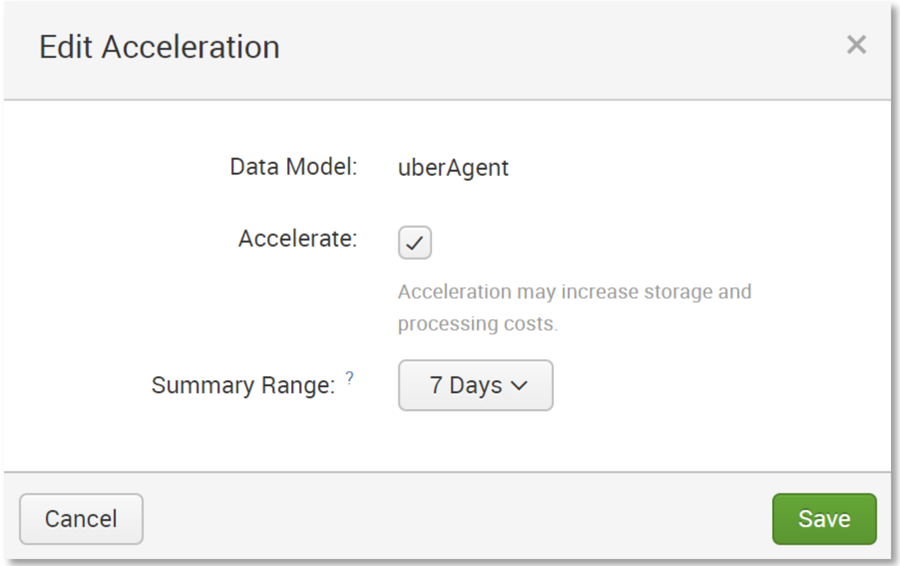
Data model object

Search

Fields

# Acceleration

- A data model can optionally be **accelerated:**



Edit Acceleration

Data Model: uberAgent

Accelerate: ☑

Acceleration may increase storage and processing costs.

Summary Range: ? 7 Days

# Field Extraction

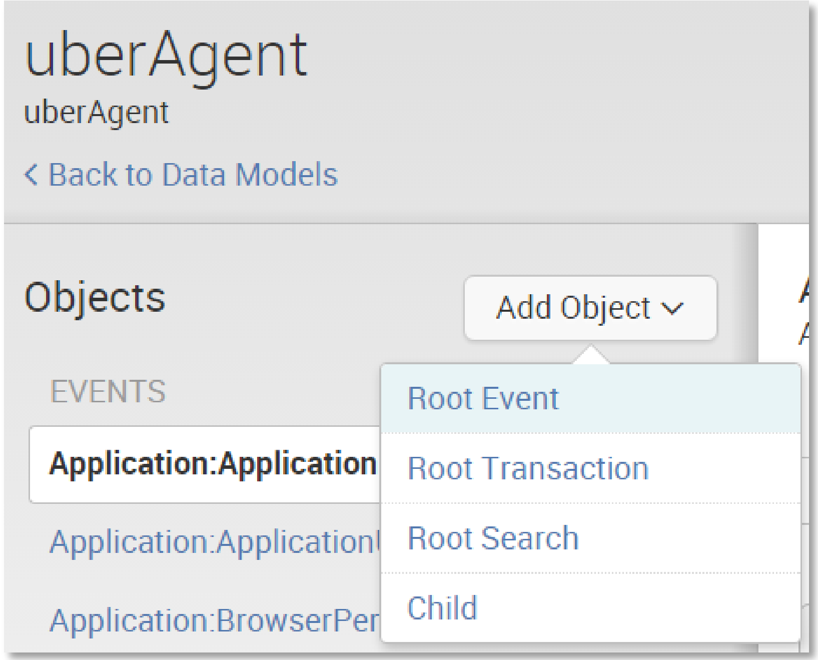
- Normally Splunk extracts fields from raw text data at **search time**
- When a data model is accelerated, a field extraction process is added to **index time**
  - Pro: better search performance
  - Con: higher indexer utilization

# HPAS

- Extracted data model fields are stored in the high-performance analytics store (HPAS)
- Created on the **indexers**
  - .tsidx files
  - Parallel to the regular event buckets
  - Not replicated in an indexer cluster

# Caveats

- Only data model **event** hierarchies can be accelerated:



uberAgent

uberAgent

< Back to Data Models

Objects

Add Object ⌄

EVENTS

Application:Application

Application:Application

Application:BrowserPer

Root Event

Root Transaction

Root Search

Child

# Caveats

- Once a data model is accelerated, it **cannot be edited**
  - Simple to work around by disabling acceleration before edits and re-enabling it after

# Under the Hood

# HPAS Population

- The high-performance analytics store is populated by **scheduled searches**
  - Run every 5 mins
- The HPAS spans a user-defined **time range**
  - Older events are purged automatically to limit disk usage
  - Maintenance process runs every 30 minutes

# Populating Searches

- One auto summarizing search is added to the scheduler per data model *object*
  - These searches have a **low priority**
  - Total number of these searches is **limited**
- New in **Splunk 6.3**: parallel summarization
  - 2 concurrent search jobs to build summary files instead of 1

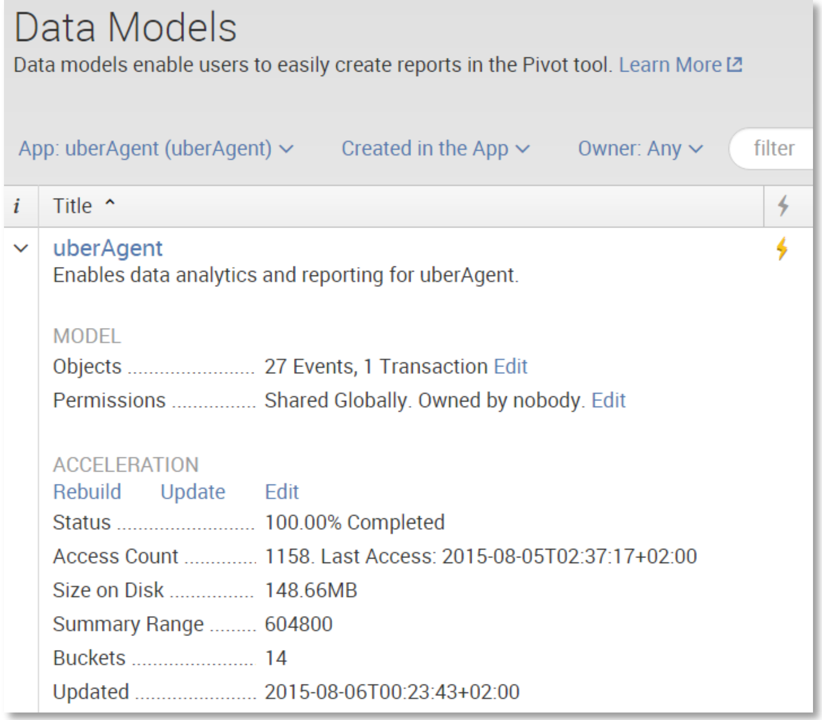
# Populating Searches

- Configuration in *limits.conf*:
  - max_searches_perc
    - Percentage of **system-wide** concurrent searches the scheduler can run
    - Default: 50
  - auto_summary_perc
    - Percentage of **scheduler se** ization
    - Default: 50

50% of 50%:
Only **25%** of all concurrent searches are available for data model acceleration

# Check Status

- From the UI: (Settings > Data Models)
- UI bug was fixed in 6.2.3



Data Models

Data models enable users to easily create reports in the Pivot tool. Learn More

uberAgent

Enables data analytics and reporting for uberAgent.

MODEL

Objects ........................ 27 Events, 1 Transaction Edit

Permissions ............... Shared Globally. Owned by nobody. Edit

ACCELERATION

Rebuild Update Edit

Status ............................ 100.00% Completed

Access Count ............. 1158. Last Access: 2015-08-05T02:37:17+02:00

Size on Disk ............... 148.66MB

Summary Range ......... 604800

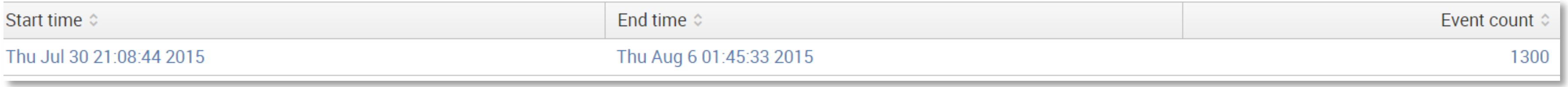
Buckets ........................ 14

Updated ....................... 2015-08-06T00:23:43+02:00

# Check Status

- From a search:

```
| tstats summariesonly=t min(_time) as min,
         max(_time) as max count from datamodel=uberAgent
| eval "Start time"=strftime(min, "%c")
| eval "End time"=strftime(max, "%c")
| eval "Event count"=count
| fields "Start time" "End time" "Event count"
```



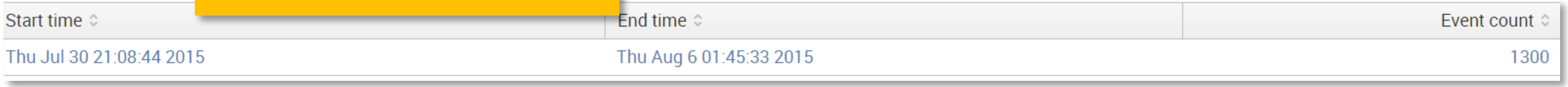
| Start time | End time | Event count |
|---|---|---|
| Thu Jul 30 21:08:44 2015 | Thu Aug 6 01:45:33 2015 | 1300 |

# Check Status

- From a search:

```
| tstats summariesonly=t min(_time) as min,
         max(_time) as max count from datamodel=uberAgent
| eval "Sta    time"=strftime(min, "%c")
| eval              me(max, "%c")
| eval              nt
| field             nd time" "Event count"
```

**Summariesonly:**
Searches the
HPAS only



| Start time | End time | Event count |
|---|---|---|
| Thu Jul 30 21:08:44 2015 | Thu Aug 6 01:45:33 2015 | 1300 |

## Data models and

# Apps

# Enabling Acceleration

- In *datamodels.conf*:

```
[uberAgent]
acceleration = 1
acceleration.earliest_time = -1w
```

# Data Model Definition

- Filename: *modelname.json*
- Directory:

  $SPLUNK_HOME\etc\apps\appname\default\data\models
- Resides on the **search heads**
- Is sent to the **indexers** as part of the replication bundle

# Data Model Definition

- If you have the data model definition on multiple independent search heads, you get **multiple copies** of the HPAS:

```
$SPLUNK_DB
  index
    datamodel_summary
      bucket_id
        search_head_or_pool_id
```

One of these per (independent) search head

# Searching

accelerated data models

# What is Accelerated?

- The HPAS is used only with:
  - Pivot (UI and the *pivot* command)
  - The *tstats* command
- Not accelerated:
  - Regular searches
  - The *datamodel* command

# Search Commands

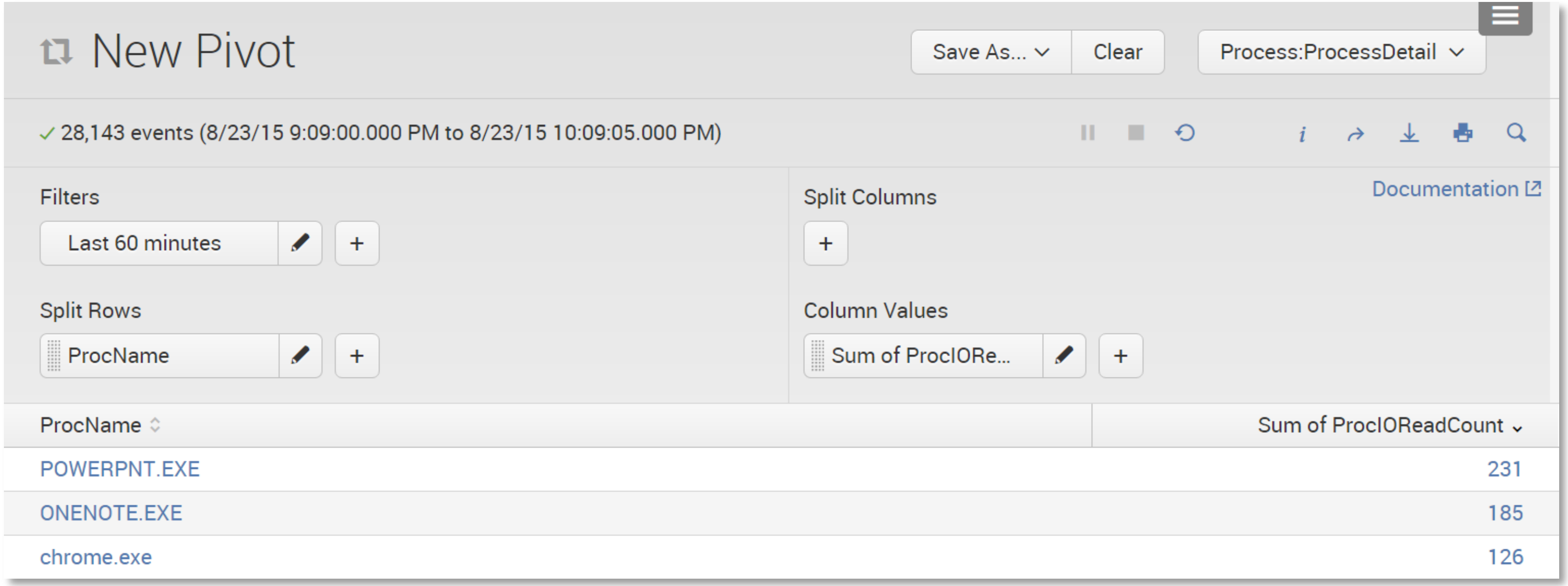
- *Tstats*
  - More familiar syntax
  - Does not support realtime searches
- *Pivot*
  - „Different“ syntax & capabilities
  - Supports realtime searches

# Tstats: The Principle

- Must be the 1st command in the search pipline
- Used in *prestats* mode
- Followed by:
  - *Stats*
  - *Chart*
  - *Timechart*

# Learning Tstats

- Build a sample search in Pivot Editor



✓ 28,143 events (8/23/15 9:09:00.000 PM to 8/23/15 10:09:05.000 PM)

Filters: Last 60 minutes

Split Rows: ProcName

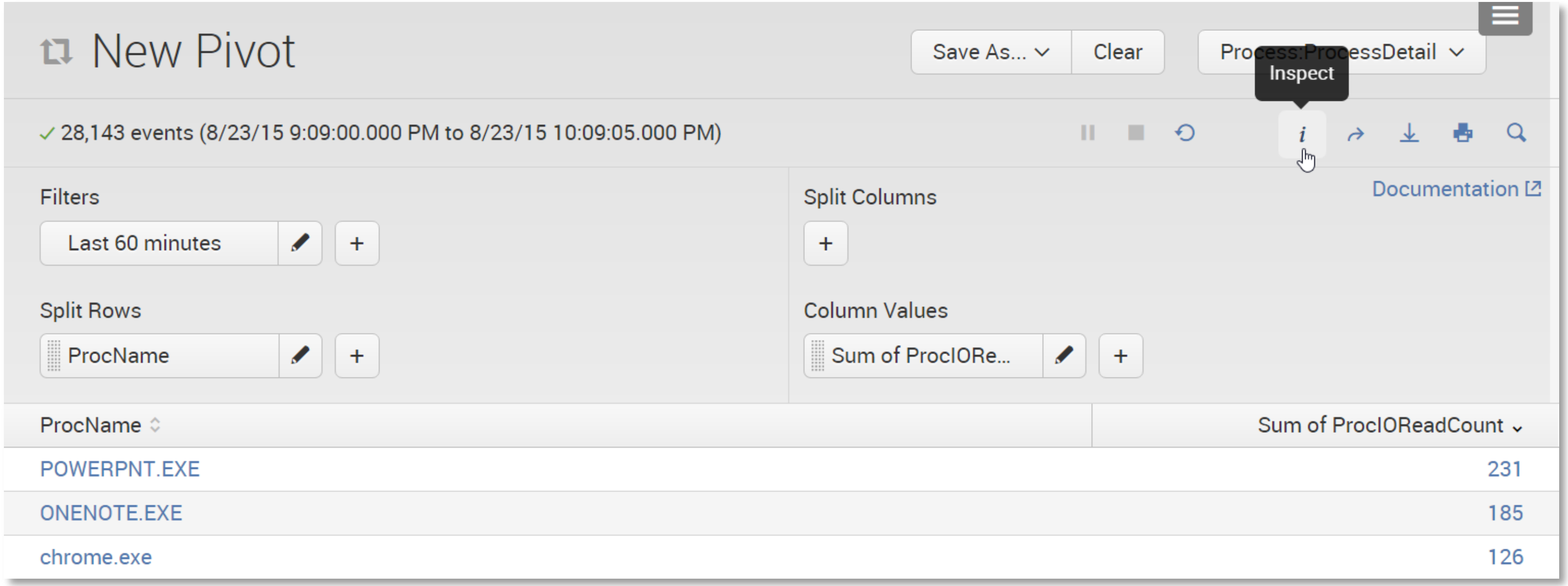
Split Columns

Column Values: Sum of ProcIORe...

| ProcName | Sum of ProcIOReadCount |
|---|---|
| POWERPNT.EXE | 231 |
| ONENOTE.EXE | 185 |
| chrome.exe | 126 |

# Learning Tstats

- Inspect the underlying search



New Pivot

✓ 28,143 events (8/23/15 9:09:00.000 PM to 8/23/15 10:09:05.000 PM)

Filters: Last 60 minutes

Split Rows: ProcName

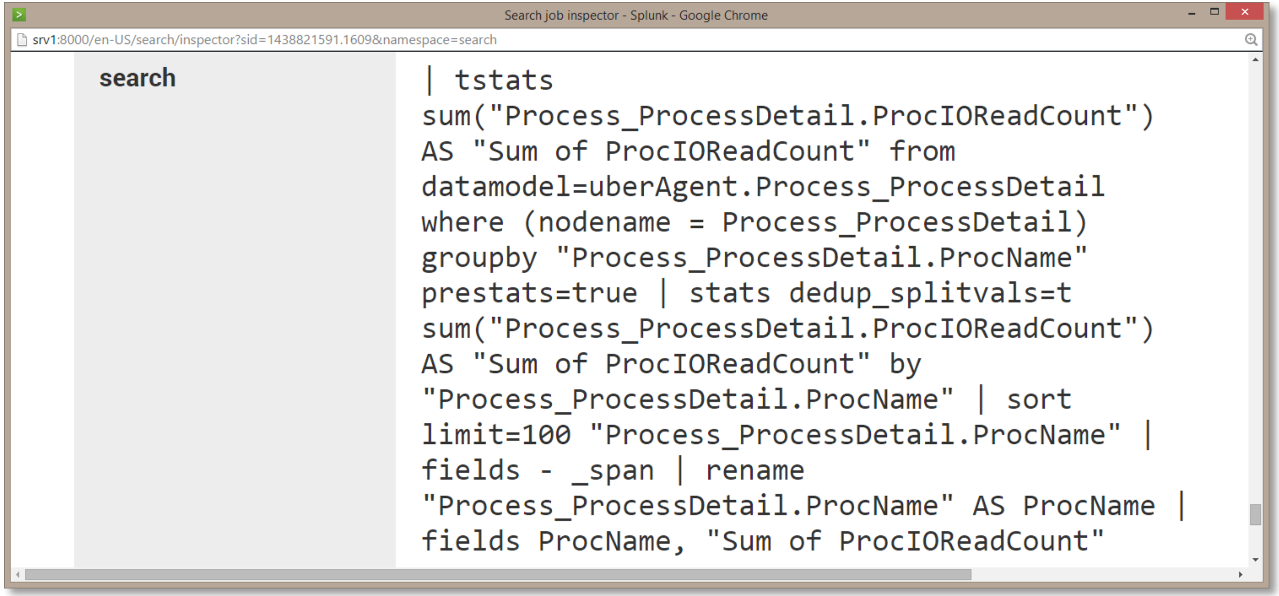
Split Columns

Column Values: Sum of ProcIORe...

| ProcName | Sum of ProcIOReadCount |
|---|---|
| POWERPNT.EXE | 231 |
| ONENOTE.EXE | 185 |
| chrome.exe | 126 |

# Learning Tstats

- Copy the underlying search


Search job inspector - Splunk - Google Chrome
srv1:8000/en-US/search/inspector?sid=1438821591.1609&namespace=search

search

```
| tstats
sum("Process_ProcessDetail.ProcIOReadCount")
AS "Sum of ProcIOReadCount" from
datamodel=uberAgent.Process_ProcessDetail
where (nodename = Process_ProcessDetail)
groupby "Process_ProcessDetail.ProcName"
prestats=true | stats dedup_splitvals=t
sum("Process_ProcessDetail.ProcIOReadCount")
AS "Sum of ProcIOReadCount" by
"Process_ProcessDetail.ProcName" | sort
limit=100 "Process_ProcessDetail.ProcName" |
fields - _span | rename
"Process_ProcessDetail.ProcName" AS ProcName |
fields ProcName, "Sum of ProcIOReadCount"
```

# Underlying Search

```
| tstats sum("Process_ProcessDetail.ProcIOReadCount")
  AS "Sum of ProcIOReadCount"
  from datamodel=uberAgent.Process_ProcessDetail
  where (nodename = Process_ProcessDetail)
  groupby "Process_ProcessDetail.ProcName" prestats=true
| stats dedup_splitvals=t
  sum("Process_ProcessDetail.ProcIOReadCount")
  AS "Sum of ProcIOReadCount"
  by "Process_ProcessDetail.ProcName"
| sort limit=100 "Process_ProcessDetail.ProcName"
| fields - _span
| rename "Process_ProcessDetail.ProcName" AS ProcName
| fields ProcName, "Sum of ProcIOReadCount"
```

# Let‘s Simplify

```
| tstats
  sum("Process_ProcessDetail.ProcIOReadCount")
  from datamodel=uberAgent.Process_ProcessDetail
  where (nodename = Process_ProcessDetail)
  groupby "Process_ProcessDetail.ProcName"
  prestats=true
| stats dedup_splitvals=t
  sum("Process_ProcessDetail.ProcIOReadCount")
  as "Sum of ProcIOReadCount"
  by "Process_ProcessDetail.ProcName"
```

# Walkthrough

```
| tstats
  sum("Process_ProcessDetail.ProcIOReadCount")
  from datamodel=uberAgent.Process_ProcessDetail
  where (nodename = Process_ProcessDetail)
          roces          tail.Pr
          rue
|         o_spl
  sum("Process_ProcessDetail.ProcIOReadCount")
  as "Sum of ProcIOReadCount"
  by "Process_ProcessDetail.ProcName"
```

Stats function

Data model object

Data model field

# Walkthrough

```
| tstats
  sum("Process_ProcessDetail.ProcIOReadCount")
  from datamodel=uberAgent.Process_ProcessDetail
  where (nodename = Process_ProcessDetail)
  groupby "Process_ProcessDetail.ProcName"
  prest  s=true
| sta        litvals=t
  su         rocessDetail.ProcIOReadCount")
  as         cIOReadCount"
  by "Process_ProcessDetail.ProcName"
```

Data model & object

# Walkthrough

```
| tstats
  sum("Process_ProcessDetail.ProcIOReadCount")
  from datamodel=uberAgent.Process_ProcessDetail
  where (nodename = Process_ProcessDetail)
  groupby "Process_ProcessDetail.ProcName"
  prestats=true
| stats dedup_splitvals=t
  sum("Process_ProcessDetail.Proc           ")
  as "Sum of ProcIOReadCount"
  by "Process_ProcessDetail.Proc
```

Field to group by

# Walkthrough

```
| tstats
  sum("Process_ProcessDetail.ProcIOReadCount")
  from datamodel=uberAgent.Process_ProcessDetail
  where (nodename = Process_ProcessDetail)
  groupby "Process_ProcessDetail.ProcName"
  prestats=true
| stats dedup_splitvals=t
  sum("      ocess_ProcessDetail.ProcIOReadCount")
  as             cIOReadCount"
  by             ocessDetail.ProcName"
```

Prestats mode

# Walkthrough

```
| tstats
  sum("Process_P                ocIOReadCount")
  from datamodel                ess_ProcessDetail
  where (nodena                 cessDetail)
  groupby "P     ce             .ProcName"
  prestats=true
| stats dedup_splitvals=t
  sum("Process_ProcessDetail.ProcIOReadCount")
  as "Sum of ProcIOReadCount"
  by "Process_ProcessDetail.ProcName"
```

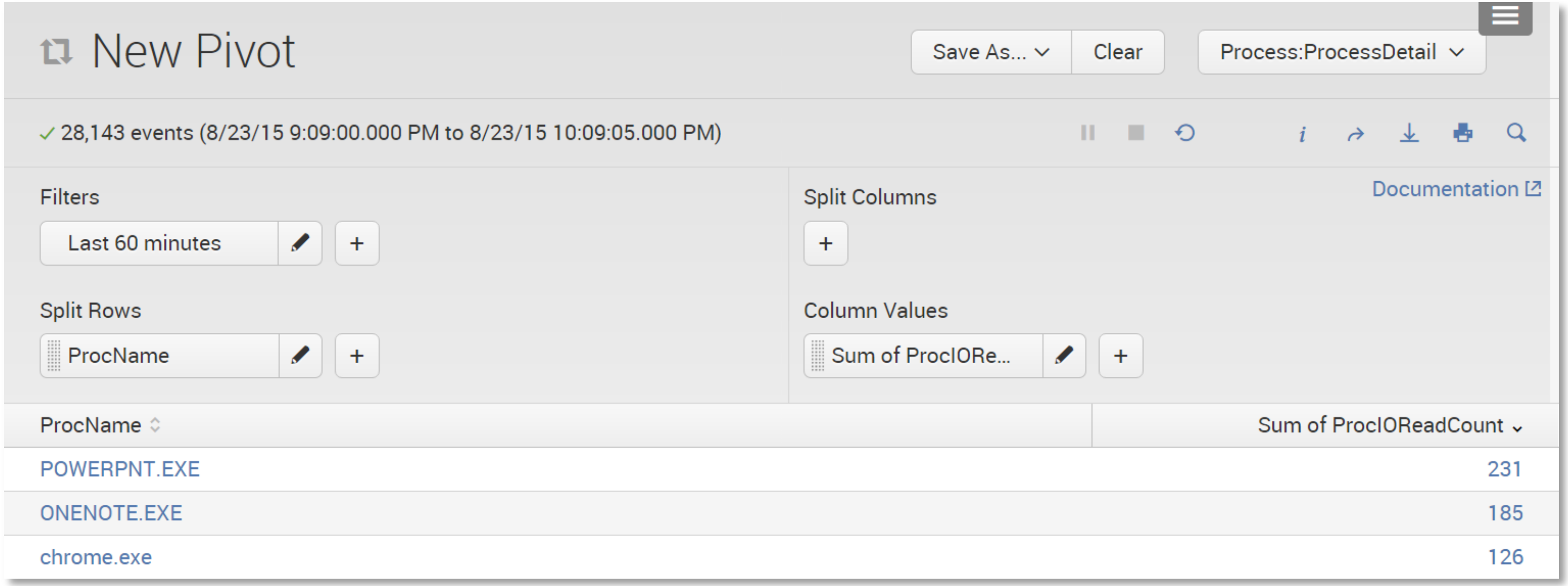
Stats command mirrors earlier tstats command

# Pivot: The Principle

- „Different“ syntax
- Only searches data models
- Must be the 1st command in the search pipline

# Learning Pivot

- Build a sample search in Pivot Editor



✓ 28,143 events (8/23/15 9:09:00.000 PM to 8/23/15 10:09:05.000 PM)

Filters: Last 60 minutes

Split Rows: ProcName

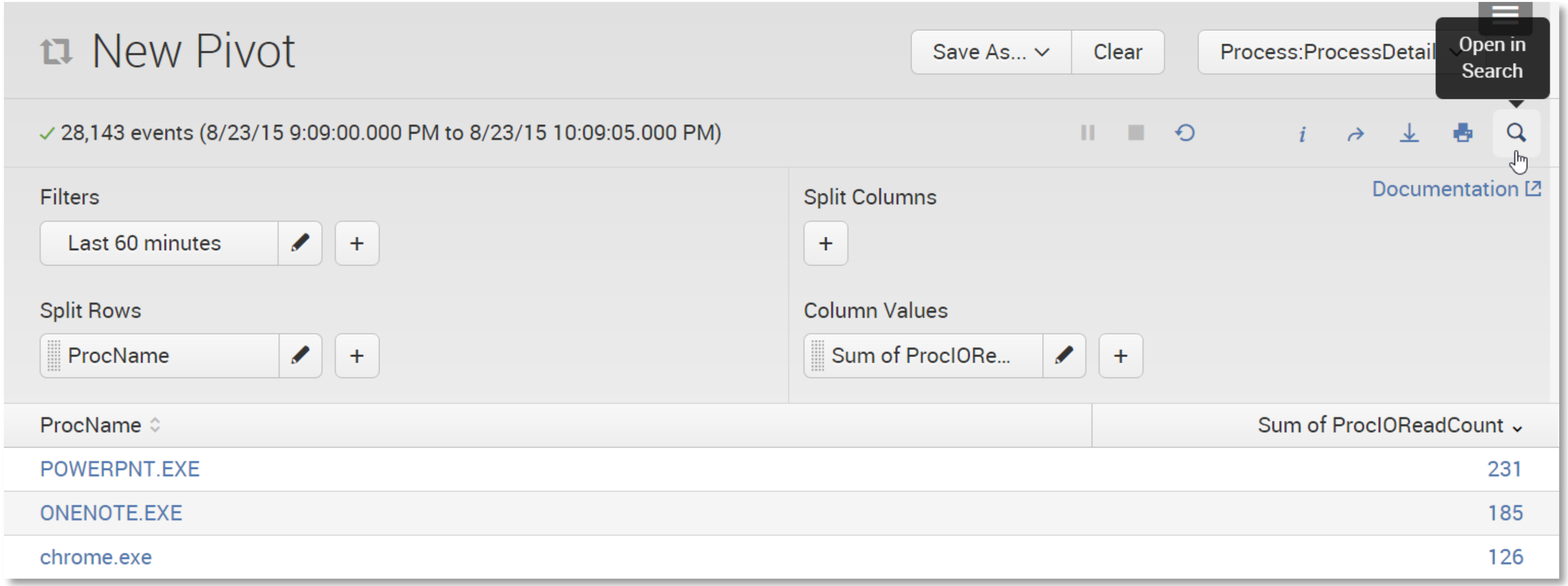
Split Columns

Column Values: Sum of ProcIORe...

| ProcName | Sum of ProcIOReadCount |
|---|---|
| POWERPNT.EXE | 231 |
| ONENOTE.EXE | 185 |
| chrome.exe | 126 |

# Learning Pivot

- Open in Search



✓ 28,143 events (8/23/15 9:09:00.000 PM to 8/23/15 10:09:05.000 PM)

Filters: Last 60 minutes

Split Columns

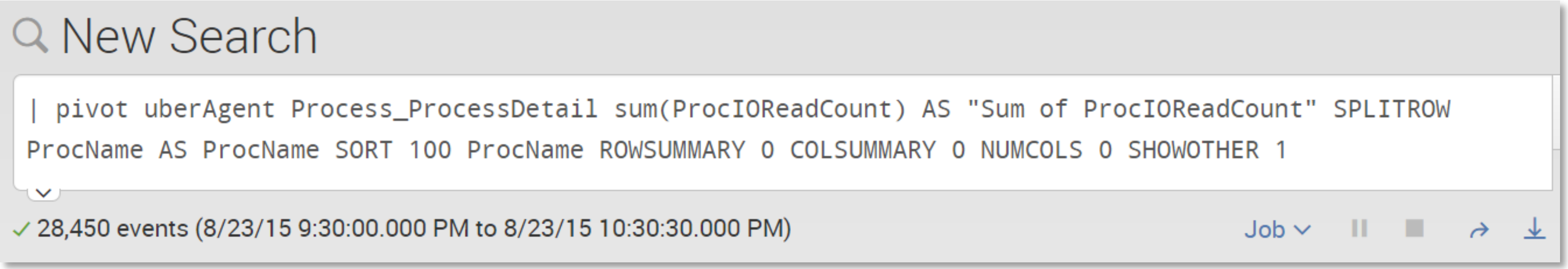
Split Rows: ProcName

Column Values: Sum of ProcIORe...

| ProcName | Sum of ProcIOReadCount |
|---|---|
| POWERPNT.EXE | 231 |
| ONENOTE.EXE | 185 |
| chrome.exe | 126 |

# Learning Pivot

- Copy the underlying search



```
| pivot uberAgent Process_ProcessDetail sum(ProcIOReadCount) AS "Sum of ProcIOReadCount" SPLITROW
ProcName AS ProcName SORT 100 ProcName ROWSUMMARY 0 COLSUMMARY 0 NUMCOLS 0 SHOWOTHER 1
```

✓ 28,450 events (8/23/15 9:30:00.000 PM to 8/23/15 10:30:30.000 PM)

# Underlying Search

```
| pivot uberAgent Process_ProcessDetail
  sum(ProcIOReadCount) as "Sum of ProcIOReadCount"
  splitrow ProcName as ProcName
  sort 100 ProcName
  rowsummary 0
  colsummary 0
  numcols 0
  showother 1
```

# Let‘s Simplify

```
| pivot uberAgent Process_ProcessDetail
  sum(ProcIOReadCount) as "Sum of ProcIOReadCount"
  splitrow ProcName as ProcName
```

# Walkthrough

```
| pivot uberAgent Process_ProcessDetail
  sum(ProcIOReadCount) as "Sum of ProcIOReadCount"
  splitrow ProcName as ProcName
```

Data model name

Data model object

# Walkthrough

```
| pivot uberAgent Process_ProcessDetail
  sum(ProcIOReadCount) as "Sum of ProcIOReadCount"
  splitrow ProcName as ProcName
```

Stats function

Data model field

# Walkthrough

```
| pivot uberAgent Process_ProcessDetail
  sum(ProcIOReadCount) as "Sum of ProcIOReadCount"
  splitrow ProcName as ProcName
```

Field to group by

# Wrap Up

# Wrap Up

- This talk covered **persistent** data model acceleration
- There is also **ad hoc** data model acceleration
  - Applied only in the Pivot UI
  - Automatically enabled
  - Takes place on the search head
  - Summaries are deleted when the Pivot Editor is left

# Thank you!

Enjoy the conference!

# Resources

- Design data models and objects

  http://docs.splunk.com/Documentation/Splunk/latest/Knowledge/Designdatamodelobjects

- Manage data models

  http://docs.splunk.com/Documentation/Splunk/latest/Knowledge/Managedatamodels

- Accelerate data models

  http://docs.splunk.com/Documentation/Splunk/latest/Knowledge/Acceleratedatamodels