Enriching Your Data Using The Latest Features Of DB Connect

Jack Coates

Splunk

.conf2016

splunk>

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.

Use Cases For Structured Data In Splunk

Index machine data from databases, such as logs or sales records

Enrich machine data with high-level data, such as customer records

Update structured databases with Splunk info, such as risk scores

Interactively browse structured and unstructured data from Splunk reports

What’s Happened Since Last Year?

Performance Improvements

User Experience Improvements

Database Improvements

Output Improvements

WHAT'S NEW, DBX TWO?

Performance!

- Input system redesigned
- Lookup system redesigned
- Java caching redesigned
- Front end tuned
- Scale and performance test automation
  - Documented scaling expectations
  - JVM fixes
  - FetchSize tuned

User Experience!

- SQL Editor improvements
  - Automatic vs Editor modes
- Advanced mode inputs
- Time selection improvements
- Save As Search
- dbxquery options
  - Max rows hardcode removed in favor of **maxrows** option (default 100,000)
  - **output** JSON or CSV

Databases!

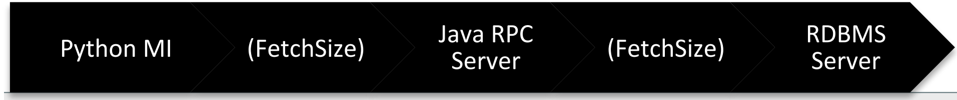
- Integrated Authentication to Microsoft SQL Server from Linux using Kerberos
- SparkSQL support
- Easier Oracle SSL
  - 2 way or 1 way
- AWS RDS RedShift
- AWS RDS Aurora
- MemSQL 5
- Also documented "SQL tips for Splunkers"

Administration!

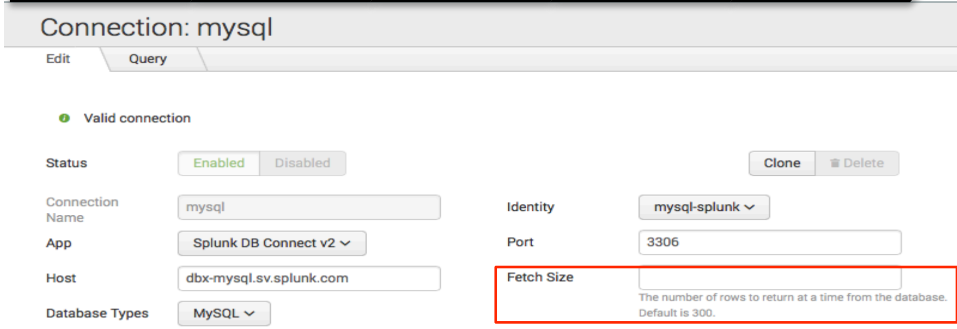
- Ability to reload RPC server and refresh JDBC drivers without restarting Splunk
- Logging improvements
- UPSERT support
  - If row exists, UPDATE; else INSERT
- Custom commands: dbxlookup, dbxoutput
- Modular alert support

Control Over FetchSize

- Say you're asking for "TOP 1000" rows in your query
- The JDBC client and server negotiate their own transference... Which affects the whole streaming pipeline
- Control this with the FetchSize parameter
- Default was 10, is now database-specific
  - Oracle == 300
  - MySQL-ish == -1 (streams instead of chunks)
- Too big is bad (OOM Errors!!)
- Too small is bad (Super Slow!!)
- DBA will want to discuss



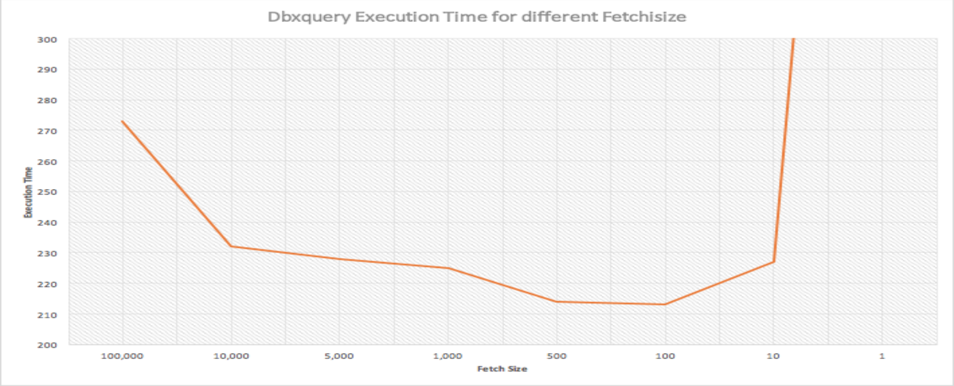
Python MI → (FetchSize) → Java RPC Server → (FetchSize) → RDBMS Server



Connection: mysql

● Valid connection

Status	Enabled / Disabled		
Connection Name	mysql	Identity	mysql-splunk
App	Splunk DB Connect v2	Port	3306
Host	dbx-mysql.sv.splunk.com	Fetch Size	
Database Types	MySQL		The number of rows to return at a time from the database. Default is 300.



Microsoft SQL Servers

- 2 platforms, 2 drivers, 3 auth methods = 18 potential scenarios

DBX: OS, driver	**SQL Server Account (dbuser)**	**Domain Account – Mixed Mode (AD\user)** ***Must use "Domain" field in Identity***	**Integrated Auth – Kerberos (AD\user)** ***Must use "Domain" field in Identity***
Windows, MS	✔ "MS SQL Server using MS Generic Driver"	✔ "MS SQL Server using MS Generic Driver with Windows Authentication"	✔ "MS SQL Server using MS Generic Driver with Windows Authentication" *Must run Splunk service as the domain user*
Windows, jTDS	✔ "MS SQL Server using jTDS Driver"	✔ "MS SQL Server using jTDS Driver with Windows Authentication"	✖
Linux, MS	✔ "MS SQL Server using MS Generic Driver"	✖	✔ "MS SQL Server using MS Generic Driver with Kerberos Authentication"
Linux, jTDS	✔ "MS SQL Server using jTDS Driver"	✔ "MS SQL Server using jTDS Driver with Windows Authentication"	✖

What The Heck Is Query Wrapping?

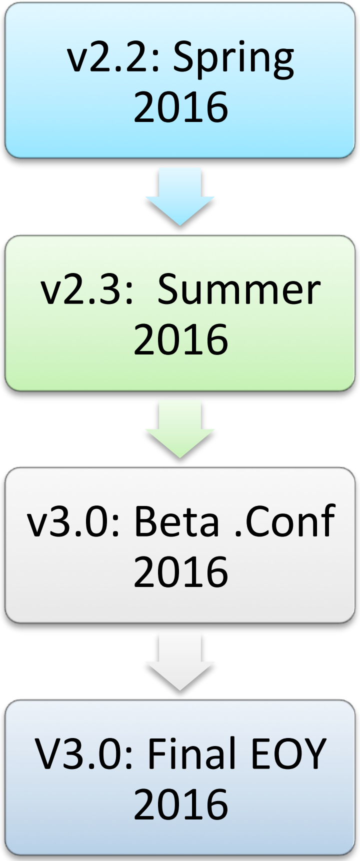
- Inline views are a handy way to handle ambiguous column names, variables, and column renames
  - Turned on by default in 2.1.0 to resolve problems with rising column use
  - Introduced performance problems, syntax errors in some SQL
- 2.2.0 adds connection-level control
  - Add `disable_query_wrapping=1` to the db_connections.conf entry
- 2.3.0 adds input and command level control (wrap=[bool])
  - Use Advanced editor to build the input without any DBX meddling
  - Add `wrap=false` to `dbxquery` command

ROADMAP

Iteration across the year's major goals:

- DB Connect for Data Exploration
  - DBXQuery improvements (2.2)
  - Exploring Schema-based data (2.3, 3.0)
- DB Connect for the Cloud
  - Support for Cloud databases (2.2, 2.3, 3.0)
  - Support for the Splunk Cloud Service (2.2, 2.3, 3.0)
- DB Connect for Modern Data
  - Support for NoSQL databases* (2.3, 3.0)
- Maintenance
  - End of Life DBX 1 (3.0)

[Spark, Hive, Cassandra]



v2.2: Spring 2016

v2.3: Summer 2016

v3.0: Beta .Conf 2016

V3.0: Final EOY 2016

https://confluence.splunk.com/display/PROD/PRD+DB+Connect+FY17

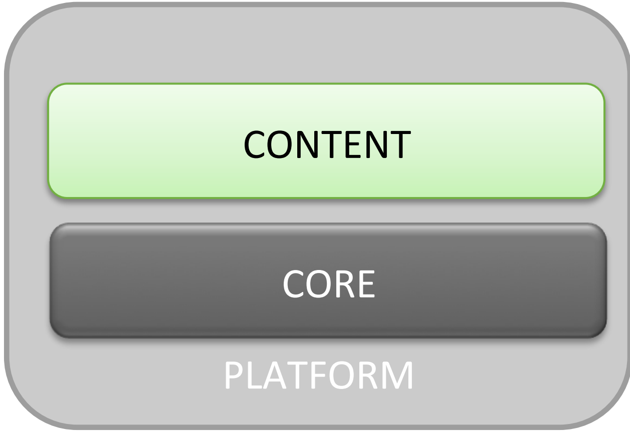
DB Connect In Splunk Context

blah blah blah blah blah **PLATFORM** *blah blah blah blah blah*

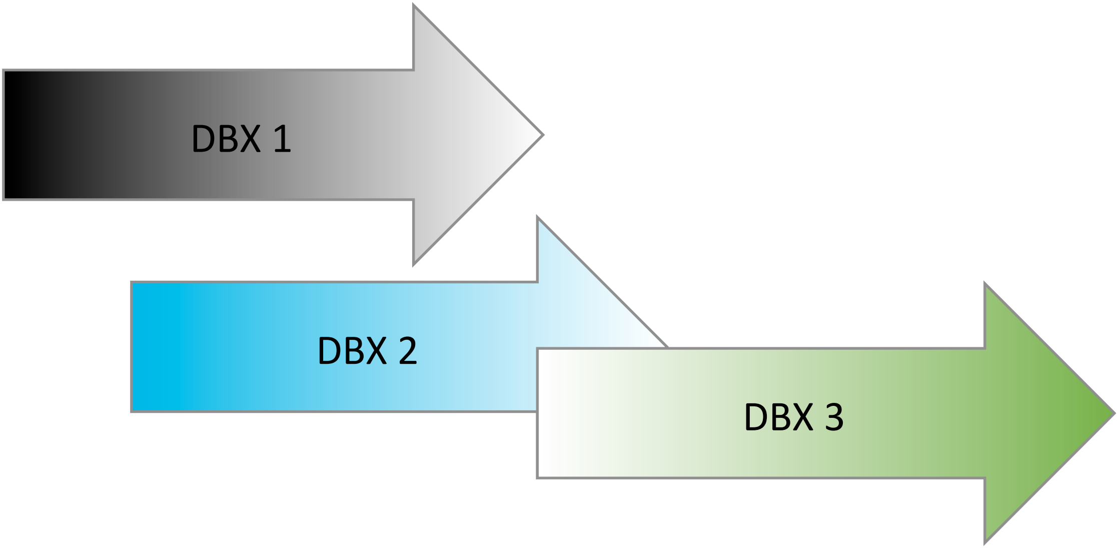
- Data wranglers have enough to focus on without Splunk getting in the way
  - More data transformation: make it easy to use SQL and SPL together
  - More data collection: make it easy to ingest data
- Platform goes beyond “Apps & Splunk” or “Content and Core”
- More Entry and Exit points for partners and customers
  - Open interfaces and tools between each functional layer of Splunk
  - Components in platform, not features in monolith
  - Mo’ DevOps - Better packaging, dependency management, SOA

Platform Functionalities

- Platform isn't about Core and Content, it's about shipped capabilities
- Functionalities are critical to Platform, regardless of how they ship
  - Capability expansion via custom commands & modular inputs
    - Human-speed, high-iteration analysis
  - Scalable, fault tolerant Service Oriented Architecture
    - Data Solutions Group to enable new data, new product
    - Add-on Builder to enable more partner use
  - Semantic abstraction between data and use case (CIM)
    - Wrangling & ETL of semi-structured data (UI improvements)



DBX Generational Movements



- DBX 2.3.0 eliminates most reasons to stay on DBX1
- DBX 1.x EOL announced with DBX 2.2 & implemented with DBX 2.3

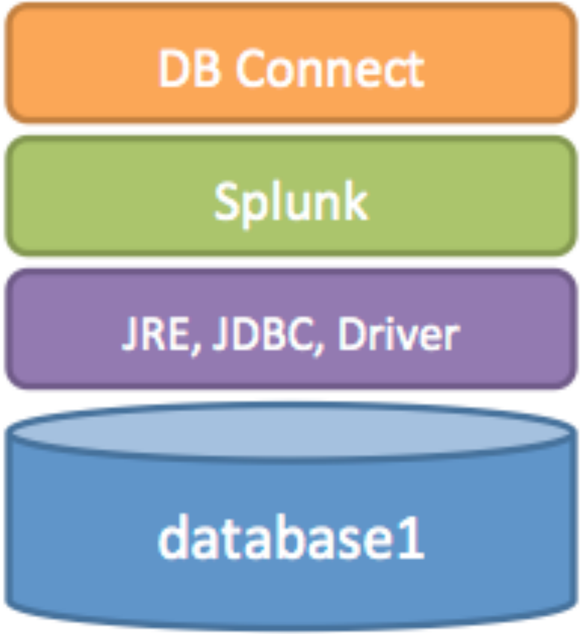
- Expect DBX 3 to extend DBX 2's place instead of being a new Splunkbase entry
- We'll stop supporting DBX 2 more gradually instead of doing a hard cliff like DBX 1

THANK YOU

.conf2016

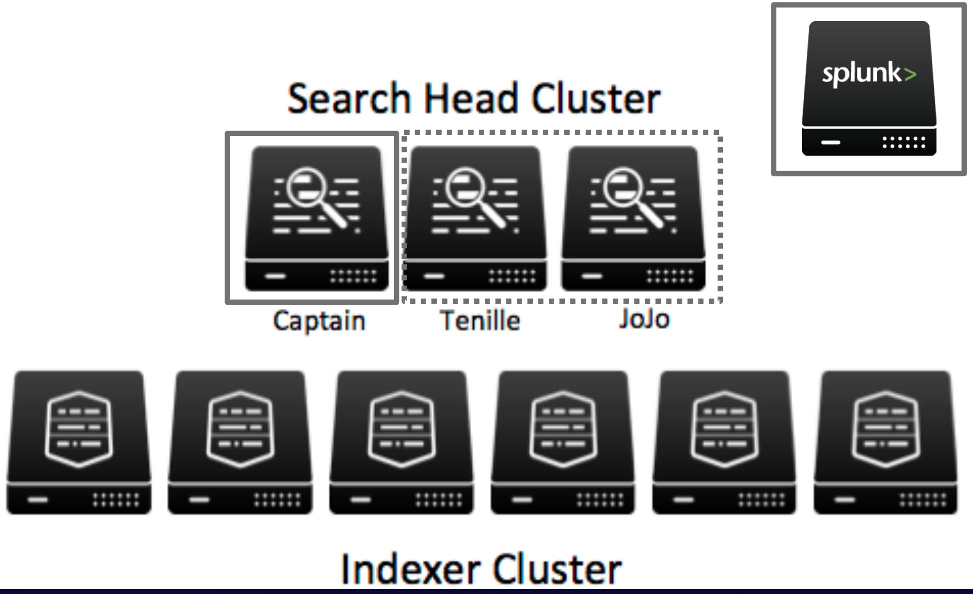
Architecture Overview

- RPC Server is the modular input interface between Splunk and Java. It runs all the time
- JDBC is the mechanism used
- Java is required
- ODBC is not currently planned
- Cross-platform use is supported



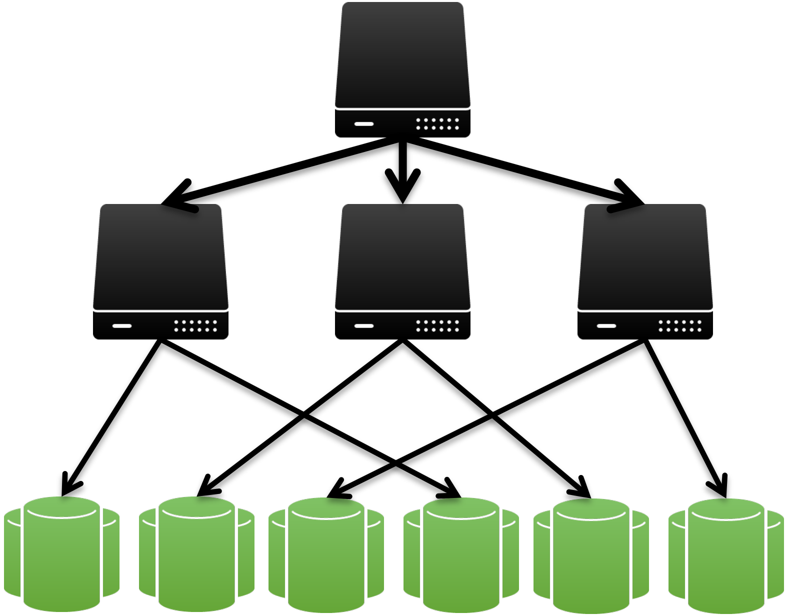
Clustering Architecture

- Installs to Search Heads or Stand-Alones
- In a SH Cluster, only the captain will run DB Connect – the others are idle
- Note that captain re-election make take some minutes, during which time DB Connect is not running.



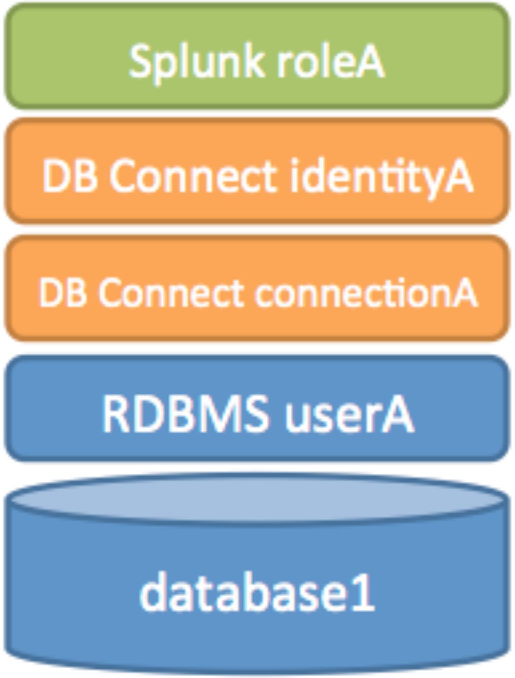
Resource Pool Overview

- Install DB Connect on a master and N resource pool nodes
- Jobs from the master will dispatch in round robin to resource pool members
- Resource pool nodes must actively receive jobs, so no dead zones
- Master does not monitor job progress



Connection Overview

- Map Splunk users to database users
- Use roles and identities to manage role based access controls
- Map identities to connections
- Look out for the read-only JDBC option, it doesn't do what you might think



Configuration File Format Changes

1.x.x

$SPLUNK_HOME/etc/apps/**dbx**/README/*.spec

- database.conf
- database_types.conf
- dblookup.conf
- inputs.conf
- java.conf

2.x.x

$SPLUNK_HOME/etc/apps/**splunk_app_db_connect**/README/*.spec

- db_connections.conf
- db_connection_types.conf
- healthlog.conf
- identities.conf
- inputs.conf

Database Connections (1 of 2)

1.x.x

database_types.conf

- Lists the supported database types, driver parameters, test queries

database.conf

- All configuration necessary for connecting to a specific database

2.x.x

db_connection_types.conf

- Lists the supported database types, driver parameters, test queries

db_connections.conf

- All configuration necessary for connecting to a specific database, *unless overridden by parameters from identities.conf*

identities.conf

- Username and password used to connect to the database (stored in standard Splunk credential store, but in a DBX-specific method)

Database Connections (2 of 2)

1.x.x

inputs.conf

- Configures database indexing scripted input behavior (tail, dump, batch, change)

dblookup.conf

- Configures database-backed lookups

java.conf

- Sets Java location and options globally

2.0.0

inputs.conf

- Configures database indexing, lookup, and output behavior
- Modular inputs are used for all three types of operations
- Java options can be set per input now

healthlog.conf

- Manage the behavior of DB Connect's self-monitoring dashboard

No More Output Format Templates

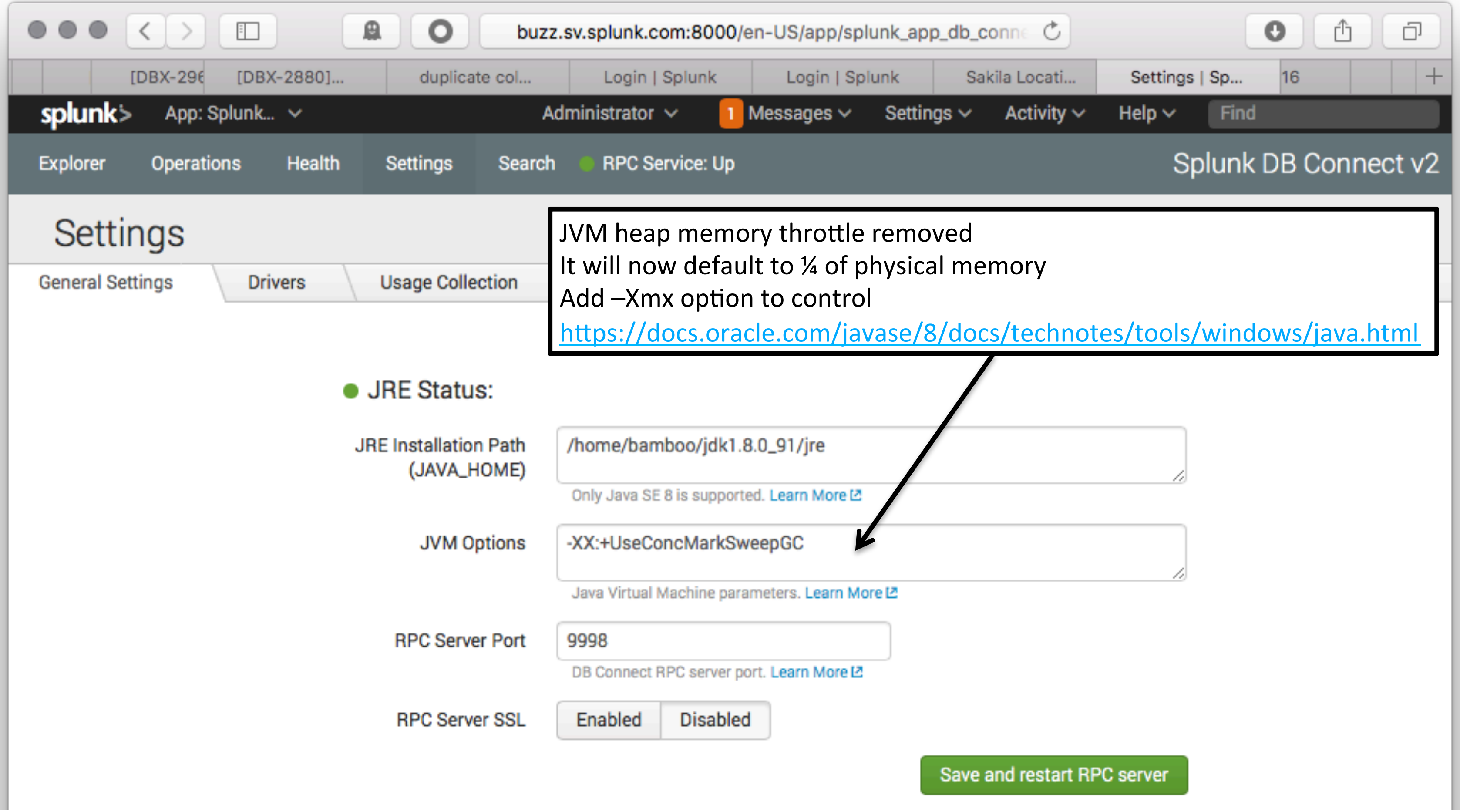
- In DBX 1, you could set a output format template to select behavior:
  - http://docs.splunk.com/Documentation/DBX/1.1.6/DeployDBX/Configuredatabasemonitoring#Configure_database_output
- In DBX 2, you just use search commands to format.
  - Want key-value? Use `eval`.
  - Want to change quoting pattern? Use `rex`.

WITH DBX 2.2+, USE CSV OUTPUT

SCREENSHOT TOUR Of New Features

.conf2016

splunk>



Settings

General Settings | Drivers | Usage Collection

JVM heap memory throttle removed
It will now default to ¼ of physical memory
Add –Xmx option to control
https://docs.oracle.com/javase/8/docs/technotes/tools/windows/java.html

● JRE Status:

JRE Installation Path (JAVA_HOME): /home/bamboo/jdk1.8.0_91/jre

Only Java SE 8 is supported. Learn More

JVM Options: -XX:+UseConcMarkSweepGC

Java Virtual Machine parameters. Learn More

RPC Server Port: 9998

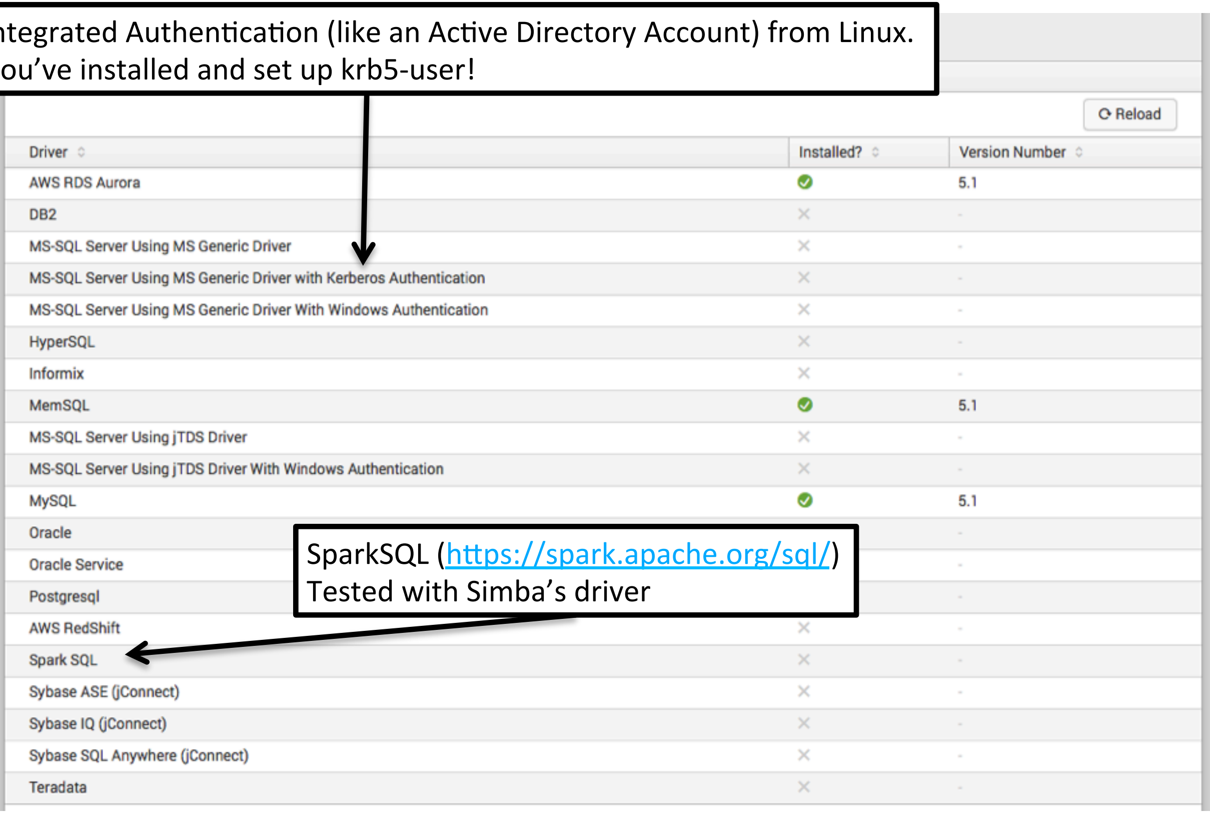
DB Connect RPC server port. Learn More

RPC Server SSL: Enabled | Disabled

Save and restart RPC server

New Connection Types

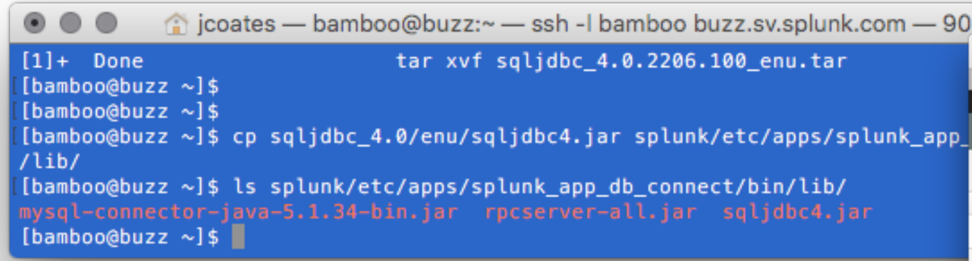
MS-SQL with Integrated Authentication (like an Active Directory Account) from Linux. This assumes you've installed and set up krb5-user!



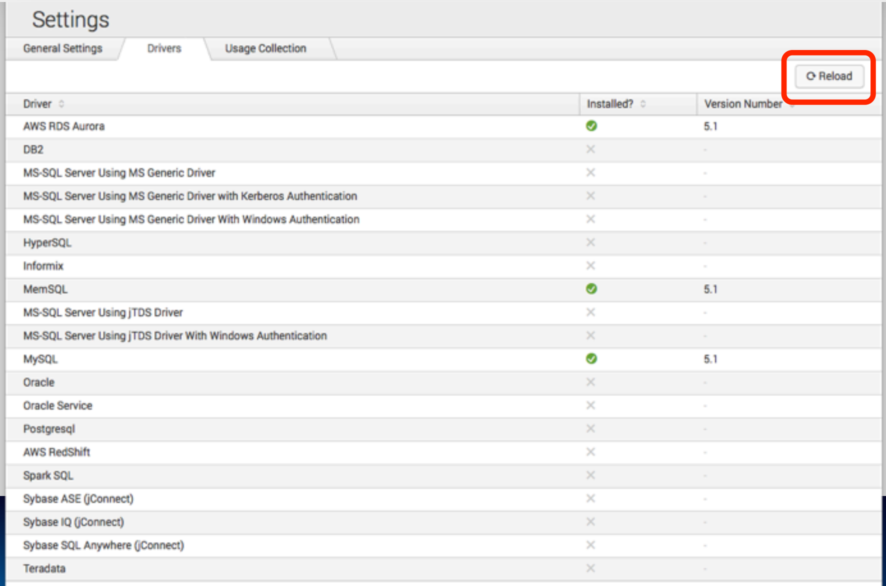
Driver	Installed?	Version Number
AWS RDS Aurora	✓	5.1
DB2	✗	-
MS-SQL Server Using MS Generic Driver	✗	-
MS-SQL Server Using MS Generic Driver with Kerberos Authentication	✗	-
MS-SQL Server Using MS Generic Driver With Windows Authentication	✗	-
HyperSQL	✗	-
Informix	✗	-
MemSQL	✓	5.1
MS-SQL Server Using jTDS Driver	✗	-
MS-SQL Server Using jTDS Driver With Windows Authentication	✗	-
MySQL	✓	5.1
Oracle		-
Oracle Service		-
Postgresql		-
AWS RedShift	✗	-
Spark SQL	✗	-
Sybase ASE (jConnect)	✗	-
Sybase IQ (jConnect)	✗	-
Sybase SQL Anywhere (jConnect)	✗	-
Teradata	✗	-

SparkSQL (https://spark.apache.org/sql/)
Tested with Simba's driver

Reload Drivers Without Splunk Restart

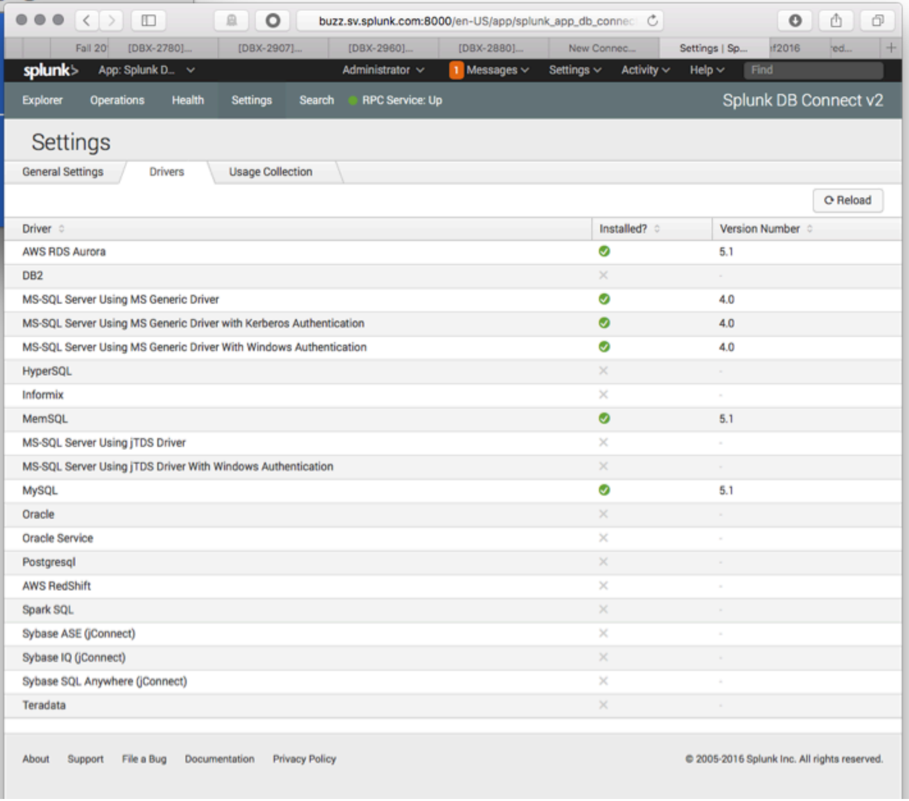

jcoates — bamboo@buzz:~ — ssh -l bamboo buzz.sv.splunk.com — 90×8

```
[1]+  Done                    tar xvf sqljdbc_4.0.2206.100_enu.tar
[bamboo@buzz ~]$
[bamboo@buzz ~]$
[bamboo@buzz ~]$ cp sqljdbc_4.0/enu/sqljdbc4.jar splunk/etc/apps/splunk_app_
/lib/
[bamboo@buzz ~]$ ls splunk/etc/apps/splunk_app_db_connect/bin/lib/
mysql-connector-java-5.1.34-bin.jar  rpcserver-all.jar  sqljdbc4.jar
[bamboo@buzz ~]$
```



Settings

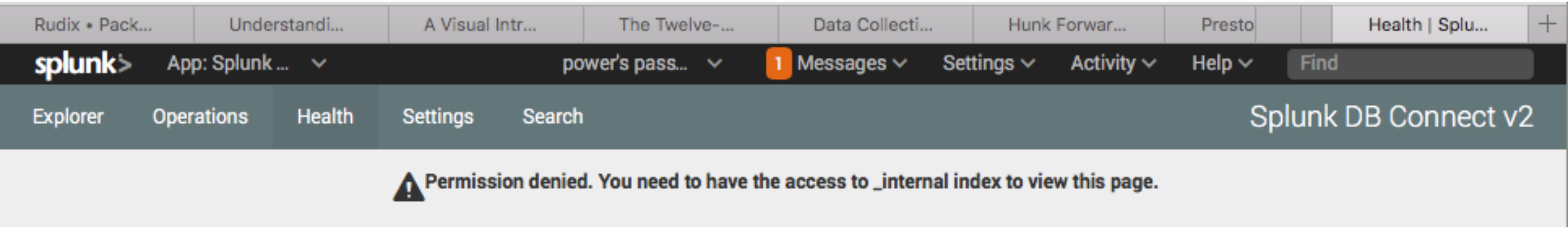
Driver	Installed?	Version Number
AWS RDS Aurora	✓	5.1
DB2	✗	-
MS-SQL Server Using MS Generic Driver	✗	-
MS-SQL Server Using MS Generic Driver with Kerberos Authentication	✗	-
MS-SQL Server Using MS Generic Driver With Windows Authentication	✗	-
HyperSQL	✗	-
Informix	✗	-
MemSQL	✓	5.1
MS-SQL Server Using jTDS Driver	✗	-
MS-SQL Server Using jTDS Driver With Windows Authentication	✗	-
MySQL	✓	5.1
Oracle	✗	-
Oracle Service	✗	-
Postgresql	✗	-
AWS RedShift	✗	-
Spark SQL	✗	-
Sybase ASE (jConnect)	✗	-
Sybase IQ (jConnect)	✗	-
Sybase SQL Anywhere (jConnect)	✗	-
Teradata	✗	-



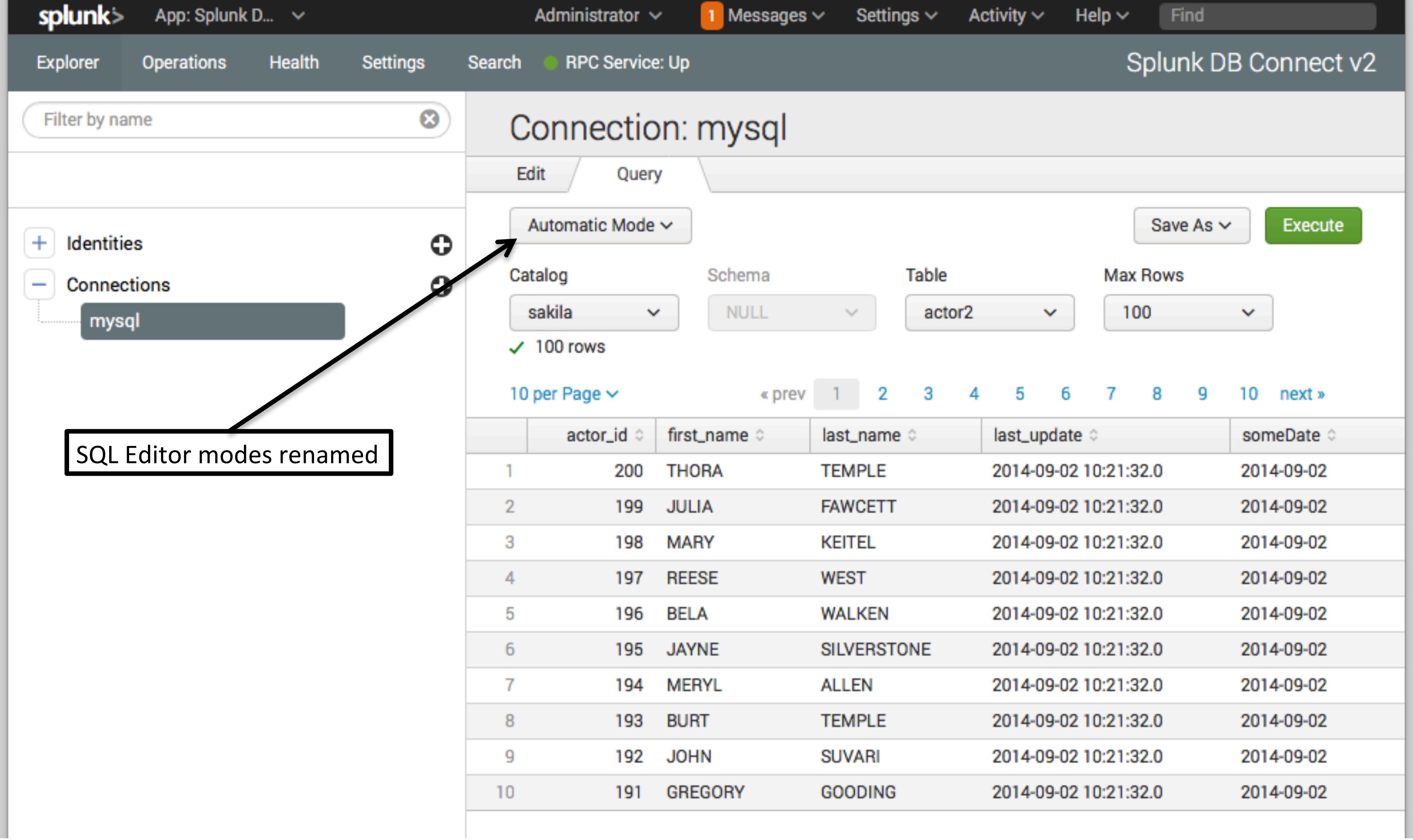
Settings

Driver	Installed?	Version Number
AWS RDS Aurora	✓	5.1
DB2	✗	-
MS-SQL Server Using MS Generic Driver	✓	4.0
MS-SQL Server Using MS Generic Driver with Kerberos Authentication	✓	4.0
MS-SQL Server Using MS Generic Driver With Windows Authentication	✓	4.0
HyperSQL	✗	-
Informix	✗	-
MemSQL	✓	5.1
MS-SQL Server Using jTDS Driver	✗	-
MS-SQL Server Using jTDS Driver With Windows Authentication	✗	-
MySQL	✓	5.1
Oracle	✗	-
Oracle Service	✗	-
Postgresql	✗	-
AWS RedShift	✗	-
Spark SQL	✗	-
Sybase ASE (jConnect)	✗	-
Sybase IQ (jConnect)	✗	-
Sybase SQL Anywhere (jConnect)	✗	-
Teradata	✗	-

Health Dashboard



⚠ Permission denied. You need to have the access to _internal index to view this page.



Connection: mysql

Catalog: sakila

Schema: NULL

Table: actor2

Max Rows: 100

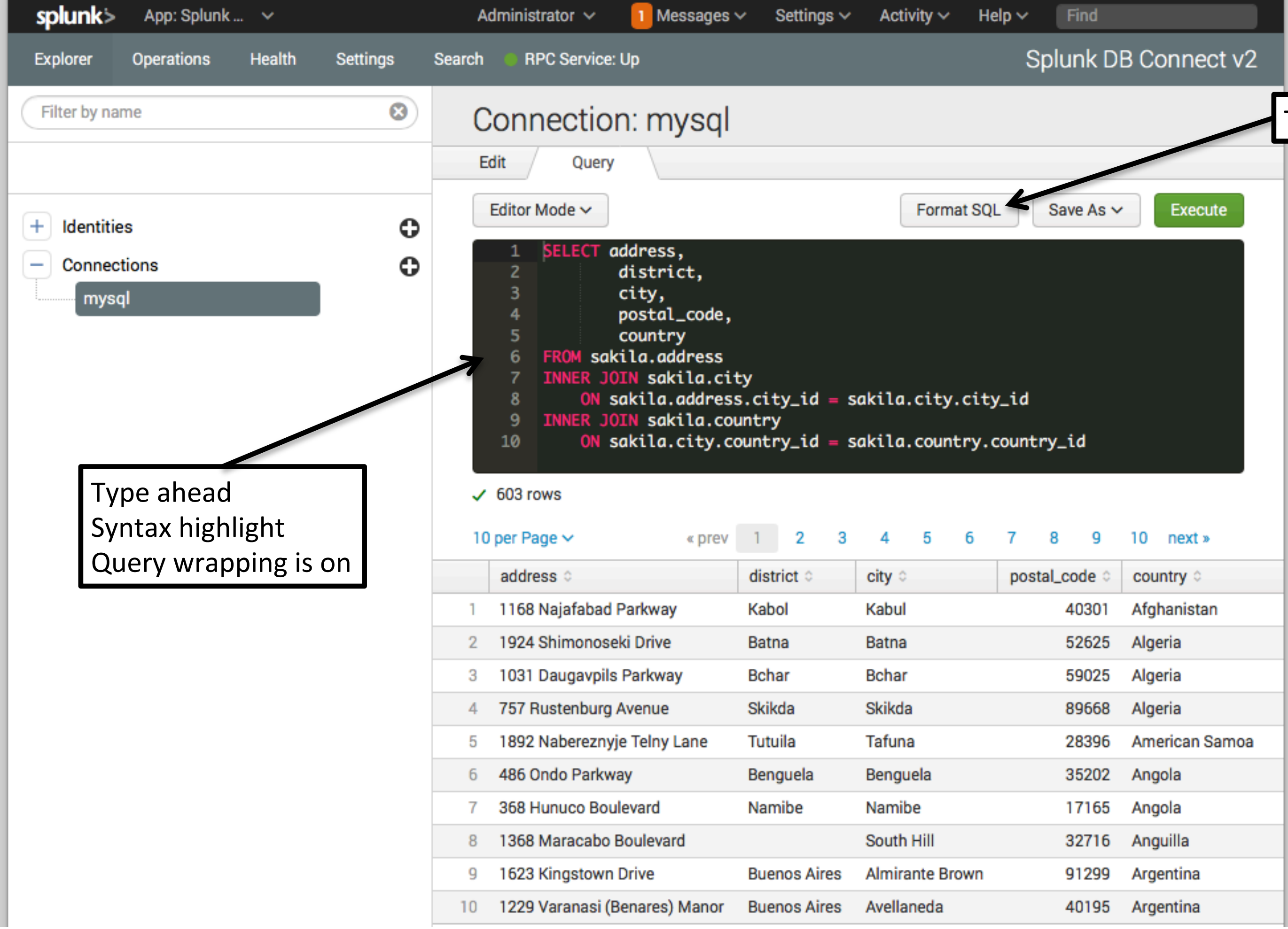
✓ 100 rows

10 per Page

« prev 1 2 3 4 5 6 7 8 9 10 next »

	actor_id	first_name	last_name	last_update	someDate
1	200	THORA	TEMPLE	2014-09-02 10:21:32.0	2014-09-02
2	199	JULIA	FAWCETT	2014-09-02 10:21:32.0	2014-09-02
3	198	MARY	KEITEL	2014-09-02 10:21:32.0	2014-09-02
4	197	REESE	WEST	2014-09-02 10:21:32.0	2014-09-02
5	196	BELA	WALKEN	2014-09-02 10:21:32.0	2014-09-02
6	195	JAYNE	SILVERSTONE	2014-09-02 10:21:32.0	2014-09-02
7	194	MERYL	ALLEN	2014-09-02 10:21:32.0	2014-09-02
8	193	BURT	TEMPLE	2014-09-02 10:21:32.0	2014-09-02
9	192	JOHN	SUVARI	2014-09-02 10:21:32.0	2014-09-02
10	191	GREGORY	GOODING	2014-09-02 10:21:32.0	2014-09-02

SQL Editor modes renamed



Connection: mysql

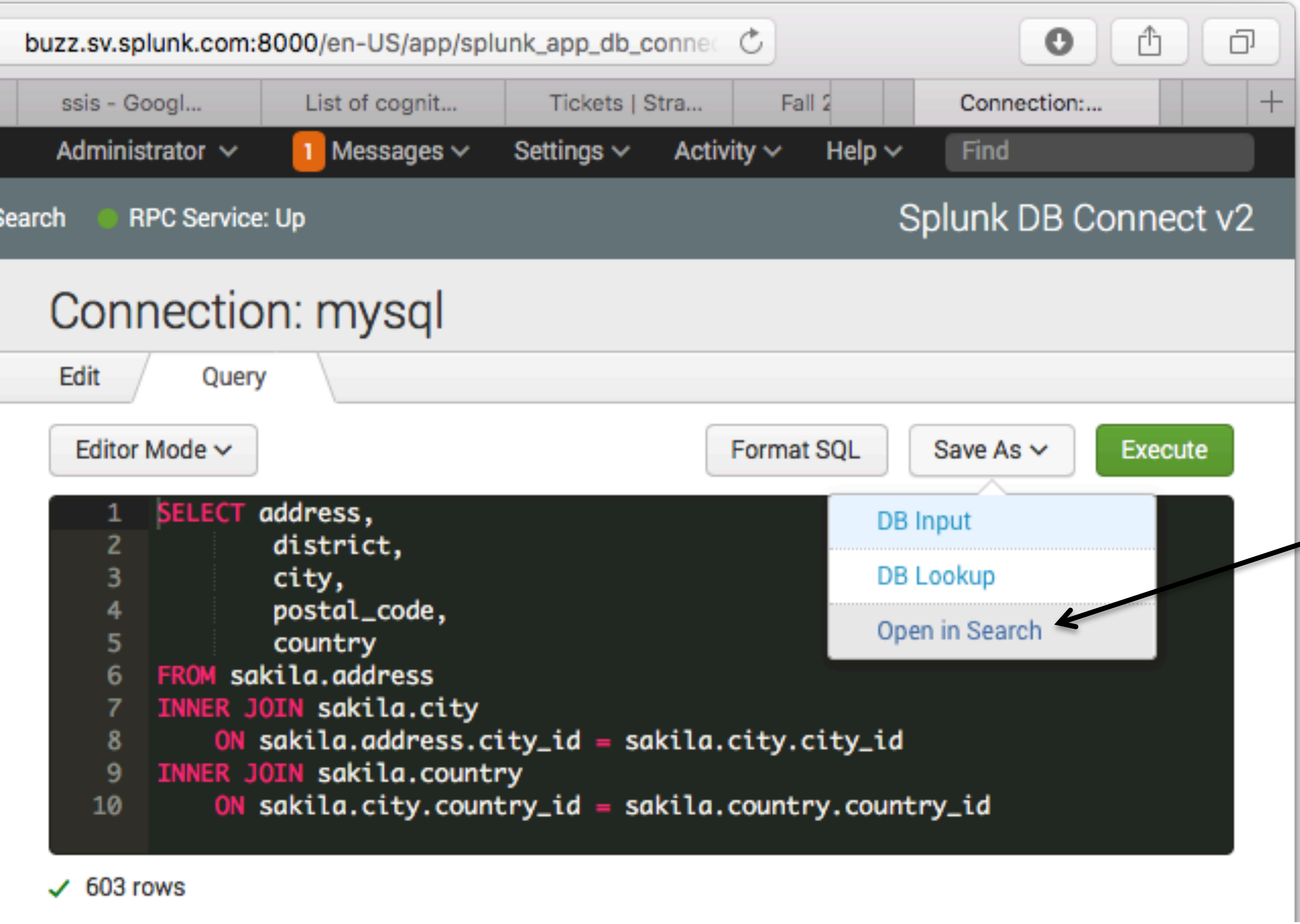
```
SELECT address,
       district,
       city,
       postal_code,
       country
FROM sakila.address
INNER JOIN sakila.city
  ON sakila.address.city_id = sakila.city.city_id
INNER JOIN sakila.country
  ON sakila.city.country_id = sakila.country.country_id
```

✓ 603 rows

	address	district	city	postal_code	country
1	1168 Najafabad Parkway	Kabol	Kabul	40301	Afghanistan
2	1924 Shimonoseki Drive	Batna	Batna	52625	Algeria
3	1031 Daugavpils Parkway	Bchar	Bchar	59025	Algeria
4	757 Rustenburg Avenue	Skikda	Skikda	89668	Algeria
5	1892 Nabereznyje Telny Lane	Tutuila	Tafuna	28396	American Samoa
6	486 Ondo Parkway	Benguela	Benguela	35202	Angola
7	368 Hunuco Boulevard	Namibe	Namibe	17165	Angola
8	1368 Maracabo Boulevard		South Hill	32716	Anguilla
9	1623 Kingstown Drive	Buenos Aires	Almirante Brown	91299	Argentina
10	1229 Varanasi (Benares) Manor	Buenos Aires	Avellaneda	40195	Argentina

That's not reversible

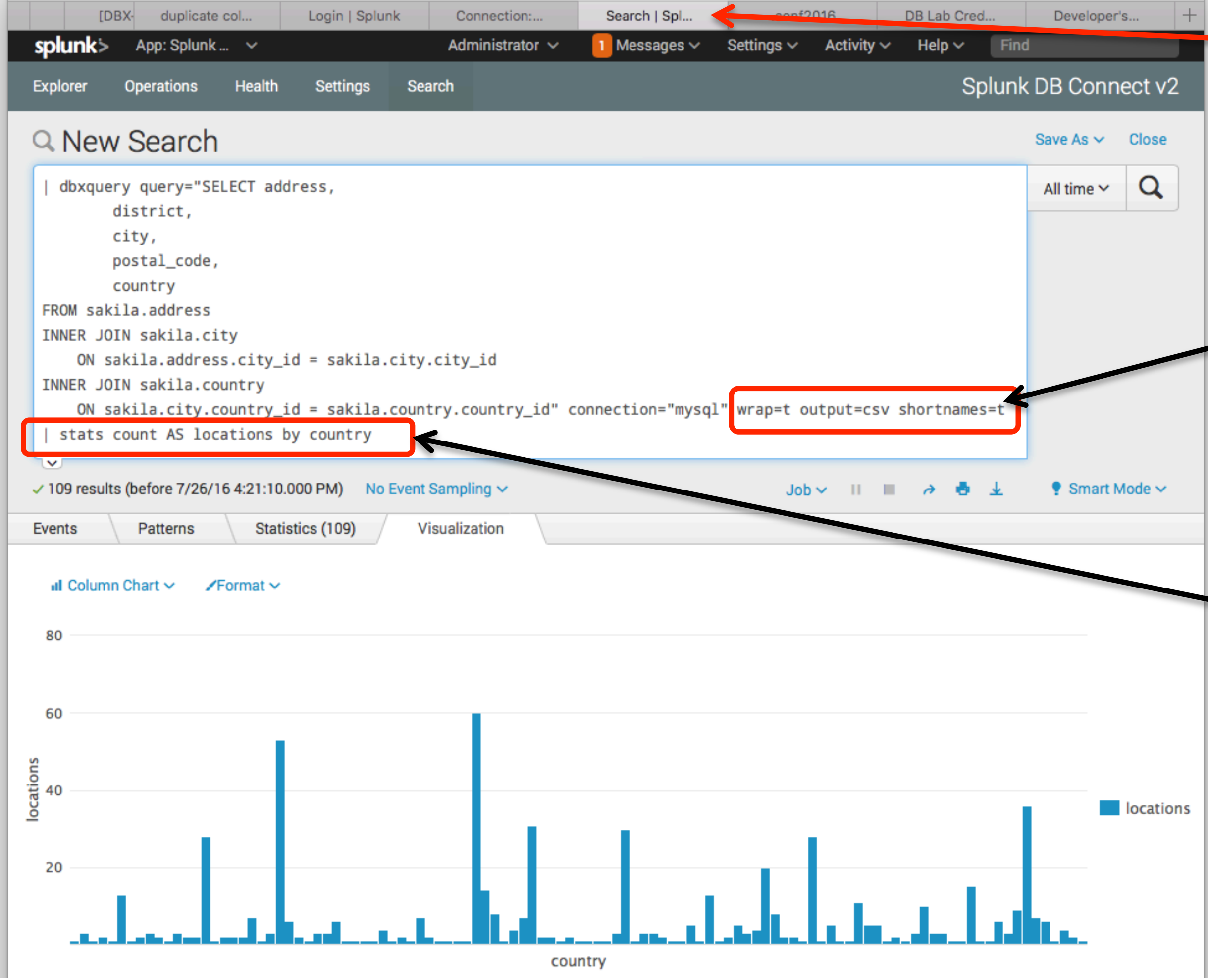
Type ahead
Syntax highlight
Query wrapping is on



```sql
SELECT address,
        district,
        city,
        postal_code,
        country
FROM sakila.address
INNER JOIN sakila.city
    ON sakila.address.city_id = sakila.city.city_id
INNER JOIN sakila.country
    ON sakila.city.country_id = sakila.country.country_id
```

✓ 603 rows

Let’s go make a dashboard panel



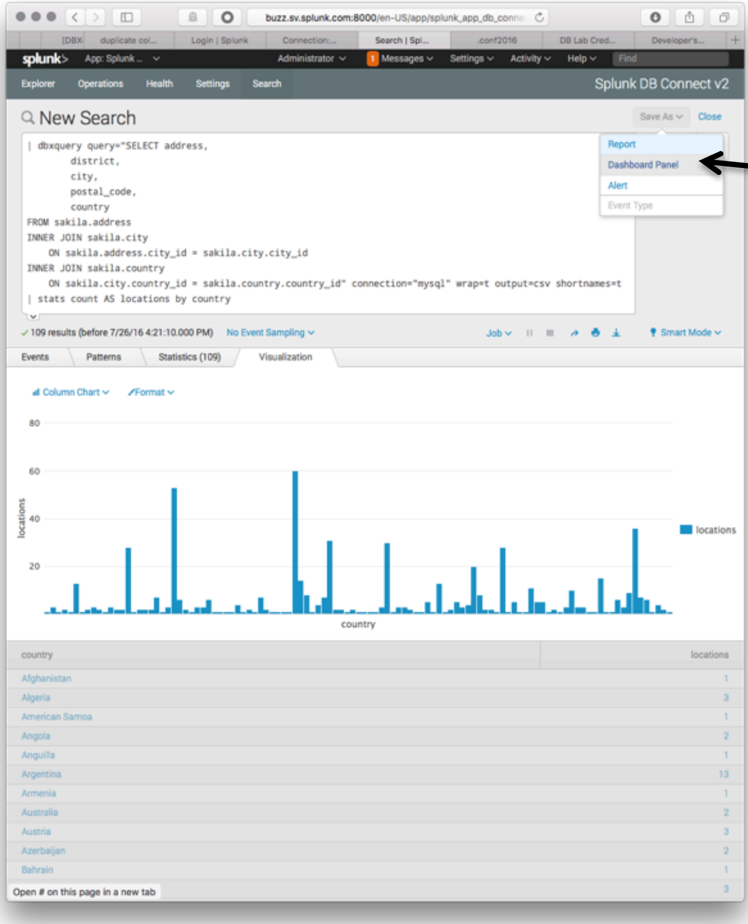
```
| dbxquery query="SELECT address,
        district,
        city,
        postal_code,
        country
FROM sakila.address
INNER JOIN sakila.city
    ON sakila.address.city_id = sakila.city.city_id
INNER JOIN sakila.country
    ON sakila.city.country_id = sakila.country.country_id" connection="mysql" wrap=t output=csv shortnames=t
| stats count AS locations by country
```

✓ 109 results (before 7/26/16 4:21:10.000 PM)   No Event Sampling

Opens in new tab

Options for dbxquery note “wrap=[t|f]”

Pipe to SPL and keep on truckin’

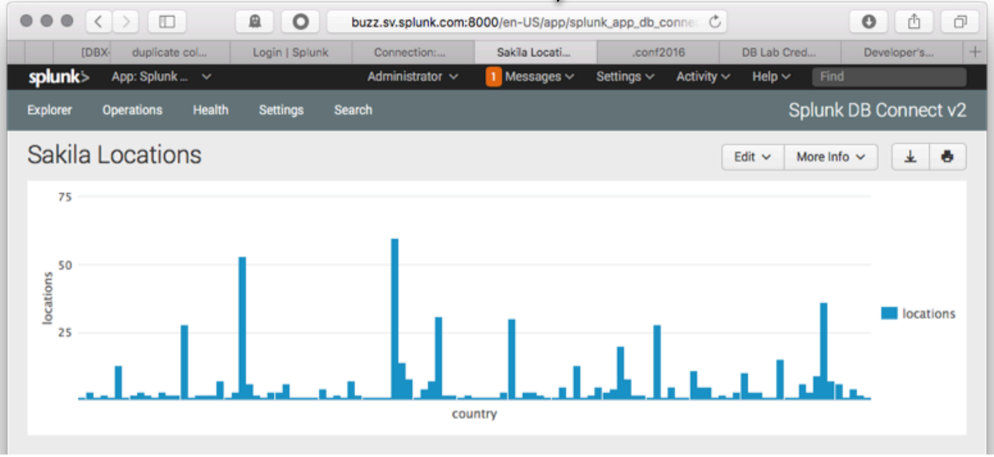


```
| dbxquery query="SELECT address,
        district,
        city,
        postal_code,
        country
FROM sakila.address
INNER JOIN sakila.city
    ON sakila.address.city_id = sakila.city.city_id
INNER JOIN sakila.country
    ON sakila.city.country_id = sakila.country.country_id" connection="mysql" wrap=t output=csv shortnames=t
| stats count AS locations by country
```

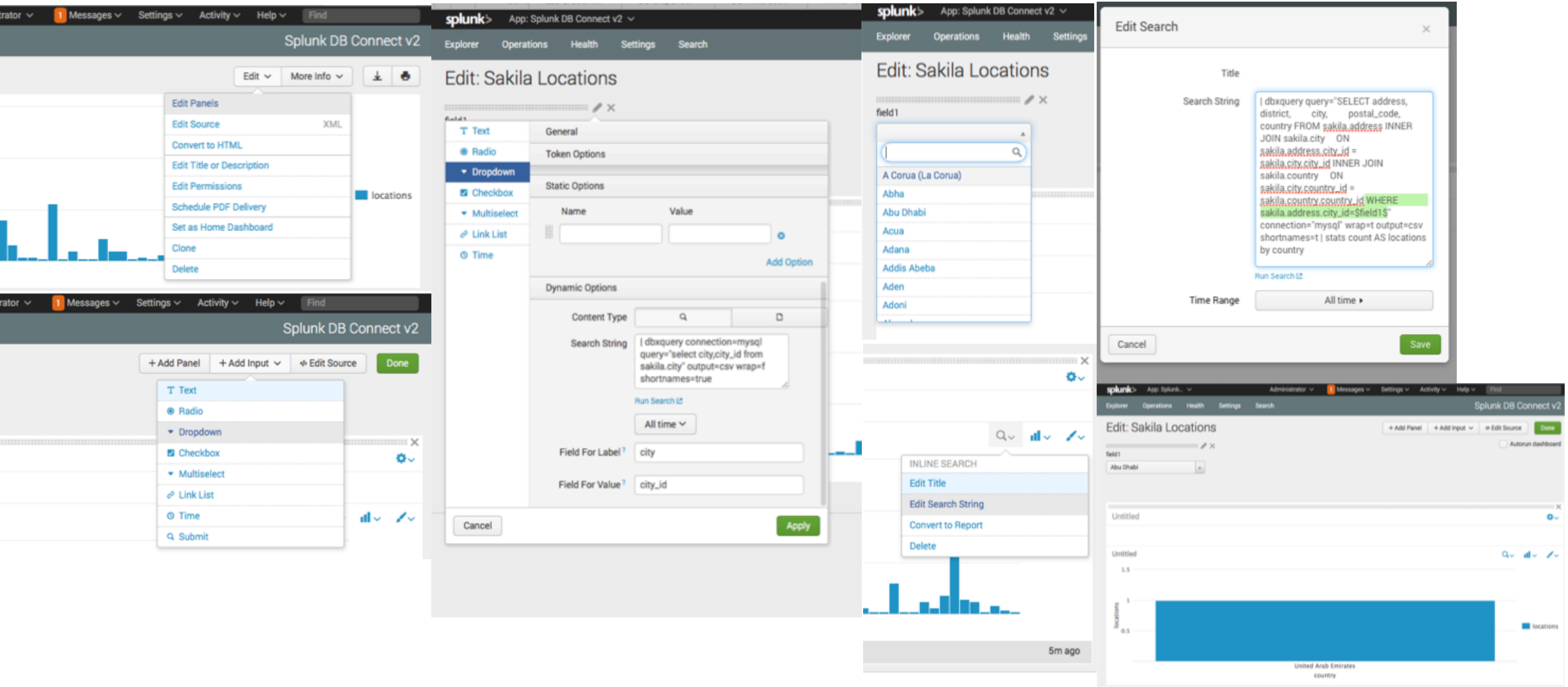
✓ 109 results (before 7/26/16 4:21:10.000 PM)

country	locations
Afghanistan	1
Algeria	3
American Samoa	1
Angola	2
Anguilla	1
Argentina	13
Armenia	1
Australia	2
Austria	3
Azerbaijan	2
Bahrain	1
Open # on this page in a new tab	3

Save as panel, boo yah



Hey, You Can Use Variables In There Too



Content Type

Search String:

```
| dbxquery connection=mysql query="select city,city_id from sakila.city" output=csv wrap=f shortnames=true
```

All time

Field For Label: city

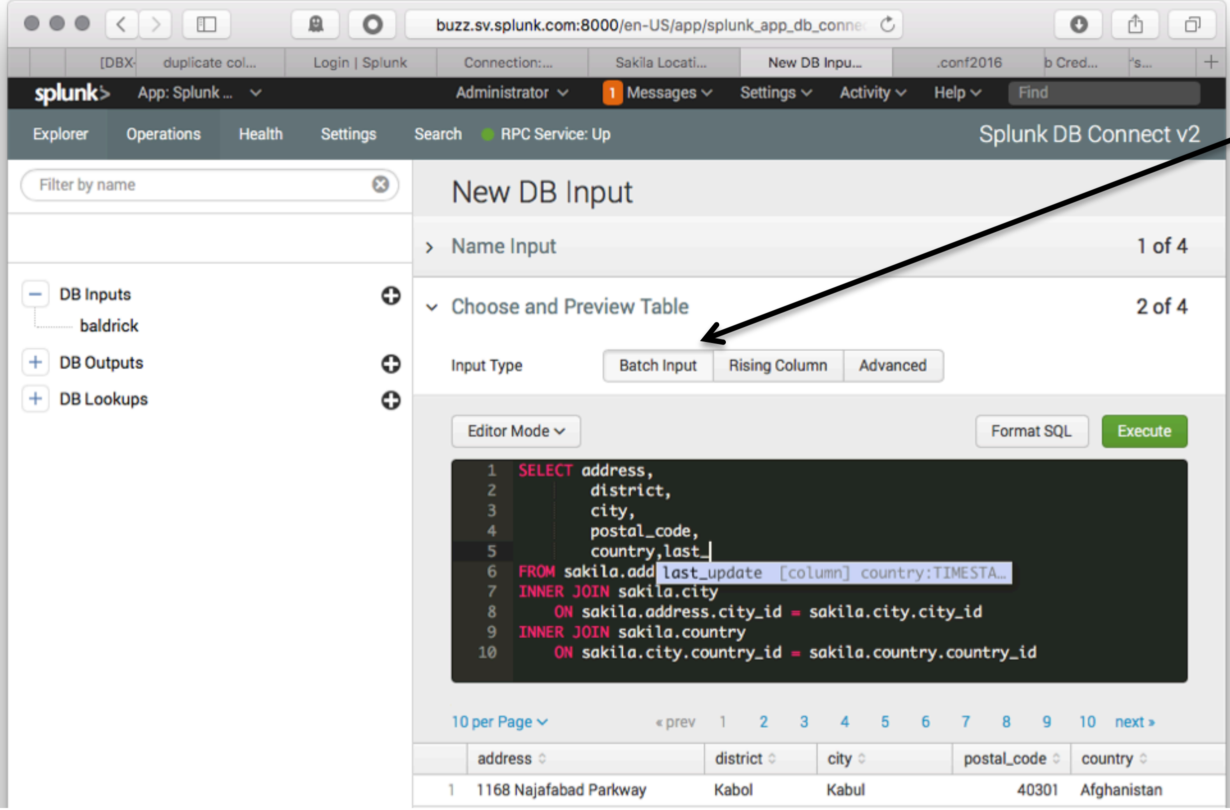
Field For Value: city_id

Search String:

```
| dbxquery query="SELECT address, district,      city,      postal_code, country FROM sakila.address INNER JOIN sakila.city    ON sakila.address.city_id = sakila.city.city_id INNER JOIN sakila.country    ON sakila.city.country_id = sakila.country.country_id WHERE sakila.address.city_id=$field1$" connection="mysql" wrap=t output=csv shortnames=t | stats count AS locations by country
```

Time Range: All time

DB Input Types



New DB Input

Name Input 1 of 4

Choose and Preview Table 2 of 4

Input Type: Batch Input | Rising Column | Advanced

```
SELECT address,
       district,
       city,
       postal_code,
       country,last_
FROM sakila.add
INNER JOIN sakila.city
   ON sakila.address.city_id = sakila.city.city_id
INNER JOIN sakila.country
   ON sakila.city.country_id = sakila.country.country_id
```

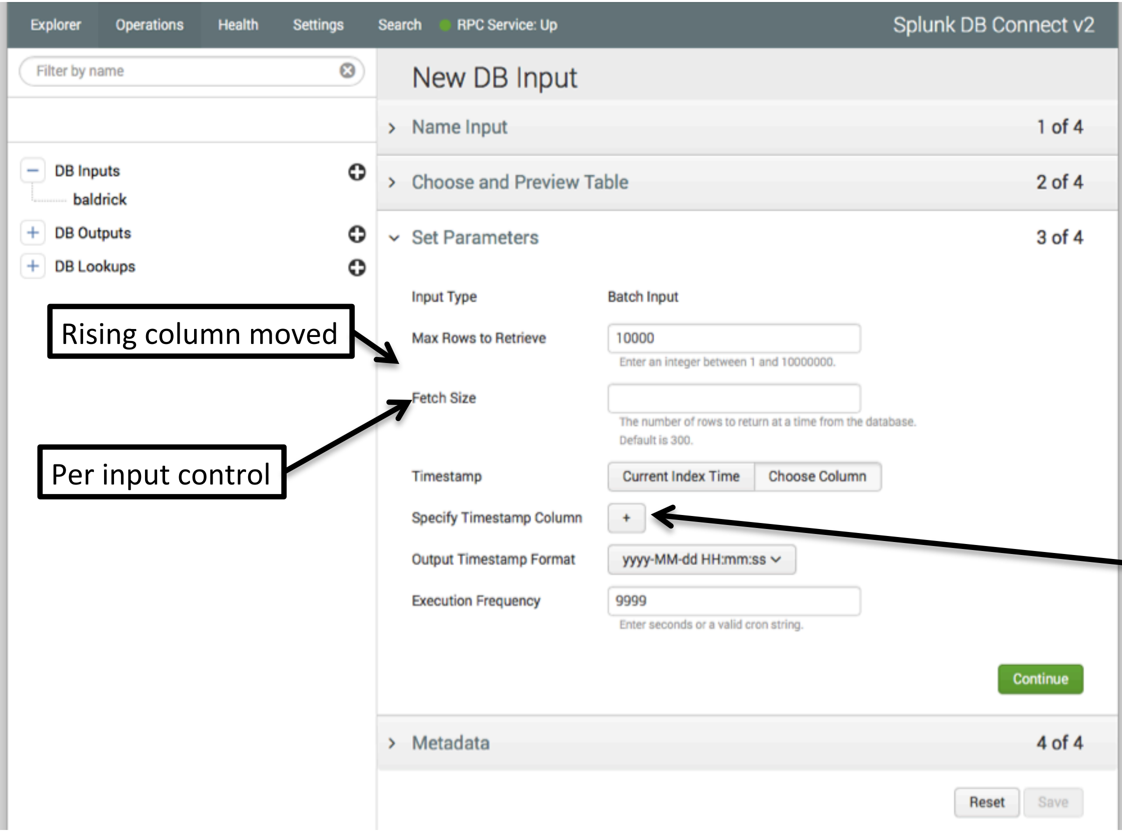
	address	district	city	postal_code	country
1	1168 Najafabad Parkway	Kabol	Kabul	40301	Afghanistan

Batch: allows automatic or editor, wrap=t

Rising: allows automatic or editor, wrap=t, Adds order by and helps you do rising col.

Advanced: only allows editor, wrap=f, you have to do your own rising col.

DB Input Times



New DB Input

Name Input 1 of 4

Choose and Preview Table 2 of 4

Set Parameters 3 of 4

Input Type: Batch Input

Max Rows to Retrieve: 10000

Enter an integer between 1 and 10000000.

Fetch Size:

The number of rows to return at a time from the database. Default is 300.

Timestamp: Current Index Time | Choose Column

Specify Timestamp Column: +

Output Timestamp Format: yyyy-MM-dd HH:mm:ss

Execution Frequency: 9999

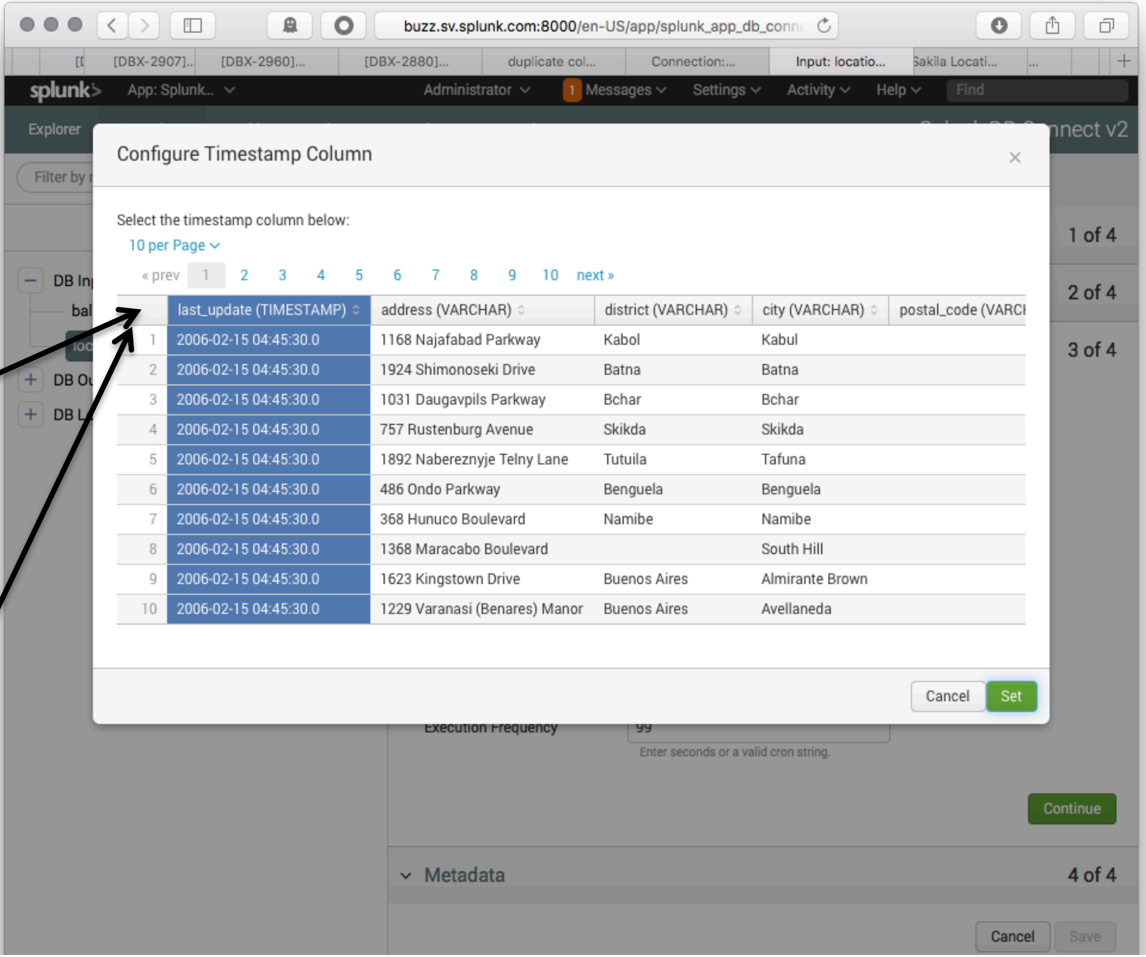
Enter seconds or a valid cron string.

Metadata 4 of 4

Rising column moved

Per input control

When you pick a column…



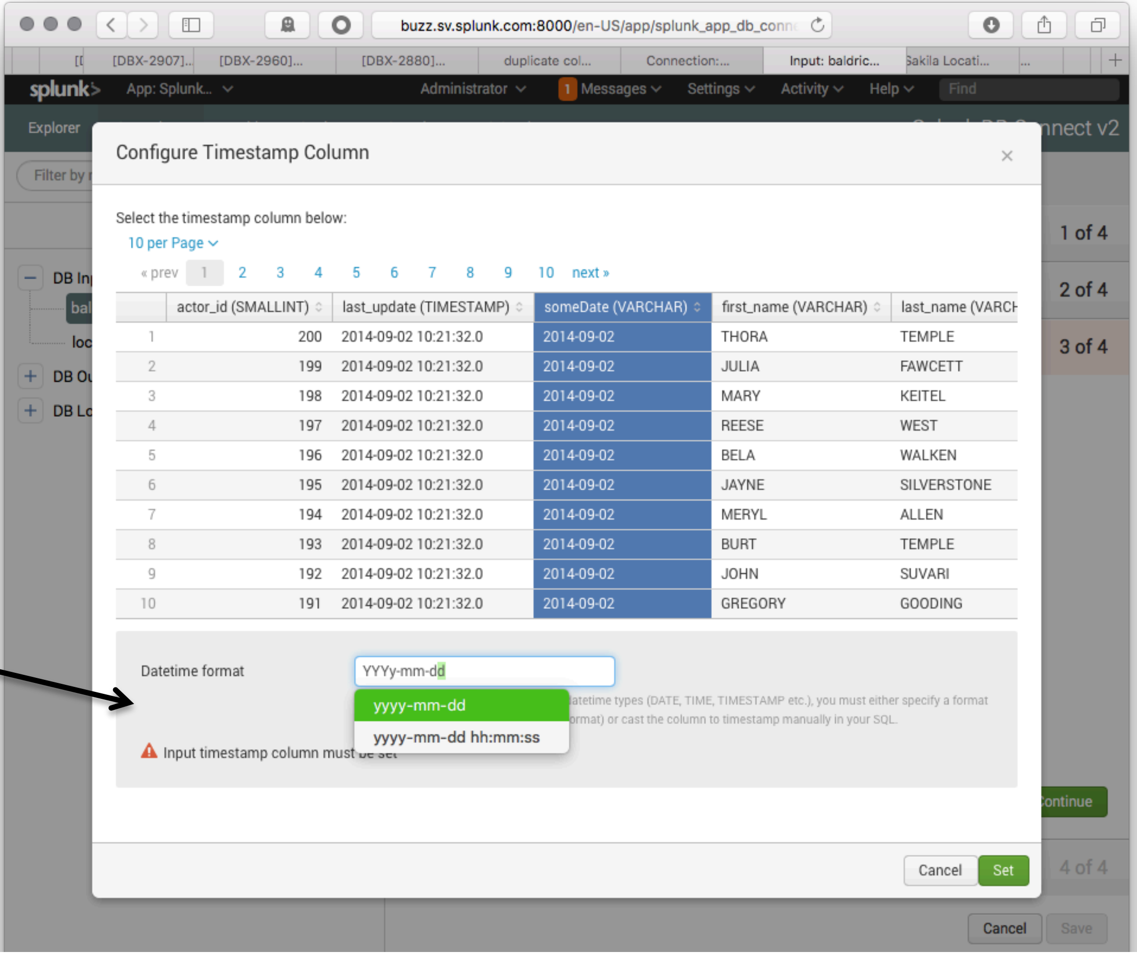
Configure Timestamp Column

Select the timestamp column below:

	last_update (TIMESTAMP)	address (VARCHAR)	district (VARCHAR)	city (VARCHAR)	postal_code (VARC
1	2006-02-15 04:45:30.0	1168 Najafabad Parkway	Kabol	Kabul	
2	2006-02-15 04:45:30.0	1924 Shimonoseki Drive	Batna	Batna	
3	2006-02-15 04:45:30.0	1031 Daugavpils Parkway	Bchar	Bchar	
4	2006-02-15 04:45:30.0	757 Rustenburg Avenue	Skikda	Skikda	
5	2006-02-15 04:45:30.0	1892 Nabereznyje Telny Lane	Tutuila	Tafuna	
6	2006-02-15 04:45:30.0	486 Ondo Parkway	Benguela	Benguela	
7	2006-02-15 04:45:30.0	368 Hunuco Boulevard	Namibe	Namibe	
8	2006-02-15 04:45:30.0	1368 Maracabo Boulevard		South Hill	
9	2006-02-15 04:45:30.0	1623 Kingstown Drive	Buenos Aires	Almirante Brown	
10	2006-02-15 04:45:30.0	1229 Varanasi (Benares) Manor	Buenos Aires	Avellaneda	

We show you the column type

If you pick a TIMESTAMP or DATETIME,
We automatically set _time to that.
If that goes wrong, try a props.conf override



Configure Timestamp Column

Select the timestamp column below:

	actor_id (SMALLINT)	last_update (TIMESTAMP)	someDate (VARCHAR)	first_name (VARCHAR)	last_name (VARCH
1	200	2014-09-02 10:21:32.0	2014-09-02	THORA	TEMPLE
2	199	2014-09-02 10:21:32.0	2014-09-02	JULIA	FAWCETT
3	198	2014-09-02 10:21:32.0	2014-09-02	MARY	KEITEL
4	197	2014-09-02 10:21:32.0	2014-09-02	REESE	WEST
5	196	2014-09-02 10:21:32.0	2014-09-02	BELA	WALKEN
6	195	2014-09-02 10:21:32.0	2014-09-02	JAYNE	SILVERSTONE
7	194	2014-09-02 10:21:32.0	2014-09-02	MERYL	ALLEN
8	193	2014-09-02 10:21:32.0	2014-09-02	BURT	TEMPLE
9	192	2014-09-02 10:21:32.0	2014-09-02	JOHN	SUVARI
10	191	2014-09-02 10:21:32.0	2014-09-02	GREGORY	GOODING

Datetime format: YYYy-mm-dd

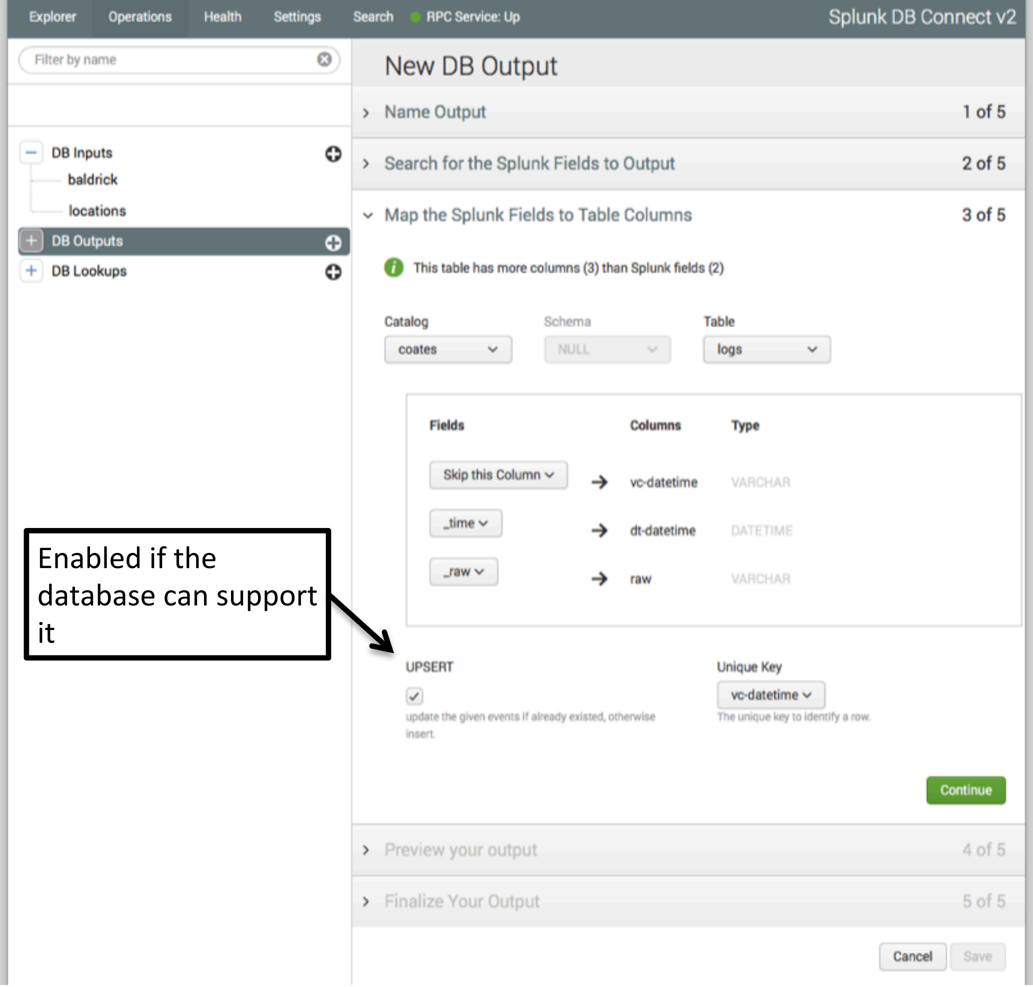
atetime types (DATE, TIME, TIMESTAMP etc.), you must either specify a format
ormat) or cast the column to timestamp manually in your SQL.

⚠ Input timestamp column must be set

If you pick something else,
we show you this formatting tool
so you can override

UPSERT

- Requires specific database support
- We went with a less elegant solution to maximize number of supported databases
  - MS SQL, DB2/Linux, Oracle, MySQL, Aurora, Redshift, Sybase IQ



New DB Output

› Name Output 1 of 5

› Search for the Splunk Fields to Output 2 of 5

⌄ Map the Splunk Fields to Table Columns 3 of 5

This table has more columns (3) than Splunk fields (2)

Catalog: coates
Schema: NULL
Table: logs

Fields		Columns	Type
Skip this Column	→	vc-datetime	VARCHAR
_time	→	dt-datetime	DATETIME
_raw	→	raw	VARCHAR

UPSERT
☑
update the given events if already existed, otherwise insert.

Unique Key: vc-datetime
The unique key to identify a row.

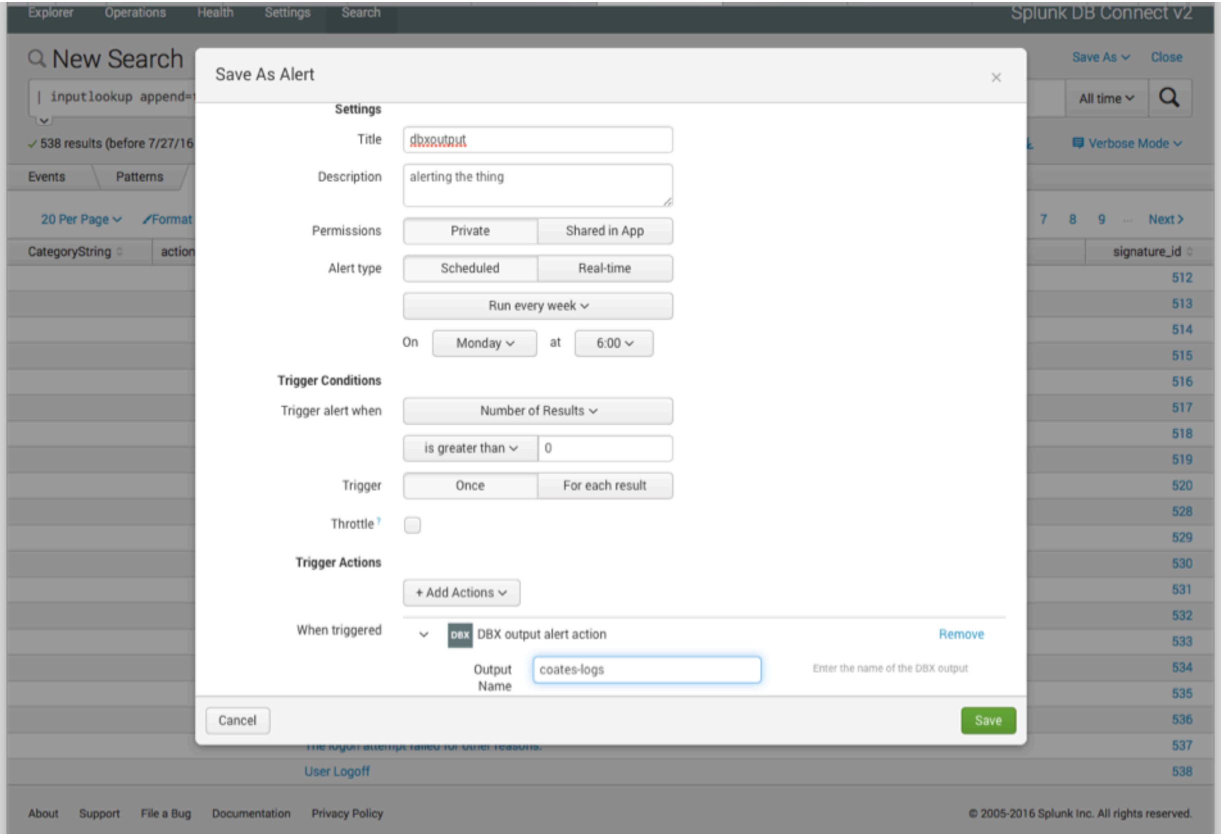
› Preview your output 4 of 5

› Finalize Your Output 5 of 5

Enabled if the database can support it

Output Modular Alert

- Lets you use an existing DB Connect Output as a modular alert
- This is useful for updating in-house systems from Splunk
- *There's also a custom command, | dbxoutput*



Save As Alert

Settings

Title: dbxoutput

Description: alerting the thing

Permissions: Private | Shared in App

Alert type: Scheduled | Real-time

Run every week

On Monday at 6:00

Trigger Conditions

Trigger alert when: Number of Results

is greater than 0

Trigger: Once | For each result

Throttle □

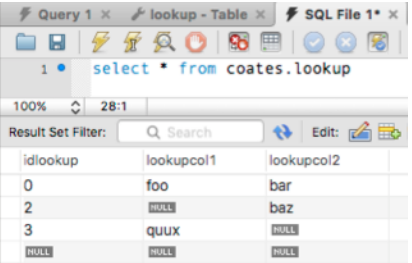
Trigger Actions

When triggered: DBX output alert action

Output Name: coates-logs

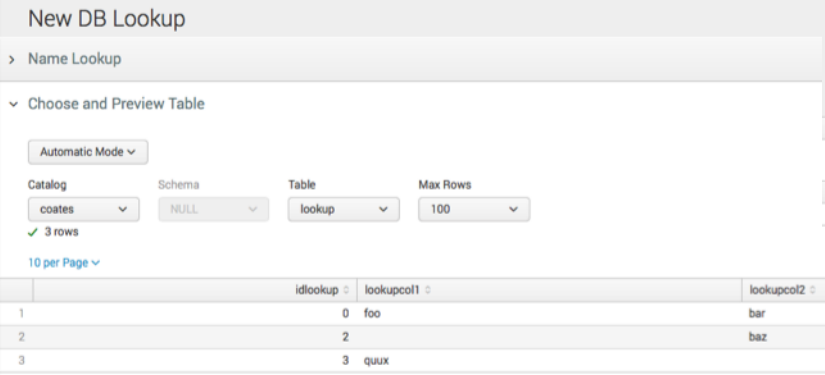
Enter the name of the DBX output

Lookups And Nulls



```
select * from coates.lookup
```

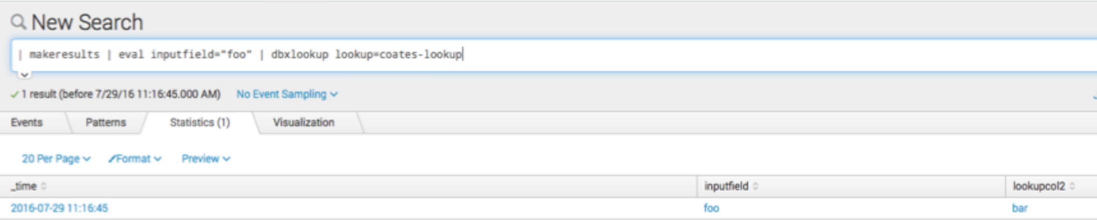
idlookup	lookupcol1	lookupcol2
0	foo	bar
2	NULL	baz
3	quux	NULL
NULL	NULL	NULL



Catalog: coates
Schema: NULL
Table: lookup
Max Rows: 100

✓ 3 rows

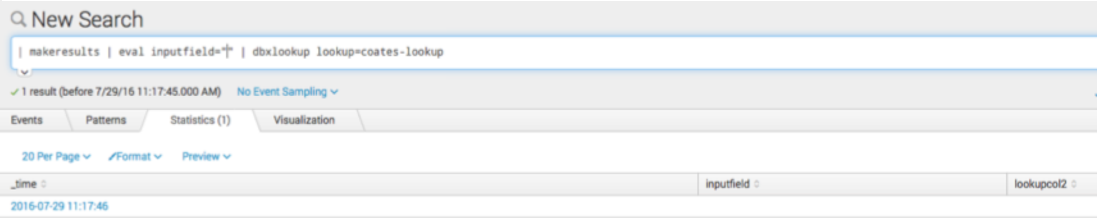
	idlookup	lookupcol1	lookupcol2
1	0	foo	bar
2	2		baz
3	3	quux	



```
| makeresults | eval inputfield="foo" | dbxlookup lookup=coates-lookup
```

✓ 1 result (before 7/29/16 11:16:45.000 AM)

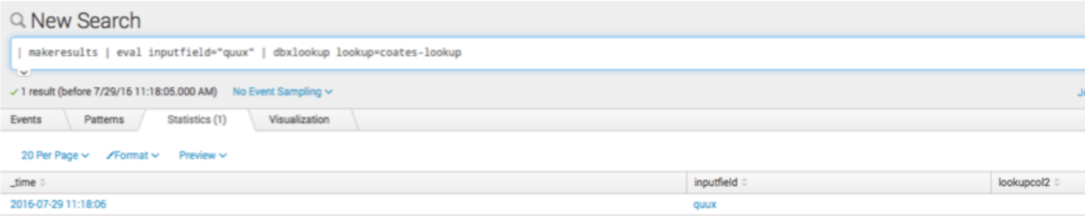
_time	inputfield	lookupcol2
2016-07-29 11:16:45	foo	bar



```
| makeresults | eval inputfield="" | dbxlookup lookup=coates-lookup
```

✓ 1 result (before 7/29/16 11:17:45.000 AM)

_time	inputfield	lookupcol2
2016-07-29 11:17:46		



```
| makeresults | eval inputfield="quux" | dbxlookup lookup=coates-lookup
```

✓ 1 result (before 7/29/16 11:18:05.000 AM)

_time	inputfield	lookupcol2
2016-07-29 11:18:06	quux	

- You can’t return idlookup 2.