Integrating Splunk into your Spring Applications

By Damien Dallimore

About Me

Developer Evangelist at Splunk

Make

Apps

JMX Splunk for JMX

JMS JMS Messaging Modular Input

SNMP SNMP Modular Input

REST REST API Modular Input

CMD Command Modular Input

Talk

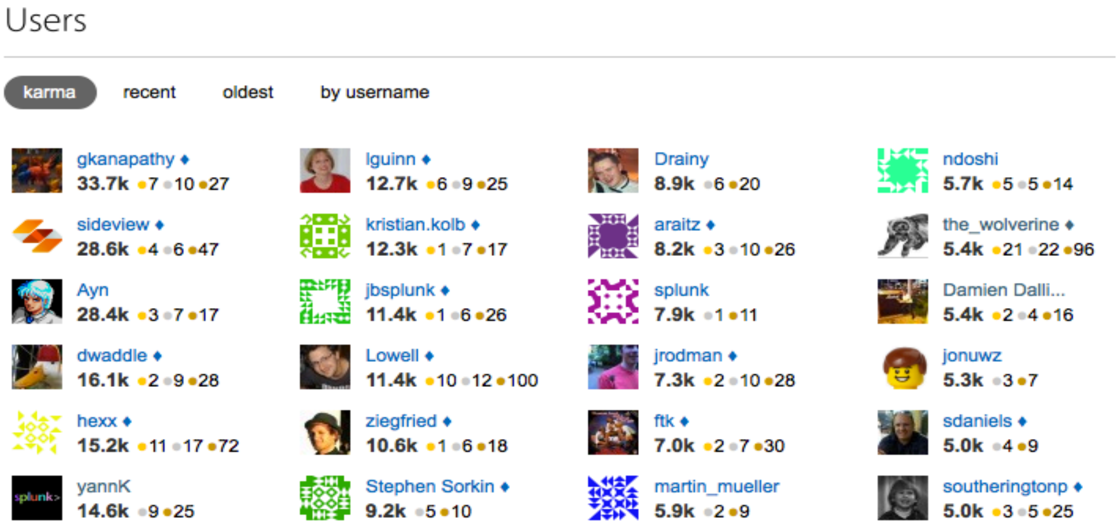
splunk>blogs

A Developer's Smorgasbord

First bite of the Cherry(py)

I didn't always work at Splunk. In fact, many moons ago I used to be a Splunk customer. At the time we were simply looking for a means to better consolidate our enterprise's numerous sources of log data into a

Came from the Splunk Community



Coder

Search or type a command | Explore | Gist | Blog | Help

Damien Dallimore
damiendallimore

Java, Shell, Python
Splunk
Worldwide
ddallimore@splunk.com

Contributions | Repositories | Public Activit

Popular repositories

- **SPLUNK4JMX** — 9
  SplunkBase app for monitoring JVM appl…
- **SplunkJavaAgent** — 7
  An instrumentation agent for tracing cod…
- **SplunkModularInputsPythonFr…** — 5
  This is a framework for building Splunk …
- **SplunkModularInputsJavaFra…** — 2
  This is a framework for building Splunk …
- **spring-integration-splunk-web…** — 2
  Demos for the Spring Integration Splunk …

Your Contributions

From Aotearoa (New Zealand)

Agenda

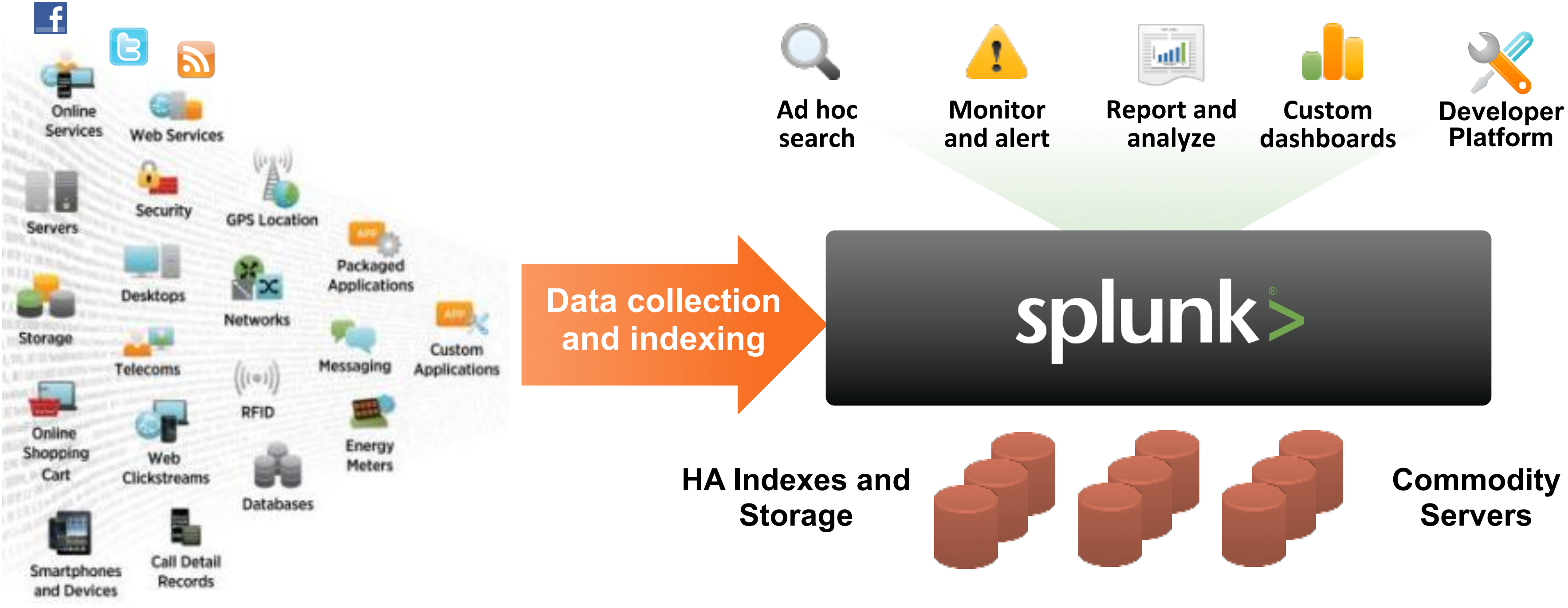
Agenda

- Splunk Overview
- Splunk Developer Platform
- Integrating Splunk and your Spring App
- Questions (time allowing, else see me after)

Splunk Overview

Lets Go Spelunking

Splunk is a Platform for Machine Data



What Does Machine Data Look Like?

Order Processing

ORDER,2012-05-21T14:04:12.484,10098213,569281734,67.17.10.12,43CD1A7B8322,SA-2100

Middleware Error

May 21 14:04:12.996 wl-01.acme.com Order 569281734 failed for customer 10098213. Exception follows: weblogic.jdbc.extensions.ConnectionDeadSQLException: weblogic.common.resourcepool.ResourceDeadException: Could not create pool connection. The DBMS driver exception was: [BEA][Oracle JDBC Driver]Error establishing socket to host and port: ACMEDB-01:1521. Reason: Connection refused

Care IVR

05/21 16:33:11.238 [CONNEVENT] Ext 1207130 (0192033): Event 20111, CTI Num:ServID:Type 0:19:9, App 0, ANI T7998#1, DNIS 5555685981, SerID 40489a07-7f6e-4251-801a-13ae51a6d092, Trunk T451.16
05/21 16:33:11.242 [SCREENPOPEVENT] SerID 40489a07-7f6e-4251-801a-13ae51a6d092 CUSTID 10098213
05/21 16:37:49.732 [DISCEVENT] SerID 40489a07-7f6e-4251-801a-13ae51a6d092

Twitter

{actor:{displayName:"Go Boys!!",followersCount:1366,friendsCount:789,link: "http://dallascowboys.com/",location:{displayName:"Dallas, TX",objectType:"place”}, objectType:"person”,preferredUsername:"B0ysF@n80",statusesCount:6072},body:"Just bought this POS device from @ACME. Doesn't work! Called, gave up on waiting for them to answer! RT if you hate @ACME!!”,objectType:"activity”,postedTime:"2012-05-21T16:39:40.647-0600”}

Machine Data Contains Critical Insights

Order Processing

Customer ID | Order ID | Product ID

ORDER,2012-05-21T14:04:12.484,10098213,569281734,67.17.10.12,43CD1A7B8322,SA-2100

Middleware Error

May 21 14:04:12.996 wl-01.acme.com Order 569281734 failed for customer 10098213.
Exception follows: weblogic.jdbc.extensions.ConnectionDeadSQLException:
weblogic.common.resourcepool.ResourceDeadE[...] Could not create poo[...] The
DBMS driver exception was: [BEA][Oracle JDBC Driver]Error establishing socket to host and port:
ACMEDB-01:1521. Reason: Connection refused

Order ID | Customer ID

Care IVR

05/21 16:33:11.238 [CONNEVENT] Ext 1207130 (0192033): Event 20111, CTI Num:ServID:Type
[...]98#1, DNIS 5555685981, SerID 40489a07-7f6e-4251-801a-
13ae51a6d092, Trunk T451.16
05/21 16:33:11.242 [SCREENPOPEVENT] SerID 40489a07-7f6e-4251-801a-13ae51a6d092
CUSTID 10098213
05/21 16:37:49.732 [DISCEVENT] SerID 40489a07-7f6e-4251-801a-13ae51a6d092

Time Waiting On Hold | Customer ID

Twitter

{actor:{displayName:"Go Boys!!",followersCount:1366,friendsCount:789,link:
"http://dallascowboys.com/",location:{dis[...]"Dallas, TX",objectType[...]
objectType:"person",preferredUsername:"B0ysF@n80",statusesCount:6072},body:"Just bought
this POS device from @ACME. Doesn't work! Called, gave up on waiting for them to answer! RT if
you hate @ACME!!",objectType:"activity",postedTime:"2012-05-21T16:39:40.647-0600"}

Twitter ID | Customer's Tweet | Company's Twitter ID

Machine Data Contains Critical Insights

Order Processing

ORDER,2012-05-21T14:04:12.484,10098213,569281734,67.17.10.12,43CD1A7B8322,SA-2100

Customer ID | Order ID | Product ID

Middleware Error

May 21 14:04:12.996 wl-01.acme.com Order 569281734 failed for customer 10098213. Exception follows: weblogic.jdbc.extensions.ConnectionDeadSQLException: weblogic.common.resourcepool.ResourceDeadE... Could not create po... The DBMS driver exception was: [BEA][Oracle JDBC Driver]Error establishing socket to host and port: ACMEDB-01:1521. Reason: Connection refused

Order ID | Customer ID

Care IVR

05/21 16:33:11.238 [CONNEVENT] Ext 1207130 (0192033): Event 20111, CTI Num:ServID:Type ...98#1, DNIS 5555685981, SerID 40489a07-7f6e-4251-801a-13ae51a6d092, Trunk T451.16
05/21 16:33:11.242 [SCREENPOPEVENT] SerID 40489a07-7f6e-4251-801a-13ae51a6d092 CUSTID 10098213
05/21 16:37:49.732 [DISCEVENT] SerID 40489a07-7f6e-4251-801a-13ae51a6d092

Time Waiting On Hold | Customer ID

Twitter

{actor:{displayName:"Go Boys!!",followersCount:1366,friendsCount:789,link: "http://dallascowboys.com/",location:{dis... "Dallas, TX",objectType... objectType:"person",preferredUsername:"B0ysF@n80",statusesCount:6072},body:"Just bought this POS device from @ACME. Doesn't work! Called, gave up on waiting for them to answer! RT if you hate @ACME!!",objectType:"activity",postedTime:"2012-05-21T16:39:40.647-0600"}

Twitter ID | Customer's Tweet | Company's Twitter ID

Splunk Developer Platform

Developer Platform

DevOps | Integrate | Build

JavaScript | Java | Python | PHP | C# | Ruby

REST API

splunk>

Splunk REST API

- An API method for all features of the platform
- Send data in
- Search and Export data out
- JSON , CSV, XML

Integrating Splunk and your Spring App

How can Splunk help out the Spring Developer ?

- During Dev/Test
  - Use Splunk to deliver deeper insights , hook in more thorough test case assertions
  - Aggregate data from your apps in development
- Integrate the Data you have collected with Splunk
  - Use Spring as the EAI backbone to build integrated data solutions , correlate data form numerous sources
- Build standalone Big Data apps
  - Let Splunk do the hard yards on the data and searching side

What are some of the hooks?

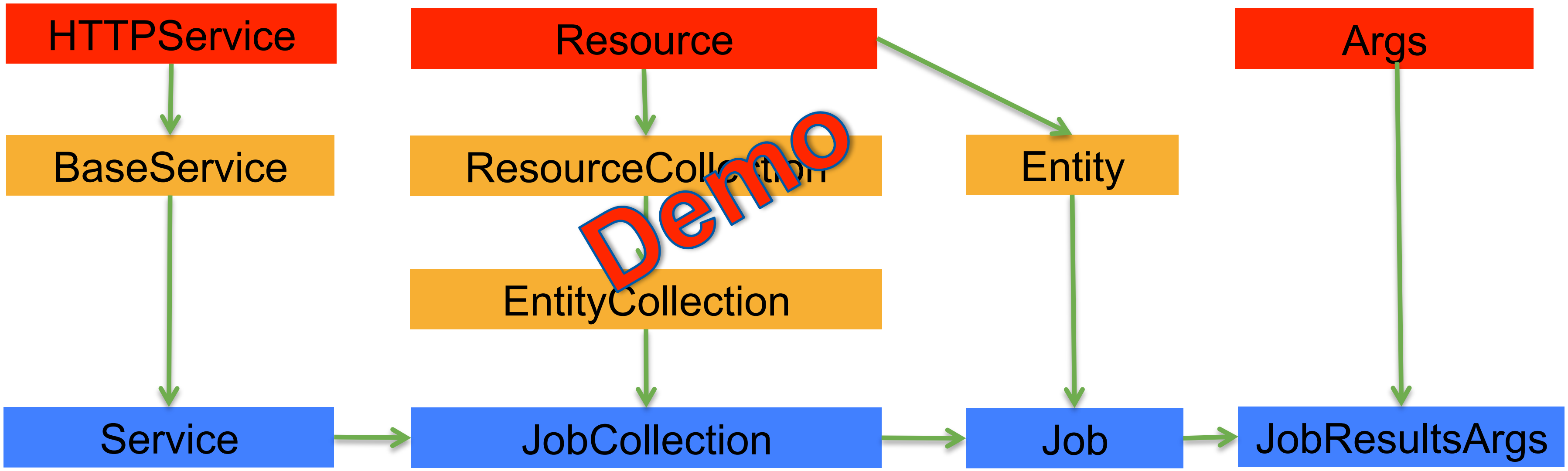
- REST API
- Splunk SDK for Java
- Spring Integration Adaptors
- Logging
- JMS Messaging
- JMX MBeans
- REST
- BCI(byte code injection) tracing

Splunk SDK for Java

Splunk SDK for Java

- Open sourced under the Apache v2.0 license
- git clone https://github.com/splunk/splunk-sdk-java.git
- JRE 6+
- Maven/Gradle Repository
- Code Examples :
  - Connect
  - Hit your first endpoint
  - Send data in
  - Search for data , Simple and Realtime
  - Scala and Groovy can play too

SDK Class Design



springone 2GX

Spring Integration Adaptors

When Spring and Splunk collide

- Developers like tools & frameworks that increase productivity
- An SDK makes it easier to use a REST API
- A declarative Enterprise Integration framework makes it easier to build solutions that need to integrate, transform, filter and route data from heterogeneous channels and data sources
- We now have the Spring Integration Splunk Adaptors to make it easier for Java Developers to integrate Splunk into their solutions utilizing a semantic they are most familiar with in the Spring framework.

Spring Integration And Splunk

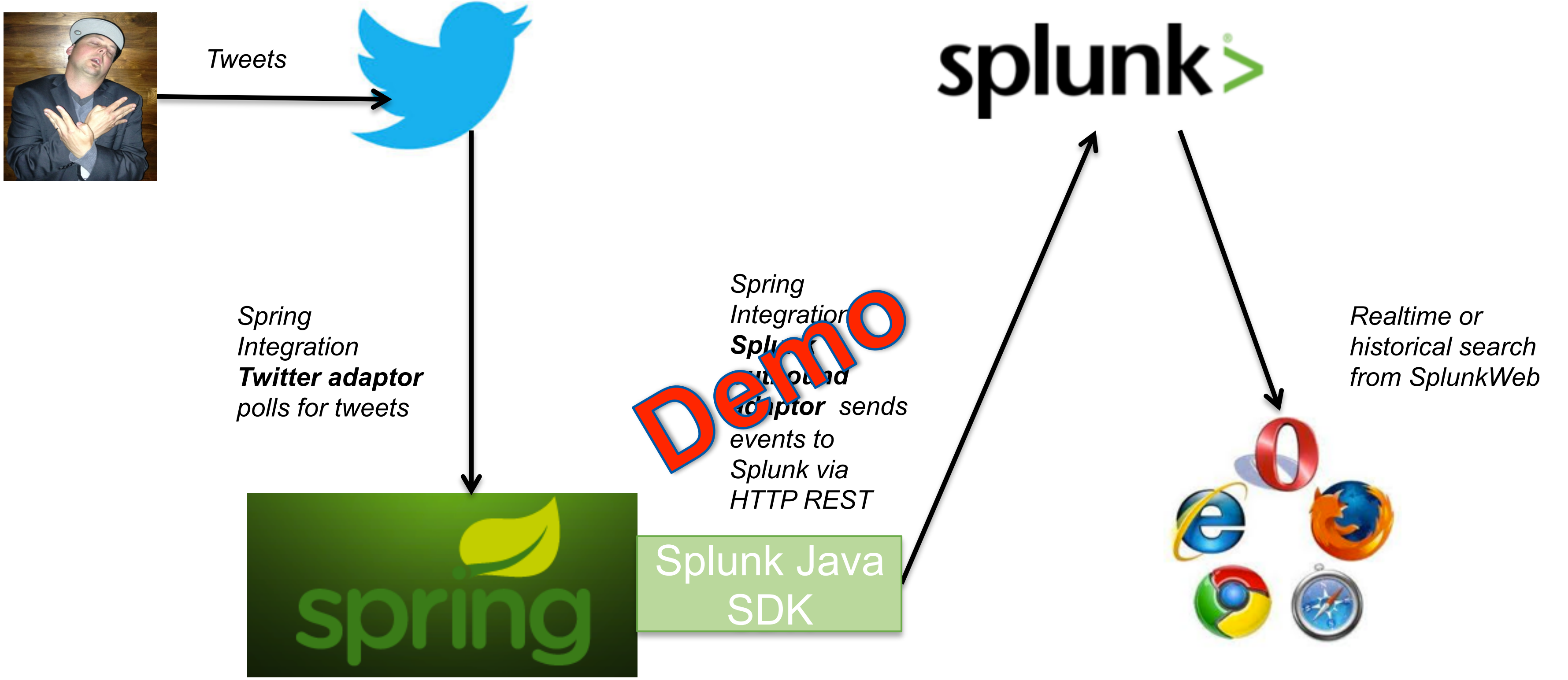
Inbound Adapter

- Used to execute Splunk searches and produce messages containing results
- Search modes: *BLOCKING, NORMAL, REALTIME, EXPORT, SAVEDSEARCH*
- Date/Time Ranges

Outbound Adapter

- Write system or application events to Splunk
- Write to a named index, submit a REST request, write to a data input bound to a server TCP port

Message payload for Splunk I/O adapters is SplunkEvent



Raw events from Twitter are transformed into best practice logging format

Logging

SplunkJavaLogging

- A logging framework to allow developers to as seamlessly as possible integrate Splunk best practice logging semantics into their code
- Transport log events to Splunk directly from your code
- Custom handler/appender implementations(REST and Raw TCP) for common Java logging frameworks .
  - LogBack
  - Log4j (Log4j2 coming also)
  - java.util.logging
- Utility classes for formatting log events
- Configurable in memory buffer to handle network outages

Developers just log as they are used to

```java
/**
 * Just log as usual, and wire up a Splunk REST/TCP appender to forward the
 * event to Splunk
 */
private static void simpleLogExample() {

    // get your logger
    Logger logger = LoggerFactory.getLogger("splunk.logger");

    // log a regular string
    logger.info("REST for the wicked");

}
```

Search

```
index=main sourcetype=logback name="Failed Login" | stats count as "Failed Logins" by app,user
```

1 result in the last 15 minutes (from 3:46:00 PM to 4:01:59 PM on Tuesday, August 7, 2012)

Export Options

Overlay: None

	app	user	Failed Logins
1	myapp	jane	3

Better

A-HA

```java
/**
 * Format the log message to adhere to Splunk best practice logging
 * semantics
 */
private static void splunkLogEventExample() {

    // get your logger
    Logger logger = LoggerFactory.getLogger("splunk.logger");

    // create a SplunkLogEvent with a date and values quoted
    SplunkLogEvent event = new SplunkLogEvent("Failed Login", "someID");

    //add Splunk CIM fields
    event.setAuthApp("myapp");
    event.setAuthUser("jane");

    // add a custom field
    event.addPair("somefieldname", "foobar");

    // log a splunk log event generated string
    logger.info(event.toString());
}
```

2012-08-07 15:54:06:644+1200 name="Failed Login" event_id="someID" app="myapp" user="jane" somefieldname="foobar"

Semantic Logging

Log anything that can add value when aggregated, charted or further analyzed

Example Bogus Pseudo-Code:

```
void submitPurchase(purchaseId)
{
        log.info("action=submitPurchaseStart, purchaseId=%d", purchaseId)
        //these calls throw an exception on error
        submitToCreditCard(...)
        generateInvoice(...)
        generateFullfillmentOrder(...)
        log.info("action=submitPurchaseCompleted, purchaseId=%d", purchaseId)
 }
```

- Create Human Readable Events
- Clearly Timestamp Events
- Use Key-Value Pairs (JSON Logging)
- Separate Multi-Value Events
- Log Unique Identifiers

Log4J config

log4j.appender.splunkrest=com.splunk.logging.log4j.appender.SplunkRestAppender
log4j.appender.splunkrest.user=admin
log4j.appender.splunkrest.pass=somepass
log4j.appender.splunkrest.host=localhost
log4j.appender.splunkrest.port=8089
log4j.appender.splunkrest.delivery=stream
log4j.appender.splunkrest.metaSource=rest
log4j.appender.splunkrest.metaSourcetype=testing
log4j.appender.splunkrest.metaIndex=main
log4j.appender.splunkrest.maxQueueSize=5MB
log4j.appender.splunkrest.dropEventsOnQueueFull=false

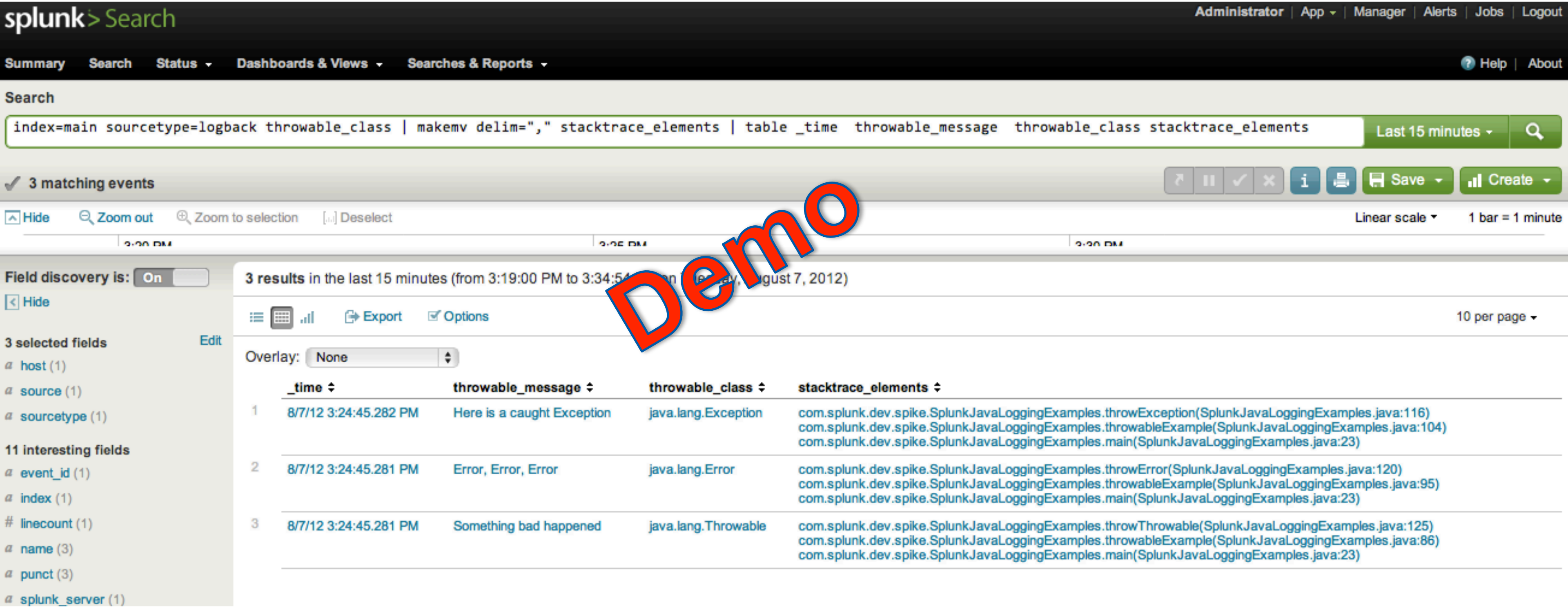
Java stacktraces are a nuisance

```
java.lang.Throwable: Something bad happened
	at com.splunk.dev.spike.SplunkJavaLoggingExamples.throwThrowable(SplunkJavaLoggingExamples.java:126)
	at com.splunk.dev.spike.SplunkJavaLoggingExamples.throwableExample(SplunkJavaLoggingExamples.java:86)
	at com.splunk.dev.spike.SplunkJavaLoggingExamples.main(SplunkJavaLoggingExamples.java:23)
java.lang.Error: Error, Error, Error
	at com.splunk.dev.spike.SplunkJavaLoggingExamples.throwError(SplunkJavaLoggingExamples.java:121)
	at com.splunk.dev.spike.SplunkJavaLoggingExamples.throwableExample(SplunkJavaLoggingExamples.java:96)
	at com.splunk.dev.spike.SplunkJavaLoggingExamples.main(SplunkJavaLoggingExamples.java:23)
java.lang.Exception: Here is a caught Exception
	at com.splunk.dev.spike.SplunkJavaLoggingExamples.throwException(SplunkJavaLoggingExamples.java:117)
	at com.splunk.dev.spike.SplunkJavaLoggingExamples.throwableExample(SplunkJavaLoggingExamples.java:105)
	at com.splunk.dev.spike.SplunkJavaLoggingExamples.main(SplunkJavaLoggingExamples.java:23)
```

SplunkJavaLogging is your friend

```java
/**
 * Log an Error/Exception/Throwable and handle the stacktrace elements in Splunk as a multi value field
 */
private static void throwableExample() {

        // get your logger
        Logger logger = LoggerFactory.getLogger("splunk.logger");

        try {

                throw new Exception("Danger Danger");

        } catch (Throwable throwable) {
                SplunkLogEvent event = new SplunkLogEvent();
                event.addThrowable(throwable);
                logger.error(event.toString());
        }

}
```

Java stacktraces in Splunk unravelled

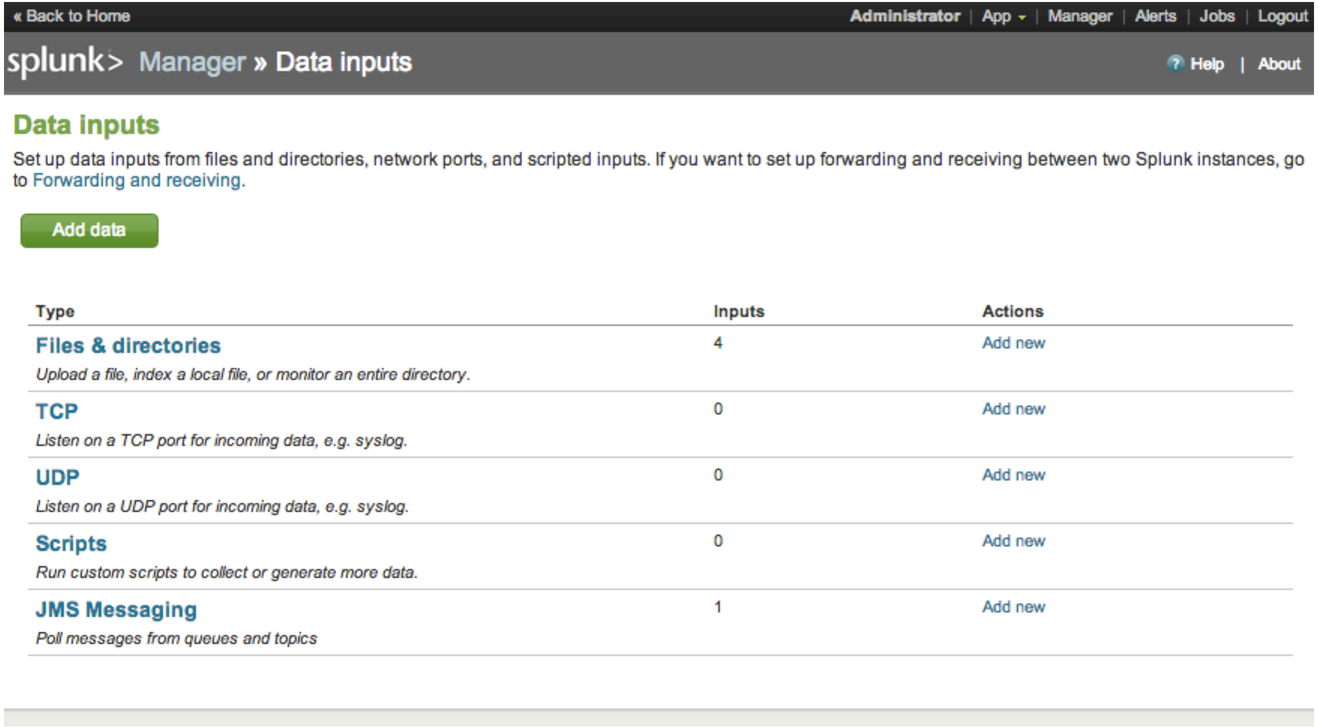


JMS Messaging

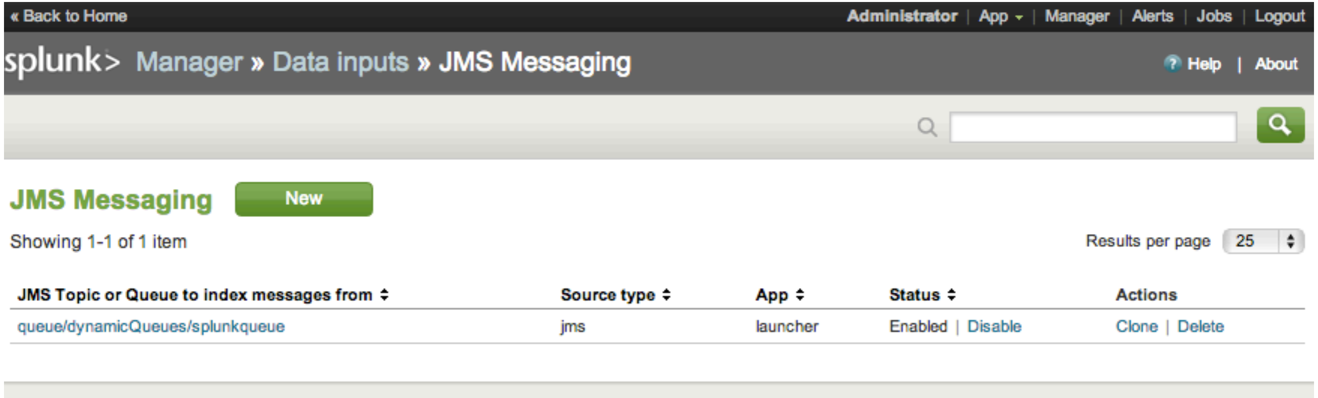
JMS Messaging Splunk Input

- JMS is an interface that abstracts your underlying MOM provider implementation
- Send messages to *parallel* queues or topics in Spring that Splunk can tap into
- Index messages from :
    - MQ Series / Websphere MQ
    - Tibco EMS
    - ActiveMQ
    - HornetQ
    - RabbitMQ
    - SonicMQ
    - JBoss Messaging
    - Weblogic JMS
    - Native JMS
    - StormMQ
- Note : Non-JMS inputs also available (Stomp , ZeroMQ)

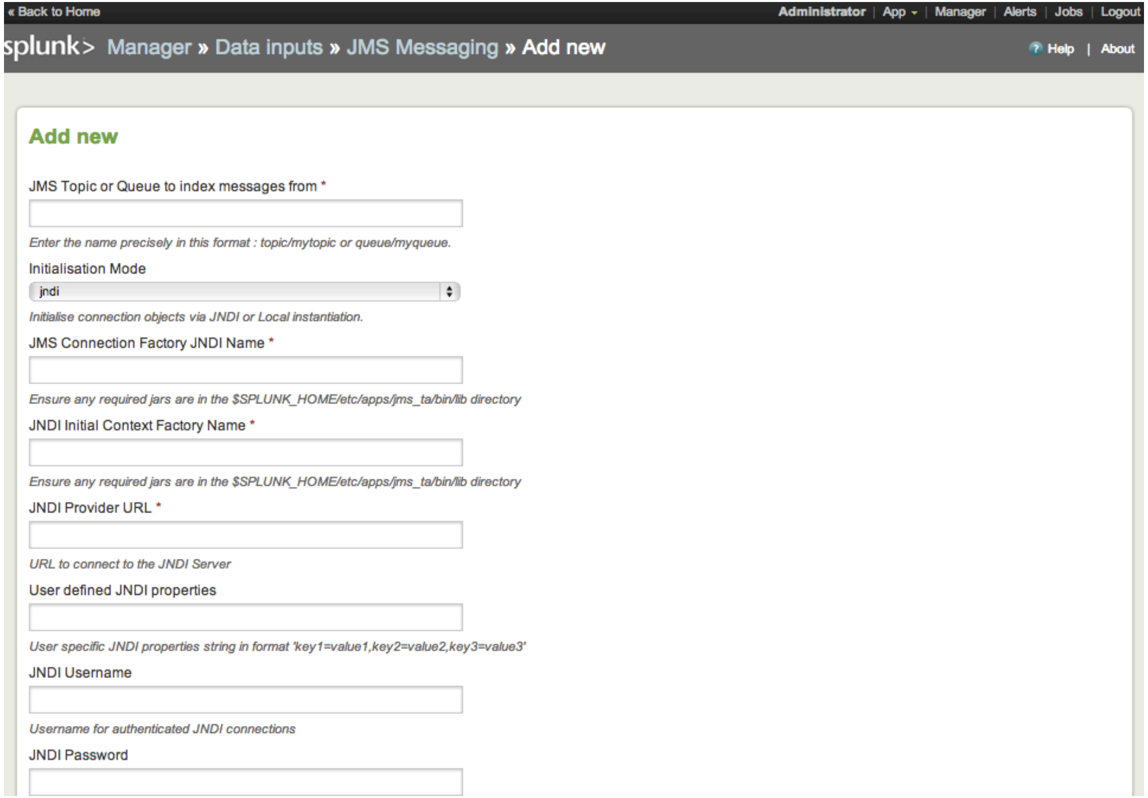
JMS input fully integrated into Splunk



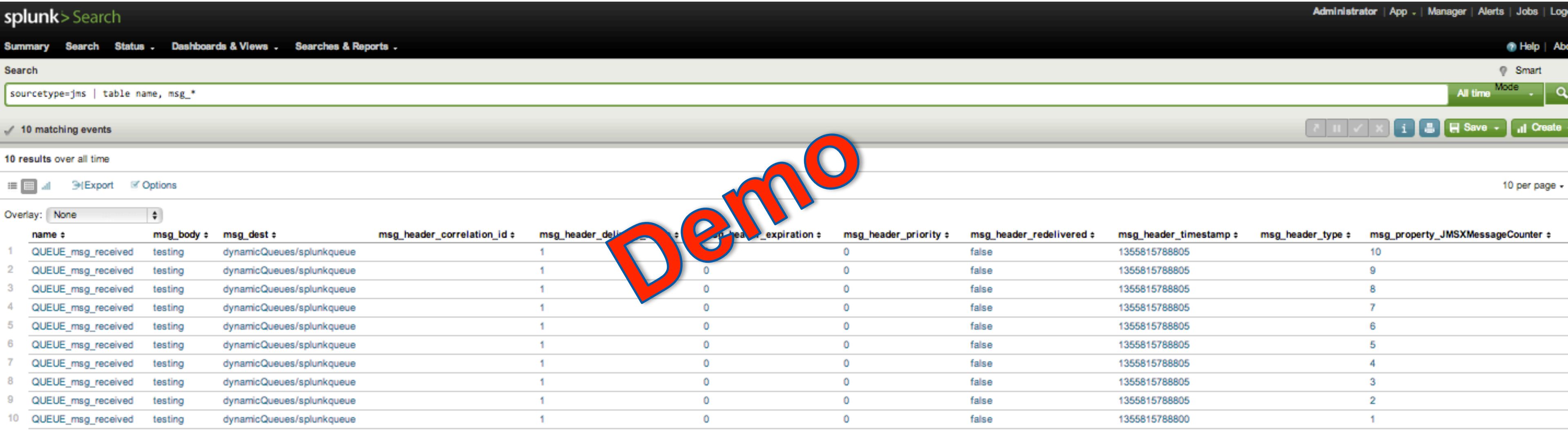
Add a new queue/topic input



Configure the properties to connect



Get instant operational visibility



	name	msg_body	msg_dest	msg_header_correlation_id	msg_header_deli[illegible]	[illegible]_expiration	msg_header_priority	msg_header_redelivered	msg_header_timestamp	msg_header_type	msg_property_JMSXMessageCounter
1	QUEUE_msg_received	testing	dynamicQueues/splunkqueue		1	[illegible]	0	false	1355815788805		10
2	QUEUE_msg_received	testing	dynamicQueues/splunkqueue		1	0	0	false	1355815788805		9
3	QUEUE_msg_received	testing	dynamicQueues/splunkqueue		1	0	0	false	1355815788805		8
4	QUEUE_msg_received	testing	dynamicQueues/splunkqueue		1	0	0	false	1355815788805		7
5	QUEUE_msg_received	testing	dynamicQueues/splunkqueue		1	0	0	false	1355815788805		6
6	QUEUE_msg_received	testing	dynamicQueues/splunkqueue		1	0	0	false	1355815788805		5
7	QUEUE_msg_received	testing	dynamicQueues/splunkqueue		1	0	0	false	1355815788805		4
8	QUEUE_msg_received	testing	dynamicQueues/splunkqueue		1	0	0	false	1355815788805		3
9	QUEUE_msg_received	testing	dynamicQueues/splunkqueue		1	0	0	false	1355815788805		2
10	QUEUE_msg_received	testing	dynamicQueues/splunkqueue		1	0	0	false	1355815788800		1

JMX

Expose Mbeans in your Spring App

- 3 tiers can be exposed
  - JVM (java.lang domain)
  - Framework / Container (Spring , Tomcat etc…)
  - Application (whatever you have coded)
- Attributes , Operations and Notifications
- Use Splunk for JMX to monitor the internals of your running Spring apps

Splunk for JMX

- Multiple connectivity options
  - rmi/iiop ,direct process attachment, mx4j http connectors
- Works with all JVM variants
- Scales out to monitor large scale JVM infrastructures

Demo

REST

REST is easy with Spring

- Create an endpoint with Spring Integration
- Splunk can poll the REST API endpoint
- Multiple authentication mechanisms
- Custom response handling / pre-processing
- Send responses in XML , JSON
- Splunk can natively index the responses and then simply search over the auto extracted fields

REST : The Data Potential

There is a world of data out there available via REST that can be brought into Splunk, correlated and enriched against your existing data, or used for entirely new uses cases that you might conceive of once you see what is available and where your data might take you.

- Twitter
- Foursquare
- LinkedIn
- Facebook
- Fitbit
- Amazon
- Yahoo
- Reddit
- YouTube
- Flickr
- Wikipedia
- GNIP
- Box

Demo

- Okta
- Datasift
- Google APIs
- Weather Services
- Seismic monitoring
- Publicly available socio-economic data
- Traffic data
- Stock monitoring
- Security service providers
- Proprietary systems and platforms
- Other "data related" software products
- **The REST "dataverse" is vast , but I think you get the point.**

BCI(Byte Code Injection) Tracing

Splunk Java Agent

An instrumentation agent for tracing code level metrics via bytecode injection, JMX attributes/operations/notification and decoded HPROF records and streaming these events directly into Splunk

https://github.com/damiendallimore/SplunkJavaAgent

- class loading
- method execution
- method timings (cumulative, min, avg, max, std deviation)
- method call tracing(count of calls, group by app/app node(for clustered systems)/thread/class/package)
- method parameter and return value capture (in progress)
- application/thread stalls , thread dumps and stacktraces
- errors/exceptions/throwables
- JVM heap analysis, object/array allocation count/size,class dumps, leak detection, stack traces, frames
- JMX attributes/operations/notifications from the JVM or Application layer MBean Domains

Design goals

- Just pull out the raw metrics , then let Splunk perform the crunching
- Format events in best practice semantic , well defined key value pairs , tagged events help correlation across distributed environment
- Low impact to the instrumented application
- No code changes required
- Flexible configuration
- Extensible
- Generic open source agent , I may have used some Splunk terms in the naming conventions, but it is still completely generic , **anyone want to collaborate !**
- Not a full blown APM solution , just pulling raw data.
- Incorporate into your Spring apps during Dev/Test to get deeper insights

Setup should be as simple as possible

This is all you pass to the JVM at startup :

-javaagent:splunkagent.jar

Everything required by the agent is built into the one single jar file

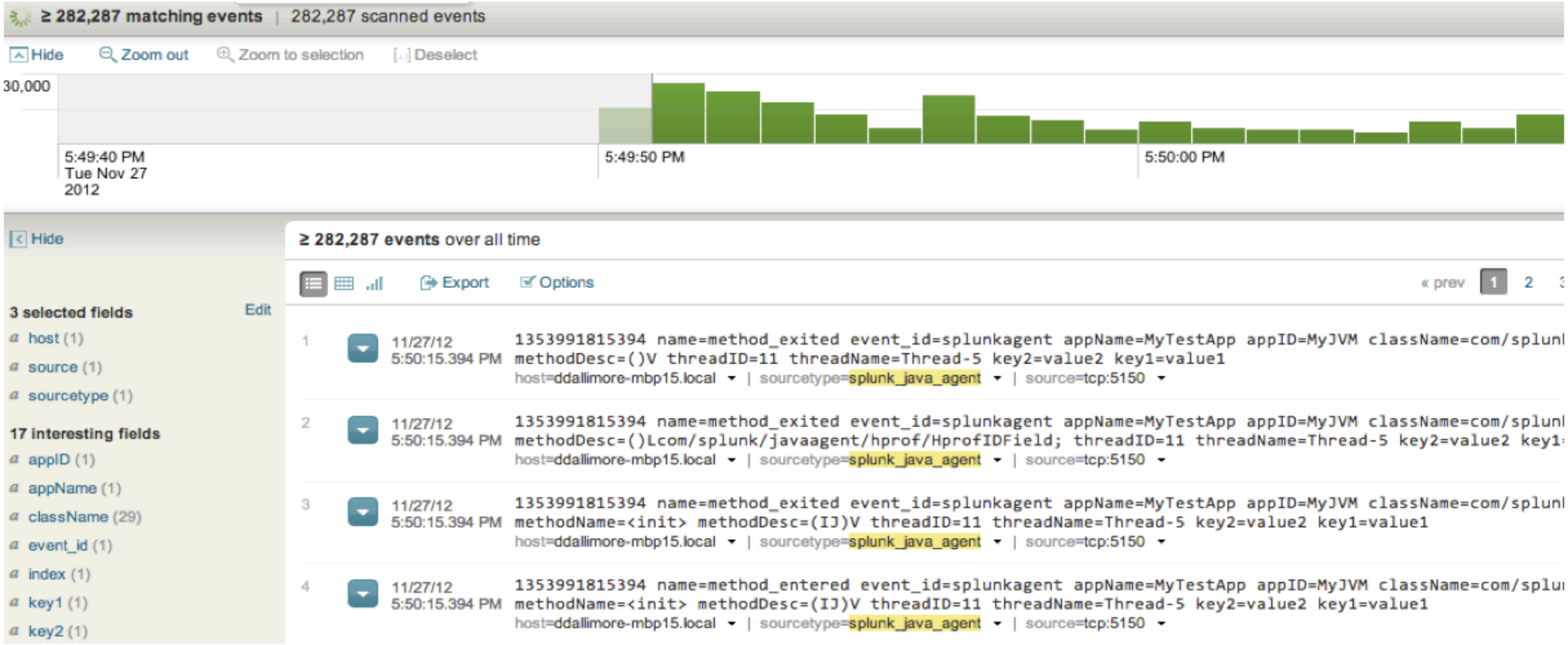
We also have a new Eclipse plugin that incorporates this functionality if you don't want to setup the JVM command line argument manually.

Configuration should allow for flexibility

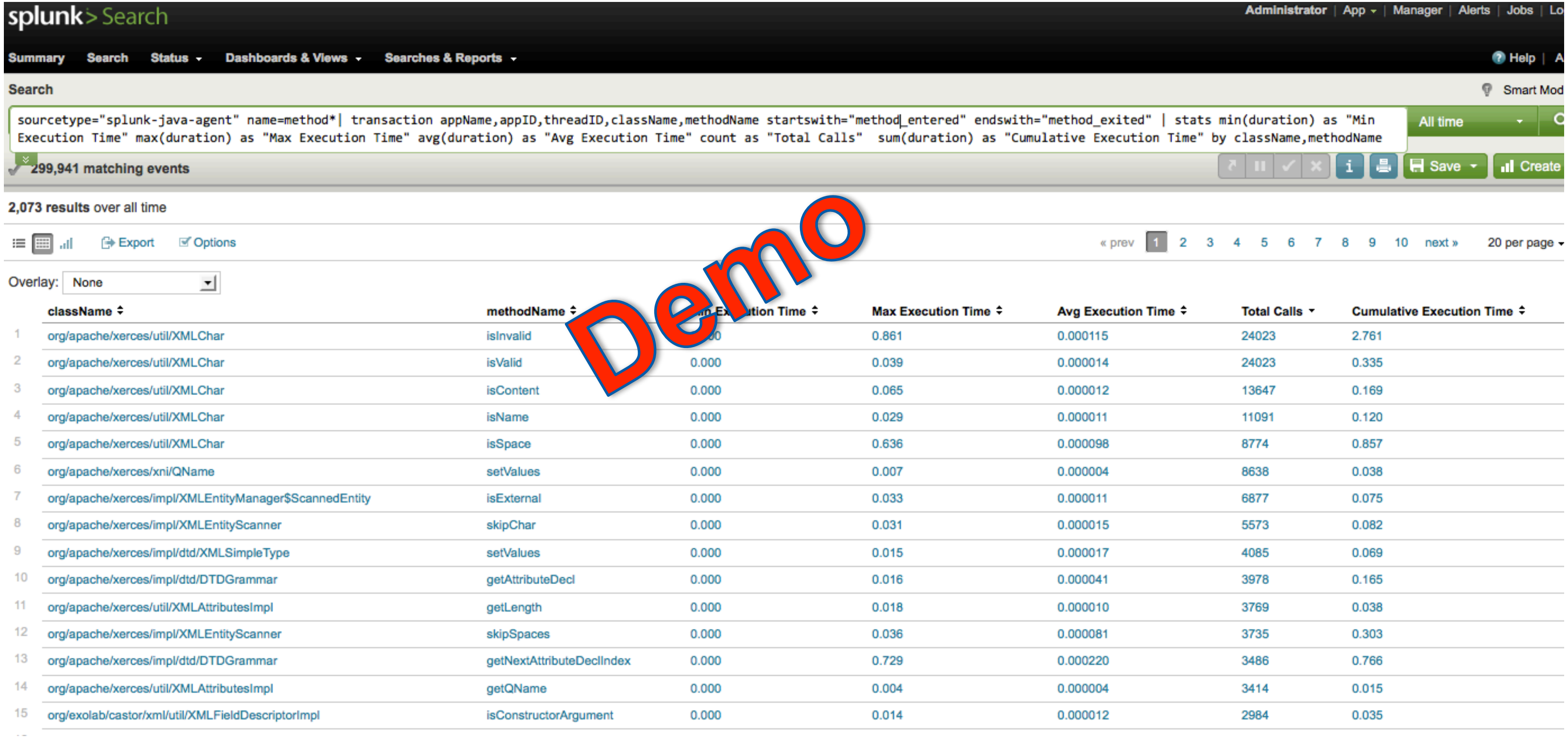
```
#---------------------
# Common Agent options
#---------------------

agent.app.name=MyTestApp
agent.app.instance=MyJVM
agent.userEventTags=key1=value1,key2=value2

#-------------------------
# Splunk Transport options
#-------------------------
#splunk.transport.internalQueueSize=10000
splunk.transport.impl=com.splunk.javaagent.transport.SplunkTCPTransport
#splunk.transport.impl=com.splunk.javaagent.transport.SplunkStdOutTransport
splunk.transport.tcp.host=ubuntu-splunk
splunk.transport.tcp.port=5150
splunk.transport.tcp.maxQueueSize=5MB
splunk.transport.tcp.dropEventsOnQueueFull=false

#------------------------------------
# Class/Method/Error Tracing options
#------------------------------------
#trace.whitelist=com/some/package/you/want/to/monitor
trace.blacklist=com/sun,sun/,java/,javax/,com/splunk/javaagent/
trace.methodEntered=true
trace.methodExited=true
trace.classLoaded=true
trace.errors=true
trace.returnValues=true
trace.parameterValues=true
```

```
#----------------------------
# HPROF Dump Collection options
#----------------------------

#periodically dump hprof file(using JMX operation call)
trace.hprof=false
trace.hprof.tempfile=mydump.hprof
#trace.hprof.tempfile=/etc/tmp/dump.hprof
#in seconds
trace.hprof.frequency=600
#trace.hprof.recordtypes=2,3,4

#-----------------------------------------------------
# JMX attribute/operation/notification collection options
#-----------------------------------------------------

#Embedded JMX polling , all other JMX config is in the JMX XML configuration file.
trace.jmx=false
#name of XML files(minus the ".xml" suffix) that should reside in the root of splunkagent.jar
trace.jmx.configfiles=jmx
#trace.jmx.configfiles=goo,foo
#in seconds
trace.jmx.default.frequency=60
#trace.jmx.goo.frequency=30
```

Raw events streamed into Splunk



Use Splunk to derive insights



	className	methodName	Min Execution Time	Max Execution Time	Avg Execution Time	Total Calls	Cumulative Execution Time
1	org/apache/xerces/util/XMLChar	isInvalid	[illegible]	0.861	0.000115	24023	2.761
2	org/apache/xerces/util/XMLChar	isValid	0.000	0.039	0.000014	24023	0.335
3	org/apache/xerces/util/XMLChar	isContent	0.000	0.065	0.000012	13647	0.169
4	org/apache/xerces/util/XMLChar	isName	0.000	0.029	0.000011	11091	0.120
5	org/apache/xerces/util/XMLChar	isSpace	0.000	0.636	0.000098	8774	0.857
6	org/apache/xerces/xni/QName	setValues	0.000	0.007	0.000004	8638	0.038
7	org/apache/xerces/impl/XMLEntityManager$ScannedEntity	isExternal	0.000	0.033	0.000011	6877	0.075
8	org/apache/xerces/impl/XMLEntityScanner	skipChar	0.000	0.031	0.000015	5573	0.082
9	org/apache/xerces/impl/dtd/XMLSimpleType	setValues	0.000	0.015	0.000017	4085	0.069
10	org/apache/xerces/impl/dtd/DTDGrammar	getAttributeDecl	0.000	0.016	0.000041	3978	0.165
11	org/apache/xerces/util/XMLAttributesImpl	getLength	0.000	0.018	0.000010	3769	0.038
12	org/apache/xerces/impl/XMLEntityScanner	skipSpaces	0.000	0.036	0.000081	3735	0.303
13	org/apache/xerces/impl/dtd/DTDGrammar	getNextAttributeDeclIndex	0.000	0.729	0.000220	3486	0.766
14	org/apache/xerces/util/XMLAttributesImpl	getQName	0.000	0.004	0.000004	3414	0.015
15	org/exolab/castor/xml/util/XMLFieldDescriptorImpl	isConstructorArgument	0.000	0.014	0.000012	2984	0.035

A couple of other integrations you may like

Let’s integrate some mobile data

Android SDK project

- Cousin to the Splunk SDK for Java, has all the same functionality and code examples are the same
- Utility classes to make common tasks easier
  - logging to Splunk
  - searching Splunk
  - **pulling system/device/sensor metrics from Android and logging to Splun**k
- Send Android data to Splunk and then use the Spring Integration Splunk Inbound adaptor to integrate this into your Spring applications.
- Community preview currently published to Github

Demo

Making Hadoop analytics easier

HUNK (Splunk Analytics for Hadoop)

- A new product offering from Splunk , currently in Beta preview
- Allows you to use the power and simplicity of Splunk to search over data locked away in HDFS
- Sits on top of HDFS as if it was a native Splunk Index
- Virtual Indexes
- So you can use the Spring Integration Splunk Inbound Adaptor to search over data in HDFS, or correlate your HDFS data with other data you have indexed and integrate it into your Spring Applications.

Splunk sits on top of HDFS

1 Point Splunk at Hadoop Cluster

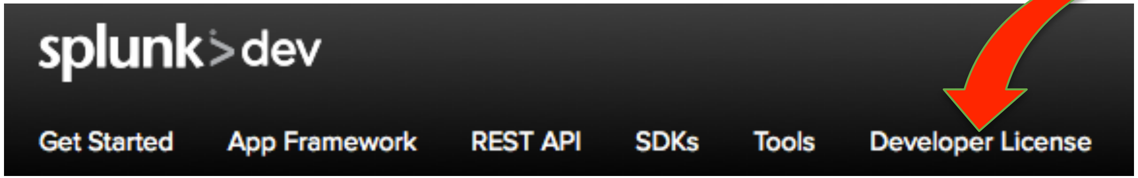
2 Immediately start exploring, analyzing and visualizing raw data in Hadoop

Housekeeping & Plugs

Where to Go for More Info

Email
devinfo@splunk.com
Portal
http://dev.splunk.com
Github
https://github.com/splunk

Twitter
@splunkdev
Blog
http://blogs.splunk.com/dev
Demos
http://demos.splunk.com



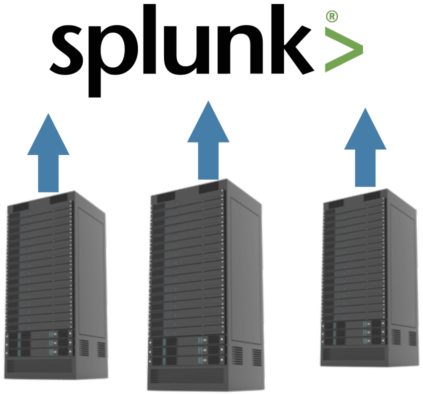
10 GB Free

Easy to Get Started

Download and install in minutes

1. Download

2. Eat your Machine Data



3. Start Splunking

.conf2013

Sept 30 – Oct 3
Las Vegas

4th Annual Event

3+ days | 100+ sessions | 30+ customer sessions
1,500+ IT Pros | 20+ partners
2 days of Splunk University
Specialist Tracks: CIO, Big Data, Executive

splunk.com/goto/conf

Links

Github Gists for SDK code examples : https://gist.github.com/damiendallimore

SDK docs at dev.splunk.com : http://dev.splunk.com/view/SP-CAAAECN

Splunk SDK for Java Github repository : https://github.com/splunk/splunk-sdk-java

Maven/Gradle/Ivy Repository : http://splunk.artifactoryonline.com/splunk/ext-releases-local

Splunk Spring Integration repository on Github : https://github.com/SpringSource/spring-integration-extensions/tree/master/spring-integration-splunk

Splunk Spring Integration demo on Github : https://github.com/damiendallimore/spring-integration-splunk-webex-demo

Splunk Eclipse plugin : http://dev.splunk.com/view/splunk-plugin-eclipse/SP-CAAAEQP

Splunk Java Logging on Github : https://github.com/splunk/splunk-library-javalogging

Links cont….

Splunk Java Agent on Github :
https://github.com/damiendallimore/SplunkJavaAgent
Splunk Android SDK on Github :
https://github.com/damiendallimore/splunk-sdk-android
Splunk REST API reference :
http://docs.splunk.com/Documentation/Splunk/latest/RESTAPI/RESTcontents
Free Splunk download : http://www.splunk.com/get?r=header
Best practice logging overview :
http://dev.splunk.com/view/logging-best-practices/SP-CAAADP6
Splunk SDK for Java videos :
http://dev.splunk.com/view/get-started/SP-CAAAECH
HUNK Beta video : http://www.splunk.com/view/SP-CAAAH2F

Contact me

Email : ddallimore@splunk.com
Twitter : @damiendallimore
Skype : damien.dallimore
Github : damiendallimore
Splunkbase : damiend
Slideshare : http://www.slideshare.net/damiendallimore

Thanks !