

Confidence In Conclusions: Leveraging Splunk For Data Driven Insights

David Uslan

Lead Data Analyst, ICF Technology

.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.

Who Are We?

- Established in 2002
- ~450 Employees
 - 100 Developers
 - 3 Analysts (“Full Time Splunkers”)
- 70,000 Paid streams per day
- 33,000 Daily Users
- 8,000 Daily Performers
- Splunking since 2012



What Is The Value Of Insight?



.conf2016

What IS Insight?

“The essence of profound insight is simplicity.”

- *Jim Collins*

The Value Of Data Driven Insights

1. Make better, more informed decisions
2. Demystify the unknown: be certain projects are effective
3. Repeat positive outcomes
4. Learn from mistakes



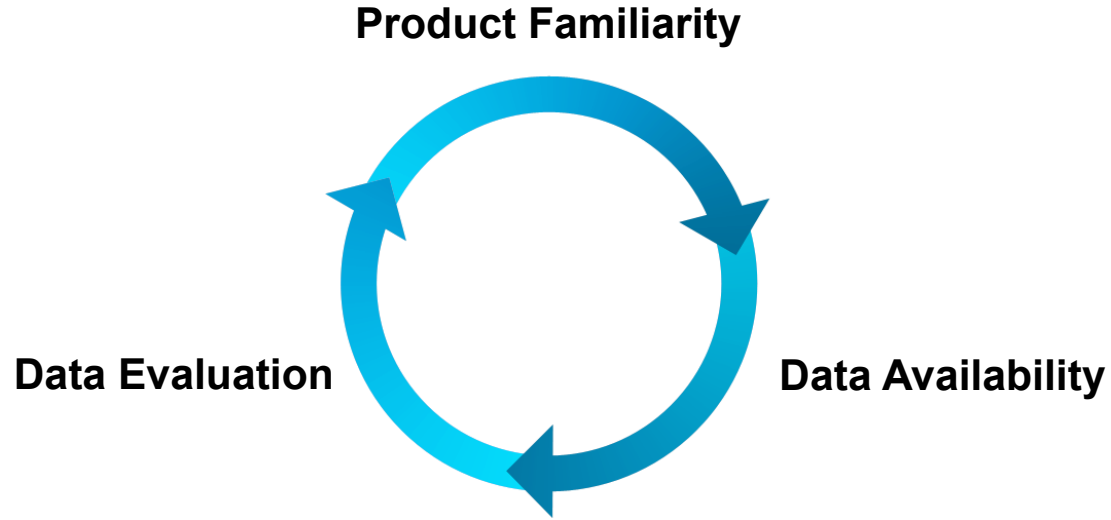
Building A Data Driven Development Process



.conf2016

splunk >

Prerequisites



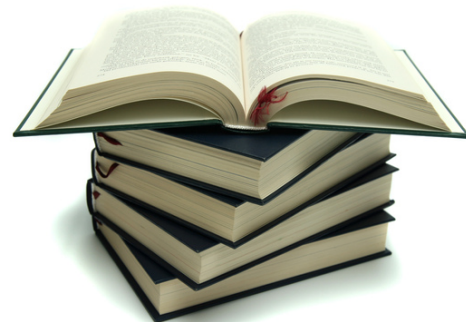
Product Familiarity

1. Know your product inside and out
 - Especially the “Happy Path”
2. Understand its strengths and weaknesses
3. Be familiar with the competition



Record Everything

1. Prioritize your happy path
2. Record as much data as possible
3. You never know what might be important in the future
4. Do it soon...



Evaluate Data

This is going to seem a little grim but...

1. Question everything
2. Trust reluctantly
3. Leave no room for doubt

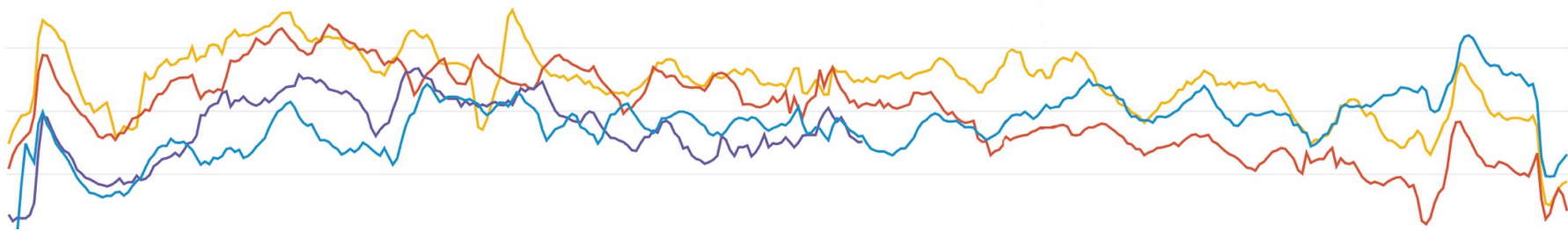
“The temptation to form premature theories upon insufficient data is the bane of our profession.”

– Sherlock Holmes



Implementing The Process

1. Develop historic baselines metrics and Key Performance Indicators (KPIs)
2. Determine relevant metrics for specific code deployments
3. Monitor KPIs before, during, and after deployment
4. Distinguish between short term monitoring and long term reporting



Review And Evaluate

1. Determine whether the project was successful
2. Were the selected KPIs appropriate?
 - Do we need new/better/different ones?
3. What questions should we have asked up front?
4. Did we adequately understand the user population?
5. **How successful was it?**

Expected	Actual	Diff	Change %
2,041,205	2,140,580	+99,375	+4.9%

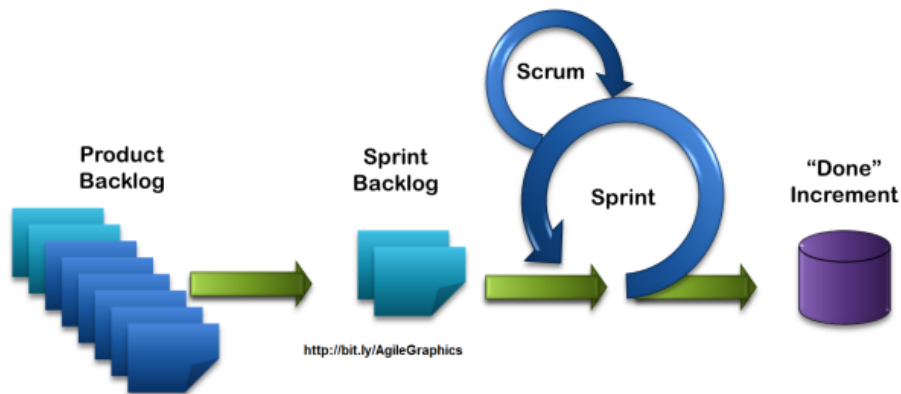
An Internal Case Study: Revenue Impacting Code Deployments

.conf2016

splunk >

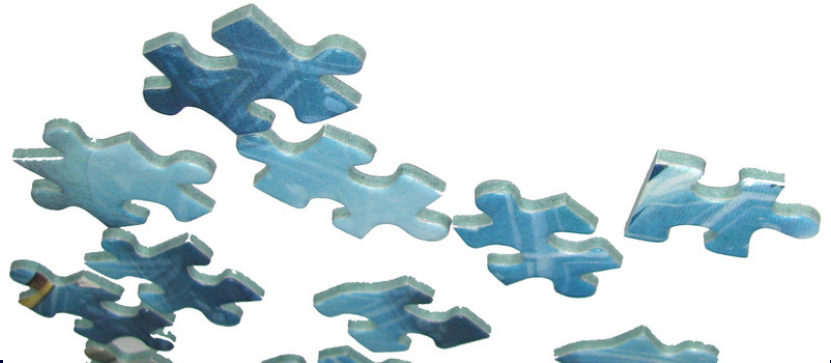
Background

1. ICF Technology uses Scrum for agile development
2. Several product teams releasing code to production
3. We monitor business and system data streams
4. On Average: 2 incidents per month



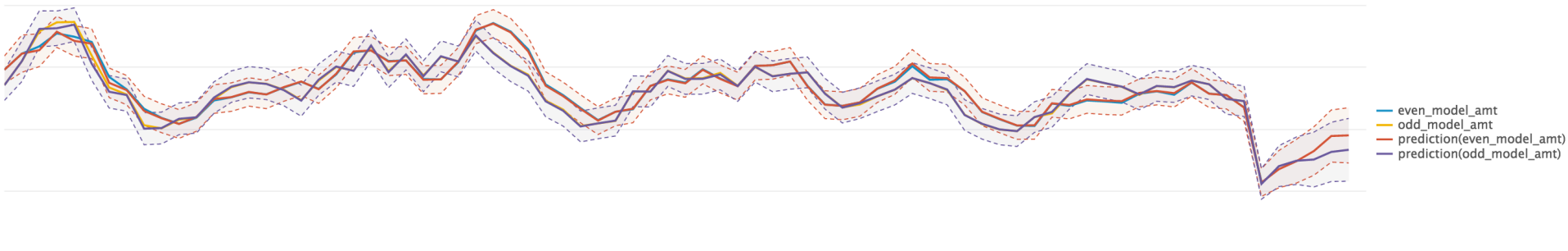
The Problem

1. Not enough visibility into impact of code deployments on production environment
2. Our Streaming Group kept having to delay or revert code releases until revenue “recovered”
3. Needed a way to determine that deployed code wasn’t breaking production environment



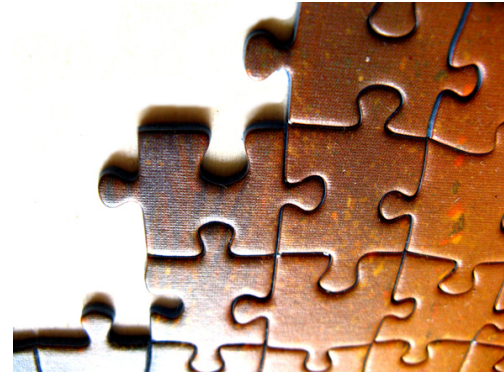
Analyze The Problem

1. Met with subject matter experts
2. Determined mechanism to deploy code to identifiable populations
3. Analyze viability of odd vs. even performerID deployments



The Proposed Solution

1. Developed dashboard to monitor code impact on populations
2. Creates baselines before each deployment
3. Determines impact during deployment
4. Allows evaluation success after appropriate period of time



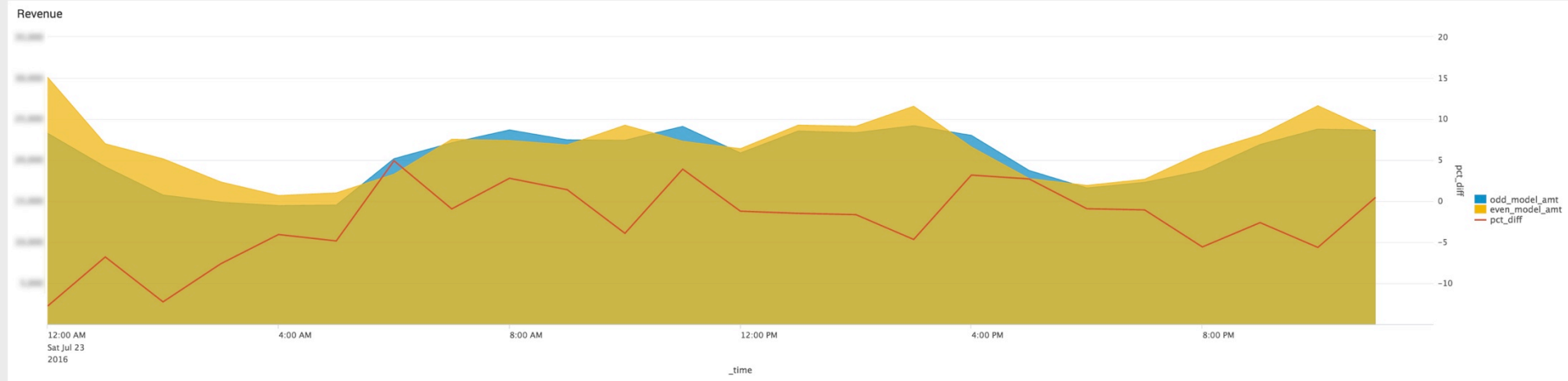
Evaluation Dashboard

Odd/Even Performer A/B Testing

Edit More Info Download Refresh

Time Frame: Yesterday | Time Span: 1 Hour | Device/Stream Type: All | Submit

- ### Panel Descriptions
- Use the Time Frame selector above to define an arbitrary and fixed time to analyze with the dashboard.
 - The Time Span selector allows you to get more granular with the time period.
 - The Device/Stream Type selector affects all but the top line of reports. Type differences can be found below:
 - **Android Tablet** - Smartphones using the iOS operating system
 - **Android Phone** - Smartphones using the iOS operating system with the iPhone browser and Smartphones web using the iOS operating system
 - **iOS Tablet** - Tablets, devices using the iOS operating system with the iPhone browser and Smartphones web using the iOS operating system
 - **iOS Phone** - Smartphones using the iOS operating system
 - The pct_diff data represents the pct_diff score relative to current vs. week ago.
 - The revenue data used to add the odd performance and the revenue data used to add the even.

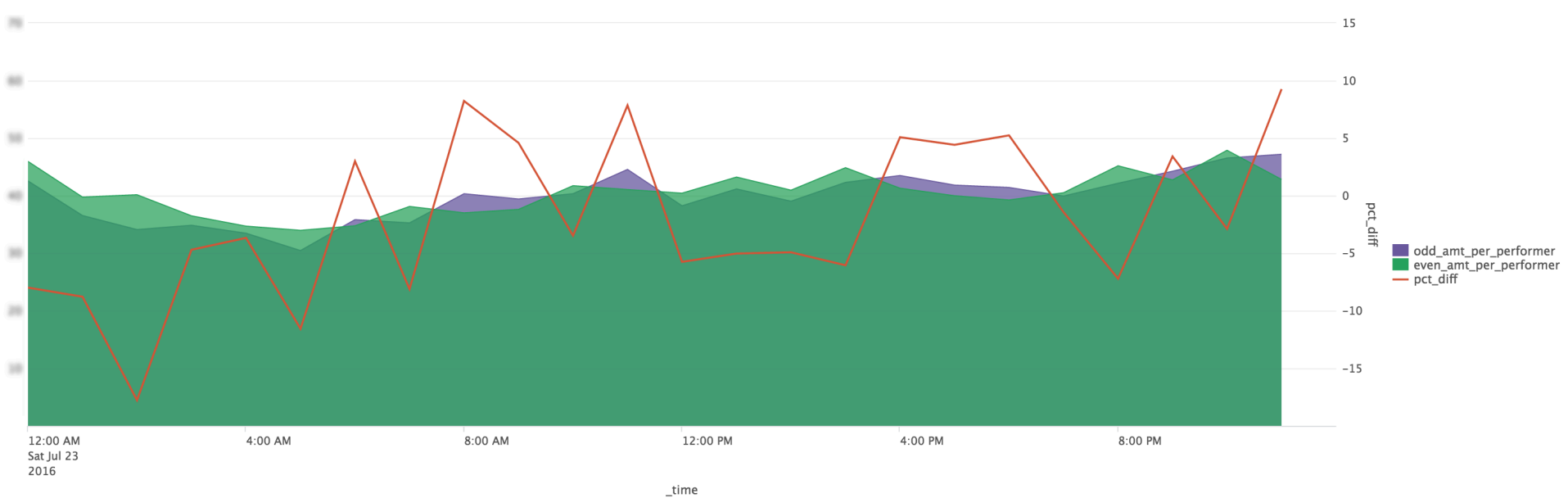


Revenue Totals for Selected Time Period

odd_model_amt	even_model_amt	week_ago_odd_model_amt	week_ago_even_model_amt	pct_diff	week_ago_pct_diff
10000.00	10000.00	10000.00	10000.00	-2.42	-3.04

Revenue Per Performer

Revenue Per Performer



Revenue Trends for Selected Time Period

1h ago

odd_amt_per_performer

even_amt_per_performer

week_ago_odd_amt_per_performer

week_ago_even_amt_per_performer

pct_diff

week_ago_pct_diff

39.86

40.58

41.72

41.81

-1.81

-0.22

What We Do Today

1. Each release is now evaluated with A/B test dashboard
2. Code deployments are first released to half the performer population
3. After a few hours, revenue is compared
4. If everything looks good, code is deployed to remaining population



Review And Evaluate

1. Determine whether the project was successful
2. Were the selected KPIs appropriate?
 - Do we need new/better/different ones?
3. What questions should we have asked up front?
4. Did we adequately understand the user population?
5. **How successful was it?**

Expected	Actual	Diff	Change %
2,041,205	2,140,580	+99,375	+4.9%

Questions?

THANK YOU

.conf2016