Linking Together Dev, Ops, and Biz Using Splunk

Takumi Fujiwara, Yokogawa Electric Corporation
Rie Tokita, Macnica Networks, Splunk Architect
Takashi Komatsubara, Splunk Senir Parner Sales Engineer

October 2018 | Version 5.1

TAKUMI FUJIWARA

takumi.fujiwara@jp.yokogawa.com

Yokogawa Electric Corporation, Japan

splunk> .conf18

RIE TOKITA
tokita-r@macnica.net

Macnica Networks

TAKASHI KOMATSUBARA
tkomatsubara@splunk.com

Splunk Japan

Biography

- Department
  - IA Systems & Service Division
    Systems Development Center, Systems Software Technology Department
    devops Group Leader
- Group Mission
  - Infrastructure Architect/Operation for Software Products for Windows・Linux and Execution of Build, Packaging, Installer Development
- Length of Experience in DevOps
  - 5 Years

Yokogawa Electric Corporation

Established: 1915

Annual Sales: $3.7B (FY2017)

Overseas Sales: 67.9%

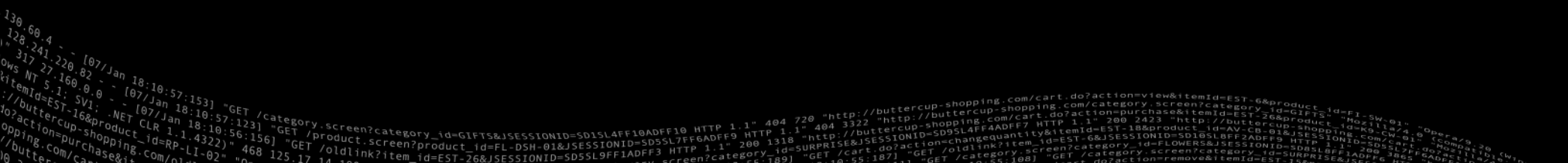
Locations: 112 WW, 59 Countries

Employees: 20K

Business Domain: Measurement, Control and Information

Customer’s industry sector: Oil, Chemical, Gas, Electric Power, Steel, Paper, Pharmaceutical, Foods

YOKOGAWA ◆ Co-innovating tomorrow™



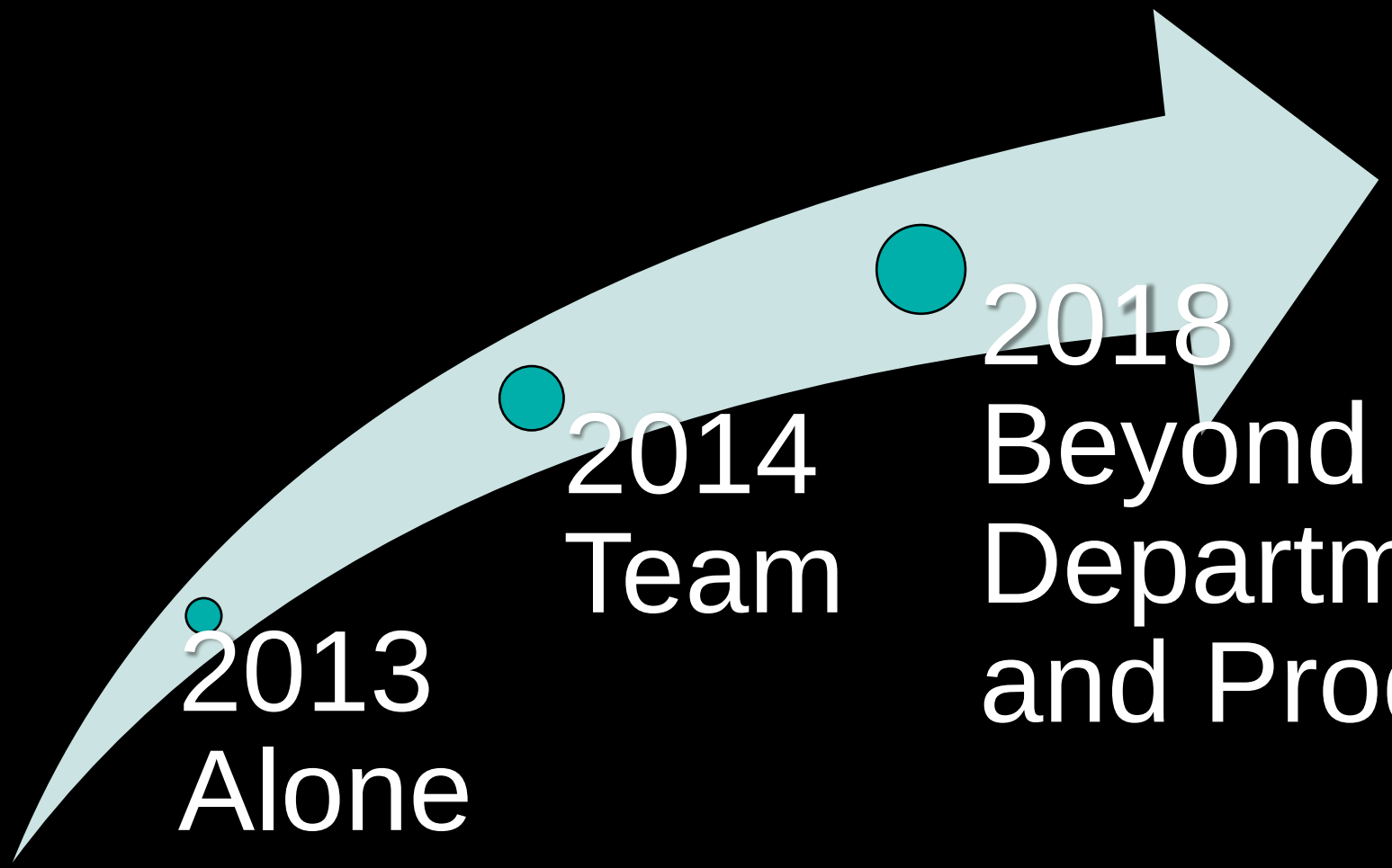
Agenda

- Looking back my five-year journey with DevOps
- Software Development Data Analysis with Splunk
- Summary

Looking Back Over My Five-year Journey With DevOps

splunk> .conf18

Our Timeline for DevOps Activities

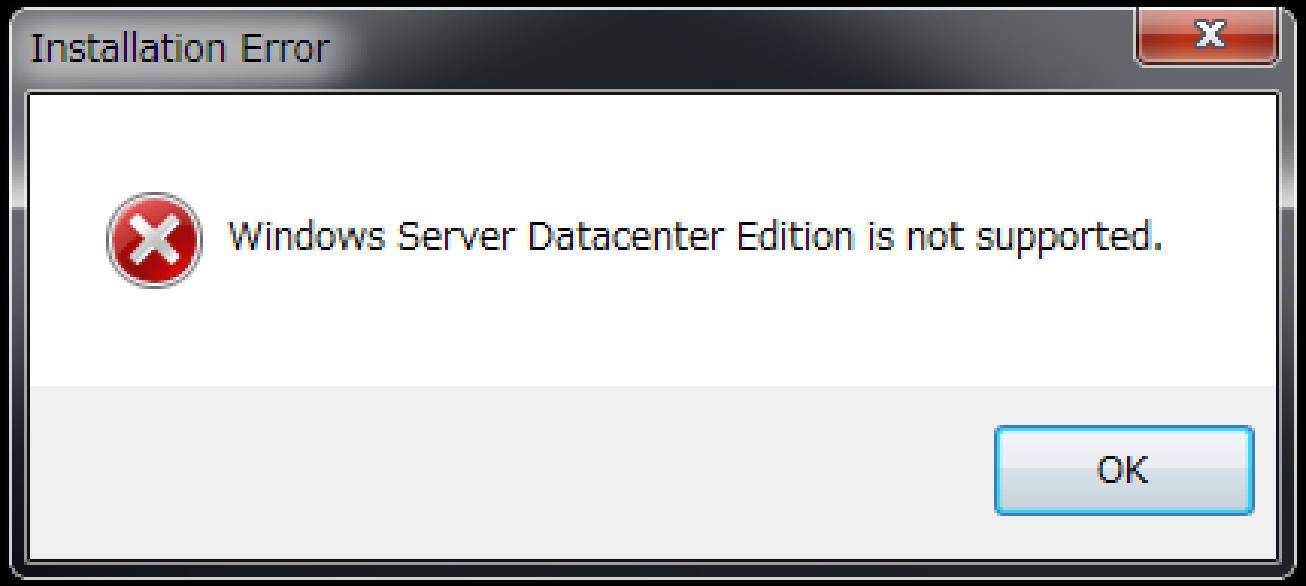


2013
Started DevOps Alone

Transferred to Another Department…

- After transfer to another department, worked as a tester in order to understand the products
- In charge of testing the products that require over 5 hours to install

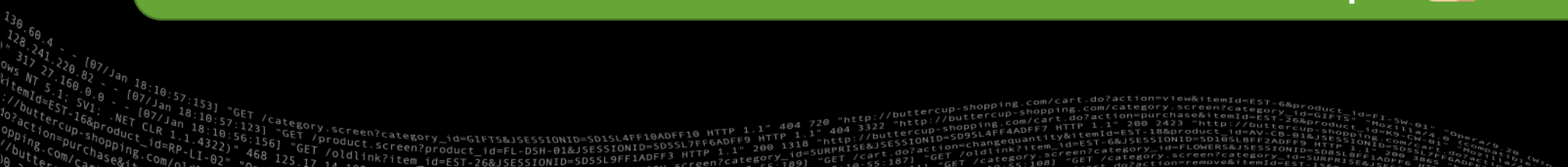
5 Hours Later…



Searching For An Automated Installation

- Automating OS and application installation with Chef
  - Successful implementation of automation
- Saved months of (internal) installation time
  - Process for Software Development
  - Shipping operation in Domestic, Overseas

Created the opportunity for understanding the effectiveness of infrastructure as Code and DevOps 👍🏻

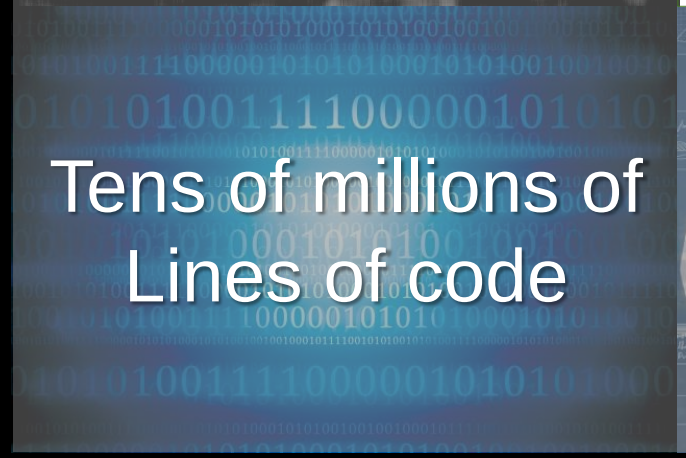


2014 Team Effort on DevOps

splunk> .conf18

The Project We Were Carrying Out

Windows



The Condition of Product Build Operation

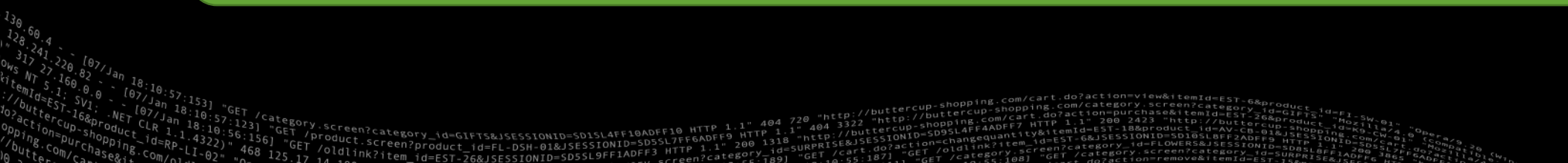
- Takes 24 hours from the start to the end of Build
- Procedure for Manual Build Operation was scattered
- Performance Control for Build-Task was poor

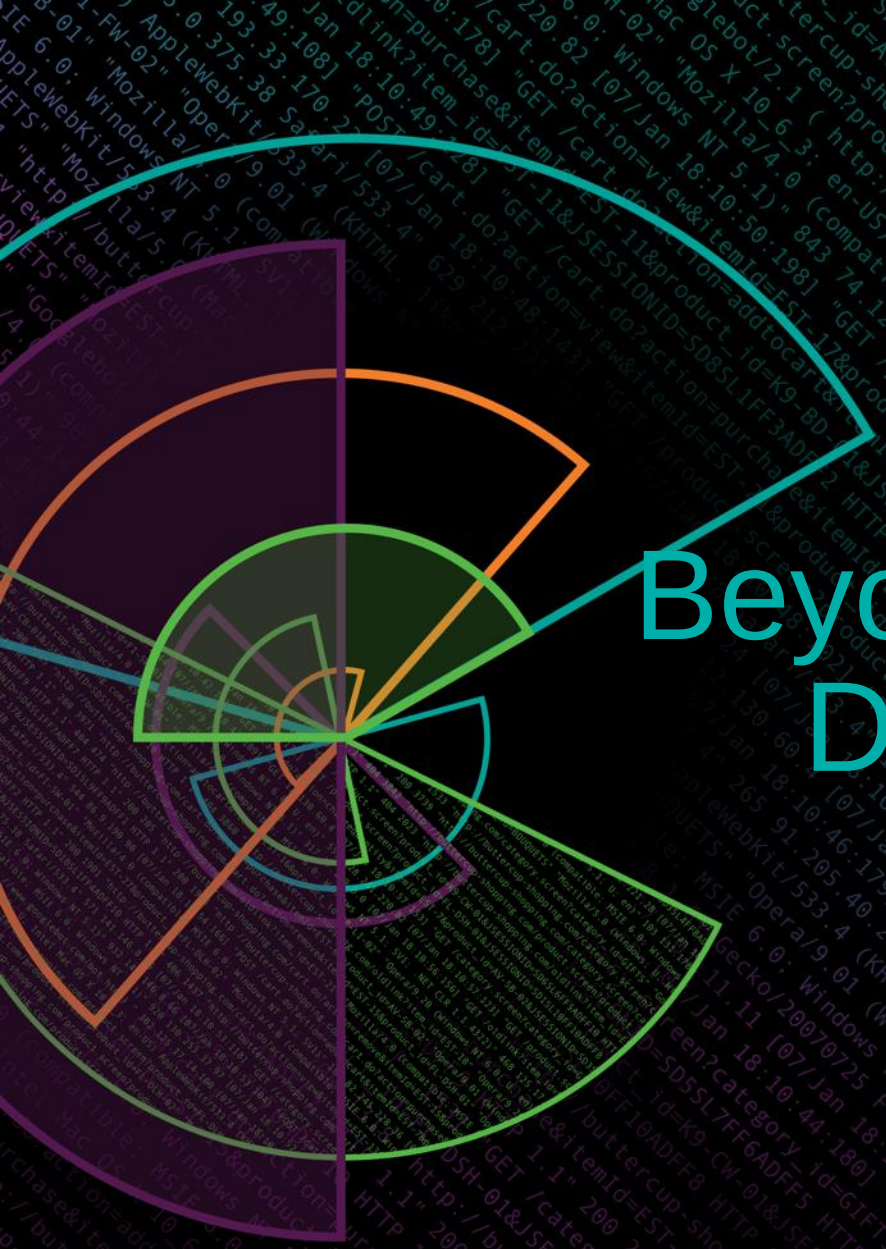
Build was slow, time-consuming operation

In Order to Actualize DevOps in Build Product Operation

- Automation and Optimization of Build Task by CI Tool
- Refactoring Build System
- Automated testing of Build results
- Improving the time-consuming transaction
- Utilizing Virtual Machines, virtual container technology, cloud Service

Result: full automation of the entire process for build and reducing the required time from 24 hours to 5 hours 🎉





2018
Beyond The Boundary of the Department and Products

splunk> .conf18


Start sharing the knowledge of DevOps beyond the department and products within the organization

DevOps Activities: Next Step

- Automated Infrastructure: Done
- Shared Version Management: Done
- One-Step-Build and Deploy: Done
- IRC and IM Bot: Done
- Shared Metrics: Next

splunk> Start changing and sharing the metrics 🎉

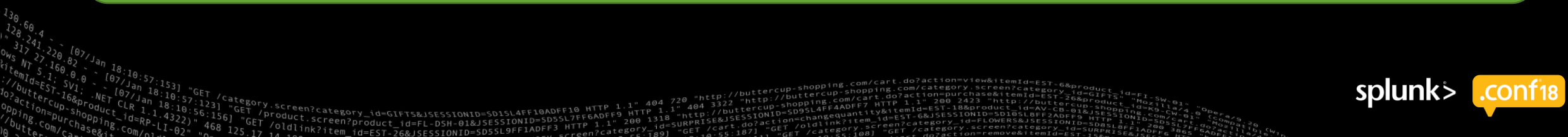
Software Development Data Analysis with Splunk

splunk> .conf18

My First Encounter With Splunk

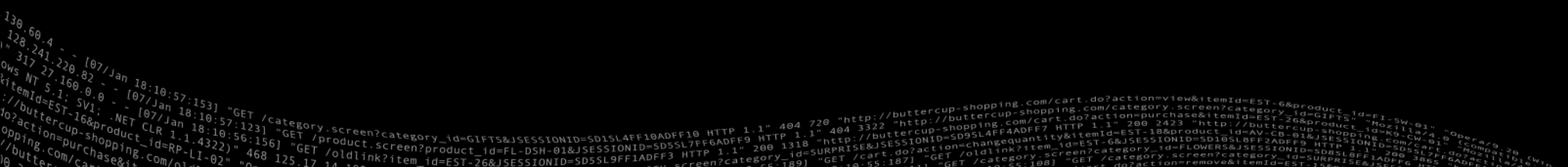
- Splunk workshop held in our office
  - Focused on Security
- Input data related to Software development as a trial…

splunk> Impressed by how easily data can be visualized 😆



Types of Software Development Data for Analysis

Types	Descriptions
1. Software Development Metrics Data	Changes in the number of lines of code Complexity
2. Activity Log of developers related to software development	Change History of Source code Man-hour
3. Defect information related to Software development	Content of Defect Cause Modified Content Date of Open and Close
4. Logs from each tool	Build, CI Tool, Automated Test

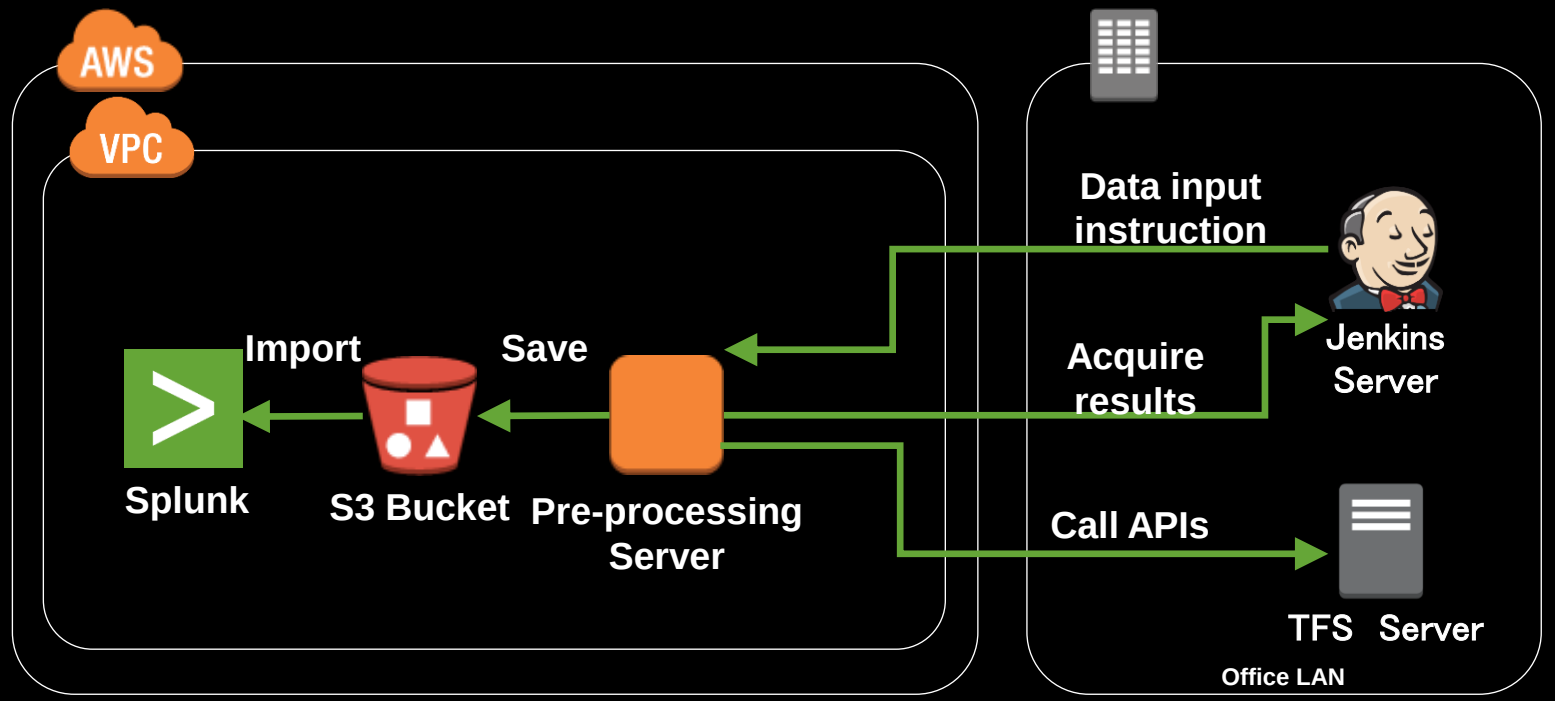


Ideal Environment for SW Development Data Analysis

- Flexible environment which is not constrained by metrics acquisition tools
- Able to combine different types of data
- Automated data collection

Realized an ideal environment with splunk> !

Flow of Development Data for Analysis



Automatically import the most recent data for analysis

List of Data

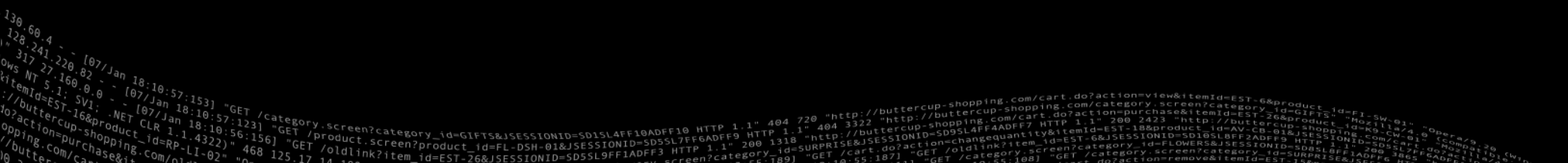
Source Type	Description	Tools
1. Method metrics	Metrics Information by Function Unit	TFS API & Source Monitor
2. Project details	Metrics Information by File Unit	TFS API & Source Monitor
3. Check in records	Logs of code modifications performed by the developer	TFS API
4. Issue tickets	Product Defect Information	Issue tracking system
5. Source similarity	Code Duplications present in source files/ violations of DRY Principle	TFS API & Simian
6. Fortify results	Security Static Analysis Result Information	Fortify
7. Issue key phrases	Defect of Product related to Key Phrase information	AWS Comprehend(NLP)
8. CI Tool logs	CI tool Logs	Jenkins

The Format of Data File

- File format
  - CSV
- Why?
  - Easy to edit its contents
  - Easy to import to Splunk

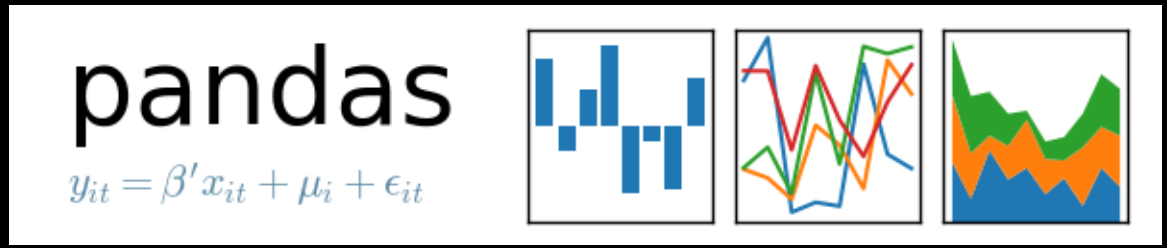
The Tasks of Pre-processing

- Delete unnecessary columns
- Add columns
  - Software version information in which the data was generated
  - Owner of each records
- Deduplication records



The Way of Data Pre-processing

- Currently, we are using a Python Data Analysis Library (Pandas).
  - https://pandas.pydata.org/
  - You can process large data quickly and conveniently.
- Before using Pandas, we had used PowerShell script for the tasks but …
  - PowerShell isn't good at processing data in CSV files.
  - For data pre-processing, You have to implement data manipulations with low-level cmdlets.
  - PowerShell requires long time for data pre-processing than Pandas.



Data Pre-processing Using Pandas

```
import pandas as pd
df = pd.DataFrame
target_df = df.from_csv('target.csv')
# Add the version column and set a value to every records.
target_df['Version'] = 'V1.01'
# Dedupe records
target_df.drop_duplicates()
# Export results to a csv file.
target_df.to_csv('result.csv')
```

target.csv

FileName	Owner
A.Cpp	Team A
B.Cpp	Team B
C.Cpp	Team C
D.Cpp	Team B
A.Cpp	Team A

result.csv

FileName	Owner	Version
A.Cpp	Team A	V1.01
B.Cpp	Team B	V1.01
C.Cpp	Team C	V1.01
D.Cpp	Team B	V1.01

splunk> .conf18

1.Visualization of Software Development Activities

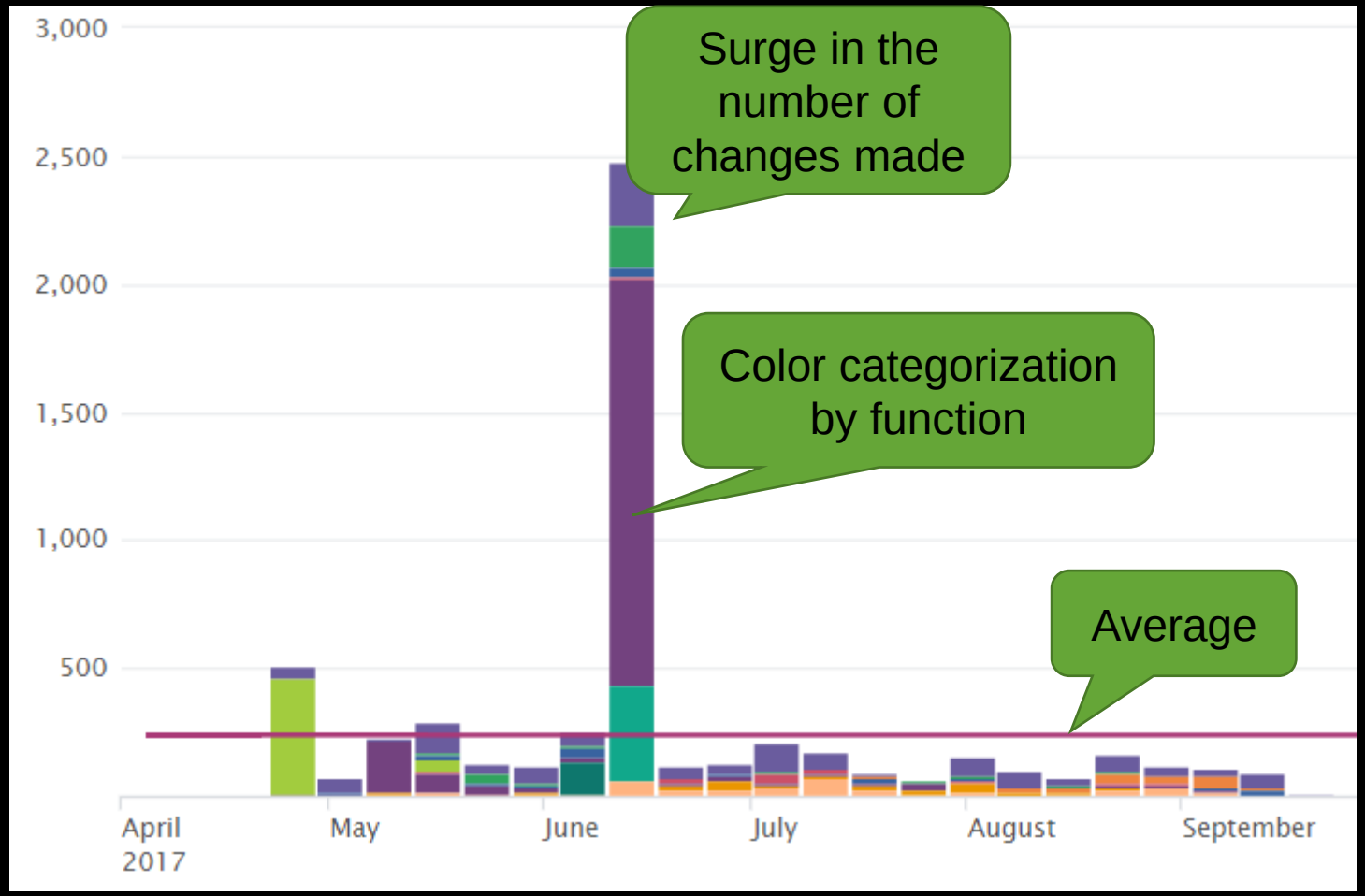
1. Visualization of Software Development Activities

- Target data:
  - Log information that records all the modifications made to the source file by each developer
    - Modified files and timestamp information in the Version Management System (TFS)
- Aggregation Method
  - Tally the number of times each engineer makes changes to the source file per function

1. Visualization of Software Development Activities

- Data can be created using TFS REST API (for VSTS, TFS2015 or later)
  - For tfvc service
    - Changesets – Get Changesets for the tfvc service http://bit.ly/2NFMW0S
    - Changesets – Get Changeset Changes for the tfvc service http://bit.ly/2NBnV6S
  - For git service
    - Commits – Get Commits for the Git service http://bit.ly/2NEpEIH
    - Commits – Get Changes for the Git service http://bit.ly/2NEqaGD
- If using older TFS servers (2008, 2010, 2012 or 2013)
  - Please use the PowerShell cmdlets which are included in Team Foundation Power Tools.

1. Visualization of Software Development Activities



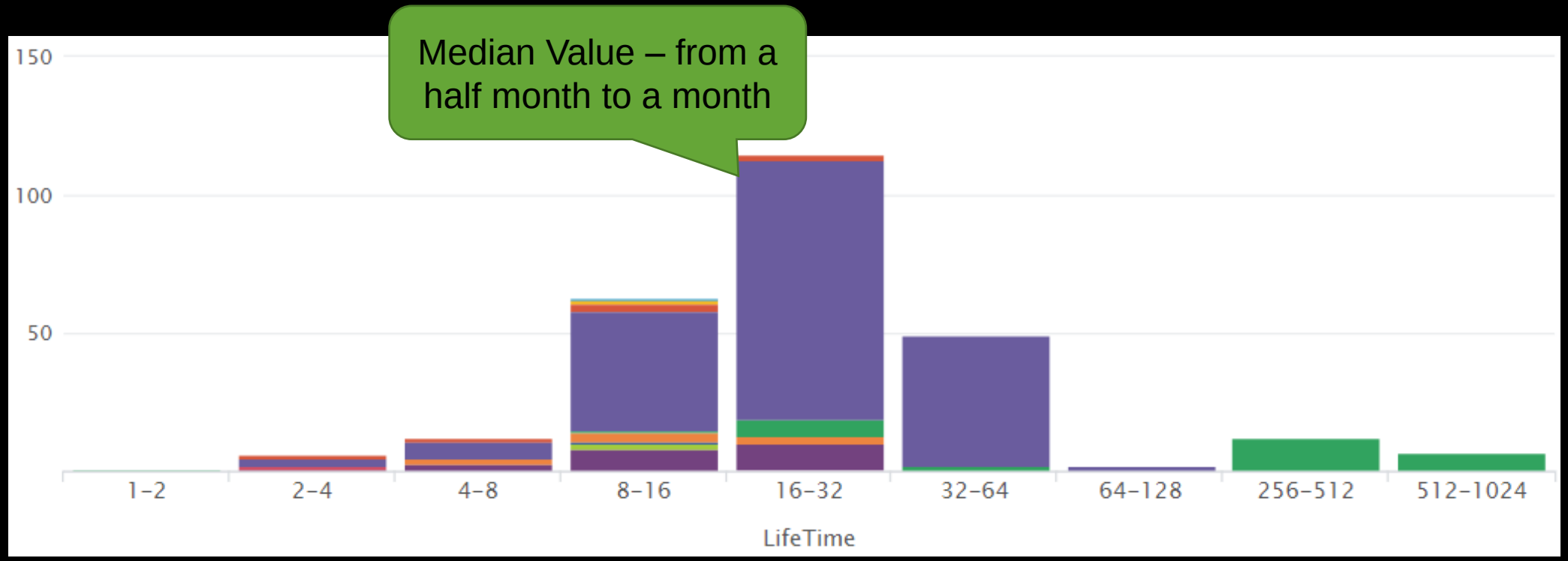
2.Incident Tickets Visualization

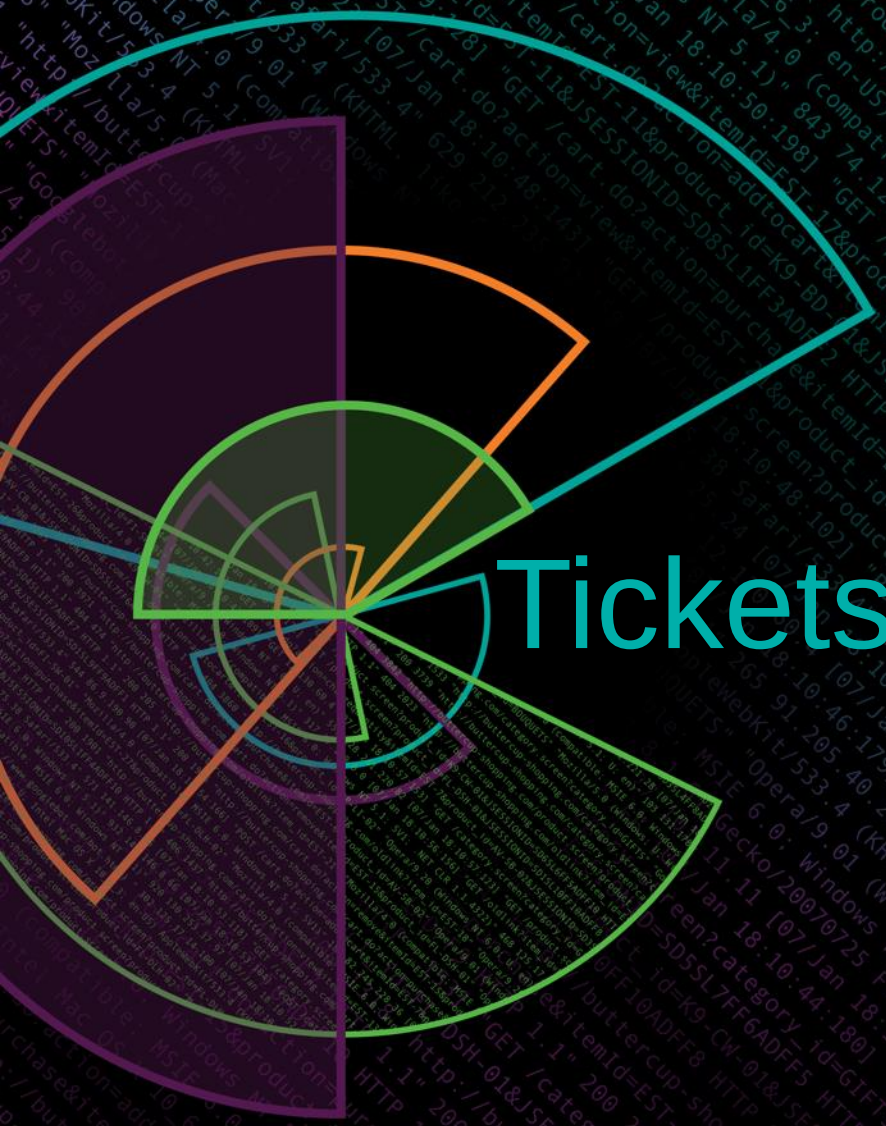
2.Incident Tickets Visualization

- Target data:
  - Product Trouble Ticket Information
    - Information about Open/Close Product Trouble Ticket
- Aggregation Method
  - Aggregate the duration of Open/Close Tickets related to product trouble per function
- Reference
  - Create based on the request in the meeting with the product manager

2.Incident Tickets Visualization

Logs from in-house Ticketing System





3.Number of Incident Tickets Related to Source Files

3. Number of Incident Tickets Related to Source Files

- Objective
  - Identify Source File where the troubles are concentrated
- Target data
  - Logs of modifications made by developers
    - Commit messages in the Version Management System (TFS), modified files, time stamps
    - Extract and aggregate the Number of Trouble Ticket that has been modified from the comment.

3. Number of Incident Tickets Related to Source Files

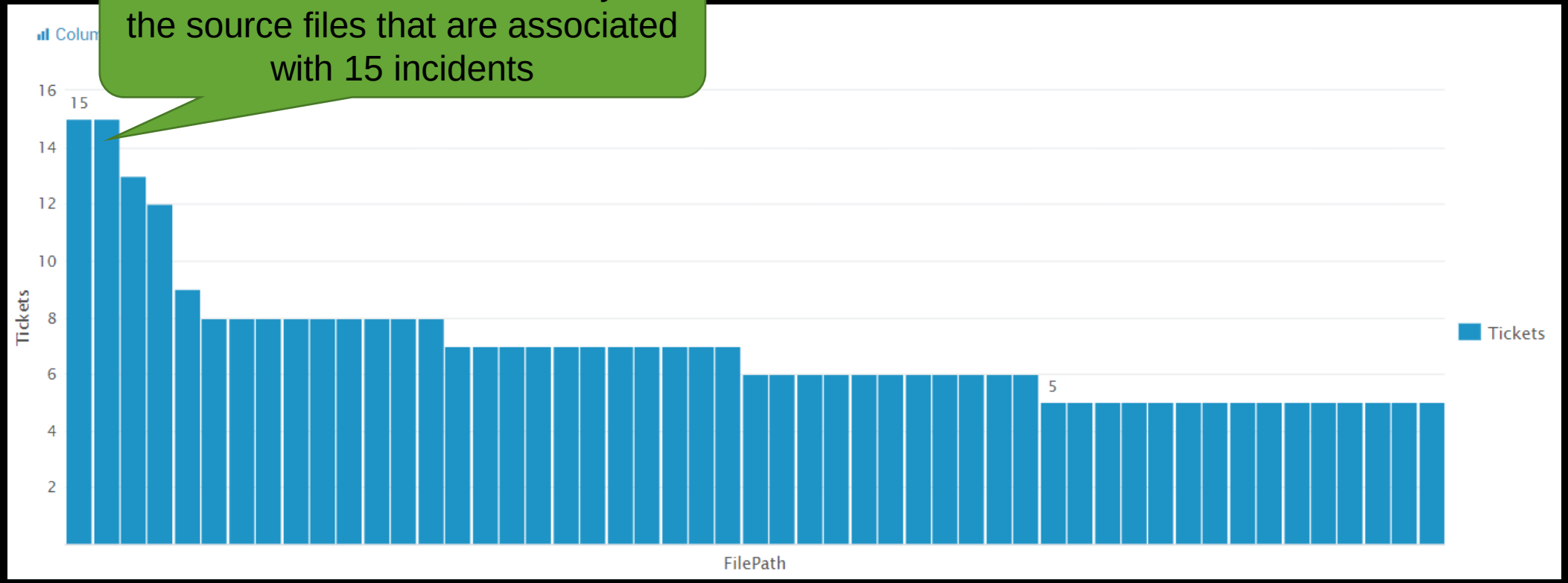
```
index=my_index sourcetype=commit_records
| regex Comment="IssueID"
| rex field=Comment "IssueID[:-]?[ ]*(?<IssueNum>[a-zA-Z]?[0-9]+)"
| stats count(FilePath) as Tickets by FilePath
| where 3 <= Tickets
| sort -Tickets
```

2018/01/26 19:00 T.Fujiwara IssueID: **12345678** Deadlock Defect Modification

Extract this portion as the IssueNum column

3. Number of Incident Tickets Related to Source Files

Immediate action is necessary on the source files that are associated with 15 incidents



4. Ai-assisted Auto-ticket Assignment

4. Ai-assisted Auto-ticket Assignment

- Target data
  - Product Trouble Ticket Information
    - Cause of Trouble and Comment on measures, Person in charge of modification

- Objective
  - Current situation
    - Incident ticket issued ➔ Supervisor assigns the ticket to an agent ➔ Owner is determined
  - What we wanted to achieve
    - Incident ticket issued ➔ Auto-assign the ticket to an agent

4. Ai-assisted Auto-ticket Assignment Case Description

- Hypothetical inquires:
  - Inquiries in the past
    - Product a crashed. ➔ Assigned to agent A
    - Product b's speed of processing is slow. ➔ Assigned to agent B
    - Want to upgrade Product c ➔ Assigned to agent C
  - New inquiry
    - Product b crashed. ➔ Who should the ticket be assigned to?

It's important to identify the "keyword"

4. Ai-assisted Auto-ticket Assignment
~Two Trials~

Splunk Machine Learning Toolkit – MLTK
https://splunkbase.splunk.com/app/2890/)

1st trial

Issue 123

Content of Trouble Description

Incident Ticket

AWS Comprehend(NLP)

MLTK Model

Extracting Keyword

2nd trial

Issue 123

Content of Trouble Description

Incident Ticket

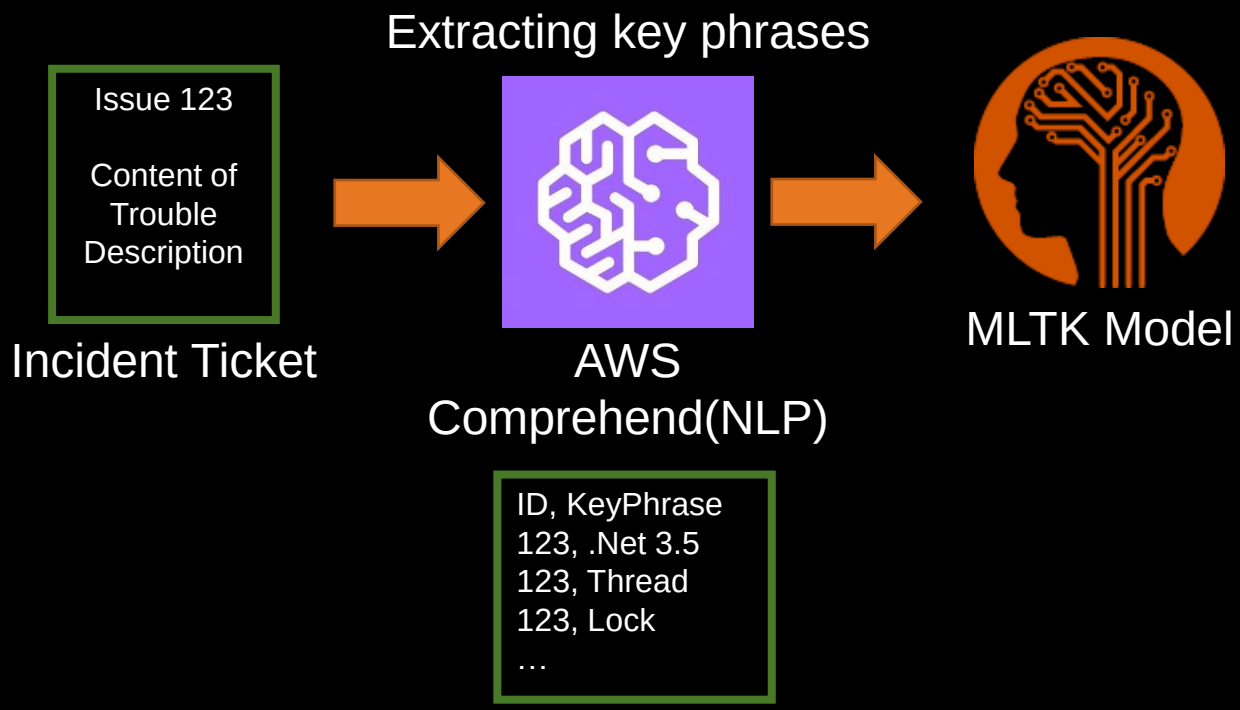
Preprocessing Steps

MLTK Model

splunk> .conf18

1st trial

4. Ai-assisted Auto-ticket Assignment
First Trial～ Extracting Key Phrases Using AWSC ～



1st trial

4. Ai-assisted Auto-ticket Assignment
~ Learning ~

Splunk

Issue 123

Trouble
Content
Description

Trouble Ticket

ID, KeyPhrase
123, .Net 3.5
123, Thread
123, Lock
…

Key Phrases

SPL

Owner	product_a	product_b	product_c
A	1	0	0
B	0	1	0
C	0	0	1
?	0	1	0

Input

MLTK Model

Output

Owner	product_a	product_b	product_c
A	1	0	0
B	0	1	0
C	0	0	1
B	0	1	0

1st trial

4. Ai-assisted Auto-ticket Assignment ~ SPL ~

```
sourcetype=“defect_ticket"
[| inputlookup UniqueVitalKeyWords
 | rename keyword as KeyPhrase
| fields KeyPhrase
| rename KeyPhrase as search]

| lookup UniqueVitalKeyWords KeyPhrase

| streamstats count as number

| mvexpand KeyPhrase

| eval Owner=Owner+":"+number

| chart count over Owner by KeyPhrase limit=0

| rex field=Owner "(?<Owner>[^:]+)"
```

- Extract only the events that contain a keyword
- Specify the keyword
- Markup the same events to expand the values into separate events in post-processing
- Expand multiple keywords into separate events
- Identify what in the expanded events is identical
- Create a matrix of the owner name and key phrases
- Display the name of the owner removing the number

1st trial

4. Ai-assisted Auto-ticket Assignment
~ Accuracy Of The Model ~

Precision	Recall	Accuracy	F1
0.77	0.68	0.68	0.72

70%+ accuracy

2nd trial

4. Ai-assisted Auto-ticket Assignment Second Trial～ TF*IDF ～

- TF*IDF
  - Data preparation process before text-mining "TF*IDF"
  - Avoid frequent words we can see in many data, pick up important words we can see several times
- Example
  - I like apple.
  - I read this English book.
  - You like this apple.
  - You read this Japanese book.

2nd trial

4. Ai-assisted Auto-ticket Assignment
~ Learning ~

Splunk

Issue 123

Trouble
Content
Description

Incident ticket

Input

MLTK TFIDF

Output

Owner	word_a	word_b	word_c
A	1	0	0
B	0	1	0
C	0	0	1
?	0	1	0

Input

MLTK Model

Output

Owner	word_a	word_b	word_c
A	1	0	0
B	0	1	0
C	0	0	1
B	0	1	0

2nd trial

4. Ai-assisted Auto-ticket Assignment ~ SPL ~

```
sourcetype=“defect_ticket"

| rex max_match=0 field=Comment "(?<word>[A-Z][a-zA-Z0-9]{2,})“

| search word!=""

| fit TFIDF word max_features=300 stop_words=english

| fields – word*
| table Owner, word*
```

Focus on words which consist of over 3 words and start from uppercase characters
And eliminate other words

stop_words specifies a language to eliminate preposition

2nd trial

4. Ai-assisted Auto-ticket Assignment
～ Accuracy Of The Model ～

Precision	Recall	Accuracy	F1
0.98	0.97	0.97	0.97

4. Ai-assisted Auto-ticket Assignment
~ Conclusion ~

- Conclusion ... TF*IDF
  - Powerful tool for text mining
  - Understand Limitations
  - Leverage already known/You-Know important words

Results of Software Development Analytics with Splunk

splunk> .conf18

Results of Software Development Analytics with Splunk

Cross-functional Collaboration based on Analysis

Visualization of Buried Issues

Proactive Detection of Issues

Utilization of AI using MLTK

How To Link Together Dev, Ops and Biz Using Splunk?

splunk> .conf18

The Areas for Which We Use Splunk

Core System

- SAP
- ARIBA
- BI

Security

- WAF
- Box

Software Development

- DevOps
- System Monitoring

Application

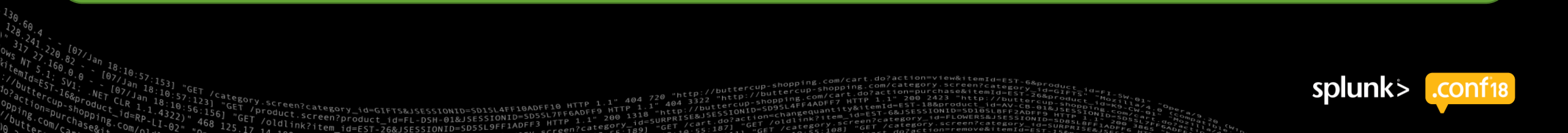
- Call Center
- Client Site Monitoring
- Install Information
- Member Site
- Product Life Cycle

splunk> .conf18

Linking Together Dev, Ops, and Biz Using Splunk

- Splunk is used in multiple divisions for various usecases.
- One of the big achievements is that we could make good collaborative working place to everyone by leveraging Splunk and bigdata.
- One of the example is quick feedback to dev team with analyzed voice communications between call center's agents and customers.

We will accelerate our DevOps by leveraging splunk>



Summary

Summary

- Splunk enables different points of view for development analysis
- Using Splunk to analyze development data enables faster decision making, support detection, and relation to the issue
- By sharing data, analytical results and insights by using Splunk, it promotes DevOps and encourages collaborations between roles such as Dev, Ops, and the business.