System Test Practice

Amy Li @ 2017/7/22

Forward-Looking Statements

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 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.

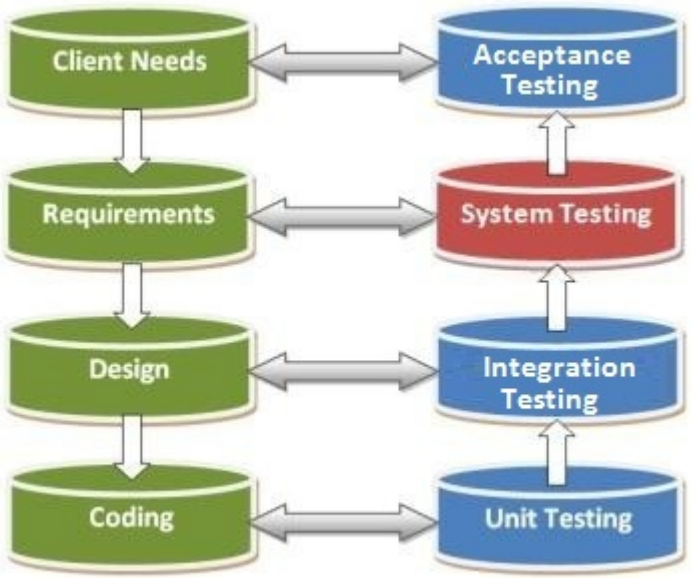
splunk> listen to your data

Agenda

- System Test in software testing
- Splunk introduction
- How do we do system test for Splunk
- A system test framework
- How to triage in system level test
- Summary

What is System Test

System Test
is testing of the software application as a whole to check if the system is complaint with the user requirements



System test Vs. Integration test

System test	Integration test
Both functional and non-functional testing are covered	Functional testing only
High level testing after integration test	low level testing after unit test
Black box test	Both black box and white box
Test cover external interface	Test only cover the inside modules

DaoCloud

IT大咖说 知识分享平台

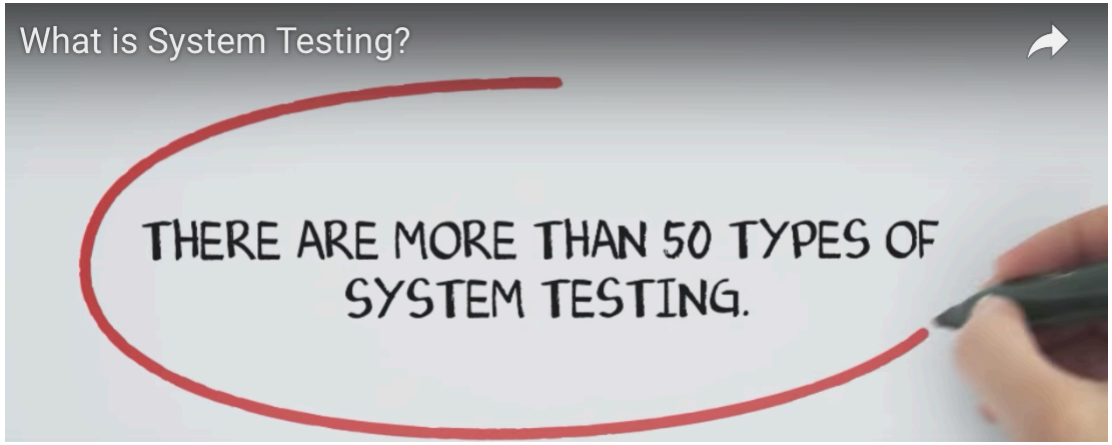
What should be covered in System test ?

splunk> listen to your data

Types of tests to include in system testing [edit]

The following examples are different types of testing that should be considered during System testing:

- Graphical user interface testing
- Usability testing
- Software performance testing
- Compatibility testing
- Exception handling
- Load testing
- Volume testing
- Stress testing
- Security testing
- Scalability testing
- Sanity testing
- Smoke testing
- Exploratory testing
- Ad hoc testing
- Regression testing
- Installation testing
- Maintenance testing[*clarification needed*]
- Recovery testing and failover testing.
- Accessibility testing, including compliance with:
  - Americans with Disabilities Act of 1990
  - Section 508 Amendment to the Rehabilitation Act of 1973
  - Web Accessibility Initiative (WAI) of the World Wide Web Consortium (W3C)



But …

DaoCloud

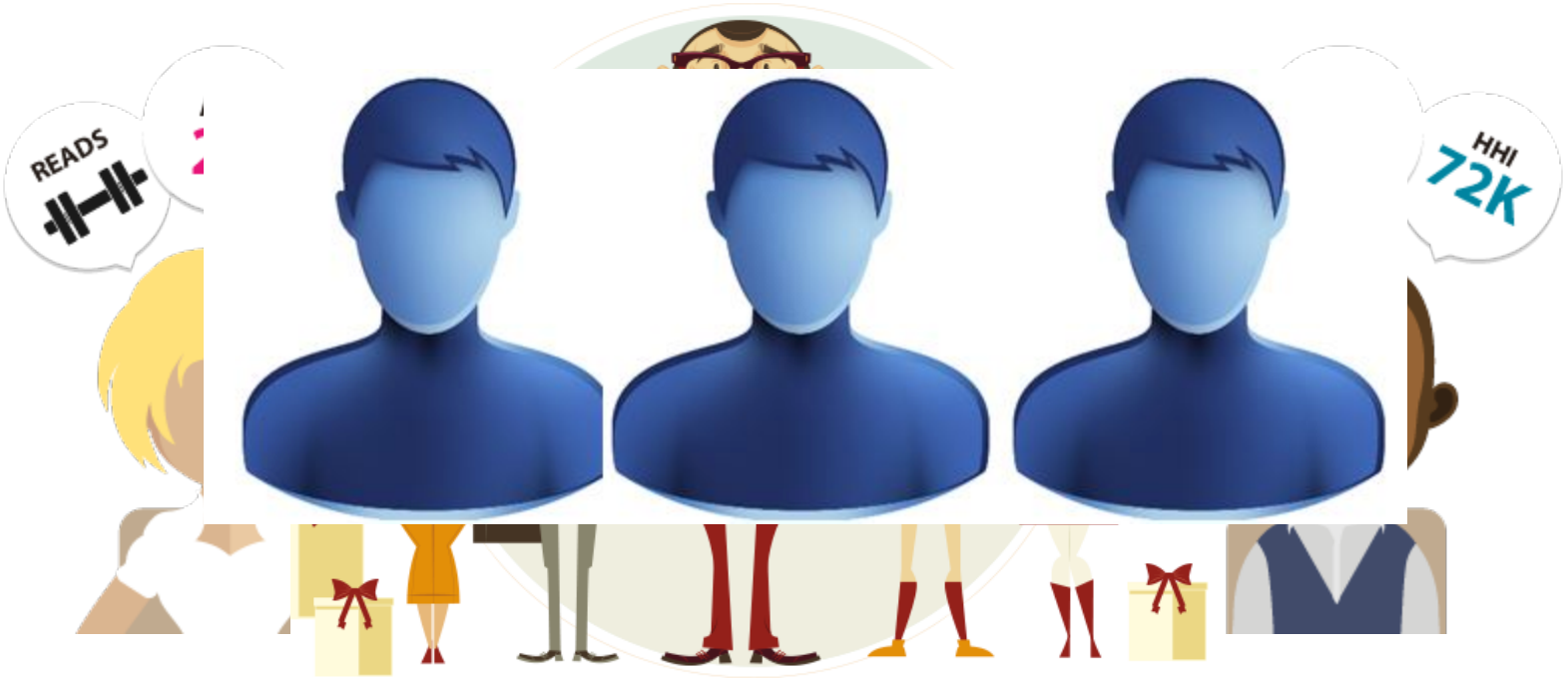
IT大咖说 知识分享平台

It Depends …

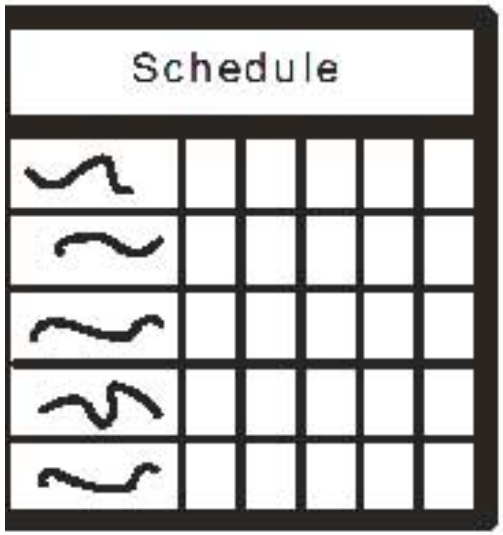
splunk> listen to your data

Product design and architecture

Product end-user



Resource and Schedule

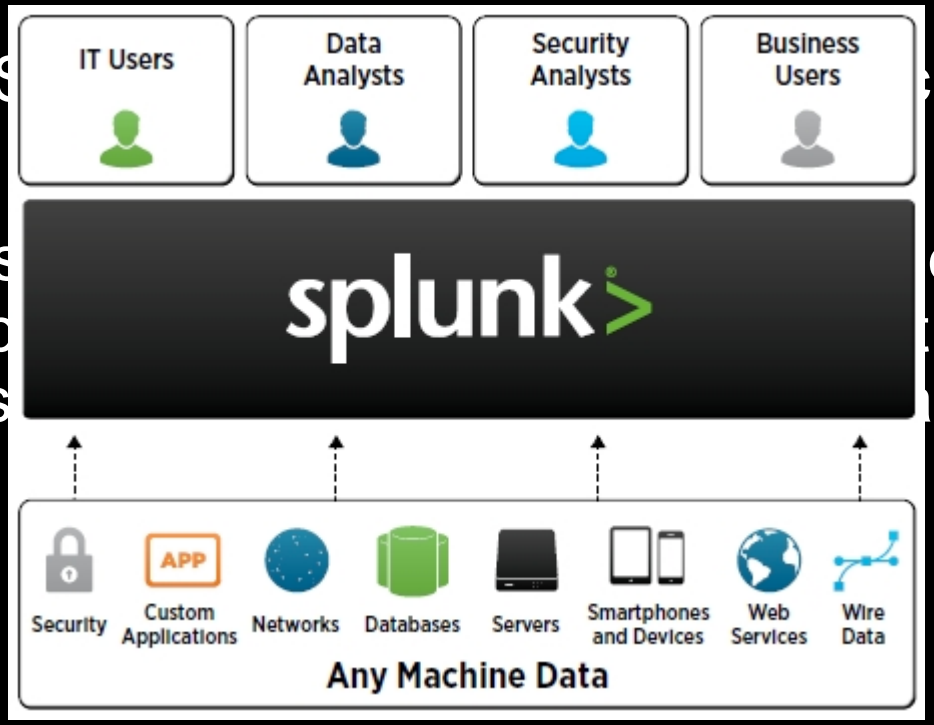


So, design your test according to requirements, and risk-driven

What is Splunk

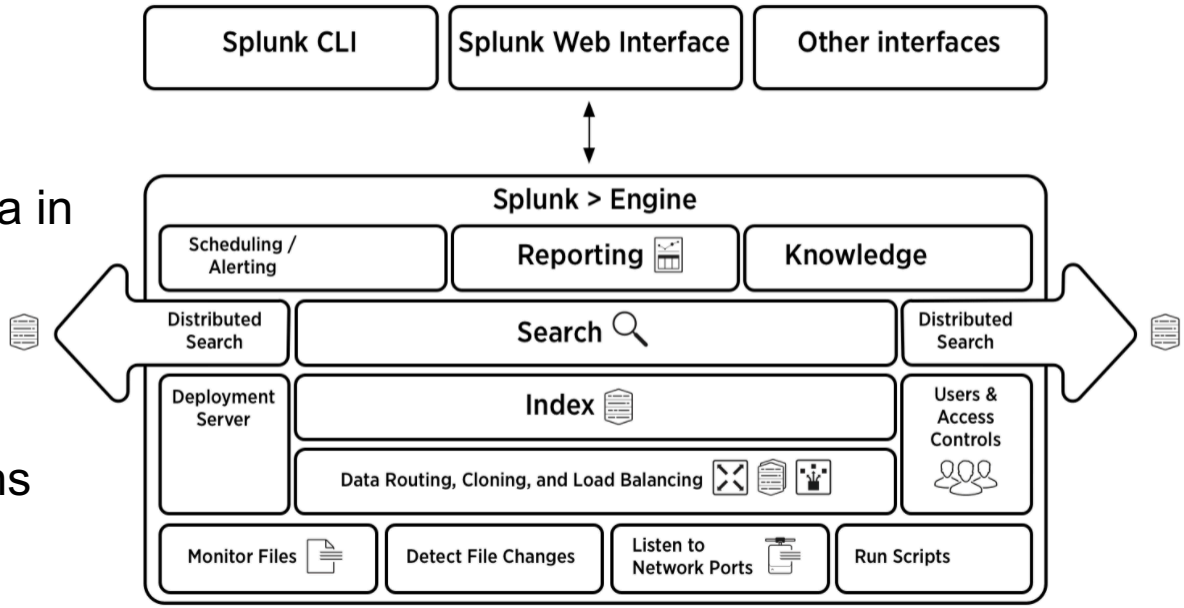
In simple words S hine data /logs

It‘s a powerful s ed to search, investigate, troub , and report on everything that's structure from



Splunk Architecture

- Distributed system
- High scalability
- Various interface to get data in
- Various Deployment
  - Cloud
  - On-Prem
- Release schedule: 3 months



System Test in Splunk

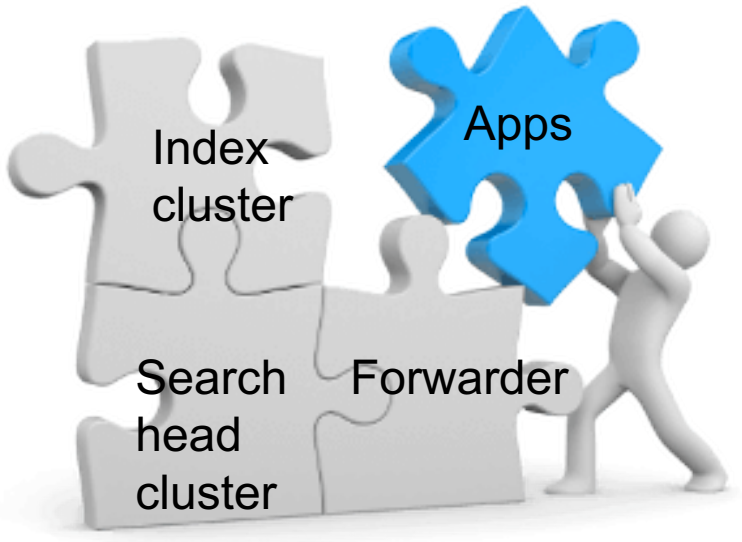
- User scenario Test
- Large-scale Test
- Reliability Test
- Scalability Test
- Stability Test
- Migration Test
- Interoperability Test
- Ad-hoc Test
- Framework development
- Tools development
- Customer case RCA

- Team setup 2 years ago
- 6 full-time employee

System Test in Splunk

User scenario Test – more like an integration test

- complex topology
- Complex configuration
- Certain volume of data



CHECK POINT

- ✓ Functionality is correct
- ✓ Performance is acceptable
- ✓ No crash or other fatal errors

System Test in Splunk

Large scale Test – Simulate large scale Customer scenario

- ✓ Deploy the product in large scale
- ✓ Manipulate the product with complex configuration and data model
- ✓ Populate data to the deployment with heavy load (26TB/7 days, 100+ scheduled search)
- ✓ Monitor the system in longevity scenario

- ✓ No System crash and Fatal error
- ✓ Resource usage under defined criteria
- ✓ Indexing rate 10% stable around the defined criteria
- ✓ Search metrics meet defined criteria

System Test in Splunk

Large-scale Test

Environment complexity
- Topology
- Configuration

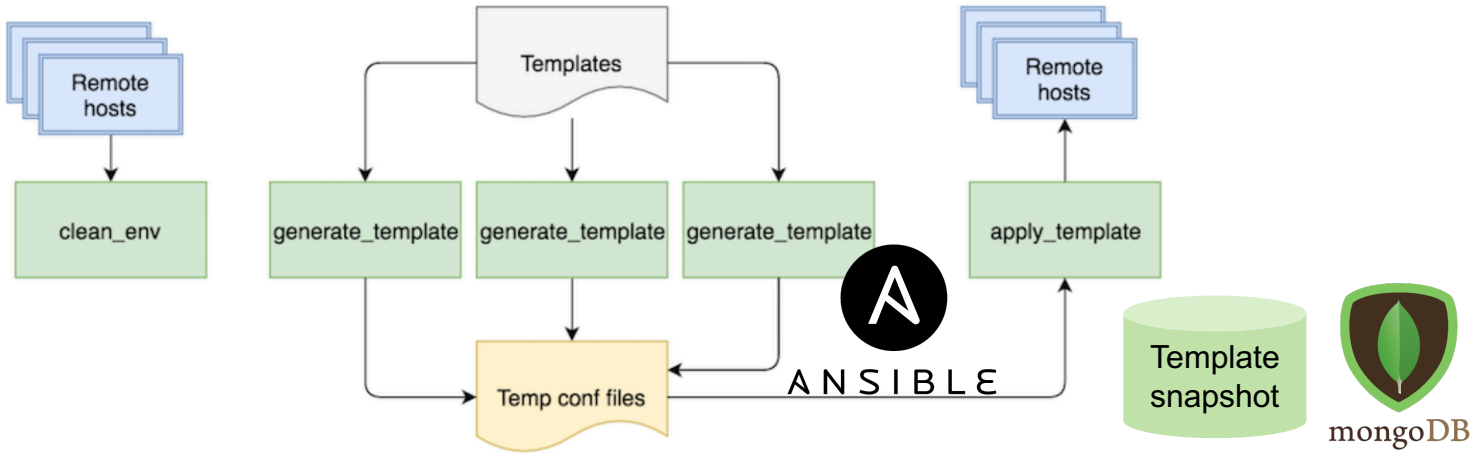
Data complexity
- Load
- Diversity
- Pattern

Behavior complexity
- Schedule
- Load distribution
- User diversity
- User access interface diversity

System Test in Splunk

Reliability Test – Explore the product with various configurations

- ✓ Configuration is infinite, but the test need to be limited
- ✓ Grouping the configurations into reasonable combinations and iterate them
- ✓ Light environment switch overhead



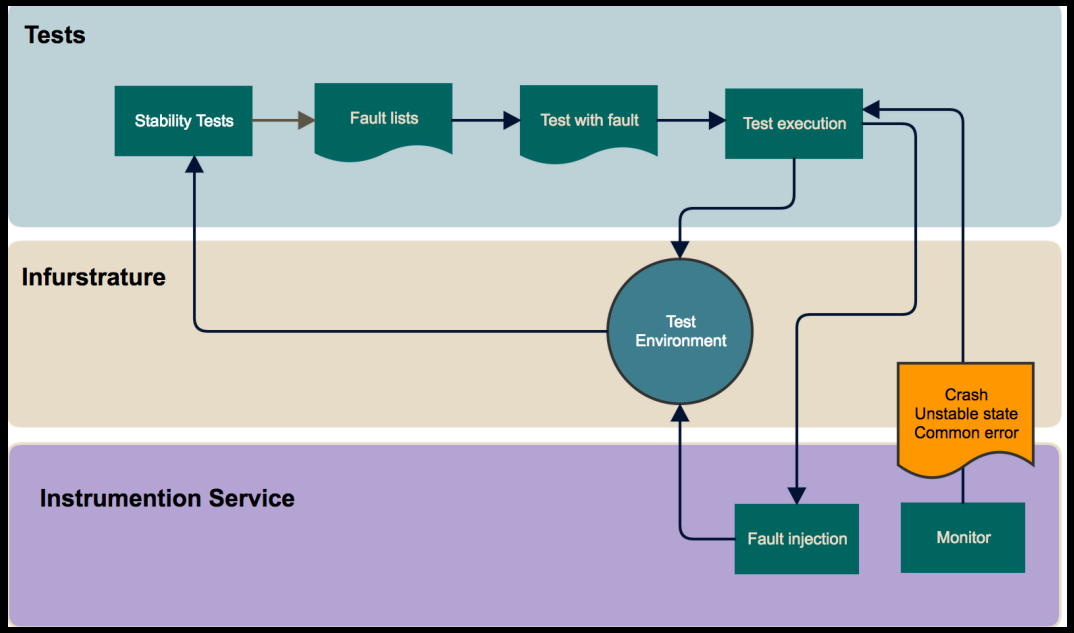
System Test in Splunk

Reliability Test

- ✓ No crash after environment switch
- ✓ Splunk Server is in health state
- ✓ No error in splunk logs
- ✓ Abnormal detect for snapshots (measure by distance with K-means)

System Test in Splunk

Stability Test – Evaluate product with limited resource or quota



Fault Injection

- ✓ CPU/Memory/Disk/Network
- ✓ Server down
- ✓ Incorrect configuration

Checkpoint

- ✓ No crash and fatal error
- ✓ Functionality could be recovered

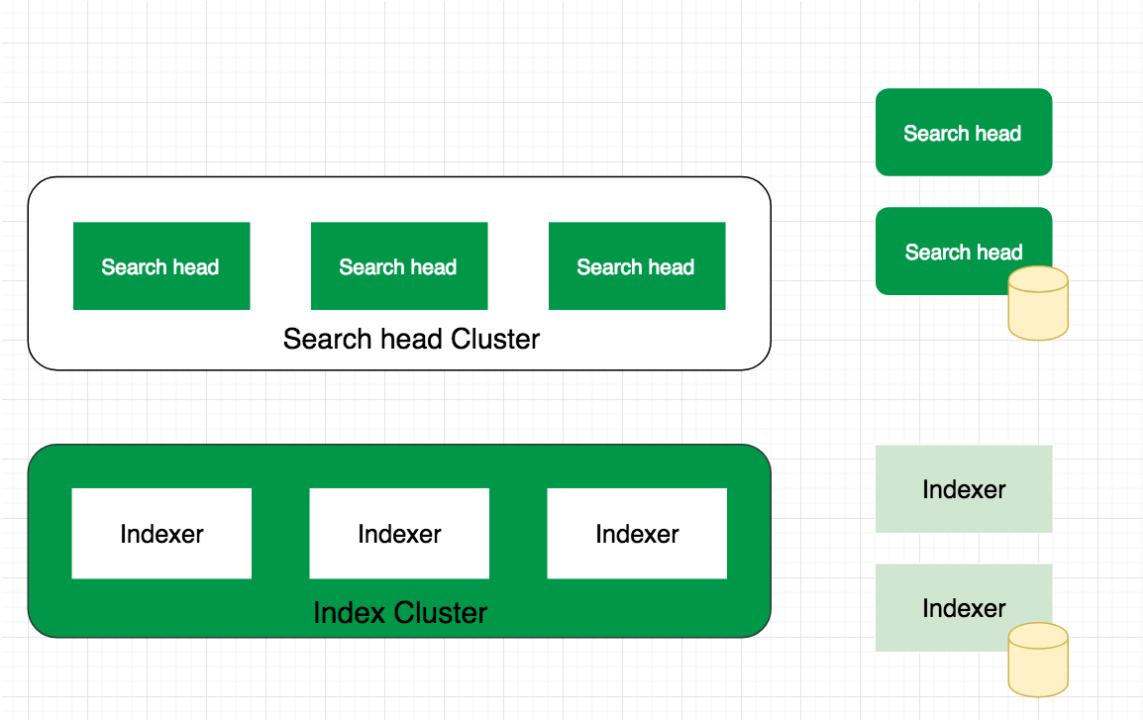
System Test in Splunk

Scalability Test – dynamic scale the product

- ✓ Scale up and Scale down
- ✓ Clean scale and dirty scale
- ✓ Concentrate on functional scalability instead of performance

CHECK POINT

- ✓ Functionality is correct after scaling
- ✓ Scaling timing is acceptable
- ✓ No crash and fatal error
- ✓ Resource usage is acceptable



System Test in Splunk

Migration test – evaluate product with migration

- ✓ Upgrade (Online/Offline)
- ✓ Data migration
- ✓ Partial deployment migration
- ✓ Upgrade in large volume environment

CHECK POINT

- ✓ Functionality is correct after upgrading
  - ✓ No regression for legacy feature
  - ✓ New feature is applied
- ✓ No crash and fatal error
- ✓ No error imported after upgrading
- ✓ System in health state
- ✓ Upgrade timing is acceptable
- ✓ Performance KPI is met

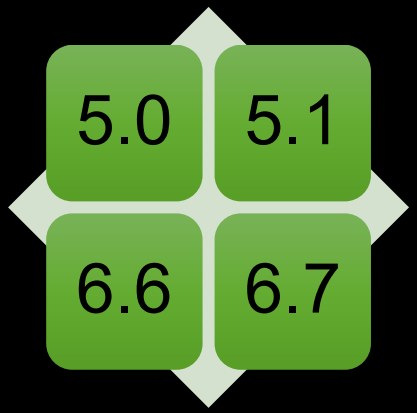
System Test in Splunk

Interoperability Test

- ✓ Cross-version interoperability
- ✓ Cross-deployment interoperability

Check point

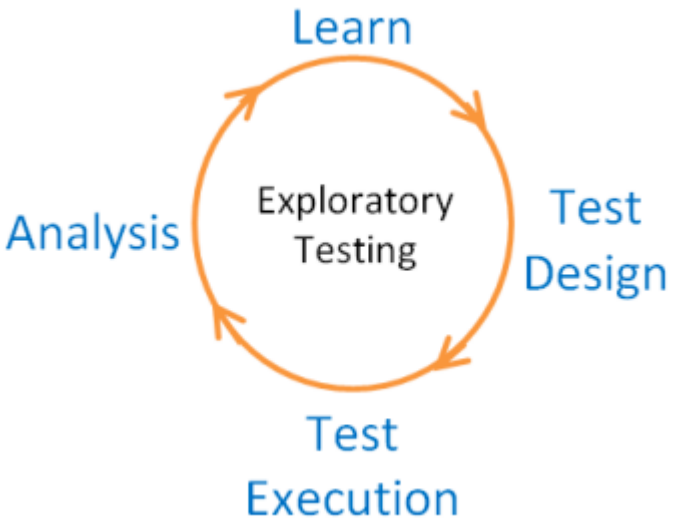
- ✓ Compatibility is guaranteed as design
- ✓ Functionality is correct in hybrid deployment (cloud + on prem)



System Test in Splunk

Ad-hoc test

- ✓ To explore new features
- ✓ Exploratory test



Test Framework

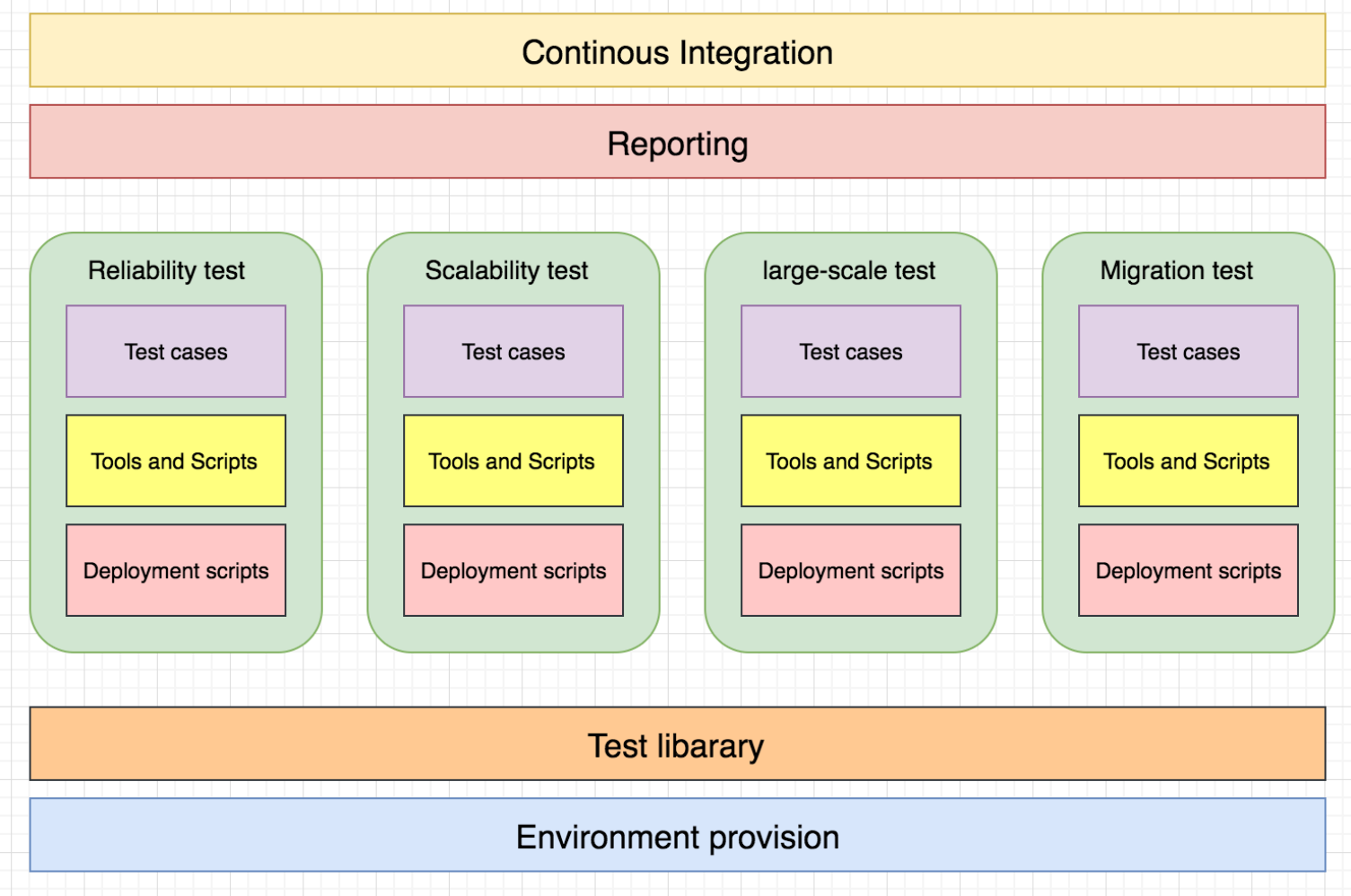
- ✓ Difference with Unit test and Feature test , there's no unified system test framework as it is various from product and test design
- ✓ System test framework is not only to manage tests but also need to manage the corresponding tools and services

Test Framework

What is a good System test Framework ?

- Best efforts for automation
- Extensive and Configurable
- Modularity and Decoupling
- Traceable

Test Framework – V1



Test Framework – V1

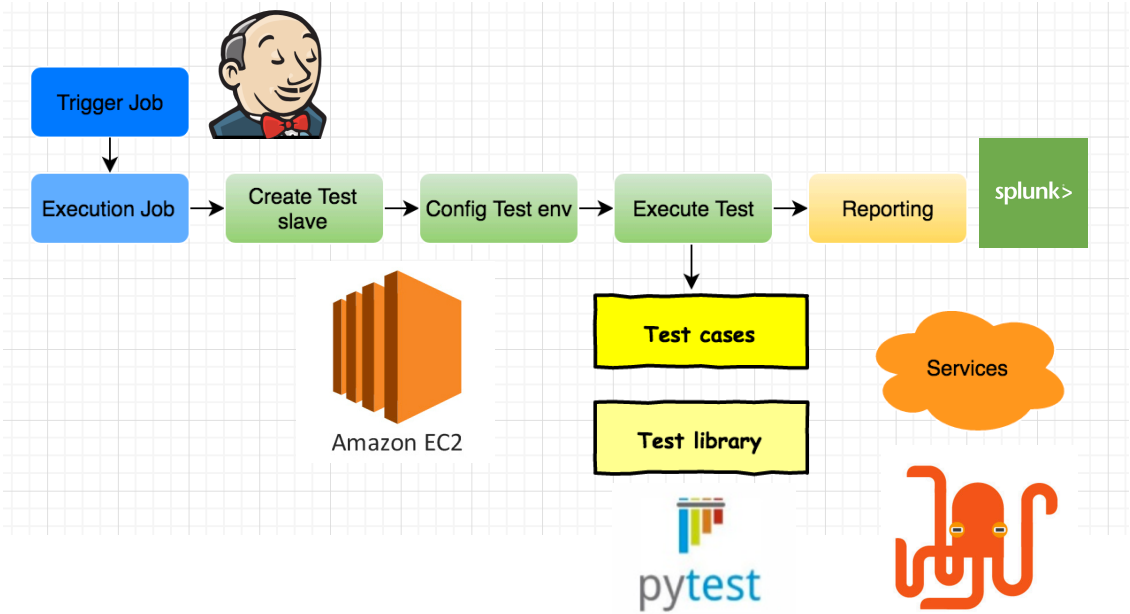
- ✓ Various test design patterns
- ✓ Duplicated work on tools and scripts
- ✓ Duplicated resource (server, storage)
- ✓ Expensive for maintain and add new test
- ✓ Winding learning curve

Test framework – an improved design

Test framework – an improved design

Test flow

- ✓ Unified CI solution
- ✓ Unified reporting framework
- ✓ Unified test library
- ✓ Flexible test scenario design
- ✓ Test related services could be dynamically imported from Service store

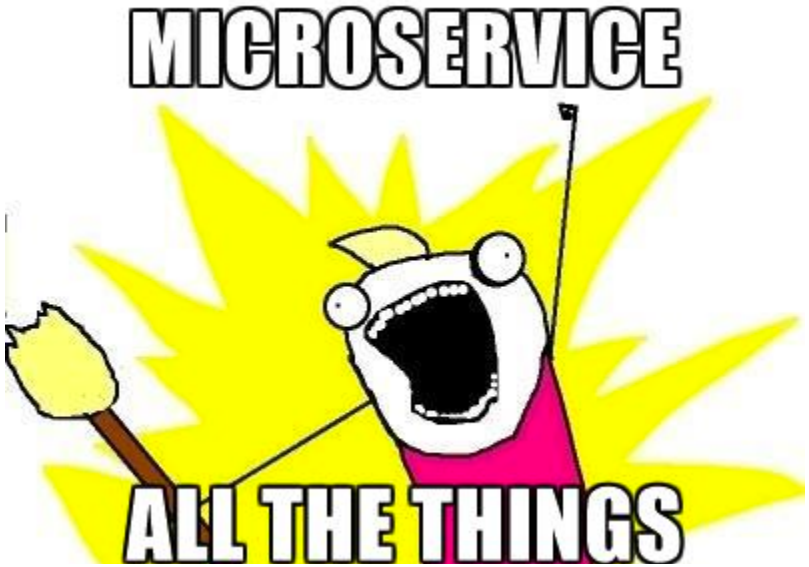


How we build the Service Store?

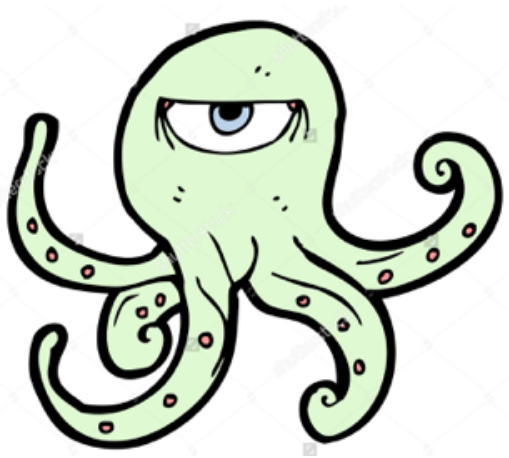
Requirement

- ✓ Serve both automation and ad-hoc test
- ✓ Dynamic import
- ✓ Asynchronous invocation
- ✓ Traceable
- ✓ Extensive and configurable
- ✓ Easy to scale
- ✓ Low Cost for operation

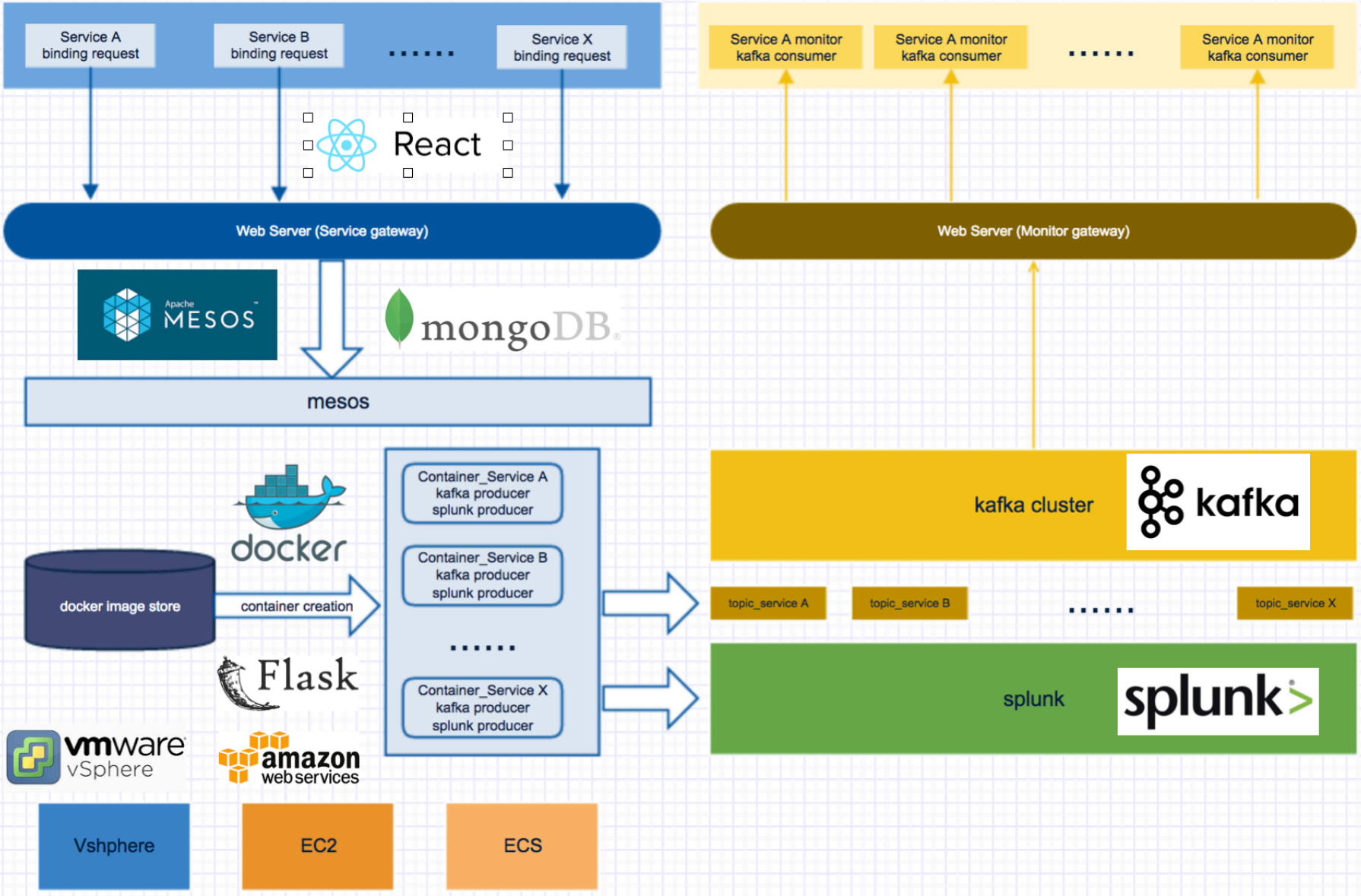
How we build the Service Store?



Octopus House



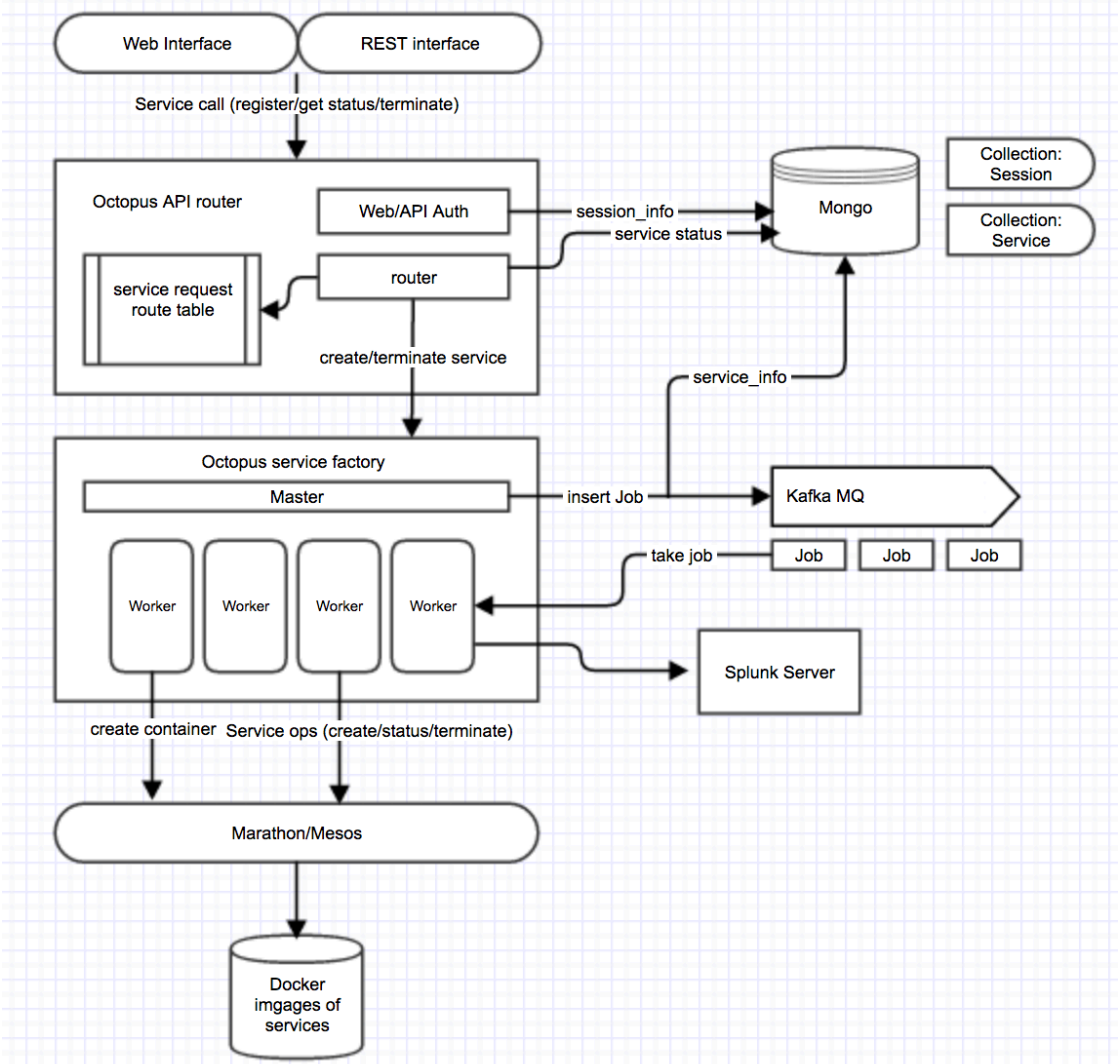
Octopus House Architecture



Octopus House

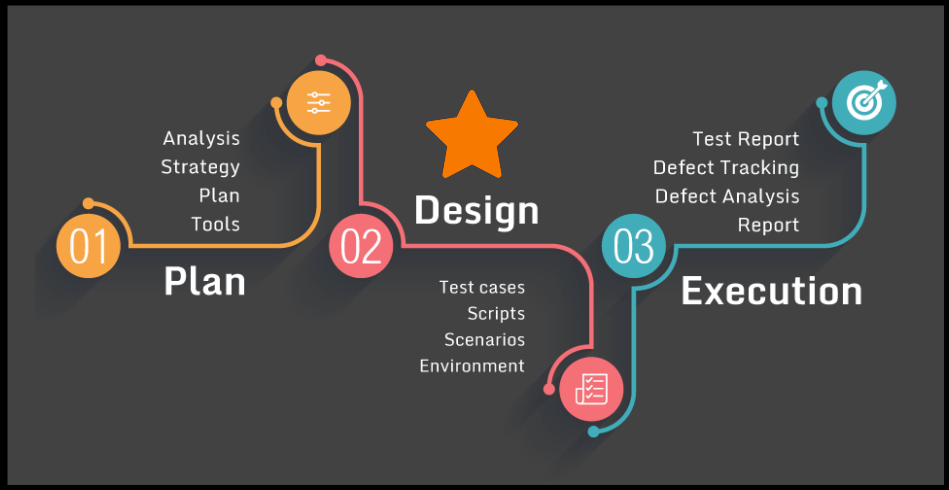
Key Components

- ✓ Interface
- ✓ API Router
- ✓ Service Factory
- ✓ Master and Worker
- ✓ Service Repository
- ✓ Meta-data DB
- ✓ Message queue
- ✓ Splunk Server



Now, when adding a new test scenario in system test …

- ✓ Concentrate the test flow and case design
- ✓ Make the design comply with the framework pattern
- ✓ Plugin the services according to your test requirements



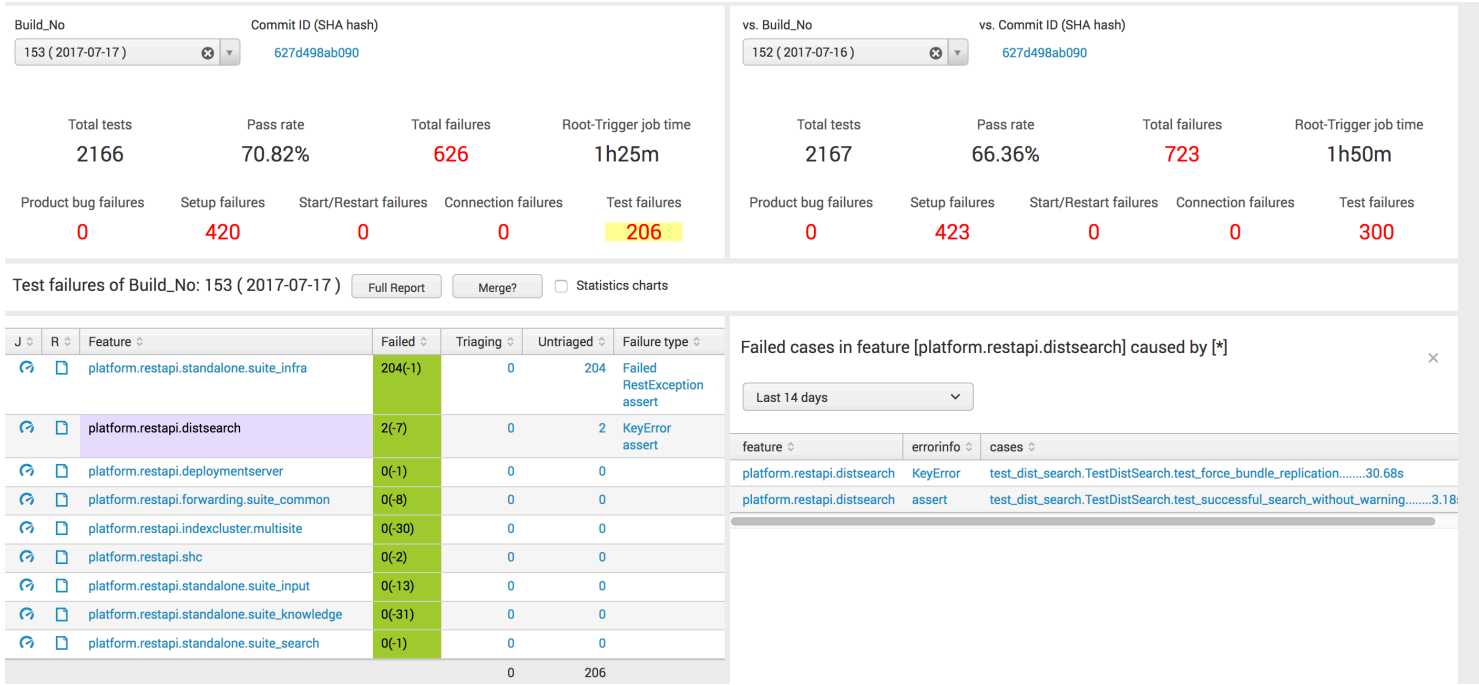
How to triage in System level test

Report Log Monitor Trending

DaoCloud | IT大咖说 知识分享平台

Triage from Test report

Get your data in and Splunk it !



Build_No: 153 (2017-07-17) | Commit ID (SHA hash): 627d498ab090

Total tests	Pass rate	Total failures	Root-Trigger job time
2166	70.82%	626	1h25m

Product bug failures	Setup failures	Start/Restart failures	Connection failures	Test failures
0	420	0	0	206

vs. Build_No: 152 (2017-07-16) | vs. Commit ID (SHA hash): 627d498ab090

Total tests	Pass rate	Total failures	Root-Trigger job time
2167	66.36%	723	1h50m

Product bug failures	Setup failures	Start/Restart failures	Connection failures	Test failures
0	423	0	0	300

Test failures of Build_No: 153 (2017-07-17) | Full Report | Merge? | Statistics charts

J	R	Feature	Failed	Triaging	Untriaged	Failure type
		platform.restapi.standalone.suite_infra	204(-1)	0	204	Failed RestException assert
		platform.restapi.distsearch	2(-7)	0	2	KeyError assert
		platform.restapi.deploymentserver	0(-1)	0	0	
		platform.restapi.forwarding.suite_common	0(-8)	0	0	
		platform.restapi.indexcluster.multisite	0(-30)	0	0	
		platform.restapi.shc	0(-2)	0	0	
		platform.restapi.standalone.suite_input	0(-13)	0	0	
		platform.restapi.standalone.suite_knowledge	0(-31)	0	0	
		platform.restapi.standalone.suite_search	0(-1)	0	0	
				0	206	

Failed cases in feature [platform.restapi.distsearch] caused by [*]

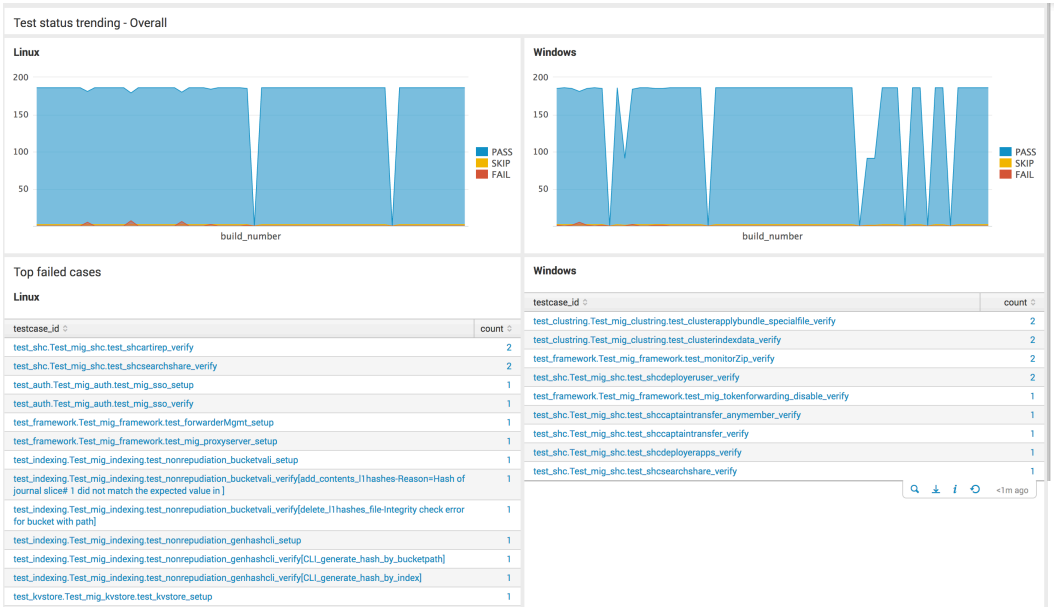
Last 14 days

feature	errorinfo	cases
platform.restapi.distsearch	KeyError	test_dist_search.TestDistSearch.test_force_bundle_replication.....30.68s
platform.restapi.distsearch	assert	test_dist_search.TestDistSearch.test_successful_search_without_warning.....3.18

splunk> listen to your data

Triage from Test report

Get your data in and Splunk it !



Abnormal trending detection

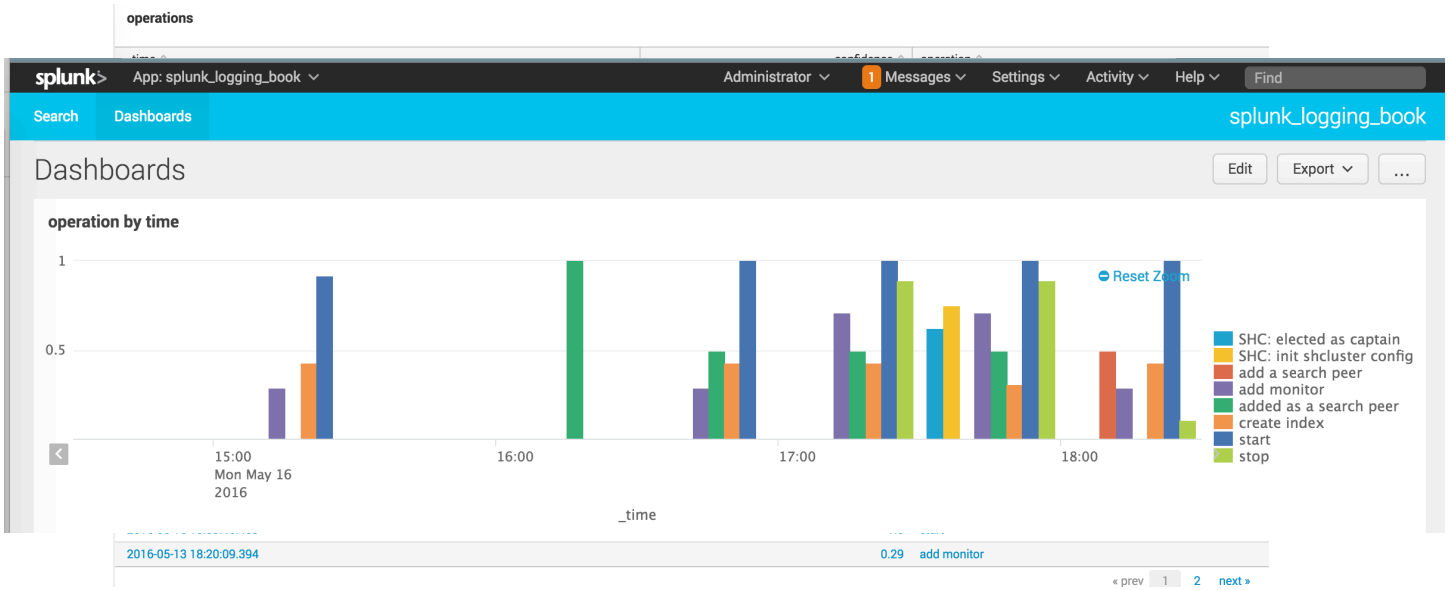
Logging Translation and Correlation

```
07-19-2017 10:05:29.655 +0800 INFO  DatabaseDirectoryManager - idx=_audit Writing a bucket manifest in hotWarmPath='/usr/local/splunk/var/lib/splunk/audit/db', pendingBucketUpdates=0 .  Reason='Buckets we
re rebuilt or tsidx-minified (bucket_count=1).'
07-19-2017 10:05:29.655 +0800 INFO  DatabaseDirectoryManager - Finished writing bucket manifest in hotWarmPath=/usr/local/splunk/var/lib/splunk/audit/db
07-19-2017 10:05:29.656 +0800 INFO  DatabaseDirectoryManager - idx=_introspection Writing a bucket manifest in hotWarmPath='/usr/local/splunk/var/lib/splunk/_introspection/db', pendingBucketUpdates=0 .  R
eason='Buckets were rebuilt or tsidx-minified (bucket_count=1).'
07-19-2017 10:05:29.656 +0800 INFO  DatabaseDirectoryManager - Finished writing bucket manifest in hotWarmPath=/usr/local/splunk/var/lib/splunk/_introspection/db
07-19-2017 10:05:29.657 +0800 INFO  DatabaseDirectoryManager - idx=main Writing a bucket manifest in hotWarmPath='/usr/local/splunk/var/lib/splunk/defaultdb/db', pendingBucketUpdates=0 .  Reason='Buckets
were rebuilt or tsidx-minified (bucket_count=1).'
07-19-2017 10:05:29.657 +0800 INFO  DatabaseDirectoryManager - Finished writing bucket manifest in hotWarmPath=/usr/local/splunk/var/lib/splunk/defaultdb/db
07-19-2017 10:25:40.337 +0800 WARN  TailReader - Access error while handling path: failed to open for checksum: '/usr/local/splunk/var/log/introspection/kvstore.log' (No such file or directory)
07-19-2017 10:25:40.416 +0800 INFO  WatchedFile - Will begin reading at offset=24997366 for file='/usr/local/splunk/var/log/introspection/kvstore.log.1'.
07-19-2017 10:25:41.316 +0800 INFO  WatchedFile - File too small to check seekcrc, probably truncated.  Will re-read entire file='/usr/local/splunk/var/log/introspection/kvstore.log'.
07-20-2017 00:00:00.204 +0800 INFO  LMStackMgr - should rollover=true because _lastRolloverTime=1500393600 lastRolloverDay=1500393600 snappedNow=1500480000
07-20-2017 00:00:00.205 +0800 INFO  LMStackMgr - finished rollover, new lastRolloverTime=1500480000
07-20-2017 00:00:00.221 +0800 INFO  IndexWriter - Creating hot bucket=hot_v1_1, idx=_telemetry, event timestamp=1500480000, reason="suitable bucket not found, number of hot buckets=0, max=3"
07-20-2017 00:00:00.222 +0800 INFO  DatabaseDirectoryManager - idx=_telemetry Writing a bucket manifest in hotWarmPath='/usr/local/splunk/var/lib/splunk/_telemetry/db', pendingBucketUpdates=0 .  Reason='A
dding bucket, bid=_telemetry~1~84E62911-3C0E-4797-BFA0-F184B47D84A1'
07-20-2017 00:00:00.223 +0800 INFO  DatabaseDirectoryManager - Finished writing bucket manifest in hotWarmPath=/usr/local/splunk/var/lib/splunk/_telemetry/db
07-20-2017 00:00:12.204 +0800 INFO  LMSlaveInfo - Detected that masterTimeFromSlave(Wed Jul 19 23:59:12 2017) < lastRolloverTime(Thu Jul 20 00:00:00 2017), meaning that the master has already rolled over.
 Ignore slave persisted usage.
07-20-2017 00:53:42.208 +0800 INFO  WatchedFile - Will begin reading at offset=0 for file='/usr/local/splunk/var/log/splunk/metrics.log'.
07-20-2017 02:38:50.214 +0800 WARN  TailReader - Access error while handling path: failed to open for checksum: '/usr/local/splunk/var/log/introspection/resource_usage.log' (No such file or directory)
07-20-2017 02:38:50.215 +0800 INFO  WatchedFile - Will begin reading at offset=24999955 for file='/usr/local/splunk/var/log/introspection/resource_usage.log.1'.
07-20-2017 02:38:51.215 +0800 INFO  WatchedFile - Will begin reading at offset=0 for file='/usr/local/splunk/var/log/introspection/resource_usage.log'.
07-20-2017 03:05:00.013 +0800 INFO  ExecProcessor - setting reschedule_ms=86399987, for command=python /usr/local/splunk/etc/apps/splunk_instrumentation/bin/instrumentation.py
07-20-2017 06:42:54.158 +0800 WARN  TailReader - Insufficient permissions to read file='/usr/local/splunk/var/log/introspection/kvstore.log' (hint: No such file or directory ,
ID: 0, GID: 0).
07-20-2017 06:42:54.158 +0800 INFO  WatchedFile - Will begin reading at offset=24996420 for file='/usr/local/splunk/var/log/introspection/kvstore.log.1'.
07-20-2017 06:42:55.160 +0800 INFO  WatchedFile - File too small to check seekcrc, probably truncated.  Will re-read entire file='/usr/local/splunk/var/log/introspection/kvstore.log'.
```

splunk> listen to your data

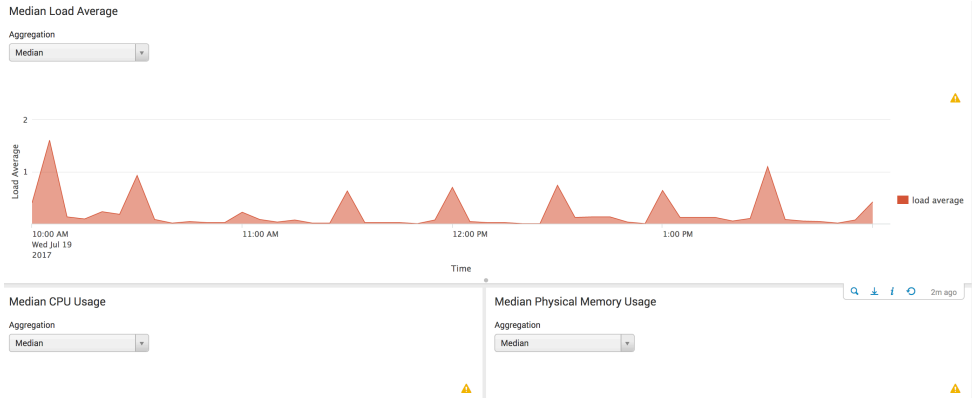
Logging Translation and Correlation

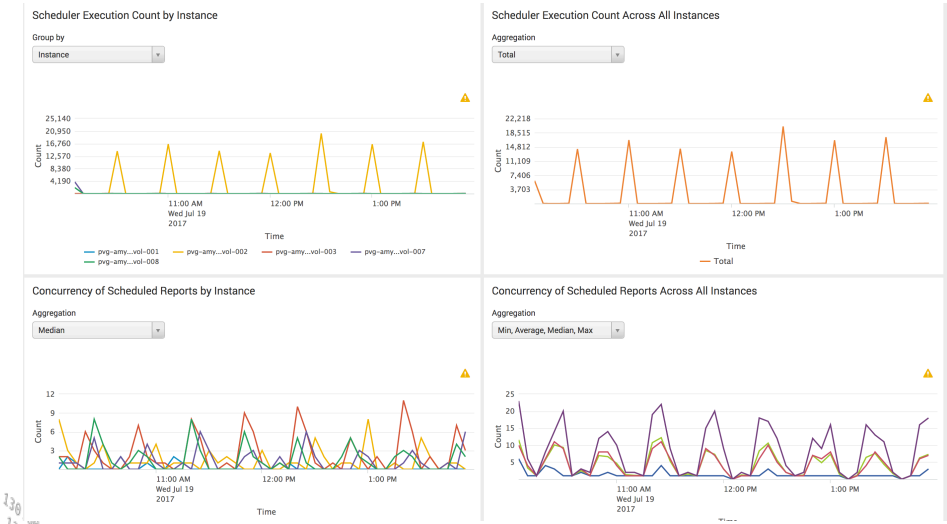
- ✓ Match machine logs to readable test operations
- ✓ Correlate the timing of operations and Errors (failures)
- ✓ Abstract the minimum operation sequence to reproduce the issue

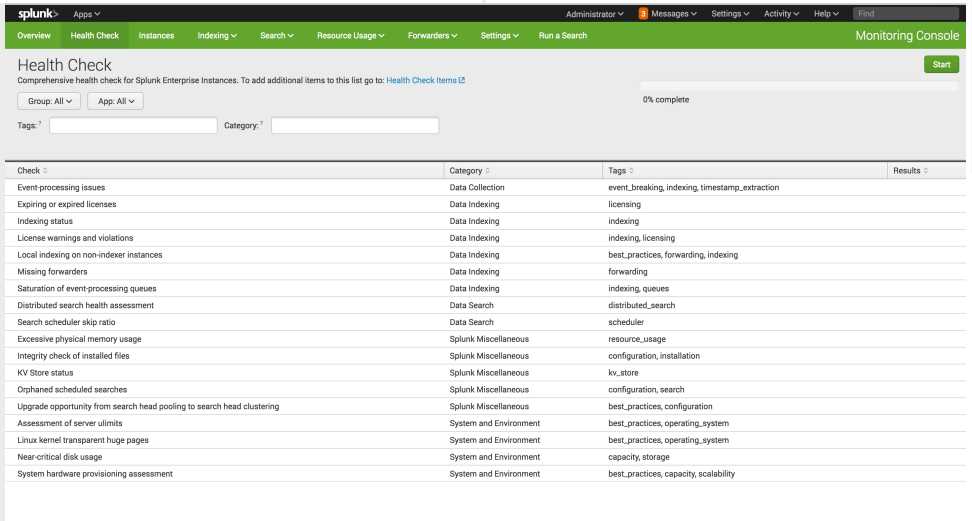


Health-Check and Monitoring

Get your data in and Splunk it !







Check	Category	Tags	Results
Event-processing issues	Data Collection	event_breaking, indexing, timestamp_extraction	
Expiring or expired licenses	Data Indexing	licensing	
Indexing status	Data Indexing	indexing	
License warnings and violations	Data Indexing	indexing, licensing	
Local indexing on non-indexer instances	Data Indexing	best_practices, forwarding, indexing	
Missing forwarders	Data Indexing	forwarding	
Saturation of event-processing queues	Data Indexing	indexing, queues	
Distributed search health assessment	Data Search	distributed_search	
Search scheduler skip ratio	Data Search	scheduler	
Excessive physical memory usage	Splunk Miscellaneous	resource_usage	
Integrity check of installed files	Splunk Miscellaneous	configuration, installation	
KV Store status	Splunk Miscellaneous	kv_store	
Orphaned scheduled searches	Splunk Miscellaneous	configuration, search	
Upgrade opportunity from search head pooling to search head clustering	Splunk Miscellaneous	best_practices, configuration	
Assessment of server ulimits	System and Environment	best_practices, operating_system	
Linux kernel transparent huge pages	System and Environment	best_practices, operating_system	
Near-critical disk usage	System and Environment	capacity, storage	
System hardware provisioning assessment	System and Environment	best_practices, capacity, scalability	

Summary

- System test is an import stage in product test lifecycle
- The coverage in System test depends on your product architecture and resource
- Automation is an import topic in System test, and more in framework design perspective
- Health check and monitoring is important in System test as some product failures may not come from specific cases
- System bug is not easy to debug, and correlation of logs will help identify the troublemaker

DaoCloud

IT大咖说 知识分享平台

THANKS

splunk> listen to your data