# AGILE PERFORMANCE TESTING IN THE REAL WORLD

**Amit Patel**
Total Performance Consulting

1


PERFORMANCE

2

The **HealthCare.gov fiasco** is a clear example of how businesses can seriously undermine their business goals when launching a new app or user platform. **Brands must do performance testing** even when their app or website has much smaller loads than HealthCare.gov.

**FastCompany** 4 Lessons From Healthcare.gov Epic Failure



3

## OTHERS…


**Nordstrom's website is crashing on one of the retailer's biggest shopping days of the year**

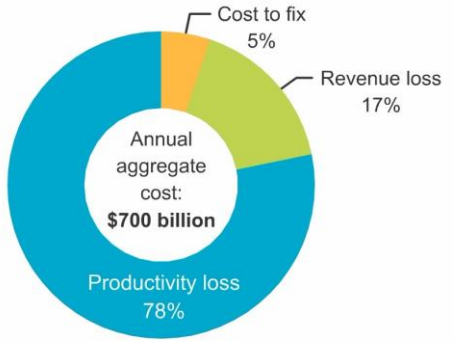



At least four retailers — Macy's, Lowe's, and U.K.-based retailers the Perfume Shop and Game — were affected by technical glitches on Black Friday that slowed purchasing activity and transaction processing on the busiest shopping day of the year. In each case, the problems were reportedly resolved within hours.



4

**IT downtime costs North American businesses $700 billion annually, mostly due to loss of employee productivity**

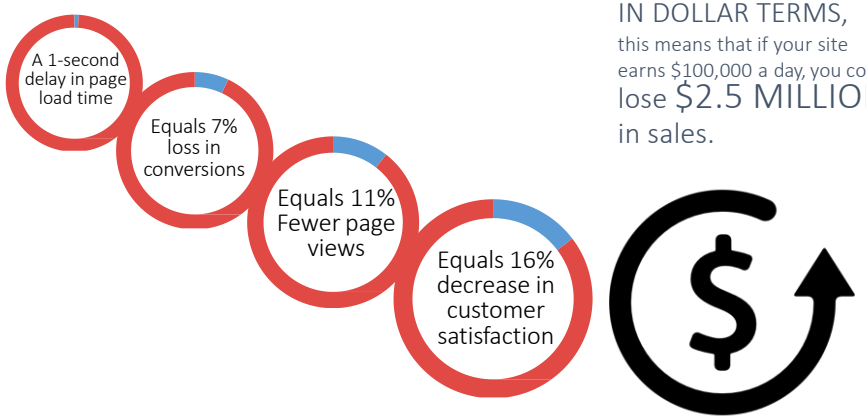



© IHS, *IHS Infonetics The Cost of Server, Application, and Network Downtime: Annual North American Enterprise Survey and Calculator;* 2016

5

5

## PERFORMANCE MATTERS




IN DOLLAR TERMS,
this means that if your site earns $100,000 a day, you could lose $2.5 MILLION in sales.

6

## NOT ALL ABOUT THE MONEY

*Is the risk of customer complaints really worth not performance testing?*

- Predicatable deployments
- Understand scalablity with or w/o cloud
- CDN performance
- Container scalablity
- Customer satisfaction
- Proper infrastructure sizing
- Autoscaling in the cloud


TOTAL performance CONSULTING

7

7

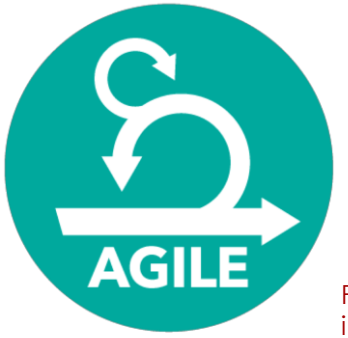

8

9

More frequent releases

Automation implemented as part of the SDLC, some CI, CD maybe not

Teams still not fully connected




Performance still often left until the end

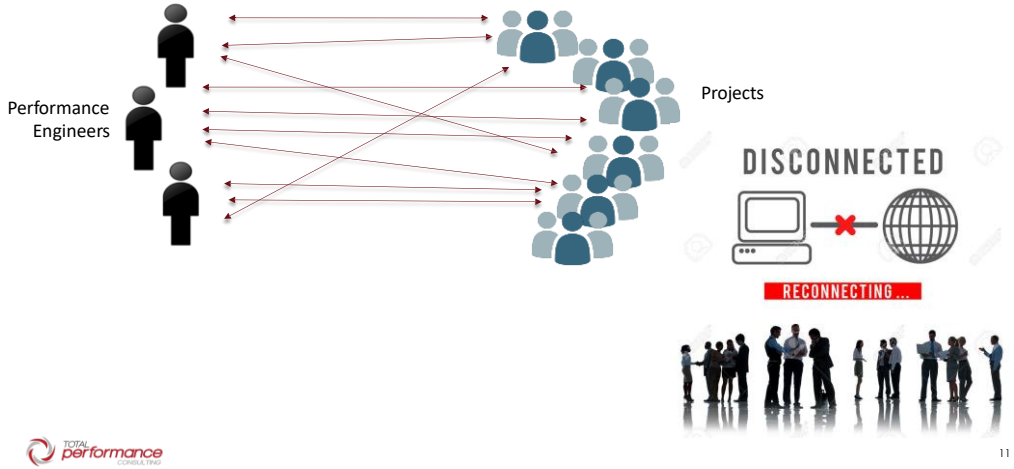
Performance often not included until major releases

TOTAL performance CONSULTING

10

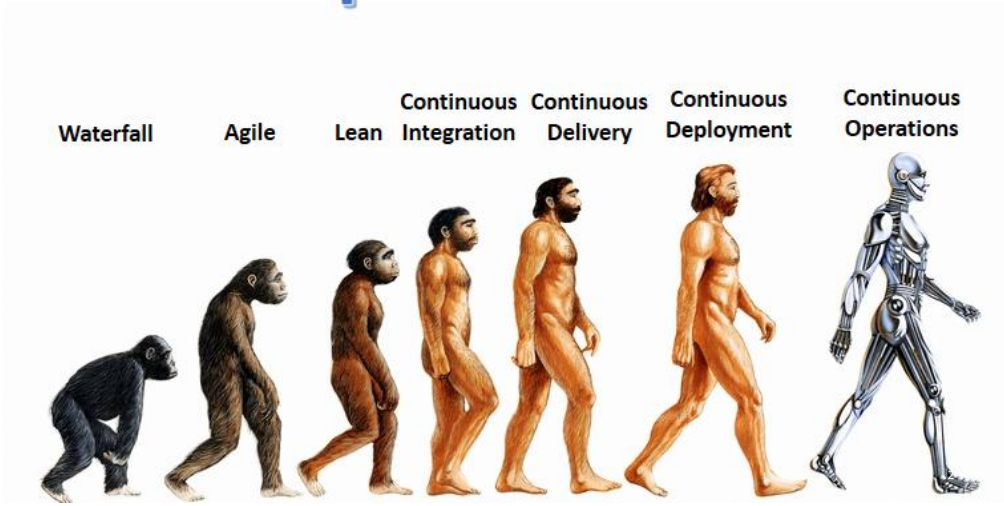
10

## CHALLENGES

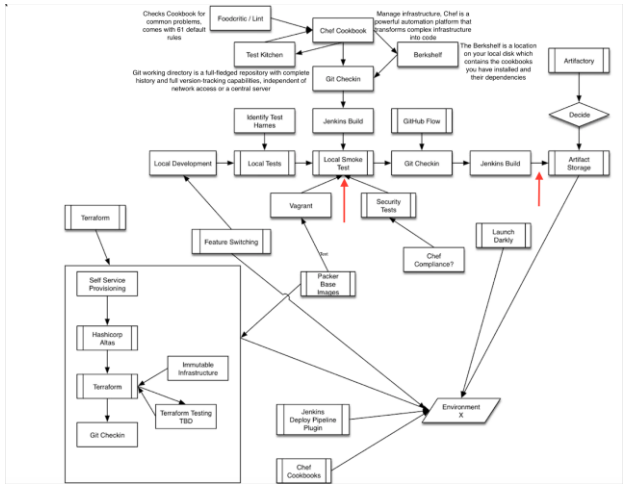



11

# DevOps Movement




12






13

## NEW REQUIREMENTS

- Cultural and Behavioral changes to the organization
- Start to think about performance early from requirements to initial architecture decisions
- Include performance as stories and backlog items (non-functional requirements)
- Integrate with different team members
- Testability
- Creating Performance Defects



14

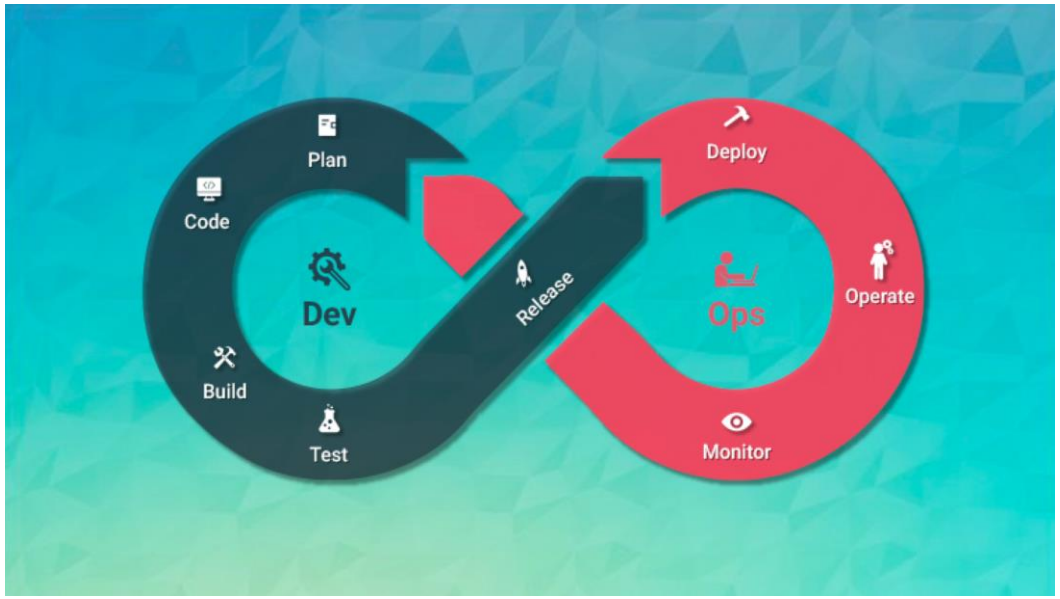
**TEAMWORK**

**MEETINGS**

**FEEDBACK**

15

15




16

## AUTOMATION
## AUTOMATION
## AUTOMATION

- DevOps emphasizes automation
- Build performance into the DevOps automation process
- More data analysis conducted
- Testing is automated
- Scripting challenges
- Service Virtualization
  - Infrastructure sizing
  - Availability of back end system
  - Test earlier (service not fully ready yet)



17

## DEVOPS TOOL STACK

| SCM | TESTING | CI | OPS | CONFIG | DEPLOYMENT | MONITORING | COLLABORATION |
|---|---|---|---|---|---|---|---|
| GitHub | JMeter | TeamCity | splunk> | | | New Relic. | |
| git | FitNesse | Bamboo | | Chef | | Nagios | |
| Bitbucket | Selenium | | | | | CloudWatch | |
| SUBVERSION | TestNG | | | puppet labs | | dynatrace | |

18

## LOAD TESTING TOOLS

### TYPICAL REQUIREMENTS:

- Project and product requirements
- Cloud based load
- Internal vs external traffic or both
- Skill sets required
- Budget

### THINGS TO ALSO CONSIDER:

- Integration with CI solutions (s)
- Monitoring integrations
- SLAs
- API testing / Service Level Testing





## CI TOOLS

- Scheduling/Triggering performance tests
- SLA pass/fail (error rates, response times, throughput…)
- Allows data to be trended over time (Response Times, Errors, Throughput…)


## APM SOLUTIONS

- Production feedback loop
- Monitoring QA and Staging environments
- Compare Pre-PROD data with PROD
- Automate collection of data

APPDYNAMICS

PINPOINT

New Relic.

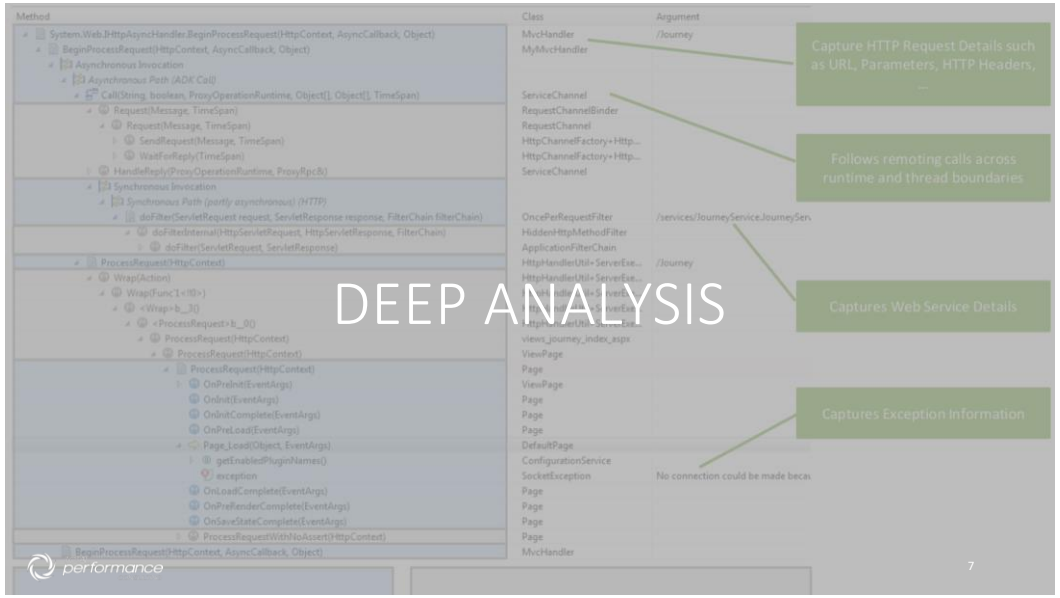
dynatrace



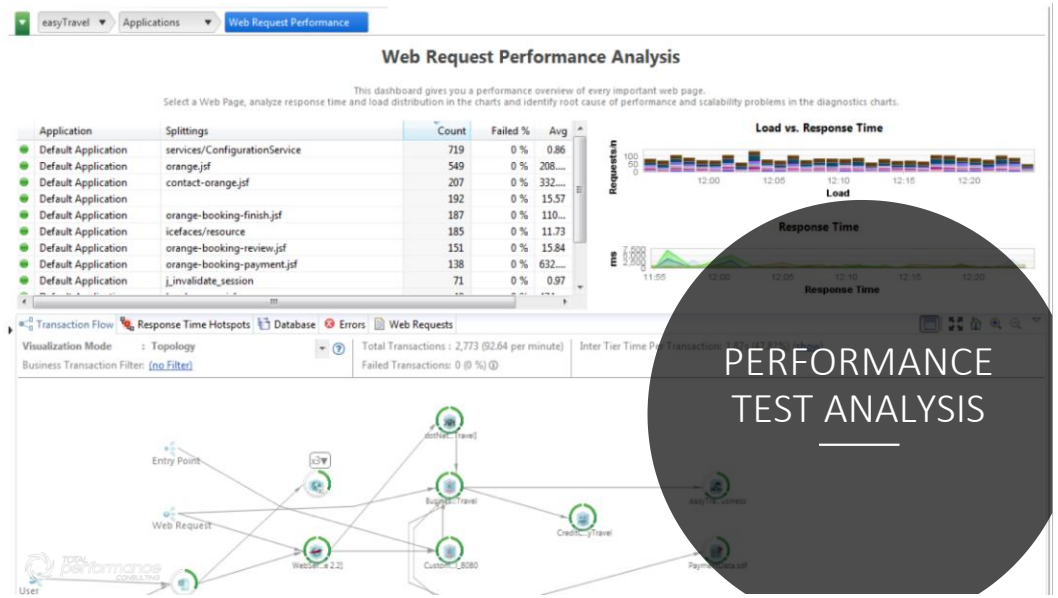


## PRODUCTION FEEDBACK LOOP

| Name | Combined Avg | Page Views | Browser Avg | Server Avg | Avg Dom Processing Duration | DomProcessing Duration (95%) | Avg Page Rendering Duration | Page Rendering Duration (95%) | Combined (95%) | Browser 95% | Backend (95%) | Impact Rate |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Search | 6.50 | 190128 | 4.60 | 1.90 | 2.29 | 6.61 | 2.30 | 8.38 | 16.05 | 12.03 | 4.02 | 3,051,364 |
| Login | 8.77 | 76081 | 1.37 | 7.41 | 1.27 | 4.32 | 0.10 | 0.22 | 34.62 | 2.33 | 32.28 | 2,633,696 |
| View Cart | 15.93 | 29486 | 14.60 | 1.33 | 14.54 | 49.75 | 0.06 | 0.12 | 50.82 | 48.52 | 2.30 | 1,498,508 |
| Add To Cart | 11.59 | 10640 | 9.10 | 2.49 | 9.05 | 34.01 | 0.05 | 0.09 | 38.35 | 33.76 | 4.59 | 408,055 |
| Checkout | 19.33 | 5998 | 18.25 | 1.08 | 18.20 | 60.56 | 0.05 | 0.11 | 61.88 | 60.00 | 1.89 | 371,168 |
| Category | 6.92 | 3841 | 3.15 | 3.76 | 2.11 | 4.60 | 1.04 | 2.95 | 32.68 | 4.28 | 28.39 | 125,505 |
| Registration | 12.33 | 2621 | 11.73 | 0.60 | 11.68 | 46.93 | 0.05 | 0.15 | 47.64 | 46.57 | 1.07 | 124,857 |
| My account | 13.26 | 3898 | 0.75 | 12.52 | 0.71 | 2.04 | 0.04 | 0.11 | 30.64 | 1.02 | 29.62 | 119,431 |
| Update password | 32.00 | 1056 | 30.08 | 1.92 | 30.00 | 93.94 | 0.08 | 0.36 | 97.23 | 92.03 | 5.21 | 102,679 |
| Delete item from cart | 6.16 | 2749 | 0.92 | 5.23 | 0.90 | 2.51 | 0.03 | 0.04 | 30.04 | 0.74 | 29.30 | 82,591 |
| Homepage | 12.96 | 1597 | 12.30 | 0.66 | 12.23 | 41.88 | 0.07 | 0.13 | 42.86 | 41.57 | 1.29 | 68,439 |
| Filter/Facet | 8.17 | 4783 | 1.37 | 6.80 | 0.86 | 6.00 | 0.51 | 0.11 | 12.45 | 1.70 | 10.75 | 59,539 |
| Search | 6.07 | 1903 | 0.71 | 5.36 | 0.68 | 1.58 | 0.03 | 0.11 | 26.96 | 1.14 | 25.81 | 51,299 |
| Login | 5.77 | 1830 | 3.82 | 1.95 | 0.65 | 2.04 | 3.17 | 15.00 | 19.44 | 14.72 | 4.71 | 35,568 |
| View Cart | 19.18 | 600 | 17.96 | 1.21 | 0.37 | 0.78 | 17.60 | 46.41 | 50.37 | 47.24 | 3.13 | 30,223 |
| Add To Cart | 7.70 | 1138 | 6.43 | 1.27 | 6.41 | 20.92 | 0.02 | 0.04 | 24.04 | 17.62 | 6.41 | 27,352 |

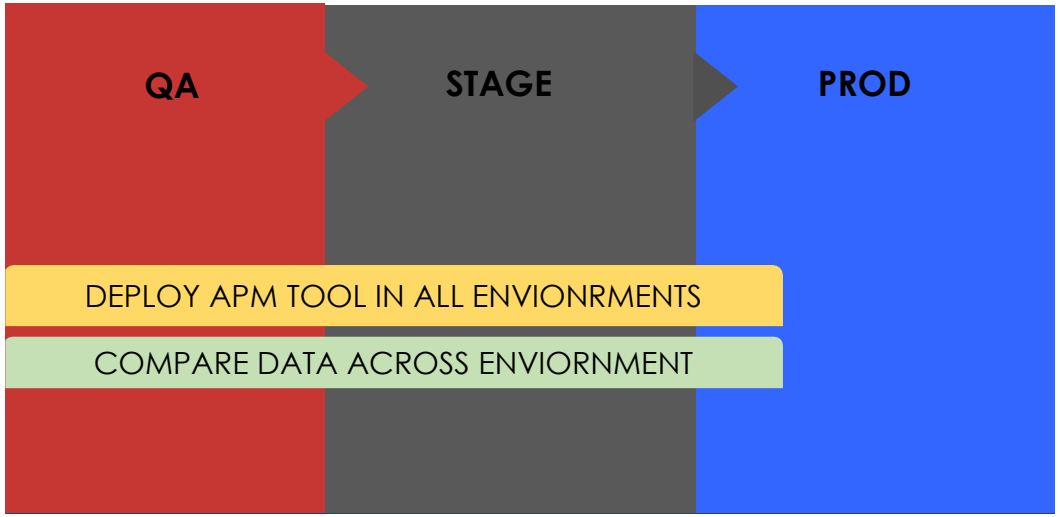

DEEP ANALYSIS
Capture HTTP Request Details such as URL, Parameters, HTTP Headers, ...
Follows remoting calls across runtime and thread boundaries
Captures Web Service Details
Captures Exception Information
performance
7

23


easyTravel
Applications
Web Request Performance
Web Request Performance Analysis
This dashboard gives you a performance overview of every important web page.
Select a Web Page, analyze response time and load distribution in the charts and identify root cause of performance and scalability problems in the diagnostics charts.
Application
Splittings
Count
Failed %
Avg
Default Application services/ConfigurationService 719 0 % 0.86
Default Application orange.jsf 549 0 % 208....
Default Application contact-orange.jsf 207 0 % 332....
Default Application 192 0 % 15.57
Default Application orange-booking-finish.jsf 187 0 % 110...
Default Application icefaces/resource 185 0 % 11.73
Default Application orange-booking-review.jsf 151 0 % 15.84
Default Application orange-booking-payment.jsf 138 0 % 632....
Default Application j_invalidate_session 71 0 % 0.97
Load vs. Response Time
Load
Response Time
Transaction Flow
Response Time Hotspots
Database
Errors
Web Requests
Visualization Mode : Topology
Business Transaction Filter: (no Filter)
Total Transactions : 2,773 (92.64 per minute)
Failed Transactions: 0 (0 %)
Entry Point
Web Request
User
PERFORMANCE TEST ANALYSIS

24

## APM DEPLOYMENT




25


26


27

## REQUIREMENTS & KPIs

- Understanding what is the goal and KPIs required for success
- Understanding cross team requirements
- Types of Tests
- Team responsibilities
- Cloud Testing
- Circuit Breaker

28

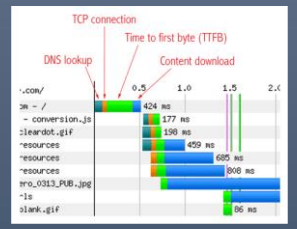
**Sample KPIs**

- Database query performance
- Traffic being sent Mb/s
- Right amount of logging
- Response times
- Time to deploy new code
- Rollback time
- Response sizes
- Autoscaling

28

## TYPES OF TESTS

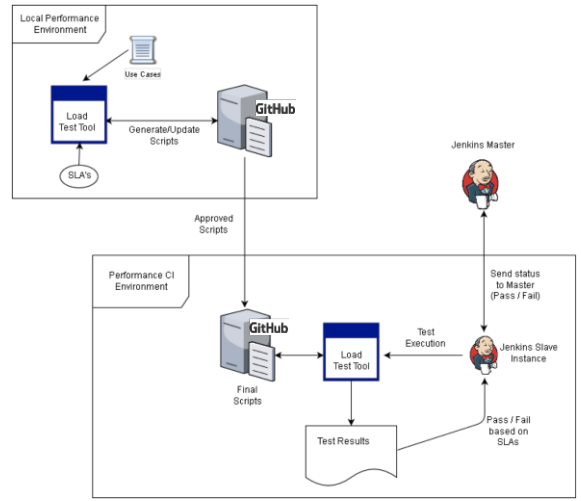
- End-User Experience tests (UI)
- Traditional load tests
  - Scaled down tests
  - Large scale tests
  - Memory leak tests
- API tests (Service level tests)
- Integration tests
- Autoscaling validation testing



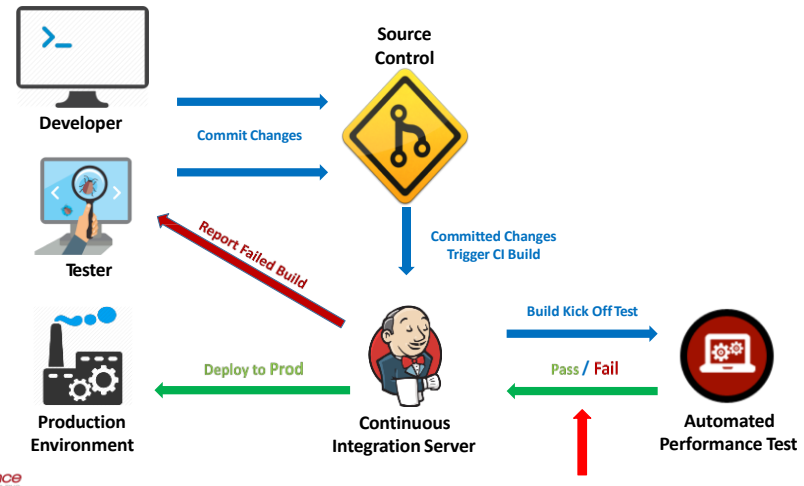





## PERFORMANCE CI ENVIRONMENT EXAMPLE



## HIGH-LEVEL AUTOMATION WORKFLOW EXAMPLE




31

31

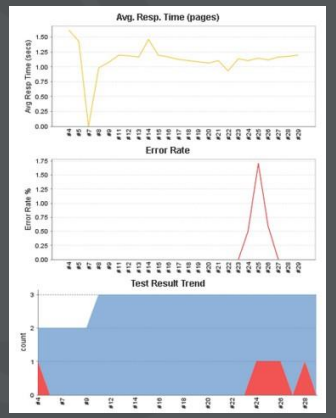
## SLAs

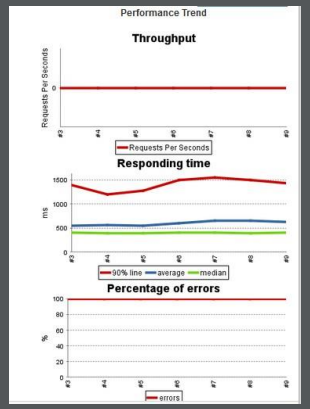
- Transaction level response times
- Request response times
- API response times
- Use case / script level response times
- Total # of Errors / Error rate
- Throughput
- Response sizes

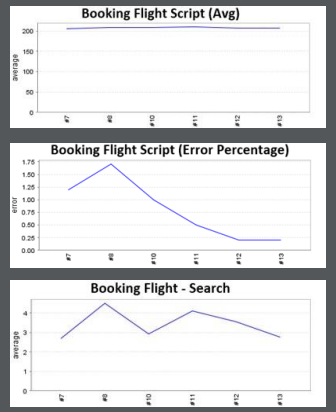
32

32

## RESULTS




TOTAL performance CONSULTING

33

33


9

34

## KEY CHALLENGES

- Integration Timeouts
- Unable to diagnose and resolve issues

**3**
Back to back months of integration call issues

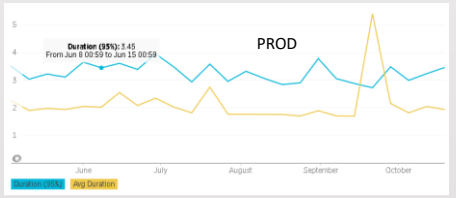
**50%**
Handled only 50% of projected peak traffic

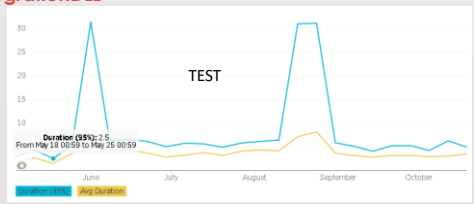
**10**
Struggled to handle 10 concurrent users without seeing integration errors

35

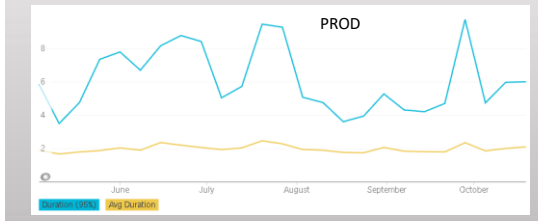
## CASE OF THE INTEGRATION CALLS

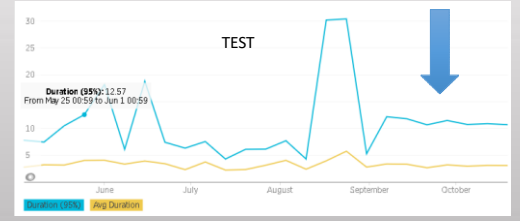
ProductIntegrationDLL




BankIntegrationDLL

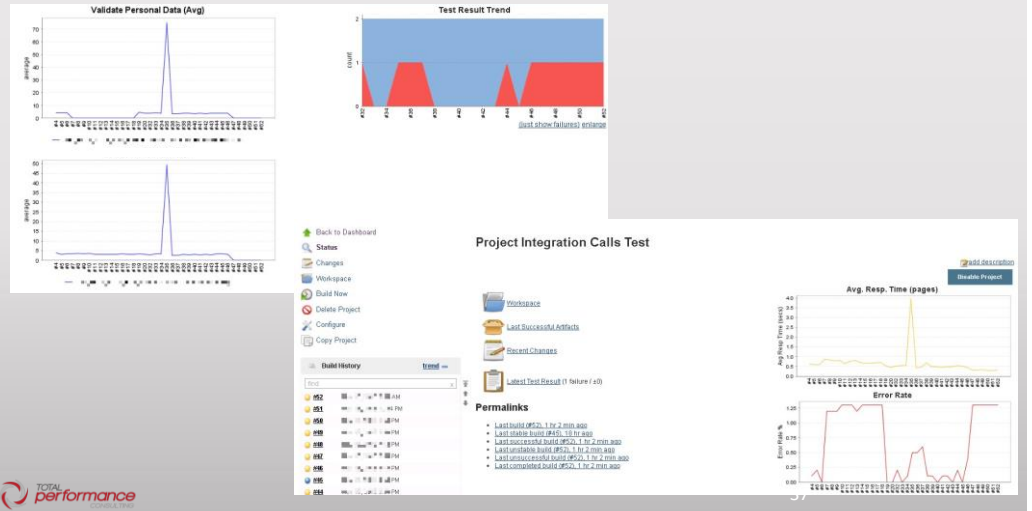



36

36

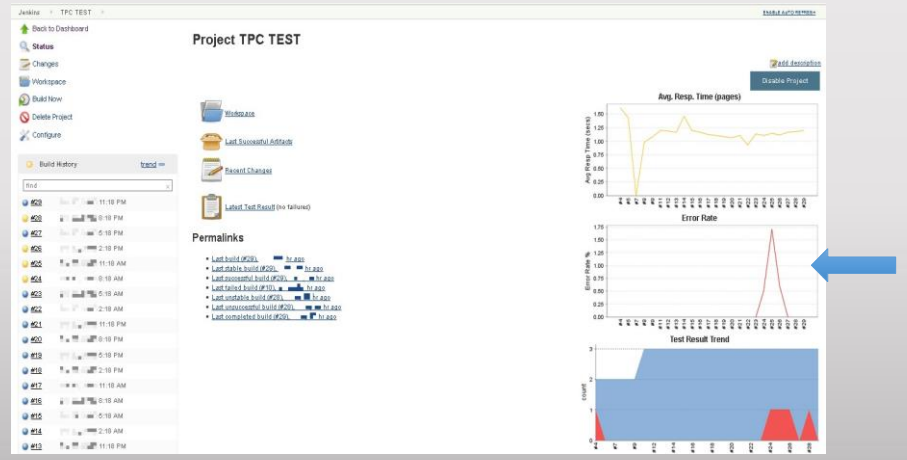
## TESTING INTEGRATION CALLS




37

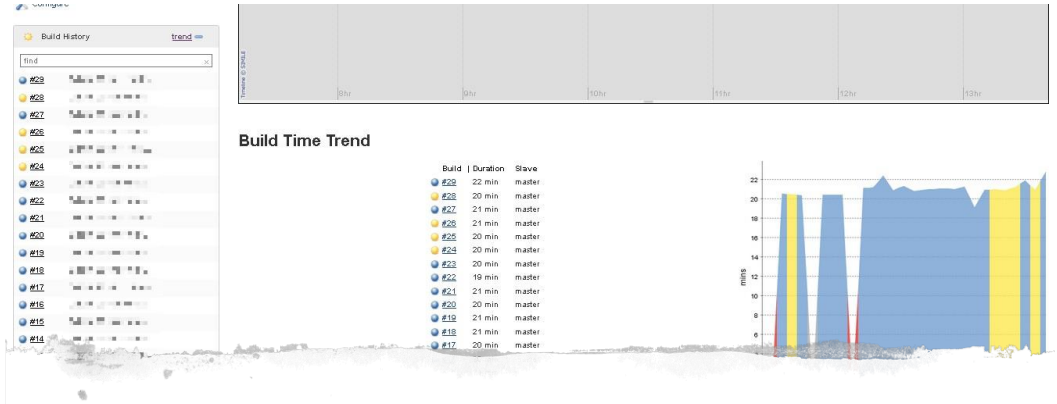

38

## CI SERVICE LEVEL TESTS







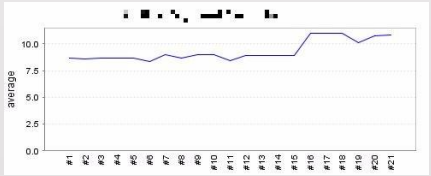




CI SERVICE LEVEL TESTS – BUILD TRENDS

## UI LEVEL TESTS – TRANSACTION TRENDS

Print

Approve Case




Review Application

Search



41

## TAKE AWAYS

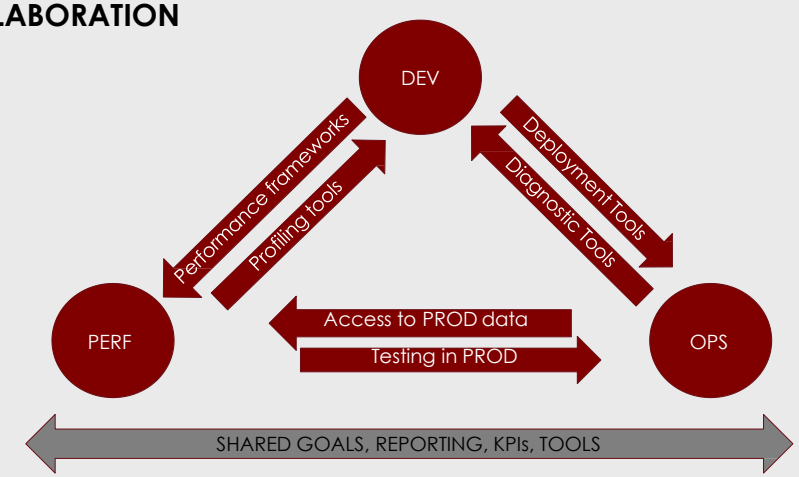
- Collaboration
- Testability
- Setting up the right tests
- Comparing results over time
- Automate result collection

AUTOMATE, LEARN, UPDATE, ITERATE



42

## COLLABORATION




43

## ADDITIONAL THOUGHTS

1. Incoporate performance into every team
2. Plan for large scale on-demand tests
3. Notifications (i.e. Emails, Chat)

44

# QUESTIONS

THANK YOU FOR ATTENDING THIS SESSION

**Amit Patel**
Total Performance Consulting
@aapatel
apatel@totalperform.com

@TotalPerform
www.totalperform.com

45