THE CUSTOMER SUCCESS PLATFORM

ApexUnit 2.x

Increasing Productivity of Apex Development

Continuous Delivery team

Tech Talk
October 15 2015

Safe Harbor

Safe harbor statement under the Private Securities Litigation Reform Act of 1995:

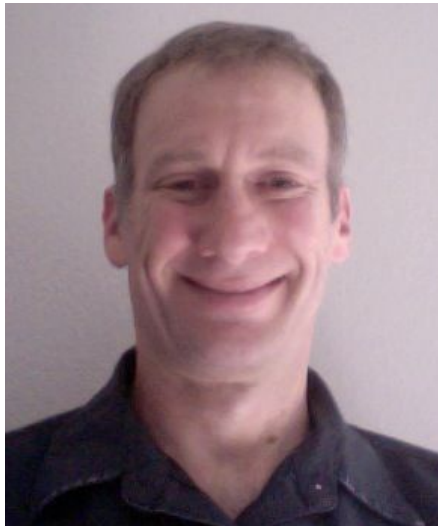
This presentation may contain forward-looking statements that involve risks, uncertainties, and assumptions. If any such uncertainties materialize or if any of the assumptions proves incorrect, the results of salesforce.com, inc. could differ materially from the results expressed or implied by the forward-looking statements we make. All statements other than statements of historical fact could be deemed forward-looking, including any projections of product or service availability, subscriber growth, earnings, revenues, or other financial items and any statements regarding strategies or plans of management for future operations, statements of belief, any statements concerning new, planned, or upgraded services or technology developments and customer contracts or use of our services.

The risks and uncertainties referred to above include – but are not limited to – risks associated with developing and delivering new functionality for our service, new products and services, our new business model, our past operating losses, possible fluctuations in our operating results and rate of growth, interruptions or delays in our Web hosting, breach of our security measures, the outcome of any litigation, risks associated with completed and any possible mergers and acquisitions, the immature market in which we operate, our relatively limited operating history, our ability to expand, retain, and motivate our employees and manage our growth, new releases of our service and successful customer deployment, our limited history reselling non-salesforce.com products, and utilization and selling to larger enterprise customers. Further information on potential factors that could affect the financial results of salesforce.com, inc. is included in our annual report on Form 10-K for the most recent fiscal year and in our quarterly report on Form 10-Q for the most recent fiscal quarter. These documents and others containing important disclosures are available on the SEC Filings section of the Investor Information section of our Web site.

Any unreleased services or features referenced in this or other presentations, press releases or public statements are not currently available and may not be delivered on time or at all. Customers who purchase our services should make the purchase decisions based upon features that are currently available. Salesforce.com, inc. assumes no obligation and does not intend to update these forward-looking statements.

IT Applications

Continuous Delivery Team



Scott Glaser

Vamshi Gandham

Adarsh Ramakrishna

Today’s Overview

1. Introduction
   - Challenges with Apex Development
2. ApexUnit
   - What is ApexUnit
   - Features
3. Demo
   - Getting started
   - Road map

Questions for You

- How many of you code/work in Apex?
- What are the challenges in testing Apex code?

Challenges Managing Apex Code Quality

1. Code Coverage
2. Grouping and Filtering Classes
3. Automated Execution and Detailed Reporting

What Is ApexUnit?

GOAL: Evolved from xUnit frameworks like JUnit and TestNG to simplify apex code quality management

- CLI based tool
- Run tests asynchronously
- Extensive code coverage reports
- Seamless integration with CI engines
- Auto validate quality gates

Code Coverage Metrics

Consolidated code coverage metrics - Actionable items for developers

- Sorted by code coverage %
- Uncovered vs Covered lines
- Org-wide code coverage
- Team code coverage
- API version of source class
- Modularity: Length of the class

Challenges

1. Code Coverage
2. Grouping and Filtering Classes
3. Automated Execution and Detailed Reporting

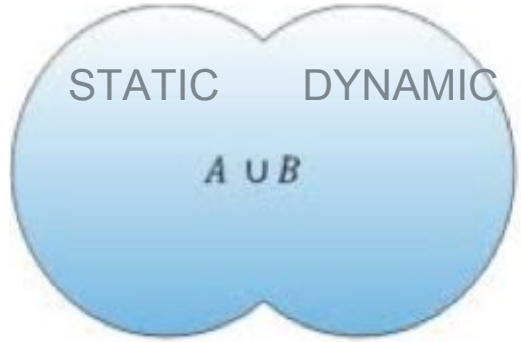
Grouping and Filtering

- Segregating ownership in a shared sfdc org
  - Ability to group test suites (xUnit)
  - Custom test suites to modularize suites
  - Applies to src code
- How to filter
  - **Static** : Manifest files
  - **Dynamic** : Regular expressions
- Multiple manifest files and regular exp.

Challenges

1. Code Coverage
2. Grouping and Filtering Classes
3. Automated Execution and Detailed Reporting

Grouping and Filtering (Contd…)



- Resultant DS = (manifest) UNION (reg. exp)
- Application:
  - **Static** : Legacy classes
  - **Dynamic** : Brand new classes
- Recommended Naming convention
  - ProjectA_ScrumTeamB_ModuleC_YourClassName

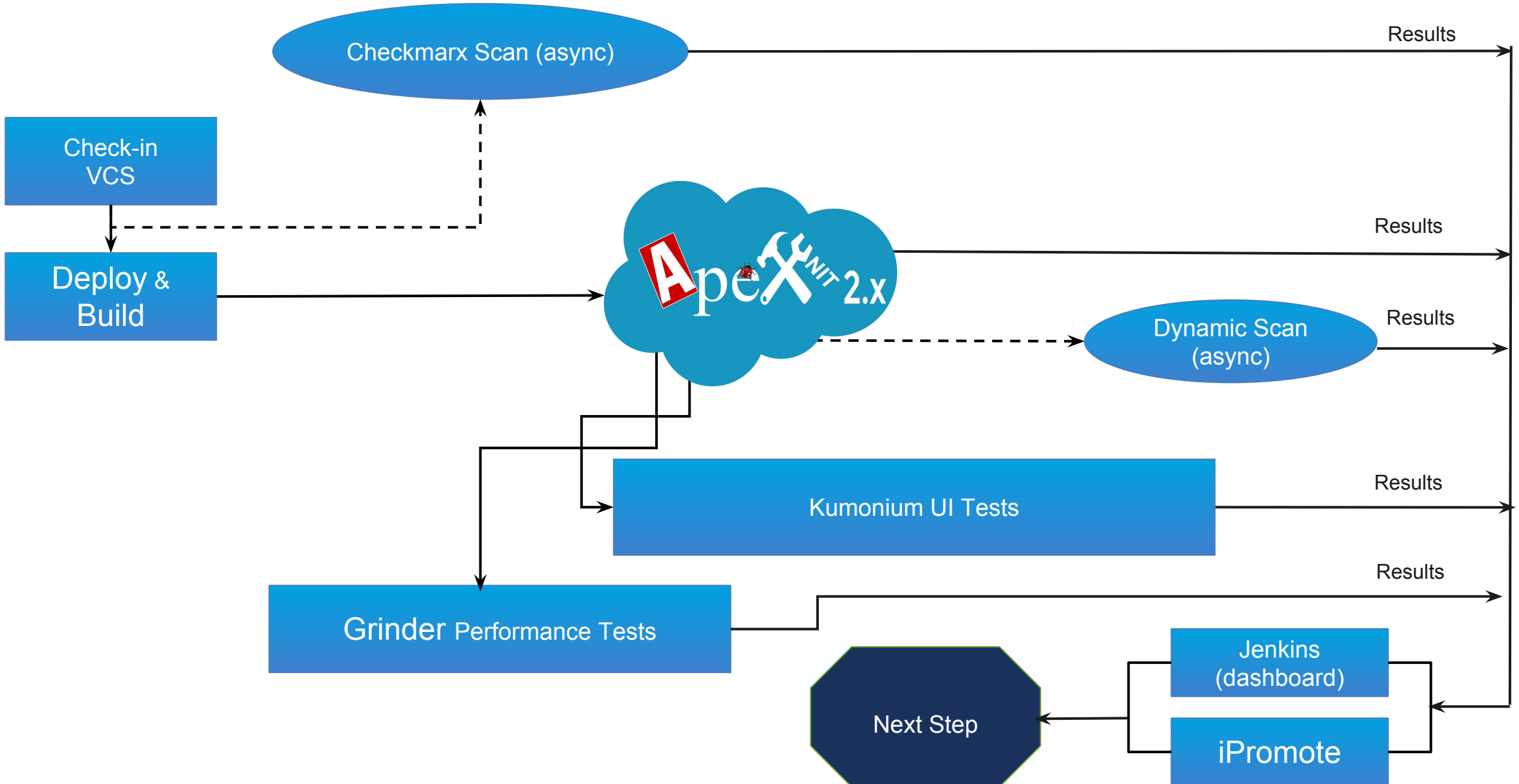
ApexUnit in CI/CD Pipeline

- Customizable code coverage threshold
- Incremental progression from 1- 100% (75-100 for customer orgs)
- Auto abort ? **PS -9 | Kill** for long running tests (customizable)
- Deep dive test reports with drill-down interface
- Seamless integration with CI engines

Challenges

1. Code Coverage
2. Grouping and Filtering Classes
3. Automated Execution and Detailed Reporting

Continuous Delivery Pipeline – Force.com



Recap

1. Challenges with Apex Development
2. ApexUnit
   - What is ApexUnit
   - Features
3. Demo
   - Getting started
   - Road Map

Demo

ApexUnit 2.x demo job

Getting Started with ApexUnit 2.x

- https://github.com/forcedotcom/ApexUnit
- Modify/Add manifest files for test/source classes
- Identify the regex for your test/source classes
- Check-in the project to SCM
- Clone and configure the jenkins job

ApexUnit 2.x Jenkins Template Job

ApexUnit 2.x wiki page

Road Map

- Code coverage at method level for each test class
- Leverage “Wave” for analytics on report
- Integration with cloud CI systems
- Templatizing CI step on CI systems (on-premis and/or cloud)

Thank you

- Team Email: IT-Tools@salesforce.com

Sample Report Using ApexUnit 2.x

ApexUnit Report

Code Coverage Summary: *

Team code coverage: 44.48% [The customized team code coverage threshold was: 80%]

Org wide code coverage: 49.00% [The customized org wide code coverage threshold was: 100%]

Test Execution Summary:

Total test classes executed: 10
Total test methods executed: 11
Test method pass count: 11
Test method fail count: 0

Apex Test Report:

Detailed Test Report

Detailed code coverage report:

Apex Class Name	API Version	Code Coverage %	#Covered Lines	#Uncovered Lines	Covered Lines	Uncovered Lines	Length Without Comments(Bytes)
SampleClass2	28.0	0.00%	0	118	-	-	2356
MobileMetricsController	30.0	0.00%	0	55	-	-	3961
WrapperExampleController	26.0	78.38%	29	8	15,17,19,23,27,30,32,33,34,35, 39,43,45,47,50,51,52,56,57,58, 59,60,61,65,72,75,77,81,83	88,90,91,92,94,95,96,98	2695
WrapperExampleController1	26.0	100.00%	29	0	6,9,12,15,17,19,23,27,30,32, 33,35,39,43,45,47,50,52,56,58, 59,60,61,65,72,75,77,81,83	-	2387
WrapperExample2	26.0	100.00%	29	0	7,10,13,16,18,20,24,28,31,33, 34,36,40,44,46,48,51,53,57,59, 60,61,62,66,73,76,78,82,84	-	2357
WrapperExample1	26.0	100.00%	29	0	7,10,13,16,18,20,24,28,31,33, 34,36,40,44,46,48,51,53,57,59, 60,61,62,66,73,76,78,82,84	-	2357
WrapperExampleController2	26.0	100.00%	29	0	16,18,20,24,28,31,33,34,35,36, 40,44,46,48,51,52,53,57,58,59, 60,61,62,66,73,76,78,82,84	-	2387

Sample Report Using ApexUnit 2.x Continued..

Duplicate Apex Class Names Across Manifest Files And Regular Expressions
Sample_Functional_Test
WrapperExampleController
PartnerApexTestResultDAOTest
CodeCoverageTest
Sample_Integration_Test
Sample_Unit_Test
SimpleClass1Test
IntegerSum_Test

Invalid/Non-existant Apex Class Names Across Manifest Files And Regular Expressions
AuthorPageConfig
AuthorPageConfigTest

Apex_Code_Coverage_Best_Practices

** Code coverage is calculated by dividing the number of unique Apex code lines executed during your test method execution by the total number of Apex code lines in all of your trigger and classes (Note: these numbers do not include lines of code within your testMethods)*

Sample ApexUnit Test Report

3 failures (-2)

55 tests (+10)
Took 0 ms.
add description

All Failed Tests

Test Name	Duration	Age
>>> HTSCConfirmPaymentControllerTest.<compile>	0 ms	4
>>> HTPrescriptiveContentControllerTest.prescriptiveContentControllerTest1	0 ms	4
>>> HTPrescriptiveContentControllerTest.prescriptiveContentControllerTest2	0 ms	4

All Test

Package
(root)

line 84, column 4: Method is not visible: [HTSCConfirmPaymentController].validateDocuSignParameters()

<<< HTPrescriptiveContentControllerTest.prescriptiveContentControllerTest1

Error Details

System.NullPointerException: Attempt to de-referen

>>> HTPrescriptiveContentControllerTest.prescriptiveConte

Failed

HTPrescriptiveContentControllerTest.prescriptiveContentControllerTest1 (from (Apexl

This test was not claimed.

Error Message

System.NullPointerException: Attempt to de-reference a null object

Stacktrace

```
Class.HTWebparController.<init>: line 26, column 1
Class.HTPrescriptiveContentController.fetchConfigurations: line 51, column 1
Class.HTPrescriptiveContentController.init: line 37, column 1
Class.HTPrescriptiveContentController.<init>: line 19, column 1
Class.HTPrescriptiveContentControllerTest.prescriptiveContentControllerTest1: 1
```

JUnit

Jenkins Runtime Parameters

Project CD_ApexUnit_2_0_Job

This build requires parameters:

Parameter	Value	Description
Salesforce_org_url	https://na14.salesforce.com	Server URL for the org *
username	rama_krishna@salesforce.com	Username for the given org *
password	•••••••••••••	Password for the above Username *
manifest_Files_For_Apex_Test_Classes_To_Execute	ManifestFile.txt,ManifestFile1.txt	Name of the manifest file that contains the list of test classes to be executed. Multiple comma separated manifest files can be provided. Please do not include spaces while providing multiple manifest files.
regex_For_Apex_Test_Classes_To_Execute	Sample*,*Test	The regex to use for the tests belonging to the team. All test classes in the given org with name beginning with this parameter will be selected to run. Multiple comma separated regexes can be provided.Please do not include spaces while providing multiple regex.
manifest_Files_For_Apex_Source_Classes_to_compute_code_coverage	ClassManifestFile.txt	Name of the class manifest file that contains the list of source classes to be executed. These source classes will be used to calculate team code coverage. Multiple comma separated manifest files can be provided. Please do not include spaces while providing multiple manifest files.
regex_For_Apex_Classes_To_Compute_Code_Coverage	Sample*	The regex used by the team for the source Classes. All the source classes in the given org with names beginning with this parameter will be used to calculate team code coverage. Multiple comma separated regexes can be provided. Please do not include spaces while providing multiple regexes
Org_Wide_Code_Coverage_Percentage_Threshold	25	Org-wide code coverage percentage threshold(0-100)
team_Code_Coverage_Percentage_Threshold	50	Team code-wide code coverage percentage threshold (0-100) (Need Team apex class prefix/class manifest file for this)
client_id	••	client id for the org for OAUTH authorization * http://salesforce.stackexchange.com/questions/40346/where-do-i-find-the-client-id-and-client-secret-of-an-existing-connected-app
client_secret	••••••••••••••••••••	client secret for the org for OAUTH authorization *

Build