



HL7 FHIR DevDays 2017



Test Driven Development II - Advanced

Richard Ettema, AEGIS.net, Inc.



Amsterdam, 15-17 November | [@fhir_furore](#) | [#fhirdevdays17](#) | [www.fhirdevdays.com](#)

Presented by

- **Name:** Richard Ettema
- **Position:**
 - Lead Consultant, AEGIS.net, Inc.
 - FHIR[®] Certified Implementer
- **Background:**
 - 34+ years IT industry experience
 - 14+ years leading HIT development/implementation efforts
 - 4+ years contributing to the HL7[®] FHIR[®] specification (focus on testing)
 - Sr. Architect / Lead Developer for the Touchstone Project
 - Author of the AEGIS WildFHIR public test server and client

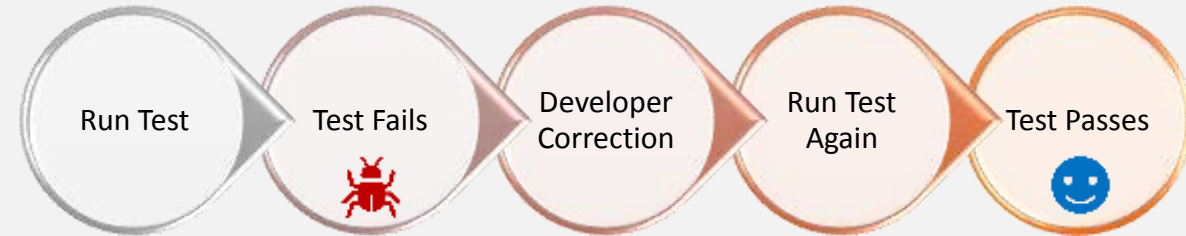
Test Driven Development with FHIR

Intro Session Review

- To ensure interoperability between applications claiming conformance to the specification, a testing framework has been established within the FHIR specification itself
<https://www.hl7.org/fhir/STU3/testing.html>
- This framework defines a Test Engine for processing a TestScript resource as a natural language, computable format of a test case
- The TestScript resource is an implementation-agnostic description of tests that allows test engines to evaluate if a FHIR implementation conforms with the FHIR specification
<https://www.hl7.org/fhir/STU3/testscript.html>

Reliable and Repeatable Testing

- Testing workflow example



- How can we trust that the correction is being tested in the same way?
Can we trust either test outcome?

Known data facilitates known, expected outcomes

We will review and examine how to ensure “Reliable and Repeatable Testing” during the Hands-on Exercises using a “Two Users, Same Data” scenario

Identifying Your Testing Criteria; a.k.a. Asserts

- Testing simple values
 - Is this the patient I expected?
 - Is this the operation response I expected?
 - Did I get multiple matches when I expected one?
- Testing specification conformance
 - Is this element required?
 - Is this code value correct?
- Testing conditionality or constraints
 - Can this element be expressed more than one way and still be compliant?
 - Is this element is only required when another element is present?

Asserts can perform...

- simple operations
 - standard assert operators are constrained to the following: **equals** | **notEquals** | **in** | **notIn** | **greaterThan** | **lessThan** | **empty** | **notEmpty** | **contains** | **notContains** | **eval**
 - specification allows for use of XPath or JSONPath expressions
- complex evaluations
 - specification allows for rules to be defined within the assert
 - specification allows for use of FHIRPath* expressions

**FHIRPath is supported within the TestScript resource and by the Touchstone test engine, but will not be a focus of this session. To learn more about FHIRPath, see <http://hl7.org/fhirpath>*

Complex Asserts

- Are used when testing conditionality, constraints or simplifying an existing test construct
- Support the use of FHIRPath
 - FHIRPath has expressions, functions, operations
 - Far more functionality than found in the simple assert operators
 - Built-in expressions such as dateTime and quantity
 - Expressions can contain operators
 - Functions like isDistinct, count, subsetOf, supersetOf
- Support Rules and Rulesets (high level language scripted logic)
 - Touchstone Rules Engine supports Groovy, Schematron and XSLT

Complex Asserts - FHIRPath Example

- FHIRPath provides a syntax neutral way to access the contents of a resource and can simplify assert logic constructs

Assert condition:

“Confirm that the searchset Bundle total number of entries is at least one (1).”

- Let’s compare XPath and JSONPath to FHIRPath...

Complex Asserts - FHIRPath Example

Comparison with XPath and JSONPath

- XPath assert

```
<assert>
  <operation value="greaterThan" />
  <path value="/Bundle/total" />
  <value value="0" />
</assert>
```

- JSONPath assert

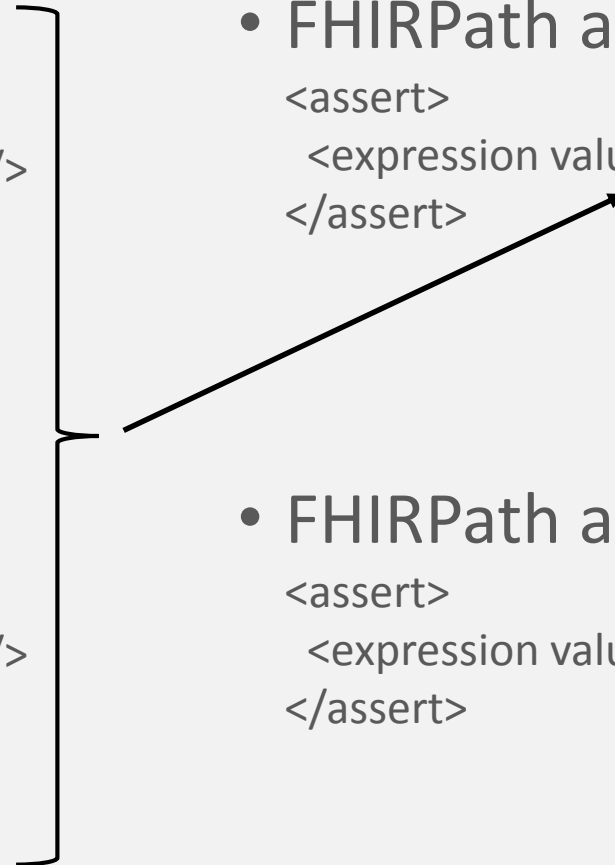
```
<assert>
  <operation value="greaterThan" />
  <path value=".total" />
  <value value="0" />
</assert>
```

- FHIRPath assert (equivalent logic)

```
<assert>
  <expression value="Bundle.total.toInteger() > 0" />
</assert>
```

- FHIRPath assert (alternative logic)

```
<assert>
  <expression value="Bundle.entry.count() > 0" />
</assert>
```



Complex Asserts - Rules and Rulesets

- The TestScript resource allows the use of Rules and Rulesets within the assert
- Rules are called during TestScript execution
 - In Touchstone, the Rules are hosted within the Test Definition folders
 - Rules may be viewed and will be displayed in Test Results if selected
- Rulesets are simply collections of one or more Rules
 - Allows for the definition of a group of Rules to be referenced within the TestScript in an efficient manner

We will review and examine Rules and Rulesets during the Hands-on Exercises

FHIR Client or Peer-to-Peer Testing

- TestScripts in Touchstone can be either Server-only, Client (Peer-to-Peer), or Multi-actor
 - **Server-only:** Touchstone initiates requests to the destination FHIR System and evaluates the response
 - **Client (Peer-to-Peer):** Touchstone waits for a request from the origin FHIR System to be received, evaluates the request, sends to the destination FHIR System, and evaluates the response
 - **Multi-actor:** Touchstone may act as the initiating system along with other FHIR Test Systems acting as either a Client or Server

Client (Peer-to-Peer) Testing - Test Setup

- Select a Client TestScript from the Test Definitions list
- Select and fill in any test data necessary for test execution
 - **Origin Test System (Client)**
 - Destination Test System (Server)
 - **(Dynamic) Variables**

Test Setup Save Execute

Name *	Scripts	Tests
FHIR3-0-1-DevDays17--TDD-2-Adv-03-FHIRClient-06-patient-search-	1	1

Origin (FHIR-Client) *

AEGIS.net, Inc. - QA WildFHIR-3-0-1-Client ▼

Destination (FHIR-Server) *

AEGIS.net, Inc. - QA WildFHIR FHIR-3-0-1 ▼

Delete	Test Script	Version	Description
✕	/FHIR3-0-1-DevDays17/TDD-2-Advanced/03-FHIRClient/JSON Format/TDD-2-Adv-03-FHIRClient-06-patient-search-json	1	FHIR Client Testing minimum data elen destination system and given.

Variables

patientFamily:

patientGiven:

patientIdentifier:

Client (Peer-to-Peer) Testing - Test Execution

Test Execution

Stop

Execute Again

Exec Id: 20171026044726824

Test Setup: FHIR3-0-1-DevDays17--TDD-2-Adv-03-FHIRClient-06-patient-search-json

Start Time: 10/26/2017 01:47:26PM

Executed By: Richard Ettema

End Time:
Organization: AEGIS.net, Inc.

Status: Waiting for Request
Origin: AEGIS.net, Inc. - QA WildFHIR-3-0-1-Client - 10.0.4.44

Duration: 35.964s

Destination: AEGIS.net, Inc. - QA WildFHIR FHIR-3-0-1 <http://10.0.4.44:8080/fhir3-0-1>
Test Scripts: 1


Refresh

Test Script Execution	Version	Latest	Spec	Description	Origin	Destination	Status	Start	End	Duration	Passed	Tests
/FHIR3-0-1-DevDays17/TDD-2-Advanced/03-FHIRClient/JSONFormat/TDD-2-Adv-03-FHIRClient-06-patient-search-json	1	1	FHIR 3.0.1	FHIR Client Testing - tests external FHIR Client and Server to search a JSON formatted Patient with minimum data elements: identifier, name.family, name.given, gender and birthDate. The origin and destination systems must support the search operation and Patient search parameters: identifier, family and given.	AEGIS.net, Inc. - QA WildFHIR-3-0-1-Client - 10.0.4.44	AEGIS.net, Inc. - QA WildFHIR FHIR-3-0-1 http://10.0.4.44:8080/fhir3-0-1	Waiting for Request	10/26/2017 01:47:28PM		34.057s	0 of 1	

Client (Peer-to-Peer) Testing - TestScript Execution

Tests

Test Name	Description	Status	Duration	
Test: PatientSearch	Test the search operation with JSON content. The expected response content is a searchset Bundle resource containing all matching Patient resource instances in JSON format.	Waiting for Request	0.966s	
Action	Description	Status	Duration	Details
Operation	search-type - Patient Origin: AEGIS.net, Inc. - QA WildFHIR-3-0-1-Client - 10.0.4.44 Destination: AEGIS.net, Inc. - QA WildFHIR FHIR-3-0-1http://10.0.4.44:8080/fhir3-0-1 <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> Submit the following request: Method: GET URL: http://touchstonedev.aegis.net:5576/1/fhir3-0-1/Patient?identifier=12345&given=Peter&family=Chalmers Headers: USER_KEY YlvSwX1dxtYkPuZPx2w Accept application/fhir+json </div>	Waiting for Request	0.003s	Type: search-type [hide] Resource: Patient Params: identifier , given , family Description: Search for matching Patients and verify the search results. Definition: ...

We will review and examine Client Testing during the Hands-on Exercises

Captured Message Exchanges

- The Exchanges dashboard in Touchstone allows users to view all captured request and response messages
- Touchstone will match an exchange with a test execution using these checks:
 - If your test server is already defined within Touchstone and is publicly accessible, the message can be matched by originating IP address
 - If that is not possible, the system attempts to use USER_KEY and ORG_KEY
 - Else, the system cannot match the message to a test execution

➔ Test Executions

➔ Test Execution

☰ History

➔ Exchanges

Exchanges

Initiated By: Executed By: Response Code: Exec Status:

Origin System:

Origin Org:

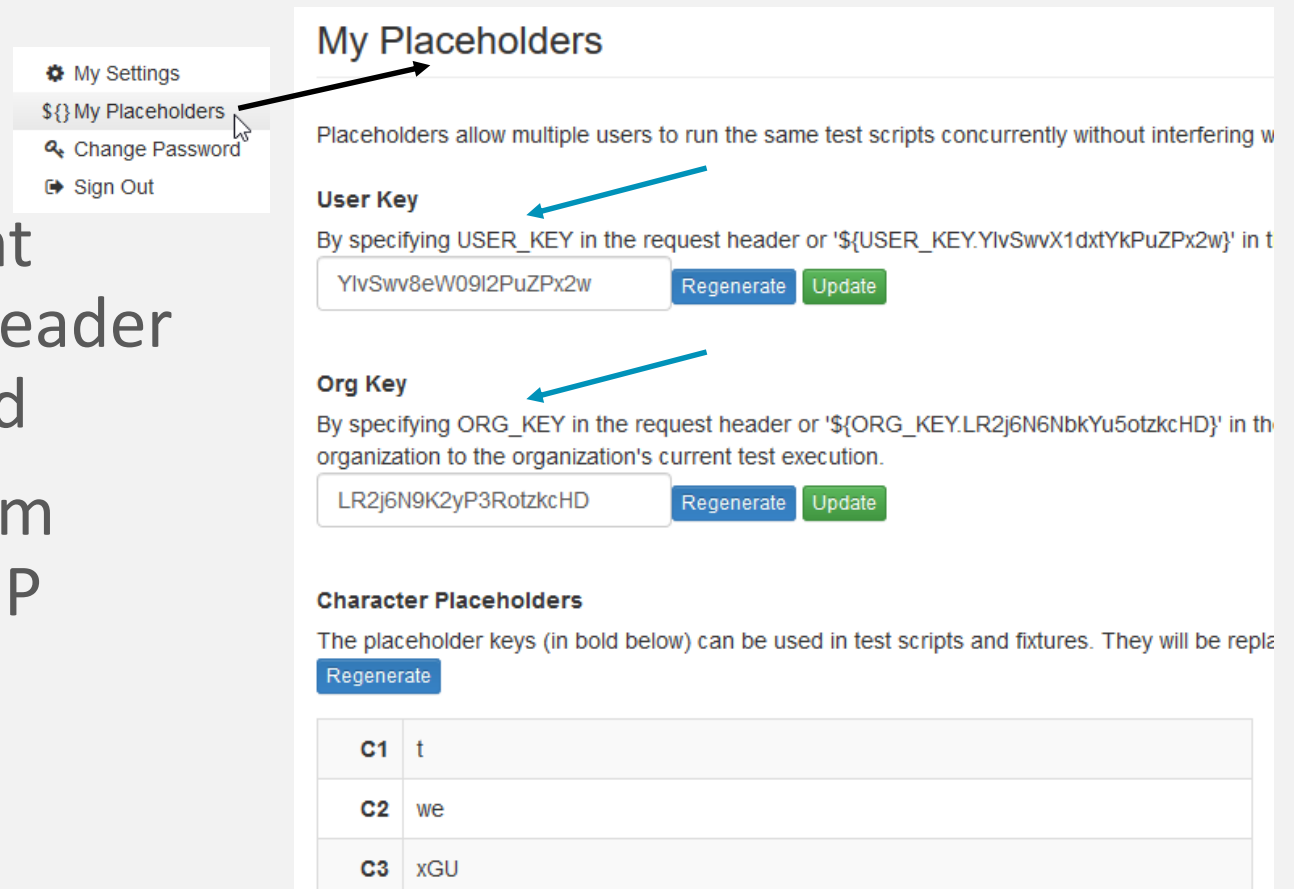
Dest System: Dest Org:

Records 1 - 10 of 10 Page Size:

Request Time	Origin	Exchange	Destination	Method	Status	Test Setup	Test Execution	Exec User	Matched
09/17/2016 05:04:04PM	WidFHIR FHIR-1.6.0 - 4.59: 159.80 AEGIS.net, Inc.	Request → Response	WidFHIR FHIR-1.6.0 - http://widfhir.aegis.n et:80/fhir/1.6.0/Patient/example AEGIS.net, Inc.	PATCH	200 OK		/Connection13/PATCH/JSON PatchPatient /connection-13-patch-patient-peer2peer-jun Step2-PatchPatient match Patient	Richard Enema	Matched via ORIGIN_IP

USER_KEY and ORG_KEY

- USER_KEY and ORG_KEY are special placeholders
- They can be sent by the client system as an HTTP request header or within the request payload
- They identify the client system when the machine name or IP address would not



My Placeholders

Placeholders allow multiple users to run the same test scripts concurrently without interfering w

User Key

By specifying USER_KEY in the request header or '\${USER_KEY.YlvSwvX1dxtYkPuZPx2w}' in t

YlvSwv8eW09I2PuZPx2w [Regenerate](#) [Update](#)

Org Key

By specifying ORG_KEY in the request header or '\${ORG_KEY.LR2j6N6NbkYu5otzkcHD}' in th organization to the organization's current test execution.

LR2j6N9K2yP3RotzkcHD [Regenerate](#) [Update](#)

Character Placeholders

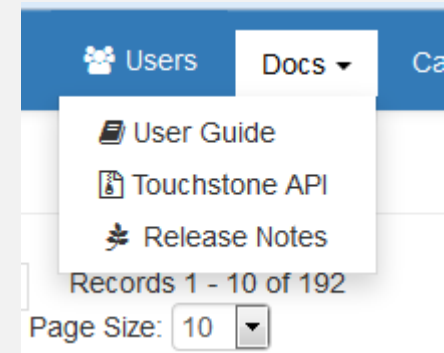
The placeholder keys (in bold below) can be used in test scripts and fixtures. They will be repl

[Regenerate](#)

C1	t
C2	we
C3	xGU

Introducing the Touchstone APIs

- Test executions can be launched and monitored via remote RESTful web services
- Provides the means to integrate Touchstone into your organization's test executions for:
 - Internal automated regressions tests
 - Continuous Integration build processes
- Documentation and example messages are available
- Supported formats – JSON and XML



Touchstone API Capabilities

- The Touchstone APIs allow for many of the same non-administrative functions as the User Interface
- Test executions launched on behalf of a remote test user via Touchstone API will be visible to all members of your organization on the Touchstone UI
- Any member of your organization can log in to the Touchstone UI and investigate test failures if needed

Touchstone API Services

POST authenticate → Authenticate with Touchstone; return new API session key

POST testExecution → Launch a new test execution for an existing Test Setup

GET testExecution → Return the status of a known test execution

GET testExecDetail → Return summary status of all test execution TestScripts

GET scriptExecDetail → Return test execution single TestScript details

GET testReport → Same as GET scriptExecDetail except returns FHIR TestReport

Touchstone API - Jenkins CI Server Integration Example

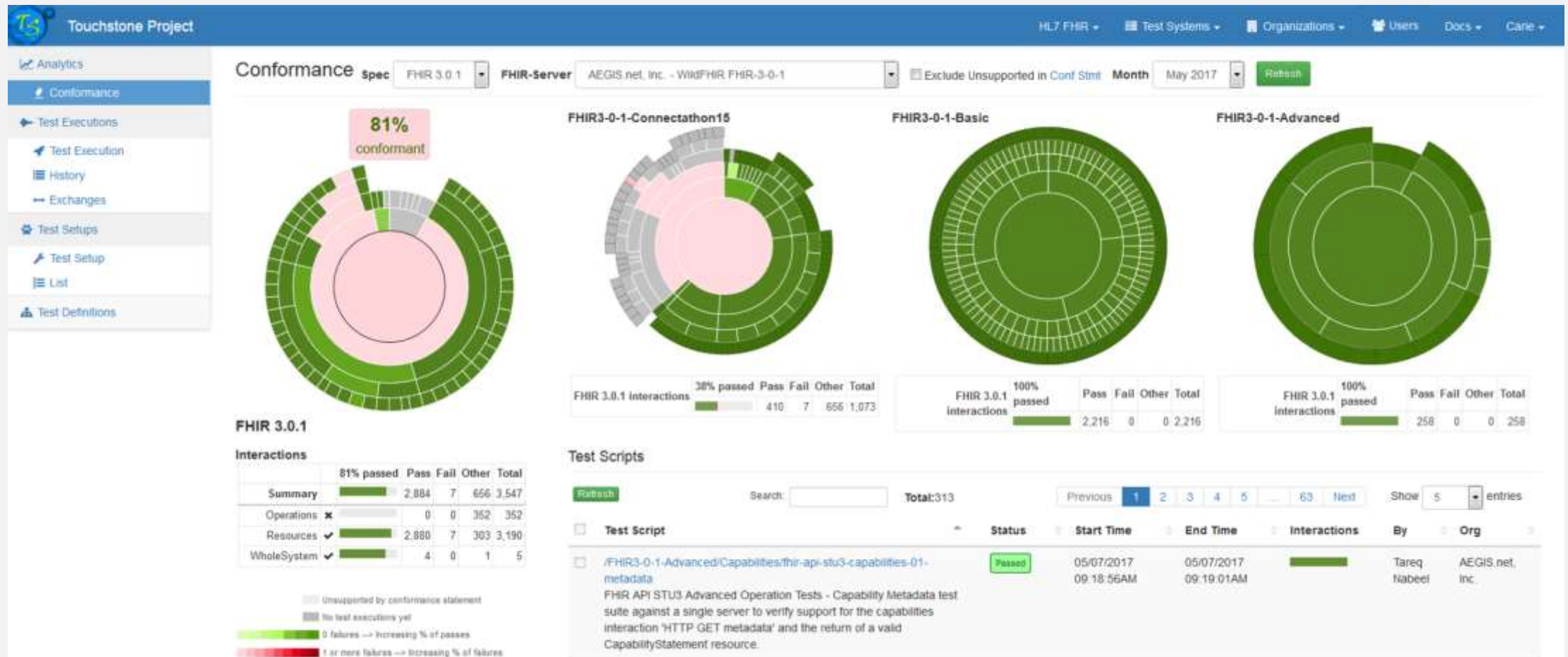
- The Touchstone User Guide documents a Jenkins CI Server Integration Example
 - <https://touchstone.aegis.net/touchstone/TouchstoneUserGuide>, section 'Touchstone API – Jenkins Integration Example'
 - Illustrates the use of the Jenkins Groovy Plugin
 - Builds on the example code from the previous 'Touchstone API – Definition' section

Due to not having a publicly available Jenkins CI Server at this event, there is no corresponding Hands-on Exercises for this topic.

Conformance Testing

- Conformance Testing validates a system against known standards
 - FHIR Specification (including Implementation Guides)
 - Version support (forward/backward compatibility)
- Continuous Conformance Testing shows that an organization is currently conformant, but also committed to remain conformant
 - Future development minded - interoperability needs to be addressed on a continuous basis
 - Development Aid - developers can integrate FHIR conformance testing into their build cycle to avoid costly conformance code rewrites later
 - Regression testing - validate that changes and enhancements are also conformant

Touchstone Conformance Analytics



Hands on Exercises

- Reliable and Repeatable Testing - Two Users, Same Data
- Complex Asserts
 - FHIRPath comparison to XPath and JSONPath
 - Rules and Rulesets
 - Advanced TestScripts
- FHIR Client / Peer-to-Peer
- Conformance Analytics Dashboard
- Connectathon Test Track
 - We will continue our review of one of the test tracks for the next HL7 FHIR Connectathon 17 event and the development of TestScripts



Discussion (Q & A)





HL7 FHIR DevDays 2017



Thank you!



Amsterdam, 15-17 November | [@fhir_furore](#) | [#fhirdevdays17](#) | [www.fhirdevdays.com](#)