02 - Test Execution - Basic Operations

Test Driven Development I - Intro – Exercise

Track: Test Driven Development I - Intro
Track lead: **Richard Ettema**
Email: richard.ettema@aegis.net

During this hands-on session of the Test Driven Development I - Intro tutorial we will explore TestScript Execution within the Touchstone Project environment. This exercise examines a series of basic FHIR operations (read, search, create, update and delete). Each operation's TestScript illustrates the use of different TestScript elements and as well as the basic mechanics of executing tests within Touchstone. The following generic steps will be used for each operation's TestScript, so you will need to perform them in sequence:

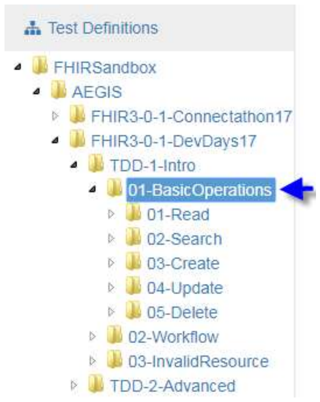
1. Create and execute a Test Setup
2. Examine the Test Execution interface
3. Examine the TestScript Execution interface

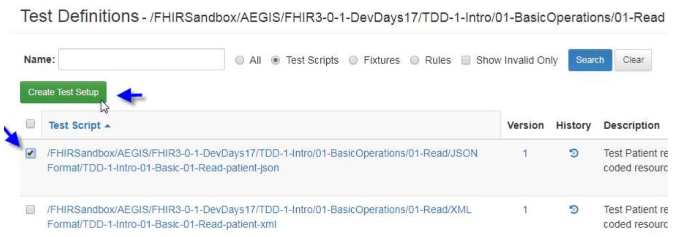
TestScripts for this exercise are in the **FHIRSandbox/AEGIS/FHIR3-0-1-DevDays18/TDD-1-Intro/02-BasicOperations Test Definitions*
***See page 4 for descriptions of each operation TestScript*

1. Create and execute a Test Setup

Related online document: Touchstone User Guide, Section 'Executing Tests'.
Creating and executing a Test Setup first starts with selecting one or more Test Definitions that will be included in the Test Setup.

- Navigate to and select the left menu bar item 'Test Definitions'. This will expand the top-level Test Definition Groups.
- Expand the **FHIRSandbox/AEGIS/FHIR3-0-1-DevDays18** Test Definition folder, the **TDD-1-Intro** folder and then **01-BasicOperations**
- Select an operation folder; e.g. **01-Read**, and the Test Definitions workspace display two TestScripts
  - Both TestScripts perform the same test(s)
  - The only difference is the FHIR syntax type of either JSON or XML
- In the Test Definitions workspace select either the JSON or XML TestScript and then select the Create Test Setup button

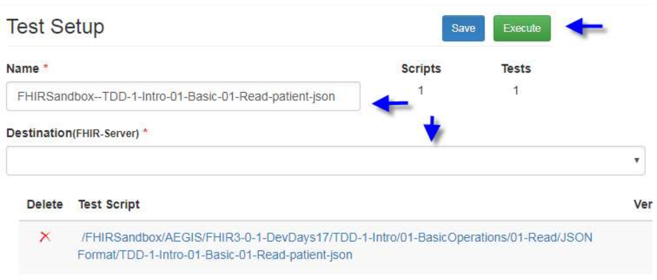

Test Definitions
- FHIRSandbox
  - AEGIS
    - FHIR3-0-1-Connectathon17
    - FHIR3-0-1-DevDays17
      - TDD-1-Intro
        - 01-BasicOperations
          - 01-Read
          - 02-Search
          - 03-Create
          - 04-Update
          - 05-Delete
        - 02-Workflow
        - 03-InvalidResource
      - TDD-2-Advanced


Test Definitions - /FHIRSandbox/AEGIS/FHIR3-0-1-DevDays17/TDD-1-Intro/01-BasicOperations/01-Read

Name: ○ All ● Test Scripts ○ Fixtures ○ Rules ☐ Show Invalid Only

	Test Script ▲	Version	History	Description
✓	/FHIRSandbox/AEGIS/FHIR3-0-1-DevDays17/TDD-1-Intro/01-BasicOperations/01-Read/JSON Format/TDD-1-Intro-01-Basic-01-Read-patient-json	1	⟲	Test Patient re coded resourc
☐	/FHIRSandbox/AEGIS/FHIR3-0-1-DevDays17/TDD-1-Intro/01-BasicOperations/01-Read/XML Format/TDD-1-Intro-01-Basic-01-Read-patient-xml	1	⟲	Test Patient re coded resourc

- The Test Setup workspace will be displayed where you
  - May change the auto-generated Test Setup name
  - Must choose a Destination Test System as the target of the test execution.
  - *You can choose the test system you created from the previous step. Please feel free to test against "AEGIS.net, Inc. - WildFHIR FHIR-3-0-1" public server to compare results.*


Test Setup

Save | Execute

Name *	Scripts	Tests
FHIRSandbox--TDD-1-Intro-01-Basic-01-Read-patient-json	1	1

Destination(FHIR-Server) *

Delete	Test Script	Ver
✗	/FHIRSandbox/AEGIS/FHIR3-0-1-DevDays17/TDD-1-Intro/01-BasicOperations/01-Read/JSON Format/TDD-1-Intro-01-Basic-01-Read-patient-json	

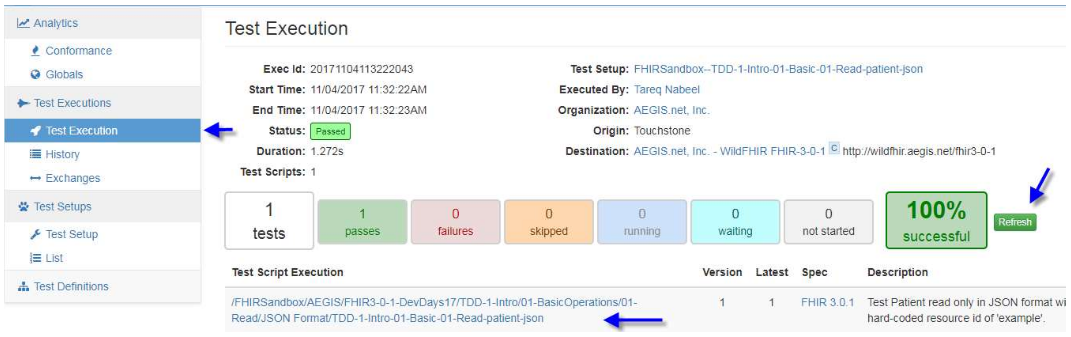
- After completing the Test Setup, select the Execute button
- The test execution will start, and you will be taken to the Test Execution screen (next step)

2. Test Execution Interface

Related online document: Touchstone User Guide, Section 'Test Execution Results'.

The Test Execution interface is where you can monitor the high-level execution status of your Test Setup.

- Refresh the browser page using the green 'Refresh' button or by directly refreshing your browser
- Explore the dashboard information
- Details of each individual TestScript Execution are available via the links under the 'Test Script Execution' column (next step)



Test Execution

Exec Id: 20171104113222043
Start Time: 11/04/2017 11:32:22AM
End Time: 11/04/2017 11:32:23AM
Status: Passed
Duration: 1.272s
Test Scripts: 1

Test Setup: FHIRSandbox--TDD-1-Intro-01-Basic-01-Read-patient-json
Executed By: Tareq Nabeel
Organization: AEGIS.net, Inc.
Origin: Touchstone
Destination: AEGIS.net, Inc. - WildFHIR FHIR-3-0-1 C http://wildfhir.aegis.net/fhir3-0-1

1 tests	1 passes	0 failures	0 skipped	0 running	0 waiting	0 not started	100% successful

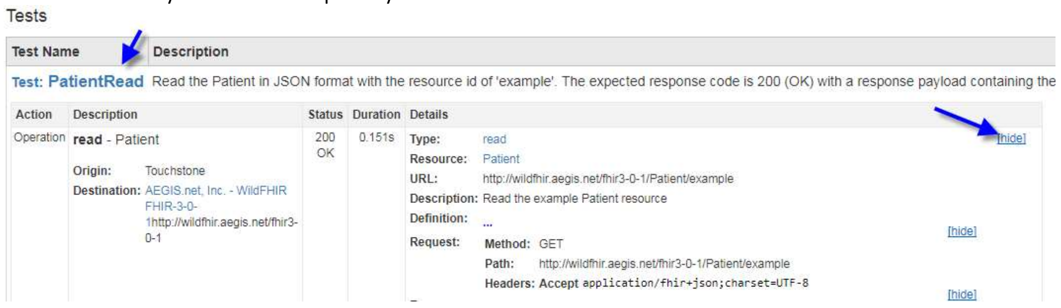
Refresh

Test Script Execution	Version	Latest	Spec	Description
/FHIRSandbox/AEGIS/FHIR3-0-1-DevDays17/TDD-1-Intro/01-BasicOperations/01-Read/JSON Format/TDD-1-Intro-01-Basic-01-Read-patient-json	1	1	FHIR 3.0.1	Test Patient read only in JSON format wi hard-coded resource id of 'example'.

3. TestScript Execution Interface

The TestScript Execution interface is where you can monitor the specific execution status of a TestScript.

- Refresh the browser page using the green 'Refresh' button or by directly refreshing your browser
- Explore the dashboard information
- Details of each individual test operation and assert are available
  - Many details are collapsed by default if there are no failures


Tests

Test Name	Description
Test: PatientRead	Read the Patient in JSON format with the resource id of 'example'. The expected response code is 200 (OK) with a response payload containing the

Action	Description	Status	Duration	Details
Operation	read - Patient Origin: Touchstone Destination: AEGIS.net, Inc. - WildFHIR FHIR-3-0-1http://wildfhir.aegis.net/fhir3-0-1	200 OK	0.151s	[hide] Type: read Resource: Patient URL: http://wildfhir.aegis.net/fhir3-0-1/Patient/example Description: Read the example Patient resource Definition: ... [hide] Request: Method: GET Path: http://wildfhir.aegis.net/fhir3-0-1/Patient/example Headers: Accept application/fhir+json;charset=UTF-8 [hide]

***

Have fun! And, remember to ask for help if you get stuck.

***

Please note:

- The exercises can be made in the hands-on area, where each track has its own table, indicated with a track sign. The track lead will be present for guidance and review.
- Exercises will only be discussed or reviewed during the HL7 FHIR DevDays 18 in Boston
- Any questions or remarks after the conference can be addressed in the FHIR chat on Zulip: https://chat.fhir.org

TestScript Descriptions

All TestScripts for this exercise use the Patient resource type. Each TestScript progressively illustrates the use of more TestScript features.

01-Read

Test the read operation for a Patient resource using a hard-coded resource id of ‘example’.
Features

- ❖ Asserts test for HTTP response code, Content-Type header and correct FHIR resource type payload
- ❖ Observe use of hard-coded resource id; *how does this limit the usefulness of this TestScript?*

02-Search

Test the search operation for Patient resources matching a hard-coded family search parameter of 'Chalmers'.
Features

- ❖ Asserts test for HTTP response code, Content-Type header and correct FHIR resource type payload
- ❖ Observe use of hard-coded search parameter; *how does this limit the usefulness of this TestScript?*

03-Create

There are two TestScripts:

- ➢ First - tests the create operation of a Patient resource only
- ➢ Second - tests the create operation with another test to search for the created Patient resource

Features

- ❖ Introduces the use of fixtures; *examine the fixture and note the static contents*
- ❖ Observe the asserts for the create operation; *why only test for HTTP response code and Location header?*
- ❖ Observe the 2nd TestScript use of the search operation; *why might this be useful/necessary?*

04-Update

Test the update operation of a Patient resource. A second test with a read operation is used to validate the update operation’s success.
Features

- ❖ Introduces the use of variables; *examine the variable definitions and use in the asserts following the read operation*
- ❖ Introduces the use of the setup element prior to all tests; *examine the setup and note the use of the conditional delete and create operations; what is the benefit of using setup versus tests?*
- ❖ Introduces the use of profiles; *asserts invoke the FHIR Validation Engine using profiles*
- ❖ Note the use of a single test with multiple operations - update followed by read; *what is the benefit of this paradigm versus individual tests (one per operation)?*

05-Delete

Test the delete operation of a Patient resource. A second test with a read operation is used to validate the delete operation’s success.
Features

- ❖ Continues the use of variables; *examine the variable definitions and their use*
- ❖ Continues the use of the setup element prior to all tests; *examine the setup and note the use of the conditional delete and create operations*

Note the use of a single test with multiple operations - delete followed by read; *what is the benefit of this paradigm versus individual tests (one per operation)?*