Testing FHIR Data Exchanges

SCORING SOFTWARE EXPLORATION REPORT

Trevor Andrus for JP Systems Data Quality

Outline

Introduction

Purpose

- FHIR
- Mitre
- ONC

Installation

Process

- Docker
- WSL
- Browser Alternative
- Demonstration

Walkthrough

Problems

- FHIR Server
- API compatibility
- Results

Conclusion

Possible applications

- Data testing
- Server testing
- Resources

Introduction

What is Inferno?
Where did Inferno come from?
Who is responsible for Inferno's development?

What is Inferno?

"Inferno is an open source tool for testing data exchanges enabled by the Fast Healthcare Interoperability Resources (FHIR) standard."

- Github Documentation

In short, Inferno was created to test FHIR servers.

FHIR servers can consist of a combination of a few elements, but, generally speaking consist of two parts:

API - (Applicaiton programming interface) - a gateway that allows two applications to communicate - in this case, it gives permissions and allows access to patient data within databases.

Databases - This is where patient data is stored. Servers can contain millions of records that can be accessed through the API.

Inferno was designed to test both of these parts - it tests both how well the data is organized according to FHIR standards, but also how the FHIR server reacts to queries for data.

Where did Inferno Come from?

Inferno is the result of a collaboration between **ONC** (Office of the National Coordinator for Health Information Technology) and **MITRE** (A federally funded research and development organization).

MITRE was experimenting with scoring softwares in a project called Crucible. ONC contracted with MITRE to develop software to test a specific ONC implementation guide built upon the Crucible framework. The collaboration has resulted in two editions of Inferno:

Program Edition - An edition of Inferno aimed specifically at testing the ONC Standardized API for Patient and Population Service

Community Edition - An edition of inferno that tests a number of FHIR Implementation Guides - users can either run existing Community Edition tests, or author their own tests using Inferno's open testing language.

ONC

- Office of national coordinator for health information technology

- Principal federal entity charged with coordination of health information technology and the electronic exchange of health information

MITRE

- FFRDC - Federally funded research and development center

- Sponsors both federally funded R&D centers and public-private partnerships

- Contracted with ONC to build Inferno on the model they developed in the Crucible project

Installing Inferno

Docker
WSL Containerization

Installing Inferno

Because of the open source nature of Inferno, the application is still in development and available through an online Github repository. To retrieve and use the program takes a couple more steps than simply downloading an .exe file.

Step 1: Download Docker

Through the instructions on Github, Inferno is only currently only available through Docker

Step 2: Enable WSL Compatability

Docker can only be implemented with a back-end powered by WSL: a linuxed based kernel for Windows operating systems

Step 3: Download the Inferno image, and run through Docker.

After these two preliminary steps are finished, the Inferno image can be downloaded and run.

Docker is a containerization platform designed to maximize ease of access to software in development

PROJECTS ARE ACCESSED THROUGH GITHUB, AND BUILT IN AN INDEPENDENT ENVIRONMENT TO PREVENT BUILD ERRORS

Rather than spend the time to optimize for multiple operating systems, Inferno developers use docker to allow the application to run on any operating system.

WSL
Containerization

(Noun), running (a program or application) in a discrete environment set up within an operating system specifically for that purpose and allocated only essential resources

Windows subsystem for Linux

The installation of Docker requires reconfiguration of Windows operating systems. This can be done through the Powershell. Virtualization capability must be enabled - meaning the native OS will allow a separate lightweight computer to be built on top of it.

Light and efficient

This eliminates the need for running an entire new operating system for a single application, and allows developers (such as those working on Inferno) to distribute their application in a controlled environment.

In short, **Docker** helps developers ensure their software runs as intended. In order to do so, we must enable the computer's containerization compatibility - this is often done by downloading the Windows Subsystem for Linux (WSL).

WE WON'T GO OVER THE SPECIFICS OF DOCKER INSTALLATION

In order to install and have Docker working properly (with WSL compatibility) there is a bit of command line interface work that is required. Those familiar with Git will have the know-how necessary to get it working without issue.

Using Inferno

Github Instructions

Imbedded Test Cases

Inferno also allows for testing through a browser-based instance

THIS LIGHT WEIGHT VERSION ALLOWS US TO VIEW SIMPLE FUNCTIONALITY

For using inferno extensively, it is suggested to download the full version using the process we previously discussed, but the Inferno Github repository is equipped with a browser-based walkthrough to illustrate its concepts.

Walkthrough
Commentary

A side by side look at the openly available Inferno Documentation

Walkthrough Outline

As provided on the Inferno Github page in the README file, these are the current instructions for creating a working instance of Inferno for testing.

- Step 1: Build a SMART-on-FHIR Sandbox
- Step 2: Open Inferno
- Step 3: Select FHIR Version and Enter FHIR Endpoint
- Step 4: Perform Discovery and Registration Tests
- Step 5: Perform Standalone Patient App Tests
- Step 6: Perform EHR Launch App Tests
- Step 7: Perform Data Access Tests
- Step 8: Review Results

Instructions Provided

- **Step 1: Build a SMART-on-FHIR Sandbox**
- **Step 2: Open Inferno**
- **Step 3: Select FHIR Version and Enter FHIR Endpoint**

Deprecated Results

This step for the most part goes as described - the server host, however has changed. Follow the redirection to the new host and continue as described.

After installation (or using the browser alternative) Inferno opens correctly

Select the FHIR version as described in the instructions, then copy the endpoint from the sandbox documentation

Instructions Provided

- **Step 4: Perform Discovery and Registration Tests**
- **Step 5: Perform Standalone Patient App Tests**
- **Step 6: Perform EHR Launch App Tests**
- **Step 7: Perform Data Access Tests**
- **Step 8: Review Results**

Deprecated Results

It seems that the testing stage of the instructions are where things start break down

Sandboxes can be created, but when linked to Inferno for testing, almost all tests fail.

This renders viewing results and performing data access tests impossible.

Despite **Inferno's** deprecated Github walkthrough, the developers have included **cloud-based test cases** within the software itself.

THIS ALLOWS US TO SEE WHAT A PROPER RESULT SHOULD LOOK LIKE

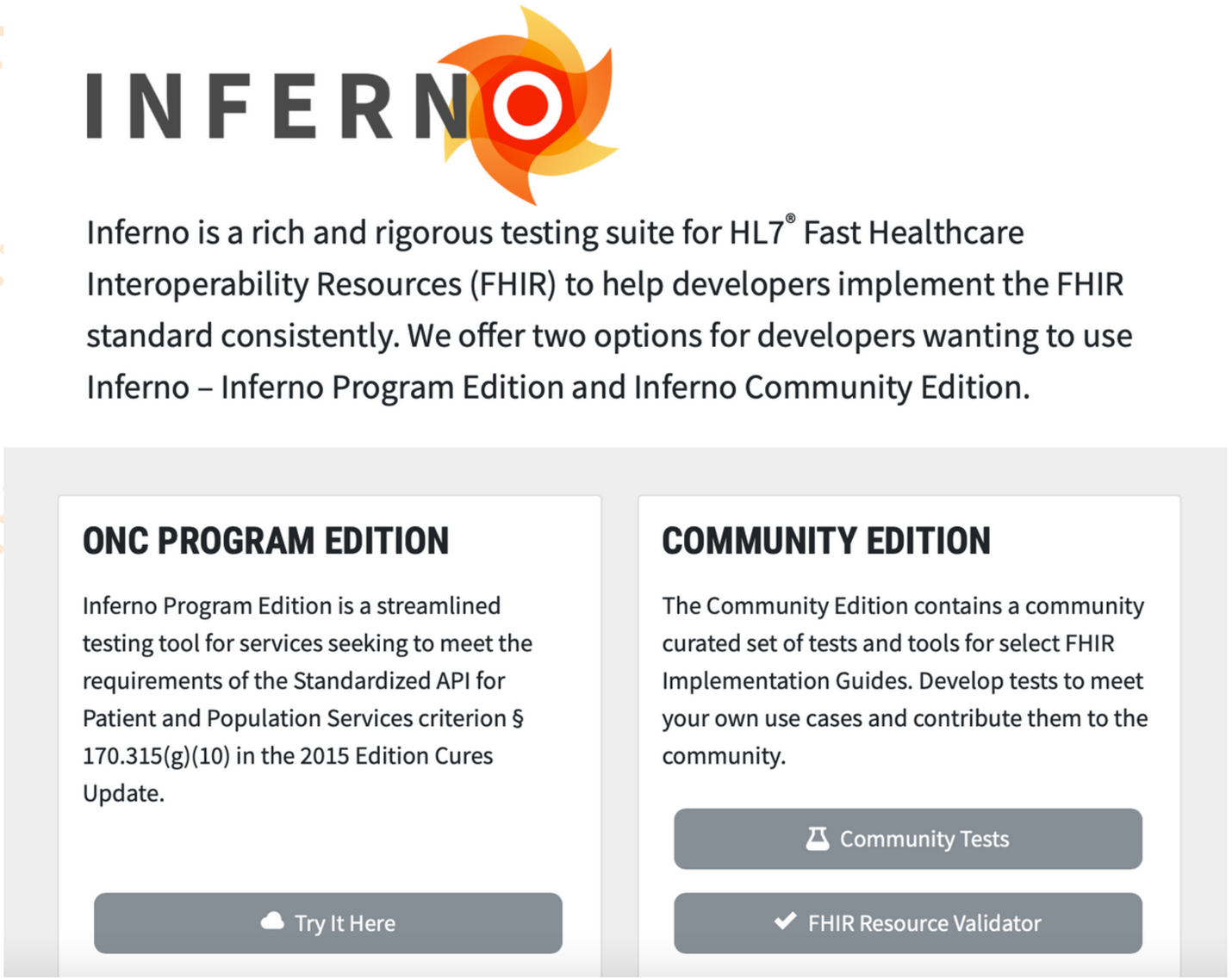
While further documentation may be required to get an instance of Inferno working on consumer data, the provided test cases gives a look at what the software is designed to do.

Inferno In Action

Demonstration with
Imbedded Test Cases

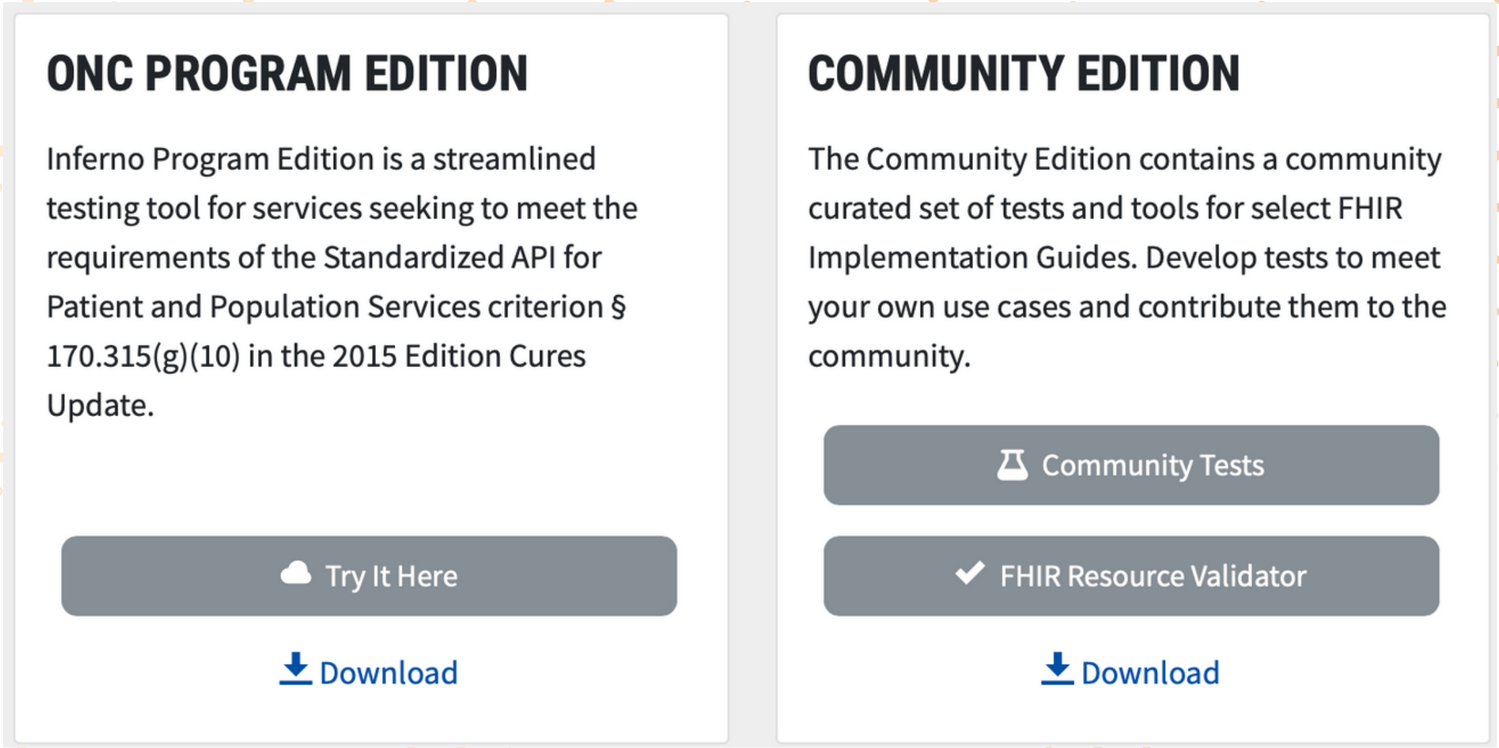
Landing Page

After traveling to https://inferno.healthit.gov/, this is the landing page we see to access the Inferno validator. We can see the two versions previously discussed.



Select Program Edition

To access the imbedded test cases, select the "Try it Here" option form the ONC program edition on the left.

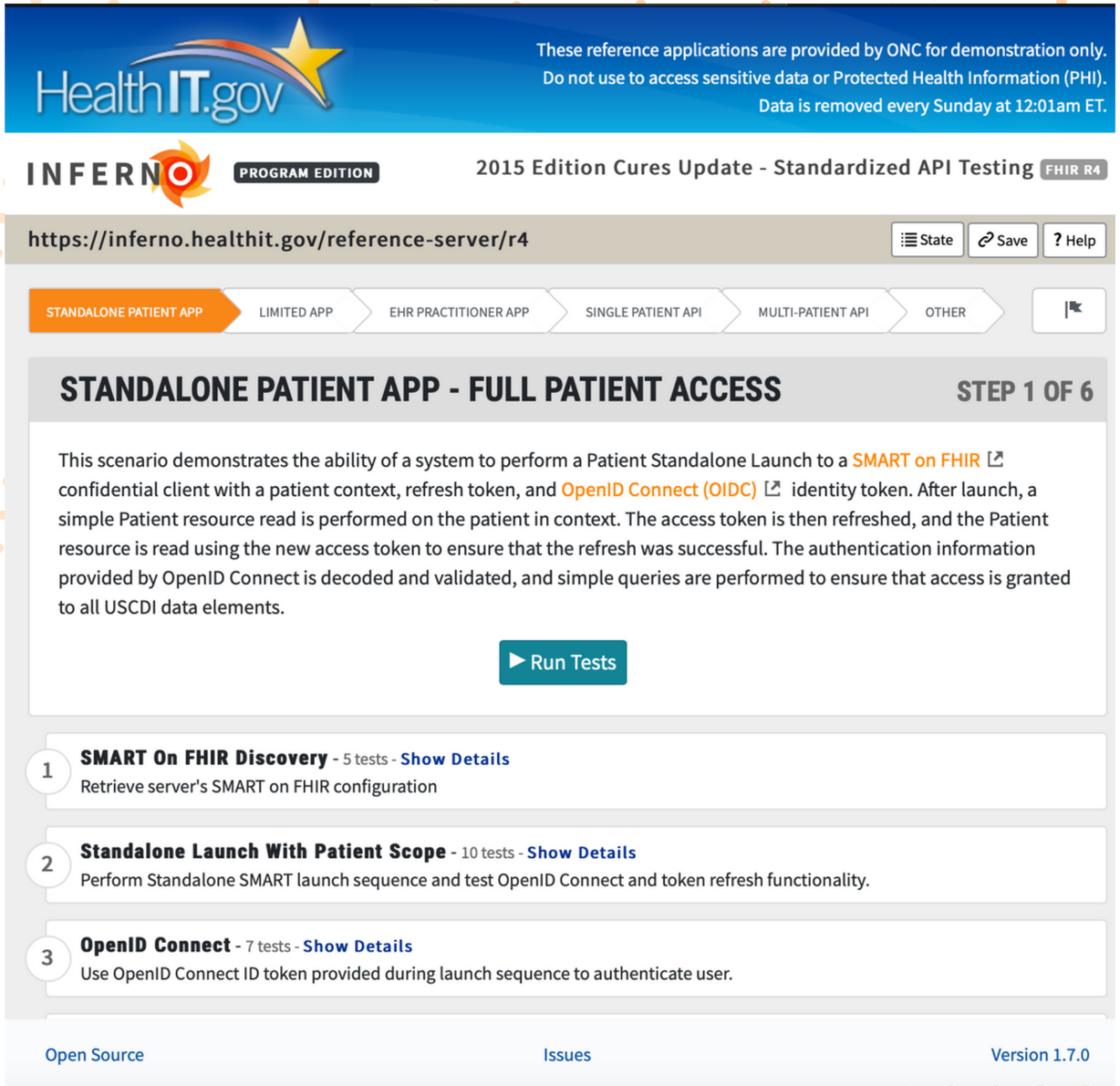


Select Inferno Reference Server

Under the 'Enter a FHIR Endpoint" dropdown menu, select "Inferno Reference Server". This allows us to access the prebuilt test cases developed by the Inferno team.

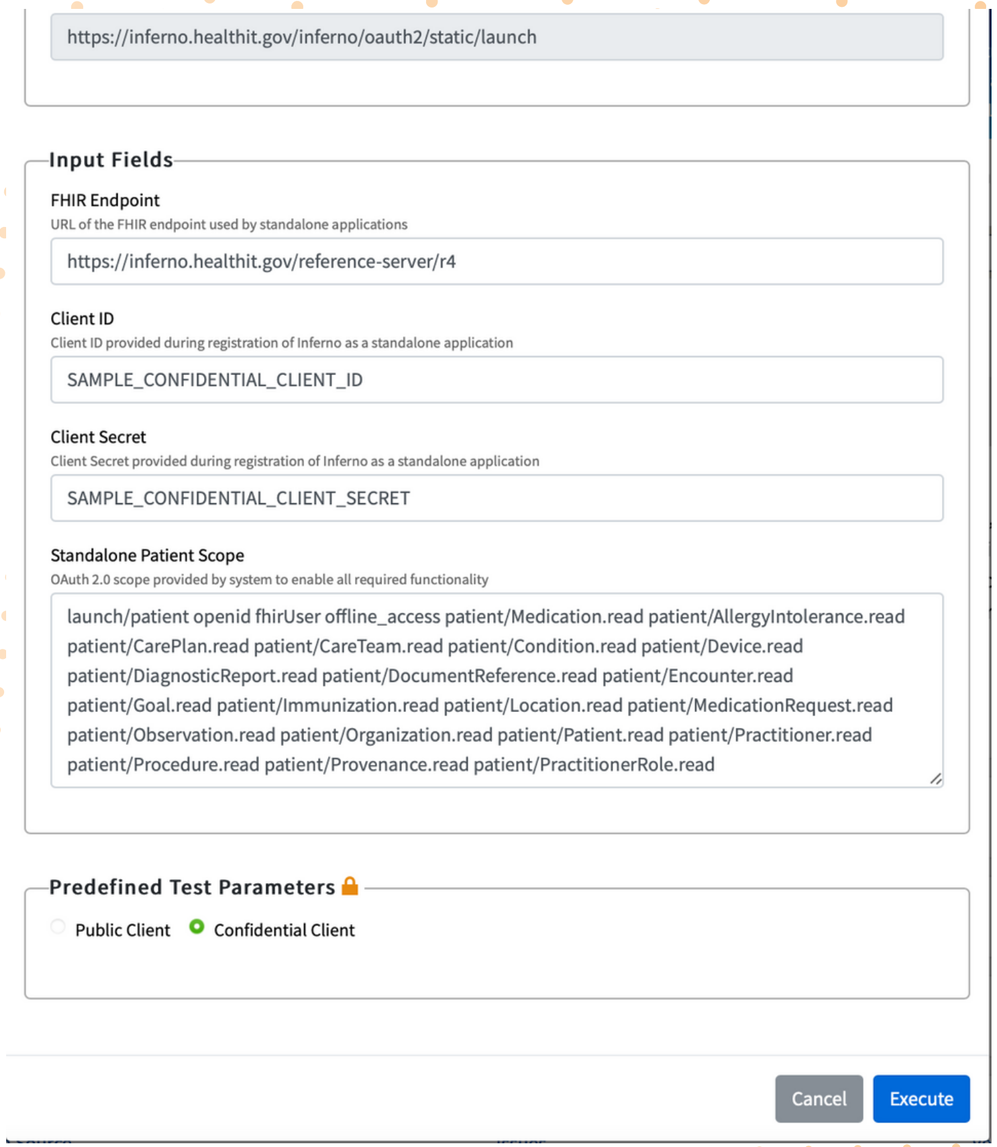
Run Tests

After selecting the Inferno Reference Server and clicking "Start Testing" You should be brought to the following page. Here we can see options for what tests Inferno will run, and under each option which tests will be performed. In this instance, we will be doing the Standalone Patient App tests - this tests if a third party app is able to access patient data on the FHIR server with proper authentication.



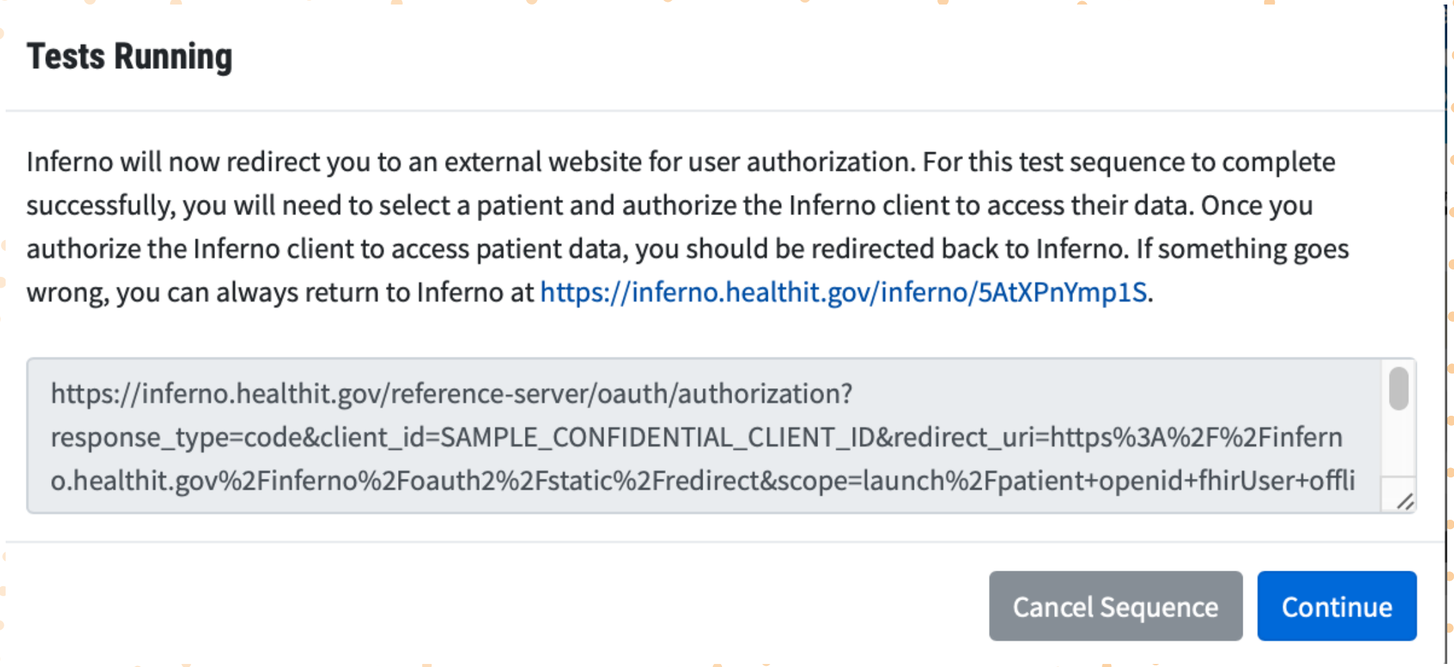
Execute

After clicking to run tests, we are presented with the following information: This pop-up window defines the FHIR endpoint for testing (the imbedded test server), and the specifications we will be testing on. All of these options should be left as default to run on the provided test server.



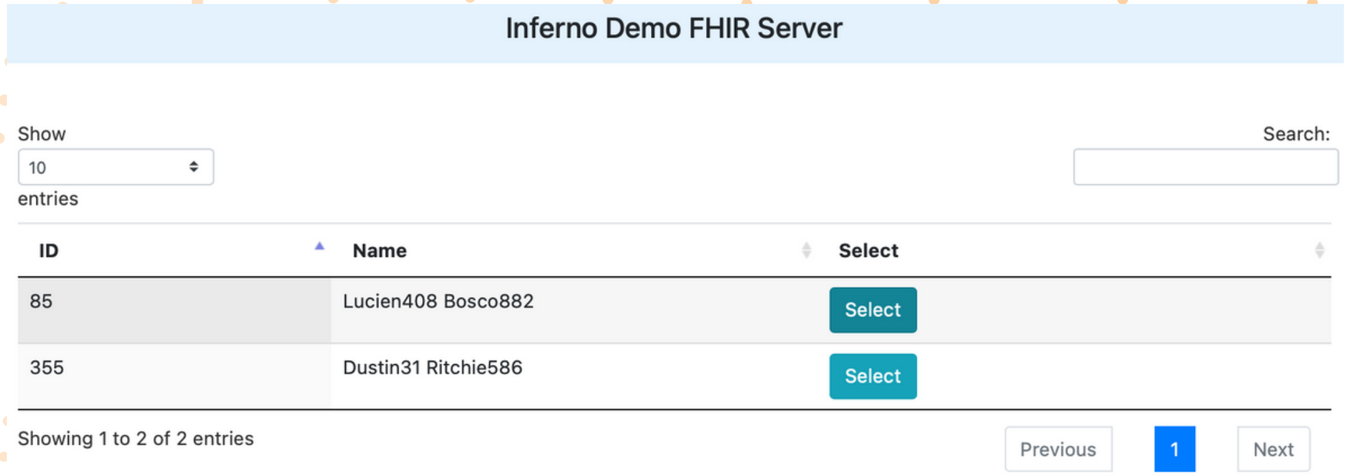
Continue

After pressing continue in this pop up window, we will be directed to an external website to define some final speciifications before receiving test results. In editions of Inferno downloaded through Docker, You would now be directed to a browser window.



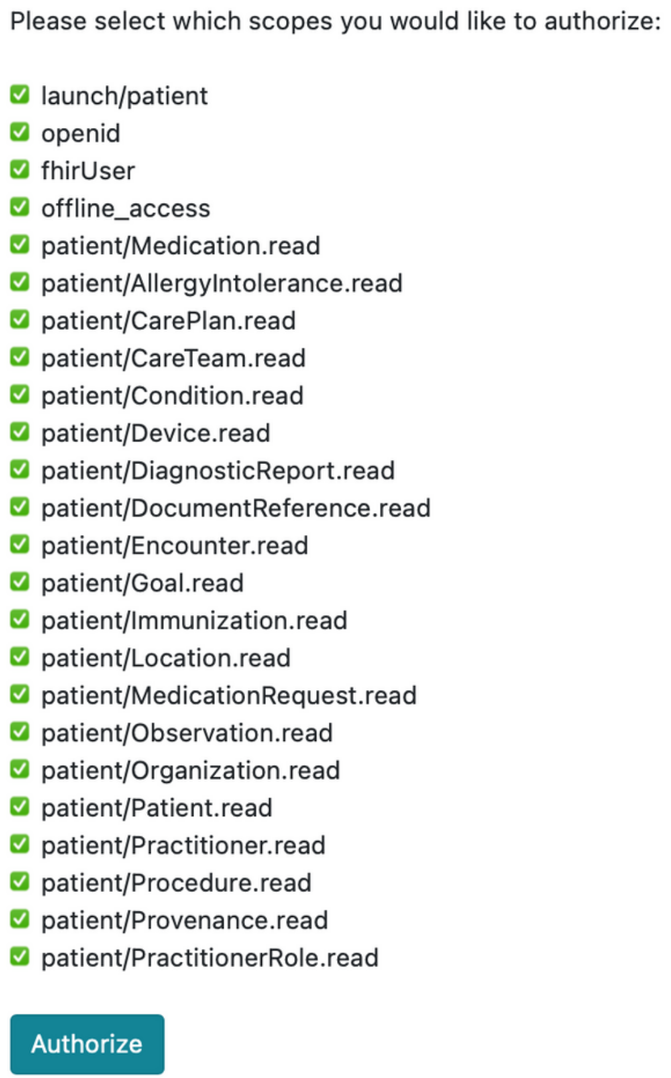
Select a Patient

After selecting continue, we are brought to the following page. here we pick a test patient to access. You may be presented with a variety of options, but all should work the same. I picked the first name for simplicity.



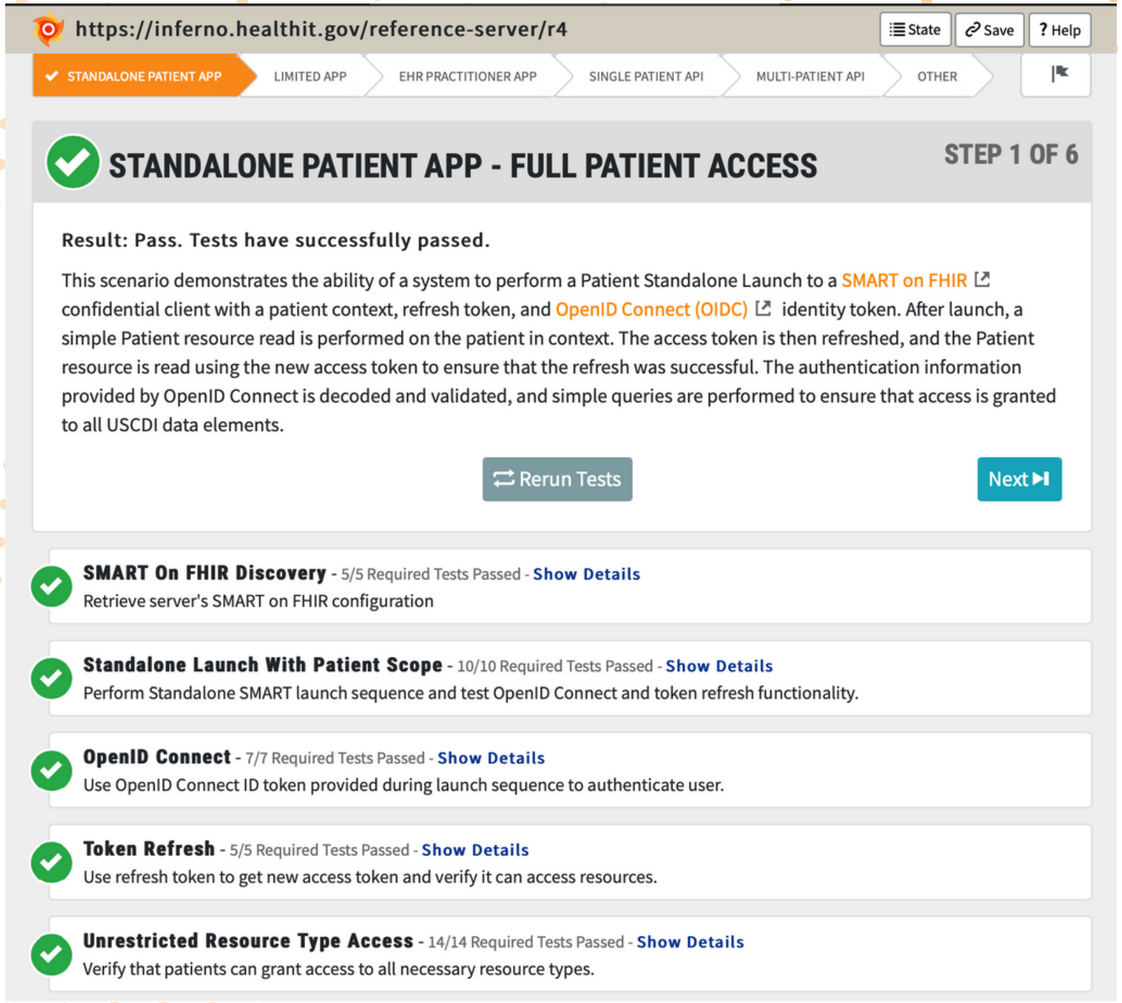
Select Authorizations

After selecting a patient, we are now asked to pick which authorizations to allow. (This simulates different 3rd party application authorizations). For example, if we unselect the "Medication Request" box, Inferno will give us an error and tell us we are unauthorized to access the medications for the patient.



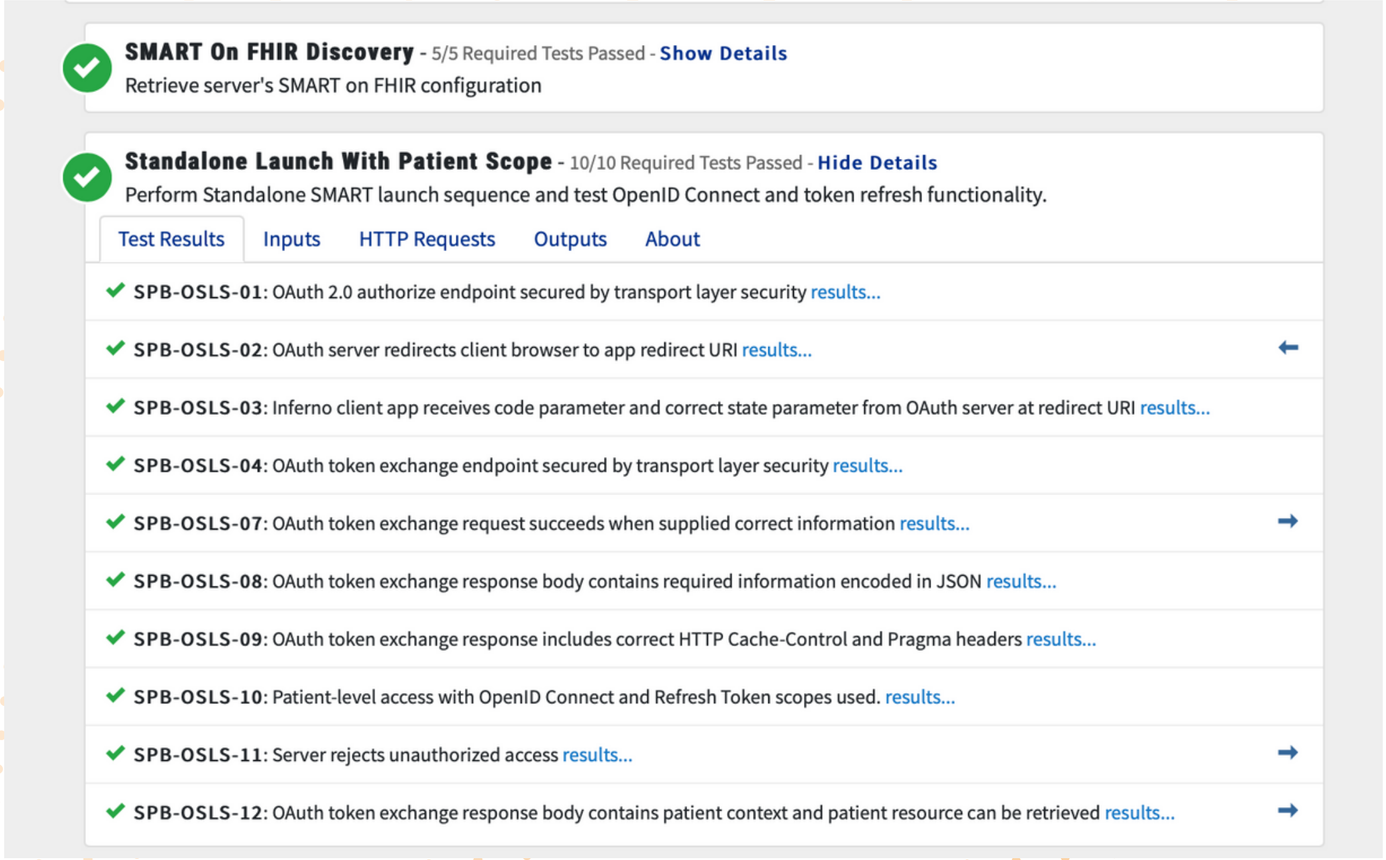
Access Results

After granting authorizations, Inferno will run tests to determine if the server abides by (In this case the ONC) implementation guides. It tests both the reactivity of the server, and the quality of the data contained within the database. From the tests shown, you can select a drop down menu to see the return codes for each individual test run.



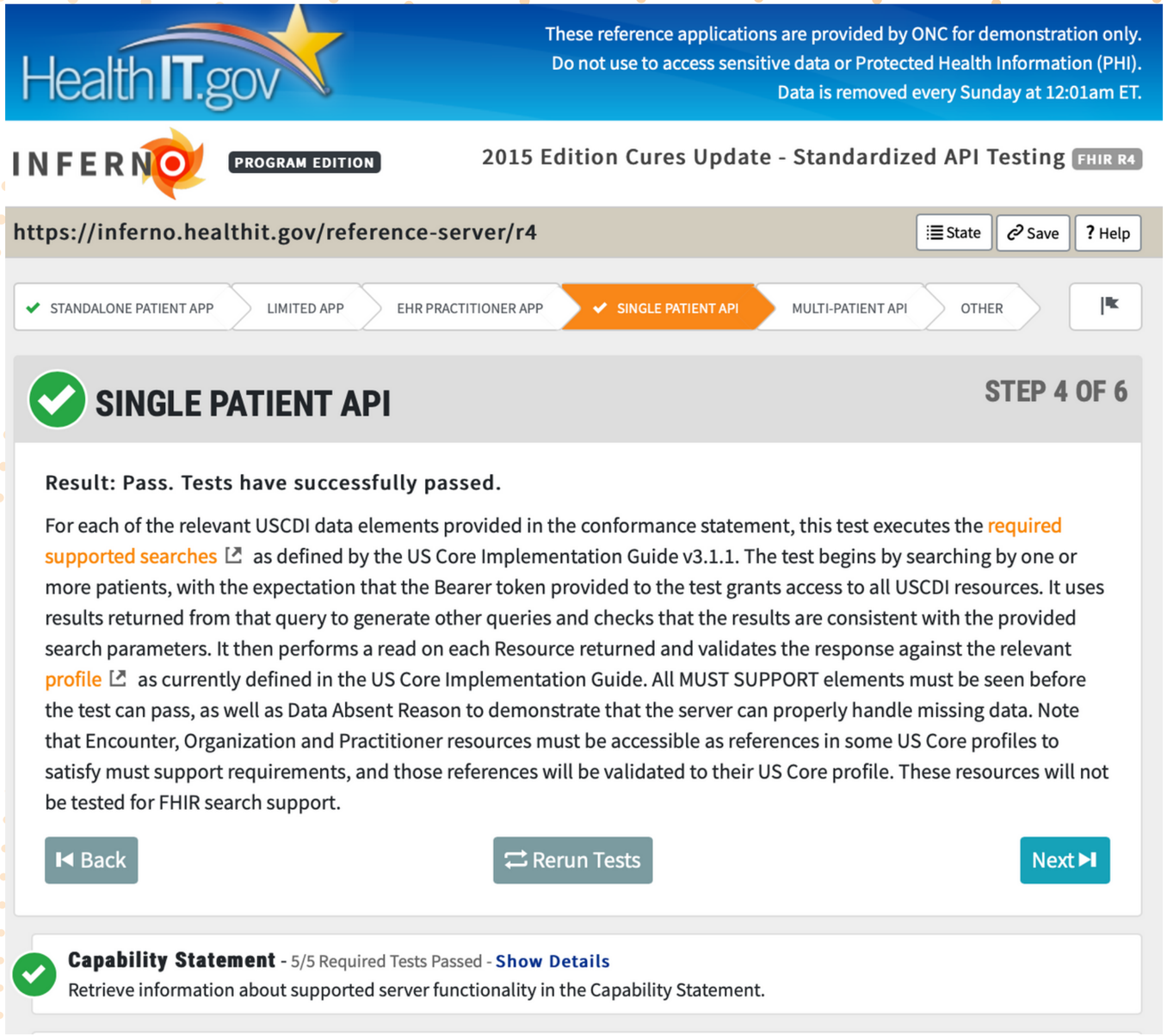
Interpreting Results

Admittedly, the return codes and responses for individual tests are pretty dense for someone (like myself) unfamiliar with building and testing servers.



Additional Testing

After performing tests for Standalone patient apps, additional tests can be run for different use cases - API access, practitioner access, and other tests can be run to assess how the FHIR server conforms to the implementation guides.



Conclusion

Impressions
Applications
Resources

Inferno Impressions

While still in development, Inferno seems a promising tool in testing the production of FHIR servers. However, the extent to which JP Systems may have use of it is questionable.

Inferno is mainly aimed at testing API and Server compatibility

Server Development - As opposed to programs like Diameter health that JP Systems currently uses, Inferno seems to be more aimed towards those developing new FHIR servers. More attention (and tests) are devoted to testing how servers respond to requests than the quality of data contained in the database.

Data Quality - While there are some tests devoted to the quality of data stored within the FHIR servers, this does not seem to be the main purpose of Inferno. While further development may include increased attention to data quality, it currently takes the back seat to server compatibility testing.

Inferno Applications

For JP Systems data quality, the current applications of Inferno are limited. It is incredibly difficult to get a working instance of the software outside of the built in test cases, and the analysis of data quality is not currently a priority.

Further development is needed before Inferno can be used effectively

Server Testing - Even in its applications as a server testing software, Inferno has a long way to come before it can be used extensively. The lack of documentation (and deprecated instructions on Github) make running the software difficult. Once running, the test results are also difficult to decode, and provide little help to those not trained technically in server architecture.

Data Quality - As was previously stated, data quality is not currently the priority for Inferno, and until (or if) improvements to this are made, Inferno may not be of much use for JP Systems data quality.

Sources

ONC is FHIR’d up: Unwrapping the new Inferno Testing Suite - Health IT Buzz

https://www.healthit.gov/buzz-blog/interoperability/onc-is-fhird-up-unwrapping-the-new-inferno-testing-suite

Inferno Program Edition Documentation

https://github.com/onc-healthit/inferno-program

Inferno Community Edition Documentation

https://github.com/onc-healthit/inferno

Github Walkthrough

https://github.com/onc-healthit/inferno/wiki/Walkthrough

Inferno

https://inferno.healthit.gov