Public Test Servers

Introduction

This page lists FHIR servers that are publicly available for testing. These are public services provided by volunteers and HL7 makes no representations concerning their safety or reliability.

In order to avoid spam etc, the servers are generally password protected. A contact is provided to get a password.

BTW: List of publically available test data (some of these test servers preload some of this data):

- [Base: What is in the specification]
- [Smart on FHIR test data]

Status

The status of publicly available FHIR servers is monitored using UptimeRobot at https://stats.uptimerobot.com/9DArDu1Jo. If you would like a publicly available FHIR server added to the UptimeRobot status page, email [Sean McIlvenna[1]].

Servers

Note that these servers are testing servers. They may be sporadically unavailable, and as the FHIR specification is a moving target, they may not always implement the latest version, or do so correctly.

- http://test.fhir.org/r2, http://test.fhir.org/r3 and http://test.fhir.org/r4 - Grahame's test server
    - Supports all resource types, all operations, xml + json
    - implementation details: open source - see [[2]]
    - supports Smart on FHIR

- HSPC Sandbox
    - http://sandbox.hspconsortium.org
    - Free DSTU2 and STU3 open sandboxes with tools for managing data. Both personal and team sandboxes available.
    - Supports both open and SMART on FHIR OAuth2 access
    - Supports app registration for SMART on FHIR apps
    - Supports all resource types, all operations
    - http://hspconsortium.org/#/
    - https://healthservices.atlassian.net/wiki/display/HSPC/Healthcare+Services+Platform+Consortium

- Vonk - .NET based FHIR Server by Firely
    - Demo servers
        - Stable: http://vonk.fire.ly (STU3 + R4)
        - Experimental: https://labs.vonk.fire.ly/ (Including R5 support)
    - Supports STU3, R4 and the R5 pre-release
    - Functionality
        - Generic FHIR Server, for all types of resources, all search parameters, xml + json
        - Supports validation (for example: POST /Patient/$validate, with a Patient resource in the body) and many plugins.
        - This test instance runs on MongoDB and therefore can do batch but not transaction. (Transactions are supported on SQL Server.)
    - Download your own instance - More information - Documentation

- HAPI FHIR Reference Server
    - Supports all resources and many FHIR features including Subscriptions, Terminology Services, etc.
    - Fully open source (Apache 2.0 License)
    - Project information: http://hapifhir.io
    - Endpoints:
        - Web UI: http://hapi.fhir.org
        - DSTU2 Endpoint: http://hapi.fhir.org/baseDstu2
        - STU3 Endpoint: http://hapi.fhir.org/baseDstu3
        - R4 Endpoint: http://hapi.fhir.org/baseR4
        - R5 Endpoint: http://hapi.fhir.org/baseR5

- http://spark.furore.com - Open source test server developed by Firely, maintained by Kufu.
    - C# reference implementation, WebApi 2.0 library, Mongo DB for storage and search.
    - The service endpoint is at http://spark.furore.com/fhir.
    - Supports DSTU 2, STU3, R4.

- Supports all resource types, all operations, xml + json
- Open source: [[3]]

- http://nprogram.azurewebsites.net - Rik Smithies/NProgram test server (STU 3)
  - patient, organization, diagnostic report, value set resources, read only, xml + json (C#)
  - DSTU 2 here http://fhir-dstu2-nprogram.azurewebsites.net/

- http://fhir.oridashi.com.au - Brett Esler/Oridashi demo servers overview
  - read-only implementation, C#, self-host web server, SQL Server DB - Facade on CIS
  - R4: 'Best Practice CIS' http://demo.oridashi.com.au:8304
  - R4: 'Medical Director CIS' http://demo.oridashi.com.au:8305
  - STU3: 'Best Practice CIS' http://demo.oridashi.com.au:8297
  - STU3: 'Medical Director CIS' http://demo.oridashi.com.au:8298
  - DSTU2: 'Best Practice CIS' http://demo.oridashi.com.au:8290
  - DSTU2: 'Medical Director CIS' http://demo.oridashi.com.au:8291

- Pyro
  - Documentation: https://pyrohealth.net/
  - Pyro Server's FHIR STU3 endpoint: https://stu3.test.pyrohealth.net/fhir
  - Implementation: Generic STU V3.0.1 FHIR Server, all resource types, .NET (C#), MS SQL or PostgreSQL
  - Supports: All search parameters, _includes, _revincludes, Search chaining, Conditional (Create, Update, Delete), History and Transaction bundles
  - Pyro Server's FHIR R4 endpoint: https://r4.test.pyrohealth.net/fhir
  - Implementation: Generic R4 V3.5.0 FHIR Server, all resource types, .NET (C#), MS SQL or PostgreSQL
  - Supports: All search parameters, _includes, _revincludes, Search chaining, Conditional (Create, Update, Delete), History and Transaction bundles

- https://fhir.smartplatforms.org - Josh Mandel / SMART Platforms
  - Open-source server in Grails (Java/Groovy) + MongoDB
  - "SMART on FHIR" Server: https://fhir-api.smartplatforms.org | https://fhir-open-api.smartplatforms.org | Source
  - "FHIR Starter" App Launcher https://fhir.smartplatforms.org | Source
- http://worden.globalgold.co.uk:8080/FHIR_a/hosted_demo.html Robert Worden / Open Mapping Software
  - Patient resource, read-only
  - Illustrates building a FHIR server on any existing application, any resource, by mapping to the application database
  - Tools to do this now available free, evolving

- https://health-samurai.io/aidbox - Health Samurai's FHIR server as a service with free layer, register and create a new server in 1 click
  - Local development installation by Docker
  - Supports all resource types, all operations, json + xml + edn + transit + yaml, All FHIR versions
  - Based on fhirbase (https://www.health-samurai.io/fhirbase)
  - Supports SQL on FHIR
  - Implementation: PostgreSQL, JVM (clojure), Kubernetes
  - Oauth 2.0, OpenID Connect, SCIM
  - Access Policies
  - Custom Resources & First Class Extensions
  - Terminology Service
  - Extensible with Custom Operations by Aidbox SDK

- Cerner's Sandbox
  - See our developer / API documentation for the resources that this server supports at http://fhir.cerner.com
  - Supports both open and OAuth 2 access
  - Contact us on our Google group for any issues or access to the OAuth 2 protected endpoints
  - Endpoints (DSTU2):
    - Open - https://fhir-open.sandboxcerner.com/dstu2/0b8a0111-e8e6-4c26-a91c-5069cbc6b1ca
    - With OAuth 2 Provider access - https://fhir-ehr.sandboxcerner.com/dstu2/0b8a0111-e8e6-4c26-a91c-5069cbc6b1ca
    - With OAuth 2 Patient access - https://fhir-myrecord.sandboxcerner.com/dstu2/0b8a0111-e8e6-4c26-a91c-5069cbc6b1ca

- http://open.epic.com/Interface/FHIR - Epic's Sandbox
  - Supports a subset of resource types, read-only
  - Runs against a functional Epic database
  - Includes online test harness for quick syntax checking
  - Contact open@epic.com for more information

- https://fhir-terminology-demo.e-imo.com/api/swagger/ui/index - IMO's FHIR Terminology Service

- AEGIS WildFHIR
  - **R4 Official v4.0.1** - http://wildfhir4.aegis.net/fhir4-0-1
    - Test client interface: http://wildfhir4.aegis.net/fhir4-0-1-gui
    - Supports the FHIR R4 Official (v4.0.1) version
    - Supports all resource types and standard operations, including batch, validate and json/xml patch, excluding transaction; search supports chained parameters, include and revinclude; support for conditional read/update/delete, create/update return preference and _summary parameter
  - R4 v4.0.0 - http://wildfhir4.aegis.net/fhir4-0-0
    - Test client interface: http://wildfhir4.aegis.net/fhir4-0-0-gui
    - Supports the FHIR R4 Official (v4.0.0) version
    - Supports all resource types and standard operations, including batch, validate and json/xml patch, excluding transaction; search supports chained parameters, include and revinclude; support for conditional read/update/delete, create/update return preference and _summary parameter
  - **STU3 Official v3.0.2** - http://wildfhir3.aegis.net/fhir3-0-2
    - Test client interface: http://wildfhir3.aegis.net/fhir3-0-2-gui

- Supports the FHIR Release 3 (v3.0.1-11917) version
- Supports all resource types and standard operations, including validate and json/xml patch, excluding transaction; search supports chained parameters; support for conditional read/delete, create/update return preference and _summary parameter
- STU3 v3.0.1 - http://wildfhir3.aegis.net/fhir3-0-1
  - Test client interface: http://wildfhir3.aegis.net/fhir3-0-1-gui
  - Supports the FHIR Release 3 (v3.0.1-11917) version
  - Supports all resource types and standard operations, including validate and json/xml patch, excluding transaction; search supports chained parameters; support for conditional read/delete, create/update return preference and _summary parameter
- **DSTU2 Official v1.0.2** - http://wildfhir2.aegis.net/fhir1-0-2
  - Test client interface: http://wildfhir2.aegis.net/fhir1-0-2-gui
  - Supports the FHIR DSTU 2.0 (v1.0.2-7202) version
  - Supports all resource types and operations, including validate, excluding transaction; support for conditional read, create/update return preference

- http://sqlonfhir-dstu2.azurewebsites.net/fhir - HealthConnex - DSTU2
  - Supports all resource types, but not all operations
  - .NET (C#) implementation on SQL Server Azure
  - No support for search chaining or profiles at present
- http://sqlonfhir-stu3.azurewebsites.net/fhir - Telstra Health (HealthConnex) - Jan 2017 San Antonio Connectathon Build
  - Supports all resource types, but not all operations
  - .NET (C#) implementation on SQL Server Azure
  - Limited support for search chaining or profiles at present (and terminology is really broken)

- http://health.gnusolidario.org:5000 GNU Health FHIR server
  - Supports read, validate and search for Patient, DiagnosticReport, Practitioner, Procedure, Observation, Condition, FamilyHistory
  - DSTU1
  - A Flask app. It's connected to the GNU Health community server database. Consequently, it's possible to create and update patients, doctors, etc. through the GNU Health frontend and the changes should be reflected on the FHIR server (indirect write support, I suppose).
  - I think our goal is to use the FHIR server as an adapter for non-GNU Health EHRs and users, since we already have synchronization between instances. But, there are other interesting possibilites, too.
  - I'm updating the code frequently and it's still in the dev stages (e.g., adding new resources frequently).
  - Documentation is here: https://en.wikibooks.org/w/index.php?title=GNU_Health/Using_the_FHIR_REST_server

- https://stu3.ontoserver.csiro.au/fhir - CSIRO's Ontoserver - STU3 (FHIR 3.0.1)
- https://r4.ontoserver.csiro.au/fhir - CSIRO's Ontoserver - R4 (still in beta)
  - Supports Terminology Services
  - CodeSystem, ValueSet, ConceptMap, StructureDefinition, and Bundle with read, create, update, delete, search
  - $expand, $lookup, $validate-code, $validate, $closure, $translate, and $batch
  - Specialised support for SNOMED CT, and LOINC

- http://fhir.i2b2.org/open/ i2b2 FHIR server - DSTU21
  - Supports read and search for Patient, Observation, Condition, Medication
  - DSTU21
  - demo calls see: http://fhir.i2b2.org
  - Connected to demo server at https://www.i2b2.org/webclient/

- https://tw171.open.allscripts.com/FHIR - Allscripts Touchworks Sandbox (OAuth)
  - DSTU 2.0 (v1.0.2) version of FHIR
  - Supports the resource types required for the Common Clinical Data Set, currently read-only
  - Supports OAuth validation
  - Supports SMART on FHIR
  - Runs against Allscripts Touchworks 17.1 test server
  - Please review our developer / API documentation at https://developer.allscripts.com/Content/fhir/content/TWFHIR17_Sandbox/
  - Contact Jeffrey Danford for more information about this endpoint
  - Contact Gunther Meyerfor OAuth setup

- https://pro171.open.allscripts.com/FHIR - Allscripts Professional Sandbox (OAuth)
  - DSTU 2.0 (v1.0.2) version of FHIR
  - Supports the resource types required for the Common Clinical Data Set, currently read-only
  - Supports OAuth validation
  - Supports SMART on FHIR
  - Runs against Allscripts Professional 17.1 test server
  - Please review our developer / API documentation at https://developer.allscripts.com/content/fhir/content/Pro171_Sandbox/
  - Contact Jeffrey Danford for more information about this endpoint
  - Contact Gunther Meyerfor OAuth setup

- https://scm163cu3.open.allscripts.com/FHIR - Allscripts Sunrise Sandbox (OAuth)
  - DSTU 2.0 (v1.0.2) version of FHIR
  - Supports the resource types required for the Common Clinical Data Set, currently read-only
  - Supports OAuth validation
  - Supports SMART on FHIR
  - Runs against Allscripts Sunrise 16.3 CU3 test server
  - Please review our developer / API documentation at https://developer.allscripts.com/Content/fhir/content/SCM163CU3_Sandbox/
  - Contact Jeffrey Danford for more information about this endpoint
  - Contact Gunther Meyerfor OAuth setup

- http://twdev.open.allscripts.com/FHIR - Allscripts STU3 Sandbox (OAuth)
  - STU 3.0 (v1.8.0) version of FHIR
  - Supports the resource types required for the Common Clinical Data Set, currently read-only

- Supports OAuth validation
- Supports SMART on FHIR
- Runs against Allscripts Touchworks 17.1 test server
- Please review our developer / API documentation at https://developer.allscripts.com/fhir
- Contact Jeffrey Danford for more information about this endpoint
- Contact Gunther Meyerfor OAuth setup

- http://52.72.172.54:8080/fhir/home - Michigan Health Information Network (open)
  - 1000s of realistic test patient records

- https://fhir.careevolution.com/Master.Adapter1.WebClient/fhir - CareEvolution HIEBus
  - DSTU2, read-write for most resources
  - STU3, read-write for most resources
  - R4, read-write for most resources
  - Open DSTU2 end-point: https://fhir.careevolution.com/apitest/fhir
  - Protected (SMART-on-FHIR) DSTU2 end-point: https://fhir.careevolution.com/Master.Adapter1.WebClient/api/fhir
  - Protected (SMART-on-FHIR) STU3 end-point: https://fhir.careevolution.com/Master.Adapter1.WebClient/api/fhir-stu3
  - Protected (SMART-on-FHIR) R4 end-point: https://fhir.careevolution.com/Master.Adapter1.WebClient/api/fhir-r4
  - Supports backend services authorization:
    - Token end point: https://fhir.careevolution.com/Master.Adapter1.WebClient/identityserver/connect/token
    - client ID: JWTClientCredentials
    - issuer: JWTClientCredentials
    - audience: https://fhir.careevolution.com/Master.Adapter1.WebClient/identityserver/connect/token
    - scope: system/*.read
    - signing certificate: https://fhir.careevolution.com/jwt.pfx with password jwt.careevolution.com
  - Contact CareEvolution if you need a login to access the protected end-point (...or for any additional information or question)

- http://fhir.nestvision.com (NestVision, a Chinese medical IT company, for Chinese implementers)
  - NEST-FHIR: a free FHIR testing platform. Functionality starts with client conformance testing against a FHIR STU3 server, via any rest extensions on chrome browser, and will become a full functional testing platform later.

- SyntheticMass - https://syntheticmass.mitre.org/fhir
  - FHIR STU3 v1.8
  - JSON only
  - 1 Million Synthetic Patients with complete medical records
  - Open Source https://github.com/synthetichealth

- The SMART Genomics API - http://genomics-advisor.smartplatforms.org:2048/static/doc/index.html
  - FHIR STU3 v1.8
  - JSON and XML
  - The SMART Genomics API is built on top of SMART on FHIR
  - Open Source https://github.com/chaiery/FHIR-Genomics-2
  - Documentation:http://genomics-advisor.smartplatforms.org:2048/static/doc/index.html

- Kaji - https://kaji.healthforge.io
  - Supports STU3
  - General purpose server supporting all resources, most search parameters, xml & json
  - Tech stack: Scala, PostgreSQL, Kubernetes

- JPA HL7 FHIR Server - http://fhir.com.vn:8181
  - Promotes HL7 FHIR development in Vietnam

- Phast's Standard Terminology Services STS - https://jade.phast.fr/resources-server/api/FHIR/
  - FHIR R4
  - JSON and XML
  - Terminology server with terminologies SNOMED CT, LOINC, EDQM Standard Terms, all FHIR-defined code systems ...
  - Support ECL for SNOMED CT
  - Resources: CodeSystem, ValueSet, ConceptMap, NamingSystem, TerminologyCapabilities
  - interactions: read, vread, update, delete, create
  - common parameters: _tag, _profile, _security, _text, _content, _list, _format, _query, _filter
  - common search parameters: _lastUpdated, _has, _type, _id
  - common search result parameters: _sort, _count, _include, _revinclude
  - operations on CodeSystem (GET & POST): lookup, validate-code, subsumes
  - operations on ValueSet (GET & POST): expand, validate-code
  - operations on ConceptMap (GET & POST): translate, closure
- NLM's HAPI FHIR servers (reset weekly):
  - https://lforms-fhir.nlm.nih.gov/baseDstu3
  - https://lforms-fhir.nlm.nih.gov/baseR4
  - 4.7 million Observations from 2048 patients (deidentified); tag: "ri-10k"
  - Also has generated data and sample resources from FHIR