Open Dental Software

FHIR Interface Specification

For Open Dental 17.1

Opendental

May 8, 2017

Table of Contents

About Open Dental FHIR

Open Dental has a RESTful API service that conforms to the FHIR standard defined by https://www.hl7.org/fhir. This FHIR service can be used to look up and create patient and appointments. For a detailed, technical description of Open Dental FHIR's capabilities, see the Conformance Statement in Appendix A.

This document describes the FHIR implementation in Open Dental v. 17.1. The version of FHIR used in this implementation is 1.0.2.

These are the resources currently accessible through the FHIR API: Appointment, Location, Organization, Patient, Practitioner, Schedule, Slot, and Subscription. These resources allow the GET method (meaning that the data can be retrieved through the API). Three resources additionally implement the POST method (resources can be created through the API): Appointment, Patient, and Subscription.

Testing Environment

Open Dental hosts a test server for developers to play with FHIR. The base URL is http://50.201.161.45:56015/opendentalfhir. The Conformance Statement on this server gives a detailed, technical description of Open Dental's FHIR capabilities. These three headers must be included in requests sent to this server:

Content-Type: application/json
Authorization: FHIRAPIKey apikey=Ztx2c32DxX0klEA5
Accept: application/json

A browser extension or other software is necessary to send request headers. Keep in mind that while this test server does not require HTTPS, Open Dental FHIR in a live production environment must use HTTPS.

Setting Up FHIR

One specific computer will need to be designated as the FHIR server. This can be the same computer that holds the Open Dental database, although if high-load performance is a concern, it would be more efficient to run the FHIR service on a different computer that is on the same network as the Open Dental database server.

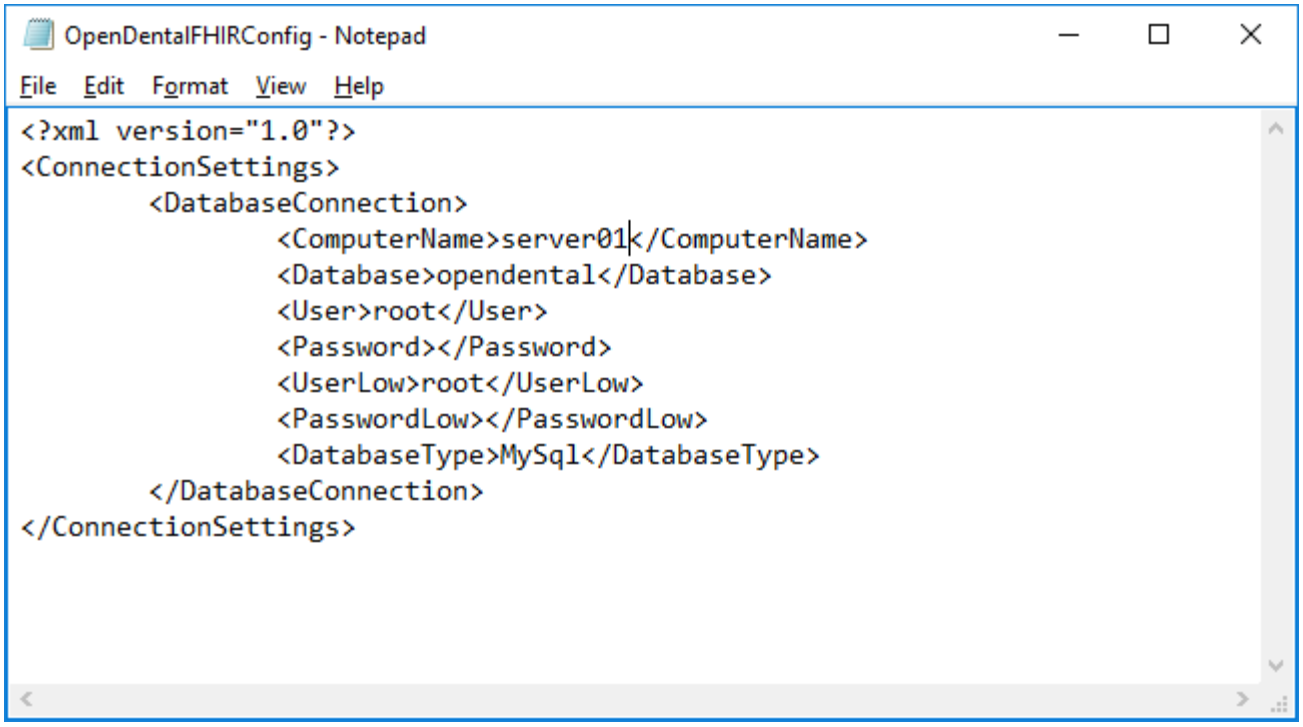
Installation Steps

Performed directly on the FHIR server, possibly through remote desktop. Be sure you are logged in as an administrative user.

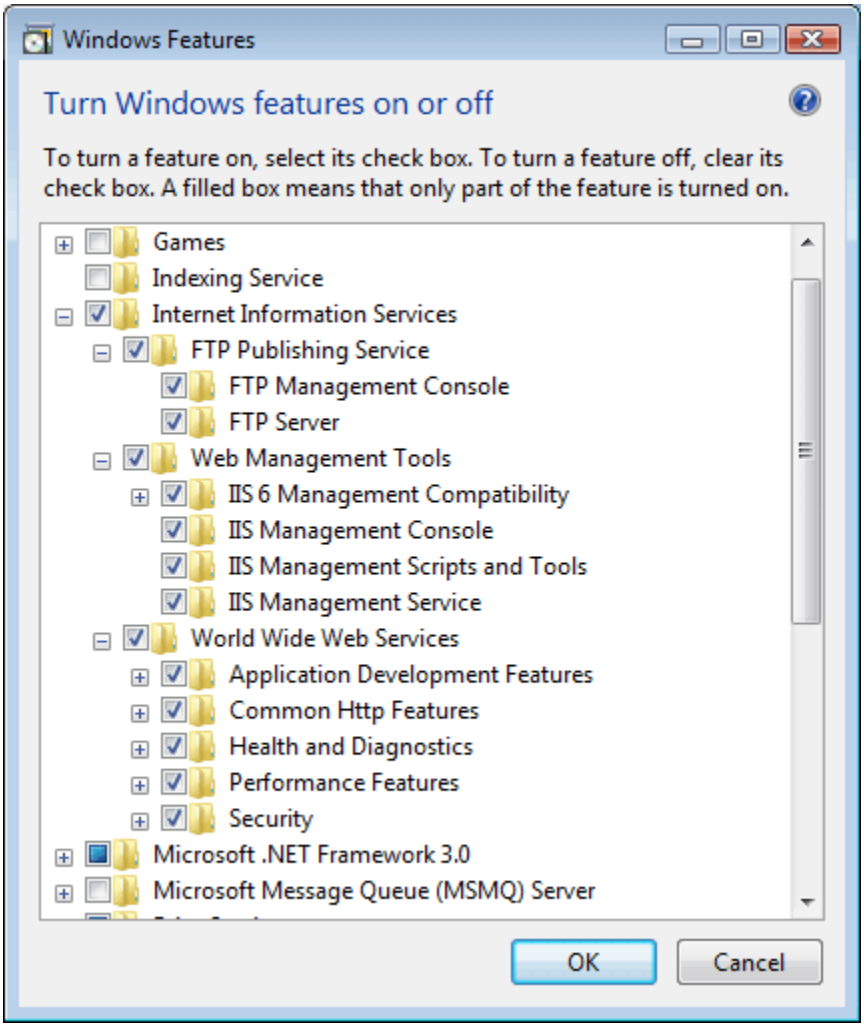
1. Make sure dotNet 4.5 is installed on the FHIR server.
2. Re-install Open Dental by running the Setup.exe from the shared A to Z folders. This will put the necessary files for FHIR in the Open Dental installation directory.
3. If you install the FHIR service on one computer, but have the MySQL service and database on

another computer, make sure that you can connect directly to the database from the FHIR server.

4. On the FHIR server, open Open Dental and log on as any user. This will ensure that the configuration file is created on that computer.
5. Look in your Open Dental application folder for a folder titled OpenDentalFHIR. Edit the file OpenDentalFHIRConfig.xml (right click, "Open with..." Notepad) to set the database connection settings.

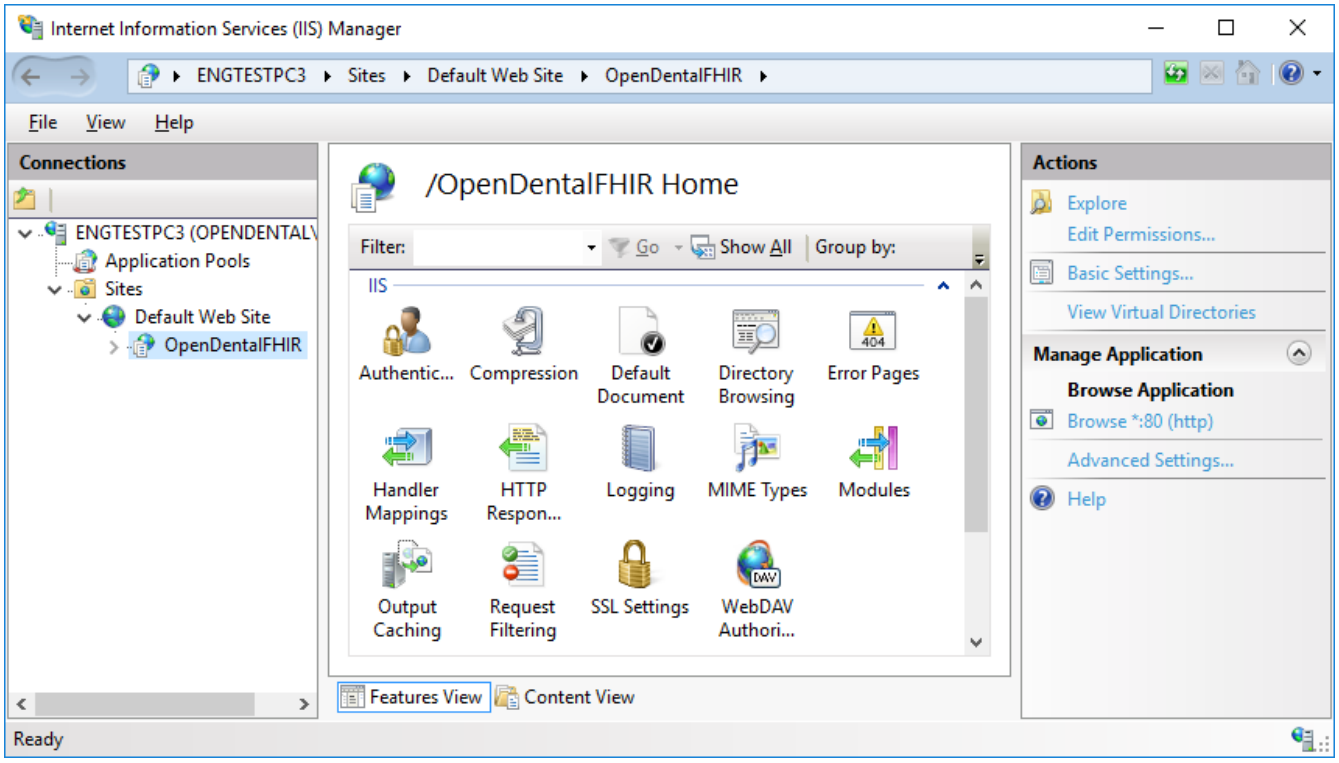


6. (If your FHIR server has Windows Server Web as an operating system, then skip this step.) Turn on more of your IIS features in Windows. In the Control Panel, select Programs and Features (aka Add/Remove Programs). At the upper left, select Turn Windows Features on or off.

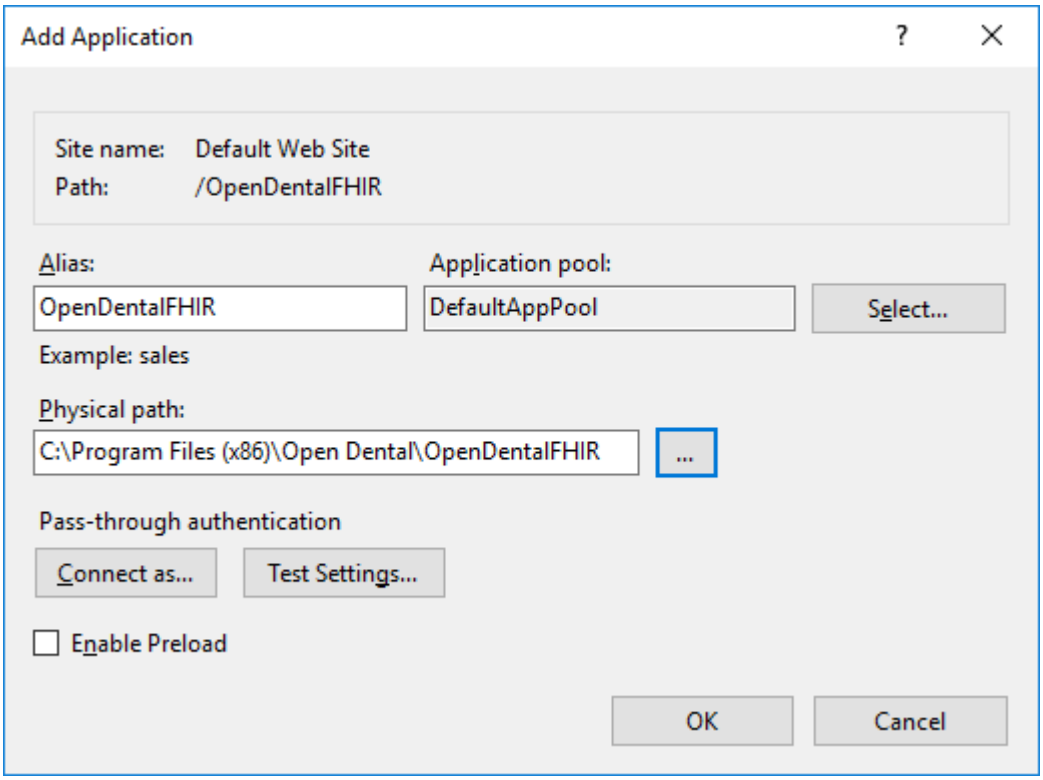


Turn on all the IIS features. In the screenshot above, the IIS folder has been expanded to show those features. If you are a power user, you can selectively only turn on the IIS features that you think will be needed.

7. Open your IIS Manager. Expand your connections and your sites.

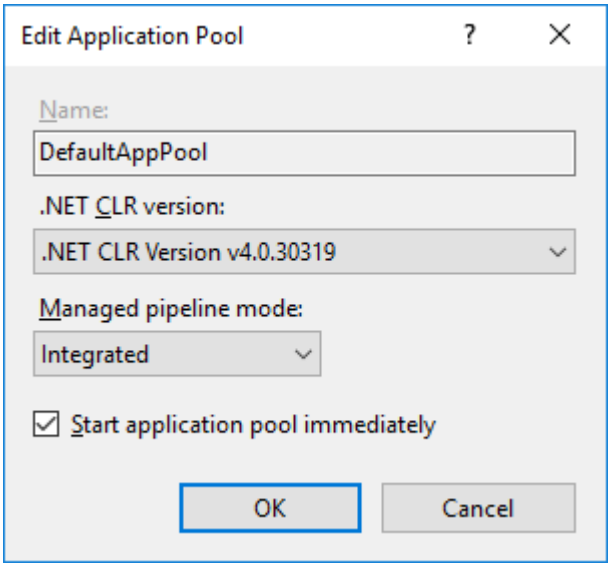


8. Right click on Default Web Site, and Add Application. If the Default Web Site does not yet exist, you must create it first and point it to c:\inetpub\wwwroot.

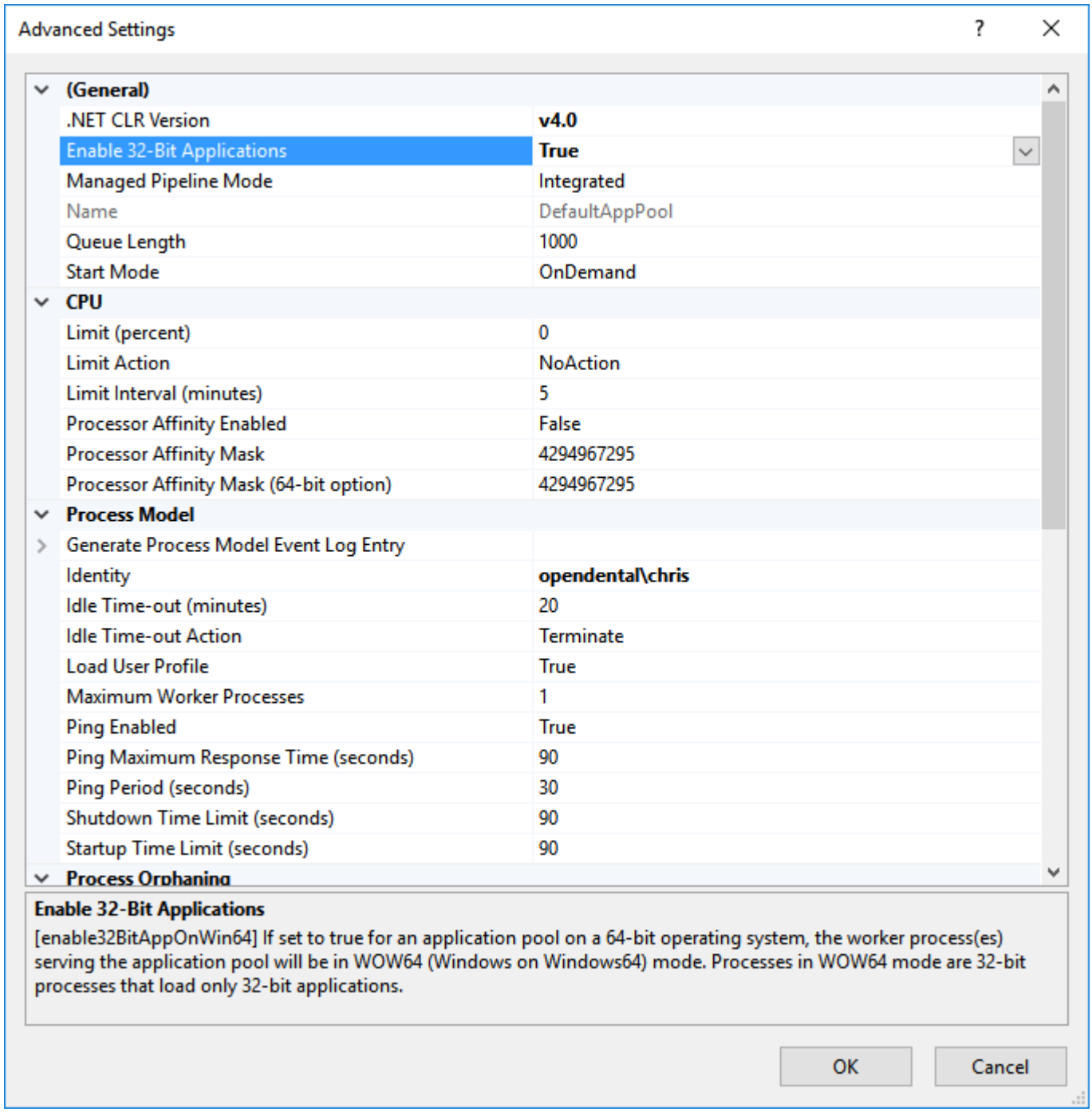


Fill out the information as above, and click OK.

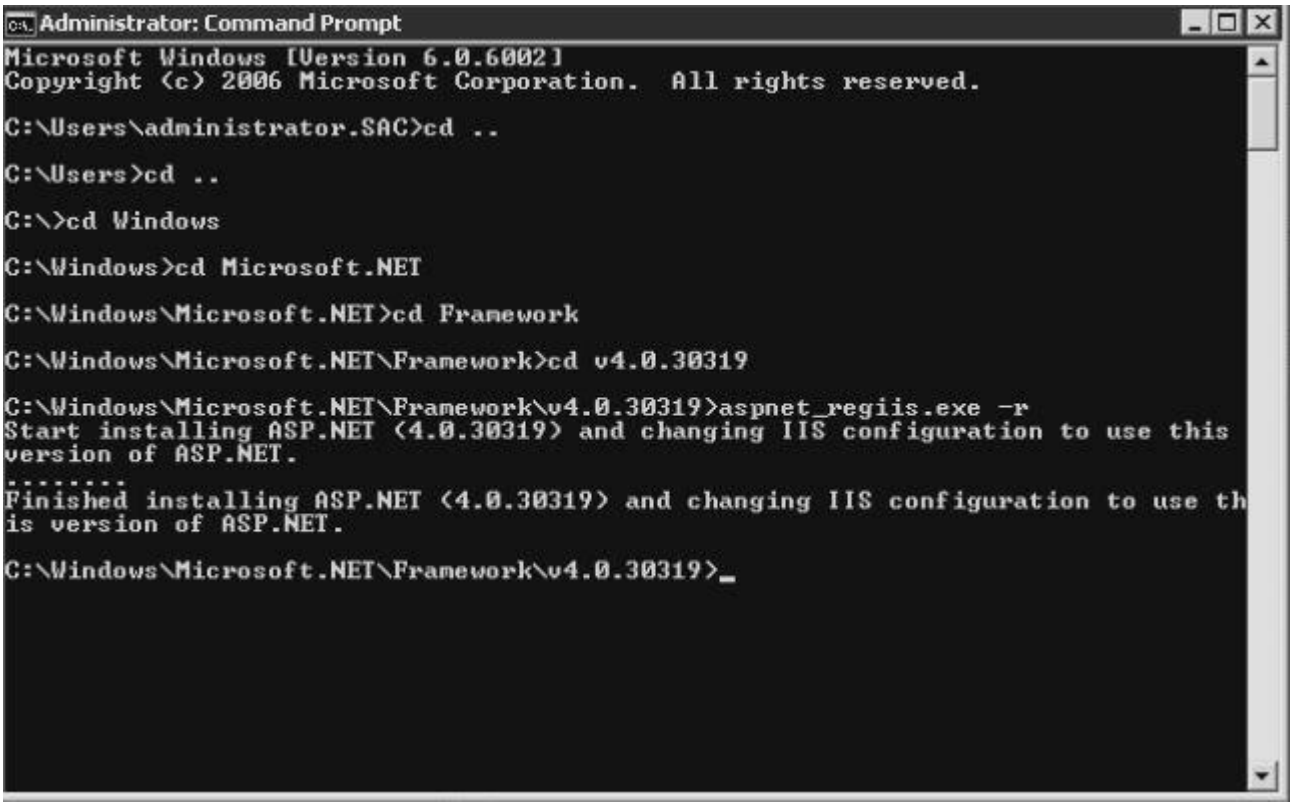
9. Edit the DefaultAppPool and make sure it is set to use .NET Framework v4.0 rather than 2.0.



10. Right-click the DefaultAppPool and select Advanced Settings...
    Set Enable 32-Bit Applications to True.



11. Register .NET Framework v4.0 with IIS as follows. This is not necessary for Windows 10.



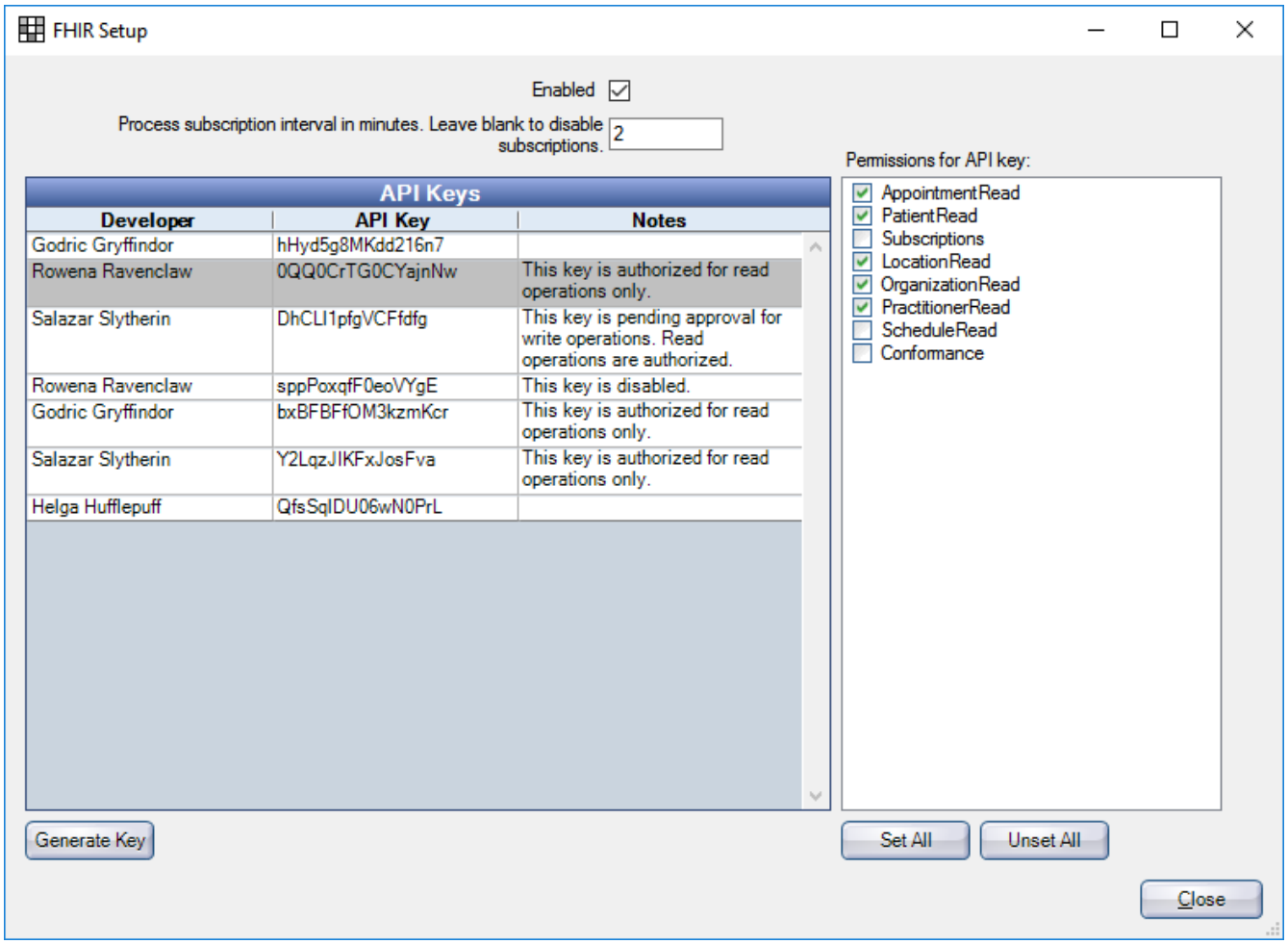
12. Launch the Open Dental program. Enable FHIR by going to Setup -> Advanced Setup -> FHIR and checking the Enabled checkbox.
13. Edit the OpenDentalFHIRConfig.xml within the OpeDentalFHIR folder of the Open Dental installation directory. This file should contain the credentials to connect to the Open Dental database.
14. FHIR requires that all communication uses HTTPS. This requires a security certificate to be installed on the server. The following webpage details that process: http://opendental.com/manual/middletiersecuritycert.html

    If you are planning on only allowing FHIR be used within a private network, you can create your own security certificate instead of purchasing one.
15. Test that the FHIR server is now working by typing into a web browser the URL of your FHIR website followed by '/conformance' (i.e. http://50.201.161.45:56015/opendentalfhir/conformance) If you receive a message stating “The only supported format is application/json”, then the server is working properly.

API Keys

When requesting data from the FHIR server, an API key must be present in the request header. API keys can be acquired from within the Open Dental program and distributed to developers. A single office can generate as many keys as it wants. These keys are free of charge if they only require the ability to read data from the FHIR server. To write data through FHIR, a small fee will be charged for each key. The Subscription resource, however, can use a read-only key to create and update Subscriptions.

To create a new API key from within Open Dental, go to Setup -> Advanced Setup -> FHIR. Click the Generate Key in the lower left. Once the key is created, you can assign permissions for the resources to which you want this API key to have access.



Resources

Important notes concerning the functionality of resources are given here. To see a complete list of resources along with the methods and search parameters supported, see the Conformance Statement.

Appointment

Fields supported:

Field	Comments
identifier	Corresponds to appointment.AptNum
status	
priority	
start	
end	
minutesDuration	
comment	Corresponds to appointment.Note
participant	Used to specify the patient, provider, and operatory

The statuses on a FHIR Appointment resource correspond to the following appointment statuses in Open Dental:

proposed – An appointment that is on the Unscheduled List
pending – An appointment with a status of Scheduled or ASAP
booked – An appointment with a status of Scheduled or ASAP
arrived – An appointment that has a value in the Time Arrived field but not in the Time Dismissed field
fulfilled – An appointment with a status of Complete
cancelled – An appointment that has been deleted
noshow – An appointment with a status of Broken

When creating an appointment, the following fields are required: Patient, location (operatory), and status. If a practitioner is not specified, the provider scheduled in that operatory for that time slot is used. If there is none, the patient's primary provider is used. If a secondary provider is not specified, the hygienist for the operatory is used if the preference to use the secondary provider from the operatory is set. Otherwise, the patient's secondary provider is used.

Location

A location corresponds to an operatory within Open Dental.

Fields supported:

Field	Comments
identifier	Corresponds to operatory.OpNum
status	

name	Correspondes to operatory.OpName
description	Correspondes to operatory.OpName
mode	Always instance
telecom	Contact information for the clinic the operatory belongs to
address	Always room
managingOrganization	The clinic the operatory belongs to

Organization

One Organization represents the practice as entered within Open Dental under Setup -> Practice. Every other Organization resource is a clinic within Open Dental.

Fields supported:

Field	Comments
identifier	0 if this Organization is the practice, otherwise clinic.ClinicNum
name	Correspondes to clinic.Abbr.
telecom	
address	
partOf	If this is a clinic, will point to the practice

Practitioner

Fields supported:

Field	Comments
identifier	Corresponds to provider.ProvNum
active	
name	
gender	Always unknown
practitionerRole	`role` will be either 'provider' or 'hygienist'. `Specialty` is drawn from the provider specialties within Open Dental. These specialties are user-editable and do not conform to any code system.
minutesDuration	
comment	Corresponds to appointment.Note
participant	Used to specify the patient, provider, and operatory

Schedule

Fields supported:

Field	Comments
identifier	An alphanumeric string that identifies the date and provider/location
actor	Either a Location or a Practitioner
planningHorizon	One full day

A Schedule resource will exist for a Practitioner if the provider has a schedule set for that day within Open Dental. Every operatory in Open Dental will have a Schedule resource for every day. If no date range is specified for a GET call, then the schedules for the next 28 days will be returned.

Slot

Fields supported:

Field	Comments
identifier	An alphanumeric string that identifies the date, provider/location, and the start and end.
schedule	
freeBusyType	
start	
end	
overbooked	If the provider is scheduled for multiple appointments at this time, will be true

Slots are divided into five, ten, or fifteen minute intervals (depending on the appointment time increment preference). A Slot that is linked to a Schedule that is linked to a Practitioner will be considered free if there is a schedule within Open Dental for that provider during that time and the provider is not scheduled for an appointment during that time. A Slot that is linked to a Location will be considered free if that operatory has a provider scheduled for that time and the operatory is an operatory considered for Web Sched and there is no appointment in that Slot.

Subscription

Fields supported:

Field	Comments
criteria	
contact	
reason	
status	
error	
channel	
end	

Subscriptions can be used to find out about changes that occur to Patients and Appointments. The channel type that is supported is rest-hook, so when a change occurs, an empty POST request is sent to the channel endpoint. A notification will be sent anytime there is a change in the Patient or Appointment database tables, so it is possible that a notification will be sent even though the resource as returned by FHIR has the exact same fields.

To uses Subscriptions, the Open Dental eConnector service must be running. The interval at which the service sends out notifications can be set in the Open Dental program in Setup -> Advanced Setup -> FHIR.

Use Cases

All the following use cases can be performed on the demo server using the specified URLs.

Use Case 1 – Appointments for Date

Find all appointments scheduled for clinic Hogwarts Hospital Wing for September 1st, 2016.

- The client will find out the id for the Hogwarts Hospital Wing clinic by querying the Organization resource:
  http://50.201.161.45:56015/opendentalfhir/organization?name=Hogwarts%20Hospital%20Wing
- Then the client will need all the Locations that have Hogwarts Hospital Wing for their organization (we'll say the id for Hogwarts Hospital Wing is 1):
  http://50.201.161.45:56015/opendentalfhir/location?organization=Organization/1
- The client can now query the Appointment resources (suppose that the location returned above has the id of 2 and 3):
  http://50.201.161.45:56015/opendentalfhir/appointment?location=2,3&date=2016-09-01&status=booked

Use Case 2 – List of Patients

Keep an updated list of patients who have Madame Pomfrey as a provider

- The client will find out the id for Madame Pomfrey by querying the Practitioner resource:

http://50.201.161.45:56015/opendentalfhir/practitioner?family=pomfrey&given=madame

- This will return a Practitioner resource. In this case the id for that resource will be 1.
- The client will then issue a query against the patient resources like the following: http://50.201.161.45:56015/opendentalfhir/patient?careprovider=Practitioner/1
- To be informed of new patients that are assigned Madame Pomfrey as their provider, the client then creates a Subscription resource like this one:

```
{
  "criteria": "patient?careProvider=Practitioner/1",
  "contact": [
    {
      "system": "email",
      "value": "andrew@friendsofopendental.com",
      "use": "work",
      "rank": 1,
      "period": {
        "start": "2016-08-01T08:00:00"
      }
    }
  ],
  "reason": "To provider surveys to Madame Pomfrey's patients to
determine her quality of care",
  "status": "requested",
  "channel": {
    "type": "rest-hook",
    "endpoint": "http://www.friendsofopendental.com/fhirpatients"
  },
  "end": "2018-01-01T00:00:00"
}
```

Then the client will perform a POST request it to this URL: http://50.201.161.45:56015/opendentalfhir/subscription

- Every few minutes Open Dental's EConnector will check active subscriptions and if any match the criteria, it will send an empty POST request to the endpoint URL (http://www.friendsofopendental.com/fhirpatients) that was included when the Subscription was created.
- When the client receives that POST request, it can issue the same query with a parameter for the time that it last checked: http://50.201.161.45:56015/opendentalfhir/patient?careprovider=Practitioner/1&_lastupdated=ge2016-09-21T18:37:10

Use Case 3 – Find an opening and create an appointment

Find a time where an appointment is not scheduled in the operatory named 'Madame Pomfrey's Operatory' for March 17th, 2017, and create an appointment for a patient named Penelope Clearwater with the provider Madame Pomfrey. Create the patient if she does not exist.

- The client will first find the operatory id for the operatory using this query:

http://50.201.161.45:56015/opendentalfhir/location?name=madame+pomfrey%27s+operatory

- Using the returned id of 1, issue a query for the schedule for that location for that day: http://engtestpc3/opendentalfhir/schedule?actor=Location/1&date=2017-03-17
- The id from that resource will be 20170317L1. That id will be used for the 'schedule' parameter for the Slot resource: http://engtestpc3/opendentalfhir/slot?schedule=20170317L1&fb-type=free
- Using the list of available slots, we can now pick a time for the appointment. We'll pick 8:00 AM for this example.
- Then to find the patient for this appointment, we will issue this query: http://50.201.161.45:56015/opendentalfhir/patient?family=clearwater&given=penelope
- If this returns 0 results, we will need to create the patient by submitting a POST request to http://50.201.161.45:56015/opendentalfhir/patient

```
{
  "name": [
    {
      "use": "usual",
      "family": "Clearwater",
      "given": "Penelope"
    }
  ],
  "telecom": [
    {
      "system": "phone",
      "value": "(123)456-7890",
      "use": "home"
    }
  ],
  "gender": "female",
  "birthDate": "1996-09-19"
}
```

- The value from the Location header of the response will be the id for the patient on the appointment (in this case, 157).
- The next step is to find the id for the provider Madam Pomfrey. This is the query that will be used: http://50.201.161.45:56015/opendentalfhir/practitioner?family=pomfrey&given=madame
- Using the id of 1 returned from the last query, we can construct the appointment resource.

```
{
  "status": "booked",
  "priority": 5,
  "start": "2017-03-17T08:00:00",
  "end": "2017-03-17T08:40:00",
  "minutesDuration": 40,
  "participant": [
    {
      "type": [
```

```
        {
          "code": [
            {
              "system": "http://hl7.org/fhir/participant-type",
              "code": "PART"
            }
          ]
        }
      ],
      "actor": {
        "reference": "Patient/157",
      }
    },
    {
      "type": [
        {
          "code": [
            {
              "system": "http://hl7.org/fhir/participant-type",
              "code": "PPRF"
            }
          ]
        }
      ],
      "actor": {
        "reference": "Practitioner/1",
      },
    },
    {
      "type": [
        {
          "code": [
            {
              "system": "http://hl7.org/fhir/participant-type",
              "code": "PART"
            }
          ]
        }
      ],
      "actor": {
        "reference": "Location/1",
      }
    }
  ]
}
```

- Now, posting to http://50.201.161.45:56015/opendentalfhir/appointment should return an HTTP status code of 201.

Appendix A – Conformance Statement

This Conformance Statement describes in detail the implementation of Open Dental FHIR. An up-to-date Conformance Statement can always be retrieved from http://50.201.161.45:56015/opendentalfhir/conformance


```
{
  "url": "http://50.201.161.45:56015/opendentalfhir/conformance/open-dental-conformance",
  "version": "1.0.1",
  "status": "active",
  "experimental": true,
  "publisher": "Open Dental Software, Inc.",
  "contact": [
    {
      "name": "Brian Miller",
      "telecom": [
        {
          "system": "phone",
          "value": "5033635432",
          "use": "work",
          "rank": 1
        },
        {
          "system": "email",
          "value": "brian@opendental.com",
          "use": "work",
          "rank": 2
        }
      ]
    },
    {
      "name": "Chris McGehee",
      "telecom": [
        {
          "system": "phone",
          "value": "5033635432",
          "use": "work",
          "rank": 2
        },
        {
          "system": "email",
          "value": "chris@opendental.com",
          "use": "work",
          "rank": 1
        }
      ]
    }
```

```
  ],
  "date": "2016-08-22T00:00:00",
  "description": "This conformance statement describes the RESTful
API that accesses the Open Dental database.",
  "requirements": "Published to give operational support to third-
party vendors wishing to interface with the Open Dental database.",
  "kind": "instance",
  "software": {
    "name": "Open Dental",
    "version": "16.3.0.0",
    "releaseDate": "2016-08-11T00:00:00"
  },
  "implementation": {
    "description": "FHIR server for a dental office running Open
Dental",
    "url": "http://engtestpc3:56015/opendentalfhir"
  },
  "fhirVersion": "1.0.2",
  "acceptUnknown": "no",
  "format": [
    "application/json"
  ],
  "rest": [
    {
      "mode": "server",
      "documentation": "This FHIR server can be used to look up
patient demographics and appointments and to find openings where
appointments can be scheduled. A client system can keep itself in
sync with this server through Subscriptions.",
      "security": {
        "cors": false,
        "description": "This FHIR server uses API keys to enforce
security. Each API key has a set of permissions that allow access to
certain resources. These keys can be generated and managed within the
Open Dental program (Setup -> Advanced Setup -> FHIR). These API keys
are only valid for the dental office that generated the API key. Once
a client has an API key, it can include it in the header with the
format 'Authorization: FHIRAPIKey apikey=mysupersecretapikey'."
      },
      "resource": [
        {
          "type": "Appointment",
          "interaction": [
            {
              "code": "read"
            },
            {
              "code": "search-type"
            },
```

```
          {
               "code": "create"
          ],
          "versioning": "no-version",
          "readHistory": false,
          "updateCreate": false,
          "conditionalCreate": false,
          "conditionalUpdate": false,
          "conditionalDelete": "not-supported",
          "searchParam": [
            {
              "name": "patient",
              "type": "reference",
              "documentation": "The patient on the appointment.
Corresponds to appointment.PatNum.",
              "target": [
                "Patient"
              ]
            },
            {
              "name": "location",
              "type": "reference",
              "documentation": "The operatory in which the
appointment is scheduled. Corresponds to appointment.Op.",
              "target": [
                "Location"
              ]
            },
            {
              "name": "identifier",
              "type": "string",
              "documentation": "The logical Id of the appointment.
This is the same value as appointment.AptNum."
            },
            {
              "name": "practitioner",
              "type": "reference",
              "documentation": "The provider or hygienist on the
appointment. Corresponds to appointment.ProvNum or
appointment.ProvHyg.",
              "target": [
                "Practitioner"
              ]
            },
            {
              "name": "status",
              "type": "token",
              "documentation": "The status of the appointment. A
status of 'booked' corresponds to an appointment.AptStatus of
```

```
Scheduled or ASAP. A status of 'cancelled' corresponds to an
appointment that has been deleted. A status of 'noshow' corresponds
to an appointment.AptStatus of Broken'. A status of 'proposed'
corresponds to an appointment.AptStatus of Planned. A status of
'pending' corresponds to a an appointment.AptStatus of UnschedList. A
status of 'fulfilled' corresponds to an appointment.AptStatus of
'complete'."
            },
            {
              "name": "date",
              "type": "date",
              "documentation": "The start time of the appointment.
This corresponds to appointment.AptDateTime."
            },
            {
              "name": "_lastUpdated",
              "type": "date",
              "documentation": "The last time any of the fields in
the appointment were modified (this includes fields that are stored
in the database and not transferred through FHIR). This corresponds
to appointment.DateTStamp."
            }
          ]
        },
        {
          "type": "Location",
          "interaction": [
            {
              "code": "read"
            },
            {
              "code": "search-type"
            }
          ],
          "versioning": "no-version",
          "readHistory": false,
          "updateCreate": false,
          "conditionalCreate": false,
          "conditionalUpdate": false,
          "conditionalDelete": "not-supported",
          "searchParam": [
            {
              "name": "address",
              "type": "string",
              "documentation": "The physical address of the clinic to
which the operatory is assigned. Corresponds to any value in
clinic.Address, clinic.Address2, clinic.City, clinic.State,
clinic.Zip."
            },
```

```
            {
              "name": "address-city",
              "type": "string",
              "documentation": "The city of the clinic to which the
operatory is assigned. Corresponds to clinic.City."
            },
            {
              "name": "address-postalcode",
              "type": "string",
              "documentation": "The zip code of the clinic to which
the operatory is assigned. Corresponds to clinic.Zip."
            },
            {
              "name": "address-state",
              "type": "string",
              "documentation": "The state of the clinic to which the
operatory is assigned. Corresponds to clinic.State."
            },
            {
              "name": "identifier",
              "type": "string",
              "documentation": "The logical Id of the location. This
is the same value as operatory.OperatoryNum."
            },
            {
              "name": "name",
              "type": "string",
              "documentation": "The name or abbreviation of the
operatory. Corresponds to operatory.OpName or operatory.Abbrev."
            },
            {
              "name": "organization",
              "type": "reference",
              "documentation": "The clinic to which the operatory is
assigned. Corresponds to operatory.ClinicNum.",
              "target": [
                "Organization"
              ]
            },
            {
              "name": "status",
              "type": "token",
              "documentation": "The status 'active' corresponds to an
operatory that is not hidden. The status 'inactive' corresponds to an
operatory that is hidden."
            }
          ]
        },
        {
```

```
          "type": "Organization",
          "interaction": [
            {
              "code": "read"
            },
            {
              "code": "search-type"
            }
          ],
          "versioning": "no-version",
          "readHistory": false,
          "updateCreate": false,
          "conditionalCreate": false,
          "conditionalUpdate": false,
          "conditionalDelete": "not-supported",
          "searchParam": [
            {
              "name": "address",
              "type": "string",
              "documentation": "The physical or billing address of
the clinic or practice. Corresponds to any value in clinic.Address,
clinic.Address2, clinic.City, clinic.State, clinic.Zip,
clinic.BillingAddress, clinic.BillingAddress2, clinic.BillingCity,
clinic.BillingState, clinic.BillingZip."
            },
            {
              "name": "address-city",
              "type": "string",
              "documentation": "The city of the clinic or practice.
Corresponds to clinic.City or clinic.BillingCity."
            },
            {
              "name": "address-postalcode",
              "type": "string",
              "documentation": "The zip code of the clinic or
practice. Corresponds to clinic.Zip or clinic.BillingZip."
            },
            {
              "name": "address-use",
              "type": "string",
              "documentation": "All addresses are considered 'work'
so that is the only value that will return results."
            },
            {
              "name": "identifier",
              "type": "string",
              "documentation": "The logical Id of the clinic or
practice. This is the same value as clinic.ClinicNum, however the
dental practice can be represented as a Clinic resource with a
```

```
logical Id of 0."
            },
            {
              "name": "name",
              "type": "string",
              "documentation": "The name of the clinic or practice.
Corresponds to clinic.Abbr."
            },
            {
              "name": "partof",
              "type": "reference",
              "documentation": "In this implementation, all clinics
are a partof the practice.",
              "target": [
                "Organization"
              ],
              "chain": [
                "name"
              ]
            }
          ]
        },
        {
          "type": "Patient",
          "interaction": [
            {
              "code": "read"
            },
            {
              "code": "search-type"
            },
            {
              "code": "create"
            },
          ],
          "versioning": "no-version",
          "readHistory": false,
          "updateCreate": false,
          "conditionalCreate": false,
          "conditionalUpdate": false,
          "conditionalDelete": "not-supported",
          "searchParam": [
            {
              "name": "active",
              "type": "token",
              "documentation": "Whether a patient is active or not. A
value of 'true' corresponds to a patient.PatStatus of Patient. A
value of 'false' corresponds to a patient.PatStatus of anything other
than Patient."
```

```
            },
            {
              "name": "address",
              "type": "string",
              "documentation": "The address of the patient.
Corresponds to any value in patient.Address, patient.Address2,
patient.City, patient.State, patient.Zip, or patient.Country."
            },
            {
              "name": "address-city",
              "type": "string",
              "documentation": "The patient's city. Corresponds to
patient.City."
            },
            {
              "name": "address-postalcode",
              "type": "string",
              "documentation": "The patient's zip code. Corresponds
to patient.Zip."
            },
            {
              "name": "address-state",
              "type": "string",
              "documentation": "The patient's state. Corresponds to
patient.State."
            },
            {
              "name": "address-country",
              "type": "string",
              "documentation": "The patient's country. Corresponds to
patient.Country."
            },
            {
              "name": "birthdatedate",
              "type": "date",
              "documentation": "The patient's birthdate. This
corresponds to patient.Birthdate."
            },
            {
              "name": "careprovider",
              "type": "reference",
              "documentation": "The patient's primary provider.
Corresponds to patient.PriProv.",
              "target": [
                "Practitioner"
              ]
            },
            {
              "name": "deathdate",
```

```
              "type": "date",
              "documentation": "The date time the patient passed
away. This corresponds to patient.DateTimeDeceased."
            },
            {
              "name": "deceased",
              "type": "token",
              "documentation": "Whether a patient deceased or not. A
value of 'true' corresponds to a patient.PatStatus of Deceased or a
value is entered in patient.DateTimeDeceased. A value of 'false'
corresponds to a patient.PatStatus of anything other than Deceased
and no value is entered in papatient.DateTimeDeceased."
            },
            {
              "name": "email",
              "type": "string",
              "documentation": "The patient's email. Corresponds to
patient.Email."
            },
            {
              "name": "family",
              "type": "string",
              "documentation": "The patient's family (last) name.
Corresponds to patient.LName."
            },
            {
              "name": "gender",
              "type": "token",
              "documentation": "The patient's gender. Corresponds to
patient.Gender. The genders 'other' and 'unknown' correponds to
Unknown in Open Dental."
            },
            {
              "name": "given",
              "type": "string",
              "documentation": "The patient's given (first or middle)
name. Corresponds to patient.FName or patient.MiddleI."
            },
            {
              "name": "identifier",
              "type": "token",
              "documentation": "This can match either the business id
of the patient or the patient's Social Security Number. This
corresponds to patient.PatNum or patient.SSN. If intending to search
for the SSN, the token's system should be
'http://hl7.org/fhir/sid/us-ssn'."
            },
            {
              "name": "language",
```

```
              "type": "string",
              "documentation": "The patient's primary language. A
two- or three-letter ISO langauge tag is accepted. Because Open
Dental only stores a patient's langauge and not their language
region, searches for en-US, en-UK, or en-CA will match all patients
whose language is English."
            },
            {
              "name": "name",
              "type": "string",
              "documentation": "The patient's name. Corresponds to a
portion of patient.FName, patient.MiddleI, or patient.LName."
            },
            {
              "name": "organization",
              "type": "reference",
              "documentation": "The clinic to which the patient is
assigned. Corresponds to patient.ClinicNum.",
              "target": [
                "Organization"
              ]
            },
            {
              "name": "phone",
              "type": "string",
              "documentation": "A patient's phone number. Corresponds
to patient.HmPhone, patient.WkPhone, or patient.WirelessPhone."
            },
            {
              "name": "telecom",
              "type": "string",
              "documentation": "A patient's phone number or email.
Corresponds to patient.HmPhone, patient.WkPhone,
patient.WirelessPhone, or patient.Email."
            },
            {
              "name": "_lastUpdated",
              "type": "date",
              "documentation": "The last time any of the fields in
for the patient were modified (this includes fields that are stored
in the database and not transferred through FHIR). This corresponds
to patient.DateTStamp."
            },
            {
              "name": "includePhoto",
              "type": "token",
              "documentation": "Custom search parameter. To reduce
bandwidth, the Patient.photo element is excluded by default. To have
the photo included, this parameter must be included with a value of
```

```
'true'."
            },
            {
              "name": "phoneNumberMatch",
              "type": "string",
              "documentation": "Custom search parameter. This
parameter can be used to search for patients whose phone number is a
partial match. Any non-digit characters are ignored both in the
parameter value and in the database value."
            }
          ]
        },
        {
          "type": "Practitioner",
          "interaction": [
            {
              "code": "read"
            },
            {
              "code": "search-type"
            }
          ],
          "versioning": "no-version",
          "readHistory": false,
          "updateCreate": false,
          "conditionalCreate": false,
          "conditionalUpdate": false,
          "conditionalDelete": "not-supported",
          "searchParam": [
            {
              "name": "family",
              "type": "string",
              "documentation": "The practitioner's family (last)
name. Corresponds to provider.LName."
            },
            {
              "name": "given",
              "type": "string",
              "documentation": "The practitioner's given (first or
middle) name. Corresponds to provider.FName or provider.MI."
            },
            {
              "name": "identifier",
              "type": "string",
              "documentation": "The logical id of the practitioner.
This is the same value as provider.ProvNum."
            },
            {
              "name": "name",
```

```
              "type": "string",
              "documentation": "The practitioner's name. Corresponds
to a portion of provider.FName, provider.MI, or patient.provider."
            },
            {
              "name": "role",
              "type": "token",
              "documentation": "The provider's role in the practice.
The parameter values supported are 'provider' and hygienist'. The
former corresponds to provider.IsSecondary=0 and the latter
corresponds to provider.IsSecondary=1."
            },
            {
              "name": "specialty",
              "type": "string",
              "documentation": "The provider's specialty. These
parameter values are the values set up within Open Dental as provider
specialties."
            }
          ]
        },
        {
          "type": "Schedule",
          "interaction": [
            {
              "code": "read"
            },
            {
              "code": "search-type"
            }
          ],
          "versioning": "no-version",
          "readHistory": false,
          "updateCreate": false,
          "conditionalCreate": false,
          "conditionalUpdate": false,
          "conditionalDelete": "not-supported",
          "searchParam": [
            {
              "name": "actor",
              "type": "reference",
              "documentation": "The actor on the Schedule.",
              "target": [
                "Location",
                "Practitioner"
              ]
            },
            {
              "name": "date",
```

```
              "type": "date",
              "documentation": "The date of the schedule. Unless an
identifier parameter is included, the dates being searched must be a
finite date range."
            },
            {
              "name": "identifier",
              "type": "string",
              "documentation": "The logical id of the practitioner."
            }
          ]
        },
        {
          "type": "Slot",
          "interaction": [
            {
              "code": "read"
            },
            {
              "code": "search-type"
            }
          ],
          "versioning": "no-version",
          "readHistory": false,
          "updateCreate": false,
          "conditionalCreate": false,
          "conditionalUpdate": false,
          "conditionalDelete": "not-supported",
          "searchParam": [
            {
              "name": "fb-type",
              "type": "token",
              "documentation": "The free/busy status of the
appointment. The code system must be
http://hl7.org/fhir/ValueSet/slotstatus"
            },
            {
              "name": "identifier",
              "type": "string",
              "documentation": "A Slot Identifier. The logical id of
the Slot. A parameter of either 'identifier' or 'schedule' must be
included."
            },
            {
              "name": "schedule",
              "type": "reference",
              "documentation": "The Schedule Resource that we are
seeking a slot within. A parameter of either 'identifier' or
'schedule' must be included.",
```

```
              "target": [
                "Schedule"
              ]
            },
            {
              "name": "start",
              "type": "date",
              "documentation": "The start time of the slot."
            }
          ]
        },
        {
          "type": "Subscription",
          "interaction": [
            {
              "code": "read"
            },
            {
              "code": "create"
            },
            {
              "code": "update"
            },
            {
              "code": "delete"
            },
            {
              "code": "search-type"
            }
          ],
          "versioning": "no-version",
          "readHistory": false,
          "updateCreate": false,
          "conditionalCreate": false,
          "conditionalUpdate": false,
          "conditionalDelete": "not-supported",
          "searchParam": [
            {
              "name": "contact",
              "type": "string",
              "documentation": "The value for a contact field for the
Subscription."
            },
            {
              "name": "criteria",
              "type": "string",
              "documentation": "Rule for server push criteria."
            },
            {
```

```
              "name": "payload",
              "type": "string",
              "documentation": "Mimetype to send, or blank for no
payload."
            },
            {
              "name": "status",
              "type": "token",
              "documentation": "The status of the Subscription.
System must be http://hl7.org/fhir/ValueSet/subscription-status"
            },
            {
              "name": "type",
              "type": "token",
              "documentation": "The type of the Subscription channel.
System must be http://hl7.org/fhir/ValueSet/subscription-channel-
type"
            },
            {
              "name": "url",
              "type": "uri",
              "documentation": "The endpoint of the Subscription
channel."
            },
            {
              "name": "identifier",
              "type": "string",
              "documentation": "The logical id of the practitioner."
            }
          ]
        }
      ],
      "interaction": [
        {
          "code": "search-system"
        }
      ],
      "searchParam": [
        {
          "name": "_id",
          "type": "string",
          "documentation": "The logical id of the resource."
        }
      ]
    }
  ],
  "id": "open-dental-conformance"
}
```