ASU College of Health Solutions
ARIZONA STATE UNIVERSITY

Department of Biomedical Informatics

BMI 593: Prototype of Medical Device Integration Knowledge Base and Expert System Queries

ASSIGNMENT Final Project

Prototype of Medical Device Integration Knowledge Base and Expert System Queries

Submitted By:
Bridget A. Moorman, CCE, Col, USAF (Ret)
10 July 2017

Abstract

A medical device integration ontology and knowledge base prototype was built using Protégé. The medical device ontology focused on those medical devices' characteristic that were important to determine if a specific medical device had networking ability, any networking transformation capability for that medical device and the sensor and setting data available from the medical device to be presented to an interface broker and/or electronic health record (EHR) application. Due to time constraints, only nine specific medical device models were included in the prototype. Due to the lack of expert system front-ends, the reasoning extension to Protégé was used to simulate an expert system query. The prototype demonstrated that the ontology and knowledge base were robust enough to meet the project requirement intent. Future considerations for a more robust prototype would entail inclusion of more specific medical device models, use of standard nomenclature terminology for class names and inclusion of location and unit ontologies.

Introduction

Integration/interfacing of Medical Devices to an Electronic Health Record (EHR)

Moorman, B., (2008) provided an overview of integrating medical device systems and information into an EHR. Figure 1 provides a graphical representation of the overview. The process to implement medical device integration into an EHR consists of identifying a medical device, its networking capability, transforming that networking capability if needed, aggregating the data output from the medical device, and then mapping the data to the EHR or other consumer of the data from the sensors or setting of the medical device. It can be very difficult and time-consuming to implement a medical device integration system as there are many different vendors of medical devices with different sensors and networking capabilities.

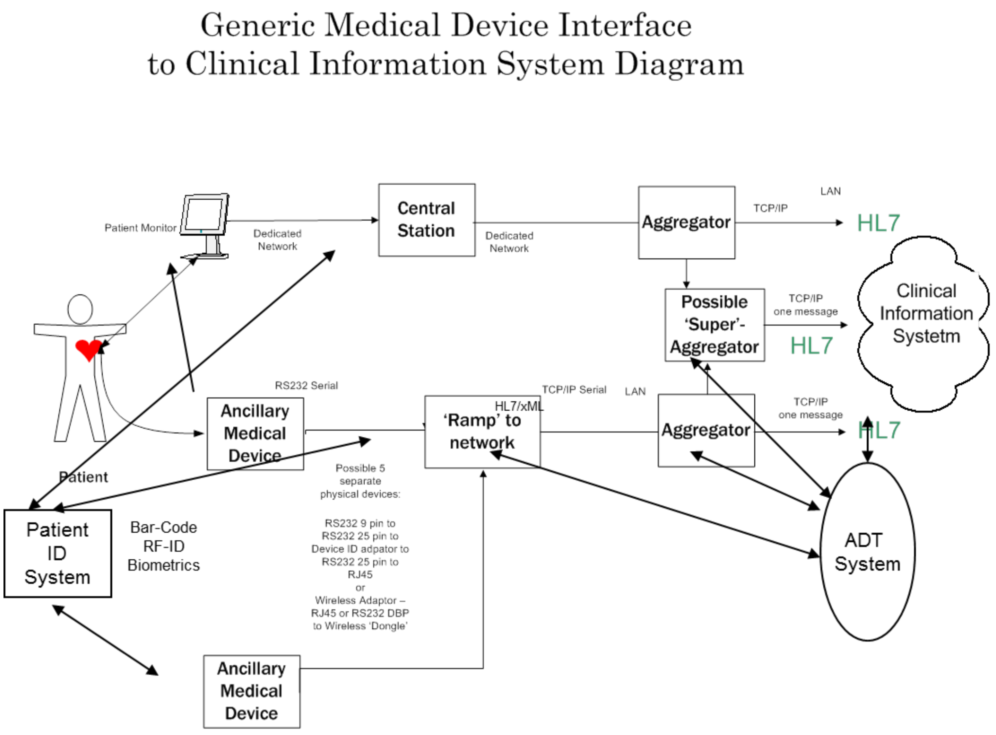

Generic Medical Device Interface
to Clinical Information System Diagram
Patient Monitor
Dedicated Network
Central Station
Dedicated Network
Aggregator
TCP/IP
LAN
HL7
Possible 'Super'- Aggregator
TCP/IP one message
HL7
Clinical Information Systetm
RS232 Serial
Ancillary Medical Device
'Ramp' to network
HL7/xML
TCP/IP Serial
LAN
Aggregator
TCP/IP one message
HL7
Patient
Patient ID System
Bar-Code
RF-ID
Biometrics
Possible 5 separate physical devices:
RS232 9 pin to RS232 25 pin to Device ID adpator to RS232 25 pin to RJ45 or Wireless Adaptor – RJ45 or RS232 DBP to Wireless 'Dongle'
ADT System
Ancillary Medical Device

Figure 1. Moorman Depiction of Medical Device Interfacing/Integration (Moorman, B., 2008)

Many times a medical device does not have the ability to send data or will require a networking transformation to integrate into the system and the EHR. In addition, depending on the medical device integration vendor, there may not be a ‘driver’ which converts the medical device data to a format that can be accepted by the EHR. This preparatory phase of identification of the medical device, networking and data state can be a very time consuming and a mostly manual process. Having a quick way to identify the networking and sensor constraints described above would make the job of implementing the medical device integration system easier and less labor intensive. To speed this process as well as manage the implementation, an expert system with a knowledge base that would automate the identification of the medical device capabilities would help.

Literature Review

A brief review of the work done in this medical device integration area since the Moorman, B., (2008) article has shown many ideas for development of “middleware” management techniques (table of results). Interestingly, nearly all of the techniques require medical devices to have networking capability as well as sentience to report what they are and what they deliver and/or a device directory which has a knowledge base of the available devices and what they deliver (device specifications, meta-data, sensor and setting data). At the same time as these middleware techniques have been published, the three device integration vendors mentioned in the Moorman, B. (2008) article have been ‘absorbed’ by larger companies (Moengain, B. (2015), Densford, F. (2015), Falcon Capital Partners, LLC. (2013)). Moreover, many more of the EHR vendors and medical device vendors now offer medical device aggregation and integration solutions.

Table 1

Medical Device Integration Literature Review Results

Authors	**Title**	**System Description**	**Comments**
Kliem, A., Boelke, A., Grohnert, A., & Traeder, N. (2016)	A reconfigurable middleware for on-demand integration of medical devices. (Kliem et al, 2016)	A middleware platform for medical device integration that “handles standard and proprietary devices, can adapt to the rapidly changing sensor and device environment, and….. preserves interoperability at the application level, if devices are replaced.” They describe a Device Directory as one of the components of their system which they envision as a directory service for medical devices comprising of specific knowledge about the medical devices to be accessed by the aggregator if an unknown medical device is	The main assumption here is that there is a device directory available and the authors envision a Global Device Directory which would contain several files about each type of device and its characteristics to aid in integration. The information needed from the DD would be meta-data about the device such as vendor, class, unique identification, networking, and security. The system described is assumed to be dynamic in nature

Table 1

Medical Device Integration Literature Review Results

Authors	**Title**	**System Description**	**Comments**
		"discovered" and needs to be integrated. Once integrated or connected to the middleware, device setting and sensor data would also be available for integration to a system using the middleware for an interface.	and not manual as most currently available medical device integration systems.
Cabri, G., De Mola, F., & Leonardi, L. (2007)	Agent-based plug-and-play integration of role-enabled medical devices. (Cabri et al, 2007)	The authors describe an agent-based solution for medical device integration which is comprised of three agents that interface, process and manage the medical device data to requestors of the data.	They assume that the medical device has networking capability and they state that device vendors should modify their device software to accommodate the different roles they assign the agents so that this middleware, which is also dynamic, could be enabled for medical device

Table 1

Medical Device Integration Literature Review Results

Authors	**Title**	**System Description**	**Comments**
			integration.
King, A., Procter, S., Andresen, D., Hatcliff, J., Warren, S., Spees, W., . . . Weininger, S. (2009).	An open test bed for medical device integration and coordination. (King et al, 2009)	The authors developed a java machine virtual integration engine which provided device connection and medical device data integration and closed-loop control.	There was an assumption that the device had networking capability and they stated that devices should have a Java Virtual Machine (JVM) capability or 'dongle' which could manage the Java Messaging Service which managed the publish and subscribe mechanism underlying their integration system. They also were going to expand their work using publicly available data streams from medical devices, so they also do not have a

Table 1

Medical Device Integration Literature Review Results

Authors	**Title**	**System Description**	**Comments**
			knowledge base of medical devices and their characteristics available to use their integration engine.
Gregorczyk, D., Bubhaus, T., & Fischer, S. (2012)	A proof of concept for medical device integration using web services. (Gregorczyk, D, Bubhaus, T, Fischer, S., 2012)	The authors in this paper did a proof of concept using the Device Profile for Web Services (DPWS) in a specific clinical area and use case for medical device integration.	For their demonstration, one of the devices in actual use (the surgical microscope) had to be replaced with another in the same device class as the original one did not have networking capability. In addition, they required that the devices output information that was in the IEEE 11073 standard, which is not a

Table 1

Medical Device Integration Literature Review Results

Authors	**Title**	**System Description**	**Comments**
			widely implemented capability. Yet again, there was no knowledge base available or built by them for device integration purposes. They concluded that while theoretically possible to use DPWS for integration, several critical issues remain: lack of patient context, no ability to use non-networkable devices, lack of security protocols, and lack of semantic interoperability capability (unless the device can output in IEEE 11073 format).

Table 1

Medical Device Integration Literature Review Results

Authors	**Title**	**System Description**	**Comments**
Kliem, A., Hovestadt, M., & Kao, O. (2012).	Security and communication architecture for networked medical devices in mobility-aware eHealth environments. (Kliem, A., Hovestadt, M., & Kao, O., 2012)	The authors in this paper describe a middleware approach to manage medical device security and access to an aggregator for integration. They specify a backend which has a global device directory and master and provider authorities which rely upon local device directories. The aggregator connects to the backend and the device.	This paper describes an attempt to solve the issue in the previous paper with regard to security. However, it is theoretical and does rely upon the device directories or knowledge bases to function correctly for the device discovery function.
Hao, A., & Wang, L. (2015).	Medical device integration model based on the internet of things. (Hao, A., & Wang, L., 2015)	The authors in this paper also describe a middleware approach that is dynamic and uses a four layer model to define their approach: devices abstract layer, device adaptation layer, data access	They require the device to have networking capability and imply there is a knowledge base about the devices and the data they can output.

Table 1

Medical Device Integration Literature Review Results

Authors	**Title**	**System Description**	**Comments**
		layer and data abstract layer. The devices abstract layer is further subdivided into standardized output devices, non-standardized output devices, and imaging devices. The device adaptation layer is further subdivided into collection of inspection items (sensors), device type (class), and communication mode. The device access layer deals with the hardware connection for networking and assumes that all of the devices have networking capability. Lastly, the data abstract layer (also the data filtering layer) is further subdivided into inspection items	

Table 1

Medical Device Integration Literature Review Results

Authors	**Title**	**System Description**	**Comments**
		(sensor items), inspection end value (sensor value), reference value, and inspection conclusions.	
Foundation for Intelligent Physical Agents (FIPA), FIPA Device Ontology Specification. (2002)	Foundation for Intelligent Physical Agents (FIPA), FIPA Device Ontology Specification. http://www.fipa.org/specs/fipa00091/SI00091E.html) (FIPA, 2002)	They reference the FIPA device ontology specification at the component level.	Upon review of the ontology, it describes a device by its components at a lower level that would not necessarily be needed for device integration. Additionally, if does not include the types of sensors used for physiological monitoring. Moreover, it does not describe the types of setting or outputs normally seen with medical devices.

Current medical device knowledge bases are held by the medical device vendors and/or third party aggregator vendors (example: product listing of device drivers) and are considered proprietary. There are other related medical device nomenclature and standards efforts: FDA UID and GMDN database (US Food and Drug Administration, 2017); IEEE 11073 Rosetta Stone and data standards (Moorman, B. 2010); and, IHE PCD (IHE 2017) and PCHA guidelines (Personal Connected Health Alliance,2017). However, these do not contain knowledge bases of device information available for querying which marry meta (networking), sensor and setting data.

Project Description and Constraints

Project Description

For this project, I wanted to build and demonstrate the concept of a knowledge base and expert system that would assist a healthcare organization in their medical device integration effort. The idea was to input into an expert system (or reasoner) with an associated knowledge base a medical device inventory with the result of the expert system providing a list of the devices that are already integrate-able, need network transformation, and are not integrate-able. As a secondary output, I wanted to list the available sensor and setting outputs available for the interface broker to the EHR. As detailed in Table 1, some other possible approaches for this knowledge are that the devices could report their status and information via publish/subscribe functionality/capability with interface managing that, however, would still need a device directory or device knowledge base.

For a previous class, BMI 598, I did a search and review of any currently available knowledge bases and ontologies that could assist me in this project (Moorman, B., 2017) The conclusion of that research stated that:

“A medical device integration system implementation spans different nomenclature and ontological domains, some of which compete across the nomenclature and ontological areas. There are also gaps in the areas of cross-domain attributes as well as specificity in some item hierarchies. “ (Moorman, B., 2017, pg 10)

This conclusion meant that for this project I had to build the device ontology from ‘scratch’ and in the end was not able to incorporate any of the ontologies or nomenclatures that were available.

As part of this project, I also reviewed available knowledge base and expert system applications that could be used for my knowledge base. I wanted to use an ontologically based system that was open source, free, currently available, currently supported and able to output the knowledge base and any results in a standard format. Table 2 details the results of that search. In the end, I chose Protégé Desktop version 5.2.0-win to build the ontology and knowledge base and used its reasoner extension to mimic an expert system interface. This did change the project scope such that feeding the expert a medical device inventory list was not possible; instead, queries were constructed against the knowledge base so that specific devices having the query characteristics were retrieved.

Constraints

There were several constraints in this project: limited time for project (7.5 weeks) to a build proof of concept system, limited availability of knowledge base and expert system products, limited to no ontologies to inherit or incorporate into the medical device ontology, and limited access to publicly available information on medical devices and their characteristics. For example, the knowledge base lacks specific units to be associated with the sensor readings. Unfortunately, an open source effort to build units (QUDT, 2017) for clinical and device use had not developed the ontology for the units used in medical devices yet.

As another example, access to medical device specifications and manuals is limited, so only those that were openly available via the internet could be used for building the ontology and knowledge base and these tended to be of older 'legacy' devices. Lastly, due to the time constraints, I was only able to model several medical device types and instances in the knowledge base. There are thousands of different medical devices available on the market and used by healthcare organizations, so 'building out' the knowledge base would take much more time than was available for this project and would need to be an ongoing effort which incorporated medical devices as they were released onto the market.

Table 1
Knowledge Base and Expert System Review Results

Application	Comments
Expertise to go-http://expertise2go.com/e2g3g/	Rule based system, Last updated in 2014
Sweet Rules- http://sweetrules.projects.semwebcentral.org	Rules based system. Last updated in 2005
JLog- http://jlogic.sourceforge.net	No date; sample screens are 1990's MacIntosh
CLIPS http://www.clipsrules.net	Is a commercial product for sale; update is 2016-2017
Open Rules- http://openrules.com/index.htm	Rules based system. Last updated in 2014
Jess- http://herzberg.ca.sandia.gov/jess/index.shtml	Last update 2008
Protégé 5.2.0-win - http://protege.stanford.edu	Ontology based; open source KB (free). Last updated in 2016. Output is resource description framework (rdf) or eXtensible Markup Language (xml).

Methods and Results

Methods

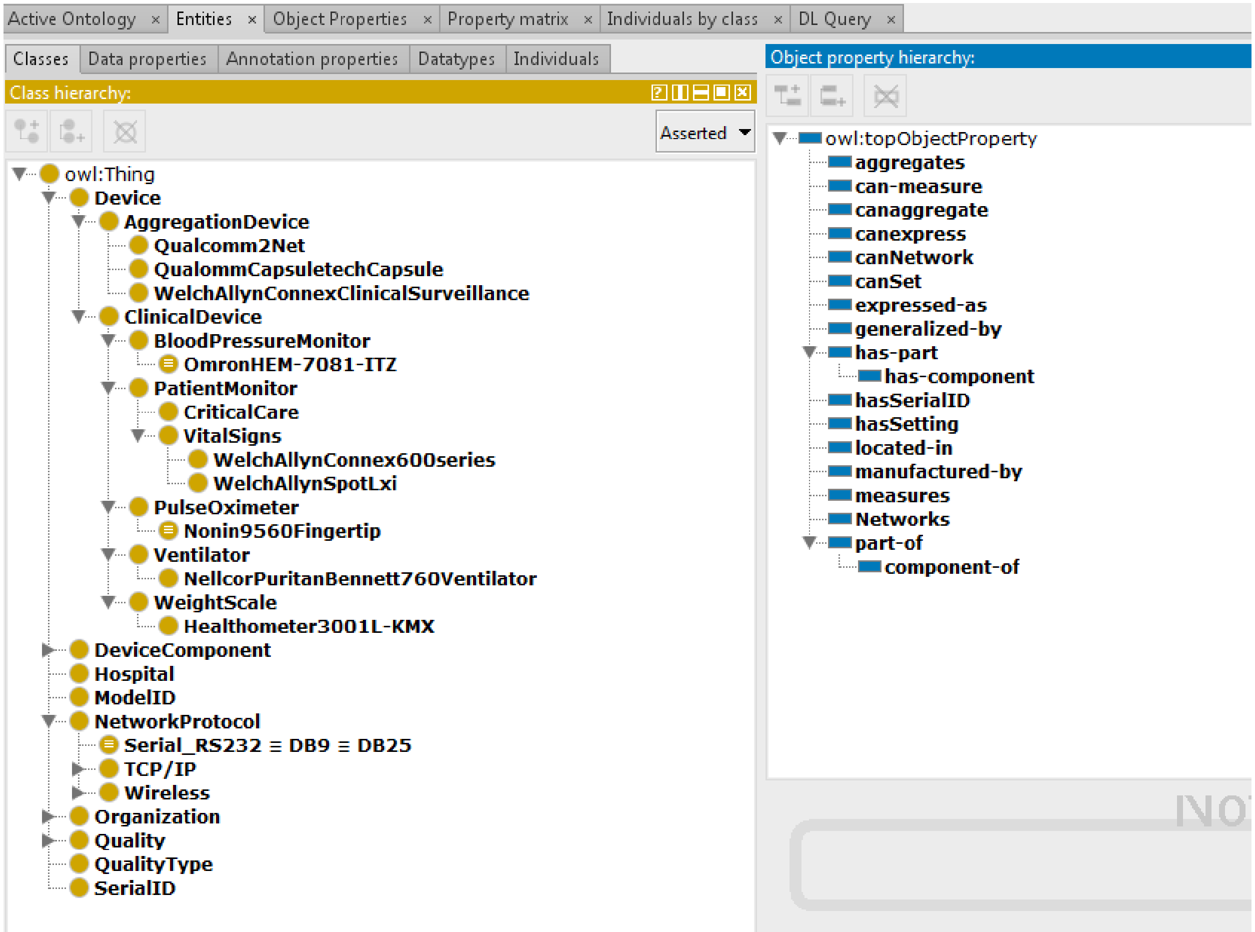
The first task was to learn how to use Protégé. For that, I used a tutorial provided by the University of Manchester (Horridge, M., 2011) which illustrates the features of Protégé as well as the type of thinking required to use the tool while building a “Pizza Ontology and Knowledge Base.” From there I built the general medical device ontology by class and characteristic, incorporated specific medical device instances and then built queries as proof that the knowledge base was constructed appropriately for the task of identifying device characteristics key for integration.

Results

Ontology and Knowledge Base.

Several screenshots of the final ontology, knowledge base and query results are shown below. Figure 2 shows the basic ontology to include the classes (left side of figure) and object properties (right side of figure) . In the Protégé approach to ontologies and knowledge bases, the classes are shown as a hierarchy. As shown in Figure 2, the classes of device, device component, Hospital, ModelID, Network Protocol, Organization, Quality, etc are on an equal level of the hierarchy. Moreover, the device class is further subdivided into the device types: aggregation and clinical. Lastly, the clinical device class is further divided into device types and then specific models of those device types. Figures 3, 4, 5 and 6 show the rest of the classes and subclasses hierarchies in more detail.

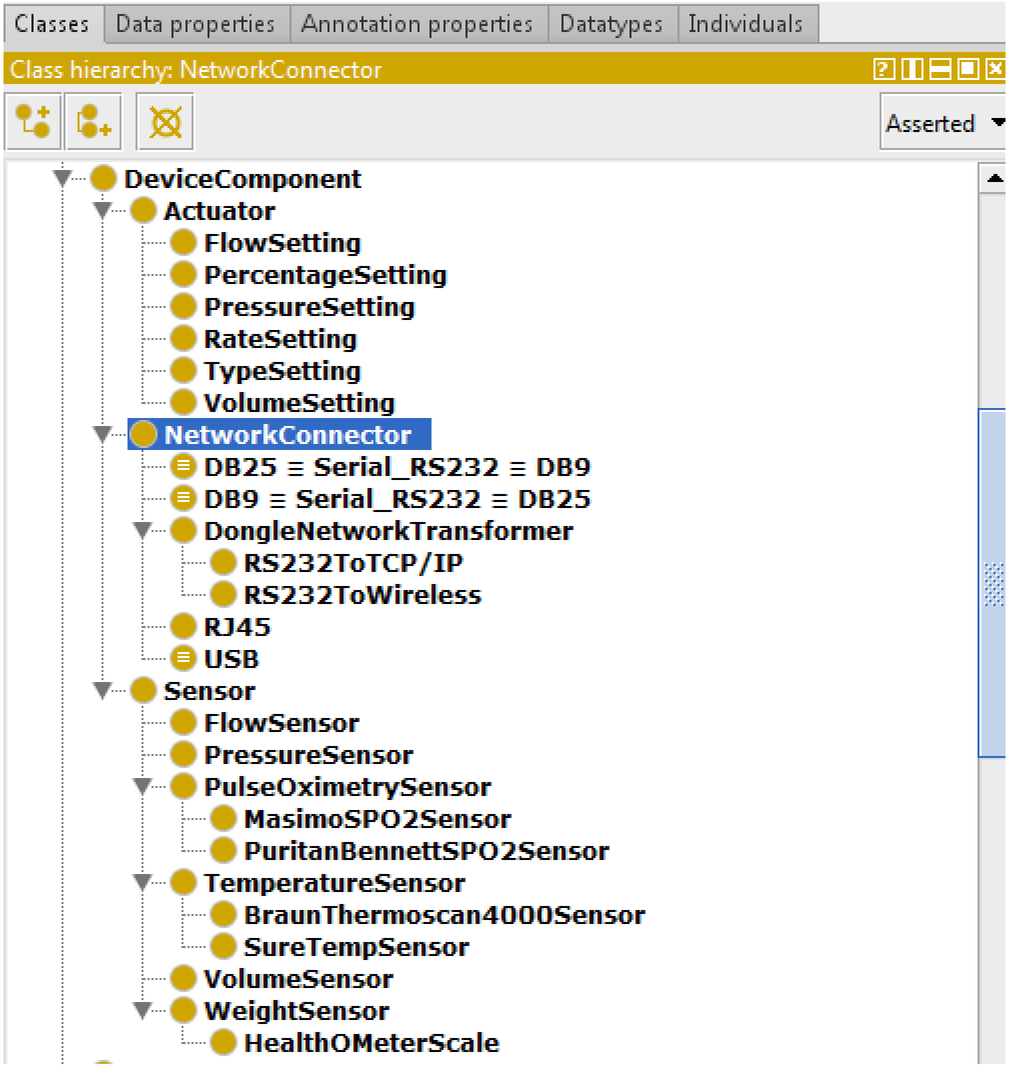
The object properties can be thought of actions or capabilities that can be assigned to the different classes and/or establish relationships between the classes. In the case of this ontology, object properties of aggregates, networks, and measures are key object properties that were used to establish the key class characteristics and relationships required for medical device integration and query development.



```
owl:Thing
  Device
    AggregationDevice
      Qualcomm2Net
      QualommCapsuletechCapsule
      WelchAllynConnexClinicalSurveillance
    ClinicalDevice
      BloodPressureMonitor
        OmronHEM-7081-ITZ
      PatientMonitor
        CriticalCare
        VitalSigns
          WelchAllynConnex600series
          WelchAllynSpotLxi
      PulseOximeter
        Nonin9560Fingertip
      Ventilator
        NellcorPuritanBennett760Ventilator
      WeightScale
        Healthometer3001L-KMX
  DeviceComponent
  Hospital
  ModelID
  NetworkProtocol
    Serial_RS232 ≡ DB9 ≡ DB25
    TCP/IP
    Wireless
  Organization
  Quality
  QualityType
  SerialID
```

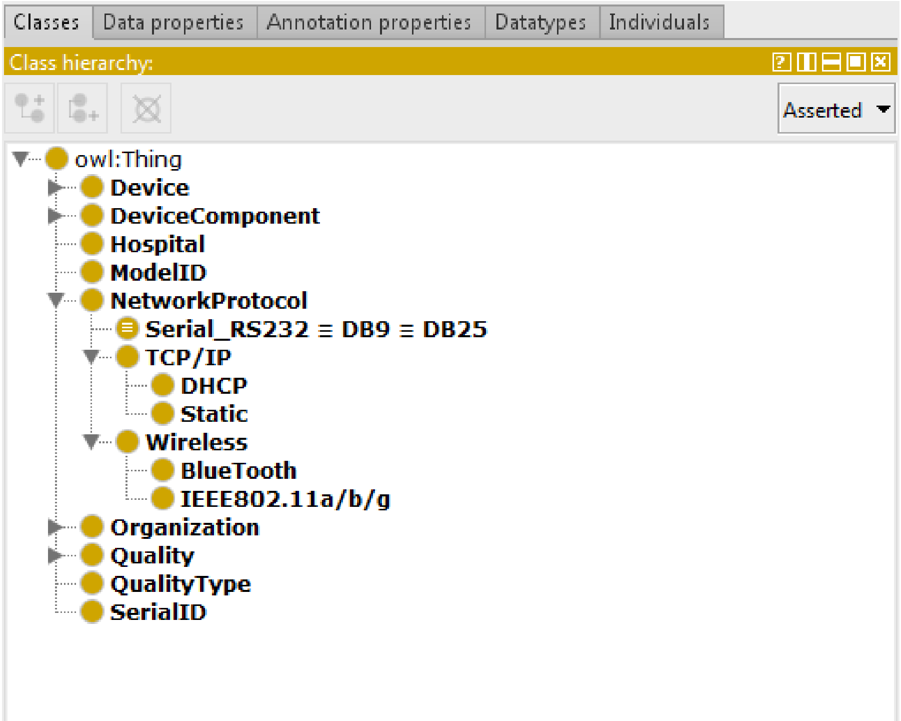
```
owl:topObjectProperty
  aggregates
  can-measure
  canaggregate
  canexpress
  canNetwork
  canSet
  expressed-as
  generalized-by
  has-part
    has-component
  hasSerialID
  hasSetting
  located-in
  manufactured-by
  measures
  Networks
  part-of
    component-of
```

Figure 2. Medical Device Class Hierarchy and Object Properties



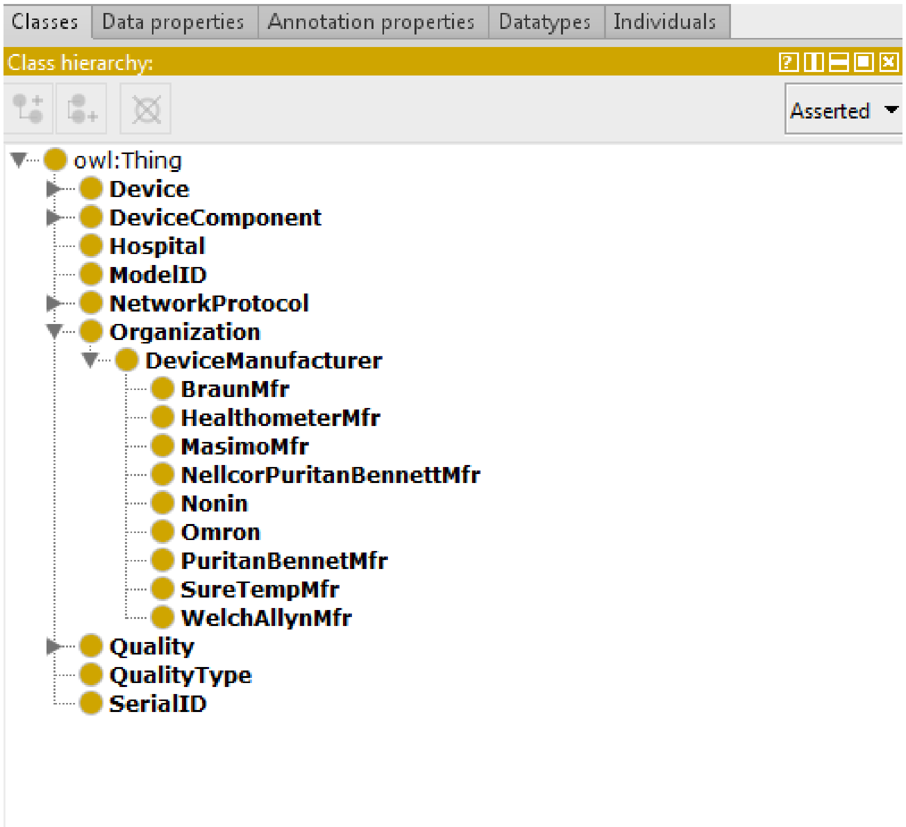
```
DeviceComponent
    Actuator
        FlowSetting
        PercentageSetting
        PressureSetting
        RateSetting
        TypeSetting
        VolumeSetting
    NetworkConnector
        DB25 ≡ Serial_RS232 ≡ DB9
        DB9 ≡ Serial_RS232 ≡ DB25
        DongleNetworkTransformer
            RS232ToTCP/IP
            RS232ToWireless
        RJ45
        USB
    Sensor
        FlowSensor
        PressureSensor
        PulseOximetrySensor
            MasimoSPO2Sensor
            PuritanBennettSPO2Sensor
        TemperatureSensor
            BraunThermoscan4000Sensor
            SureTempSensor
        VolumeSensor
        WeightSensor
            HealthOMeterScale
```

Figure 3. Medical Device Subclasses Hierarchy



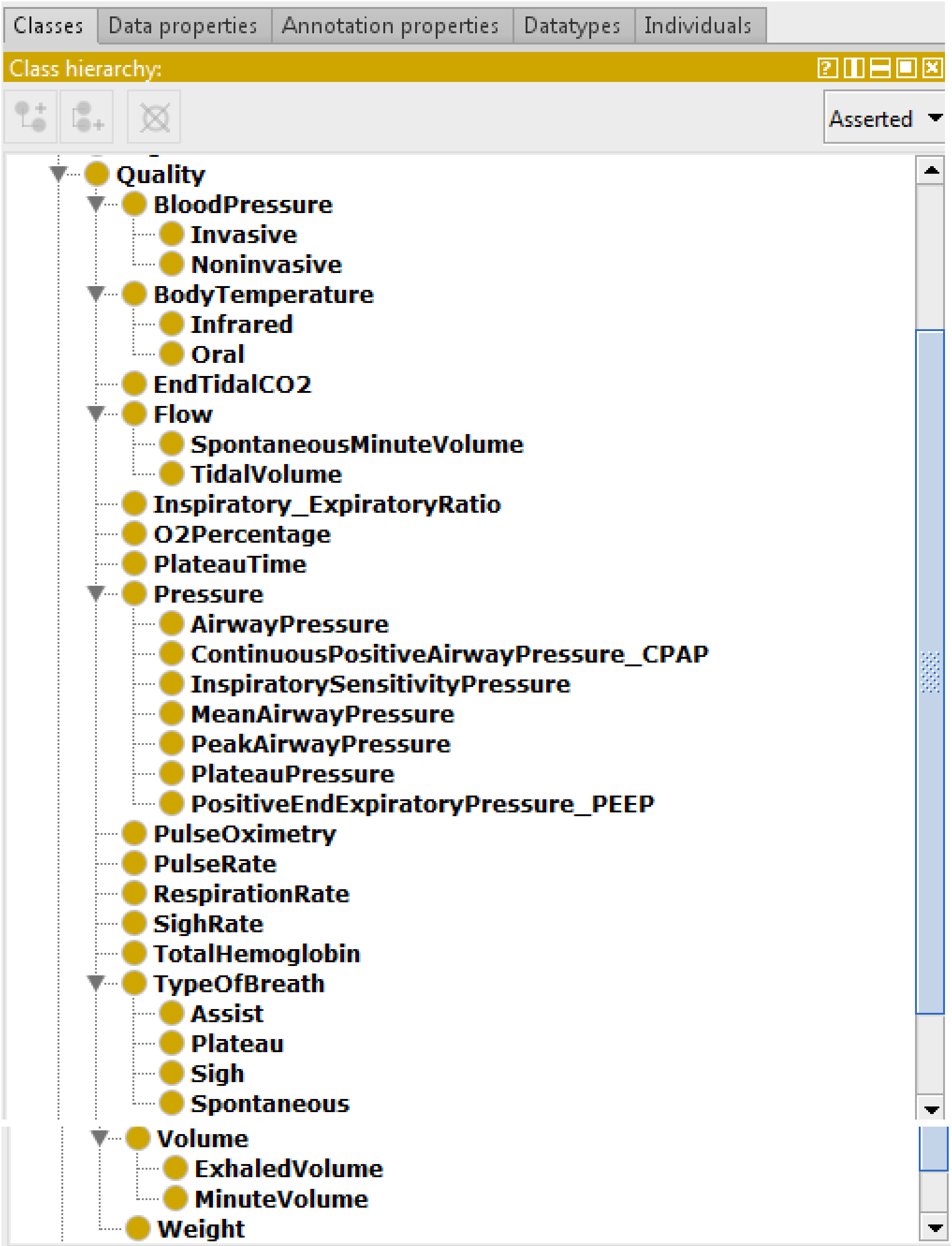
- owl:Thing
  - Device
  - DeviceComponent
  - Hospital
  - ModelID
  - NetworkProtocol
    - Serial_RS232 ≡ DB9 ≡ DB25
    - TCP/IP
      - DHCP
      - Static
    - Wireless
      - BlueTooth
      - IEEE802.11a/b/g
  - Organization
  - Quality
  - QualityType
  - SerialID

Figure 4: Network Protocol Subclass Hierarchy



- owl:Thing
  - Device
  - DeviceComponent
  - Hospital
  - ModelID
  - NetworkProtocol
  - Organization
    - DeviceManufacturer
      - BraunMfr
      - HealthometerMfr
      - MasimoMfr
      - NellcorPuritanBennettMfr
      - Nonin
      - Omron
      - PuritanBennetMfr
      - SureTempMfr
      - WelchAllynMfr
  - Quality
  - QualityType
  - SerialID

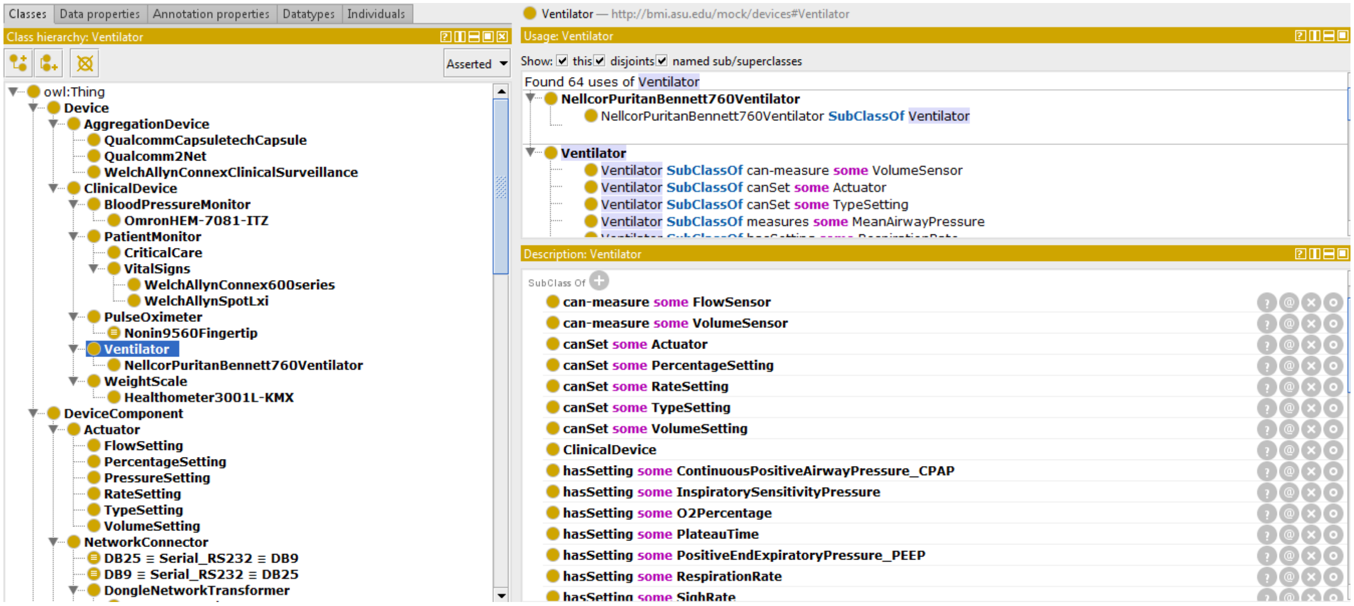
Figure 5: Organization (Medical Device Manufacturer) Class and Subclass Hierarchy



- Quality
  - BloodPressure
    - Invasive
    - Noninvasive
  - BodyTemperature
    - Infrared
    - Oral
  - EndTidalCO2
  - Flow
    - SpontaneousMinuteVolume
    - TidalVolume
  - Inspiratory_ExpiratoryRatio
  - O2Percentage
  - PlateauTime
  - Pressure
    - AirwayPressure
    - ContinuousPositiveAirwayPressure_CPAP
    - InspiratorySensitivityPressure
    - MeanAirwayPressure
    - PeakAirwayPressure
    - PlateauPressure
    - PositiveEndExpiratoryPressure_PEEP
  - PulseOximetry
  - PulseRate
  - RespirationRate
  - SighRate
  - TotalHemoglobin
  - TypeOfBreath
    - Assist
    - Plateau
    - Sigh
    - Spontaneous
  - Volume
    - ExhaledVolume
    - MinuteVolume
  - Weight

Figure 6: Quality (Sensor and Setting Readings) Class and Subclass Hierarchy

Figure 7 below demonstrates how the classes and object properties are used to build characteristics and relationships. In the example shown, a clinical device of a ventilator has been assigned an objecct property of ‘can measure’ some Flowsensor. In this case the object property of 'can measure' is affilliated with the subclass of Flowsensor and assigned as a characteristic of a ventilator.



Classes | Data properties | Annotation properties | Datatypes | Individuals

Class hierarchy: Ventilator

Asserted

```
owl:Thing
  Device
    AggregationDevice
      QualcommCapsuletechCapsule
      Qualcomm2Net
      WelchAllynConnexClinicalSurveillance
    ClinicalDevice
      BloodPressureMonitor
        OmronHEM-7081-ITZ
      PatientMonitor
        CriticalCare
        VitalSigns
          WelchAllynConnex600series
          WelchAllynSpotLxi
      PulseOximeter
        Nonin9560Fingertip
      Ventilator
        NellcorPuritanBennett760Ventilator
      WeightScale
        Healthometer3001L-KMX
  DeviceComponent
    Actuator
      FlowSetting
      PercentageSetting
      PressureSetting
      RateSetting
      TypeSetting
      VolumeSetting
    NetworkConnector
      DB25 ≡ Serial_RS232 ≡ DB9
      DB9 ≡ Serial_RS232 ≡ DB25
      DongleNetworkTransformer
```

Ventilator — http://bmi.asu.edu/mock/devices#Ventilator

Usage: Ventilator

Show: ☑ this ☑ disjoints ☑ named sub/superclasses

Found 64 uses of Ventilator

```
NellcorPuritanBennett760Ventilator
    NellcorPuritanBennett760Ventilator SubClassOf Ventilator

Ventilator
    Ventilator SubClassOf can-measure some VolumeSensor
    Ventilator SubClassOf canSet some Actuator
    Ventilator SubClassOf canSet some TypeSetting
    Ventilator SubClassOf measures some MeanAirwayPressure
    Ventilator SubClassOf hasSetting some RespirationRate
```

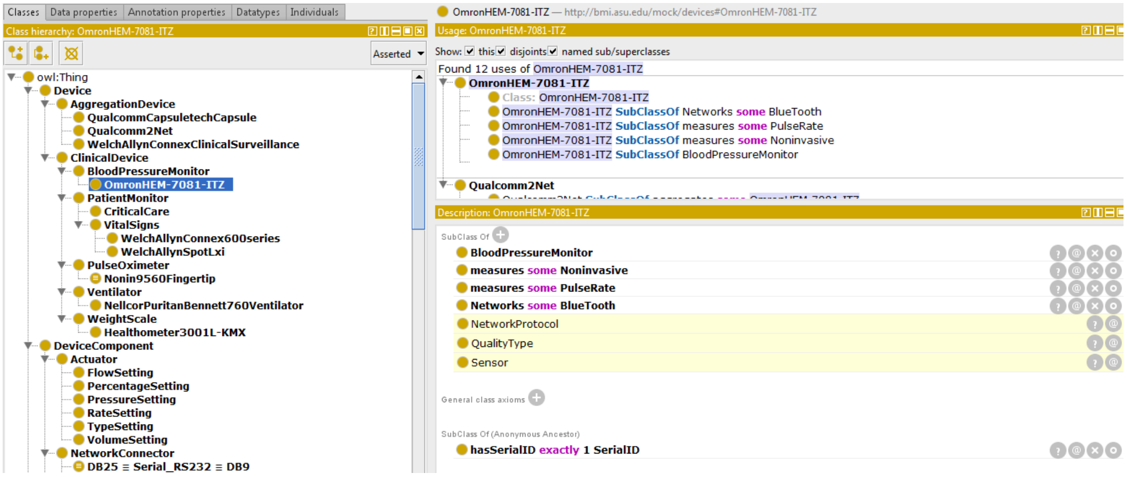
Description: Ventilator

SubClass Of

- can-measure some FlowSensor
- can-measure some VolumeSensor
- canSet some Actuator
- canSet some PercentageSetting
- canSet some RateSetting
- canSet some TypeSetting
- canSet some VolumeSetting
- ClinicalDevice
- hasSetting some ContinuousPositiveAirwayPressure_CPAP
- hasSetting some InspiratorySensitivityPressure
- hasSetting some O2Percentage
- hasSetting some PlateauTime
- hasSetting some PositiveEndExpiratoryPressure_PEEP
- hasSetting some RespirationRate
- hasSetting some SighRate

Figure 7: Clinical Device Ventilator Characteristics

Figure 8 below shows an example of a specific blood pressure model and its characteristics. It inherits some characeristics from a general blood pressure monintor, however, in this case the object property of networks is unique to this specific vendor model of blood pressure monitor so the affiliation is made at the model level and not the general device type class level.



Classes | Data properties | Annotation properties | Datatypes | Individuals

Class hierarchy: OmronHEM-7081-ITZ

Asserted

```
owl:Thing
  Device
    AggregationDevice
      QualcommCapsuletechCapsule
      Qualcomm2Net
      WelchAllynConnexClinicalSurveillance
    ClinicalDevice
      BloodPressureMonitor
        OmronHEM-7081-ITZ
      PatientMonitor
        CriticalCare
        VitalSigns
          WelchAllynConnex600series
          WelchAllynSpotLxi
      PulseOximeter
        Nonin9560Fingertip
      Ventilator
        NellcorPuritanBennett760Ventilator
      WeightScale
        Healthometer3001L-KMX
  DeviceComponent
    Actuator
      FlowSetting
      PercentageSetting
      PressureSetting
      RateSetting
      TypeSetting
      VolumeSetting
    NetworkConnector
      DB25 ≡ Serial_RS232 ≡ DB9
```

OmronHEM-7081-ITZ — http://bmi.asu.edu/mock/devices#OmronHEM-7081-ITZ

Usage: OmronHEM-7081-ITZ

Show: ☑ this ☑ disjoints ☑ named sub/superclasses

Found 12 uses of OmronHEM-7081-ITZ

```
OmronHEM-7081-ITZ
    Class: OmronHEM-7081-ITZ
    OmronHEM-7081-ITZ SubClassOf Networks some BlueTooth
    OmronHEM-7081-ITZ SubClassOf measures some PulseRate
    OmronHEM-7081-ITZ SubClassOf measures some Noninvasive
    OmronHEM-7081-ITZ SubClassOf BloodPressureMonitor

Qualcomm2Net
    Qualcomm2Net SubClassOf aggregates some OmronHEM-7081-ITZ
```

Description: OmronHEM-7081-ITZ

SubClass Of

- BloodPressureMonitor
- measures some Noninvasive
- measures some PulseRate
- Networks some BlueTooth
- NetworkProtocol
- QualityType
- Sensor

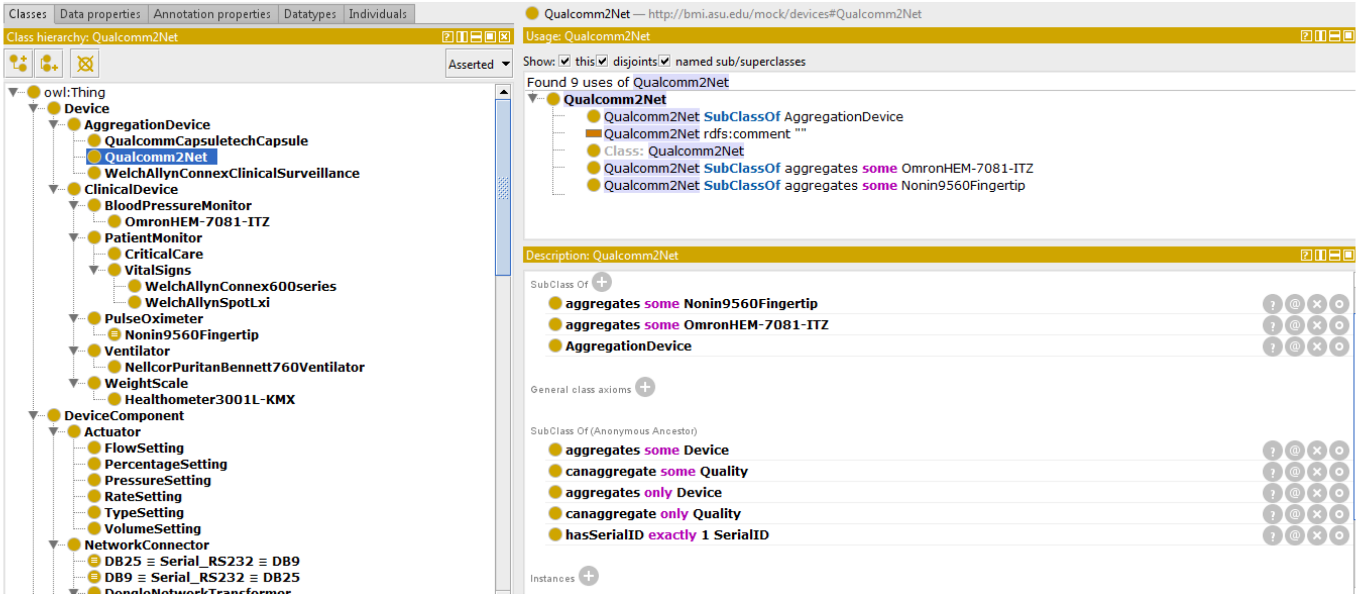
General class axioms

SubClass Of (Anonymous Ancestor)

- hasSerialID exactly 1 SerialID

Figure 8: Omron Blood Pressure Monitor Characteristics

In Figure 9 below, a last example of a device class is shown: the aggregation device. This device is the aggregator and interface point for all of the medical device data to the interface broker or EHR. As seen below the object property of aggregates is used to identify and affiliate which devices the aggregation device can aggregate. In this case the Qualcomm2Net is shown to aggregate two devices: the Omron blood pressure monitor and the Nonin pulse oximeter. This particular device is actually used in a patient home to aggregate medical device data from personal device that might be used in remote monitoring programs for patients with chronic diseases. It was included in the proof of concept to show that the onotology and knowledge base design need not only model hospital based scenarios.



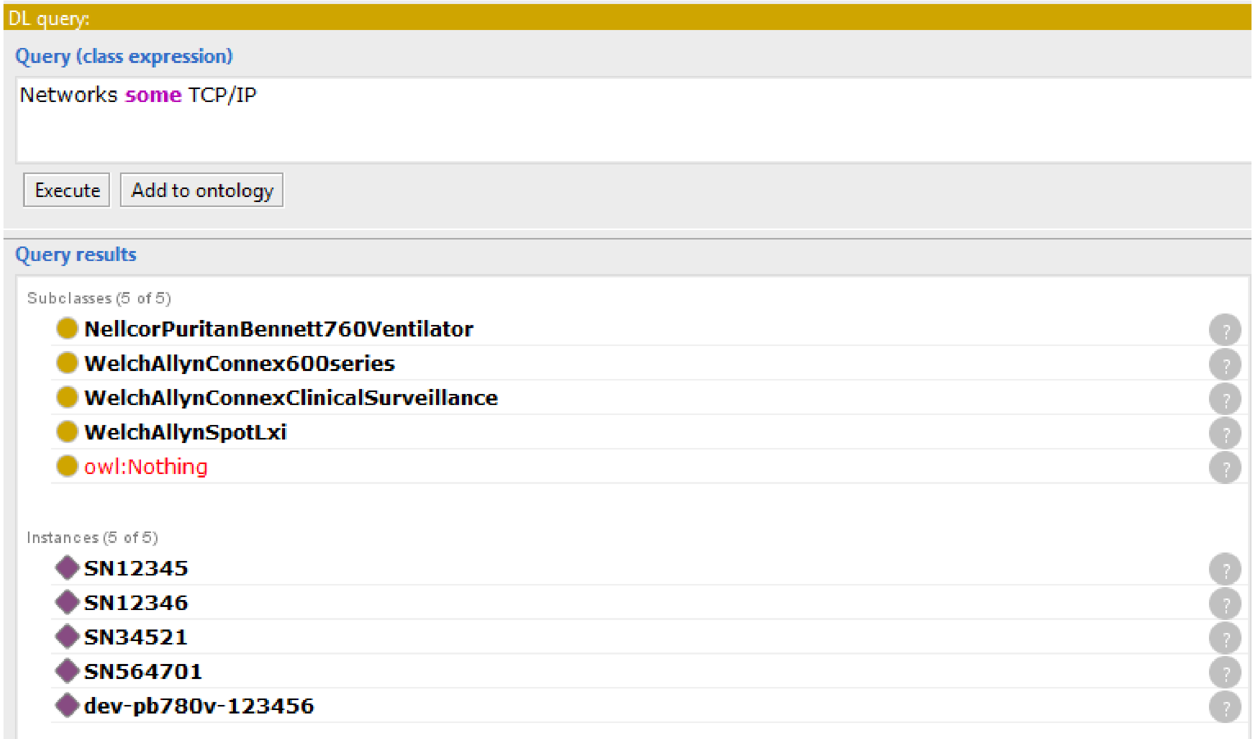
```
owl:Thing
  Device
    AggregationDevice
      QualcommCapsuletechCapsule
      Qualcomm2Net
      WelchAllynConnexClinicalSurveillance
    ClinicalDevice
      BloodPressureMonitor
        OmronHEM-7081-ITZ
      PatientMonitor
        CriticalCare
        VitalSigns
          WelchAllynConnex600series
          WelchAllynSpotLxi
      PulseOximeter
        Nonin9560Fingertip
      Ventilator
        NellcorPuritanBennett760Ventilator
      WeightScale
        Healthometer3001L-KMX
  DeviceComponent
    Actuator
      FlowSetting
      PercentageSetting
      PressureSetting
      RateSetting
      TypeSetting
      VolumeSetting
    NetworkConnector
      DB25 ≡ Serial_RS232 ≡ DB9
      DB9 ≡ Serial_RS232 ≡ DB25
      DongleNetworkTransformer
```

```
Qualcomm2Net — http://bmi.asu.edu/mock/devices#Qualcomm2Net

Show: ☑ this ☑ disjoints ☑ named sub/superclasses
Found 9 uses of Qualcomm2Net
Qualcomm2Net
    Qualcomm2Net SubClassOf AggregationDevice
    Qualcomm2Net rdfs:comment ""
    Class: Qualcomm2Net
    Qualcomm2Net SubClassOf aggregates some OmronHEM-7081-ITZ
    Qualcomm2Net SubClassOf aggregates some Nonin9560Fingertip

SubClass Of
    aggregates some Nonin9560Fingertip
    aggregates some OmronHEM-7081-ITZ
    AggregationDevice

General class axioms

SubClass Of (Anonymous Ancestor)
    aggregates some Device
    canaggregate some Quality
    aggregates only Device
    canaggregate only Quality
    hasSerialID exactly 1 SerialID

Instances
```

Figure 9: Qualomm2Net Aggregation Device Characteristics

Expert System Simulation: Queries.

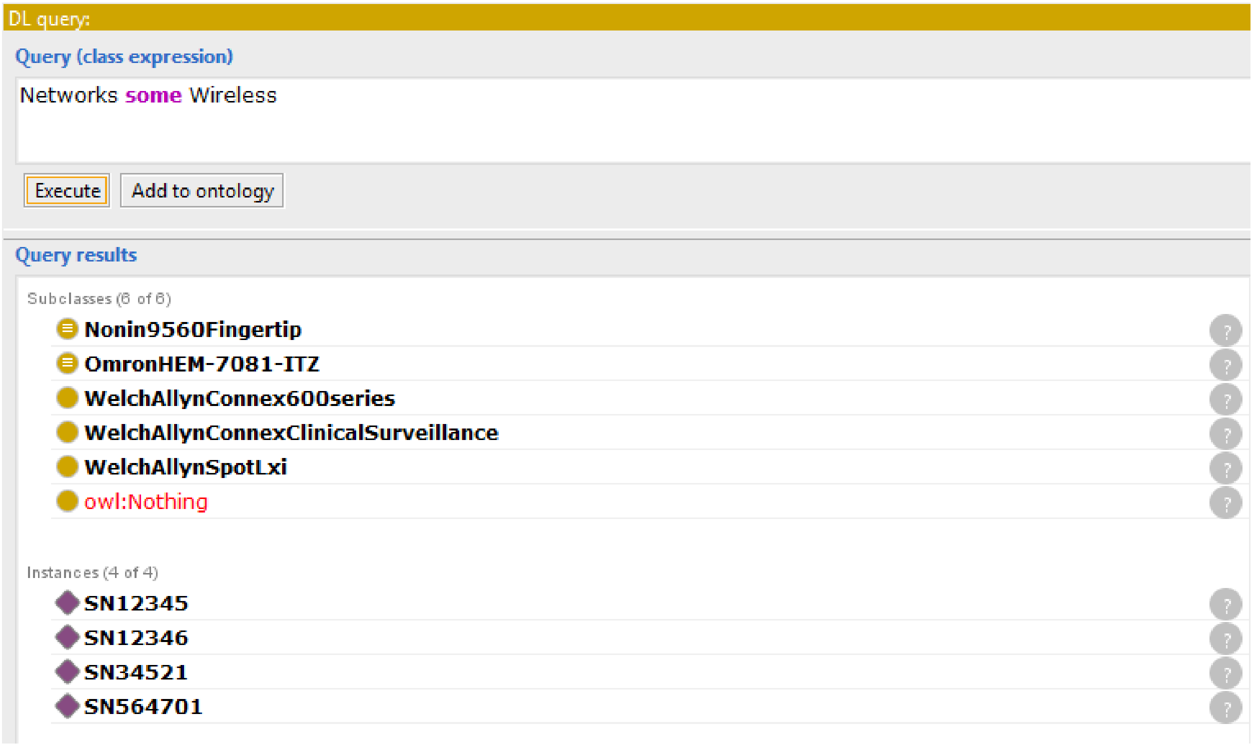
Figures 10 through 14 below show the results of three specific queries that demonstrate the key information that would be valuable when implementing amedical devic implementation system. Figure 10 show a query of the knowedge base about which medical devices have the ability to use the TCP/P protocol to network. As shown in the result window, there are four meical device models which have that characterisitc. In Protege one can also have specific instances of a class and in this case, dummy serial numbers were used to assign instances so that the proof of concept could show that the query would also retrieve the actual devices that have this capacitility and not just the model. Figure 11 shows a similar query result only this time for wireless networking capability.



```
Networks some TCP/IP
```

```
Subclasses (5 of 5)
  ● NellcorPuritanBennett760Ventilator
  ● WelchAllynConnex600series
  ● WelchAllynConnexClinicalSurveillance
  ● WelchAllynSpotLxi
  ● owl:Nothing

Instances (5 of 5)
  ◆ SN12345
  ◆ SN12346
  ◆ SN34521
  ◆ SN564701
  ◆ dev-pb780v-123456
```

Figure 10: Query Results for TCP/IP



Networks some Wireless

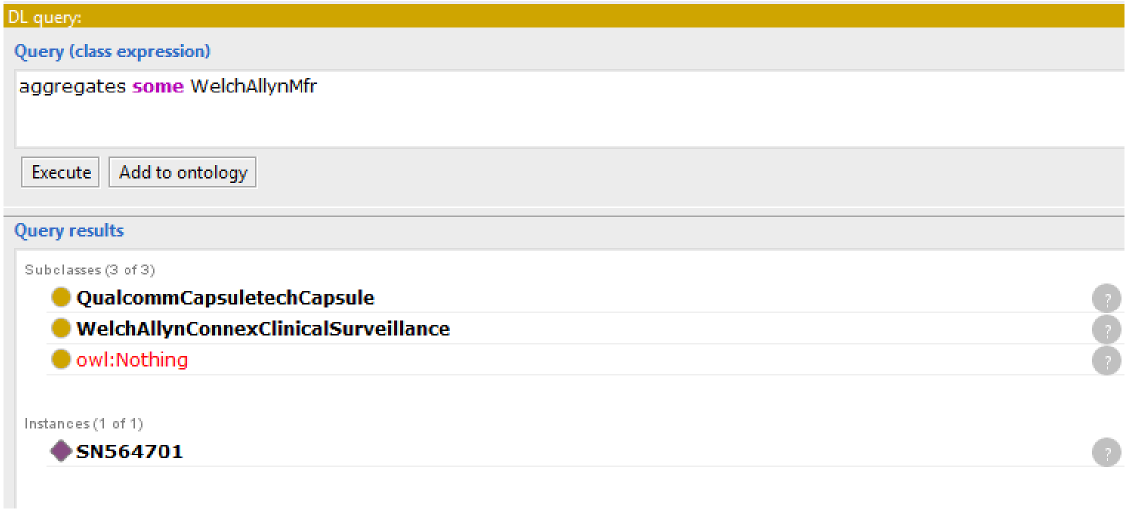
Subclasses (6 of 6)

- Nonin9560Fingertip
- OmronHEM-7081-ITZ
- WelchAllynConnex600series
- WelchAllynConnexClinicalSurveillance
- WelchAllynSpotLxi
- owl:Nothing

Instances (4 of 4)

- SN12345
- SN12346
- SN34521
- SN564701

Figure 11: Query Result for Wireless



aggregates some WelchAllynMfr

Subclasses (3 of 3)

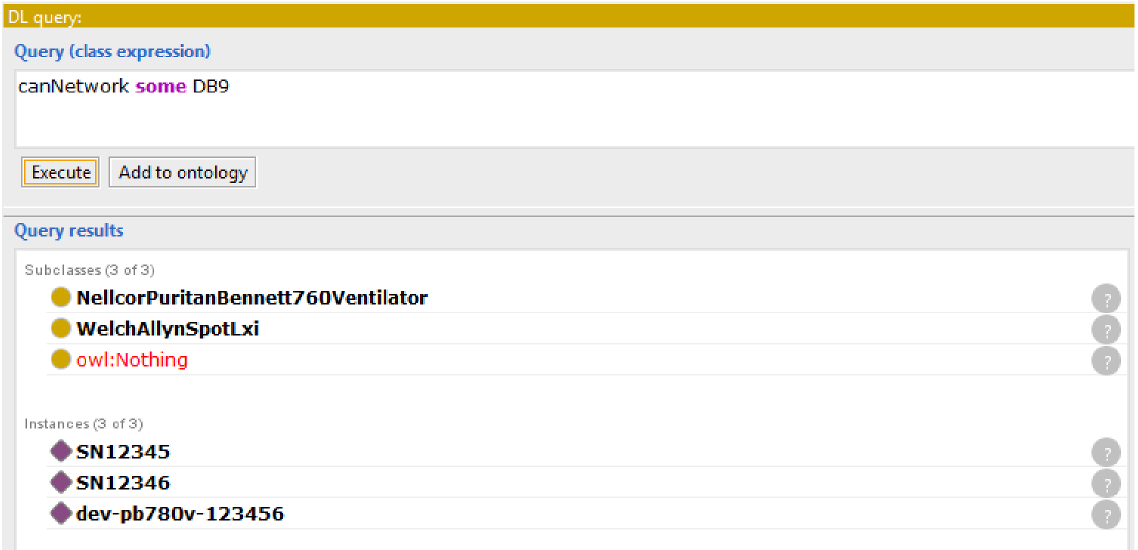
- QualcommCapsuletechCapsule
- WelchAllynConnexClinicalSurveillance
- owl:Nothing

Instances (1 of 1)

- SN564701

Figure 12: Query results for Welch-Allyn Aggregation

Figure 13 shows a query of the knowledge base to determine what type of medical devices have DB9 network connectors. This question is an indirect way to determine if a network transformation is needed for the medical device to connect to the network and integrate its data to an EHR. In Figure 3, there is a subclass hierarchy of network dongle that could also be queried to determine if there is a network transformation dongle available for that medical device.



```
canNetwork some DB9
```

Subclasses (3 of 3)

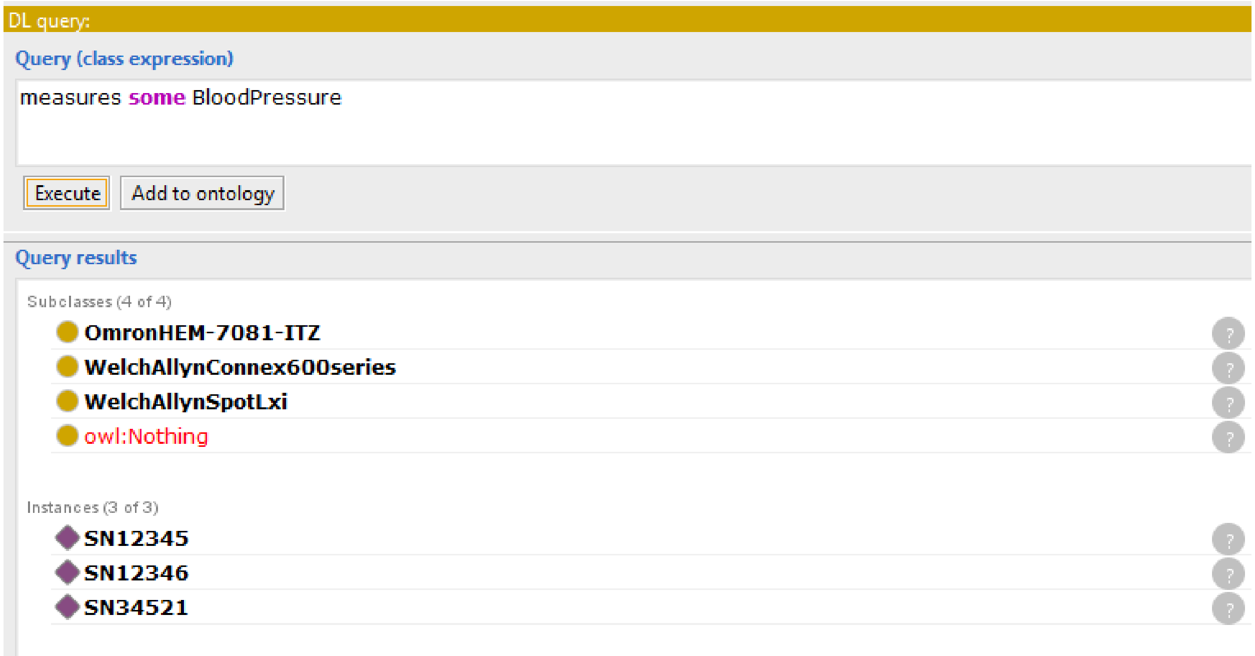
- NellcorPuritanBennett760Ventilator
- WelchAllynSpotLxi
- owl:Nothing

Instances (3 of 3)

- SN12345
- SN12346
- dev-pb780v-123456

Figure 13: Query Results of DB9 Network Connector

Figure 14 below demonstrates the secondary object of the project to show what sensor and setting outputs are available with a medical device inventory by showing the query results of which medical devices in the knowledge base measure blood pressure. As the results window shows, there are three devices which offer that sensory information for aggregation and integration.



measures some BloodPressure

Subclasses (4 of 4)

- OmronHEM-7081-ITZ
- WelchAllynConnex600series
- WelchAllynSpotLxi
- owl:Nothing

Instances (3 of 3)

- SN12345
- SN12346
- SN34521

Figure 14: Query Results for Medical Devices that Measure Blood Pressure

Conclusion and Future Directions

The purpose of this project was to build a prototype knowledge base and expert system to assist in implementing a medical device integration system to an interface broker or EHR and in a rudimentary way, the onotology and knowledge base built demonstrated the concept. A recent literature review a well as my extensive experience and knowledge regarding implementing medical device integration systems showed that the current direction for medical device integration is to build middleware solutions that either query a medical device directory or direct device manufacturers to build network, sensor and setting sentience into the device to be queried by the middleware. There is no such medical device directory openly available nor do most medical devices have the sentient capability described. Therefore, this project built a prototype that has not been built before.

Due to the project constraints, Protégé was used to build a medical device ontology with some specific medical device model information. For the expert system functionality the reasoner function extension to Protégé was used to query the knowledge base. Specific screen shots of the ontology, knowledge base and query results were shown to demonstrate the effectiveness of the prototype.

As per the description of the project, there were quite a few constraints to the project: time limitations, knowledge base and expert system limitations, and medical device characteristics availability limitations. As a result, simulation of an expert system functionality was demonstrated versus the original intent to have an expert system output the medical device integration characteristics of a medical device inventory that would be compared to the knowledge base. Moreover, only nine separate medical device models were modeled in the knowledge base. Nevertheless, I picked nine separate medical device models that provided a good overall distribution of networking, sensor and setting capabilities attributed to most medical device classes.

The most difficult parts of this project were learning Protégé in such a short time and then building the generic medical device integration ontology so that when a specific model was entered into the knowledge base, a fairly repeatable process could be used for each model. This involved identifying the device class, devices attributes and then assigning those attributes by associating the different object properties along with the other classes to the medical device class and model. This knowledge base is missing the location information ontology and units ontology. This was due to the time constraints as those ontologies would need to also be designed and implemented within the project as there was no complete ontology representing those correctly for this project that could be inherited into the medical device integration ontology and knowledge base. For the location ontology, specific reference to healthcare organizations and care locations would be necessary.

When designing the medical device integration ontology, I named the generic medical device classes. In the future, some of the specific nomenclatures already developed (FDA could be used for those generic names. Another nomenclature that could possibly be used in the quality class for sensor and setting names is the data standard nomenclatures developed by IEEE and LOINC. In addition, the prototype could be more robust with the addition of more specific medical device model knowledge. Moreover, the ontology could include the data types that are available in a standard configuration versus in a vendor proprietary configuration. Lastly, an expert system 'front-end' could be developed that would allow comparison of a healthcare organization's medical device inventory with the knowledge base to automatically determine the 'integrate-ability' status of their medical device inventory.

I'd like to thank Davide Sottara, PhD, Arizona State University and Mayo Institute, for his guidance and assistance in building this prototype.

References

Cabri, G., De Mola, F., & Leonardi, L. (2007). Agent-based plug-and-play integration of role-enabled medical devices. Paper presented at the 111-121. doi:10.1109/HCMDSS-MDPnP.2007.18

Densford, F. (2015) Update: Qualcomm buys Capsule, Look to “Internet of Medical Things”. 16 Sep 2015. Mass Device. http://www.massdevice.com/qualcomm-buys-capsule-tech/. Accessed 5 Jul 2017.

Falcon Capital Partners, LLC. (2013). Nant Healthcare, Inc, acquires iSirona, Inc. Jan 2013. http://www.falconllc.com/nant-healthcare-inc-aquires-isirona/ Accessed 5 Jul 2017.

Foundation for Intelligent Physical Agents (FIPA), FIPA Device Ontology Specification. (2002) http://www.fipa.org/specs/fipa00091/SI00091E.html, 2002. Accessed 29 June 2017.

Gregorczyk, D., Bubhaus, T., & Fischer, S. (2012). A proof of concept for medical device integration using web services. Paper presented at the 1-6. doi:10.1109/SSD.2012.6198124

Hao, A., & Wang, L. (2015). Medical device integration model based on the internet of things. The Open Biomedical Engineering Journal, 9(1), 256-261. doi:10.2174/1874120701509010256

Horridge, M. (2011) A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools Edition 1.3. 24 Mar 2011. The University at Manchester; OWL@Manchester. Protégé OWL Tutorial.

http://owl.cs.manchester.ac.uk/publications/talks-and-tutorials/protg-owl-tutorial/. Accessed 15 Jun 2017.

IHE (2017) Integrating the Healthcare Enterprise. IHE Patient Care Device. http://www.ihe.net/Patient_Care_Devices/. Accessed 5 Jul 2017.

King, A., Procter, S., Andresen, D., Hatcliff, J., Warren, S., Spees, W., . . . Weininger, S. (2009). An open test bed for medical device integration and coordination. Paper presented at the 141-151. doi:10.1109/ICSE-COMPANION.2009.5070972

Kliem, A., Hovestadt, M., & Kao, O. (2012). Security and communication architecture for networked medical devices in mobility-aware eHealth environments. Paper presented at the 112-114. doi:10.1109/MobServ.2012.15

Kliem, A., Boelke, A., Grohnert, A., & Traeder, N. (2016). A reconfigurable middleware for on-demand integration of medical devices. Irbm, 37(4), 198-209. doi:10.1016/j.irbm.2016.05.003

Moengain, B. (2015) CPC, Nuvon to Merge. Healthcare IT News, 11 Sep 2015. http://www.healthcareitnews.com/news/cpc-nuvon-merge. Accessed 5 Jul 2017.

Moorman, B. (2008). Biomedical device interfacing to clinical information systems: A primer. Biomedical Instrumentation & Technology, 42(3), 205-208. doi:10.2345/0899-8205(2008)42[205:BDITCI]2.0.CO;2

Moorman, B. (2010). Medical device interoperability: Overview of key initiatives. Biomedical Instrumentation & Technology, 44(2), 132-138. doi:10.2345/0899-8205-44.2.133

Moorman, B. (2017) BMI 598 Final Project: Initial Assessment of Nomenclatures and Ontologies to Support Development of an Expert System to Support Implementation of a Medical Device Integration System to an EHR. 6 May 2017.

Personal Connected Health Alliance. (2017) Continua Design Guidelines. http://www.pchalliance.org/continua-design-guidelines. Accessed 5 Jul 2017.

QUDT.org. (2017) QUDT- Quantities, Units, Dimensions and Data Types Ontologies Catalog. 8 May 2017. http://www.qudt.org/release2/qudt-catalog.html. Accessed 5 Jul 2017.

US Food and Drug Administration. Unique Device Identification-UDI. (2017) https://www.fda.gov/MedicalDevices/DeviceRegulationandGuidance/UniqueDeviceIdentification/. Accessed 5 Jul 2017.