

# Modelado de procesos continuos para la toma de decisiones

Guillermo Lio<sup>1</sup>, Gustavo Placer<sup>1,2</sup>, Nestor R. Barraza<sup>3</sup>

<sup>1</sup> Universidad Nacional General Sarmiento,  
Buenos Aires, Argentina

<sup>2</sup> Universidad Interamericana,  
Buenos Aires, Argentina

<sup>3</sup> Universidad Nacional de Tres de Febrero,  
Caseros, Argentina  
glio@ungs.edu.ar

## Resumen

*En este trabajo se presenta la aplicación de una ontología para la descripción general, exploración y consultas de una planta piloto industrial, la cual integra distintos lazos de control como temperatura, nivel y caudal para usos didácticos y de investigación, con el fin de luego poder escalar el sistema a procesos industriales de gran escala. El desarrollo propuesto se enmarca en la integración dirigida por el conocimiento (knowledge driven approach) utilizando las herramientas y estándares de World Wide Web Consortium (W3C). Estas herramientas aportan una semántica formal de un ámbito determinado de la realidad, permitiendo una comprensión clara de procesos continuos complejos. Este diseño permite flexibilizar la comunicación entre el nivel de campo de la pirámide del CIM (Computer integrated manufacturing) con los niveles superiores, lo cual aporta un análisis rápido para la toma de decisiones.*

## 1. Introducción

En el ámbito del control se incorporan rápidamente los avances en el campo del software debido a la importancia que éste tiene en todas las disciplinas técnicas.

Hoy en día es necesaria una integración vertical entre el control [1], la informática y la representación de los datos, para lo cual es necesaria la creación de mecanismos de representación del conocimiento comunes y cada vez con un mayor nivel de abstracción en la representación de la información, para realizar procesos de toma de decisión más complejos en forma automática [2].

El lenguaje y la semántica en el campo del control automático juegan un papel muy importante hoy en día, siendo las ontologías y la Web Semántica algunas de las herramientas más utilizadas. Las ontologías son especificaciones formales que proporcionan una vía para representar conocimiento en un dominio específico con el propósito de facilitar la comunicación, reusar y compartir información entre usuarios donde el modelado ontológico pone énfasis en la formalización de los conocimientos, y el modelado de objetos representa conceptualmente los datos requeridos por una base de datos. Las ontologías permiten transformar la web en un espacio de conocimiento además de un espacio de información y permiten la reutilización de recursos heterogéneos [3], [4].

La informática está produciendo una evolución hacia el uso de modelos de conocimiento. La tendencia es unir los esfuerzos e ideas de los campos de la ingeniería del software e ingeniería de control para poder conseguir esta evolución.

El uso del software para ingeniería de control necesita incrementar el nivel de representación para hacer frente a las demandas actuales y futuras, permitiendo el acceso a los datos a través de la Web.

Técnicamente, las herramientas relacionadas con las ontologías son adecuadas para conseguir este nivel de representación necesario. Es posible realizar modelos de conocimiento del dominio de la ingeniería de control con ontologías, aunque los lenguajes existentes hoy en día no permiten una total representación de las estructuras de conocimiento necesarias, debiendo implementar parte de la semántica mediante procesadores externos.

En este trabajo proponemos integrar esquemas de representación ontológicos con los conceptos de la ingeniería de procesos, aplicando una capa semántica

entre el plano de procesos y los clientes. La Fig. 1 muestra la arquitectura propuesta.

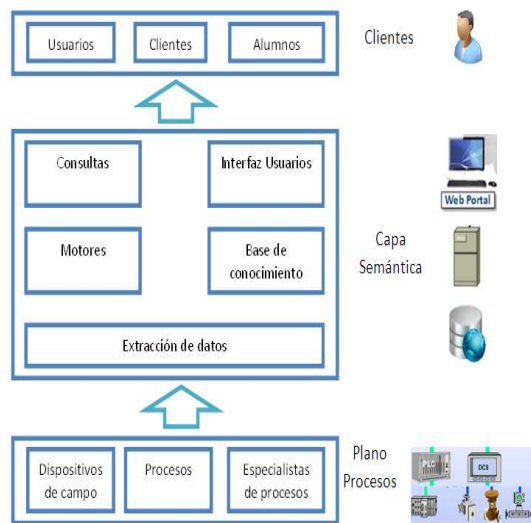


Fig. 1. Integración propuesta.

Para llevar a cabo esta representación propuesta, se presenta un diseño de una planta piloto industrial simple basado en ontologías. Este diseño permite separar en módulos los elementos (eventos, algoritmos, equipos, sensores y actuadores, etc.) del sistema de control con una alta especificación y un bajo acoplamiento, lo cual permite una facilidad de comunicación y posible reutilización.

La planta modelizada, integra cinco lazos de control. La misma cuenta con dos recipientes contenedores de agua, 2 bombas centrífugas, sensores de nivel, caudal, presión y temperatura, actuadores del tipo válvula globo, proporcional, resistencia calefactora y radiador con forzador.

## 2. Herramienta de desarrollo

Para la representación del conocimiento en primer lugar se realiza una formalización de este conocimiento, luego se implementan herramientas y estándares del W3C en torno a la creación de la Web Semántica. Esto impulsó el desarrollo de herramientas y lenguajes específicos para ontologías. El lenguaje que se utiliza para la representación se denomina OWL (Web Ontology Language) el mismo consta de clases, relaciones, funciones, instancias, entre otros. Permite describir conceptos y además cuenta con un conjunto de operadores como intersección, unión, negación. El OWL está basado en una lógica descriptiva. Para representar los datos en la Web generalmente se utiliza el lenguaje RDF [6], expresa proposiciones utilizando vocabularios formales y precisos, permitiendo el acceso y uso del

World Wide Web. Es un estándar que puede ser utilizado para representar la información de manera que las aplicaciones informáticas puedan utilizarla de manera escalable.

La herramienta elegida para construir este trabajo fue el Protegé [7], que utiliza el formalismo OWL/RDF con una misma apariencia en la interfaz y el funcionamiento, además tiene carácter de distribución bajo licencia de código abierto.

Los componentes básicos de la herramienta son [7]:

**Clases:** representan conceptos que son agrupaciones de individuos que comparten un conjunto de propiedades.

**Slots:** recogen las propiedades que distinguen a las clases. Sobre los tipos de valores que aparecerán en los slots se pueden establecer restricciones por medio de las facetas (cardinalidad, valores por defecto, dominio, rango, etc.)

**Instancias:** representan individuos concretos de una clase, correspondiendo a elementos reales en el mundo representado. La distinción entre instancia como concepto de la ontología e individuo como elemento en el mundo externo es importante. En el caso de las lógicas descriptivas y OWL dos instancias en la ontología pueden referirse a un mismo individuo: es una de las consecuencias de la asunción de mundo abierto.

**Axiomas:** Protegé ofrece un lenguaje para la imposición de restricciones sobre los slots de forma más flexible que por medio de las facetas predefinidas.

## 3. Esquema de representación

La elección del sub-dominio será la primer tarea a desarrollar ya que con una buena elección del mismo se podrán manifestar las bondades y ventajas del modelado del conocimiento en el campo del software para la ingeniería del control. La metodología para la construcción de ontologías tienen las siguientes facetas:

- Definir el alcance de la ontología.
- Evaluar la posibilidad de reutilización de otras ontologías y/o partes de ontología.
- Enumerar los términos que aparecen en el dominio.
- Definir la taxonomía de conceptos. De arriba abajo, de abajo arriba o combinando las dos.
- Definir las propiedades.
- Definir las facetas.
- Definir las instancias.

Comprobar las estructuras creadas.

## 4. Modelo de la Planta

Para la representación jerárquica de los equipos se siguió el estándar de la norma ISA-S88 [8], la misma permite una especificación en detalle de los sistemas de control. Se definió en primer lugar la clase de mayor jerarquía

“Planta Piloto” en mención al nombre que lleva la planta real que se encuentra en los laboratorios de Universidad. De esta se desprenden dos grandes subclases “Lazo Control” y “Tipo Lazo Control”. Este diseño tiene la ventaja de separar, por un lado, el lazo de control con sus tres componentes fundamentales (Actuador, Sensor, Controlador), obteniendo una arquitectura general para todos los lazos existentes. Por el otro lado, tenemos los tipos de lazo para esta planta (caudal, presión, temperatura, nivel) pudiendo asignarle, gracias a las propiedades de objeto y restricciones definidas, una descripción de los equipos que las integran con la ventaja de ser de fácil exploración, búsqueda y de consulta. En los ejemplos propuestos a lo largo del trabajo puede verificarse que la estructura elegida es acorde a la ingeniería de automatización y control, utilizando los mismos conceptos del control industrial para la descripción de los lazos.

Para volcar la información pertinente (marca, modelo, datos técnicos, etc.) a cada uno de los equipos de la planta se utilizaron los individuos en conjunto con las propiedades de datos y de anotación, pudiendo así demostrar la facilidad con la que se pueden obtener los datos de un equipo en particular de cualquier lazo en un sistema de control con múltiples lazos, cosa que es muy importante para el ingeniero de control y/o de procesos. Este tipo de modelado fomenta compartir el conocimiento entre distintos sistemas, representándolos de manera intuitiva y extensible, pudiendo obtener a futuro la ventaja de que los usuarios puedan explorar, buscar y consultar sobre todos los equipos a través de la web semántica.

Este trabajo, propone además, una validación en la Planta Piloto de la Universidad y así luego poder hacerlo escalable al ámbito de las aplicaciones industriales a gran escala.

## 5. Estructura

Las clases se representarán con un círculo naranja y sus nombres siempre empiezan con mayúsculas. Tenemos 2 clases de mayor jerarquía “PlantaPiloto” y “ComponentesLazoControl”, de las cuales se desprenden sus sub-clases:

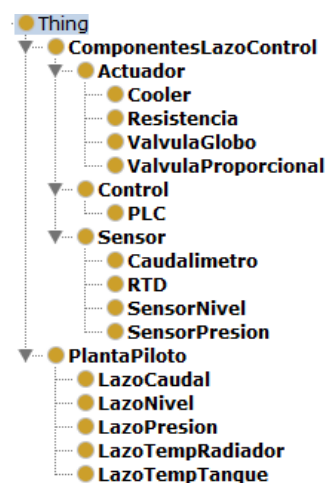


Fig. 2. Clases de mayor jerarquía.

La clase “ComponentesLazoControl” tiene todos los componentes de un lazo de control típico. Fig. 3.

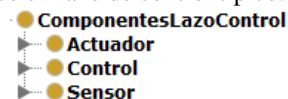


Fig. 3. SubClases ComponentesLazoControl.

Aquí se ven las sub-clases que se desprenden de “LazoControl”. Tenemos la clase “Actuador”, “Control” y “Sensor”. En cada una de ellas tenemos las sub-clases Fig. 4:

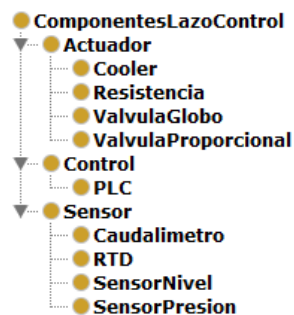


Fig. 4. SubClasesLazoControl.

Aquí vemos la estructura total de la clase “ComponentesLazoControl”.

## 6. Propiedades de objeto

Las propiedades representan relaciones entre objetos. En OWL existen dos tipos de propiedades: propiedades de objeto y propiedades de datos. Las propiedades de objeto tienen como elementos: Nombre, Dominio y Rango. Cada propiedad de objeto debe tener su inversa. Si una propiedad enlaza al elemento A con el B, debe haber una propiedad inversa que enlace el elemento B con el A. Se simbolizarán con un rectángulo azul y

siempre empiezan en minúsculas. La propiedad “hasElementoControl” se usa para relacionar los componentes de un lazo de control. Por ejemplo: Un lazo de control tiene un elemento de control llamado PLC. A su vez “PLC” es un elemento de control del lazo. El dominio es la clase mayor “PlantaPiloto” y el rango limita la propiedad “Control”. En la Fig. 5 podemos ver las propiedades y las propiedades inversas:

- hasElementoControl
- hasElementoFlujo
- hasEquipoPotencia
- hasInstrumentoMedicion
- isElementoControlOf
- isElementoFlujoOf
- isEquipoPotenciaOf
- isInstrumentoMedicionOf

Fig. 5. Propiedades Inversas.

Las descripciones de cada propiedad son como la que se muestra para “hasElementoControl”, donde apreciamos su propiedad inversa “isElementoControlOf”, su dominio “PlantaPiloto” y su rango “Control” Fig. 6:

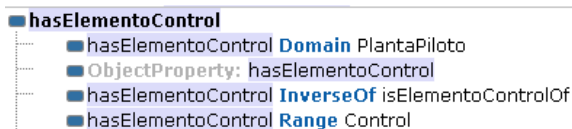


Fig. 6. Descripción.

Este tipo de relaciones establecen una conexión necesaria y suficiente entre ambas clases, permitiendo que el razonador detecte automáticamente los distintos lazos internos y pueda clasificarlos con la clase correspondiente.

## 7. Definiciones de clases y su comportamiento

El diseño de la ontología fue desarrollado de una manera que permita una representación adecuada que contenga los datos básicos para su identificación.

Una vez creadas las propiedades, se procedió y describir el comportamiento de las clases para los distintos lazos de control que posee la planta. Las mismas se utilizan para crear restricciones en las clases para las ontologías OWL. Generalmente el nombre de la propiedad debería sugerir la restricción propuesta.

Hay 3 categorías de restricciones:

- Restricciones de cuantificación.
- Restricciones de cardinalidad.
- Restricciones de valor

Las restricciones de cuantificación se componen de los siguientes elementos:

- Cuantificador existencial ( $\exists$ ), el cual permite indicar la existencia de al menos un objeto. En protégé la palabra clave “some” es usado para denotar  $\exists$ .

- Cuantificador universal ( $\forall$ ), el cual permite indicar la existencia de todos los objetos. En protégé la palabra clave es “only” es usado para denotar  $\forall$ .

Usamos la restricción “some” para indicar las restricciones a las clases de los diferentes lazos que conforman la totalidad de la planta. Por ejemplo para la clase “LazoCaudal” se aplicó la restricción “some”, el lazo de caudal tiene como elemento de control al PLC. O sea, el lazo si o si tiene que tener un controlador para su funcionamiento. El mismo lazo caudal nos quedará restringido a los siguientes componentes como mínimo Fig. 7:

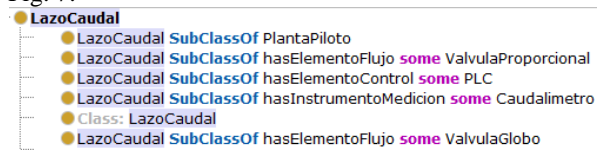


Fig. 7. Restricciones LazoCaudal.

De la misma manera, se definieron para la planta piloto los distintos lazos de control en cuestión Fig. 8:



Fig. 8. Tipos Lazo Control.

A cada uno de los lazos se le colocó las restricciones pertinentes a cada uno de los componentes que forman el lazo en cuestión quedando de la siguiente manera Fig. 9:

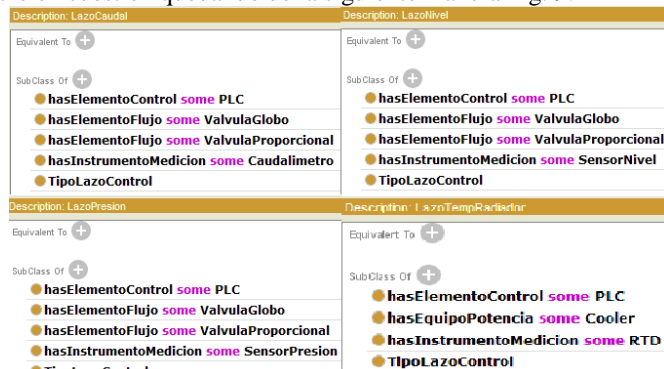


Fig. 9. Restricción Lazos.

## 8. Propiedades de datos

Las propiedades de datos se usan para relacionar un individuo con un XML Schema Datatype value o un literal RDF. Las mismas empiezan siempre con minúsculas y su nombre debe hacer referencia a la relación que se le va a dar. Se simbolizan con un

rectángulo verde. Por ejemplo, una válvula tiene diámetro 1 pulgada Fig. 10.

● **ValvulaGlobo** ■ **hasDiámetro "1Pulgada"**

Fig. 10. Propiedad de datos de ValvulaGlobo.

Dependiendo de las relaciones, se definen las distintas propiedades de datos, para nuestro sistema se definieron las siguientes Fig. 11:

- **hasComunicacion**
- **hasDiámetro**
- **hasFrecuencia**
- **hasPotencia**
- **hasPresion**
- **hasRango**
- **hasSalida**
- **hasVoltaje**

Fig. 11. Propiedades de Datos para el sistema.

## 9. Propiedades de anotación

Las propiedades de anotación permiten generar notas en las clases, individuos, cabeceras de ontologías e individuos. Por ejemplo se puede definir como propiedad al "Fabricante" para describir la marca de un equipo dado. Para nuestra descripción se utilizaron las siguientes propiedades de anotación Fig. 12:

- **Fabricante**
- **Modelo**
- **NroSerie**
- **ProteccionIP**
- **Tipo**

Fig. 12. Propiedades de anotación.

Las mismas se utilizaron para describir los distintos equipos que hay en la planta piloto.

## 10. Individuos

Los individuos representan objetos del dominio de interés y son también conocidos como instancias. Las clases OWL están formadas por individuos y pueden ser organizadas dentro de una jerarquía de clases y subclases conocida como taxonomía. Por ejemplo dentro de la clase Caudalímetro se creó el individuo o instancia "EndressHauser". Para la descripción del mismo se pueden apreciar las propiedades de anotación, así también como las propiedades de datos Fig. 13:

The screenshot shows two panels. The top panel, titled 'Annotations: EndresshauserPromag', lists the following properties and values:
 

- Fabricante**: Endress Hauser
- Modelo**: Promag
- NroSerie**: 33FW25
- ProteccionIP** [type: integer]: 67
- Tipo**: Caudalimetro

 The bottom panel, titled 'Property assertions: EndresshauserPromag', lists the following data property assertions:
 

- hasPotencia**: "15VA"
- hasRango**: "110000m3/h"
- hasDiámetro**: "1Pulg"
- hasFrecuencia**: "50Hz"
- hasComunicacion**: "RS485"
- hasRango**: "20-55V"

Fig. 13. Propiedades del Caudalímetro.

## 11. Ontograf

Esta herramienta da apoyo a navegar de forma interactiva en las relaciones de las ontologías OWL. Contiene varios diseños apoyados por la organización de forma automática dando la estructura de su ontología. Múltiples compatibilidades: subclase, individuos, dominios, propiedades de los objetos de rango y la equivalencia. Utilizando esta herramienta observamos la estructura y relaciones de la clase LazoControl Fig. 14.

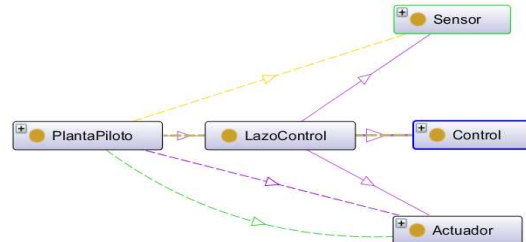


Fig. 14. Estructura Planta Piloto con OntoGraf.

Permite ver el URI en cada uno de los elementos. Un URI puede ser clasificado como un localizador, un nombre o ambos. Un localizador de recursos uniforme (URL) es un URI que, además de la identificación de un recurso, proporciona medios de actuar sobre o la obtención de una representación del recurso mediante la descripción de su mecanismo de acceso a la red primaria o "ubicación". En el ejemplo anterior podemos observar para la clase Sensor la dirección URI:

<http://www.semanticweb.org/ModeladoPlantaPiloto#Sensor>

asi también como la dependencia de la clase LazoControl Fig. 15.



Fig. 15. Dirección URI y superclases para Sensor.

Ejemplo: estructura y dependencia de la clase Actuador. Fig. 16:

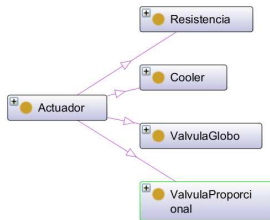


Fig. 16. Estructura Actuador en OntoGraf.

Con las propiedades podemos generar los distintos tipos de lazo de control contenidos en la planta piloto Fig. 17:

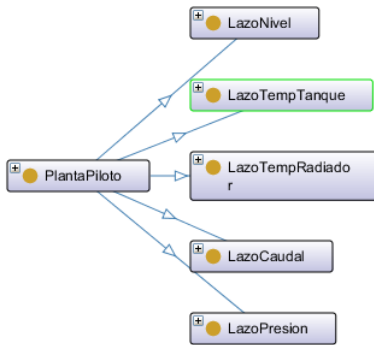


Fig. 17. Estructura Tipo Lazos en OntoGraf.

Para el caudalímetro podemos ver que es un Sensor que usa se usa en el Lazo de Caudal y que tiene como individuo al EndresshauserPromag Fig. 18.

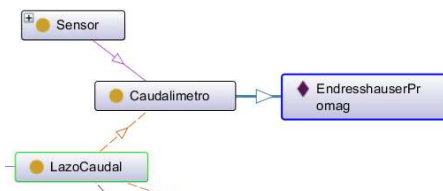


Fig. 18. Estructura Caudalímetro en OntoGraf..

## 12. Consultas SPARQL

SPARQL es un lenguaje de consulta semántica de bases de datos, capaz de recuperar y manipular los datos

almacenados [9]. Una consulta SPARQL contiene tres cláusulas principales: SELECT, WHERE y un filtro.

Se utilizó la consulta SPARQL para buscar los instrumentos que componen los distintos lazos de control de la planta piloto, así también como las características de los equipos que los integran.

En algunas de las consultas propuestas, la cláusula SELECT propone mostrar el Tipo de Lazo de control, Tipo de Instrumento y el Instrumento en sí. En otras consultas el SELECT pretende mostrar las características de un equipo en particular.

Para obtener los resultados propuestos, en la cláusula WHERE, se utilizaron y combinaron las siguientes sentencias:

?a rdfs:subClassOf ?b: "a" es subclase de "b".

?a owl:onProperty ?b: "b" contiene las propiedades de objeto.

?a owl:someValuesFrom ?b: Restricción, cuantificador existencial "some". En "b" muestra las restricciones.

FILTER regex(str(?a), "string"): Filtra los "string" dentro de "a".

Filter isLiteral(?a): Filtra los "literales" que se encuentran en "a".

Ejemplo 1: Para consultar los instrumentos que integran el lazo de Caudal de la Fig. 19, se utilizó la sentencia descrita de la Fig. 20.

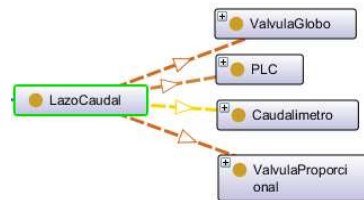


Fig. 19. Instrumentos del Lazo Caudal.

```

SELECT ?Lazo ?Tipo_de_Instrumento ?Instrumento
WHERE {
  ?Lazo rdfs:subClassOf ?b.
  ?b owl:onProperty ?Tipo_de_Instrumento.
  ?b owl:someValuesFrom ?Instrumento.
  FILTER regex(str(?Lazo), "LazoCaudal").
}
  
```

Fig. 20. Consulta SPARQL para el lazo de caudal.

En la Fig. 21 vemos el resultado arrojado para la consulta:

Lazo	Tipo_de_Instrumento	Instrumento
LazoCaudal	hasElementoFlujo	ValvulaGlobo
LazoCaudal	hasElementoControl	PLC
LazoCaudal	hasInstrumentoMedicion	Caudalimetro
LazoCaudal	hasElementoFlujo	ValvulaProporcional

Fig. 21. Resultado de la consulta SPARQL para el lazo de caudal.

Ejemplo 2: Para consultar los instrumentos que integran el lazo de temperatura del tanque de la Fig. 22., se utilizó la sentencia de la Fig. 23.

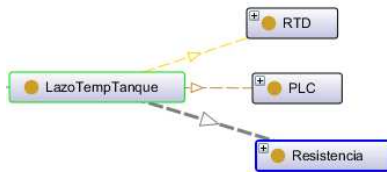


Fig. 22. Instrumentos del Lazo Temp. Tanque.

```
SELECT ?Lazo ?Tipo_de_Instrumento ?Instrumento
WHERE {?Lazo rdfs:subClassOf ?b.
?b owl:onProperty ?Tipo_de_Instrumento.
?b owl:someValuesFrom ?Instrumento.
FILTER regex(str(?Lazo), "LazoTempTanque").
}
```

Fig. 23. Consulta SPARQL para el Lazo Temp. Tanque.

En la Fig. 24 vemos el resultado arrojado para la consulta:

Lazo	Tipo_de_Instrumento	Instrumento
LazoTempTanque	hasInstrumentoMedicion	RTD
LazoTempTanque	hasElementoControl	PLC
LazoTempTanque	hasEquipoPotencia	Resistencia

Fig. 24. Resultado de la consulta SPARQL para el lazo de temperatura del tanque.

Ejemplo 3: Para consultar las características del instrumento en particular “Belimo” Fig. 25, se utilizó la sentencia de la Fig. 26.

Fig. 25. Datos de la válvula proporcional Belimo.

```
SELECT ?Propiedades ?Datos
WHERE {?a ?Propiedades ?Datos.
filter isLiteral(?Datos).
FILTER regex(str(?a), "Belimo")
}
```

Fig. 26. Consulta SPARQL para el individuo Belimo.

En la Fig. 27 vemos el resultado arrojado para la consulta:

Propiedades	Datos
Fabricante	"Belimo"@
Modelo	"LRB24"@
hasPresion	"6Bar"@
hasVoltaje	"24V"@
hasDiametro	"1/2Pulg"@
hasPotencia	"1.5W"@

Fig. 27. Resultado de la consulta SPARQL para el individuo Belimo.

Aquí vemos que el resultado de la consulta nos muestra todas las características del equipo Belimo, como su modelo, presión, voltaje, diámetro, potencia.

### 13. Conclusiones y Trabajos Futuros

Con las herramientas utilizadas de la Web Semántica hemos establecido que son totalmente aplicables a la Ingeniería de Control pues permiten una modelización y cohesión con muy poco acoplamiento entre los módulos, lo que nos da la posibilidad de una reutilización del código establecido en diferentes aplicaciones industriales.

La utilización de XML Schema permite un acoplamiento eficiente entre los diferentes niveles de la pirámide del CIM, lo cual facilitaría la toma de decisiones empresariales incluyendo la utilización de la nube, por ejemplo, utilizando los estándares del IFML (Interaction Flow Modeling Lenguaje) que permite el diseño interactivo de interfaz gráfica y de fácil programación. De esta manera, es posible realizar aplicaciones en las que los elementos que aparecen en la interfaz gráfica, con la que el usuario interactúa, estén relacionados con los conceptos teóricos subyacentes si éstos están recogidos en una ontología.

Utilizando las herramientas de la Web Semántica nos proponemos hacer como trabajo futuro el diseño de un Scada en la Web, para utilizar lo realizado en investigación y diseño de un laboratorio remoto. Esta tecnología proporcionaría una plataforma estándar que se ocuparía del monitoreo y control de procesos a través de Internet.

### 14. References

[1]. KALOGERAS, A. Vertical integration of enterprise industrial systems utilizing web services. IEEE Transactions On Industrial Informatics, v. 2, n. 2, p. 120-128, 2006

[2]. Murray, R. M.; Åström, K. J.; Boyd, S. P.; Brockett, R. W.; Stein, G. (2003) Future directions in control in an information-rich world, IEEE Control Systems Magazine, 23(2), pp 20 – 33

- [3]. Barrera Milagros, Núñez Haydemar, Ramos Esmeralda; (2012). Ingeniería Ontológica. Universidad central de Venezuela, Escuela de Computación. ISSN 1316-6239. Enero 2012.
- [4]. Fernando Roda, Estanislao Musulin and Marta Basualdo; (2012). Esquemas de representación ontológica para la integración de datos en los sistemas de información de planta. 1° Simposio Argentino de Informática Industrial, SII 2012; 41 JAIHO - SII 2012 - ISSN: 2313-9102. 2012
- [5]. Bissell, C. (1993) A new way of talking: aspects of the creation of the language of control engineering, Faculty of Technology/Systems Architecture Group Internal Report (SAG/1993/RR31/CCB), November 1993
- [6]. Klyne, G.; Carroll, J. J. (eds.) (2004) Resource Description Framework (RDF): concepts and abstract syntax, W3C Recommendation, 10 February 2004.
- [7]. Wang, H.; Rector, A.; Drummond, N.; Horridge, M.; Seidenberg, J.; Noy, N.; Musen, M.; Redmond, T.; Rubin, D.; Tu, S.; Tudorache, T. (2006) Frames and OWL Side by Side, In 9th International Protégé Conference, Stanford, CA, USA
- [8]. ISA: ANSI/ISA-S88.01 - 1995, Batch Control, Part I: Models and Terminology (1995)
- [9]. <https://www.w3.org/TR/rdf-sparql-query/>