

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
VALPARAISO - CHILE



“INTEGRANDO DATOS DE GASTOS DE LA
ADMINISTRACIÓN PÚBLICA CHILENA PARA
MEJORAR LA TRANSPARENCIA A NIVEL DE
MUNICIPIO ”

FRANK QUIROZ ORLANDI

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN INFORMÁTICA

Profesor Guía: Carlos Buil

Profesor Correferente: Cecilia Reyes

Septiembre - 2021

RESUMEN

Resumen— El diluvio de datos entrega una clara visión de lo que es hoy en día la red, un sinfín de información siendo intercambiada, usada, obtenida, etcétera, tanto es así que hacen falta mecanismos que permitan hacer un uso apropiado de esta. La aparición de los principios FAIR trajo una gran ayuda para las distintas áreas del conocimiento y el consumo, lo que conlleva a que se crearan cada vez más ecosistemas de información. La idea de sumar la información de la administración pública chilena a este movimiento puede permitir que esta sea usada para mejorar tanto como los trabajos de los gobiernos como para darle más mecanismos de evaluación a la ciudadanía que espera lo mejor de estos.

Palabras Clave— Principios de datos FAIR; datos enlazados; bases de datos de grafos; web semántica; Integración de datos; Datos Abiertos

ABSTRACT

Abstract— The deluge of data provides a clear vision of what the network is today, an endless amount of information being exchanged, used, obtained, etc., so much so that mechanisms are needed to make appropriate use of it. The emergence of the FAIR principles brought a great help to the different areas of knowledge and consumption, which led to the creation of more and more information ecosystems. The idea of adding Chilean public administration information to this movement can allow it to be used to improve both the work of governments and to provide more evaluation mechanisms to citizens who expect the best from them.

Keywords— FAIR data principles; linked data; graph databases; semantic web; data integration; Open Data

GLOSARIO

API: Application Programming Interface.

CCA: Creative Commons Atribución.

CSV: comma-separated values.

DI: Departamento de Informática.

DIG: Domain-specific Insight Graphs.

FAIR: Find, Access, Interoperate and Reuse.

IRI: Internationalized Resource Identifier.

OKG: Open Knowledge Graph.

ONG: Organizacion No Gubernamental.

OWL: Web Ontology Language.

RDF: Resource Description Framework.

RDFS: Resource Description Framework schema.

SAAM: Smithsonian American Art Museum.

URI: uniform resource identifier.

URL: Uniform Resource Locator.

UTFSM: Universidad Técnica Federico Santa María.

W3C: World Wide Web Consortium.

XML: Extensible Markup Language.

XSD: XML Schema Definition.

ÍNDICE DE CONTENIDOS

RESUMEN	I
ABSTRACT	II
GLOSARIO	III
ÍNDICE DE FIGURAS	VI
ÍNDICE DE TABLAS	VII
ÍNDICE DE CÓDIGOS	VIII
INTRODUCCIÓN	1
CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA	2
1.1 Problema general	2
1.2 Objetivos	4
1.2.1 Objetivo general	4
1.2.2 Objetivos Específicos	4
1.3 Alcance	4
CAPÍTULO 2: MARCO CONCEPTUAL	5
2.1 Ley de Transparencia	6
2.2 Consejo para la transparencia	6
2.2.1 Portal de Transparencia	6
2.3 ChileCompra	7
2.3.1 MercadoPublico	7
2.4 Web Semántica	7
2.4.1 URI	9
2.4.2 XML	11
2.4.3 RDF	12

2.4.4	Vocabularios: RDFS y OWL	15
2.4.5	SPARQL	17
2.4.6	Wikimedia	20
2.4.7	DBpedia	20
2.5	Integración de datos	21
2.6	FAIR Principles	21
2.6.1	Principios	21
2.6.2	FAIRificación	22
2.7	METHONTOLOGY	23
2.8	Protégé	24
2.9	Tarql	25
2.10	TripleStore	26
2.10.1	Apache Jena	26
2.11	Antecedentes	27
CAPÍTULO 3: PROPUESTA DE SOLUCIÓN		32
3.1	Recopilación de Recursos	32
3.2	Desarrollo de vocabulario	36
3.3	SPARQL como lenguaje de integración y transformación de datos . . .	39
3.4	Almacenamiento de RDF y publicación	42
CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN		44
4.1	Presentación de solución	44
CAPÍTULO 5: CONCLUSIONES Y TRABAJO FUTURO		53
5.1	Trabajo Futuro	54
REFERENCIAS BIBLIOGRÁFICAS		55

ÍNDICE DE FIGURAS

2.1	Pila de tecnologías de la Web Semántica	8
2.2	Ejemplo de una URI	9
2.3	Ejemplo en Turtle.	14
2.4	Mismo ejemplo en XML/RDF.	14
2.5	Grafo generado.	15
2.6	Ejemplo de base de datos RDF que utiliza RDFS y OWL	16
2.7	Consulta para obtener recursos y sus etiquetas	18
2.8	Consulta para obtener recursos, propiedades literales y su valor	19
2.9	Diagrama del proceso de FAIRificación	23
2.10	Arquitectura Apache Jena	27
2.11	La ontología SAAM.	28
2.12	Ejemplo de grafo de conocimiento para el caso de trata de personas.	29
2.13	La organización de alto nivel de la ontología BioKNO.	30
2.14	La nueva arquitectura diseñada para el ecosistema KnetMiner.	31
3.15	Portal de Transparencia Municipalidad de Valparaiso	33
3.16	Pagina de inicio Mercado Publico.	34
3.17	Interfaz de datos abiertos de chilecompra	35
3.18	Interfaz de descarga masiva de chilecompra	35

3.19	Acercamiento a sección Personal y remuneraciones del organismo	36
3.20	Tareas de conceptualización de METHONTOLOGY.	37
3.21	Interfaz de Query	42
3.22	Interfaz de carga de datos	43
4.23	Acceso a plantillas de personal de todos los organismos.	44
4.24	Ontología completa realizada para la integración.	46
4.25	Registro de carga de licitaciones de un mes en Fuseki.	47
4.26	Lista preliminar de datos en el servidor de Pubby.	49
4.27	Muestras de Datos en Pubby	50
4.28	Documentación WIDOCO para el Proyecto.	51

ÍNDICE DE TABLAS

1	Resultados de la consulta ilustrada en la figura 2.7	19
2	Resultados de la consulta ilustrada en la figura 2.8	20
3	Las siguiente tabla forma parte de un archivo .CSV que contiene información sobre empresas.	26
4	La siguiente tabla entrega una muestra de los resultados de la QUERY 5 .	48

ÍNDICE DE CÓDIGOS

1 Mapeo de la tabla 3	25
2 Ejemplo proveedores	39
3 Ejemplo proveedores	40
4 Ejemplo proveedores corrigiendo datos	41
5 QUERY en búsqueda de conocimiento antes no accesible.	48
6 Ejemplos de triples obtenidos.	52

INTRODUCCIÓN

En 2010, el surgimiento de una sociedad basada en los datos fue destacado por una revista de noticias con una portada dedicada a *"the data deluge"* [Economist, 2010] (o *diluvio de datos* en español). La recopilación, el intercambio y el procesamiento de datos e información han sido actividades cada vez más fundamentales para mejorar nuestras vidas, que están repercutiendo en campos como los negocios, la industria, la medicina, la agricultura, humanidades e incluso la justicia. Esto ha sido posible gracias a los avances de la informática y a la difusión de los estándares de Internet, que han proporcionado los medios técnicos para tratar grandes cantidades de datos interconectados y han creado una cultura que favorece el intercambio y distribución de información. Los datos se han vuelto tan fundamentales en las ciencias y en otras áreas que un grupo de interesados publicó los "principios FAIR" [2.6.1], que establecen que los datos deben ser Encontrables, Accesibles, Interoperables y Reutilizables, principios que si bien han sido formalizados, en la actualidad son normas que se han buscado implementar por años. Organizaciones como el W3C han promovido la idea de aprovechar la World Wide Web tradicional para crear la web semántica [2.4], que permite publicar y consumir una web de datos, de forma similar a lo que hacemos con la más conocida web de documentos. Los proyectos de datos abiertos enlazados han surgido como realizaciones concretas de esa visión, donde el adjetivo "abierto" hace hincapié en los aspectos legales y sociales y en la necesidad de hacer que la información sea libremente accesible y reutilizable. Esta línea de pensamiento a llevado a gente de distintas áreas a llevar a cabo sus propios planes de integración de datos para aprovechar las ventajas que la web semántica trae a sus campos. Para abordar este flujo de pensamiento se planea llevar a cabo un proceso de integración[2.5] y FAIRificación[2.6.2] de datos de la administración pública chilena, área que si bien tiene el concepto de datos abiertos aplicado, no presenta un esquema que permita obtener nuevo conocimiento a partir de los datos ni otorga la oportunidad de ser reutilizados.

CAPÍTULO 1

DEFINICIÓN DEL PROBLEMA

En este capítulo se describirá el problema a resolver, comenzando con el problema general, la información recopilada sobre este, reconocimiento de los participantes y el entorno, finalizando los objetivos y el alcance del problema acotado.

1.1. Problema general

El 20 de abril del año 2009 el estado puso en marcha la ley 20.285 [Biblioteca del Congreso Nacional, 2008], o mejor conocida como ley de Transparencia, esta ley consiste en respetar y cautelar la publicidad de los actos, resoluciones, procedimientos y documentos de la Administración, así como las de sus fundamentos y en facilitar el acceso a esa información a cualquier persona. Obliga además, establece que toda persona tiene derecho a solicitar y recibir información de cualquier órgano de la Administración del Estado, creando una entidad llamada consejo de transparencia que tiene como objetivo promover la transparencia, fiscalizar su cumplimiento y garantizar el acceso a la información.

La presencia del consejo de transparencia conllevó la creación del portal de transparencia del estado chileno, lo que permitió que un gran número de organismos del estado subieran su información a la web para estar al alcance de la gente y se pudiera regular de forma apropiada lo que provocó que esta plataforma acumulara una gran cantidad de data.

Dentro de los requerimientos de la ciudadanía, uno de ellos es la facilidad de saber sobre como se gastan los impuestos y los recursos estatales, esto está dentro del portal de transparencia llevando a los usuarios a ChileCompra, la plataforma donde los organismos chilenos licitan y compran para los proyectos, la información referente a esta se abordará debidamente en la sección 2.3.

Las cantidades grandes de información traen consigo la necesidad de hacer accesible y

reutilizable dicho conocimiento, la "herramienta" mas conocida para dicho proceso es hacer uso de los conceptos de web semántica, con estos se puede lograr extraer nuevo conocimiento gracias a la facilidad para operar los datos con maquinas que otorga este modelo de web, pero en la actualidad los datos de las plataformas mencionadas no cuentan con las herramientas de integración ni los modelos necesarios para llevar a cabo estos conceptos a cabildada, por lo que hay que buscar, identificar y/o desarrollar un modelo de datos común para que se pueda aprovechar como está prevista esta información, tanto para las personas como para las maquinas haciendo de esta información reutilizable, en esta época la falta de estas herramientas genera un problema a gran escala.

Si se aterriza como es debido el problema a un ámbito mas local, los ciudadanos se verían beneficiados con saber como se gastan los recursos del estado y de sus propios bolsillos (impuestos) en sus localidades por medio de los municipios, el organismo del estado mas cercano a ellos con responsabilidad directa en este asunto, a su vez ganar mas conocimiento de los datos que se registran en los portales de transparencia mejoraría la gestión de las mismas municipalidades y mejoraría la toma de decisiones tanto en gobiernos de turno o en periodos de transición.

Con todo esto en mente el problema se reduce en lograr que la gestión publica local obtenga una herramienta de integración y una base de datos que siga los estándares de los Principios FAIR para que pueda ser reutilizada en distintos niveles de trabajo o para conocimiento general del publico.

1.2. Objetivos

1.2.1. Objetivo general

Lograr integrar los datos de chileCompra con los datos de las municipalidades siguiendo los principios FAIR ¹ para lograr un ambiente digital donde la información pueda ser usada, estudiada y analizada tanto como por computadores como personas.

1.2.2. Objetivos Específicos

1. Lograr obtener información de municipios por medio del portal de transparencia, buscando una forma de automatizar su obtención para trabajo futuro.
2. Obtener los datos necesarios desde chileCompra referentes a los organismos del estado mencionados anteriormente, en un formato en el que puedan ser procesados y organizados.
3. Crear un vocabulario basado en los datos obtenidos debidamente documentado.
4. Trabajar con una herramienta que permita procesar los datos obtenidos haciendo uso del vocabulario desarrollado asegurando que cumplan los principios FAIR.
5. Habilitar una plataforma adecuada para consultar los datos.

1.3. Alcance

Como el problema en general se extiende en todas las entidades estatales, tratar ese problema en general seria un proyecto demasiado grande, por lo que en particular se desarrollara ambientado solo a municipalidades limitadas por la cantidad de datos a procesar, unas dos o tres de momento, con lo que se obtendrá el personal, las licitaciones efectuadas y las adjudicaciones a distintos proveedores.

¹FAIR Principles: <https://www.go-fair.org/fair-principles/>

CAPÍTULO 2

MARCO CONCEPTUAL

A continuación, se procederá a describir el marco conceptual que actuará como fundamento teórico para la integración de datos sobre gastos localizado a municipios. Éste, se compondrá de los siguientes elementos:

- Ley de Transparencia
- Consejo de transparencia
- ChileCompra
- Web Semántica
- Integración de datos
- FAIR Principles
- Methontology
- Protégé
- Tarql
- Triplestore
- Antecedentes

2.1. Ley de Transparencia

La ley de transparencia ² es una ley que otorga a toda persona el Derecho de acceder a la información pública, funcionando como una llave para acceder a la información pública de toda institución pública. Esta ley se desarrolla en 2 dimensiones:

- Transparencia activa: El acceso permanente a información a través de los sitios web de los organismos públicos.
- Derecho a acceso a la información: El deber que tienen los organismos públicos de recibir solicitudes de información y entregar ésta, salvo que exista motivo de secreto o reserva. Dicho derecho está reconocido en el artículo 8º de la Constitución.

En la práctica la ley, si bien permite obtener la información requerida y existe un consejo que se encarga de que esto funcione, esta información no se puede encontrar de forma expedita ni tampoco se pueden hacer grandes análisis de forma fácil.

2.2. Consejo para la transparencia

El Consejo para la Transparencia ³ es una corporación autónoma de derecho público, con personalidad jurídica y patrimonio propio, creado por la Ley de Transparencia de la Función Pública y de Acceso a la Información de la Administración del Estado.

Su principal labor es velar por el buen cumplimiento de dicha ley, la que fue promulgada el 20 de agosto de 2008 y entra en vigencia el 20 de abril de 2009.

2.2.1. Portal de Transparencia

El portal de transparencia ⁴ es una plataforma del Estado de Chile, orientada a facilitar acceso de las personas a la información pública al permitirles realizar solicitudes de

²Ley de transparencia: <http://bcn.cl/1uuq2>

³Consejo para la transparencia: <https://www.consejotransparencia.cl/quienes-somos/>

⁴Portal Transparencia: <https://www.portaltransparencia.cl/PortalPdT>

información y revisar la información de Transparencia Activa. En esta plataforma se encuentra un repositorio amplio con la información de distintos organismos del estado.

2.3. ChileCompra

ChileCompra ⁵ es la institución que administra la plataforma de compras públicas de Chile, esta es un marco regulatorio único. Esta institución administra la plataforma de licitaciones del estado por lo que maneja la información de todos o la gran mayoría de los gastos de las instituciones públicas. Estas compras incluyen desde material de oficina, medicamentos y alimentos hasta servicios de transporte o consultorías. Al contar con tal cantidad de información esta plataforma es una herramienta muy poderosa para analizar el trabajo y los gastos de las organizaciones públicas pero esto es bastante complicado de hacer para una persona natural.

2.3.1. MercadoPublico

MercadoPublico ⁶ es una plataforma electrónica donde los organismos públicos de Chile realizan los procesos de compras y los proveedores ofrecen sus productos y servicios, espacio de oferta y demanda administrado por ChileCompra.

2.4. Web Semántica

La Web Semántica es un conjunto de actividades propuestas por la W3C con el objetivo de generar tecnologías para la publicación de datos en la Web, de tal manera que sean procesables por las máquinas. Se basa en la idea de añadir metadatos semánticos y ontológicos para describir el contenido y las relaciones entre los datos publicados. Con esto se logra mejorar la interoperatividad de Internet pues los programas de computadora podrán acceder a la información en un lenguaje que comparte el mismo vocabulario y modelo de datos, para así procesar su contenido, razonar en base a éste y combinarlo

⁵Chile compras: <https://www.chilecompra.cl/que-es-chilecompra/>

⁶MercadoPublico: <https://www.mercadopublico.cl/Home/Contenidos/QueEsMercadoPublico?esNuevaHome=true>

para resolver problemas cotidianos automáticamente.

Es posible rastrear los orígenes de esta idea hasta una propuesta temprana de la World Wide Web en 1986 [Berners-Lee y CERN, 1989] y el subsecuente trabajo de Berners-Lee [Berners-Lee, 1992] donde se prevé la necesidad de una evolución del contenido Web desde objetos legibles por las personas a información semántica orientada a máquinas. Para lograr los objetivos de la Web Semántica se han generado múltiples estándares y tecnologías como son RDF, RDF Schema, XML Schema, OWL, SKOS y SPARQL. Estas tecnologías dan la capacidad de vincular información arbitraria en la Web y se utilizan actualmente para generar datos enlazados (Linked Data), creando así una “*red de las cosas del mundo, descrita por los datos en la Web*” [Berners-Lee, 2011].

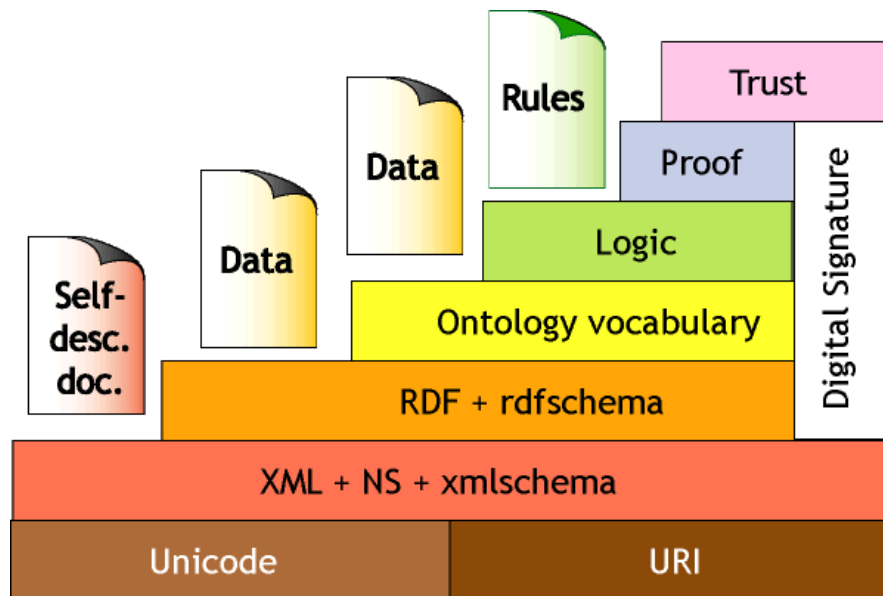


Figura 2.1: Pila de tecnologías de la Web Semántica

De Semantic Web - XML2000 - Tim Berners-Lee [Berners-Lee, 2014]

En las siguientes secciones se describirán mas detalladamente algunos de los conceptos presentados en la figura 2.1 comenzando desde la base, con las tecnologías del hipertexto (URI en 2.4.1 y XML en 2.4.2), para luego pasar a las tecnologías de la Web Semántica mas estandarizadas (RDF en 2.4.3 y RDFS y OWL en 2.4.4) y terminar con el lenguaje de consulta SPARQL en la sección 2.4.5.

2.4.1. URI

Una URI (del las siglas del ingles *Uniform Resource Identifier*) es una cadena de caracteres utilizada para identificar un recurso inequívocamente. Si bien en un principio solo acepta un subconjunto de caracteres ASCII (el alfabeto en ingles y los numeros) y “codigo por ciento” (en el cual, por ejemplo ‘%26’ es ‘&’) el estandar fue extendido el 2005 a IRI (*Internationalized Resource Identifier*) que puede contener caracteres Unicode/ISO 10646, incluyendo chino, japones, coreano, entre otros [Gangemi y Presutti, 2006]. En este documento no se diferenciará entre URI e IRI.

La sintaxis actual de una URI, definida por Berners-Lee et al. el 2005 [Masinter *et al.*, 2005], se presenta en la figura 2.2.

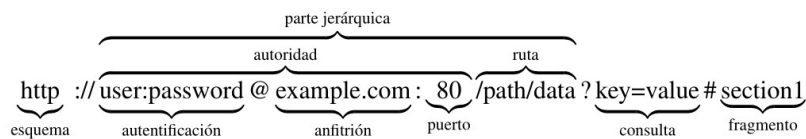


Figura 2.2: Ejemplo de una URI

Una URI está compuesta por:

- El **esquema** (*scheme*), que es utilizado para informar el protocolo de resolución. Algunos esquemas requieren las dos barras. Algunos ejemplos de esquemas populares son `http:`, `ftp:` y `mailto:`.
- La autoridad, que se divide en tres partes como sigue:
 - Una **autenticación** opcional compuesta por un usuario (*user*), dos puntos y una contraseña (*password*) terminando en un arroba (`@`).
 - Un **anfitrión** (*host*) que puede ser un nombre registrado de una dirección IP. Representa la localización del recurso representado.
 - Un número de **puerto** opcional (*port*) separado del anfitrión por dos puntos.

- Una **ruta** (*path*) que es una secuencia de segmentos separados por una barra (/) que comienza con una de las mismas. Generalmente representa una localización en el sistema de archivos del anfitrión, pero no es necesario.
- Una **consulta** opcional (*query*) que comienza con un signo de interrogación (?) y, a pesar de no tener una sintaxis definida, usualmente presenta pares atributo-valor.
- Un **fragmento** opcional (*fragment*) que comienza con un numeral (#) seguido por un identificador secundario al recurso principal.

Podemos notar que una URL (*Uniform Resource Locator*) es una URI, pero no debemos confundir los términos pues una URL además de identificar un recurso Web permite obtener una representación del mismo generalmente en formato HTML via HTTP. Es decir, una URL es una URI que apunta a un recurso físico en la web, mientras que una URI no necesariamente debe apuntar a una localización que exista realmente.

En RDF se utilizan URIs para representar recursos y relaciones, generalmente se prefieren URLs para tenerla capacidad de ingresar en ellas (dereferenciar) y ver información relevante del recurso representado. Por lo regular las URIs utilizadas como recursos RDF utilizan el esquema HTTP, no incluyen autenticación, puerto ni consulta y definen sus recursos y vocabulario por medio de la ruta y el fragmento.

Por ejemplo, el proyecto WikiData⁷ organiza sus recursos y propiedades utilizando diferentes rutas. Podemos encontrar instancias como las siguientes:

`http://www.wikidata.org/entity/Q298`

`http://www.wikidata.org/prop/direct/P31`

Como vemos, se utiliza el esquema HTTP y el anfitrión es `www.wikidata.org` en ambos, pero los recursos se encuentran bajo `/entity/` mientras que las propiedades están en `/prop/`. En este caso particular, el recurso Q298 representa Chile y la propiedad P31 es *instance of* (“instancia de”).

⁷`www.wikidata.org`

2.4.2. XML

XML es el acrónimo del inglés *Extensible Markup Language*. Es un lenguaje de marcado desarrollado por el W3C para almacenar datos de manera legible tanto por personas como por máquinas. El diseño de XML busca la simplicidad, generalidad y usabilidad a través de internet [Bray, 2008].

XSD (del inglés *XML Schema Definition*) es una recomendación del W3C que especifica formalmente la estructura y restricciones de los contenidos de un fichero XML de manera precisa, más allá de las normas sintácticas del propio lenguaje XML. A diferencia de otros lenguajes para XML, XSD especifica también tipos de datos y sus restricciones, logrando un estándar a la hora de representar datos [23]. Presenta 19 tipos de datos básicos: `anyURI`, `base64Binary`, `boolean`, `date`, `dateTime`, `decimal`, `double`, `duration`, `float`, `hexBinary`, `gDay`, `gMonth`, `gMonthDay`, `gYear`, `gYearMonth`, `NOTATION`, `QName`, `string` y `time`. Además permite la creación de nuevos tipos de datos por medio de tres mecanismos:

- **Restricción:** Reduce los valores que puede tomar un dato.
- **Lista:** Permite una secuencia de valores.
- **Unión:** Permite la elección de valores de diferentes tipos.

Gracias a estos factores un fichero XML Schema puede representar un modelo de datos completo y robusto, con relaciones entre las entidades y asignaciones de tipos de datos básicos.

Estas características hacen de XSD el estándar para representar información literal⁸ en una base de datos RDF. Por ejemplo, números y cadenas de caracteres se pueden guardar usando XSD como sigue:

```
^^"2.9"<http://www.w3.org/2001/XMLSchema#decimal>  
"texto^^"<http://www.w3.org/2001/XMLSchema#string> "some text"@en
```

⁸Información que no es un recurso RDF (URI), incluye números, cadenas de caracteres, etc.

Como podemos ver, tanto el valor como el tipo de dato se almacenan en la misma cadena de caracteres. La información literal en una base de datos RDF se guarda con el valor entre comillas ("), luego se pueden utilizar `^^` para denotar el tipo de dato (generalmente definido por XSD), o un `@` para denotar el idioma de la cadena (`xml:lang`). Estos ejemplos en particular muestran que "2.9" es un decimal, "texto" es una cadena de caracteres y "some text" es una cadena de caracteres en inglés.

2.4.3. RDF

RDF (del inglés *Resource Description Framework*) es una familia de especificaciones del W3C diseñada como un modelo de datos para metadatos. Fue adoptado como una recomendación del W3C en 1999, mientras que la especificación 1.0 fue publicada el 2004 y la 1.1 el 2014 [Bikakis *et al.*, 2013].

El modelo de datos RDF se basa en la idea de hacer declaraciones sobre recursos Web (URIs) en forma de expresiones $\langle \textit{sujeto}, \textit{predicado}, \textit{objeto} \rangle$ que son llamados triples RDF. El *sujeto* indica el recurso mientras que el *predicado* denota la relación (también llamada propiedad) de este con el *objeto* . Así podemos decir que un triple RDF sigue la clásica notación **entidad - atributo - valor** de los modelos orientados a objetos, permitiendo gracias a su simpleza modelar todo tipo de conceptos abstractos.

Digamos U un conjunto de URIs, B un conjunto de recursos anónimos y L un conjunto de literales XSD, podemos denotar un triple RDF como:

$$\langle s, p, o \rangle \in (U \cup B) \times U \times (U \cup B \cup L)$$

Además, un conjunto de triples RDF sera representado naturalmente por un grafo dirigido⁹(con s , p como nodos y o como el arco que los une). Esta característica faculta la tecnologia para ser parte fundamental de la Web semántica pues permite relacionar información de diferentes fuentes sin mayor problema y representarla en un esquema fácilmente identificable.

Llamaremos vocabulario a la definición de conceptos y relaciones (propiedades) utilizados para describir y representar un área de conocimiento. Otro concepto a tener en

⁹específicamente por un *property graph* (grafo con propiedades)

cuenta son las ontologías, aunque no existe una clara división entre estas y los vocabularios, generalmente se les considera mas complejas y formales. En la Web Semántica una colección de triples RDF puede denotar un vocabulario o una ontología.

El vocabulario incluido en la especificación RDF es muy básico y por ello fue extendido con RDF Schema, por lo que la gran mayoría de las bases de datos RDF actuales contienen al menos ambos vocabularios.

RDF es un modelo abstracto con varios formatos de serialización, por lo que la codificación de un triple varía dependiendo del tipo de archivo en el que se guarde.

RDF/XML [Beckett y McBride, 2004] fue la primera codificación estándar para serializar RDF (en un archivo XML) y si bien es potente, es difícil de leer por las personas. Por esto utilizaremos Turtle [Beckett *et al.*, 2014] y N-Triples [Beckett, 2014], pues su simpleza los hace ideales para el entendimiento y procesamiento de los triples.

Algunas de las reglas para escribir un archivo Turtle (`.ttl`) son las siguientes:

1. Un triple es denotado por tres URIs separadas por espacio y terminadas en un punto.
2. Si una línea termina en punto y coma (;) la siguiente línea mantiene el *sujeto* y solo se deben escribir las URIs del *predicado* y el *objeto*.
3. Se pueden definir prefijos con `@prefix name: <URI>..` Esta característica es particularmente útil cuando trabajamos con vocabularios ya definidos.

N-Triples es una subsección del lenguaje Turtle y su característica principal es la facilidad que presenta a la hora de analizar su sintaxis. En un archivo N-Triples (`.nt`) solo se pueden escribir tres URIs seguidas de un punto para representar un triple RDF y todo lo que esté después de un numeral (#) se considera un comentario.

En las figuras siguientes podemos ver un ejemplo básico de la utilización de Turtle para la generación de triples (figura 2.3), el mismo ejemplo escrito en XML/RDF (figura 2.4) y como éstos pueden ser visualizados como un grafo dirigido (2.5).

Tanto 2.3 como 2.4 presentan la misma información, pero en formatos distintos. Turtle es un formato fácil de leer tanto por humanos como por computadoras mientras que

XML/RDF, si bien es más difícil de leer por los humanos, es el estándar para la transmisión de metadatos por Internet. Por su parte, la misma información es presentada en forma de grafo en la figura 2.5.

```
1 @prefix ex: <http://www.example.com/> .
2 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
3 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
4
5 ex:fundador rdf:type      rdf:Property .
6 ex:nombre   rdf:type      rdf:Property .
7 ex:utfsm    rdf:type      ex:universidad ;
8             ex:fundador   ex:federico ;
9             ex:pagina     <http://www.usm.cl/> .
10 ex:federico ex:nombre     "Federico Santa Maria"^^xsd:string .
```

Figura 2.3: Ejemplo en Turtle.

```
1 <?xml version="1.0" encoding="utf-8" ?>
2 <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3     xmlns:ex="http://www.example.com/">
4
5     <rdf:Property rdf:about="http://www.example.com/fundador" />
6     <rdf:Property rdf:about="http://www.example.com/nombre" />
7
8     <rdf:Description rdf:about="http://www.example.com:utfsm">
9         <rdf:type rdf:resource="http://www.example.com/universidad"/>
10        <ex:fundador>
11            <rdf:Description rdf:about="http://www.example.com/federico">
12                <ex:nombre rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
13                    Federico Santa Maria
14                </ex:nombre>
15            </rdf:Description>
16        </ex:fundador>
17        <ex:pagina rdf:resource="http://www.usm.cl/" />
18    </rdf:Description>
19
20 </rdf:RDF>
```

Figura 2.4: Mismo ejemplo en XML/RDF.

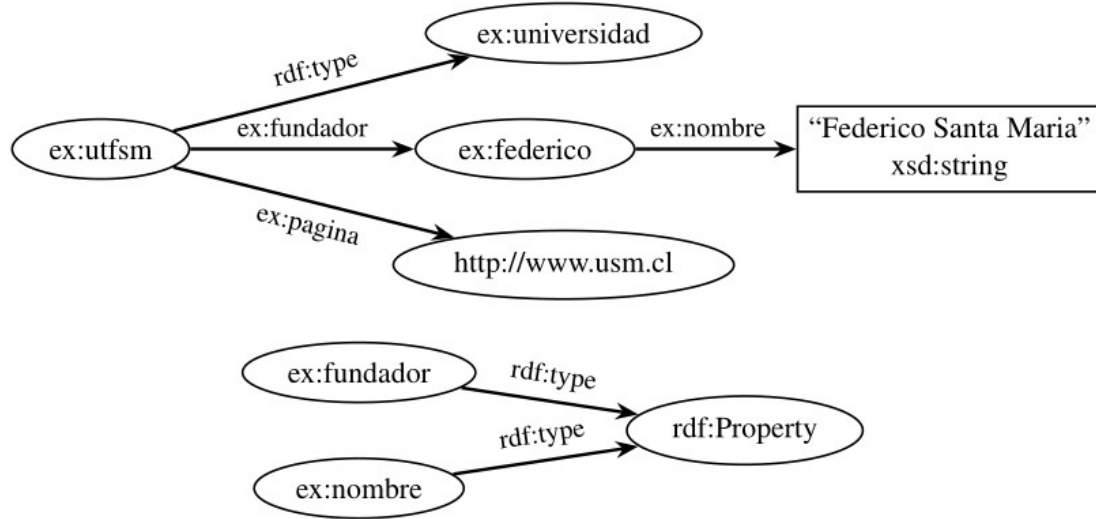


Figura 2.5: Grafo generado.

2.4.4. Vocabularios: RDFS y OWL

RDFS (de las siglas del inglés *Resource Description Framework Schema*) es un vocabulario que extiende RDF proveyendo un conjunto de clases y propiedades que mejoran la creación de modelos como son: **Class** para declarar clases, **subClassOf** para denotar herencia, **range** y **domain** para el rango y dominio de cierta propiedad (**rdf:Property**), entre otras. RDF fue presentado en 1998 e introducido finalmente como recomendación del W3C el 2004 [Bikakis *et al.*, 2013]. RDFS provee el vocabulario necesario para la construcción de cualquier ontología más avanzada y por ello es la base de otros vocabularios como OWL y SKOS. La especificación completa del vocabulario puede encontrarse en [Brickley *et al.*, 2014].

OWL (del inglés *Web Ontology Language*) es una familia de lenguajes para la creación de ontologías complejas. Agrega lógica computacional para que las relaciones hechas con este lenguaje puedan ser procesadas con el fin de verificar la consistencia de la información o generar información implícita.

La versión actual de OWL se conoce como “OWL 2” y fue publicada el 2009 como una revisión y extensión de la versión inicial publicada el 2004 [Bikakis *et al.*, 2013].

Generalmente cuando se habla de “OWL” nos referimos a la versión del 2004.

Algunas de las características mas utilizadas de RDFS y OWL en las bases de datos RDF actuales son:

- **rdfs:label**, utilizado para denotar la etiqueta (generalmente el nombre) de un recurso.
- **owl:ObjectProperty**, utilizado para denotar que una propiedad apuntará exclusivamente a recursos RDF definidos por URIs (otros objetos).
- **owl:DatatypeProperty**, utilizado para denotar que una propiedad apuntará exclusivamente a literales, es decir información (generalmente XSD) en forma de números, cadenas de caracteres, fechas, entre otros.

La figura 2.6 presenta una pequeña base de datos RDF que utiliza los vocabularios de RDFS y OWL. Especificamente se denotan, por medio de **rdfs:label** , las etiquetas “Federico Santa Maria” y “Universidad Técnica Federico Santa Maria” para los recursos **ex:federico** y **ex:ut fsm** respectivamente. Por otro lado se define la propiedad **ex:fundador** como **owl:ObjectProperty** , para siempre apunte a un objeto RDF denotado por una URI y **ex:nombre**(y **ex:apellido**) como **owl:DatatypeProperty** de manera que se utilicen para denotar información exclusivamente literal.

```
1 @prefix ex:    <http://www.example.com/> .
2 @prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
3 @prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
4 @prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
5 @prefix owl: <http://www.w3.org/2002/07/owl#> .
6
7 ex:fundador rdf:type    owl:ObjectProperty .
8 ex:nombre   rdf:type    owl:DatatypeProperty .
9 ex:apellido  rdf:type    owl:DatatypeProperty .
10 ex:ut fsm    rdf:type    ex:universidad ;
11             ex:fundador ex:federico ;
12             ex:pagina   <http://www.usm.cl/> .
13 ex:federico ex:nombre   "Federico"^^xsd:string .
14 ex:federico ex:apellido "Santa Maria"^^xsd:string .
15 ex:federico rdfs:label  "Federico Santa Maria"^^xsd:string .
16 ex:ut fsm    rdfs:label  "Universidad Tecnica Federico Santa Maria"@es .
17 ex:ut fsm    rdfs:label  "Federico Santa Maria Technical University"@en .
```

Figura 2.6: Ejemplo de base de datos RDF que utiliza RDFS y OWL

Actualización de la base de datos presentada en la figura 2.3 utilizando algunas características típicas de RDFS y OWL.

2.4.5. SPARQL

SPARQL (del inglés *SPARQL Protocol and RDF Query Language*) es un lenguaje estandarizado para consultar grafos RDF, se constituyó como una recomendación oficial por la W3C en el 2008 [Bikakis *et al.*, 2013]. La versión actual de SPARQL es la 1.1 [The W3C SPARQL Working Group, 2013].

SPARQL provee un set completo de operaciones analíticas para sus consultas definidas directamente en la especificación. Particularmente provee 4 formas de consultas:

- **SELECT:** Retorna valores en forma de tabla.
- **CONSTRUCT:** Retorna valores en forma de triples RDF.
- **ASK :** Retorna un resultado binario (**True/False**) indicando si la consulta tiene o no resultados.
- **DESCRIBE :** Retorna un grafo RDF con contenido que el administrador del *endpoint* SPARQL considere información útil.

A excepción de **DESCRIBE**, las demás consultas necesitan un bloque **WHERE** con sintaxis similar a Turtle en el cual se determinan las restricciones de la búsqueda en forma de triples RDF en los cuales cualquier elemento puede ser una variable (las cuales siempre comienzan con el carácter **?** seguido de un nombre).

SPARQL provee múltiples funciones como son las condicionales (**if**, **exists**, etc), de conversión (**str**, **lang**, etc), de comprobación (**isNumber**, **isBlank**, etc) y modificadores de respuesta como **ORDER BY**, **DISTINCT**, **REDUCED**, **LIMIT** y **OFFSET**. Una descripción completa del lenguaje puede encontrarse en [Prud'hommeaux, 2008] y en [The W3C SPARQL Working Group, 2013].

```
1  PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2
3  SELECT DISTINCT ?recurso ?etiqueta WHERE {
4    ?recurso rdfs:label ?etiqueta .
5    FILTER (LANG(?etiqueta) = "" || LANG(?etiqueta) = "es")
6  } LIMIT 10
```

Figura 2.7: Consulta para obtener recursos y sus etiquetas

Al ejecutar esta consulta en la base de datos de la figura 2.6 obtendremos los recursos que tienen etiquetas sin idioma o en español, y dichas etiquetas.

En la figura 2.7 vemos un ejemplo de una consulta SPARQL para obtener recursos con etiquetas definidas en la base de datos mostrada en la figura 2.6. La línea 3 de esta consulta indica, por medio de `SELECT DISTINCT`, que queremos obtener todos los valores no duplicados¹⁰ de `?recurso` y `?etiqueta` que cumplan con las restricciones especificadas en la sección `WHERE` (línea 4 y 5). En dicha sección se hacen dos aseveraciones: la primera, el triple:

`?recurso rdfs:label ?etiqueta`

que indica la búsqueda en toda la base de datos de triples que calcen con dicho triple, es decir recursos a los que llamaremos `?recurso` que tengan una relación `rdfs:label` con algo a lo que llamaremos `?etiqueta`. En este caso los triples que calzan con dicha restricción son las líneas 15, 16 y 17 de la figura 2.6.

La segunda aseveración de la sección `WHERE` de esta consulta es el filtro `FILTER (LANG(?etiqueta)="" || LANG(?etiqueta) = "es")` en el cual se le aplica el operador `LANG` a los resultados obtenidos en la variable `?etiqueta`. Dicho operador retorna el lenguaje de la cadena de caracteres en este caso se filtra el resultado de la línea 17 pues `LANG("Federico Santa Maria Technical University"@en)` retorna en y los resultados permitidos son `""` (nada) o `"es"`.

¹⁰Una base de datos RDF puede tener la misma información, es decir los mismos triples RDF, más de una vez. Esto ocurre muchas veces debido a la generación automática de información por lo que se aconseja utilizar la cláusula `DISTINCT` a no ser que se requiera lo contrario.

Por último la línea 6 de la consulta (de la figura 2.6) indica que a lo más queremos obtener 10 resultados por medio de `LIMIT 10`, así el retorno de esta consulta nos entregará la tabla 1.

Tabla 1: Resultados de la consulta ilustrada en la figura 2.7

<code>?recurso</code>	<code>?etiqueta</code>
<code>ex:federico</code>	<code>"Federico Santa Maria"^^xsd:string</code>
<code>ex:utfsm</code>	<code>"Universidad Tecnica Federico Santa Maria"@es</code>

La figura 2.8 muestra otro ejemplo de consulta SPARQL, en esta podemos notar como la propiedad también puede ser una variable. Al igual que el ejemplo anterior buscamos todos los resultados distintos que coincidan con la sección `WHERE`, en este caso dicha sección impone dos restricciones en forma de triples. El primero (línea 5) es:

`?recurso ?propiedad ?valor`

como podemos notar este triple está compuesto por tres variables, por lo que por sí solo no filtra nada (pues toda la base de datos calza con él) en particular este triple nos indica como llamaremos cada parte de un triple cualquiera.

```

1  PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2  PREFIX owl: <http://www.w3.org/2002/07/owl#>
3
4  SELECT DISTINCT ?recurso ?propiedad ?valor WHERE {
5    ?recurso ?propiedad ?valor .
6    ?propiedad rdf:type owl:DatatypeProperty .
7  } LIMIT 10

```

Figura 2.8: Consulta para obtener recursos, propiedades literales y su valor

Al ejecutar esta consulta en la base de datos de la figura 2.6 obtendremos los recursos que tienen propiedades definidas literales y el valor de asociado a las mismas

`?propiedad rdf:type owl:DatatypeProperty`

en el indicamos que las propiedades deben ser de tipo `owl:DatatypeProperty`, es decir deben ser propiedades literales. Podemos notar que los datos que calzan con dicha restricción son las líneas 8 y 9 de la figura 2.6, en particular las propiedades `ex:nombre` y `ex:apellido`, así los resultados de esta consulta estarán denotados en la tabla 2.

Tabla 2: Resultados de la consulta ilustrada en la figura 2.8

?recurso	?propiedad	?valor
ex:federico	ex:nombre	"Federico"^^xsd:string
ex:federico	ex:apellido	"Santa Maria"^^xsd:string

2.4.6. Wikimedia

Wikimedia¹¹ es una fundación sin fines de lucro que alberga Wikipedia y otros proyectos relacionados con el conocimiento libre, bajo la premisa de "Queremos que sea más fácil para todos compartir lo que saben"; esto se ve reflejado en la cantidad de herramientas que han desarrollado para este fin.

2.4.7. DBpedia

DBpedia [Auer *et al.*, 2007] es un esfuerzo de la comunidad de origen público para extraer contenido estructurado de la información creada en varios proyectos de Wikimedia. Esta información estructurada se asemeja a un gráfico de conocimiento abierto (OKG) que está disponible para todos en la Web. Un gráfico de conocimiento es un tipo especial de base de datos que almacena el conocimiento en una forma legible por máquina y proporciona un medio para recopilar, organizar, compartir, buscar y utilizar la información. Google utiliza un enfoque similar para crear esas tarjetas de conocimiento durante la búsqueda.

Los datos de DBpedia se sirven como datos vinculados, lo que está revolucionando la forma en que las aplicaciones interactúan con la Web. Se puede navegar por esta Web

¹¹Fundacion wikimedia: https://meta.wikimedia.org/wiki/Wikimedia_Foundation

de hechos con navegadores web estándar, rastreadores automáticos o realizar consultas complejas con lenguajes de consulta similares a SQL (por ejemplo, SPARQL).

2.5. Integración de datos

Integración de datos¹² es el proceso que permite combinar datos heterogéneos de muchas fuentes diferentes en la forma y estructura de una única aplicación. Esto facilita que diferentes tipos de información, tales como matrices de datos, documentos y tablas, sean fusionados por usuarios, organizaciones y aplicaciones para un uso personal, de procesos de negocio o de funciones. La integración soporta el procesamiento analítico de grandes conjuntos de datos alineando, combinando y presentando cada conjunto de informaciones de departamentos organizacionales y fuentes de datos remotas y externas, para cumplir con los objetivos del integrador. La integración de datos se implementa generalmente en un data warehouse mediante software especializado que aloja grandes almacenes de datos de recursos internos y externos. Los datos se extraen, se mezclan y se presentan de forma unificada. Por ejemplo y en este caso, usar los datos de ChileCompra para obtener información de los proveedores, las compras y operaciones de una municipalidad para obtener un informe detallado de las conexiones de dichos proveedores, cuanto se está gastando etc.

2.6. FAIR Principles

2.6.1. Principios

FAIR Principles o los principios FAIR se diseñaron pensando en la ciencia abierta basada en datos y asistida por máquinas. El objetivo final de seguir los principios FAIR es que tanto las máquinas como las personas puedan encontrar, acceder, interoperar y, por tanto, reutilizar los objetos de investigación de los demás. En el documento original de FAIR de Wilkinson [Wilkinson *et al.*, 2016] se formularon los principios, pero (conscientemente) no se incluyeron detalles sobre las opciones de implementación.

¹²integración de datos:<https://www.powerdata.es/integracion-de-datos>

estos principios son:

- Findable (Que se pueda encontrar): El primer paso para que los datos se puedan reutilizar es poder encontrarlos. Los datos y los metadatos deben ser fáciles de encontrar tanto para humanos como para máquinas.
- Accessible (Accesible): Al encontrar los datos requeridos se debe saber cómo acceder a ellos de forma expedita ya sea con autorizaciones o autenticaciones.
- Interoperable (Interoperable): Los datos suelen tener que integrarse con otros datos. Además, los datos deben interoperar con aplicaciones o flujos de trabajo para su análisis, almacenamiento y procesamiento.
- Reusable (Reutilizable): El objetivo final de FAIR es optimizar la reutilización de los datos. Para lograrlo, los metadatos y los datos deben estar bien descritos para que puedan ser replicados y/o combinados en diferentes entornos.

2.6.2. FAIRificación

Los Principios de Datos FAIR se aplican a los metadatos, los datos y la infraestructura de apoyo. La mayor parte de los requisitos de localización y accesibilidad pueden lograrse a nivel de metadatos. La interoperabilidad y la reutilización requieren más esfuerzos a nivel de los datos. El proceso de FAIRificación¹³ es una lista de pasos a seguir para lograr que un dominio de datos siga los principios FAIR. Este proceso consta de:

- Recuperar los datos que no son FAIR.
- Analizar los datos.
- Definir modelo semántico.
- Hacer que los datos sean vinculables.
- Asignar licencia.

¹³<https://www.go-fair.org/fair-principles/fairification-process/>

- Definir los metadatos del conjunto de datos.
- Publicar los datos FAIRificados.

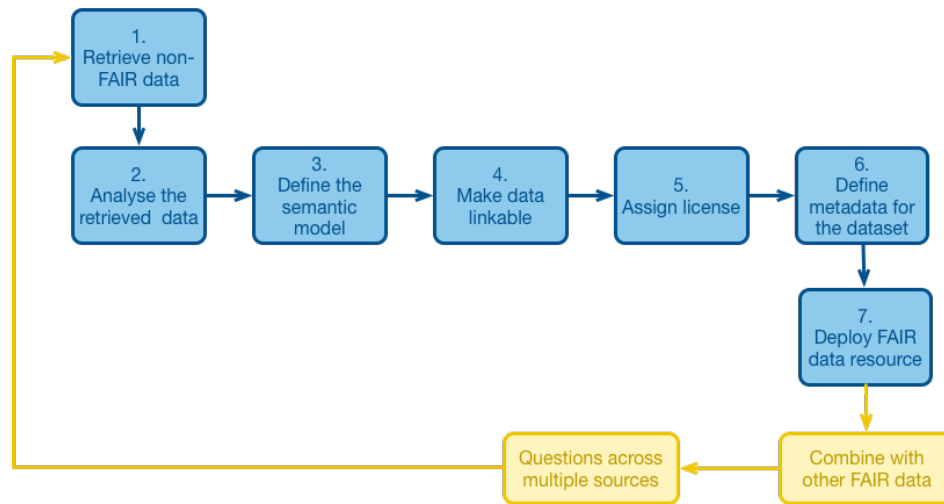


Figura 2.9: Diagrama del proceso de FAIRificación

2.7. METHONTOLOGY

Metodología para la confección de ontologías, desarrollado por la Facultad de informática de la universidad Politécnica de Madrid [Corcho *et al.*, 2005], permite construir ontologías en el nivel de conocimientos, y tiene sus raíces en las actividades identificadas por el proceso de desarrollo de software propuesto por la organización IEEE¹⁴ y en otras metodologías de ingeniería de conocimientos. Proporciona guías sobre cómo llevar a cabo el desarrollo de la ontología a través de las actividades de especificación, conceptualización, formalización, implementación y mantenimiento. A continuación se describe brevemente en qué consiste cada una de estas actividades:

- La actividad de especificación permite determinar por qué se construye la ontología, cuál será su uso, y quiénes serán sus usuarios finales.
- La actividad de conceptualización se encarga de organizar y convertir una percepción informal del dominio en una especificación semi-formal, para lo cual utiliza

¹⁴<https://www.ieee.org/>

un conjunto de representaciones intermedias (RRII), basadas en notaciones tabulares y gráficas, que pueden ser fácilmente comprendidas por los expertos de dominio y los desarrolladores de ontologías. El resultado de esta actividad es el modelo conceptual de la ontología.

- La actividad de formalización se encarga de la transformación de dicho modelo conceptual en un modelo formal o semi computable.
- La actividad de implementación construye modelos computables en un lenguaje de ontologías (Ontolingua, RDF Schema, OWL, etc.). La mayor parte de las herramientas de ontologías permiten llevar a cabo esta actividad de manera automática. Por ejemplo, Protégé 2.8.
- La actividad de mantenimiento se encarga de la actualización y/o corrección de la ontología, en caso necesario.

2.8. Protégé

Protégé es un plataforma gratuita de código abierto desarrollada por la Universidad de Stanford ¹⁵. La arquitectura de plug-in de Protégé puede adaptarse para construir aplicaciones basadas en ontologías, tanto simples como complejas. Los desarrolladores pueden integrar el resultado de Protégé con sistemas de reglas u otros solucionadores de problemas para construir una amplia gama de sistemas inteligentes. Cuenta con el apoyo activo de una sólida comunidad de usuarios y desarrolladores que responden a preguntas, escriben documentación y contribuyen con plug-ins, es totalmente compatible con las últimas especificaciones OWL 2 Web Ontology Language y RDF del World Wide Web Consortium, está basado en Java, es extensible y proporciona un entorno plug-and-play que lo convierte en una base flexible para la creación rápida de prototipos y el desarrollo de aplicaciones.

¹⁵Protégé: <https://protege.stanford.edu/>

2.9. Tarql

Tarql es una herramienta de línea de comandos hecha en Java diseñada para convertir archivos .csv en triples RDF usando SPARQL[2.4.5] el contenido del archivo CSV se introduce en la consulta como una tabla enlazada. Esto permite manipular los datos CSV utilizando toda la potencia de la sintaxis de SPARQL 1.1, y en particular la generación de RDF mediante consultas CONSTRUCT, la documentación y ejemplos de uso se pueden encontrar en la página web del proyecto <https://tarql.github.io/> en el código 1 se puede ver como se realiza el mapeo de la tabla 3.

```
1 PREFIX ex: <http://ex.org/a#>
2 PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
3
4 CONSTRUCT {
5   ?URI a ex:Organization;
6       ex:permalink ?permalink;
7       ex:name ?company;
8       ex:employees ?numEmployees;
9       ex:category ?category;
10      ex:city ?city;
11      ex:state ?state;
12      ex:foundationDate ?fundedDate;
13      ex:raisedAmt ?amount;
14      ex:raisedCurrency ?raisedCurrency;
15      ex:round ?round;
16 }
17 FROM <file:TechCrunchcontinentalUSA.csv>
18 WHERE {
19   BIND (URI(CONCAT('http://ex.org/companies/', ?permalink)) AS ?URI)
20   BIND (xsd:integer(?numEmps) AS ?numEmployees)
21   BIND (xsd:decimal(?raisedAmt) AS ?amount)
22 }
```

Código 1: Mapeo de la tabla 3

Tabla 3: Las siguiente tabla forma parte de un archivo .CSV que contiene información sobre empresas.

permalink	company	numEmps	category	city	state	fundedDate	raisedAmt	raisedCurrency	round
lifelock	LifeLock		web	Tempe	AZ	1-May-07	6850000	USD	http://example.com/b
mycityfaces	MyCityFaces	7	web	Scottsdale	AZ	1-Jan-08	50000	USD	seed
flypaper	Flypaper		web	Phoenix	AZ	1-Feb-08	3000000	USD	a

2.10. TripleStore

Un **Triplestore** o almacén de triples es una base de datos especialmente diseñada para el almacenamiento y recuperación de triples a través de consultas semánticas [Rusher,] como las mencionadas en la sección 2.4. Al ser una base de datos de grafos, los triplestores almacenan los datos como una red de objetos con enlaces materializados entre ellos. Esto hace que los triplestores RDF sean la opción preferida para gestionar datos altamente interconectados. Los triplestores son más flexibles y menos costosos que una base de datos relacional, por ejemplo.

La base de datos RDF, a menudo llamada base de datos de grafos semánticos, también es capaz de manejar potentes consultas semánticas y de utilizar la inferencia para descubrir nueva información a partir de las relaciones existentes.

Algunos motores de almacenamiento de triples son AnzoGraph¹⁶, AllegroGraph¹⁷ y JenaTDB¹⁸.

2.10.1. Apache Jena

Apache Jena es un Framework de código abierto de Java específicamente para construir aplicaciones de web semantica y Linked data. El framework es compuesto por distintas APIs que interactúan entre sí para procesar datos RDF. entre estas APIs. Un vistazo a la arquitectura de este Framework se podrá ver en la figura 2.10.

¹⁶https://www.cambridgesemantics.com/anzograph/?utm_campaign

¹⁷<https://allegrograph.com/>

¹⁸<http://jena.apache.org/documentation/tdb/>

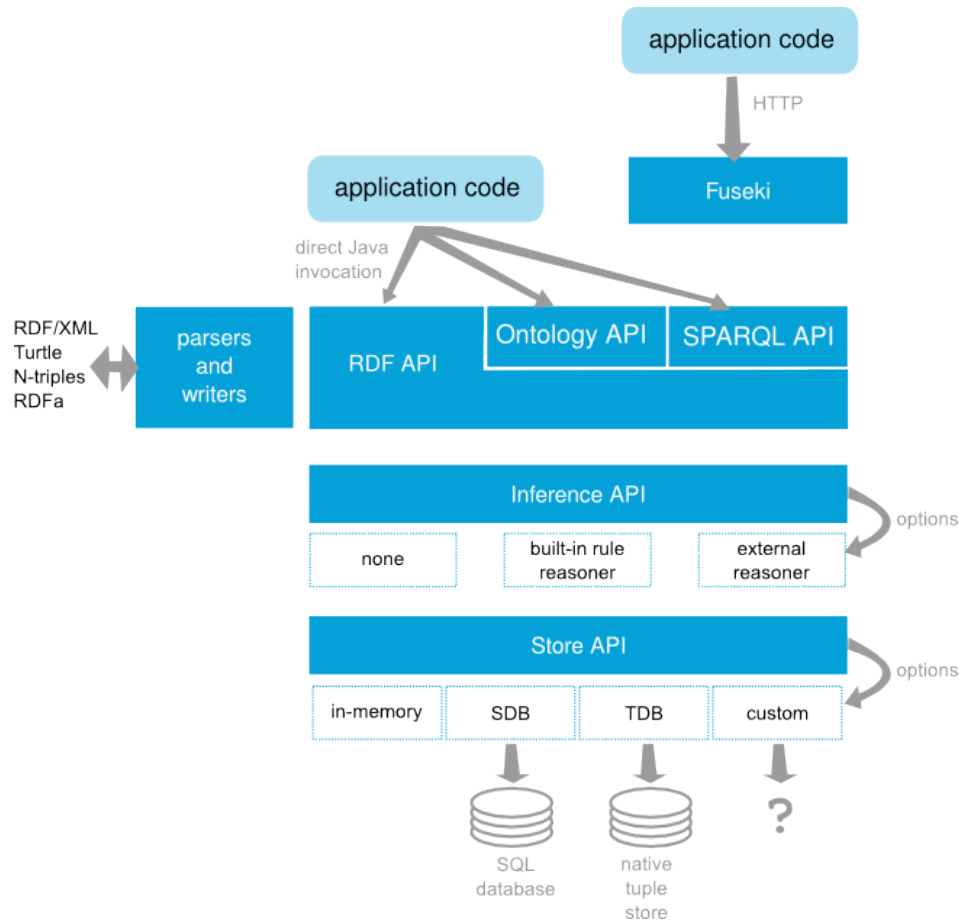


Figura 2.10: Arquitectura Apache Jena

2.11. Antecedentes

En la sección 1.1 se habló sobre cómo distintas áreas han trabajado para obtener una integración de datos para adaptarse a esta época de los datos, como ejemplo de esto, está el caso del museo Smithsonian desarrollado por Pedro Szekely [Szekely *et al.*, 2013] donde se describe el paso a paso para lograr hacer un mapeo de los datos del museo que incluye la selección de las ontologías de dominio, el mapeo de las tablas de la base de datos en RDF, la vinculación de los datos con otras fuentes relacionadas y la curación de los datos resultantes para garantizar datos de alta calidad, esto se ve reflejado en la figura 2.11.

adicional en sus registros.

Por otro lado, se puede aplicar esta tecnología para combatir contra el crimen, como también realiza Pedro Szekely en [Szekely *et al.*, 2015] donde se describe el uso de DIG para generar un grafo de conocimiento para combatir la trata de personas. DIG¹⁹(Domain-specific Insight Graphs) es un sistema de indexación, búsqueda y análisis específico del dominio. El sistema DIG aprovecha el software de código abierto de vanguardia combinado con una arquitectura abierta y un conjunto flexible de **API** para facilitar la integración de una variedad de herramientas de extracción y análisis. La aplicación de DIG para combatir la trata de personas está en uso hoy en día. El sistema se implementó en ese tiempo en seis agencias de aplicación de la ley y varias ONG que están utilizando el sistema de varias maneras para combatir la trata de personas, como localizar víctimas o investigar organizaciones que se dedican a la trata de personas. En la aplicación implementada a partir de julio de 2015, había 60 millones de anuncios con aproximadamente 162,000 anuncios nuevos por día (con nuevos anuncios actualizados cada hora). El número de objetos (sujetos RDF) era de 1.4 mil millones y el número de objetos de características es de 222 millones.

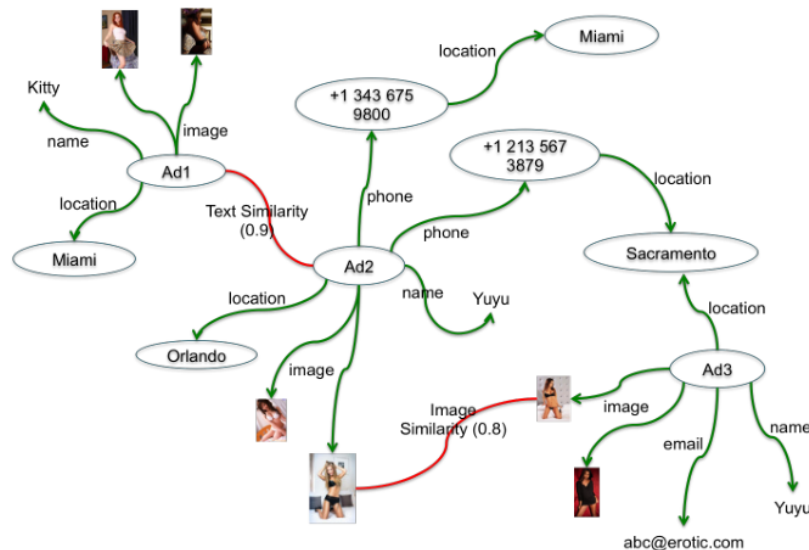


Figura 2.12: Ejemplo de grafo de conocimiento para el caso de trata de personas.

¹⁹Domain-specific Insight Graphs: <http://usc-isi-i2.github.io/dig/>

Otro caso también digno de mencionar es el de Marco Brandizi en el *Journal of Integrative Bioinformatics* [Brandizi et al., 2018] donde usaron un sistema híbrido de datos enlazados y bases de datos de grafos para crear una red de conocimiento biológico que siguiera los estándares de los FAIR Principles, esto específicamente centrado en el área de la ciencia de la vida. Para este fin crearon una Ontología ligera y general, que permitió modelar una gran variedad de entidades de las ciencias de la vida de forma sencilla, esta ontología es llamada *Bio-Knowledge Network Ontology* o BioKNO²⁰, con ella y un software llamado KnetMiner crearon una arquitectura que obtenía la información, trataba los datos con distintas herramientas como Tarql²¹ para asegurar seguir los principios FAIR. Dicha ontología y arquitectura se pueden observar en las Figuras 2.13 y 2.14 respectivamente.

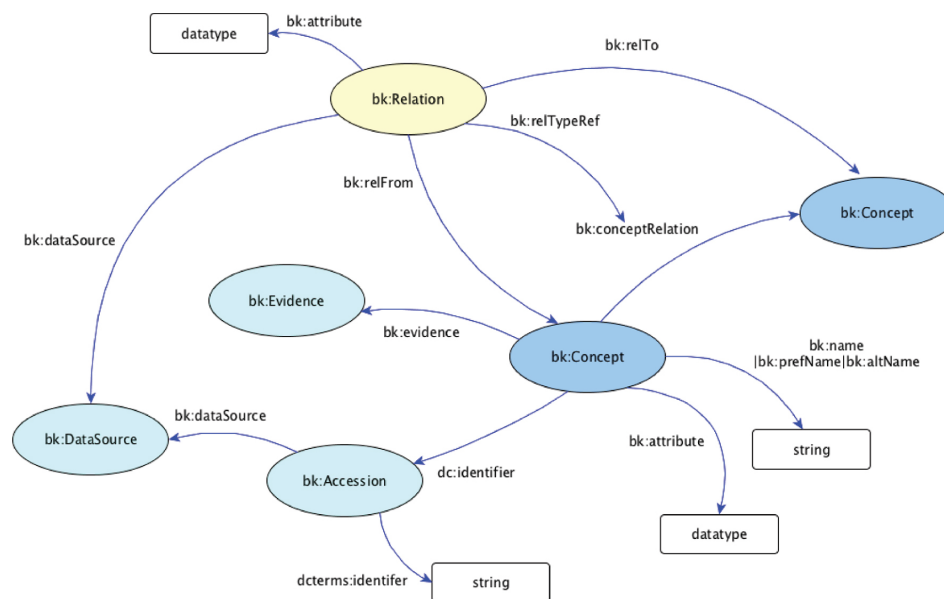


Figura 2.13: La organización de alto nivel de la ontología BioKNO.

Las entidades biológicas específicas se definen como subclases de `bk:Concept` y los tipos de relación (es decir, las propiedades de los objetos OWL) entre ellas se basan en subpropiedades de `bk:conceptRelation`. `bk:Relation` puede utilizarse para modelar

²⁰BioKNO, The Biological Knowledge Network Ontology: <https://github.com/Rothamsted/bioknet-onto>

²¹Tarql: <https://github.com/tarql/tarql>

relaciones reificadas. Tanto los conceptos como las relaciones reificadas pueden tener bk:attribute y otros elementos adjuntos.

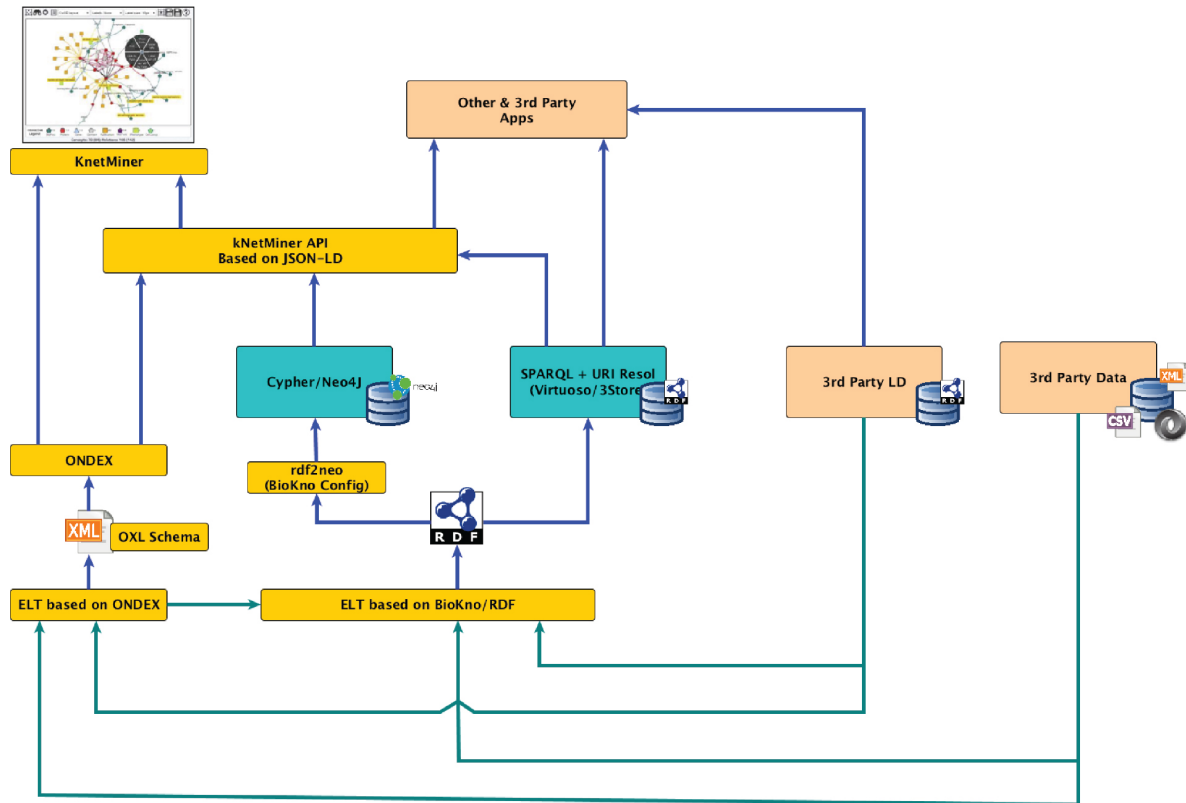


Figura 2.14: La nueva arquitectura diseñada para el ecosistema KnetMiner.

El modelado basado en BioKNO permite tanto la adquisición de datos (extracción, carga, transformación o ELT) como la consulta de datos de sus aplicaciones y las de terceros. El RDF sirve para la publicación de datos abiertos y la integración con otros conjuntos de datos.

CAPÍTULO 3

PROPUESTA DE SOLUCIÓN

La propuesta de solución será en resumidas cuentas seguir un proceso de FAIRificación como se menciona en la sección 2.6.2 para los datos de la administración pública, con el fin de conseguir un dominio que tenga datos y metadatos que permite encontrar la información, sea accesible, pueda ser operado por distintas aplicaciones o entidades y sea reutilizable.

3.1. Recopilación de Recursos

En base al problema planteado en la sección 1.1 se procede a realizar distintas etapas para el desarrollo de la solución a proponer, lo primero es Investigar el portal de transparencia y ver que datos se pueden obtener y hacer un análisis de los recursos a mano, para esto se accederá a la web del consejo de transparencia y se elegirá un organismo municipal para tomar de ejemplo, para este caso en específico se elegirá la municipalidad de Valparaíso presente en la figura 3.15



INTEGRANDO DATOS DE GASTOS DE LA ADMINISTRACIÓN PÚBLICA CHILENA PARA MEJORAR LA TRANSPARENCIA A NIVEL DE MUNICIPIO



Figura 3.15: Portal de Transparencia Municipalidad de Valparaíso

La figura 3.15, obtenida desde portalTransparencia mostrara el desglose en categorías de la información perteneciente a la municipalidad seleccionada, así en cada entidad del estado, se podrá observar que muchas categorías presentan enlaces externo representado por su correspondiente icono, estas son plataformas distintas desde donde se buscara cruzar datos e información con los del organismo.

Uno de los enlaces externos mas importante para este proyecto sera el *Sistema de compras Públicas*, que llevara hacia la web del mercado publico²² [2.3.1] como muestra la figura3.16, plataforma donde los organismos chilenos licitan y compran a distintos proveedores, Si se busca de donde vienen y van los datos de esta web, se podrá obtener esta pagina esta conectada directamente con otro portal que es **Chilecompra** [2.3]

²²<https://www.mercadopublico.cl/Home>



Figura 3.16: Pagina de inicio Mercado Publico.

como se observara en la figura 3.16.

Chilecompra tiene un apartado interno de datos abiertos, figura 3.17 donde aparece un apartado de descarga masiva, en este se encontraría el método para obtener la información de las licitaciones y ordenes de compra del país separado por mes, esto sera beneficioso para el proyecto ya que en las listas de licitaciones aparecerá gran cantidad de datos que serán usados en el proceso de desarrollo de la ontología en la sección 3.2.



Figura 3.17: Interfaz de datos abiertos de chilecompra

Al entrar al apartado de *Descarga masiva* uno se enfrentara a la siguiente interfaz mostrada en la figura 3.18.

ChileCompra Datos Abiertos

Compras históricas del Estado | Compras del Estado durante el año

Selección Descargas /

Descargas masivas

< Volver a Selección Descargas

Con el objetivo de disponer datos históricos de **órdenes de compra (OC)** y **licitaciones (Lic)** para todos los usuarios, se han habilitado links de descarga de registros históricos, los cuales agrupan los datos por mes.

El alcance de los archivos descargables abarca desde el año 2007 a la actualidad con un día de desfase con respecto a los datos en la base de datos productiva. Esta actualización se realiza todos los días entre las 12:00 a 14:00 horas.

Para la facilidad de los usuarios, los descargables corresponden a tablas en formato *.csv* con menos de 1 millón de registros, para que puedan ser consumidas mediante los softwares comúnmente utilizados por los usuarios (Excel, OpenOffice, SpreadSheet, LibreOffice, Office, entre otros).

La cantidad de información mensual puede variar según sea el caso. Los archivos comprimidos tienen un tamaño entre 30 MB y 150 MB. Esta condición, y el flujo de datos de la banda ancha utilizada, determinará el tiempo de descarga

Para descargar directamente a través de una URL:

Órdenes de Compra: <https://transparenciachc.blob.core.windows.net/oc-da/año-mes.zip>

Licitaciones: <https://transparenciachc.blob.core.windows.net/lic-da/año-mes.zip>

Como ejemplo, si el usuario desea descargar las órdenes de compra de marzo del 2019, solo debe reemplazar el año y el mes en el link *OC*.

Esto sería: <https://transparenciachc.blob.core.windows.net/oc-da/2019-3.zip>

Para una descripción más detallada de lo anterior, revisar el siguiente link: [Instructivo de Descarga \(PDF\)](#)

Reporte de Órdenes de Compra

Figura 3.18: Interfaz de descarga masiva de chilecompra

Dicha web, entrega ciertas instrucciones de descarga para ser operado por software de descargas u otros medio además de la descarga directa mas abajo.

Por otro lado en la plataforma de portal de transparencia aun se puede obtener más información pero la categoría que esta directamente relacionada con las licitaciones será la de **Personal y remuneraciones** lo que otorgaría la capacidad de obtener la información del personal del organismo. Al obtener que información se usara se podrá elaborar un vocabulario haciendo uso de una metodología de trabajo especial en la sección 3.2.

04. Personal y remuneraciones

- > Personal a Contrata
- > Personal de Planta
- > Personal sujeto al Código de Trabajo
- > Personas naturales contratadas a honorarios
- > Escala de Remuneraciones

Figura 3.19: Acercamiento a sección Personal y remuneraciones del organismo

3.2. Desarrollo de vocabulario

Este vocabulario sera usado para definir luego el modelo semántico de los datos, a la hora de crear un vocabulario no se suele partir de cero, sino que se basa en ontologías y vocabularios preexistentes, lo que favorece la comunicación entre personas, agentes inteligentes y sistemas. En este caso se buscara usar de base ontologías tales como *The Organization Ontology*, *GoodRelations* y *Public Contracts Ontology*.

The Organization Ontology²³ es una ontología base para la mayoría de vocabularios de estructuras organizativas, permite respaldar los datos de una gran cantidad de dominios con esta estructura. Está diseñada para permitir extensiones específicas de dominio para añadir la clasificación de las organizaciones y los roles, así como extensiones para apoyar la información vecina, como las actividades de la organización.

GoodRelations²⁴ Es un vocabulario estandarizado para datos de productos, precios

²³org: <https://www.w3.org/TR/vocab-org/>

²⁴gr: <http://www.heppnetz.de/ontologies/goodrelations/v1>

y empresas, destinado a mejorar la visibilidad, la búsqueda de información y ser usado en aplicaciones novedosas para las áreas implicadas. *Public Contracts Ontology*²⁵ surge como una alternativa para expresar la data sobre contratos públicos basada en vocabularios disponibles como *GoodRelations*. La existencia de este ultimo y su ausencia de componentes al español motivo al desarrollo del vocabulario que se espera crear.

Para la confección de este vocabulario se usara la metodología *METHONTOLOGY* descrita en [Corcho *et al.*, 2005], como se menciona en la sección 2.7 esta metodología consta de 4 actividades, la primera, la *especificación* ya se discutió al principio de la sección 3.1, por otro lado la tarea de *conceptualización* consta de once tareas como muestra la figura 3.20 .

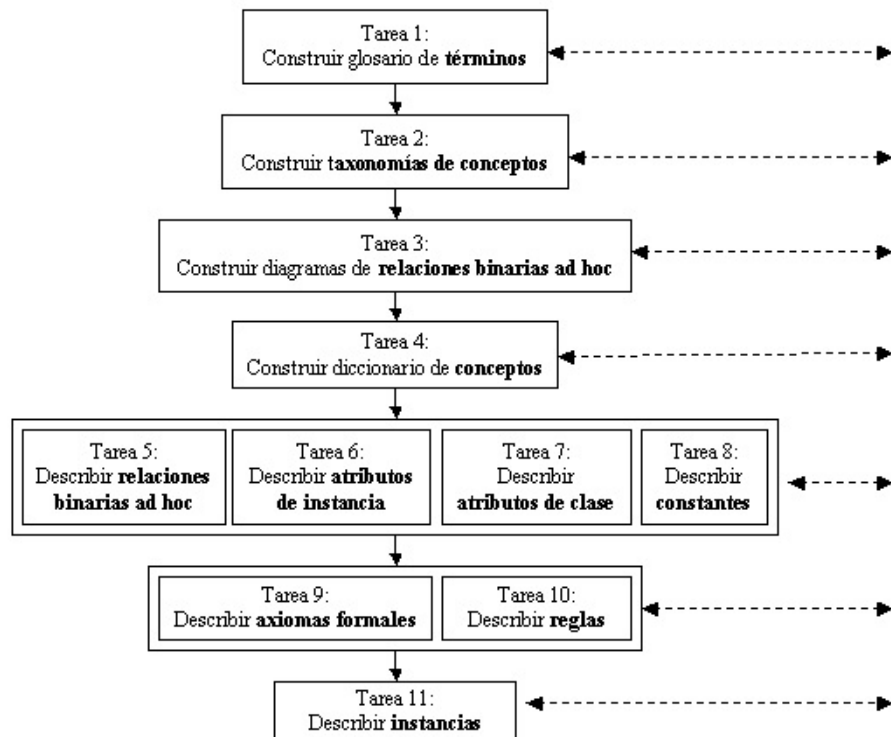


Figura 3.20: Tareas de conceptualización de METHONTOLOGY.

Primero se debe definir cuales son los actores involucrados dentro de los datos, en el ejemplo de la sección 3.1 se ve que se obtendrán plantillas .csv con la información tabulada, con esto se puede identificar los términos y construir un glosario de termino

²⁵pc: <https://github.com/opendatacz/public-contracts-ontology>

en base a ellos. Las relaciones entre entidades, estas se podrán inferir de las tablas por ejemplo la licitación *A* fue licitado por el organismo *MunicipioA*, información que eventualmente aparece en las tablas. Los atributos de dichas entidades, revisando las columnas de las tablas por la lógica y sus variaciones, con documentación sobre que significa cada columna, esta información se podrá obtener en documentación sobre las licitaciones en la web de Chilecompra en el apartado de *Sobre Datos Abiertos*²⁶, ahí se llevara al usuario hacia la API de Mercado Publico, donde esta el manual de licitaciones y ordenes de compra ²⁷, con la especificación y tipo de datos de la mayoría de Datos de las tablas. Esta información se cruza con las tablas ya que la descarga masiva es una generalización del uso de la API pues recopila distintas consultas de licitaciones que hace el sistema, por lo que la documentación de la API se puede tomar como referencia a los datos de las tablas. Una vez realizadas estas tareas, se llevara a cabo la *formalización* de la ontología para luego implementarla usando el software Protégé mencionado en la sección 2.8.

El uso de Protégé para el desarrollo de ontologías sera lo siguiente a realizar, esto se podra efectuar tanto por la interfaz web de la plataforma webProtege como por la versión de escritorio disponible para descargar en su web como se introdujo en la sección 2.8. Protégé otorga una gran cantidad de herramientas para trabajar por su arquitectura de Plug-in, es bastante simple de usar y de personalizar, si es necesario y resulta útil, se pueden importar las ontologías que se usaran de base, así se podrá tener acceso a las clases y propiedades necesarias para el desarrollo de un nuevo vocabulario. En la pestaña de **Entities**, se trabajara en el desarrollo de las clases a partir de la conceptualización y formalización anterior, junto con las *ObjectProperty* y las *dataProperty* ya mencionadas en la sección 2.4.4. Se deberá tener especial cuidado a la hora de crear estas propiedades para asegurar bien los Dominios y Rangos correspondientes.

Con el vocabulario terminado se exportara en formato Turtle (**.ttl**) este formato sera usado luego en la sección 3.4, por ahora se deberá continuar con la transformación e integración de datos usando **SPARQL**.

²⁶<http://datosabiertos.chilecompra.cl/Home/SobreDatosAbiertos>

²⁷<https://api.mercadopublico.cl/modules/DocuApoyo.aspx>

3.3. SPARQL como lenguaje de integración y transformación de datos

En la sección 3.1 se discutió sobre como y que datos se obtendrán de los portales Chilecompra y Transparencia, esta información se obtendrá como tablas .csv, se utilizaran transformaciones RDF basadas en consultas SPARQL CONSTRUCT[2.4.5] y conversiones CSV basadas en la herramienta TARQL explicado en la sección 2.9, que, de nuevo, explota SPARQL para mapear estructuras de tablas a RDF. Estas transformaciones suelen ser fáciles de escribir y sólo requieren conocimientos básicos de SPARQL.

Las clases deberán ser separadas por puntos dentro de la consulta, agregando en ellas todas las propiedades que debe contener a la hora de formar las URI como muestra el código 2 tomando como base un ejemplo de consulta CONSTRUCT con la clase Proveedores.

```
1 PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
2 PREFIX mpCL: <http://www.semanticweb.org/mercadoPublicoCL#>
3
4 CONSTRUCT {
5   ?PRO a mpCL:Proveedor;
6       mpCL:codigoProveedor ?CodigoProveedor;
7       mpCL:codigoSucursalProveedor ?CodigoSucursalProveedor;
8       mpCL:rutProveedor ?RutProveedor;
9       mpCL:nombreProveedor ?NombreProveedor;
10      mpCL:descripcionProveedor ?DescripcionProveedor;
11      mpCL:razonSocialProveedor ?RazonSocialProveedor;
12      mpCL:ofrece ?OFE;
13      mpCL:tieneRubro ?R1, ?R2, ?R3;
14      .
15 }
```

Código 2: Ejemplo proveedores

Para generar las URI se necesita usar la clausula **WHERE** donde se pondrán las restricciones para la consulta, estas restricciones en este caso formaran los triples como

corresponde según se redacten, para seguir con el ejemplo de el código 2 se continuara con la muestra de una clausula **WHERE** para esta clase en el código 3.

URI

```
1 PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
2 PREFIX mpCL: <http://www.semanticweb.org/mercadoPublicoCL#>
3
4 ...
5 WHERE{
6     BIND("http://data.semanticmp.cl/resources/" AS ?baseURI)
7     BIND (URI(CONCAT(?baseURI,'proveedor/',?CodigoProveedor)) AS ?PRO)
8     BIND(URI(CONCAT(?baseURI, 'rubro/',MD5(LCASE(?Rubro1)))) AS ?R1)
9     BIND(URI(CONCAT(?baseURI, 'rubro/',MD5(LCASE(?Rubro2)))) AS ?R2)
10    BIND(URI(CONCAT(?baseURI, 'rubro/',MD5(LCASE(?Rubro3)))) AS ?R3)
11    ...
12 }
```

Código 3: Ejemplo proveedores

La clausula **BIND** permite asignar el contenido de una expresión a una variable, esto sera usado para obtener desde los **.csv** las variables de las columnas **?CodigoProveedor**, **?Rubro1**, **?Rubro2**, **?Rubro3**, transformarlas en **URI** y luego entregarlas para armar los triples con la clausula **CONSTRUCT** mencionada antes. Este punto es crucial para este proyecto pues en la generación de **URI** con las clausulas **BIND** se puede asegurar el cumplimiento de el principio de encontrabilidad de los Fair Principles, ya que se puede asegurar un identificador único para el recurso en la web usando información de las mismas tablas como lo es el **codigoProveedor**.

Otra tarea que recae en usar la Clausula **WHERE** es ir corrigiendo y modificando los datos que pueden estar con formatos distintos a los que uno esperaría recibir en una consulta en las tablas, para esto se debe usar la documentación pertinente de las tablas y las Clausulas **BIND** para corregir esas entradas de datos para luego agregarlas al modelo del **CONSTRUCT** como muestra el código 4

```
1 PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
2 PREFIX mpCL: <http://www.semanticweb.org/mercadoPublicoCL#>
3
4 ...
5 WHERE{
6     ...
7     BIND (IF(?FechaInicio != "1900-01-01", ?FechaInicio, "NA") AS ?
FechaInicio_Fix)
8     BIND(COALESCE(
9         IF(CONTAINS(STR(?Modalidad),"30_DAYS"),"Pago a 30 dias",1/0),
10        IF(CONTAINS(STR(?Modalidad),"45_DAYS"),"Pago a 45 dias",1/0),
11        "Otra modalidad de pago"
12    ) AS ?Modalidad_fix
13 )
14
15     BIND(IF(?VisibilidadMonto=1, "si","no") as ?VisibilidadMonto_fix)
16
17     BIND(IF(?TipoConvocatoria=1, "Abierta","Cerrada") as ?
TipoConvocatoria_fix)
18     ...
19 }
```

Código 4: Ejemplo proveedores corrigiendo datos

Así ya se tendría listo el mapeo correspondiente, este tendrá que hacerse para cada archivo .csv que se desee agregar. Al tenerlo listo se podrá correr Tarql con los parámetros correspondientes el archivo con el las consultas SPARQL y la tabla a trabajar.

```
tarql mapa.sparql tabla.csv
```

Este comando entregara los triples RDF correspondientes de esa tabla, este proceso hará que los datos puedan ser vinculados entre si, estos triples se deberán guardar en una base de datos adaptada para los triples rdf como lo seria una Triplestore 2.10.

3.4. Almacenamiento de RDF y publicación

Dado que los datos que se recopilaran serán esencialmente abiertos, el vocabulario que se creara facilitara la publicación de la información que será procesable por máquinas, tal como lo recomiendan los principios FAIR. La adopción de estándares SPARQL y RDF/OWL hace que los datos sean accesibles e interoperables. Para lograr esto se debe almacenar los triples que se crearan con la herramienta Tarql en la sección 3.3, con este objetivo en mente se usará parte del Framework de Apache Jena del cual se habló en la sección 2.10.1, específicamente se trabajará con Fuseki²⁸, servidor SPARQL que puede presentar datos RDF y responder a consultas SPARQL a través de HTTP, y JenaTDB la triplestore de ApacheJena que viene integrada en Fuseki. En las figuras 3.21 y 3.22 se observa la ya implementación de fuseki con su interfaz de query y de carga de datos.

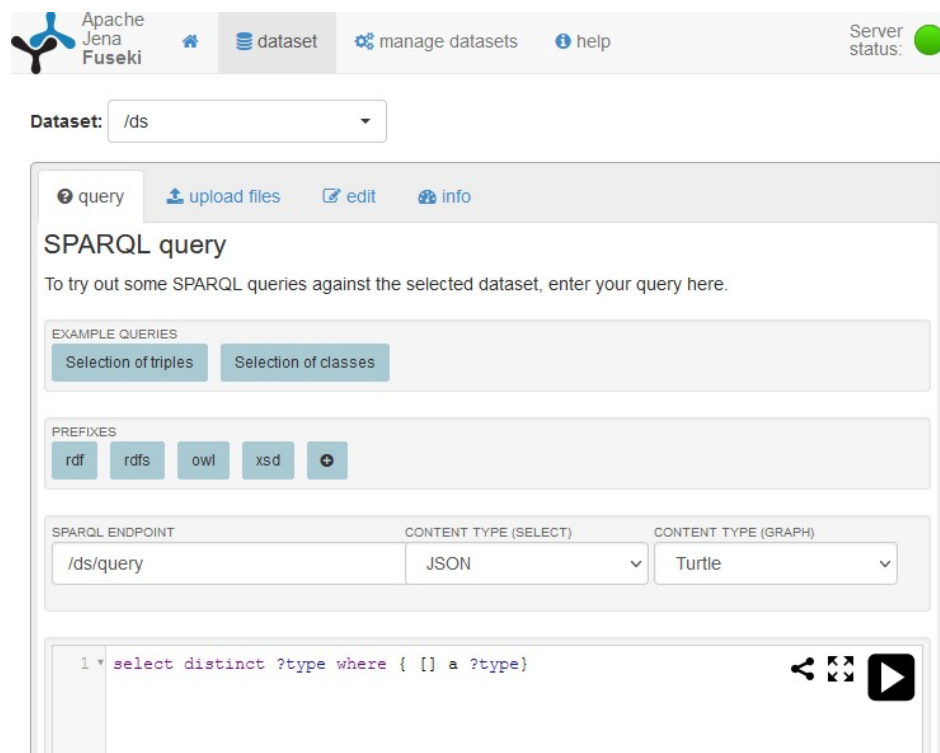


Figura 3.21: Interfaz de Query

Fuente: Elaboración Propia

²⁸<https://jena.apache.org/documentation/fuseki2/index.html>

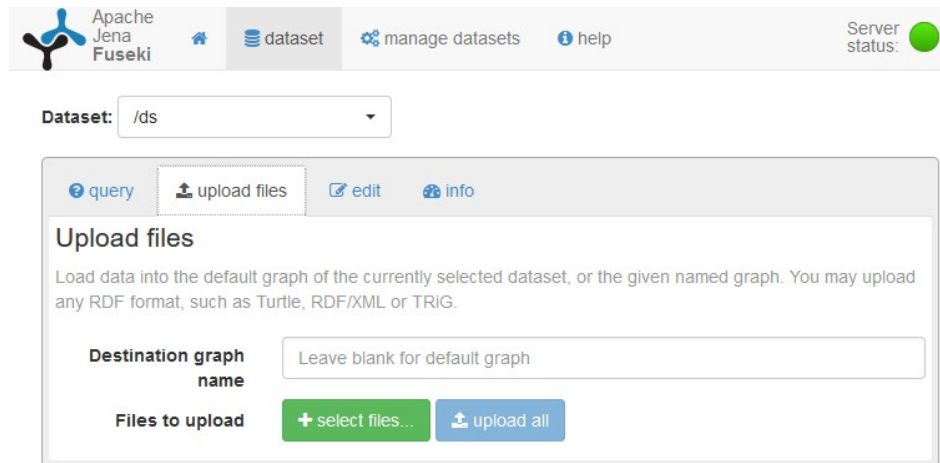


Figura 3.22: Interfaz de carga de datos

Fuente: Elaboracion Propia

Un punto crucial antes de la publicación de la data es la documentación, se debe documentar debidamente la ontología para que esta pueda ser interoperable y asegurar así cumplir con los principios, para esto se considera usar un asistente para generar la documentación de ontologías llamada WIDOCO[Garijo, 2017], para crear esta documentación, usara la ontología generada al final de la sección 3.2.

Para terminar se deberá asegurar la accesibilidad mencionada al principio de esta sección, para esto se usara un Frontend genérico para datos enlazados llamado Pubby, este permitirá el acceso solo con la URI a los recursos y entregara su información relacionada en su RDF.

Al terminar la implementación de Pubby se asegurara, al menos localmente, la accesibilidad ya mencionada. Llevar a cabo toda esta propuesta permitirá entre otras cosas buscar que ofertas recibió una licitación que fue pedida por cierto Municipio o que proveedores suelen adjudicarse las peticiones de algún organismo idealmente llegando a saber que personas están involucradas a ciertas compras, llegando a ese punto tendremos una integración de datos exitosa.

CAPÍTULO 4

VALIDACIÓN DE LA SOLUCIÓN

En este capítulo validaremos la propuesta presentada en el capítulo 3, para esto se realizó la propuesta en base a los resultados de la investigación previa.

4.1. Presentación de solución

Con lo estipulado en las secciones del capítulo anterior, se realizó una integración de datos paso a paso, primero se determinó cómo obtener la información. Por el sistema buscado que tiene el portal de transparencia para obtener las plantillas de personal por organismo se llegó a la conclusión de que lo mejor sería descargar directamente la plantilla de todos los organismos²⁹ ya que esto era fácilmente realizable sin incursionar en mayores problemas más que por el tamaño de dichos archivos, esto se puede realizar en la interfaz que se muestra en la figura 4.23.



Figura 4.23: Acceso a plantillas de personal de todos los organismos.

Estos datos son de modificación mensual, así que para trabajos futuros se buscara automatizar la obtención de esta data. Por otro lado como se consideró en la propuesta, se efectuó la descarga masiva de las licitaciones resguardadas en Chilecompra, las pe-

²⁹Portal de transparencia activa, todos los organismos:<http://www.portaltransparencia.cl/opendata/dataset/transparencia-activa-publicada-en-el-portal>

ticiones de descarga masiva son mensuales por lo que, en caso de ser necesario se ira agregando toda la información mes a mes. Estos datos están bajo la Licencia Creative Commons Atribución 4.0 Internacional³⁰.

Con la información obtenida se comenzó el proceso de confección de vocabulario usando al metodología mencionada en la sección 3.2, llegando a realizar la Conceptualizacion en base a las tablas de datos y la documentación de la API de chileCompra, se llegaron a establecer las entidades implicadas en el común de estos archivos y sus relaciones, esto conlleva a formalizar la ontología usando subclases de *GoodRelations*, *The Organization Ontology* y *Public Contracts Ontology*. Con esto listo se procedió a usar Protégé para confeccionar el vocabulario con las clases y propiedades obtenidas generando la ontología que muestra la figura 4.24.

³⁰Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by/4.0/> o envíe una carta a Creative Commons, PO Box 1866, Mountain View, CA 94042, USA. esta información es importante para asegurar el cumplimiento del principio de reutilización.

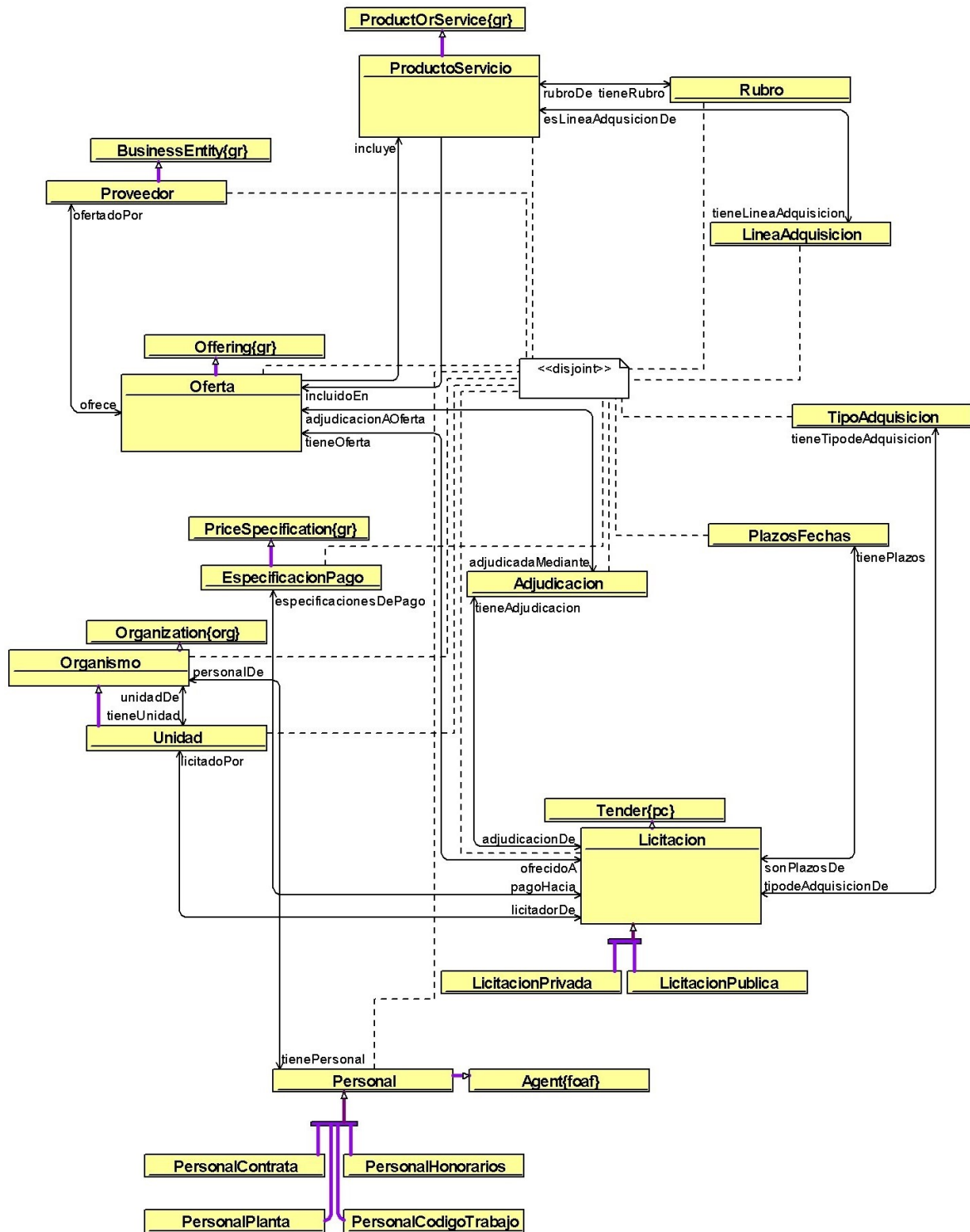
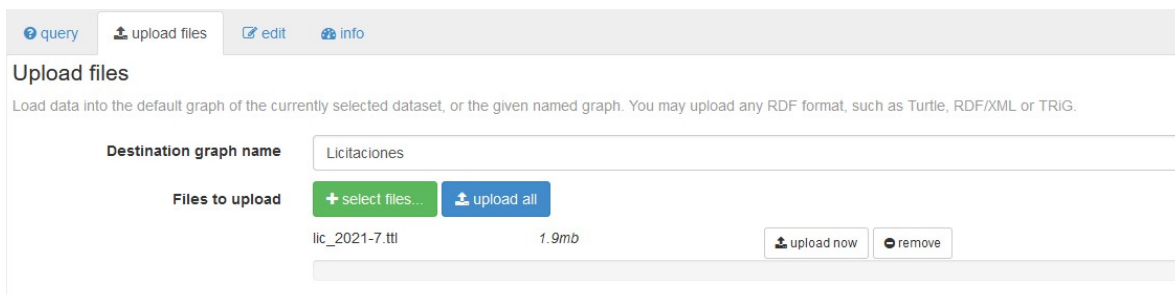


Figura 4.24: Ontología completa realizada para la integración.

Con el Vocabulario listo se procedió a crear las consultas que usara **Tarql** para realizar el mapeo, este se realizara con 5 archivos de mapeo, uno por cada tipo de plantilla de personal, esto por el volumen de datos y para lograr definir bien el tipo de contrato del personal (a contrata, honorarios, planta o código del trabajo) y un archivo para el tratamiento de las licitaciones. Nuevamente recordar que este proceso podrá ser automatizado para un trabajo a futuro con un sistema a mayor escala.

Estos datos estan licenciados bajo la Licencia Creative Commons Atribución 4.0 Internacional. Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by/4.0/> o envíe una carta a Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

Al terminar con el mapeo se obtuvo un grupo de archivos con todos los triples desplegados de esta data, luego se realizo la carga de triples a **jenaTDB** usando Fuseki como muestra la figura 4.25.



The screenshot shows the 'Upload files' interface of the Fuseki web application. At the top, there are tabs for 'query', 'upload files', 'edit', and 'info'. The 'upload files' tab is active. Below the tabs, there's a heading 'Upload files' and a subtext: 'Load data into the default graph of the currently selected dataset, or the given named graph. You may upload any RDF format, such as Turtle, RDF/XML or TRIG.' The main form has a 'Destination graph name' field containing 'Licitaciones'. Below this, there's a 'Files to upload' section. It contains a green '+ select files...' button and a blue 'upload all' button. A file named 'lic_2021-7.ttl' with a size of '1.9mb' is listed. At the bottom right of the file list, there are 'upload now' and 'remove' buttons.

Figura 4.25: Registro de carga de licitaciones de un mes en Fuseki.

Luego de cargar todos los datos se procedió a realizar una query buscando conocimiento que no se podía obtener antes, esta query se puede ver en el Código 5 donde el resultado arrojado lo entrega la tabla 4.

```

1 prefix mpCL: <http://www.semanticweb.org/mercadoPublicoCL#>
2 SELECT ?URITrabajador ?NombreTrabajador ?cargo ?nombreOrganizacion ?
   Licitacion
3 WHERE {
4   ?URITrabajador mpCL:nombre_completo ?NombreTrabajador;
5   mpCL:cargo_o_funcion ?cargo;
6   mpCL:personalDe ?org.
7   ?org mpCL:nombreOrg ?nombreOrganizacion;
8   mpCL:tieneUnidad ?und.
9   ?und mpCL:licitadorDe ?Licitacion.
10  filter(contains(UCASE(?cargo),"FINANZAS") && contains(UCASE(?
   nombreOrganizacion),"QUILPUE"))
11 }

```

Código 5: QUERY en búsqueda de conocimiento antes no accesible.

Tabla 4: La siguiente tabla entrega una muestra de los resultados de la QUERY 5

URITrabajador	NombreTrabajador	cargo	nombreOrganizacion	Licitacion
http://data.semanticmp.cl/resources/organismo/87633/personal/planta/a99744604e5f748826f7a5704de820	XIMENA TORRES VILLAROE	SECRETARIA EN DEPTO. DE FINANZAS	I. MUNICIPALIDAD DE QUILPUE	2428-57-LE21
http://data.semanticmp.cl/resources/organismo/87633/personal/planta/a99744604e5f748826f7a5704de820	XIMENA TORRES VILLAROE	SECRETARIA EN DEPTO. DE FINANZAS	I. MUNICIPALIDAD DE QUILPUE	2428-58-L121
http://data.semanticmp.cl/resources/organismo/87633/personal/contrata/ca885c04e46809267c0067c2d0e24	SELMA LARA GATICA	ADMINISTRATIVO EN EL DEPTO. DE FINANZAS Y CONTABILIDAD	I. MUNICIPALIDAD DE QUILPUE	2428-57-LE21
http://data.semanticmp.cl/resources/organismo/87633/personal/contrata/ca885c04e46809267c0067c2d0e24	SELMA LARA GATICA	ADMINISTRATIVO EN EL DEPTO. DE FINANZAS Y CONTABILIDAD	I. MUNICIPALIDAD DE QUILPUE	2428-58-L121
http://data.semanticmp.cl/resources/organismo/87633/personal/honorarios/ca885c04e46809267c0067c2d0e24	SELMA LARA GATICA	COMISION DE FINANZAS Y PRESUPUESTO	I. MUNICIPALIDAD DE QUILPUE	2428-57-LE21
http://data.semanticmp.cl/resources/organismo/87633/personal/honorarios/ca885c04e46809267c0067c2d0e24	SELMA LARA GATICA	COMISION DE FINANZAS Y PRESUPUESTO	I. MUNICIPALIDAD DE QUILPUE	2428-59-L121
http://data.semanticmp.cl/resources/organismo/87633/personal/contrata/a73749064a56c2f33604550c70b76	GONZALO ITURRIETA CONTRERAS	PROFESIONAL DEL DEPTO DE ADMINISTRACION Y FINANZAS	I. MUNICIPALIDAD DE QUILPUE	2428-57-LE21
http://data.semanticmp.cl/resources/organismo/87633/personal/contrata/a73749064a56c2f33604550c70b76	GONZALO ITURRIETA CONTRERAS	PROFESIONAL DEL DEPTO DE ADMINISTRACION Y FINANZAS	I. MUNICIPALIDAD DE QUILPUE	2428-58-L121
http://data.semanticmp.cl/resources/organismo/87633/personal/contrata/a73749064a56c2f33604550c70b76	GONZALO ITURRIETA CONTRERAS	PROFESIONAL DEL DEPTO DE ADMINISTRACION Y FINANZAS	I. MUNICIPALIDAD DE QUILPUE	2428-59-L121
http://data.semanticmp.cl/resources/organismo/87633/personal/planta/c302551f9b470940510bc32b4084571e	MARCIA MARTINEZ GIUSTO	SECRETARIA EN DEPTO. DE FINANZAS Y CONTABILIDAD	I. MUNICIPALIDAD DE QUILPUE	2428-57-LE21
http://data.semanticmp.cl/resources/organismo/87633/personal/planta/c302551f9b470940510bc32b4084571e	MARCIA MARTINEZ GIUSTO	SECRETARIA EN DEPTO. DE FINANZAS Y CONTABILIDAD	I. MUNICIPALIDAD DE QUILPUE	2428-58-L121
http://data.semanticmp.cl/resources/organismo/87633/personal/planta/c302551f9b470940510bc32b4084571e	MARCIA MARTINEZ GIUSTO	SECRETARIA EN DEPTO. DE FINANZAS Y CONTABILIDAD	I. MUNICIPALIDAD DE QUILPUE	2428-59-L121
http://data.semanticmp.cl/resources/organismo/87633/personal/planta/c302551f9b470940510bc32b4084571e	MARCIA MARTINEZ GIUSTO	FINANZAS Y CONTABILIDAD	I. MUNICIPALIDAD DE QUILPUE	2428-57-LE21
http://data.semanticmp.cl/resources/organismo/87633/personal/honorarios/62b49463d775893049876c3157c29b65	NICOLAS ARANDA FERNANDEZ	COMISION FINANZAS Y PRESUPUESTO	I. MUNICIPALIDAD DE QUILPUE	2428-57-LE21
http://data.semanticmp.cl/resources/organismo/87633/personal/honorarios/62b49463d775893049876c3157c29b65	NICOLAS ARANDA FERNANDEZ	COMISION FINANZAS Y PRESUPUESTO	I. MUNICIPALIDAD DE QUILPUE	2428-58-L121
http://data.semanticmp.cl/resources/organismo/87633/personal/honorarios/62b49463d775893049876c3157c29b65	NICOLAS ARANDA FERNANDEZ	COMISION FINANZAS Y PRESUPUESTO	I. MUNICIPALIDAD DE QUILPUE	2428-59-L121
http://data.semanticmp.cl/resources/organismo/87633/personal/honorarios/e050108015c598fa7661a89391246c	KAREN PIZARRO PALMA	Apoyo administrativo en dDepto. De Finanzas y Contabilidad	I. MUNICIPALIDAD DE QUILPUE	2428-57-LE21
http://data.semanticmp.cl/resources/organismo/87633/personal/honorarios/e050108015c598fa7661a89391246c	KAREN PIZARRO PALMA	Apoyo administrativo en dDepto. De Finanzas y Contabilidad	I. MUNICIPALIDAD DE QUILPUE	2428-58-L121
http://data.semanticmp.cl/resources/organismo/87633/personal/honorarios/e050108015c598fa7661a89391246c	KAREN PIZARRO PALMA	Apoyo administrativo en dDepto. De Finanzas y Contabilidad	I. MUNICIPALIDAD DE QUILPUE	2428-59-L121
...				

Se puede obtener mas información de el resultado de esta consulta en el repositorio del proyecto accediendo a la tabla en el siguiente enlace [Respuesta de Query]. Esta consulta pregunta por los trabajadores del área de finanzas de la municipalidad de Quilpué que tuvieron directa relación con licitaciones hechas por el organismo, la respuesta otorga información de los responsables y las licitaciones en las que participaron.

Con esto se confirma que el vocabulario cumple su propósito. Además se habilito el servidor de Pubby para tener una vista mas ordenada de los datos esto se muestra en la figura 4.26.

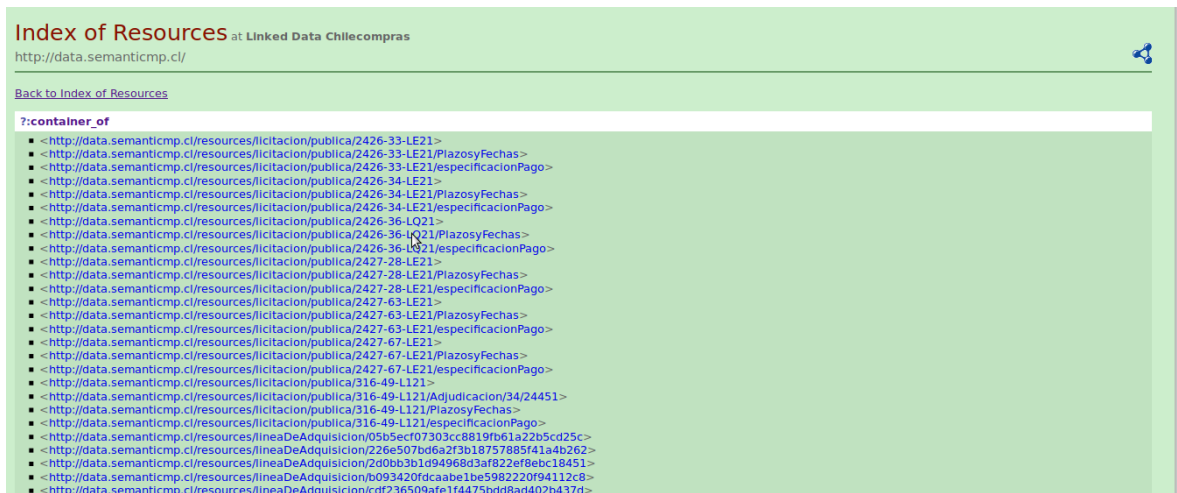


Figura 4.26: Lista preliminar de datos en el servidor de Pubby.

También se logra obtener las propiedades de los recursos guardados como muestran las figuras 4.27a, 4.27b y 4.27c, en estas se logra ver el verdadero resultado del mapeo realizado, este consigue seguir los principios FAIR consiguiendo identificadores únicos para cada recurso, al usar Pubby se tiene un acceso a la información expedito, como se hablo antes al seguir los estándares RDF/OWL y SPARQL los datos son interoperables pero para asegurar se uso WIDOCO como Documentación para que puedan efectivamente ser usados por distintos estamentos como muestra la figura 4.28.

2426-33-LE21 at Linked Data Chilecompras
<http://data.semanticmp.cl/resources/licitacion/publica/2426-33-LE21>

<code>:cantidadReclamos</code>	461
<code>:codigo</code>	8832025
<code>:codigoEstadoLicitacion</code>	6
<code>:codigoExterno</code>	2426-33-LE21
<code>:contrato</code>	Contrato Requiere subscripción
<code>:descripcion</code>	reparación de viviendas de material ligero del programa de Habitabilidad 2019, ubicados en distintos sectores de la comuna de Valparaíso
<code>:esBaseTipo</code>	no
<code>:esRenovable</code>	no
<code>:especificacionesDePago</code>	< http://data.semanticmp.cl/resources/licitacion/publica/2426-33-LE21/especificacionPago >
<code>:estado</code>	Cerrada
<code>:estadoEtapas</code>	1
<code>:estadoPublicidadOfertas</code>	no muestra la publicidad de oferta técnica
<code>:etapas</code>	1
<code>:extensionPlazo</code>	No extiende
<code>:fechaUsuario</code>	no presenta fechas definidas por usuario
<code>:informada</code>	no
<code>:licitadoPor</code>	< http://data.semanticmp.cl/resources/organismo/87402 >
<code>Is :licitadorDe of</code>	< http://data.semanticmp.cl/resources/organismo/87402/unidad/3418 >
<code>:linkURL</code>	http://www.mercadopublico.cl/fichaLicitacion.html?idLicitacion=2426-33-LE21
<code>:nombre</code>	Serv. de reparación de viviendas para 17 familias
<code>:numeroOfertantes</code>	1

(a) Datos de una licitación

87402 at Linked Data Chilecompras
<http://data.semanticmp.cl/resources/organismo/87402>

<code>:codigoOrg</code>	87402
<code>Is :licitadoPor of</code>	< http://data.semanticmp.cl/resources/licitacion/publica/2426-33-LE21 > < http://data.semanticmp.cl/resources/licitacion/publica/2426-34-LE21 > < http://data.semanticmp.cl/resources/licitacion/publica/2426-36-LQ21 > < http://data.semanticmp.cl/resources/licitacion/publica/2426-39-LE21 > < http://data.semanticmp.cl/resources/licitacion/publica/2427-28-LE21 > < http://data.semanticmp.cl/resources/licitacion/publica/2427-63-LE21 > < http://data.semanticmp.cl/resources/licitacion/publica/2427-67-LE21 >
<code>:nombreOrg</code>	ILUSTRE MUNICIPALIDAD DE VALPARAISO
<code>Is :personalDe of</code>	[66283 values]
<code>:tienePersonal</code>	[26818 values]
<code>:tieneUnidad</code>	< http://data.semanticmp.cl/resources/organismo/87402/unidad/3418 > < http://data.semanticmp.cl/resources/organismo/87402/unidad/3419 >
<code>rdf:type</code>	Organismo
<code>Is :unidadDe of</code>	< http://data.semanticmp.cl/resources/organismo/87402/unidad/3418 > < http://data.semanticmp.cl/resources/organismo/87402/unidad/3419 >

[As Turtle](#) | [As RDF/XML](#)

(b) Datos de un Organismo

57ed69ac99cd62fa0452de30d8c254b7 at Linked Data Chilecompras
<http://data.semanticmp.cl/resources/organismo/7233/personal/codigoTrabajo/57ed69ac99cd62fa0452de30d8c254b7>

<code>:calificacionProfesional</code>	MEDICO SIN INFORMACION Sin Informacion
<code>:cargo_o_funcion</code>	GABINETE PSICOTECNICO MEDICO PSICOTECNICO
<code>:fecha_inicio</code>	01/04/2014
<code>:fecha_termino</code>	31/12/2015 31/12/2016 Indefinido
<code>:nombre_completo</code>	MIGUEL ANGEL DE LA ROSA MANZUR
<code>:personalDe</code>	< http://data.semanticmp.cl/resources/organismo/7233 >
<code>Is :tienePersonal of</code>	< http://data.semanticmp.cl/resources/organismo/7233 >
<code>rdf:type</code>	PersonalCodigoTrabajo

[As Turtle](#) | [As RDF/XML](#)

(c) Datos de un Miembro del personal de algún organismo

Figura 4.27: Muestras de Datos en Pubby

Ontology Specification Draft

4. Términos de [Ontology Name] para clases, propiedades y propiedades de datos [volver a índice](#)

Esta sección introduce más detalles sobre cada clase y propiedad definida por [Ontology Name].

4.1. Clases

[licitacion](#)
[licitacion privada](#)
[licitacion publica](#)
[linea adquisicion](#)
[oferta](#)
[organismo](#)
[personal](#)

[Personal a Contrata](#)
[Personal Codigo del Trabajo](#)
[Personal contratado a Honorarios](#)
[Personal de planta](#)
[plazos fechas](#)
[producto servicio](#)

[proveedor](#)
[rubro](#)
[tipo adquisicion](#)
[unidad](#)

adjudicacion
[Volver a índice o índice de clases](#)

IRI: <http://www.semanticweb.org/mercadoPublicoCL#Adjudicacion>

Clase que representa la adjudicacion de una licitacion en cuestion

en dominio de

[Adjudicacion a oferta](#) ^{op}, [Adjudicacion de](#) ^{op}, [cantidad adjudicada](#) ^{dp}, [fecha aprobacion](#) ^{dp}, [monto linea adjudica](#) ^{dp}, [numero aprobacion](#) ^{dp}, [tipo aprobacion](#) ^{dp}

en rango de

[Adjudicada mediante](#) ^{op}, [Tiene una adjudicacion asociada](#) ^{op}

disjunto a

[especificacion pago](#) ^c, [licitacion](#) ^c, [linea adquisicion](#) ^c, [oferta](#) ^c, [organismo](#) ^c, [personal](#) ^c, [plazos fechas](#) ^c, [producto servicio](#) ^c, [proveedor](#) ^c, [rubro](#) ^c, [tipo adquisicion](#) ^c, [unidad](#) ^c

(a)

Personal de planta
[Volver a índice o índice de clases](#)

IRI: <http://www.semanticweb.org/mercadoPublicoCL#PersonalPlanta>

Personal de planta del organismo

tiene superclases

[personal](#) ^c

disjunto a

[Personal Codigo del Trabajo](#) ^c, [Personal a Contrata](#) ^c, [Personal contratado a Honorarios](#) ^c

plazos fechas
[Volver a índice o índice de clases](#)

IRI: <http://www.semanticweb.org/mercadoPublicoCL#PlazosFechas>

Clase que contendra distintas fechas asignadas a la licitacion

en dominio de

[acto apertura economica](#) ^{dp}, [acto apertura tecnica](#) ^{dp}, [adjudicacion](#) ^{dp}, [cierre](#) ^{dp}, [creacion](#) ^{dp}, [entrega antecedentes](#) ^{dp}, [estimada adjudicacion](#) ^{dp}, [estimada firma](#) ^{dp}, [final](#) ^{dp}, [inicio](#) ^{dp}, [publicacion](#) ^{dp}, [publicacion respuestas](#) ^{dp}, [son plazos de](#) ^{op}, [soporte fisico](#) ^{dp}, [visita terreno](#) ^{dp}

en rango de

[tiene plazos](#) ^{op}

disjunto a

[adjudicacion](#) ^c, [especificacion pago](#) ^c, [licitacion](#) ^c, [linea adquisicion](#) ^c, [oferta](#) ^c, [organismo](#) ^c, [personal](#) ^c, [producto servicio](#) ^c, [proveedor](#) ^c, [rubro](#) ^c, [tipo adquisicion](#) ^c, [unidad](#) ^c

Figura 4.28: Documentación WIDOCO para el Proyecto.

Esta documentación se puede obtener completamente por el repositorio del proyecto accediendo a la Especificación de Ontología, aquí se presenta el resultado obtenido por la herramienta WIDOCO.

El proceso de FAIRificación llega a su fin al tener la documentación del vocabulario, ya que con esto se cumplen todas las etapas de este ya que se obtuvieron datos no FAIR con una licencia establecida, se creo un vocabulario usando **Protégé** con los datos estableciendo un modelo semántico, se establecieron las relaciones entre ellos y se

logro hacer el enlace entre información por medio de las URI y la herramienta **Tarql**, ejemplos de esto ultimo estarán presentes en el código 6.

```
1 <http://data.semanticmp.cl/resources/licitacion/publica/2427-28-LE21> <  
    http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.  
    semanticweb.org/mercadoPublicoCL#LicitacionPublica> .  
2  
3 <http://data.semanticmp.cl/resources/licitacion/publica/2427-28-LE21> <  
    http://www.semanticweb.org/mercadoPublicoCL#licitadoPor> <http://data.  
    semanticmp.cl/resources/organismo/87402> .  
4  
5 <http://data.semanticmp.cl/resources/proveedor/1365794> <http://www.  
    semanticweb.org/mercadoPublicoCL#tieneRubro> <http://data.semanticmp.  
    cl/resources/rubro/bb1d0ee5e245397167c73a68e49fc8c8> .
```

Código 6: Ejemplos de triples obtenidos.

Todos los triples obtenidos por el mapeo realizado por Tarql fueron revisados para cumplir con los principios FAIR se puede encontrar el total de triples trabajados en el repositorio del proyecto en el Almacenamiento. Queda a pendiente para trabajo futuro la publicación de esta información con un triplestore publico ya que esto supera el alcance de la memoria.

CAPÍTULO 5

CONCLUSIONES Y TRABAJO FUTURO

El diluvio de datos como se hablo al principio de este documento entrega una clara visión de lo que es hoy en día la red, un sinfín de información siendo intercambiada, usada, obtenida, etcétera, tanto es así que hacen falta mecanismos que permitan hacer un uso apropiado de esta. La aparición de los principio FAIR trajo una gran ayuda para las distintas áreas del conocimiento y el consumo, lo que conlleva a que se crearan cada vez mas ecosistemas de información.

Los modelos de datos apropiados y los estándares en la información recopilada son claramente la clave para lograr las metas que proponen los ya mencionados principios, que a su vez han sido un pilar para mejorar el acceso digital al conocimiento de distintas áreas, tanto a nivel académico como practico. En los últimos años, soluciones como los grafos de conocimiento y tecnologías como las bases de datos de grafos se han hecho populares en contextos comerciales y de producción. En parte, enfoques tradicionalmente académicos como los datos enlazados han salido del ámbito académico y han enriquecido estos nuevos desarrollos.

El trabajo de esta memoria fue el lograr incorporar estos estándares a un nivel mas cercano, mas específicamente al nivel del ciudadano común que paga impuestos y quiere saber que se hace con ellos, para este objetivo se logro recabar información pertinente a la administración publica con la que se logro efectuar un modelo semántico que permitió vincular correctamente los datos llevando a cabo un tratamiento de estos para que sean usados para cualquier fin pertinente de forma reiterada, manteniendo su individualidad. Si bien la propuesta de solución puede ser llevada a cabo en un alcance mas amplio, se logro llevar a cabo los objetivos que se establecieron en la sección 1.2 aterrizando el problema como se propuso, esto facilito el desarrollo del vocabulario haciendo uso de la documentación correspondiente de los datos obtenidos. Al lograr esto el mapeo de datos que fue usado de parte de Tarql logro moldear la información para que cumpliera

con los requisitos presentados. Ciertamente hay puntos que no se pudo abordar como lo son la publicación de los datos ya que, por lo acotado del proyecto no traería una relevancia real a el ecosistema de datos ya que este no esta completamente creado, habría que realizar este proceso a un espectro mas amplio de datos y la verdad eso se va de los alcances que una memoria podría tener, sin considerar que necesita una mayor cantidad de procesamiento y mas tiempo de estudio, además de depender directamente del auspicio de la administración publica pues seria una herramienta que debe ser desarrollada por el gobierno.

5.1. Trabajo Futuro

El desarrollo de un vocabulario que permita la integración de TODOS los datos de la administración publica es un objetivo que se podría realizar a largo plazo, realizar el proceso de esta memoria a todos los datos del portal de transparencia ciertamente seria un objetivo ambicioso que además necesitaría del trabajo en conjunto de el equipo de desarrollo y el equipo que mantiene los datos para lograr tener mayor nivel de vinculación entre distintas bases de datos al rededor de el portal (Licitaciones, Plantillas de personal, fondos, compras, etcetera). Este nivel de integración además debe respetar los principios FAIR para lograr un nivel de interoperatividad y reutilización que permitirá obtener un gran nivel de conocimiento para mejorar la toma de decisiones o simplemente informar.

REFERENCIAS BIBLIOGRÁFICAS

- [Auer *et al.*, 2007] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., e Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. En Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., y Cudré-Mauroux, P., editores, *The Semantic Web*, pp. 722–735, Berlin, Heidelberg. Springer Berlin Heidelberg.
- [Beckett, 2014] Beckett, D. (2014). RDF 1.1 N-Triples.
- [Beckett *et al.*, 2014] Beckett, D., Berners-Lee, T., Prud’hommeaux, E., y Carothers, G. (2014). Rdf 1.1 turtle. *World Wide Web Consortium*, pp. 18–31.
- [Beckett y McBride, 2004] Beckett, D. y McBride, B. (2004). Rdf/xml syntax specification (revised). *W3C recommendation*, 10(2.3).
- [Berners-Lee, 2011] Berners-Lee, T. (2011). Linked data-design issues (2006),.
- [Berners-Lee, 2014] Berners-Lee, T. (2014). Semantic web architecture.
- [Berners-Lee y CERN, 1989] Berners-Lee, T. y CERN, M. (1989). The original proposal of the www. URL <http://www.w3.org/History/1989/proposal.html>, pp. 706–716.
- [Berners-Lee, 1992] Berners-Lee, T. J. (1992). The world-wide web. *Computer networks and ISDN systems*, 25(4-5):454–459.
- [Biblioteca del Congreso Nacional, 2008] Biblioteca del Congreso Nacional (2008). Ley 20285 SOBRE ACCESO A LA INFORMACIÓN PÚBLICA | Biblioteca del congreso nacional | ley chile.
- [Bikakis *et al.*, 2013] Bikakis, N., Tsinaraki, C., Gioldasis, N., Stavarakantonakis, I., y Christodoulakis, S. (2013). The xml and semantic web worlds: technologies, interoperability and integration: a survey of the state of the art. En *Semantic hyper/multimedia adaptation*, pp. 319–360. Springer.

- [Brandizi *et al.*, 2018] Brandizi, M., Singh, A., Rawlings, C., y Hassani-Pak, K. (2018). Towards fairer biological knowledge networks using a hybrid linked data and graph database approach. *Journal of Integrative Bioinformatics*, 15(3):20180023.
- [Bray, 2008] Bray, T. (2008). Extensible Markup Language (Xml) 1.0 (Fifth edition).
- [Brickley *et al.*, 2014] Brickley, D., Guha, R. V., y McBride, B. (2014). Rdf schema 1.1. *W3C recommendation*, 25:2004–2014.
- [Corcho *et al.*, 2005] Corcho, O., Fernández-López, M., Gómez-Pérez, A., y López-Cima, A. (2005). Construcción de ontologías legales con la metodología methontology y la herramienta webode. *Law and the Semantic Web. Legal Ontologies, Methodologies, Legal Information Retrieval, and Applications*, pp. 142–157.
- [Economist, 2010] Economist, T. (2010). The data deluge. *The Economist*.
- [Gangemi y Presutti, 2006] Gangemi, A. y Presutti, V. (2006). The bourne identity of a web resource. En *Proceedings of Identity Reference and the Web Workshop (IRW) at the WWW Conference*.
- [Garijo, 2017] Garijo, D. (2017). Widoco: a wizard for documenting ontologies. En *International Semantic Web Conference*, pp. 94–102. Springer.
- [Masinter *et al.*, 2005] Masinter, L., Berners-Lee, T., y Fielding, R. T. (2005). Uniform resource identifier (uri): Generic syntax. *Network Working Group: Fremont, CA, USA*.
- [Prud’hommeaux, 2008] Prud’hommeaux, E. (2008). Sparql query language for rdf, w3c recommendation. <http://www.w3.org/TR/rdf-sparql-query/>.
- [Rusher,] Rusher, J. Triple store, 2001. *Last Accessed: http://www.w3.org/2001/sw/Europe/events/20031113-storage/positions/rusher.html*.
- [Szekely *et al.*, 2015] Szekely, P., Knoblock, C. A., Slepicka, J., Philpot, A., Singh, A., Yin, C., Kapoor, D., Natarajan, P., Marcu, D., Knight, K., Stallard, D., Karunamoorthy, S. S., Bojanapalli, R., Minton, S., Amanatullah, B., Hughes, T., Tamayo,

- M., Flynt, D., Artiss, R., Chang, S.-F., Chen, T., Hiebel, G., y Ferreira, L. (2015). Building and using a knowledge graph to combat human trafficking. En *Proceedings of the 14th International Semantic Web Conference (ISWC 2015)*.
- [Szekely *et al.*, 2013] Szekely, P., Knoblock, C. A., Yang, F., Zhu, X., Fink, E. E., Allen, R., y Goodlander, G. (2013). Connecting the smithsonian american art museum to the linked data cloud. En Cimiano, P., Corcho, O., Presutti, V., Hollink, L., y Rudolph, S., editores, *The Semantic Web: Semantics and Big Data*, pp. 593–607, Berlin, Heidelberg. Springer Berlin Heidelberg.
- [The W3C SPARQL Working Group, 2013] The W3C SPARQL Working Group (2013). SPARQL 1.1 Overview.
- [Wilkinson *et al.*, 2016] Wilkinson, M. D., Dumontier, M., y Aalbersberg (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, 3(1):160018.