



UNIVERSIDAD TECNOLÓGICA NACIONAL

Facultad Regional Santa Fe

DOCTORADO EN INGENIERÍA

MENCIÓN EN INGENIERÍA EN SISTEMAS DE INFORMACIÓN

**Tesis Doctoral**

“SOPORTE PARA ENRIQUECER LA REPRESENTACIÓN  
DE ENTIDADES EN UNA ONTOLOGÍA”

Ing. Mariela G. Rico

Director Dra. María Rosa Galli

Codirector Dr. Omar Chiotti

Santa Fe, Argentina.

Agosto de 2011

Rico, Mariela Guadalupe

Soporte para enriquecer la representación de entidades en una ontología. - 1a ed. -  
Santa Fe : el autor, 2011.

287 p. ; 29x21 cm.

ISBN 978-987-33-1236-6

1. Ontología. I. Título

CDD 111

Fecha de catalogación: 29/09/2011

UNIVERSIDAD TECNOLÓGICA NACIONAL

Facultad Regional Santa Fe

Comisión de Posgrado

Se presenta esta Tesis en cumplimiento de los requisitos exigidos por la Universidad Tecnológica Nacional para la obtención del grado académico de Doctor en Ingeniería, mención Sistemas de Información

“SOPORTE PARA ENRIQUECER LA REPRESENTACIÓN  
DE ENTIDADES EN UNA ONTOLOGÍA”

POR

ING. MARIELA G. RICO

DIRECTOR DRA. MARÍA ROSA GALLI

CODIRECTOR DR. OMAR CHIOTTI

JURADOS DE TESIS

DRA. REGINA MOTZ

DR. LUIS OLSINA

DR. PABLO VILLARREAL

SANTA FE, ARGENTINA.

AGOSTO DE 2011



*A Lucía, Lucas y Marcelo.*

*A mis padres.*

*A la memoria de mi abuelo Angel.*



# Índice General

Índice de Tablas	VII
Índice de Figuras	IX
Prólogo	XIII
Resumen	XIX
Reconocimientos	XXI
<b>Capítulo 1. Introducción</b>	<b>1</b>
1.1. Ontologías . . . . .	1
1.1.1. Categorización de las ontologías . . . . .	5
1.1.1.1. Tipos de ontologías según la riqueza de su estructura interna . . . . .	5
1.1.1.2. Tipos de ontologías según el tema de conceptualización . . . . .	6
1.1.2. Campos de aplicación de las ontologías . . . . .	7
1.2. Interoperabilidad Semántica. Ontologías y Contextos . . . . .	10
1.2.1. El problema del matching de ontologías . . . . .	13
1.3. Propuestas y Objetivos de esta Tesis . . . . .	14
1.3.1. Método para enriquecer la representación de las entidades en una ontología . . . . .	14
1.3.2. Framework para dar soporte al reuso de ontologías . . . . .	15
1.3.3. Lenguaje de representación de características contextuales . . . . .	16
1.4. Conclusiones . . . . .	16
<b>Capítulo 2. Método para Enriquecer la Representación de las Entidades en una Ontología</b>	<b>19</b>
2.1. Introducción . . . . .	19
2.2. Ámbito de Aplicación y Objetivo . . . . .	21
2.3. Descripción General del Método . . . . .	23
2.3.1. Identificación de los requerimientos de la ontología . . . . .	24
2.3.1.1. Describir el dominio . . . . .	25
2.3.1.2. Determinar el objetivo y alcance de la ontología . . . . .	26
2.3.1.3. Identificar las principales entidades del dominio, sus relaciones y características . . . . .	27

2.3.2.	Generación de una ontología base . . . . .	27
2.3.3.	Formalización de la ontología . . . . .	30
2.3.4.	Verificación de la ontología . . . . .	33
2.3.4.1.	Identificar la representación de las entidades del dominio en la ontología . . . . .	34
2.3.4.2.	Analizar la representación de las entidades del dominio, sus relaciones y características . . . . .	35
2.3.4.3.	Verificar la coherencia lógica de la ontología . . . . .	38
2.3.5.	Enriquecimiento de la ontología . . . . .	39
2.3.5.1.	Añadir los elementos faltantes . . . . .	39
2.3.5.2.	Mejorar la representación de las entidades del dominio, sus relaciones y características . . . . .	40
2.3.5.3.	Designar un término puente que refiera a la entidad de dominio . . . . .	45
2.3.6.	Validación de la ontología . . . . .	46
2.4.	Análisis Comparativo de la Calidad de Ontologías . . . . .	47
2.4.1.	Dimensiones de calidad . . . . .	48
2.4.2.	Evaluación de las dimensiones de calidad . . . . .	48
2.4.2.1.	Evaluación de la dimensión sintáctica . . . . .	49
2.4.2.2.	Evaluación de la dimensión semántica . . . . .	52
2.4.2.3.	Evaluación de la dimensión pragmática . . . . .	57
2.5.	Conclusiones . . . . .	58
<b>Capítulo 3. Aplicación del Método Propuesto a un Caso de Estudio</b>		<b>61</b>
3.1.	Caso de Estudio 1: Una Ontología para una e-Colaboración . . . . .	61
3.1.1.	Identificación de los requerimientos de la ontología . . . . .	62
3.1.1.1.	Describir el dominio . . . . .	62
3.1.1.2.	Determinar el objetivo y alcance de la ontología . . . . .	68
3.1.1.3.	Identificar las principales entidades del dominio, sus relaciones y características . . . . .	69
3.1.2.	Generación de una ontología base . . . . .	71
3.1.3.	Formalización de la ontología . . . . .	72
3.1.4.	Verificación de la ontología base . . . . .	72
3.1.4.1.	Verificación de la representación de la entidad socio de negocio . . . . .	73
3.1.4.2.	Verificación de la representación de la entidad programa de provisión . . . . .	74
3.1.4.3.	Verificación de la representación de la entidad ítem . . . . .	75
3.1.5.	Enriquecimiento de la ontología base . . . . .	77
3.1.5.1.	Enriquecimiento de la representación de la entidad socio de negocio . . . . .	77
3.1.5.2.	Enriquecimiento de la representación de la entidad programa de provisión . . . . .	80
3.1.5.3.	Enriquecimiento de la representación de la entidad ítem . . . . .	81

3.1.6.	Verificación de la ontología enriquecida . . . . .	87
3.1.7.	Validación de la ontología . . . . .	89
3.2.	Análisis Comparativo de la Calidad de las Ontologías . . . . .	91
3.2.1.	Evaluación de la dimensión sintáctica . . . . .	92
3.2.2.	Evaluación de la dimensión semántica . . . . .	93
3.2.3.	Evaluación de la dimensión pragmática . . . . .	95
3.3.	Evaluación de los Procesos de Matching de Ontologías . . . . .	96
3.3.1.	Matching entre las ontologías DNE Base y Privada . . . . .	97
3.3.2.	Matching entre las ontologías DNE Contextualizada, uso previsto industria de envases, y Privada . . . . .	99
3.3.3.	Matching entre las ontologías DNE Contextualizada comple- ta y Privada . . . . .	101
3.3.4.	Evaluación de los alignments obtenidos . . . . .	103
3.3.5.	Análisis de otros sistemas de matching . . . . .	105
3.4.	Conclusiones . . . . .	108
 <b>Capítulo 4. Framework para dar Soporte al Reuso de Ontologías</b>		<b>111</b>
4.1.	Introducción . . . . .	111
4.2.	Framework para el Desarrollo de Ontologías . . . . .	114
4.2.1.	Ontología base . . . . .	116
4.2.2.	Ontologías de características primitivas . . . . .	117
4.2.2.1.	Ontología de características primitivas temporal . . . . .	118
4.2.2.2.	Ontología de características primitivas espacial . . . . .	119
4.2.2.3.	Ontología de características primitivas de proceso . . . . .	120
4.2.2.4.	Ontología de características primitivas funcional . . . . .	121
4.2.3.	Ontología de especificación de contexto . . . . .	122
4.2.4.	Ontología enriquecida . . . . .	126
4.3.	Caso de Estudio 2: Desarrollo de una Ontología de Interfaz . . . . .	129
4.3.1.	Identificación de los requerimientos de la ontología . . . . .	130
4.3.2.	Generación y formalización de una ontología base . . . . .	131
4.3.2.1.	Ontología base . . . . .	131
4.3.3.	Verificación de la ontología base . . . . .	132
4.3.3.1.	Verificación de la representación de la entidad ma- terial de empaque . . . . .	132
4.3.3.2.	Verificación de la representación de la entidad pro- ducto terminado . . . . .	133
4.3.4.	Enriquecimiento de la ontología base . . . . .	135
4.3.4.1.	Ontología de especificación de contexto . . . . .	135
4.3.4.2.	Ontología enriquecida . . . . .	139
4.3.5.	Verificación y validación de la ontología enriquecida . . . . .	141
4.4.	Herramientas para la Implementación del Framework Propuesto . . . . .	143
4.5.	Conclusiones . . . . .	145

<b>Capítulo 5. Lenguaje de Representación de Características Contextuales</b>	<b>147</b>
5.1. Introducción . . . . .	147
5.2. Ontología de Referencia . . . . .	154
5.3. Lenguaje General de Representación de Características . . . . .	159
5.3.1. Meta-ontología de características contextuales . . . . .	159
5.3.1.1. Context . . . . .	160
5.3.1.2. Feature . . . . .	161
5.3.1.3. RelationalFeature . . . . .	161
5.3.1.4. IntrinsicFeature . . . . .	162
5.3.1.5. QualityUniversal . . . . .	163
5.3.1.6. SimpleQualityUniversal . . . . .	164
5.3.1.7. ComplexQualityUniversal . . . . .	165
5.3.1.8. Set . . . . .	166
5.3.1.9. QualityStructure . . . . .	166
5.3.1.10. QualityDimension . . . . .	167
5.3.1.11. QualityMultiDimension . . . . .	168
5.3.1.12. UnitOfMeasure . . . . .	170
5.3.1.13. PhysicalDimension . . . . .	171
5.3.2. Implementación del lenguaje en la herramienta MetaXys . . . . .	171
5.3.2.1. Meta-ontología implementada en Ecore . . . . .	172
5.3.2.2. Sintaxis concreta . . . . .	174
5.4. Aplicación del Lenguaje al Caso de Estudio 1 . . . . .	176
5.4.1. Enriquecimiento de la ontología base . . . . .	177
5.4.1.1. Enriquecimiento de la representación de la entidad socio de negocio . . . . .	177
5.4.1.2. Enriquecimiento de la representación de la entidad programa de provisión . . . . .	179
5.4.1.3. Enriquecimiento de la representación de la entidad ítem . . . . .	185
5.5. Conclusiones . . . . .	196
<b>Capítulo 6. Conclusiones y Trabajos Futuros</b>	<b>199</b>
6.1. Principales Contribuciones . . . . .	199
6.1.1. Método para enriquecer la representación de las entidades en una ontología . . . . .	200
6.1.2. Framework para dar soporte al reuso de ontologías . . . . .	201
6.1.3. Lenguaje de representación de características contextuales . . . . .	202
6.2. Trabajos Futuros . . . . .	203
<b>Apéndice A. Representación Gráfica de Ontologías</b>	<b>205</b>
<b>Apéndice B. Ontología DNE Base Formalizada en OWL</b>	<b>207</b>
<b>Apéndice C. Detalle de Cálculo de Medidas para la Evaluación Semántica</b>	<b>223</b>

**Bibliografía**

**233**



# Índice de Tablas

3.1. DERO: Descripción del dominio. Preguntas de competencia . . . . .	68
3.2. DERO: Relaciones entre las entidades del dominio . . . . .	70
3.3. DERO: Características de las entidades del dominio . . . . .	71
3.4. Resultado de la verificación . . . . .	78
3.5. Algunos indicadores para la dimensión sintáctica . . . . .	92
3.6. Algunos indicadores para la dimensión semántica . . . . .	93
3.7. <i>Alignments</i> $A_1$ y $E_1$ de elementos involucrados en la representación de la entidad producto . . . . .	99
3.8. <i>Alignments</i> $A_2$ y $E_2$ de elementos involucrados en la representación de la entidad producto . . . . .	100
3.9. <i>Alignments</i> $A_3$ y $E_3$ de elementos involucrados en la representación de la entidad producto . . . . .	102
3.10. Comparación de <i>alignments</i> obtenidos sobre elementos involucrados en la representación de la entidad producto . . . . .	104
3.11. Comparación de <i>alignments</i> obtenidos sobre todas las entidades consideradas en la e-colaboración . . . . .	105
3.12. Comparación de <i>alignments</i> obtenidos por distintos sistemas de <i>matching</i> para la entidad producto entre las ontologías DNE Base y Privada . . . . .	106
3.13. Comparación de <i>alignments</i> obtenidos por distintos sistemas de <i>matching</i> para la entidad producto entre las ontologías DNE Con- textualizada y Privada . . . . .	107
4.1. DERO: Relaciones entre las entidades del dominio . . . . .	130
4.2. DERO: Características de las entidades del dominio . . . . .	130
4.3. Resultado de la verificación . . . . .	134
C.1. Detalles del cálculo de la interpretabilidad y claridad de la ontología DNE Base . . . . .	224
C.2. Detalles del cálculo de la interpretabilidad y claridad de la ontología DNE Contextualizada (hoja 1) . . . . .	225
C.3. Detalles del cálculo de la interpretabilidad y claridad de la ontología DNE Contextualizada (hoja 2) . . . . .	226
C.4. Detalles del cálculo de la interpretabilidad y claridad de la ontología DNE Contextualizada (hoja 3) . . . . .	227

C.5. Detalles del cálculo de la interpretabilidad y claridad de la ontología DNE Contextualizada (hoja 4) . . . . .	228
C.6. Detalles del cálculo del principio 1: Entidades . . . . .	228
C.7. Detalles del cálculo del principio 2: Uso previsto de entidades . . .	229
C.8. Detalles del cálculo del principio 3: Relaciones entre entidades . .	229
C.9. Detalles del cálculo del principio 4: Características como entidades	230
C.10. Detalles del cálculo del principio 5: Características comunes . . . .	231

# Índice de Figuras

2.1. Una visión general del método propuesto . . . . .	24
2.2. Diagrama de actividades del proceso: Identificación de los requerimientos de la ontología . . . . .	25
2.3. Diagrama de actividades del proceso: Verificación de la ontología . . . . .	34
2.4. Diagrama de actividades del proceso: Enriquecimiento de la ontología . . . . .	40
2.5. Ejemplo de característica simple . . . . .	42
2.6. Ejemplo de característica simple medible . . . . .	43
2.7. Ejemplo de característica compleja . . . . .	44
3.1. DERO: Descripción del dominio. Procesos involucrados en el Modelo de Colaboración Socio-a-Socio . . . . .	64
3.2. DERO: Descripción del dominio. Elementos de la relación colaborativa entre los socios de negocio . . . . .	65
3.3. DERO: Descripción del dominio. Ejemplo de la información intercambiada para acordar sobre un programa de provisión a nivel de PMP . . . . .	65
3.4. DERO: Descripción del dominio. DNE basado en XML . . . . .	66
3.5. DERO: Descripción del dominio. Escenario de motivación: Procesamiento del contenido de un DNE . . . . .	69
3.6. DERO: Descripción del dominio. Escenario de motivación: Traducción del contenido de un DNE . . . . .	70
3.7. Ontología DNE Base . . . . .	72
3.8. Una porción de la ontología DNE Enriquecida con la representación de la entidad socio de negocio . . . . .	79
3.9. Una porción de la ontología DNE Enriquecida con la representación de la entidad programa de provisión . . . . .	81
3.10. Una porción de la ontología DNE Enriquecida con la representación de la entidad producto . . . . .	82
3.11. Una porción de la ontología DNE Enriquecida con la representación de las características cantidad y precio . . . . .	86
3.12. Ontología DNE Enriquecida . . . . .	88
3.13. Características de la entidad producto que no son entidades . . . . .	90
3.14. Entidades y características relacionadas a la entidad producto . . . . .	91
3.15. Características medibles . . . . .	91
3.16. Representación de la entidad producto en la ontología DNE Base . . . . .	97

3.17. Ontología Privada . . . . .	98
3.18. Representación de la entidad producto en la ontología DNE Contextualizada, uso previsto industria de envases . . . . .	100
3.19. Representación de la entidad producto en la ontología DNE Contextualizada, usos previstos industrias de envases y láctea . . . . .	102
4.1. Sinopsis del <i>framework</i> propuesto . . . . .	115
4.2. Porción de la Ontología Base (OB) para el <i>XML Schema Planing Schedule</i> (XMLS-PS) del estándar OAGIS . . . . .	117
4.3. Ejemplo de una Ontología de Especificación de Contexto (OEC) para el Proceso Colaborativo Programa Maestro de la Producción (PC-PMP), características temporales . . . . .	123
4.4. Ejemplo de una Ontología de Especificación de Contexto (OEC) para el Proceso Colaborativo Programa Maestro de la Producción (PC-PMP), características espaciales . . . . .	125
4.5. Ejemplo de una Ontología de Especificación de Contexto (OEC) para el Proceso Colaborativo Programa Maestro de la Producción (PC-PMP), característica producto . . . . .	126
4.6. Ejemplo de una Ontología Enriquecida (OE) del <i>XML Schema PlanningSchedule</i> (XMLSPS) para el Proceso Colaborativo Programa Maestro de la Producción (PCPMP) . . . . .	128
4.7. Porción de una ontología privada de la industria láctea . . . . .	131
4.8. Ontología Base (OB) . . . . .	132
4.9. Ontología de Especificación de Contexto (OEC) para la industria láctea (IL), característica capacidad de un material de empaque . . . . .	136
4.10. Ontología de Especificación de Contexto (OEC) para la industria láctea (IL), característica tamaño de un material de empaque . . . . .	137
4.11. Ontología de Especificación de Contexto (OEC) para la industria láctea (IL), característica embalaje de un material de empaque . . . . .	138
4.12. Una porción de la Ontología Enriquecida . . . . .	140
4.13. Elementos de la Ontología Enriquecida que modelan productos . . . . .	142
4.14. Características de productos modeladas como propiedades en la Ontología Enriquecida . . . . .	143
4.15. Entidades relacionadas con productos en la Ontología Enriquecida . . . . .	143
5.1. Entidades de la realidad y sus representaciones . . . . .	149
5.2. Relaciones entre abstracción, modelo, lenguaje de modelado y representación cognitiva de un dominio dado . . . . .	150
5.3. Relaciones entre la representación cognitiva de un dominio, lenguaje de modelado, lenguaje general de representación y un sistema de categorías ontológicas formal . . . . .	151
5.4. Fragmento de la ontología fundacional definida por Guizzardi (2005), extendida con elementos ontológicos definidos por Dockhorn Costa y otros (2006) . . . . .	155
5.5. Meta-ontología de características contextuales . . . . .	159
5.6. Porción de la meta-ontología implementada en Ecore . . . . .	173

5.7. MetaXys: Ontología DNE Base . . . . .	174
5.8. MetaXys: Jerarquía de términos . . . . .	175
5.9. Ontología DNE Base . . . . .	177
5.10. MetaXys: Enriquecimiento de la característica país . . . . .	178
5.11. MetaXys: Nueva representación de la característica dirección . . .	179
5.12. MetaXys: Representación de la dimensión país . . . . .	179
5.13. Representación posible de las características horizonte y fecha . .	180
5.14. MetaXys: Enriquecimiento de la característica horizonte . . . . .	181
5.15. MetaXys: Nueva representación de la entidad programa de provi- sión . . . . .	182
5.16. MetaXys: Representación de la dimensión intervalo calendario . .	184
5.17. MetaXys: Representación de la dimensión física asociada a unidades de medida . . . . .	184
5.18. MetaXys: Nueva representación de la entidad programa de provi- sión . . . . .	185
5.19. MetaXys: Representación de la dimensión intervalo calendario . .	185
5.20. MetaXys: Nueva representación de la entidad ítem . . . . .	186
5.21. MetaXys: Representación de la dimensión intervalo calendario . .	187
5.22. MetaXys: Nueva representación de la entidad ítem . . . . .	187
5.23. MetaXys: Representación de la unidad de medida . . . . .	188
5.24. MetaXys: Nueva representación de la entidad ítem . . . . .	189
5.25. MetaXys: Enriquecimiento de la característica producto . . . . .	190
5.26. MetaXys: Enriquecimiento de la característica capacidad . . . . .	191
5.27. MetaXys: Enriquecimiento de la característica producto . . . . .	192
5.28. MetaXys: Eliminación de propiedades asociadas a la entidad ítem	193
5.29. MetaXys: Nueva representación de la entidad ítem . . . . .	193
5.30. MetaXys: Representación de la unidad de medida . . . . .	194
5.31. MetaXys: Otro uso previsto de la entidad producto: Envase . . . .	195
A.1. Representación gráfica de ontologías . . . . .	205



# Prólogo

En Ciencias de la Computación, las ontologías comenzaron a convertirse en un concepto relevante en la década de 1990, relacionadas sobre todo al trabajo en adquisición de conocimientos. Desde aquel momento hasta el presente, el uso de las ontologías se ha diseminado en diferentes comunidades y con diversos propósitos, de los cuales la interoperabilidad semántica y el intercambio e integración de información son uno de los principales. Algunas de las áreas más importantes y los objetivos con los cuales se utilizan las ontologías se describen en la primera parte del Capítulo 1.

Para lograr la interoperabilidad semántica es necesario que el emisor y el receptor le asignen el mismo sentido a la información transmitida. No basta con tener plena accesibilidad a los datos, los datos accedidos deben poder ser procesados e interpretados correctamente por todas las partes involucradas. Con este objetivo se han realizado propuestas basadas en el uso de una ontología común. Sin embargo, estas propuestas no se pueden aplicar en todos los casos. Existe un importante número de ontologías desarrolladas por distintos grupos, con diferentes conocimientos y guiados por objetivos dispares. Ante la necesidad de un intercambio de información es altamente probable que distintas ontologías, definidas a distintos niveles de detalle y que se solapan sólo en cierta medida, estén involucradas. En estos casos se ha reconocido al *matching* de ontologías como una solución posible.

*Matching* es el proceso de encontrar correspondencias entre los elementos de distintas ontologías, y *alignment* es el resultado de dicho proceso. Tener la capacidad de encontrar un *alignment* correcto entre ontologías es esencial para asegurar la interoperabilidad semántica en entornos abiertos, donde conviven múltiples ontologías heterogéneas. Con este objetivo se han propuesto hasta la fecha varias soluciones al problema de *matching*; sin embargo, los *alignments* que

se obtienen dependen en gran medida de la calidad de la representación de las entidades en las correspondientes ontologías.

Existen propuestas tendientes a mejorar la representación de las entidades en una ontología; aún así, lograr un *alignment* correcto sigue siendo una tarea difícil, más aún tratándose de ontologías que pertenecen a contextos diferentes (entendiendo por contexto las circunstancias en las cuales algo existe o se produce). Cuando la ontología se construye con el propósito de lograr interoperabilidad semántica en el intercambio de información en entornos abiertos, es fundamental que la metodología considere los posibles contextos existentes y cómo ellos afectan la interpretación de las entidades y sus características. Es por ello que en el marco de esta tesis se propone un método que, sin desaprovechar ideas útiles de otras metodologías y métodos existentes, provee una estrategia para descubrir y representar las entidades y sus características, cuya interpretación depende del contexto en el que se las considere. Dicho método se presenta en el Capítulo 2. La ontología que se obtiene de la aplicación del método recibe el nombre de ontología contextualizada. También en el mencionado capítulo se presentan algunas medidas comparativas que se utilizan para el análisis de la calidad de ontologías y se proponen otras en base a los principios de diseño seguidos en el método propuesto.

En el Capítulo 3 se realiza una validación empírica del método presentado en el Capítulo 2, aplicándolo a un caso de estudio en el dominio empresarial. Luego se compara la calidad de la ontología obtenida (contextualizada) respecto de la ontología base de la cual se parte, aplicando las medidas descritas en el Capítulo 2, y se muestra cómo afecta al *alignment* el disponer de ontologías contextualizadas.

El modelado de las entidades y sus características contextuales requiere, como primera medida, la identificación de los posibles contextos y sus características. En el Capítulo 4 se presenta un *framework*, cuyo objetivo es dar soporte en la tarea de descripción de contextos complejos a partir del reuso de las ontologías de los contextos que los componen. La propuesta se basa en la idea de representar las características de contextos simples haciendo uso de ontologías existentes. Luego, las características de los contextos complejos se representan referenciando elementos de ontologías de contextos simples y complejos. También en el mismo capítulo se muestra cómo se puede utilizar el *framework* en el proceso

de enriquecimiento del método propuesto en el Capítulo 2, aplicándolos juntos a un caso de estudio, y se realiza una breve discusión sobre posibles herramientas para la implementación del *framework*.

Para poder explicitar las características contextuales de las entidades en una ontología se necesita contar también con un lenguaje de modelado para representarlas de forma concisa, completa y sin ambigüedades. Este es el objetivo del Capítulo 5. Inicialmente se muestra la necesidad de un lenguaje que refleje las leyes generales que describen la realidad. Con este objetivo, el lenguaje propuesto toma como referencia una ontología fundacional que comprende un sistema de categorías del mundo real. En el mencionado capítulo se describe la sintaxis abstracta del lenguaje y la primera versión de la herramienta MetaXys que lo implementa. También se muestra cómo se puede enriquecer la representación de entidades en una ontología existente mediante la herramienta.

Finalmente, el Capítulo 6 presenta las conclusiones de la tesis y los trabajos futuros a ser encarados.

Los resultados parciales del trabajo realizado en esta tesis han sido divulgados, o están en vías de publicación, a través de las siguientes publicaciones:

Trabajos en revistas internacionales:

- Rico, Mariela; Caliusco, María Laura; Chiotti, Omar y Galli, María Rosa. «A method for semi-automatic development of a domain ontology from structured and semi-structured data sources». *Expert Systems*. Aceptado con cambios para una segunda revisión.
- Rico, Mariela; Taverna, María Laura; Caliusco, María Laura; Chiotti, Omar y Galli, María Rosa (2009). «Adding semantics to electronic business documents exchanged in collaborative commerce relations». *Journal of Theoretical and Applied Electronic Commerce Research*, **4(1)**, pp. 72–90.

Trabajos en capítulos de libro:

- Rico, Mariela; Caliusco, María Laura; Galli, María Rosa y Chiotti, Omar (2009). «Towards semantic interoperability between information systems». En: Vijayan Sugumaran y Jon Atle Gulla (Eds.), *Applied semantic web*

*technologies: Using semantics in intelligent information processing*, en edición. Taylor & Francis.

- Rico, Mariela; Caliusco, María Laura; Galli, María Rosa y Chiotti, Omar (2011). «An ontology-based framework to semantically describe XML-based business documents». En: Ejub Kajan (Ed.), *Electronic business interoperability: Concepts, opportunities, and challenges*, pp. 457–481. IGI Global.

Trabajos en *proceedings* de congresos:

- Rico, Mariela; Caliusco, María Laura; Reynares, Emiliano; Chiotti, Omar y Galli, María Rosa (2010). «A strategy for analyzing and enriching domain ontologies». En: *XVI Congreso Argentino de Ciencias de la Computación, CACIC 2010. Workshop Innovación en Sistemas de Software. Buenos Aires, Argentina, 18 al 22 de octubre, 2010*. Publicado en CD-ROM.
- Rico, Mariela; Caliusco, María Laura; Taverna, María Laura; Chiotti, Omar y Galli, María Rosa (2009). «A method for making contextual features explicit in domain ontologies». En: Antônio Carlos da Rocha Costa y Gerson Geraldo H. Cavalheiro (Eds.), *XXXV Latin American Informatics Conference CLEI 2009. Pelotas, RS, Brasil, 22 al 25 de septiembre, 2009*. Publicado en CD-ROM.
- Rico, Mariela; Caliusco, María Laura; Chiotti, Omar y Galli, María Rosa (2008). «Toward the improvement of alignment between heterogeneous domain ontologies». En: *XXXIV Conferencia Latinoamericana de Informática CLEI 2008. Santa Fe, Argentina, 8 al 12 de septiembre, 2008*, pp. 579–588.
- Rico, Mariela; Caliusco, María Laura; Taverna, María Laura; Chiotti, Omar y Galli, María Rosa (2008). «Collaborative commerce in supply chains: An approach to facilitate semantic interoperability». En: *Jornadas de Informática Industrial: Cadenas de Abastecimiento, JII 2008 SC, 37 JAIIO. Santa Fe, Argentina, 8 al 12 de septiembre, 2008*, pp. 1–14.
- Rico, Mariela; Taverna, María Laura; Caliusco, María Laura; Galli, María Rosa y Chiotti, Omar (2007). «A comprehensive framework for representing

semantics via context and ontology in the collaborative commerce area». En: Virgilio Almeida y Ricardo Baeza-Yates (Eds.), *Proceedings 5th Latin American Web Congress, LA-WEB 2007. Santiago de Chile, October 31–November 2, 2007*, pp. 110–119. IEEE Computer Society, Los Alamitos, CA.

- Rico, Mariela; Taverna, María Laura; Caliusco, María Laura; Chiotti, Omar y Galli, María Rosa (2007). «Towards semantic interoperability in collaborative e-commerce through contexts and ontologies». En: *Libro de actas: Colloquer Iberoamérica 2007: Collaborative Electronic Commerce Technology and Research Conference. Córdoba, Argentina: Universidad Nacional de Córdoba, 6 al 9 de noviembre, 2007*, pp. 37–46. Fundación Mediterránea, 1ª edición.
- Rico, Mariela; Caliusco, María Laura; Chiotti, Omar y Galli, María Rosa (2007). «Considering subcontexts and multiple contexts for defining semantic relations between complex domains». En: *XXXIII Latin American Conference on Informatics CLEI 2007. San José, Costa Rica, 9 al 12 de octubre, 2007*.
- Rico, Mariela; Caliusco, María Laura; Chiotti, Omar y Galli, María Rosa (2006). «Combining contexts and ontologies: A case study and a conceptual proposal». En: *ECAI 2006 workshop: Contexts and Ontologies: Theory, Practice and Applications. Riva del Garda, Italia, August 28, 2006*, pp. 47–49.
- Rico, Mariela; Caliusco, María Laura; Chiotti, Omar y Galli, María Rosa (2006). «Combining contexts and ontologies: A case study». En: John Debenham (Ed.), *IFIP International Federation for Information Processing, Professional Practice in Artificial Intelligence*, volumen 218, pp. 111–120. Springer, Boston.

### Convenciones Tipográficas

Los estilos de letra que se muestran a continuación se utilizan en esta tesis para distinguir declaraciones de elementos de ontologías, elementos de lenguajes

de programación o elementos de interfaces de usuario, del castellano común. Sin embargo, estas convenciones no se utilizan en tablas o títulos donde tal distinción no es necesaria.

- Los elementos de las ontologías (términos, propiedades, relaciones, instancias y axiomas) se escriben en **sans serif**.
- Los elementos de los lenguajes de ontologías se escriben en **typewriter**.
- Los elementos de interfaces de usuario se escriben en '**este estilo**'.

Así mismo, las palabras que pertenecen a un idioma diferente al castellano, o que no están aceptadas por la Real Academia Española, se escriben en *este estilo*.

# Resumen

Cuando una ontología se desarrolla con el objetivo de ser utilizada en un proceso de intercambio de información entre distintos contextos es necesario que se representen adecuadamente las entidades y características de las entidades que tienen diferente interpretación en dichos contextos. El objetivo principal de esta tesis es proponer un método semiautomático de desarrollo de ontologías que, sin desaprovechar ideas útiles de otras metodologías y métodos, provee una estrategia para descubrir y representar las entidades y sus características, cuya interpretación depende del contexto en el que se las considere. Este método se caracteriza por incorporar una etapa de enriquecimiento en la representación de las entidades, que se compone de dos estrategias: el reuso de ontologías preexistentes, ampliamente aceptadas por la comunidad ontológica; y la aplicación de un número de principios de diseño. Para la primera estrategia se propone un *framework* que, además de ayudar en la creación de ontologías que representan características de contextos, se puede utilizar para enriquecer la representación de las entidades y sus características. Para la segunda estrategia se propone un lenguaje de representación de características contextuales y un prototipo de una herramienta que lo implementa.

El principal aporte de esta tesis consiste en ofrecer al ingeniero ontológico una guía que muestra detalladamente el modo en que debe analizar el problema y los pasos que debe seguir para resolverlo, cuando se enfrenta a una situación de heterogeneidad semántica en el intercambio de información entre diferentes contextos.



# Reconocimientos

Esta tesis es el resultado del esfuerzo, dedicación y colaboración de varias personas e instituciones, a quienes debo mi entero reconocimiento.

Quiero comenzar agradeciendo a mi directora, la Dra. María Rosa Galli, quien creyó en mi capacidad para recorrer con éxito este camino antes que yo misma. Fue ella la que me instó a postularme en la convocatoria a becas para docentes de la Universidad Tecnológica Nacional, Formación de Posgrado, cuando yo ya había desistido de mi interés por la investigación y me había embarcado en el ejercicio de la profesión, en el área privada. A ella le debo, en primer lugar, darme la oportunidad de retornar a una actividad que me atrajo mucho antes de obtener mi título de grado.

También quiero expresar mi gratitud al Dr. Omar Chiotti, mi codirector, quien me introdujo en la investigación. Él me dio la posibilidad de integrarme a un grupo de alumnos y docentes que, movidos por el sólo afán de aprender, de conocer más allá de los contenidos mínimos de una carrera de grado, se reunía a estudiar temas novedosos. Ese conjunto de personas tomó forma como grupo de investigación gracias al trabajo y esfuerzo del Dr. Chiotti, quien convocó también a otros docentes investigadores y los sumó a nuestro incipiente grupo. Ese grupo se llamó GIDSATD (Grupo de Investigación y Desarrollo de Apoyo a la Toma de Decisiones) y sentó una de las bases fundamentales para lo que hoy es el CIDISI (Centro de Investigación y Desarrollo de Ingeniería en Sistemas de Información), lugar donde actualmente desarrollo mis tareas de investigación.

En suma, las discusiones, los distintos puntos de vista, la orientación, la motivación y el apoyo permanentes, los recursos materiales a su alcance, la lectura y corrección de esta tesis son sólo algunos de los invaluables aportes de mis directores.

Otras dos personas jugaron un papel fundamental en el desarrollo de las

ideas presentadas en esta tesis: la Dra. María Laura Caliusco y la Magister María Laura Taverna. Con Ma. Laura Caliusco trabajamos estrechamente los últimos seis años. Ella fue una influencia muy valiosa. Laura es dueña de un espíritu emprendedor y tiene la capacidad de ver siempre el lado positivo de las cosas y saber sacar provecho de resultados adversos. María Laura Taverna siempre fue fuente de una visión distinta y enriquecedora, fundamental en una actividad interdisciplinaria como ésta. Estoy en deuda con ambas por todos estos años de amistad y trabajo conjunto.

Agradezco también a las ingenieras Gabriela García y Estefanía Reidel por su esfuerzo y dedicación para que la propuesta pudiera contar con la primera versión del prototipo que implementa el lenguaje general de representación de características.

Los integrantes del CIDISI, de una u otra forma, también hicieron su aporte para que esta tesis pudiera concretarse. Porque es un grupo humano invaluable, capaz de crear un ambiente de trabajo ideal, donde siempre se encuentran personas dispuestas a intercambiar ideas, compartir conocimientos o simplemente brindar el apoyo emocional tan necesario en esta tarea. Mi reconocimiento especial a todas las “brujas”, con quienes compartimos la difícil tarea de mantener un equilibrio entre la necesidad de lograr tanto el desarrollo profesional como el personal.

También quiero agradecer el respaldo económico de instituciones como la Universidad Tecnológica Nacional, que me otorgó una beca de posgrado para docentes de la UTN, y la Facultad Regional Santa Fe que me brindó el espacio físico y recursos tecnológicos disponibles para facilitar mi tarea. Mi reconocimiento también a otras instituciones que brindaron su aporte económico mediante subsidios a los siguientes proyectos:

- *Estrategias y Herramientas para la Gestión de la Semántica de la Información en las Organizaciones*. Ente financiador: Secretaría de Ciencia y Tecnología del Rectorado de la Universidad Tecnológica Nacional. Período: 01 de enero de 2010 al 31 de diciembre de 2013.
- *Tecnología de Información para el Desarrollo de Procesos de Negocio Colaborativos*. Ente financiador: Agencia Nacional de Promoción Científica y Tecnológica. Período: 22 de abril de 2009 al 21 de abril de 2012.

- *TI para Desarrollar Procesos de Integración que Implementan Procesos Colaborativos*. Ente financiador: Consejo Nacional de Investigaciones Científicas y Técnicas. Período: junio de 2009 a junio de 2011.
- *Modelo y herramientas para la gestión semántica de información y conocimiento distribuido*. Ente financiador: Agencia Nacional de Promoción Científica y Tecnológica. Período: 1 de abril de 2007 al 31 de marzo de 2010.
- *Sistema de Información Soporte de Procesos de Negocio Colaborativos*. Ente financiador: Secretaría de Ciencia y Tecnología del Rectorado de la Universidad Tecnológica Nacional. Período: 30 de noviembre de 2006 al 31 de diciembre de 2009.
- *Soporte para el Diseño de Sistemas de información Business-to-Business*. Ente financiador: Consejo Nacional de Investigaciones Científicas y Técnicas. Período: 1 de enero de 2005 al 31 de diciembre de 2007.
- *Sistema de Información para Soporte de Procesos de Negocio Colaborativos de un Modelo P2P para Gestionar la Cadena de Suministros*. Ente financiador: Agencia Nacional de Promoción Científica y Tecnológica. Período: 1 de enero de 2003 al 31 de diciembre de 2007. Inicio de actividades: 1 de febrero de 2005.
- *Sistema de Información para Soporte de Procesos de Negocio Colaborativos de un modelo P2P para gestionar la cadena de suministros*. Ente financiador: Secretaría de Ciencia y Tecnología del Rectorado de la Universidad Tecnológica Nacional. Período: 1 de enero de 2004 al 31 de diciembre de 2006. Inicio de actividades: 1 de enero de 2005.

Sin embargo, no obstante el apoyo y colaboración recibidos en el aspecto académico, la posibilidad de llegar a esta instancia también es gracias a mi familia y amigos, que estuvieron siempre a mi lado. Mi eterno agradecimiento a mis padres, quienes me sostuvieron desde un comienzo haciendo todos los esfuerzos a su alcance para que yo pudiera formarme en una carrera académica, además de apoyarme incondicionalmente durante todo este tiempo y siempre. Así mismo, el sostén del Dr. Héctor Rico Nesa fue fundamental para que yo pudiera cumplir con las responsabilidades académicas asumidas. A Carlos Pauli y Marta Ferrero,

muchas gracias también por su amistad y por ayudarme desde sus respectivos lugares.

Finalmente, agradezco infinitamente a Marcelo su amor y respaldo sin límites, por comprender mis necesidades y por haberme dado dos hijos maravillosos que, siendo tan pequeños y sin tener la posibilidad de elegir, tuvieron que compartir el tiempo de su mamá con todo lo que implica el desarrollo de una tesis. No creo poder retribuir adecuadamente todo cuanto ellos me brindan.

Mariela Rico

Santa Fe, Argentina, Junio 2011

## **Introducción**

Este capítulo describe brevemente el ámbito en el que se enmarca esta tesis. Con este propósito se comienza desambiguando el término “ontología”, se presenta una clasificación de las mismas en el área de las Ciencias de la Computación y se describen los campos de aplicación principales y objetivos con los cuales se las utiliza (Sección 1.1). A continuación se introduce el problema de la interoperabilidad semántica al compartir información entre sistemas que pertenecen a distintos contextos (Sección 1.2) y se describen las propuestas y objetivos planteados en esta tesis para atacarlo (Sección 1.3). El capítulo finaliza con las conclusiones (Sección 1.4).

### **1.1. Ontologías**

Hacia fines del siglo XX y comienzos del XXI, las ontologías se convirtieron en un área importante de investigación (Gómez-Pérez y otros, 2004). Actualmente, las ontologías se utilizan ampliamente con diferentes propósitos (por ejemplo, gestión del conocimiento, procesamiento de lenguaje natural, comercio electrónico, integración inteligente de información, etc.) en comunidades tales como Ingeniería del Conocimiento, Inteligencia Artificial, Ingeniería de Software y la Web Semántica, entre otras (Breitman y otros, 2007; Gómez-Pérez y otros, 2004).

El término ontología fue introducido en Filosofía, en el siglo XIX, por el filósofo alemán Rudolf Gockel, en su *Lexicon Philosophicum*, para distinguir el estudio del “ser” del estudio de varias clases de seres en las Ciencias Naturales.

Como disciplina filosófica, la creación de ontologías se ocupa de proporcionar sistemas de categorías que dan cuenta de una cierta visión del mundo (Breitman y otros, 2007).

En Ciencias de la Computación, las ontologías empezaron a convertirse en un concepto relevante en la década de 1990, relacionadas sobre todo con el trabajo en adquisición de conocimiento (Staab y Studer, 2009). Desde la perspectiva de la Inteligencia Artificial, la definición de ontología citada con mayor frecuencia es la de Grüber (1993):

“Una ontología es una especificación explícita de una conceptualización”.

A partir de esta definición inicial se han desarrollado varias caracterizaciones de ontologías, dando como resultado la siguiente definición de Studer y otros (1998):

“Una ontología es una especificación explícita, formal, de una conceptualización compartida”.

Aquí, “especificación” significa una representación declarativa; la representación es “formal” porque la ontología debe ser procesable por una computadora (Gašević y otros, 2006) y es “explícita” porque todos los elementos se deben definir claramente (Breitman y otros, 2007). “Conceptualización” es una visión abstracta y simplificada del mundo; cada conceptualización se basa en los conceptos, objetos y otras entidades que se supone existen en un área de interés, y las relaciones que existen entre ellos (Gašević y otros, 2006). Por último, “compartida” refleja la idea de que una ontología representa conocimiento consensuado que ha sido acordado por un grupo de personas, generalmente como resultado de un proceso social (Staab y Studer, 2009).

Sin embargo, en Ciencias de la Computación no existe una definición universal de ontología. Una de las razones es el amplio espectro de posibles usos de las ontologías (Breitman y otros, 2007). Diferentes definiciones proporcionan puntos de vista distintos y complementarios de una misma realidad. Algunos autores proveen definiciones que son independientes de los procesos seguidos para construir la ontología y de su uso en las aplicaciones, mientras que otras definiciones están influenciadas por su proceso de desarrollo (Gómez-Pérez y

otros, 2004). Una revisión de algunas de estas definiciones y las relaciones existentes entre ellas se puede encontrar en (Gómez-Pérez y otros, 2004) y (Gašević y otros, 2006). Un enfoque más formal para la definición de ontología se da en (Guarino y otros, 2009).

Independientemente de la definición adoptada, es importante comprender que las ontologías se utilizan para describir una gran variedad de modelos de un dominio dado. El espectro va desde taxonomías sencillas, como la del *Open Directory Project* (ODP, 2010), hasta modelos muy sofisticados, escritos en variantes de la lógica de primer orden (Breitman y otros, 2007). Sin embargo, lo que hace que un modelo se pueda considerar una ontología es que se trate de un modelo de conocimiento compartido y consensuado por una comunidad (Gómez-Pérez y otros, 2004).

Las ontologías son teorías lógicas que proveen reglas para interpretar explícitamente la sintaxis de lo que se declara. La interpretación de la ontología no se deja librada a las personas que leen los diagramas o a los sistemas que las implementan, se especifica explícitamente de modo que se limiten las posibles interpretaciones de lo que se declara (Euzenat y Shvaiko, 2007).

En el marco de esta tesis, el término ontología se utiliza para referir a un artefacto que representa la semántica de un dominio dado. La ontología se compone de los siguientes elementos:

**Términos:** son palabras, o grupo de palabras, que representan entidades del dominio o características de entidades, que se pueden considerar entidades en sí mismas. Ejemplos de términos son: **Producto**, **Organización**, **Persona**, **LecheEntera**, **Lugar**, etc.

Por convención, cada palabra que designa un término es un sustantivo singular que comienza en mayúsculas. Para los nombres compuestos, las palabras segunda y siguientes comienzan en mayúsculas, sin espacios entre ellas. Esta forma de escritura se conoce como notación *Camel Case* (Svátek y otros, 2009).

**Propiedades:** representan características de entidades en un dominio dado que no se consideran entidades en sí mismas. Por ejemplo, el nombre de una organización, la descripción de un producto, etc.

Por convención, los nombres de las propiedades son sustantivos singulares que comienzan en minúsculas. Para los nombres compuestos, las palabras segunda y siguientes comienzan en mayúsculas, sin espacios entre ellas.

**Relaciones:** son elementos que unen otros elementos de la ontología. Las relaciones se pueden clasificar de acuerdo a los tipos de elementos que relacionan. Existen relaciones entre términos, relaciones entre instancias, pero también existen relaciones entre diferentes tipos de elementos, por ejemplo, entre términos e instancias. Las relaciones entre términos se pueden dividir en jerárquicas (por ejemplo, `Leche esUn Producto`), mereológicas (por ejemplo, `DepartamentoProducción parteDe EmpresaProducción`), equivalencia semántica (sinónimos, por ejemplo, `Artículo equivalenteA Producto`), y particulares (definidas por el desarrollador de la ontología, por ejemplo, `Producto tieneMarca MarcaComercial`), entre otras. Una clasificación de las relaciones se puede encontrar en (Caliusco, 2005).

Generalmente, para nombrar las relaciones se utilizan verbos como palabra principal, y para los nombres compuestos se aplican las mismas combinaciones de mayúsculas y minúsculas que para las propiedades.

**Instancias:** refieren a individuos de una entidad de dominio o de una característica de una entidad en un dominio dado. Un término y sus instancias se unen por la relación `instanciaDe`. Por ejemplo, `LecheEnteraLvTb1L instanciaDe LecheEntera`. Para nombrar las instancias se aplican las mismas convenciones que para los términos.

**Axiomas:** sirven para representar sentencias que son siempre ciertas en un dominio (Grüber, 1993). Los axiomas generalmente se formalizan en algún lenguaje lógico. Habitualmente se utilizan para representar restricciones sobre las entidades de dominio que no se pueden definir formalmente con los otros elementos de la ontología (Gómez-Pérez y otros, 2004). Por ejemplo en lógica descriptiva, el axioma “ $\forall$  tieneTamaño Tamaño” establece que el rango de la relación `tieneTamaño` es sólo el conjunto de las instancias del término `Tamaño`.

### 1.1.1. Categorización de las ontologías

Gómez-Pérez y otros (2004) clasifican a las ontologías de acuerdo a la riqueza de su estructura interna y al tema que conceptualizan. Para el primer grupo se basan en la clasificación propuesta por Lassila y McGuinness (2001), mientras que para el segundo extienden la clasificación de van Heijst y otros (1997).

#### 1.1.1.1. Tipos de ontologías según la riqueza de su estructura interna:

- Vocabularios controlados, es decir, una lista finita de términos. Un ejemplo típico de esta categoría es un catálogo.
- Glosarios, es decir, una lista de términos y sus significados que se indican como declaraciones en lenguaje natural.
- Tesauros, que proporcionan alguna semántica adicional entre los términos. Por ejemplo, proveen información de relaciones de sinónimos, pero no de una jerarquía explícita.
- Jerarquías es-un informales, tomadas de especificaciones de jerarquías de términos tales como la de Yahoo!®. Estas jerarquías no son una subclase estricta.
- Jerarquías es-un formales, que permiten explotar la herencia.
- Jerarquías es-un formales que incluyen instancias del dominio.
- Marcos. La ontología incluye clases y sus propiedades, que pueden ser heredadas por las clases de los niveles más bajos de la taxonomía es-un formal.
- Ontologías que expresan restricción de valor. Estas son ontologías que pueden imponer restricciones sobre los valores que puede tomar una propiedad.
- Ontologías que expresan restricciones lógicas generales. Éstas son las más expresivas. Se pueden especificar restricciones lógicas de primer orden entre los términos usando lenguajes de ontología expresivos.

### 1.1.1.2. Tipos de ontologías según el tema de conceptualización:

- Ontologías de representación del conocimiento, que capturan las primitivas de representación utilizadas para formalizar el conocimiento en virtud de un paradigma de representación del conocimiento dado.
- Ontologías generales, que se utilizan para representar conocimiento de sentido común reutilizable entre dominios. Estas ontologías incluyen vocabulario relacionado con cosas, eventos, tiempo, espacio, causalidad, comportamiento, función, mereología<sup>1</sup>, etc.
- Ontologías de nivel superior, que describen conceptos muy generales y proveen nociones generales a las que todos los términos raíz en ontologías existentes deben ser enlazados.
- Ontologías de dominio, que son reutilizables en un dominio específico dado (medicina, farmacia, ingeniería, derecho, empresa, automóvil, etc.). Estas ontologías proporcionan vocabularios de términos dentro de un dominio y sus relaciones, sobre las actividades que tienen lugar en ese dominio, y sobre las teorías y principios elementales que rigen dicho dominio. Existe un límite bien definido entre las ontologías de dominio y las de nivel superior. Los términos en las ontologías de dominio son por lo general especializaciones de términos ya definidos en las ontologías de nivel superior, y lo mismo podría ocurrir con las relaciones.
- Ontologías de tarea (Guarino, 1998), que describen el vocabulario relacionado con una tarea o actividad genérica (como diagnóstico, programación, venta, etc.) mediante la especialización de los términos de las ontologías de nivel superior. Las ontologías de tarea proporcionan un vocabulario sistemático de los términos empleados para resolver problemas asociados con tareas, que pueden o no pertenecer al mismo dominio.
- Ontologías de dominio-tarea, que son ontologías de tarea reutilizables en un dominio dado, pero no entre dominios. Son independientes de la aplicación.
- Ontologías de método, que dan definiciones de los términos y relaciones

---

<sup>1</sup> Estudio formal de las propiedades lógicas de la relación de parte y todo.

relevantes aplicados para especificar un proceso de razonamiento con el fin de lograr una tarea particular (Tijerino y Mizoguchi, 1993).

- Ontologías de aplicación, que contienen todas las definiciones necesarias para modelar el conocimiento requerido para una aplicación particular. Las ontologías de aplicación a menudo extienden y especializan el vocabulario del dominio y de las ontologías de tarea para una aplicación dada.

### 1.1.2. Campos de aplicación de las ontologías

En Ciencias de la Computación, las ontologías se adoptaron inicialmente en el área de Inteligencia Artificial para facilitar el intercambio de conocimientos y su reutilización (Davies y otros, 2003; Fensel, 2004). Hoy en día, su uso se ha generalizado en varias áreas. Existe una conciencia creciente entre investigadores y desarrolladores respecto a que las ontologías no son sólo útiles en sistemas basados en conocimiento, sino en todos los sistemas de software (Gašević y otros, 2006).

En la actualidad, los campos de aplicación principales de las ontologías incluyen la Web Semántica, gestión del conocimiento, comercio electrónico, multimedia y gráficos, computación en malla (*grid computing*), entornos de computación ubicua (*pervasive computing*), Ingeniería de Software, medicina, ingeniería, química, turismo, biología y bioinformática, entre otros.

En la Web Semántica, las ontologías permiten servicios inteligentes (agentes de información y de búsqueda, filtros de información, integración inteligente de información, etc.) (Gašević y otros, 2006). Se utilizan en portales, en sistemas de recomendación, para dar soporte al procesamiento de lenguaje natural y al aprendizaje en línea o *e-Learning*, entre otros.

En gestión del conocimiento, las ontologías se utilizan principalmente para los siguientes tres propósitos generales (Abecker y van Elst, 2009): (a) soporte en la búsqueda, recuperación, y personalización del conocimiento; (b) servir de base para la recopilación, integración y organización de la información; y (c) soporte en la visualización de conocimiento.

En la integración de información intra-organización, una ontología global provee una vista conceptual independiente de las fuentes de datos que se acceden a través de ella (Calvanese y otros, 2002). En la integración de información

inter-organización existe una red de sistemas autónomos (pares), donde cada uno de ellos puede usar diferente terminología y modelos de metadatos para representar sus datos, aún si refieren a la misma área de interés. En este escenario no es posible basar todos los pares en una ontología global. Por lo tanto es necesario identificar y caracterizar las relaciones entre sus ontologías. Esto es una operación de *matching*. Luego dichas relaciones se pueden utilizar para responder a las consultas (Euzenat y Shvaiko, 2007).

En los servicios web semánticos, las ontologías pueden describir los tipos de servicios, los datos que consumen y proveen, los procesos de los cuales los servicios son parte e, igualmente importante, las relaciones entre todo lo anterior (Duke y Richardson, 2006).

El continuo desarrollo de bases de datos a gran escala crea la necesidad de un marco de conocimiento integrado, en el que los contenidos se deben relacionar entre sí. En una ontología neutral se pueden establecer correspondencias entre diferentes representaciones de las entidades en distintas bases de datos (Gaitanou, 2007). Sin embargo, no siempre es posible adoptar una ontología neutral. Las aplicaciones de diferentes empresas necesitan integrarse y cooperar de forma explícita, a pesar que esas aplicaciones y sus estructuras de datos se han diseñado para lograr distintos objetivos (Rico y otros, 2009). En general, las ontologías en estas empresas reflejan dichas estructuras.

El comercio electrónico, en sus escenarios negocio-a-cliente (*Business-to-Customer*, B2C), negocio-a-negocio (*Business-to-Business*, B2B) y negocio-a-empleado (*Business-to-Employee*, B2E), se basa en el intercambio de información entre partes interesadas utilizando una infraestructura de telecomunicaciones. Las ontologías permiten: (a) realizar consultas integradas, automáticas y simultáneas sobre varios comerciantes (Léger, 2002); (b) traducir diferentes representaciones utilizadas entre las empresas y aprovechar las diversas ontologías de productos y servicios de varios e-negocios (Gaitanou, 2007); (c) personalizar y adaptar la presentación a las necesidades específicas de los dispositivos móviles (Léger, 2002), entre otras cosas.

En los entornos de computación ubícua, varios problemas se pueden manejar con el uso de ontologías, entre ellos: (a) descubrimiento semántico de servicios, que determinan la posibilidad de reemplazar artefactos y la compatibilidad de las conexiones (Krummenacher y otros, 2007); (b) sensibilidad al contexto para que

varias entidades puedan tener una comprensión común del mismo (por “contexto” se entiende información física, por ejemplo, lugar, tiempo, información ambiental, información personal, etc.) (Krummenacher y otros, 2007); (c) interoperabilidad entre diferentes entidades al basarse en conceptos comunes, bien definidos, y compartir un modelo conceptual completo del contexto (Ranganathan y Campbell, 2003).

En Ingeniería de Software, las ontologías facilitan el proceso de desarrollo en las distintas etapas que lo componen. En base a ellas se puede: (a) deducir preguntas para obtener los requerimientos funcionales y no funcionales (Zhao y otros, 2009); (b) tener trazabilidad entre las etapas de análisis y diseño del desarrollo (Gašević y otros, 2009); (c) realizar validación de modelos, aprovechando las transformaciones de modelos y los servicios de razonamiento de las ontologías (Gašević y otros, 2009); (d) mejorar el conocimiento acerca de los patrones de software (Zhao y otros, 2009); (e) mejorar el mantenimiento al permitir rastrear el conocimiento en el que se basó el diseño (Dietrich y Elgar, 2007); (f) mejorar la gestión de las pruebas (Zhu y Huo, 2005), entre otros.

En los sistemas multiagente (*multi-agent systems*, MAS), las ontologías proveen una base semántica para la comunicación y cooperación entre los agentes involucrados (Staab y Studer, 2004). Las ontologías privadas de los agentes proporcionan el marco conceptual que les permite construir un modelo de su entorno, con el cual pueden razonar sobre cómo las acciones que llevan a cabo afectan a su entorno y cómo esos cambios los llevan a alcanzar sus metas; estas ontologías describen el tipo de entidades que los agentes encuentran, las propiedades de esas entidades y las relaciones entre ellas. Las ontologías públicas, que comparten todos los agentes en el MAS, apoyan a los agentes en su interoperabilidad, particularmente en la comunicación y el intercambio de información. Estas ontologías proporcionan una descripción de la infraestructura del MAS (qué tipo de registros emplea, el tipo de protocolos, etc.), y un vocabulario común para que los agentes comprendan el contenido de los mensajes que intercambian (las capacidades de los agentes, información básica sobre cómo contactarlos, el protocolo de interacción, la confiabilidad, la reputación, la seguridad, etc.) (Sycara y Paolucci, 2004). Cuando dos agentes autónomos y diseñados independientemente se encuentran, tienen la oportunidad de intercambiar mensajes, pero pocas posibilidades de entenderse si no comparten

el mismo lenguaje de contenido y la ontología. En este caso es útil ayudar a estos agentes a realizar un *matching* de sus ontologías para traducir sus mensajes, o bien integrar axiomas puente en sus propios modelos (Euzenat y Shvaiko, 2007).

En las aplicaciones geoespaciales, las ontologías se han utilizado para acceder e intercambiar datos e implementar estándares (Breitman y otros, 2007).

En simulación, las ontologías se han utilizado para proveer un vocabulario preciso de simulación y un lenguaje para los componentes de simulación y los dominios en los cuales estos componentes residen (Bell y otros, 2007).

En el área legal, las ontologías se han construido con varios objetivos: recuperación de información, recuperación de la ley, vinculación normativa, gestión del conocimiento y razonamiento jurídico (Casanovas y otros, 2006).

En resumen, y para concluir esta sección, existen muchos campos de aplicación de las ontologías, de los cuales aquí se han mencionado algunos. Fikes (2001) reconoce como claves las siguientes aplicaciones: (a) colaboración entre personas; (b) colaboración entre agentes de software; (c) interoperación; (d) educación; y (e) modelado. Por su parte, Grüninger y Lee (2002) también reconocen el uso de las ontologías para inferencia computacional, y para la reutilización y organización del conocimiento.

Como se puede observar en casi todos los escenarios de aplicación mencionados, la interoperabilidad semántica y el intercambio e integración de información son uno de los propósitos principales de la aplicación de las ontologías. La integración de recursos web, documentos de intranet y bases de datos, así como la cooperación de servicios web y aplicaciones de empresa son escenarios que requieren caracterización semántica de su contenido y/o funcionalidad (Staab y Studer, 2004).

## 1.2. Interoperabilidad Semántica. Ontologías y Contextos

Interoperabilidad semántica es la facultad de atribuir a cada pieza de conocimiento importada la correcta interpretación (Euzenat, 2001). “Correcta interpretación” significa que el receptor utiliza la información transmitida en el mismo sentido según lo previsto por el emisor.

Cuando se trata de evaluar cómo comprende un sistema de información una expresión que proviene de otro sistema de información, existen varios niveles de interoperabilidad posibles (Euzenat, 2001):

1. Codificación: el sistema es capaz de segmentar la representación en caracteres.
2. Léxico: el sistema es capaz de segmentar la representación en palabras (o símbolos).
3. Sintáctico: el sistema es capaz de estructurar la representación en oraciones estructuradas (o fórmulas o afirmaciones).
4. Semántico: el sistema es capaz de construir el significado proposicional de la representación.
5. Semiótico: el sistema es capaz de construir el significado pragmático de la representación (o su significado en contexto).

Lograr la interoperabilidad en cada uno de estos niveles implica lograr la interoperabilidad en todos los niveles anteriores al considerado.

La heterogeneidad semántica es uno de los problemas más importantes y más difíciles de abordar cuando se trata de la interoperabilidad y el intercambio e integración de información (Kashyap y Sheth, 1997). En un sistema distribuido y abierto como la Web Semántica y muchas otras aplicaciones como las mencionadas en la sección anterior, la heterogeneidad no se puede evitar. Diferentes actores tienen diferentes intereses y hábitos, utilizan distintas herramientas y conocimiento, generalmente a diferentes niveles de detalle. Estas diversas razones para la heterogeneidad conducen a distintas formas de heterogeneidad y distintas soluciones posibles (Euzenat y Shvaiko, 2007). Para establecer un intercambio de información eficiente, la interoperabilidad tiene que ser provista tanto a nivel técnico (los tres primeros niveles identificados por Euzenat (2001)) como a nivel informativo. En resumen, el intercambio de información no sólo debe proporcionar la plena accesibilidad a los datos, también requiere que los datos accedidos puedan ser procesados e interpretados por el otro sistema (Wache y otros, 2001).

Con el fin de lograr la interoperabilidad a nivel semántico, el significado de la información que se intercambia tiene que ser entendido a través de los sistemas. Para ello, algunas propuestas se basan en el uso de una ontología común (Firat y otros, 2005; Wache y Stuckenschmidt, 2001). Sin embargo, existen escenarios donde el uso de una única ontología de referencia no es posible. Por ejemplo, en un escenario de relaciones colaborativas vía Internet entre dos empresas, sus aplicaciones necesitan integrarse y cooperar de forma explícita, a pesar que esas aplicaciones y sus estructuras de datos se hayan diseñado para lograr distintos objetivos (Rico y otros, 2009). Otro escenario lo constituyen agentes de software autónomos y diseñados independientemente, que a menudo requieren la colaboración de otros agentes para cumplir con sus objetivos. Debido a que estos agentes generalmente poseen ontologías distintas tienen pocas posibilidades de entenderse (Euzenat y Shvaiko, 2007). En estos casos, la mera utilización de ontologías no reduce la heterogeneidad sino que aumenta los problemas de heterogeneidad a un nivel superior (Euzenat y Shvaiko, 2007).

Para ejemplificar uno de los escenarios, considérense dos empresas que acuerdan establecer una relación colaborativa vía Internet para comerciar sus productos. Ambas empresas tratan con envases: una es una industria dedicada a su fabricación, la otra es una industria láctea que los utiliza como insumos. Sin embargo, ellas se interesan por diferentes aspectos de los envases: el fabricante se preocupa por las características técnicas de su producción, el margen que deja su comercialización, etc. La industria láctea se interesa en los posibles proveedores de envases para sus productos, los tiempos de reposición, etc. A ambas empresas les importan las dimensiones de los envases y los precios. Aún así pueden considerar estas características de manera diferente: los precios pueden incluir impuestos y gastos de envío o no, y se pueden expresar en monedas diferentes; las dimensiones pueden estar dadas en términos de alto, ancho y profundidad de los envases y ser indicadas en centímetros, o pueden especificar una capacidad expresada en unidades de volumen. Más aún, el fabricante puede organizar los envases según su tipo en: envases primarios, los que están en contacto directo con el contenido; envases secundarios, utilizados para agrupar varios envases primarios; y envases terciarios, utilizados para la manipulación a granel, el almacenamiento y transporte. Mientras, la industria láctea puede no clasificar los envases y tenerlos asociados al tipo de producto que envasan. Todavía más importante aún es el uso

que cada empresa hace de los envases: mientras para la empresa que los fabrica representan sus productos, su razón de ser, para la industria láctea los envases son la envoltura de sus propios productos. Como consecuencia de todo lo anterior, estas dos empresas tienen ontologías heterogéneas, donde la entidad envase se modela considerando diferentes características y con distinto grado de detalle.

Debido a un acuerdo de colaboración, estas dos empresas tienen que interactuar. Para hacerlo sin problemas y al no contar con una ontología común de referencia, necesitan encontrar las correspondencias entre las entidades en sus respectivas ontologías. Las correspondencias pueden expresar que lo que se llama producto en la ontología del fabricante representa lo que se llama tipo de envase en la ontología de la industria láctea. El proceso de encontrar estas correspondencias se llama *matching* de ontologías. El resultado de un *matching* de ontologías, llamado *alignment*, es un conjunto de correspondencias entre dos o más ontologías que expresa, con distintos grados de precisión, las relaciones entre las ontologías bajo consideración (Euzenat y Shvaiko, 2007).

### 1.2.1. El problema del *matching* de ontologías

Tener la capacidad de encontrar un *alignment* correcto entre ontologías es esencial para asegurar la interoperabilidad semántica en entornos abiertos, donde conviven múltiples ontologías heterogéneas. Varias soluciones al problema de *matching* se han propuesto hasta la fecha y se las ha clasificado desde diferentes puntos de vista, por ejemplo, bases de datos (Rahm y Bernstein, 2001), sistemas de información (Wache y otros, 2001) e inteligencia artificial (Kalfoglou y Schorlemmer, 2003). Estas soluciones aprovechan distintas propiedades de las ontologías (por ejemplo, las estructuras, las instancias de datos, la semántica o las etiquetas) y utilizan técnicas de diferentes campos (por ejemplo, estadísticas y análisis de datos, aprendizaje de máquina, razonamiento automatizado y la lingüística). Estas soluciones hacen frente a problemas similares (en general, la reducción de los problemas de heterogeneidad terminológica y conceptual), pero difieren en la manera de combinar y explotar sus resultados (Euzenat y Shvaiko, 2007).

En todos los casos, los resultados del proceso de *matching* son altamente dependientes de la calidad de la representación de las entidades en las

correspondientes ontologías.

Con el objetivo de proveer una mejor base para el proceso de *matching*, Stuckenschmidt y van Harmelen (2005) presentan una metodología para enriquecer jerarquías de términos por medio de una red de términos, vinculados a los de la jerarquía a través de las relaciones de sinónimo, hipónimo e hiperónimo. Esta red de términos se basa en una terminología común que debe ser lo suficientemente general como para ser utilizada por todas las ontologías que deben someterse al mismo proceso de *matching*, pero lo suficientemente específica como para poder hacer definiciones significativas. Otra propuesta para mejorar la entrada al proceso de *matching* es la de Sváb-Zamazal y otros (2008), quienes proponen sistematizar la asignación de nombres a los términos.

Si bien estas propuestas conducen a mejorar la representación de las entidades, lograr un *alignment* correcto sigue siendo una tarea difícil, más aún tratándose de ontologías que pertenecen a contextos diferentes –entendiendo contexto como las circunstancias en las cuales algo existe o se produce (Rico y otros, 2007)–. Una ontología se compone de un conjunto de elementos (términos, propiedades, relaciones, axiomas) que describen entidades. La evidencia en la literatura de Inteligencia Artificial, Ciencia Cognitiva, Lingüística, Epistemología y Sociología del Conocimiento muestra que una ontología, como cualquier otra representación explícita del mundo modelado, depende de un conjunto de supuestos implícitos, sin importar cuánto se hayan preocupado sus diseñadores por lograr que sea lo más “objetiva” posible (Bouquet y otros, 2004a). Las técnicas de *matching* aplicadas a estas ontologías no pueden descubrir correspondencias sobre entidades o características que no se han representado explícitamente (Rico y otros, 2008, 2011).

### **1.3. Propuestas y Objetivos de esta Tesis**

#### **1.3.1. Método para enriquecer la representación de las entidades en una ontología**

Las entidades representadas en una ontología, o sus características, pueden tener diferentes interpretaciones y/o representaciones en función del contexto en el que se las considera. Estas características, denominadas características

contextuales, generalmente están implícitas en la ontología, y así su interpretación en un contexto diferente puede conducir a equivocaciones. Las técnicas de *matching* no pueden descubrirlas ni procesarlas. Por lo tanto, ellas deben ser representadas de manera explícita para permitir la interpretación semántica de una entidad en contextos diferentes, y así poder lograr la integración de datos a nivel semántico.

Es objetivo de esta tesis proponer un método sistemático para enriquecer la representación de las entidades en una ontología, explicitando sus características contextuales, de modo que esta ontología permita el intercambio de información en sistemas abiertos donde participan diferentes contextos. Esta ontología enriquecida, obtenida de la aplicación del método propuesto, se denomina ontología contextualizada. Dicho método se presenta en el Capítulo 2.

### **1.3.2. Framework para dar soporte al reuso de ontologías**

El modelado de características contextuales requiere como primera medida la identificación de los posibles contextos y sus características. En el área de modelado semántico de datos la noción de contexto se utiliza para empaquetar información y dar una cierta interpretación de esta información. El contexto actúa como un contenedor de un conjunto de características que afectan el significado de la información (Rico y otros, 2006). En un contexto, se puede distinguir un conjunto de características que describen las circunstancias en las cuales algo existe u ocurre, tales como condiciones geopolíticas, factores que influyen en un proceso, y así sucesivamente.

Esta noción de contexto se puede utilizar para representar particiones del mundo real. Estas particiones se pueden crear teniendo en cuenta algunas dimensiones básicas tales como la temporal, espacial, funcional, estructural, de procedimiento y de comportamiento, entre otras (Acker y Porter, 1994; Theodorakis, 2001). Así, es posible llegar a una clasificación de contexto de acuerdo a diferentes dimensiones. A su vez, los contextos pueden ser simples, cuando se basan en una única dimensión, o complejos, cuando se basan en más de una dimensión.

El desarrollo de ontologías contextualizadas se vería facilitado si se contara con un *framework* adecuado que le de soporte. Este *framework* debería permitir

integrar ontologías preexistentes facilitando su reutilización y, además, debería dar soporte a la tarea de hacer explícitas las características contextuales de las entidades. Para ello debería considerar contextos simples y complejos. Estos contextos complejos deberían poder referenciar a contextos simples en su definición. La presente tesis tiene también como objetivo el desarrollo de un *framework* adecuado para esta tarea, el cual se presenta en el Capítulo 4.

### 1.3.3. Lenguaje de representación de características contextuales

Cuando no es posible reutilizar ontologías para modelar características contextuales, es necesario contar con una herramienta que permita representarlas de manera apropiada. Existen herramientas para el modelado de ontologías ampliamente difundidas en la comunidad de Ingeniería Ontológica, como es el caso de Protégé (Gennari y otros, 2003; Knublauch, 2003; Noy y otros, 2000), WebODE (Arpírez Vega y otros, 2003) y OntoEdit (Sure y otros, 2002), entre otras. Si bien es posible utilizar estas herramientas, ellas no permiten la incorporación automática de todos los elementos ontológicos requeridos para modelar las características contextuales. De esta forma surge como otro objetivo de esta tesis el desarrollo de una herramienta que permita modelar tanto características contextuales simples como complejas, automatizando la incorporación de todos los elementos ontológicos necesarios para tal representación. El lenguaje de representación de características contextuales y un prototipo de una herramienta que lo implementa se presentan en el Capítulo 5.

## 1.4. Conclusiones

La interoperabilidad semántica y el intercambio e integración de información son una de las aplicaciones más frecuentes de las ontologías en la actualidad. Para lograr estos propósitos existen dos estrategias bien diferenciadas: proveer una ontología común o de referencia, a la cual todas las partes interesadas deben adherir, o realizar un *matching* entre las ontologías involucradas. Esta última estrategia es la que se aplica con mayor frecuencia en entornos abiertos, donde conviven múltiples ontologías heterogéneas, por la dificultad que conlleva lograr

una ontología común.

Con el objetivo de realizar el *matching* de ontologías se han propuesto varias soluciones hasta la fecha. Sin embargo, los resultados obtenidos dependen en gran medida de la calidad de la representación de las entidades en las correspondientes ontologías.

Las propuestas tendientes a mejorar la representación de las entidades tienen una carencia importante: ellas no consideran que las ontologías involucradas se pueden utilizar para interoperar entre distintos contextos. Las características propias de cada contexto afectan la manera en la que se interpretan las entidades y sus características representadas en las ontologías. Más aún, generalmente la representación de las entidades depende de un conjunto de supuestos implícitos, que las técnicas de *matching* no pueden inferir. Es por ello que se necesita contar con metodologías y herramientas que permitan descubrir y representar adecuadamente las entidades y sus características en las ontologías que se utilizan para realizar una comunicación entre diferentes contextos. Esto define el objetivo principal de esta tesis.

En el siguiente capítulo se presenta un método sistemático para enriquecer la representación de las entidades en una ontología.



## **Método para Enriquecer la Representación de las Entidades en una Ontología**

En este capítulo se propone un método para enriquecer la representación de las entidades en una ontología. Luego de una breve introducción (Sección 2.1), se describe el ámbito de aplicación y el objetivo del método propuesto (Sección 2.2). A continuación se presenta una visión general del método y se describe en detalle cada uno de los procesos que lo componen (Sección 2.3). Posteriormente se presentan algunas medidas comparativas para el análisis de la calidad de ontologías y se proponen otras, a la luz de los principios de diseño seguidos en el método propuesto (Sección 2.4). El capítulo finaliza con las conclusiones (Sección 2.5).

### **2.1. Introducción**

Como lo indican Sure y otros (2009), las ontologías constituyen activos valiosos que poco a poco, pero de manera continua, van ganando reconocimiento y uso en un conjunto amplio de disciplinas, como se describió en el Capítulo 1 de esta tesis. Sin embargo, su creación y gestión no son tareas sencillas. La Ingeniería Ontológica es la disciplina que estudia los principios, métodos y herramientas para iniciar, desarrollar y mantener ontologías. Una de las áreas más importantes de esta disciplina es la que trata con el proceso y los aspectos metodológicos, es decir, cómo proporcionar directrices y asesoramiento a los (potenciales) desarrolladores de ontologías.

Desde mediados de 1990 se han venido desarrollando metodologías y métodos para construir ontologías, ya sea desde cero, mediante la reutilización y reingeniería de ontologías preexistentes, mediante un proceso de mezcla de ontologías (*ontology merging*) o mediante el uso de un enfoque de aprendizaje de ontologías (*ontology learning*) (Gómez-Pérez y otros, 2004).

Estas metodologías y métodos se pueden clasificar en dos grandes grupos. El primero es el grupo de metodologías basadas en la experiencia, representado por la metodología de Grüninger y Fox (1995), definida en el marco del proyecto TOVE (*TOronto Virtual Enterprise*), y la metodología de Uschold y King (1995) –extendida por Uschold y Grüninger (1996)–, basada en la experiencia de desarrollar la ontología de empresa (*the Enterprise Ontology*). El segundo es el grupo de metodologías que proponen un conjunto de actividades para desarrollar ontologías en función de su ciclo de vida y el refinamiento del prototipo, tal como la metodología METHONTOLOGY (Gómez-Pérez y otros, 2004), el método de desarrollo de ontología 101 (Noy y McGuinness, 2001) y la metodología definida por Brusa y otros (2008). Generalmente, el primer grupo de metodologías es apropiado cuando los propósitos y requisitos de la ontología son claros; mientras que el segundo grupo es útil cuando el entorno es dinámico y difícil de entender y los objetivos no están claros desde el principio (Cristani y Cuel, 2004). Un resumen de las principales metodologías y métodos propuestos se puede encontrar en (Gómez-Pérez y otros, 2004).

Estas metodologías y métodos se han diseñado para cumplir los requerimientos de diversos campos de aplicación específicos. Ellos proporcionan una guía para identificar las entidades y relaciones relevantes en el área de interés. La representación de estas entidades es altamente dependiente de los usos que se intenta dar a la ontología y las características propias del área de aplicación quedan implícitas en ella. Cuando la ontología se construye con el propósito de lograr interoperabilidad semántica en el intercambio de información en entornos abiertos, es fundamental que la metodología considere los posibles contextos existentes y cómo ellos afectan la interpretación de las entidades y sus características. Es por ello que en el marco de esta tesis se propone un método que, sin desaprovechar ideas útiles de otras metodologías y métodos existentes, provee una estrategia para descubrir y representar las entidades y sus características, cuya interpretación depende del contexto en el que se las considere.

## 2.2. **Ámbito de Aplicación y Objetivo**

El método aquí propuesto es útil cuando existe el requerimiento de intercambiar información entre distintos contextos y no es posible imponer una ontología global para todos ellos.

En el proceso de intercambio de información entre contextos distintos, se pueden presentar diferentes tipos de conflictos de heterogeneidad a nivel ontológico. En la literatura se pueden encontrar diversas clasificaciones de tipos de heterogeneidad de ontologías (Benerecetti y otros, 2001; Bouquet y otros, 2004a; Hameed y otros, 2004; Kashyap y Sheth, 1997; Klein, 2001; Wache y otros, 2001). En particular, Euzenat y Shvaiko (2007) consideran las siguientes: sintáctica, terminológica, conceptual y semiótica.

- La heterogeneidad sintáctica se produce cuando dos ontologías no están expresadas en el mismo lenguaje, por ejemplo, OWL versus KIF.
- La heterogeneidad terminológica se produce debido a variaciones en los nombres para referirse a las mismas entidades en diferentes ontologías. Esto puede ser causado por el uso de diferentes lenguajes naturales, por ejemplo **Report** versus **Informe**; diferentes sub-lenguajes técnicos, por ejemplo **Informe** versus **Memo**; diferente ortografía aceptable, uso de abreviaturas, etc., por ejemplo **Documento de Negocio Electrónico** versus **DNE**; o el uso de sinónimos, por ejemplo **Informe** versus **Documento**. También puede ocurrir que se utilice la misma palabra en diferentes ontologías para referirse a distintas entidades, por ejemplo **Reporte** para referir a un informe versus **Reporte** para referir a una noticia.
- La heterogeneidad conceptual se presenta por las diferencias en la representación del mismo dominio de interés. Estas diferencias se pueden agrupar en dos clases principales (Euzenat y otros, 2007):
  - Las diferencias metafísicas tienen que ver con la forma en la cual el mundo se descompone en piezas, es decir, qué entidades, propiedades y relaciones se representan en la ontología. Benerecetti y otros (2001) identifican tres razones importantes por las cuales estas diferencias se producen:

- ★ Las diferencias de cobertura se producen cuando dos ontologías describen diferentes dominios, posiblemente superpuestos, al mismo nivel de detalle y desde una única perspectiva; por ejemplo, dos ontologías pertenecientes a diferentes departamentos de una empresa, una al departamento de recursos humanos, la otra al departamento de mercadotecnia, y que contienen información sobre ventas.
- ★ Las diferencias de granularidad se producen cuando dos ontologías describen el mismo dominio desde la misma perspectiva, pero a distinto nivel de detalle; por ejemplo, dos ontologías que contienen información de ventas, una de ellas describiendo ventas mensuales por familia de productos, la otra describiendo las ventas diarias por ítem.
- ★ Las diferencias de perspectiva se producen cuando dos ontologías describen el mismo dominio, pero desde un punto de vista diferente; por ejemplo, dos ontologías que describen la información de ventas, una de ellas desde la perspectiva de la dimensión temporal, la otra desde la perspectiva de la dimensión geográfica.
- Las diferencias epistemológicas tienen que ver con las afirmaciones que se hacen sobre las entidades. Diferentes ontologías pueden contener distintas afirmaciones (a veces contradictorias) acerca de las mismas entidades. Por ejemplo, la entidad cliente se representa en una ontología por un término que es subtérmino de otro que representa personas (**Cliente esUn Persona**); mientras que en otra ontología, la misma entidad cliente se representa por un término que es subtérmino de otro que representa organizaciones (**Cliente esUn Organización**).
- La heterogeneidad semiótica se refiere a cómo las personas interpretan a las entidades. Generalmente, las personas interpretan a las entidades con respecto a su uso, aunque dichas entidades tengan exactamente la misma interpretación semántica. La intención de uso de las entidades tiene un gran impacto sobre su interpretación. Por ejemplo, si el término **Producto** aparece en una ontología de una industria de envases, un sistema de *matching* de ontologías concluiría que es equivalente al término **Producto** en una

ontología de una industria láctea, mientras que una persona probablemente no incurriría en este error.

La heterogeneidad sintáctica se suele abordar en el plano teórico cuando se establecen equivalencias entre las construcciones de los diferentes lenguajes. La heterogeneidad terminológica y algunas de las diferencias que conducen a la heterogeneidad conceptual pueden tratarse por medio de diferentes estrategias de *matching*. Sin embargo, la heterogeneidad semiótica es difícil de detectar por una computadora, y aún más difícil de resolver, porque está fuera de su alcance. En el ejemplo anterior, mientras el propósito del término **Producto** en la ontología de la industria de envases es representar una variedad de diferentes tipos de envases, el propósito de este término en la ontología de la industria láctea es representar leche, queso, yogur, etc. Por lo tanto, el *matching* de entidades que no están destinadas a ser utilizadas en el mismo contexto a menudo está propenso a errores. El método propuesto en esta tesis es particularmente útil para hacer frente al problema de la heterogeneidad semiótica.

Con el fin de representar una entidad, se pueden construir dos conjuntos de características de dicha entidad: uno que varía según el contexto en el que se considera la entidad –características contextuales– y otro en el que no. Ambos conjuntos podrían estar implícitos en la ontología; hacerlos explícitos puede contribuir a una mejor representación de la entidad. Ya que la representación de las características contextuales depende del contexto en el que se considera la entidad y las entidades a menudo están propensas a sufrir problemas de heterogeneidad semiótica, la presente tesis se centra en la representación de las características contextuales de estas entidades.

El objetivo principal del método propuesto es el desarrollo de una ontología para permitir el intercambio de información entre contextos diferentes. Para ello se propone mejorar la representación de las entidades, haciendo explícitas sus características contextuales en la ontología. En la sección siguiente se describen los procesos que componen el método propuesto.

## 2.3. Descripción General del Método

El método consta de seis procesos: Identificación de los requerimientos de la ontología, Generación de una ontología base, Formalización de la ontología,

Verificación de la ontología, Enriquecimiento de la ontología y Validación de la ontología, que se muestran en la Figura 2.1. Esta figura, y las subsiguientes referidas al método, se basan en el Meta-Modelo de Ingeniería de Procesos de Software y Sistemas, versión 2.0 (*Software & Systems Process Engineering Meta-Model 2.0, SPEM 2.0*) (Object Management Group, 2008). Cada uno de los procesos tiene varias actividades, requiere de una decisión para finalizar y tiene un resultado específico. Los procesos de Verificación, Enriquecimiento y Validación de la ontología generalmente requieren ciclos iterativos, hasta lograr que la ontología cumpla con los requerimientos del usuario y un conjunto de criterios de diseño. La ejecución de estos procesos está a cargo de un ingeniero ontológico, quien es asistido por expertos del dominio. A continuación se describe cada uno de los procesos y las actividades que los componen.

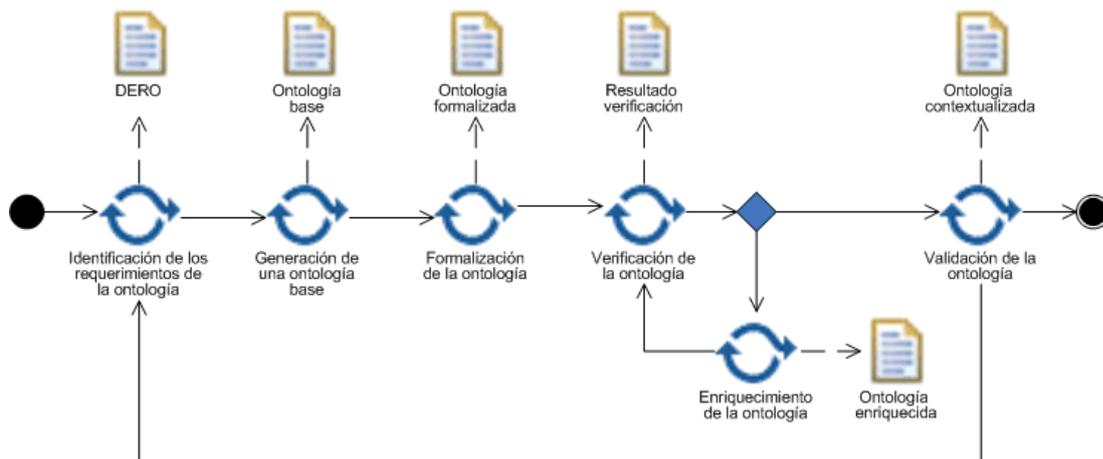


Figura 2.1: Una visión general del método propuesto

### 2.3.1. Proceso 1: Identificación de los requerimientos de la ontología

Una ontología se desarrolla para cumplir un propósito. El objetivo de este proceso es identificar los requerimientos que la ontología debe satisfacer. Para ello, el proceso se divide en tres actividades: Describir el dominio; Determinar el objetivo y alcance de la ontología; e Identificar las principales entidades del dominio, sus relaciones y características, que se muestran en la Figura 2.2. La salida de este proceso es un Documento de Especificación de Requerimientos de la Ontología (DERO). Este documento contiene una descripción del dominio; el

objetivo y alcance de la ontología en lenguaje natural; y una lista de las entidades del dominio, sus relaciones y características, requeridas para el intercambio de información entre distintos contextos.

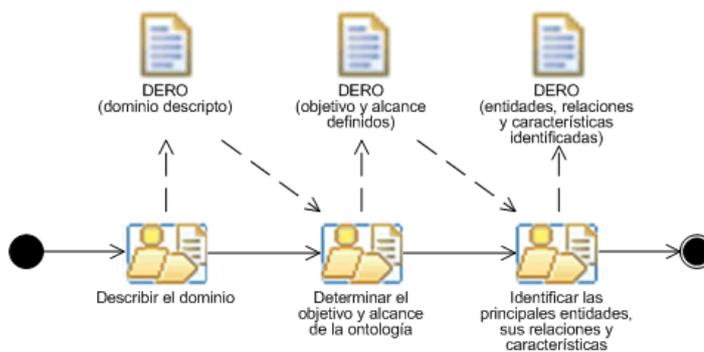


Figura 2.2: Diagrama de actividades del proceso: Identificación de los requerimientos de la ontología

El proceso termina cuando el ingeniero ontológico y los expertos del dominio consideran que los requerimientos de la ontología han sido suficientemente capturados, es decir, que no es necesario seguir analizando más información. Sin embargo, podría darse el caso de que en etapas posteriores se reconozca la falta de información y, en esa situación, se deba volver a este proceso.

#### 2.3.1.1. Actividad 1.1: Describir el dominio

Con el fin de comprender el dominio de aplicación de la ontología, es necesario comenzar con una descripción informal del mismo. Como propusieron Noy y McGuinness (2001) en su metodología, la pregunta principal a responder es: ¿qué dominio va a cubrir la ontología? Para ello, es muy importante la participación de los usuarios a través de reuniones. El propósito de estas reuniones es identificar a los expertos del dominio, las fuentes de información y los contextos involucrados, y adquirir el conocimiento del dominio necesario para llevar a cabo las actividades manuales de los procesos de Verificación y Enriquecimiento de la ontología.

Siguiendo la idea propuesta por Brusa y otros (2008), se podrían utilizar escenarios de motivación y preguntas de competencia como una forma para describir el dominio.

Esta descripción del dominio (que incluye la descripción de las fuentes de información identificadas; la identificación de los expertos del dominio, los

contextos involucrados y las características que los describen; la descripción de los escenarios de motivación y las preguntas de competencia) constituye la salida de esta actividad y se incorpora al DERO.

### **2.3.1.2. Actividad 1.2: Determinar el objetivo y alcance de la ontología**

El objetivo de la ontología especifica para qué será utilizada la misma. Al proporcionar escenarios, es posible comprender la motivación de la ontología propuesta en términos de sus aplicaciones (Grüninger y Fox, 1995; Uschold y Grüninger, 1996). La ontología obtenida con este método debe representar la semántica de la información que se va a intercambiar entre distintos contextos. No obstante, los usuarios son quienes necesitan la ontología y, por lo tanto, los que tienen que especificar su objetivo.

El alcance limita la ontología especificando qué se debe incluir en la representación de las entidades del dominio. Con este objetivo, la siguiente pregunta debe ser formulada: ¿a qué tipo de preguntas debería dar respuesta la información contenida en la ontología? La respuesta a esta pregunta puede cambiar durante el proceso de desarrollo de la ontología, pero siempre limitará su alcance.

Para responder a la pregunta planteada en el párrafo anterior se pueden utilizar los escenarios de motivación, ya que a partir de ellos surgen un conjunto de preguntas que plantean demandas sobre la ontología subyacente. Estas preguntas se pueden considerar como requerimientos que, presentados en forma de preguntas, una ontología debe ser capaz de responder. Grüninger y Fox (1995) y Uschold y Grüninger (1996) denominan a este tipo de preguntas “preguntas de competencia informales”, ya que aún no están expresadas en el lenguaje formal de la ontología. Según estos autores, las preguntas de competencia se deben formular en niveles de complejidad, de manera tal que para responder preguntas más complejas se requiera la solución a preguntas más simples.

Para determinar el alcance de la ontología también puede ser de utilidad identificar y caracterizar a los posibles usuarios de la ontología (Uschold y Grüninger, 1996).

Esta es una actividad esencial en el desarrollo de una ontología, ya que la identificación de las entidades del dominio, sus relaciones y características a ser

representadas, dependerá del objetivo y el alcance de la ontología.

La salida de esta actividad, es decir, la definición del objetivo y alcance de la ontología en lenguaje natural, se suman al DERO.

### **2.3.1.3. Actividad 1.3: Identificar las principales entidades del dominio, sus relaciones y características**

El objetivo de esta actividad es especificar las entidades principales del dominio, sus características y sus relaciones con otras entidades del dominio, que se requieren para el intercambio de información entre contextos diferentes.

Las entidades del dominio, sus características y relaciones se pueden obtener explorando la documentación generada en las actividades anteriores. Por ejemplo, dadas las preguntas de competencia informales, se puede extraer el conjunto de términos usados para expresar la pregunta; éstos serán la base para la determinación de las entidades principales.

En particular, las características también se pueden obtener buscando y representando el conocimiento subyacente que, según los expertos del dominio, se debe poseer a fin de tener una adecuada interpretación del significado de las entidades del dominio en distintos contextos. Por lo tanto, la comunicación con los expertos del dominio se hace imprescindible a lo largo de toda esta actividad.

Dado que el objetivo principal de la ontología a desarrollar es el intercambio de información entre diferentes contextos, las características identificadas deben clasificarse en contextuales y no contextuales. A tal fin, se requiere formular la siguiente pregunta: ¿existen características de entidades, cuyos significados pueden variar en función del contexto en el que se las considera? La respuesta podría cambiar durante el proceso de desarrollo de la ontología.

La lista de las entidades principales del dominio, sus características y sus relaciones, constituye la salida de esta actividad y se agrega al DERO.

## **2.3.2. Proceso 2: Generación de una ontología base**

El objetivo de este proceso es generar una ontología base, que constituye su salida. Para ello, y a partir de la información recabada en el DERO, se puede: realizar una actividad de conceptualización como la descrita por Gómez-Pérez y otros (2004) para su metodología METHONTOLOGY; aplicar una estrategia

de abajo hacia arriba, de arriba hacia abajo, o desde el medio hacia afuera para identificar términos en la ontología (Uschold y Grüninger, 1996); o bien aplicar un proceso semiautomático de aprendizaje de ontologías sobre las fuentes de información identificadas en el proceso previo, al describir el dominio.

La actividad de conceptualización de una ontología, propuesta por METHONTOLOGY, organiza y convierte una vista informal de un dominio en una especificación semiformal, utilizando un conjunto de representaciones intermedias basadas en tablas y gráficos que pueden ser entendidos por los expertos del dominio y los desarrolladores de la ontología. Esta actividad de conceptualización consta de once tareas para identificar el conjunto de términos a ser incluidos en la ontología, sus jerarquías, propiedades y relaciones, y describir los axiomas formales y reglas que se utilizan para comprobar restricciones y para inferir valores de propiedades.

La estrategia de abajo hacia arriba propone identificar primero los términos más específicos y luego generalizarlos en términos más abstractos. Con la estrategia de arriba hacia abajo los términos más abstractos se identifican primero y luego se especializan en términos más específicos. Por último, la estrategia desde el medio hacia afuera recomienda identificar primero los términos básicos centrales, y a continuación especializarlos y generalizarlos según se necesite. Uschold y Grüninger (1996) afirman que esta última estrategia establece un equilibrio en cuanto al nivel de detalle.

Sin embargo, adquirir conocimiento para desarrollar una ontología es un cuello de botella debido a que requiere mucho tiempo y recursos (Gómez-Pérez y otros, 2004). Es así que el aprendizaje de ontologías cobra importancia, ya que su objetivo es el de desarrollar métodos y técnicas que permitan reducir el esfuerzo necesario para el proceso de adquisición de conocimiento.

Con el aprendizaje de ontologías, el desarrollo de una ontología se realiza de forma semiautomática, a partir de distintos tipos de fuentes (Gómez-Pérez y Manzano-Macho, 2003). Según el tipo de fuente, se aplican distintas técnicas para extraer la estructura y el contenido de la ontología. Por ejemplo, para extraer una ontología de un texto se aplican técnicas de análisis de lenguaje natural. Basándose en el tipo de fuente utilizada para el aprendizaje de ontologías, Maedche y Staab (2001) realizan la siguiente clasificación de técnicas de aprendizaje: (a) a partir de texto, (b) a partir de diccionario, (c) a partir

de base de conocimiento, (d) a partir de esquemas semiestructurados y (e) a partir de esquemas relacionales. Una revisión bibliográfica de las técnicas de aprendizaje de ontologías se puede encontrar en (Gómez-Pérez y Manzano-Macho, 2003) y (Drumond y Girardi, 2008).

De todas las propuestas efectuadas para el aprendizaje de ontologías, las que consideran como entrada a los esquemas relacionales y los esquemas semiestructurados son las de mayor aplicabilidad en el intercambio de información. En la actualidad, las organizaciones se están moviendo a páginas web dinámicas, conocidas como sitios de uso intensivo de datos (Fraternali, 1999), para llevar adelante actividades de comercio electrónico, que generalmente se implementan utilizando bases de datos relacionales y documentos basados en XML (*eXtensible Markup Language*).

En caso de aplicar una técnica de aprendizaje de ontologías, las actividades de este proceso dependen de la técnica utilizada. Por su parte, la elección de la técnica de aprendizaje depende del tipo de fuente de datos. Considerando fuentes de datos estructuradas (tales como bases de datos) o semiestructuradas (por ejemplo, HTML, XML, RDF, etc.), algunos ejemplos de técnicas que se pueden utilizar son:

- Para fuentes de datos semiestructuradas:
  - La propuesta de Delteil y otros (2002) tiene por objetivo conocer, del grafo RDF completo, nuevos términos específicos del dominio para enriquecer la ontología de la cual participan las anotaciones RDF.
  - El método de Papatheodorou y otros (2002) tiene por objeto construir taxonomías, utilizando un enfoque de minería de datos a partir de repositorios de dominio escritos en XML o RDF.
  - La propuesta de Volz y otros (2003) trata de capturar la semántica de un XML, o XML Schema, mediante la traducción de símbolos no terminales y terminales en términos y relaciones de la ontología, mediante la aplicación de un conjunto de reglas.
- Para fuentes de datos estructuradas:
  - El método de Johannesson (1994) tiene como fin la traducción de un modelo relacional en un modelo conceptual. El objetivo es producir un

esquema que posea la misma información que el original.

- El método de Kashyap (1999) utiliza los esquemas de bases de datos para construir una ontología, que luego se refina mediante un conjunto de consultas que son de interés para los usuarios de esas bases de datos.
- La propuesta de Stojanovic y otros (2002) trata de construir ontologías *light*<sup>1</sup> a partir de esquemas conceptuales de bases de datos, mediante el uso de un conjunto de reglas para establecer correspondencias de conceptos de la base de datos relacional a términos semánticamente equivalentes en la ontología.

Aunque estas técnicas de aprendizaje generan ontologías poco expresivas, en esta tesis se propone su uso para un rápido desarrollo de una ontología base, a partir de la cual se comienza un proceso iterativo. En los siguientes procesos del método, esta ontología base se analiza comprobando si cumple los requerimientos descritos en el DERO generado en el Proceso 1, y puede ser modificada de acuerdo a los resultados de dicho análisis.

### 2.3.3. Proceso 3: Formalización de la ontología

El objetivo de este proceso es implementar la ontología en algún lenguaje apropiado para transformarla en un modelo computable; por ende su salida es una ontología formalizada. Cuando la ontología base se conceptualiza con la ayuda de un editor de ontologías como WebODE o Protégé por ejemplo, la ontología se puede implementar automáticamente en diferentes lenguajes de representación. En cambio, si en el proceso anterior se aplica una técnica de aprendizaje, y ésta da por resultado sólo un modelo conceptual, es necesario formalizar la ontología en algún lenguaje. En caso contrario, si la técnica de aprendizaje resulta en una ontología formalizada, se debe analizar si el lenguaje utilizado por la técnica es el adecuado a las necesidades y, de ser necesario, realizar las transformaciones que correspondan.

La elección del lenguaje en el cual será implementada la ontología es una decisión clave en el proceso de desarrollo (Gómez-Pérez y otros,

---

<sup>1</sup> La comunidad ontológica distingue entre ontologías *lightweight*, que incluyen términos, taxonomías de términos, relaciones entre términos y propiedades de los términos; y ontologías *heavyweight* que agregan axiomas y restricciones a las ontologías *lightweight* (Gómez-Pérez y otros, 2004).

2004). Esta elección debería basarse en la necesidad de representación del conocimiento y mecanismos de inferencia, y no en las preferencias personales de los desarrolladores de las ontologías. Para ayudar en esta tarea, Corcho y Gómez-Pérez (2000) propusieron un *framework* que permite analizar y comparar la expresividad y capacidad de razonamiento de los lenguajes de ontologías.

Se pueden distinguir dos grandes grupos de lenguajes: los que se basan en Inteligencia Artificial y los que se basan en la Web. En el primer grupo, los paradigmas de representación del conocimiento subyacente se basan en: lógica de primer orden –por ejemplo, KIF (Genesereth y Fikes, 1992)–; marcos combinados con lógica de primer orden –por ejemplo, CycL (Lenat y Guha, 1989), Ontolingua (Farquhar y otros, 1997), OCML (Motta, 1999) y FLogic (Kifer y otros, 1995)–; y lógica descriptiva –por ejemplo, LOOM (MacGregor, 1991)–. También se creó OKBC (Chaudhri y otros, 1998) como un protocolo para acceder a ontologías implementadas en diferentes lenguajes con un paradigma de representación del conocimiento basado en marcos.

Los lenguajes basados en Web explotan las características de la Web. Su sintaxis se basa en lenguajes de marcado existentes, tales como HTML (Raggett y otros, 1999) y XML (Bray y otros, 2008), cuyo propósito no es el desarrollo de ontologías, sino la presentación de datos en el primer caso y el intercambio de datos en el segundo. SHOE (Luke y Heflin, 2000), una extensión de HTML, es un lenguaje que combina marcos y reglas. XOL (Karp y otros, 2000), RDF (Beckett, 2004) y RDF Schema (Brickley y Guha, 2004) se basan en XML. RDF es un lenguaje basado en redes semánticas. RDF Schema extiende RDF con primitivas basadas en marcos. La combinación de RDF y RDF Schema se conoce como RDF(S). OIL, DAML+OIL y OWL son extensiones de RDF(S). OIL (Fensel y otros, 2000) agrega primitivas de representación de conocimiento basadas en marcos y su semántica formal se basa en lógica descriptiva. DAML+OIL (van Harmelen y otros, 2001) agrega primitivas de representación del conocimiento basadas en lógica descriptiva a RDF(S). OWL (Bechhofer y otros, 2004) es el sucesor de DAML+OIL (cubre la mayoría de sus características y renombra la mayoría de sus primitivas). OWL es una combinación de tres sub-lenguajes, contruidos cada uno sobre la base del anterior con expresividad incremental y diseñados para adaptarse a las diferentes comunidades de desarrolladores y usuarios. OWL Lite está diseñado para dar soporte en la construcción de

jerarquías de clasificaciones simples y restricciones simples, pero es relativamente eficiente en el aspecto computacional. OWL DL refleja los fundamentos de la lógica descriptiva de su predecesor, DAML+OIL. OWL DL proporciona la máxima expresividad, pero también garantiza que todas las conclusiones sean computables y terminarán en un tiempo finito; incluye todas las construcciones del lenguaje OWL, aunque impone ciertas restricciones sobre su uso. OWL Full aporta máxima expresividad y la libertad sintáctica de RDF, pero no garantiza la completitud ni la decidibilidad computacional (Gómez-Pérez y otros, 2004).

Además de la expresividad y capacidad de razonamiento del lenguaje elegido, también es importante considerar la existencia de herramientas de desarrollo que soporten dicho lenguaje. Editores de ontologías como WebODE o Protégé por ejemplo, permiten que la ontología se pueda implementar automáticamente en diferentes lenguajes de representación.

Teniendo en cuenta que el objetivo principal del método propuesto en esta tesis es el desarrollo de una ontología para permitir el intercambio de información entre distintos contextos, se considera a OWL como el lenguaje más adecuado para implementar dicha ontología. OWL es una recomendación del *World Wide Web Consortium* (W3C), como resultado del trabajo desarrollado por el *Web Ontology Working Group*, es decir, alcanzó un estado estable en su definición y se impulsa su amplia aceptación por parte de la comunidad de la Web Semántica. OWL está diseñado para ser usado en aplicaciones que necesitan procesar el contenido de la información, y no sólo presentar información a las personas. OWL provee un mecanismo de interpretabilidad del contenido web mejor que los mecanismos soportados por XML, RDF, y RDF Schema, proporcionando mayor poder expresivo junto con una semántica formal (McGuinness y van Harmelen, 2004). El *OWL Working Group* (inactivo al momento de escribirse esta tesis) produjo luego OWL 2, también como una recomendación del W3C, que refina y extiende OWL, pero puede sufrir alguna modificación cuando la versión 1.1 del *XML Schema Definition Language* (XSD) se convierta en recomendación del W3C, ya que OWL 2 utiliza los tipos de datos allí definidos (World Wide Web Consortium, 2009).

#### 2.3.4. Proceso 4: Verificación de la ontología

Verificación refiere a la actividad técnica, llevada a cabo por los desarrolladores, que garantiza que la ontología, sus entornos de software y documentación asociados son correctos con respecto a un marco de referencia, en cada etapa y entre las etapas del ciclo de vida (Gómez-Pérez, 1999). La verificación implica asegurar que las definiciones de la ontología implementan correctamente los requerimientos de la ontología y las preguntas de competencia (Gómez-Pérez y otros, 2004).

El objetivo de este proceso es verificar la ontología base formalizada (la salida del Proceso 3) comprobando si esta ontología cumple un conjunto de criterios de diseño y los requerimientos descritos en el DERO (la salida del Proceso 1), particularmente la salida de la Actividad 1.3: la lista de las entidades principales del dominio, sus características y sus relaciones. La salida de este proceso es un documento que reporta cómo se cumplen esos requerimientos.

La ontología base formalizada, resultante del Proceso 3, podría presentar algunos problemas para satisfacer los requerimientos antes mencionados, por ejemplo porque la estructura de la fuente de datos, a la que se aplicó una técnica de aprendizaje de ontologías (Proceso 2), podría no haber sido creada con el propósito de compartir información entre contextos. Así, la técnica de aprendizaje de ontologías aplicada no puede identificar características contextuales de las entidades.

Además, para cada pregunta de competencia informal, debe haber entidades, características, o relaciones en la ontología, que intuitivamente se requieren para responder la pregunta (Uschold y Grüninger, 1996).

La principal decisión que debe tomarse para la finalización de este proceso es la de si las entidades del dominio requeridas, sus relaciones y características, están representadas en la ontología, y si esa representación es adecuada. Si esto es así, se debe realizar la validación de la ontología (Proceso 6). De lo contrario, se debe realizar el enriquecimiento de la ontología (Proceso 5). Este enriquecimiento puede consistir en la incorporación de elementos faltantes en la ontología como así también una mejora en la representación de las entidades, sus características y relaciones. Estos procesos de Verificación y Enriquecimiento de la ontología pueden requerir varios ciclos hasta que la ontología cumpla los requerimientos del

usuario y criterios de diseño.

El Proceso 4 consta de tres actividades: Identificar la representación de las entidades del dominio en la ontología; Analizar la representación de las entidades del dominio, sus relaciones y características; y Verificar la coherencia lógica de la ontología, que se muestran en la Figura 2.3.



Figura 2.3: Diagrama de actividades del proceso: Verificación de la ontología

#### 2.3.4.1. Actividad 4.1: Identificar la representación de las entidades del dominio en la ontología

El objetivo de esta actividad es determinar si las entidades del dominio, sus relaciones y características, necesarios para el intercambio de información entre distintos contextos, están representados en la ontología base formalizada, y qué elementos de la ontología se utilizan para su representación.

Con la ayuda de los expertos del dominio, el ingeniero ontológico trata de identificar cuáles son los elementos de la ontología que representan cada entidad requerida del dominio, sus relaciones y características. Dependiendo de cómo se ha modelado la fuente de datos, una entidad del dominio podría estar representada por un único término o por un conjunto de elementos de la ontología. Más aún, es posible que algunas entidades del dominio, sus relaciones y características no estén representados en absoluto.

La salida de esta actividad es una lista donde se consigna por cada entidad del dominio, relación y característica, qué elemento o elementos de la ontología la representan. Esta lista integra el documento que reporta el resultado del proceso de Verificación.

#### **2.3.4.2. Actividad 4.2: Analizar la representación de las entidades del dominio, sus relaciones y características**

Con el fin de guiar y evaluar el desarrollo de ontologías se han propuesto diferentes enfoques cualitativos (Gómez-Pérez, 1999; Gruber, 1995) y cuantitativos (Alani y Brewster, 2006; Brewster y otros, 2004; Tartir y otros, 2005). Dado que el objetivo de esta actividad es identificar problemas de representación en la ontología, se propone el uso de enfoques cualitativos. Esta verificación cualitativa tiene que ser realizada por el ingeniero ontológico en colaboración con los expertos del dominio, siguiendo los criterios descriptos a continuación.

Gruber (1995) considera un conjunto fundamental de criterios de diseño para guiar el desarrollo de ontologías. Cuando el propósito de la ontología resultante es el intercambio de conocimiento y la interoperabilidad entre los programas que la usan, estos criterios son: claridad, coherencia, extensibilidad, enfoque de codificación mínima y compromisos ontológicos mínimos.

1. Claridad. Una ontología debería comunicar efectivamente el significado pretendido de los términos definidos, proporcionando definiciones formalizadas con axiomas y completas (con condiciones necesarias y suficientes) siempre que sea posible. Todas las definiciones deberían ser documentadas en lenguaje natural.
2. Coherencia. Una ontología debería ser coherente, es decir, realizar inferencias que sean consistentes con las definiciones. Si una sentencia que se puede inferir de los axiomas contradice una definición o ejemplo dado informalmente, la ontología es incoherente.
3. Extensibilidad. Una ontología debería ser diseñada para anticipar los usos del vocabulario compartido. Deberían poder definirse nuevos términos para usos especiales basados en el vocabulario existente, de forma tal que no se requiera la revisión de las definiciones existentes.
4. Enfoque de codificación mínima. La conceptualización debería ser especificada a nivel de conocimiento, sin depender de una codificación particular a nivel de símbolos. Las elecciones de representación, que se

hacen exclusivamente por la conveniencia de notación o implementación, deben reducirse al mínimo.

5. Compromisos ontológicos mínimos. Una ontología debería requerir el compromiso ontológico mínimo suficiente que permita soportar las actividades de intercambio de conocimiento. Ya que el compromiso ontológico se basa en el uso consistente del vocabulario, el mismo se puede minimizar definiendo sólo aquellos términos que son esenciales para la comunicación del conocimiento.

Por su parte, Gómez-Pérez (1999) considera los siguientes criterios para verificar el contenido de una ontología: consistencia, completitud y concisión.

1. Consistencia refiere a la incapacidad de obtener conclusiones contradictorias de las definiciones de entrada válidas. Una ontología es semánticamente consistente si y sólo si sus definiciones son semánticamente consistentes. Una definición es consistente si y sólo si la definición individual es consistente y no se pueden inferir sentencias contradictorias usando otras definiciones y axiomas. Este criterio es equivalente al de coherencia de Gruber (1995).
2. Completitud refiere a la extensión, grado, cantidad o cobertura para la cual la información en una ontología cubre la información del mundo real (Gómez-Pérez, 1995). La completitud de una definición depende del nivel de detalle acordado en la ontología general. Se puede decir que una definición es completa si nada se ha olvidado. Es decir, si todo lo que se supone debe estar en la definición está en la definición o se puede inferir de los axiomas. Para averiguar si una definición formal es completa, en primer lugar, se busca determinar si la definición cumple con los criterios estructurales para una definición completa –un predicado definido por condiciones necesarias y suficientes (Gruber, 1995)–. En segundo lugar, se determina si el dominio y el rango de las relaciones y las funciones están delimitados exactamente y con precisión. En tercer lugar, se detecta si la generalización/especialización de un término determinado representa con exactitud y precisión las superclases/subclases de una clase dada en el mundo real. Por último, se busca un conjunto completo de atributos en cada

definición de término. Una definición informal escrita en lenguaje natural es completa si expresa el mismo conocimiento que la definición formal.

3. Concisión refiere a si toda la información recogida en la ontología es útil y precisa. La concisión no implica ausencia de redundancias. A veces, un cierto grado de redundancia controlada puede ser útil en las definiciones. Una ontología es concisa si no almacena definiciones innecesarias o inútiles, si no existen redundancias explícitas entre las definiciones, y las redundancias no se pueden deducir usando otras definiciones y axiomas.

Otros criterios que han demostrado ser útiles para la construcción de ontologías son: la representación de conocimiento disjunto y exhaustivo, la minimización de la distancia semántica entre términos hermanos y la estandarización de nombres (Arpírez Vega y otros, 1998).

1. Representación de conocimiento disjunto y exhaustivo. Si el conjunto de los subtérminos de un término es disjunto, es posible definir una descomposición disjunta. La descomposición es exhaustiva si el término padre se define completamente.
2. Minimización de la distancia semántica entre términos hermanos. Los términos similares se agrupan y representan como subtérminos de un término y deberían ser definidos usando las mismas primitivas, mientras que los términos con menos similitud se representan más lejos en la jerarquía.
3. Estandarización de nombres. Para facilitar la comprensión de la ontología se deberían utilizar las mismas convenciones de nomenclatura para nombrar términos relacionados.

Teniendo en cuenta que el objetivo del método propuesto en esta tesis es el desarrollo de una ontología para permitir el intercambio de información entre distintos contextos, la verificación de la completitud de dicha ontología merece una atención especial. Para cumplir con el propósito de la ontología se necesita representar las entidades y características contextuales. Estas características de las entidades se pueden detectar en base a las salidas generadas en procesos anteriores. A tal fin, en esta tesis se propone un conjunto de preguntas guía,

cuyas respuestas pueden ayudar a identificar las características que necesitan ser representadas. Las preguntas son:

- ¿Existe alguna característica no representada de una entidad del dominio que puede ser inferida por una persona, pero no por una computadora? Si la respuesta es sí, ¿esta característica podría ser inferida de forma equivocada en otros contextos distintos del contexto considerado? Si la respuesta es sí, esta característica debe ser representada.
- ¿Las representaciones y los significados de las características contextuales de las entidades, o las entidades del dominio, están completamente representados en la ontología? Si la respuesta es no, deben ser representados.
- ¿Cuáles son las dimensiones utilizadas para especificar características de las entidades? ¿Son las mismas independientemente del contexto en el que se consideran esas características? Si la respuesta es no, ¿están representadas en la ontología? Si la respuesta es no, estas dimensiones deben ser representadas.

Todos estos criterios –los de Gruber (1995), Gómez-Pérez (1999), Arpírez Vega y otros (1998) y las preguntas guía– hacen a la calidad de una ontología y deben tenerse en cuenta al momento de verificar la representación de las entidades de dominio. Estos criterios son una guía para el desarrollo y la verificación de la ontología. Ninguno de ellos se puede medir de manera directa (Vrandečić, 2009).

La salida de esta actividad consiste en incorporar al documento que reporta el resultado del proceso de Verificación, por cada entidad, característica y relación detallada allí, cuáles de estos criterios no se satisfacen.

#### **2.3.4.3. Actividad 4.3: Verificar la coherencia lógica de la ontología**

Corresponde también en este proceso verificar la coherencia lógica de la ontología. Ya que la ontología está formalizada, se pueden utilizar los servicios de un razonador a este fin.

Uno de los principales servicios ofrecidos por un razonador es probar si un término es subtérmino de otro. Al realizar estas pruebas en todos los términos de una ontología, es posible que un razonador calcule la jerarquía de

términos inferida de la ontología. Otro de los servicios comunes que ofrecen los razonadores es la comprobación de coherencia. En base a la descripción (condiciones) de un término, el razonador puede comprobar si es o no posible que el término tenga instancias. Un término se considera incoherente si no puede tener instancias (Horridge y otros, 2004).

El resultado de esta verificación es la salida de esta actividad, que se suma al documento que reporta el resultado del proceso de Verificación.

### **2.3.5. Proceso 5: Enriquecimiento de la ontología**

El objetivo de este proceso es mejorar la representación de las entidades del dominio en la ontología base, de acuerdo con los resultados de la verificación. Las entradas a este proceso son la ontología base formalizada (Proceso 3) y el documento que reporta cómo se cumplen los requerimientos (Proceso 4). La salida es una ontología enriquecida que debe volver a verificarse (Proceso 4).

La principal decisión que debe tomarse para finalizar este proceso es determinar si la ontología enriquecida cumple los requerimientos capturados en el DERO.

El proceso consiste de tres actividades: Añadir los elementos faltantes; Mejorar la representación de las entidades del dominio, sus relaciones y características; y Designar un término puente que refiera a la entidad de dominio, que se muestran en la Figura 2.4. Las dos primeras actividades podrían llevarse a cabo de forma simultánea.

#### **2.3.5.1. Actividad 5.1: Añadir los elementos faltantes**

En el proceso anterior, se generó una lista de las entidades del dominio que no están representadas en la ontología base y, para cada entidad del dominio, una lista de las características y las relaciones con otras entidades del dominio faltantes. El objetivo de esta actividad es crear los elementos de la ontología necesarios para representar esas entidades, características y relaciones faltantes. Para ello se deben tener en cuenta los criterios de diseño discutidos por Gruber (1995), Gómez-Pérez (1999) y Arpírez Vega y otros (1998), las preguntas guía propuestas en esta tesis, y otros principios de diseño que se detallan en la próxima actividad.

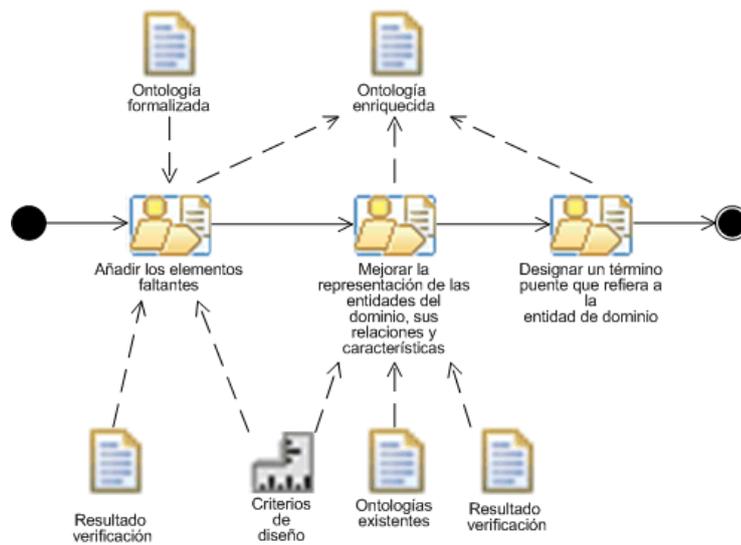


Figura 2.4: Diagrama de actividades del proceso: Enriquecimiento de la ontología

### 2.3.5.2. Actividad 5.2: Mejorar la representación de las entidades del dominio, sus relaciones y características

En el proceso anterior se identificaron las entidades del dominio requeridas, sus relaciones y características, cuyas representaciones no son adecuadas. El objetivo de esta actividad es mejorar esas representaciones. Para ello se proponen dos alternativas de representación: la reutilización de ontologías existentes y la aplicación de principios de diseño.

#### *Reutilización de ontologías existentes*

La comunidad de Ingeniería Ontológica considera beneficiosa la integración de ontologías que poseen una representación bien desarrollada respecto de un cierto tema, permitiendo así que se puedan compartir y reutilizar conocimientos mutuos (Noy y Hafner, 1997). Siguiendo esta idea, se debe contemplar la posibilidad de importar en la ontología formalizada ontologías existentes y ampliamente aceptadas, o porciones de ellas, para enriquecer la representación de las entidades, sus relaciones y características. Ejemplos de tales ontologías son:

- La ontología *OWL-Time* (Hobbs y Pan, 2004) para representar la mayor parte de las entidades y relaciones de tiempo. Esta ontología provee

un vocabulario adecuado para expresar hechos acerca de las relaciones topológicas entre instantes, intervalos y eventos, junto con información sobre duraciones y sobre fechas y tiempos.

- La ontología de Códigos de Países ISO 3166<sup>2</sup> para representar los nombres oficiales de los países tal como se indica en la norma ISO 3166<sup>3</sup>.
- Una porción de una ontología que representa los códigos de monedas que se publican en la norma ISO 4217:2008, como la ontología MLC (*Music Licensing Company Message Suite Standard*)<sup>4</sup> para la representación de monedas y fondos.
- Una porción de una ontología que representa el estándar internacional ISO 80000, sucesor de ISO 31, para representar magnitudes físicas y unidades de medida, como la del Centro de las Naciones Unidas para la Facilitación del Comercio y Transacciones Electrónicas (UN/CEFACT)<sup>5</sup>.

En el Capítulo 4 se presenta un *framework* que es de utilidad en la implementación de esta alternativa de representación, ya que permite relacionar la representación de entidades y sus características con ontologías que representan las características de contextos simples y complejos. Como se describe en el capítulo mencionado, estas ontologías de contextos se pueden basar en ontologías de características primitivas, algunas de las cuales pueden ser las enumeradas en el párrafo anterior.

#### *Aplicación de principios de diseño*

Sin embargo, no siempre es posible la reutilización de ontologías preexistentes. En este caso, en esta tesis se proponen algunos principios de diseño para mejorar la representación de las entidades y sus características.

Para enriquecer la representación de una característica de entidad, se debe comenzar por identificar si esa característica es simple o compleja. Una característica de entidad simple es una cualidad que no conlleva otras

---

<sup>2</sup> <http://www.daml.org/2001/09/countries/iso-3166-ont> (último acceso, 17/02/2011)

<sup>3</sup> [http://www.iso.org/iso/english\\_country\\_names\\_and\\_code\\_elements](http://www.iso.org/iso/english_country_names_and_code_elements) (últ. acc., 15/02/2011)

<sup>4</sup> <http://www.ifpi.org/pcs/dd/pcssDictionary.html> (último acceso, 15/02/2011)

<sup>5</sup> [http://www.unece.org/cefact/codesfortrade/codes\\_index.htm](http://www.unece.org/cefact/codesfortrade/codes_index.htm) (último acceso, 15/02/2011)

cualidades, y se asocia con una representación unidimensional (Guizzardi, 2005). Su representación consta de, al menos, dos elementos en la ontología:

- (a) un término que representa la dimensión, el conjunto de valores posibles;
- (b) una relación entre este término y el término que representa a la entidad caracterizada.

La Figura 2.5 muestra un ejemplo de característica simple. En este caso, la característica país se representa por el término **DimensiónPaís**, que representa el conjunto de valores posibles, y la relación **enPaís** entre los términos **DimensiónPaís** y **Dirección**.



Figura 2.5: Ejemplo de característica simple

Cuando una característica es medible, es decir, su dimensión representa cantidades, esta dimensión tiene una unidad de medida asociada (gramos, metros,  $\text{cm}^3$ , dólares, etc.). Esta unidad de medida afecta la granularidad de la dimensión, pero no su estructura. A su vez, una unidad de medida tiene una dimensión física asociada. Densidad, longitud, frecuencia, masa, tiempo, velocidad, etc., son dimensiones físicas. Sólo las cantidades de la misma dimensión se pueden sumar o comparar. Por ende, una característica medible se representa en la ontología por seis elementos:

- (a) un término que representa la dimensión, el conjunto de valores posibles;
- (b) una relación entre este término y el término que representa a la entidad caracterizada;
- (c) un término que representa la unidad de medida de la dimensión;
- (d) una relación entre este término y el término que representa la dimensión;
- (e) un término que representa la dimensión física de la unidad de medida;
- (f) una relación entre este término y el término que representa la unidad de medida.

La Figura 2.6 muestra la representación de la característica peso. El término `DimensiónPeso`, que representa la representación unidimensional, se relaciona a su unidad de medida a través de la relación `medidoEn`. El término `UnidadDeMedida` representa la unidad de medida de la dimensión peso, cuyos valores posibles serán números reales positivos, independientemente que el peso se exprese en kilogramos o toneladas por ejemplo. El término `DimensiónFísica` representa la dimensión física de la unidad de medida y se relaciona con el término que la representa a través de la relación `tieneDimensión`.

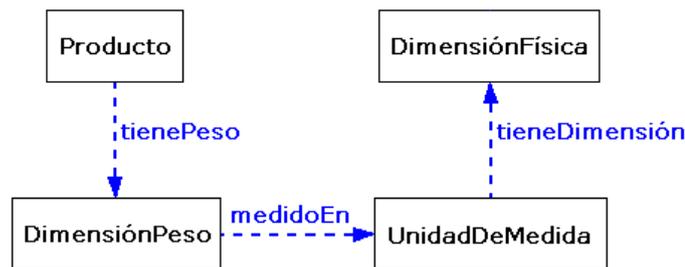


Figura 2.6: Ejemplo de característica simple medible

Una característica de entidad compleja es una cualidad que tiene otras cualidades, y se asocia con un conjunto de dimensiones integrales que se pueden separar de todas las demás dimensiones (Guizzardi, 2005). Una dimensión integral es aquella en la que no es posible asignar un valor a un objeto en una dimensión sin darle un valor en las otras. Por ejemplo, tamaño se puede representar en término de las dimensiones de profundidad, ancho y alto. Estas dimensiones son integrales. Por el contrario, las dimensiones de peso y profundidad se dice que son separables. Con el fin de mejorar la representación de una característica compleja, los siguientes elementos deben estar en la ontología:

- (a) un término que representa el conjunto de dimensiones integrales;
- (b) una relación entre este término y el término que representa a la entidad caracterizada;
- (c) para cada dimensión integral, un término que la representa y una relación entre este término y el término que representa el conjunto de dimensiones integrales;
- (d) para cada término que representa una dimensión integral medible debe estar: la relación con el término que representa la unidad de medida, el

término que representa la unidad de medida de la dimensión, un término que representa la dimensión física de la unidad de medida, y una relación entre este término y el término que representa la unidad de medida.

La Figura 2.7 muestra un ejemplo de característica de entidad compleja. El término **MultidimensiónTamaño** representa el conjunto de dimensiones integrales y está asociado al término que representa a la entidad caracterizada, **Producto**, a través de la relación **tieneTamaño**. Los términos **DimensiónAncho**, **DimensiónProfundidad** y **DimensiónAlto** representan las tres dimensiones integrales que componen la multidimensión, y están asociados al término que la representa a través de las relaciones **definidoPorDimAncho**, **definidoPorDimProf** y **definidoPorDimAlto** respectivamente. En este caso, todas las dimensiones integrales son medibles, por lo cual los términos que las representan están asociados al término **UnidadDeMedida** a través de las relaciones **medidoEn**.

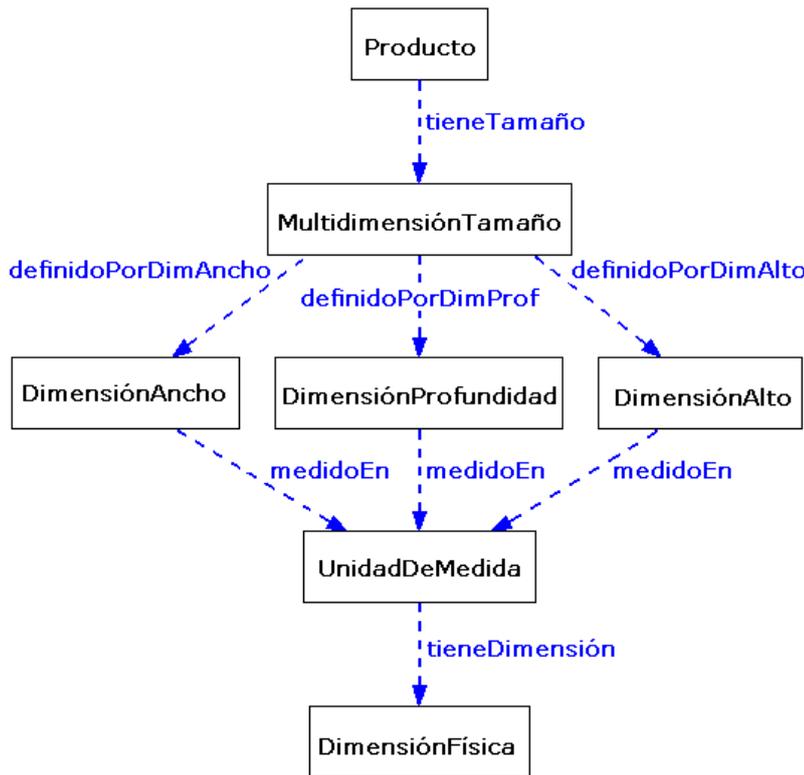


Figura 2.7: Ejemplo de característica compleja

Además, se deben agregar las restricciones de existencia, cardinalidad y axiomas de clausura para garantizar que la multidimensión está compuesta por las respectivas dimensiones integrales. Estas deben ser condiciones necesarias y

suficientes. Otra restricción que debe incorporarse es aquella que impone que las dimensiones integrales deben ser disjuntas. En lógica descriptiva, para el ejemplo de la Figura 2.7, esto es:

$$\begin{aligned}
& \forall \text{ definidoPorDimAncho.DimensiónAncho} \\
& \exists \text{ definidoPorDimAncho.DimensiónAncho} \\
& \quad = 1 \text{ definidoPorDimAncho} \\
& \forall \text{ definidoPorDimProf.DimensiónProfundidad} \\
& \exists \text{ definidoPorDimProf.DimensiónProfundidad} \\
& \quad = 1 \text{ definidoPorDimProf} \\
& \forall \text{ definidoPorDimAlto.DimensiónAlto} \\
& \exists \text{ definidoPorDimAlto.DimensiónAlto} \\
& \quad = 1 \text{ definidoPorDimAlto} \\
& \text{DimensiónAncho} \cap \text{DimensiónProfundidad} \subseteq \perp \\
& \text{DimensiónAncho} \cap \text{DimensiónAlto} \subseteq \perp \\
& \text{DimensiónAlto} \cap \text{DimensiónProfundidad} \subseteq \perp
\end{aligned}$$

En el Capítulo 5 se presenta un lenguaje de representación de características contextuales y una primera versión de una herramienta que lo implementa que facilitan la aplicación de los principios de diseño propuestos aquí.

### 2.3.5.3. Actividad 5.3: Designar un término puente que refiera a la entidad de dominio

Para tener una ontología que pueda ser útil para el intercambio de información entre distintos contextos, un aspecto clave a representar en la ontología son los posibles usos de las entidades de dominio, ya que estos usos dependen del contexto en el que se consideran esas entidades. Los usos previstos de una entidad de dominio en el contexto considerado se deben representar mediante términos, llamados “puente” debido a que permiten vincular diferentes significados y representaciones de la misma entidad de dominio en distintos contextos. Estos términos también se deben interpretar como la representación de características contextuales de la entidad, porque el uso previsto depende del contexto en el que se considera la entidad de dominio. Por lo tanto, es necesario determinar si existe un término que designe el uso previsto de cada entidad de dominio, y si dicho término está ausente se lo debe agregar a la ontología.

Estos términos puente también deben estar relacionados con los elementos que representan la entidad de dominio, cuyo uso previsto ellos representan. Una entidad de dominio podría estar representada por un único elemento o un conjunto de elementos. En el primer caso, se debe agregar una relación entre este elemento y el término puente. En el segundo caso, se debe elegir el término más representativo y, a continuación, se debe agregar una relación entre este término y el término puente. Como el término puente representa una característica contextual de entidad, también debe estar relacionado con el término que representa su dimensión.

Una vez enriquecida la ontología, ésta debe ser verificada (Proceso 4).

### **2.3.6. Proceso 6: Validación de la ontología**

Una vez que la ontología cumple los criterios impuestos en el proceso de Verificación, tiene que ser validada. La validación garantiza que las definiciones de la ontología realmente modelan el mundo real para el cual la ontología fue creada (Gómez-Pérez y otros, 2004). Entonces, validar la ontología significa comprobar si la ontología cumple con los requerimientos que figuran en el DERO, pero ya no desde el punto de vista técnico sino desde el punto de vista de los usuarios.

Durante este proceso, la comunicación con los expertos del dominio es esencial. La principal decisión que debe tomarse para la finalización de este proceso es que la ontología responda a las necesidades de los usuarios, según el uso previsto de la ontología. Los expertos del dominio son quienes toman esta decisión. La ontología que se obtiene como salida de este proceso se denomina “ontología contextualizada”.

Las preguntas de competencia sirven a dos fines. En el proceso de Verificación se utilizan para determinar si la ontología puede responderlas. Una vez que un motor de inferencia puede dar respuestas razonablemente completas y coherentes, a criterio de un experto del dominio, a las preguntas de competencia (consideradas teoremas), el esfuerzo de desarrollo se considera terminado (Obrst y otros, 2007). El proceso de Validación debe comprobar si las preguntas de competencia en realidad plantean las preguntas adecuadas para el propósito de la ontología (de Almeida Falbo, 2004). Si se descubre que las preguntas de

competencia planteadas no son las adecuadas, éstas deben ser reformuladas, haciéndose necesario revisar todo lo hecho desde el proceso de Identificación de requerimientos.

Parte del proceso de Validación se puede realizar en forma automática, si las preguntas de competencia están representadas de manera formal, o semiautomática mediante heurísticos específicos o juicios humanos. A fin de formalizar las preguntas de competencia, se pueden utilizar lenguajes de consulta como RDQL (*RDF Data Query Language*) (Seaborne, 2004) u OWL-QL (Fikes y otros, 2004). RDQL es una implementación de un lenguaje de consulta como SQL para RDF. Éste trata a RDF como datos y provee consultas con patrones de tripletas y restricciones sobre un único modelo de RDF. OWL-QL se diseñó para diálogos pregunta-respuesta entre agentes de software que utilizan OWL. Por lo tanto, OWL-QL es adecuado cuando se necesita llevar a cabo una inferencia en la consulta.

Otros aspectos a evaluar, como el grado de comprensión y la facilidad de uso desde el punto de vista del usuario, son juicios humanos y por lo tanto no automatizables.

## 2.4. Análisis Comparativo de la Calidad de Ontologías

El proceso de construcción de ontologías ha sido tradicionalmente un arte más que un proceso sistemático, a pesar de la existencia de metodologías para apoyar este proceso. La correcta aplicación de las metodologías ayuda a garantizar una cierta calidad en la ontología resultante, aunque la calidad que ofrecen estas metodologías no ha sido hasta ahora medida cuantitativa o cualitativamente. No existen modelos maduros para medir los procesos de construcción de ontologías. Sin embargo, los desarrolladores de ontologías necesitan una manera de evaluar la ontología resultante y, posiblemente, guiar el proceso de construcción y cualquier actividad de refinamiento (Brank y otros, 2005). Con este objetivo, diversos autores (Burton-Jones y otros, 2005; Colomb, 2002; Stvilia, 2007) identifican algunas dimensiones de evaluación de ontologías, las cuales se describen a continuación.

### 2.4.1. Dimensiones de calidad

La calidad no es una propiedad de algo, sino un juicio, de modo tal que debe estar en relación con algún propósito; y generalmente implica el reconocimiento de soluciones de diseño de compromiso (Colomb, 2002).

Las características relevantes para evaluar la calidad de ontologías, utilizadas para interoperabilidad de los sistemas de información, se pueden clasificar en tres dimensiones: sintáctica, semántica y pragmática. Estas tres dimensiones respectivamente responden a las preguntas: ¿la ontología es sintácticamente correcta, una computadora la puede “leer”? ¿La ontología cubre el dominio de interés, la computadora la puede entender? Y ¿la ontología es de utilidad, cumple el propósito para el cual se la desarrolló?

La calidad en su dimensión sintáctica es la corrección sintáctica. Cuando el desarrollo de la ontología está soportado por una herramienta *case*, la dimensión sintáctica es la riqueza y la complejidad de la sintaxis antes que la corrección, ya que la herramienta debe ser capaz de señalar estos errores al ingeniero ontológico. La calidad en su dimensión semántica es cuán bien la ontología refleja su universo de discurso, mientras que la calidad en su dimensión pragmática es cuán útil es la ontología. La evaluación a lo largo de estas dimensiones es, en la práctica, una evaluación de la capacidad para cumplir un propósito. Cuanto más refleja la ontología su universo de discurso, más grande y más compleja tiende a ser y, por lo tanto, más difícil de entender y utilizar. También existen ventajas y desventajas comparativas entre las dimensiones sintáctica y semántica y entre las dimensiones sintáctica y pragmática.

### 2.4.2. Evaluación de las dimensiones de calidad

Como se mencionó en la Sección 2.3.4.2, con el objetivo de evaluar la calidad de las ontologías, diversos enfoques cuantitativos y cualitativos se han considerado en la literatura. En dicha sección se describieron algunos enfoques cualitativos con el propósito de identificar problemas de representación en las ontologías. Respecto a los enfoques cuantitativos, Brank y otros (2005) agrupan las propuestas en base al nivel de evaluación y distinguen los siguientes: nivel léxico, de vocabulario o de datos; jerarquía o taxonomía; otras relaciones semánticas; nivel de contexto o aplicación; nivel sintáctico; estructura, arquitectura, diseño; y propuestas de

criterios múltiples. Algunos de estos niveles se pueden utilizar para evaluar las distintas dimensiones de calidad de una ontología, cuyo propósito es el intercambio de información entre diferentes contextos. A continuación se describen los indicadores más relevantes para cada una de las dimensiones como así también se proponen otros.

#### 2.4.2.1. Evaluación de la dimensión sintáctica

La calidad de una ontología en su dimensión sintáctica se puede evaluar considerando los indicadores de los niveles de jerarquía o taxonomía, otras relaciones semánticas y nivel sintáctico propuestos por Brank y otros (2005).

##### *Jerarquía o taxonomía*

En este nivel, algunos indicadores son:

- Ancho ( $A$ ) es el número promedio de subtérminos en un término (Colomb, 2002; Supekar y otros, 2004), también llamado riqueza de herencia (Tartir y otros, 2005).

Sea  $T$  el número total de términos en una ontología. Sea  $ST_i$  el número de subtérminos directos del término  $i$ . Luego,

$$A = \frac{\sum_{i=1}^T ST_i}{T}$$

Una ontología con un valor bajo de  $A$  sería de naturaleza vertical, lo que podría reflejar que la ontología representa un conocimiento muy detallado; mientras que una ontología con un valor alto de  $A$  sería de naturaleza horizontal, lo que significaría que la ontología representa una amplia gama de conocimientos generales.

- Profundidad ( $PF$ ) es el número promedio de pasos (arcos, caminos) en la jerarquía, desde el término raíz hasta un término hoja (Colomb, 2002), también llamado nivel de detalle promedio o precisión (Stvilia, 2007).

Sea  $TH$  el número total de términos hoja en una ontología. Sea  $P_i$  el número de pasos para llegar desde el término raíz hasta el término hoja  $i$ . Luego,

$$PF = \frac{\sum_{i=1}^{TH} P_i}{TH}$$

- Tamaño es el número de términos hoja (Colomb, 2002).
- Especificidad es el número promedio de instancias asociadas con un término hoja (Colomb, 2002). Se rige por el ancho y la profundidad (Supekar y otros, 2004). Intuitivamente, una ontología más específica es indicativo de una alta calidad de los conocimientos. Lógicamente, este indicador se puede calcular sobre una ontología “poblada”, es decir, una ontología que contiene la información que se va a intercambiar entre los distintos contextos involucrados.
- La complejidad se puede medir por el ancho y profundidad de la ontología, el número total de términos, el número total de relaciones, el número total de atributos y el número total de instancias (Stvilia, 2007; Supekar y otros, 2004).

Una ontología contextualizada debería ser más compleja que otra no contextualizada, ya que justamente el proceso de Enriquecimiento hace explícitas entidades y características contextuales agregando términos, propiedades, relaciones y axiomas a la ontología base.

### *Otras relaciones semánticas*

La ontología puede contener otras relaciones, además de las relaciones jerárquicas. Tartir y otros (2005) proponen medir la Riqueza de las Relaciones ( $RR$ ) como la proporción del número de relaciones no jerárquicas definidas en la ontología sobre la suma total del número de relaciones jerárquicas y no jerárquicas.

Sea  $RNJ$  el número de relaciones no jerárquicas. Sea  $RJ$  el número de relaciones jerárquicas. Luego,

$$RR = \frac{RNJ}{(RNJ + RJ)}$$

Una ontología que contiene muchas relaciones que no son jerárquicas es más rica que una taxonomía con sólo relaciones jerárquicas.

#### *Nivel sintáctico*

En este nivel sólo se considera la riqueza y la complejidad de la sintaxis, ya que se asume que la ontología se desarrolla con alguna herramienta *case* que puede señalar los errores sintácticos.

- Riqueza ( $R$ ), también conocida como expresividad (Colomb, 2002) o expresividad semántica (Supekar y otros, 2004), refiere a la proporción de características del lenguaje de ontología que se han utilizado en una ontología (por ejemplo, si incluye términos y axiomas, o sólo términos) (Burton-Jones y otros, 2005).

Sea  $Y$  el total de características sintácticas disponibles en el lenguaje de ontología. Sea  $Z$  el total de características sintácticas utilizadas en la ontología. Luego,

$$R = \frac{Z}{Y}$$

Ontologías más ricas son más valiosas para un agente de software, por ejemplo.

#### *Otros indicadores*

La Riqueza de Atributos ( $RA$ ) (Tartir y otros, 2005), que es el número promedio de propiedades por término, se puede calcular como otro indicador de esta dimensión de calidad. Se computa como el total de atributos para todos los términos ( $AT$ ) dividido por el total de términos ( $T$ ), es decir

$$RA = \frac{AT}{T}$$

Tartir y otros (2005) asumen que, en general, cuanto más atributos (propiedades) se definen, más conocimientos transmite la ontología. Sin embargo, considerando los principios de diseño propuestos en esta tesis, se puede observar

que esta suposición es relativa. En el método propuesto, la utilización de las propiedades se reserva para la representación de características que no se pueden considerar entidades en sí mismas y que no son medibles. Las otras características se representan con un conjunto de relaciones, términos y axiomas, según sea su tipo. Este hecho no implica que la ontología transmite menos conocimiento, como se muestra en el Capítulo 3.

#### **2.4.2.2. Evaluación de la dimensión semántica**

La calidad de una ontología en su dimensión semántica se puede evaluar teniendo en cuenta los indicadores del nivel léxico, de vocabulario o de datos descriptos por Tartir y otros (2005). La representación de conocimientos previos (información del contexto), es decir, costumbres, expectativas, prácticas comerciales y regulaciones, que generalmente no están representadas explícitamente en la ontología, también deberían ser considerados (Colomb, 2002). Según Supekar y otros (2004), este conocimiento previo es una característica no cuantificable y, además, su incorporación a la ontología incrementa la complejidad de la misma. Sin embargo, alguna información de contexto se debe agregar a la ontología para permitir la interpretación de las entidades representadas en ella en diferentes contextos sin malos entendidos.

##### *Nivel léxico, de vocabulario o de datos*

En este nivel, la atención se centra en los términos, instancias, etc., que se han incluido en la ontología y el vocabulario utilizado para representar o identificar a estos elementos. En este nivel, los indicadores más representativos son:

- Interpretabilidad ( $I$ ), también llamado naturalidad (Stvilia, 2007), refiere al significado de las palabras usadas en la ontología para definir los términos, las propiedades, etc. (Burton-Jones y otros, 2005). Se evalúa comprobando si las palabras utilizadas en la ontología existen en otra fuente semántica independiente, por ejemplo, una base de datos léxica genérica y completa como WordNet<sup>6</sup> o una base de datos léxica específica de un dominio.

---

<sup>6</sup> <http://www.cogsci.princeton.edu/~wn>

Sea  $PTP$  el número total de palabras distintas utilizadas para definir términos y propiedades en una ontología. Sea  $PS$  el número de palabras utilizadas para definir términos y propiedades que tienen por lo menos un sentido listado en WordNet. Luego,

$$I = \frac{PS}{PTP}$$

Los nombres de términos y propiedades pueden ser palabras simples (por ejemplo **Organization**) o frases (por ejemplo **itemDescription**). Para la interpretabilidad se controla la existencia de las palabras individuales en WordNet (por ejemplo **Organization**, **item** y **Description**).

- Claridad ( $C$ ) refiere a si el entorno lingüístico de los términos es comprensible –no se debe confundir con el criterio de claridad definido por Gruber (1995)–. Esto se mide calculando para cada nombre de término o propiedad en la ontología, el número de sentidos de ese nombre en su conjunto en otra fuente semántica independiente (ya sea que se trate de una sola palabra o de una frase) (Burton-Jones y otros, 2005).

Sea  $N_i$  el nombre de un término o propiedad  $i$  en la ontología. Sea  $A_i$  el número de sentidos de  $N_i$  en WordNet. Sea  $TN$  el total de nombres de términos o propiedades en la ontología. Luego,

$$C = \frac{\sum_{i=1}^{TN} A_i}{TN}$$

Idealmente, la ontología debería usar palabras con significados precisos, es decir,  $C = 1$ , para que sea comprensible por una computadora.

El uso de WordNet como fuente semántica puede arrojar resultados distorsionados, ya que WordNet no contiene frases.

#### *Otros indicadores*

En el marco de esta tesis, el término ontología refiere a un artefacto de representación de la semántica de un dominio de discurso dado. Como tal, provee descripciones para entidades, y relaciones y características de esas entidades.

Un modelo de medición de la calidad de la información de una ontología necesita evaluar la calidad de los mapeos de las entidades del dominio en los términos, propiedades, relaciones, etc., utilizados para su representación (Stvilia, 2007). Es decir, el modelo tiene que medir si la ontología representa las entidades del dominio de forma completa, coherente y exacta en relación a los contextos en los que la ontología se utilizará.

Como se mencionó en la Sección 2.3.4.2, en el método propuesto la evaluación de la ontología se realiza siguiendo un enfoque cualitativo, ya que el objetivo es identificar problemas de representación. En este sentido, en esta tesis se siguen los siguientes principios:

1. Una entidad del dominio debe tener un término principal en su representación.
2. Cada uso previsto de una entidad debe tener un término puente que lo designe. Si la entidad tuviera un único uso previsto en diferentes contextos, este término puente sería el término principal de la representación de la entidad.
3. Una relación entre entidades se representa por una relación entre los términos principales que representan dichas entidades.
4. Una característica de una entidad, que se puede considerar una entidad en sí misma, debe ser representada por un término. Se distinguen tres casos:
  - a) Una característica simple se representa por: un término que representa la dimensión, el conjunto de valores posibles; y una relación entre este término y el término que representa a la entidad caracterizada.
  - b) Una característica simple medible se representa por: un término que representa la dimensión; una relación entre este término y el término que representa a la entidad caracterizada; un término que representa a la unidad de medida de la dimensión; una relación entre este término y el término que representa la dimensión; un término que representa la dimensión física de la unidad de medida; y una relación entre este término y el término que representa la unidad de medida.

- c) Una característica compleja se representa por: un término que representa el conjunto de dimensiones integrales; una relación entre este término y el término que representa a la entidad caracterizada; para cada dimensión integral, un término que la representa y una relación entre este término y el término que representa el conjunto de dimensiones integrales; para cada término que representa una dimensión integral medible debe estar la relación con el término que representa la unidad de medida, el término que representa la unidad de medida de la dimensión, un término que representa la dimensión física de la unidad de medida y una relación entre este término y el término que representa la unidad de medida.
5. Una característica de una entidad, que no se puede considerar una entidad en sí misma, debe ser representada por una propiedad.

Con el propósito de evaluar los mapeos de las entidades del dominio, sus relaciones y características, en los elementos utilizados para su representación, es posible contabilizar cuántos de estos principios se cumplen respecto del total de entidades, relaciones y características identificadas en la ontología, según corresponda.

Considerando una representación correcta a aquella que cumple con el principio correspondiente y un vector de pesos  $\alpha$  tal que

$$\alpha_k = \begin{cases} 0 & : \text{ si el elemento } k \text{ no está representado} \\ 0,5 & : \text{ si el elemento } k \text{ está representado de forma incompleta} \\ 1 & : \text{ si el elemento } k \text{ está bien representado} \end{cases}$$

se propone calcular los indicadores para los principios antes mencionados de la siguiente forma:

- Indicador del principio 1: Entidades ( $PE$ ): mide el grado en que las entidades están representadas por términos.

Sea  $E$  el número total de entidades. Luego,

$$PE = \frac{\sum_{k=1}^E \alpha_k}{E}$$

- Indicador del principio 2: Uso Previsto de las Entidades ( $PU$ ): mide el grado en que los usos previstos tienen un término puente que los represente.

Sea  $U$  el total de usos previstos para todas las entidades. Luego,

$$PU = \frac{\sum_{k=1}^U \alpha_k}{U}$$

- Indicador del principio 3: Relaciones entre Entidades ( $PR$ ): mide el grado en que las relaciones entre entidades se representan por relaciones entre los términos que representan a dichas entidades.

Sea  $RE$  el total de relaciones identificadas para todas las entidades. Luego,

$$PR = \frac{\sum_{k=1}^{RE} \alpha_k}{RE}$$

- Indicador del principio 4.a: Características Simples como Entidades ( $PCS$ ): mide el grado en que las características se representan de acuerdo a lo indicado por el principio.

Sea  $CS$  el total de características simples identificadas para todas las entidades. Luego,

$$PCS = \frac{\sum_{k=1}^{CS} \alpha_k}{CS}$$

- Indicador del principio 4.b: Características Simples Medibles como Entidades ( $PCM$ ): mide el grado en que las características medibles se representan de acuerdo a lo indicado por el principio.

Sea  $CM$  el total de características simples medibles identificadas para todas las entidades. Luego,

$$PCM = \frac{\sum_{k=1}^{CM} \alpha_k}{CM}$$

- Indicador del principio 4.c: Características Complejas como Entidades ( $PCC$ ): mide el grado en que las características complejas se representan de acuerdo a lo indicado por el principio.

Sea  $CC$  el total de características complejas identificadas para todas las entidades. Luego,

$$PCC = \frac{\sum_{k=1}^{CC} \alpha_k}{CC}$$

- Indicador del principio 5: Características Comunes ( $PCc$ ): mide el grado en que las características comunes se representan por propiedades.

Sea  $Cc$  el total de características comunes identificadas para todas las entidades. Sea  $\alpha_k$  como se definió antes. Luego,

$$PCc = \frac{\sum_{k=1}^{Cc} \alpha_k}{Cc}$$

Estos indicadores se pueden aplicar tanto para comparar ontologías entre sí como para evaluar el grado de cumplimiento de los principios antes enunciados por parte de una ontología dada.

### 2.4.2.3. Evaluación de la dimensión pragmática

La calidad de una ontología en su dimensión pragmática tiene que ver esencialmente con la complejidad de la conceptualización. Un modelo más complejo sería difícil de comprender y utilizar tanto por una persona como por una computadora o agente de software. Burton-Jones y otros (2005) proponen tres indicadores para evaluar la utilidad de la ontología para las personas y agentes de software, independientemente de su sintaxis o semántica. Ellos son:

- Exactitud analiza si las afirmaciones que hace una ontología son “ciertas”. Esto es difícil de determinar de forma automática, sin un mecanismo de aprendizaje o un sistema de mantenimiento de la veracidad. En la actualidad, un experto de dominio evalúa la exactitud. Stvilia (2007) considera a este indicador en relación con alguna fuente de referencia

estable, es decir, si la información es legítima o válida en relación a un diccionario, o un conjunto de restricciones y normas del dominio, o ambos.

- Integralidad es una medida del tamaño de la ontología. Ontologías más grandes son más propensas a ser representaciones completas de sus dominios y a proveer información a las personas y agentes de software.
- Relevancia ( $RL$ ) evalúa si la ontología satisface los requerimientos específicos del agente de software. Esto requiere un cierto conocimiento de las necesidades del agente antes de la evaluación. Esta métrica es tosca, ya que comprueba si la ontología contiene el tipo de información que el agente de software utiliza (por ejemplo, propiedad, subtérmino, etc.), en lugar de la semántica necesaria para realizar tareas específicas (por ejemplo, subtérminos específicos que se requieren para interpretar una consulta específica de un agente).

Sea  $CN$  el conjunto de sentencias en la ontología y  $S$  el cardinal de ese conjunto. Sea  $SR$  el tipo de sintaxis relevante para un agente de software. Sea  $ISR$  el número de sentencias de  $CN$  que usan  $SR$ . Luego,

$$RL = \frac{ISR}{S}$$

## 2.5. Conclusiones

En este capítulo se presentó un método para desarrollar ontologías que tengan como objetivo facilitar el intercambio de información entre distintos contextos. Cuando una ontología se desarrolla con este objetivo, las entidades y sus características que tienen diferente interpretación en dichos contextos necesitan ser representadas adecuadamente. Para ello, el método se basa en una estrategia para descubrir características contextuales de las entidades y representarlas en la ontología.

Se han diseñado diversas metodologías y métodos para desarrollar ontologías, algunas de las cuales presentan ideas útiles que se incorporaron al método propuesto aquí. Además, debido a la tendencia actual al uso de aplicaciones Web para el intercambio de información, cuyas fuentes de datos son estructuradas o semiestructuradas, el método se presentó haciendo uso de

técnicas de aprendizaje de ontologías, que permiten una generación automática acelerando el proceso de desarrollo.

Uno de los usos más frecuentes de las ontologías es representar la semántica de la información a intercambiar. Muchas veces el significado de las entidades y sus características, que constituyen la información, depende del contexto y esta situación debe representarse en la ontología. Sin embargo, estas entidades y características no suelen estar representadas de manera apropiada en las fuentes de datos, haciendo imposible que las técnicas de aprendizaje las inferan correctamente.

Para hacer frente a este problema, pero al mismo tiempo aprovechar las ventajas de un rápido desarrollo, el método propone modificar manualmente la ontología generada de forma automática. Para mejorar la representación de las entidades y sus características, el método propone dos estrategias diferenciadas: (a) reutilizar ontologías existentes y ampliamente aceptadas, o partes de ellas, para representar entidades temporales y relaciones, cantidades físicas y unidades de medida, y nombres oficiales de países, entre otros ejemplos; (b) cuando no es posible reutilizar ontologías existentes para mejorar la representación de las entidades y sus características, el método propone identificar si la característica es simple o compleja y propone una estrategia de representación a seguir en cada caso.

Mediante el uso de términos puente se relacionan diferentes significados y representaciones de una misma entidad de dominio en distintos contextos. Esto permite representar un aspecto clave en la ontología: los usos previstos de las entidades de dominio. Estos usos dependen del contexto en el que las entidades de dominio se consideran.

El resultado es un rápido desarrollo de una ontología de dominio, a partir de fuentes de datos estructuradas o semiestructuradas, que representa la semántica de las fuentes de datos necesarias para el intercambio de información entre contextos diferentes.

En consecuencia, el método propuesto consta de dos partes: una automatizada y una manual. La parte automatizada se realiza por medio de herramientas que se han desarrollado a tal fin, cuyo desempeño ha sido evaluado por sus autores, y por lo tanto está fuera del alcance de esta tesis. La efectividad de la parte manual del método se probó aplicándolo a casos de estudio para

proveer una validación empírica. Una de dichas aplicaciones se presenta en el capítulo siguiente.

Debido a que el método propuesto trata de mejorar la representación de las entidades en la ontología, se propone realizar la verificación de la ontología resultante utilizando criterios cualitativos. No obstante, y con el propósito de permitir comparar ontologías, en este capítulo se describieron también un conjunto de indicadores para evaluar cuantitativamente la calidad de las ontologías, cuyo objetivo es el intercambio de información entre distintos contextos. En este sentido, se propusieron nuevos indicadores para evaluar la dimensión semántica de la calidad de una ontología, a la luz de los principios de diseño incorporados en el método propuesto. La aplicación de todos estos indicadores se muestra en el capítulo siguiente sobre las ontologías base y contextualizada.

## Aplicación del Método Propuesto a un Caso de Estudio

En este capítulo se muestra la eficacia del método propuesto en el Capítulo 2, aplicándolo a un caso de estudio (Sección 3.1). Dicha aplicación también permite ilustrar el modo de realizar el análisis en cada uno de los pasos propuestos. Luego se compara la calidad de la ontología obtenida respecto de la ontología base, calculando las medidas descritas en la Sección 2.4 del Capítulo 2 (Sección 3.2), y se muestra la influencia de las ontologías contextualizadas en los resultados de un proceso de *matching* (Sección 3.3). Finalmente se presentan las conclusiones (Sección 3.4).

### 3.1. Caso de Estudio 1:

#### Una Ontología para Intercambiar Información en una e-Colaboración

En esta sección se muestra la aplicación del método propuesto en el capítulo anterior a un caso de estudio. Este caso pertenece al dominio empresarial y se basa en una relación colaborativa entre dos socios de negocio que son empresas de manufactura: una industria de envases y una industria láctea. La industria de envases provee a la industria láctea los envases necesarios para sus productos. Se eligió este caso ya que representa un tipo de relación muy frecuente entre empresas que participan de un canal industrial y, por lo tanto, representativo del mismo. Si bien el caso de estudio corresponde a una relación muy frecuente, a

efectos de evitar la difusión de datos privados, las estructuras de datos utilizadas no pertenecen a empresas en particular, sino que se generaron en función de las características de los datos utilizados y la información intercambiada entre ellas a través de Documentos de Negocio Electrónicos. Dado que las herramientas de software utilizadas para realizar el proceso de *matching* entre la ontología del Documento de Negocio Electrónico y la ontología privada sólo soportan definiciones en idioma inglés, dichas ontologías debieron ser expresadas en este idioma. En este capítulo solo se presenta la parte de las ontologías que contienen las entidades que requieren modificaciones para representar las características cuya semántica depende del contexto.

### **3.1.1. Proceso 1: Identificación de los requerimientos de la ontología**

#### **3.1.1.1. Actividad 1.1: Describir el dominio**

Para llevar adelante esta actividad se deben mantener reuniones con representantes de ambas industrias. De estas reuniones debe surgir la identificación de las fuentes de información y los expertos del dominio, quienes constituyen un elemento esencial en el proceso de desarrollo de la ontología y la respuesta a preguntas tales como: (a) ¿Qué dominio va a cubrir la ontología? (b) ¿Qué contextos se pueden identificar en el dominio? (c) ¿Cuáles son las características que describen estos contextos? A continuación se presenta una breve descripción del dominio, la cual forma parte del DERO.

Para implementar la relación colaborativa, las industrias aplican el Modelo de Colaboración Socio-a-Socio (Villarreal y otros, 2004), que propone una gestión descentralizada de colaboraciones a largo plazo entre dos empresas de producción en una cadena de suministro. Este modelo permite a cada socio mantener la confidencialidad y autonomía de sus procesos de negocio y estructuras de datos mientras se intercambian información. Un socio de negocio puede manejar relaciones simultáneas con diferentes socios en forma independiente, evitando posibles situaciones conflictivas. Este modelo define un proceso de negocio colaborativo dividido en tres subprocesos:

- Consenso a nivel de Planificación Agregada de la Producción (PAP). Su

objetivo es hacer que las empresas arriben a un consenso sobre un plan de provisión de material a nivel PAP. En este nivel, los socios de negocio acuerdan sobre los productos que quieren colaborar (a nivel de familia de productos), los períodos de esa colaboración (el horizonte de tiempo es de entre seis y dieciocho meses) y las cantidades aproximadas de estos productos.

- Consenso a nivel de Programa Maestro de la Producción (PMP). En este subproceso, las empresas tienen que arribar a un consenso acerca de un programa de provisión de material a nivel PMP. En este nivel, los productos se definen en el mayor nivel de detalle requerido por la empresa proveedora. Además, los períodos y cantidades se especifican a un mayor nivel de detalle. La empresa cliente especifica la cantidad de material y la fecha de entrega requeridos. La empresa proveedora define la fecha de provisión y el tamaño de las órdenes de provisión, teniendo en cuenta un acuerdo marco. Esta es información “probable”.
- Consenso a nivel de Programa de Órdenes de Provisión (POP). Su objetivo es hacer que los socios de negocio arriben a un consenso sobre la definición de un cronograma de órdenes de provisión. En este nivel se gestiona información “cierta”. Los períodos indican el día exacto en el que los productos estarán disponibles para su envío. Se define un programa de órdenes de producción y provisión para ambos socios de negocio.

Cada uno de estos niveles constituye un contexto diferente, donde la información tiene una interpretación distinta. Por ejemplo, en el nivel de PAP los productos se definen a nivel de familias, mientras que en el nivel de PMP se definen en el mayor nivel de detalle.

El Modelo de Colaboración Socio-a-Socio implica la coordinación de dos tipos de procesos de negocio (Villarreal y otros, 2007): procesos privados y procesos colaborativos (Figura 3.1). Los primeros se ejecutan de forma independiente por cada socio, mientras que los segundos se diseñan conjuntamente de manera abstracta y los ejecuta cada socio a través de procesos de negocio de interfaz (PNI) y procesos de negocio de integración. Un proceso de negocio de interfaz es responsable del rol que desempeña un socio de negocio en un proceso colaborativo; define el comportamiento público del socio en términos de

las actividades que soportan la recepción y el envío de mensajes con sus socios. Un proceso de negocio de interfaz es el esqueleto de un proceso de negocio de integración, que incorpora los aspectos privados y la lógica de negocio interna necesarios para el cumplimiento del rol del socio. Los aspectos privados incluyen las actividades que producen la información que se envía al otro socio, y las que procesan la información recibida de él.

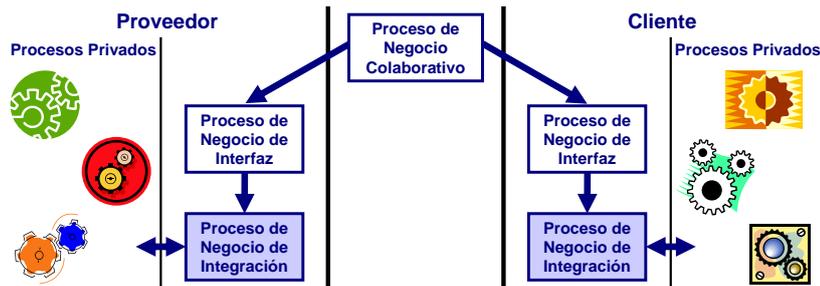


Figura 3.1: DERO: Descripción del dominio. Procesos involucrados en el Modelo de Colaboración Socio-a-Socio

En los mensajes, los socios intercambian Documentos de Negocio Electrónicos (DNEs). Un DNE es una estructura de datos estandarizada que reemplaza los documentos de negocio tradicionales tales como datos de punto de venta y pronósticos de órdenes, entre otros. Además, soporta el intercambio de información requerida para ejecutar los procesos colaborativos (Caliusco, 2005). La Figura 3.2 ilustra la colaboración entre la industria de envases (el proveedor) y la industria láctea (el cliente). En esta figura se muestra el protocolo de interacción “Solicitud Programa de Provisión”. El propósito de este protocolo es llegar a un acuerdo sobre el programa de provisión que se define a nivel de PMP. Con este objetivo, se envía un DNE como parte de cada mensaje intercambiado entre las industrias. Los mensajes están encabezados por un acto de comunicación (*request* –solicitud–, *agree* –aceptación–, *refuse* –rechazo–) que indica la intención del mensaje. La Figura 3.3 muestra un ejemplo de la información contenida en un DNE. La sintaxis y estructura del DNE basado en XML, utilizado para intercambiar esta información, se muestra en la Figura 3.4. Este documento se expresa en XML Schema (Biron y Malhotra, 2004; Thompson y otros, 2004) y es una de las fuentes de datos consideradas para generar la ontología base del Proceso 2 del método propuesto. Dicho documento es una adaptación del documento `PurchaseOrder.xsd` que forma parte del estándar de documentos de

negocio OAGIS® 9.4.1 (Open Applications Group, nd).

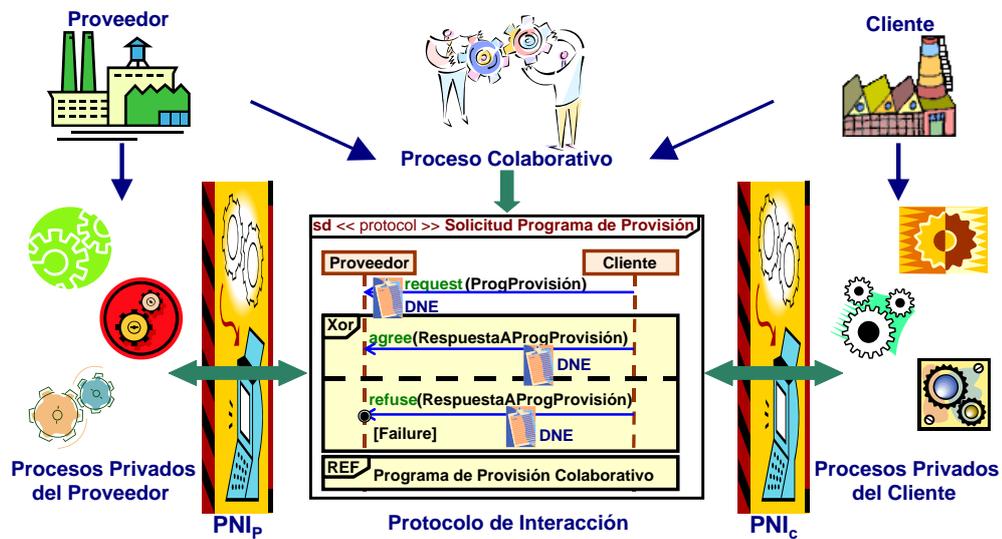


Figura 3.2: DERO: Descripción del dominio. Elementos de la relación colaborativa entre los socios de negocio

		Compañía Láctea Santa Fe Teniente General Richieri 15 Firmat – Santa Fe (03465) Contacto: Juan Pérez			Programa de Provisión N° 122364 Fecha: 12/08/09	
Proveedor: All Packaging Dirección: 216 Post Road, Buchanan, MI 49107 Contacto: Lisa Lee						
Horizonte: 01/09 – 31/10						
PERÍODO	PRODUCTO				CANTIDAD	PRECIO
	Id	Marca	Tipo	Capacidad		
1/09 – 7/09	20320101	yy	carton	1000	4400	0,40
	20320102	yy	carton	2900	2880	0,41
	20070231	yy	plastic	196	1600	0,45
	20070232	yy	plastic	250	1800	0,47
	20070235	yy	plastic	1000	6500	0,49
8/09 – 14/09	20320101	zz	carton	1000	2200	0,37
	20320102	zz	carton	2900	8064	0,35
	20070232	zz	plastic	250	1800	0,39
	20070235	zz	plastic	1000	6500	0,38
8/09 – 14/09	...	...	...	...	...	...
	...	...	...	...	...	...

Figura 3.3: DERO: Descripción del dominio. Ejemplo de la información intercambiada para acordar sobre un programa de provisión a nivel de PMP

Debido a que una de las premisas del modelo de colaboración es la de preservar la individualidad y confidencialidad de las empresas involucradas, no es apropiado definir un significado global de su información privada; la información

```

<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.EBD.org"
  xmlns="http://www.EBD.org"
  elementFormDefault="qualified">
  <xsd:complexType name="Agent">
    <xsd:sequence>
      <xsd:element name="phone" type="xsd:string" maxOccurs="unbounded"/>
      <xsd:element name="email" type="xsd:string" maxOccurs="unbounded"/>
      <xsd:element name="fax" type="xsd:string" maxOccurs="unbounded"/>
      <xsd:element name="name" type="xsd:string"/>
      <xsd:element name="mailingAddress" type="Address"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="Address">
    <xsd:sequence>
      <xsd:element name="country" type="xsd:string"/>
      <xsd:element name="zip" type="xsd:integer"/>
      <xsd:element name="city" type="xsd:string"/>
      <xsd:element name="street" type="xsd:string"/>
      <xsd:element name="state" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="Organization">
    <xsd:complexContent>
      <xsd:extension base="Agent">
        <xsd:sequence>
          <xsd:element name="contactPerson" type="Personnel"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="Personnel">
    <xsd:complexContent>
      <xsd:extension base="Agent">
        <xsd:sequence>
          <xsd:element name="position" type="xsd:string"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="EBDLine">
    <xsd:sequence>
      <xsd:element name="partNumber" type="xsd:integer"/>
      <xsd:element name="period" type="xsd:string"/>
      <xsd:element name="itemDescription" type="xsd:string"/>
      <xsd:element name="itemName" type="xsd:string"/>
      <xsd:element name="quantity" type="xsd:integer"/>
      <xsd:element name="price" type="xsd:float"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="EBD">
    <xsd:sequence>
      <xsd:element name="supplier" type="Organization"/>
      <xsd:element name="customer" type="Organization"/>
      <xsd:element name="ebdNumber" type="xsd:integer"/>
      <xsd:element name="horizon" type="xsd:string"/>
      <xsd:element name="ebdDate" type="xsd:string"/>
      <xsd:element name="ebdName" type="xsd:string"/>
      <xsd:element name="lineCollection" type="EBDLine" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:element name="ReplenishmentProgram" type="EBD"/>
</xsd:schema>

```

Figura 3.4: DERO: Descripción del dominio. DNE basado en XML

contenida en los DNEs debe ser traducida a los sistemas de información privados de cada socio de negocio.

A fin de facilitar el proceso de traducción, un documento XML proporciona

la sintaxis para el intercambio de información pero no la semántica asociada a esta información. XML se creó para facilitar el intercambio de datos entre diferentes aplicaciones, fuentes de datos y sistemas operativos. Debido a que los datos dentro de un documento XML están etiquetados, el documento contiene la información necesaria para reconocerlos, extraerlos y manipularlos. Sin embargo, el uso de XML no garantiza que las partes involucradas en el intercambio de información, sean estas personas o programas de computación, entiendan correctamente la semántica del contenido del documento XML, ya que dicha semántica está implícita. En el documento XML de la Figura 3.4, los elementos están delimitados por etiquetas de inicio y fin. XML permite crear etiquetas propias, por ejemplo, `<EBD>` o `<quantity>`, que tienen alguna semántica implícita para las personas, pero carecen de interpretación desde el punto de vista computacional. Una computadora no sabe qué es `quantity` ni cómo el concepto de `quantity` se relaciona con, por ejemplo, el concepto de `UnitOfMeasure`.

Debido a esto, el uso de XML no garantiza que los sistemas de información de los socios entiendan el contenido de un determinado mensaje, ya que pueden utilizar diferentes términos y estructuras de metadatos para representar sus datos, incluso cuando se refieren a la misma área de interés. Por ejemplo, mientras la industria láctea representa los datos de fecha en el formato “AAAA-MM-DD”, la industria de envases los representa en el formato “M/D/AA”. Asimismo, ambas empresas utilizan diferentes monedas para expresar los valores monetarios. Para lograr una verdadera comunicación, sin malos entendidos, los socios de negocio deben ponerse de acuerdo sobre el significado de la información intercambiada en los DNEs. Con este objetivo se decide el desarrollo de una ontología que represente la semántica de dicha información, la cual se desarrolla mediante la aplicación del método propuesto.

Los expertos del dominio son los que tienen la responsabilidad del acuerdo del programa de provisión; son los que aparecen como las personas de contacto en la Figura 3.3.

Los principales contextos involucrados son: el contexto de la industria láctea, el contexto de la industria de envases y el contexto del proceso colaborativo (en sus niveles PAP, PMP y POP). Respecto de este último contexto, en una primera versión de la ontología sólo se considera al contexto PMP. También, dentro de cada empresa pueden coexistir otros contextos debido a que las áreas o departamentos

(Producción, Finanzas, etc.) suelen tener su propia visión de la empresa, pero por simplicidad no se consideran aquí.

También se deben identificar los escenarios de motivación principales en los que participa la ontología. En este caso son dos: el procesamiento del contenido de un DNE y la “población” de un DNE. El primer escenario se produce cuando llega un DNE; el proceso de negocio de integración del receptor debe traducir su contenido para que lo utilicen los procesos privados, de acuerdo a la semántica de los correspondientes contextos de la empresa. El segundo escenario ocurre cuando el proceso de negocio de integración debe enviar un DNE, para lo cual debe previamente poblar ese DNE con información generada en los correspondientes contextos de la empresa, de acuerdo a la semántica del proceso colaborativo. Estos escenarios se dividieron en subescenarios para hacerlos más simples. A modo de ejemplo, las Figuras 3.5 y 3.6 muestran el primer escenario principal y uno de sus subescenarios.

Los expertos del dominio deben elaborar una lista de preguntas que se espera que la ontología de un DNE responda. La Tabla 3.1 muestra algunas de ellas a modo de ejemplo.

---

PC <sub>1</sub> : ¿Cuál es el rol desempeñado (proveedor o cliente) por el socio $x$ en el DNE $y$ ?
PC <sub>2</sub> : ¿Quién es la persona de contacto del socio $x$ ?
PC <sub>3</sub> : ¿Cuál es el horizonte de tiempo del DNE $x$ ?
PC <sub>4</sub> : Según el DNE $x$ , ¿qué productos y en qué cantidades se habrán recibido/enviado para la fecha $y$ ?

---

Tabla 3.1: DERO: Descripción del dominio. Algunas preguntas de competencia

### 3.1.1.2. Actividad 1.2: Determinar el objetivo y alcance de la ontología

La segunda actividad de este proceso es definir el objetivo y el alcance de la ontología. En este sentido, el objetivo de la ontología es describir semánticamente la información que debe intercambiarse entre las empresas para lograr un acuerdo sobre un programa de provisión de productos. Esta ontología se utiliza luego en un proceso de traducción.

Inicialmente el alcance de la ontología se limita a dar respuesta a las siguientes preguntas generales:

<b>Nombre:</b> Procesamiento del contenido de un DNE		<b>Nro:</b> EM 001
<b>Objetivo:</b> Procesar la información contenida en el DNE recibido		<b>Versión:</b> 1
<b>Actores:</b> <ul style="list-style-type: none"> <li>Proceso de negocio de integración del socio receptor</li> </ul> <b>Precondiciones:</b> <ul style="list-style-type: none"> <li>El sistema está en ejecución</li> </ul> <b>Evento Disparador:</b> <ul style="list-style-type: none"> <li>Recepción de un DNE</li> </ul> <b>Poscondiciones:</b> <ul style="list-style-type: none"> <li>Éxito: DNE procesado</li> <li>Fracaso: DNE no procesado</li> </ul> <b>Requerimientos Especiales:</b>		
<b>Flujo Normal</b>		<b>Excepción</b>
1. El sistema determina el acto de comunicación 2. El sistema invoca al EM 002: Traducción del contenido de un DNE 3.a El acto es un <i>request</i> (solicitud). El sistema invoca al EM 003: Procesamiento de solicitud de un DNE 3.b El acto es un <i>agree</i> (aceptación). El sistema invoca al EM 004: Procesamiento de aceptación de un DNE 3.c El acto es un <i>refuse</i> (rechazo). El sistema invoca al EM 005: Procesamiento del rechazo de un DNE 4. El escenario motivador termina.		El sistema no puede determinar el acto de comunicación, presenta un mensaje de error y va al Paso 4 (Flujo Normal)  El acto no es ninguno de los permitidos, el sistema presenta un mensaje de error y va al Paso 4 (Flujo Normal)
<b>Términos Principales:</b> <ul style="list-style-type: none"> <li>DNE, acto de comunicación</li> </ul> <b>Observaciones:</b>		

Figura 3.5: DERO: Descripción del dominio. Escenario de motivación: Procesamiento del contenido de un DNE

- ¿Qué entidades componen un DNE?
- ¿Qué características distinguen a cada entidad del DNE?
- ¿Cómo se relacionan entre sí las entidades del DNE?

### 3.1.1.3. Actividad 1.3: Identificar las principales entidades del dominio, sus relaciones y características

Teniendo en cuenta los resultados de las actividades anteriores se derivan las siguientes entidades relevantes: los socios de negocio, el programa de provisión y los ítems involucrados en ese programa. Los socios de negocio refieren a las industrias que participan de la colaboración. El programa de provisión refiere al acordado entre los socios de negocio. Los ítems indican los productos requeridos,

<b>Nombre:</b> Traducción del contenido de un DNE		<b>Nro:</b> EM 002
<b>Objetivo:</b> Traducir la información contenida en el DNE a los procesos internos de acuerdo a la semántica de los correspondientes contextos de la empresa		<b>Versión:</b> 1
<b>Actores:</b> <ul style="list-style-type: none"> <li>Proceso de negocio de integración del socio receptor</li> </ul> <b>Precondiciones:</b> <ul style="list-style-type: none"> <li>DNE recibido</li> <li>Reglas de conversión definidas</li> </ul> <b>Evento Disparador:</b> <ul style="list-style-type: none"> <li>Invocación desde EM 001</li> </ul> <b>Poscondiciones:</b> <ul style="list-style-type: none"> <li>Éxito: Contenido del DNE traducido</li> <li>Fracaso: Contenido del DNE no traducido</li> </ul> <b>Requerimientos Especiales:</b>		
<b>Flujo Normal</b>		<b>Excepción</b>
1. El sistema ejecuta las reglas de conversión definidas desde la ontología del DNE hacia las ontologías privadas 2. El escenario motivador termina.		El sistema no encuentra las reglas de conversión, presenta un mensaje de error y va al Paso 2 (Flujo Normal)
<b>Términos Principales:</b> <ul style="list-style-type: none"> <li>DNE, reglas de conversión</li> </ul> <b>Observaciones:</b>		

Figura 3.6: DERO: Descripción del dominio. Escenario de motivación: Traducción del contenido de un DNE

los períodos en que estos productos se necesitan, sus cantidades y precios. Los productos son los fabricados por la industria de envases, y los envases de los productos de la industria láctea.

La Tabla 3.2 muestra las relaciones entre estas entidades del dominio. El contenido de esta tabla debe leerse de izquierda a derecha. Por ejemplo, un programa de provisión contiene ítems.

	Socio de Negocio	Programa de Provisión	Ítem
Socio de Negocio		esClienteEn	provee
		esProveedorEn	solicita
Programa de Provisión	esEnviadoPor		contiene
	esRecibidoPor		
Ítem	esProvistoPor	estaContenidoEn	
	esSolicitadoPor		

Tabla 3.2: DERO: Relaciones entre las entidades del dominio

Analizando la información del programa de provisión (Figura 3.3) se derivan las características de entidades que figuran en la Tabla 3.3. Con el fin de decidir si

la característica de una entidad es contextual se formulan las siguientes preguntas: ¿la interpretación de la característica de la entidad depende del contexto en el que se la considere? ¿Cuáles son las posibles interpretaciones? Por ejemplo, se plantearon las siguientes preguntas:

- El principio y el fin de horizonte ¿se definen por un mes y un año o por un día y un mes? Por ejemplo, el valor “01/09” de Horizonte en la Figura 3.3 ¿representa “1 de septiembre” o “Enero 2009”?
- ¿Qué significa la característica Fecha? ¿La fecha se define por un día, un mes y un año? ¿O se define por un mes, un día y un año?
- La característica Cantidad ¿refiere a unidades de productos o unidades de paquetes de producto?
- Precio ¿incluye impuestos?

	Característica Contextual	Característica No Contextual
Socio de Negocio	Dirección	Nombre Organización
		Persona de Contacto
Programa de Provisión	Horizonte	Número
	Fecha	
Ítem	Período	
	Producto	
	Cantidad	
	Precio	

Tabla 3.3: DERO: Características de las entidades del dominio

### 3.1.2. Proceso 2: Generación de una ontología base

Dado que la fuente de datos considerada está implementada en XML Schema (Figura 3.4), se utilizó la tecnología *Semantic XML* de TopQuadrant<sup>TM1</sup> para obtener la ontología base que se muestra en la Figura 3.7. Esta ontología se denomina ontología DNE Base. La notación utilizada en esta tesis para la representación gráfica de ontologías se describe en el Apéndice A (página 205).

<sup>1</sup> <http://www.topquadrant.com/>

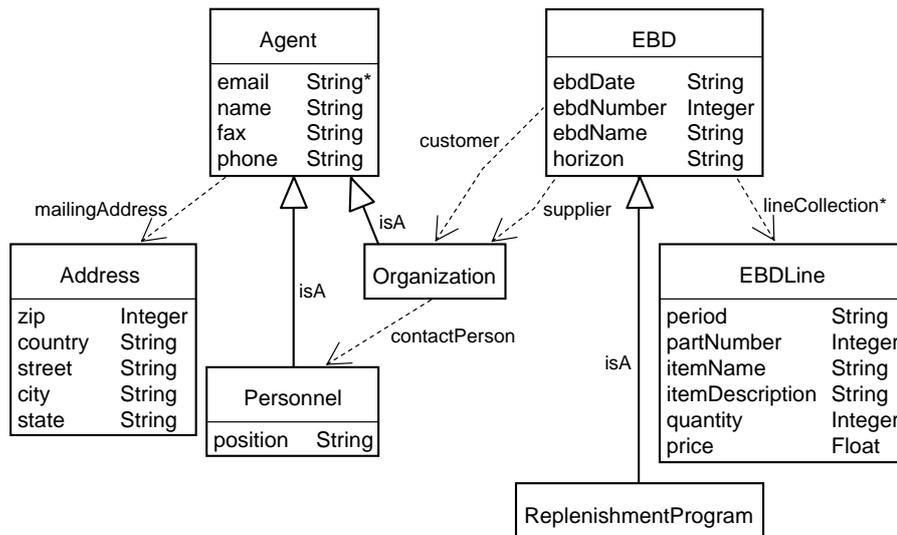


Figura 3.7: Ontología DNE Base

### 3.1.3. Proceso 3: Formalización de la ontología

La técnica de aprendizaje aplicada en el proceso anterior, *Semantic XML*, está implementada en la herramienta TopBraid Composer<sup>TM</sup>, edición Maestro<sup>2</sup>. Esta herramienta permite convertir un documento escrito en XML en una ontología escrita en RDF u OWL. La ontología DNE Base se formalizó en OWL (su código se muestra en el Apéndice B, página 207).

### 3.1.4. Proceso 4: Verificación de la ontología base

Con el fin de comprobar si la ontología base cumple con los requerimientos descritos en el DERO, en primer lugar se deben identificar los elementos de la ontología DNE Base que representan a las entidades del dominio, sus relaciones y características, y luego observar esta representación para verificar el cumplimiento de un conjunto de criterios de diseño de ontologías. Este proceso se lleva a cabo para cada entidad del dominio identificada en la Actividad 1.3.

La verificación de la coherencia lógica de la ontología base se realizó con el uso del razonador RACER<sup>3</sup> y el marco de pruebas proporcionado por el editor Protégé-OWL (Knublauch y otros, 2004). Los resultados de estas pruebas se presentan a continuación, junto con la descripción de la verificación de cada

<sup>2</sup> A esta herramienta la comercializa la empresa TopQuadrant<sup>TM</sup> y, debido a que la técnica aplicada no está disponible en la edición libre, se utilizó una versión de evaluación.

<sup>3</sup> <http://www.sts.tu-harburg.de/~r.f.moeller/racer/>

entidad.

#### 3.1.4.1. Verificación de la representación de la entidad socio de negocio

Los socios de negocio asumen dos roles diferentes: proveedor y cliente. Ambos roles están representados por los siguientes elementos de la ontología DNE Base (Figura 3.7): los términos **Agent** y **Organization**, sus propiedades y la relación **isA** entre ellos.

Un socio de negocio se relaciona con un programa de provisión a través de las relaciones **esClienteEn** y **esProveedorEn**, y con un ítem a través de las relaciones **provee** y **solicita** (Tabla 3.2, página 70). Sin embargo, ninguna de estas relaciones está representada en la ontología. Para poder acceder a ambas entidades desde un socio de negocio se necesitan definir al menos las relaciones entre un socio de negocio y un programa de provisión, para poder acceder así también a los items provistos y solicitados por los socios de negocio.

De acuerdo a la Tabla 3.3 (página 71), las principales características de un socio de negocio son las siguientes: nombre de la organización, persona de contacto y dirección. En la ontología DNE Base, el nombre de la organización está representado por la propiedad **name** del término **Agent**. La persona de contacto está representada por la relación **contactPerson** entre los términos **Organization** y **Personnel**, los términos **Personnel** y **Agent**, sus propiedades y la relación **isA** entre ellos. Por último, la dirección es una característica contextual, que está representada por el término **Address**, sus propiedades y la relación **mailingAddress** entre los términos **Agent** y **Address**.

Esta representación de la entidad socio de negocio presenta algunos inconvenientes. En primer lugar, la jerarquía formada por los términos **Agent**, **Organization** y **Personnel** no incluye en su modelado que **Organization** y **Personnel** son disjuntos; tampoco se expresa que la descomposición de **Agent** en **Organization** y **Personnel** es exhaustiva. La ausencia de la restricción de subterminos disjuntos también fue corroborada con la verificación de la coherencia lógica de la ontología.

Por otra parte, el término **Address** representa una dirección de correo postal. Las características típicas de una dirección de correo son: calle, número de la calle, piso, departamento, ciudad, código postal, provincia o estado, y país. Algunas de ellas, como piso y departamento, no están representadas en la ontología DNE

Base, pero este hecho no impide una correcta interpretación de la dirección de correo postal en diferentes contextos y, por lo tanto, se decide no representarlas.

En cambio, la representación de la característica país no es adecuada. Su representación es la propiedad `country` del término `Address`, cuyo tipo de datos es `string`. Esta forma de representación no revela si la característica país refiere a un código, tal como el de la norma ISO 3166, o a un nombre. Así, no satisface el criterio de completitud, ya que las características que se utilizan para identificar un país no están representadas. El sistema de información, que tiene que manejar esta información, necesita saber cómo procesarla. Esta representación de la característica país también se puede ver como un exceso en la aplicación del criterio de compromiso ontológico mínimo. Al estar esta característica modelada como una propiedad cuyo tipo de dato es `string`, le da a las partes la libertad de instanciarla con valores como 54, AR, Arg. o Argentina, pero esto no es suficiente para permitir una comunicación sin malos entendidos. La introducción de país como una entidad permite establecer restricciones de integridad sobre los datos. Si dos direcciones postales pertenecen a un mismo país es posible asegurar que tendrán el mismo código de identificación de la norma ISO 3166, por ejemplo. Como resultado, la representación de la característica país necesita ser mejorada.

#### **3.1.4.2. Verificación de la representación de la entidad programa de provisión**

La entidad programa de provisión y sus características están representadas, en la ontología DNE Base, por los términos `ReplenishmentProgram` y `EBD`, sus propiedades y la relación `isA` entre estos términos (Figura 3.7, página 72). Las relaciones `supplier` y `customer` (entre los términos `EBD` y `Organization`) representan a las relaciones `esEnviadoPor` y `esRecibidoPor` respectivamente de la Tabla 3.2 (página 70), mientras que la relación `lineCollection` entre los términos `EBD` y `EBDLine` representa la relación `contiene` de la tabla mencionada.

Si bien esta representación es incompleta, ya que la jerarquía cuyo término raíz es `EBD` tiene un único término descendiente `ReplenishmentProgram`, permite a los procesos que la tratan hablar de un `ReplenishmentProgram`; también facilita la extensibilidad, ya que otros tipos de documentos (plan de provisión, cronograma de órdenes de provisión) se pueden agregar a la jerarquía sin necesidad de modificar lo ya definido.

De acuerdo a la Tabla 3.3 (página 71), las características de un programa de provisión son: el número, que identifica el programa de provisión; la fecha en la que se hizo el programa de provisión; y el horizonte, que representa el período de tiempo durante el cual el programa es válido.

El número de un programa de provisión está representado por la propiedad `ebdNumber` del término EBD, cuyo tipo de datos es `integer`.

El período de tiempo durante el cual el programa es válido está representado por la propiedad `horizon` del término EBD, cuyo tipo de datos es `string`. Esta forma de representación no revela si la característica horizonte se expresa como una cantidad de tiempo o como un intervalo. Así, no se satisfacen los criterios de completitud, compromiso ontológico mínimo ni enfoque de codificación mínima. Al igual que la característica país analizada en la sección anterior, esta representación es un exceso en la aplicación del criterio de compromiso ontológico mínimo: modelar esta característica como una propiedad cuyo tipo de dato es `string` no asegura una comunicación sin malos entendidos. Además, representar entidades en lugar de sólo datos provee cierta independencia de la codificación y resolución específica de una aplicación. Así, considerar la característica horizonte como una entidad temporal separada permitirá luego referirse a la misma con distintos niveles de precisión. Si en un contexto dado se dice que el horizonte es Enero 2010 - Marzo 2010 y en otro contexto se dice que el horizonte es 01/01/2010 - 31/03/2010 es posible determinar que ambas referencias son consistentes. Luego, la representación de esta característica debe ser mejorada.

Otra característica que debe ser mejorada es la fecha, que está representada por la propiedad `ebdDate` del término EBD. Dado que el tipo de datos de esta propiedad es `string`, no queda claro si `ebdDate` refiere a un día, mes y año; o un mes, día, y año; o cualquier otro valor que se le asigne a la instancia. Se podría imponer que todas las fechas sean representadas en algún formato, tal como día/mes/año, pero esto sería una clase de enfoque de codificación que se intenta minimizar.

#### 3.1.4.3. Verificación de la representación de la entidad ítem

La entidad ítem y sus características están representadas, en la ontología DNE Base, por el término EBDLine y sus propiedades (Figura 3.7, página 72). Sin embargo, las relaciones de la entidad ítem con las entidades socio de negocio y

programa de provisión no están representadas en la ontología. Para poder acceder al programa de provisión que referencia un ítem, y así también a su proveedor y su comprador, es suficiente con agregar la relación `estaContenidoEn` de la Tabla 3.2 (página 70).

De acuerdo a la Tabla 3.3 (página 71), las características de la entidad ítem son: el período en el cual los ítems se necesitan, los productos que se intercambian, las cantidades de ellos y los precios en los que se los comercializa.

El período está representado por la propiedad `period` del término `EBDLine`, cuyo tipo de datos es `string`. Esta forma de representación presenta los mismos problemas que la característica horizonte mencionados en la sección anterior.

La característica cantidad está representada por la propiedad `quantity` del término `EBDLine`, cuyo tipo de datos es `integer`. Además de no cumplir esta representación con el criterio de enfoque de codificación mínima, en este punto surgen algunas preguntas: ¿en qué unidad de medida está expresada la cantidad? Ésta podría referir a unidades de productos o unidades de paquetes de productos, por ejemplo. Cualquier sistema de información que deba tratar con tal información ¿hará una correcta interpretación de ella? Si existe cualquier posibilidad de malos entendidos, la unidad de medida debe ser representada.

Análogo a la característica cantidad se puede hacer el análisis sobre la característica precio, que representa el valor monetario de cada producto comercializado. Esta característica está representada por la propiedad `price` del término `EBDLine`, cuyo tipo de datos es `float`. En este caso, surge la siguiente pregunta: ¿cuál es la moneda en la que se expresa el precio?

La característica producto está representada por las siguientes propiedades del término `EBDLine`: `partNumber`, cuyo tipo de datos es `integer`; e `itemName` e `itemDescription`, cuyos tipos de dato son `string`. `partNumber` representa el identificador del producto de la Figura 3.3 (Id, página 65), pero `itemName` e `itemDescription` no representan, a primera vista al menos, ninguna de las otras características de producto listadas en la Figura 3.3: Marca, Tipo y Capacidad. `itemDescription` puede referir a una descripción en lenguaje natural del producto. Sin embargo, la propiedad `itemName` debería ser reemplazada para representar adecuadamente las características marca, tipo y capacidad del producto. Además, teniendo en cuenta que un producto es tanto un elemento elaborado por la industria de envases como un envase para algún producto de la industria láctea,

esta representación de producto no satisface los criterios de claridad, coherencia ni completitud. Una solución a este problema es considerar a producto también como una entidad de dominio y crear su propia representación.

En el caso que nos ocupa, definir un producto en términos de un identificador no representa un ejemplo de enfoque de codificación. Si bien la elección de un identificador puede ser una decisión a nivel simbólico, en el dominio empresarial es normal referirse a los productos por sus identificadores numéricos. Es parte de la convención de las empresas que los identificadores de productos se codifiquen como enteros. Tanto es así que existen estándares que sirven como sistemas de clasificación de los productos como es el caso de eCI@ss (ETIM, nd) y UNSPSC (UNSPSC, 2010).

La Tabla 3.4 resume los resultados de la verificación.

### **3.1.5. Proceso 5: Enriquecimiento de la ontología base**

#### **3.1.5.1. Enriquecimiento de la representación de la entidad socio de negocio**

En el proceso anterior se concluyó que la representación de la característica país debía ser mejorada. También se explicó la conveniencia de modelarla como una entidad de dominio en sí misma. La característica país de una dirección de correo postal puede considerarse como una entidad de dominio, y ser representada por medio del término **Country** en lugar de una propiedad. En este caso, se necesita una relación formal (Guizzardi, 2005) que una los términos **Address** y **Country**.

A fin de satisfacer el criterio de extendibilidad (Gruber, 1995), además de los de completitud y compromiso ontológico mínimo mencionados en la Sección 3.1.4.1, el término **Country** y sus posibles valores se pueden importar de una ontología que modele los nombres oficiales de los países, tal como se indica en la norma ISO 3166. La Figura 3.8 muestra una porción de la ontología DNE Enriquecida, que contiene la ontología DNE Base (Figura 3.7, página 72), más todas las modificaciones introducidas. En dicha figura, el prefijo **country:** indica que el elemento correspondiente pertenece a la ontología de países importada.

Para cumplir con la completitud, a esta ontología DNE Enriquecida se le agregaron también las restricciones de jerarquía de **Agent** exhaustiva y de

	Representación			Criterio no cumplido
	Término principal	Relación	Propiedad	
<b>Entidad</b>				
Socio de Negocio				Representación de conocimiento disjuncto y exhaustivo <sup>1</sup>
Proveedor		Organization		
Cliente		Organization		
Programa de Provisión	ReplenishmentProgram			
Ítem	EBDLine			
<b>Relación</b>				
esClienteEn				No representada
esProveedorEn				No representada
provee				No representada
solicita				No representada
esEnviadoPor		supplier		Estandarización de nombres <sup>1</sup>
esRecibidoPor		customer		Estandarización de nombres <sup>1</sup>
contiene		lineCollection		Estandarización de nombres <sup>1</sup>
esProvistoPor				No representada
esSolicitadoPor				No representada
estaContenidoEn				No representada
<b>Característica no contextual</b>				
Nombre Organización			name	Estandarización de nombres <sup>1</sup>
Persona de Contacto	Personnel	contactPerson		
Número			ebdNumber	
<b>Característica contextual</b>				
Dirección	Address	mailingAddress		Estandarización de nombres <sup>1</sup>
calle			street	
número de calle				No representada
piso				No representada
departamento				No representada
ciudad			city	
código postal			zip	
provincia o estado			state	
país			country	Compleitud <sup>2</sup>
Horizonte			horizon	Exceso compr. ontol. mínimo <sup>3</sup>
Fecha			ebdDate	Compleitud <sup>2</sup>
Período			period	Exceso de codificación mínima <sup>3</sup>
Producto				Compleitud <sup>2</sup>
id			partNumber	Exceso de codificación mínima <sup>3</sup>
descripción			itemName	Compleitud <sup>2</sup>
marca			itemDescription	Claridad <sup>3</sup>
tipo				Coherencia <sup>3</sup>
capacidad				Concisión <sup>2</sup>
Cantidad			quantity	No representada
Precio			price	No representada
				No representada
				Exceso de codificación mínima <sup>3</sup>
				Dim. p/ especific. característica
				Dim. p/ especific. característica

<sup>1</sup> (Arpírez Vega y otros, 1998)<sup>2</sup> (Gómez-Pérez, 1995)<sup>3</sup> (Gruber, 1995)

Tabla 3.4: Resultado de la verificación

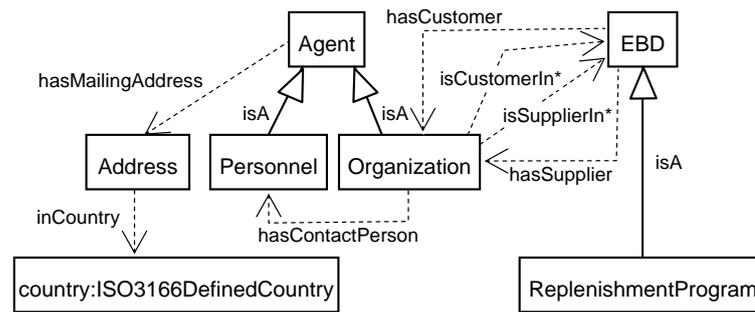


Figura 3.8: Una porción de la ontología DNE Enriquecida con la representación de la entidad socio de negocio

subtérminos (*Organization* y *Personnel*) disjuntos. En lógica descriptiva esto es:

$$\text{Agent} \equiv \text{Organization} \cup \text{Personnel}$$

$$\text{Organization} \cap \text{Personnel} \subseteq \perp$$

El código OWL siguiente muestra esto mismo:

```
<owl:Class rdf:ID="Agent">
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:ID="Personnel"/>
        <owl:Class rdf:ID="Organization"/>
      </owl:unionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

<owl:Class rdf:about="#Personnel">
  <rdfs:subClassOf rdf:resource="#Agent"/>
  <owl:disjointWith>
    <owl:Class rdf:about="#Organization"/>
  </owl:disjointWith>
</owl:Class>

<owl:Class rdf:about="#Organization">
  <rdfs:subClassOf rdf:resource="#Agent"/>
  <owl:disjointWith>
    <owl:Class rdf:about="#Personnel"/>
  </owl:disjointWith>
</owl:Class>
```

También, los nombres de las relaciones *mailingAddress*, *contactPerson*, *supplier* y *customer* se cambiaron a *hasMailingAddress*, *hasContactPerson*,

`hasSupplier` y `hasCustomer` respectivamente, para cumplir con la convención de nombres de relaciones, que deben empezar con un verbo (Horridge, 2009).

Para poder acceder a un programa de provisión desde un socio de negocio se definieron las relaciones faltantes, `isCustomerIn` e `isSupplierIn`, entre los términos `Organization` y `EBD`. La relación `isCustomerIn` es inversa de la relación existente renombrada `hasCustomer` y la relación `isSupplierIn` es inversa de la relación `hasSupplier`.

### 3.1.5.2. Enriquecimiento de la representación de la entidad programa de provisión

Una de las características a ser enriquecida es el horizonte. De acuerdo a la Figura 3.3 (página 65), ejemplo de un programa de provisión, esta característica debe ser representada como un intervalo calendario, ya que un horizonte no es sólo una cantidad de tiempo sino que también tiene una posición en la línea de tiempo.

Muchas personas y proyectos han invertido sus esfuerzos en el problema de representar el conocimiento temporal (Ferguson, 2003; Hobbs y Pan, 2004). Particularmente, Hobbs y Pan (2004) han desarrollado una subontología básica, expresada en OWL y denominada *OWL-Time*, que proporciona la mayoría de las entidades y relaciones temporales básicas que la mayoría de las aplicaciones sencillas necesitan, es decir, un vocabulario para expresar hechos acerca de relaciones topológicas entre instantes, intervalos y eventos, junto con información sobre duraciones, fechas y horas. Su utilización ayuda también a minimizar el compromiso ontológico sobre conceptos temporales.

Tomando en cuenta la ontología *OWL-Time*, la característica horizonte de la entidad programa de provisión debe ser representada por medio de un término diferente, derivado del término `CalendarClockInterval` y vinculado al término `EBD` por una relación formal. Esto se muestra en la Figura 3.9, donde el prefijo `time:` indica que el elemento correspondiente pertenece a la ontología *OWL-Time* importada, y el prefijo `tzont:` indica que el elemento correspondiente pertenece a una ontología de zonas horarias importada por la ontología *OWL-Time*.

Otra característica a mejorar es la fecha. En este caso, la fecha refiere a un día calendario y no a cualquier intervalo de 24 horas de duración (Hobbs y Pan, 2004). Para representarla también se utiliza la ontología *OWL-Time*.

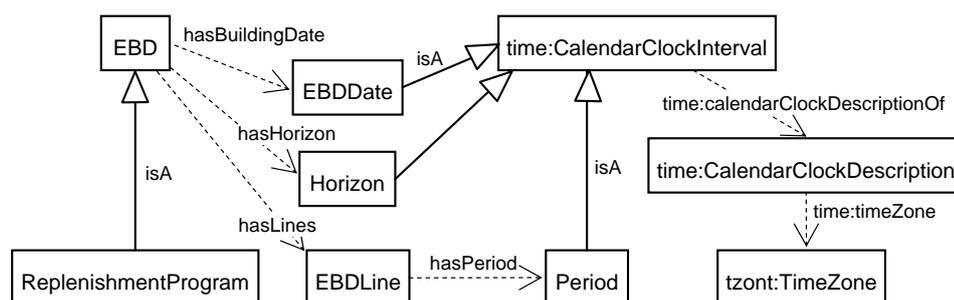


Figura 3.9: Una porción de la ontología DNE Enriquecida con la representación de la entidad programa de provisión

La característica fecha se representa por medio de un nuevo término derivado del término `CalendarClockInterval`, y vinculado al término `EBD` por una relación formal.

Para cumplir con la convención de nombres, la relación `lineCollection` se renombró como `hasLines`.

### 3.1.5.3. Enriquecimiento de la representación de la entidad ítem

De acuerdo a las conclusiones de la Sección 3.1.4.3, la representación de todas las características de la entidad ítem debe ser mejorada: período, producto, cantidad y precio.

Análogo a lo que se realizó con la característica horizonte, la característica período tiene que ser representada por medio de un término derivado del término `CalendarClockInterval` (Figura 3.9).

La característica producto necesita su propia representación como entidad de dominio, por lo cual se agrega el término `Product` y se lo relaciona con el término `EBDLine` (Figura 3.10). Las propiedades `PartNumber` e `ItemDescription` tienen que ser propiedades del término `Product` y no del término `EBDLine`.

Producto es una característica compleja de la entidad ítem, caracterizado a su vez por la marca impresa en los envases, el tipo de material con el que se hacen los envases y la capacidad de los mismos. Cada una de estas características tiene una dimensión asociada. El término `TrademarkDimension` representa la dimensión de la característica marca. Esta dimensión no es métrica, sino una enumeración de valores posibles. De la misma manera, el término `TypeDimension` representa la dimensión tipo, que es una enumeración de valores posibles (tales como *“carton”* y *“plastic”*). Por último, el término `CapacityDimension` representa

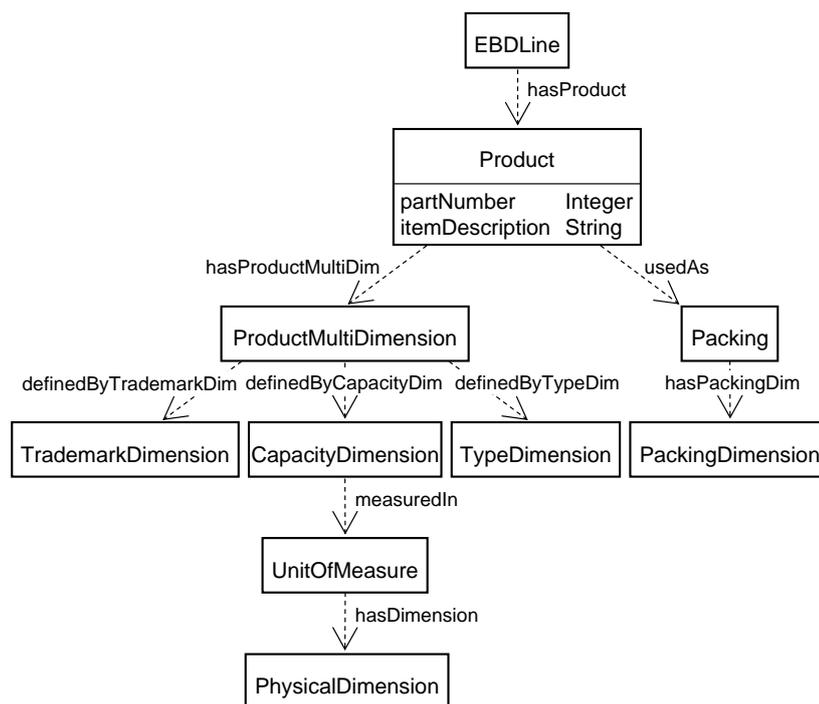


Figura 3.10: Una porción de la ontología DNE Enriquecida con la representación de la entidad producto

la dimensión de la característica capacidad. Esta dimensión es métrica, es decir, sus valores posibles están en el conjunto de los números no negativos (196, 250, etc.). Puesto que la capacidad de los envases es una cantidad asociada a una unidad de medida, el término `CapacityDimension` tiene una relación con el término `UnitOfMeasure`, quien a su vez está relacionado al término `PhysicalDimension`. Las tres dimensiones representadas por los términos `TrademarkDimension`, `TypeDimension` y `CapacityDimension` son dimensiones integrales de acuerdo a Guizzardi (2005) y componen una multidimensión representada por el término `ProductMultiDimension`, asociado a su vez al término `Product`.

El modelado de características complejas como producto se facilitaría si se contara con un lenguaje de modelado y una herramienta que inserte de forma automática todos los elementos necesarios para una representación adecuada y haga las validaciones correspondientes, como se mostrará en el Capítulo 5.

Si la formalización de la ontología se realiza en OWL, las dimensiones que son una enumeración de valores posibles se pueden implementar como clases enumeradas nombradas. Una clase enumerada nombrada se define listando las instancias que puede contener la clase (término) y definiendo tal enumeración

como una condición necesaria y suficiente de la clase. Por ejemplo, el código OWL siguiente muestra la definición del término `TypeDimension` como una clase enumerada nombrada, que sólo puede ser instanciada con las instancias `TypeDimensionPlastic` y `TypeDimensionCarton`:

```
<owl:Class rdf:ID="TypeDimension">
  <owl:equivalentClass>
    <owl:Class>
      <owl:oneOf rdf:parseType="Collection">
        <TypeDimension rdf:ID="TypeDimensionPlastic"/>
        <TypeDimension rdf:ID="TypeDimensionCarton"/>
      </owl:oneOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
```

Además, para garantizar que una instancia del término `ProductMultiDimension` está compuesta por una instancia del término `TrademarkDimension`, una instancia del término `TypeDimension` y una instancia del término `CapacityDimension`, y que la multidimensión se compone solamente de las tres dimensiones antes mencionadas, se deben agregar las siguientes restricciones en lógica descriptiva:

$$\begin{aligned}
& \forall \text{ definedByCapacityDim.CapacityDimension} \\
& \exists \text{ definedByCapacityDim.CapacityDimension} \\
& \quad = 1 \text{ definedByCapacityDim} \\
& \forall \text{ definedByTrademarkDim.TrademarkDimension} \\
& \exists \text{ definedByTrademarkDim.TrademarkDimension} \\
& \quad = 1 \text{ definedByTrademarkDim} \\
& \forall \text{ definedByTypeDim.TypeDimension} \\
& \exists \text{ definedByTypeDim.TypeDimension} \\
& \quad = 1 \text{ definedByTypeDim} \\
& \text{TrademarkDimension} \cap \text{TypeDimension} \subseteq \perp \\
& \text{TrademarkDimension} \cap \text{CapacityDimension} \subseteq \perp \\
& \text{CapacityDimension} \cap \text{TypeDimension} \subseteq \perp
\end{aligned}$$

En OWL esto mismo es:

```

<owl:Class rdf:ID="ProductMultiDimension">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Restriction>
          <owl:allValuesFrom rdf:resource="#CapacityDimension"/>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#definedByCapacityDim"/>
          </owl:onProperty>
        </owl:Restriction>
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:ID="definedByCapacityDim"/>
          </owl:onProperty>
          <owl:someValuesFrom rdf:resource="#CapacityDimension"/>
        </owl:Restriction>
        <owl:Restriction>
          <owl:cardinality rdf:datatype=
            "http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#definedByCapacityDim"/>
          </owl:onProperty>
        </owl:Restriction>
        <owl:Restriction>
          <owl:allValuesFrom rdf:resource="#TrademarkDimension"/>
          <owl:onProperty>
            <owl:ObjectProperty rdf:ID="definedByTrademarkDim"/>
          </owl:onProperty>
        </owl:Restriction>
        <owl:Restriction>
          <owl:someValuesFrom rdf:resource="#TrademarkDimension"/>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#definedByTrademarkDim"/>
          </owl:onProperty>
        </owl:Restriction>
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#definedByTrademarkDim"/>
          </owl:onProperty>
          <owl:cardinality rdf:datatype=
            "http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
        </owl:Restriction>
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#definedByTypeDim"/>
          </owl:onProperty>
          <owl:allValuesFrom rdf:resource="#TypeDimension"/>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

```

```

    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="definedByTypeDim"/>
      </owl:onProperty>
      <owl:someValuesFrom rdf:resource="#TypeDimension"/>
    </owl:Restriction>
  </owl:Restriction>
  <owl:Restriction>
    <owl:cardinality rdf:datatype=
      "http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
    <owl:onProperty>
      <owl:ObjectProperty rdf:about="#definedByTypeDim"/>
    </owl:onProperty>
  </owl:Restriction>
</owl:intersectionOf>
</owl:Class>
</owl:equivalentClass>
</owl:Class>

<owl:Class rdf:ID="CapacityDimension">
  <owl:disjointWith>
    <owl:Class rdf:ID="TrademarkDimension"/>
    <owl:Class rdf:ID="TypeDimension"/>
  </owl:disjointWith>
</owl:Class>

<owl:Class rdf:about="#TrademarkDimension">
  <owl:disjointWith rdf:resource="#TypeDimension"/>
  <owl:disjointWith rdf:resource="#CapacityDimension"/>
</owl:Class>

<owl:Class rdf:about="#TypeDimension">
  <owl:disjointWith>
    <owl:Class rdf:about="#TrademarkDimension"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#CapacityDimension"/>
</owl:Class>

```

Por último, se debe considerar la adición de términos puente. El uso que se intenta dar a la entidad producto es diferente dependiendo de la industria (contexto) en el que se la considera. La entidad producto refiere tanto a los productos fabricados por la industria de los envases como a los envases de los productos de la industria láctea. Luego, en la ontología DNE Base, se deben agregar dos términos puente: **Product** y **Packing**, para representar el uso previsto de la entidad en el contexto de la industria de envases y en el contexto de la industria láctea, respectivamente. Como **Product** ya está en la ontología,

sólo se debe agregar **Packing** y relacionarlo con el término **Product**. Como los términos puente representan características contextuales, ellos tienen que estar relacionados también con los términos que representan su dimensión. En el contexto de la industria láctea, los envases se asocian a una dimensión que es una enumeración de valores posibles.

Otra característica de la entidad ítem a ser enriquecida es la cantidad pedida de cada producto. Dado que ésta es una característica simple se asocia con una representación unidimensional, cuyos valores están en el conjunto de los números positivos. Para representar esta característica apropiadamente y seguir el criterio de enfoque de codificación mínima, se deben agregar tres elementos a la ontología DNE Base (Figura 3.11): (a) un término **QuantityDimension** que denota la dimensión; (b) una relación (**measuredIn**) entre este término y el término **UnitOfMeasure**, que representa la unidad de medida de la dimensión; y (c) una relación (**hasQuantity**) entre el término **QuantityDimension** y el término que representa la entidad ítem caracterizada (**EBDLine**). En el Capítulo 5 se mostrará una herramienta que implementa un lenguaje que permite modelar todos estos elementos de manera automática y realiza todas las validaciones necesarias.

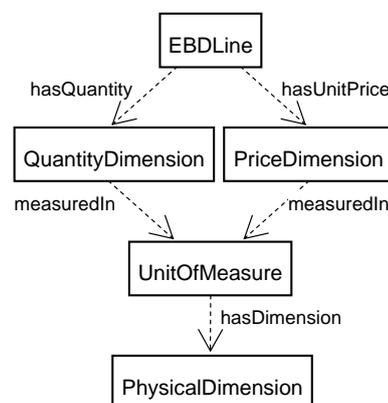


Figura 3.11: Una porción de la ontología DNE Enriquecida con la representación de las características cantidad y precio

Precio también es una característica simple, que representa el valor monetario de cada producto comercializado y no incluye impuestos. Sin embargo, el precio está representado por una propiedad. A fin de dar más flexibilidad al modelo, la característica precio debería ser representada por una relación formal entre los términos **EBDLine** y **PriceDimension** (Figura 3.11). Este último

término representa la dimensión precio, cuyos valores están en el conjunto de los números reales. La unidad monetaria asociada con el precio no está representada, así que se debe agregar también una relación entre los términos `PriceDimension` y `UnitOfMeasure` a este fin.

Por último, para poder acceder al programa de provisión que referencia un ítem se agrega la relación `isLineOf` entre los términos `EBDLine` y `EBD`, que es inversa de la relación existente renombrada `hasLines`.

La Figura 3.12 muestra un bosquejo de la ontología DNE Enriquecida.

### 3.1.6. Proceso 4: Verificación de la ontología enriquecida

La ontología base obtenida del Proceso 3 (Sección 3.1.2) se modificó según se describió en el Proceso 5 (Sección 3.1.4), haciendo uso del editor de ontologías Protégé-OWL. De esta forma se obtuvo la ontología enriquecida automáticamente en OWL. Luego se procedió a verificar la ontología DNE Enriquecida, obteniéndose las siguientes conclusiones:

- La ontología es completa para el propósito para el cual se la desarrolló.
  - A las entidades originalmente identificadas (Sección 3.1.1.3) se le sumó producto como entidad relevante, para el cual se creó una representación especial teniendo en cuenta su diferente interpretación en el contexto de la industria de envases y en el contexto de la industria láctea.
  - Todas las relaciones necesarias entre las entidades y todas sus características se representaron en la ontología.
  - La jerarquía formada por el término `Agent` se mejoró en su definición, al incluirse las restricciones faltantes de términos disjuntos y descomposición exhaustiva.
- La ontología es concisa. El único elemento que se detectó como innecesario, la propiedad `itemName`, se eliminó de la ontología.
- La ontología es más clara.

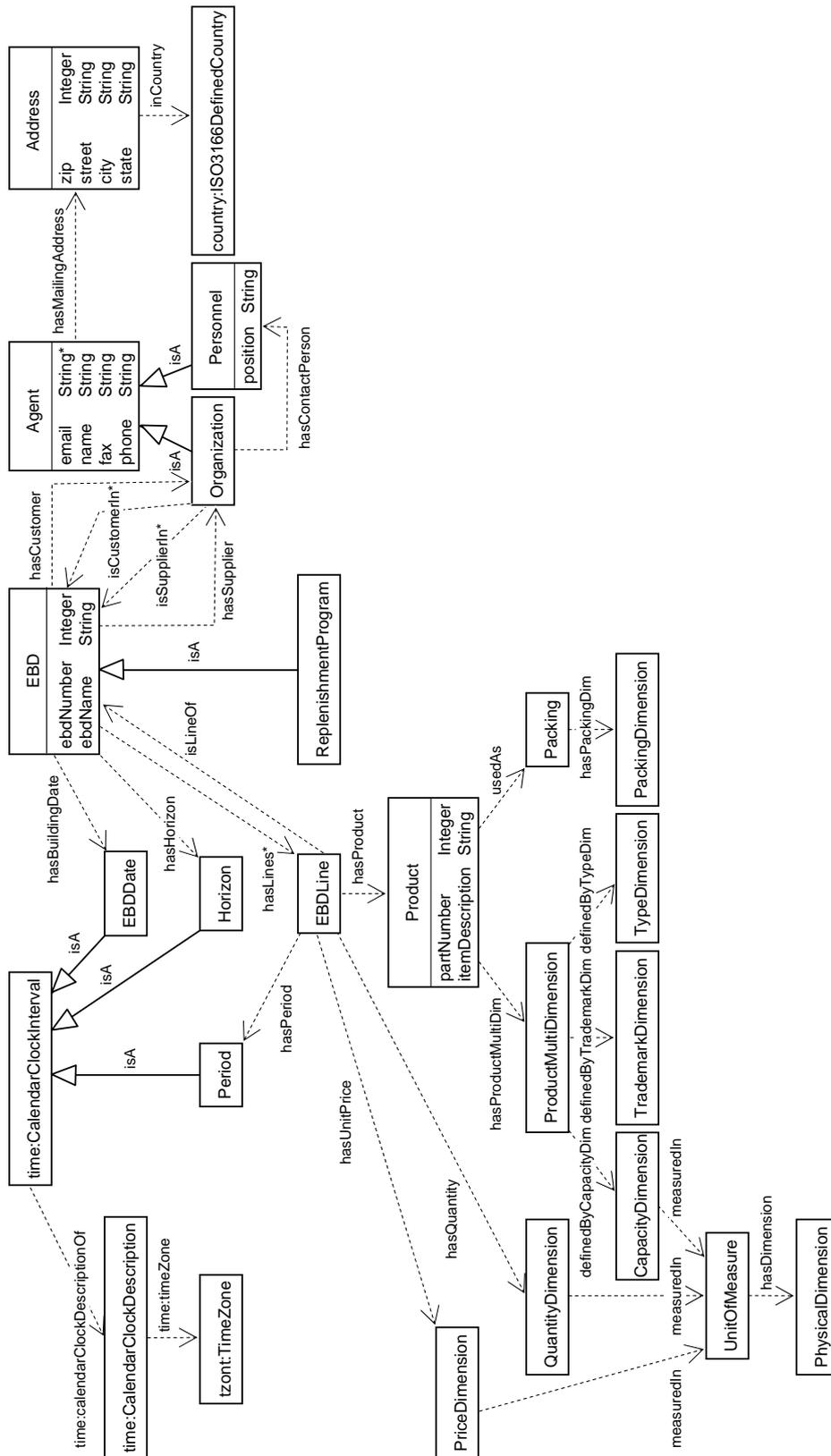


Figura 3.12: Ontología DNE Enriquecida

- La representación de las características se enriqueció para permitir una correcta interpretación en distintos contextos. Por ejemplo, con la nueva representación de la característica producto (ahora entidad) y sus características, los algoritmos de *matching* deberían descubrir que un producto así representado se corresponde con un envase para los productos en el contexto de la industria láctea.
- Se modificaron los nombres de algunos elementos en la ontología para respetar la convención de nombres adoptada.
- La ontología permite ser extendida para modelar otros documentos, además del programa de provisión, por ejemplo el plan de provisión y el cronograma de órdenes de provisión.
- Se redujeron los efectos de enfoque de codificación y compromisos ontológicos.

Además, la verificación de la coherencia lógica de la ontología no arrojó resultados negativos.

En base a los resultados obtenidos de este proceso de verificación, se debe decidir si se continúa enriqueciendo la ontología o se procede a validar la misma. En este caso se opta por la segunda opción ya que no es necesario seguir enriqueciendo la ontología.

### 3.1.7. Proceso 6: Validación de la ontología

Con el objetivo de realizar la validación de la ontología enriquecida, se debe someter a consideración de los expertos del dominio las respuestas obtenidas por la ontología a las preguntas de competencia.

Las preguntas de competencia, formalizadas en el lenguaje de consultas SPARQL (Prud'hommeaux y Seaborne, 2008), se definieron de manera estratificada, tal como sugieren Grüninger y Fox (1995). A modo de ejemplo, se muestran algunas de ellas.

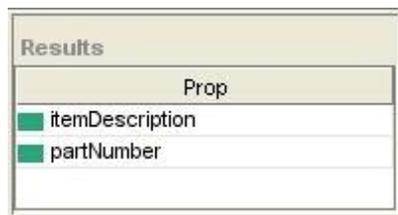
**Consulta 1:**

```

PREFIX ebd: <http://www.owl-ontologies.com/EnrichedEBDOnt.owl#>
SELECT ?Prop
WHERE
{
  ?Prop rdfs:domain ebd:Product .
  ?Prop ?z owl:DatatypeProperty
}

```

Esta consulta codifica la pregunta de competencia “¿cuáles son las características de la entidad producto que no han sido consideradas entidades en sí mismas?” El resultado obtenido fue el siguiente (Figura 3.13):



Prop
itemDescription
partNumber

Figura 3.13: Características de la entidad producto que no son entidades

**Consulta 2:**

```

PREFIX ebd: <http://www.owl-ontologies.com/EnrichedEBDOnt.owl#>
SELECT ?Rel ?MultiDimODim ?RelDim ?Dim
WHERE
{
  ?Rel rdfs:domain ebd:Product .
  ?Rel ?z owl:ObjectProperty .
  ?Rel rdfs:range ?MultiDimODim .
  ?RelDim rdfs:domain ?MultiDimODim .
  ?RelDim ?v owl:ObjectProperty .
  ?RelDim rdfs:range ?Dim
}

```

Esta consulta implementa la pregunta de competencia “¿con qué entidades, o características consideradas entidades, se relaciona la entidad producto?” El resultado obtenido fue el siguiente (Figura 3.14):

Results			
Rel	MultiDimODim	RelDim	Dim
hasProductMultiDim	ProductMultiDimension	definedByTrademarkDim	TrademarkDimension
hasProductMultiDim	ProductMultiDimension	definedByTypeDim	TypeDimension
hasProductMultiDim	ProductMultiDimension	definedByCapacityDim	CapacityDimension
usedAs	Packing	hasPackingDim	PackingDimension

Figura 3.14: Entidades y características relacionadas a la entidad producto

### Consulta 3:

```

PREFIX ebd: <http://www.owl-ontologies.com/EnrichedEBDOnt.owl#>
SELECT ?Dim
WHERE
{
  ebd:measuredIn ?a ?Dim FILTER (?a = rdfs:domain)
}

```

Esta consulta refiere a la pregunta de competencia “¿cuáles características son medibles?” El resultado obtenido fue el siguiente (Figura 3.15):

Results	
Dim	
CapacityDimension	or QuantityDimension or PriceDimension

Figura 3.15: Características medibles

Considerando aceptables los resultados obtenidos se da por finalizado el desarrollo de la ontología. En este momento, la ontología DNE Enriquecida pasa a denominarse ontología DNE Contextualizada.

## 3.2. Análisis Comparativo de la Calidad de las Ontologías Base y Contextualizada

En esta sección se analizan los valores obtenidos al aplicar los indicadores de cada una de las dimensiones de calidad descritas en la Sección 2.4 del Capítulo 2, calculados para las ontologías DNE Base y DNE Contextualizada de

la Sección 3.1. El propósito es evaluar la mejora obtenida con el enriquecimiento de la ontología DNE Base, obtenida por un método automático de aprendizaje.

### 3.2.1. Evaluación de la dimensión sintáctica

La Tabla 3.5 muestra los valores de los indicadores para esta dimensión.

Dimensión Sintáctica		
Indicador	DNE Base	DNE Contextualizada
<b>Nivel de Jerarquía o Taxonomía</b>		
Complejidad		
Ancho (A)	0,4286	0,6250
Profundidad (PF)	1,6000	1,8889
Tamaño (total de términos hoja, TH)	5	36
Número total de términos (T)	7	48
Número total de relaciones	8	94
Relaciones jerárquicas (RJ)	3	30
Relaciones no jerárquicas (RNJ)	5	64
Número total de atributos (AT)	20	47
<b>Otras Relaciones Semánticas</b>		
Riqueza de las relaciones (RR)	0,6250	0,6809
<b>Nivel Sintáctico</b>		
Riqueza (R)	0,2000	0,5333
Restricciones existenciales	0	22
Restricciones universales	25	35
Restricciones de valor	0	0
Restricciones de cardinalidad máxima	0	21
Restricciones de cardinalidad mínima	4	6
Restricciones de cardinalidad	21	40
Propiedad transitiva	0	1
Propiedad funcional	0	19
Propiedad simétrica	0	0
Propiedad inversa	0	32
Clase equivalente	0	10
Complemento	0	0
Disyunto	0	64
Todos diferentes	0	0
Diferente	0	0
Riqueza de atributos (RA)	2,8571	0,9792

Tabla 3.5: Algunos indicadores para la dimensión sintáctica

Como era de esperar, la ontología DNE Contextualizada es más compleja que la ontología DNE Base (todos los valores de la columna DNE Base son menores que los correspondientes de la columna DNE Contextualizada para el indicador Complejidad), pero es más rica en cuanto a relaciones semánticas (RR) y la proporción de características del lenguaje utilizadas (riqueza del nivel sintáctico, R). La especificidad no se puede calcular ya que estas ontologías no se han poblado aún con la información a ser intercambiada.

En cuanto a la riqueza de atributos (RA), la ontología DNE Contextualizada tiene un valor inferior a la ontología DNE Base. Esto se debe a que muchas propiedades de esta última ontología se convirtieron en términos para poder enriquecer su representación. Si bien Tartir y otros (2005) asumen que a mayor valor de este indicador, más conocimientos transmite la ontología, también se puede concluir que a mayor valor de este indicador, mayores son las posibilidades de tener características pobremente representadas, como es el caso de la ontología DNE Base.

### 3.2.2. Evaluación de la dimensión semántica

La Tabla 3.6 muestra los valores obtenidos para los indicadores tanto del nivel léxico, de vocabulario o de datos (Tartir y otros, 2005) como de los indicadores propuestos en esta tesis para medir los principios de diseño, para las ontologías DNE Base y DNE Contextualizada. En el Apéndice C (página 223) se presentan detalles del cálculo para estos indicadores.

<b>Dimensión Semántica</b>		
<b>Indicador</b>	<b>DNE Base</b>	<b>DNE Contextualizada</b>
<b>Nivel léxico, de vocabulario o de datos</b>		
Interpretabilidad (I)	0,9615	0,9293
Claridad (C)	3,5185	1,5368
<b>Principio 1: Entidades (PE)</b>	1,0000	1,0000
<b>Principio 2: Uso Previsto de Entidades (PU)</b>	0,6667	1,0000
<b>Principio 3: Relaciones entre Entidades (PR)</b>	0,3000	0,6000
<b>Principio 4: Características como Entidades</b>		
a. Características Simples (PCS)	0,5000	1,0000
b. Características Simples Medibles (PCM)	0,4167	0,7500
c. Características Complejas (PCC)	0,0000	1,0000
<b>Principio 5: Características Comunes (PCc)</b>	0,7273	0,7273

Tabla 3.6: Algunos indicadores para la dimensión semántica

En el nivel léxico, de vocabulario o de datos se puede observar que la ontología DNE Base es, en principio, más interpretable que la ontología DNE Contextualizada. En la ontología DNE Base, de veintiséis (26) palabras distintas utilizadas para nombrar términos y propiedades, veinticinco (25) tienen al menos un sentido en WordNet (Tabla C.1, página 224). En cambio, la ontología DNE Contextualizada tiene noventa y dos (92) palabras con al menos un sentido en WordNet, de las noventa y nueve (99) utilizadas (Tabla C.5, página 228). Sin

embargo, se debe considerar que la ontología DNE Contextualizada importa las ontologías *OWL-Time* y *Country* y que de las siete (7) palabras que no tienen un sentido en WordNet, sólo una pertenece a la ontología DNE Contextualizada (la misma palabra que no tiene un sentido en la ontología DNE Base, “EBD”). De las seis palabras restantes, tres pertenecen a la ontología *OWL-Time* y tres a la ontología *Country*. Si sólo se consideran los elementos propios de la ontología DNE Contextualizada, sin tener en cuenta a los importados, entonces la Interpretabilidad (I) de esta ontología es de 0,9714 (Tabla C.3, página 226), un 1,03% mayor a la interpretabilidad de la ontología DNE Base.

Respecto de la claridad (C) se puede observar que la ontología DNE Contextualizada es más clara que la ontología DNE Base (cuanto más cerca a uno es el valor, más clara es la ontología). Sin embargo, este indicador presenta el inconveniente que en WordNet las frases no están registradas. Esto produce una distorsión en el valor obtenido, ya que ambas ontologías utilizan frases para nombrar tanto a términos como a propiedades. Si no se consideran las frases para calcular este indicador se obtienen los siguientes valores: 5 (= 95/19) para la ontología DNE Base y 4,06 (= 146/36) para la ontología DNE Contextualizada, con lo cual esta última sigue siendo más clara. De todas maneras, la forma de cálculo de este indicador debería ser mejorado. Una alternativa sería aplicar una estrategia similar a la empleada por Castano y otros (2004) para manejar nombres compuestos. Este trabajo está fuera del alcance de esta tesis.

En cuanto a los valores de los indicadores de los principios seguidos en esta tesis se puede observar que:

- Todas las entidades inicialmente identificadas están correctamente representadas en ambas ontologías (PE = 1).
- La ontología DNE Base no contempla todos los usos previstos de las entidades (PU), como sí lo hace la ontología DNE Contextualizada. No considerarlos puede acarrear malas interpretaciones a la hora de intercambiar información en un entorno abierto.
- Ninguna de las ontologías representa todas las relaciones identificadas (PR < 1 en ambos casos). La ontología DNE Contextualizada incorpora sólo las relaciones necesarias para responder a las preguntas de competencia, sin incurrir en redundancias.

- La ontología DNE Base falla en la representación de las características como entidades (en ninguno de los tres casos –PCS, PCM y PCC– el indicador supera a 0,5) e implica que existe información de contexto implícita en la representación de estas características. La ontología DNE Contextualizada no alcanza el valor 1 para el indicador de las características simples medibles (PCM) debido a que las características temporales se modelan utilizando la ontología *OWL-Time*, que no considera la representación de la dimensión física (tiempo) asociada a la unidad de medida.
- Ambas ontologías obtienen el mismo valor para el indicador de características comunes (PCc), el cual es inferior a uno ya que no todas las características requieren ser explícitamente representadas (por ejemplo, el número de la calle de una dirección).

### 3.2.3. Evaluación de la dimensión pragmática

Para la evaluación de esta dimensión se presentaron tres indicadores, propuestos por Burton-Jones y otros (2005): exactitud, integralidad y relevancia. Respecto de la exactitud se puede decir que las afirmaciones realizadas por ambas ontologías son ciertas, sólo que esas afirmaciones se hacen a un nivel de detalle diferente en cada una de las ontologías.

La integralidad es una medida del tamaño de la ontología. En base a los valores obtenidos para los indicadores de las dimensiones sintáctica y semántica, se puede decir que la ontología DNE Contextualizada es más integral que la ontología DNE Base, ya que representa con mayor detalle las entidades, sus relaciones y características.

La relevancia no se puede determinar para las ontologías DNE Base y DNE Contextualizada, ya que se desconocen las necesidades propias de un agente de software que pudiera hacer uso de estas ontologías.

### 3.3. Evaluación de los Procesos de Matching de Ontologías

Para poder traducir la información contenida en la ontología del DNE a los procesos internos de los socios de negocio (escenario de motivación EM002, Sección 3.1.1.1) es necesario que se hayan definido previamente las reglas de conversión correspondientes (Rico y otros, 2009). Con este objetivo, en tiempo de diseño de la colaboración se debe identificar, caracterizar y establecer las correspondencias entre las entidades representadas en las ontologías intervinientes, es decir, realizar un *matching* entre las mismas. El objetivo de esta sección es mostrar cómo el enriquecimiento de la representación de las entidades en las ontologías ofrece, a las técnicas de *matching*, mayor posibilidad de encontrar correspondencias correctas.

Con tal fin se considera el proceso de intercambio de la información contenida en un DNE y el sistema de información de uno de los socios de negocio: la industria láctea. Para analizar los *alignments* obtenidos al someter al proceso de *matching* las ontologías involucradas, se toma como ontología “destino” una ontología privada de la industria láctea, y como ontología “origen” la ontología del DNE con distintos grados de enriquecimiento (ontología base; ontología contextualizada, enriquecida con los usos previstos por la industria de envases; y ontología contextualizada completa, enriquecida con los usos previstos por las industrias de envases y láctea).

Dado que las ontologías “origen” y “destino” consideradas no refieren al mismo contexto, el grado de solapamiento es reducido. Por tal razón, para disminuir el tamaño de la presentación, el análisis comparativo de los *alignments* se centra en la representación de la entidad producto, común a ambas ontologías y con usos previstos distintos según el contexto en el que se la considere.

Diversas herramientas y técnicas para realizar el *matching* de ontologías están disponibles. Una lista de varias de ellas se puede encontrar en (Shvaiko y Euzenat, 2011). Sin embargo, no todas ellas son adecuadas para operar en sistemas abiertos, debido a sus limitaciones en términos de altos tiempos de cálculo y soporte restringido a diferentes lenguajes de especificación de ontologías (Castano y otros, 2007).

Si bien se utilizaron varias de estas herramientas, los resultados que se

analizan inicialmente corresponden a los obtenidos con el *plugin* H-Match 1.5 para Protégé<sup>4</sup> debido a las siguientes consideraciones: (a) H-Match puede tratar con diferentes lenguajes de especificación de ontologías, particularmente OWL Full, en el cual están implementadas las ontologías a ser sometidas al proceso de *matching*; (b) H-Match considera tanto los nombres dados a los elementos de la ontología como las relaciones que ellos tienen con otros elementos; (c) H-Match se concibió específicamente para operar en sistemas abiertos distribuidos; (d) H-Match puede tratar con diferentes niveles de riqueza en la descripción de las ontologías, ya que se puede configurar dinámicamente de acuerdo a las características del escenario de *matching* considerado.

El algoritmo H-Match provee un ranking de similaridad entre los elementos de dos ontologías. Para ello calcula una afinidad semántica de dos elementos teniendo en cuenta sus afinidades lingüísticas y contextuales. Las primeras refieren a los nombres de los elementos de las ontologías y sus significados. Las segundas refieren a las propiedades, relaciones y términos relacionados directamente con un término dado.

A continuación se analizan los resultados de los procesos de *matching* entre las ontologías DNE, para cada grado de enriquecimiento considerado, y privada.

### 3.3.1. Matching entre las ontologías DNE Base y Privada

Como se muestra en la Figura 3.7, en la ontología DNE Base producto no está representado como entidad; sólo algunas propiedades del término EBDLine refieren a él (Figura 3.16): `partNumber`, `itemName` e `itemDescription`, como se describió la Sección 3.1.4.3.

EBDLine	
<code>period</code>	String
<code>partNumber</code>	Integer
<code>itemName</code>	String
<code>itemDescription</code>	String
<code>quantity</code>	Integer
<code>price</code>	Float

Figura 3.16: Representación de la entidad producto en la ontología DNE Base

La Figura 3.17 muestra parte de la ontología Privada considerada. Teniendo en cuenta que el producto al que se hace referencia en la ontología DNE Base

<sup>4</sup> <http://islab.dico.unimi.it/hmatch/downloads.php>

es tanto un elemento elaborado por la industria de envases como un envase para algún producto de la industria láctea, en la ontología Privada corresponde considerar a los términos `ExternalMaterial` y `PackingMaterial`, sus propiedades y la relación `isA` entre ellos como representación de la entidad producto.

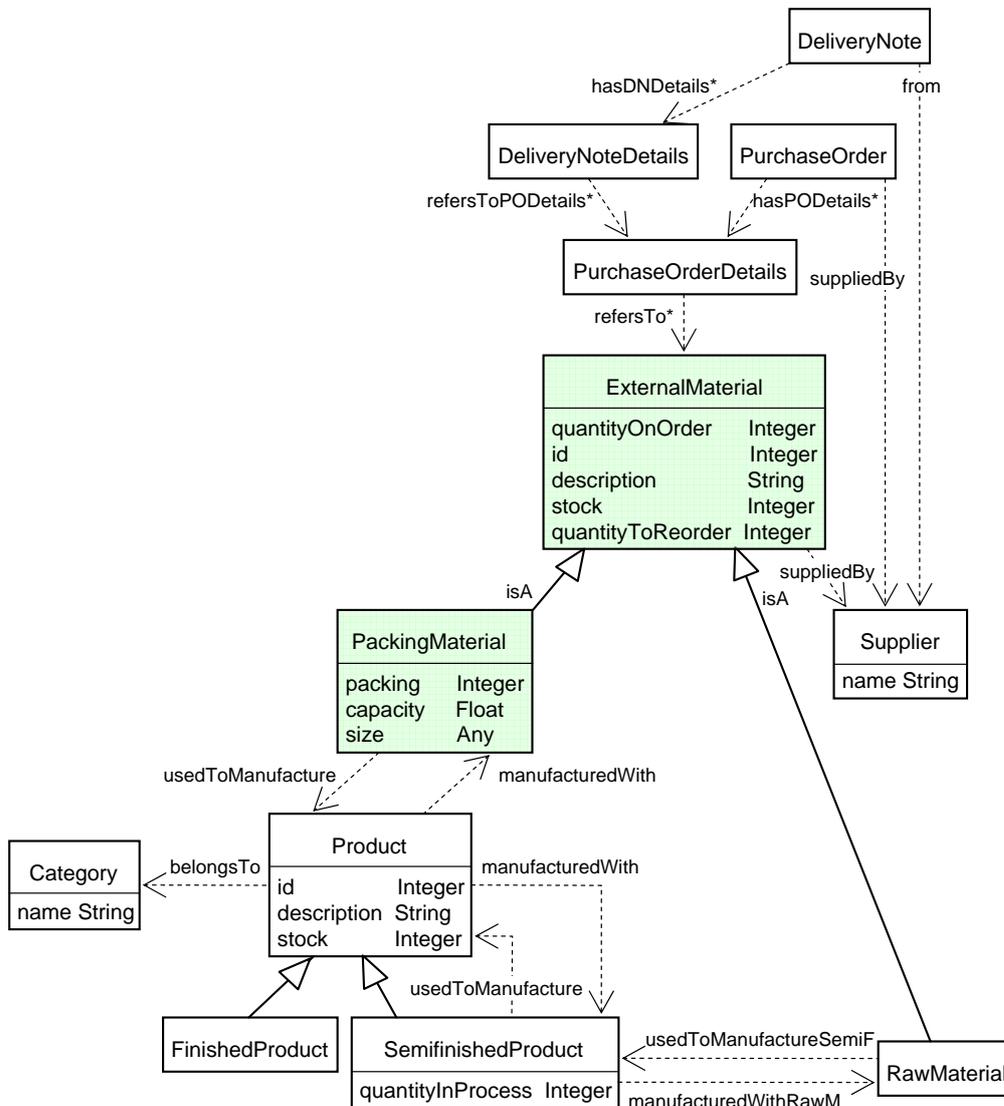


Figura 3.17: Ontología Privada

La ejecución del proceso de *matching* automático entre las ontologías antes mencionadas arrojó el *alignment*  $A_1$ , que se muestra en la parte izquierda de la Tabla 3.7, para los elementos involucrados en la representación de la entidad producto. En la misma tabla se pueden observar, a la derecha, las correspondencias esperadas, surgidas del análisis del ingeniero ontológico. La fila sombreada señala la coincidencia entre los *alignments*  $A_1$  y  $E_1$ .

<i>Alignment</i> H-Match $A_1$			<i>Alignment</i> Esperado $E_1$	
Ontología DNE Base	Ontología Privada	Afinidad semántica	Ontología DNE Base	Ontología Privada
<b>Propiedad</b>			<b>Propiedad</b>	
itemName	name	0,8	itemName	capacity
itemDescription	description	0,8	itemDescription	description

Tabla 3.7: *Alignments*  $A_1$  y  $E_1$  de elementos involucrados en la representación de la entidad producto

En la ontología DNE Base, la propiedad `partNumber` representa el número de parte asignado al envase por su fabricante; no tiene equivalente en la ontología Privada. En coincidencia con este hecho, en el *alignment*  $A_1$  no se observan correspondencias para este elemento.

La propiedad `itemName` en la ontología DNE Base se utiliza para describir la marca impresa en los envases, el tipo de material de los envases y la capacidad de los mismos. De estas características sólo la capacidad está representada en la ontología Privada (la propiedad `capacity` del término `PackingMaterial`). Sin embargo, en el *alignment*  $A_1$  `itemName` tiene una correspondencia con la propiedad `name` asociada a los términos `Supplier` y `Category`, lo cual no es correcto.

Finalmente, la propiedad `itemDescription` utilizada para dar una descripción en lenguaje natural del producto en la ontología DNE Base, se representa en la ontología Privada por la propiedad `description` del término `ExternalMaterial`. Esto se corresponde con el resultado observado en el *alignment*  $A_1$ .

Es importante notar que, para la representación de producto en la ontología DNE Base, el *alignment* obtenido con H-Match no arroja correspondencias que relacionen correctamente elementos de la ontología DNE Base con los términos `ExternalMaterial` y `PackingMaterial` de la ontología Privada de la industria láctea.

### 3.3.2. Matching entre las ontologías DNE Contextualizada, uso previsto industria de envases, y Privada

A continuación se considera una ontología “origen” contextualizada, en la cual sólo se representa el uso previsto de las entidades por parte de la industria de envases. La Figura 3.18 muestra una porción de dicha ontología. En esta ontología, producto está representado como entidad (el término `Product`). La

ontología “destino” es la misma ontología Privada utilizada en la sección anterior.

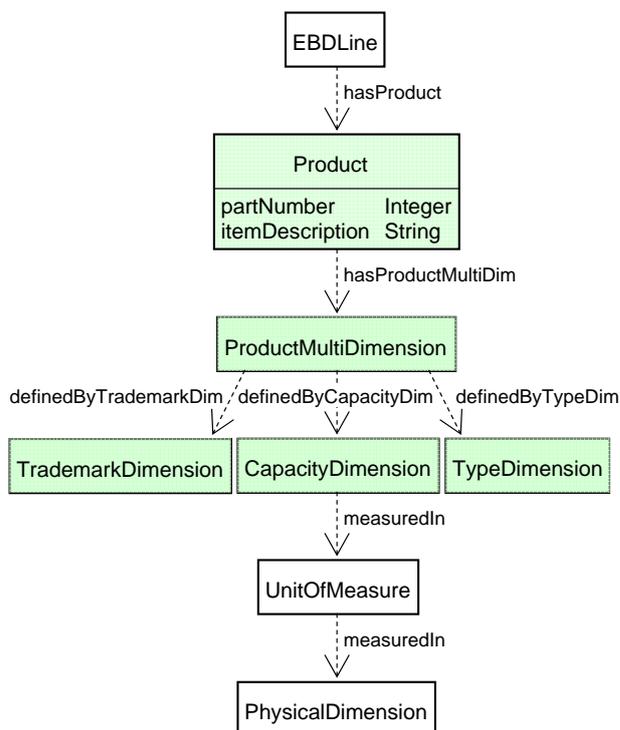


Figura 3.18: Representación de la entidad producto en la ontología DNE Contextualizada, según su uso previsto en la industria de envases

El *matching* entre ambas ontologías dio por resultado las correspondencias que se muestran en la parte izquierda de la Tabla 3.8, para los elementos intervinientes en la representación de la entidad producto. A la derecha de dicha tabla se observan las correspondencias esperadas.

<i>Alignment</i> H-Match $A_2$			<i>Alignment</i> Esperado $E_2$	
Ontología DNE Contextualizada	Ontología Privada	Afinidad semántica	Ontología DNE Contextualizada	Ontología Privada
Uso previsto industria de envases			Uso previsto industria de envases	
<b>Término</b>			<b>Término</b>	
Product	Product	1	Product	ExternalMaterial
ProductMultiDimension	Product	0,7	Product	PackingMaterial
TrademarkDimension	Product	0,7		
TypeDimension	Product	0,7		
CapacityDimension	Product	0,7	CapacityDimension	capacity (*)
<b>Propiedad</b>			<b>Propiedad</b>	
itemDescription	description	0,8	itemDescription	description

(\*) capacity es una propiedad

Tabla 3.8: *Alignments*  $A_2$  y  $E_2$  de elementos involucrados en la representación de la entidad producto

La representación de producto en la ontología DNE Contextualizada refiere al uso previsto de dicha entidad por parte de la industria de envases. Este hecho hace que varios elementos de la ontología DNE Contextualizada, que representan características de la entidad producto como la marca y el tipo del mismo, no tengan un equivalente en la ontología Privada de la industria láctea.

En la Tabla 3.8 se puede observar que mientras se espera que el término **Product** se corresponda con los términos **ExternalMaterial** y **PackingMaterial**, el *alignment*  $A_2$  lo relaciona con el término **Product** de la ontología Privada, lo cual no es correcto. A pesar de estar producto representado como entidad en la ontología DNE Contextualizada, esto no es suficiente para relacionarlo correctamente con su representación en la ontología Privada de la industria láctea.

Por otro lado, el sistema H-Match no está preparado para encontrar la correspondencia esperada  $\langle \text{CapacityDimension}, \text{capacity} \rangle$ , que implica relacionar un término de una ontología con una propiedad de la otra ontología.

### 3.3.3. Matching entre las ontologías DNE Contextualizada completa y Privada

En la ontología DNE Contextualizada considerada en esta sección, la entidad producto se representa teniendo en cuenta su uso previsto tanto por la industria de envases como por la industria láctea. La Figura 3.19 muestra dicha representación.

El *matching* entre esta ontología DNE Contextualizada y la ontología Privada de las secciones previas da por resultado el *alignment*  $A_3$  que se muestra a la izquierda de la Tabla 3.9, para los elementos involucrados en la representación de la entidad producto. En la parte derecha de dicha tabla se pueden observar las correspondencias esperadas.

De los resultados mostrados en la Tabla 3.9 se puede concluir que sólo cuando se representó la entidad producto con el uso previsto por la industria láctea, el sistema de *matching* H-Match encontró correspondencias correctas entre los términos principales de las representaciones de la entidad producto en el contexto del proceso colaborativo y en el contexto de la industria láctea. A pesar de ello, este sistema de *matching* sigue asignando un valor de afinidad semántica de 1 a la correspondencia entre los términos **Product** de ambas ontologías. Una

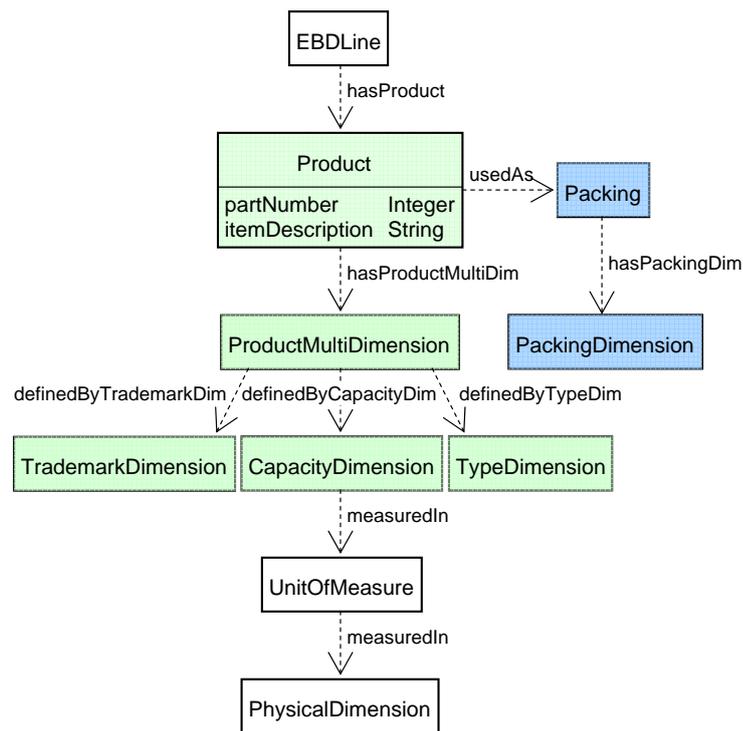


Figura 3.19: Representación de la entidad producto en la ontología DNE Contextualizada, según sus usos previstos en las industrias de envases y láctea

<i>Alignment</i> H-Match $A_3$			<i>Alignment</i> Esperado $E_3$	
Ontología DNE Contextualizada	Ontología Privada	Afinidad semántica	Ontología DNE Contextualizada	Ontología Privada
Usos previstos ind. envase y láctea			Usos previstos ind. envase y láctea	
<b>Término</b>			<b>Término</b>	
Product	Product	1	Product	ExternalMaterial
ProductMultiDimension	Product	0,7	Product	PackingMaterial
TrademarkDimension	Product	0,7		
TypeDimension	Product	0,7		
CapacityDimension	Product	0,7	CapacityDimension	capacity (*)
Packing	ExternalMaterial	0,892	Packing	ExternalMaterial
Packing	PackingMaterial	0,892	Packing	PackingMaterial
Packing	RawMaterial	0,892		
<b>Propiedad</b>			<b>Propiedad</b>	
itemDescription	description	0,8	itemDescription	description

(\*) capacity es una propiedad

Tabla 3.9: *Alignments*  $A_3$  y  $E_3$  de elementos involucrados en la representación de la entidad producto

razón para ello puede ser que sólo se enriqueció la representación de las entidades en una de las ontologías involucradas, la ontología “origen”, ya que modificar la ontología “destino” implicaría, en este caso, afectar a los sistemas de información de la industria láctea que usan dicha ontología. En otros trabajos, donde se

enriqueció la representación de las entidades en las dos ontologías sometidas al proceso de *matching*, se obtuvieron mejores resultados respecto de este tipo de correspondencias (Rico y otros, 2008, 2011).

También es importante notar que el sistema H-Match asigna el mismo valor de afinidad semántica a las correspondencias <Packing, PackingMaterial> y <Packing, RawMaterial>. Este sistema de *matching* no puede discernir cuál de las correspondencias es correcta, a pesar que en la ontología Privada los términos PackingMaterial y RawMaterial son disjuntos entre sí, y que el término PackingMaterial tiene una relación particular definida con el término Product, que el término RawMaterial carece.

### 3.3.4. Evaluación de los alignments obtenidos

Con el fin de evaluar el grado de mejora en los resultados de un proceso de *matching* al enriquecer las ontologías sometidas a dicho proceso, se definen las siguientes métricas:

- Índice de mejora esperada ( $IE$ ) es una medida que evalúa el grado de mejora potencial para un proceso de *matching* respecto de un *alignment* de referencia.

Sea  $Er$  el número total de correspondencias esperadas para un proceso de *matching* aplicado a ontologías de referencia. Sea  $Ee$  el número total de correspondencias esperadas para un proceso de *matching* aplicado a las ontologías de referencia enriquecidas. Luego,

$$IE = \begin{cases} \frac{Ee - Er}{Er} & \text{si } Er > 0 \\ Ee & \text{si } Er = 0 \end{cases}$$

- Índice de mejora de correspondencias correctas ( $IC$ ) es una medida que evalúa cuánto influye el enriquecimiento de las ontologías para permitir que el proceso de *matching* encuentre mayor cantidad de correspondencias correctas.

Sea  $Cr$  el número total de correspondencias correctas encontradas en el proceso de *matching* aplicado a ontologías de referencia. Sea  $Ce$  el número

total de correspondencias correctas encontradas en el proceso de *matching* aplicado a las ontologías de referencia enriquecidas. Luego,

$$IC = \begin{cases} \frac{Ce - Cr}{Cr} & \text{si } Cr > 0 \\ Ce & \text{si } Cr = 0 \end{cases}$$

La Tabla 3.10 muestra, para las tres ejecuciones de *matching* efectuadas –*Alignment* (1), *Alignment* (2) y *Alignment* (3)–, el número de correspondencias devueltas por el sistema de *matching* H-Match (A), el número de correspondencias que se espera que el proceso de *matching* devuelva (E), el número de correspondencias correctas encontradas por H-Match ( $C = A \cap E$ ), el índice de mejora esperada (IE) respecto del *Alignment* (1) de referencia y el índice de mejora de correspondencias correctas (IC) respecto del *Alignment* (1) de referencia. *Alignment* (1) refiere al obtenido de someter al proceso de *matching* a las ontologías DNE Base y Privada; el (2), a las ontologías DNE Contextualizada, con el uso previsto de la entidad producto por parte de la industria de envases, y Privada; y el (3), a las ontologías DNE Contextualizada completa (con los usos previstos de la entidad producto tanto por parte de la industria de envases como de la industria láctea) y Privada.

	<i>Alignments</i>		
	(1)	(2)	(3)
A	2	6	9
E	2	4	6
$C = A \cap E$	1	1	3
IE (%)		100	200
IC (%)		0	200

Tabla 3.10: Comparación de *alignments* obtenidos sobre elementos involucrados en la representación de la entidad producto

En la Tabla 3.10 se puede observar que los valores del índice de mejora esperada (IE) aumentan a medida que la ontología “origen” se enriquece: para el *Alignment* (2) la mejora es del 100% y para el *Alignment* (3) la mejora es del 200%, ambas respecto del *Alignment* (1) tomado como referencia. Analizando el índice de mejora de correspondencias correctas (IC), también se observa que para el *Alignment* (3) la mejora es del 200% respecto del *Alignment* (1) de referencia.

La Tabla 3.11 muestra los resultados completos de los *Alignments* (1) y (3). Para la entidad socio de negocio se recuperan cuatro correspondencias correctas

antes de ser enriquecida, y seis luego de este proceso (una mejora del 50%). La entidad programa de provisión no tiene una representación en la ontología Privada considerada, y por esta razón no existen correspondencias entre los elementos que la representan en la ontología “origen” y los elementos de la ontología Privada.

	DNE Base - Ontología Privada				DNE Contextualizada - Ontología Privada			
	Socio de Negocio	Programa de Provisión	Ítem	Total	Socio de Negocio	Programa de Provisión	Ítem	Total
A	22	0	9	31	38	0	65	103
E	12	0	2	14	14	0	6	20
$C = A \cap E$	4	0	1	5	6	0	3	9
IE (%)					16,67	0,00	200,00	42,86
IC (%)					50,00	0,00	200,00	80,00

Tabla 3.11: Comparación de *alignments* obtenidos sobre todas las entidades consideradas en la e-colaboración

En general se puede observar que el proceso de enriquecimiento aplicado a la ontología base, obtenida de un proceso de aprendizaje de ontologías, aumenta la cantidad de elementos en la ontología contextualizada resultante, y con ello también aumentan las correspondencias esperadas (cuyo incremento es, en este caso, de un 42,86% respecto del *Alignment* (1) de referencia). Este enriquecimiento de la ontología base provee de una entrada más rica al proceso de *matching* y le da la posibilidad al sistema para que recupere mayor cantidad de correspondencias correctas (el incremento en la cantidad de correspondencias correctamente recuperadas fue del 80% respecto del *Alignment* (1) de referencia).

### 3.3.5. Análisis de otros sistemas de matching

Con el fin de independizar los resultados obtenidos de la herramienta de *matching* utilizada, la evaluación también se realizó con los sistemas que obtuvieron los mejores desempeños en la campaña 2010 de la Iniciativa de Evaluación de *Alignments* de Ontologías (*Ontology Alignment Evaluation Initiative*, OAEI), en las sesiones *benchmark*, anatomía y conferencia (Euzenat y otros, 2010), disponibles al momento de escribirse esta tesis.

Los sistemas de *matching* *AgreementMaker*<sup>5</sup>, RiMOM<sup>6</sup> y Falcon-AO<sup>7</sup> se

<sup>5</sup> <http://agreementmaker.org/>

<sup>6</sup> <http://keg.cs.tsinghua.edu.cn/project/RiMOM/>

<sup>7</sup> <http://ws.nju.edu.cn/falcon-ao/>

utilizaron para realizar el *matching* entre las ontologías DNE Base y Privada de la industria láctea, y entre las ontologías DNE Contextualizada completa y la misma ontología Privada, con parámetros de configuración similares a los utilizados para ejecutar el sistema H-Match. Los *alignments* obtenidos, respecto de la representación de la entidad producto, se muestran en las Tablas 3.12 y 3.13 y se comparan con los *alignments* producidos con H-Match.

Ontología DNE Base	Ontología Privada	Afinidad semántica			
		H-Match	AgreementMaker	RiMOM	Falcon-AO
<b>Propiedad</b>					
partNumber					
itemName	name	0,8000			
itemDescription	description	0,8000	0,7502		0,7775
A		2	1	0	1
E		2	2	2	2
$C = A \cap E$		1	1	0	1

Tabla 3.12: Comparación de *alignments* obtenidos por distintos sistemas de *matching* para la entidad producto entre las ontologías DNE Base y Privada

En la Tabla 3.12 se puede observar que dos de los nuevos sistemas considerados (*AgreementMaker* y Falcon-AO) recuperan la misma cantidad de correspondencias correctas que el sistema H-Match a la vez que obtienen menos correspondencias (A), reduciendo la posibilidad de error. La fila sombreada en la tabla refiere a la única correspondencia esperada y correctamente encontrada.

En la Tabla 3.13 se observa que todos los nuevos sistemas considerados recuperan una correspondencia correcta menos que H-Match, y también obtienen menor cantidad de correspondencias (A) que este último. Respecto del índice de mejora de correspondencias correctas y tomando como *alignment* de referencia al correspondiente a cada sistema en la Tabla 3.12, se observa que el enriquecimiento de la ontología base produce un incremento de este índice para todos los sistemas considerados (las filas sombreadas en la tabla refieren a las correspondencias esperadas correctamente recuperadas). En cuanto a los valores de la afinidad semántica de las correspondencias correctas, sólo en una correspondencia (*<itemDescription, description>*) el valor obtenido por el sistema H-Match es superado.

La fila sombreada y subrayada en la Tabla 3.13 marca una correspondencia que, aunque no obtiene una afinidad semántica mayor o igual a 0,70, es interesante y representa una mejora respecto de los resultados obtenidos con H-Match. También se puede observar que todos los sistemas considerados fallan

Ontología DNE Contextualizada	Ontología Privada	Afinidad semántica			
		H-Match	AgreementMaker	RiMOM	Falcon-AO
<b>Término</b>					
Product	Product	1,0000	1,0000	1,0000	0,9987
ProductMultiDimension	Product	0,7000			
ProductMultiDimension	SemifinishedProduct			0,7666	
TrademarkDimension	Product	0,7000			
TypeDimension	Product	0,7000			
TypeDimension	ExpirationType			0,7676	
CapacityDimension	Product	0,7000			
Packing	ExternalMaterial	0,8920			
Packing	PackingMaterial	0,8920	0,7088	0,7903	0,7281
Packing	RawMaterial	0,8920			
PackingDimension					
<b>Propiedad</b>					
partNumber					
itemDescription	description	0,8000	0,7179	0,9396	0,7516
<b>Relación</b>					
hasProduct					
hasProductMultiDim					
definedByTrademarkDim	suppliedBy			0,7864	
definedByCapacityDim	capacity		0,6336		
definedByTypeDim					
usedAs					
hasPackingDim	packing		0,7055		0,7561
A		9	4	6	4
E		6	6	6	6
$C = A \cap E$		3	2	2	2
IC (%)		200,00	100,00	200,00	100,00

Tabla 3.13: Comparación de *alignments* obtenidos por distintos sistemas de *matching* para la entidad producto entre las ontologías DNE Contextualizada y Privada

en establecer las correspondencias entre la representación de la entidad producto en las ontologías Contextualizada y Privada: el término **Product** de la ontología Contextualizada tiene una afinidad semántica de 1, o muy próxima a él, con el término **Product** de la ontología Privada, en lugar de ser relacionado a los términos **ExternalMaterial** y **PackingMaterial**. Mientras, sólo el sistema H-Match relaciona correctamente el término **Packing** de la ontología Contextualizada con los términos **ExternalMaterial** y **PackingMaterial** de la ontología Privada. Como contrapartida, parecería que los nuevos sistemas sí pueden discernir entre los términos **PackingMaterial** y **RawMaterial**, ya que no encuentran una correspondencia entre los términos **Packing** y **RawMaterial** como sí lo hace el sistema H-Match.

En resumen, del análisis de los sistemas de *matching* considerados se pueden obtener algunas conclusiones:

1. Los sistemas considerados son incapaces de descubrir correspondencias

sobre entidades o características no representadas en las ontologías, como se observó en el *matching* entre las ontologías DNE Base y Privada.

2. Todos los sistemas tienen problemas para descubrir correspondencias del tipo  $\langle \text{Product}, \text{ExternalMaterial} \rangle$  y  $\langle \text{Product}, \text{PackingMaterial} \rangle$ . Con el objetivo de probar si el modelado de los usos previstos de la entidad producto tiene influencia sobre este problema, se reemplazó la relación *usedAs* entre los términos *Product* y *Packing* de la ontología DNE Contextualizada por una relación de equivalencia entre estos términos, que tiene mayor fuerza semántica que una relación particular. Sin embargo, todos los sistemas de *matching* considerados, incluido H-Match, arrojaron prácticamente los mismos resultados obtenidos con la relación *usedAs*.
3. Todos los sistemas considerados encuentran correspondencias entre términos, entre propiedades, entre relaciones o entre propiedades y relaciones, pero no entre términos y propiedades por ejemplo, como es el caso de la correspondencia esperada  $\langle \text{CapacityDimension}, \text{capacity} \rangle$ . Este tipo de correspondencias también debería ser tratado por los sistemas de *matching*.

### 3.4. Conclusiones

En este capítulo se mostró la aplicación del método propuesto en el capítulo anterior a un caso de estudio, el cual plantea una colaboración entre dos industrias pertenecientes a distintos contextos (el contexto de la industria láctea y el contexto de la industria de envases), poniendo especial énfasis en el modo en que se debe realizar el análisis en cada etapa de los procesos. También se puso en evidencia la necesidad de contar con una herramienta que de soporte al modelado de las características simples y complejas de las entidades y realice las validaciones correspondientes.

Siguiendo los lineamientos del método propuesto se logró una ontología más rica en su aspecto semántico, completa y concisa. La aplicación de esta ontología a un proceso de *matching* mostró que la hipótesis planteada al comienzo de esta tesis es correcta: el enriquecimiento de las ontologías, haciendo explícitas las características contextuales de las entidades, mejora los resultados de los procesos

---

de *matching*, independientemente de la herramienta utilizada para su ejecución. Se mostró que las ontologías contextualizadas tienen mayor potencial para la obtención de *alignments* correctos, aunque los algoritmos de *matching* deban aún mejorar su performance para disminuir la probabilidad de recuperación de correspondencias incorrectas.



## Framework para dar Soporte al Reuso de Ontologías

En este capítulo se presenta un *framework*, cuyo objetivo es dar soporte en la tarea de descripción de contextos complejos a partir del reuso de las ontologías de los contextos que los componen. La propuesta se basa en la idea de representar las características de contextos simples haciendo uso de ontologías existentes. Luego, las características de los contextos complejos se representan referenciando elementos de ontologías de contextos simples y complejos. El capítulo comienza con una breve introducción (Sección 4.1) donde, además, se especifican algunas definiciones relevantes de términos que se utilizan en el capítulo. A continuación se presenta el *framework* propuesto (Sección 4.2) y se muestra cómo se lo puede utilizar en el proceso de Enriquecimiento del método propuesto en el Capítulo 2, aplicándolos juntos a un caso de estudio (Sección 4.3). Luego se realiza una breve discusión sobre posibles herramientas para la implementación del *framework* (Sección 4.4) y, finalmente, se exponen las conclusiones (Sección 4.5).

### 4.1. Introducción

La noción de contexto se puede utilizar para representar particiones del mundo real, teniendo en cuenta algunas dimensiones básicas como la temporal, espacial, funcional, estructural, de procedimiento y de comportamiento, entre otras (Acker y Porter, 1994; Theodorakis, 2001). Cada una de estas dimensiones constituye un contexto diferente que se puede representar por una ontología que capture las características propias de dicha dimensión. En el marco de esta tesis,

a estos contextos se los denomina simples.

Generalmente, los contextos requieren más de una dimensión para su representación. En esta tesis se denomina complejo a un contexto que surge de la combinación de dos o más contextos simples o complejos. Por ejemplo, un contexto puede ser espacial y temporal a la vez, o bien un contexto temporal puede estar inserto dentro de otro contexto temporal. La representación de un contexto complejo se realiza por medio de una ontología que combina las ontologías de los contextos que lo componen. Por ejemplo, una ontología que represente las características del contexto del proceso colaborativo del Caso de Estudio 1 se puede crear tomando elementos de otras ontologías, como las que representan características del contexto temporal, espacial, de proceso, etc.

Esta propuesta de representación de contextos simples y complejos a través del uso de ontologías se contrapone a otras propuestas en el área de modelado semántico de datos. Por ejemplo, Bouquet y otros (2004b) proponen el uso del lenguaje C-OWL, una extensión de la semántica de OWL, para representar ontologías contextuales. Para los autores, la ontología representa el contexto en el que la interpretación de un elemento es cierta. Una ontología está contextualizada cuando su contenido se mantiene local y se vincula con el contenido de otras ontologías vía mapeos explícitos usando reglas puente. Estas reglas puente representan las relaciones equivalente a, más general que, menos general que, compatible e incompatible. Rifaieh y Benharkat (2006) siguen esta idea, pero combinan lógica descriptiva con lógica modal para formalizar las ontologías contextuales. Estas propuestas tienen por objeto dar soporte a aplicaciones asociadas con distintas representaciones de las mismas entidades. Sin embargo, ellas no proporcionan un mecanismo para caracterizar un contexto a través de un conjunto de características ni una forma de relacionar las características de un contexto con las entidades consideradas en dicho contexto.

También en el área de modelado semántico de datos, pero con el objetivo de abordar el problema de la heterogeneidad semántica en los sistemas de información globales, Kashyap y Sheth (1997) proponen que cada objeto exportado por un sistema al sistema global posea un contexto de definición formado por coordenadas contextuales. Estas coordenadas se obtienen de términos en ontologías específicas de dominio y se vinculan a atributos de objetos en bases de datos. La interoperabilidad se logra con respecto a una consulta, cuyo

contexto también se expresa como un conjunto de coordenadas contextuales. La respuesta se obtiene de comparar los contextos de la consulta y de los objetos del sistema global. Otras propuestas a partir de este enfoque son las de Wache y Stuckenschmidt (2001) y Firat y otros (2005). Estas ideas son adecuadas para integrar diferentes bases de datos que describen su contenido usando descriptores derivados de ontologías. Sin embargo, cuando se debe tratar con contextos complejos, la descripción de un contexto como un conjunto de coordenadas contextuales puede no ser suficiente. Un contexto como el del proceso colaborativo del Caso de Estudio 1 requiere de un conjunto de términos, un conjunto de relaciones, un conjunto de axiomas y la relación con otros contextos para su representación, lo cual no se puede lograr sólo con coordenadas contextuales.

De lo expuesto en el párrafo anterior se deduce que la actividad de describir un contexto puede tener un alto grado de complejidad. Con el fin de dar soporte en esta tarea, en esta tesis se propone un *framework* que ayude en la creación de las ontologías correspondientes a contextos complejos. Para ello, este *framework* se compone de diversas ontologías, que representan características de distintos contextos, y permite combinarlas con el objetivo de describir características de contextos complejos.

Por otra parte, como el proceso de Enriquecimiento de ontologías, correspondiente al método presentado en el Capítulo 2, incentiva la reutilización de ontologías existentes para hacer explícitas entidades y características contextuales, el *framework* propuesto también es de utilidad en dicho proceso. Una entidad, o sus características, pueden tener diferentes interpretaciones y/o representaciones en función del contexto en el que se las considera. Por ejemplo en el Caso de Estudio 1, una entidad se interpreta como producto en el contexto de la industria de envases y la misma entidad se interpreta como envase en el contexto de la industria láctea. Para mostrar esta situación, la ontología que representa dicha entidad y sus características se puede enriquecer referenciando a las correspondientes ontologías de los contextos (complejos) de la industria de envases y de la industria láctea, para hacer explícita la correcta interpretación de la entidad en los diferentes contextos. De esta forma, la ontología enriquecida permite el intercambio de información sin imponer un significado global. Para apoyar el proceso de Enriquecimiento, el *framework* incluye también a las ontologías enriquecidas y las ontologías base correspondientes.

En suma, este *framework* potencia el reuso de ontologías debido a que:

- La creación de las ontologías que describen características de contextos complejos se realiza reutilizando ontologías que describen características de contextos simples y complejos.
- Las ontologías que representan características de contextos se pueden utilizar para enriquecer la representación de todas las entidades que correspondan ser interpretadas en dichos contextos.

Para la presentación del *framework*, en la próxima sección se utiliza el Caso de Estudio 1.

## 4.2. Framework para el Desarrollo de Ontologías

Como se introdujo en la sección anterior, el *framework* tiene por objeto dar soporte en el desarrollo de ontologías que describen características de contextos complejos, pero además se puede utilizar para dar soporte en el proceso de Enriquecimiento de ontologías, específicamente en la tarea de hacer explícitas entidades y características contextuales de las entidades. Para ello, el *framework* reutiliza ontologías existentes para describir características de distintos contextos, aprovechando porciones de ontologías ampliamente aceptadas por la comunidad ontológica o que se respaldan en estándares.

La ontología enriquecida, obtenida del Proceso 5 del método propuesto, se puede pensar constituida por elementos de tres orígenes diferentes: elementos provenientes de la ontología base, de la cual se parte; elementos de la ontología que describe las características de los contextos involucrados en el intercambio de información; y elementos propios de la ontología enriquecida, que se deben agregar para relacionar a los dos primeros grupos de elementos.

Dependiendo del método empleado para desarrollar la ontología base, ésta puede consistir de elementos que sólo refieren a la estructura de fuentes de datos (por ejemplo si se aplicó una técnica de aprendizaje de ontologías), o puede tener, además, elementos que refieren a algunas características de contextos. La diferencia entre la ontología base y la enriquecida radica en que la ontología base

carece de elementos que refieran a los contextos involucrados en el intercambio de información, y por esta razón surge la necesidad de enriquecerla.

Por su parte, la ontología que describe las características de los contextos involucrados en el intercambio de información generalmente se asocia a un contexto complejo, y puede referenciar elementos de las ontologías que representan características de los contextos simples o complejos que lo conforman.

La Figura 4.1 muestra las relaciones entre los cuatro tipos de ontologías que se combinan en el *framework* general propuesto. Dichas ontologías son: Ontología Base (OB), Ontologías de Características Primitivas (OCP), Ontología de Especificación de Contexto (OEC) y Ontología Enriquecida (OE).

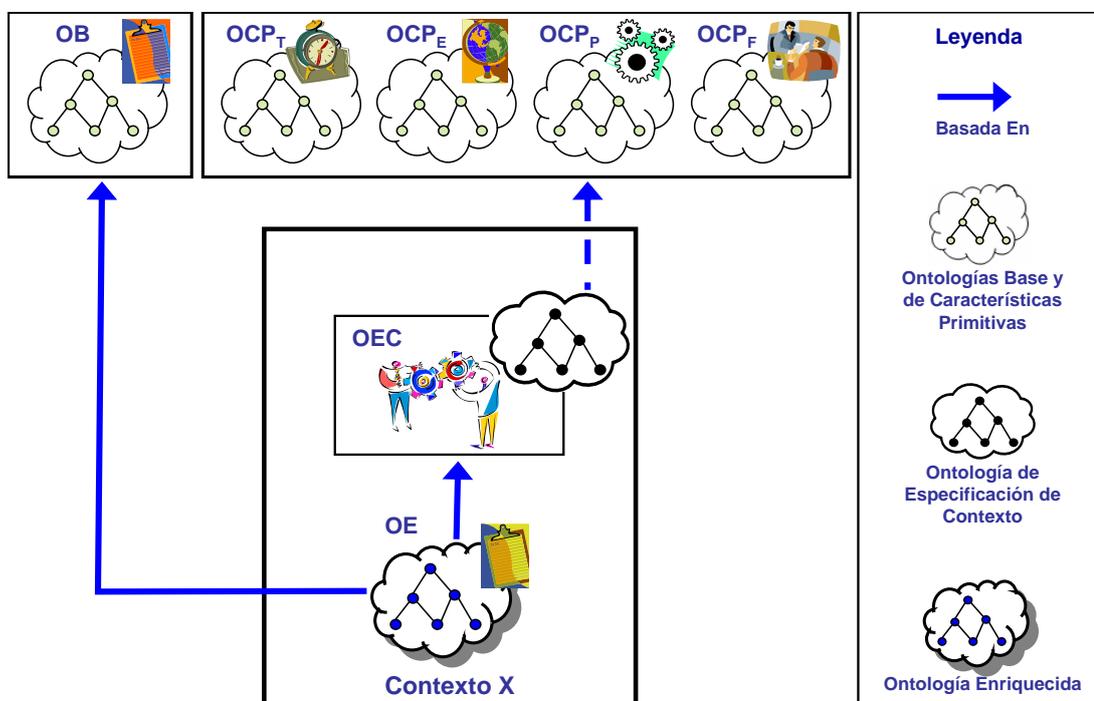


Figura 4.1: Sinopsis del *framework* propuesto

Para una mayor claridad en su presentación, en la Figura 4.1 se considera un contexto complejo (Contexto X), en función del cual se enriquece una ontología (OE). Esta Ontología Enriquecida contiene elementos que provienen de la Ontología Base (OB) y elementos de la ontología que describe las características del contexto complejo X, denominada Ontología de Especificación de Contexto (OEC). A su vez, esta Ontología de Especificación de Contexto puede tener elementos que provienen de ontologías que describen características de contextos simples (temporal, espacial, de proceso, funcional, etc.), denominadas Ontologías

de Características Primitivas (OCP). También puede contener elementos que se originan en otras ontologías de contextos complejos, pero por simplicidad no se representan en el *framework*. Estas ontologías se describen a continuación.

#### 4.2.1. Ontología base

En procesos de intercambio de información, generalmente dicha información está contenida en documentos de texto, bases de datos o documentos XML, entre otras fuentes. Esto hace que las técnicas de aprendizaje de ontologías sean particularmente adecuadas para crear una Ontología Base a partir de la cual comenzar todo el proceso de desarrollo. Estas técnicas facilitan el desarrollo de la ontología cuando la fuente de información tiene varios cientos de elementos, lo cual es común en la práctica. Sin embargo, existen otras metodologías y métodos que se pueden seguir para obtener esta ontología, como los mencionados en el Capítulo 2.

Cuando la Ontología Base (esquina superior izquierda de la Figura 4.1, OB) se crea a partir de la aplicación de una técnica de aprendizaje de ontologías a las fuentes de información, generalmente consiste de términos genéricos que refieren a las unidades representacionales de la estructura de las fuentes de información, y no contiene instancias. De esta forma, la Ontología Base puede ser reutilizada en diferentes contextos y con diferentes objetivos. Por ejemplo en el Caso de Estudio 1, la ontología resultado del Proceso 2: Generación de una ontología base (Sección 3.1.2, página 71) es una ontología que describe semánticamente el documento XML Schema de un DNE. Si los socios de negocio acuerdan utilizar la misma estructura de DNE en los distintos subprocesos que componen el proceso colaborativo, esta ontología será reutilizada en diferentes contextos (PAP, PMP y POP) y con diferentes objetivos (consenso sobre un plan de provisión de material, consenso sobre un programa de provisión de material y consenso sobre la definición de un cronograma de órdenes de provisión).

En el caso particular de los DNEs, las empresas podrían decidir emplear DNEs estándares. Por ejemplo, la Figura 4.2 muestra una porción de la ontología obtenida al aplicar la técnica de aprendizaje de ontologías propuesta por Volz y otros (2003) al *XML Schema PlanningSchedule* (XMLS-PS) del estándar OAGIS (Rowell y Feblowitz, 2002). También existen algunos estándares que

tienen una ontología asociada que se puede utilizar como Ontología Base. Por ejemplo, han aparecido algunas iniciativas para desarrollar ontologías acerca de las especificaciones de RosettaNet (Haller y otros, 2008; Kotinurmi y Vitvar, 2006).

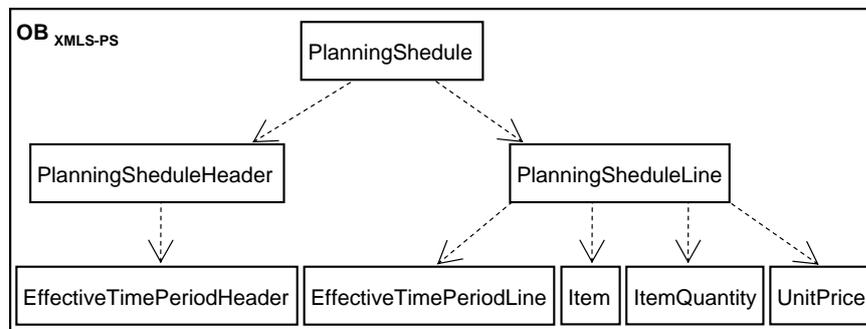


Figura 4.2: Porción de la Ontología Base (OB) para el *XML Schema Planning Schedule* (XMLS-PS) del estándar OAGIS

Cuando la Ontología Base se genera con otra metodología, diferente de las técnicas de aprendizaje, o bien se toma como Ontología Base a una que se enriqueció previamente con un propósito distinto del de intercambio de información, esta Ontología Base probablemente contenga otros elementos, además de los estructurales, que refieran a características propias de los contextos involucrados con el objetivo para el cual se la creó.

#### 4.2.2. Ontologías de características primitivas

Las Ontologías de Características Primitivas (OCP) sólo tienen elementos para describir la semántica de las características de contextos simples. Estos contextos se pueden clasificar en diferentes tipos de acuerdo a la perspectiva o dimensión que ellos describen. Por ejemplo, Acker y Porter (1994) han identificado las siguientes dimensiones básicas: de comportamiento, estructural, funcional, temporal y procedural, entre otras. Por su parte, en su estudio del uso de contexto, Theodorakis (2001) ha identificado las siguientes dimensiones o perspectivas: temporal, espacial, funcional y puntos de vista, entre otras. Cada una de estas dimensiones o perspectivas puede tener una Ontología de Características Primitivas asociada. En la Figura 4.1, área central superior (página 115), se representan algunas de estas ontologías a modo de ejemplo (temporal,  $OCP_T$ ; espacial,  $OCP_E$ ; de proceso,  $OCP_P$ ; funcional,  $OCP_F$ ).

#### 4.2.2.1. Ontología de características primitivas temporal

Una Ontología de Características Primitivas Temporal (OCP<sub>T</sub>) describe las características de un contexto temporal, es decir, aquel en el cual la dimensión temporal lleva a considerar la información de acuerdo al momento de tiempo al cual la información refiere (Theodorakis, 2001). Por ejemplo, la demanda de un producto tenía un comportamiento determinado antes de la crisis económica mundial en 2009 y otro comportamiento después de ese momento. Ambos períodos, antes y después de la crisis, constituyen diferentes contextos.

Muchas personas y proyectos han invertido sus esfuerzos en el problema de representar conocimiento temporal. Entre las propuestas de representación desarrolladas se destacan las basadas en espacio de estados, en esquemas de fechas, las de propagación de restricciones o encadenamiento antes/después, las basadas en duración y los modelos formales (Allen, 1983, 1991). Cada una de estas propuestas tiene sus virtudes y debilidades y, por ende, son más o menos apropiadas según la situación en la que van a ser aplicadas.

Allen (1983), entre sus importantes aportes a la solución de la problemática, describe las características que debiera tener una representación temporal y establece que:

- La representación debe permitir imprecisión significativa (por ejemplo, A antes que B).
- La representación debe permitir incertidumbre de información (no se conoce la relación exacta entre dos tiempos, pero sí alguna restricción sobre cómo podrían estar relacionados; por ejemplo, A antes o superpuesto con B).
- La representación debe permitir variar la granularidad del razonamiento (nanosegundos, días, años, etc.).
- El modelo debe soportar la persistencia.
- Se deben poder representar eventos disjuntos (por ejemplo, el evento A ocurre antes o después que el evento B).

Como parte del proyecto *DARPA Agent Markup Language* (DAML) (DAML.org, 2006), que finalizó a comienzos del 2006 y cuyo

objetivo era desarrollar un lenguaje y herramientas para facilitar el concepto de la Web Semántica, el esfuerzo *DAML-Time* tenía como objetivo desarrollar una ontología representativa de tiempo, que exprese conceptos temporales y propiedades comunes a cualquier formalización de tiempo (Ferguson, 2003). Varios sitios, contratistas DAML y otros, han desarrollado ontologías de tiempo (por ejemplo, DAML-S, Cycorp, CMU, Kestrel y Teknowledge, entre otros).

Particularmente, Hobbs y Pan (2004) han desarrollado una subontología básica de tiempo, expresada en OWL, que es mucho más simple que la ontología completa *DAML-Time*. Esta subontología, denominada *OWL-Time*, presenta las características descritas por Allen (1983) y proporciona la mayoría de los conceptos básicos temporales y las relaciones que la mayoría de las aplicaciones sencillas necesitan, es decir, un vocabulario para expresar hechos acerca de las relaciones topológicas entre instantes, intervalos y eventos, junto con información sobre duraciones y sobre fechas y horas. Esta ontología está lo suficientemente madura como para ser utilizada como una Ontología de Características Primitivas Temporal.

#### 4.2.2.2. Ontología de características primitivas espacial

Una Ontología de Características Primitivas Espacial ( $OCP_E$ ) describe las características de un contexto espacial. Un contexto espacial es aquel que permite agrupar la información por lugar o región de origen (Theodorakis, 2001). Por ejemplo, las regiones, en las cuales están situadas las empresas que mantienen una relación colaborativa, tienen su propia economía y marco legal. Cada región constituye un contexto diferente.

Tomando esta definición como base, se podría pensar que la división política del planeta sería suficiente para clasificar la información por procedencia. Sin embargo, existen otros atributos que pueden ayudar a la caracterización de un contexto espacial, como por ejemplo el idioma que se utiliza para la comunicación oral y escrita, la moneda de uso corriente, etc.

Por otro lado, son ampliamente conocidas las bondades de los sistemas de información geográfica (*Geographic Information System*, GIS). Entre las distintas funcionalidades provistas por un GIS se incluye la capacidad de realizar operaciones espaciales (Goodchild y Kemp, 1990). Los GIS almacenan datos de diverso tipo: datos sobre seres humanos; actividades humanas; datos

demográficos: edad, sexo, etnia y estado civil, educación; datos sobre vivienda: calidad, costo; migración; transporte; datos económicos: ingresos personales, empleo, ocupaciones, industria, crecimiento regional. Además, involucran datos naturales sobre el medio ambiente y los recursos naturales: geología, suelo, vegetación, hidrología; así como también datos que generalmente no se perciben como “naturales”: la red de transporte y las fronteras políticas, entre otros.

A pesar de existir una serie de iniciativas que vienen trabajando en el desarrollo de una ontología GIS (Bitters, 2005; Islam y otros, 2004; Linková y Nedbal, 2007), considerar todas las características de un GIS con el fin de describir un contexto espacial no sólo es ambicioso sino también innecesario en muchos casos.

Algo más modesto y manejable sería tener distintas porciones de ontologías que modelen características de un contexto espacial, como por ejemplo los países, las monedas, las unidades de medida, etc. Para mencionar algunas, las ontologías de: códigos de países ISO 3166; códigos de monedas que se publican en la norma ISO 4217:2008; y el estándar internacional ISO 80000, sucesor de ISO 31, para representar magnitudes físicas y unidades de medida, se podrían usar a tal fin.

#### **4.2.2.3. Ontología de características primitivas de proceso**

Una Ontología de Características Primitivas de Proceso (OCP<sub>P</sub>) describe las características de un contexto de proceso. En el estándar ISO 9001 (ISO, 2000) se define proceso como un conjunto de actividades correlacionadas, o que interactúan, que transforman entradas en salidas. En general, un proceso se caracteriza por: (a) un conjunto de actividades/transformaciones que se realizan como parte del proceso; (b) un conjunto de recursos que se consumen, se producen, se refinan, se transforman o se usan en el proceso; (c) un objetivo o meta explícita; y (d) un conjunto de restricciones o reglas que gobiernan o controlan el proceso. Por ejemplo, considerando las relaciones colaborativas del Caso de Estudio 1, existe la necesidad de definir un tipo de contexto de proceso que permita agrupar las actividades, recursos y restricciones de los procesos. El Modelo de Colaboración Socio-a-Socio define tres niveles de colaboración: Consenso a nivel de Planificación Agregada de la Producción (PAP), Consenso a nivel de Programa Maestro de la Producción (PMP) y Consenso a nivel del Programa de Órdenes de Provisión (POP). La información contenida en los DNEs puede tener diferentes

interpretaciones de acuerdo al nivel (el contexto) en el cual se la usa.

La Web Semántica y las tecnologías de servicios de la Web Semántica proporcionan técnicas de conocimiento adecuadas para representar procesos (Hepp y otros, 2005). En este sentido, las iniciativas más relevantes son OWL-S (Martin y otros, 2004) y WSMO (Feier y otros, 2005).

OWL-S es una ontología basada en OWL, desarrollada en el marco del programa DAML. Esta ontología suministra, a los proveedores de servicios web, un conjunto básico de constructores de lenguaje de marcado para describir las propiedades y capacidades de sus servicios web en forma inequívoca e interpretable por la computadora, y permite a las personas y agentes de software descubrir, invocar, componer y supervisar de forma automática los recursos web que ofrecen servicios, bajo restricciones especificadas (Martin y otros, 2004).

WSMO proporciona especificaciones ontológicas de los elementos básicos de servicios de la Web Semántica, tratando de resolver las deficiencias de OWL-S (Lara y otros, 2005). WSMO está acompañado de un lenguaje formal, *Web Service Modeling Language* (WSML), que permite escribir anotaciones de servicios web de acuerdo con el modelo conceptual. Además, existe un entorno de ejecución (*Web Service Modelling eXecution environment*, WSMX<sup>1</sup>) para el descubrimiento dinámico, selección, mediación, invocación y la interoperación de los servicios de la Web Semántica basados en la especificación WSMO. Teniendo en cuenta las características de WSMO, esta ontología es una candidata para la descripción de las características de un contexto de proceso.

#### 4.2.2.4. Ontología de características primitivas funcional

Una Ontología de Características Primitivas Funcional (OCP<sub>F</sub>) describe las características de un contexto funcional. Un contexto funcional es el que se crea con el propósito de representar los diferentes roles que las personas o los conceptos pueden jugar (Theodorakis, 2001). Por ejemplo, en una relación colaborativa como la del Caso de Estudio 1, las empresas participantes asumen diferentes roles (cliente y proveedor). Cada rol constituye un contexto diferente.

Los conceptos de rol y modelo de rol han atraído a investigadores de las áreas de Ingeniería de Software Orientado a Objetos y Orientado a Agente. Dos metodologías reconocidas en estas áreas son Gaia (Wooldridge y otros, 2000) y

---

<sup>1</sup> <http://www.wsmx.org/>

MaSE (DeLoach y otros, 2001). Gaia es una metodología para análisis y diseño orientado a agente y MaSE es una metodología de propósito general para analizar, diseñar y desarrollar sistemas multiagente heterogéneos.

En Gaia, un rol se define por cuatro atributos: responsabilidades, permisos, actividades y protocolos. Las responsabilidades determinan la funcionalidad de un rol y son análogas a los objetivos definidos en MaSE. Los permisos son los “derechos” (generalmente recursos de información) asociados a un rol e identifican los recursos que están disponibles para que un rol pueda cumplir sus responsabilidades. MaSE no tiene el concepto de permiso; en general se asume que cada recurso se encapsula en un único rol, que proporciona una interfaz para el resto del sistema. Las actividades de Gaia definen cálculos de un rol, que se realizan sin la interacción con otros roles. Estas actividades se corresponden directamente con las actividades de tareas concurrentes en MaSE. Los protocolos definen cómo los roles Gaia interactúan entre sí y se definen por seis atributos: propósito, iniciador, respondedor, entradas, salidas y procesamiento. En MaSE, los protocolos se definen con más detalle al modelarlos como tareas concurrentes (DeLoach y otros, 2001).

En el marco del proyecto europeo *K4Care*, se desarrolló la ontología Gaia<sup>2</sup>, implementada en el lenguaje OWL, para describir semánticamente los conceptos de la metodología Gaia (Hajnal y otros, 2007). Esta ontología se podría utilizar, en principio, para describir roles.

Otras Ontologías de Características Primitivas se pueden utilizar para describir las características de otros tipos de contextos. Las Ontologías de Características Primitivas pueden ser estándares, es decir, ontologías acordadas por toda la comunidad. Estas ontologías pueden tener instancias.

### 4.2.3. Ontología de especificación de contexto

Una Ontología de Especificación de Contexto (OEC) (área central de la Figura 4.1, página 115) refiere a las características particulares que describen un contexto complejo. Esta ontología se puede construir tomando términos de

---

<sup>2</sup> <http://code.google.com/p/petriagents/downloads/detail?name=GAIA.owl>, (último acceso 17/02/2011)

las Ontologías de Características Primitivas (descriptas en la sección anterior) y agregando otros términos necesarios para describir características típicas del contexto dado, o se puede construir ad hoc. La Ontología de Especificación de Contexto incluye instancias.

La Figura 4.3 muestra un ejemplo de una ontología de este tipo para el contexto del Proceso Colaborativo a nivel de Programa Maestro de la Producción,  $OEC_{PC-PMP}$ , del Caso de Estudio 1, cuyo objetivo es representar las características principales del contexto del proceso colaborativo a ese nivel. Los términos en esta ontología pueden describir características como la moneda en la que se expresan los precios, la unidad de medida en la que se indican los volúmenes, el horizonte de tiempo para la colaboración y los productos involucrados en la colaboración, entre otras. Para una mejor visualización, la Figura 4.3 sólo muestra los términos que derivan de la Ontología de Características Primitivas Temporal, otras características se muestran en figuras subsiguientes; los rectángulos con bordes punteados representan instancias.

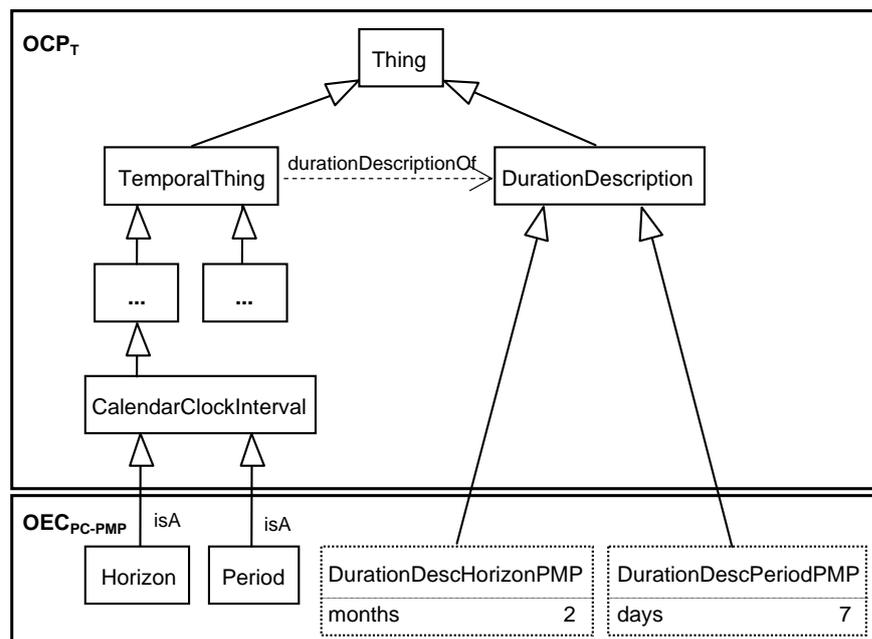


Figura 4.3: Ejemplo de una Ontología de Especificación de Contexto (OEC) para el Proceso Colaborativo Programa Maestro de la Producción (PC-PMP), características temporales

En la Figura 4.3 se puede observar que la característica horizonte de tiempo (representada por el término **Horizon**) y la característica período (representada por el término **Period**) derivan ambas del término **CalendarClockInterval** de la

Ontología de Características Primitivas Temporal ( $OCP_T$ ). En el contexto del proceso colaborativo, los socios de negocio han acordado que cada programa maestro tendrá una duración de dos meses y que los períodos, dentro de ese programa, serán de siete días. Por esta razón, en la Ontología de Especificación de Contexto se crean dos instancias del término `DurationDescription` de la Ontología de Características Primitivas Temporal, `DurationDescHorizonPMP` y `DurationDescPeriodPMP`, y se definen dos restricciones de valor: una para indicar que todas las instancias del término `Horizon` van a estar relacionadas con la instancia `DurationDescHorizonPMP` a través de la relación `durationDescriptionOf`, y otra para indicar que todas las instancias del término `Period` van a estar relacionadas con la instancia `DurationDescPeriodPMP` a través de la relación `durationDescriptionOf`. En lógica descriptiva esto es:

$$\begin{aligned} \text{Horizon} &\sqsubseteq (\exists \text{ ocpt} : \text{durationDescriptionOf}.\{\text{DurationDescHorizonPMP}\}) \\ \text{Period} &\sqsubseteq (\exists \text{ ocpt} : \text{durationDescriptionOf}.\{\text{DurationDescPeriodPMP}\}) \end{aligned}$$

donde el prefijo `ocpt:` indica que la relación `durationDescriptionOf` pertenece a la Ontología de Características Primitivas Temporal, la cual se importó en la Ontología de Especificación de Contexto.

La Ontología de Características Primitivas Temporal utilizada es la ontología *OWL-Time* mencionada en la Sección 4.2.2.1.

Por su parte, las características moneda (en la que se expresan los precios), unidad de cantidad (en la que se expresan las cantidades) y unidad de volumen (en la que se expresan las capacidades) se representan por los términos `PriceDimension`, `QuantityDimension` y `CapacityDimension` respectivamente en la Ontología de Especificación de Contexto (Figura 4.4). Estas características tienen asociada una unidad de medida (`UnitOfMeasure`) que pertenece a la Ontología de Características Primitivas Espacial ( $OCP_E$ ), también importada en la Ontología de Especificación de Contexto. En el contexto del proceso colaborativo, el dinero se expresa en dólares estadounidenses, las cantidades en unidades y los volúmenes en  $\text{cm}^3$ .

La Ontología de Características Primitivas Espacial utilizada se basa en la propuesta de modelado de cantidades físicas realizada por Gruber (1995).

En el contexto del proceso colaborativo, un producto refiere tanto al producto fabricado por la industria de envases como al envase de algún

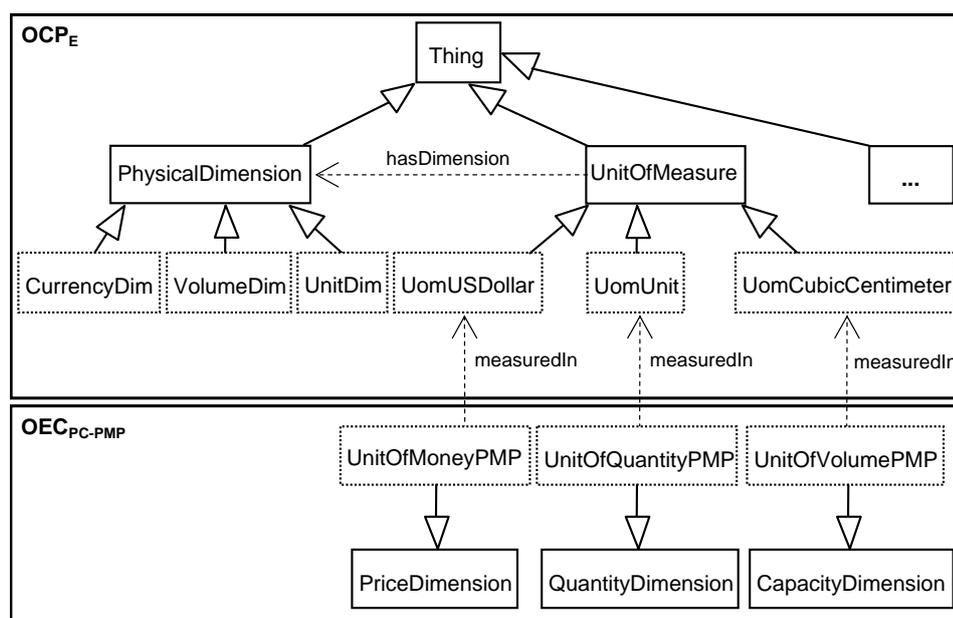


Figura 4.4: Ejemplo de una Ontología de Especificación de Contexto (OEC) para el Proceso Colaborativo Programa Maestro de la Producción (PC-PMP), características espaciales

producto de la industria láctea. Tal como se describió en el Caso de Estudio 1, Sección 3.1.5.3 (página 81), producto es una característica contextual compleja de la entidad ítem, caracterizado a su vez por la marca impresa en los envases, el tipo de material con el que se hacen los envases y la capacidad de los mismos. De acuerdo al análisis realizado en la sección antes mencionada, la representación de producto se modela en la Ontología de Especificación de Contexto como se muestra en la Figura 4.5.

En dicha figura se puede observar que la Ontología de Especificación de Contexto posee términos para describir características propias del contexto del proceso colaborativo, que no derivan ni están relacionados a términos de las Ontologías de Características Primitivas; tal es el caso de los términos **Packing**, **PackingDimension**, **Product**, **ProductMultiDimension**, **TrademarkDimension** y **TypeDimension** que refieren a los productos comercializados entre los socios de negocio, tal como se los representa en este contexto.

Además, en la Ontología de Especificación de Contexto se define una restricción de valor para indicar que todas las instancias del término **ProductMultiDimension** van a estar relacionadas con la instancia **UnitOfVolumePMP** del término **CapacityDimension** a través de la relación

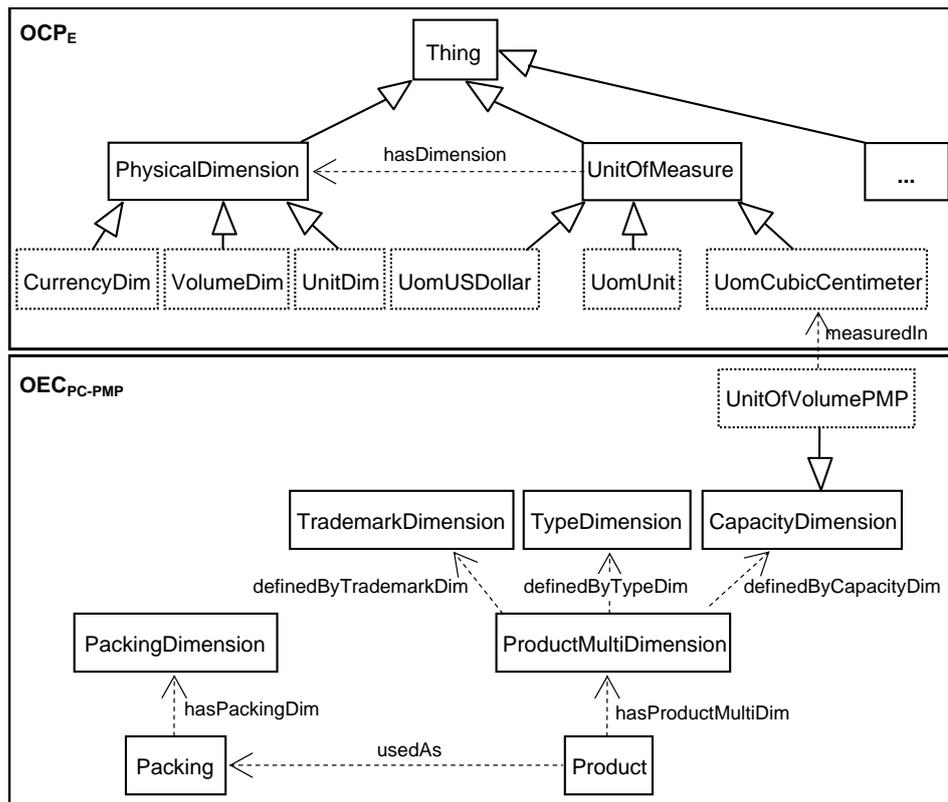


Figura 4.5: Ejemplo de una Ontología de Especificación de Contexto (OEC) para el Proceso Colaborativo Programa Maestro de la Producción (PC-PMP), característica producto

definedByCapacityDim, estableciendo de esta forma que la capacidad de los productos se mide en  $\text{cm}^3$ .

En suma, las Figuras 4.3 (página 123), 4.4 (página 125) y 4.5 muestran cómo se representan las características particulares del contexto del Proceso Colaborativo a nivel de Programa Maestro de la Producción en la Ontología de Especificación de Contexto  $\text{OEC}_{\text{PC-PMP}}$ . Esta ontología puede ser reutilizada cada vez que sea necesario interpretar alguna entidad o característica en el marco de este contexto.

#### 4.2.4. Ontología enriquecida

La interpretación semántica de la información intercambiada entre dos partes, sean estas socios de negocio, agentes de software, etc., depende del contexto en el cual se considera dicha información. Mientras la estructura sintáctica de una fuente de información se puede describir a través de

una Ontología Base creada con una técnica de aprendizaje de ontologías (Sección 4.2.1), la semántica de esa información requiere de una ontología más elaborada que la describa. En esta tesis se denomina Ontología Enriquecida (OE) a dicha ontología.

Esta Ontología Enriquecida permite relacionar términos de la Ontología Base con términos de la Ontología de Especificación de Contexto. La Figura 4.6 muestra un ejemplo de una Ontología Enriquecida del *XML Schema PlanningSchedule* para el Proceso Colaborativo Programa Maestro de la Producción. En la parte inferior de la figura se pueden observar todos los términos de la Ontología Base, que describe la estructura del XML Schema, y cómo estos términos se relacionan con términos de la Ontología de Especificación de Contexto, que describe las características del contexto del proceso colaborativo. Las flechas gruesas en la figura indican que tanto la Ontología Base  $OB_{\text{XMLS-PS}}$  como la Ontología de Especificación de Contexto  $OEC_{\text{PC-PMP}}$  se importan en la Ontología Enriquecida, es decir se reutilizan, y así no se definen cada vez que se necesitan estas ontologías.

Debido a las relaciones definidas entre los términos de la Ontología Base y los términos de la Ontología de Especificación de Contexto, y debido a los valores de las instancias de esta última ontología, el uso del *XML Schema PlanningSchedule* en el contexto del proceso colaborativo programa maestro de la producción implica que:

- (a) las instancias del término `EffectiveTimePeriodHeader` se interpreten como un horizonte de tiempo de dos meses de duración (todas las instancias del término `Horizon` se relacionan con la instancia `DurationDescHorizonPMP`, cuya propiedad `months` tiene valor 2, Sección 4.2.3);
- (b) las instancias del término `EffectiveTimePeriodLine` se interpreten como un período de siete días de duración (todas las instancias del término `Period` se relacionan con la instancia `DurationDescPeriodPMP`, cuya propiedad `days` tiene valor 7, Sección 4.2.3);
- (c) las instancias del término `ItemQuantity` se interpreten como unidades (la instancia `UnitOfQuantityPMP` del término `QuantityDimension` se relaciona con la instancia `UomUnit` del término `UnitOfMeasure`, Figura 4.6);

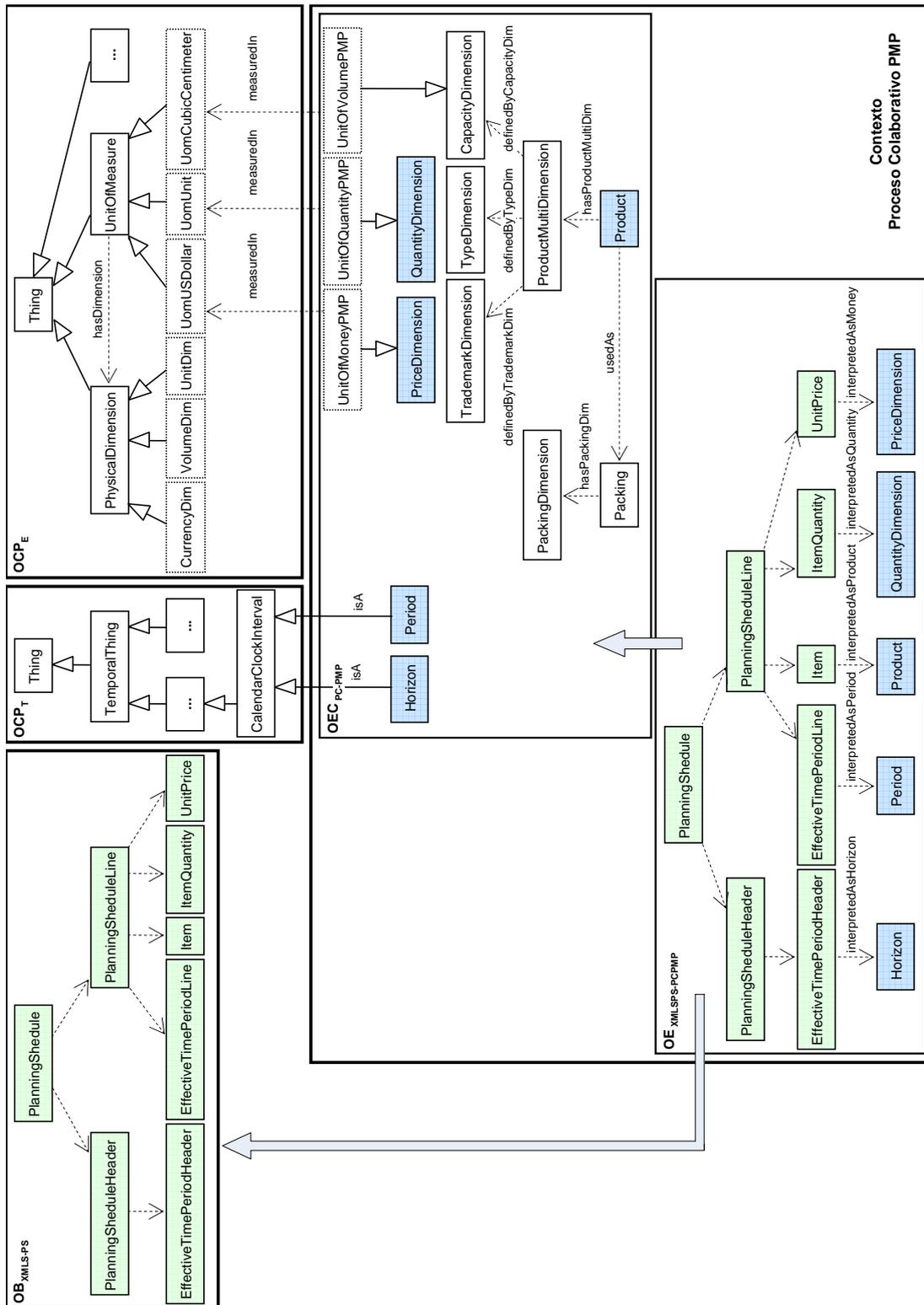


Figura 4.6: Ejemplo de una Ontología Enrichida (OE) del *XML Schema PlanningSchedule* (XMLSPS) para el Proceso Colaborativo Programa Maestro de la Producción (PCPMP)

- (d) las instancias del término `UnitPrice` se interpreten como dólares estadounidenses (la instancia `UnitOfMoneyPMP` del término `PriceDimension` se relaciona con la instancia `UomUSDollar` del término `UnitOfMeasure`, Figura 4.6); y
- (e) las instancias del término `Item` se interpreten como productos con la estructura definida en la Ontología de Especificación de Contexto correspondiente (Figura 4.6).

Definiendo la Ontología Enriquecida de esta forma, la semántica de las entidades se puede especificar refiriendo las instancias apropiadas del conjunto de elementos que representan las características del contexto. Hacer explícitas las características contextuales de las entidades no sólo clarifica la semántica de la información intercambiada, sino también posibilita la reusabilidad de la misma estructura sintáctica de la fuente de información en diferentes contextos, ya que diferentes características dependientes del contexto se pueden asociar con la misma estructura. Por ejemplo, el mismo *XML Schema PlanningSchedule* se podría utilizar para intercambiar la información necesaria para ejecutar los tres subprocesos propuestos por el Modelo de Colaboración Socio-a-Socio (Planificación Agregada de la Producción, Programa Maestro de la Producción, y Programa de Órdenes de Provisión). De esta forma, la misma Ontología Base  $OB_{XMLS-PS}$  se vincularía con otra Ontología de Especificación de Contexto, por ejemplo la de la Planificación Agregada de la Producción, donde las instancias del término `Horizon` tendrían una duración de dieciocho meses y el término `Product` tendría otra estructura, ya que en este contexto se definen a nivel de familia de productos.

### 4.3. Caso de Estudio 2: Desarrollo de una Ontología de Interfaz

El objetivo de esta sección es mostrar cómo el *framework* propuesto (Sección 4.2) se puede utilizar junto con el método descrito en el Capítulo 2 para el desarrollo de una ontología de interfaz de una empresa, que quiere hacer públicos sus productos y requerimientos de materiales con el fin de facilitar el descubrimiento de nuevos clientes y proveedores.

### 4.3.1. Identificación de los requerimientos de la ontología

Sea, por ejemplo, la industria láctea del Caso de Estudio 1. El objetivo de una ontología de interfaz para esta industria es el de representar sus productos y requerimientos de materiales de forma tal de facilitar la búsqueda de información que puedan realizar otras industrias o clientes que deseen encontrar nuevos proveedores y clientes. Esta ontología estará disponible en el sitio web de la empresa. De esta forma será directamente accesible por agentes de software con capacidad para procesarla y las personas podrán consultar su contenido navegando a través del sitio o completando un formulario, generado a partir de la ontología, que se traducirá al lenguaje de consultas SPARQL. La información contenida en la ontología debería permitir responder inicialmente la siguiente pregunta: ¿cuáles son los productos de la industria, sus características y los materiales requeridos para su fabricación?

Las entidades a representar en la ontología son: los productos que la industria ofrece a la venta, es decir, productos terminados; y los requerimientos para la elaboración de dichos productos, es decir, materias primas y envases. Para este caso de estudio sólo se consideran los envases a efectos de disminuir el tamaño del problema. Las Tablas 4.1 y 4.2 muestran las relaciones entre las entidades y las características de las entidades respectivamente.

	Producto Terminado	Material de Empaque
Producto Terminado		envasadoCon
Material de Empaque	usadoParaEnvasar	

Tabla 4.1: DERO: Relaciones entre las entidades del dominio

	Característica Contextual	Característica No Contextual
Producto Terminado		Código
		Descripción
		Presentación
Material de Empaque	Capacidad	Código
	Tamaño	Descripción
	Embalaje	

Tabla 4.2: DERO: Características de las entidades del dominio

### 4.3.2. Generación y formalización de una ontología base

La industria láctea almacena su información en bases de datos relacionales. La Figura 4.7 muestra una porción de una ontología privada de la industria láctea, obtenida a partir de dichas bases de datos, traducida al inglés por las mismas razones esgrimidas en el Caso de Estudio 1 e implementada en OWL.

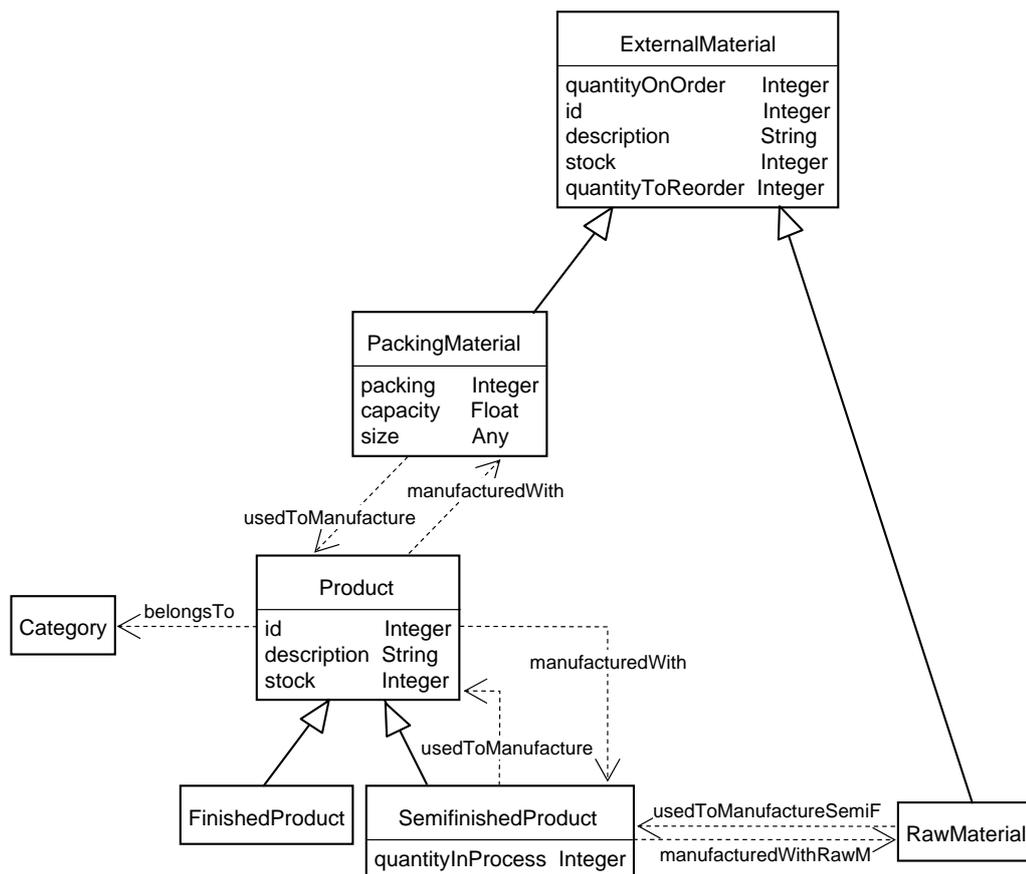


Figura 4.7: Porción de una ontología privada de la industria láctea

#### 4.3.2.1. Ontología base

La industria láctea no desea hacer pública toda su ontología privada. Por ejemplo, el término `ExternalMaterial` (Figura 4.7), que representa tanto a las materias primas como al material de empaque, tiene entre sus propiedades información del punto de repedido (`quantityToReorder`), cantidad en trámite de recepción (`quantityOnOrder`) y disponibilidad actual (`stock`), entre otras. En consecuencia, la Ontología Base (Figura 4.8) se crea a partir de esta ontología privada.

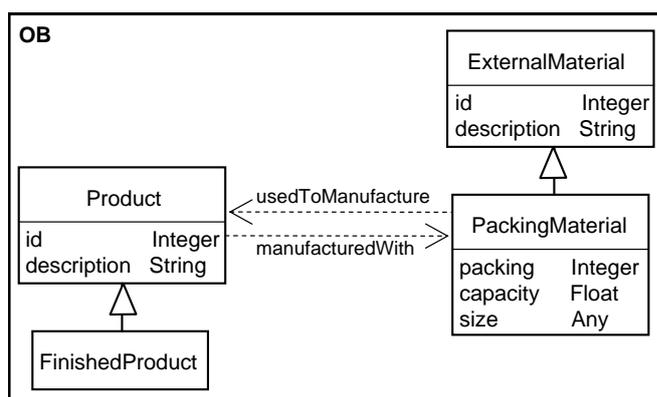


Figura 4.8: Ontología Base (OB)

### 4.3.3. Verificación de la ontología base

#### 4.3.3.1. Verificación de la representación de la entidad material de empaque

La entidad material de empaque está representada en la Ontología Base por los términos `ExternalMaterial` y `PackingMaterial`, sus propiedades y la relación `isA` entre ellos. Esta entidad se relaciona con la entidad producto terminado a través de la relación `usadoParaEnvasar` (Tabla 4.1, página 130), representada en la Ontología Base por la relación `usedToManufacture`.

De acuerdo a la Tabla 4.2 (página 130), las características a considerar de un material de empaque son: el código, la descripción, la capacidad, el tamaño y el embalaje. En la Ontología Base, las características código y descripción están representadas por las propiedades `id` y `description` del término `ExternalMaterial` respectivamente. Las características capacidad, tamaño y embalaje están representadas por las propiedades `capacity`, `size` y `packing` del término `PackingMaterial` respectivamente.

Esta forma de representación presenta algunos inconvenientes. Por ejemplo, la capacidad del material de empaque, representada por la propiedad `capacity`, es un caso de exceso en el criterio de enfoque de codificación mínima. Esta propiedad tiene asociado un tipo de dato `float` que no permite saber en qué unidad se expresa; es necesario identificar su unidad de medida. De hecho, en la industria láctea considerada, la capacidad de los envases correspondientes a productos líquidos se expresa en  $\text{cm}^3$ , mientras que la correspondiente a productos sólidos se expresa en gramos.

La característica tamaño, representada por la propiedad `size`, exhibe otro problema. Conceptualmente, la propiedad `size` representa el tamaño de un material de empaque, pero la estructura de representación asociada a esta propiedad depende de la forma de dicho material. Por ejemplo, si el material de empaque tiene forma de caja, su tamaño se describe por tres características: alto, ancho y profundidad. Por el contrario, si el material de empaque tiene forma de tubo (generalmente vaso o botella), su tamaño se describe por dos características: diámetro y alto. Además, es necesario explicitar si estas características representan medidas exteriores o interiores del material de empaque. Así, esta representación de la característica tamaño es un caso de exceso del criterio de compromisos ontológicos mínimos.

La representación de la característica embalaje, la propiedad `packing`, ofrece un problema similar. Esta característica refiere a la agrupación de envases pequeños en otros de mayor tamaño, y representa la cantidad de unidades de material de empaque pequeño que se pueden acomodar en uno de mayor tamaño. Se reconocen tres categorías de material de empaque: primario, secundario y terciario. Un material de empaque primario puede tener asociado uno secundario o uno terciario. Por ejemplo, en el caso de la industria láctea considerada, las cajas *tetra brik* (material de empaque primario) se colocan en cajas de cartón corrugado (material de empaque secundario) y éstas, a su vez, en *pallets* (material de empaque terciario) para su traslado. En cambio, las botellas plásticas se suelen embalar directamente en *pallets* especialmente diseñados para el traslado y su posterior disposición en góndolas. El tipo de material de empaque asociado al embalaje de uno dado no está explícito en la representación de esta característica.

#### 4.3.3.2. Verificación de la representación de la entidad producto terminado

Por su parte, la entidad producto terminado está representada en la Ontología Base por los siguientes elementos: los términos `Product` y `FinishedProduct`, sus propiedades y la relación `isA` entre ellos. La relación `envasadoCon` entre un producto terminado y un material de empaque (Tabla 4.1, página 130) está representada por la relación `manufacturedWith` en la ontología.

De acuerdo a la Tabla 4.2 (página 130), las características a considerar de un producto terminado son: el código, la descripción y la presentación

del producto. Las características código y descripción están representadas por las propiedades `id` y `description` del término `Product` respectivamente. La característica presentación está representada por la entidad material de empaque y la relación `manufacturedWith` entre los términos `Product` y `PackingMaterial`. Esta representación de características de un producto no presentaría inconvenientes a la hora de su interpretación en diferentes contextos.

Otro aspecto a considerar son los posibles usos de las entidades material de empaque y producto terminado en diferentes contextos. En principio se podría pensar que el material de empaque, si bien es producto para los sistemas de información de la industria que lo fabrica, otros sistemas en cualquier otro contexto lo interpretarán como material utilizado para envasar o embalar cosas. Por el contrario, producto terminado puede significar cosas diferentes según el contexto en el cual se lo considere, por ejemplo automóvil, medicamento, mueble, lácteo, entre otros. En este caso entonces, es necesario especificar mejor su significado para evitar malas interpretaciones.

La Tabla 4.3 resume los resultados de la verificación.

	Representación			Criterio no cumplido
	Término principal	Relación	Propiedad	
<b>Entidad</b>				
Producto Terminado	FinishedProduct			Uso previsto
Material de Empaque	PackingMaterial			
<b>Relación</b>				
envasadoCon usadoParaEnvasar		manufacturedWith usedToManufacture		
<b>Característica no contextual</b>				
Código			id	
Descripción			description	
Presentación	PackingMaterial	manufacturedWith		
Código			id	
Descripción			description	
<b>Característica contextual</b>				
Capacidad			capacity	Exceso de codificación mínima
				Dim. p/ especific. característica
Tamaño			size	Exceso compr. ontol. mínimo
				Dim. p/ especific. característica
Embalaje			packing	Exceso compr. ontol. mínimo
				Dim. p/ especific. Característica

Tabla 4.3: Resultado de la verificación

#### 4.3.4. Enriquecimiento de la ontología base

De la sección anterior surge que las características que deben ser mejoradas en su representación son la capacidad, el tamaño y el embalaje de los materiales de empaque y el significado de la entidad producto terminado. Con este objetivo se desarrolla en primer lugar una Ontología de Especificación de Contexto para describir las características del contexto de la industria láctea considerada. Luego se desarrolla una Ontología Enriquecida para la correcta interpretación de la información en el contexto de dicha industria. Con este objetivo, la Ontología Enriquecida reutiliza la Ontología de Especificación de Contexto antes mencionada.

##### 4.3.4.1. Ontología de especificación de contexto

###### *Enriquecimiento de la representación de la entidad material de empaque*

Para el enriquecimiento de la representación de la entidad material de empaque se reutiliza la Ontología de Características Primitivas Espacial, y se agregan otros elementos necesarios siguiendo los principios de diseño propuestos en esta tesis.

La Figura 4.9 muestra la representación asociada a la capacidad de un material de empaque en la Ontología de Especificación de Contexto. Esta característica simple tiene al término `CapacityDimension` como núcleo central de su representación, del cual derivan los términos `SolidsCapacityDimension` y `LiquidCapacityDimension`, disjuntos entre sí. La unidad de medida asociada a la capacidad de un material de empaque está representada por el término `UnitOfMeasure` de la Ontología de Características Primitivas Espacial, importada en la Ontología de Especificación de Contexto para la industria láctea, y la relación `measuredIn` entre los términos `UnitOfMeasure` y `CapacityDimension`.

Además, en la Ontología de Especificación de Contexto se definen dos restricciones de valor: una para indicar que todas las instancias del término `LiquidCapacityDimension` van a estar relacionadas con la instancia `UomCubicCentimeter` del término `UnitOfMeasure` a través de la relación `measuredIn`, y otra para indicar que todas las instancias del término

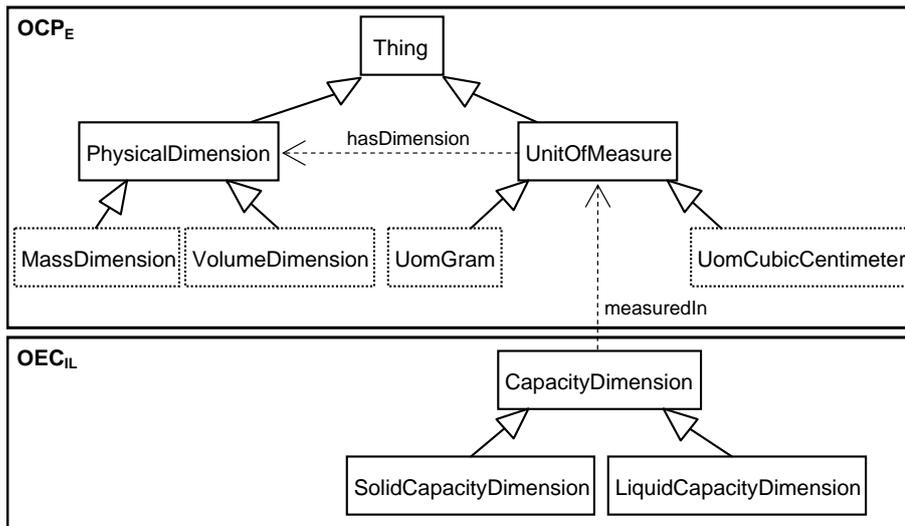


Figura 4.9: Ontología de Especificación de Contexto (OEC) para la industria láctea (IL), característica capacidad de un material de empaque

`SolidsCapacityDimension` van a estar relacionadas con la instancia `UomGram` del término `UnitOfMeasure` a través de la relación `measuredIn`. En lógica descriptiva esto es:

$$\begin{aligned} \text{LiquidCapacityDimension} &\subseteq (\exists \text{ measuredIn} . \{ \text{ocpe} : \text{UomCubicCentimeter} \}) \\ \text{SolidsCapacityDimension} &\subseteq (\exists \text{ measuredIn} . \{ \text{ocpe} : \text{UomGram} \}) \end{aligned}$$

donde el prefijo `ocpe:` indica que las instancias correspondientes pertenecen a la Ontología de Características Primitivas Espacial.

Para la representación de la característica compleja tamaño se adopta una solución similar (Figura 4.10). El término central es `SizeMultiDimension`, del cual derivan los términos `TubeSizeMultiDimension` y `BoxSizeMultiDimension`, disjuntos entre sí. La multidimensión `TubeSizeMultiDimension` está formada por dos dimensiones integrales, representadas por los términos `DiameterDimension` y `HighDimension`. La multidimensión `BoxSizeMultiDimension` está formada por tres dimensiones integrales, representadas por los términos `HighDimension`, `WidthDimension` y `LengthDimension`. Todas estas dimensiones integrales son disjuntas entre sí.

Además, para garantizar que cada instancia de una multidimensión está compuesta por una instancia de cada dimensión integral, y que la multidimensión se compone solamente de esas dimensiones integrales, se deben agregar las siguientes restricciones en lógica descriptiva:

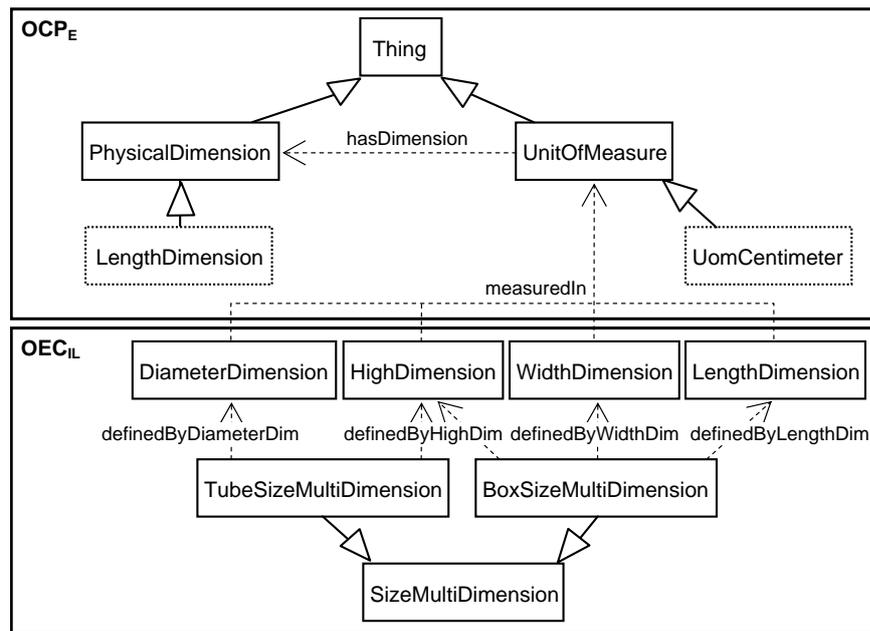


Figura 4.10: Ontología de Especificación de Contexto (OEC) para la industria láctea (IL), característica tamaño de un material de empaque

- Para el término **TubeSizeMultiDimension**:

$$\begin{aligned} & \forall \text{ definedByDiameterDim.DiameterDimension} \\ & \exists \text{ definedByDiameterDim.DiameterDimension} \\ & \quad = 1 \text{ definedByDiameterDim} \\ & \forall \text{ definedByHighDim.HighDimension} \\ & \exists \text{ definedByHighDim.HighDimension} \\ & \quad = 1 \text{ definedByHighDim} \end{aligned}$$

- Para el término **BoxSizeMultiDimension**:

$$\begin{aligned} & \forall \text{ definedByHighDim.HighDimension} \\ & \exists \text{ definedByHighDim.HighDimension} \\ & \quad = 1 \text{ definedByHighDim} \\ & \forall \text{ definedByWidthDim.WidthDimension} \\ & \exists \text{ definedByWidthDim.WidthDimension} \\ & \quad = 1 \text{ definedByWidthDim} \\ & \forall \text{ definedByLengthDim.LengthDimension} \\ & \exists \text{ definedByLengthDim.LengthDimension} \\ & \quad = 1 \text{ definedByLengthDim} \end{aligned}$$

También se define, para cada una de las dimensiones integrales, una restricción de valor para indicar que todas las instancias del término que representa la dimensión van a estar relacionadas con la instancia `UomCentimeter` del término `UnitOfMeasure` a través de la relación `measuredIn`. En lógica descriptiva esto es:

$$\text{DiameterDimension} \subseteq (\exists \text{ measuredIn.}\{\text{ocpe : UomCentimeter}\})$$

$$\text{HighDimension} \subseteq (\exists \text{ measuredIn.}\{\text{ocpe : UomCentimeter}\})$$

$$\text{WidthDimension} \subseteq (\exists \text{ measuredIn.}\{\text{ocpe : UomCentimeter}\})$$

$$\text{LengthDimension} \subseteq (\exists \text{ measuredIn.}\{\text{ocpe : UomCentimeter}\})$$

La representación de la característica simple embalaje requiere la definición de una dimensión contenedor (el término `ContainerDimension` en la Figura 4.11) y tres dimensiones derivadas de ésta (`PrimaryContainerDimension`, `SecondaryContainerDimension` y `TertiaryContainerDimension`), disjuntas entre sí.

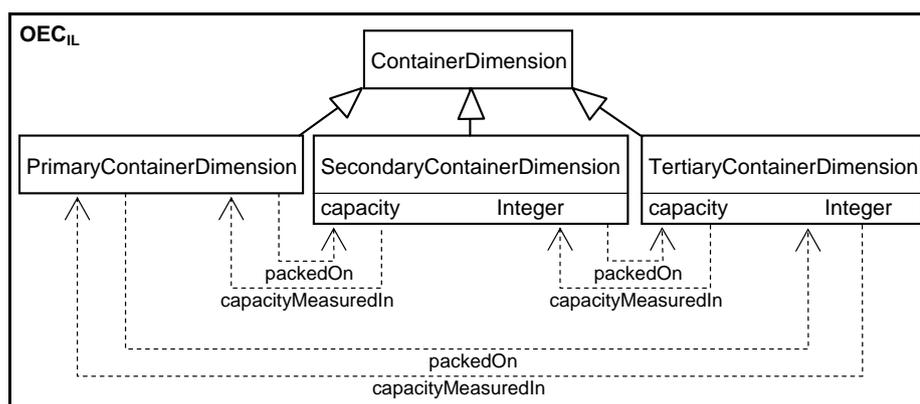


Figura 4.11: Ontología de Especificación de Contexto (OEC) para la industria láctea (IL), característica embalaje de un material de empaque

Un contenedor primario se puede embalar en uno secundario o uno terciario. Un contenedor secundario sólo se puede embalar en uno terciario. Para asegurar estas condiciones, la relación `packedOn` tiene los siguientes axiomas sobre el término `SecondaryContainerDimension` de su dominio, como condiciones necesarias y suficientes:

$$\forall \text{ packedOn.TertiaryContainerDimension}$$

$$\exists \text{ packedOn.TertiaryContainerDimension}$$

La capacidad de un contenedor terciario se mide en término de las unidades de contenedores primarios o secundarios que pueden acomodarse en él. La

capacidad de un contenedor secundario se mide en término de las unidades de contenedores primarios que pueden acomodarse en él. Para garantizar el cumplimiento de estas condiciones, la relación `capacityMeasuredIn` tiene los siguientes axiomas definidos sobre el término `SecondaryContainerDimension` de su dominio:

$$\begin{aligned} &\forall \text{ capacityMeasuredIn.PrimaryContainerDimension} \\ &\exists \text{ capacityMeasuredIn.PrimaryContainerDimension} \end{aligned}$$

Sobre la misma relación `capacityMeasuredIn`, pero para el término `TertiaryContainerDimension`, se agrega una restricción de cardinalidad para asegurar la definición de sólo una relación, ya sea con un contenedor primario o uno secundario.

#### *Enriquecimiento de la representación de la entidad producto terminado*

La entidad producto terminado requiere una desambiguación de su significado. En el contexto de la industria láctea, producto terminado refiere a producto lácteo (leche, queso, manteca, etc.). Una posible solución para la desambiguación es definir el término `DairyProduct` en lugar del término `FinishedProduct` en la Ontología Base, pero de esta forma se pierde el vínculo directo con la ontología privada que dio origen a la Ontología Base. Por el contrario, si se define producto lácteo como sinónimo de producto terminado se evitaría que alguien o algo, por ejemplo un agente de software que busca información, mal interprete el significado de producto terminado, al tiempo que se mantiene el vínculo con la ontología privada de la industria láctea. En la Ontología de Especificación de Contexto se define entonces el término `DairyProduct`, que luego se relaciona con el término `FinishedProduct` en la Ontología Enriquecida.

#### **4.3.4.2. Ontología enriquecida**

La Ontología Enriquecida importa las Ontologías Base (Figura 4.8, página 132) y de Especificación de Contexto para la industria láctea considerada (Figuras 4.9, página 136; 4.10, página 137; y 4.11, página 138). Este hecho se visualiza en la Figura 4.12: los elementos que tienen prefijo `ob:` corresponden a la Ontología Base y los que tienen prefijo `oec:` corresponden a la Ontología de Especificación de Contexto.

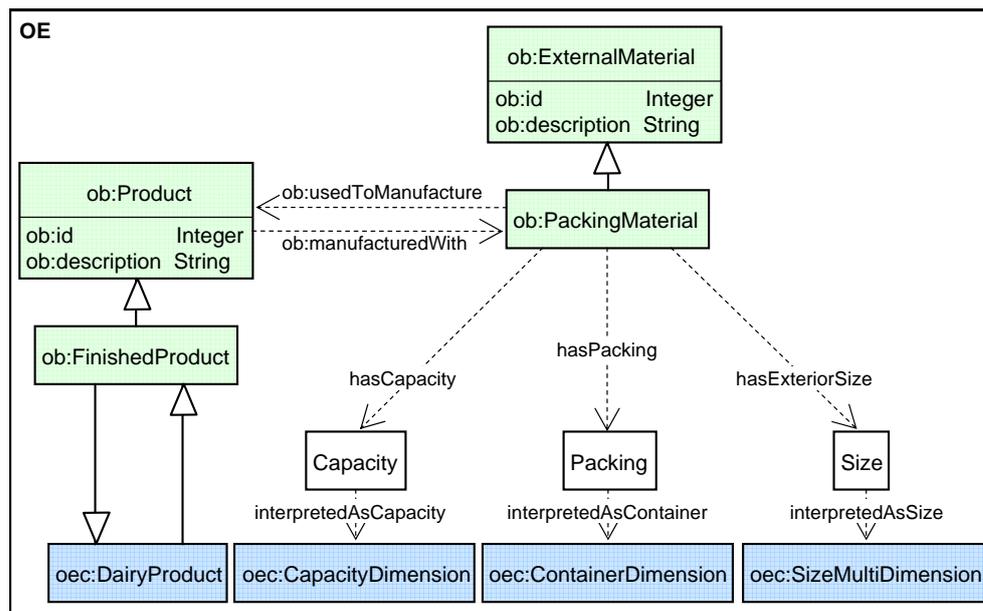


Figura 4.12: Una porción de la Ontología Enriquecida

Siguiendo los principios de diseño propuestos en esta tesis, para mejorar la representación de la característica capacidad de un material de empaque y poder asociarle una unidad de medida, es necesario representarla por medio de un término en lugar de una propiedad. Las características tamaño y embalaje de un material de empaque también deben ser representadas por términos. Luego, en la Ontología Enriquecida se deben eliminar las propiedades `capacity`, `size` y `packing` (importadas con la Ontología Base) y se deben agregar los términos homónimos correspondientes, asociándolos al término `PackingMaterial` a través de relaciones (`hasCapacity`, `hasExteriorSize` y `hasPacking`, Figura 4.12).

Del análisis de la Ontología Enriquecida se puede concluir que, en el contexto de la industria láctea considerada:

- La capacidad de un material de empaque se define en  $\text{cm}^3$  o gramos, según sea el tipo de la instancia relacionada (`LiquidCapacityDimension` o `SolidsCapacityDimension`) a la instancia correspondiente del término `Capacity` a través de la relación `interpretedAsCapacity`.
- El tamaño de un material de empaque se expresa en función de su largo, ancho y alto, o su diámetro y alto, según sea el tipo de la instancia asociada (`BoxSizeMultiDimension` o `TubeSizeMultiDimension`) a la instancia correspondiente del término `Size` a través de la relación `interpretedAsSize`.

- El embalaje de un material de empaque es de tipo secundario o terciario según sea el tipo de la instancia relacionada (`SecondaryContainerDimension` o `TertiaryContainerDimension`) a la instancia correspondiente del término `Packing` a través de la relación `interpretedAsContainer`. Para asegurar esta condición se deben agregar los axiomas que garanticen que la relación `interpretedAsContainer` sólo tiene en el rango una instancia de alguno de los dos términos antes mencionados.
- Las entidades producto terminado y producto lácteo son equivalentes.

#### 4.3.5. Verificación y validación de la ontología enriquecida

La verificación de la Ontología Enriquecida arroja las siguientes conclusiones:

- La ontología es completa para el propósito para el cual se la desarrolló. Todas las entidades, sus características y relaciones requeridas están representadas en la ontología.
- La ontología es concisa. Todos los elementos de la Ontología Base identificados como innecesarios para el propósito de la ontología se eliminaron de la misma y por lo tanto no forman parte de la Ontología Enriquecida.
- La ontología es más clara. La representación de las entidades, sus características y relaciones se enriqueció para hacer explícitas características que podrían ser mal interpretadas en otros contextos diferentes al de la industria láctea considerada. También se respetan las convenciones de nombres establecidas.
- La ontología cumple con los requisitos de enfoque de codificación y compromisos ontológicos mínimos.
- La ontología no arrojó resultados negativos al verificarse su coherencia lógica con un razonador.

Al no requerir la Ontología Enriquecida un nuevo ciclo de mejora, se procede a su validación. Con este objetivo se formalizan las preguntas de competencia en el lenguaje de consultas SPARQL. A continuación se muestran algunas de ellas.

**Consulta 1:**

```

SELECT  ?elem1 ?type ?elem2
WHERE   { ?elem1 ?type ?elem2 .
         FILTER regex(str(?elem1), "product", "i")
       }
ORDER BY ?elem1 ?type

```

Esta consulta codifica la pregunta de competencia “¿qué elementos de la ontología se utilizan para modelar productos?” El resultado obtenido fue el siguiente (Figura 4.13):

Results		
elem1	type	elem2
ob:FinishedProduct	rdf:type	owl:Class
ob:FinishedProduct	rdfs:subClassOf	ob:Product
ob:FinishedProduct	owl:equivalentClass	oec:DairyProduct
ob:Product	rdf:type	owl:Class
oec:DairyProduct	rdf:type	owl:Class
oec:DairyProduct	owl:equivalentClass	ob:FinishedProduct

Figura 4.13: Elementos de la Ontología Enriquecida que modelan productos

**Consulta 2:**

```

SELECT  ?property ?product
WHERE   { ?property rdfs:domain ?product .
         ?property ?z owl:DatatypeProperty .
         FILTER regex(str(?product), "product", "i")
       }
ORDER BY ?property ?product

```

Esta consulta implementa la pregunta “¿cuáles son las características de producto modeladas como propiedades?” El resultado se muestra en la Figura 4.14.

Results	
property	product
ob:description	ob:Product
ob:id	ob:Product

Figura 4.14: Características de productos modeladas como propiedades en la Ontología Enriquecida

### Consulta 3:

```

SELECT   ?entity ?relation ?product

WHERE    { ?relation rdfs:domain ?product .
           ?relation ?z          owl:ObjectProperty .
           ?relation rdfs:range  ?entity .
           FILTER regex(str(?product), "product", "i")
         }

ORDER BY ?entity

```

Esta consulta muestra la implementación de la pregunta de competencia “¿con qué entidades se relaciona producto?” La Figura 4.15 muestra el resultado de esta consulta.

Results		
entity	relation	product
ob:Category	ob:belongsTo	ob:Product
ob:PackingMaterial	ob:manufacturedWith ↔ ob:usedToManufacture	ob:Product

Figura 4.15: Entidades relacionadas con productos en la Ontología Enriquecida

## 4.4. Herramientas para la Implementación del Framework Propuesto

Desarrollar ontologías es una actividad compleja y que demanda tiempo, y más aún si los desarrolladores tienen que implementarlas directamente en un lenguaje de ontologías, sin ningún tipo de herramienta de apoyo (Gómez-Pérez y otros, 2004). Para facilitar esta tarea se crearon herramientas de desarrollo de ontologías. Estas herramientas proporcionan interfaces de usuarios que ayudan a

llevar a cabo algunas de las principales actividades del proceso de desarrollo de ontologías, como la conceptualización, la ejecución y la comprobación de coherencia. Además de las funciones comunes de edición y navegación, estas herramientas dan soporte a la documentación de la ontología, la exportación e importación de la ontología hacia y desde diferentes formatos y lenguajes de ontologías, la edición gráfica de la ontología, la gestión de bibliotecas de ontologías, etc. Ejemplos de tales herramientas son WebODE (Arpírez Vega y otros, 2003), SWOOP (Kalyanpur y otros, 2006), KAON (Maedche y otros, 2003), OntoEdit (Sure y otros, 2002) y Protégé (Gennari y otros, 2003; Knublauch, 2003; Noy y otros, 2000), entre otras.

Algunas herramientas dan soporte a la aplicación de una metodología particular de desarrollo de ontologías. Por ejemplo, el proceso de desarrollo de ontologías en OntoEdit se basa en la metodología On-To-Knowledge (Sure y otros, 2004), el de WebODE en la metodología METHONTOLOGY (Gómez-Pérez y otros, 2004). Otras herramientas, por ejemplo Protégé, no soportan ninguna metodología en especial.

Para el desarrollo y reutilización de los distintos tipos de ontologías que forman parte del *framework* propuesto se puede utilizar cualquier herramienta que provea soporte, como mínimo, para:

- Editar ontologías (crear/editar términos, relaciones, propiedades, axiomas formales e instancias).
- Cargar, editar y grabar ontologías en formato OWL.
- Importar ontologías en formato OWL.

Es deseable que la herramienta permita, además:

- Editar ontologías en forma gráfica.
- Cargar, editar y grabar ontologías en otros formatos, por ejemplo RDF Schema, archivos XML, archivos XML Schema, bases de datos relacionales.
- Importar ontologías en otros formatos como los mencionados en el punto anterior.

- Realizar inferencias de clasificación (clasificar automáticamente términos en una taxonomía), verificar la coherencia (si un término puede tener instancias) y efectuar algún tipo de prueba que ayude a detectar errores comunes cometidos en la edición de una ontología.
- El desarrollo colaborativo y la gestión de versiones de ontologías.
- Dar soporte en el modelado de las características contextuales de las entidades.

Para la implementación del *framework* propuesto en este capítulo se utilizó el *plug-in* Protégé-OWL, una extensión de la plataforma de desarrollo Protégé que permite editar archivos OWL y bases de datos, y satisface todos los requerimientos antes detallados y casi todos los deseables.

Sin embargo, en estas herramientas el modelado de las características contextuales de una entidad está sujeto a la capacidad del ingeniero ontológico de no omitir ninguno de los elementos requeridos para una correcta representación de las mismas.

## 4.5. Conclusiones

En este capítulo se presentó un *framework* para dar soporte al reuso de ontologías. Diferentes ontologías del *framework* definen separadamente el significado de las características de distintos contextos, lo cual facilita la reutilización de estas ontologías al componerlas para la descripción de contextos complejos. También este *framework* permite el reuso de ontologías preexistentes. De esta forma, distintas ontologías basadas en estándares se pueden agregar al *framework* presentado.

Además, empleando la idea de contexto, la propuesta da soporte a la representación de algunas características de las entidades que generalmente están implícitas en una ontología, lo cual permite que el *framework* se utilice como soporte al proceso de Enriquecimiento de ontologías del método presentado en el Capítulo 2.

Este *framework* define una nueva estrategia para combinar contextos y ontologías con el propósito de definir la semántica de los datos, evitando la

definición de un significado global, lo cual suele ser una solución difícil o incluso imposible en entornos abiertos. La propuesta utiliza ontologías no sólo para definir el significado de la información sino también para caracterizar el contexto en el cual este significado es cierto. Debido a que las características del contexto afectan el significado de la información, se propone establecer relaciones entre estas ontologías. La principal ventaja de este *framework* es que potencia el reuso de ontologías permitiendo organizar las ontologías existentes. La principal desventaja de esta propuesta es la complejidad de las tareas de mantenimiento de las ontologías. La propuesta permite relacionar un conjunto de ontologías desarrolladas por diferentes personas. El hecho que estos desarrolladores puedan pertenecer a diferentes contextos hace las tareas de mantenimiento más difíciles.

La necesidad de desarrollar un conjunto de ontologías para describir las características de un contexto complejo podría verse como una desventaja. Sin embargo, varias ontologías desarrolladas en distintas disciplinas se pueden utilizar para modelar las Ontologías de Especificación de Contexto. En este capítulo se han identificado potenciales ontologías con este objetivo. Particularmente en el contexto temporal, *OWL-Time* es una ontología madura y ampliamente adoptada para representar entidades temporales. En el contexto espacial, y considerando que el propósito final es definir una semántica de contexto fácil de usar, considerar todas las características de un GIS para un contexto espacial no sólo es ambicioso sino innecesario. No obstante, existen varios estándares que se pueden utilizar para representar características de un contexto espacial (por ejemplo la norma ISO 3166 para representar los códigos de países, la norma ISO 4217:2008 para representar los códigos de monedas, etc.). Las ontologías desarrolladas para representar servicios web se pueden utilizar para modelar características de un contexto de proceso. Finalmente parece necesario algún esfuerzo para desarrollar una ontología capaz de describir contextos funcionales, aunque se podría analizar el empleo de las ontologías desarrolladas para modelar los roles de agentes de software.

El *framework* propuesto se aplicó a dos casos de estudio diferentes. Primero se utilizó el Caso de Estudio 1, descrito en el Capítulo 3, para facilitar la comprensión del *framework*. Luego se mostró cómo el *framework* propuesto se puede utilizar para dar soporte en el proceso de Enriquecimiento del método propuesto en el Capítulo 2, aplicándolos al desarrollo de una ontología de interfaz.

## Lenguaje de Representación de Características Contextuales

En este capítulo se describe un lenguaje de representación de características para dar soporte a la representación de características contextuales cuando se desarrolla una ontología. Inicialmente se muestra la necesidad de un lenguaje que refleje las leyes generales que describen la realidad (Sección 5.1). Con este objetivo, el lenguaje propuesto toma como referencia una ontología fundacional que comprende un sistema de categorías del mundo real (Sección 5.2). Se describe la sintaxis abstracta del lenguaje y la herramienta MetaXys que implementa parte del mismo (Sección 5.3). Luego se muestra cómo se puede enriquecer la representación de entidades en una ontología existente mediante la herramienta (Sección 5.4) y se finaliza con las conclusiones (Sección 5.5).

### 5.1. Introducción

Para poder explicitar las características contextuales de las entidades en una ontología se necesita contar con: (a) un método que de soporte al descubrimiento de dichas características (el cual se presentó en el Capítulo 2); y (b) un lenguaje de modelado que permita representarlas de forma concisa, completa y sin ambigüedades. Este último es el objetivo del presente capítulo.

En la actualidad se cuenta con varias herramientas que se pueden utilizar para el desarrollo de ontologías, algunas de las cuales ya se han mencionado en esta tesis –WebODE, Protégé, etc.–. En el Capítulo 3 se utilizó el editor Protégé-OWL para enriquecer la representación de las entidades en una ontología y formalizar

dicha ontología en OWL. Como se vio en dicho capítulo, la inserción de todos los elementos necesarios para la adecuada representación de las características simples y complejas de las entidades está sujeta a la capacidad del ingeniero ontológico de no omitir ninguno de los elementos requeridos. Los editores actuales de ontologías no pueden dar soporte a esta tarea, es decir, comprobar que todos los elementos necesarios sean agregados e impedir la eliminación o modificación de elementos que dejen inconsistente la representación de las entidades. Con el fin de subsanar estas limitaciones de los editores, en este capítulo se presenta un lenguaje para modelar las características de las entidades y un prototipo de una herramienta para dar soporte a este modelado y formalizar automáticamente la ontología en OWL.

Una entidad es todo aquello que existe, incluyendo objetos, procesos, cualidades y estados, en los siguientes tres niveles (Smith y otros, 2006):

- Nivel 1: entidades en la realidad como objetos, procesos, cualidades, estados, etc.
- Nivel 2: representaciones cognitivas de esta realidad por parte de algún sujeto cognitivo. Una representación cognitiva es una representación cuyas unidades son ideas, pensamientos o creencias en la mente de una persona.
- Nivel 3: concretizaciones de estas representaciones cognitivas en artefactos de representación. Un artefacto de representación sirve para hacer públicas las representaciones cognitivas a otros agentes, tanto personas como computadoras. Los elementos constitutivos de los artefactos de representación refieren a entidades en la realidad (Nivel 1).

La Figura 5.1 muestra la relación entre los niveles antes mencionados y sus elementos constitutivos.

Una representación cognitiva sólo existe en la mente de una persona o comunidad. Con el objetivo de documentarla, comunicarla y analizarla, las personas transforman gradualmente dicha representación cognitiva en artefactos de representación de diversos tipos. En este proceso ignoran alguna información que no es de interés (abstraen) y agrupan información importante en base a características comunes (clasifican). Para crear un artefacto de representación se necesita de un lenguaje. Las líneas punteadas entre elementos del Nivel 3 y

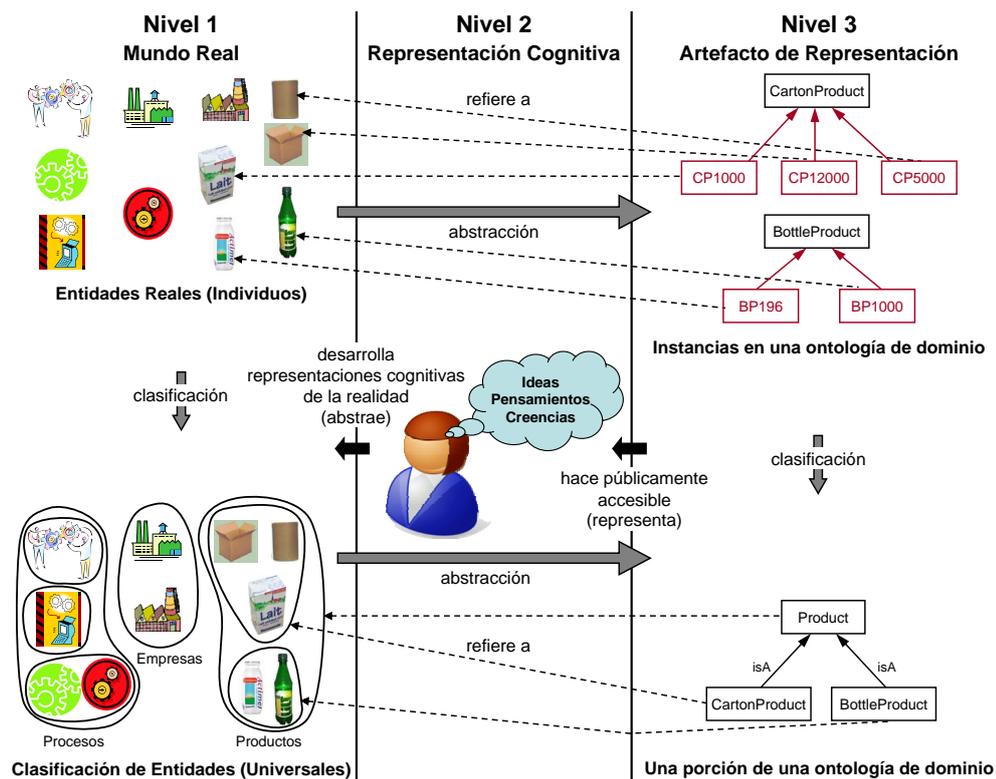


Figura 5.1: Entidades de la realidad y sus representaciones

elementos del Nivel 1 de la Figura 5.1 indican que las relaciones entre elementos de un artefacto de representación y entidades de la realidad no son directas, sino a través de una representación cognitiva (Smith y otros, 2006).

En el área de modelado, el artefacto de representación se denomina modelo; y el lenguaje para crearlo, lenguaje de modelado. Un lenguaje de modelado debe proporcionar un conjunto de primitivas de modelado que se puedan utilizar para expresar directamente abstracciones relevantes del dominio (Guizzardi, 2007). La Figura 5.2 representa las relaciones entre una abstracción y su representación, y entre estos y la representación cognitiva de un dominio y el lenguaje de modelado gráfico (LMG).

Un lenguaje de modelado comprende una sintaxis abstracta y una sintaxis concreta a usar en la creación, edición y mantenimiento de los modelos expresados en ese lenguaje (Mellor y otros, 2004).

En lenguajes gráficos, el conjunto de primitivas de modelado gráfico disponibles constituye la sintaxis concreta y la sintaxis abstracta del lenguaje se define en términos de un metamodelo (Guizzardi, 2007). Este metamodelo se enriquece con condiciones dadas en algún lenguaje de descripción de restricciones

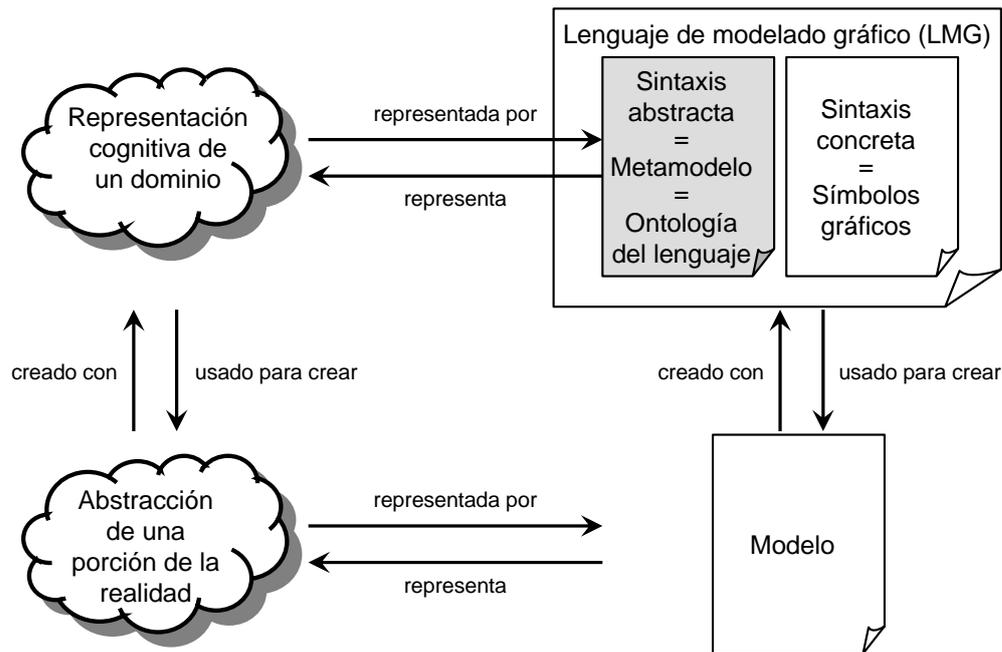


Figura 5.2: Relaciones entre abstracción, modelo, lenguaje de modelado y representación cognitiva de un dominio dado

como OCL (*Object Constraint Language*)<sup>1</sup>, lógica de primer orden u otros. Estas condiciones restringen la sintaxis del lenguaje definiendo el conjunto de sentencias bien formadas del lenguaje.

Un metamodelo es un modelo explícito de los bloques de construcción y reglas necesarias para combinar estos bloques, con el fin de construir modelos específicos válidos dentro de un dominio de interés (Gašević y otros, 2006). Este metamodelo es el resultado de un proceso de abstracción, clasificación y generalización en el dominio del problema del lenguaje de modelado (Mellor y otros, 2004). El metamodelo del lenguaje es una representación concreta de la visión del mundo subyacente a ese lenguaje; dicha visión está embebida en las primitivas de modelado del lenguaje (Guizzardi, 2007). Esto caracteriza un metamodelo como una ontología (Gašević y otros, 2006; Guizzardi, 2007), la ontología del lenguaje.

Ya que el objetivo de este capítulo es obtener un lenguaje de modelado que permita representar las características contextuales de las entidades, es importante que este lenguaje incluya primitivas que representen las leyes generales que describen la realidad. Adhiriendo a las ideas de Guarino (1994, 2009) para

<sup>1</sup> [http://www.omg.org/technology/documents/modeling\\_spec\\_catalog.htm#OCL](http://www.omg.org/technology/documents/modeling_spec_catalog.htm#OCL)

representación de conocimiento, al modelar una característica es importante “marcarla” con una categoría ontológica, dotada de una axiomatización adecuada, con el fin de asegurar que se comunica el significado previsto, y se excluyen por lo menos los malentendidos más graves. Con este objetivo, se puede definir un lenguaje general de representación de características (LGRC) que se pueda utilizar para desarrollar la ontología del lenguaje para un dominio dado (Figura 5.3). Es decir, la ontología de este lenguaje general de representación de características debe ser una meta-ontología que represente un conjunto de categorías del mundo real, independiente del dominio (Guizzardi, 2007).

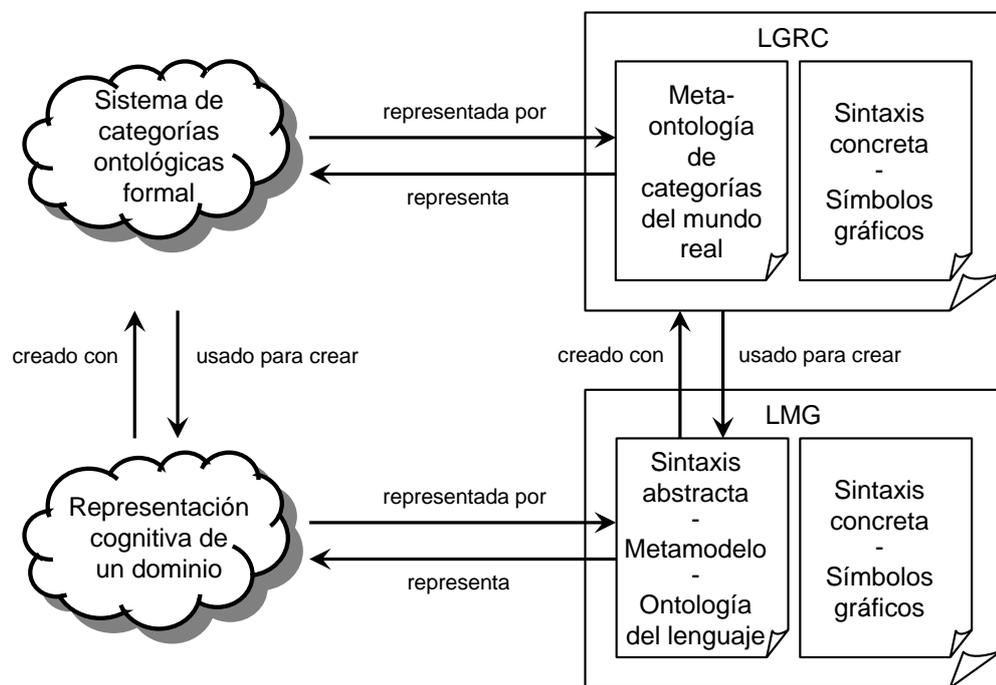


Figura 5.3: Relaciones entre la representación cognitiva de un dominio, lenguaje de modelado, lenguaje general de representación y un sistema de categorías ontológicas formal

La idoneidad de un lenguaje para crear modelos en un dominio determinado depende de cuánto se asemejen la estructura de los modelos construidos con ese lenguaje y la estructura de las abstracciones de dominio que se supone esos modelos representan (Guizzardi, 2007). Esto es válido tanto para los lenguajes específicos de dominio como para los lenguajes independientes del dominio. Para poder comparar dichas estructuras, la estructura de las abstracciones del dominio también debe ser accesible a través de una descripción explícita y formal de la correspondiente porción de la realidad en términos de un artefacto concreto,

que se denomina ontología de referencia para ese dominio. La idea es que una ontología de referencia debe ser construida con el único objetivo de hacer la mejor descripción posible de la realidad con respecto a un cierto nivel de granularidad y punto de vista.

La Ontología Formal es la rama de la Filosofía que se ocupa de las estructuras ontológicas formales que se aplican a todos los dominios (por ejemplo teoría de las partes, teoría de conjuntos, tipos y creación de instancias, identidad, dependencia, unidad). Una ontología, cuando se construye con las teorías desarrolladas por la Ontología Formal en la Filosofía, se llama ontología fundacional. Guizzardi (2005) desarrolló una ontología fundacional denominada UFO (*Unified Foundational Ontology*) que comprende un sistema de categorías del mundo real y se puede utilizar como base ontológica para evaluar, rediseñar y proporcionar la semántica del mundo real para los lenguajes de representación de ontologías. En esta tesis se adopta la ontología UFO como ontología de referencia para desarrollar la meta-ontología del lenguaje general de representación de características.

Además de una sintaxis abstracta, como ya se mencionó, un lenguaje necesita una sintaxis concreta y herramientas para manipularlo.

Si bien existen propuestas como la del Object Management Group (2009) para usar UML (*Unified Modeling Language*) como lenguaje de modelado de ontologías, son varios los inconvenientes de su uso. Considerando que UML no es sólo una notación gráfica (sintaxis concreta) y que la noción de modelo UML se corresponde semánticamente a la noción de ontología (Gašević y otros, 2006) (sintaxis abstracta), se pueden comparar los lenguajes de ontologías y UML. Algunas de las incompatibilidades más relevantes entre ellos son:

- Los lenguajes de ontologías no tienen una separación rígida entre metaniveles. Por ejemplo en OWL Full, una instancia de una clase puede ser otra clase.
- La relación de generalización/especialización de UML estimula la reutilización de comportamiento, mientras que la idea detrás de esta relación en los lenguajes de ontologías es la de la teoría de conjuntos.
- Los paquetes UML no están al mismo metanivel que los contenedores RDF (**Bag**, **Seq** y **Alt**) y las características de presentación de los paquetes los

hacen inadecuados para la representación de los contenedores RDF.

- Los lenguajes de ontologías tienen la habilidad de construir clases utilizando operaciones booleanas (unión, intersección y complemento) y cuantificadores. En UML no existe una primitiva que se corresponda con esto.
- La disjunción en UML requiere que las clases disjuntas tengan un supertipo en común, lo cual no es el caso en OWL.
- La falta de semántica de conjunto y teoría de modelos confiables en UML impide el uso de razonadores automáticos sobre modelos UML.
- En UML no es posible que una instancia exista sin una clase que defina su estructura. La solución propuesta a este inconveniente en el perfil de UML desarrollado por el Object Management Group (2009) no permite aplicar las metodologías que inician con las instancias y derivan las clases de ellas.
- En UML, el alcance de una relación se limita a las subclasses de la clase en la que se define. El nombre de una asociación se puede duplicar en un diagrama dado, y cada ocurrencia puede tener una semántica distinta. En OWL, una relación se define por defecto con rango y dominio *Thing* (una clase universal de la cual todo deriva). Por consiguiente, una relación dada puede aplicarse en principio a cualquier clase. Así, el nombre de una relación tiene un alcance global y es la misma relación siempre que aparece.

Aún así, las organizaciones que desarrollan ontologías pueden necesitar aprovechar modelos de datos y procesos preexistentes representados en UML, diagramas de entidad-relación (ER) u otro lenguaje de modelado. Con este propósito, el Object Management Group (2009) desarrolló *The Ontology Definition Metamodel* (ODM). ODM es una familia de meta-modelos MOF (*Meta-Object Facility*), mapeos entre estos meta-modelos y mapeos hacia y desde UML, y un conjunto de perfiles que permiten el modelado de ontologías mediante el uso de herramientas basadas en UML. Los meta-modelos que componen ODM reflejan la sintaxis abstracta de varias representaciones de conocimiento estándar y lenguajes de modelado conceptual que: (a) han sido adoptados por otros organismos internacionales de estandarización (por ejemplo, RDF, RDF Schema

y OWL del W3C); (b) se encuentran en proceso de aprobación (por ejemplo, *Common Logic* y *Topic Maps* por la ISO); o (c) se consideran estándares de facto de la industria (ER).

Sin embargo, el mismo Object Management Group (2009) remarca que es muy importante entender, por parte de quienes están influenciando el uso de artefactos existentes para el desarrollo de ontologías, que “lo que hace a un buen modelo de componentes de software orientado a objetos” no necesariamente hace a una buena ontología. Una vez que un modelo particular ha sido traducido a OWL por ejemplo, se necesita garantizar que el modelo resultante respalda las afirmaciones deseadas para el dominio considerado. En otras palabras, a menudo se necesita una reestructuración importante.

Dadas las desventajas que presenta UML y las soluciones poco naturales propuestas para un ingeniero ontológico, en el marco de esta tesis se decidió desarrollar el lenguaje general de representación de características a partir de ECore<sup>2</sup>. ECore es un meta-meta-modelo muy similar pero más simple que MOF (Gašević y otros, 2006). Las similitudes residen en su capacidad para especificar las clases y sus características estructurales y de comportamiento, la herencia, los paquetes y la reflexión. Se diferencian en el área del ciclo de vida, las estructuras de tipo de datos, las relaciones entre paquetes y los aspectos complejos de las asociaciones (Steinberg y otros, 2008).

También en el marco de esta tesis se desarrolló una primera versión de una herramienta que implementa la meta-ontología del lenguaje general de representación de características, y permite enriquecer la representación de las características de una ontología ya creada. Esta herramienta se denomina MetaXys (*Metamodel eXtension system*) y se implementó con *Eclipse Modeling Framework Project* (EMF).

## 5.2. Ontología de Referencia

La sintaxis abstracta del lenguaje general de representación de características debe contener un sistema de categorías ontológicas para representar las leyes generales que describen la realidad. Con este objetivo se

---

<sup>2</sup> <http://download.eclipse.org/modeling/emf/emf/javadoc/2.6.0/org/eclipse/emf/ecore/package-summary.html#details>

desarrolla una meta-ontología.

A fin de lograr que esta meta-ontología refleje lo mejor posible la realidad, se toma como ontología de referencia una porción de la ontología fundacional UFO, desarrollada por Guizzardi (2005), y se la extiende con elementos ontológicos definidos por Dockhorn Costa y otros (2006). Esta ontología de referencia así definida representa las abstracciones de un sistema de categorías ontológicas. La idea es que cuanto más se asemejen la meta-ontología del lenguaje general de representación de características y la ontología de referencia, más adecuado será el mencionado lenguaje para crear modelos en un dominio determinado. La Figura 5.4 muestra un pequeño extracto de los elementos más relevantes de la ontología de referencia para el objetivo de este capítulo, es decir, aquellos elementos que dan fundamento ontológico a las características de entidades.

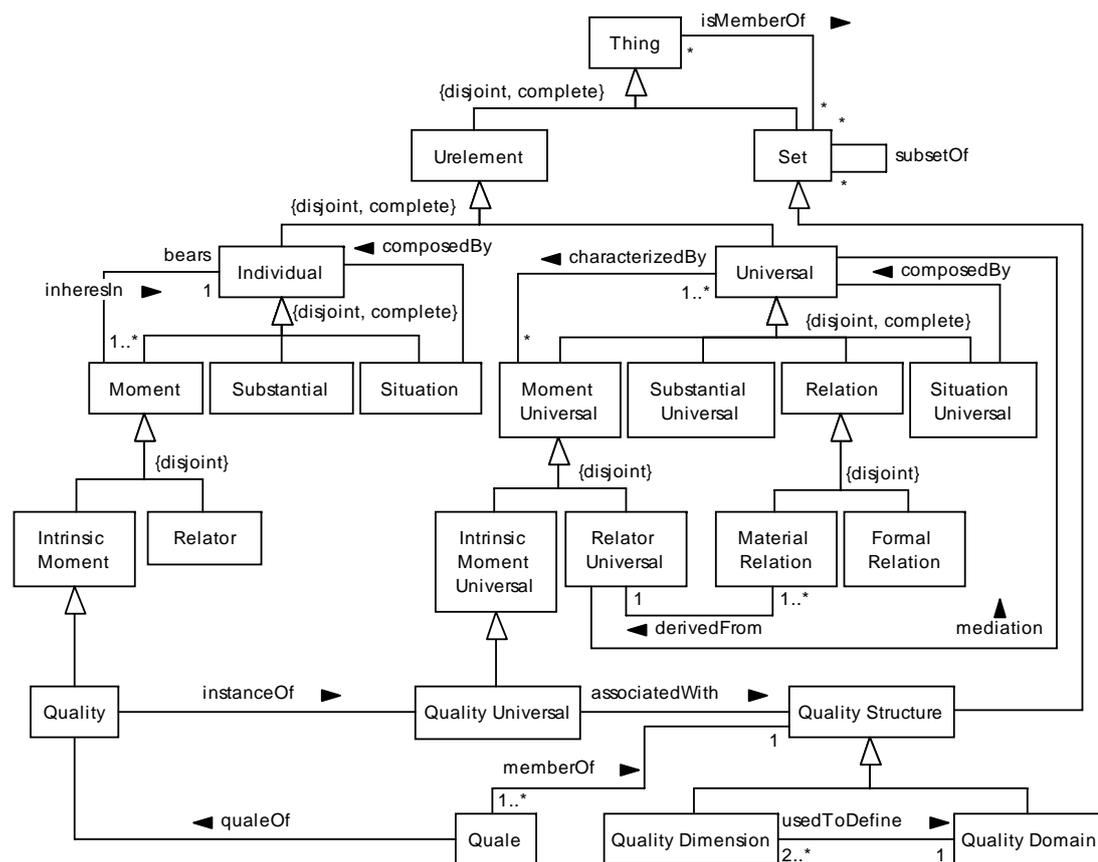


Figura 5.4: Fragmento de la ontología fundacional definida por Guizzardi (2005), extendida con elementos ontológicos definidos por Dockhorn Costa y otros (2006)

Una característica clave en esta ontología es la distinción entre las categorías de conjuntos (**Set**) y elementos primordiales (**Urelement**). Los conjuntos son clases

en el sentido matemático. Los elementos primordiales son entidades que no son conjuntos, pero pueden ser elementos de un conjunto.

Los elementos primordiales se clasifican en individuos (**Individual**) y universales (**Universal**). Un elemento primordial debe ser un individuo o un universal, pero no puede ser ambos. Intuitivamente, los individuos refieren a instancias, mientras los universales refieren a tipos.

Los individuos se pueden categorizar como momento (**Moment**), sustancial (**Substantial**) y situación (**Situation**). Los universales se pueden categorizar como momento universal (**Moment Universal**), sustancial universal (**Substantial Universal**) y situación universal (**Situation Universal**).

Un momento es un individuo que es inherente, y por lo tanto existencialmente dependiente de otro individuo llamado su portador. Es decir, un momento no se puede concebir independientemente del individuo que lo porta; por ejemplo, el color de una botella, el peso de una caja, una creencia de una persona, un acuerdo entre socios de negocio, una orden de compra entre un proveedor y un cliente, etc.

Un sustancial es un individuo que no es inherente a otro individuo, es decir, que no es un momento. Ejemplos son una empresa, un producto, un distrito postal, una máquina, etc.

Una situación es un individuo que se compone de otros individuos, sean estos momentos, sustanciales u otras situaciones. Dada la definición de contexto en el marco de esta tesis, un contexto constituye una situación.

Un momento universal (**Moment Universal**) es un universal que sólo se instancia por individuos de tipo momento. Un sustancial universal (**Substantial Universal**) es un universal que se instancia sólo por individuos de tipo sustancial. Finalmente, una situación universal (**Situation Universal**) es un universal que sólo se instancia por individuos de tipo situación.

Los momentos pueden ser: (a) intrínsecos (**Intrinsic Moment**), es decir, cualidades tales como un peso, modos tales como un pensamiento; y (b) relacionales (**Relator**), por ejemplo ser padre o ser hijo, pero también objetos sociales tales como una orden de compra. Los momentos intrínsecos dependen de un único individuo; los relacionales dependen de una pluralidad de individuos. Del mismo modo, los momentos universales pueden ser intrínsecos (**Intrinsic Moment Universal**) o relacionales (**Relator Universal**). Un momento intrínseco universal es

un universal cuyas instancias dependen existencialmente de una única entidad (llamada su portador). Un momento relacional universal es uno cuyas instancias dependen existencialmente de una pluralidad de entidades.

Una cualidad universal (**Quality Universal**) es un momento intrínseco universal que está asociado con una estructura de cualidad (**Quality Structure**). El término **Quality Structure** refiere a dimensiones de cualidad (**Quality Dimension**) y dominios de cualidad (**Quality Domain**).

Una dimensión de cualidad es una estructura unidimensional que es una dimensión de representación en la cognición humana. Por ejemplo, peso (una cualidad universal) está asociado a una estructura de representación unidimensional que es la semirrecta de los números reales positivos incluyendo el cero.

Un dominio de cualidad es un conjunto de dimensiones integrales que se pueden separar de todas las demás dimensiones. Una dimensión integral es una en la cual no es posible asignarle a un objeto un valor en una dimensión sin asignarle un valor en las otras. Por ejemplo, color puede ser representado mediante la mezcla por adición de los tres colores luz primarios con que se forma: el rojo, el verde y el azul<sup>3</sup>. Cada uno de estos colores luz primarios tiene una dimensión asociada que indica la intensidad de ese color en la formación del color resultante. Estas son dimensiones integrales en el sentido que se debe especificar un valor a cada una de ellas para definir el color. Aunque el intervalo de valores podría ser cualquiera, es frecuente que cada color primario se codifique con un byte (8 bits). Así, de manera usual, la intensidad de cada una de las componentes se mide según una escala que va del 0 al 255. Los dominios de cualidad están dotados de ciertas estructuras geométricas que limitan las relaciones entre las dimensiones que los conforman. Un punto en un dominio de cualidad n-dimensional se llama **Quale** y es un vector que representa las varias dimensiones de cualidad que dependen mutuamente en un dominio de cualidad; por ejemplo, el amarillo es (255,255,0).

Las cualidades (**Quality**) son momentos intrínsecos que instancian cualidades universales. Las cualidades se pueden categorizar como simples (**Simple Quality**) o complejas (**Complex Quality**). Una cualidad simple es una que no porta otras cualidades; por ejemplo, el peso de una cosa. Por el contrario, una

---

<sup>3</sup> Existen otros sistemas de colores. Por ejemplo el modelo de color RYB se forma con rojo, amarillo y azul como colores primarios; el modelo de color CMYK se basa en la mezcla de pigmentos de los colores cian, magenta, amarillo y negro para crear otros colores.

calidad compleja es una que porta otras cualidades; por ejemplo, color. Las cualidades universales instanciadas por cualidades simples y complejas se llaman respectivamente cualidades simples universales (**Simple Quality Universal**) y cualidades complejas universales (**Complex Quality Universal**). Las cualidades simples universales están siempre asociadas con dimensiones de cualidad y viceversa. Las cualidades complejas universales están siempre asociadas con dominios de cualidad.

Para una cualidad universal dada puede haber más de una representación posible. Por ejemplo, color se puede representar en términos del modelo de color RGB (del inglés *Red, Green, Blue*; “rojo, verde, azul”), pero también se puede representar en término de las dimensiones de brillo, tono y saturación, o se puede representar por una simple enumeración de valores posibles. En los dos primeros casos, la cualidad universal color está asociada a un dominio de cualidad y por lo tanto se la considera una cualidad compleja universal, mientras que en el último caso está asociada a una representación unidimensional y se la trata como una cualidad simple universal.

Dentro de la categoría de los universales (**Universal**) existe otra categoría llamada relación (**Relation**). Las relaciones son entidades que unen otras entidades. Estas relaciones se dividen en formales (**Formal Relation**) y materiales (**Material Relation**).

Las relaciones formales se dan entre dos o más entidades directamente, sin que intervengan más individuos. Ejemplos de éstas son las relaciones de instanciación, asociación, comparación (mayor que, más liviano que, etc.), entre otros.

Las relaciones materiales, por el contrario, tienen una estructura material e incluyen ejemplos como empleos, matrículas, órdenes de compra y acuerdos de negocio. Estas relaciones materiales derivan de los momentos relacionales universales (**Relator Universal**). Los elementos relacionados a través de una relación material están mediados por momentos relacionales **Relator**. Por ejemplo, una matrícula es un momento relacional que vincula un estudiante con una institución educativa. La relación de mediación (**mediation**) se da entre un universal (**Universal**: estudiante, institución educativa) y un momento relacional universal (**Relator Universal**: matrícula), si y sólo si cada instancia del universal es mediada por una instancia de un momento relacional universal.

## 5.3. Lenguaje General de Representación de Características

Como ya se mencionó, un lenguaje se compone de una sintaxis abstracta o estructura, y una sintaxis concreta. La sintaxis abstracta del lenguaje general de representación de características la constituye una meta-ontología que representa la visión del mundo subyacente a este lenguaje, es decir, un sistema de categorías ontológicas, que se describe a continuación. La primera versión de un prototipo de una herramienta que implementa la sintaxis concreta se describe en la Sección 5.3.2.

### 5.3.1. Meta-ontología de características contextuales

La sintaxis abstracta del lenguaje general de representación de características es una meta-ontología que se basa en la ontología de referencia descrita en la Sección 5.2, particularmente en la rama de los universales. Esta meta-ontología propuesta se resume en la Figura 5.5.

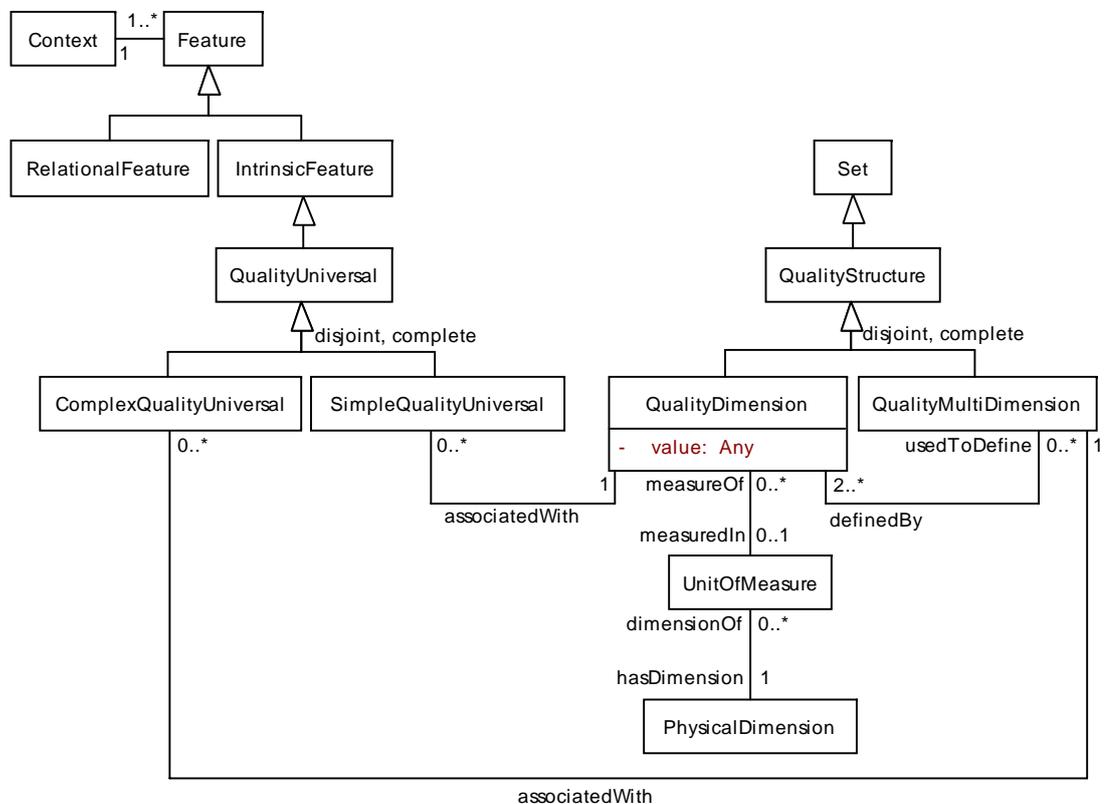


Figura 5.5: Meta-ontología de características contextuales

A continuación se describe cada uno de los términos que forman esta meta-ontología, sus propiedades, relaciones, restricciones y axiomas. Las restricciones y axiomas se definen en lógica descriptiva.

### 5.3.1.1. Context

#### *Semántica*

El término **Context** representa un universal cuyas instancias refieren a contextos. **Context** representa una entidad compuesta, cuyos componentes son el conjunto de características que describen las circunstancias en las cuales algo existe u ocurre. Así, en términos de la ontología fundacional, **Context** representa una situación universal (**Situation Universal**).

#### *Propiedades*

- **id**: String [1]. Representa el identificador del elemento.

#### *Relaciones*

- **characterizedBy**: Feature [1..\*]. Relaciona un contexto a una o más características que describen las circunstancias del contexto. Esta relación es inversa de la relación **characterizes**.

#### *Restricciones*

1. El identificador debe ser único.
2. Un contexto debe estar relacionado con al menos una característica.

$\geq 1$  characterizedBy

#### *Axiomas*

Ninguno.

### 5.3.1.2. Feature

#### *Semántica*

El conjunto de características de un contexto, que afectan el significado de las entidades interpretadas en él, es existencialmente dependiente de un contexto dado y se representa con el término **Feature**. Así, este término representa un momento universal (**Moment Universal**) de la ontología fundacional.

#### *Propiedades*

- **id**: **String** [1]. Representa el identificador del elemento.

#### *Relaciones*

- **characterizes**: **Context** [1]. Relaciona una característica al contexto que describe. Esta relación es inversa de la relación **characterizedBy**.

#### *Restricciones*

1. El identificador debe ser único.
2. Una característica debe estar relacionada con exactamente un contexto.

$$= 1 \text{ characterizes}$$

#### *Axiomas*

1. La jerarquía que tiene al término **Feature** como raíz debe ser exhaustiva.

$$\text{Feature} \equiv \text{RelationalFeature} \cup \text{IntrinsicFeature}$$

### 5.3.1.3. RelationalFeature

#### *Semántica*

El término **RelationalFeature** representa momentos relacionales universales (**Relator Universal**) de la ontología fundacional, es decir, representa características

que están relacionadas a una pluralidad de entidades. A efectos de la presente tesis, se incluye sólo para completar la jerarquía de características de un contexto.

*Propiedades*

Ninguna.

*Relaciones*

- isA: Feature.

*Restricciones*

Ninguna.

*Axiomas*

1. `RelationalFeature` e `IntrinsicFeature` son disjuntos, es decir, una instancia que es miembro de `RelationalFeature` no puede ser miembro de `IntrinsicFeature` y viceversa.

$$\text{RelationalFeature} \cap \text{IntrinsicFeature} \subseteq \perp$$

#### 5.3.1.4. `IntrinsicFeature`

*Semántica*

El término `IntrinsicFeature` representa momentos intrínsecos universales (Intrinsic Moment Universal) de la ontología fundacional, es decir, representa características relativas a un contexto, que no dependen de la relación con otras entidades.

*Propiedades*

Ninguna.

*Relaciones*

- isA: Feature.

*Restricciones*

Ninguna.

*Axiomas*

1. `IntrinsicFeature` y `RelationalFeature` son disjuntos, es decir, una instancia que es miembro de `IntrinsicFeature` no puede ser miembro de `RelationalFeature` y viceversa.

$$\text{IntrinsicFeature} \cap \text{RelationalFeature} \subseteq \perp$$

**5.3.1.5. QualityUniversal***Semántica*

El término `QualityUniversal` representa una cualidad universal; es una característica intrínseca (`IntrinsicFeature`) que se asocia a una estructura de cualidad (`QualityStructure`).

Para la misma cualidad universal puede haber más de una representación posible. Cuál estructura de cualidad se adopta depende de la perspectiva y el nivel de detalle requerido de la representación, pero en un contexto dado sólo puede haber una representación asociada a la cualidad universal.

*Propiedades*

Ninguna.

*Relaciones*

- `associatedWith`: `QualityStructure` [1]. Relaciona una cualidad universal a su representación. Esta relación es inversa de la relación `inverseOfAssociatedWith`.
- `isA`: `IntrinsicFeature`.

*Restricciones*

1. Una cualidad universal debe estar relacionada con exactamente una representación en un contexto dado.

$$= 1 \text{ associatedWith}$$

*Axiomas*

1. La jerarquía que tiene al término `QualityUniversal` como raíz debe ser exhaustiva.

$$\text{QualityUniversal} \equiv \text{SimpleQualityUniversal} \cup \text{ComplexQualityUniversal}$$

**5.3.1.6. SimpleQualityUniversal***Semántica*

Las cualidades simples universales (`SimpleQualityUniversal`) son cualidades universales instanciadas por cualidades simples, es decir, cualidades que no portan otras cualidades, y se asocian siempre a dimensiones de cualidad (`QualityDimension`). Así, el término `SimpleQualityUniversal` representa cualidades simples universales de la ontología fundacional.

*Propiedades*

Ninguna.

*Relaciones*

Ninguna.

*Restricciones*

1. Una cualidad simple universal debe estar siempre relacionada con una dimensión de cualidad, por lo cual se restringe la relación `associatedWith:QualityStructure` a sólo instancias del término `QualityDimension`.

$$\forall \text{ associatedWith.QualityDimension}$$

*Axiomas*

1. `SimpleQualityUniversal` y `ComplexQualityUniversal` son disjuntos, es decir, una instancia que es miembro de `SimpleQualityUniversal` no puede ser miembro de `ComplexQualityUniversal` y viceversa.

$$\text{SimpleQualityUniversal} \cap \text{ComplexQualityUniversal} \subseteq \perp$$

### 5.3.1.7. ComplexQualityUniversal

#### *Semántica*

Las cualidades complejas universales (ComplexQualityUniversal) son cualidades universales instanciadas por cualidades complejas, es decir, cualidades que portan otras cualidades, y se asocian siempre a multidimensiones de cualidad (QualityMultiDimension). El término ComplexQualityUniversal representa cualidades complejas universales de la ontología fundacional (Complex Quality Universal).

#### *Propiedades*

Ninguna.

#### *Relaciones*

Ninguna.

#### *Restricciones*

1. Una cualidad compleja universal debe estar siempre relacionada con una multidimensión de cualidad. Debido a esto, se restringe la relación associatedWith: QualityStructure a sólo instancias del término QualityMultiDimension.

$$\forall \text{ associatedWith. QualityMultiDimension}$$

#### *Axiomas*

1. ComplexQualityUniversal y SimpleQualityUniversal son disjuntos, es decir, una instancia que es miembro de ComplexQualityUniversal no puede ser miembro de SimpleQualityUniversal y viceversa.

$$\text{ComplexQualityUniversal} \cap \text{SimpleQualityUniversal} \subseteq \perp$$

### 5.3.1.8. Set

#### *Semántica*

El término Set representa conjuntos en el sentido matemático, sin adherir a ninguna teoría de conjuntos en particular. Se incluye sólo para caracterizar aspectos de otros elementos en la meta-ontología.

#### *Propiedades*

- id: String [1]. Representa el identificador del elemento.

#### *Relaciones*

- isSubSetOf: Set [\*]. Representa la operación de inclusión de conjuntos.

#### *Restricciones*

1. El identificador debe ser único.

#### *Axiomas*

Ninguno.

### 5.3.1.9. QualityStructure

#### *Semántica*

El término QualityStructure representa conjuntos no vacíos; refiere a dimensiones de calidad (QualityDimension) y multidimensiones de calidad (QualityMultiDimension).

#### *Propiedades*

Ninguna.

#### *Relaciones*

- inverseOfAssociatedWith: QualityUniversal [0..\*]. Relaciona una representación a cero o más cualidades universales. Esta relación es inversa de la relación associatedWith.

- isA: Set.

#### *Restricciones*

Ninguna.

#### *Axiomas*

1. La jerarquía que tiene al término **QualityStructure** como raíz debe ser exhaustiva.

$$\text{QualityStructure} \equiv \text{QualityDimension} \cup \text{QualityMultiDimension}$$

### **5.3.1.10. QualityDimension**

#### *Semántica*

El término **QualityDimension** representa la dimensión de representación asociada a una cualidad simple universal (**SimpleQualityUniversal**). También puede representar una dimensión integral que forma parte de una multidimensión de cualidad.

Una dimensión de cualidad no requiere ser un conjunto denso, puede ser uno discreto como por ejemplo una enumeración de valores posibles.

Las dimensiones de cualidad que representan cantidades tienen una unidad de medida asociada, representada por el término **UnitOfMeasure**.

#### *Propiedades*

- **value: Any**. Representa el valor asignado a la cualidad. Su tipo, **Any**, significa que éste será definido al momento de modelar una dimensión de cualidad concreta.

#### *Relaciones*

- **measuredIn: UnitOfMeasure [0..1]**. Relaciona una dimensión de cualidad medible a una unidad de medida en un contexto dado. Esta relación es inversa de la relación **measureOf**.

- `usedToDefine`: `QualityMultiDimension` [0..\*]. Relaciona una dimensión de cualidad integral a la multidimensión que conforma. Esta relación es inversa de la relación `definedBy`.
- `isA`: `QualityStructure`.

### *Restricciones*

1. Una dimensión de cualidad medible debe estar relacionada como máximo con una unidad de medida en un contexto dado.

$$\leq 1 \text{ measuredIn}$$

2. Una dimensión de cualidad, si está directamente asociada a una cualidad universal, debe serlo con una cualidad simple universal, por lo cual se restringe la relación `inverseOfAssociatedWith`: `QualityUniversal` a sólo instancias del término `SimpleQualityUniversal`.

$$\forall \text{ inverseOfAssociatedWith.SimpleQualityUniversal}$$

### *Axiomas*

1. `QualityDimension` y `QualityMultiDimension` son disjuntos, es decir, una instancia que es miembro de `QualityDimension` no puede ser miembro de `QualityMultiDimension` y viceversa.

$$\text{QualityDimension} \cap \text{QualityMultiDimension} \subseteq \perp$$

#### **5.3.1.11. `QualityMultiDimension`**

##### *Semántica*

El término `QualityMultiDimension` representa el conjunto de dimensiones integrales (Guizzardi, 2005) asociadas a la representación de una cualidad compleja universal (`ComplexQualityUniversal`); representa un dominio de cualidad (`Quality Domain`) de la ontología fundacional.

Las multidimensiones de calidad se definen en términos de un producto cartesiano de sus dimensiones de calidad constitutivas. Las reglas de formación de las tuplas que son miembros de una multidimensión de calidad deben obedecer a las restricciones que relacionan sus dimensiones de calidad, pero éstas sólo se conocen al momento de modelar una multidimensión de calidad concreta.

#### *Propiedades*

Ninguna.

#### *Relaciones*

- **definedBy:** QualityDimension [2..\*]. Relaciona una multidimensión de calidad a las dimensiones integrales que la conforman. Esta relación es inversa de la relación **usedToDefine**.
- **isA:** QualityStructure.

#### *Restricciones*

1. Una multidimensión de calidad debe estar formada por dos dimensiones de calidad como mínimo.

$\geq 2$  definedBy

2. Todas las dimensiones integrales que conforman una multidimensión deben ser disjuntas entre sí. Esto no se puede especificar hasta el momento de modelar una multidimensión concreta.
3. Una multidimensión de calidad debe estar asociada con una calidad compleja universal, por lo cual se restringe la relación **inverseOfAssociatedWith: QualityUniversal** a sólo instancias del término **ComplexQualityUniversal**.

$\forall$  inverseOfAssociatedWith.ComplexQualityUniversal

*Axiomas*

1. `QualityMultiDimension` y `QualityDimension` son disjuntos, es decir, una instancia que es miembro de `QualityMultiDimension` no puede ser miembro de `QualityDimension` y viceversa.

$$\text{QualityMultiDimension} \cap \text{QualityDimension} \subseteq \perp$$

**5.3.1.12. UnitOfMeasure***Semántica*

El término `UnitOfMeasure` representa las unidades de medida asociadas a las dimensiones de cualidad medibles. Las unidades de medida tienen asociada una dimensión física, representada por el término `PhysicalDimension`.

*Propiedades*

- `id: String [1]`. Representa el identificador del elemento.

*Relaciones*

- `measureOf: QualityDimension [0..*]`. Relaciona una unidad de medida a las dimensiones de cualidad que se miden en dicha unidad en un contexto dado. Esta relación es inversa de la relación `measuredIn`.
- `hasDimension: PhysicalDimension [1]`. Relaciona una unidad de medida a su dimensión. Esta relación es inversa de la relación `dimensionOf`.

*Restricciones*

1. El identificador debe ser único.
2. Una unidad de medida debe estar relacionada con exactamente una dimensión física.

$$= 1 \text{ hasDimension}$$

*Axiomas*

Ninguno.

### 5.3.1.13. PhysicalDimension

#### *Semántica*

El término `PhysicalDimension` representa las dimensiones de las unidades de medida.

Densidad, longitud, frecuencia, masa, tiempo, velocidad, etc., son dimensiones físicas. Sólo las cantidades de la misma dimensión se pueden sumar o comparar.

#### *Propiedades*

- `id`: `String [1]`. Representa el identificador del elemento.

#### *Relaciones*

- `dimensionOf`: `UnitOfMeasure [0..*]`. Relaciona una dimensión física a las unidades de medida de dicha dimensión. Esta relación es inversa de la relación `hasDimension`.

#### *Restricciones*

1. El identificador debe ser único.

#### *Axiomas*

Ninguno.

## 5.3.2. Implementación del lenguaje en la herramienta **MetaXys**

MetaXys es una herramienta prototipo que permite implementar el lenguaje general de representación de características, cuya sintaxis abstracta se acaba de presentar en la subsección anterior. El principal objetivo de MetaXys es permitir enriquecer la representación de las entidades y sus características en una ontología existente.

El alcance de la primera versión de la herramienta se limita a las siguientes funcionalidades:

- Cargar, gestionar en memoria y almacenar ontologías implementadas en el lenguaje OWL. Las ontologías deben respetar las siguientes restricciones:
  - La ontología no debe contener instancias. El principal objetivo de la herramienta es enriquecer la representación de las entidades y sus características, lo cual es parte de la etapa de modelado, la instanciación no forma parte de ella.
  - Los dominios y rangos de las relaciones no pueden estar especificados por múltiples términos, sólo se admite uno. El rango de una relación puede no estar definido.
  - La ontología puede tener axiomas, clases anónimas y subpropiedades, pero éstos no se visualizan en esta primera versión de la herramienta.
- Visualizar los términos, relaciones y propiedades de la ontología.
- Permitir el enriquecimiento de las entidades y sus características a través de las primitivas de modelado del lenguaje: cualidades simples y complejas universales, dimensiones y multidimensiones de cualidad.
- Validar el desarrollo de la ontología realizado por el usuario haciendo uso de la meta-ontología del lenguaje implementada en la herramienta.

La herramienta se implementó en EMF<sup>4</sup>. EMF es un *framework* potente con la facilidad de generación de código para crear aplicaciones basadas en Java a partir definiciones simples de modelos (Steinberg y otros, 2008).

### 5.3.2.1. Meta-ontología implementada en Ecore

La Figura 5.6 presenta la meta-ontología que se implementó en Ecore. Ecore es el meta-meta-modelo que utiliza EMF para representar modelos.

En Ecore, la clase `EClass` se utiliza para modelar clases. Éstas se identifican con un nombre y pueden tener atributos y referencias. También, para soportar la herencia, una clase puede referir a otras clases como sus supertipos. En la meta-ontología, los términos tienen propiedades y relaciones, inclusive relaciones de herencia, así es que se utiliza la clase `EClass` para implementar los términos de la meta-ontología, que en la Figura 5.6 corresponden a las cajas. En la parte

---

<sup>4</sup> <http://www.eclipse.org/modeling/emf/?project=emf>

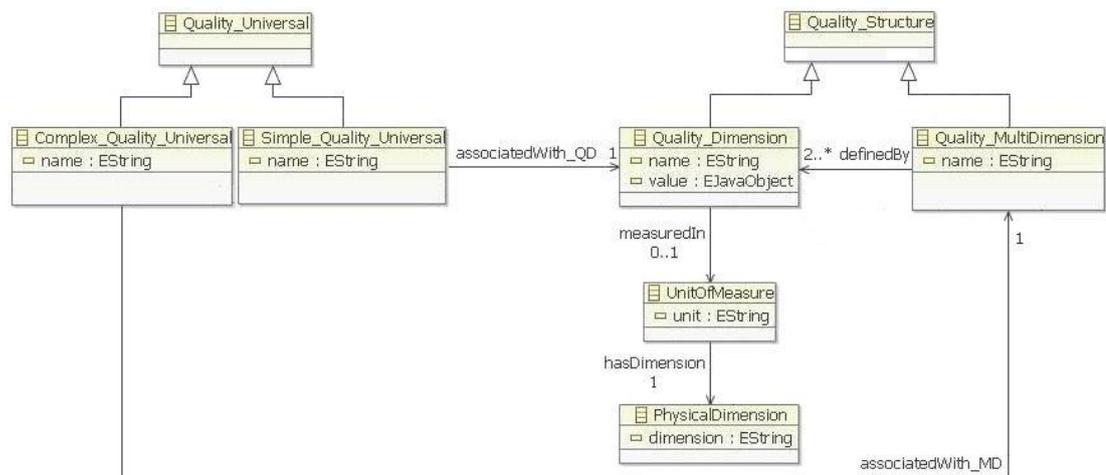


Figura 5.6: Porción de la meta-ontología implementada en Ecore

superior central de cada caja se lee el nombre de la instancia, que corresponde al nombre del término de la meta-ontología que se implementa. Las relaciones de herencia se representan con flechas con punta en forma de triángulo relleno blanco (la flecha apunta hacia el término “padre”).

La clase `EAttribute` de Ecore modela atributos de una clase `EClass`. Se identifican por un nombre y tienen un tipo que corresponde a alguno de los tipos primitivos u objeto definidos en Java. En la meta-ontología, las propiedades tienen características que se corresponden con las de `EAttribute`, por lo cual se implementan como instancias de esta clase de Ecore. En la Figura 5.6, la implementación de las propiedades se visualiza en la parte central de cada caja, encerrada entre dos líneas horizontales, en el formato `nombrePropiedad : TipoDeDato`.

La clase `EReference` se utiliza en Ecore para modelar las asociaciones entre clases; modela un extremo de dicha asociación. Al igual que los atributos, las referencias se identifican por un nombre y tienen un tipo. Sin embargo, este tipo debe ser la `EClass` en el otro extremo de la asociación. Si la asociación es navegable en la dirección opuesta, deberá haber otra referencia para representar esta bidireccionalidad. Una referencia especifica un límite inferior y uno superior de su multiplicidad. Por último, una referencia especifica si está siendo utilizada para representar un tipo más fuerte de la asociación, denominada de composición, que implica una relación todo-parte y donde la “vida” de la parte termina cuando termina la del todo. En la meta-ontología existen relaciones del tipo asociación,

donde los dos términos participantes tienen la misma importancia en la relación, es decir, no son de composición. Al igual que las asociaciones, las relaciones en la meta-ontología tienen un nombre y un sentido (una orientación) que apunta hacia el término que es el rango de la relación. Una relación puede ser la inversa de otra relación, y así se produce la “bidireccionalidad”. Asociada a una relación puede haber restricciones de cardinalidad que indican el número mínimo, máximo o de igualdad de esa relación en la que puede participar un individuo. Debido a estas similitudes, las relaciones de la meta-ontología se implementan como instancias de la clase `EReference`, que en la Figura 5.6 se representan por flechas dirigidas en el sentido de la clase que es el destino de la asociación; en términos de la meta-ontología, el rango de la relación. El nombre y los límites inferior y superior, cuando existen, se indican junto a la flecha que representa la relación.

### 5.3.2.2. Sintaxis concreta

Una vez iniciada la ejecución de la herramienta MetaXys y habiendo seleccionado un archivo con extensión `owl` para cargar en memoria, se visualiza una ventana como la que se muestra en la Figura 5.7.

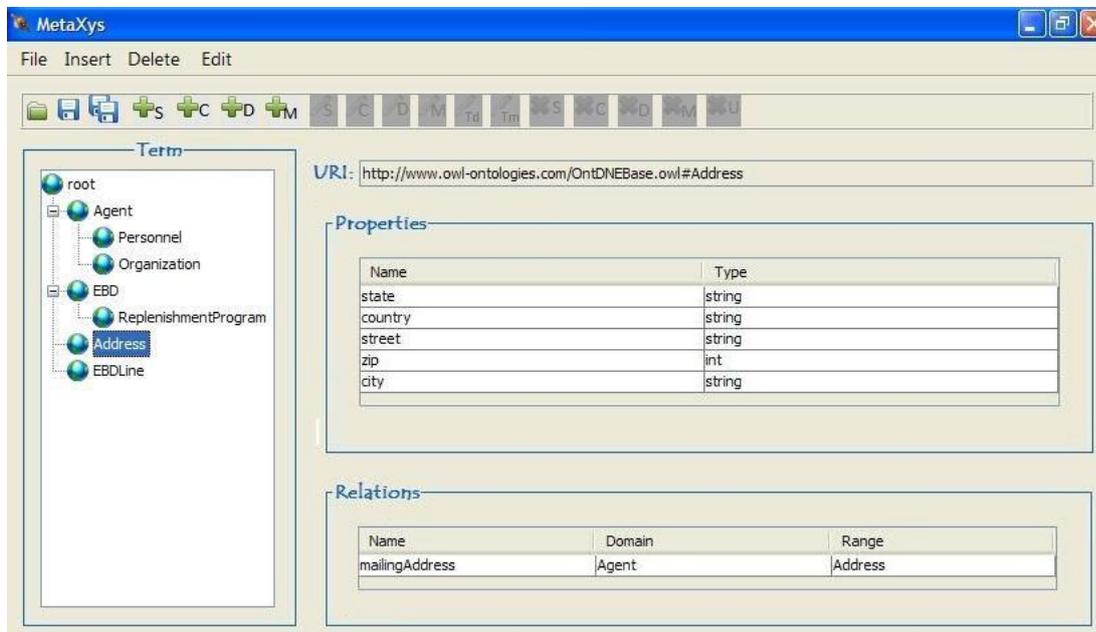


Figura 5.7: MetaXys: Ontología DNE Base

En la parte superior de la ventana, la barra de herramientas muestra habilitados los botones cuyas acciones se pueden ejecutar para el estado actual

de la aplicación.

En el lado izquierdo de la ventana se visualiza la jerarquía de términos de la ontología abierta, es decir, la jerarquía de instancias de la clase **EClass** de **Ecore** creada al abrir la ontología. Esta jerarquía muestra las relaciones de herencia entre las instancias de la clase **EClass** y se representa en forma de árbol. Cada rama del árbol cuya visualización se puede extender se muestra con un signo más (+); cada rama cuya visualización se puede contraer se muestra con un signo menos (-) (Figura 5.8). Siguiendo la notación simbólica de otras herramientas de edición de ontologías, cada término en el árbol (cada instancia de la clase **EClass**) se representa con un círculo y un texto a su lado que es el nombre del término (por ejemplo,  **EBDLine**).

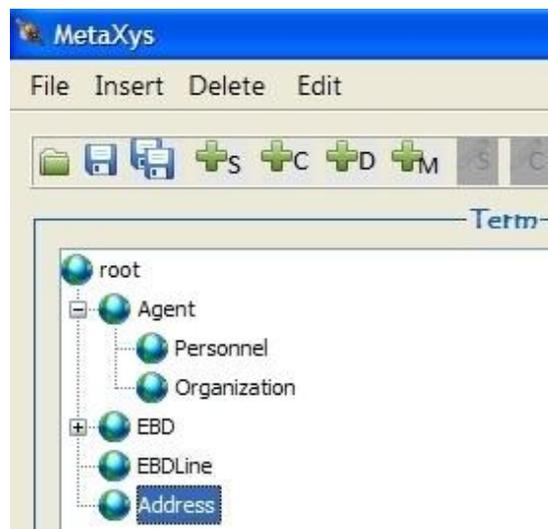


Figura 5.8: MetaXys: Jerarquía de términos

Las acciones que se pueden realizar con MetaXys tienen que ver con la inserción, edición y eliminación de las cuatro primitivas más importantes del lenguaje: cualidades simples universales, cualidades complejas universales, dimensiones de cualidad y multidimensiones de cualidad. Todas ellas se crean como instancias de la clase **EClass**, ya que representan términos de la ontología, y por lo tanto se visualizan dentro de la jerarquía de términos con la simbología que corresponde a ellos.

Retornando al análisis de la Figura 5.7, en el lado derecho de la ventana se lee el '**URI**' (*Unique Resource Identifier*), las propiedades ('**Properties**') y las relaciones ('**Relations**') del término seleccionado (**Address** en este caso).

Las propiedades son instancias de la clase `EAttribute` de `Ecore` y no tienen un símbolo gráfico asociado, su representación es textual: sus nombres y tipos de dato se muestran en forma de tabla.

Las relaciones son instancias de la clase `EReference` de `Ecore` y, al igual que las propiedades, no tienen una representación gráfica asociada. Sus nombres, dominios y rangos se muestran como columnas en una tabla. A diferencia de otras herramientas de edición de ontologías, las relaciones que se visualizan corresponden tanto a las que tienen al término seleccionado en su dominio como en su rango.

## **5.4. Aplicación del Lenguaje de Representación de Características al Caso de Estudio 1**

En esta sección se muestra cómo se puede utilizar el lenguaje general de representación de características al Caso de Estudio 1 (Capítulo 3).

El Caso de Estudio 1 tiene por objetivo desarrollar una ontología para que dos socios de negocio intercambien información en una e-Colaboración. Con este propósito, en el Capítulo 3 se siguieron los pasos definidos en el método propuesto en el Capítulo 2.

En el proceso de Identificación de los requerimientos de la ontología se identificó como fuente de información un documento expresado en XML Schema, que representa la sintaxis y estructura de un DNE utilizado por los socios de negocio para intercambiar información.

Con base en este documento, en el Proceso 2 del método propuesto se generó una ontología base, denominada ontología DNE Base, que se muestra nuevamente en la Figura 5.9 para facilitar la lectura.

Posteriormente, en el proceso de Verificación de dicha ontología base, se determinó que la representación de algunas entidades y características en la ontología necesitaba ser enriquecida para minimizar posibles malos entendidos en el intercambio de información. La Tabla 3.4 (página 78) resume los resultados de esa verificación.

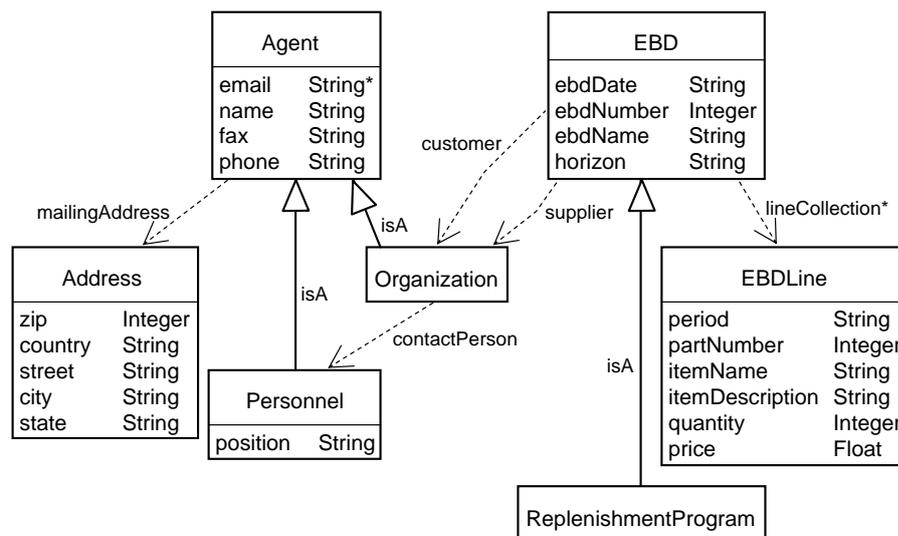


Figura 5.9: Ontología DNE Base

### 5.4.1. Enriquecimiento de la ontología base

A diferencia de lo que se hizo en el Capítulo 3, en este capítulo se realiza el enriquecimiento en la representación de las entidades y sus características a través de la herramienta MetaXys para mostrar su uso.

#### 5.4.1.1. Enriquecimiento de la representación de la entidad socio de negocio

Del proceso de Verificación surgió la necesidad de mejorar la representación de la característica país, asociada a la dirección de un socio de negocio. País, representada en la ontología DNE Base por la propiedad `country` (Figura 5.9), es una característica simple que se puede considerar una entidad en sí misma. Siguiendo los principios de diseño establecidos en esta tesis, esta característica se debe representar por un término que corresponda a la dimensión país (el conjunto de valores posibles) y una relación entre este término y el término que representa a la entidad caracterizada, `Address` en este caso. En la herramienta MetaXys esto se logra con la opción **Insert Quality Dimension** (+D), que abre una ventana (Figura 5.10) que permite ingresar el nombre del término que representará a la dimensión y seleccionar el término que representa a la entidad caracterizada. Como la dimensión es una enumeración de valores posibles, no debe tener una unidad de medida asociada y, por esta razón, se deja sin marcar la casilla de

verificación ‘**Unit Of Measure?**’ en la parte inferior de la ventana.

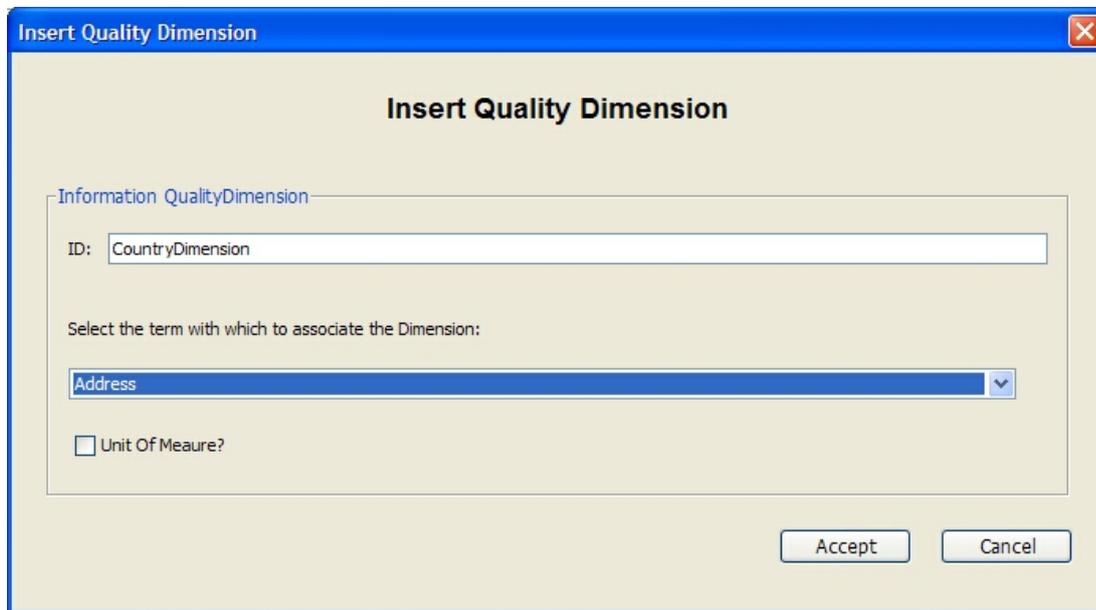


Figura 5.10: MetaXys: Enriquecimiento de la característica país

Al aceptar la inserción haciendo clic en el botón ‘**Accept**’ de la ventana, la herramienta también da la posibilidad de eliminar una propiedad asociada al término que representa a la entidad caracterizada. En este caso, se elimina la propiedad **country** que corresponde a la antigua representación. La Figura 5.11 muestra algunos resultados de estas acciones: **country** ya no es una propiedad del término **Address** y se creó un nuevo término **CountryDimension**, que está asociado al término **Address** a través de la relación **addressAssociatedWithCountryDimension**.

El término **CountryDimension** es una instancia del término **Quality.Dimension** de la meta-ontología implementada en **Ecore** y tiene una propiedad denominada **valueCountryDimension**, cuyo tipo de datos no está definido (**any**) (Figura 5.12). Esta representación permite que, en la etapa de instanciación, varias direcciones tengan asociado el mismo país, representado de igual manera para todas las direcciones, y el tipo de dato de la propiedad **valueCountryDimension** lo decida el ingeniero ontológico según desee o necesite.

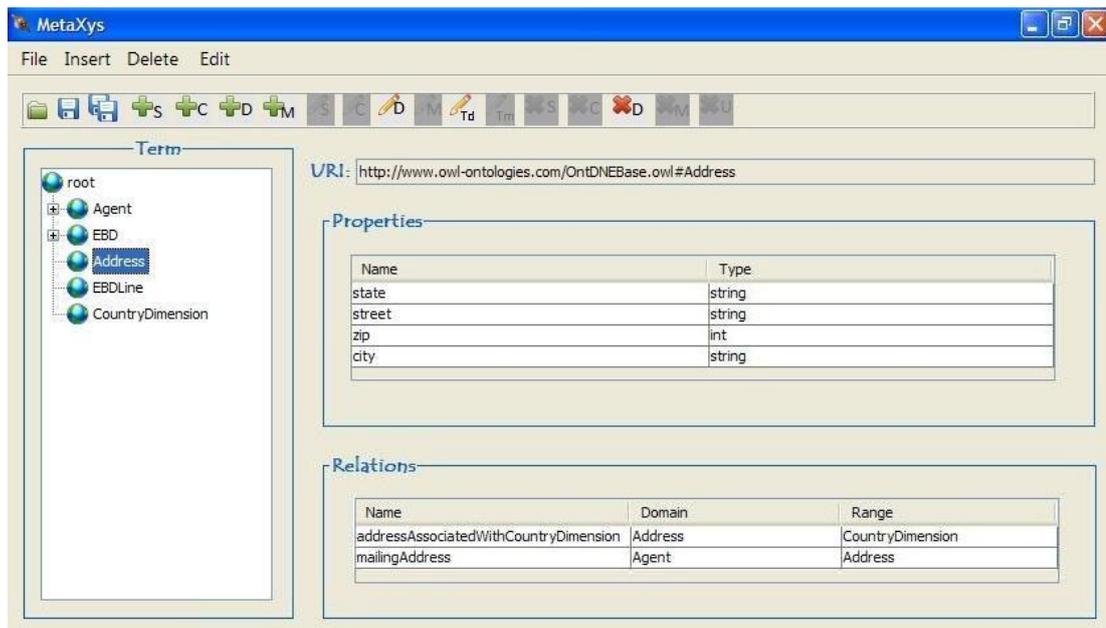


Figura 5.11: MetaXys: Nueva representación de la característica dirección

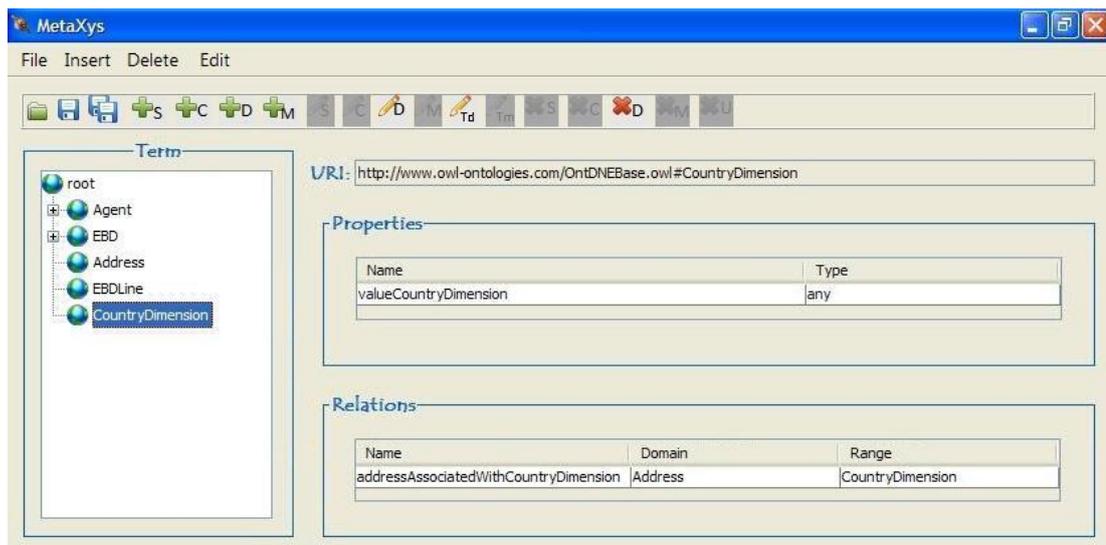


Figura 5.12: MetaXys: Representación de la dimensión país

#### 5.4.1.2. Enriquecimiento de la representación de la entidad programa de provisión

La entidad programa de provisión tiene dos características temporales que requieren una mejor representación: el horizonte de tiempo de la colaboración y la fecha de creación del DNE que lo representa.

La característica horizonte está representada en la ontología DNE Base por la propiedad `horizon` asociada al término `EBD`, mientras que la característica

fecha está representada por la propiedad `ebdDate` asociada al mismo término (Figura 5.9, página 177). Ambas son características simples medibles que, siguiendo los principios de diseño definidos en esta tesis, se deberían representar por los siguientes elementos (Figura 5.13):

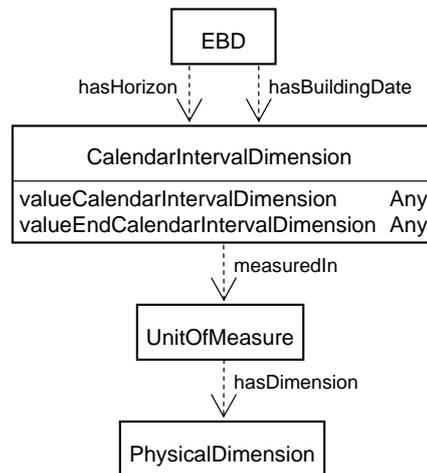


Figura 5.13: Representación posible de las características horizonte y fecha

- (a) Un término que represente la dimensión (tanto horizonte como fecha son intervalos calendarios y podrían representarse por la misma dimensión, `CalendarIntervalDimension`).
- (b) Una relación entre este término y el término que representa a la entidad caracterizada (en este caso sería el término `EBD`; las relaciones, `hasHorizon` y `hasBuildingDate`).
- (c) Un término que representa a la unidad de medida de la dimensión (según la información de ejemplo que se dispone del contenido de un DNE –Figura 3.3, página 65–, la característica horizonte podría medirse en meses y la característica fecha en días, que serían instancias del término `UnitOfMeasure`).
- (d) Una relación entre este término y el término que representa la dimensión (`measuredIn`).
- (e) Un término que representa la dimensión física de la unidad de medida (en este caso la unidad de medida de ambas características corresponde a la dimensión tiempo, que sería una instancia del término `PhysicalDimension`).

- (f) Una relación entre este término y el término que representa la unidad de medida (`hasDimension`).

Esta representación de las características horizonte y fecha, tal como se muestra en la Figura 5.13, no se puede implementar con esta primera versión de la herramienta MetaXys, ya que la misma no permite dos relaciones distintas entre un término que representa una entidad (el término `EBD`) y una dimensión de cualidad (el término `CalendarIntervalDimension`), es decir, no se pueden definir ambas relaciones (`hasHorizon` y `hasBuildingDate`), sino sólo una de ellas.

Una alternativa que sí es viable en MetaXys es representar a las características horizonte y fecha como cualidades simples universales con sus propios términos, y relacionar éstos con la entidad caracterizada y con las dimensiones de cualidad correspondientes. Con este objetivo se puede elegir la opción ‘**Insert Simple Quality Universal**’ (+s) de MetaXys, que abre una ventana como la que se muestra en la Figura 5.14.



The screenshot shows a dialog box titled "Insert Simple Quality Universal". It contains two main sections for inputting information. The first section, "Information SimpleQualityUniversal", has an "ID:" field with the text "Horizon" and a dropdown menu labeled "Select the term with which to associate the Simple Quality Universal:" with "EBD" selected. The second section, "Information QualityDimension", has an "ID:" field with "CalendarIntervalDimension" and a checked checkbox labeled "Unit Of Measure?". At the bottom right, there are "Accept" and "Cancel" buttons.

Figura 5.14: MetaXys: Enriquecimiento de la característica horizonte

En la parte superior de la ventana se ingresa el nombre que se le va a asignar al término que represente la cualidad simple (se comienza con la representación de la característica horizonte por lo cual se ingresa `Horizon`) y se selecciona de

una lista el término que representa a la entidad caracterizada (EBD). En la parte media de la ventana se indica el nombre del término que representa la dimensión de cualidad asociada a la cualidad simple (CalendarIntervalDimension) y si esta dimensión tiene una unidad de medida (casilla de verificación ‘**Unit Of Measure?**’). En la Figura 5.15 se pueden observar algunos de los resultados producidos al hacer clic en el botón ‘**Accept**’.

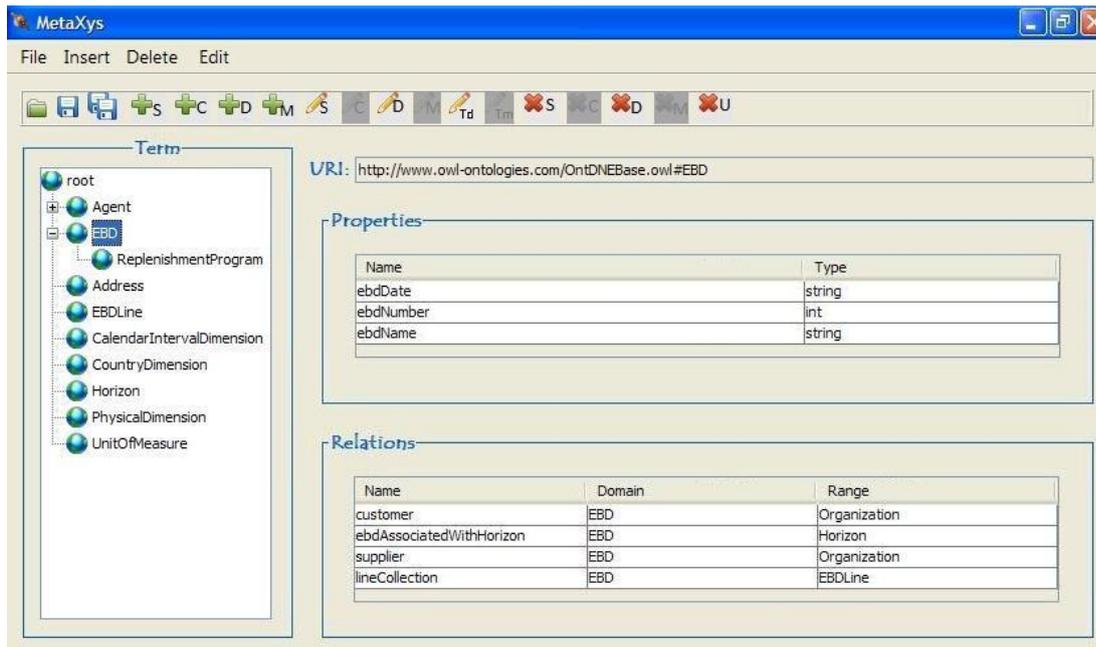


Figura 5.15: MetaXys: Nueva representación de la entidad programa de provisión

Al igual que en el caso de inserción de una dimensión de cualidad visto para la característica país, la herramienta MetaXys también permite aquí eliminar una propiedad que pudiera haber formado parte de la representación de la característica horizonte. Entonces se elimina la propiedad **horizon** y se agregan los siguientes elementos:

- (a) El término **Horizon** que representa la característica horizonte. Este término es instancia del término **Simple\_Quality\_Universal** de la meta-ontología implementada en Ecore.
- (b) La relación **ebdAssociatedWithHorizon** entre los términos **EBD**, que representa a la entidad caracterizada, y **Horizon**.
- (c) El término **CalendarIntervalDimension** que representa la dimensión, con su propiedad **valueCalendarIntervalDimension** (Figura 5.16).

`CalendarIntervalDimension` es instancia del término `Quality_Dimension` de la meta-ontología implementada en Ecore.

- (d) La relación `horizonAssociatedWithCalendarIntervalDimension` entre los términos `Horizon`, que representa a la característica horizonte, y `CalendarIntervalDimension` que representa su dimensión. Esta relación es instancia de la relación `associatedWith_QD` de la meta-ontología implementada en Ecore.
- (e) El término `UnitOfMeasure`, que representa la unidad de medida de la dimensión. Este término es instancia de su homónimo en la meta-ontología implementada en Ecore.
- (f) La relación `calendarIntervalDimensionMeasuredInUnitOfMeasure` entre los términos `CalendarIntervalDimension` y `UnitOfMeasure`. Esta relación es instancia de la relación `measuredIn` en la meta-ontología implementada en Ecore.
- (g) El término `PhysicalDimension`, que representa la dimensión física de la unidad de medida (Figura 5.17). Este término es instancia de su homónimo en la meta-ontología implementada en Ecore.
- (h) La relación `unitOfMeasureHasDimensionPhysicalDimension` entre los términos `UnitOfMeasure` y `PhysicalDimension`. Esta relación es instancia de la relación `hasDimension` de la meta-ontología implementada en Ecore.

Respecto de la representación mostrada en la Figura 5.13 (página 180), esta representación de la característica horizonte agrega el término `Horizon` que en esta tesis se considera innecesario, pero permite hacer explícito el hecho que horizonte se debe interpretar como un intervalo calendario, cuya unidad de medida podría ser, por ejemplo, meses, a lo cual correspondería una dimensión física temporal. Esto no está representado en la ontología DNE Base original (Figura 5.9, página 177).

Realizando los mismos pasos de inserción de una cualidad simple universal se puede obtener la representación de la característica fecha. En este caso se elimina la propiedad `ebdDate` y en su lugar se crea un nuevo término `EBDDate` (Figura 5.18). Este término se relaciona con el término `EBD` a través de la

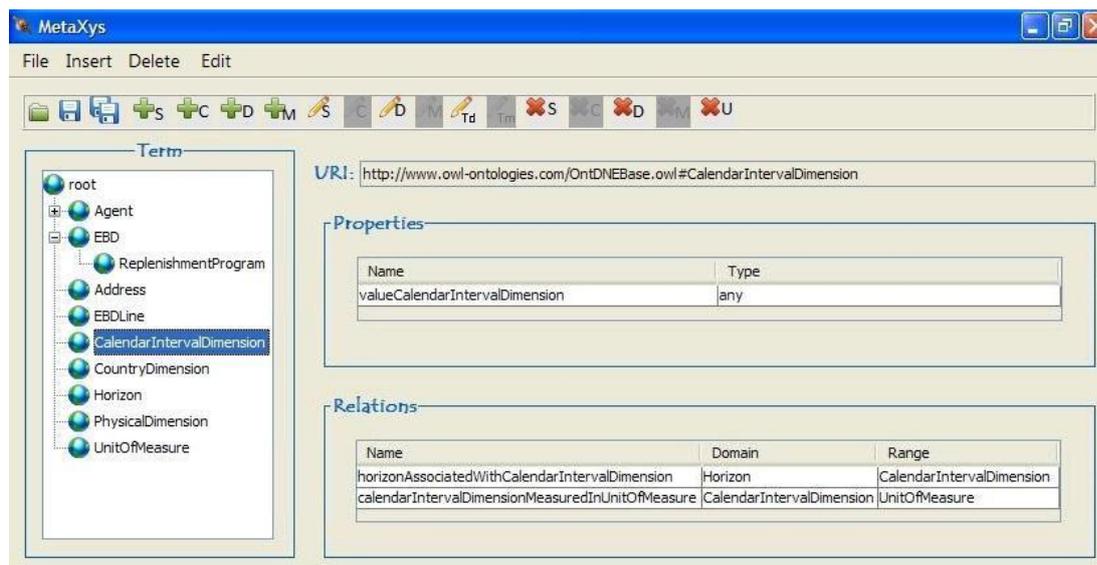


Figura 5.16: MetaXys: Representación de la dimensión intervalo calendario

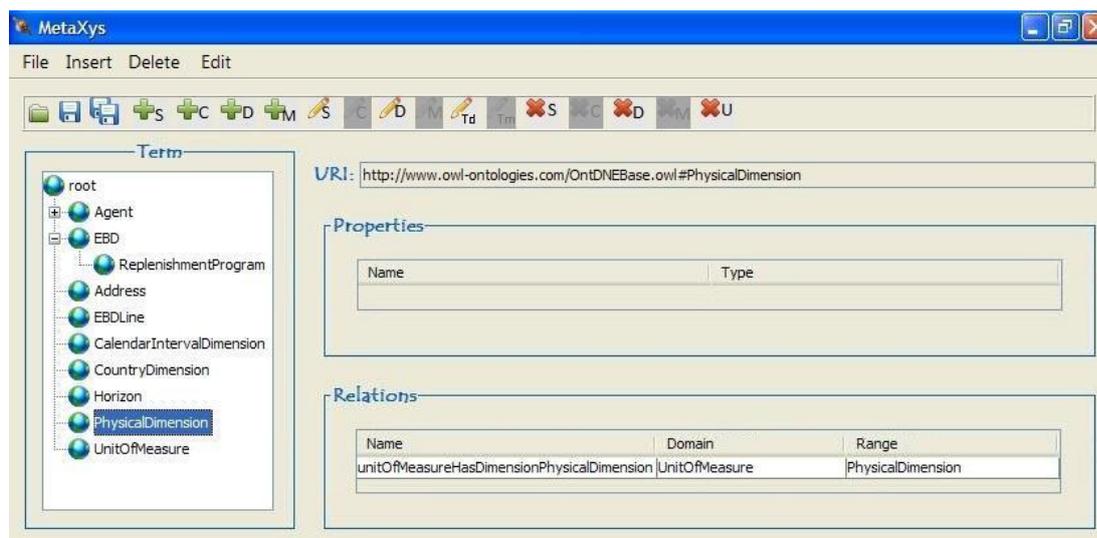


Figura 5.17: MetaXys: Representación de la dimensión física asociada a unidades de medida

relación `ebdAssociatedWithEBDDate`, y con el término `CalendarIntervalDimension` a través de la relación `ebdDateAssociatedWithCalendarIntervalDimension`. El término `CalendarIntervalDimension` es el que se creó en oportunidad de enriquecer la característica horizonte y, por lo tanto, mantiene sus relaciones con los otros términos: `Horizon` y `UnitOfMeasure` (Figura 5.19).

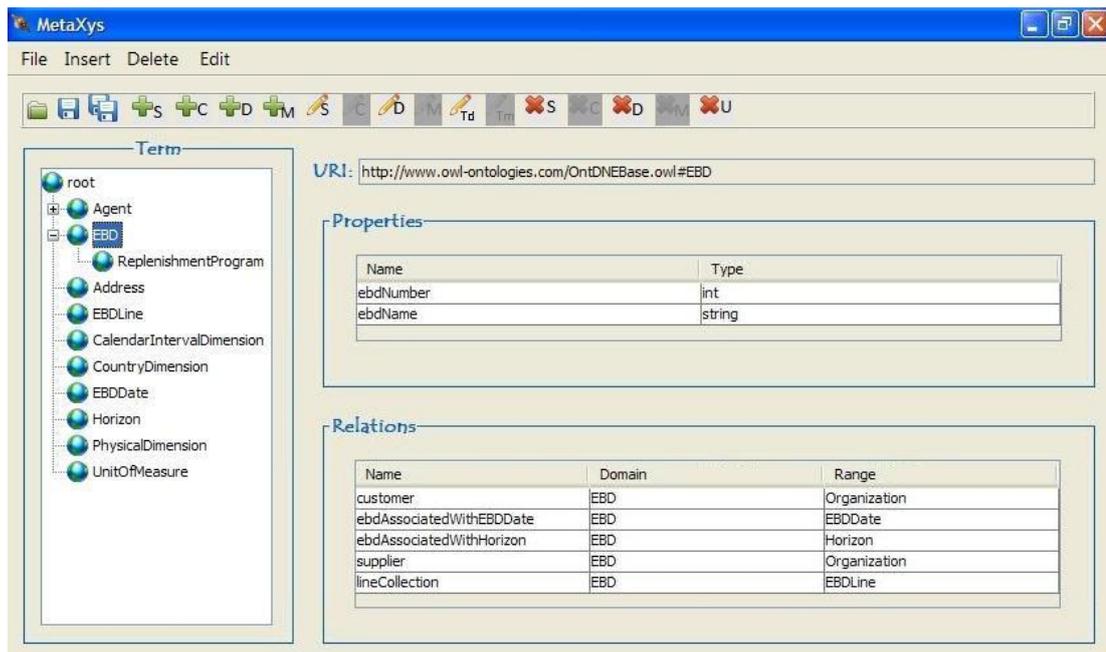


Figura 5.18: MetaXys: Nueva representación de la entidad programa de provisión

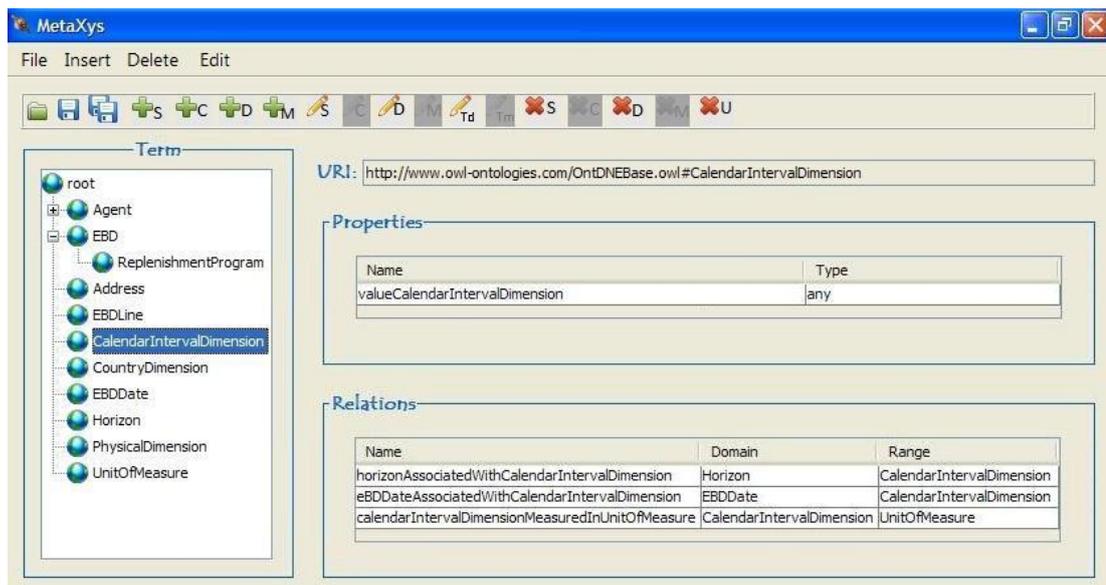


Figura 5.19: MetaXys: Representación de la dimensión intervalo calendario

#### 5.4.1.3. Enriquecimiento de la representación de la entidad ítem

Según los resultados de la verificación, las características período, producto, cantidad y precio de un ítem deben ser mejoradas en su representación.

Ya que período es la única característica temporal a enriquecer de la entidad ítem, es posible representarla en MetaXys como se hubiera querido hacer con

las características horizonte y fecha del programa de provisión (Figura 5.13, página 180). Sin embargo, esto implicaría aplicar distintos criterios de diseño para los mismos tipos de características, y en esta tesis no se considera una práctica adecuada. Por tal motivo, se representa a la característica período como una cualidad simple universal, como se hizo con las características horizonte y fecha. La Figura 5.20 muestra algunos de los resultados de esta decisión.

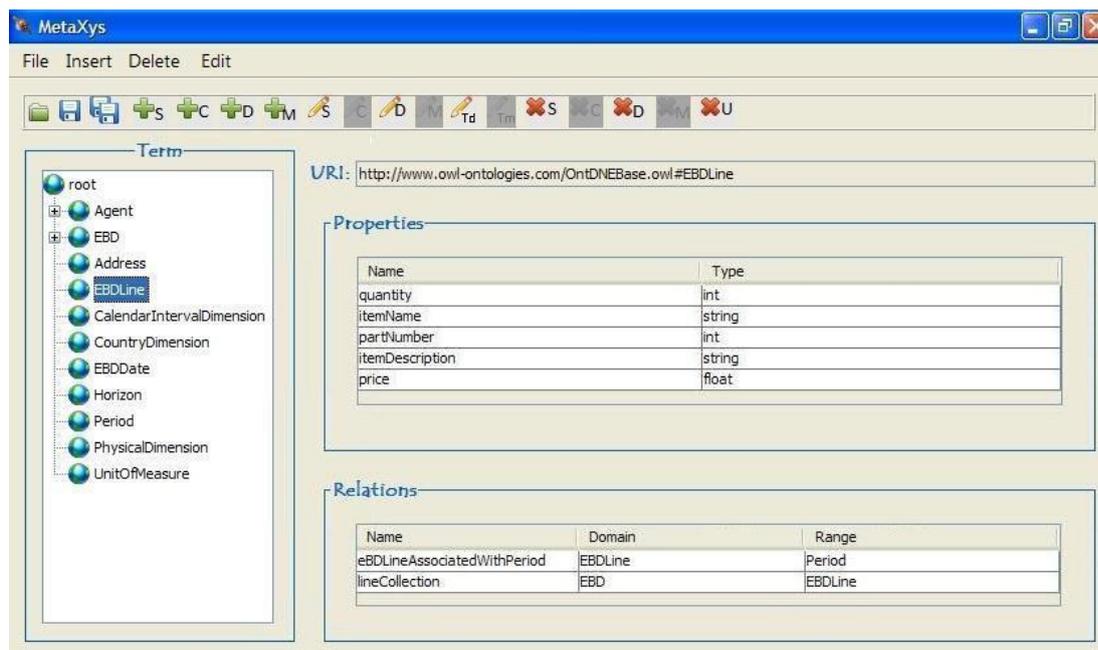


Figura 5.20: MetaXys: Nueva representación de la entidad ítem

Se eliminó la antigua representación de la característica período (la propiedad `period` asociada al término `EBDLine`, Figura 5.9, página 177) y en su lugar se creó el término `Period`. Este término se relaciona con el término `EBDLine` a través de la relación `eBDLineAssociatedWithPeriod`, y con el término `CalendarIntervalDimension` a través de la relación `periodAssociatedWithCalendarIntervalDimension`. Ahora el término `CalendarIntervalDimension` se relaciona con los términos `Horizon`, `Period`, `EBDDate` y `UnitOfMeasure` (Figura 5.21).

Cantidad y precio son también características simples medibles, al igual que el período, pero no comparten la dimensión de cualidad con éste. Para el enriquecimiento de estas características entonces se hace uso de la opción 'Insert Quality Dimension' (+D), como se hizo con el enriquecimiento de la característica país. La Figura 5.22 muestra un resultado parcial de la

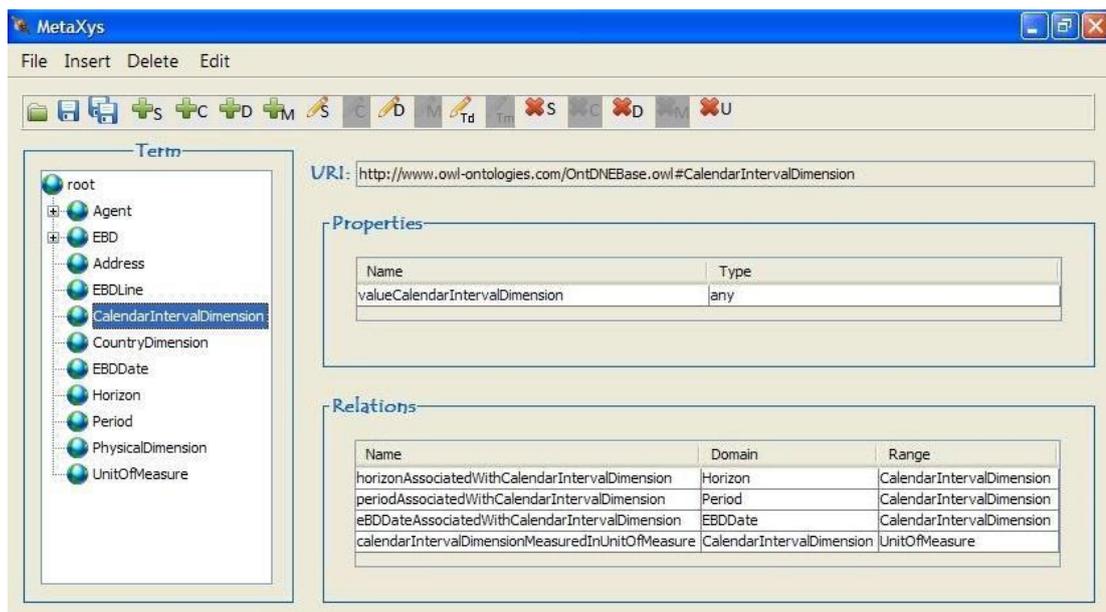


Figura 5.21: MetaXys: Representación de la dimensión intervalo calendario  
representación de la característica cantidad.

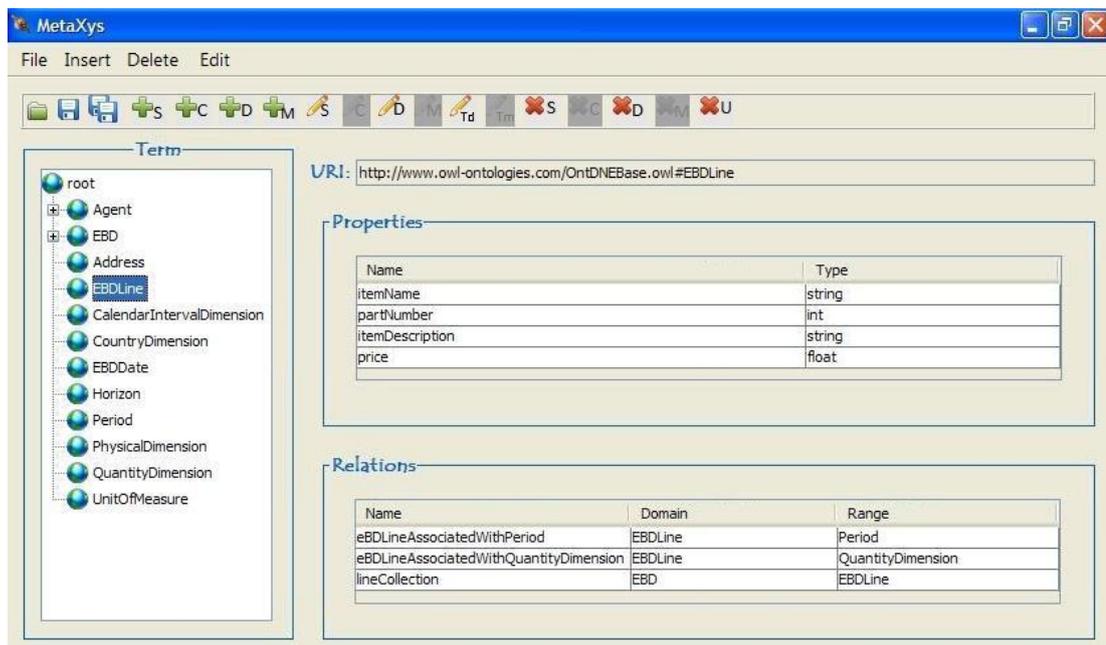


Figura 5.22: MetaXys: Nueva representación de la entidad ítem

La propiedad `quantity`, que representaba a la característica cantidad (Figura 5.9, página 177), se reemplazó por el término `QuantityDimension`, que representa la dimensión cantidad. Este término `QuantityDimension` se relaciona con el término que representa la entidad caracterizada (`EBDLine`) a

través de la relación `eBDLineAssociatedWithQuantityDimension`, y con el término `UnitOfMeasure` a través de la relación `quantityDimensionMeasuredInUnitOfMeasure`. Para las cantidades empleadas en el DNE de ejemplo (Figura 3.3, página 65), la unidad de medida podría ser “unidad” o “unidades de paquetes”, por ejemplo. La dimensión física asociada a estas unidades de medida podría ser “unidad”. El término `UnitOfMeasure` se creó en oportunidad del enriquecimiento de la característica horizonte y, por lo tanto, se relaciona ahora con los términos `CalendarIntervalDimension`, `QuantityDimension` y `PhysicalDimension` (Figura 5.23).

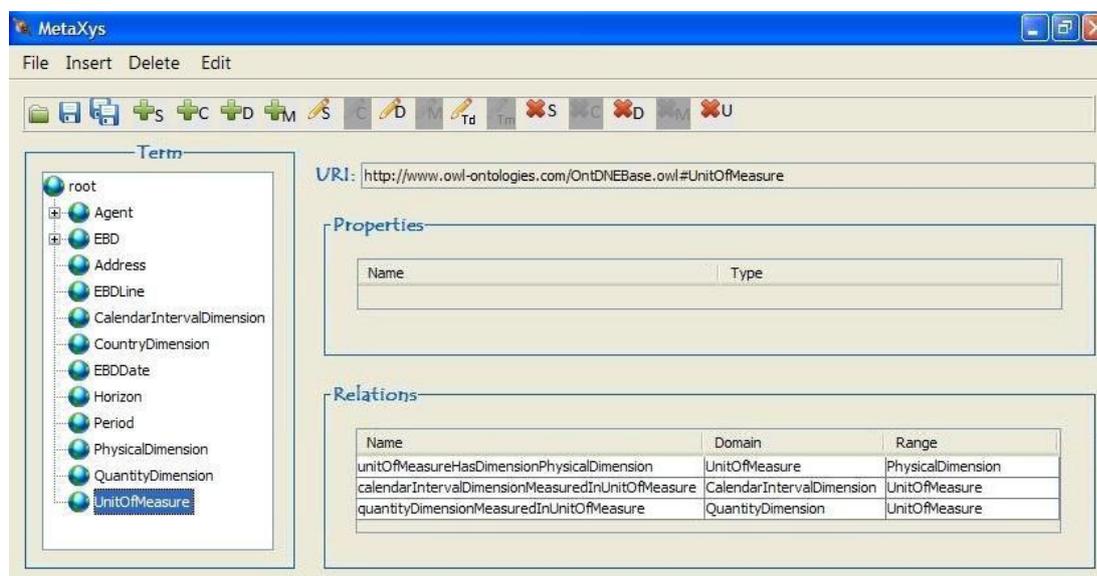


Figura 5.23: MetaXys: Representación de la unidad de medida

Realizando las mismas decisiones de diseño se llega a la representación de la característica precio, que se muestra de manera parcial en la Figura 5.24.

En este caso se eliminó la propiedad `price` y en su lugar se creó el término `PriceDimension` para representar la dimensión de cualidad asociada, cuya unidad de medida podría ser dólares estadounidenses, pesos argentinos, euros, etc., correspondiendo a la dimensión física “moneda”.

En cambio, producto es una característica compleja. Según los principios de diseño definidos en esta tesis, esta característica se debe representar por los siguientes elementos: (a) un término que represente el conjunto de dimensiones integrales; (b) una relación entre este término y el término que representa a la entidad caracterizada; (c) para cada dimensión integral, un término que la represente y una relación entre este término y el término que representa el

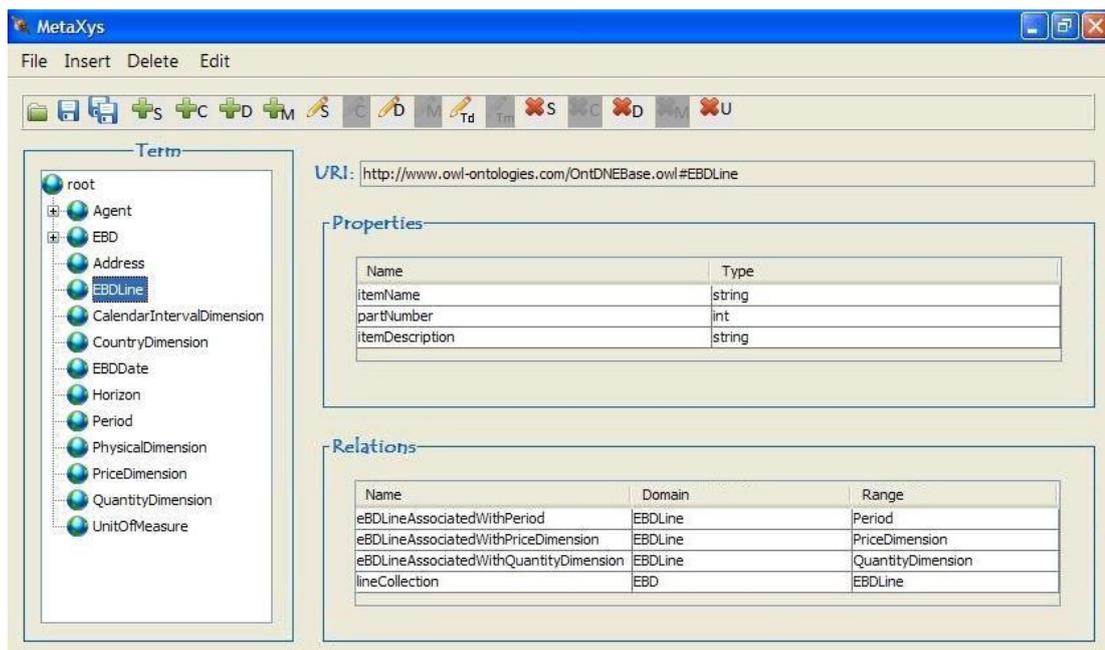


Figura 5.24: MetaXys: Nueva representación de la entidad ítem

conjunto de dimensiones integrales; (d) para cada término que representa una dimensión integral medible debe estar: la relación con el término que representa la unidad de medida, el término que representa la unidad de medida de la dimensión, un término que representa la dimensión física de la unidad de medida y una relación entre este término y el término que representa la unidad de medida. Pero además, producto necesita su propia representación como entidad. En la ontología DNE Base (Figura 5.9, página 177) algunas propiedades pueden referir a un producto (`partnumber` o `itemDescription`, por ejemplo), pero ninguna lo representa como entidad. La herramienta MetaXys permite el enriquecimiento de producto y sus características, agregando los elementos necesarios para su adecuada representación, con la opción 'Insert Complex Quality Universal' (+C). La Figura 5.25 muestra la ventana que se abre al seleccionarla.

En la parte superior de la ventana se debe ingresar el nombre de la cualidad compleja universal a crear, **Product** en este caso, y seleccionar el término de la ontología DNE Base con el cuál se lo va a relacionar: **EBDLine**. También se debe elegir un nombre para la multidimensión de la cualidad compleja (**ProductMultiDimension**) y agregar tantas dimensiones integrales como requiera la multidimensión. Esto último se logra pulsando el botón 'Add Dimension' en la parte inferior de la ventana. En este caso las dimensiones integrales son tres:

**Insert Complex Quality Universal**

Information ComplexQualityUniversal

ID: Product

Select the term with which to associate the Complex Quality Universal:

EBDLine

Information MultiDimension

ID: ProductMultiDimension

Dimensions:

Dimension	Unit Of Measure

Add Dimension Delete Dimension

Accept Cancel

Figura 5.25: MetaXys: Enriquecimiento de la característica producto

TrademarkDimension, TypeDimension y CapacityDimension. Las dos primeras son una enumeración de valores posibles y la tercera es una característica simple medible (Figura 5.26). Al finalizar estas acciones, la ventana de inserción de una cualidad compleja universal queda como se muestra en la Figura 5.27.

Al pulsar el botón '**Accept**' en la ventana de la Figura 5.27, la herramienta permite que se borren tantas propiedades asociadas al término que representa la entidad caracterizada como dimensiones integrales formen la multidimensión. Con este objetivo presenta una ventana como la que se muestra en la Figura 5.28.

En el lado izquierdo de la ventana de la Figura 5.28 se observan todas las propiedades asociadas al término que representa la entidad caracterizada (EBDLine en este caso). Al hacer clic en alguna de ellas, automáticamente la propiedad seleccionada se elimina del lado izquierdo y pasa a visualizarse en

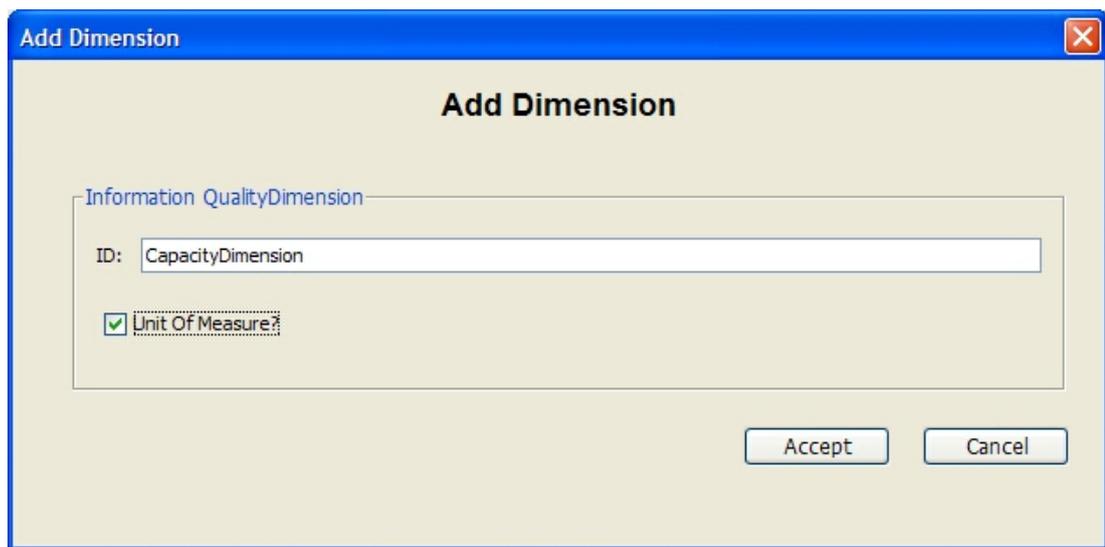


Figura 5.26: MetaXys: Enriquecimiento de la característica capacidad

el lado derecho. Al pulsar el botón '**Accept**' se eliminan todas las propiedades listadas en el lado derecho de la ventana.

Según los resultados de la verificación, la característica producto no tiene una propiedad que la represente como entidad en la ontología DNE Base, sólo algunas propiedades que refieren a características de un producto (`partNumber` e `itemDescription`, asociadas al término `EBDLine`). Al crearse un término que represente a la entidad producto, estas características deberían asociarse a él y eliminarse su relación con el término `EBDLine`. Esto hace que las propiedades `partNumber` e `itemDescription` no sean elegibles para su eliminación, sólo se necesita modificar sus dominios, lo cual no se puede realizar con esta primera versión de la herramienta MetaXys. En cambio, el resultado de la verificación indicó que la propiedad `itemName`, asociada al término `EBDLine`, no era de utilidad y debía eliminarse. Se selecciona entonces dicha propiedad en la ventana de la Figura 5.28 y se pulsa el botón '**Accept**' para obtener la creación de todos los elementos necesarios para la representación de producto (la Figura 5.29 muestra una representación parcial):

- (a) El término `Product` que representa la característica compleja. Este término es instancia del término `Complex_Quality_Universal` de la meta-ontología implementada en Ecore.
- (b) El término `ProductMultiDimension` que representa el conjunto de

**Insert Complex Quality Universal**

Information ComplexQualityUniversal

ID:

Select the term with which to associate the Complex Quality Universal:

---

Information MultiDimension

ID:

Dimensions:

Dimension	Unit Of Measure
CapacityDimension	Yes
TypeDimension	No
TrademarkDimension	No

Figura 5.27: MetaXys: Enriquecimiento de la característica producto

dimensiones integrales. Este término es instancia del término `Quality_MultiDimension` de la meta-ontología implementada en Ecore.

- (c) La relación `productAssociatedWithProductMultiDimension` entre los términos `Product` y `ProductMultiDimension`. Esta relación es instancia de la relación `associatedWith_MD` de la meta-ontología implementada en Ecore.
- (d) Los términos `TrademarkDimension`, `TypeDimension` y `CapacityDimension` que representan a cada una de las dimensiones integrales. Estos términos son instancias del término `Quality_Dimension` de la meta-ontología implementada en Ecore.
- (e) Las relaciones `productMultiDimensionDefinedByTrademarkDimension`, `productMultiDimensionDefinedByTypeDimension` y

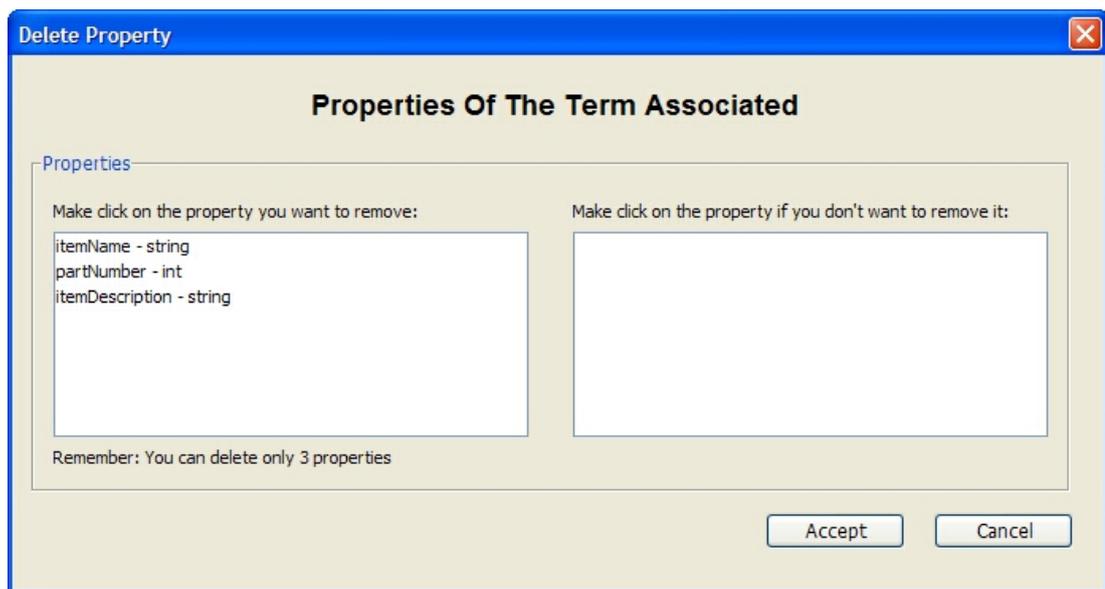


Figura 5.28: MetaXys: Eliminación de propiedades asociadas a la entidad ítem

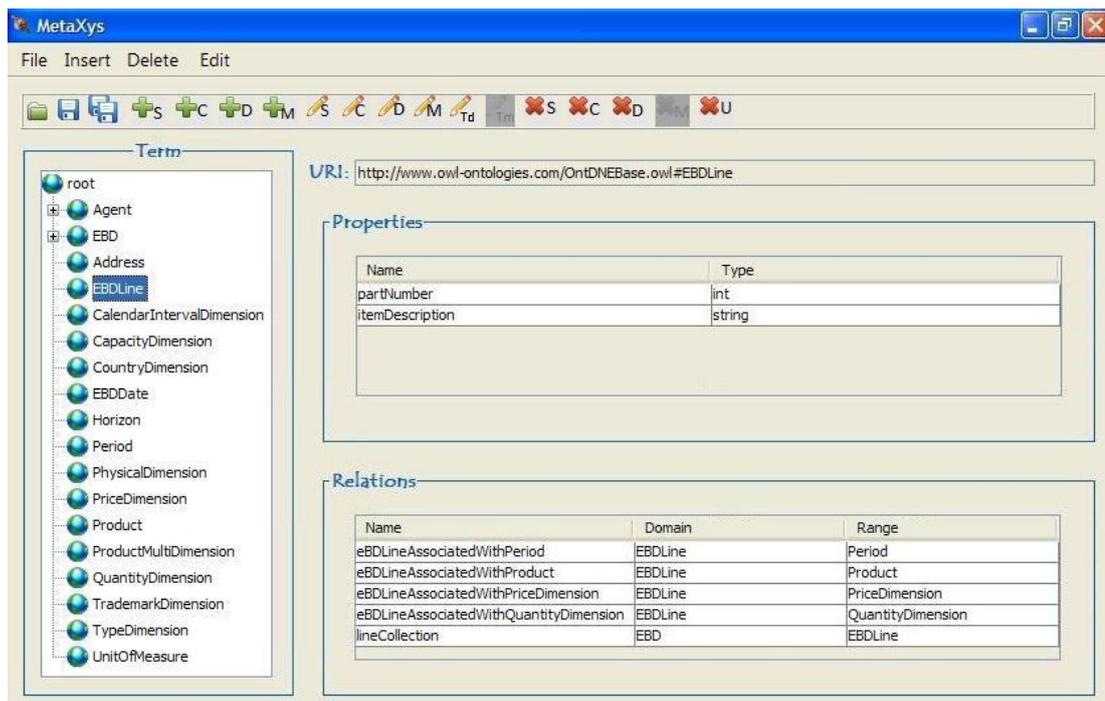


Figura 5.29: MetaXys: Nueva representación de la entidad ítem

productMultiDimensionDefinedByCapacityDimension entre el término ProductMultiDimension, el conjunto de dimensiones integrales, y cada uno de los términos que representan las dimensiones integrales correspondientes (TrademarkDimension, TypeDimension y CapacityDimension). Estas relaciones son instancias de la relación definedBy de la meta-ontología

implementada en Ecore.

- (f) La relación `capacityDimensionMeasuredInUnitOfMeasure` entre los términos `CapacityDimension` y `UnitOfMeasure`. Esta relación es instancia de la relación `measuredIn` de la meta-ontología implementada en Ecore.

El término `UnitOfMeasure` es el mismo que se creó con el enriquecimiento de la característica `horizonte`, que ahora se relaciona con los términos `CapacityDimension`, `PriceDimension`, `PhysicalDimension`, `CalendarIntervalDimension` y `QuantityDimension` (Figura 5.30).

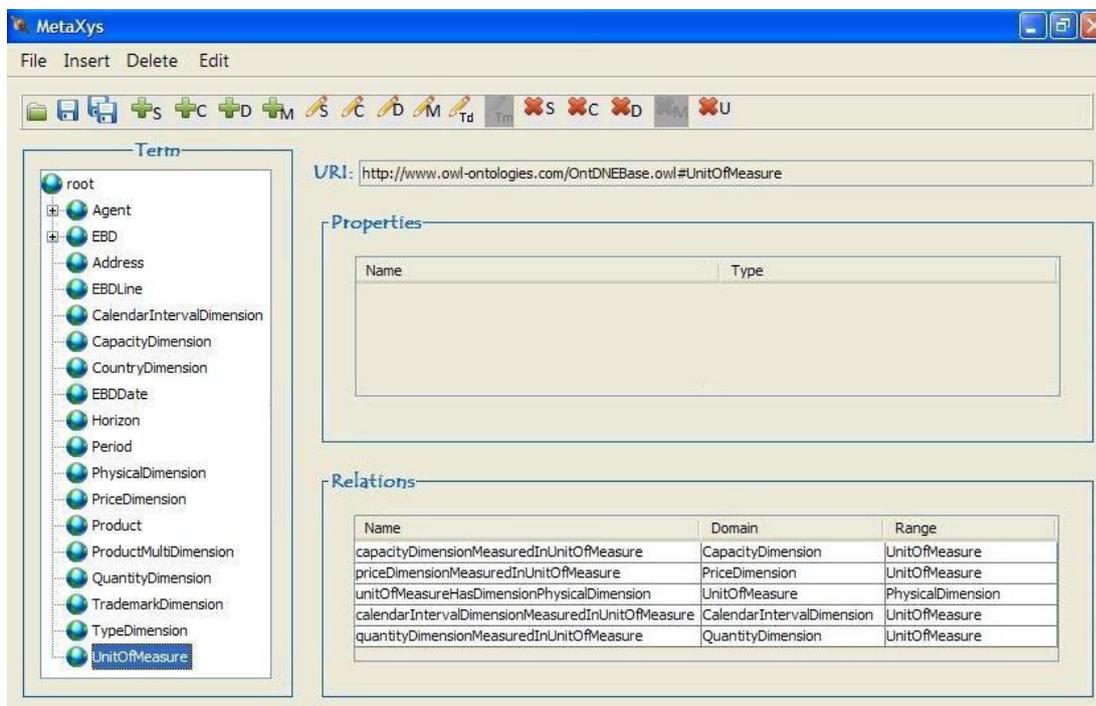


Figura 5.30: MetaXys: Representación de la unidad de medida

De todas las entidades representadas en la ontología, `product` tiene la particularidad de poseer más de un uso previsto: se puede interpretar como los productos de la industria de los envases y también como los envases de los productos de la industria láctea. Siguiendo los principios de diseño definidos en esta tesis, debe haber un término, llamado puente, por cada uso previsto de la entidad. Con el enriquecimiento de la característica `product` se agregó un término `Product` que puede servir a efectos de representar el uso previsto de la entidad en el contexto de la industria de los envases. Sin embargo, la ontología carece de un término que pueda representar el uso previsto de la entidad en

el contexto de la industria láctea, por ejemplo, **Packing**. Así como producto se considera una característica de la entidad ítem, envase también puede serlo, pero debido al uso que se le da en el contexto de la industria láctea, envase es una característica simple y no compleja como producto. La herramienta MetaXys permite el agregado de características simples como en este caso, con la opción ‘**Insert Simple Quality Universal**’ (✚), con la cual se abre la ventana que se muestra en la Figura 5.31.

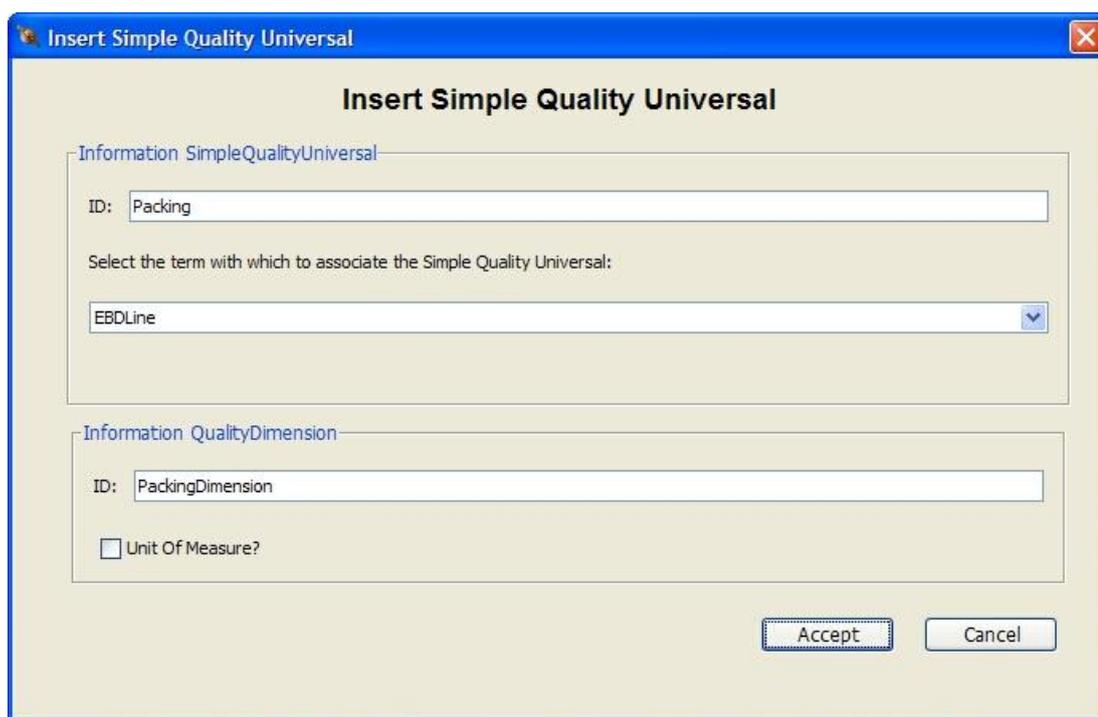


Figura 5.31: MetaXys: Otro uso previsto de la entidad producto: Envase

Al igual que en el caso de la cualidad compleja, esta ventana solicita el ingreso del nombre de la cualidad simple a crear (**Packing**) y el nombre del término de la ontología DNE Base con el cual se lo va a relacionar. En este caso en particular, el término **Product** no pertenece a la ontología DNE Base original, sino que se lo creó con la herramienta MetaXys al enriquecer la característica producto. Una limitación de la primera versión de la herramienta es la de no permitir establecer relaciones entre términos que derivan de la meta-ontología, más allá de las previstas. Esto impide definir la relación **usedAs** entre los términos **Product** y **Packing** como se hizo en el Capítulo 3. Como envase también se considera una característica de la entidad ítem, se establece entonces la relación con el término que la representa (**EBDLine**).

En el contexto de la industria láctea, los envases están asociados a una dimensión que es una enumeración de valores posibles, y por lo tanto no tienen unidad de medida asociada. Esto se indica en la parte inferior de la ventana de la Figura 5.31, dejando sin seleccionar la casilla de verificación correspondiente.

## 5.5. Conclusiones

En este capítulo se presentó un lenguaje para enriquecer la representación de las características de entidades en una ontología. La propuesta se basa en la idea de hacer explícitas aquellas características cuya interpretación depende del contexto en el que se las considere.

El lenguaje propuesto consta de una sintaxis abstracta y una sintaxis concreta, implementada en la primera versión de la herramienta MetaXys.

Como se vio en la aplicación del lenguaje al Caso de Estudio 1 (Sección 5.4), gracias a la meta-ontología (sintaxis abstracta) implementada en la herramienta es posible modelar características simples, medibles y no medibles, y características complejas de las entidades de forma automática y sin omitir ninguno de los elementos requeridos para una adecuada representación. También se pueden eliminar elementos correspondientes a las antiguas representaciones de las entidades y sus características, permitiendo así realizar una depuración de elementos en la ontología enriquecida resultante, pero manteniendo al mismo tiempo consistentes dichas representaciones. Además, se realizan todas las validaciones necesarias para cada tipo de característica. Por ejemplo, con este lenguaje no se puede crear una representación donde una multidimensión de cualidad esté definida por una única dimensión, o donde una cualidad simple universal esté asociada a una multidimensión.

Una ontología tal como la enriquecida es una representación más precisa que la que se puede obtener creándola directamente en OWL, por ejemplo. Esto se debe a que la herramienta MetaXys usada en su creación adhiere a una meta-ontología mucho más rica que la subyacente a OWL. Por el contrario, OWL se diseñó para mantener propiedades computacionales como tratabilidad y decidibilidad, que en general están ausentes en lenguajes más expresivos.

Para subsanar esta brecha entre un lenguaje adecuado para la representación de características contextuales y un lenguaje computacionalmente aceptable,

la herramienta MetaXys permite almacenar la ontología en el formato `owl` y mantiene información necesaria para poder recuperar decisiones de diseño hechas con la herramienta en la propiedad `<rdfs:comment>`, una propiedad `annotation` predefinida por OWL. De esta forma se logra disponer de un lenguaje apropiado para el modelado, pero manteniendo al mismo tiempo la posibilidad de implementar el modelo (ontología) resultante en un lenguaje ampliamente difundido en la comunidad ontológica y que es una recomendación de la W3C. La posibilidad de almacenar la ontología enriquecida en el formato `owl` permite, además, que la misma sea editada por cualquier otro editor de ontologías que reconozca dicho formato y que supla los requerimientos de edición que, por las limitaciones de esta primera versión, no se pueden realizar con la herramienta MetaXys.



## Conclusiones y Trabajos Futuros

El objetivo de este capítulo es presentar las conclusiones de la tesis, resaltando las principales contribuciones realizadas (Sección 6.1) y los trabajos futuros a ser encarados (Sección 6.2).

### 6.1. Principales Contribuciones

Este trabajo de investigación parte de la observación que el uso de ontologías en el intercambio e integración de información no garantiza por sí solo que los sistemas de información interpreten correctamente la información transmitida. En muchos sistemas distribuidos y abiertos (por ejemplo, la Web Semántica, la colaboración entre agentes de software autónomos y diseñados independientemente, la colaboración vía Internet entre empresas) es impensable utilizar una única ontología global para establecer la comunicación. Generalmente, cada una de las partes involucradas cuenta con sus propias ontologías, de modo que se hace necesario establecer las correspondencias entre las entidades representadas en dichas ontologías, es decir, realizar un *matching* entre las mismas, para que la comunicación entre los distintos sistemas sea posible.

Una de las razones de la dificultad de las tareas de *matching* es que las ontologías se diseñan con ciertos conocimientos implícitos y en un contexto determinado, que no son parte de la especificación de las ontologías y, por lo tanto, no están disponibles para que los sistemas de *matching* puedan aprovecharlos. Debido a esto, los resultados obtenidos por distintas propuestas de solución al problema de *matching* dependen de la calidad de la representación de las entidades

en las correspondientes ontologías.

Cuando se utilizan ontologías con esta problemática para establecer una comunicación entre partes que pertenecen a contextos diferentes, las posibilidades de incurrir en malas interpretaciones aumentan, sobre todo porque las técnicas de *matching* aplicadas no pueden descubrir correspondencias sobre entidades o características que no se han representado explícitamente.

Para atacar este problema, desde dentro de los sistemas de *matching*, se han usado estrategias como reutilizar resultados de *matching* previos, utilizar corpus específicos de dominio u ontologías específicas de dominio, entre otras. También se han realizado propuestas para mejorar la calidad de las ontologías que se someten a un proceso de *matching* y en este sentido esta tesis hace su aporte. Aún así, los sistemas de *matching* tienen problemas para encontrar *alignments* correctos y por tanto es un área en la que todavía hay mucho por investigar y mejorar.

En esta tesis se propuso un método sistemático para enriquecer la representación de las entidades en una ontología, explicitando sus características contextuales, de modo que esta ontología permita el intercambio de información en sistemas abiertos donde participan contextos diferentes.

Con el propósito de dar soporte al proceso de Enriquecimiento en la representación de las entidades, en esta tesis se desarrollaron dos estrategias complementarias: el modelado de características contextuales como características simples y complejas, y el reuso de ontologías ampliamente aceptadas por la comunidad ontológica o que se sustentan en estándares.

A continuación se discuten las contribuciones en más detalle.

### **6.1.1. Método para enriquecer la representación de las entidades en una ontología**

Existen varias metodologías y métodos que se han propuesto para desarrollar ontologías, cumpliendo con los requerimientos de diversos campos de aplicación específicos. Si bien estas metodologías proporcionan una guía para identificar las entidades y relaciones relevantes en el área de interés, las características propias del área de aplicación (contexto) suelen quedar implícitas en la ontología.

El método propuesto se basa en la idea de hacer explícitas las características

contextuales de las entidades para reducir las posibles malas interpretaciones que puedan hacer los sistemas que deben tratarlas. Este método es particularmente útil para hacer frente al problema de la heterogeneidad semiótica, ya que mediante el uso de términos puente se pueden relacionar diferentes significados y representaciones de una misma entidad de dominio en distintos contextos.

Teniendo en cuenta que, en general, la información que se intercambia se origina en fuentes de datos estructuradas y semiestructuradas, el método propuesto promueve el uso de técnicas de aprendizaje de ontologías para un rápido desarrollo de una ontología base, la cual se puede enriquecer luego manualmente. Así mismo, el método impulsa el reuso de ontologías existentes, ampliamente aceptadas por la comunidad ontológica o que se sustentan en estándares.

Cuando el reuso de ontologías no es posible, el método propone principios de diseño para representar entidades, usos previstos de las entidades, relaciones entre entidades, características simples medibles y no medibles, características complejas y características comunes. Con el propósito de evaluar estos principios de diseño, en esta tesis se presentaron un conjunto de indicadores que miden el cumplimiento de los mismos.

Las ontologías obtenidas a partir del método propuesto proveen una entrada más rica a los sistemas de *matching*, aportando los elementos requeridos por los mismos para encontrar las correspondencias adecuadas.

La efectividad del método propuesto se probó aplicándolo a casos de estudio para proveer una validación empírica.

### 6.1.2. Framework para dar soporte al reuso de ontologías

Para poder modelar características contextuales se requiere, como primera medida, identificar los contextos involucrados y sus características. Es así que en esta tesis se propuso un *framework* general que utiliza ontologías para describir las características de contextos simples y complejos. Luego, una entidad se puede representar haciendo uso de este *framework* relacionándola a los contextos en los que deba ser interpretada, sin imponer un significado global de la misma.

Además, este *framework* potencia el reuso de ontologías debido a que:

- La creación de las ontologías que describen características de contextos complejos se realiza reutilizando ontologías que describen características de

contextos simples.

- Las ontologías que representan características de contextos se pueden utilizar para enriquecer la representación de todas las entidades que correspondan ser interpretadas en dichos contextos.
- Las ontologías que representan la estructura sintáctica de fuentes de información se pueden relacionar con diferentes ontologías que representan las características de distintos contextos.

La utilidad del *framework* se ilustró con su aplicación a dos casos de estudio.

### 6.1.3. Lenguaje de representación de características contextuales

Cuando no es posible utilizar ontologías existentes para la representación de las características de las entidades es preciso contar con un lenguaje de modelado. Dado que los lenguajes de modelado existentes no son adecuados para dar soporte en la representación de las características contextuales de las entidades, en esta tesis se propuso un lenguaje de modelado de características general que se pueda utilizar a tal fin.

Para la sintaxis abstracta del lenguaje, se propuso una meta-ontología que adopta a la ontología fundacional UFO como referencia con el objetivo de que las ontologías creadas con este lenguaje se asemejen lo más posible a la realidad.

Para la sintaxis concreta y notación, se desarrolló un prototipo de una herramienta, denominada MetaXys, que brinda soporte en la representación de características simples y complejas de las entidades, agregando automáticamente todos los elementos requeridos para tal representación.

La herramienta MetaXys permite almacenar la ontología en el formato `owl`, con lo cual se posee un lenguaje adecuado para el modelado de las características y, al mismo tiempo, una ontología implementada en un lenguaje computacionalmente aceptable que, además, puede ser editada por otros editores que reconozcan el formato `owl`.

Para mostrar la utilidad del lenguaje propuesto, éste se aplicó en el proceso de Enriquecimiento de la ontología base, desarrollada a partir de una técnica de aprendizaje de ontologías.

## 6.2. Trabajos Futuros

A partir de la presente tesis, los trabajos futuros de investigación se enmarcan en los puntos que se detallan a continuación:

- Mejorar el proceso de Enriquecimiento del método propuesto mediante la aplicación de patrones de diseño de ontologías, dado que existen tipos de problemas en el diseño que se pueden resolver mediante la aplicación de soluciones comunes. Esto podría mejorar la eficiencia y efectividad del proceso como así también la calidad de la ontología obtenida.
- Profundizar el estudio de la representación más adecuada de los roles que asume una entidad física.
- Avanzar en el estudio e identificación de ontologías que puedan ser utilizadas para describir otros contextos simples, además del temporal, espacial, funcional y de procedimientos considerados en esta tesis.
- Avanzar en la implementación de la herramienta MetaXys incorporando los siguientes aspectos: creación de una ontología desde cero; posibilidad de importar ontologías; manejo de ontologías instanciadas, dominios y rangos de las relaciones especificados por múltiples términos; creación de términos con operaciones de conjunto (unión, intersección y complemento) y cuantificadores; visualización de todos los elementos que conforman una ontología; manejo de axiomas en lógica descriptiva y gestión de contextos.
- Estudiar alternativas que faciliten las tareas de mantenimiento de las ontologías que conforman el *framework*.
- Habiendo reconocido la importancia de contar con herramientas que den soporte en la decisión de cuáles ontologías utilizar en el proceso de Enriquecimiento, queda por analizar el efecto que ciertas características producen sobre la calidad de las ontologías según el propósito para el cual se las requiere.



## Representación Gráfica de Ontologías

La Figura A.1 muestra la representación gráfica de ontologías que se utiliza en esta tesis. El significado de los elementos gráficos es el siguiente:

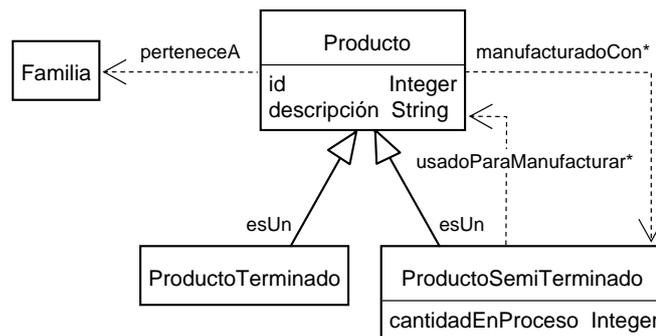


Figura A.1: Representación gráfica de ontologías

- Las cajas representan términos. Un término tiene un nombre, que se representa por una palabra o grupo de palabras en la parte superior central de la caja, por ejemplo **Producto**.
- Las filas dentro de una caja representan las propiedades de un término. A cada propiedad le corresponde una fila (comenzando en la segunda fila), por ejemplo **id**.
- Las flechas representan relaciones entre términos. Los nombres de las relaciones se representan por una palabra o grupo de palabras al lado de la flecha. Las flechas con líneas llenas y punta en forma de triángulo relleno blanco corresponden a relaciones jerárquicas (**esUn**; la flecha apunta hacia el

término “padre”); las flechas con líneas punteadas corresponden a relaciones particulares, por ejemplo **perteneceA** (la flecha apunta hacia el destino de la relación).

## Ontología DNE Base Formalizada en OWL

En este apéndice se muestra la ontología DNE Base, formalizada en el lenguaje OWL. Los elementos específicos que agrega la herramienta utilizada para la formalización se eliminaron del código.

```
<?xml version="1.0"?> <rdf:RDF
  xmlns="http://www.owl-ontologies.com/OntDNEBase.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.owl-ontologies.com/OntDNEBase.owl">

  <owl:Ontology rdf:about=""/>

  <owl:Class rdf:ID="Organization">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Agent"/>
    </rdfs:subClassOf>

    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:allValuesFrom>
          <owl:Class rdf:ID="Personnel"/>
        </owl:allValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID="contactPerson"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
```

```

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:ObjectProperty rdf:about="#contactPerson"/>
    </owl:onProperty>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:ID="EBDLine">
  <rdfs:subClassOf
    rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="itemName"/>
      </owl:onProperty>
      <owl:allValuesFrom
        rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
      </owl:Restriction>
    </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="partNumber"/>
      </owl:onProperty>
      <owl:allValuesFrom
        rdf:resource="http://www.w3.org/2001/XMLSchema#integer"/>
      </owl:Restriction>
    </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:cardinality
        rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >1</owl:cardinality>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="period"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

```

```
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</owl:cardinality>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#itemName"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#integer"/>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:ID="quantity"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#period"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:ID="itemDescription"/>
    </owl:onProperty>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</owl:cardinality>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#itemDescription"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>
```

```

        </owl:onProperty>
    </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
    <owl:Restriction>
        <owl:cardinality
            rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
            >1</owl:cardinality>
        <owl:onProperty>
            <owl:DatatypeProperty rdf:about="#partNumber"/>
        </owl:onProperty>
    </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
    <owl:Restriction>
        <owl:cardinality
            rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
            >1</owl:cardinality>
        <owl:onProperty>
            <owl:DatatypeProperty rdf:about="#quantity"/>
        </owl:onProperty>
    </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
    <owl:Restriction>
        <owl:allValuesFrom
            rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
        <owl:onProperty>
            <owl:DatatypeProperty rdf:ID="price"/>
        </owl:onProperty>
    </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
    <owl:Restriction>
        <owl:onProperty>
            <owl:DatatypeProperty rdf:about="#price"/>
        </owl:onProperty>
        <owl:cardinality
            rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
            >1</owl:cardinality>
    </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>

```

```
<owl:Class rdf:ID="Address">
  <rdfs:subClassOf
    rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="city"/>
      </owl:onProperty>
      <owl:cardinality
        rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:allValuesFrom
        rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:about="#city"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:allValuesFrom
        rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="state"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="country"/>
      </owl:onProperty>
      <owl:allValuesFrom
        rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
```

```
    <owl:DatatypeProperty rdf:ID="street"/>
  </owl:onProperty>
  <owl:allValuesFrom
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#integer"/>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:ID="zip"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:cardinality>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#street"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#state"/>
    </owl:onProperty>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:cardinality>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#country"/>
    </owl:onProperty>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:cardinality>
  </owl:Restriction>
</rdfs:subClassOf>
```

```
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:cardinality>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#zip"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:ID="ReplenishmentProgram">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="EBD"/>
  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:about="#EBD">
  <rdfs:subClassOf
    rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:allValuesFrom rdf:resource="#EBDLine"/>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="lineCollection"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="customer"/>
      </owl:onProperty>
      <owl:allValuesFrom rdf:resource="#Organization"/>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:about="#customer"/>
      </owl:onProperty>
      <owl:cardinality
```

```

    rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</owl:cardinality>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:ID="ebdDate"/>
    </owl:onProperty>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:cardinality>
    </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:ID="horizon"/>
    </owl:onProperty>
    <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:ObjectProperty rdf:ID="supplier"/>
    </owl:onProperty>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:cardinality>
    </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:cardinality>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#horizon"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>

```

```
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:allValuesFrom rdf:resource="#Organization"/>
    <owl:onProperty>
      <owl:ObjectProperty rdf:about="#supplier"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</owl:cardinality>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:ID="ebdNumber"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#ebdNumber"/>
    </owl:onProperty>
    <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#integer"/>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:ID="ebdName"/>
    </owl:onProperty>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</owl:cardinality>
  </owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#ebdDate"/>
    </owl:onProperty>
    <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:Restriction>
</rdfs:subClassOf>
```

```

    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:allValuesFrom
        rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:about="#ebdName"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:minCardinality
        rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >1</owl:minCardinality>
      <owl:onProperty>
        <owl:ObjectProperty rdf:about="#lineCollection"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:about="#Personnel">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Agent"/>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:allValuesFrom
        rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="position"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:cardinality
        rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >1</owl:cardinality>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:about="#position"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

```

```
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:about="#Agent">
  <rdfs:subClassOf
  rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:minCardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:minCardinality>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="email"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="phone"/>
      </owl:onProperty>
      <owl:minCardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:minCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="mailingAddress"/>
      </owl:onProperty>
      <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:about="#phone"/>
      </owl:onProperty>
      <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

```
</owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:ID="name"/>
    </owl:onProperty>
    <owl:cardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:ID="fax"/>
    </owl:onProperty>
    <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:Restriction>
  </rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#name"/>
    </owl:onProperty>
    <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:Restriction>
  </rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:ObjectProperty rdf:about="#mailingAddress"/>
    </owl:onProperty>
    <owl:allValuesFrom rdf:resource="#Address"/>
    </owl:Restriction>
  </rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:minCardinality
      rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:minCardinality>
```

```
<owl:onProperty>
  <owl:DatatypeProperty rdf:about="#fax"/>
</owl:onProperty>
</owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:allValuesFrom
      rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <owl:onProperty>
      <owl:DatatypeProperty rdf:about="#email"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>

<owl:ObjectProperty rdf:about="#customer">
  <rdfs:domain rdf:resource="#EBD"/>
  <rdfs:range rdf:resource="#Organization"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:about="#mailingAddress">
  <rdfs:domain rdf:resource="#Agent"/>
  <rdfs:range rdf:resource="#Address"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:about="#contactPerson">
  <rdfs:range rdf:resource="#Personnel"/>
  <rdfs:domain rdf:resource="#Organization"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:about="#lineCollection">
  <rdfs:domain rdf:resource="#EBD"/>
  <rdfs:range rdf:resource="#EBDLine"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:about="#supplier">
  <rdfs:range rdf:resource="#Organization"/>
  <rdfs:domain rdf:resource="#EBD"/>
</owl:ObjectProperty>

<owl:DatatypeProperty rdf:about="#zip">
  <rdfs:domain rdf:resource="#Address"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:about="#city">
  <rdfs:domain rdf:resource="#Address"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:about="#quantity">
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
  <rdfs:domain rdf:resource="#EBDLine"/>
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:about="#name">
  <rdfs:domain rdf:resource="#Agent"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:about="#email">
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#Agent"/>
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:about="#fax">
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#Agent"/>
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:about="#street">
  <rdfs:domain rdf:resource="#Address"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:about="#state">
  <rdfs:domain rdf:resource="#Address"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:about="#horizon">
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#EBD"/>
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:about="#phone">
  <rdfs:domain rdf:resource="#Agent"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="#partNumber">
  <rdfs:domain rdf:resource="#EBDLine"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="#itemDescription">
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#EBDLine"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="#position">
  <rdfs:domain rdf:resource="#Personnel"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="#country">
  <rdfs:domain rdf:resource="#Address"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="#ebdName">
  <rdfs:domain rdf:resource="#EBD"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="#price">
  <rdfs:domain rdf:resource="#EBDLine"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="#period">
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#EBDLine"/>
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:about="#ebdNumber">
  <rdfs:domain rdf:resource="#EBD"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="#ebdDate">
  <rdfs:domain rdf:resource="#EBD"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="#itemName">
  <rdfs:domain rdf:resource="#EBDLine"/>
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
</rdf:RDF>
```

## Detalle de Cálculo de Medidas para la Evaluación Semántica

En este apéndice se muestran detalles de los cálculos realizados para la evaluación de las ontologías DNE Base y DNE Contextualizada en la dimensión semántica.

La Tabla C.1 muestra las palabras y frases utilizadas en la definición de términos y propiedades de la ontología DNE Base. También se indica si dichas palabras o frases tienen una entrada en Wordnet y, en ese caso, con cuántos sentidos distintos. En gris se resalta la única palabra no encontrada en WordNet (“EBD”). Al final de la tabla están los resultados parciales y finales de los indicadores de interpretabilidad y claridad para la ontología DNE Base.

Las Tablas C.2, C.3, C.4 y C.5 muestran las palabras y frases utilizadas en la definición de términos y propiedades de la ontología DNE Contextualizada y las ontologías importadas por ésta: *OWL-Time* y *Country*. También se indica si dichas palabras o frases tienen una entrada en Wordnet y, en ese caso, con cuántos sentidos distintos. En gris se resaltan las palabras no encontradas en WordNet. Al finalizar el listado de palabras correspondientes a la ontología DNE Contextualizada se muestra el cálculo parcial de los indicadores de interpretabilidad y claridad para esta ontología, sin considerar las palabras de las ontologías *OWL-Time* y *Country* importadas (Tabla C.3). En la última tabla (C.5) se muestran los resultados parciales y finales de los indicadores de interpretabilidad y claridad para la ontología DNE Contextualizada completa (con *OWL-Time* y *Country* incluidas).

Las Tablas C.6, C.7, C.8, C.9 y C.10 muestran los detalles del cálculo para

<b>Ontología DNE Base</b>			
<b>Término (N)</b>	<b>Palabra</b>	<b>Está en WordNet (1: Sí – 0: No)</b>	<b>A</b>
Address	ADDRESS	1	8
Agent	AGENT	1	6
<b>EBD</b>	<b>EBD</b>	<b>0</b>	<b>0</b>
EBDLine	LINE	1	0
Organization	ORGANIZATION	1	7
Personnel	PERSONNEL	1	2
ReplenishmentProgram	REPLENISHMENT PROGRAM	1 1	0
<b>Propiedad (N)</b>			
city	CITY	1	3
country	COUNTRY	1	5
ebdDate	DATE	1	0
ebdName	NAME	1	0
ebdNumber	NUMBER	1	0
email	EMAIL	1	1
fax	FAX	1	1
horizon	HORIZON	1	4
itemDescription	DESCRIPTION	1	0
itemName			0
name			6
partNumber	PART	1	0
period	PERIOD	1	8
price	PRICE	1	7
phone	PHONE	1	3
position	POSITION	1	16
quantity	QUANTITY	1	3
state	STATE	1	8
street	STREET	1	5
zip	ZIP	1	2
<b>PTP</b>	<b>26</b>		
<b>PS</b>		<b>25</b>	
<b>Interpretabilidad (I)</b>		<b>0,9615</b>	
<b>TN</b>	<b>27</b>		
<b>Total de sentidos en WordNet</b>			<b>95</b>
<b>Claridad (C)</b>			<b>3,5185</b>

Tabla C.1: Detalles del cálculo de la interpretabilidad y claridad de la ontología DNE Base

los indicadores propuestos para la evaluación de los principios seguidos en esta tesis para el modelado de entidades, sus relaciones y características.

<b>Ontología DNE Contextualizada</b>			
<b>Término (N)</b>	<b>Palabra</b>	<b>Está en WordNet (1: Sí – 0: No)</b>	<b>A</b>
Address	ADDRESS	1	8
Agent	AGENT	1	6
CapacityDimension			0
	CAPACITY	1	
	DIMENSION	1	
EBD	EBD	0	0
EBDate			0
	DATE	1	
EBDLine			0
	LINE	1	
Horizon	HORIZON	1	4
Organization	ORGANIZATION	1	7
Packing	PACKING	1	3
PackingDimension			0
Period	PERIOD	1	8
Personnel	PERSONNEL	1	2
PhysicalDimension			0
	PHYSICAL	1	
PriceDimension			0
	PRICE	1	
Product	PRODUCT	1	6
ProductMultiDimension			0
	MULTIDIMENSIONAL	1	
QuantityDimension			0
	QUANTITY	1	
ReplenishmentProgram			0
	REPLENISHMENT	1	
	PROGRAM	1	
TrademarkDimension			0
	TRADEMARK	1	
TypeDimension			0
	TYPE	1	
UnitOfMeasure			0
	UNIT	1	
	MEASURE	1	
<b>Propiedad (N)</b>			
city	CITY	1	3
ebdName			0
	NAME	1	
ebdNumber			0
	NUMBER	1	
email	EMAIL	1	1
fax	FAX	1	1
itemDescription			0
	DESCRIPTION	1	
name			6
partNumber			0
	PART	1	
phone	PHONE	1	3
position	POSITION	1	16
state	STATE	1	8

Tabla C.2: Detalles del cálculo de la interpretabilidad y claridad de la ontología DNE Contextualizada (hoja 1)

<b>Ontología DNE Contextualizada</b>			
<b>Propiedad (N)</b>	<b>Palabra</b>	<b>Está en WordNet (1: Sí – 0: No)</b>	<b>A</b>
street	STREET	1	5
zip	ZIP	1	2
<b>PTP</b>	<b>35</b>		
<b>PS</b>		<b>34</b>	
<b>Interpretabilidad (I)</b>		<b>0,9714</b>	
<b>TN</b>	<b>34</b>		
<b>Total de sentidos en WordNet</b>			<b>89</b>
<b>Claridad (C)</b>			<b>2,6176</b>

<b>Ontología OWL-Time</b>			
<b>Término (N)</b>	<b>Palabra</b>		
time:CalendarClockDescription	CALENDAR	1	0
	CLOCK	1	
time:DurationDescription	DURATION	1	0
time:InstantThingPair	INSTANT	1	0
	THING	1	
	PAIR	1	
time:TemporalThing	TEMPORAL	1	0
time:Event	EVENT	1	
time:InstantEvent			0
time:IntervalEvent	INTERVAL	1	0
time:InstantThing			0
time:Instant			2
time:IntervalThing			0
time:Interval			4
time:ProperInterval	PROPER	1	0
time:CalendarClockInterval			0
time:ProperIntervalThing			0
time:TemporalEntity	ENTITY	1	0
time:TemporalUnit	UNIT	1	0
tzont:TimeZone	TIME	1	0
	ZONE	1	

<b>Propiedad (N)</b>			
time:year	YEAR	1	4
time:second	SECOND	1	10
time:month	MONTH	1	2
time:week	WEEK	1	3
time:minute	MINUTE	1	6
time:hour	HOUR	1	4
time:dayOfWeekField	DAY	1	0
	FIELD	1	

Tabla C.3: Detalles del cálculo de la interpretabilidad y claridad de la ontología DNE Contextualizada (hoja 2)

<b>Ontología DNE Contextualizada</b>			
<b>Ontología OWL-Time Propiedad (N)</b>	<b>Palabra</b>	<b>Está en WordNet (1: Sí – 0: No)</b>	<b>A</b>
time:day			10
time:dayOfYearField			0
time:years	YEARS	1	3
time:months	MONTHS	0	0
time:weeks	WEEKS	0	0
time:days	DAYS	1	1
time:hours	HOURS	1	2
time:minutes	MINUTES	1	1
time:seconds	SECONDS	0	0
time:instantPairDurationDescriptionDataType			0
	DATA	1	
	TYPE	1	
time:durationDescriptionDataType			0
time:calendarClockDescriptionDataType			0
time:inCalendarClockDataType			0
	IN	1	
<b>Ontología Country</b>			
<b>Término (N)</b>	<b>Palabra</b>		
country:Country	COUNTRY	1	5
country:AssociatedState			0
	ASSOCIATED	1	
country:FullyDependentTerritory			0
	FULLY	1	
	DEPENDENT	1	
	TERRITORY	1	
country:IndependenceDisputedState			0
	INDEPENDENCE	1	
	DISPUTED	1	
country:IndependentState			0
	INDEPENDENT	1	
country:OccupiedTerritory			0
	OCCUPIED	1	
country:UnclaimableTerritory			0
	UNCLAIMABLE	0	
country:VirtuallyAnnexedTerritory			0
	VIRTUALLY	1	
	ANNEXED	1	
country:VirtualTerritory			0
	VIRTUAL	1	
country:ISO3166DefinedCountry			0
	ISO	0	
	DEFINED	1	
<b>Propiedad (N)</b>			
country:nameLocal			0
	LOCAL	1	
country:nameEnglishLong			0
	ENGLISH	1	
	LONG	1	
country:nameLocalLong			0

Tabla C.4: Detalles del cálculo de la interpretabilidad y claridad de la ontología DNE Contextualizada (hoja 3)

<b>Ontología DNE Contextualizada</b>			
<b>Ontología Country Propiedad (N)</b>	<b>Palabra</b>	<b>Está en WordNet (1: Sí – 0: No)</b>	<b>A</b>
country:countryCodeISO3166Alpha2	CODE	1	0
	ALPHA	1	
country:modificationTimestamp	MODIFICATION	1	0
	TIME	1	
	STAMP	1	
country:countryNameISO3166OfficialName	OFFICIAL	1	0
country:countryCodeUNNumeric3	UN	1	0
	NUMERIC	1	
country:revocationEffective	REVOCATION	1	0
	EFFECTIVE	1	
country:nameEnglish			0
country:countryCodeISO3166Alpha3			0
country:deletionTimestamp	DELETION	1	0
country:creationTimestamp	CREATION	1	0
country:effectiveFrom	FROM	0	0
country:countryNameISO3166Short	SHORT	1	0
<b>PTP</b>	<b>99</b>		
<b>PS</b>		<b>92</b>	
<b>Interpretabilidad (I)</b>		<b>0,9293</b>	
<b>TN</b>	<b>95</b>		
<b>Total de sentidos en WordNet</b>			<b>146</b>
<b>Claridad (C)</b>			<b>1,5368</b>

Tabla C.5: Detalles del cálculo de la interpretabilidad y claridad de la ontología DNE Contextualizada (hoja 4)

<b>Entidad</b>	<b>DNE Base</b>		<b>DNE Contextualizada</b>	
	$\omega$	<b>Término</b>	$\omega$	<b>Término</b>
Socio de Negocio				
Proveedor	1	Organization	1	Organization
Cliente	1	Organization	1	Organization
Programa de Provisión	1	ReplenishmentProgram	1	ReplenishmentProgram
Ítem	1	EBDLine	1	EBDLine
$\sum_{i=1,E} \omega_i$	4		4	
E		4		4
PE		1		1

Tabla C.6: Detalles del cálculo del principio 1: Entidades

Uso Previsto Entidad	DNE Base		DNE Contextualizada	
	$\omega$	Término	$\omega$	Término
Socio de Negocio				
Proveedor	1	Organization	1	Organization
Cliente	1	Organization	1	Organization
Programa de Provisión	1	ReplenishmentProgram	1	ReplenishmentProgram
Ítem	1	EBDLine	1	EBDLine
Producto				
Producto	0	---	1	Product
Envase	0	---	1	Packing
$\sum_{i=1,U} \omega_i$	4		6	
U		6		6
PU		0,6667		1

Tabla C.7: Detalles del cálculo del principio 2: Uso previsto de entidades

Relación entre Entidades	DNE Base		DNE Contextualizada	
	$\omega$	Relación	$\omega$	Relación
es cliente en (Socio de Negocio - Programa de Provisión)	0	---	1	isCustomerIn
es proveedor en (Socio de Negocio - Programa de Provisión)	0	---	1	isSupplierIn
provee (Socio de Negocio - Ítem)	0	---	0	---
solicita (Socio de Negocio - Ítem)	0	---	0	---
es enviado por (Programa de Provisión - Socio de Negocio)	1	supplier	1	hasSupplier
es recibido por (Programa de Provisión - Socio de Negocio)	1	customer	1	hasCustomer
contiene (Programa de Provisión - Ítem)	1	lineCollection	1	hasLines
es provisto por (Ítem - Socio de Negocio)	0	---	0	---
es solicitado por (Ítem - Socio de Negocio)	0	---	0	---
esta contenido en (Ítem - Programa de Provisión)	0	---	1	isLineOf
$\sum_{i=1,R} \omega_i$	3		6	
R		10		10
PR		0,3		0,6

Tabla C.8: Detalles del cálculo del principio 3: Relaciones entre entidades

Característica Simple	DNE Base			DNE Contextualizada			
	$\omega$	Propiedad	Relación	Término	$\omega$	Relación	Término
Dirección	1	mailingAddress	Address	Address	1	hasMailingAddress	Address
País	0,5	country	Address	Address	1	inCountry	country:ISO3166DefinedCountry
Marca	0				1	definedByTrademarkDim	TrademarkDimension
Tipo	0				1	definedByTypeDim	TypeDimension
Persona de Contacto	1	contactPerson	Personnel	Personnel	1	hasContactPerson	Personnel
$\sum_{i=1,CS} \omega_i$	2,5				5		
CS					5		
PCS					0,5		
Característica Simple Medible	DNE Base			DNE Contextualizada			
	$\omega$	Propiedad	Relación	Término	$\omega$	Relación	Término
Horizonte	0,5	horizon			0,5	hasHorizon	Horizon
Fecha	0,5	ebsdDate			0,5	time:calendarClockDescriptionOf	time:CalendarClockDescription
Período	0,5	period			0,5	time:calendarClockDescriptionOf	time:CalendarClockDescription
Capacidad	0				0,5	hasPeriod	Period
Cantidad	0,5	quantity			1	time:calendarClockDescriptionOf	time:CalendarClockDescription
Precio	0,5	price			1	definedByCapacityDim	CapacityDimension
$\sum_{i=1,CM} \omega_i$	2,5				6	measuredIn	UnitOfMeasure
CM					6	hasDimension	PhysicalDimension
PCM					0,4167	hasQuantity	QuantityDimension
					0,7500	measuredIn	UnitOfMeasure
					0,7500	hasDimension	PhysicalDimension
Característica Compleja	DNE Base			DNE Contextualizada			
	$\omega$	Propiedad	Relación	Término	$\omega$	Relación	Término
Producto	0				1	hasProductMultiDim	ProductMultiDimension
					1	definedByTrademarkDim	TrademarkDimension
					1	definedByTypeDim	TypeDimension
					1	definedByCapacityDim	CapacityDimension
					1	measuredIn	UnitOfMeasure
					1	hasDimension	PhysicalDimension
$\sum_{i=1,CC} \omega_i$	0				1		
CC					1		
PCC					0		

Tabla C.9: Detalles del cálculo del principio 4: Características como entidades

Características Comunes	DNE Base		DNE Contextualizada	
	$\omega$	Propiedad	$\omega$	Propiedad
Calle	1	street	1	street
Número de Calle	0		0	
Piso	0		0	
Departamento	0		0	
Ciudad	1	city	1	city
Código Postal	1	zip	1	zip
Provincia o Estado	1	state	1	state
Id	1	partNumber	1	partNumber
Descripción	1	itemDescription	1	itemDescription
Nombre Organización	1	name	1	name
Número	1	ebdNumber	1	ebdNumber
$\sum_{i=1,U} \omega_i$	8		8	
Cc		11		11
PCc		0,7273		0,7273

Tabla C.10: Detalles del cálculo del principio 5: Características comunes



# Bibliografía

- ABECKER, ANDREAS y VAN ELST, LUDGER (2009). «Ontologies for knowledge management». En: Steffen Staab y Rudi Studer (Eds.), *Handbook on ontologies*, International handbooks on information systems, pp. 713–734. Springer-Verlag Berlin Heidelberg, 2ª edición.
- ACKER, LIANE y PORTER, BRUCE (1994). «Extracting viewpoints from knowledge bases». En: *Proceedings of the Twelfth National Conference on Artificial Intelligence, AAAI-94. Seattle, Washington, July 31–August 4*, volumen 1, pp. 547–552. American Association for Artificial Intelligence, Menlo Park, CA, USA.
- ALANI, HARITH y BREWSTER, CHRISTOPHER (2006). «Metrics for ranking ontologies». En: Denny Vrandečić; Mari Carmen Suárez-Figueroa; Aldo Gangemi y York Sure (Eds.), *Proceedings of the 4th International EON Workshop, 15th International World Wide Web Conference. Edinburgh, UK, May 22–26*, pp. 24–30.
- ALLEN, JAMES F. (1983). «Maintaining knowledge about temporal intervals». *Communications of the ACM*, **26(11)**, pp. 832–843.
- ALLEN, JAMES F. (1991). «Time and time again: The many ways to represent time». *International Journal of Intelligent Systems*, **6(4)**, pp. 341–355.
- ARPÍREZ VEGA, JULIO CÉSAR; CORCHO, OSCAR; FERNÁNDEZ-LÓPEZ, MARIANO y GÓMEZ-PÉREZ, ASUNCIÓN (2003). «WebODE in a nutshell». *AI Magazine*, **24(3)**, pp. 37–48.
- ARPÍREZ VEGA, JULIO CÉSAR; GÓMEZ-PÉREZ, ASUNCIÓN; LOZANO TELLO, ADOLFO y ANDRADE N. P. PINTO, HELENA SOFÍA (1998). «(ONTO)<sup>2</sup>Agent: An ontology-based WWW broker to select ontologies». En: Asunción

- Gómez-Pérez y V. Richard Benjamins (Eds.), *Proceedings of the Workshop on Applications of Ontologies and Problem-Solving Methods, held in conjunction with ECAI'98. Brighton, UK, August 24–25*, pp. 16–24.
- BECHHOFFER, SEAN; VAN HARMELEN, FRANK; HENDLER, JIM; HORROCKS, IAN; MCGUINNESS, DEBORAH L.; PATEL-SCHNEIDER, PETER F. y STEIN, LYNN ANDREA (2004). «OWL web ontology language reference». *W3C Recommendation 10 February 2004*, The World Wide Web Consortium.  
<http://www.w3.org/TR/owl-ref/>
- BECKETT, DAVE (2004). «RDF/XML syntax specification (revised)». *W3C Recommendation 10 February 2004*, The World Wide Web Consortium.  
<http://www.w3.org/TR/REC-rdf-syntax/>
- BELL, DAVID; DE CESARE, SERGIO; LYCETT, MARK; MUSTAFEE, NAVONIL y TAYLOR, SIMON J. E. (2007). «Semantic web service architecture for simulation model reuse». En: David J. Roberts; Georgios K. Theodoropoulos y Abdulmotaleb El Saddik (Eds.), *Proceedings of the 11th IEEE International Symposium on Distributed Simulation and Real-Time Applications. Chania, Crete Island, October 22–24*, pp. 129–136. IEEE Computer Society, Washington, DC, USA.
- BENERECETTI, MASSIMO; BOUQUET, PAOLO y GHIDINI, CHIARA (2001). «On the dimensions of context dependence: Partiality, approximation, and perspective». En: Varol Akman; Paolo Bouquet; Richmond H. Thomason y Roger A. Young (Eds.), *Modeling and Using Context. Third International and Interdisciplinary Conference, CONTEXT 2001. Dundee, UK, July 27–30. Proceedings*, volumen 2116 de *LNCS*, pp. 59–72. Springer-Verlag Berlin Heidelberg.
- BIRON, PAUL V. y MALHOTRA, ASHOK (2004). «XML Schema part 2: Datatypes second edition». *W3C Recommendation 28 October 2004*, The World Wide Web Consortium.  
<http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/datatypes.html>
- BITTERS, BARRY (2005). *A geographical ontology of objects in the visible domain*. Ph.D. dissertation, College of Social Sciences, The Florida State

University.

[http://etd.lib.fsu.edu/theses\\_1/submitted/etd-04112005-124616/unrestricted/BittersBDissertation.pdf](http://etd.lib.fsu.edu/theses_1/submitted/etd-04112005-124616/unrestricted/BittersBDissertation.pdf)

BOUQUET, PAOLO; EUZENAT, JÉRÔME; FRANCONI, ENRICO; SERAFINI, LUCIANO; STAMOU, GIORGOS y TESSARIS, SERGIO (2004a). «D2.2.1 Specification of a common framework for characterizing alignment». *Deliverable KWEB/2004/D2.2.1/v1.2*, KnowledgeWeb Consortium.

<http://knowledgeweb.semanticweb.org/semanticportal/deliverables/D2.2.1v2.pdf>

BOUQUET, PAOLO; GIUNCHIGLIA, FAUSTO; VAN HARMELEN, FRANK; SERAFINI, LUCIANO y STUCKENSCHMIDT, HEINER (2004b). «Contextualizing ontologies». *Web Semantics: Science, Services and Agents on the World Wide Web*, **1(4)**, pp. 325–343.

BRANK, JANEZ; GROBELNIK, MARKO y MLADENIĆ, DUNJA (2005). «A survey of ontology evaluation techniques». En: *Proceedings of the Conference on Data Mining and Data Warehouses, SiKDD 2005. Ljubljana, Slovenia, October 17*, pp. 166–169.

BRAY, TIM; PAOLI, JEAN; SPERBERG-MCQUEEN, C. M.; MALER, EVE y YERGEAU, FRANÇOIS (2008). «Extensible markup language (XML) 1.0 (fifth edition)». *W3C Recommendation 26 November 2008*, The World Wide Web Consortium.

<http://www.w3.org/TR/2008/REC-xml-20081126/>

BREITMAN, KARIN KOOGAN; CASANOVA, MARCO ANTONIO y TRUSZKOWSKI, WALTER (2007). *Semantic Web: Concepts, technologies and applications*. NASA monographs in systems and software engineering. Springer-Verlag London Limited.

BREWSTER, CHRISTOPHER; ALANI, HARITH; DASMAHAPATRA, SRINANDAN y WILKS, YORICK (2004). «Data driven ontology evaluation». En: *Proceedings of the Fourth International Conference on Language Resources and Evaluation, LREC 2004. Lisbon, Portugal, May 26–28*, pp. 164–168. European Language Resources Association.

- BRICKLEY, DAN y GUHA, RAMANATHAN V. (2004). «RDF vocabulary description language 1.0: RDF Schema». *W3C Recommendation 10 February 2004*, The World Wide Web Consortium.  
<http://www.w3.org/TR/rdf-schema/>
- BRUSA, GRACIELA; CALIUSCO, MARÍA LAURA y CHIOTTI, OMAR (2008). «Towards ontological engineering: A process for building a domain ontology from scratch in public administration». *Expert Systems*, **25(5)**, pp. 484–503.
- BURTON-JONES, ANDREW; STOREY, VEDA C.; SUGUMARAN, VIJAYAN y AHLUWALIA, PUNIT (2005). «A semiotic metrics suite for assessing the quality of ontologies». *Data & Knowledge Engineering*, **55(1)**, pp. 84–102.
- CALIUSCO, MARÍA LAURA (2005). *Soporte para la definición semántica de documentos de negocio electrónicos en relaciones de e-colaboración*. Tesis doctoral, Universidad Tecnológica Nacional, Facultad Regional Santa Fe, Santa Fe, Argentina. ISBN 987-43-9158-8.
- CALVANESE, DIEGO; DE GIACOMO, GIUSEPPE y LENZERINI, MAURIZIO (2002). «Description logics for information integration». En: Antonis Kakas y Fariba Sadri (Eds.), *Computational logic: Logic programming and beyond*, volumen 2408 de *LNCS*, pp. 41–60. Springer-Verlag Berlin Heidelberg.
- CASANOVAS, POMPEU; CASELLAS, NÚRIA; VALLBÉ, JOAN-JOSEP; POBLET, MARTA; BENJAMINS, V. RICHARD; BLÁZQUEZ, MERCEDES; PEÑA, RAÚL y CONTRERAS, JESÚS (2006). «Semantic Web: A legal case study». En: John Davies; Rudi Studer y Paul Warren (Eds.), *Semantic web technologies: Trends and research in ontology-based systems*, pp. 259–280. John Wiley & Sons, Ltd., Chichester, UK.
- CASTANO, SILVANA; FERRARA, ALFIO y MONTANELLI, STEFANO (2007). «Consensus-driven formation of semantic communities of peers». *Deliverable D2.2, ESTEEM Emergent Semantics and cooperation in multi-knowledge Environments*.  
[http://www.dis.uniroma1.it/~esteem/docs/Deliverable dic06/Milano/UNIMI-D22.pdf](http://www.dis.uniroma1.it/~esteem/docs/Deliverable_dic06/Milano/UNIMI-D22.pdf)

- CASTANO, SILVANA; FERRARA, ALFIO; MONTANELLI, STEFANO y RACCA, GIANPAOLO (2004). «Semantic information interoperability in open networked systems». En: Mokrane Bouzeghoub; Carole Goble; Vipul Kashyap y Stefano Spaccapietra (Eds.), *Semantics of a Networked World. Semantics for Grid Databases. First International IFIP Conference, ICSNW 2004. Paris, France, June 17–19. Revised Selected Papers*, volumen 3226 de LNCS, pp. 215–230. Springer-Verlag Berlin Heidelberg.
- CHAUDHRI, VINAY K.; FARQUHAR, ADAM; FIKES, RICHARD; KARP, PETER D. y RICE, JAMES P. (1998). «Open knowledge base connectivity 2.0.3». *Technical report April 9, 2008*, Artificial Intelligence Center of SRI International and the Knowledge Systems Laboratory of Stanford University.  
<http://www.ai.sri.com/~okbc/okbc-2-0-3.pdf>
- COLOMB, ROBERT M. (2002). «Quality of ontologies in interoperating information systems». *Technical report 18/02 ISIB-CNR*, National Research Council, Institute of Biomedical Engineering, ISIB-CNR, Padova, Italy.  
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.100.2321&rep=rep1&type=pdf>
- CORCHO, OSCAR y GÓMEZ-PÉREZ, ASUNCIÓN (2000). «A roadmap to ontology specification languages». En: Rose Dieng y Olivier Corby (Eds.), *Knowledge Engineering and Knowledge Management Methods, Models, and Tools. 12th International Conference, EKAW 2000. Juan-les-Pins, France, October 2–6. Proceedings*, volumen 1937 de LNCS, pp. 80–96. Springer-Verlag Berlin Heidelberg, London, UK.
- CRISTANI, MATTEO y CUEL, ROBERTA (2004). «Methodologies for the semantic web: State-of-the-art of ontology methodology». *AIS SIGSEMIS Bulletin*, **1(2)**, pp. 103–112. Theme: SW Challenges for KM.
- DAML.ORG (2006). «DAML. The DARPA agent markup language homepage». Último acceso Febrero 26, 2011.  
<http://www.daml.org/>
- DAVIES, JOHN; FENSEL, DIETER y VAN HARMELEN, FRANK (2003). *Towards the semantic web: Ontology-driven knowledge management*. John Wiley & Sons.

- DE ALMEIDA FALBO, RICARDO (2004). «Experiences in using a method for building domain ontologies». En: *Proceedings of the 16th International Conference on Software Engineering and Knowledge Engineering, SEKE'2004. International Workshop on Ontology in Action, OIA'2004. Banff, Alberta, Canada, June 21*, pp. 474–477. Knowledge Systems Institute Graduate School.
- DELOACH, SCOTT A.; WOOD, MARK F. y SPARKMAN, CLINT H. (2001). «Multiagent systems engineering». *International Journal of Software Engineering and Knowledge Engineering*, **11(3)**, pp. 231–258.
- DELTEIL, ALEXANDRE; FARON, CATHERINE y DIENG, ROSE (2002). «Building concept lattices by learning concepts from RDF graphs annotating web documents». En: Uta Priss; Dan Corbett y Galia Angelova (Eds.), *Conceptual Structures: Integration and Interfaces. 10th International Conference on Conceptual Structures, ICCS 2002. Borovets, Bulgaria, July 15–19. Proceedings*, volumen 2393 de LNCS, pp. 191–204. Springer-Verlag Berlin Heidelberg.
- DIETRICH, JENS y ELGAR, CHRIS (2007). «Towards a web of patterns». *Web Semantics: Science, Services and Agents on the World Wide Web*, **5(2)**, pp. 108–116.
- DOCKHORN COSTA, PATRICIA; GUIZZARDI, GIANCARLO; ALMEIDA, JOÃO PAULO A.; FERREIRA PIRES, LUÍS y VAN SINDEREN, MARTEN (2006). «Situations in conceptual modeling of context». En: *10th IEEE International Enterprise Distributed Object Computing Conference Workshops, EDOCW 2006. Hong Kong, China, October 16–20*, p. 6. IEEE Computer Society, Washington, DC.
- DRUMOND, LUCAS y GIRARDI, ROSARIO (2008). «A survey of ontology learning procedures». En: Fred Freitas; Heiner Stuckenschmidt; Helena Sofia Pinto; Andreia Malucelli y Óscar Corcho (Eds.), *Proceedings of the 3rd Workshop on Ontologies and their Applications. Salvador, Bahia, Brazil, October 26*, volumen 427 de CEUR workshop proceedings. CEUR-WS.org.
- DUKE, ALISTAIR y RICHARDSON, MARC (2006). «A semantic service-oriented architecture for the telecommunications industry». En: John Davies; Rudi

- Studer y Paul Warren (Eds.), *Semantic web technologies: Trends and research in ontology-based systems*, pp. 281–299. John Wiley & Sons, Ltd., Chichester, UK.
- ETIM (nd). «ETIM. Der standard der elektrobranche». *Standard*, ETIM Deutschland e.V.  
<http://www.etim.de/>
- EUZENAT, JÉRÔME (2001). «Towards a principled approach to semantic interoperability». En: Asunción Gómez Pérez; Michael Grüninger; Heiner Stuckenschmidt y Michael Uschold (Eds.), *Workshop on Ontologies and Information Sharing at the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI-2001. Seattle, WA, US, August 4–5*, pp. 19–25.
- EUZENAT, JÉRÔME; BACH, THANH LE; DIENG, ROSE; DUC, CHAN LE; NAPOLI, AMEDEO; ZIMMERMANN, ANTOINE; ... y MAYNARD, DIANA (2007). «Knowledge web 2.2: Heterogeneity in the semantic web». *Work package KWEB/2004/WP2.2 synthesis/0.1*, KnowledgeWeb Consortium.  
<http://disi.unitn.it/~p2p/RelatedWork/Matching/kweb22-full.pdf>
- EUZENAT, JÉRÔME; FERRARA, ALFIO; MEILICKE, CHRISTIAN; PANE, JUAN; SCHARFFE, FRANÇOIS; SHVAIKO, PAVEL; STUCKENSCHMIDT, HEINER; ŠVÁB ZAMAZAL, ONDŘEJ; SVÁTEK, VOJTĚCH y DOS SANTOS, CÁSSIA TROJAHN (2010). «Results of the ontology alignment evaluation initiative 2010». En: Pavel Shvaiko; Jérôme Euzenat; Fausto Giunchiglia; Heiner Stuckenschmidt; Ming Mao y Isabel Cruz (Eds.), *Proceedings of the 5th International Workshop on Ontology Matching (OM-2010), collocated with the 9th International Semantic Web Conference (ISWC-2010). Shanghai, China, November 7*, volumen 689 de *CEUR workshop proceedings*, pp. 85–117. CEUR-WS.org.
- EUZENAT, JÉRÔME y SHVAIKO, PAVEL (2007). *Ontology matching*. Springer-Verlag Berlin Heidelberg, London, 2ª edición.
- FARQUHAR, ADAM; FIKES, RICHARD y RICE, JAMES (1997). «The ontolingua server: A tool for collaborative ontology construction». *International Journal of Human-Computer Studies*, **46(6)**, pp. 707–727.

FEIER, CRISTINA; ROMAN, DUMITRU; POLLERES, AXEL; DOMINGUE, JOHN; STOLLBERG, MICHAEL y FENSEL, DIETER (2005). «Towards intelligent web services: Web service modeling ontology (WSMO)». En: *Proceedings of the International Conference on Intelligent Computing, ICIC. Hefei, China, August 23–26*, .  
<http://aran.library.nuigalway.ie/xmlui/bitstream/handle/10379/417/feie-et-al-2005.pdf?sequence=1>

FENSEL, DIETER (2004). *Ontologies: A silver bullet for knowledge management and electronic commerce*. Springer-Verlag Berlin Heidelberg, 2ª edición.

FENSEL, DIETER; HORROCKS, IAN; HARMELEN, FRANK VAN; DECKER, STEFAN; ERDMANN, MICHAEL y KLEIN, MICHEL C. A. (2000). «OIL in a nutshell». En: Rose Dieng y Olivier Corby (Eds.), *Knowledge Engineering and Knowledge Management Methods, Models, and Tools. 12th International Conference, EKAW 2000. Juan-les-Pins, France, October 2–6. Proceedings*, volumen 1937 de *LNCIS*, pp. 1–16. Springer-Verlag Berlin Heidelberg, London, UK.

FERGUSON, GEORGE (2003). «DAML-time homepage». Último acceso Marzo 3, 2011.  
<http://www.cs.rochester.edu/~ferguson/daml/>

FIKES, RICHARD (2001). «Multi-use ontologies». PowerPoint slides. Computer Science Department, Stanford University.  
<http://www.ksl.stanford.edu/people/fikes/cs222/2001/ontologies/Ontologies.PPT>

FIKES, RICHARD; HAYES, PATRICK y HORROCKS, IAN (2004). «OWL-QL. A language for deductive query answering on the semantic web». *Web Semantics: Science, Services and Agents on the World Wide Web*, **2(1)**, pp. 19–29.

FIRAT, AYKUT; MADNICK, STUART y MANOLA, FRANK (2005). «Multi-dimensional ontology views via contexts in the ECOIN semantic interoperability framework». En: Pavel Shvaiko; Jérôme Euzenat; Alain Leger; Deborah L. McGuinness y Holger Wache (Eds.), *Contexts and Ontologies:*

- Theory, Practice and Applications. Papers from the 2005 AAI Workshop. Pittsburgh, Pennsylvania, July 9*, pp. 1–8. AAAI Press, Menlo Park, California.
- FRATERNALI, PIERO (1999). «Tools and approaches for developing data-intensive web applications: A survey». *ACM Computing Surveys*, **31(3)**, pp. 227–263.
- GAITANOU, PANOREA (2007). «Ontology semantics and applications». En: *Proceedings of 2nd International Conference on Metadata and Semantics Research (CD-ROM). Corfu Island, Greece, October 11–12*, Ionian University, University of Alcalá.
- GAŠEVIĆ, DRAGAN; DJURIĆ, DRAGAN y DEVEDŽIĆ, VLADAN (2006). *Model driven architecture and ontology development*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- GAŠEVIĆ, DRAGAN; KAVIANI, NIMA y MILANOVIĆ, MILAN (2009). «Ontologies and software engineering». En: Steffen Staab y Rudi Studer (Eds.), *Handbook on ontologies*, International handbooks on information systems, pp. 593–615. Springer-Verlag Berlin Heidelberg, 2ª edición.
- GENESERETH, MICHAEL R. y FIKES, RICHARD E. (1992). «Knowledge interchange format. Version 3.0. Reference manual». *Technical report Logic-92-1*, Stanford Logic Group. Stanford University.  
<http://logic.stanford.edu/kif/Hypertext/kif-manual.html>
- GENNARI, JOHN H.; MUSEN, MARK A.; FERGERSON, RAY W.; GROSSO, WILLIAM E.; CRUBÉZY, MONICA; ERIKSSON, HENRIK; NOY, NATALYA F. y TU, SAMSON W. (2003). «The evolution of Protégé-2000: An environment for knowledge-based systems development». *International Journal of Human-Computer Studies*, **58(1)**, pp. 89–123.
- GÓMEZ-PÉREZ, ASUNCIÓN (1995). «Some ideas and examples to evaluate ontologies». En: *Proceedings of the 11th Conference on Artificial Intelligence for Applications, CAIA '95. Los Angeles, CA, February 20–23*, pp. 299–305. IEEE Computer Society, Washington, DC, USA.
- GÓMEZ-PÉREZ, ASUNCIÓN (1999). «Evaluation of taxonomic knowledge in ontologies and knowledge bases». En: Brian R. Gaines; Rob Kremer y Mark A.

- Musen (Eds.), *Proceedings of the Twelfth Workshop on Knowledge Acquisition, Modeling and Management, KAW'99. Banff, Alberta, Canada, October 16–21*, volumen 2, pp. 6.8.1–6.8.18. SRDG Publications, University of Calgary.
- GÓMEZ-PÉREZ, ASUNCIÓN; FERNÁNDEZ-LÓPEZ, MARIANO y CORCHO, OSCAR (2004). *Ontological engineering*. Advanced information and knowledge processing. Springer-Verlag London Limited, New York, 2ª edición.
- GÓMEZ-PÉREZ, ASUNCIÓN y MANZANO-MACHO, DAVID (2003). «A survey of ontology learning methods and techniques». *Deliverable 1.5*, OntoWeb Consortium. Project number IST-2000-29243.  
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.93.3714&rep=rep1&type=pdf>
- GOODCHILD, MICHAEL F. y KEMP, KAREN K. (1990). «NCGIA core curriculum in GIS». National Center for Geographic Information and Analysis, University of California. Santa Barbara, CA. Último acceso Marzo 4, 2011.  
<http://www.geog.ubc.ca/courses/klink/gis.notes/ncgia/toc.html>
- GRÜBER, THOMAS (1993). «A translation approach to portable ontology specification». *Knowledge Acquisition*, **5(2)**, pp. 199–220.
- GRUBER, THOMAS R. (1995). «Toward principles for the design of ontologies used for knowledge sharing». *International Journal of Human-Computer Studies*, **43(5-6)**, pp. 907–928.
- GRÜNINGER, MICHAEL y FOX, MARK S. (1995). «Methodology for the design and evaluation of ontologies». En: Douglas Skuce (Ed.), *Workshop on Basic Ontological Issues in Knowledge Sharing at the Fourteenth International Joint Conference on Artificial Intelligence, IJCAI-95. Montréal, Québec, Canada, August 20–25*, pp. 6.1–6.10.
- GRÜNINGER, MICHAEL y LEE, JINTAE (2002). «Ontology applications and design. Introduction». *Communications of the ACM*, **45(2)**, pp. 39–41.
- GUARINO, NICOLA (1994). «The ontological level». En: Roberto Casati; Barry Smith y Graham White (Eds.), *Philosophy and the cognitive science*, pp. 443–456. Hölder-Pichler-Tempsky, Vienna.

- GUARINO, NICOLA (1998). «Formal ontology and information systems». En: Nicola Guarino (Ed.), *Proceedings of the First International Conference on Formal Ontology in Information Systems, FOIS'98. Trento, Italy, June 6–8*, pp. 3–15. IOS Press, Amsterdam.
- GUARINO, NICOLA (2009). «The ontological level: Revisiting 30 years of knowledge representation». En: Alexander Borgida; Vinay Chaudhri; Paolo Giorgini y Eric Yu (Eds.), *Conceptual modeling: Foundations and applications*, volumen 5600 de *LNCS*, pp. 52–67. Springer-Verlag Berlin Heidelberg.
- GUARINO, NICOLA; OBERLE, DANIEL y STAAB, STEFFEN (2009). «What is an ontology?» En: Steffen Staab y Rudi Studer (Eds.), *Handbook on ontologies*, International handbooks on information systems, pp. 1–17. Springer-Verlag Berlin Heidelberg, 2ª edición.
- GUIZZARDI, GIANCARLO (2005). *Ontological foundations for structural conceptual models*. Tesis doctoral, University of Twente, Enschede, The Netherlands. Telematica instituut fundamental research series. Volume 015. Telematica Instituut.
- GUIZZARDI, GIANCARLO (2007). «On ontology, ontologies, conceptualizations, modeling languages, and (meta)models». En: Olegas Vasilecas; Johann Eder y Albertas Caplinskis (Eds.), *Proceeding of the 2007 Conference on Databases and Information Systems IV: Selected Papers from the Seventh International Baltic Conference DB&IS'2006*, pp. 18–39. IOS Press, Amsterdam, The Netherlands.
- HAJNAL, AKOS; PEDONE, GIANFRANCO y VARGA, LASZLO (2007). «Ontology-driven agent code generation for home care in Protégé». En: *Proceedings of the 10th International Protégé Conference. Budapest, Hungary, July 15–18*, pp. 91–93.
- HALLER, ARMIN; GONTARCZYK, JEDRZEJ y KOTINURMI, PAAVO (2008). «Towards a complete SCM ontology: The case of ontologising RosettaNet». En: Roger L. Wainwright y Hisham Haddad (Eds.), *Proceedings of the 2008 ACM Symposium on Applied Computing, SAC '08. Fortaleza, Ceara, Brazil, March 16–20*, pp. 1467–1473. ACM, New York, NY, USA.

- HAMEED, ADIL; PREECE, ALUN y SLEEMAN, DEREK (2004). «Ontology reconciliation». En: Steffen Staab y Rudi Studer (Eds.), *Handbook on ontologies*, International handbooks on information systems, capítulo 12, pp. 231–250. Springer-Verlag Berlin Heidelberg.
- HEPP, MARTIN; LEYMAN, FRANK; DOMINGUE, JOHN; WAHLER, ALEXANDER y FENSEL, DIETER (2005). «Semantic business process management: A vision towards using semantic web services for business process management». En: *Proceedings of the IEEE International Conference on e-Business Engineering, ICEBE '05. Beijing, China, October 18–20*, pp. 535–540. IEEE Computer Society, Washington, DC, USA.
- HOBBS, JERRY R. y PAN, FENG (2004). «An ontology of time for the semantic web». *ACM Transactions on Asian Language Information Processing (TALIP)*, **3(1)**, pp. 66–85.
- HORRIDGE, MATTHEW (2009). «A practical guide to building OWL ontologies using Protégé 4 and CO-ODE Tools. Edition 1.2». *Protégé owl tutorial*, The University Of Manchester.  
[http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP4\\_v1.2.pdf](http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1.2.pdf)
- HORRIDGE, MATTHEW; KNUBLAUCH, HOLGER; RECTOR, ALAN; STEVENS, ROBERT y WROE, CHRIS (2004). «A practical guide to building OWL ontologies using the Protégé-OWL plugin and CO-ODE tools. Edition 1.0». *Protégé owl tutorial*, The University Of Manchester.  
[http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP3\\_v1.0.pdf](http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP3_v1.0.pdf)
- ISLAM, AKM SAIFUL; BERMUDEZ, LUIS; BERAN, BORA; FELLAH, STEPHANE y PIASECKI, MICHAEL (2004). «Ontology for geographic information - metadata (iso 19115:2003), version 2.0». Último acceso Diciembre 18, 2006.  
<http://loki.cae.drexel.edu/~wbs/ontology/iso-19103.htm>
- ISO, TECHNICAL COMMITTEE 176/SC 2 (2000). «Quality management systems. Requirements». *Standard ISO 9001:2000*, International Organization for Standardization.

- JOHANNESSON, PAUL (1994). «A method for transforming relational schemas into conceptual schemas». En: *Proceedings of the Tenth International Conference on Data Engineering. Houston, Texas, USA, February 14–18*, pp. 190–201. IEEE Computer Society, Washington, DC, USA.
- KALFOGLOU, YANNIS y SCHORLEMMER, MARCO (2003). «Ontology mapping: The state of the art». *The Knowledge Engineering Review*, **18(1)**, pp. 1–31.
- KALYANPUR, ADITYA; PARSIA, BIJAN; SIRIN, EVREN; GRAU, BERNARDO CUENCA y HENDLER, JAMES (2006). «Swoop: A web ontology editing browser». *Web Semantics: Science, Services and Agents on the World Wide Web*, **4(2)**, pp. 144–153.
- KARP, PETER D.; CHAUDHRI, VINAY K. y THOMERE, JEROME F. (2000). «XOL: An XML-based ontology exchange language. Version 0.5». *Technical note 676*, AI Center, SRI International, Menlo Park, CA.  
<http://www.ai.sri.com/pubs/files/676.pdf>
- KASHYAP, VIPUL (1999). «Design and creation of ontologies for environmental information retrieval». En: Brian R. Gaines; Rob Kremer y Mark A. Musen (Eds.), *Proceedings of the Twelfth Workshop on Knowledge Acquisition, Modeling and Management, KAW'99. Banff, Alberta, Canada, October 16–21*, pp. 5.4.1–5.4.18. SRDG Publications, University of Calgary.
- KASHYAP, VIPUL y SHETH, AMIT (1997). «Semantic heterogeneity in global information systems: The role of metadata, context and ontologies». En: Michael P. Papazoglou y Gunter Schlageter (Eds.), *Cooperative information systems: Trends and directions*, pp. 139–178. Academic Press.
- KIFER, MICHAEL; LAUSEN, GEORG y WU, JAMES (1995). «Logical foundations of object-oriented and frame-based languages». *Journal of the ACM*, **42(4)**, pp. 741–843.
- KLEIN, MICHEL (2001). «Combining and relating ontologies: An analysis of problems and solutions». En: Asunción Gómez Pérez; Michael Grüninger; Heiner Stuckenschmidt y Michael Uschold (Eds.), *Workshop on Ontologies and Information Sharing at the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI-2001. Seattle, WA, US, August 4–5*, pp. 53–62.

KNUBLAUCH, HOLGER (2003). «An AI tool for the real world: Knowledge modeling with Protégé». *JavaWorld*. (June 20, 2003).

<http://www.javaworld.com/javaworld/jw-06-2003/jw-0620-protege.html?>

KNUBLAUCH, HOLGER; FERGERSON, RAY W.; NOY, NATALYA F. y MUSEN, MARK A. (2004). «The Protégé OWL plugin: An open development environment for semantic web applications». En: Sheila A. McIlraith; Dimitris Plexousakis y Frank van Harmelen (Eds.), *The Semantic Web - ISWC 2004. Third International Semantic Web Conference. Hiroshima, Japan, November 7-11. Proceedings*, volumen 3298 de LNCS, pp. 229-243. Springer-Verlag Berlin Heidelberg.

KOTINURMI, PAAVO y VITVAR, TOMAS (2006). «Adding semantics to rosettaNet specifications». En: Les Carr; David De Roure; Arun Iyengar; Carole A. Goble y Michael Dahlin (Eds.), *Proceedings of the 15th International Conference on World Wide Web, WWW 2006. Edinburgh, Scotland, UK, May 23-26*, pp. 1059-1060. ACM, New York, NY, USA.

KRUMMENACHER, RETO; LAUSEN, HOLGER y STRANG, THOMAS (2007). «Analyzing the modeling of context with ontologies». En: Walteneagus Dargie; Thomas Springer y Bruno Klauser (Eds.), *Context-Awareness for Self-Managing Systems (Devices, Applications and Networks). International Workshop CASEMANS 2007. Toronto, Canada, May 13. Proceedings*, pp. 11-22. VDE Verlag.

LARA, RUBÉN; POLLERES, AXEL; LAUSEN, HOLGER; ROMAN, DUMITRU; DE BRUIJN, JOS y FENSEL, DIETER (2005). «D4.1v0.1 A conceptual comparison between WSMO and OWL-S». *WSMO working draft 05 January 2005*, Digital Enterprise Research Institute (DERI).

<http://www.wsmo.org/2004/d4/d4.1/v0.1/20050106/>

LASSILA, ORA y MCGUINNESS, DEBORAH (2001). «The role of frame-based representation on the semantic web». *Electronic Transactions on Artificial Intelligence (ETAI)*, **6(5)**.

<http://www.ep.liu.se/ea/cis/2001/005/>

- LENAT, DOUGLAS B. y GUHA, RAMANATHAN V. (1989). *Building large knowledge-based systems. Representation and inference in the cyc project*. Addison-Wesley Longman Publishing Co., Inc., 1ª edición.
- LÉGER, ALAIN (2002). «Successful scenarios for ontology-based applications v1.0». *Deliverable 2.1*, OntoWeb Consortium. Project number IST-2000-29243. <http://www.sti-innsbruck.at/fileadmin/documents/deliverables/Ontoweb/D2.1.pdf>
- LINKOVÁ, ZDEŇKA y NEDBAL, RADIM (2007). «Ontology approach to integration of geographical data». En: *Proceedings of the 1st Workshop Evolutionary Techniques in Data-processing, WETDAP 2007. In Conjunction with Znalosti (Knowledge) 2007. February 21–23*, pp. 35–41. Faculty of Electrical Engineering and Computer Science, Ostrava, Czech Republic.
- LUKE, SEAN y HEFLIN, JEFF (2000). «SHOE 1.01». *Proposed specification April 28*, Parallel Understanding Systems Group, Department of Computer Science, University of Maryland. SHOE project. <http://www.cs.umd.edu/projects/plus/SHOE/spec.html>
- MACGREGOR, ROBERT M. (1991). «Inside the LOOM description classifier». *ACM SIGART Bulletin*, **2(3)**, pp. 88–92.
- MAEDCHE, ALEXANDER; MOTIK, BORIS; STOJANOVIC, LJILJANA; STUDER, RUDI y VOLZ, RAPHAEL (2003). «Ontologies for enterprise knowledge management». *IEEE Intelligent Systems*, **18(2)**, pp. 26–33.
- MAEDCHE, ALEXANDER y STAAB, STEFFEN (2001). «Ontology learning for the semantic web». *IEEE Intelligent Systems*, **16(2)**, pp. 72–79.
- MARTIN, DAVID; BURSTEIN, MARK; HOBBS, JERRY; LASSILA, ORA; MCILRAITH, SHEILA; NARAYANAN, SRINI; PAOLUCCI, MASSIMO; PARSIA, BIJAN; PAYNE, TERRY; SIRIN, EVREN; SRINIVASAN, NAVEEN y SYCARA, KATIA (2004). «OWL-S: Semantic markup for web services». *W3C Member Submission 22 November 2004*, The World Wide Web Consortium. <http://www.w3.org/Submission/OWL-S/>

- MCGUINNESS, DEBORAH L. y VAN HARMELEN, FRANK (2004). «OWL web ontology language overview». *W3C Recommendation 10 February 2004*, The World Wide Web Consortium.  
<http://www.w3.org/TR/2004/REC-owl-features-20040210/>
- MELLOR, STEPHEN J.; SCOTT, KENDALL; UHL, AXEL y WEISE, DIRK (2004). *MDA distilled: Principles of model-driven architecture*. Addison-Wesley Professional.
- MOTTA, ENRICO (1999). *Reusable components for knowledge modelling. Principles and case studies in parametric design*. IOS Press, Amsterdam, The Netherlands.
- NOY, NATALYA F. y MCGUINNESS, DEBORAH L. (2001). «Ontology development 101: A guide to creating your first ontology». *Technical report KSL-01-05*, Stanford Knowledge Systems Laboratory.  
<http://ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness.pdf>
- NOY, NATALYA FRIDMAN; FERGERSON, RAY W. y MUSEN, MARK A. (2000). «The knowledge model of Protégé-2000: Combining interoperability and flexibility». En: Rose Dieng y Olivier Corby (Eds.), *Knowledge Engineering and Knowledge Management Methods, Models, and Tools. 12th International Conference, EKAW 2000. Juan-les-Pins, France, October 2–6. Proceedings*, volumen 1937 de *LNCS*, pp. 17–32. Springer-Verlag Berlin Heidelberg, London, UK.
- NOY, NATALYA FRIDMAN y HAFNER, CAROLE D. (1997). «The state of the art in ontology design: A survey and comparative review». *AI Magazine*, **18(3)**, pp. 53–74.
- OBJECT MANAGEMENT GROUP, OMG (2008). «Software & systems process engineering meta-model specification. Version 2.0». *Standard formal/2008-04-01*, Object Management Group.  
<http://www.omg.org/spec/SPEM/2.0/PDF>
- OBJECT MANAGEMENT GROUP, OMG (2009). «Ontology definition metamodel.

- Version 1.0». *Specification formal/2009-05-01*, Object Management Group.  
<http://www.omg.org/spec/ODM/1.0/>
- OBRST, LEO; CEUSTERS, WERNER; MANI, INDERJEET; RAY, STEVE y SMITH, BARRY (2007). «The evaluation of ontologies. Toward improved semantic interoperability». En: Christopher J. O. Baker y Kei-Hoi Cheung (Eds.), *Semantic Web. Revolutionizing knowledge discovery in the life sciences*, capítulo 7, pp. 139–158. Springer Science+Business Media.
- ODP (2010). «The open directory project».  
<http://www.dmoz.org/>
- OPEN APPLICATIONS GROUP, OAGI (nd). «OAGIS®9.4.1». *Standard 9.4.1*, Open Applications Group.  
<http://www.schemacentral.com/sc/oagis941/ss.html>
- PAPATHEODOROU, CHRISTOS; VASSILIOU, ALEXANDRA y SIMON, BERND (2002). «Discovery of ontologies for learning resources using word-based clustering». En: Philip Barker y Samuel Rebelsky (Eds.), *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications, ED-MEDIA 2002. Denver, Colorado, USA, June 24–29*, pp. 1523–1528. AACE.
- PRUD'HOMMEAUX, ERIC y SEABORNE, ANDY (2008). «SPARQL query language for RDF». *W3C Recommendation 15 January 2008*, The World Wide Web Consortium.  
<http://www.w3.org/TR/rdf-sparql-query/>
- RAGGETT, DAVE; HORS, ARNAUD LE y JACOBS, IAN (1999). «HTML 4.01 specification». *W3C Recommendation 24 December 1999*, The World Wide Web Consortium.  
<http://www.w3.org/TR/html401/>
- RAHM, ERHARD y BERNSTEIN, PHILIP A. (2001). «A survey of approaches to automatic schema matching». *The VLDB Journal*, **10(4)**, pp. 334–350.
- RANGANATHAN, ANAND y CAMPBELL, ROY H. (2003). «Ontologies in a pervasive computing environment». En: *Workshop on Ontologies in Distributed Systems at the Eighteenth International Joint Conference on*

- Artificial Intelligence, IJCAI-03. Acapulco, Mexico, August 9–15*, pp. 10–14. ACM Press.
- RICO, MARIELA; CALIUSCO, MARÍA LAURA; CHIOTTI, OMAR y GALLI, MARÍA ROSA (2006). «Combining contexts and ontologies: A case study and a conceptual proposal». En: Pavel Shvaiko; Jérôme Euzenat; Alain Léger; Deborah L. McGuinness y Holger Wache (Eds.), *Proceedings of the 2nd International Workshop on Contexts and Ontologies: Theory, Practice and Applications (C&O-2006). Riva del Garda, Italy, August 28*, pp. 47–49.
- RICO, MARIELA; CALIUSCO, MARÍA LAURA; CHIOTTI, OMAR y GALLI, MARÍA ROSA (2008). «Toward the improvement of alignment between heterogeneous domain ontologies». En: *XXXIV Conferencia Latinoamericana de Informática CLEI 2008. Santa Fe, Argentina, September 8–12*, pp. 579–588.
- RICO, MARIELA; CALIUSCO, MARÍA LAURA; GALLI, MARÍA ROSA y CHIOTTI, OMAR (2011). «Towards semantic interoperability between information systems». En: Vijayan Sugumaran y Jon Atle Gulla (Eds.), *Applied semantic web technologies: Using semantics in intelligent information processing*, Taylor & Francis. En edición.
- RICO, MARIELA; TAVERNA, MARÍA LAURA; CALIUSCO, MARÍA LAURA; CHIOTTI, OMAR y GALLI, MARÍA ROSA (2009). «Adding semantics to electronic business documents exchanged in collaborative commerce relations». *Journal of Theoretical and Applied Electronic Commerce Research*, **4(1)**, pp. 72–90.
- RICO, MARIELA; TAVERNA, MARÍA LAURA; CALIUSCO, MARÍA LAURA; GALLI, MARÍA ROSA y CHIOTTI, OMAR (2007). «A comprehensive framework for representing semantics via context and ontology in the collaborative commerce area». En: Virgilio Almeida y Ricardo Baeza-Yates (Eds.), *5th Latin American Web Congress, LA-WEB 2007. Santiago de Chile, Chile, October 31–November 2*, pp. 110–119. IEEE Computer Society, Los Alamitos, CA.
- RIFAIEH, RAMI y BENHARKAT, NABILA (2006). «From ontology phobia to contextual ontology use in enterprise information systems». En: David Taniar

- y Johanna Wenny Rahayu (Eds.), *Web semantics and ontology*, pp. 115–164. Idea Group Publishing, Hershey, Pennsylvania, USA.
- ROWELL, MICHAEL y FEBLOWITZ, MARK (2002). «OAGIS 8.0 design document version 1.1». *Documen 20021024-1*, Open Applications Group, Inc.  
[http://www.oagi.org/oagi/downloads/designdocument/OAGIS 8.0 Design Document 011.pdf](http://www.oagi.org/oagi/downloads/designdocument/OAGIS_8.0_Design_Document_011.pdf)
- SEABORNE, ANDY (2004). «RDQL - A query language for RDF». *W3C member submission 9 January 2004*, Hewlett-Packard.  
<http://www.w3.org/Submission/RDQL/>
- SHVAIKO, PAVEL y EUZENAT, JÉRÔME (2011). «Ontology matching projects». Último acceso Marzo 9, 2011.  
<http://www.ontologymatching.org/projects.html>
- SMITH, BARRY; KUSNIERCZYK, WACLAW; SCHOBBER, DANIEL y CEUSTERS, WERNER (2006). «Towards a reference terminology for ontology research and development in the biomedical domain». En: Olivier Bodenreider (Ed.), *Proceedings of the Second International Workshop on Formal Biomedical Knowledge Representation: “Biomedical Ontology in Action” (KR-MED 2006), collocated with the 4th International Conference on Formal Ontology in Information Systems (FOIS-2006). Baltimore, Maryland, USA, November 8*, volumen 222 de *CEUR workshop proceedings*, pp. 57–65. CEUR-WS.org.
- STAAB, STEFFEN y STUDER, RUDI (Eds.) (2004). *Handbook on ontologies*. International handbooks on information systems. Springer-Verlag Berlin Heidelberg.
- STAAB, STEFFEN y STUDER, RUDI (Eds.) (2009). *Handbook on ontologies*. International handbooks on information systems. Springer-Verlag Berlin Heidelberg, 2ª edición.
- STEINBERG, DAVE; BUDINSKY, FRANK; PATERNOSTRO, MARCELO y MERKS, ED (2008). *EMF: Eclipse modeling framework*. Eclipse series. Addison-Wesley Professional, 2ª edición.

- STOJANOVIC, LJILJANA; STOJANOVIC, NENAD y VOLZ, RAPHAEL (2002). «Migrating data-intensive web sites into the semantic web». En: *Proceedings of the 2002 ACM Symposium on Applied Computing, SAC '02. Madrid, Spain, March 10–14*, pp. 1100–1107. ACM Press, New York, NY, USA.
- STUCKENSCHMIDT, HEINER y VAN HARMELEN, FRANK (2005). *Information sharing on the semantic web*. Advanced information and knowledge processing. Springer-Verlag Berlin Heidelberg, Germany.
- STUDER, RUDI; BENJAMINS, V. RICHARD y FENSEL, DIETER (1998). «Knowledge engineering: Principles and methods». *Data & Knowledge Engineering*, **25(1-2)**, pp. 161–197.
- STVILIA, BESIKI (2007). «A model for ontology quality evaluation». *First Monday*, **12(12)**.  
<http://frodo.lib.uic.edu/newjournals/index.php/FM/article/viewArticle/453>
- SUPEKAR, KAUSTUBH; PATEL, CHINTAN y LEE, YUGYUNG (2004). «Characterizing quality of knowledge on semantic web». En: Valerie Barr y Zdravko Markov (Eds.), *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference. Miami Beach, Florida, USA, May 12–14*, AAAI Press.
- SURE, YORK; ERDMANN, MICHAEL; ANGELE, JUERGEN; STAAB, STEFFEN; STUDER, RUDI y WENKE, DIRK (2002). «OntoEdit: Collaborative ontology engineering for the semantic web». En: Ian Horrocks y James A. Hendler (Eds.), *The Semantic Web - ISWC 2002. First International Semantic Web Conference. Sardinia, Italy, June 9–12. Proceedings*, volumen 2342 de LNCS, pp. 221–235. Springer-Verlag Berlin Heidelberg.
- SURE, YORK; STAAB, STEFFEN y STUDER, RUDI (2004). «On-to-knowledge methodology». En: Steffen Staab y Rudi Studer (Eds.), *Handbook on ontologies*, International handbooks on information systems, capítulo 6, pp. 117–132. Springer-Verlag Berlin Heidelberg.
- SURE, YORK; STAAB, STEFFEN y STUDER, RUDI (2009). «Ontology engineering methodology». En: Steffen Staab y Rudi Studer (Eds.), *Handbook on ontologies*,

International handbooks on information systems, pp. 135–152. Springer-Verlag Berlin Heidelberg, 2<sup>a</sup> edición.

SVÁB-ZAMAZAL, ONDĚJ; SVÁTEK, VOJTĚCH; MEILICKE, CHRISTIAN y STUCKENSCHMIDT, HEINER (2008). «Testing the impact of pattern-based ontology refactoring on ontology matching results». En: Pavel Shvaiko; Jérôme Euzenat; Fausto Giunchiglia y Heiner Stuckenschmidt (Eds.), *Proceedings of the 3rd International Workshop on Ontology Matching (OM-2008), collocated with the 7th International Semantic Web Conference (ISWC-2008). Karlsruhe, Germany, October 26*, volumen 431 de *CEUR workshop proceedings*. CEUR-WS.org.

SVÁTEK, VOJTĚCH; ŠVÁB ZAMAZAL, ONDĚJ y PRESUTTI, VALENTINA (2009). «Ontology naming pattern sauce for (human and computer) gourmets». En: Eva Blomqvist; Kurt Sandkuhl; Francois Scharffe y Vojtěch Svátek (Eds.), *Proceedings of the Workshop on Ontology Patterns (WOP 2009), collocated with the 8th International Semantic Web Conference (ISWC-2009). Washington D.C., USA, October 25*, volumen 516 de *CEUR workshop proceedings*, pp. 171–178. CEUR-WS.org.

SYCARA, KATIA P. y PAOLUCCI, MASSIMO (2004). «Ontologies in agent architectures». En: Steffen Staab y Rudi Studer (Eds.), *Handbook on ontologies*, International handbooks on information systems, capítulo 17, pp. 343–364. Springer-Verlag Berlin Heidelberg.

TARTIR, SAMIR; ARPINAR, I. BUDAK; MOORE, MICHAEL; SHETH, AMIT P. y ALEMAN-MEZA, BOANERGES (2005). «OntoQA: Metric-based ontology quality analysis». En: Doina Caragea; Vasant Honavar; Ion Muslea y Raghu Ramakrishnan (Eds.), *Proceedings of IEEE Workshop on Knowledge Acquisition from Distributed, Autonomous, Semantically Heterogeneous Data and Knowledge Sources (KADASH). Houston, TX, November 27*, pp. 45–53. IEEE Computer Society.

THEODORAKIS, MANOS (2001). *Contextualization: An abstraction mechanism for information modeling*. Doctoral dissertation, Department of Computer Science, University of Crete, Crete, Greece.

- THOMPSON, HENRY S.; BEECH, DAVID; MALONEY, MURRAY y MENDELSON, NOAH (2004). «XML schema part 1: Structures second edition». *W3C Recommendation 28 October 2004*, The World Wide Web Consortium.  
<http://www.w3.org/TR/xmlschema-1/>
- TIJERINO, YURI A. y MIZOGUCHI, RIICHIRO (1993). «MULTIS II: Enabling end-users to design problem-solving engines via two-level task ontologies». En: Nathalie Aussenac-Gilles; Guy A. Boy; Brian R. Gaines; Jean-Gabriel Ganascia; Yves Kodratoff y Marc Linster (Eds.), *Knowledge Acquisition for Knowledge-Based Systems. 7th European Workshop, EKAW '93. Toulouse and Caylus, France, September 6–10. Proceedings*, volumen 723 de LNCS, pp. 340–359. Springer-Verlag Berlin Heidelberg.
- UNSPSC (2010). «UNSPSC homepage».  
<http://www.unspsc.org/>
- USCHOLD, MIKE y GRÜNINGER, MICHAEL (1996). «Ontologies: Principles, methods and applications». *Knowledge Engineering Review*, **11(2)**, pp. 93–155.
- USCHOLD, MIKE y KING, MARTIN (1995). «Towards a methodology for building ontologies». En: Douglas Skuce (Ed.), *Workshop on Basic Ontological Issues in Knowledge Sharing at the Fourteenth International Joint Conference on Artificial Intelligence, IJCAI-95. Montréal, Québec, Canada. August 20–25*, . Also available as AIAI-TR-183 from AIAI, The University of Edinburgh.
- VAN HARMELEN, FRANK; PATEL-SCHNEIDER, PETER F. y HORROCKS, IAN (2001). «Reference description of the DAML+OIL (March 2001) ontology markup language». *Revision 4.2*, www-rdf-logic forum.  
<http://www.daml.org/2001/03/reference.html>
- VAN HEIJST, GERTJAN; SCHREIBER, GUUS y WIELINGA, BOB J. (1997). «Using explicit ontologies in KBS development». *International Journal of Human-Computer Studies*, **46(2-3)**, pp. 183–292.
- VILLARREAL, PABLO; CALIUSCO, MARÍA LAURA; GALLI, MARÍA ROSA; SALOMONE, ENRIQUE y CHIOTTI, OMAR (2004). «Decentralized process management for inter-enterprise collaboration». En: B. Sahay (Ed.), *Emerging*

- issues in supply chain management*, pp. 293–310. MacMillan India Limited, New Delhi, India.
- VILLARREAL, PABLO; ROA, JORGE; CHIOTTI, OMAR y SALOMONE, ENRIQUE (2007). «Aligning the business solution with the technological solution in the development of B2B collaborations». En: *Libro de actas: Collecter Iberoamérica 2007: Collaborative Electronic Commerce Technology and Research Conference. Córdoba, Argentina, November 6–9*, pp. 63–72. Universidad Nacional de Córdoba, Fundación Mediterránea.
- VOLZ, RAPHAEL; OBERLE, DANIEL; STAAB, STEFFEN y STUDER, RUDI (2003). «OntoLiFT prototype». *Deliverable 11*, WonderWeb: Ontology infrastructure for the semantic web. IST Project 2001-33052.  
<http://wonderweb.semanticweb.org/deliverables/documents/D11.pdf>
- VHRANDEČIĆ, DENNY (2009). «Ontology evaluation». En: Steffen Staab y Rudi Studer (Eds.), *Handbook on ontologies*, International handbooks on information systems, pp. 293–313. Springer-Verlag Berlin Heidelberg, 2ª edición.
- WACHE, HOLGER y STUCKENSCHMIDT, HEINER (2001). «Practical context transformation for information system interoperability». En: Varol Akman; Paolo Bouquet; Richmond H. Thomason y Roger A. Young (Eds.), *Modeling and Using Context. Third International and Interdisciplinary Conference, CONTEXT 2001. Dundee, UK, July 27–30. Proceedings*, volumen 2116 de LNCS, pp. 367–380. Springer-Verlag Berlin Heidelberg.
- WACHE, HOLGER; VÖGELE, THOMAS; VISSER, UBBO; STUCKENSCHMIDT, HEINER; SCHUSTER, GADI; NEUMANN, HORST y HÜBNER, SEBASTIAN (2001). «Ontology-based integration of information. A survey of existing approaches». En: Asunción Gómez Pérez; Michael Grüninger; Heiner Stuckenschmidt y Michael Uschold (Eds.), *Workshop on Ontologies and Information Sharing at the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI-2001. Seattle, WA, US, August 4–5*, pp. 108–117.
- WOOLDRIDGE, MICHAEL; JENNINGS, NICHOLAS R. y KINNY, DAVID (2000). «The gaia methodology for agent-oriented analysis and design». *Journal of Autonomous Agents and Multi-Agent Systems*, **3(3)**, pp. 285–312.

- WORLD WIDE WEB CONSORTIUM, OWL WORKING GROUP (2009). «OWL 2 web ontology language document overview». *W3C Recommendation 27 October 2009*, The World Wide Web Consortium.  
<http://www.w3.org/TR/2009/REC-owl2-overview-20091027/>
- ZHAO, YAJING; DONG, JING y PENG, TU (2009). «Ontology classification for semantic-web-based software engineering». *IEEE Transactions on Services Computing*, **2(4)**, pp. 303–317.
- ZHU, HONG y HUO, QINGNING (2005). «Developing a software testing ontology in UML for a software growth environment of web-based applications». En: Hongji Yang (Ed.), *Software evolution with UML and XML*, pp. 263–295. Idea Group Publishing.