



Un DSL para Trazabilidad Alimentaria

Bioing. Emanuel Ceferino Caluva

Director: Dr. Gustavo Rossi

Codirector: Ing. MSc. Andrés F. Moltoni

Tesis presentada para obtener el grado de
Magister en Ingeniería de Software

Facultad de Informática
Universidad Nacional de La Plata

La Plata, Mayo de 2018

AGRADECIMIENTOS

A mi director Gustavo Rossi y a mi codirector Andrés Moltoni que me han brindado el tiempo necesario para guiarme en esta tesis.

Al INTA por brindarme la posibilidad de realizar esta Maestría.

A Claudia Pons por su ayuda en el área de Perfiles UML y el lenguaje OCL.

Y a mi familia por siempre estar presente.

ÍNDICE GENERAL

1.	Introducción.....	8
1.1	Objetivos	11
1.2	Estructura de la Tesis	12
2.	Trabajos relacionados	14
2.1	Sistemas de trazabilidad en general.....	14
2.2	Sistemas de trazabilidad alimentaria	15
2.3	Trazabilidad con DSLs	17
3.	Producción de alimentos y Trazabilidad.....	18
3.1	Responsabilidades.....	19
3.2	Calidad y Trazabilidad.....	20
3.3	Modelo de datos de un sistema de trazabilidad	23
3.4	Gestión del ciclo de vida del producto	28
3.5	Tecnologías necesarias para los sistemas	29
3.6	Internet of Things (IoT – <i>Internet de las Cosas</i>)	31
4.	Domain-Specific Languages (DSL) – Perfiles UML.....	37
4.1	DSL	37
4.2	Perfiles UML	38
4.3	Object Constraint Language (OCL)	41
4.4	MDA y los Perfiles UML.....	42
4.5	DSL y UML.....	44
5.	Desarrollo de Perfil UML para trazabilidad alimentaria.....	46
5.1	El Metamodelo del Dominio	47
5.1.1	El Paquete Trazabilidad	47
5.1.2	El Paquete Calidad.....	50
5.2	El Perfil UML.....	51
5.3	Restricciones en OCL	56
5.4	Optimización al perfil diseñado	59
5.5	Evaluación del PerfilTrazabilidadCalidad	61
5.5.1	Análisis de indicadores	61
5.5.2	Análisis cualitativo	63

5.6	El perfil PerfilTrazabilidadCalidad y la IoT	66
6.	Ejemplo de Aplicación del Perfil UML al Dominio Apícola	73
6.1	Importancia de la producción Apícola en Argentina	73
6.2	Descripción del dominio Apícola	73
6.3	Aplicación del PerfilTrazabilidadCalidad	79
6.3.1	Curación	80
6.3.2	Multiplicación.....	81
6.3.3	Colocación de alzas	83
6.3.4	Cosecha	84
6.3.5	Desoperculado y envasado.....	85
6.4	Aplicación de PerfilTrazabilidadCalidad y PerfilIoT	87
7.	Conclusiones.....	90
8.	Trabajos Futuros	92
9.	Referencias	93

ÍNDICE DE FIGURAS

Figura 1 Modelo de datos centrado en el producto para trazabilidad (Gemessi, 2010).....	26
Figura 2 Modelo de datos para trazabilidad (Bechini et al., (2005))	27
Figura 3 Ciclo de vida del producto.....	28
Figura 4 Ciclo de la agricultura inteligente.....	33
Figura 5 Virtualización de una caja de pescado, con información logística, para facilitar el control y manejo remoto	34
Figura 6 Modelo de Dominio para la IoT (Bauer, 2013).....	36
Figura 7 Lenguajes específicos de dominio (DSLs) ampliamente utilizados.	37
Figura 8 Arquitectura de metamodelos definida por la OMG	39
Figura 9 Estereotipo	39
Figura 10 Valor Etiquetado	40
Figura 11 Patrón Lonely Compound. (Robert et al., (2009)).....	41
Figura 12 Patrón Superfluous Child. (Robert et al., (2009))	41
Figura 13 Esquema de Model-Driven Architecture. (García-Díaz et al., (2010))	43
Figura 14 DSL y Perfil UML representando los mismos objetos	45
Figura 15 Escenario de la cadena productiva que involucra responsables (flechas grandes), actividades (estrellas) y TRUs (cajas).	47
Figura 16 Diagrama de clases UML del metamodelo para sistemas de trazabilidad.....	48
Figura 17 Patrones de Comportamiento (Bechini et al., (2005))	49
Figura 18 Perfil UML del metamodelo presentado	52
Figura 19 Perfil UML simplificado mediante el uso de patrones de optimización.....	60
Figura 20 Ejemplo de aplicación del metamodelo	64
Figura 21 Ejemplo de aplicación del perfil UML	65
Figura 22 Perfil SysML para sistemas IoT (Costa et al., (2016))	68
Figura 23 Perfil UML (Corredor et al., (2012))	69
Figura 24 Interacción del paquete IoT con los paquetes Calidad y Trazabilidad	70
Figura 25 Perfil IoT propuesto	71

Figura 26 Planilla de campo para control de las actividades en el apiario. (Dini & Bedascarrasbure, 2011)	75
Figura 27 Diagrama de flujo de la miel y cera dentro de la sala de extracción.....	76
Figura 28 Aplicación del PerfilTrazabilidadCalidad	79
Figura 29 Ejemplo de aplicación en actividad de Curación	81
Figura 30 Ejemplo de aplicación en actividad de Multiplicación.....	82
Figura 31 Ejemplo de aplicación en actividad de Colocación de Alzas	83
Figura 32 Ejemplo de aplicación en actividad de Cosecha	84
Figura 33 Ejemplo de aplicación en actividades de Desoperculado, Envasado, Adquisición y Suministro	86
Figura 34 Aplicación de los perfiles PerfilTrazabilidadCalidad y PerfilIoT	88
Figura 35 Ejemplo de aplicación de ambos perfiles a una actividad de Cosecha con tecnologías IoT	89

ÍNDICE DE TABLAS

Tabla 1 Comparación entre diferentes tipos de trazabilidad.....	22
Tabla 2 Diferencias entre Código de Barras y RFID.....	30
Tabla 3 Definición de Estereotipo EntidadTrazable.....	51
Tabla 4 Definición de Estereotipo Lote	53
Tabla 5 Definición de Estereotipo Actividad	53
Tabla 6 Definición de Estereotipo Responsable.....	53
Tabla 7 Definición de Estereotipo Movimiento	53
Tabla 8 Definición de Estereotipo Alteración	54
Tabla 9 Definición de Estereotipo División.....	54
Tabla 10 Definición de Estereotipo Union.....	54
Tabla 11 Definición de Estereotipo Suministro	54
Tabla 12 Definición de Estereotipo Adquisicion	55
Tabla 13 Definición de Estereotipo RequisitoDeCalidad.....	55
Tabla 14 Definición de Estereotipo RCCualitativo	55
Tabla 15 Definición de Estereotipo RCNumerico.....	55
Tabla 16 Definición de Estereotipo Identificador	56
Tabla 17 Definición de Estereotipo TipoEstado	56
Tabla 18 Listado de objetos de la cadena apícola	79

1. INTRODUCCIÓN

En los últimos años la trazabilidad ha ganado terreno en casi todas las áreas productivas, dejando de ser exclusiva de la industria de alimentos, no solo por su importancia a la hora de disminuir riesgos y evitar pérdidas substanciales, sino también por las responsabilidades que conlleva para con los actores de toda la cadena productiva. El incremento de exigencias y controles por parte de la Unión Europea y de Estados Unidos ha tenido también un papel fundamental en la expansión del uso de este concepto. Esta expansión ha llegado a abarcar áreas como la industria automotriz, aeronáutica, distribución logística, electrónica de consumo, construcción, desarrollo de software, farmacéutica, salud, entre otras. Además, la trazabilidad se ha convertido en un área que involucra aspectos sociales, tecnológicos y administrativos, generando que, por culpa de esta interdisciplinariedad, no exista un marco teórico común en el área de trazabilidad, lo que afecta su desarrollo e implementación en industrias complejas como la alimentaria.

La incorporación de sistemas informáticos de trazabilidad se ha convertido en una pieza clave para el éxito de cualquier empresa que desee incorporarse a los grandes mercados globalizados. Estos sistemas deben estar respaldados en una infraestructura basada en Tecnologías de la Información y Comunicación que permitan la rápida identificación de productos, y la captación y distribución de la información para su posterior procesamiento.

La importancia que tienen estos sistemas es directamente proporcional a la complejidad del dominio al que se apliquen. Las diferencias entre los distintos tipos de productos a los cuales se aplica el concepto de trazabilidad hacen casi imposible la estandarización y generalización de tecnologías y sistemas informáticos que abarque la totalidad de ellos. Incluso, dentro de una misma área, se dificulta lograr esta generalización, como ocurre, por ejemplo, en la industria alimentaria donde identificar una cabeza de ganado es mucho más sencillo que identificar un grano de maíz, lo que define, casi inevitablemente, la resolución del sistema de trazabilidad que se pretenda desarrollar. Por esto es necesaria la creación de modelos por áreas productivas y de esta forma lograr una “*estandarización específica*” para el área en cuestión. Además, otro de los grandes problemas para la trazabilidad es

la gran diferencia de mecanismos y conceptos aplicados entre los diferentes eslabones de la cadena productiva, llevando esto a generar grandes conflictos para el intercambio adecuado de la información recabada durante los diferentes procesos.

Estas dificultades hacen que la tarea de crear sistemas de trazabilidad para la industria de alimentos sea una actividad compleja cuando se abarca más de un eslabón productivo y donde cada eslabón se rige mediante distintos criterios o conceptos de lo que implican la calidad y la trazabilidad. Las cantidades y tipos de información requerida son diferentes si se toman en cuenta distintos mercados, distintos actores, y, lógicamente, distintos productos. Además, la granularidad (tamaño de la mínima unidad de un producto que se puede identificar) de cada sistema varía de acuerdo al producto en cuestión y define la calidad de la información que se puede obtener. Esto recae en la necesidad de facilitar el proceso de desarrollo de los sistemas de trazabilidad y de control de calidad, y que se puedan adaptar fácilmente a diferentes dominios o diferentes requisitos dentro de un mismo dominio.

Verdouw et al. (2016), clasifica a la mayoría de los problemas antes mencionados en “*complejidades de la cadena de suministro de alimentos*”:

- Complejidad de la red de alimentos: engloba los problemas generados debido a la gran variedad de interacciones que se dan dentro de la red, como por ejemplo empresas multinacionales interactuando con pequeños productores, lo que genera que, por ejemplo, un objeto particular tenga diferentes representaciones debido a los distintos intereses y propósitos de cada compañía.
- Complejidad de los objetos de la cadena: involucra los objetos que se manipulan en determinados puntos de la cadena y que dependen del tipo de alimento en cuestión. Por ejemplo, en agricultura se manejan semillas, alimentos para ganado, fertilizantes, etc. Y esto afecta directamente a la granularidad de cualquier sistema de trazabilidad que se desee implementar.
- Complejidad de los procesos de los alimentos: toma en cuenta la dificultad que representa la gran cantidad de procesos distintos que puede tener un mismo tipo de producto agrícola para obtener diferentes productos finales, o lo diferente que puede

ser el mismo proceso si se toman en cuenta diferentes productos, lo que implica una gran variedad de procesos y tipos de flujo de alimentos durante los mismos.

- Complejidad del control de los procesos: se basa en la dificultad que representa la cambiante demanda de producto, los cambios de preferencia de los consumidores y la turbulencia de los mercados, lo cual deriva en una gran necesidad de optimización y re-planificación continua de los procesos.

Actualmente, la cadena de suministro es tratada, por lo general, separadamente del sistema de trazabilidad, sin considerar que ambas coexisten y a pesar que cada eslabón pueda funcionar de forma independiente, el impacto final no es mutuamente excluyente. Es por ello, y por los problemas antes mencionados, que el diseño de un sistema de trazabilidad requiere un replanteamiento a fondo y la reorganización de toda la cadena de suministro; para ello, como se dijo antes, se necesita unificar las bases teóricas desde las cuales se generen los diseños compatibles entre sí que permitan compartir información; esto es imprescindible, dado que todos los actores relacionados con la cadena de suministro necesitarán implementar la trazabilidad o tendrán dificultades para mantenerse en el negocio.

Otro problema del desarrollo de software de trazabilidad, es el elevado costo que estos tienen debido a la “*personalización*” que requieren para cada área en cuestión, lo que pone la situación aún más difícil para los pequeños productores de alimentos y empresas de procesamiento, ya que carecen de la capacidad financiera para afrontar los costos, la información de trazabilidad adecuada y conocimientos suficientes para ponerla en práctica; por esto la responsabilidad recae en la comunidad académica, que es la encargada de simplificar los métodos de desarrollo; y los gobiernos, para generar los mecanismos pertinentes para que este primer actor de la cadena conozca y pueda implementar la trazabilidad de acuerdo a sus capacidades y necesidades.

La necesidad de personal capacitado también representa un desafío para la integración de este tipo de sistemas, ya que muchas veces debido a las limitaciones técnicas y la falta de conciencia, se hacen necesarios programas de capacitación y educación eficaces para aumentar la capacidad técnica y el nivel de conciencia de los participantes; lo que a su vez disminuye

la resistencia inicial a la adopción de estos sistemas de trazabilidad, la cual muchas veces es generada por falta de voluntad o por falta de conocimiento de los beneficios.

Para solucionar muchas de estas problemáticas se puede considerar a los Domain-Specific Languages (DSL) que brindan la posibilidad de desarrollar, como su nombre lo indica, lenguajes adaptados al dominio en cuestión. Al estudiar a los DSLs aparecen, consecuentemente, los UML-Profiles (Perfiles UML) que se pueden considerar como un subtipo de DSL siempre que se cumplan ciertas características del dominio. Estos últimos tomaron fuerza cuando la preocupación más importante dejó de ser el poner énfasis en la definición de la estructura y calidad del código final, y paso a ser el dedicar más tiempo y esfuerzo al modelado del sistema.

1.1 Objetivos

Lo antes mencionado ha motivado en gran medida la realización de esta tesis, proponiendo lograr como objetivo final de la misma un DSL para el dominio de trazabilidad de alimentos que facilite el desarrollo de aplicaciones y la adaptación de las mismas a las complicaciones que puedan surgir dentro de los diferentes eslabones del propio dominio.

En cuanto a los objetivos específicos de la presente tesis, se pretende:

- Realizar un estudio comparativo de los modelos y patrones de diseño existentes en el dominio de la trazabilidad de productos de la industria de alimentos y las restricciones inherentes al mismo.
- Descubrir las primitivas de modelización necesarias para mejorar el diseño de software de trazabilidad alimentaria.
- Desarrollar una especialización de un lenguaje de modelización estándar (por ejemplo UML) para satisfacer los requisitos del dominio.
- Realizar un ejemplo de aplicación del lenguaje obtenido en el subdominio de la trazabilidad de productos apícolas.

El primer objetivo permitirá establecer los requerimientos básicos a tener en cuenta como punto de partida para el desarrollo del DSL, identificando las posibles mejoras a los modelos existentes, tanto desde el punto de vista conceptual como el estructural. Llegando así a abordar el segundo ob-

jetivo. Para esto se tendrán en cuenta áreas temáticas como los sistemas de producción, ciclo de vida de los productos, revisión de hardware, modelos de datos de trazabilidad de productos.

Tomando como punto de partida los resultados obtenidos de los primeros dos objetivos, se pretende llevar a cabo el desarrollo propuesto en el tercer objetivo detallando las características necesarias para el Lenguaje Especifico de Dominio que se desarrolle.

El ámbito de desarrollo laboral del Tesista se encuentra enmarcado en el diseño y desarrollo de tecnologías de monitoreo para la industria apícola, la cual, no se ha visto exenta del incremento en la demanda de controles de calidad y sistemas de trazabilidad, pero que tampoco estas demandas han sido acompañadas por la aplicación de tecnologías acordes. Por lo tanto, dentro del dominio general de la producción alimenticia, se ha elegido el subdominio de los productos apícolas, principalmente miel, para utilizar el DSL desarrollado a modo de ejemplo de aplicación, para comprobar su utilidad, realizando un análisis de las ventajas y/o desventajas encontradas, y de esta forma cubrir el cuarto objetivo específico planteado en este trabajo.

1.2 Estructura de la Tesis

La organización general de la estructura de esta Tesis se rige de acuerdo a los objetivos planteados en la sección anterior.

En primera instancia, se llevara a cabo una descripción del estado del arte en el área de la trazabilidad, la calidad y la producción de alimentos, presentando las bases conceptuales del dominio de aplicación al que se orienta el desarrollo.

Luego se realizará un estudio comparativo de las características más importantes de los modelos de datos de los sistemas de trazabilidad existentes; y un repaso de las tecnologías de registro e identificación de productos más utilizadas actualmente. Dentro de esta última categoría se pretende hacer especial hincapié en la Internet de las Cosas o “*Internet of Things (IoT)*”. Esta tecnología está brindando muchas ventajas en los últimos años en muchas áreas productivas, una de las cuales es la trazabilidad. Las principales ventajas se ven en la identificación de unidades, la transmisión de la información y su procesamiento en tiempo real.

Se realizará, también, una revisión de los conceptos más importantes en cuanto a DSLs y Perfiles UML, analizando sus ventajas y desventajas; y

su importancia para el desarrollo de sistemas utilizando arquitectura MDA (Model-Driven Architecture).

Se llevará a cabo una descripción detallada del lenguaje específico diseñado, incluyendo las restricciones del mismo. Para esto primero se presentará un diagrama de clases del metamodelo en el cual se basará el DSL incluyendo una descripción de las clases que lo componen y la relación entre ellas.

Con el fin de comprobar la utilidad del DSL se presentará un capítulo donde se pretende presentar la realidad del sector productivo apícola en la argentina, y como se conforma su ciclo productivo; haciendo especial hincapié en la producción de miel y el proceso que esta sigue desde la colmena hasta el consumidor final, para luego aplicar a este proceso el DSL desarrollado.

También se agregará una sección de conclusiones sobre la totalidad del desarrollo de este trabajo donde se analizarán los resultados obtenidos comentando pros y contras del DSL diseñado, y presentando, para finalizar, una sección de trabajos futuros donde se plantearan los posibles trabajos que se pueden desprender de lo desarrollado en esta tesis.

2. TRABAJOS RELACIONADOS

Desde el punto de vista de la trazabilidad existen varios trabajos que se centran en la importancia de crear un modelo de datos lo suficientemente preciso para minimizar la cantidad de información necesaria para llevar a cabo una trazabilidad efectiva, y disminuir los problemas de interoperabilidad que existen entre los eslabones productivos.

2.1 Sistemas de trazabilidad en general

El trabajo realizado por *Tursi et al.* (2007) aborda el problema de muchos sistemas de trazabilidad que fracasan en trazar correctamente el ciclo de vida del producto, al realizar la recolección de información desde fuentes heterogéneas, lo que generalmente se debe a problemas de interoperabilidad, ya que la información frecuentemente se encuentra dispersa dentro de las organizaciones. Como solución presenta un sistema centrado en el producto, ya que este es el único elemento en común a toda la cadena productiva y todo el agregado de valor se ve reflejado directamente en él. Para esto propone un modelo de datos donde toda la información técnica del producto se encuentra inmersa en el producto mismo.

Khabbazi et al. (2009), plantean los inconvenientes más comunes de la trazabilidad en la industria de la manufactura en general, tomando como ejemplo la fabricación de automóviles y sus partes, y cómo la información de trazabilidad se almacena en diferentes sistemas y en fuentes de datos heterogéneas; y luego presenta un diagrama Entidad-Relación para modelizar los datos de un sistema de trazabilidad centrado en el producto, que pretende resolver esos problemas e incorpora características de calidad de los productos. Se comprueba que el modelo presentado hace que el pasaje desde el análisis del sistema hacia la implementación sea más rápido y menos complicado. Se propone utilizar sistemas de información integrados con funciones de realimentación en tiempo real para mejorar el flujo de la información, lo que a su vez brinda ventajas en cuanto a la personalización y al control en tiempo real del flujo de material, mediante la captación de indicadores de calidad y de procesos. Por otro lado también propone la utilización de una base de datos integrada que facilita el acceso a la información mediante interfaces adaptadas a tal fin.

Rinconballesteros (2016) realiza una revisión de las complicaciones más comunes que enfrentan los desarrolladores de sistemas de trazabilidad, principalmente debido a problemas de adopción de los sistemas por parte de las empresas y/o actores en general, ya sea por falta de educación al respecto, o por falta de recursos financieros que permitan la implementación de la infraestructura necesaria para montar dichos sistemas. También plantea los pilares básicos para el éxito de un sistema de trazabilidad y estos son: regulación, seguridad y calidad, preocupación social, preocupación económica y preocupación tecnológica.

2.2 Sistemas de trazabilidad alimentaria

Bechini et al. (2008), presentan un diagrama de clases UML definiendo uno de los modelos de datos más aceptados en el área de trazabilidad de productos alimentarios al día de hoy, realizando una descripción de lo que los autores llaman “*Patrones de Comportamiento*” genéricos a cualquier cadena de producción. Además realiza un repaso de las tecnologías más utilizadas para la identificación del lote y los mecanismos de intercambio de información que se utilizan actualmente dentro de la cadena de suministro, promoviendo el uso de XML (eXtensible Markup Language) y SOAP (Simple Object Access Protocol) como tecnologías de comunicación que permiten dicho intercambio de manera segura y facilitando el acceso de los diferentes actores. Los autores realizan un análisis de resultados desarrollando algunos ejemplos de uso del modelo obtenido en el dominio de la trazabilidad de alimentos, puntualmente en lechería y vitivinicultura. El modelo de datos presentado por *Bechini et al.*, ha servido de base para el metamodelo considerado en el presente trabajo, realizando al mismo algunas modificaciones para que se adecue a los objetivos de esta tesis.

Manikas & Manos (2009) basan su trabajo en varios estándares internacionales ISO que definen los requerimientos mínimos de información para diseñar e implementar un sistema de trazabilidad de alimentos eficiente. Además presentan mecanismos de manejo de la información mediante la estandarización y tipificación de los mensajes que permiten el almacenamiento y comunicación de la misma, considerando que esto es parte fundamental de la efectividad de un sistema de trazabilidad. Con estos conceptos se presenta un modelo de datos que consiste en tres fases, acordes a las características del dominio de productos lácteos considerado para este tra-

bajo. Las tres fases son: el ambiente natural, que considera tanto al animal como a la granja donde se produce la leche; la transformación, que representa todos los procesos que sufre la leche para obtener los diferentes productos lácteos; y la distribución, que involucra principalmente el almacenamiento en los locales de venta y el traslado del producto a los depósitos. El modelo presentado se orienta a una plataforma web que facilita el intercambio seguro y eficiente de información mediante la utilización de una base de datos central. También se propone la implementación en PML (Physical Markup Language), que es una tecnología basada en XML que fue creada con la intención de ser un estándar generalizado para describir objetos físicos en sistemas de monitoreo remoto.

Ruiz-García et al. (2010) parten de un análisis sobre el tipo y la cantidad de información necesarios en trazabilidad, luego realizan una revisión de las tecnologías de recolección de información y monitoreo logístico de productos mediante GIS (Geographic Information Systems), proponiendo una recolección de datos automatizada para disminuir los errores; y, a diferencia de otros estudios, propone que la trazabilidad no sea “*one-up/one-down*” como generalmente se realiza, sino que sugiere un modelo en el que toda la información sea accesible para todos los actores en cualquier momento. Dicho modelo está basado en una arquitectura orientada a servicios (SOA) y la utilización de Servicios Web.

El trabajo de *Gemesi* (2010) consiste en una tesis de maestría, que en primer lugar realiza un estudio de la posibilidad de establecer una trazabilidad electrónica completa a través de la cadena de suministro de granos, tratando de determinar las mejores prácticas para llevarlo a cabo. Para esto, el autor, realiza un análisis de la dinámica de las unidades trazables, del diseño de un modelo de información que integra trazabilidad dentro del concepto del modelado de datos centrado en el producto y también, los métodos más comunes de intercambio de información para lograr una trazabilidad electrónica en toda la cadena productiva. En segundo lugar la tesis explora como los GIS se pueden usar para rastrear o trazar granos a nivel de la granja, asociando información de trazabilidad con localizaciones geográficas. Para esto desarrolla una base de datos geográfica basada en GIS. Esto propone un modelo de datos de referencia basado en la trazabilidad desde la granja que detecta la presencia de organismos genéticamente modifica-

dos (GMO) y cumple con las regulaciones actuales para la producción de granos.

2.3 Trazabilidad con DSLs

Panetto & Petin (2005) describen formalmente un framework para la automatización de sistemas de producción. Aquí se utilizan los estereotipos UML para facilitar el entendimiento de los sistemas en las etapas tempranas del proceso de ingeniería de sistemas.

El trabajo presentado por *García-Díaz et al.* (2010), utiliza los principios de la MDE (Model-driven Engineering o Ingeniería Basada en Modelos) para crear un framework denominado Talisman MDE. En este framework se combinan los principios de la MDA y las SF (Software Factories o Fábricas de Software) donde los autores toman las mejores características de cada metodología, utilizando Perfiles UML y DSLs respectivamente. Luego el framework es puesto a prueba creando aplicaciones para el control de la trazabilidad alimentaria en el área de lácteos. Las aplicaciones fueron creadas exclusivamente para cada lechería participante del estudio mediante el uso únicamente de modelos y sin programación adicional.

3. PRODUCCIÓN DE ALIMENTOS Y TRAZABILIDAD

Los productores y comercializadores de alimentos tienen una responsabilidad muy grande con los consumidores en cuanto a la inocuidad y seguridad de sus productos. Las enfermedades transmitidas por alimentos son un problema que toma cada vez más relevancia debido a la globalización del mercado alimentario. Los participantes de la cadena de abastecimiento deben garantizar que los productos que suministran son aptos para el consumo y no representan ningún peligro para la salud de los consumidores.

Los factores que favorecen la contaminación de los alimentos pueden surgir en cualquier eslabón de la cadena de producción y comercialización de los alimentos, por lo que todos los participantes son responsables del buen estado de los mismos, incluso los consumidores. Algunos de los factores pueden ser malas prácticas agrícolas, falta de higiene de los manipuladores, almacenamiento incorrecto del producto o materia prima, falta de cocción, etc.

En años recientes se ha incrementado el uso de agroquímicos y de Organismos Genéticamente Modificados (OGM) en aras de incrementar los rindes de las producciones agropecuarias, esto no solo ha generado importantes avances en muchos campos productivos, sino que también ha provocado una gran preocupación debido al desconocimiento, en muchos casos, de los efectos secundarios y las cuestiones morales, éticas y económicas que implican. Esto ha derivado en la demanda creciente, por parte de un sector de la población, de productos “*libres de contaminación*”, impulsando, de esta forma, aún más la necesidad de conocer la historia de los alimentos que llegan a la mesa.

La trazabilidad surge como una de las herramientas para prevenir, identificar y/o corregir la aparición de algún tipo contaminación de los alimentos, en cualquiera de sus etapas de producción; y/o para demostrar las técnicas de producción y los ingredientes que se utilizaron en los diferentes productos.

En este capítulo se dará una visión global de cómo se conforma un sistema de trazabilidad de alimentos, tomando en cuenta los aspectos básicos de calidad y trazabilidad, para luego abarcar los aspectos más complejos necesarios para el desarrollo de este trabajo.

3.1 Responsabilidades

Una de las consecuencias de la aplicación de sistemas de trazabilidad es la identificación de responsables para cada actividad que se realice en la cadena de producción.

Slorach (2002) plantea 3 niveles de responsabilidad en cuanto a la generación y preservación de la inocuidad de los alimentos:

- Primaria: Recae en los productores, procesadores y comerciantes de alimentos. Son los encargados de garantizar que los alimentos que producen y manipulan sean inocuos y respeten las características básicas que las leyes alimentarias impongan.
- Secundaria: Son las autoridades de supervisión. Su tarea es establecer las normas de inocuidad y asegurar que los sistemas de control sean adecuados y se apliquen de acuerdo a dichas normas. Realizan actividades de control directo (sobre importaciones por ej.) y brindan información sobre alimentos.
- Terciaria: Abarca a los consumidores. Son responsables de la higiene, almacenamiento y preparación de los alimentos en su hogar.

Esta clasificación plantea la necesidad de disponer de los medios y herramientas para facilitar la comunicación, la información y el control de los productos y actividades a cargo de los distintos responsables, a fin de minimizar la ocurrencia de los factores causantes de contaminación de los alimentos.

Los responsables primarios “*deben operar según las buenas prácticas de manufactura/higiene/agricultura*”. Partiendo de esto se plantea que todas las actividades deben ser evaluadas en busca de posibles puntos de peligro, para luego analizar los riesgos asociados. Esto resulta en la necesidad de identificar los puntos críticos de control y de elaborar un sistema para controlar dichos puntos, tomando en cuenta, también, la documentación necesaria, los procedimientos de comprobación y las medidas correctivas. El conjunto de estas medidas se conoce como “*Análisis de los Peligros y Puntos Críticos de Control*” (HACCP por su sigla en inglés). (*Slorach, 2002*)

El planteo de este sistema toma en cuenta la dificultad de poder inspeccionar la calidad en el producto final, ya que es muy difícil reacondicio-

nar un producto terminado para que vuelva a ser apto para el consumo humano; de esta forma se favorece el control en cada una de las etapas de la producción disminuyendo la cantidad de productos rechazados por falta de inocuidad al final de la cadena y las consecuentes pérdidas económicas que esto ocasionaría.

3.2 Calidad y Trazabilidad

Las Normas ISO 9000-2000, si bien aplican a todos los productos y no solo a los alimentos, definen a la calidad como *“el grado en el que un conjunto de características inherentes a un objeto (producto, servicio, proceso, persona, organización, sistema o recurso) cumple con los requisitos”* (International Organization for Standardization, 2015).

La Comisión Europea, por su parte, es una institución de la EU (Unión Europea) que se encarga de proponer las nuevas legislaciones y ejecutar las políticas de Europa (European Commission, 2002). Tanto la Comisión Europea como el Codex Alimentarius definen la trazabilidad como *“la habilidad para rastrear y seguir el movimiento de un alimento a través de una etapa de producción, procesado o distribución específica”*.

El Codex Alimentarius contiene normas alimentarias que protegen la salud de los consumidores y fomenta prácticas leales de comercio. El mismo fue creado por la FAO (Organización de las Naciones Unidas para la Alimentación y la Agricultura) y la OMS (Organización Mundial de la Salud), y se encuentra administrado por la comisión que lleva su mismo nombre (Codex Alimentarius Commission, 2006). Los *“Principios para la rastreabilidad/rastreo de alimentos”* del Codex Alimentarius comprenden el contexto, el fundamento, el diseño y la aplicación de la rastreabilidad/rastreo de productos como una herramienta que puede ser utilizada por la autoridad competente en un sistema de inspección y certificación de alimentos. Entre los fundamentos más importantes que se plantean en estos principios, se destaca el hecho de que la trazabilidad como herramienta debería mejorar la eficacia y la eficiencia de las actividades necesarias para los sistemas de inspección y certificación de alimentos; así también se resalta que la rastreabilidad en sí misma no es suficiente para garantizar la inocuidad de los alimentos, a menos que se utilice con otras medidas y requisitos apropiados a tal fin. También se expresa que la trazabilidad debe contribuir a facilitar el comercio sobre la base de una descripción exacta de

los productos y proteger al consumidor de prácticas engañosas de comercio. En todos los casos, se debe justificar a la trazabilidad en el contexto del sistema de inspección y certificación de alimentos, por lo que el ámbito y la medida de aplicación de la herramienta deberían ser coherentes con la necesidad del contexto. En cuanto a cuestiones de diseño, el Codex plantea que la trazabilidad se puede llevar a cabo en todas las etapas o en etapas específicas, de acuerdo a los objetivos del sistema; además se especifica que las características del sistema de inspección y certificación deberían ser transparentes y ponerse a disposición de las autoridades gubernamentales de control cuando así lo requieran. En cuanto a la aplicación de los sistemas de trazabilidad, el Codex establece varias condiciones básicas necesarias para tal fin, entre las que se menciona la no restricción del comercio (más de lo necesario), la viabilidad técnica y económica de los sistemas de trazabilidad, la evaluación de los riesgos de aplicación, la evaluación “*caso por caso*” de la factibilidad de implementación y, la posibilidad de que países importadores colaboren con la implementación de la trazabilidad en los casos en que los países exportadores en vías desarrollo no posean las condiciones para cumplir los objetivos de dichos sistemas. (Codex Alimentarius Commission, 2006).

Por su parte el Servicio Nacional de Sanidad y Calidad Agroalimentaria (SENASA, 2015) y la Administración Nacional de Medicamentos, Alimentos y Tecnología Médica (ANMAT, 2017) definen la trazabilidad como “*el conjunto de aquellos procedimientos preestablecidos y autosuficientes que permiten conocer el histórico, la ubicación y la trayectoria de un producto o lote de productos a lo largo de la cadena de suministros en un momento dado, a través de unas herramientas determinadas*”.

Todas estas instituciones contemplan como requisito operativo fundamental los conceptos de *trazabilidad hacia atrás y hacia adelante*, así también como la *trazabilidad interna*. El primero consiste en conocer los ingredientes que forman parte de un producto, envases y otros materiales utilizados, además de identificar a los proveedores, también es conocido como “*rastreabilidad*” y es especialmente útil para detectar el origen de posibles problemas en calidad; el segundo permite conocer dónde se ha vendido o distribuido un lote determinado de un producto alimenticio y para qué se ha utilizado, corresponde a la “*trazabilidad*”, propiamente dicha, y es útil principalmente cuando se deben reclamar productos de los eslabo-

nes siguientes de la cadena debido a la detección de algún defecto; por su parte la trazabilidad interna, o de proceso, corresponde a la identificación de productos y procesos dentro de la propia empresa (ANMAT, 2017). En la Tabla 1 se resumen las principales diferencias entre los tres tipos de trazabilidad mencionados

En términos prácticos, el estándar del GS1 representa uno de los estándares más ampliamente utilizados como herramientas para el control de la calidad y la trazabilidad. El mismo fue creado en el año 2005 por una gran cantidad de expertos de diferentes países y compañías, y en su definición utiliza la definición de trazabilidad de la norma ISO 9000-2000. Este estándar proporciona “*las tuercas y tornillos*” que otras iniciativas, como ISO o Codex Alimentarius, requieren o recomiendan, ya que los mismos dictan la recopilación específica de información para realizar el seguimiento de los productos hacia el origen y hacia el destino, pero ninguno especifica la forma en que esto debe hacerse. En este punto, el GS1 proporciona estándares para la identificación automática con códigos de barras (EAN-UCC), para el comercio electrónico (utiliza dos lenguajes estándares de mensajes complementarios: GS1 EANCOM y GS1 XML), para la sincronización global de datos y para la identificación con RFID. (GS1, 2018)

Trazabilidad hacia atrás	Trazabilidad interna	Trazabilidad hacia adelante
<ul style="list-style-type: none"> • De quién se reciben los productos • Qué se ha recibido exactamente • Cuándo • Qué se hizo con los productos cuando se recibieron 	<ul style="list-style-type: none"> • Cuándo los productos se dividen, cambian o mezclan • Qué es lo que se crea • A partir de qué se crea • Cómo se crea • Cuándo • Identificación del producto final 	<ul style="list-style-type: none"> • A quién se entrega • Qué se ha vendido exactamente • Cuándo

Tabla 1 Comparación entre diferentes tipos de trazabilidad

Una de las ventajas de la trazabilidad en la cadena productiva es, como se ha dicho, que permite la identificación y el aislamiento de los productos contaminados y la implementación de medidas correctivas al momento de detectar la incidencia de algún evento peligroso, esto resulta en la prevención de los eventos que amenacen la inocuidad de los alimentos disminuyendo el impacto mediante la identificación precisa de los productos o lotes de productos afectados, y especificando: ¿Qué ocurrió?, ¿Cómo y Cuándo ha ocurrido? y ¿Quién ha sido responsable?. Esto genera valor

agregado a la calidad general de los productos alimenticios al favorecer el cumplimiento de las normas de calidad y las expectativas de los consumidores. A pesar de lo antes mencionado, debe considerarse a la trazabilidad como una herramienta complementaria a otras que buscan el aseguramiento de la calidad y el manejo seguro de los productos; por lo que la trazabilidad por sí misma no es una condición suficiente para generar seguridad a los procesos involucrados en la cadena de producción. (Opara, 2003) (Evizal et al., (2016))

Una de las maneras más efectivas de implementar seguridad alimentaria y trazabilidad es el monitoreo del flujo de productos, su calidad y los parámetros de los procesos a través de la cadena productiva y unirlos a cada transición en el estado de esos productos (Azram & Atan, 2011). Uno de los mayores desafíos para la industria de productos perecederos, como la industria alimentaria, es asegurar que los productos mantienen continuamente la cadena de frío durante los distintos procesos desde el productor al consumidor.

Por su parte las regulaciones de la Unión Europea y los Estados Unidos, así también como las normas ISO, definen como fundamental a la trazabilidad *Un Paso Arriba/Un Paso Abajo (one-up/one-down traceability)* para cada eslabón de la cadena de suministros. Por lo que de esta forma cada compañía debe recopilar la información de la compañía que le suministró los productos, la que los transportó desde y hacia ella, y a quien fueron vendidos (European Commission, 2002) (International Organization for Standardization, 2015).

3.3 Modelo de datos de un sistema de trazabilidad

En la literatura existente se han descrito varios modelos de datos para el manejo de la información en la trazabilidad.

Uno de los modelos propuestos basa su estructura en 4 aspectos (Jansen-Vullers et al., (2003)):

- La integridad física del lote: determina la resolución o precisión del sistema de trazabilidad.
- La colección de datos: representada por los datos de trazado y de procesos.
- La identificación de productos y los procesos asociados.

- El reporte de los datos obtenidos desde el sistema.

Otro de los modelos más utilizados está enfocado en la calidad y se basa en la definición de elementos fundamentales como el *Traceable Resource Unit* (TRU) y la *actividad primitiva*. El primero se define como un elemento único cuyas características son únicas desde el punto de vista de la trazabilidad; y la *actividad primitiva* se define como una operación básica de la cadena productiva (Kim et al., (1995)). Las TRU representan las entidades más pequeñas, identificables unívocamente, correspondientes a objetos físicos de la cadena de valor, por lo que el éxito de los sistemas de trazabilidad se basa en trazar y rastrear estas entidades a través de dicha cadena. Determinar cuáles son las Entidades Trazables es uno de los mayores retos al momento de crear modelos dinámicos de trazabilidad para una cadena de valor determinada, por ej., si el batch de producción es muy grande, la precisión del sistema de trazabilidad se verá comprometida. (Manikas & Manos, 2009). Un sistema de trazabilidad debe trazar tanto a los productos como a las actividades (Kim et al., (1995)) (Bechini et al., (2005)) (Ruiz-García et al., (2010)).

Gemesi (2010), por su parte, subclasifica a las TRU en dos tipos: los batch de producción y las unidades de intercambio. Los primeros son unidades trazables a las que la materia prima y los ingredientes son incorporados antes de ser transformados en nuevas unidades de intercambio. Una unidad de intercambio es un ítem listo para el transporte y/o el almacenamiento que debe ser gestionado a través de la cadena de suministro. Una unidad de intercambio consiste en una o más unidades trazables.

Ruiz-García et al. (2010), determinaron cuatro requerimientos básicos que se deben cumplir para poder desarrollar el modelo de un sistema de trazabilidad:

- Cada actor en la cadena de valor debe ser responsable solo de su tarea.
- Cada elemento en la cadena de suministro es un punto de inicio para una consulta.
- Los datos deben ser remitidos hasta el origen de la cadena de valor.
- La información procesada de cada elemento de la cadena debe estar disponible.

En primer lugar, el modelo debe ser diseñado para cada empresa que forma parte de la cadena productiva, lo que puede ser definido como trazabilidad interna. La información relacionada a los procesos y materiales que forman al producto final se obtiene gracias a un buen modelo de trazabilidad interna, por lo que es fundamental una buena estructura y un oportuno registro de las propiedades de entrada/salida para un control dinámico de los procesos de producción. Por otro lado, el modelo de datos de la trazabilidad también debe ser definido para toda la cadena de producción, basándose en la información obtenida de cada eslabón de la cadena. Este nivel de modelado de datos permite realizar una trazabilidad basada en la TRU desde el origen hasta el consumidor final. (Gemessi, 2010)

Tomando en cuenta todo lo anterior se definen dos formas de administrar la información de trazabilidad de la cadena (Gemessi, 2010):

- Se almacena la información localmente y solo se comparte la identificación del producto con los demás participantes de la cadena. Lo que genera que solo se pueda trazar los productos un paso a la vez.
- La información sigue al producto durante todo el recorrido a través de la cadena.

Para que la trazabilidad al nivel de las TRU sea efectiva se debe implementar un sistema que administre eficientemente los recursos, debido a que la información está almacenada en fuentes muy heterogéneas en las diferentes compañías. Por esto, el modelo de datos de trazabilidad debe proveer una representación integral de los contenidos de estas diversas fuentes y suministrar un punto de acceso único a la información de los productos a través de su ciclo de vida, mediante el mapeo de las estructuras de las fuentes de los datos y de su contenido (Khabbazi et al., (2009)) (Tursi et al., (2007)).

Varios autores consideran que el modelado de datos centrado en el producto (Figura 1) es la solución a los problemas de interoperabilidad que se presentan en la implementación de trazabilidad en la cadena completa de la industria alimentaria. Este modelado supone la necesidad de trazar y rastrear información relevante al ciclo de vida del producto (Gemessi, 2010).

Además de centrar el modelo en el producto, otros autores proponen tener en cuenta 3 características fundamentales para el modelado: la ubica-

ción, el estado y la calidad del producto. La ubicación, desde el punto de vista de la logística, corresponde a la localización geográfica, fecha y hora, y estado logístico del producto en la cadena de producción. El estado se define en función de los procesos que se aplican en el flujo físico de productos, considerando las condiciones y las etapas del proceso durante la producción y distribución. La calidad se toma en cada etapa de la cadena productiva donde esta pueda verse afectada. (Manikas & Manos, 2009)

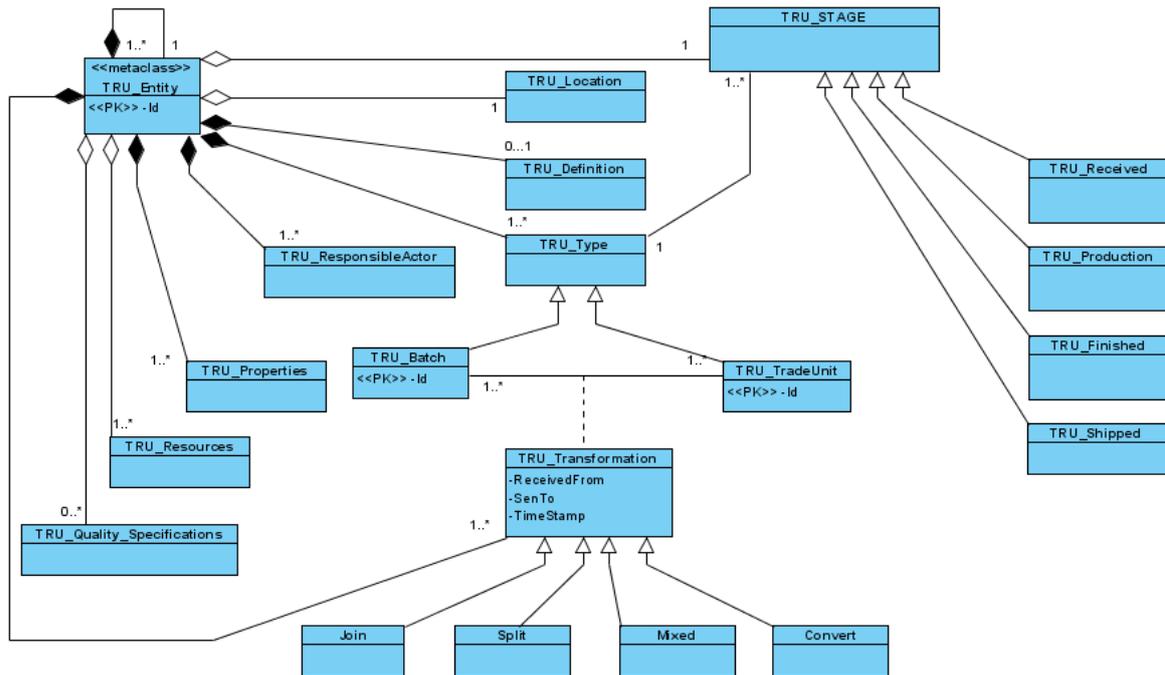


Figura 1 Modelo de datos centrado en el producto para trazabilidad (Gemesi, 2010)

Bechini et al. (2008), plantea que el modelo de datos debe ser lo suficientemente general para que sea aplicable a cualquier tipo de producto y, además, debe proveer los medios para identificar unívocamente a cada TRU y almacenar la información de las TRU y sus relaciones. El modelo que propone (Figura 2), basado en un Diagrama de Clases UML, consiste en dos paquetes, *Trazabilidad (Traceability)* y *Calidad (Quality)*, que agrupan las diferentes clases. El primero contiene las entidades que permiten rastrear y trazar al producto durante su ciclo de vida. El segundo posee las entidades que representan las características de calidad del producto. Este modelo será estudiado con más detalle en Sección 5.1.

Otro modelo de datos fue propuesto por Tursi et al. (2007), para resolver el problema de la interoperabilidad en los sistemas de manufactura. Él planteó que muchos de los problemas del manejo de información de la

trazabilidad de productos derivan de la heterogeneidad de las aplicaciones utilizadas para el manejo de la información dentro de las empresas.

Esto genera problemas de interoperabilidad debido a las diferentes representaciones de las estructuras de datos entre los distintos protocolos de las aplicaciones. *Tursi et al. (2007)*, propone, por lo tanto, un modelado de la información orientado al producto que permite que cada sistema de información perciba al producto de la misma forma. Para llevar esto a cabo se propone capturar toda la información relevante para proporcionar una estructura estandarizada de modelado y poder así mapear la información de cada producto desde diferentes aplicaciones.

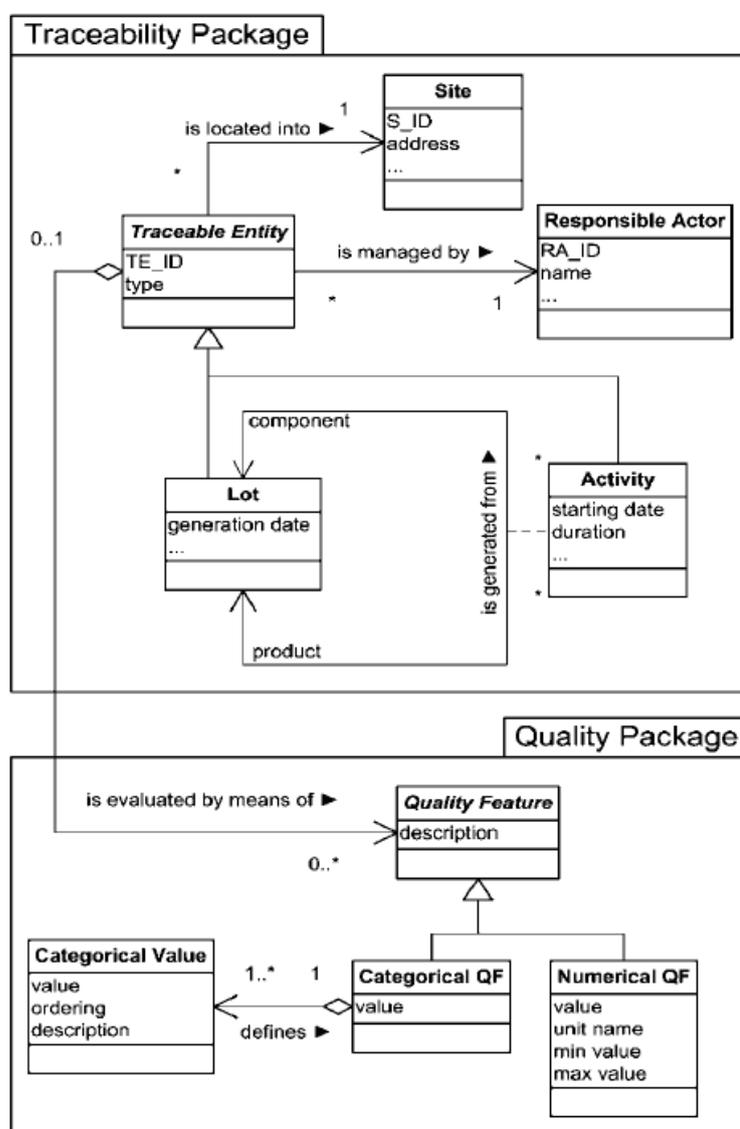


Figura 2 Modelo de datos para trazabilidad (Bechini et al., (2005))

Los modelos presentados coinciden en que el centrado en el producto representa un factor clave para la interoperabilidad y un mejor manejo del

intercambio de información. El que más simplifica la cantidad de información necesaria para el modelado de datos es el presentado por *Bechini et al.*, (2005). Este modelo es tomado como la principal referencia para el desarrollo de esta tesis.

3.4 Gestión del ciclo de vida del producto

Para crear el modelo de datos que llevará la información del producto a lo largo del ciclo de vida del producto físico, es necesario definir el ciclo de vida del producto, es decir, las etapas por las que atraviesa cada producto (Figura 3):

1. Desarrollo: abarca el diseño del producto en conjunto con su entorno de producción.
2. Producción: incluye subetapas como el planeamiento de los materiales necesarios, la extracción de la materia prima, los procesos de producción, los test de seguridad y calidad, entre otros.
3. Distribución: abarca el almacenamiento y distribución del producto terminado.



Figura 3 Ciclo de vida del producto

4. Uso: incluye las actividades que ocurren durante el uso del producto, tomando en cuenta soporte y mantenimiento.
5. Reciclado y/o descarte: representa la etapa en la que el producto ya no se encuentra en uso y toma en cuenta la destrucción y reciclado del producto.

Generalmente los diferentes sistemas de trazabilidad toman en cuenta, para el registro de datos, las etapas 2, 3 y 4. (Gemessi, 2010)

3.5 Tecnologías necesarias para los sistemas

En esta sección se pretende realizar un estudio del estado del arte de las tecnologías disponibles y comúnmente utilizadas para facilitar la identificación de productos y recolección de la información relacionada a éstos.

En 2008, *Bechini et al.*, definieron el problema conocido como “*first mile*” o “*de la primera milla*”. Este problema plantea que la facilidad para obtener información precisa de los productos aumenta a medida que nos alejamos del origen de los mismos, es decir nos posicionamos en las etapas finales de la producción y el procesado. Por el contrario, cuanto más nos acercamos a la granja de donde se extrae la materia prima, por ej., la precisión de los datos disminuye, al aumentar la dificultad para la recolección de los mismos. A partir de esto se deduce que aumentar la precisión de la información registrada durante la *first mile* es de vital importancia para la efectividad de los sistemas de trazabilidad. Además sugiere que, siempre que sea posible, la recolección de datos se realice de forma automática, para evitar errores humanos en la recolección de información.

La técnica más aceptada para la recolección de información sin la interrupción del flujo de trabajo es la del etiquetado de cada TRU con una etiqueta (*tag*) tal que transporta la información básica para identificarlo, la que puede ser leída con un lector adecuado a dicha información. Las tecnologías más usadas son el código de barras y el RFID (Identificación por Radio Frecuencias) (Evizal et al., (2016)). Desde el punto de vista práctico para la trazabilidad, el RFID posee grandes ventajas frente a los códigos de barras, a pesar de su costo superior. El RFID puede ser leído a una tasa mucho más rápida, la distancia de detección es mayor y sin necesidad de apuntar específicamente a un lugar con el lector como ocurre con los código de barras. En la Tabla 2 se observa una comparación entre ambas tecnologías.

Criterios	Código de Barras	RFID
Método de lectura	Requiere verificación visual directa	No requiere línea de visión
Ubicación de ítems	Identifica el ítem en un punto específico de la línea	No provee ubicación física absoluta.
Velocidad de lectura	Menor rapidez (caja por caja)	Mayor rapidez (Pallet completo)
Durabilidad	Se degrada	Mayor durabilidad
Almacenamiento de datos	Solamente 1 ítem	Varios ítems

Flexibilidad de información	Poca flexibilidad	Actualización en tiempo real
Seguridad	Baja	Alta (código de acceso)
Costo	Bajo	Alto

Tabla 2 Diferencias entre Código de Barras y RFID

Cada tag tiene asignado un string, o clave, único que permite la identificación unívoca del lote, remitiendo a una base de datos donde se almacena la información relativa al mismo. Para asegurar la unicidad del string existen varios estándares usados mundialmente que permiten generarlos. Un claro ejemplo de etiquetado es el *Tag de Oreja o Caravana* que se utiliza en el ganado, el cual codifica, con códigos alfanuméricos, información específica, como puede ser la fecha de nacimiento, la granja de donde proviene, información sobre las vacunas, etc., y que hoy día está siendo reemplazado por la *Caravana Electrónica* que utiliza la tecnología RFID.

Como se mencionó, la identificación única de cada TRU en una escala global es fundamental para el éxito de la implementación de los sistemas de trazabilidad. Como se mencionó en la Sección 3.2, uno de los estándares más utilizados actualmente es el estándar GS1 que provee un manejo más efectivo de la cadena de suministros al brindar herramientas estandarizadas que permiten a cada actor dentro de la cadena comunicarse en un lenguaje global. Esto permite un estándar genérico, global y único para cada participante de la cadena aliviando las presiones de identificación de las compañías y sus productos, mientras facilita el intercambio de información entre las demás compañías de la cadena. (Gemesis, 2010)

El procesamiento de la gran cantidad de datos generados, para obtener información útil, ha sido resuelto mediante la utilización de un ambiente de servicios web basado en los estándares del Open Geospatial Consortium (OGC) (Steinberger, Rothmund, & Auernhammer, 2006). El OGC es una organización sin fines de lucro que desarrolla estándares de acceso libre para la comunidad geoespacial (OCG, 2017). El uso de los sistemas geográficos de información (GIS, en inglés), ha facilitado el manejo, análisis y presentación de la información, relacionando la localización geográfica de materias primas y productos, con otros atributos relacionados, principalmente en las producciones agropecuarias y el transporte. (Gemesis, 2010)

Otra herramienta que se ha vuelto muy importante en los últimos años para el intercambio de información es el Extensible Markup Language

(XML). Este provee un formato común para la representación de los datos y el intercambio en el comercio electrónico (e-commerce, en inglés) vía internet. Esto permite que no se necesite ningún manejo especial de los datos cuando se utilizan los protocolos más comunes de internet. A su vez, diferentes grupos industriales han creado extensiones de XML que satisfacen sus necesidades particulares, en cuanto a estructura y formato de la información se refiere. (Gemesis, 2010)

La tecnología no solo es útil para identificar el producto, sino también para verificar que los requerimientos de calidad sean debidamente satisfechos. Algunas de las especificaciones que se deben cumplir suelen ser el tamaño, la firmeza, el sabor, la acidez; entre otras características físicas y químicas. Se han desarrollado métodos no destructivos para medir algunas de estas características, por ejemplo imágenes por infrarrojo y por resonancia magnética para medir la firmeza o detectar objetos peligrosos dentro de los productos. También existen otros dispositivos que permiten analizar las características químicas de las muestras de productos, la superficie del equipamiento y el aire, para detectar contaminantes microbianos peligrosos. Por otro lado, debido a la demanda de la trazabilidad genética, se han desarrollado tecnologías, PCR de tiempo real por ejemplo, para el análisis de la composición genética de productos y de esta manera monitorear los OGM. Otro tipo de monitoreo que se lleva a cabo es el del ambiente donde se manipulan los productos, registrando por ejemplo, temperatura, humedad, composición del aire, etc., mediante el uso de biosensores y analizadores de gases. (Opara, 2003)

3.6 Internet of Things (IoT – *Internet de las Cosas*)

La mayoría de las tecnologías mencionadas en la sección anterior se utilizan actualmente, pero con la limitación de que la información no es procesada y analizada en tiempo real. Estas tecnologías han logrado mejorar considerablemente las condiciones de la producción de alimentos, su eficiencia y seguridad, pero es necesaria una mejora de las condiciones debido a la sobredemanda de alimentos que existe en la actualidad.

La IoT promete superar estos inconvenientes realizando nuevos tipos de controles, mediante una red inteligente de objetos interconectados sensibles al contexto y que permiten ser identificados, sensados y controlados remotamente.

La IoT brinda las siguientes ventajas (Sundmaeker et al., (2016)):

- Mejor sensado y monitoreo de la producción, incluyendo uso de recursos, desarrollo de los cultivos, comportamiento animal y procesamiento de alimentos.
- Mejor entendimiento de las condiciones agrícolas, como las condiciones climáticas y medioambientales, bienestar animal, pestes, malezas y enfermedades; y creación de conocimiento acerca del manejo más adecuado.
- Control remoto y sofisticado de cultivos, procesado y logística por parte de actores y robots. Aplicación precisa de pesticidas y fertilizantes, robots para eliminación automatizada de malezas, control remoto de las condiciones ambientales durante el transporte, etc.
- Mejora de la trazabilidad y el monitoreo de la calidad de los alimentos mediante el control remoto de la ubicación y las condiciones de los envíos y productos.
- Incremento de la conciencia en sustentabilidad y salud de los consumidores mediante nutrición personalizada, domótica y “*wereables*”.

La agricultura de precisión representa actualmente uno de los ejemplos que sienta los pilares fundamentales para solucionar el problema de la *first mile*. Los sistemas de producción agropecuarios son, hoy en día, más precisos, así también como los mecanismos de registro. Se ha ganado transparencia en pos del aseguramiento de la calidad, permitiendo por ejemplo conocer donde y cuando se aplica un fertilizante o un insecticida. Para esto se han desarrollado sistemas de recolección de datos automatizados para la maquinaria agrícola basados en el estándar ISO11783, que es la norma que especifica la comunicación entre los dispositivos electrónicos utilizados en la maquinaria agrícola (Ruiz-Garcia, Steinberger, & Rothmund, 2010). Por ejemplo, los tractores puede ser localizados y dirigidos precisamente, utilizando datos de satélites, para incrementar la productividad especialmente en máquinas grandes: una máquina pulverizadora requiere un sistema de guía avanzado para evitar la superposición de las aplicaciones de agroquímicos. Esto mejora la productividad al mismo tiempo que disminuye la contaminación. Sin embargo solo los productores

innovadores aplican estas tecnologías, y el uso de la información recopilada es limitado (Sundmaeker, Verdouw, Wolfert, & Pérez-Freire, 2016).

La *agricultura inteligente* va un poco más allá del concepto de agricultura de precisión, ya que basa las tareas de manejo no solo en la ubicación sino también en datos, relacionándolos y mejorándolos por el contexto y el conocimiento de la situación. Las tareas son disparadas por eventos en tiempo real, lo que hace necesaria la reconfiguración de las características en tiempo real para llevar a cabo acciones ágiles, especialmente ante cambios drásticos en las condiciones del entorno. Estas características generalmente incluyen asistencia inteligente en implementación, mantenimiento y uso de la tecnología (Figura 4) (Sundmaeker et al., (2016)).

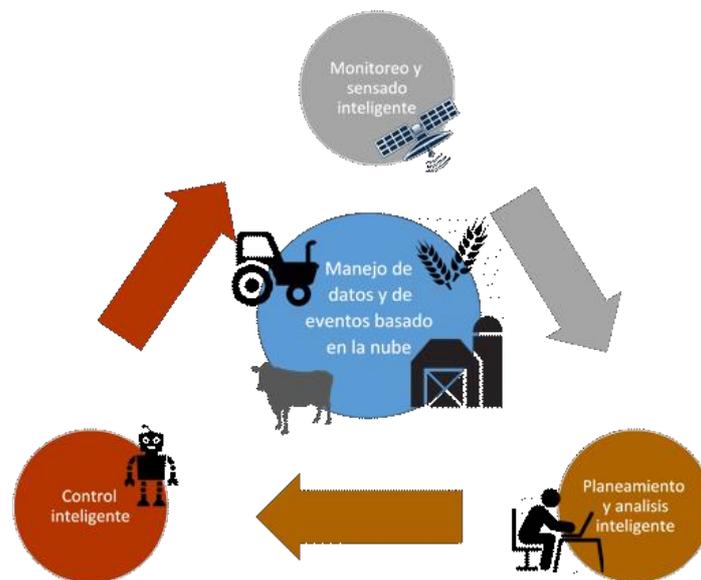


Figura 4 Ciclo de la agricultura inteligente

Desde el punto de vista de la trazabilidad y rastreabilidad la IoT brinda soluciones sofisticadas para el manejo remoto de los empaques y productos desde el origen al consumidor final (Figura 5).

Como se ha mencionado en secciones anteriores actualmente las compañías implementan trazabilidad de forma cerrada, es decir, son sistemas internos y estáticos, implementando el principio “*one step up/one step down*” sin considerar siquiera la posibilidad de implementar sistemas abiertos que sirvan dinámicamente al cambio de socios comerciales.

El uso de sensores para monitoreo remoto de variables físicas y químicas está cada vez más divulgado, mediante la utilización de tecnologías inalámbricas como el Wi-Fi, Bluetooth, GPRS y ZigBee, pero uno de los mayores inconvenientes, y que resulta crítico a la hora de la adopción de

estas tecnologías, es su costo de implantación al considerar grandes volúmenes de productos. Además muchas de las tecnologías de sensores (sensores químicos por ejemplo) están aún en etapas experimentales (Sundmaeker et al., (2016)).

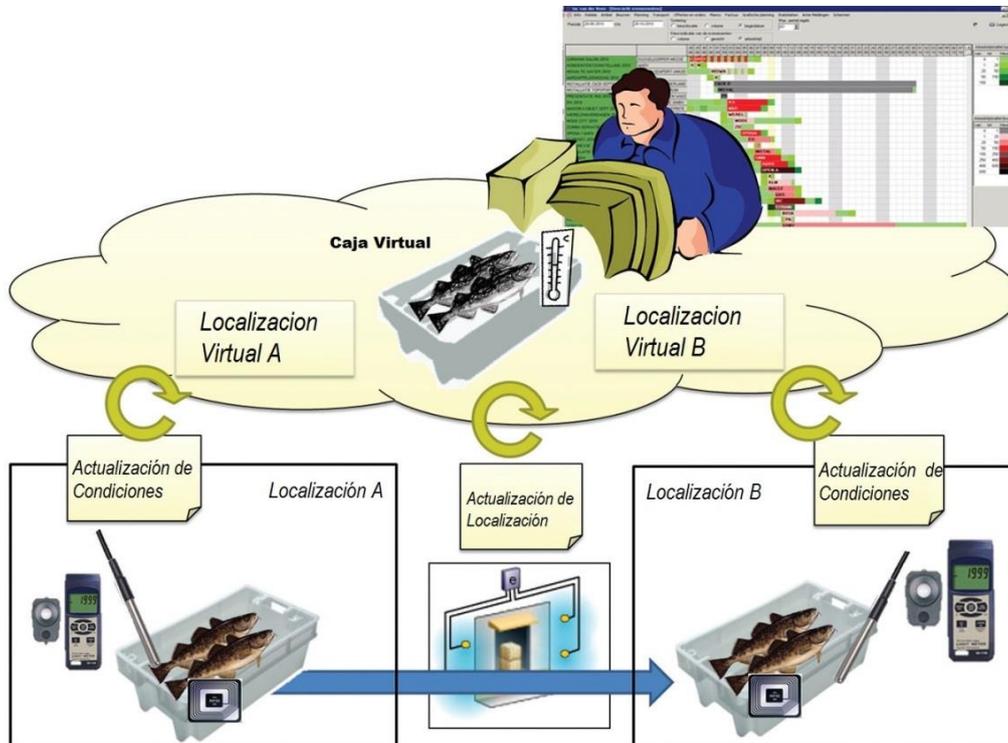


Figura 5 Virtualización de una caja de pescado, con información logística, para facilitar el control y manejo remoto

La virtualización se usa para referenciar a representaciones digitales de objetos de la vida real o imaginarios. La virtualización es un enfoque prometedor para lograr incorporar exitosamente las IoT en las cadenas de suministro alimentario, y de esa forma apalear la gran cantidad de desafíos que estas presentan. La virtualización permite simular y optimizar los procesos de los alimentos usando software en vez de llevar a cabo experimentos físicos. Además brinda la posibilidad de ser usada dinámicamente en la gestión operativa de las cadenas de suministro. Debido a esto, las cadenas de alimentos pueden ser monitoreadas, controladas, planificadas y optimizadas a distancia y en tiempo real, partiendo de los objetos virtuales. (Verdouw et al., (2016))

Como se ha dicho, en la IoT, las entidades físicas tienen contrapartes digitales (virtuales), los objetos se vuelven “*conscientes del contexto*”, y pueden interactuar comunicarse, intercambiar información y conocimiento.

Toda la información obtenida de los sensores se representa en objetos virtuales, lo que brinda grandes posibilidades que están por encima de la trazabilidad y la rastreabilidad, tales como gestión del desvío de la calidad de los alimentos, re-planificación y optimización de las funcionalidades. En todo este sistema, el internet actúa como andamiaje de almacenamiento y comunicación que sostiene a las representaciones virtuales de objetos asociando toda la información relevante entre ellos.

Las entidades virtuales tienen dos propiedades fundamentales (Bauer, 2013):

- La entidad virtual está asociada, generalmente, a una única entidad física, a la cual representa; pero una entidad física puede estar asociada a múltiples entidades virtuales, por ejemplo, una representación por cada dominio de aplicación. Cada entidad virtual es un artefacto digital que debe tener un único ID, y que puede ser clasificado en artefacto activo o pasivo. Los activos pueden ser software en ejecución o servicios que ejecutan otros servicios o recursos, y los pasivos son softwares pasivos como por ejemplo entradas de base de datos que representan digitalmente a la entidad física.
- Idealmente las entidades virtuales son representaciones sincronizadas de un conjunto de características de la entidad física. Lo que significa que los parámetros digitales relevantes se actualizan con cualquier cambio en la entidad física, y viceversa en casos donde algún parámetro de la entidad física puede ser modificado o controlado.

El concepto de Entidad Aumentada representa una composición de una entidad virtual y la entidad física a la que está asociada, resaltando el hecho de que los objetos reales forman parte de los procesos digitales gracias a esta entidad.

Desde el punto de vista de la IoT, existen tres tipos de dispositivos básicos: sensores, tags y actuadores. Los sensores proveen información o datos acerca de las entidades físicas que están monitoreando. Los tags, como se ha mencionado en secciones anteriores, sirven para identificar a las entidades físicas. Los actuadores pueden modificar el estado en el que se encuentran las entidades físicas.

En la Figura 6 se presenta un modelo de dominio de la IoT. En este se pueden ver resumidas las relaciones entre las entidades y los objetos antes mencionados.

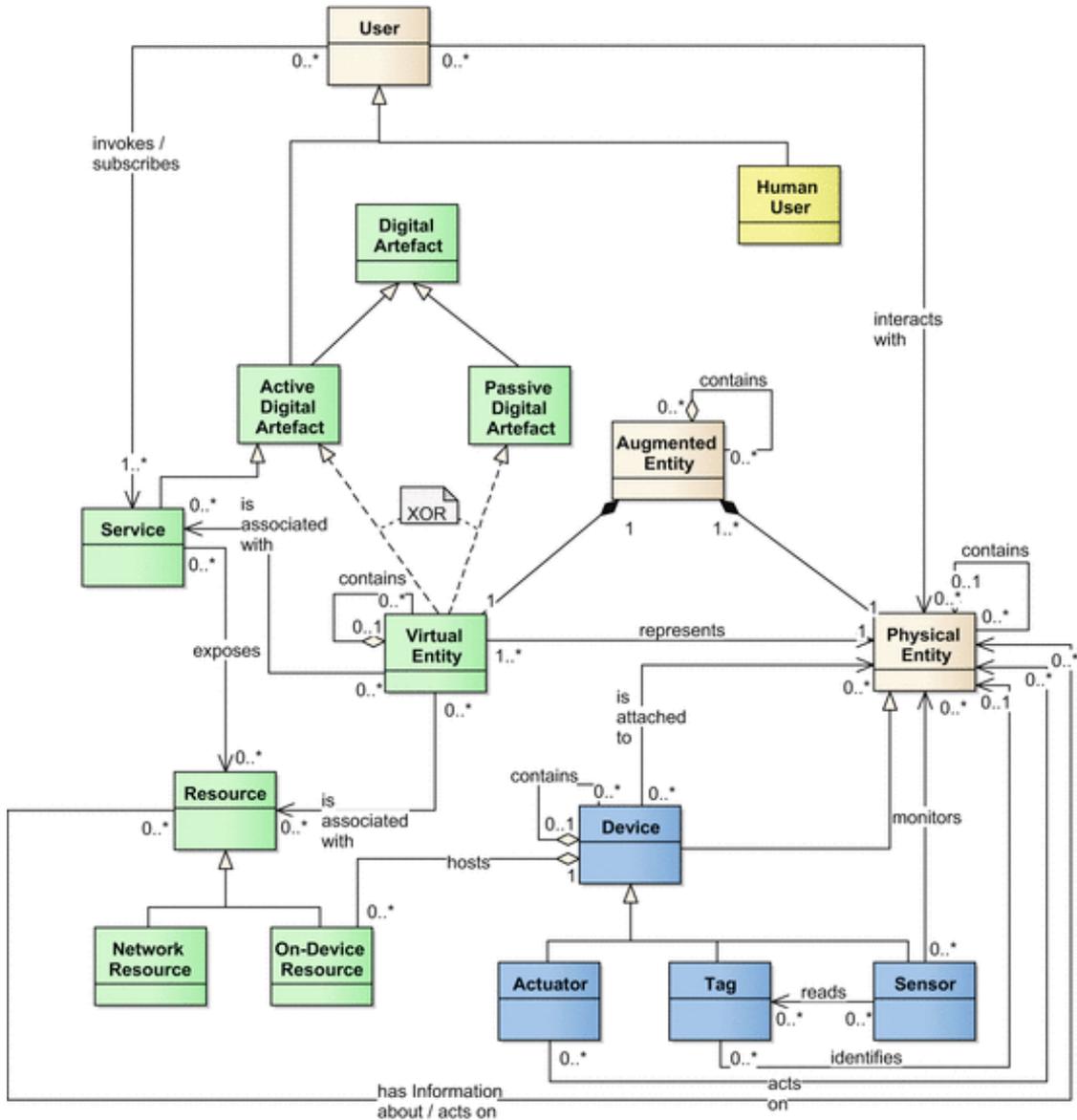


Figura 6 Modelo de Dominio para la IoT (Bauer, 2013)

4. DOMAIN-SPECIFIC LANGUAGES (DSL) – PERFILES UML

4.1 DSL

Los Lenguajes Específicos del Dominio (Domain-Specific Languages – DSL), son llamados, también, lenguajes de aplicación, especializados, de propósito especial, entre otros. Se utilizan para lograr expresividad en determinado dominio. Esto se logra mediante la utilización de características del lenguaje adaptadas particularmente al dominio de aplicación, lo que, al ser comparado con los Lenguajes de Programación de Propósito General (General-Purpose Programming Languages – GPL) ofrece grandes beneficios en cuanto a productividad y costos de mantenimiento, debido a su mayor facilidad de uso para el dominio en cuestión. También brindan la posibilidad de ser utilizados por programadores con menor experticia en programación, lo que abre el dominio de aplicación a un grupo mayor de desarrolladores de software. (Mernik et al., (2005))

En la Figura 7 se observan algunos ejemplos de DSLs ampliamente conocidos, y su dominio de aplicación.

DSL	Application Domain
BNF	Syntax specification
Excel	Spreadsheets
HTML	Hypertext web pages
LaTeX	Typesetting
Make	Software building
MATLAB	Technical computing
SQL	Database queries
VHDL	Hardware design

*Figura 7 Lenguajes específicos de dominio (DSLs) ampliamente utilizados.
(Mernik, Heering, & Sloane, 2005)*

Las ventajas que ofrece un DSL son:

- Incorpora notaciones particulares del dominio que generalmente los GPL no permiten definir. Esto está directamente relacionado con la productividad.
- Permite incorporar estructuras específicas del dominio que no siempre pueden ser mapeadas en objetos o funciones tradicionales.
- Permite el análisis, verificación, optimización y transformación en sus estructuras, de manera más sencilla que un GPL.

Las librerías son buenos competidores de los DSL, ya que para la mayoría de los casos representan la solución más efectiva tomando en cuenta la relación Costo-Efectividad.

Los DSL pueden tener distintos grados de ejecutabilidad e, incluso, no ser ejecutables. Esto lleva a que muchas veces los programas sean llamados especificaciones, definiciones o descripciones, para los casos no ejecutables.

Los DSL cumplen un papel muy importante a la hora de favorecer la reutilización de software. En el marco de los grandes sistemas de software permite la reutilización de código fuente, diseño de software, gramática del lenguaje y abstracciones del dominio. (Mernik et al., (2005))

4.2 Perfiles UML

El Unified Modeling Language (UML) es un lenguaje gráfico de propósito general ampliamente utilizado por sus ventajas en expresividad para modelar sistemas. Este lenguaje es definido por el Object Management Group (OMG).

En algunos casos se necesita contar con lenguajes más específicos para modelar ciertas características especiales de un dominio particular. Para lograr esto el OMG ofrece dos alternativas: una opción es crear un lenguaje alternativo a UML y otra opción es extender el lenguaje UML especializando o restringiendo algunos conceptos, sin alterar la semántica de UML. (OMG, 2015)

El OMG define una arquitectura de cuatro niveles, como se puede ver en la Figura 8, para describir los elementos que forman parte del modelado de un sistema: M0 (Instancias), M1 (Modelo del Sistema), M2 (Metamodelo – Modelo del modelo) y M3 (Meta-Metamodelo). Para definir los elementos del nivel M3 se utiliza un lenguaje especial que se denomina MOF (Meta-Object Facility) que es un lenguaje para describir lenguajes de modelado (UML o el propio MOF, por ejemplo) (OMG, 2014).

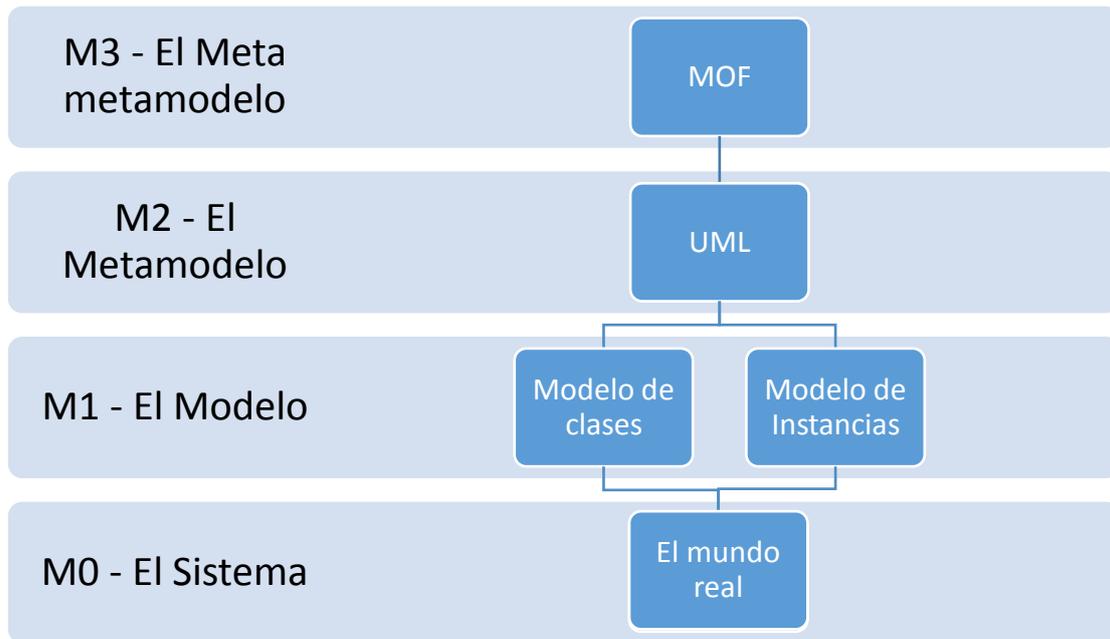


Figura 8 Arquitectura de metamodelos definida por la OMG

Un Perfil se define con el estereotipo «profile» en el paquete UML que extiende a un metamodelo (o a otro perfil) (Nivel M2 en la arquitectura definida por la OMG). Esto se logra utilizando tres mecanismos definidos por UML (Fuentes & Vallecillo, 2004):

- Los estereotipos: se define por un nombre y por los elementos del metamodelo a los que puede asociarse. Se utiliza el estereotipo «stereotype» dentro de la caja que lo contiene. (Figura 9)

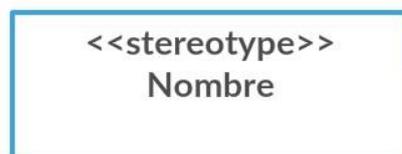


Figura 9 Estereotipo

- Las restricciones: son las condiciones que deben cumplir los elementos del metamodelo que han sido estereotipados. Éstas pueden expresarse en lenguaje natural o en Object Constraint Language (OCL) el cual forma parte de UML.
- Los valores etiquetados: son meta atributos adicionales que se asocian a una metaclase del metamodelo extendido. Se definen por un nombre y un tipo. (Figura 10)

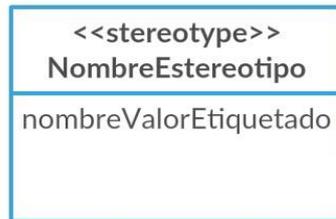


Figura 10 Valor Etiquetado

Las principales ventajas de los Perfiles UML son: la posibilidad de definir nuevos vocabularios para un dominio específico, notaciones particulares para elementos ya existentes, agregar nueva semántica, dar sintaxis para constructores que no tienen notación y agregar información en modelos para su transformación.

Debido a que el mecanismo de extensión de los perfiles UML está definido dentro de la infraestructura de UML, esto define los mecanismos usados para adaptar metamodelos existentes a plataformas específicas, dominios o modelados de procesos de software. Esto también permite que los mecanismos de extensión sean soportados por las herramientas UML existentes. Esta característica es una de las principales ventajas que ofrecen los perfiles por sobre otros mecanismos de especialización (Giachetti et al., (2009a), (2009b))

Existen muchos patrones de diseño para optimizar la construcción de los Perfiles UML, pero se mencionaran los casos en los que los patrones generan una disminución considerable en la cantidad de estereotipos necesarios, ya que los perfiles son generalmente más legibles cuando son compactos. Se presentaran dos patrones de optimización que se aplican en los casos en los que se definen estereotipos que no son estrictamente necesarios, por lo que el objetivo es identificar estos patrones y reemplazarlos por construcciones más simples (Robert et al., (2009)):

- Patrón “*Composición Solitaria*” o “*Lonely Compound*”: Se aplica cuando un estereotipo se introduce como una composición de otro. En estos casos, el primer estereotipo no tiene atributos y no participa en ninguna otra relación, por esto puede ser removido y reemplazado con un atributo en el segundo estereotipo. Por ej., en la Figura 11(a) se puede ver como el estereotipo <<TestCase>> está relacionado al estereotipo <<TestContext>>

mediante una composición, por lo que puede ser reemplazado por un atributo simple como se muestra en la Figura 11(b)

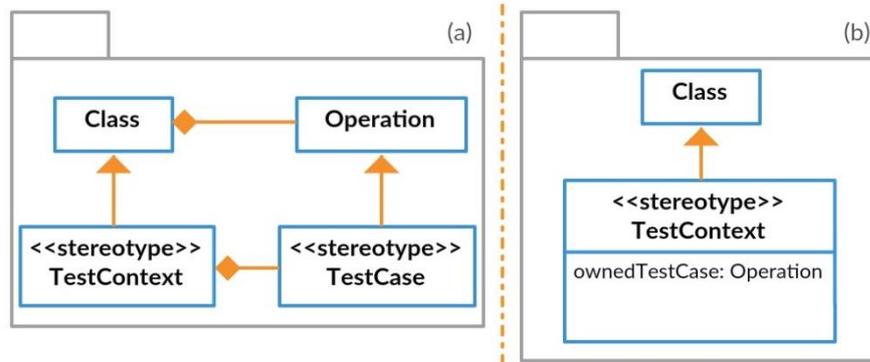


Figura 11 Patrón Lonely Compound. (Robert et al., (2009))

- El patrón “Hijo Superfluo” o “Superfluous Child”: se aplica cuando un estereotipo tiene uno o más estereotipos “hijo” que, al igual que en el caso anterior, no poseen atributos y no participan en ninguna otra relación. Por lo que estos estereotipos pueden ser removidos y reemplazados por un atributo enumerado en el estereotipo “padre”. Por ej., en la Figura 12(c) el estereotipo <<Task>> tiene dos hijos que luego son reemplazados por una enumeración como se muestra en la Figura 12(d).

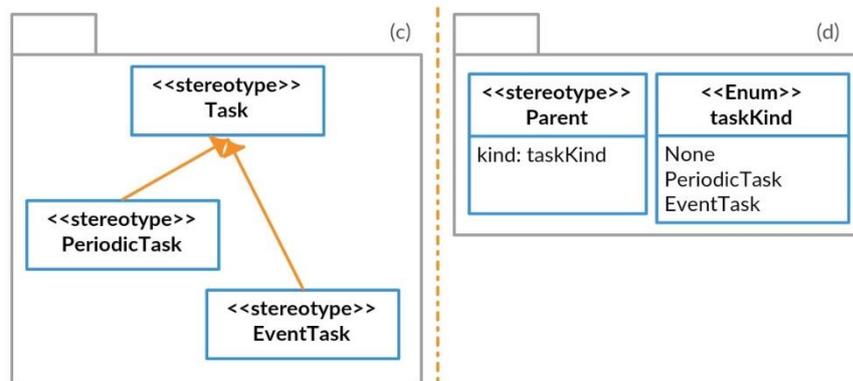


Figura 12 Patrón Superfluous Child. (Robert et al., (2009))

4.3 Object Constraint Language (OCL)

Los diagramas UML generalmente no son lo suficientemente específicos por sí solos para plasmar todas las características relevantes de una especificación. Una de esas características son las restricciones y habitualmente se definen en lenguaje natural, lo que tarde o temprano genera ambigüedades. Con el fin de evitar estas ambigüedades se han creado lenguajes

formales para definir restricciones, pero al costo de ser demasiado complejos para desarrolladores sin un fuerte conocimiento matemático.

A fin de sortear estos obstáculos, se ha creado el OCL. Es un lenguaje formal de fácil lectura y escritura.

Es un lenguaje de especificación puro, por lo que cuando una sentencia OCL es evaluada se obtiene un valor como resultado. No puede cambiar nada en el modelo, lo que significa que el estado de un sistema no va a cambiar por causa de la evaluación de una sentencia OCL, a pesar de que estas se pueden utilizar para especificar un cambio de estado.

OCL no es un lenguaje de programación por lo que no puede ser usado para crear lógica de programa o control de flujo. (OMG, 2014)

4.4 MDA y los Perfiles UML

La Model-Driven Architecture (MDA), definida por el OMG, permite considerar a los modelos como punto central del desarrollo de grandes sistemas, evitando tomar en cuenta cuestiones particulares de implementación en las primeras etapas del desarrollo, mejorando su portabilidad. Uno de los objetivos principales de la MDA es obtener valor de los modelos al mejorar la capacidad de lidiar con la complejidad e interdependencia de los sistemas y maximizar la reusabilidad de los mismos. Existen varias formas de lograr esto (OMG, 2014):

- Mejorando la comunicación al utilizar los modelos como vehículos, ofreciendo herramientas al equipo de desarrollo para lograr el consenso y mejorar el entendimiento.
- Utilizando los modelos para automatizar el proceso de implementación, lo que involucra la creación de modelos independientes de la plataforma (PIM – Platform Independent Model), que luego son transformados en un modelo específico para la plataforma particular (PSM – Platform Specific Model).
- Utilizar los datos de los modelos (métricas, validaciones, estadísticas) para asesorar en la toma de decisiones, monitoreo y calidad.
- Simular y ejecutar los modelos para validar y/o localizar errores de los mismos.

- Los modelos bien definidos pueden aportar gran cantidad de información (libros de procesos, documentación, etc.)

La MDA está basada en tres niveles conceptuales como se muestra en la Figura 13. El primero consiste en un modelo altamente abstracto, el Modelo Independiente de la Computación (CIM - Computation Independent Model), que modela los requerimientos del sistema definiendo el entorno donde el mismo va a operar. El segundo es el PIM, que se obtiene transformando al CIM, disminuyendo su nivel de abstracción a un modelo computacional, pero aun independiente de la plataforma computacional usada. El PIM define la funcionalidad del sistema, en base a soluciones de diseño para los requerimientos planteados en el CIM, pero sin tomar en cuenta detalles de implementación.

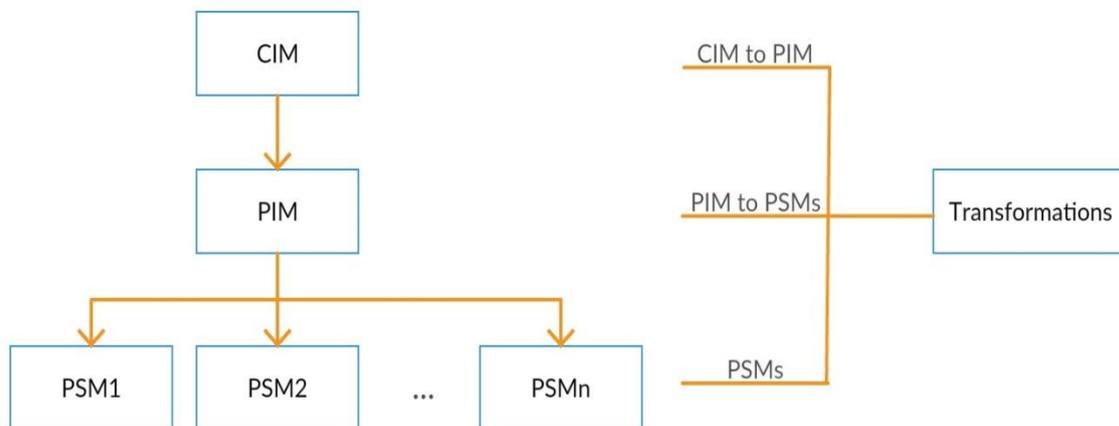


Figura 13 Esquema de Model-Driven Architecture. (García-Díaz et al., (2010))

A partir del PIM, mediante el uso de transformaciones, se obtiene el tercer y último nivel, el PSM. Se puede obtener uno o más PSMs, dependiendo de la cantidad de plataformas tecnológicas que se desee abarcar. Esto logra una gran independencia entre la definición de la funcionalidad y la plataforma que lo va a implementar. (Fuentes & Vallecillo, 2004) (Weidemann et al., (2008)).

Se pretende que las transformaciones mencionadas sean lo más automatizadas posibles. Y es aquí que los Perfiles UML cobran importancia, al momento de definir el PSM y las reglas de transformación. Al usar los Perfiles UML para definir el modelo de la plataforma, se garantiza la consistencia con UML, lo que cumple uno de los requisitos para el éxito de la aplicación de MDA: la utilización de estándares para modelización. Con este procedimiento se pretende “marcar” los elementos del PIM con los

estereotipos del Perfil de una plataforma para obtener su correspondiente PSM. Una marca puede representar una indicación de cómo transformarse, requisitos extra funcionales o de calidad. (Fuentes & Vallecillo, 2004) (Giachetti et al., (2009a), (2009b))

4.5 DSL y UML

Existen diferentes opiniones entre los expertos respecto a la MDA y la utilización de DSLs o Perfiles UML en su desarrollo. Una de las opiniones más neutrales se basa en que el desarrollo con MDA y Perfiles UML es útil cuando se usa con moderación, y que son mayormente válidos en el dominio de las Tecnologías de Información y Comunicación (TICs). Por otro lado los detractores de los Perfiles UML para el desarrollo con MDA se basan en lo extenso del lenguaje UML y su poca especificidad a la hora de representar un dominio particular. (García-Díaz et al., (2010))

La crítica anterior desconsidera la gran utilidad que poseen los Perfiles UML al permitir la reutilización de parte, o de la totalidad, del metamodelo de referencia, por la que los desarrolladores pueden crear modelos con una terminología y datos específicos más especializados. Además los propios desarrolladores realizan validaciones a los modelos para así obtener una semántica UML más ajustada al dominio. Las clases estereotipadas generalmente igualan la terminología de un DSL y las relaciones estereotipadas en un perfil UML son las mismas que se usan para la terminología gráfica de un DSL. Lo que convierte a ambos (Perfiles UML y DSLs) en enfoques que pueden expresar conceptos en varios niveles de abstracción arbitrarios. (Dalgarno & Fowler, 2008) (Robert et al., (2009))

Es posible, entonces, utilizar UML para modelar el CIM y el PIM en un desarrollo MDA, para luego implementar un DSL o un Perfil UML que permita transformar el PIM en PSM. Como se ha mencionado antes, un DSL o un Perfil UML pueden usarse indistintamente sin riesgo de perder especificidad para un dominio particular, siempre que se definan correctamente las reglas de transformación que cada enfoque ofrece y se trabaje principalmente en el ámbito de las TICs. Esto se debe a que en este ámbito se pueden utilizar Tecnologías Orientadas a Objetos para describir los problemas, ya que los profesionales del mismo están familiarizados con esto; en cambio a profesionales de otras áreas no les interesa aprender el lengua-

je básico de modelización de sistemas sino que prefieren modelizar directamente el dominio de su interés. (Thomas, 2004)

En la Figura 14 se aprecia un ejemplo de la diferencia fundamental entre un DSL y un Perfil UML. Allí se puede ver que el DSL es más entendible para los humanos y tiene un mayor nivel de abstracción, pero como ambos modelos representan los mismos objetos, se pueden crear mapeos entre ellos de manera transparente a los usuarios. (García-Díaz et al. (2010)) (Abouzahra et al., (2005)) (Silingas et al., (2009))

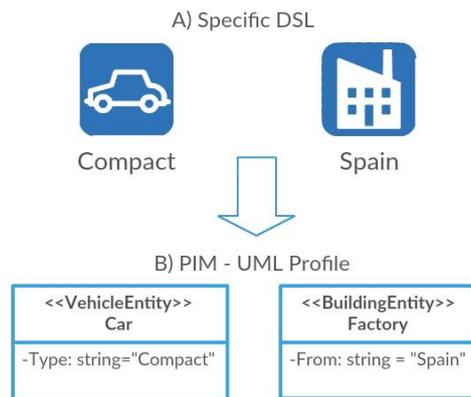


Figura 14 DSL y Perfil UML representando los mismos objetos

5. DESARROLLO DE PERFIL UML PARA TRAZABILIDAD ALIMENTARIA

Partiendo de los pasos sugeridos por Fuentes y Vallecillo para crear un Perfil UML (Fuentes & Vallecillo, 2004) y de las características necesarias para los sistemas de trazabilidad alimentaria, mencionadas en el Capítulo 2, en este capítulo se llevara a cabo el desarrollo de un Perfil UML que facilite la implementación de estos sistemas. Este desarrollo se hace posible ya que el dominio para este modelo está enmarcado dentro de las Tecnologías para la Comunicación y la Información (TICs), ya que abarca diferentes sistemas de registro y sensado del estado de las muestras de alimento y su posterior procesamiento para el análisis de los datos recogidos a lo largo de toda la cadena de producción; y como se expresó en la Sección 4.5 esto favorece la utilización de Perfiles UML para el diseño de modelos especializados en el dominio.

Como se mencionó en la Sección 3.3, existen modelos ampliamente aceptados que intentan resolver los problemas de interoperabilidad de la información e intentan ser aplicables a la trazabilidad en cualquier sistema de producción. Estos modelos son los presentados por *Tursi et al.* (2007), *Bechini et al.* (2008), *Manikas & Manos* (2009) y *Gemesi* (2010). Para la realización de esta tesis se ha tomado a estos autores como referencia para crear un modelo acorde a la temática de la misma.

Un escenario típico de una cadena productiva se muestra en Figura 15. En dicha figura, una caja representa una *TRU* que puede representar una cantidad de productos empaquetados o procesados bajo las mismas condiciones (*Lote*); o a un grupo de productos que comparten algunas características como tamaño, tipo, categoría, entre otros (*Batch*). Una estrella representa una *actividad*, que puede ser, por ejemplo, empaquetado, procesamiento, distribución o venta. Las flechas grandes representan a los *Responsables* de la cadena de suministro, son los ejecutantes de las diferentes actividades y pueden ser personas físicas o jurídicas.

Al considerar que cada TRU es producida por una actividad, entonces se puede afirmar que cada TRU tiene un responsable y, por lo tanto éste también es responsable de la veracidad en la información relativa a la misma.

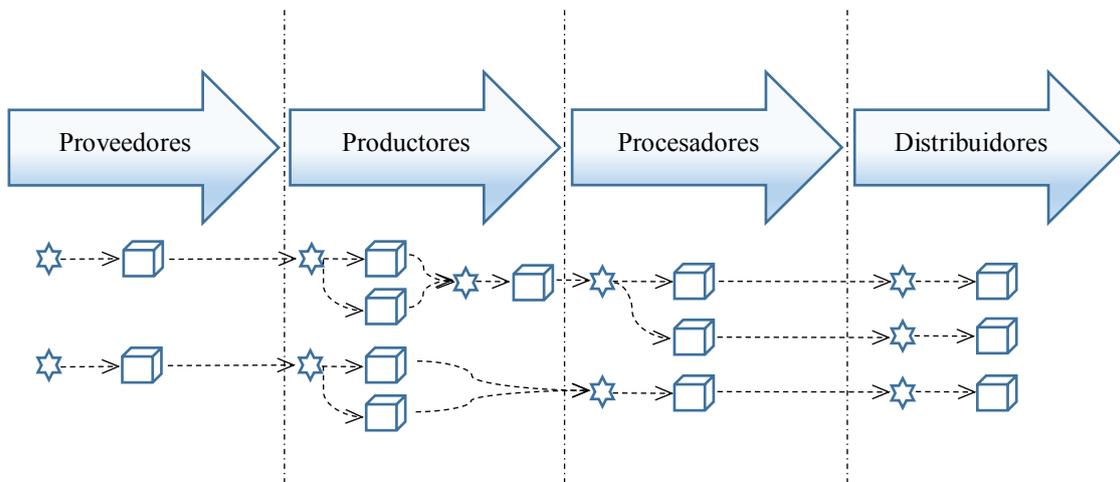


Figura 15 Escenario de la cadena productiva que involucra responsables (flechas grandes), actividades (estrellas) y TRUs (cajas).

Se pretende lograr un modelo centrado en el producto para asegurar la trazabilidad de una TRU a lo largo de su ciclo de vida. Para esto, la entidad que represente en el modelo a la TRU debe contener información relevante al producto, como su identificación, una descripción del mismo, resultados de los test de calidad, la ubicación, el estado, etc. Como se mencionó anteriormente este tipo de modelo favorece la interoperabilidad tanto dentro de la misma empresa como entre los diferentes actores a lo largo de la cadena.

5.1 El Metamodelo del Dominio

El metamodelo está compuesto por dos paquetes UML: Calidad y Trazabilidad. (Figura 16)

5.1.1 El Paquete Trazabilidad

El paquete Trazabilidad contiene como clase central a la *EntidadTrazable*, la cual es una clase abstracta que modela las características de dos entidades básicas de la trazabilidad: *Actividad* y *Lote*. Esta clase posee el atributo *Estado* que representa los posibles estados en los que se puede encontrar la entidad trazable. Como se dijo anteriormente, identificar las Entidades Trazables es una de las tareas más complicadas al momento de implementar estos modelos.

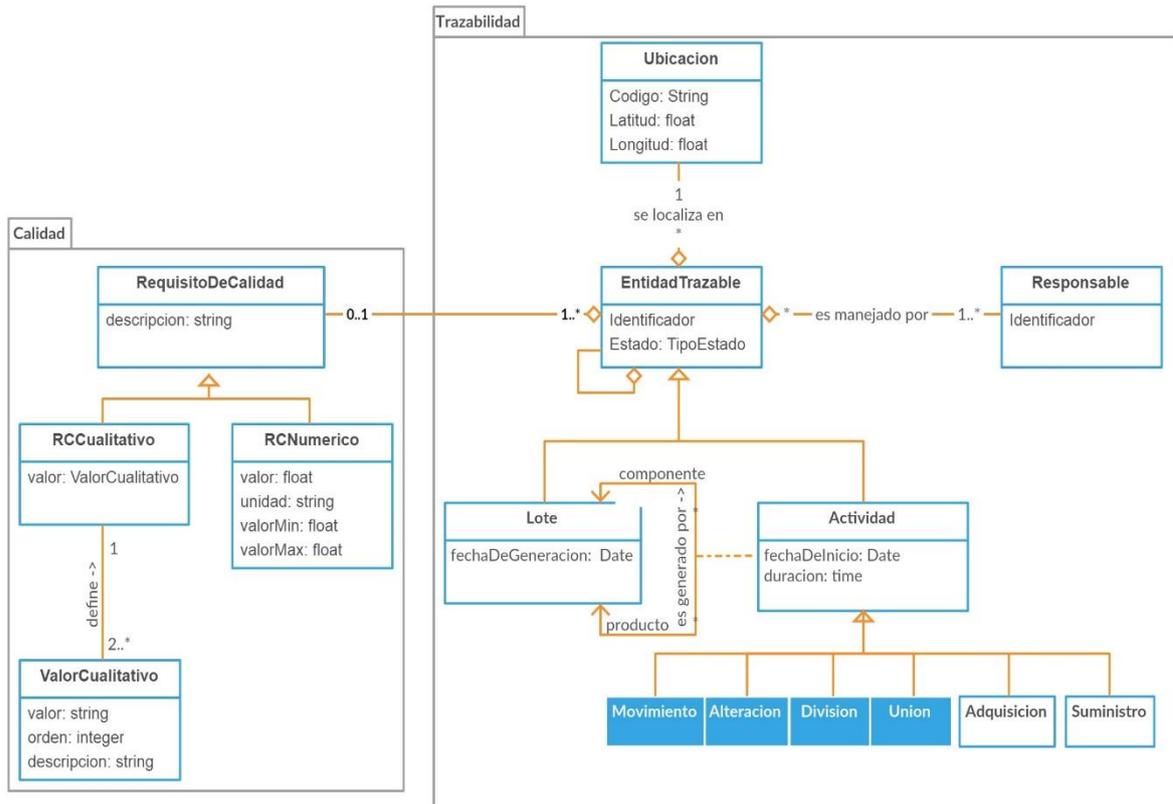


Figura 16 Diagrama de clases UML del metamodelo para sistemas de trazabilidad

Cada entidad trazable posee un único *Responsable* y una única *Ubicación* en un momento dado.

El *Responsable* puede representar a una persona física o jurídica, encargada de la manipulación de un lote dado en un momento particular. También pueden ser considerados como responsables la naturaleza y el consumidor final, que corresponden al inicio y al final, respectivamente, de la cadena productiva, aunque en algunos casos la naturaleza también puede considerarse como fin de la cadena para los casos en que algún producto defectuoso es descartado.

La clase *Ubicación* representa a la localización física del lote dentro de la cadena productiva. Esta ubicación puede estar representada por coordenadas geográficas, dependiendo del tipo de producto o del tamaño del lote; y por un código de identificación.

Por su parte la clase *Actividad* representa cualquier modificación que se realice en las características del producto (*Lote*), ya sea desde el punto de vista de su composición (Subclases coloreadas de azul en el diagrama); o del responsable de la misma (Subclases *Suministro* y *Adquisición*). La clase *Actividad* representa uno de los objetos más importantes desde el

punto de vista de establecer los vínculos entre las diferentes etapas del ciclo de vida de un producto.

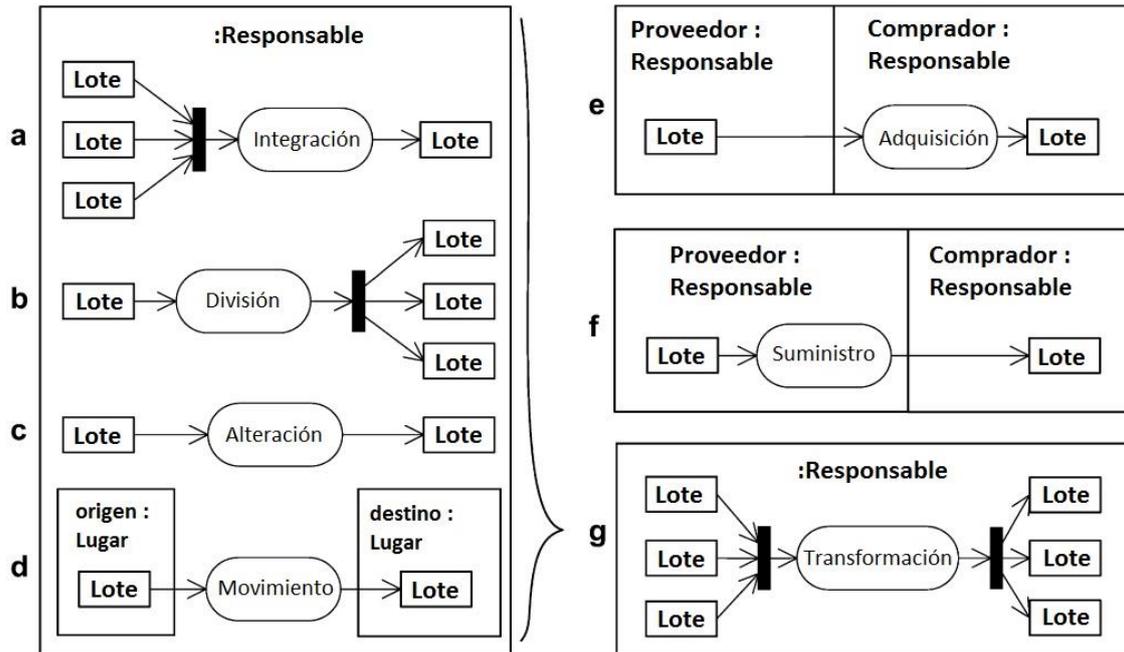


Figura 17 Patrones de Comportamiento (Bechini et al., (2005))

Particularmente, las subclases que representan alguna transformación física en el lote se describen a continuación, y son conocidos como “*Patrones de Comportamiento*” (Figura 17):

- Integración (o unión) del Lote: un número “n” de lotes (con n mayor a uno) se integra en un único lote. Ejemplos: empaquetado, mezcla.
- División del Lote: varios lotes se generan a partir de un único lote. Ejemplos: Corte, separación del lote.
- Alteración del Lote: de ser necesario, se crea un nuevo lote a partir de la actividad de alteración del lote inicial. Ejemplos: freezado, cocción, secado.
- Movimiento del Lote: Se mueve un lote de un lugar a otro, por lo que varía la ubicación del mismo. Dependiendo del sistema de registro se puede asentar solo las ubicaciones inicial y final, o se puede registrar el recorrido completo.

En todos los casos en los que la identificación del lote cambie, el responsable de los lotes crea una asociación entre el o los lotes previos a la actividad y el o los lotes posteriores a la actividad.

Para realizar una Adquisición, un responsable (comprador) adquiere un producto de otro responsable (proveedor) registrando la operación y permitiendo de esta forma realizar la Trazabilidad del lote. De manera similar, en la actividad de Suministro, el proveedor le transfiere el lote al comprador, dejando asentada dicha transferencia para permitir llevar a cabo el Rastreo de un lote.

Debido a que en los patrones de comportamiento el responsable de la transformación también tiene que ser el responsable del lote, primero se debe llevar a cabo una actividad de Adquisición por parte de dicho responsable, antes de poder realizar la transformación. Esto implica que si hay una adquisición también debe haber una actividad de suministro por parte de otro responsable.

La asociación llamada “*es generado por*” define lo antes mencionado, asegurando que cada lote es creado de uno o más lotes, y esto a su vez se lleva a cabo mediante una actividad.

5.1.2 El Paquete Calidad

En el paquete Calidad se representan todos los posibles controles de calidad, y sus resultados, necesarios en cada etapa de la cadena productiva para asegurar el buen estado del lote.

El objeto central de este paquete es la clase abstracta *RequisitoDeCalidad*, de la cual derivan dos subclases que identifican dos tipos de controles de calidad. Si el control puede ser representado de forma numérica se está en presencia de la clase *RCNumerico*, la cual se define en términos del valor numérico, una unidad de medida y el rango de valores que puede tomar. En el caso en que el control se define en términos cualitativos, la clase en cuestión es *RCCualitativo* y se define en términos de un conjunto de valores cualitativos. Estos últimos se representan mediante la clase *ValorCualitativo*, que posee una descripción, el valor y el orden de apreciación relativo a todos los posibles valores cualitativos.

Si bien para este desarrollo se tomó como base el metamodelo sugerido por *Bechini et al.* (2005), en conjunto con otros como los de *Gemesi* (2010), *Manikas & Manos* (2009) y *Khabbazi et al.* (2009), se realizaron modificaciones conceptuales al mismo, ya que dicho metamodelo consideraba que luego de cada transformación se debía modificar el identificador del lote, basado en que en sus consideraciones un lote solo podía tener una

única ubicación y un único responsable. Las modificaciones para este trabajo consistieron en darle a un lote la posibilidad de tener más de una ubicación a lo largo de la cadena productiva y también la posibilidad de no cambiar necesariamente de identificador de lote al ser sometido a una actividad. Una de las situaciones disparadoras del cambio de identificación de lote, por ejemplo, puede ser el cambio de responsable del lote, dependiendo de la situación particular en que ocurra. De esta manera se considera que se quita un poco el foco a la actividad y se centra el modelo en el producto, al mismo tiempo que se beneficia la interoperabilidad entre los diferentes actores de la cadena y se facilita el registro de la información.

5.2 El Perfil UML

En la Figura 18 se presenta el Perfil UML propuesto.

En el perfil se puede ver como se han extendido las metaclasses *Class*, *AssociationClass*, *Operation* y *Property* del metamodelo UML. Este perfil contiene 15 estereotipos que están relacionados a las clases, atributos y asociaciones definidas en el metamodelo correspondiente detallado en la sección anterior, además de una enumeración para definir los posibles valores que puede tomar el valor etiquetado “*Estado*”.

A continuación se detallan los estereotipos declarados y las restricciones asociadas a cada uno. Esto es presentado en la Tabla 3 a Tabla 17. Además, en la sección siguiente se definen las restricciones planteadas, en el lenguaje OCL.

Nombre	EntidadTrazable
Clase Base	Class
Descripción	Es una clase abstracta que representa a todos los objetos trazables del sistema
Restricciones	<ul style="list-style-type: none"> - Una EntidadTrazable, en todo momento, debe tener solo una Ubicación - Una EntidadTrazable no tiene necesariamente un RequisitoDeCalidad - Una EntidadTrazable tiene en todo momento un único ID.
Valores Etiquetados	Estado

Tabla 3 Definición de Estereotipo EntidadTrazable

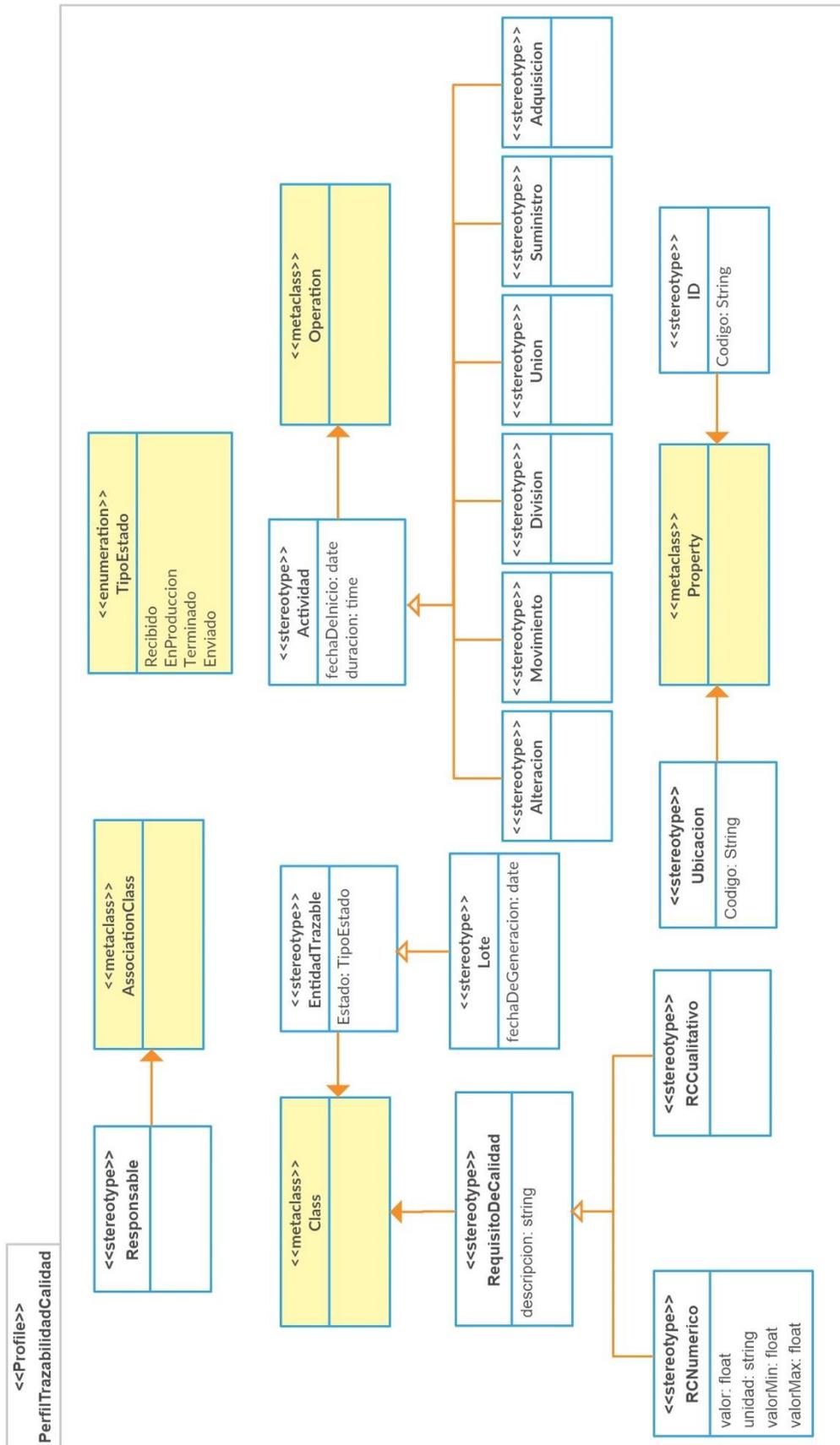


Figura 18 Perfil UML del metamodelo presentado

Nombre	Lote
Clase Base	Class.
Descripción	Representa al producto, o grupo de productos, que va a ser trazado o rastreado a lo largo de la cadena de producción.
Restricciones	<ul style="list-style-type: none"> - Un Lote no tiene Operaciones. - La ubicación del Lote, debe ser necesariamente la misma de su responsable y de la actividad que se ejecuta sobre el lote - Un Lote debe tener al menos un Responsable
Valores Etiquetados	fechaDeGeneracion

Tabla 4 Definición de Estereotipo Lote

Nombre	Actividad
Clase Base	Operation.
Descripción	Representa cualquier modificación que se le realice al Lote.
Restricciones	<ul style="list-style-type: none"> - Una Actividad siempre es llevada a cabo por un responsable.
Valores Etiquetados	fechaDeInicio, duración

Tabla 5 Definición de Estereotipo Actividad

Nombre	Responsable
Clase Base	AssociationClass
Descripción	Clase que representa al Responsable directo de la manipulación del Lote en determinado momento. Este Responsable es, a su vez, responsable de las actividades por las que atraviesa el Lote durante su responsabilidad.
Restricciones	<ul style="list-style-type: none"> - Un Responsable tiene en todo momento un único Identificador.
Valores Etiquetados	Identificador

Tabla 6 Definición de Estereotipo Responsable

Nombre	Movimiento
Clase Base	Operation
Descripción	Clase que representa cualquier actividad de traslado de un lote.
Restricciones	<ul style="list-style-type: none"> - Una operación de Movimiento necesariamente genera un cambio del atributo Ubicación del Lote en cuestión.
Valores Etiquetados	

Tabla 7 Definición de Estereotipo Movimiento

Nombre	Alteración
Clase Base	Operation
Descripción	Clase que representa las actividades de alteración del lote que no modifican la cantidad de lotes considerados.
Restricciones	
Valores Etiquetados	

Tabla 8 Definición de Estereotipo Alteración

Nombre	División
Clase Base	Operation
Descripción	Clase que representa a la actividad de División de un lote en varios lotes nuevos.
Restricciones	- La cantidad de lote Componente debe ser menor que la de lote Producto
Valores Etiquetados	

Tabla 9 Definición de Estereotipo División

Nombre	Unión
Clase Base	Operation
Descripción	Clase que representa a la actividad de Unión de varios lotes en un lote nuevo
Restricciones	- La cantidad de lote Componente debe ser mayor que la de lote Producto
Valores Etiquetados	

Tabla 10 Definición de Estereotipo Union

Nombre	Suministro
Clase Base	Operation
Descripción	Clase que representa a la actividad de Suministro de un lote de un Responsable a otro
Restricciones	- Solo se lleva a cabo cuando el Responsable del lote Componente es distinto del Responsable del lote Producto.
Valores Etiquetados	

Tabla 11 Definición de Estereotipo Suministro

Nombre	Adquisicion
---------------	-------------

Clase Base	Operation
Descripción	Clase que representa a la actividad de Adquisicion de un lote que proviene de otro Responsable
Restricciones	- Sólo se lleva a cabo cuando el Responsable del lote Componente es distinto al Responsable del lote Producto.
Valores Etiquetados	

Tabla 12 Definición de Estereotipo Adquisicion

Nombre	RequisitoDeCalidad
Clase Base	Class
Descripción	Clase abstracta que representa los requisitos de calidad que debe cumplir una Entidad Trazable para ser considerada “de calidad”
Restricciones	
Valores Etiquetados	Descripcion

Tabla 13 Definición de Estereotipo RequisitoDeCalidad

Nombre	RCCualitativo
Clase Base	RequisitoDeCalidad, Class
Descripción	Requisitos que se definen en términos cualitativos
Restricciones	- Un RCCualitativo debe tener un ValorCualitativo. - Deben existir al menos dos ValorCualitativo definidos
Valores Etiquetados	Valor

Tabla 14 Definición de Estereotipo RCCualitativo

Nombre	RCNumerico
Clase Base	RequisitoDeCalidad, Class
Descripción	Requisitos que se definen en términos de valores numéricos, una unidad de medida y el rango de valores que puede tomar.
Restricciones	- Las variables valor, valorMax y valorMin no pueden estar vacías
Valores Etiquetados	valor, unidad, valorMax, valorMin

Tabla 15 Definición de Estereotipo RCNumerico

Nombre	ID
Clase Base	Property
Descripción	Representa un Identificador único para toda la cadena productiva que puede representar un Responsable o una EntidadTrazable

Restricciones	
Valores Etiquetados	Código

Tabla 16 Definición de Estereotipo Identificador

Nombre	TipoEstado
Clase Base	Enumeration
Descripción	El tipo TipoEstado es una enumeración donde se especifican las clasificaciones de los estados que puede tomar una entidad trazable, como por ejemplo: Recibido, EnProduccion, Terminado y Enviado
Restricciones	
Semántica	

Tabla 17 Definición de Estereotipo TipoEstado

5.3 Restricciones en OCL

A. Una EntidadTrazable, en todo momento, debe tener solo una Ubicación

```
context UML::Class
inv: self.isStereotyped("EntidadTrazable") implies
self.allAttributes() ->
exists(att | att.isStereotyped("Ubicacion")->size() = 1)
```

B. Una EntidadTrazable no tiene necesariamente un RequisitoDeCalidad

```
context UML::Class
inv: self.isStereotyped("EntidadTrazable") implies
self.allAttributes() ->
exists(att | att.isStereotyped("RequisitoDeCalidad")->
size() >= 0)
```

C. Una EntidadTrazable tiene en todo momento un único ID.

```
context UML::Class
inv: self.isStereotyped("EntidadTrazable") implies
self.allInstances()->forall(et1,et2|
et1<>et2 implies et1.ID <> et2.ID)
```

D. Un Lote no tiene Operaciones.

```
context UML::Class
inv: (self.isStereotyped("Lote") implies
self.allOperations() -> size() = 0)
```

E. La ubicación del Lote, debe ser necesariamente la misma de su responsable y de la actividad que se ejecuta sobre el lote.

```

context UML::Class
inv: self.isStereotyped("Lote") implies
(self.allAttributes() -> exists(att1,att2 |
att1.isStereotyped("Ubicacion"))
and att2.isStereotyped("Responsable")
and (att2.allAttributes() ->
exists(att3|att3.isStereotyped("Ubicacion")
and att1=att3))
and (att2.allOperations() ->
exists(op1|op1.isStereotyped("Actividad")
and op1.allAttributes() ->
exists(att4|att4.isStereotyped("Ubicacion")
and att1=att4))))
    
```

F. Un Lote debe tener al menos un Responsable

```

context UML::Class
inv: self.isStereotyped("Lote") implies
self.allAttributes() ->
exists(att1 | att1.isStereotyped("Responsable")
and att1[Producto] -> size() > 0
and att1[Componente] -> size() > 0)
    
```

G. Una actividad siempre es llevada a cabo por un responsable

```

context UML::AssociationClass
inv: self.isStereotyped("Responsable") implies
self.allOperations() ->
exists(o1|o1.isStereotyped("Actividad")
or o1.isStereotyped("Union")
or o1.isStereotyped("Division")
or o1.isStereotyped("Movimiento")
or o1.isStereotyped("Alteracion")
or o1.isStereotyped("Suministro")
or o1.isStereotyped("Adquisicion"))
    
```

H. Una Ubicación tiene en todo momento un código de identificación.

```

context UML::Class
inv: self.isStereotyped("Lote")
or self.isStereotyped("Responsable") implies
(self.allAttributes() ->
exists(att | att.isStereotyped("Ubicacion")
and att.codigo <> ""))
    
```

```

context UML::Operation
inv: self.isStereotyped("Actividad") implies
(self.allAttributes() ->
  exists(att | att.isStereotyped("Ubicacion")
  and att.codigo <> ""))

```

I. Un Responsable tiene en todo momento un único ID

```

context UML::AssociationClass
  inv: self.isStereotyped("Responsable") implies
self.AllInstances() -> forAll(r1,r2|r1 <> r2 implies r1.ID
<> r2.ID)

```

J. Una operación de Movimiento necesariamente genera un cambio del atributo Ubicación del Lote en cuestión.

```

context Responsable::<<Movimiento>>movimiento(:Lote):Lote
post: self.Componente.allAttributes() ->
exists(att1|att1.isStereotyped("Ubicacion")
and self.Producto.allAttributes() ->
exists(att2|att2.isStereotyped("Ubicacion")
and att1 <> att2))

```

K. Para una operación de División la cantidad de lote Componente debe ser menor que la de lote Producto

```

context Responsable::<<Division>>Division(:Lote):Lote
let n : Integer = self.Componente -> size() in
let m : Integer = self.Producto -> size() in
post: n < m

```

L. Para una operación de Unión la cantidad de lote Componente debe ser mayor que la de lote Producto

```

context Responsable::<<Union>>union(:Lote):Lote
let n : Integer = self.Componente -> size() in
let m : Integer = self.Producto -> size() in
post: n > m

```

M. Una Adquisición y un Suministro solo se llevan a cabo cuando el Responsable del lote Componente es distinto del Responsable del lote Producto

```

Context Responsable::<<Suministro>>suministro(:Lote):Lote
pre: Lote.Responsable[Componente] <> Lote.Responsable[Producto]

```

```
context Responsable::<<Adquisicion>>adquisición(:Lote):Lote  
pre: Lote.Responsable[Componente] <> Lo-  
te.Responsable[Producto]
```

N. Un RCCualitativo debe tener un valor y Deben existir al menos dos ValorCualitativo definidos

```
context UML::Class  
inv: self.isStereotyped("RCCualitativo") implies  
self.valor <> 0 and valorCualitativo.allIntances() ->  
size() >=2
```

O. Las variables valor, valorMax y valorMin no pueden estar vacías.

```
context UML::Class  
inv: self.isStereotyped("RCNumerico") implies (self.valor  
<> 0 and self.valorMax <> 0 and self.valorMin <> 0)
```

5.4 Optimización al perfil diseñado

Basado en la descripción sobre patrones de diseño optimización de perfiles, llevada a cabo en la Sección 4.2, se pretende aplicar el patrón *Superfluos Child* al Perfil UML de Figura 18.

Al observar el estereotipo Actividad y sus seis subclases estereotipadas, se puede deducir que se cumplen los requisitos para la aplicación del patrón mencionado, ya que las subclases carecen de atributos, y no participan en ninguna otra relación. Para aplicar el patrón se deben suprimir las seis subclases, que serán reemplazadas por un valor etiquetado de nombre "actividad" y cuyos valores pueden ser los valores que toma la enumeración "TipoActividad": *Alteracion, Movimiento, Division, Union, Suministro y Adquisicion*.

Por lo tanto, el Perfil UML resultante es el que se puede observar en la Figura 19. En esta figura se aprecia como la aplicación del patrón hace que el Perfil UML diseñado sea más legible, sin tener que modificar, al menos en este caso, las restricciones detalladas en la sección anterior.

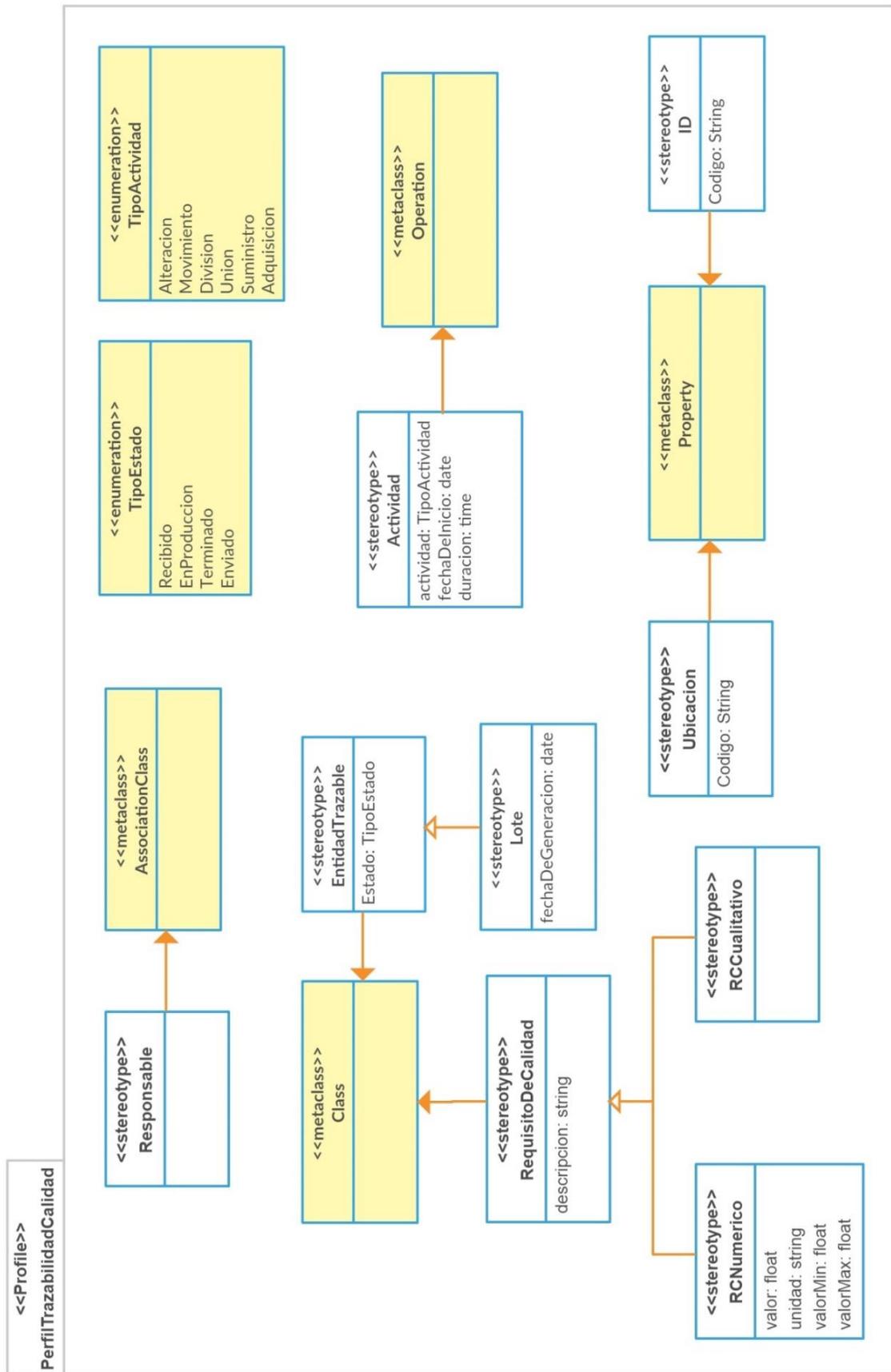


Figura 19 Perfil UML simplificado mediante el uso de patrones de optimización.

5.5 Evaluación del PerfilTrazabilidadCalidad

En esta sección se pretende realizar una evaluación de los beneficios que brinda el Perfil UML propuesto en la Sección 5.2 con su versión optimizada presentada en la Sección 5.4. Se analizará una serie de indicadores con el fin de realizar una evaluación cuantitativa del perfil diseñado. Y también se llevará a cabo un análisis cualitativo de un caso de aplicación donde se aplique el PerfilTrazabilidadCalidad.

5.5.1 Análisis de indicadores

Para este análisis se tomó como referencia el trabajo de *Robert et al.* (2009), donde los autores presentan un conjunto de tres métricas que determinan cuantitativamente cuán bueno es un Perfil UML.

Las métricas son:

- ANLA (*Average Number Location Application o Aplicación del Promedio de Ubicaciones*): Da el número promedio de elementos UML a los que los estereotipos de un Perfil UML pueden ser aplicados y se calcula como:

$$anla = \frac{1}{n} \sum_{i=1}^n M_i$$

Donde n es el número de estereotipos del perfil y M_i es la cantidad de metaclases a las que el estereotipo i -ésimo se puede aplicar.

El estereotipo puede ser aplicado a cualquier subclase de la metaclase que extiende. Por lo que se obtendrá un *anla* mayor cuando las metaclases extendidas se encuentren más cerca del elemento “raíz” del metamodelo referenciado por el perfil que de las “hojas”.

- ALDIT (*Average Leaf Depth of Inheritance Tree o Profundidad Promedio de la Hoja en el Árbol de Herencia*): Representa la longitud promedio del árbol de herencia en los estereotipos.

$$aldit = \frac{1}{n} \sum_{i=1}^n L_i$$

Donde n es el número de estereotipos del perfil y L_i es la cantidad de estereotipos “*padre*” del i -ésimo estereotipo. $L_i=0$ si el estereotipo no especializa a alguna superclase que sea estereotipada.

- *ASWA (Average Stereotype With Attributes o Promedio de Estereotipos con Atributos)*: Como su nombre lo indica, da un promedio de la cantidad de estereotipos que tienen atributos:

$$aswa = \frac{1}{n} \sum_{i=1}^n A_i$$

Donde n es el número de estereotipos del perfil y $A_i = 1$ si el estereotipo tiene uno o más atributos, o $A_i = 0$ en cualquier otro caso.

El ANLA da un indicio de cuánto depende el perfil del metamodelo UML. Un ANLA alto representa un perfil que extiende metaclases cercanas a los elementos raíz del metamodelo, y debido a que la semántica de los mismos es más general, el perfil resultante no será tan dependiente del metamodelo UML. Las métricas ALDIT y ASWA dan una idea de la complejidad del perfil considerado. Por su parte ASWA también muestra si el perfil puede ser parametrizado por los usuarios.

Para estudiar estas métricas se tomara en cuenta el perfil original, de la Figura 18.

Al aplicar las definiciones de las métricas al perfil se obtiene:

- $n = 15$
- ANLA: Los estereotipos que extienden la metaclase “*Class*” pueden ser aplicados también a las metaclases “*Component*” y “*AssociationClass*”. Los que extienden a la metaclase “*AssociationClass*” solo pueden ser aplicados a esa metaclases. Lo mismo ocurre con los estereotipos que extienden a la metaclase “*Operation*”. Por su parte los estereotipos que extienden a la metaclase “*Property*” también pueden ser aplicados a las metaclases “*AssociationEnd*” y “*Attribute*”. De esta forma la fórmula del ANLA da el siguiente resultado.

$$anla = \frac{29}{15} = 1.93$$

Este resultado es bajo, lo cual significa que el perfil UML diseñado es altamente dependiente de la semántica de UML.

- ALDIT: para el cálculo de esta métrica se tienen 10 estereotipos cuyas superclases son a su vez estereotipos. Por lo que de la fórmula de *aldit* se obtiene:

$$aldit = \frac{10}{15} = 0.67$$

- ASWA: en este caso la cantidad de estereotipos que poseen algún atributo es de 7, entonces:

$$aswa = \frac{7}{15} = 0.47$$

- De los resultados del ALDIT y ASWA podemos ver que el perfil tiene una complejidad media, con un árbol de herencia corto y que brinda buenas posibilidades de parametrización por parte de los usuarios (en aproximadamente un 50 % de los estereotipos).

5.5.2 Análisis cualitativo

Para esta comparación se analizará un ejemplo sencillo aplicando el perfil UML desarrollado, para luego revisar las diferencias.

Un ejemplo que se observa en *Bechini et al.* (2005), es el caso de la producción de vinos, donde se presentan requisitos de calidad numéricos y cualitativos, además de un ejemplo de una actividad de compra, y los objetos involucrados en dicha actividad. Basado en esto, se presenta en la Figura 20, un ejemplo de características similares, pero basado en una actividad de traslado, desde una bodega al almacén, que involucra a tres responsables y las consecuentes actividades de adquisición y suministro del lote entre ellos. Además, incorpora la definición de los requisitos de calidad del vino. Al mismo tiempo, en la Figura 21 se observa el mismo ejemplo pero con la aplicación del PerfilTrazabilidadCalidad al metamodelo.

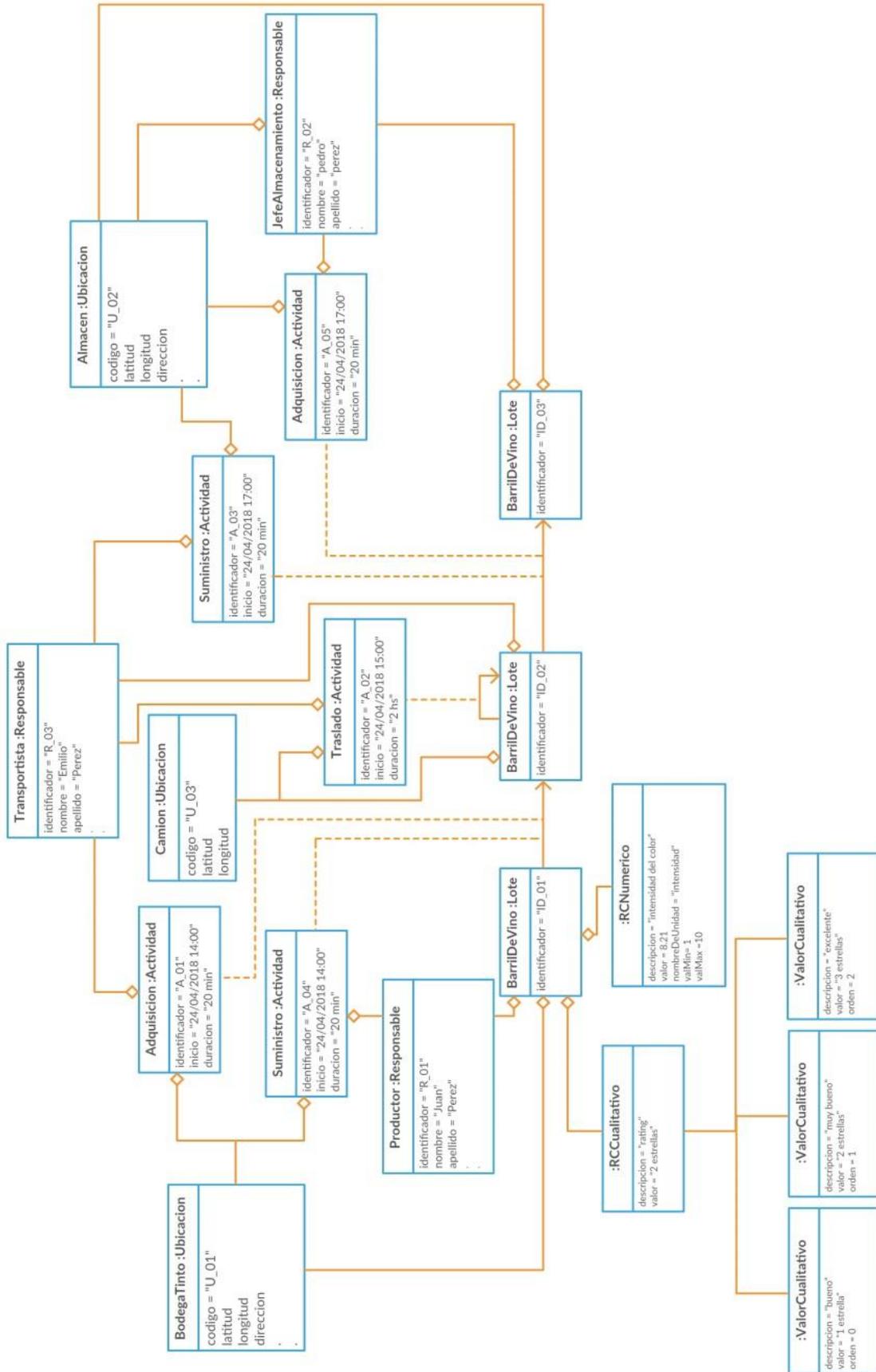


Figura 20 Ejemplo de aplicación del metamodelo

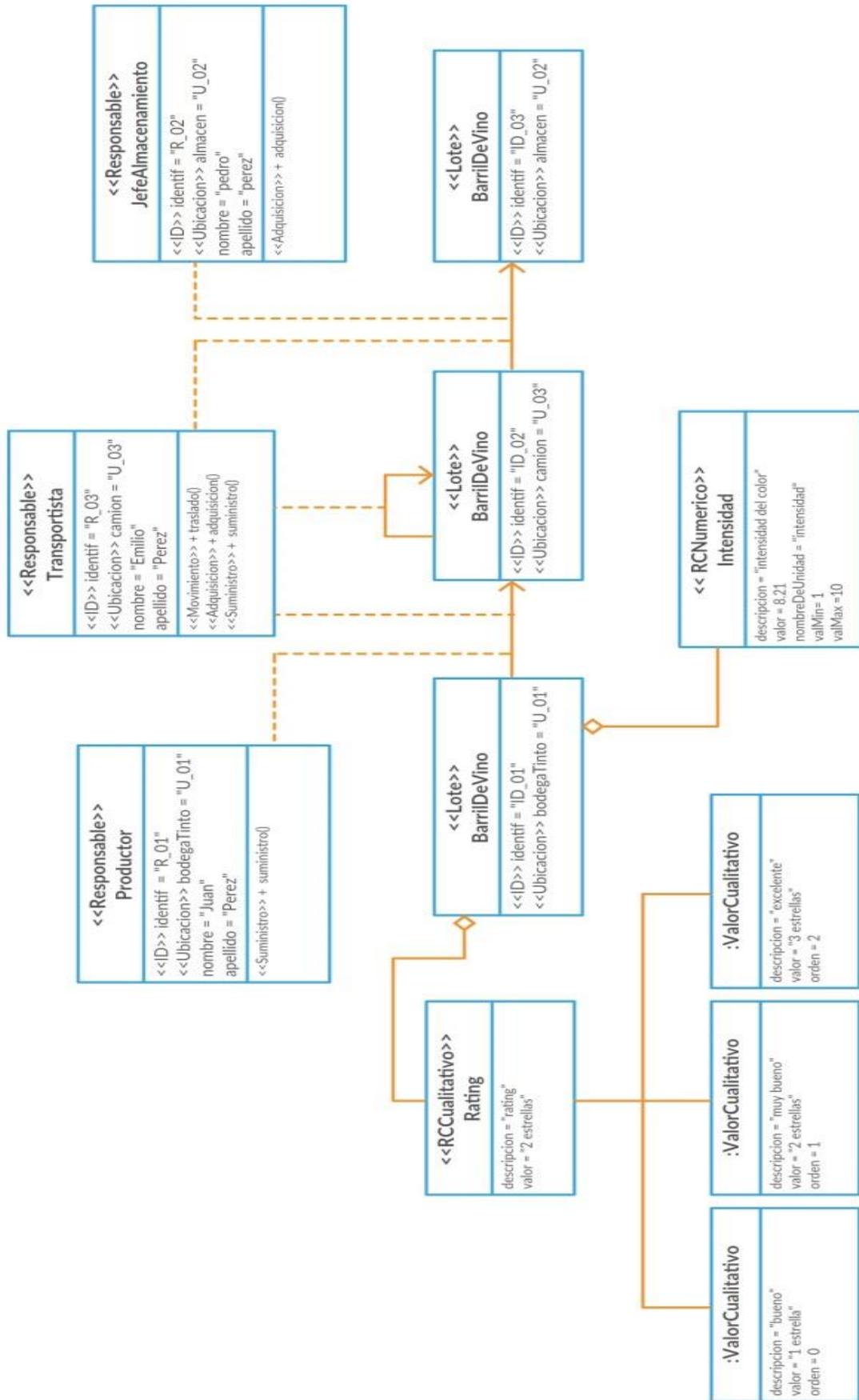


Figura 21 Ejemplo de aplicación del perfil UML

Cabe aclarar que los atributos faltantes de las actividades y las ubicaciones en esta figura se han omitido por el bien de la simplicidad, pero idealmente deberían ser incorporados como comentarios a las clases que los contienen. Es evidente la diferencia entre ambos diagramas, ya que se observa la disminución considerable de los objetos presentes al aplicar el perfil, la consiguiente disminución de las relaciones entre los objetos restantes, y la consecuente simplificación del metamodelo original, lo que lo hace más comprensible al ojo del desarrollador y facilita el intercambio de opiniones y criterios entre los distintos involucrados en el proceso del desarrollo del software. Además, tomando en cuenta lo que significa conceptualmente una *Actividad*, es más adecuado, desde el punto de vista abstracto, que esta sea representada como una operación en los diagramas de clases y que los responsables de dichas actividades sean los que lleven a cabo tales operaciones, al mismo tiempo que asocian al lote componente con el lote producto. También se puede apreciar que el modelo de la Figura 21 se enfoca más en el producto que el modelo de la Figura 20, ya que en torno a los productos se llevan a cabo las actividades, mediante la intervención de los responsables involucrados y la aplicación de los controles de calidad, quedando esto mejor representado luego de la aplicación del PerfilTrazabilidadCalidad.

Este análisis se puede replicar en diferentes ámbitos del dominio de la trazabilidad alimentaria, y para otras actividades, distintas de la actividad de compra mostrada en la Figura 20, una muestra de esto se lleva a cabo en la Sección 6.3 donde se reproduce la aplicación del PerfilTrazabilidadCalidad a las diferentes actividades del dominio de la trazabilidad de productos apícolas, ampliando la visión que presenta ejemplo de las Figura 20 y Figura 21.

5.6 El perfil PerfilTrazabilidadCalidad y la IoT

En la Sección 3.6 se comentó acerca de la importancia que está teniendo la tecnología de la IoT en los últimos años en el área de producción de alimentos y, por lo tanto, también en los sistemas de trazabilidad y de control de calidad de los mismos. En esta sección se pretende demostrar cómo el perfil diseñado puede ser implementado también en sistemas que contemplen tecnologías de IoT sin realizar grandes modificaciones, lo que

le da una gran flexibilidad y compatibilidad con posibles nuevas tecnologías.

Como ya se ha mencionado, la IoT representa una de las mejores respuestas para solucionar el problema de la “*first mile*”. Esto implica que cada sistema de producción de alimentos debe incorporar en las diferentes etapas productivas algún dispositivo capaz de recopilar información relevante al proceso en cuestión y, automáticamente, enviar esta información para su procesamiento. Existen actualmente varios Perfiles UML que intenta modelar este tipo de sistemas que cuentan con una parte física (Hardware) y una parte virtual (Software). A continuación se describen dos de esos perfiles que sirven como referencia para esta sección del trabajo.

Thramboulidis & Christoulakis (2015) presentaron el diseño de un Perfil UML para la explotación de la IoT en los sistemas de manufactura. Para llevar a cabo esto, describieron el “*componente mecatrónico o cyberfísico*”, que está compuesto por partes mecánicas, electrónicas y software altamente acoplados. Los autores extendieron y especializaron las metaclasses UML para facilitar el proceso de transformación de interfaces tradicionales orientadas a objetos, a interfaces de tipo REST. El perfil creado se llama UML4IoT y es utilizado para anotar, en el modelo UML la parte “*cyber*” del componente mecatrónico, aquellos artefactos del modelo que representan las propiedades del componente. El UML4IoT está basado en el modelo de objetos presentado en el protocolo OMA Lightweight M2M de la Open Mobile Alliance (OMA) para el manejo de dispositivos M2M (machine-to-machine) o IoT.

En *Costa et al.* (2016), se presenta un Perfil SysML llamado SysML4IoT (Figura 22). El lenguaje SysML es, a su vez, un Perfil UML que especifica un lenguaje de modelado de sistemas. El perfil presentado se basa en el metamodelo de la Figura 6.

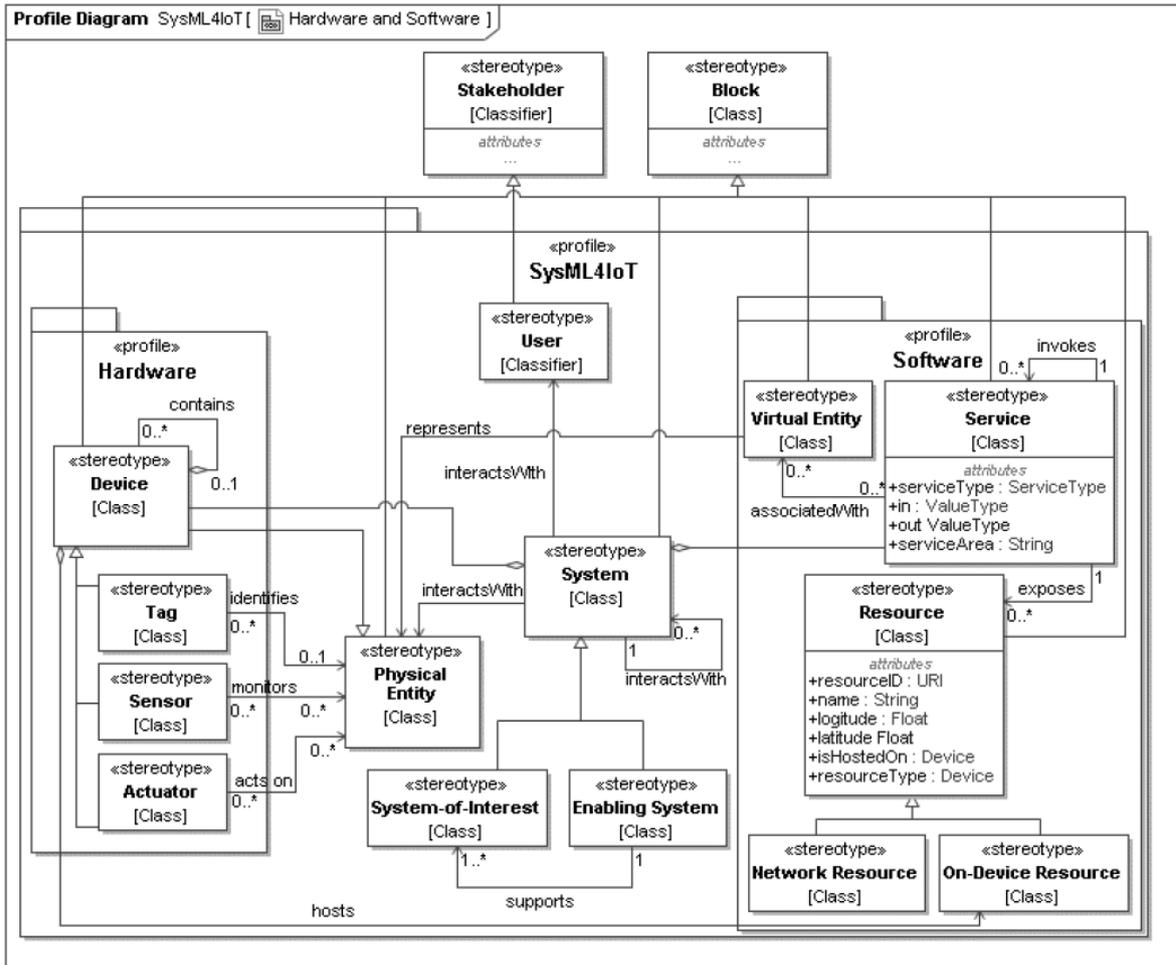


Figura 22 Perfil SysML para sistemas IoT (Costa et al., (2016))

Corredor et al. (2012), presentaron un perfil UML orientado a las tecnologías IoT, que denominaron “*Smart Space Modeling Language o SsML*”. Este está compuesto por dos enfoques que definen dos modelos: el “*Smart Object Model o SOM*”, que define los aspectos del procesamiento relacionados al sensado y las capacidades de actuación de los objetos Smart, y el modelo del contexto que manejan encapsulándolos en recursos *RESTful*; y el “*Environment Context Model o ECM*”, que se enfoca en describir los comportamientos de alto nivel, las interacciones y la información de contexto del espacio Smart. En la Figura 23 se puede observar el perfil aplicado al enfoque del SOM.

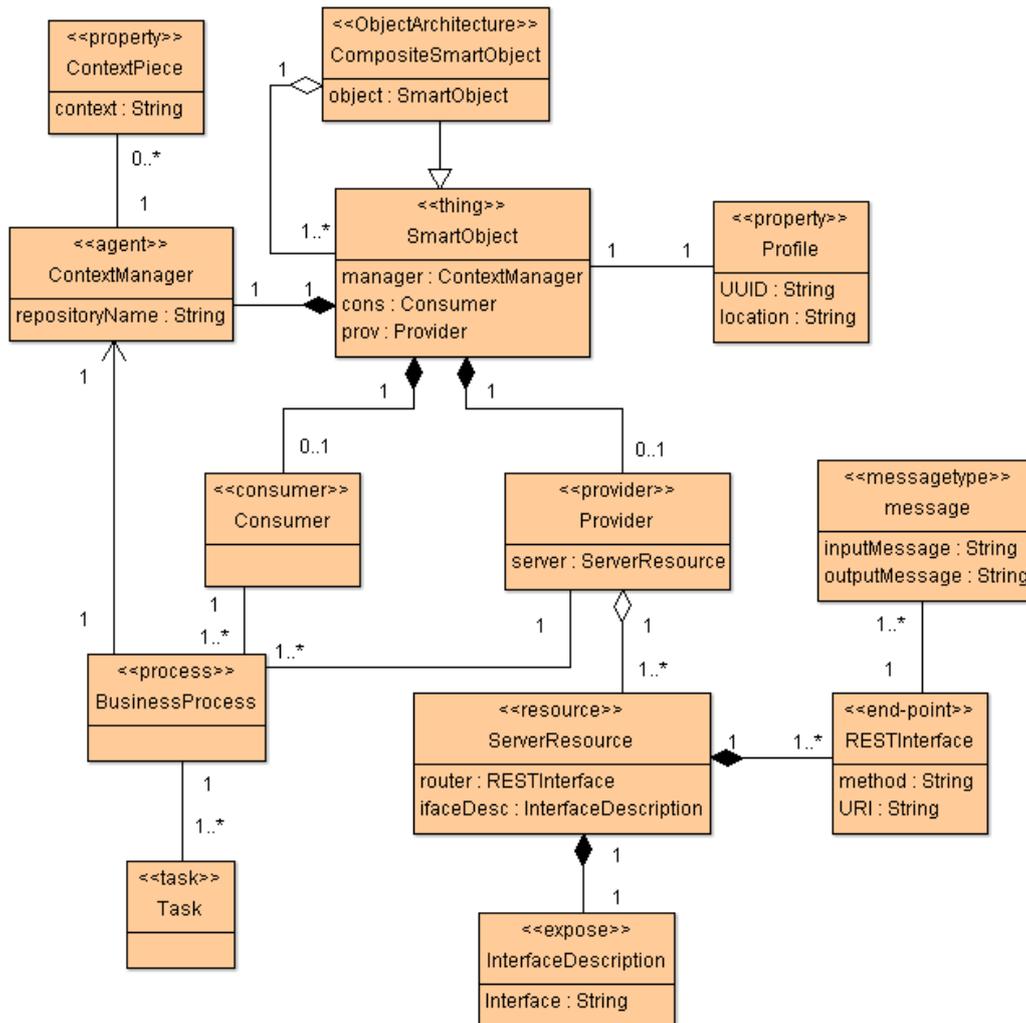


Figura 23 Perfil UML (Corredor et al., (2012))

Tomando estos autores como referencia se puede elaborar una versión sintetizada de un perfil UML que tome en cuenta el perfil desarrollado en la Sección 5.2 que no incorpora el concepto de IoT y un nuevo perfil que tome en cuenta esto último.

Para llevar a cabo esto, primero se debe presentar el metamodelo completo para ver cómo interactúan los paquetes Calidad y Trazabilidad del metamodelo considerado originalmente, con el nuevo paquete IoT. Esto es presentado en la Figura 24. En la figura se observan los tres paquetes, con sus respectivas clases diferenciadas por color, donde el verde, el azul y el rosa representan a clases de los paquetes Calidad, Trazabilidad y IoT respectivamente.

La clase Lote especializa a la EntidadFísica al mismo tiempo que también especializa a la EntidadTrazable. Este punto representa la principal unión entre los paquetes mencionados. La clase Responsable, del paquete Trazabilidad, especializa a la clase Usuario, del paquete IoT. Otro punto importante a destacar es que los dispositivos sensor, tag y actuador interactúan de manera directa con las entidades trazables, ya que monitorean, identifican y actúan, respectivamente, sobre las mismas.

A continuación se realizara una breve descripción de un Perfil UML del modelo IoT, que como se ha mencionado no será descrito de manera detallada ya que excede los objetivos de esta tesis. El PerfilIoT, Figura 25, se basa en los perfiles propuestos por (Corredor, Bernardos, Iglesias, & Casar, 2012), (Thramboulidis & Christoulakis, 2015) y (Costa, Pires, Delicato, Li, & Zomaya, 2016).

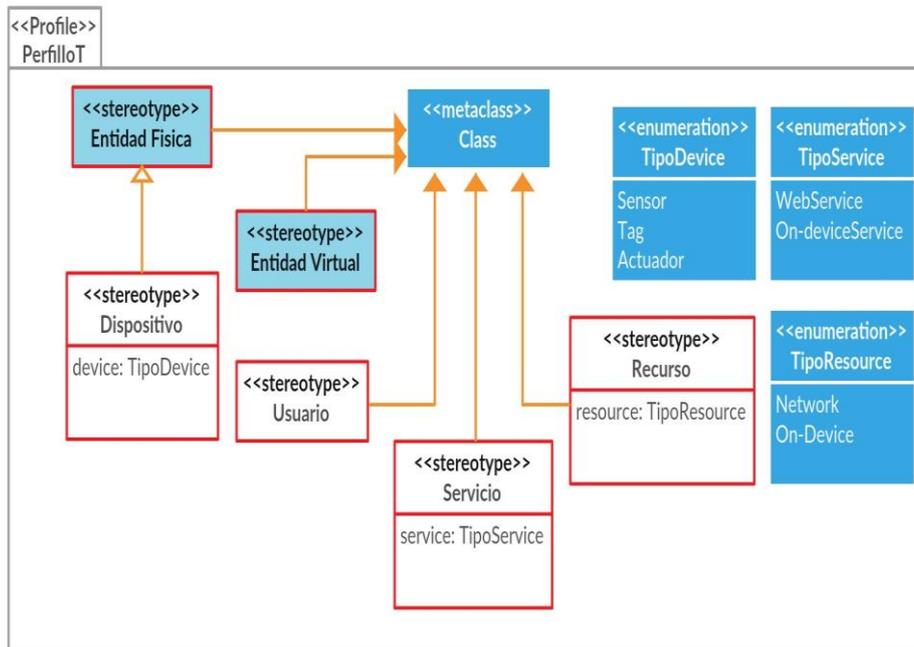


Figura 25 Perfil IoT propuesto

Vale recalcar que para realizar este perfil se han considerado algunas simplificaciones, como por ejemplo no tomar en cuenta el cliente y el servidor, para el modelo y tampoco se han especificado restricciones para el perfil.

El PerfilIoT está compuesto por seis estereotipos y tres enumeraciones que definen a los valores etiquetados *service*, *resource* y *device*.

En el próximo capítulo se dedicará una sección del mismo a ejemplificar el uso de este perfil para el ámbito apícola, en conjunto con el Perfil-TrazabilidadCalidad y de esta forma completar la demostración de que el perfil presentado en la Sección 5.2 es flexible a la incorporación de nuevos paquetes y/o tecnologías lo que lo hace robusto ante estas posibilidades.

6. EJEMPLO DE APLICACIÓN DEL PERFIL UML AL DOMINIO

APÍCOLA

6.1 Importancia de la producción Apícola en Argentina

Con una producción de aproximadamente 60.000 toneladas por año, la Argentina es el tercer productor de miel del mundo y el segundo exportador, según datos de la FAO. Alrededor del 95% de la producción se exporta, y el 80% de las exportaciones tienen por destino Estados Unidos, Alemania y Japón. Representan solo US\$168 millones anuales, pero tienen un alto impacto en las economías regionales del centro del país.

A pesar de que el consumo mundial se ha incrementado en gran medida en los últimos años, la producción de miel a nivel mundial enfrenta una crisis importante. Esto se debe principalmente al avance de la agricultura por encima de la ganadería, ya que esta actividad utiliza insecticidas y herbicidas que son incompatibles con la apicultura. Además otro gran problema es la aparición de mieles adulteradas con jarabe de arroz provenientes de China, que desestabilizan los precios del mercado al ser ofrecidas a un precio considerablemente menor que las mieles sin adulterar.

En el país existe una gran informalidad en la producción apícola, como resultado de muchos años sin controles eficaces. Esta informalidad hace que la industria penda de un hilo.

Actualmente se están tomando medidas para erradicar la informalidad del sector. Las principales medidas se basan en la creación de sistemas de trazabilidad integrales, que además de regularizar esta situación y agilizar los trámites de exportación, sirvan también para satisfacer la demanda creciente de los mercados internacionales de aumentar los controles en toda la cadena productiva y para agregar valor al producto.

6.2 Descripción del dominio Apícola

Un apicultor promedio en Argentina posee entre 100 y 200 colmenas. Estas se encuentran distribuidas generalmente entre 3 y 4 apiarios ubicados en distintos puntos geográficos. El apicultor suele visitar los apiarios para realizar tareas de mantenimiento y control de las colmenas cada 10 o 15 días en épocas de máxima actividad (primavera – verano) y un poco menos

frecuentemente durante el otoño e invierno. Actualmente se requiere que se lleve registro manual de todas las actividades que se realizan en cada visita al apiario, incluyendo registros de alimentación suplementaria, movimientos de colmenas entre apiarios, multiplicación de colmenas, los medicamentos que se han utilizado para curar las colmenas, recambio de reinas, entre otras actividades (Figura 26).

Además, se debe identificar cada colmena y, dependiendo de la granularidad que se pretenda, también podría identificarse cada uno de los cuadros en cada colmena, lo que, como desventaja, representaría un trabajo mucho mayor a la hora de la recolección de la información relevante. Esto es parte del problema de la “*first mile*” descrito en la Sección 3.5, y se pretende abordarlo mediante la utilización de tags RFID en cada colmena de manera que cuando se cargan los datos, estos queden asociados al tag de la colmena correspondiente. La ubicación de cada apiario puede ser descrita incluyendo datos como los tipos de floración existente dentro el radio de pecoreo (distancia que la abeja puede recorrer desde la colmena en busca de néctar y polen) que normalmente es de 3 km. A esta descripción también se puede agregar las actividades realizadas en los campos aledaños como fechas de aplicación de insecticidas o herbicidas por ejemplo. De esta forma se puede caracterizar el tipo de miel que se está obteniendo del apiario en cuestión.

Al momento de la cosecha los cuadros son transportados desde el apiario directamente a una sala de extracción de miel. Actualmente la forma en que se realiza este proceso en nuestro país no distingue la procedencia del cuadro que se va a procesar, es decir, se extrae todo en conjunto y luego se mezclan los cuadros indistintamente si venían de uno u otro apiario y de esta forma un tambor de miel, al salir de la sala de extracción, contiene miel de distintos tipos y procedencias. En el sistema de trazabilidad que se propone, y que se viene estudiando hace algunos años en el Instituto Nacional de Tecnología Agropecuaria (INTA), se sugiere que se registren las colmenas de las que proviene la miel y de ser posible se procese la producción de apiarios con características florales similares, evitando así la mezcla de distintos tipos florales, y obteniendo mieles “*mono-florales puras*”, lo que en el mercado tiene una apreciación mayor.

Nº de RENAPA		Observaciones (Anotar en esta columna lo que NO se encuentra en las columnas anteriores)	
Empresa: Responsable: Distancia (km):			
APIARIO	Código:	Material Melario - Cosecha	kg Neto Totales
	Nombre:		Nº de Lote de Extracción
	Recambio Multiplicación - Movimientos	Cambio Panales	Cantidad de Alzas Retiradas
			Cantidad de Alzas Colocadas
			Ingreso
			Destino
	Tratamientos Sanitarios	Alimentación	Logrados
			Reinas Fecundadas
			Celdas Reales
	Categorización	Cantidad por Colmena	Tipo
Cantidad			
Nº de Lote del Producto			
PLANILLA DE CAMPO	Fecha	Nombre Comercial del Producto	
		Cambio de Panales	
		Tipo de Alimento	
PLANILLA DE CAMPO	Cantidad de Colmenas	Tipo III	
		Tipo II	
		Tipo I	
PLANILLA DE CAMPO	Colmenas Muertas		

Figura 26 Planilla de campo para control de las actividades en el apiario.
(Dini & Bedascarrasbure, 2011)

Por lo antes mencionado, cada tambor tiene asociadas todas las colmenas de las cuales provino la miel que lo compone. En cuanto a la identificación del tambor, actualmente se está utilizando una metodología por la cual cada tambor que se utiliza ya posee un código asignado en su fabricación, de esta forma también se puede trazar su historia. Todas las actividades que se realizan dentro de la sala de extracción se pueden observar de manera más detallada en la siguiente Figura 27.

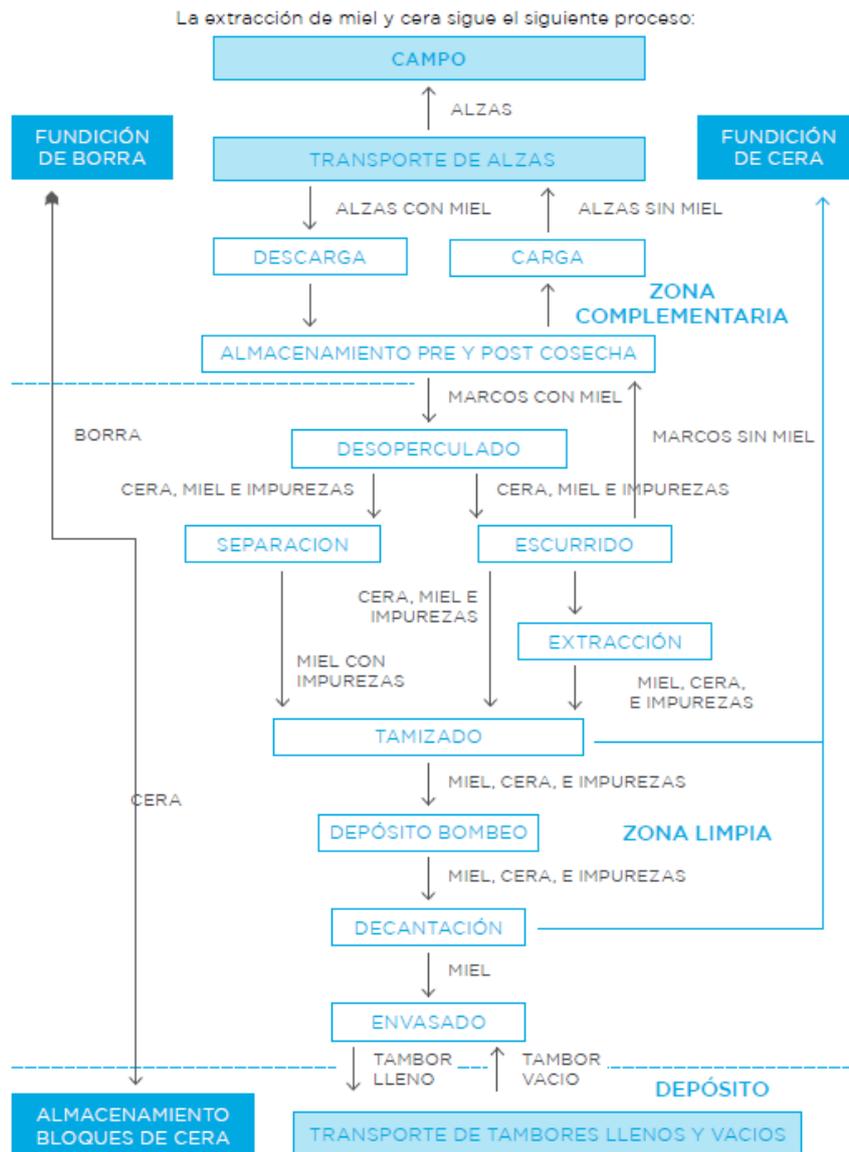


Figura 27 Diagrama de flujo de la miel y cera dentro de la sala de extracción (Ministerio de Agroindustria - República Argentina, 2016)

El siguiente paso en la cadena productiva, si se va a exportar a granel, es el traslado a una empresa acopiadora/exportadora. En esta se almacenan los tambores hasta tener una cantidad suficiente para exportar. Durante este transporte y almacenamiento se pretende que existan registros de tempera-

tura a la que son expuestos los tambores como un indicador de calidad, ya que las altas temperaturas incrementan la proliferación de agentes potencialmente tóxicos (por ej.: HMF – hidroximetil furfural: potencial agente carcinógeno) en la miel y que los mercados internacionales se encargan de controlar constantemente. La exportación se concreta mediante la utilización de barcos como medio de transporte de los tambores, los cuales son introducidos dentro de contenedores sin refrigeración. Por este motivo también es recomendable en esta etapa registrar la temperatura a la que son expuestos los tambores, ya que muchos de los cargamentos tienen como destino final países del hemisferio norte y deben cruzar la línea del ecuador, siendo sometidos, por lo tanto, a altas temperaturas.

En el caso que se vaya a fraccionar la miel obtenida en la sala de extracción, se registran las características de los envases utilizados y cada uno es identificado con el número de lote correspondiente al tambor del que proviene la miel que contiene y luego es enviado a los centros de venta a nivel regional o nacional.

Desde el punto de vista de la calidad, entre otras cosas, la miel debe cumplir ciertos estándares nacionales e internacionales que marcan los límites máximos en su composición en cuanto a la presencia de compuestos químicos y contaminantes microbiológicos. El exceso de alguno de estos contaminantes puede sugerir una falta general de higiene o malos manejos en el proceso de extracción, envasado y/o almacenamiento. Por esto los análisis químicos deben llevarse a cabo en laboratorios habilitados a tal fin, y conservarse como indicadores de la inocuidad del producto.

Algunos de los requisitos de calidad más importantes a nivel nacional según el Código Alimentario Argentino (ANMAT, 2010) son los siguientes:

- Humedad, por refractometría, Máx: 18,0%.
- Cenizas a 550-600°C: Máx: 1,0%.
- Azúcares reductores. Mín: 60%
- Sacarosa aparente. Máx: 10%
- Acidez, Máx: 40 miliequivalentes/kg. El valor de pH normal se encuentra comprendido entre 3.0 y 4.5

- Índice de diastasa (Escala de Gothe), Mín: 8. (Enzima encargada de la hidrólisis del almidón en glucosa. A mayor valor mayor calidad de la miel)
- Hidroximetilfurfural, Máx: 40 mg/kg. ; Max: 15 mg/Kg para mieles con escaso contenido de enzimas naturales. (Se produce por la descomposición térmica de glúcidos)

De todo lo descrito hasta aquí se pueden obtener los siguientes objetos que forman parte de la cadena productiva de la miel (Tabla 18):

Instancia	Clase
Cuadros/Alzas/Colmena	Lote
Apicultor	Responsable
RevisionDelApiario	Actividad
ApliqueDeMedicamentos	Actividad
AlimentacionSuplementaria	Actividad
Multiplicacion	Actividad
RecambioDeReinas	Actividad
Cosecha	Actividad
SaludDeLaColonia	RCCualitativo
CategoriaDeColonia	RCCualitativo
TransporteASala	Actividad
Apiario	Ubicacion
SalaDeExtraccion	Ubicacion
Tambor	Lote
ResponsableDeSala _n	Responsable
Carga/descargaDeAlzas	Actividad
AlmacenamientoDeAlzas	Actividad
Desoperculado	Actividad
Escurrido/Separacion	Actividad
Tamizado	Actividad
Bombeo	Actividad
Decantación	Actividad
Envasado	Actividad
CantidadDeHMF	RCNumerico
IndiceDiastasa	RCNumerico
Acidez	RCNumerico

Humedad	RCNumerico
Cenizas	RCNumerico
SacarosaAparente	RCNumerico
AzucaresReductores	RCNumerico
TransporteAAcopio	Actividad
Almacenamiento/Acopio	Actividad
UbicacionAlmacenamiento	Ubicacion
EncargadoDeAlmacenamiento	Responsable
Exportador	Responsable
Contenedor	Lote
TrasladoAlExterior	Actividad
ImportadorEnExterior	Responsable
UbicacionDestino	Ubicacion

Tabla 18 Listado de objetos de la cadena apícola

6.3 Aplicación del PerfilTrazabilidadCalidad

Cabe mencionar que para los siguientes ejemplos se han utilizados identificadores y fechas ficticias con el fin de simplificar la legibilidad de este trabajo. Así también como se han omitido varios atributos que en la realidad son necesarios para llevar a cabo una trazabilidad efectiva como lo son los datos personales del responsable, las propiedades y características del lote, etc. Se considera que los datos omitidos y simplificados no afectan al resultado final de esta tesis.

En la Figura 28 se observa la aplicación del PerfilTrazabilidadCalidad a una actividad genérica, a modo de ejemplo.

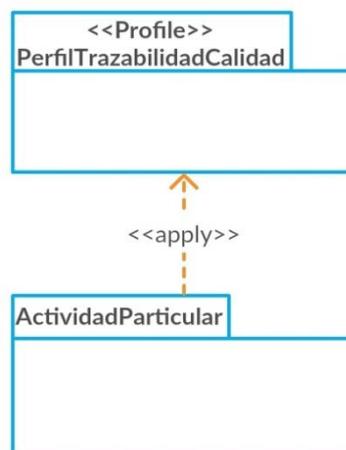


Figura 28 Aplicación del PerfilTrazabilidadCalidad

6.3.1 Curación

En el primer ejemplo a analizar (Figura 29) se puede ver una actividad de Curación. Esta actividad se lleva a cabo luego de detectar la presencia de ácaros, como la varroa, u otras enfermedades producidas por virus u hongos. Esta actividad se lleva a cabo sin la presencia de Alzas melarias en la colmena, ya que de ser así podrían quedar restos del medicamento utilizado en la miel que se pretende cosechar. Las alzas no serán colocadas hasta que finalice el periodo de carencia del compuesto químico colocado. Por este motivo también las curaciones se deben llevar a cabo antes del comienzo de la temporada primaveral y/o luego de la cosecha, en otoño.

En cuanto al diagrama UML presentado se observa que la actividad de curación es llevada a cabo por el Apicultor (*<<Responsable>>*). La línea de puntos que conecta a éste con la asociación entre la Colmena (*<<Lote>>*) y sí misma, representa a la relación que existe entre el responsable y el lote. Mediante esta relación, el objeto Apicultor se encarga de ejecutar la Operación de curación sobre la Colmena generando una colmena curada, pero sin necesidad de modificar el identificador de Lote de la colmena. De esta forma la operación de curación es una instancia del estereotipo *<<Alteracion>>* quien a su vez es una extensión de la metaclassa *<<Operation>>*.

Cabe destacar que los atributos correspondientes a la actividad son detallados con comentarios lo que facilita la comprensión del diagrama. En dicho comentario se especifican la *fechaDeInicio*, la *duración* y la *ubicación* donde se lleva a cabo la actividad.

También se puede observar la utilización del estereotipo *<<RCCualitativo>>* que representa al resultado obtenido luego de la curación, definiendo los tres casos en que la mortandad se mantiene, aumenta o disminuye, como parámetros de referencia del estado de salud de la colmena.

Otra cuestión a mencionar es el cumplimiento de las restricciones. En este ejemplo se puede ver como la ubicación del lote es coincidente con la ubicación donde se realiza la actividad: “APIA0002”, además del hecho de que la colmena y el apicultor tienen su identificador *<<ID>>*, por mencionar algunas.

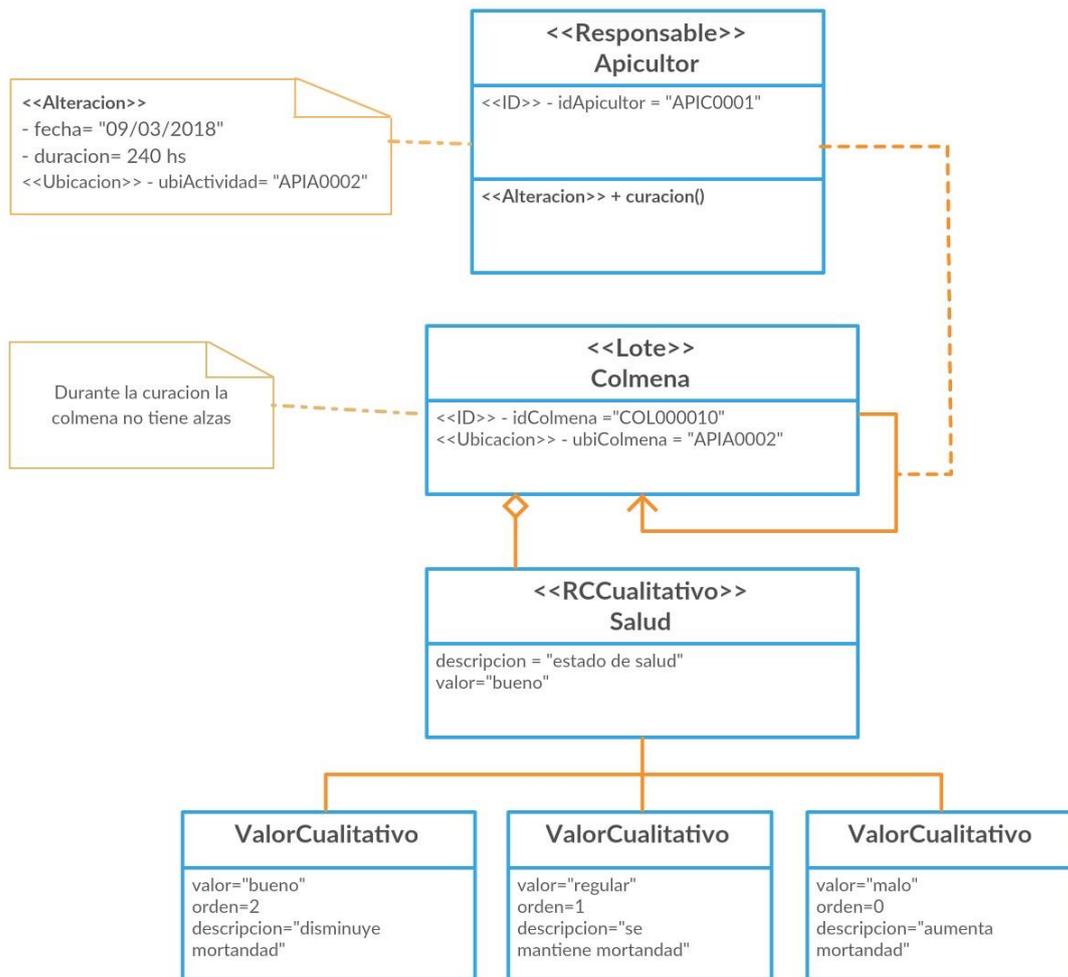


Figura 29 Ejemplo de aplicación en actividad de Curación

6.3.2 Multiplicación

En la Figura 30 se muestra un diagrama de clases con aplicación del perfil desarrollado, donde se puede ver la actividad de Multiplicación, la cual es una instancia, paradójicamente, del estereotipo `<<División>>`.

La multiplicación consiste en separar la colonia de una colmena para formar dos colonias independientes en dos, o más, nuevas colmenas. Esto se lleva a cabo cuando la colonia en cuestión llega a una cantidad de población lo suficientemente grande como para poder formar las dos colonias mencionadas y es un mecanismo para evitar la enjambrazón (proceso por el cual gran parte de la colonia abandona la colmena definitivamente al superar la cantidad de población que la colmena puede soportar) y también para aumentar la producción. Esto puede estar acompañado, o no, por la colocación de nuevas reinas. Existen otros métodos también, pero no serán tomados en cuenta por cuestiones de simplicidad.

En el diagrama presentado no se toma en cuenta la colocación de reinas nuevas. Ahí se observa como al aplicar la actividad de multiplicación por parte del Apicultor, la colmena “COL000001” se divide en “COL000002” y en sí misma. Vale aclarar que aquí se pueden realizar dos caminos posibles, el primero es el explicado, donde de una colmena se obtiene una segunda, y la primera mantiene su identificador; y el segundo consistiría en que la colmena “COL000001” generaría dos colmenas totalmente nuevas (“COL000002” y “COL000003” por ejemplo) y dejaría de existir.

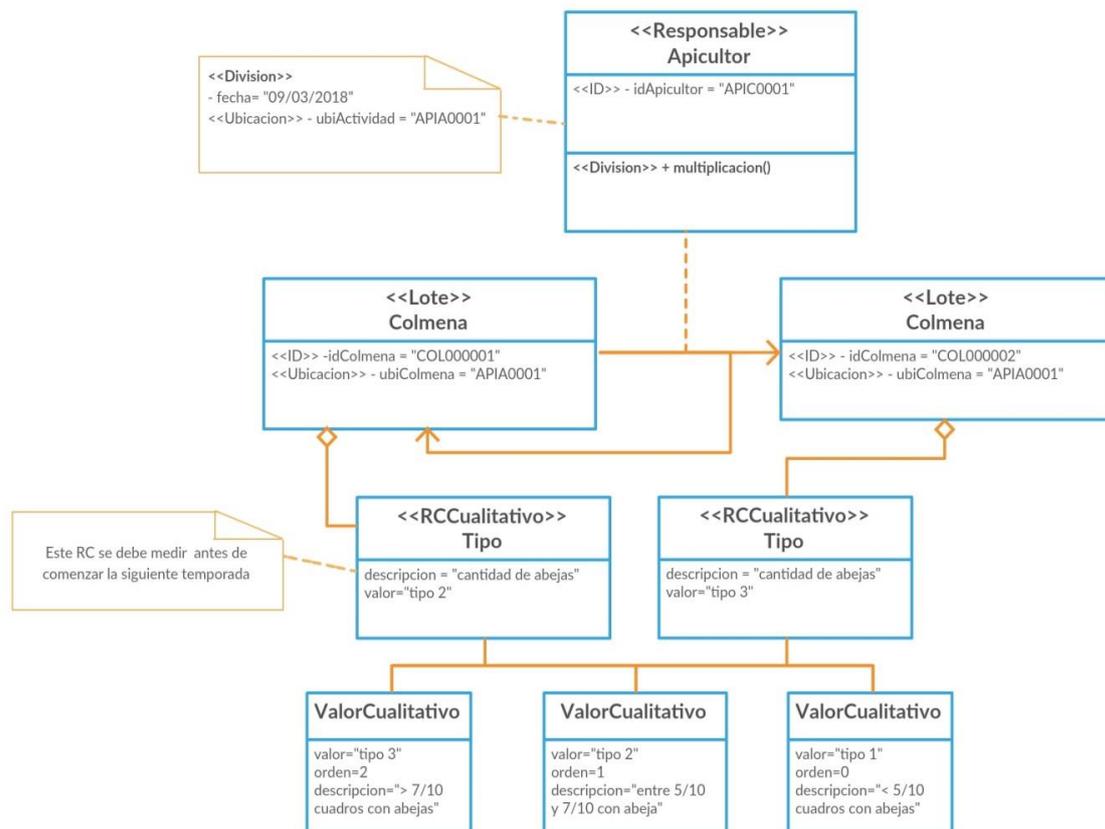


Figura 30 Ejemplo de aplicación en actividad de Multiplicación

Aquí también se puede ver que las colmenas no tienen Alzas melarias (<<Lote>>) ya que esta actividad se lleva a cabo al finalizar la temporada productiva, es decir cuando las alzas ya fueron cosechadas.

Como multiplicación se realiza al finalizar la temporada productiva, antes de comenzar la siguiente temporada se debe verificar el estado de las colmenas, para esto se utiliza un indicador que se puede medir a simple vista, y es la cantidad de abejas que se observa. La cantidad de cuadros cubiertos con abejas al levantar la tapa de la colmena se clasifica en 3 tipos o ca-

tegorías: Tipo I, Tipo II y Tipo III, de menor a mayor cantidad de cuadros cubiertos respectivamente.

6.3.3 Colocación de alzas

En este caso, en la Figura 31 se puede ver cómo funciona la actividad de `<<Union>>`. El ejemplo representa la operación de colocación de Alzas melarias encima de una Colmena, por lo que la colmena en cuestión, que puede o no tener alzas previamente colocadas, se une a las alzas conteniéndolas hasta que éstas estén listas para ser cosechadas. Esto se hace visible en dicha figura, donde el lote Colmena se une a los sublotes Alza, donde el primero no cambia su identificación pero cambia su contenido debido a que ahora contiene dos nuevas alzas. Esto representa una de las mejoras que se plantean respecto a los metamodelos vistos en la bibliografía, donde luego de esta actividad el lote resultante se debería renombrar, sin embargo aquí no se hace y representa fielmente a la realidad donde la colmena real sigue siendo la misma antes y después de la colocación de alzas.

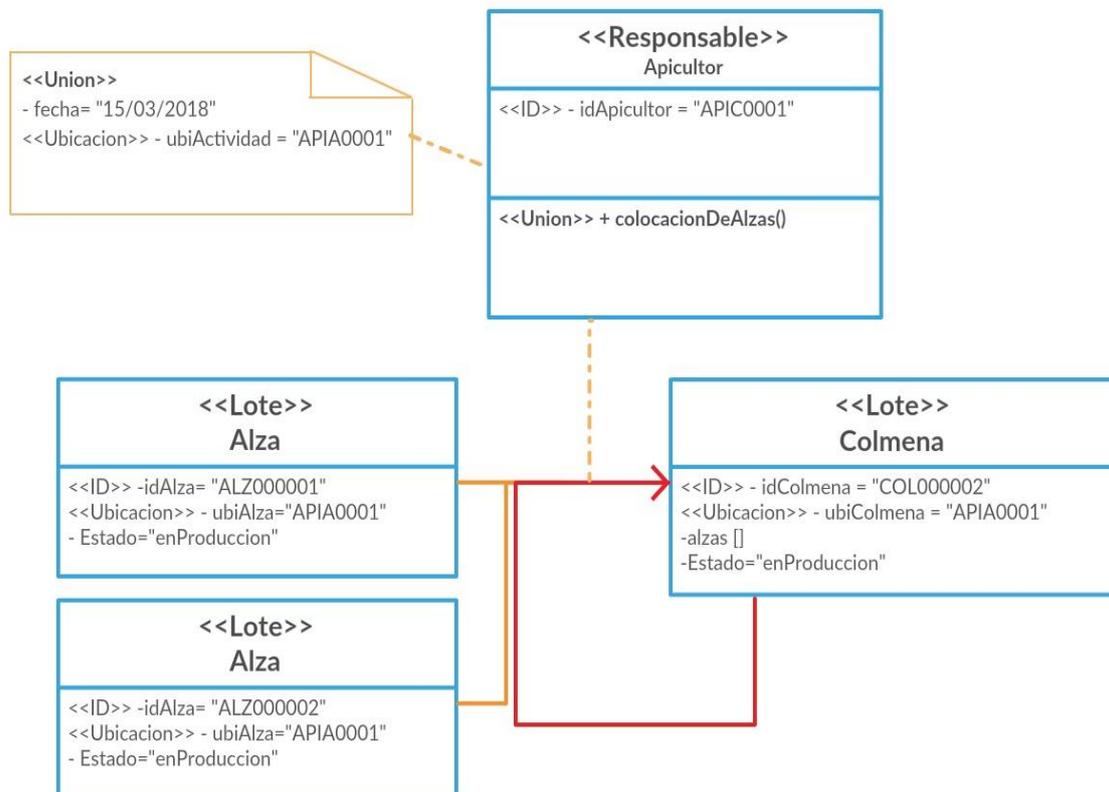


Figura 31 Ejemplo de aplicación en actividad de Colocación de Alzas

También se observa el atributo *Estado* en “*enProduccion*” de la clase *Alza* y permanecerá así hasta que sus cuadros se completen de miel y esté listo para ser cosechado, momento en el cual pasará a “*terminado*”.

6.3.4 Cosecha

En la Figura 32 se representa la actividad de cosecha de miel.

Al momento de la cosecha las Alzas que están en la colmena se encuentran listas para ser transportadas a la sala de extracción, por lo que la Operación de Cosecha implica una división de la colmena en: Colmena + Alzas que la componen.

Las alzas serán trasladadas a la sala de extracción mientras que la colmena quedara en el apiario, y acto seguido, si todavía existe tiempo para producir más miel, se colocaran nuevas alzas mediante una actividad de Unión como se describió en la sección 6.3.3, en caso contrario la colmena será preparada para afrontar la temporada invernal. Aquí ocurre algo similar a la actividad de colocación de alzas pero a la inversa, donde a pesar que le son retirados los sublotes Alza a la Colmena, lo que cambia su composición, esta sigue manteniendo su identificación, ya que la colmena sigue siendo la misma.

En el ejemplo considerado en la figura se observa como la colmena resultante todavía se encuentra en estado “*enProduccion*”, razón por la cual se le han colocado 2 nuevas alzas.

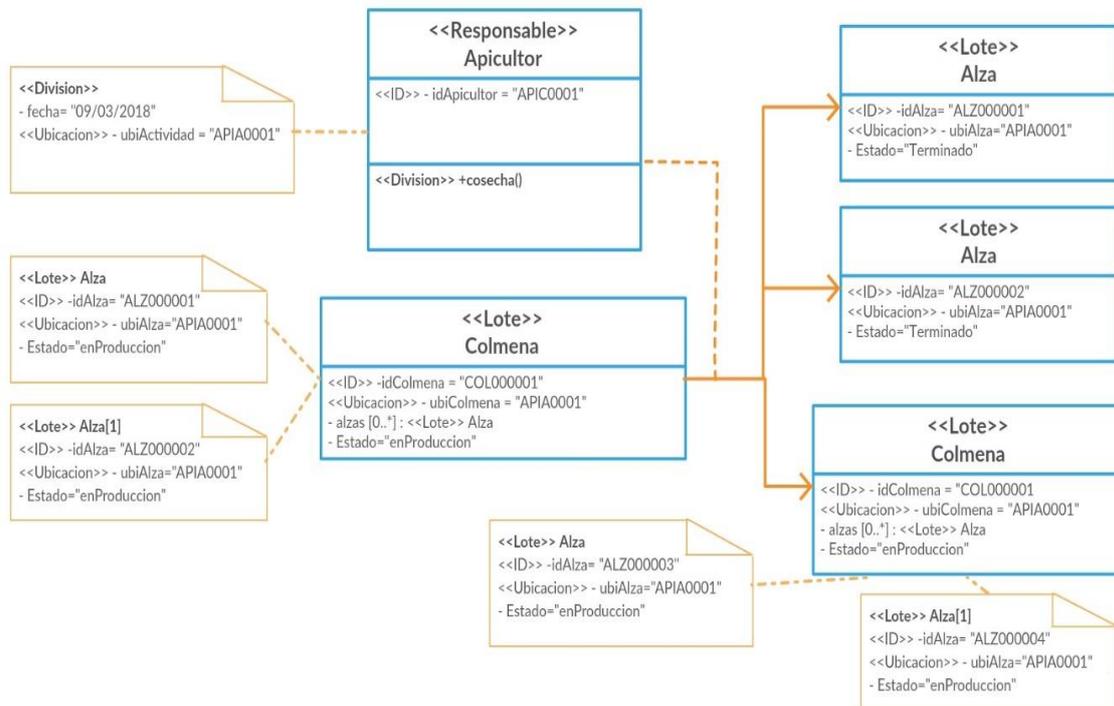


Figura 32 Ejemplo de aplicación en actividad de Cosecha

6.3.5 Desoperculado y envasado

En la Figura 33 se revisará un ejemplo de aplicación de algunas actividades que se llevan a cabo dentro de la sala de extracción.

Luego de la cosecha las alzas son transportadas a la sala de extracción. Una vez allí, alguien se encarga de recibirlas (actividad de adquisición no reflejada en la Figura 33) y de comenzar el proceso de extracción.

El RespDeSala1 se encarga de colocar las “n” alzas cosechadas en la desoperculadora y luego se escurre la miel, uniéndose en el lote de miel denominado “MIELSUCIA” en la figura. Es por esta razón que este lote posee una lista de alzas.

En el ejemplo se considera que el RespDeSala1 le realiza una transferencia de responsabilidades al RespDeSala2 para que este último realice las actividades siguientes. Por esto el primero realiza una actividad de Suministro al segundo, mientras que el segundo realiza una actividad de adquisición del primero. En este proceso no se modifica el lote “MIELSUCIA”.

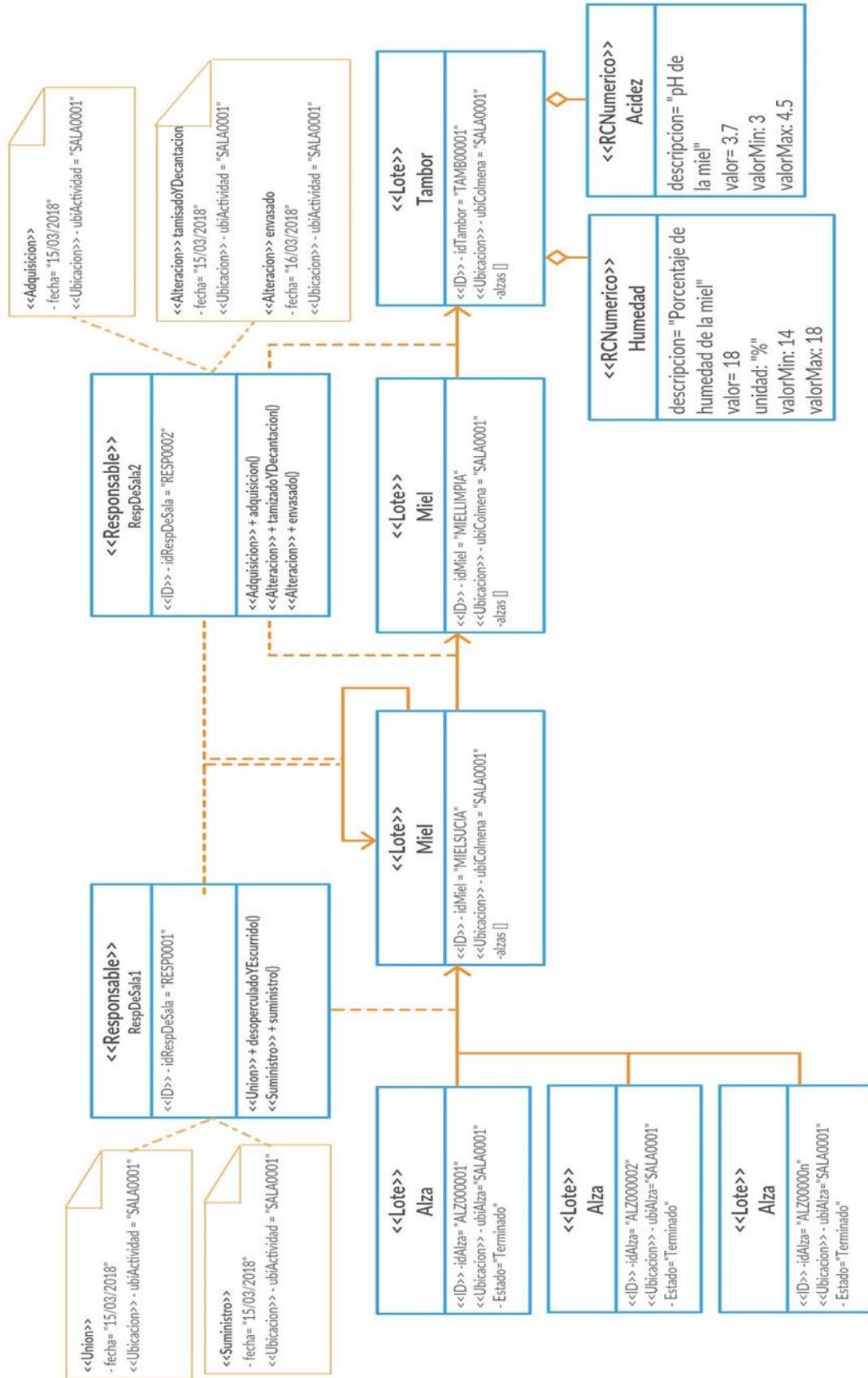


Figura 33 Ejemplo de aplicación en actividades de Desoperculado, Envasado, Adquisición y Suministro

Luego de realizada la transferencia, el RespDeSala2 somete a la “MIELSUCIA” a la operación de tamizadoYDecantacion para obtener un nuevo lote de miel denominado “MIELLIMPIA”. La que luego será envasada creando un nuevo <<Lote>> Tambor, cuyo identificador quedara asociado a todas las Alzas que produjeron la miel que contiene.

Una vez envasado se realizan los controles de calidad de miel pertinentes. A modo de ejemplo se muestran los controles realizados sobre la Humedad y la Acidez de la miel producida.

Un punto a resaltar de este diagrama es que se puede apreciar como un Responsable puede ejecutar muchas actividades siempre que sean consecutivas, y no necesita realizar actividades de adquisición y suministro cada vez que pasa de una actividad a otra (siempre que él mismo sea el responsable de esas actividades), como se sugería en la bibliografía revisada.

6.4 Aplicación de PerfilTrazabilidadCalidad y PerfilIoT

Para el siguiente ejemplo se ha considerado la aplicación de ambos perfiles presentados en esta Tesis al dominio apícola, con el fin de demostrar lo que se planteó en la Sección 5.6, y comprobar la facilidad de adaptación del perfil diseñado a nuevas tecnologías, en este caso la tecnología de la Internet de las Cosas.

En la Figura 34 se puede ver como se aplican ambos perfiles a la actividad de cosecha de miel, de lo que resulta el siguiente diagrama (Figura 35) en el cual se representa una situación en la que cada objeto trazable es identificado por un tag RFID, al igual que el apicultor responsable de la actividad y de la colmena en cuestión. Los tags son leídos por un lector RFID que registra los códigos de identificación de cada objeto.

También se incorpora un data logger que registra el peso de la colmena y la humedad de la miel, estas son variables que representan de manera fehaciente el estado de madurez de la miel y por consiguiente si es factible llevar a cabo la cosecha. El apicultor consulta dichas variables mediante un servicio web que le informa al instante lo que se está registrando en el data logger.

Existe también una entidad virtual por cada entidad física representada en el esquema. En esta entidad virtual se registra la información que se rescata de los dispositivos asociados a las entidades físicas.

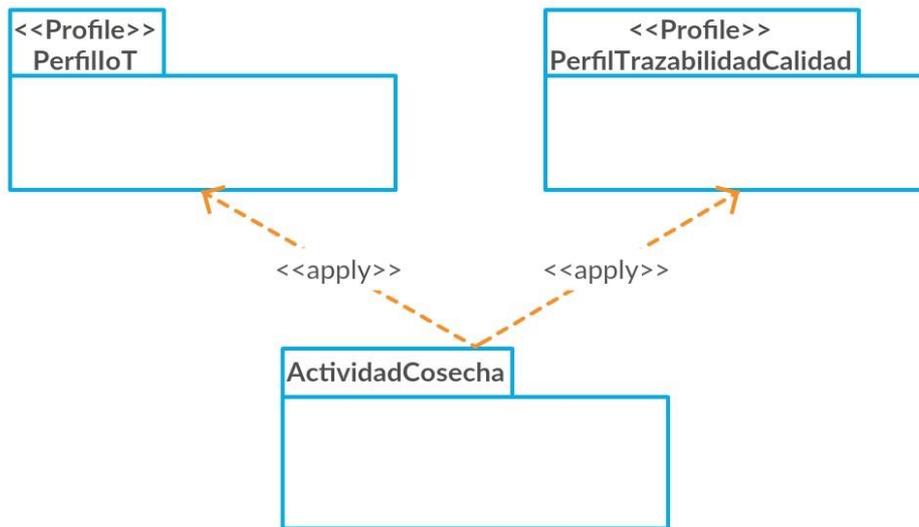


Figura 34 Aplicación de los perfiles PerfilTrazabilidadCalidad y PerfilloT

Con el fin de simplificar el esquema se han omitido las entidades virtuales de las alzas melarias, por lo que solo se ha agregado al diagrama la de la colmena virtual. Con la misma finalidad, también se han omitido los tags de las alzas melarias definidas en comentarios.

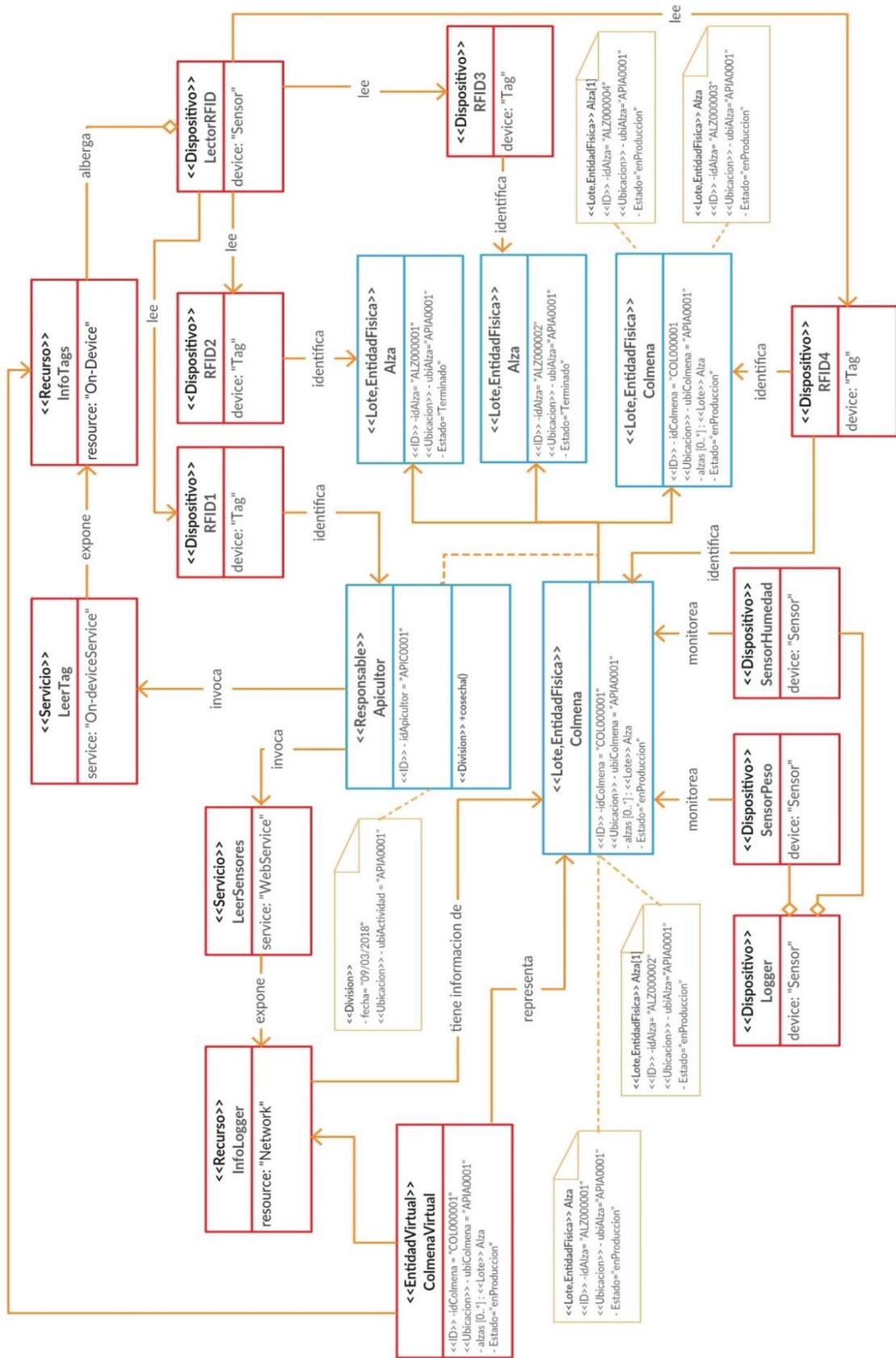


Figura 35 Ejemplo de aplicación de ambos perfiles a una actividad de Cosecha con tecnologías IoT

7. CONCLUSIONES

En este trabajo se han estudiado los Perfiles UML como una alternativa viable a los DSL tradicionales para resolver los problemas que se presentan en el área de control de calidad y trazabilidad de los alimentos.

Se ha determinado que los mayores problemas que enfrenta este sector productivo se deben a la falta de unificación de criterios en cuanto a lo que representa la trazabilidad del producto para cada actor de la cadena productiva y en qué cantidad de información están dispuestos a compartir con el resto de los actores. Esto deriva en diferencias substanciales en el tipo y cantidad de información que se recolecta de los productos, y también en los medios que se utilizan para tal fin. Como solución a estos problemas de interoperabilidad se ha propuesto un modelo para el diseño de sistemas de trazabilidad enfocado en el producto, considerando que es éste el que recorre todos, o gran parte, de los eslabones en las diferentes formas que va tomando a lo largo de la cadena.

El modelo propuesto presenta ventajas en cuanto al registro de las actividades que se realizan sobre los lotes. Ya que, a diferencia de los modelos vistos en la bibliografía, evita realizar operaciones de suministro y adquisición cuando no hay cambio de responsable del producto; y también se evita cambiar de identificación de lote luego de cada actividad, llevándose a cabo el cambio solo en los casos en que sea estrictamente necesario.

Se ha estudiado el uso de la tecnología IoT como una forma de maximizar los controles y la automatización de los procesos, al mismo tiempo que se facilita el acceso a la información en tiempo real de los responsables de las diferentes etapas. Se ha concluido que esta tecnología resulta ser significativamente útil a la hora de realizar modificaciones en los procesos y/o retirar productos del mercado debido a la detección de fallas o defectos en los mismos. También se concluye que esta tecnología representa una de las mejores soluciones al problema presentado como *first mile*, ya que brinda grandes posibilidades de monitoreo y control en el punto de origen de la producción de alimentos, así también como nuevas formas de identificar los alimentos aumentando la granularidad del sistema de trazabilidad.

El Perfil UML desarrollado se ha evaluado cuantitativamente mediante la aplicación de una serie de métricas que permiten dar cuenta que si

bien el perfil es muy dependiente de la semántica UML, posee una complejidad media, lo que lo hace relativamente fácil de interpretar y además facilita la posibilidad de ser parametrizado por los usuarios. Desde el punto de vista cualitativo se ha realizado una comparación con un ejemplo sencillo de aplicación del Perfil UML donde se aprecian las grandes ventajas al momento de interpretar el modelo, ya que ha disminuido notablemente la cantidad de objetos presentes y las relaciones entre ellos, dándole también un significado conceptualmente más correcto a las actividades que se realizan sobre los lotes al considerarlas como operaciones, y generando un diagrama que representa de manera más adecuada la realidad de las interacciones que se llevan a cabo entre los involucrados.

También se ha realizado un análisis de la compatibilidad del Perfil UML desarrollado con las tecnologías IoT, concluyendo que el PerfilTrazabilidadCalidad es compatible y robusto frente a la utilización de nuevas tecnologías de trazabilidad y control de calidad.

Por último, se ha aplicado el Perfil UML a la cadena de producción apícola. Estudiando algunos ejemplos de actividades propias de la apicultura se obtuvieron buenos resultados representando los objetos, respetando las restricciones propias de la definición del perfil, y logrando una buena modelización del dominio apícola mediante la utilización del PerfilTrazabilidadCalidad.

Como conclusión general se puede resaltar que los objetivos fueron satisfechos en su totalidad presentando un estudio comparativo de los modelos encontrados en la bibliografía y analizando patrones de diseño de los mismos; lo que sirvió como punto de partida para llevar a cabo el desarrollo del DSL para el dominio planteado, utilizando los conceptos de Perfiles UML.

8. TRABAJOS FUTUROS

Dentro de las implicancias a futuro se considera que una de las más importantes es la utilización de alguna herramienta de diseño UML para lograr automatizar el diseño de sistemas para el dominio de la producción de alimentos mediante la utilización del Perfil UML presentado en la Sección 5.2. Esto brindará resultados cuantitativos más detallados para poder comparar este Perfil UML con otros perfiles estandarizados.

También, desde el punto de vista de la obtención de parámetros de comparación, se puede realizar un estudio de las diversas métricas existentes para los diagramas de clases UML y sus posibles usos para el análisis del perfil diseñado en este trabajo, ampliando de esta forma el análisis cuantitativo llevado a cabo aquí y poniendo un punto de partida para futuros nuevos perfiles que puedan derivar de éste.

Si bien en este trabajo se consideró el dominio de la producción de alimentos, los trabajos futuros se pretende que estén dedicados a estudiar la viabilidad de aplicación de este perfil para otros dominios similares, con la menor cantidad de modificaciones posibles. Ejemplos de estos dominios pueden ser la producción de medicamentos, electrónica, desarrollo de software, productos de consumo en general, etc.; donde cada dominio en particular significa diferentes maneras y diferentes tecnologías para llevar a cabo el control y monitoreo de los productos.

Otro desprendimiento del actual trabajo es el análisis del impacto del Perfil TrazabilidadCalidad y los metamodelos presentados, en el mundo real, cuantificando diferencias, tanto económicas (infraestructura, personal e inversiones necesarias) como sociales (predisposición a la adopción y/o a la capacitación), respecto a los modelos existentes.

Se propone desarrollar una arquitectura MDA, donde se utilizará el perfil UML como transformación entre el CIM y el PIM, pudiendo también combinarlo con algún otro DSL tradicional de manera de poder aprovechar al máximo las ventajas de ambos DSLs, para luego aplicar perfiles UML propios de diferentes plataformas a lo anterior y así obtener los PSMs para un sistema de trazabilidad alimentaria multiplataforma, mejorando así la interoperabilidad.

9. REFERENCIAS

- Abouzahra, A., Bézivin, J., Del Fabro, M. D., & Jouault, F. (2005). A practical approach to bridging domain specific languages with UML profiles. *Proceedings of the Best Practices for Model Driven Software Development at OOPSLA*, 5.
- ANMAT. (2010). *Código Alimentario Argentino - Capítulo X*. Obtenido de ANMAT - Administracion Nacional de Medicamentos, Alimentos y Tecnología Médica: http://www.anmat.gov.ar/alimentos/normativas_alimentos_caa.asp
- ANMAT. (2017). *Portafolio Educativo - Capitulo 10*. Obtenido de ANMAT - Administracion Nacional de Medicamentos, Alimentos y Tecnología Médica: http://www.anmat.gov.ar/portafolio_educativo/home.asp
- Azram, N. A., & Atan, R. b. (2011). Software engineering traceability method for food ingredient tracing. *2011 Malaysian Conference in Software Engineering*, (págs. 141-145). Johor Bahru, Malasia.
- Bauer, M. (2013). IoT Reference Model. En A. Bassi, R. van Kranenburg, M. Bauer, S. Lange, M. Fiedler, S. Meissner , & T. Kramp, *Enabling Things To Talk* (págs. 113-162). Berlin, Heidelberg: Springer.
- Bechini, A., Cimino, M. G., Marcelloni, F., & Tomasi, A. (2008). Patterns and technologies for enabling supply chain traceability through collaborative e-business. *Information and Software Technology*, 50(4), 342-359.
- Bechini, A., Cimino, M., Lazzerini, B., Marcelloni, F., & Tomasi, A. (2005). A General Framework for Food Traceability. *2005 Symposium on Applications and the Internet Workshops*, (págs. 366-369). Trento, Italia.
- Codex Alimentarius Commission. (2006). *Codex Alimentarius - CAC/GL 60-2006*. Recuperado el 08 de Mayo de 2017, de Food and Agriculture Organization (FAO) and World Health Organization (WHO) of the United Nations: <http://www.fao.org/fao-who-codexalimentarius>
- Corredor, I., Bernardos, A. M., Iglesias, J., & Casar, J. R. (2012). Model-driven methodology for rapid deployment of smart spaces based on resource-oriented architectures. *Sensors*, 12(7), 9286-9335.

- Costa, B. R., Pires, P. F., Delicato, F. C., Li, W., & Zomaya, A. Y. (2016). Design and Analysis of IoT Applications: A Model-Driven Approach. *14th Intl Conf on Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech)* (págs. 392-399). IEEE.
- Dalgarno, M., & Fowler, M. (2008). UML vs. domain-specific languages. *Methods and Tools*, 16(2), 2-8.
- Dini, C., & Bedascarrasbure, E. (2011). *Manual de apicultura para ambientes subtropicales*. Tucuman: INTA.
- European Commission. (2002). *Food Law General Requirements*. Obtenido de <https://ec.europa.eu/>
- Evizal, A., Rahim, S., Rahman, T., & Rosa, S. (2016). 10 - Traceability Software for the Food Industry. En M. Espiñeira, & F. J. Santaclara, *Advances in Food Traceability Techniques and Technologies. Improving Quality Throughout the Food Chain* (págs. 191-206). Woodhead Publishing.
- Fuentes, L., & Vallecillo, A. (2004). Una introducción a los perfiles UML. *Novática*, 168, 6-11.
- García-Díaz, V., Fernández-Fernández, H., Palacios-González, E., G-Bustelo, B. C., Sanjuán-Martínez, O., & Lovelle, J. M. (2010). TALISMAN MDE: Mixing MDE principles. *Journal of Systems and Software*, 83(7), 1179-1191.
- Gemesi, H. G. (2010). Food Traceability Information Modeling and Data Exchange and GIS Based Farm Traceability Model Design and Application. *Graduate Theses and Dissertations*. Ames, Iowa.
- Giachetti, G., Marín, B., & Pastor, O. (2009a). Using UML as a domain-specific modeling language: A proposal for automatic generation of UML profiles. *Advanced Information Systems Engineering - 21st International Conference, CAiSE*, (págs. 110-124). Amsterdam, Netherlands.
- Giachetti, G., Marín, B., & Pastor, O. (2009b). Using UML Profiles to Interchange DSML and UML Models. *Third International Conference on Research Challenges in Information Science, RCIS*.

- GS1. (12 de 11 de 2018). *GS1 Argentina - El lenguaje mundial de los negocios*. Obtenido de www.gs1.org.ar
- International Organization for Standardization. (2015). ISO 9000 - Quality Management Standards.
- Jansen-Vullers, M., van Dorp, C., & Beulens, A. (2003). Managing traceability information in manufacture. *International Journal of Information Management*, 23, 395–413.
- Khabbazi, M., Ismail, N., Ismail, M., & Mousavi, S. (2009). Data Modeling of Traceability Information for Manufacturing Control System. *2009 International Conference on Information Management and Engineering*, (págs. 633-637). Kuala Lumpur.
- Kim, H. M., Fox, M. S., & Gruninger, M. (1995). Ontology of quality for enterprise modeling. *Fourth Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises* (págs. 105-116). IEEE.
- Manikas, I., & Manos, B. (2009). Design of an integrated supply chain model for supporting traceability of dairy products. *International Journal of Dairy Technology*, 62, 126–138.
- Mernik, M., Heering, J., & Sloane, A. (2005). When and how to develop domain-specific languages. *ACM Computing Surveys*, 37(4), 316-344.
- Ministerio de Agroindustria - Republica Argentina. (2016). *Guía de Buenas Prácticas Apícolas y de Manufactura*.
- OCG. (2017). *Open Geospacial Consortium*. Obtenido de <http://www.opengeospatial.org/>
- OMG. (2014). *Documents Associated With Object Constraint Language (OCL), Version 2.4 - formal-14-02-02.pdf*. Obtenido de <http://www.omg.org>
- OMG. (2014). *MDA Guide rev. 2.0 - OMG Document ormsc/2014-06-01*. Obtenido de <http://www.omg.org>
- OMG. (2014). *Meta Object Facility (MOF) Specification - formal/2014-04-05*. Obtenido de <http://www.omg.org/>
- OMG. (2015). *Unified Modelling Language (UML) Specification - formal/2015-03-01*. Obtenido de <http://www.omg.org>
- Opara, L. (2003). Traceability in agriculture and food supply chain: a review of basic concepts, technological implications, and future prospects. *Food, Agriculture & Environment*, 1(1), 101-106.

- Panetto, H., & Petin, J.-F. (2005). Metamodelling of production systems process models using UML stereotypes. *International Journal of Internet and Enterprise Management*, 3(2), 155-169.
- Rinconballesteros, D. L. (2016). *Conceptualizacion de la trazabilidad en la cadena de abastecimiento*. Bogotá. Colombia: Universidad Distrital Francisco Jose de Caldas.
- Robert, S., Gérard, S., Terrier, F., & Lagarde, F. (2009). A Lightweight Approach for Domain-Specific Modeling Languages Design. *35th Euromicro Conference on Software Engineering and Advanced Applications*. Patras, Greece.
- Ruiz-Garcia, L., Steinberger, G., & Rothmund, M. (February de 2010). A model and prototype implementation for tracking and tracing agricultural batch products along the food chain. *Food Control*, 21(2), 112-121.
- SENASA. (2015). *Servicio Nacional de Sanidad y Calidad Agroalimentaria*. Obtenido de <http://www.senasa.gov.ar>
- Silingas, D., Vitiutinas, R., Armonas, A., & Nemuraite, L. (2009). Domain-Specific modeling environment based on UML Profiles. *15th International Conference on Information and Software Technologies*, (págs. 167-177). Kaunas, Lithuania.
- Slorach, S. A. (28-30 de Enero de 2002). *Enfoques integrados para la gestión de inocuidad de los alimentos a lo largo de toda la cadena alimentaria*. Recuperado el 05 de Mayo de 2017, de Foro Mundial FAO/OMS de las Autoridades de Reglamentación sobre Inocuidad de los Alimentos - Depósito de documentos de la FAO: http://www.fao.org/docrep/meeting/004/y1956s.htm#P42_5573
- Steinberger, G., Rothmund, M., & Auernhammer, H. (2006). Agricultural process data service (APDS). *World congress agricultural engineering for a better world* (págs. 271–272). Bonn: VDI-Verlag.
- Sundmaeker, H., Verdouw, C., Wolfert, S., & Pérez-Freire, L. (2016). Internet of Food and Farm 2020. *River Publishers*, 129-151.
- Thomas, D. (2004). MDA: revenge of the modelers or UML utopia? *IEEE Software*, 21(3), 15-17.
- Thramboulidis, K., & Christoulakis, F. (2015). UML4IoT - A UML profile to exploit IoT in cyber-physical manufacturing systems. *Computers in Industry*.

- Tursi , A., Dassisti, M., & Panetto, H. (2007). Products information interoperability in manufacturing systems. *Ottavo Convegno AITeM (Associazione Italiana Tecnologia Meccanica)*. Montecatini Terme, Italia.
- Verdouw, C., Wolfert, J., Beulens, A., & Rialland, A. (2016). Virtualization of food supply chains with the internet of things. *Journal of Food Engineering*,, 176, 128-136.
- Weidemann, T., Hüttner, T., & Hanisch, F. (2008). Model-Driven Architecture for an Interactive Ajax Mapping Platform . En *Model Driven Software Engineering - Transformations and Tools* (págs. 118-134). Berlin.