



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

TRAZABILIDAD DE PRODUCTOS UTILIZANDO ONS Y UN SISTEMA
DE MODELAMIENTO DE OBJETOS TRAZABLES

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN
COMPUTACIÓN

ANDRÉS IGNACIO PÉREZ GALAZ

PROFESOR GUÍA:
JOSÉ PIQUER GARDNER

MIEMBROS DE LA COMISIÓN:
NELSON BALOIAN TATARYAN
JENS HARDINGS PERL

SANTIAGO DE CHILE
NOVIEMBRE 2010

Resumen Ejecutivo

Hace un tiempo ya que la trazabilidad ha dejado de ser un beneficio y ha pasado a ser una necesidad. Un claro ejemplo de la importancia de este tema se puede ver en empresas dedicadas al rubro alimenticio, ya que el tener un historial de sus productos, desde el momento de su creación hasta su venta o uso, resulta ser algo vital para ellas. Esta clara necesidad de trazabilidad, junto con poder aplicar las diversas tecnologías existentes hoy en día, resultó ser un gran motivante para desarrollar una solución para satisfacer dicha necesidad.

El problema que fue posible observar es que en la actualidad el control de trazabilidad, en muchas empresas, aún no es visto como una necesidad vital e importante, o bien, no es controlada de manera eficiente. Sea cual sea de estos escenarios, la falta de un control de trazabilidad puede llevar a la toma errónea de decisiones, ya que no existe la información disponible que permita tener una visión mas clara de la situación, lo que puede terminar resultando muy costoso.

Para lograr llevar una trazabilidad completa de los productos y así facilitar las decisiones de negocio, se creó un sistema de trazabilidad con ciertas características, entre las cuales destacan su flexibilidad, adaptabilidad y modularidad. Además dicho sistema se construyó de tal manera que fuera posible utilizar las tecnologías como ONS (*Object Naming Service*), ampliando así la capacidad de poder consultar sobre la trazabilidad de un producto en cualquier parte del mundo.

El sistema construido logró ser un sistema capaz de adaptarse a cualquier tipo de negocio, entregando la trazabilidad de sus productos. Además se integró un módulo destinado a la interacción con ONS.

Agradecimientos

Agradezco a mi hermano y madre por su preocupación, apoyo y cariño durante todos estos años. A mi padre, que en paz descanse, por todo lo que me enseñó en todo ámbito de cosas, muchas gracias viejito. Sin el cariño, esfuerzo y apoyo de mi madre y padre no estaría escribiendo esto.

También a mi novia por toda su ayuda, por sus palabras de apoyo cuando las necesité, por su compañía en los momentos difíciles, por su amor y por todas las noches desveladas que estuvo a mi lado.

Finalmente quiero darle las gracias a Rodrigo Arenas (a.k.a. Gogo) por su gran ayuda, confianza y por todo lo que me ha enseñado durante el tiempo que llevo trabajando para él.

Andrés Pérez Galaz.

Índice General

Resumen Ejecutivo	I
Agradecimientos	II
Introducción	1
Antecedentes	4
Especificación del Problema	9
A . Descripción del Problema	9
A .1. Diversidad en modelos de negocio	9
A .2. Integración con otras tecnologías	10
B . Requisitos de la Solución	11
Descripción de la Solución	15
A . Diseño Base de Datos	16
B . Arquitectura del software	17
C . Funcionalidades SMOT	18
C .1. Crear Objeto Trazable	20
C .2. Creación de Usuario	25
C .3. Agregar Base de Datos Externa	26
C .4. Insertar Registro	29
C .5. Búsquedas	32
D . Validación de la Solución	45
D .1. Ejemplo Taller Mecánico	45
D .2. Ejemplo Productos Congelados	48
D .3. Ejemplo Validación Consulta ONS	54
D .4. Eficiencia en consultas	57
Conclusión	57

Índice de figuras

1.	Interacción del Modelo, la Vista y el Controlador	8
2.	Diseño Base de Datos	17
3.	Ejemplo MVC del Sistema	19
4.	Creación Objeto Trazable	20
5.	Error Creación Objeto	21
6.	Creación Objeto con Atributos Tipo MOT y Búsqueda	22
7.	Listado Objetos Creados	23
8.	Modificar Objeto Creado	24
9.	Crear Usuario	26
10.	Listado Usuarios	27
11.	Listado Base de Datos Externas	28
12.	Listado Usuarios	29
13.	Ingreso Registro	30
14.	Agregar Registro con Atributos de MOT y Búsqueda	31
15.	Búsqueda por Valor	33
16.	Resultado Búsqueda por Valor	34
17.	Búsqueda local - seleccionar MOT	35
18.	Búsqueda local - Condiciones Búsqueda	36
19.	Autoingreso	37
20.	Ejemplo del uso de la opción Historia	38
21.	Listado de Búsquedas Guardadas	40
22.	Búsqueda Guardada - Atributo Variable	41
23.	Consulta ONS	43
24.	Respuesta Consulta ONS	44
25.	Ejemplo taller mecánico - Primera Consulta	46

26.	Ejemplo taller mecánico - Segunda Consulta	47
27.	Ejemplo taller mecánico - Tercera Consulta	48
28.	Ejemplo Cámara de Frío - Primera Consulta	49
29.	Ejemplo Cámara de Frío - Segunda Consulta	50
30.	Ejemplo Cámara de Frío - Tercera Consulta	51
31.	Ejemplo Cámara de Frío - Cuarta Consulta	52
32.	Ejemplo Cámara de Frío - Quinta Consulta	53
33.	Ejemplo Validación Consulta ONS - Ingreso EPC	54
34.	Ejemplo Validación Consulta ONS - Lista de Servicios	55
35.	Ejemplo Validación Consulta ONS - Trazabilidad Producto	56

Introducción

El tema de la trazabilidad para una empresa no debe ser un tema menor. La importancia de ésta como mecanismo para el seguimiento y conocimiento de la historia de un producto ha sido ampliamente reconocida por distintas organizaciones de ámbito internacional.

Una de las razones del por qué la trazabilidad es tan importante es que el seguimiento de la vida de un producto puede aportar información suficiente para saber todos los elementos que han entrado en su producción, pero también todos los caminos que se han seguido hasta su comercialización. Como consecuencia, el poseer la información adecuada puede ayudar a determinar con más precisión la responsabilidad por los defectos en los productos o de problemas de seguridad en ellos. Del mismo modo, si se produjese un accidente se podrían localizar de forma fácil y precisa no sólo los lotes de producción, sino todos aquellos elementos que puedan haber estado en el origen del problema, minimizando así las pérdidas económicas y disminuyendo al máximo posibles tragedias. Un ejemplo claro de esto último es el caso de las vacas locas que generó pérdidas de alrededor de los 7 mil millones de dólares a la industria ganadera. Otro ejemplo, menos dramático, pero igualmente ilustrativo, es el caso de una leche vencida que fue comercializada junto con leches en perfecto estado. Sin un conocimiento claro de la producción de ese producto, la empresa podría incluso retirar toda la producción del mercado, provocándole pérdidas gigantescas, en comparación a que si solo retirara del mercado el lote correspondiente al de la leche vencida, previa consulta de la información del historial del producto.

El darse cuenta de que la trazabilidad ya no es un beneficio si no que una importantísima necesidad es solo el primer paso, ya que surgen ciertos problemas a los cuales es necesario buscarle solución. El primero de ellos es que la cantidad de información rescatada del seguimiento de un producto (desde su producción hasta su comercialización) puede llegar a ser

considerable, luego se hace necesario un sistema capaz de manejar toda esa información de manera eficiente y rápida, ya que no tiene ningún sentido disponer de todos esos datos y no poder hacer nada con ellos. El segundo problema que surge es que cada empresa tiene su propio modelo de negocio, por lo tanto, no necesariamente un sistema de trazabilidad utilizado en el modelo de negocio de una empresa podrá ser usado en el modelo de negocio de otra y obtener los mismos resultados satisfactorios. Por lo tanto, si una empresa desea invertir en trazabilidad y emplear un sistema para el manejo de ella, dicho sistema deberá ser hecho a la medida de la empresa, de acuerdo a su modelo de negocio, lo cual claramente puede causar un gasto de tiempo y de recursos considerable.

Frente a estos problemas, en la empresa donde presto mis servicios, surge la oportunidad de desarrollar un sistema que no solo sea capaz de tomar toda la información de un producto, manejarla de manera eficiente y rápida, y mostrarla de forma clara y transparente, sino que además de ser un sistema que sin importar el modelo de negocio de la empresa, se acople a dicho modelo y a las necesidades de la empresa y obtener así resultados satisfactorios, sin tener que partir desde cero con el desarrollo, ahorrándose así gran cantidad de recursos.

Si bien un sistema como el descrito anteriormente bastaría para que una empresa llevara una trazabilidad adecuada de sus productos, existen tecnologías que harían de este sistema uno más completo, aún más eficiente. Estas tecnologías que se mencionan son las de identificación por radio frecuencia (**RFID**), que se encuentra asociado al código electrónico de productos (**EPC**) y ONS (***Object Naming Service***), un sistema de resolución basado en DNS (***Domain Name Service***) para obtener información acerca de un EPC. La integración de estas tecnologías permitirá establecer una red global única, facilitando así que el producto traspase fronteras con toda la información de sus procesos, sus características, sus ventas, sus compras, etc.

Luego de analizar y evaluar diversas alternativas de desarrollo, se logró dar con una solución al problema de la diversidad en los modelos de negocios: La implementación de un sistema modular, flexible y adaptable, con una meta base de datos, permitiendo que, sin importar el tipo de negocio, no se hará necesaria una reestructuración ni tampoco un rediseño. Esta adaptabilidad del sistema permite además que no haya restricción respecto de la natu-

raleza del objeto a trazar, es decir, que la trazabilidad se puede aplicar tanto a información abstracta (como boletas y facturas), como también al momento o lugar de la prestación de un servicio o bien a objetos concretos como insumos y productos. De ahora en adelante nos referiremos a este sistema como a un **Sistema de Modelamiento de Objetos Trazables** (SMOT).

SMOT, además de presentar las características mencionadas, también posee una que permite llevar a cabo la integración con las tecnologías recién mencionadas. En este caso nos referimos a que posee un módulo destinado a realizar consultas ONS de modo de que pueda obtener la historia del objeto dado su EPC.

Como se detallará más adelante, el sistema se puso a prueba frente a distintos modelos de negocio obteniendo resultados satisfactorios en cuanto a la adaptabilidad del sistema al problema planteado, mostrando eficacia, rapidez en la muestra de resultados, al igual que en la realización de consultas ONS para obtener la información de un producto.

Antecedentes

A continuación se detallan los conceptos presentes en este documento y que formaron parte del desarrollo de la tesis.

- **SMOT: Sistema de Modelamiento de Objetos Trazables.** Se refiere al sistema implementado para esta tesis y que consiste en un software capaz de manejar la trazabilidad de un objeto de naturaleza variable, es decir, el objeto a trazar puede ser algo abstracto (como por ejemplo una orden de compra), o una prestación de servicio, u objetos más concretos tales como productos o insumos, o prácticamente cualquier cosa que se desee trazar. SMOT ha sido desarrollado íntegramente en el lenguaje de programación Perl, además del uso de Javascript y AJAX. La gran virtud de este sistema es su adaptabilidad a cualquier problema de negocio sin necesidad de modificar su estructura. Además su característica modular permite que sea posible agregar diferentes módulos, dependiendo de las necesidades de la empresa.
- **RFID: Radio Frequency Identification (*Identificación por Radio Frecuencia*).** Corresponde a un sistema de almacenamiento y recuperación de datos remoto. La premisa bajo la cual se debe entender lo que es un sistema RFID es que uno va marcando ítems con tags, los cuales son dispositivos de radio muy pequeños y de bajo costo. Los tags RFID guardan algún número identificador único que permite diferenciarlos unos de otros. Son estos tags los que le envían un mensaje a los lectores RFID. El lector devuelve información correspondiente al número de identificación del tag desde una base de datos [6]. Básicamente un RFID se compone de un pequeño chip IC y de una antena que recibe y envía una onda de radio para comunicarse [5].

- **EPC: Electronic Product Code** (*Código Electrónico de Producto*). Corresponde al identificador único que se le asigna a un producto. Este término fue creado por la organización sin fines de lucro EPC-Global [5]. El código de producto electrónico es codificado en un tag RFID, luego, el identificador único mencionado en el punto anterior corresponde al EPC [7].
- **EPCIS: Electronic Product Code Information System** (*Sistema de Información de Código Electrónico de Producto*). El objetivo de EPCIS es permitir que diferentes aplicaciones aprovechen los datos del Código Electrónico de Producto (EPC) a través del intercambio de datos relacionados con EPC, tanto dentro de la empresa como a través de varias de ellas. [8]
- **ONS: Object Name Service** (*Servicio de Nombre de Objetos*). Provee un servicio de búsqueda que retorna la URL de un EPCIS dado un EPC [2]. Es muy similar a DNS, de hecho, utiliza el sistema DNS para realizar la búsqueda. En este caso, ONS convierte el EPC dado en la URI del EPCIS (mientras que DNS convierte la dirección de dominio, es decir la URL, en direcciones IP) [1]. La búsqueda realizada por ONS es para entregar información acerca del EPC dado como también de los distintos servicios asociados a él. ONS es capaz de entregar punteros a servicios de información del dueño del objeto del EPC dado [1]. Los pasos que se deben seguir para realizar una consulta son:
 - Se obtiene un EPC por medio de una lectura a un tag RFID.
 - El lector RFID envía el EPC leído, como una secuencia de bits, a un servidor local.
Ejemplo: 10 000 0000000000000000 000000000000000011000 0000000000000000110010000
 - El servidor local convierte la secuencia de bits en una forma URI.
Ejemplo: urn:epc:id:sgtin:0614141.000024.400
 - El servidor local presenta la URI a un Solucionador de ONS local.
Ejemplo: urn:epc:id:sgtin:0614141.000024.400
 - El Solucionador convierte la URI en un nombre de dominio y ejecuta una consulta DNS usando el registro NAPTR para ese dominio.
Ejemplo: 000024.0614141.sgtin.id.onsepc.com
 - La consulta DNS retorna una serie de respuestas que contienen URLs que apuntan a uno o más servicios.

- El Solucionador local extrae el URL del registro DNS obtenido. Ejemplo: `http://epc-is.example.com/epc-wsd1.xml`
- **DNS: Domain Name Service** (*Servicio de Nombre de Dominio*). La explicación más básica que se puede dar para entender lo que hace el DNS es que transforma el nombre de dominio en su dirección IP asociada. Gracias a DNS es posible hoy en día poder navegar por la Web, mandar correos electrónicos, etc. de manera fácil y cómoda [4]. La manera en que realiza dicho mapeo desde un nombre de dominio a su dirección IP asociada, es a través de consultas en la base de datos, en las cuales se encuentra el nombre del dominio con su respectiva IP (y más información que en este momento no toma relevancia para entender el funcionamiento de DNS). En cuanto a la seguridad y la performance con la que trabaja DNS se han realizado numerosos estudios para evaluar y corregir falencias en la seguridad y performance [3].
 - **Metabase de Datos**: En términos simples, consiste en una base de datos de metadatos, es decir, si por ejemplo se tiene una base de datos de imágenes, se puede tener una base de datos de los formatos de las imágenes, los pesos en kilobytes, vectores característicos, etc., es decir una base de datos de los datos con los que se trabaja. O bien puede ser una base de datos que contiene información sobre una o más base de datos.
 - **MVC: Model View Controller** (*Modelo Vista Controlador*). Corresponde a un patrón de diseño que se basa en la separación de la aplicación en tres capas (Modelo, Vista y Controlador) separando así la lógica de negocio de la interfaz de usuario. La gran ventaja de este patrón es que las piezas de un programa se pueden construir por separado y luego unirlos en tiempo de ejecución. Si uno de los componentes, posteriormente, se observa que funciona mal o bien se desea realizar actualizaciones, puede reemplazarse sin que las otras piezas se vean afectadas. A continuación se explica brevemente cada uno de estos componentes:
 - **Modelo**: es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o de las Vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene encomendada la responsabilidad de mantener enlaces entre el Modelo y sus Vistas, y notificar a las Vistas cuando cambia el Modelo.

- Vista: es el objeto que maneja la presentación visual de los datos representados por el Modelo. Genera una representación visual de éste y muestra los datos al usuario. Interactúa con el Modelo a través de una referencia a éste.
- Controlador: es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el Modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del Modelo o por alteraciones de la Vista. Interactúa con el Modelo a través de una referencia a este último.

Como se aprecia en la Figura 1 y también por lo explicado previamente, la interacción entre las 3 componentes puede ocurrir de diferentes formas (unidireccional y/o bidireccional), dependiendo de como se construya el sistema. Lo que debe quedar claro es que siempre que se construya de esta manera un sistema, se facilitará el manejo de los distintos componentes.

EL uso del patrón MVC presenta tanto ventajas como desventajas. Entre las ventajas se cuenta:

- La separación del Modelo de la Vista, es decir, separar los datos de la representación visual de los mismos
- Dada la independencia que hay entre las capas, se hace mucho más fácil agregar nuevos tipos de datos según sea requerido por la aplicación.
- Si se desea probar que el sistema funciona correctamente, con MVC existen maneras más sencillas para hacerlo.
- Facilita el mantenimiento en caso de errores.
- Permite el escalamiento de la aplicación.

Pero como en todo ámbito de cosas, no solo pueden haber ventajas. Las desventajas que son posibles visualizar son:

- Quizás no es tan sencillo de aprender a utilizar como lo son otros patrones de diseño.

- La separación de conceptos en capas agrega complejidad al sistema.

La figura explica de manera visual el flujo que sigue una consulta por parte de un usuario al sistema, utilizando el patrón MVC.



Figura 1: Interacción del Modelo, la Vista y el Controlador

Especificación del Problema

A continuación se discutirá sobre el problema a solucionar y el porqué es importante el encontrar una solución, además de señalar los requisitos que debe tener la solución para poder ser considerada adecuada, aceptable y eficiente.

A . Descripción del Problema

Podemos distinguir dos problemas a solucionar. A continuación se da una descripción de ellos y la relevancia de contar con una solución para ellos.

A .1. Diversidad en modelos de negocio

Como se mencionó anteriormente, para poder desarrollar un sistema de trazabilidad para una empresa, este debe cumplir con ciertos requerimientos establecidos por la empresa y ser construido de cierta forma, ya que cada empresa tiene su propio modelo de negocio, sus propios problemas de negocio, cada una vende/compra/gestiona diferentes cosas, tienen diferente estructura, etc. Por lo tanto, cada una de ellas tiene una manera distinta de llevar la trazabilidad de sus productos y el sistema utilizado para cumplir dicha tarea no tiene por que ser el mismo que utiliza cualquier otra empresa, justamente por la diferencia planteada sobre sus modelos. Luego, el problema que surge es que si una empresa decide comenzar a llevar la trazabilidad de sus productos, lo más probable es que deba destinar recursos (que es posible que no tenga), como dinero, tiempo y personal, en desarrollar un sistema a medida. O bien puede ocurrir que ya se esté utilizando un sistema para llevar la trazabilidad de los productos pero se desea actualizar dicho sistema ya que han variado ciertas características en sus problemas de negocio, luego, el sistema que utilizan actualmente estas empresas podría quedar obsoleto o necesitar aplicársele una reingeniería mayor, necesitando invertir aún más recursos lo que podría llevar a la empresa a decidirse por desechar esta opción.

Es posible ver que el gran problema a solucionar es como construir un sistema que, sin importar el tipo de negocio que se quiera abordar, sea capaz de adaptarse y llevar la trazabilidad de sus productos, sea cual sea su naturaleza, y que además, si es que ocurre algún cambio en el modelo de negocios, no sea necesario modificar la estructura principal del sistema (lo que podría terminar modificando el sistema completo).

La gran relevancia que existe en tener una solución a este problema, es que el disponer de un sistema de este tipo, que tenga ese nivel de adaptabilidad y que además sea fácil de implementar y que solo baste conocer bien el modelo de negocio, resulta en un gran ahorro en los recursos de la empresa. Ahora no será necesario gastar meses o años de desarrollo para hacer un software a la medida, si no que con esta especie de “software universal”, las empresas podrán llevar la trazabilidad de sus productos (o de lo que desee) de manera eficiente y rápida

A .2. Integración con otras tecnologías

Hoy en día existen numerosas tecnologías que ayudan a mejorar los distintos aspectos de una empresa. Si nos centramos en el área de la trazabilidad, son varias las herramientas que nos permitirían mejorar esa gestión, algunas de bajo costo y otras no tanto.

A continuación se muestra un ejemplo para ilustrar de mejor manera el problema al que se podría ver enfrentado un usuario cualquiera: Una gran empresa de reconocido prestigio produce, entre otras cosas, un producto que llamaremos ***Producto A***. Dicha empresa tiene su propio sistema de manejo de trazabilidad, registrando así cada evento de importancia en la vida de su producto, desde que es creado hasta que es vendido. Esta empresa realiza ventas tanto nacionales como internacionales. El dueño de un minimarket (de otro país) le compra a esta empresa su “Producto A” en cantidades pequeñas (comparadas a las que realiza la empresa a compradores más grandes). Si el dueño de este minimarket quisiera saber sobre algún punto en la historia del producto (el motivo pueden ser muchos, desde conocer cuando se le agregó cierto compuesto al producto hasta conocer en que lote se envió el producto pues una unidad venía defectuosa), tendría que consultarle a la empresa (vía telefónica, via email o personalmente) y seguir quizás algún tipo de protocolo para obtener la información que desea, si es que la empresa decide proveérsela. Todo este trámite podría tardar varios días, lo que probablemente no sea lo ideal para este dueño.

La pregunta que surge es: ¿Cómo poder utilizar la gran diversidad de tecnologías disponibles para hacer que la situación recién descrita tenga una solución más óptima que un llamado telefónico o un correo electrónico?.

Como se puede ver, la relevancia en resolver un problema como este, para algunas personas, puede ser de vital importancia pues se estaría expandiendo el acceso a la trazabilidad, facilitando que el producto traspase fronteras con toda la información de sus procesos y características, además de mejorar el tiempo de respuesta a la consulta del registro de un producto realizada por terceros.

B . Requisitos de la Solución

En vista de los problemas expuestos a lo largo de este documento, a continuación se detallan los requisitos que debe cumplir la solución a construir para que sea considerada como exitosa y utilizable por cualquier empresa que desee implementarla en su negocio.

- Control de acceso al sistema: El software debe contar con un control de acceso de usuarios, de modo que solamente aquellos que se encuentren registrados puedan ingresar al sistema, evitando que cualquier agente ajeno a la empresa navegue por el sistema y tenga acceso a información confidencial.
- Registro de usuarios: El sistema debe permitir que un usuario pueda registrarse de modo de poder acceder a las funcionalidades que ofrece el sistema.
- Creación de grupos y roles: Dado que cada empresa posee su propia estructura y modelo de negocio, cada una posee diferentes grupos encargados de diferentes áreas de la empresa (por ejemplo el área de ventas, el área de comercio exterior, etc), luego es necesario que puedan distinguirse dentro del sistema.
- Creación de objetos trazables: El sistema debe permitir la creación de objetos trazables, a los cuales nos referiremos en este documento como MOT's. Para ello se requiere lo siguiente:

1. Se pueda ingresar el nombre del objeto a trazar

2. Se pueda ingresar uno o más atributos del objeto.
 3. Se pueda eliminar uno o más atributos del objeto.
- Atributos de variados tipos: El sistema debe permitir que los atributos sean de diversos tipos
 - Visualización de objetos trazables disponibles: Se deben poder ver todos los objetos trazables existentes, además de poder ver todos los atributos que lo componen.
 - Agregar atributos a objetos ya existentes: Es muy probable que ocurra que a un objeto ya existente, del cual ya se lleva su trazabilidad, se le quiera agregar un nuevo atributo para ser monitoreado. Luego, el sistema debe permitir esto sin perder la trazabilidad que ya se llevaba antes de agregar el nuevo atributo.
 - Se debe poder agregar valores al objeto trazable: El sistema debe permitir ingresar valores a los atributos de un objeto trazable determinado y así estos nuevos valores equivalgan a un nuevo registro del objeto.
 - Búsqueda de Objetos Trazables: Esta es la principal componente del sistema, ya que es el que permite ver la historia de un objeto a través del tiempo. El sistema debe tener la opción de realizar tanto búsquedas locales como “externas”, es decir, búsquedas en base de datos externas (las de la propia empresa, por ejemplo). A continuación se detallan las características que deben poseer estas búsquedas:
 1. Selección de búsqueda local o externa: El sistema debe permitir elegir que tipo de búsqueda se desea realizar, si es que es una local o una externa.
 2. Nombre a las búsquedas: Debe existir la posibilidad de asignarle un nombre a una búsqueda. La razón de tener esta opción es que si se desea guardar (característica que será detallada en el siguiente punto), es necesario tener un nombre para poder identificarla.
 3. Selección de objeto: Se debe seleccionar el objeto al cual se le desea realizar la búsqueda
 4. Criterio de búsqueda: Se debe especificar sobre qué atributo se está buscando y especificar las condiciones de búsqueda, de modo que exista la opción de buscar todos los registros existentes para ese objeto o bien sólo algunos

- Encapsular búsquedas: El sistema debe poder permitir guardar las búsquedas que un usuario realice (siempre y cuando quieran ser guardadas). Dichas búsquedas, al ser ejecutadas, deben mostrar el resultado sin tener que realizar todos los pasos previos para hacer una búsqueda. La utilidad de esta opción es que los usuarios realizan la misma búsqueda muchas veces, por lo que al tener una opción de guardado, les ahorra tiempo, ya que no deben seleccionar el objeto a buscar ni establecer los criterios de búsqueda sobre los atributos, etc.
- Encadenar búsquedas: Esta opción también es de gran importancia ya que es la que permite viajar a través de distintos objetos y así conocer el hilo de una historia y poder obtener información que puede ser muy valiosa.
- Visualizar búsquedas guardadas: El sistema debe permitir mostrar las búsquedas que se encuentran guardadas de modo que el usuario pueda seleccionar la que desee y obtener los resultados de dicha búsqueda.
- Editar búsquedas guardadas: El sistema debe permitir que las búsquedas que ya se encuentran almacenadas, puedan ser modificadas, ya que puede ocurrir que el usuario quiera modificar alguna característica de la búsqueda sin perderla o sin tener que hacer una nueva.
- Eliminar búsquedas: Debe ser posible eliminar búsquedas guardadas. Estas búsquedas, luego de su eliminación, no deben aparecer en el listado de búsquedas guardadas.
- Realizar consultas ONS: El sistema debe permitir realizar consultas de tipo ONS y mostrar como resultado el listado de servicios disponibles para ese EPC.
- Registrar Base de Datos externas: Si se va a trabajar con alguna base de datos externa, se debe ingresar la información necesaria de dicha base de datos para que sea posible acceder a ella.
- Permisos a los usuarios: Se le debe poder asignar permisos a los usuarios, principalmente permisos de visualización de búsquedas y de información.
- Salida del sistema: Se debe permitir “desloguearse” del sistema. Al realizar dicha acción, el usuario ya no tendrá acceso a ninguna de las funcionalidades del sistema y deberá volver a ingresar con su username y su password.

Además de los requisitos mencionados previamente, también es necesario que nuestra solución cumpla con ciertos requisitos de calidad, tales como:

- **Performance:** Es deseable que los tiempos de respuestas de las búsquedas sean cortos, o al menos no exceder cierto límite de tiempo que haga que el usuario lo encuentre no aceptable.
- **Seguridad:** El sistema debe garantizar que una persona cualquiera no pueda acceder a alguna de las funcionalidades del software sin previamente haberse identificado ingresando su username y su password. Ante cualquier intento de ingresar sin previo registro u identificación, se debe redirigir al usuario a la página de ingreso para que se identifique.
- **Eficiencia:** El sistema debe ser capaz de responder con la rapidez apropiada a las distintas exigencias del usuario, para las distintas partes del sistema.
- **Modificabilidad:** El sistema debe estar construido de tal forma que el hacer algún cambio no signifique un costo mayor, pueda hacerse de manera rápida y sin afectar mayormente la estructura del sistema. Debe estar construido de tal manera que esté preparado para extender, cambiar o remover capacidades del sistema
- **Portabilidad:** El sistema debe ser capaz de ejecutarse en cualquier servidor que posea postgresSQL.
- **Testabilidad:** El sistema debe ser capaz de mostrar sus defectos de manera fácil y clara.

Finalmente, para poder instalar nuestro sistema, es necesario contar con los siguientes requisitos:

- Tener instalado un interprete de Perl.
- Tener instalado un servidor web.
- Tener instalado CPAN para así instalar los módulos necesarios para ejecutar el sistema correctamente.

Dados todos estos requisitos del sistema, podemos decir que los criterios de aceptación que debe cumplir la solución para ser considerada como una solución al problema como tal,

son que el sistema sólo permita acceso a aquellas personas registradas, que permita obtener la trazabilidad de un objeto de cualquier naturaleza, entregando toda la información previamente ingresada, que permita realizar búsquedas encadenadas entre diferentes objetos trazables, que permita ingresar valores a los atributos de un objeto. Finalmente que permita realizar consultas ONS entregando la lista de servicios disponibles para el número EPC entregado y que al seleccionar el servicio correspondiente a la trazabilidad del objeto, el sistema muestre la historia completa de dicho objeto.

Cumpliendo con estos criterios es posible decir que el sistema es una solución aceptable para el problema planteado. En vista de que los criterios mencionados son los básicos que debe cumplir la solución, se espera también cumplir con el resto de los requisitos de calidad y de usuario.

Es deseable que el software cuente con una buena usabilidad de modo que un usuario que desconoce totalmente el sistema sólo necesite de una pequeña capacitación para lograr un uso adecuado de todas las funcionalidades. Además de que facilite las tareas de los usuarios, no que las complique más.

Descripción de la Solución

En este capítulo se presentará la mejor solución encontrada para el problema planteado y se detallará en su totalidad, pasando por la arquitectura del software, en donde se mostrarán los distintos módulos de los que está compuesto el sistema, mostrando el diseño de la base de datos y describiendo las distintas funcionalidades del sistema.

El sistema a construir corresponde a un Sistema de Modelamiento de Objetos Trazables. De ahora en adelante nos referiremos al sistema como **SMOT**. Posee la gran característica de que es un sistema adaptable, flexible y modular que permite trazar cualquier objeto, de cualquier naturaleza, además de permitir consultas ONS para seguir la trazabilidad de productos no sólo que estén físicamente presentes si no que además de cualquier producto que se encuentre en cualquier parte del mundo.

SMOT fue desarrollado íntegramente en el lenguaje de programación Perl, junto con Javascript y AJAX para ayudar a tener una mejor usabilidad y la base de datos utilizada fue PostgreSQL.

A continuación se pasará a detallar el sistema.

A . Diseño Base de Datos

Para lograr que SMOT fuera un sistema flexible y adaptable era necesario primero lograr tener una base de datos que permitiera dichas características. En la figura a continuación se muestra el diseño de la base de datos.

Como se puede apreciar, es una base de datos bastante simple, relativamente pequeña (en comparación a algunas bases de datos con decenas de entidades). Es posible distinguir que

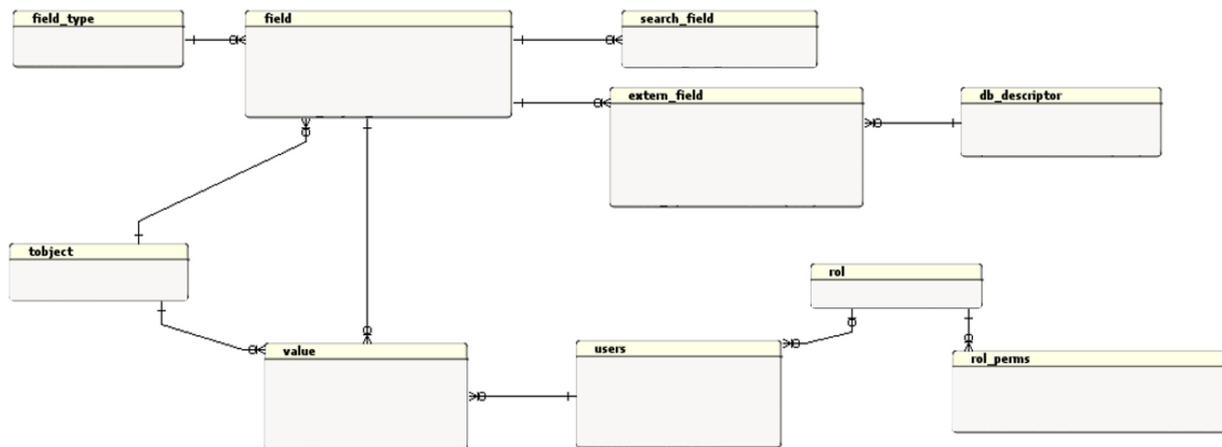


Figura 2: Diseño Base de Datos

existen tablas destinadas a almacenar información de los distintos roles que puedan existir, como también de los usuarios, o los valores de los registros, o los objetos trazables en sí (dicha información relevante se distribuye entre la tabla tobject y field). También existen unas tablas que no están relacionadas entre sí, ni tampoco con las expuestas en la Figura 2, y son las que almacenan la información de las búsquedas guardadas, tanto locales como externas. No se han mostrado en este documento pues son tablas independientes y sin efecto sobre ninguna entidad mostrada en la imagen. También, dentro del diseño de la base de datos, se creó una tabla para guardar información sobre otras base de datos, de modo de que el sistema, en caso de que sea necesario, utilizara esa información para realizar consultas a dicha base de datos externa y así mostrar los resultados o usarlos para otros propósitos.

B . Arquitectura del Software

El software está construido de manera modular para facilitar aún más el desarrollo e implementación del sistema. Los módulos están contruidos de tal forma que cualquier modificación, ya sea actualización o eliminación, a una parte de ellos, no afectará mayormente al resto de los módulos y así no tener que gastar tiempo y personal en arreglar posibles errores que ha simple vista pueden parecer menores, pero que podrían causar gran impacto enel sistema.

La arquitectura del software sigue el patrón de diseño MVC, por lo que todo el modelo de

negocio se separó de la interfaz de usuario. De este modo, cualquier modificación que requiera SMOT podrá ser tratada en su respectivo módulo, sin afectar mayormente al resto.

Como se mencionó anteriormente, SMOT está implementado íntegramente en Perl, utilizando JavaScript y Ajax para tener una mejor usabilidad. Para lograr implementar el sistema utilizando el patrón MVC, se utilizó el framework CGI::Application. La razón por la cual se decidió por utilizar este framework es porque la intención de CGI::Application es facilitar la acción de crear aplicaciones web extensibles, portables y sofisticadas. Este módulo implementa una metodología que permite que los software Web sean fáciles de diseñar, de documentar, de escribir y evolucionar. Además, CGI::Application evita emplear tecnologías y técnicas que puedan atar a un desarrollador a cualquier conjunto de herramientas, sistema operativo o servidor Web.

La Figura 3 ilustra dos ejemplos típicos de lo que ocurre en el sistema y cómo este es manejado por medio del patrón de diseño MVC.

Como se puede apreciar, para nuestro primer ejemplo el usuario comienza el flujo definiendo un objeto trazable (en la figura MOT se refiere a **M**odelo de **O**bjeto **T**razable) a través de una interfaz Web simple. Dicha petición es validada por el Controlador, el cual recurre a la capa Modelo para almacenar el nuevo objeto en la base de datos local. Este ejemplo corresponde cuando un usuario crea un nuevo objeto trazable

Para el segundo ejemplo el usuario realiza una búsqueda a través de la interfaz Web, solicitud que la valida la capa **Controlador** quien le pasa la solicitud a la capa **Modelo** la cual realiza la búsqueda y le entrega al **Controlador** los resultados, el cual, luego de juntar y ordenar la información, se la entrega a la capa de **Vista** que es la que finalmente muestra la información detallada al usuario.

C . Funcionalidades SMOT

En la presente sección, se mostrarán las distintas funcionalidades existentes en SMOT y se explicará el funcionamiento de cada una, mostrando además imágenes del sistema. Las funcionalidades presentes en SMOT son:

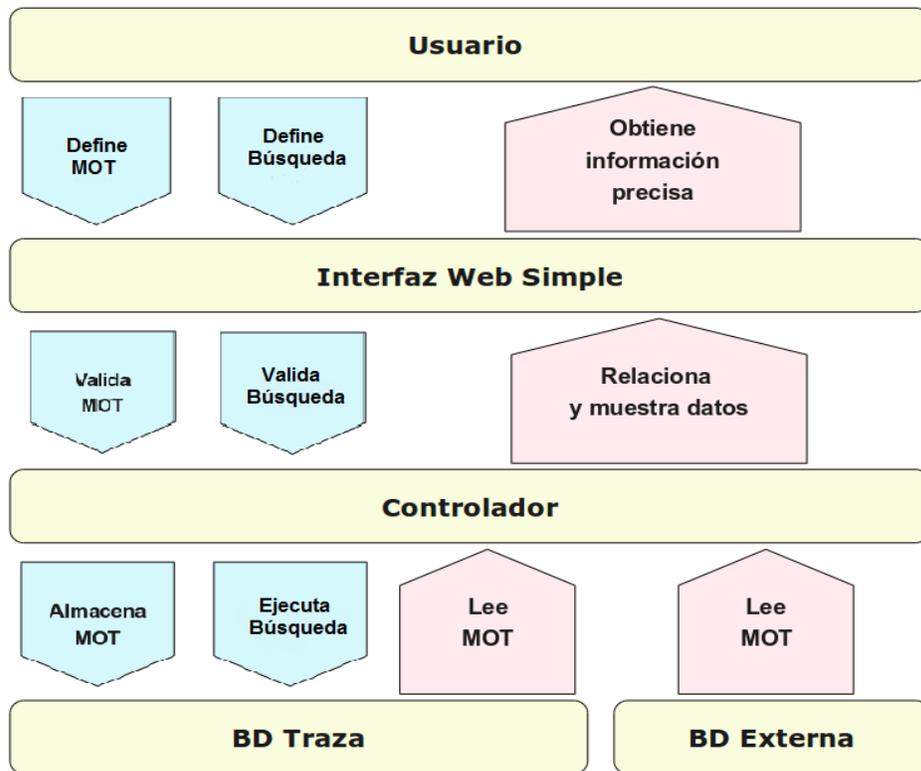


Figura 3: Ejemplo MVC del Sistema

- Creación de Objeto Trazable.
- Creación de Usuario.
- Agregar Base de Datos Externa
- Insertar Registro.
- Búsqueda Local
- Búsqueda Externa
- Guardar Búsqueda
- Consulta ONS

El sistema también tiene el módulo de ingreso de usuario y salida de usuario, pero no se describirán en el presente documento ya que no presenta diferencia alguna con cualquier

sistema que posea dichas funcionalidades y por lo tanto solo serán mencionadas.

C .1. Crear Objeto Trazable

Esta es una de las principales funcionalidades del sistema ya que es la que permite crear un objeto trazable de cualquier naturaleza. En la presente figura se muestra la interfaz a la que se encuentra enfrentado el usuario al tratar de crear un objeto.

SMOT (USER)
HOME :: Objetos :: Crear nuevo objeto

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Ver listado objetos
- Ingresar nuevo objeto

Crear un nuevo objeto

Nombre MOT

Tipo

Atributo 1: Nombre: [Borrar Atributo](#)

Atributo 2: Nombre: [Borrar Atributo](#)

«Agregar atributo»

SMOT - Sistema de Modelos de Objetos Trazables

Figura 4: Creación Objeto Trazable

Como se puede apreciar en la Figura 4, para poder crear un objeto trazable es necesario primero asignarle un nombre al objeto. De no hacerlo, si es que el usuario intenta crear el objeto, el sistema no se lo permitirá, entregándole un mensaje de error tal como se muestra en la Figura 5.

Además del nombre del objeto a crear, al usuario se le entrega la opción de agregar o bien eliminar uno o más atributos. Esta es una característica bastante importante dentro de

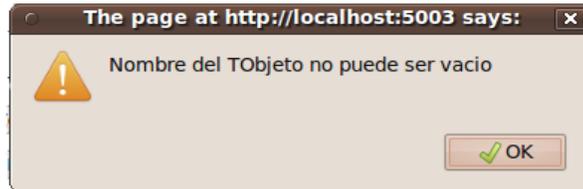


Figura 5: Error Creación Objeto

SMOT, ya que no obliga al usuario a entregar un número fijo de atributos para el objeto, no lo “amarrá” a tener que definir a priori una cantidad de atributos. Al momento de definir un atributo, el usuario debe seleccionar su tipo. Existen 5 tipos que se han definido en el sistema:

- Texto
- Fecha
- Número
- MOT
- Búsqueda

Los tres primeros tipos no necesitan mayor explicación, el usuario puede definir un atributo de tipo texto, o fecha, o número. Los últimos dos tipos son un poco más especiales. Un atributo de tipo *MOT* quiere decir que el objeto tendrá como atributo a otro objeto trazable. Esta característica permite hacer un seguimiento a más de un objeto desde un mismo objeto. Ilustraremos mejor esto último con un ejemplo:

En SMOT se encuentran creados los objetos trazables “*Neumático*”, “*Motor*” y “*Luces*”, cada uno con sus respectivos atributos. Posteriormente se desea llevar la trazabilidad de una ambulancia, en particular de las revisiones técnicas para sus distintas partes. Para ello se debe crear el objeto trazable “*Ambulancia*”, el cual posee atributos como neumáticos, luces, chequeo de aceite, motor, entre otros. Si bien el usuario podría crear los atributos como tipo texto o número, ya existen objetos trazables en el sistema para los neumáticos, las luces y el motor. De esta forma cuando se realice alguna consulta sobre el objeto *Ambulancia*, si

desea averiguar cuando fue la última revisión que se le hizo a sus neumáticos, basta con seguir la cadena a través del objeto *Neumático* que fue asociado a esa ambulancia.

Esta característica facilita en gran medida el poder seguir una cadena de acontecimientos, viajando a través de los distintos objetos.

Para el caso del tipo *Búsqueda* ocurre algo similar con el tipo *MOT*, solo que esta vez el atributo está asociado a una búsqueda. Esto quiere decir que al momento de agregar un registro (ver sección *Insertar Registro*), el valor para ese atributo será el resultado de una búsqueda elegida por el usuario.

Un ejemplo de creación de un objeto con atributos tanto de tipo MOT como de tipo Búsqueda se muestra a continuación:

The screenshot shows the SMOT (USER) interface for creating a new object. The page title is "SMOT (USER)" and the breadcrumb is "HOME :: Objetos :: Crear nuevo objeto". The navigation menu includes: < Home >, < MOTs >, < BD externas >, < Registros >, < Usuarios >, < Roles >, < Búsqueda >, < ONS Queries >, < Logout >.

The main content area is titled "Crear un nuevo objeto". It contains the following form elements:

- Nombre MOT:** A text input field containing "MOT DE PRUEBA DOS". A red circle highlights this field with the label "Objeto Existente".
- Tipo:** A dropdown menu with "Búsqueda" selected. A red circle highlights this dropdown with the label "Atributo del objeto".
- Atributo 1:** A dropdown menu with "Registro Impresion" selected.
- Atributo 2:** A dropdown menu with "MOT" selected, followed by a dropdown menu with "Registro_Codigo" selected, and a "Borrar Atributo" link.
- Atributo 3:** A dropdown menu with "Texto" selected, followed by a "Nombre:" label and a text input field containing "ATRIBUTO TRES", and a "Borrar Atributo" link.
- «Agregar atributo»:** A button to add more attributes.
- enviar:** A button to submit the form.

The footer of the page reads "SMOT - Sistema de Modelos de Objetos Trazables".

Figura 6: Creación Objeto con Atributos Tipo MOT y Búsqueda

Una vez que el nombre del objeto y sus atributos estén definidos, el usuario puede pre-

sionar “Enviar” y generar así el objeto, el cual se agregará a la lista de objetos trazables disponibles, tal como se muestra en la Figura 7, junto con un pequeño mensaje haciéndole saber al usuario que el objeto fue creado con éxito. A partir de este momento cada registro hecho para el objeto recién creado será almacenado en la base de datos y podrá ser visto al momento de realizar algún tipo de consulta sobre el objeto.

The screenshot shows the SMOT (USER) web application interface. At the top, there is a navigation menu with links: < Home >, < MOTs >, < BD externas >, < Registros >, < Usuarios >, < Roles >, < Búsqueda >, < ONS Queries >, and < Logout >. A green success message box in the top right corner reads: "OK El objeto fue insertado correctamente". Below the navigation menu, there is a sidebar with two menu items: "Ver listado objetos" and "Ingresar nuevo objeto". The main content area is titled "Listado de MOTs ingresados" and includes a sub-link "Ver objetos eliminados". Below this, there are pagination controls: "<< first < prev 1 2 3 4 next > last >>". A table with two columns, "Nombre" and "Opciones", lists the following items: "Registro Salida", "Registro_Codigo", "Cantidad_Ordenes", and "MOT DE PRUEBA". Each item has "[Editar] [Eliminar]" links. Below the table, there are more pagination controls: "<< first < prev 1 2 3 4 next last >>". At the bottom right of the page, there is a footer: "SMOT - Sistema de Modelos de Objetos Trazables".

Figura 7: Listado Objetos Creados

Los objetos pueden ser eliminados, pero solo si existe una autorización de quien corresponda. El que un objeto sea eliminado no significa que el objeto ni sus atributos hayan sido eliminados de la base de datos, pues estos se mantienen, ningún registro se borra. Lo que si ocurre es que el objeto ya no será visible para el usuario, tanto en el listado de objetos como también en las búsquedas.

Al momento de pensar en llevar la trazabilidad de un producto, es inevitable pensar qué pasaría si es que el producto cambia en el tiempo. Un ejemplo para clarificar esta idea es:

¿Qué sucede si en un comienzo se crea un objeto para llevar la trazabilidad de un producto A que posee los atributos A1, A2 y A3, y luego de 1 año la empresa decide que ahora le agregará un nuevo componente a la mezcla (el cual se lo comprará a otra empresa) y que ya no será necesario seguir la trazabilidad del atributo A2? ¿El objeto creado para seguir su trazabilidad queda obsoleto, debiéndose crear uno nuevo y con sus correspondientes atributos?.

SMOT permite realizar modificaciones en un objeto sin afectar los registros ya existentes para dicho objeto. Estas modificaciones van desde cambiarle el nombre al objeto, cambiarle los nombres de los atributos, sus tipos, hasta agregar o eliminar atributos. Al momento de obtener los registros, el sistema muestra tanto los registros antes del cambio como después de este. La trazabilidad se mantiene siempre transparente para el usuario. En la Figura 8 se muestra la interfaz que el usuario verá en caso de querer modificar algún aspecto del objeto.

SMOT (USER)
HOME :: Objetos :: Detalles MOT

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Ver listado objetos
- Ingresar nuevo objeto

MOT 35: MOT DE PRUEBA

Ver atributos eliminados

<< first < prev 1 next > last >>

ID	Etiqueta	Nombre	Tipo Atributo	Opciones
136	atributo_uno	ATRIBUTO UNO	Texto	[Editar] [Transformar] [Eliminar] [Asociar a BD externa]
137	atributo_dos	ATRIBUTO DOS	Número	[Editar] [Transformar] [Eliminar] [Asociar a BD externa]

<< first < prev 1 next > last >>

Agregar campo

enviar

SMOT - Sistema de Modelos de Objetos Trazables

Figura 8: Modificar Objeto Creado

SMOT no permitirá crear dos objetos con el mismo nombre, pero si dos objetos distintos que tengan los mismos atributos.

C .2. Creación de Usuario

En SMOT no existe la opción para que cualquier usuario que desee registrarse pueda hacerlo. Esto se debe a que en un negocio hay jerarquias, cada persona tiene cierta responsabilidad dentro de la empresa, y dichas responsabilidades deben mantenerse al momento de usar el sistema, pues no cualquiera puede registrarse y empezar a crear objetos para trazar a su antojo. Es por esto que existe el usuario *Administrador*, el cual tiene la facultad de registrar solamente a las personas autorizadas. De esta forma existe un control dentro del sistema y cada usuario se hace responsable de sus acciones dentro de SMOT.

Al momento de registrar a un usuario, la interfaz que aparece es la que se muestra en la Figura 9. En ella se puede apreciar que para registrar a un usuario se necesita llenar los campos que usualmente exige una funcionalidad como esta. La diferencia que acá existe está en el último campo a llenar, que corresponde al rol que tendrá el usuario. Como se mencionó, en una empresa existen diversas áreas (ventas, R.R.H.H., Finanzas, etc), por lo tanto cada una de las personas que trabajan en dichas áreas poseen distintas responsabilidades unas de otras, enfocan su trabajo en distintas cosas. La razón por la cual se exige este campo al momento de crear un usuario es porque así cada usuario verá en el sistema lo que solo compete a su área. Por ejemplo un usuario de ventas, cuando vea el listado de búsquedas guardadas, solamente verá el listado de búsquedas guardadas por los de su mismo grupo o por él mismo. De esta forma se evita mostrar información innecesaria, se evita que el usuario vea consultas que no le competen a su área y que además no tendría por qué hacerlas.

También se puede ver el listado de usuarios que se encuentran registrados en SMOT. Tal como se ve en la Figura 10, dentro del listado de usuarios existe la opción de editar o eliminar un usuario. Si se elige *Editar* entonces se mostrará la misma interfaz que para el ingreso de un nuevo usuario, solo que esta vez los campos estarán llenados con la información del usuario, pero se podrá editar cualquier campo. Si se elige la opción de *Eliminar* un usuario, al igual que lo descrito en la eliminación de un objeto trazable, el sistema no borra de la base de datos al usuario, si no que lo coloca en un estado de tal forma que no aparece en el listado de usuarios registrados ni tampoco aparece en los resultados de búsquedas.

Es necesario señalar que la funcionalidad de agregar nuevos usuarios, como la de editar su

SMOT (USER)
HOME :: Usuarios del Sistema :: Crear nueva usuario

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Ver usuarios
- Agregar nuevo usuario

Crear un nuevo Usuario

USERNAME

PASSWORD

NOMBRE

APELLIDO

ROL

SMOT - Sistema de Modelos de Objetos Trazables

Figura 9: Crear Usuario

información o eliminar usuarios, solo se encuentra disponible para el usuario *Administrador*, ya que no es recomendable que cualquier persona tenga acceso a dicha funcionalidad pues el sistema se desordenaría mucho. De esta forma se establece un responsable en el sistema y se establece además un orden que siempre es necesario.

C .3. Agregar Base de Datos Externa

Una de las características de SMOT es que no sólo puede interactuar con una base de datos propia, local, si no que también puede interactuar con base de datos externas. De esta forma, si una empresa ya lleva sus registros en una base de datos propia, con SMOT no es necesario migrar todos los datos, si no que el sistema permite integrar dicha base de datos de tal forma que al momento de realizar consultas, el sistema da la opción de elegir sobre que base de datos se quiere obtener información. Tal como se mencionó anteriormente, SMOT trabaja sobre postgresSQL, por lo que si se desea agregar una nueva base de datos externa y esta también es postgresSQL, no se requiere ningún tipo de configuración, pero si se desea agregar,

SMOT (USER)
HOME :: Usuarios del Sistema :: Ver listado

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Ver usuarios
- Agregar nuevo usuario

Usuarios del Sistema

<< first < prev 1 next > last >>

ID	Username	Nombre	Apellido	Rol	Opciones
0	smot_user	USER	SMOT	2	[Editar] [Eliminar]
2	vendedor	Rodrigo	Arenas	3	[Editar] [Eliminar]
3	bodeguero	Bodega	Bodega	4	[Editar] [Eliminar]

<< first < prev 1 next > last >>

SMOT - Sistema de Modelos de Objetos Trazables

Figura 10: Listado Usuarios

por ejemplo, una base de datos externa Oracle entonces en el servidor se deben realizar las configuraciones e instalaciones pertinentes para que SMOT soporte ese tipo de base de datos.

Es necesario señalar que no es necesario que la base de datos externa posea la misma estructura y definiciones que la base de datos local usada por SMOT, ya que, como se explicará mas adelante, el sistema lo que hace es obtener las tablas que componen dicha base de datos y de ellas saca la información que sea necesaria. Pero en ningún momento se crea una nueva base de datos ni tampoco se modifica ninguna, SMOT solo lee la informacion de la base de datos externa, pero en ningún caso la altera.

En la Figura 11 se observa el listado de base de datos externas que han sido agregadas al sistema al momento de realizar este documento. Como se puede apreciar, la información que se muestra es bastante importante y confidencial, por lo que, al igual que el caso de agregar nuevos usuarios al sistema, el poder ver la información sobre las base de datos con las que SMOT actualmente trabaja, es una atribución que solo personas autorizadas pueden tener.

SMOT (USER)
HOME :: Bases de Datos Externas :: Ver listado

OK
La Conexion fue insertada correctamente

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Ver conexiones de BD
- Crear una nueva conexión

Listado de Conexiones de BD

<< first < prev 1 next > last >>

ID	DSN	Username	Password	Opciones
6	dbi:Pg;dbname=base_prueba;host=200.120.119.43	bdprueba	****	[Editar] [Eliminar]

<< first < prev 1 next > last >>

SMOT - Sistema de Modelos de Objetos Trazables

Figura 11: Listado Base de Datos Externas

En nuestro caso, solo el usuario *Administrador* posee los permisos necesarios. Si se desea, es posible listar más información respecto de la base de datos, como quien la agregó, cuando se agregó, etc., pero por el momento solo nos es necesario saber esos tres campos.

Además el sistema permite editar la información entregada para dicha base de datos (tal como el DSN, el username o el password) o bien eliminar una conexión. El caso de *Eliminar* es el mismo que en los casos expuestos anteriormente, en el que no se elimina de la base de datos, si no que se cambia su estado de tal forma que no es mostrado en las distintas interfaces.

La Figura 12 muestra la interfaz para el caso que se desee agregar una nueva base de datos externa. Los campos a agregar son el DSN, el username y el password. Esta interfaz es la misma para el caso de edición.

En este caso, al igual que ver la información, es una funcionalidad a la que solo tiene acceso, en este caso, el usuario *Administrador*.

SMOT (USER)
HOME :: Usuarios del Sistema :: Ver listado

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Ver usuarios
- Agregar nuevo usuario

Usuarios del Sistema

<< first < prev 1 next > last >>

ID	Username	Nombre	Apellido	Rol	Opciones
0	smot_user	USER	SMOT	2	[Editar] [Eliminar]
2	vendedor	Rodrigo	Arenas	3	[Editar] [Eliminar]
3	bodeguero	Bodega	Bodega	4	[Editar] [Eliminar]

<< first < prev 1 next > last >>

SMOT - Sistema de Modelos de Objetos Trazables

Figura 12: Listado Usuarios

C .4. Insertar Registro

Para poder llevar la trazabilidad de un objeto es necesario que existan valores, asociados a ese objeto, a los cuales se les pueda hacer un seguimiento, de lo contrario, todo el sentido de trazabilidad se pierde. Estos valores a los que se hace referencia son los denominados *Registro*. Un registro corresponde a un “foto” del objeto en un determinado momento. Con esta “foto” se registran todas las características relevantes que posea el objeto en ese instante de tiempo. De esta forma, si se juntan todas estas “fotos” en el tiempo es posible generar una “película” del objeto y poder ver sus características a lo largo del tiempo y obtener así la trazabilidad del objeto.

SMOT posee entre sus funcionalidades la de ingresar un registro para el objeto seleccionado. Este registro se almacena en la base de datos y queda disponible para cualquier búsqueda que se realice para dicho objeto. En la Figura 13 se puede ver los campos necesarios para ingresar un registro al sistema. Primero existe un campo *Código*, el cual corresponde al código que tendrá dicho objeto dentro del sistema. La razón de ser de este campo es que para un

mismo objeto pueden existir muchos registros y lo más adecuado es que se puedan diferenciar ciertos tipos de registros con otros. Por ejemplo, quizás para un usuario del área de Ventas, le sea útil poder diferenciar todas las ventas por mes, luego en dicho campo podría colocar, por ejemplo, “VENTAS SEPTIEMBRE”. Posteriormente, cuando quiera realizar alguna búsqueda podrá usar como filtro el código del registro, haciéndole más fácil leer la información. Los demás campos que se observan en la figura corresponden a los atributos que se definieron al momento de crear el objeto, en este caso, el atributo “ATRIBUTO UNO” y el atributo “ATRIBUTO DOS”.

The screenshot shows the SMOT (USER) web application interface. At the top, there is a navigation bar with the title "SMOT (USER)" and a breadcrumb trail: "HOME :: Registros :: Detalles objeto :: Ingresar Registro". Below the navigation bar, there is a sidebar with a menu item "Nuevo Registro". The main content area is titled "Objeto MOT DE PRUEBA" and contains a form with the following elements:

- A "Código" field with a text input box.
- A table with two columns: "Field" and "Valor".
- Two rows in the table: "ATRIBUTO UNO" and "ATRIBUTO DOS", each with a corresponding text input box.
- An "enviar" button at the bottom of the form.

The footer of the page contains the text "SMOT - Sistema de Modelos de Objetos Trazables".

Figura 13: Ingreso Registro

Tal como se mencionó en la creación de un objeto trazable, es posible que un objeto posea diversos tipos de atributos. Los más comunes son los de tipo texto, número y fecha, pero también existen dos tipos de atributos que resultan muy útiles a la hora de aplicar trazabilidad. Estos son los atributos de tipo MOT y de tipo búsqueda. Para el caso de agregar un registro cuyo atributo tiene asociado un MOT, en el campo correspondiente a dicho atributo, el usuario deberá elegir el valor de una lista de códigos (correspondientes a

registros del MOT asociado). Para el caso de que el atributo tenga asociado una búsqueda, entonces el usuario deberá seleccionar el valor del atributo seleccionando primero el código del objeto del cual desea realizar la consulta. Luego debe presionar “Buscar Valor”. Al hacerlo, el sistema realizará la consulta de manera interna y colocará el resultado de la búsqueda en el campo que se encuentra al lado del nombre del atributo. La Figura 14 muestra un ejemplo de lo explicado.

The screenshot shows the SMOT (USER) interface. At the top, there is a navigation bar with the title "SMOT (USER)" and a breadcrumb trail: "HOME :: Registros :: Detalles objeto :: Ingresar Registro". Below this is a menu with links: "< Home >", "< MOTs >", "< BD externas >", "< Registros >", "< Usuarios >", "< Roles >", "< Búsqueda >", "< ONS Queries >", and "< Logout >".

The main content area is titled "Objeto MOT DE PRUEBA DOS". It contains a form with the following fields:

- A "Código" text input field.
- A table with two columns: "Field" and "Valor".

Field	Valor
nombre_producto	Producto de Prueba
Registro_Codigo	REGISTRO_CODIGO_016(
ATRIBUTO TRES	
- An "enviar" button.

The value "Producto de Prueba" in the "nombre_producto" field is circled in red. To the right of the table, there is a dropdown menu with "REGISTRO_IMPRESION" selected and a "Buscar Valor" button.

At the bottom right of the page, there is a footer: "SMOT - Sistema de Modelos de Objetos Trazables".

Figura 14: Agregar Registro con MOT y Búsqueda como atributos asociados

El ingreso de los valores no es obligatorio, solo lo es el ingreso de un código. En caso de que no se ingrese ningún valor, simplemente al momento de mostrar los resultados de una búsqueda, dichos atributos tendrán valores nulos.

C .5. Búsquedas

Las búsquedas son la funcionalidad más importante del sistema, ya que ahí se basa toda la trazabilidad. Sin las búsquedas no sería posible obtener ningún tipo de información para que el usuario la utilice para tomar todas las decisiones que se necesiten tomar. Es por esto que el principal foco de desarrollo fue en cómo construir un motor de búsqueda que fuera capaz no solo de poder entregar la historia de un objeto en particular de forma directa, por medio de una búsqueda simple, si no que también poder entregar información valiosa por medio de varias búsquedas “encadenadas”, viajando a través de distintos objetos y así tener una historia completa y fácil de leer, entregándole al usuario todo lo que necesite.

Esta sección se subdividirá en subsecciones, en donde en cada una de ellas se explicará un tipo de búsqueda en particular. La razón por la que se desarrollaron diversos tipos de búsquedas es para entregarle al usuario una gama de variedades de modo que el usuario tenga facilidades al momento de querer encontrar alguna información relevante. Cada una de las búsquedas serán explicadas y se darán ejemplos para que el lector tenga un mejor entendimiento del sistema.

C .5.1. Búsqueda por Valor

Este tipo de búsqueda se hizo pensando en que el usuario puede que no tenga mayor información del objeto que el valor de uno de sus atributos. Con esto, el usuario solo debe ingresar el valor que desea buscar y sobre que objeto desea buscarlo. El sistema permite además que el usuario entregue una fecha como condicionante de la búsqueda, de modo que se muestren los resultados cuya fecha de registro esta dentro de lo exigido por el usuario.

La Figura 15 muestra un ejemplo de búsqueda por valor, buscando sobre el objeto de prueba que hemos utilizado como objeto de prueba en este documento. Como se puede apreciar, también existe la opción de realizar una búsqueda exacta o por su estado actual. La búsqueda exacta, tal como lo dice su nombre, realiza una búsqueda en el sistema por todos los valores del objeto seleccionado que coincidan de forma exacta con el valor ingresado por el usuario. De no tener seleccionada esa opción, entonces el sistema buscará todos los valores, para el objeto seleccionado, que **contengan** el valor puesto por el usuario, es decir, si es que el usuario ingresa “café”, el sistema buscará todos los valores que contengan la palabra

The screenshot shows the SMOT (USER) web interface. At the top, there is a navigation bar with the title 'SMOT (USER)' and the path 'HOME :: Búsqueda :: Abierta'. Below this is a breadcrumb trail: '< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >'. On the left side, there is a vertical menu with the following items: 'Por Valor en Registro', 'Por Código de MOT', 'Encapsulada SMOTs', 'Encapsulada Externa', 'Búsquedas Internas Guardadas', and 'Búsquedas Externas Guardadas'. The main content area is titled 'Búsqueda por valor en registro' and contains the following elements:

- A prompt: 'Ingrese el valor, seleccione el MOT y el rango de fecha'.
- A search form with three input fields: a text box containing 'valor uno', a dropdown menu showing 'MOT DE PRUEBA', and another dropdown menu showing 'Ultimo Año'.
- Two radio buttons: 'Búsqueda exacta' (which is selected) and 'Sólo estado actual'.
- A 'Buscar' button.

 At the bottom right of the page, there is a footer: 'SMOT - Sistema de Modelos de Objetos Trazables'.

Figura 15: Búsqueda por Valor

“café”. Si la opción “Solo estado actual” se encuentra seleccionada, SMOT buscará todos los valores que tengan como fecha de ingreso la más actual.

La Figura 16 muestra el resultado que entrega la búsqueda por valor. Como se puede observar SMOT entrega todos los objetos en los que alguno de sus atributos posee el valor a buscar, que en este caso fue “valor uno”.

C .5.2. Búsqueda por Código

Este tipo de búsquedas es muy similar a la de búsqueda por valor, salvo que en vez de ingresar el valor de algún atributo para un objeto en particular, se ingresa el código del registro para ese objeto en particular. Tal como se describió en la sección de *Insertar Registro*, el código corresponde a un identificador para cada registro. La interfaz, tanto de búsqueda como la de resultados, es la misma que para las búsquedas por valor, al igual que la forma de buscar por parte de SMOT, por lo que se omitirá mostrar imágenes de este tipo de búsquedas.

SMOT (USER)
HOME :: Búsqueda :: Por Valor de Registro :: Resultados

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Resultado búsqueda de característica : valor uno

MOT DE PRUEBA

10 << first < prev **1 - 1** next > last >>

Fecha Ingreso	Código	Atributo	Valor	Registrado por
2010-09-25 20:02:04	Codigo Prueba	ATRIBUTO UNO ATRIBUTO DOS	valor uno valor dos	USER SMOT

10 << first < prev **1 - 1** next > last >>

SMOT - Sistema de Modelos de Objetos Trazables

Figura 16: Resultado Búsqueda por Valor

C .5.3. Búsqueda Local

Este tipo de búsqueda (junto con la búsqueda externa) es de las más importante dentro de la variedad de formas para obtener la historia de los objetos. Lo que la hace tan importante es que presenta una forma de realizar la consulta que le permite al usuario “customizar” las búsquedas, construirlas a su manera, de acuerdo a sus necesidades. Esta característica resulta muy útil pues así un usuario perteneciente a una área de la empresa puede realizar sus búsquedas de acuerdo a lo que él necesite, mientras que otra persona, perteneciente a otra área diferente, puede crear sus búsquedas de acuerdo a sus propias necesidades. El sistema no obliga al usuario a realizar búsquedas de un solo tipo, si no que le da la opción de crear las búsquedas a su elección.

En las siguientes figuras se muestra el funcionamiento de este tipo de búsquedas y a lo que se ve enfrentado el usuario al entrar a esta opción.

Lo primero que debe hacer el usuario al usar esta opción, es seleccionar sobre que objeto

SMOT (USER)
HOME :: Objetos :: Búsqueda :: Encapsulada

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Búsqueda Encapsulada SMOTs

Nombre de Búsqueda:

Seleccione el MOT:

Permitir Visibilidad Grupal

Sólo estado actual

SMOT - Sistema de Modelos de Objetos Trazables

Figura 17: Búsqueda local - seleccionar MOT

realizará la búsqueda. En la Figura 17 se muestra como el sistema le da la opción además de colocarle nombre a la búsqueda y además si es que desea que la búsqueda sea visible para el resto de los usuarios del mismo grupo, pero estas dos opciones las explicaremos con mas detalle más adelante. Por el momento solo nos debe importar el seleccionar el objeto.

Al seleccionar el objeto, de manera dinámica se cargaran todos los atributos que han sido definidos para ese objeto al momento de su creación. Es en esta parte donde el usuario puede crear su búsqueda a su medida, ya que el sistema provee al usuario de un número de opciones que le permiten poder hacer consultas bien específicas. La siguiente figura nos muestra las opciones mencionadas, de las cuales explicaremos la función de cada una.

Ver Esta opción permite al usuario elegir si es que desea que el atributo (y sus respectivos valores) aparezcan en el resultado de la búsqueda. La virtud que tiene esta opción es que hace más cómoda la visualización de los resultados, ya que puede suceder que a un usuario sólo le interese saber por la historia de un atributo en particular y desprecu-

ncapsulada

BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

Búsqueda Encapsulada SMOTs

Nombre de Búsqueda:

Seleccione el MOT:

Permitir Visibilidad Grupal

Ver	Autoingreso	Sumar	Contar	Historia	Variable	Fijo	Atributo	Operador	Valor
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ATRIBUTO UNO	=	<input type="text"/>				
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ATRIBUTO DOS	=	<input type="text"/>				
		<input type="checkbox"/>	<input type="checkbox"/>	Fecha:	<input type="text" value="Elija una fecha"/>				

Sólo estado actual

SMOT - Sistema de Modelos de Objetos Trazables

Figura 18: Búsqueda local - Condiciones Búsqueda

parse el resto. O puede suceder que un objeto tenga muchos atributos y al momento de ver los resultados de la consulta, la visualización de ellos no sea de lo más amigable.

Autoingreso Esta opción es para hacer un ingreso de un registro utilizando como valores para los atributos los resultados de una búsqueda. Esto puede resultar beneficioso en determinadas ocasiones, no siempre, pero le permite al usuario ahorrarle algo de tiempo. Como se observa en la Figura 19, al seleccionar “Autoingreso” sobre algún atributo, el usuario debe seleccionar el objeto y el atributo perteneciente a ese objeto.

A continuación un ejemplo para explicar lo que hace el sistema al marcar la opción de autoingreso:

Se hace una búsqueda sobre el objeto “MOT1” el cual tiene los atributos ATTR MOT1 1 y ATTR MOT1 2. Se marca la opción de autoingreso para el atributo ATTR MOT1 2 y se selecciona el objeto MOT2 y su atributo ATTR MOT2 2. Se realiza la búsqueda. Luego, de la lista de resultados, el valor del atributo ATTR 2 es un link que lleva a la

Ver	Autoingreso	Sumar	Contar	Histori
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	<input type="text" value="MOT DE PRUEBA DOS"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="text" value="ATRIBUTO TRES"/>			

Figura 19: Autoingreso

interfaz de Ingreso de Registro. Se hace click sobre el link de la segunda fila y el sistema inmediatamente nos lleva al ingreso de registros para el objeto MOT 2. Sus atributos tienen los campos autocompletados con los valores de la fila 2.

De esta forma se le ahorra tiempo al usuario dándole la oportunidad de ingresar un registro casi automáticamente.

Sumar Esta opción se utiliza cuando se desea sumar los valores de un atributo. El usuario, al seleccionar esta opción, tendrá como resultado la suma de todos los valores de ese atributo, para todos aquellos registros del objeto que sean iguales (nos referimos a que los valores de los demás atributos que no fueron seleccionados para sumarse sean iguales)

Contar Es bastante similar a la opción de sumar, salvo que acá se cuenta el número de ocurrencias de cierto atributo

Historia Esta es una de las opciones más importante en las búsquedas, ya que es la que permite viajar a través de diferentes objetos para ir viendo su historia. Con esto se quiere decir que a partir de la trazabilidad de un objeto, SMOT permite llegar a la trazabilidad de otro objeto (que puede estar directa o indirectamente relacionado con el primer objeto). Para poder realizar este tipo de búsquedas “encadenadas” es necesario que existan diversas búsquedas guardadas en el sistema (ver *Guardar Búsquedas*). En pocas palabras, esta opción consiste en asociar una búsqueda, que se encuentra guardada en el sistema, al atributo seleccionado. Luego, en los resultados de la consulta, todos los valores de la columna del atributo seleccionado tendrán un link asociado.

Dicho link corresponde a la búsqueda que se seleccionó y que usará el valor del atributo como condición. Al presionar sobre el link se mostrará el resultado para esa búsqueda asociada y así se obtendrá la trazabilidad para este otro objeto. Esta otra búsqueda podría haber tenido asociada otra búsqueda guardada y así sucesivamente.

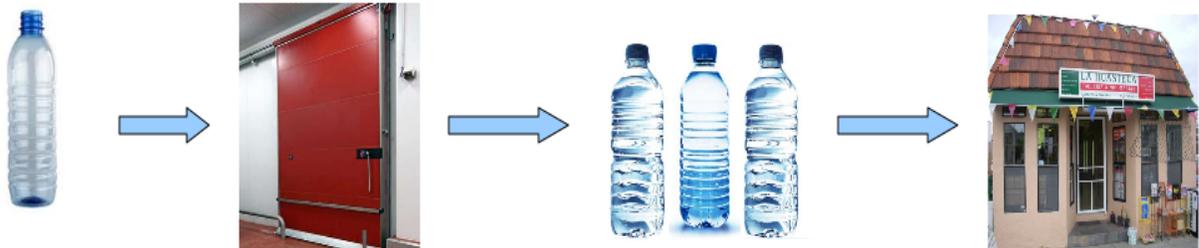


Figura 20: Ejemplo del uso de la opción Historia

En la Figura 20 se muestra un ejemplo más gráfico sobre el uso de esta opción. Lo que se muestra en la figura es que haciendo una búsqueda sobre el producto, obtengo la cámara de frío en la que se guardó el producto en determinado tiempo. Sabiendo la cámara de frío puedo obtener todos los productos que fueron almacenados durante cierto período de tiempo. Con ese listado, puedo seleccionar los diferentes productos y saber a que negocio se lo vendí. Más adelante en el documento se dará otro ejemplo y se podrán observar las ventajas de esta opción.

Variable Esta opción se usa exclusivamente para las búsquedas guardadas (o encapsuladas).

El que se seleccione un atributo como “variable” al guardar una búsqueda, significa que al usar esa búsqueda guardada, el sistema le solicitará al usuario que ingrese un valor para ese atributo, ya que es “variable”.

Fijo Con esto se quiere decir que cuando se guarde una búsqueda, este atributo no cambiará de valor, por lo que el sistema no le pedirá al usuario que ingrese un valor como condición de búsqueda para ese atributo.

Operador Esta característica es la que permite que las búsquedas sean hechas “a medida”, ya que le da la opción al usuario de que él ponga la condición de búsqueda para ese atributo, es decir, que él podría colocar, por ejemplo, que quiere buscar todos los registros del objeto “MOT DE PRUEBA” para los cuales “ATRIBUTO UNO” tiene

un valor ≤ 100 . Luego el sistema, internamente pondrá esa condición al momento de generar la consulta SQL, entregando como resultado todos aquellos registro que cumplan dicha condición

C .5.4. Búsqueda Externa

Como se mencionó anteriormente, SMOT permite interactuar con otras base de datos que no sea sobre la cual está construido SMOT. A este tipo de base de datos las llamamos “externas” y la interacción que tiene SMOT con ellas es básicamente en las búsquedas. En este caso, las llamadas “Búsquedas Externas” operan de la misma forma que las búsquedas locales, salvo una pequeña diferencia. Que en vez de seleccionar el objeto sobre el cual se va a realizar la consulta, se debe elegir la base de datos sobre la cual se consultará y, una vez hecho eso, se debe elegir la tabla de dicha base de datos. Hecho esto, aparecerá de forma dinámica, todas las columnas existentes en la tabla seleccionada. El usuario entonces opera de la misma forma como si fuera una consulta local, seleccionando que valores querrá ver, si sumará o contará valores, etc.

C .5.5. Guardar Búsqueda

El motivo por el que se agregó esta función es doble. Primero porque le ahorrará mucho tiempo a los usuarios que siempre realicen la misma búsqueda, ya que no tendrán que “armar” la búsqueda cada vez que deseen consultar por un valor, si no que basta con que guarde la búsqueda una vez y luego la ocupe cada vez que la necesite. Y segundo porque las búsquedas guardadas permiten obtener la historia de diversos objetos relacionados entre sí mediante un encadenamiento.

Las búsquedas guardadas solo son visibles en el sistema para aquellos usuarios que crearon y guardaron la búsqueda (salvo el usuario administrador, que puede ver todas las búsquedas guardadas existentes). El sistema permite que, además de que sean vista por el propio usuario, también puedan ser vistas por los pertenecientes al mismo grupo. Por ejemplo, un usuario del grupo Ventas puede dejar que los demás usuarios de su grupo vean las búsquedas que el guarda para que también las ocupen.

El proceso de guardado de búsquedas comienza cuando se quiere realizar una búsqueda

local o externa. El usuario debe ingresarle un nombre, de lo contrario, cuando trate de guardarla, el sistema le entregará un aviso de error de que debe colocarle un nombre pues de lo contrario no podrá ser guardada. Luego realiza los mismos pasos que los explicados para la búsqueda local o externa. Para poder guardar la búsqueda el usuario solo debe presionar el botón “Encapsular” y quedará guardada en la base de datos y el sistema reenviará al usuario a la interfaz donde se encuentran listadas todas las búsquedas locales (o externas, dependiendo que búsqueda realizó). La Figura 21 muestra un listado de las búsquedas guardadas. Luego, cuando un usuario desee ejecutar alguna de ellas, basta con que en el menú de búsquedas vaya a “Búsquedas Internas Guardadas” (o “Búsquedas Externas Guardadas, según sea el caso) y ejecute la búsqueda que desee.

SMOT (USER)
HOME :: Búsquedas Guardadas :: Ver listado

OK
Se encapsulo correctamente

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Listado de Búsquedas

<< first < prev 1 2 3 4 5 next > last >>

ID	Name	Opciones
45	Lista Productos Id Producto Variable (Presentacion)	[Editar] [Eliminar]
46	Lista Productos en Camara de Frio id Camara Variable	[Editar] [Eliminar]
47	Listado Camaras id Variable	[Editar] [Eliminar]
48	Producto 34 con lote 145 en camara de frio (Presentacion)	[Editar] [Eliminar]
49	Lista Productos Vendidos Id Producto Variable (Presentacion)	[Editar] [Eliminar]
50	BUSQUEDA DE PRUEBA	[Editar] [Eliminar]

<< first < prev 1 2 3 4 5 next > last >>

SMOT - Sistema de Modelos de Objetos Trazables

Figura 21: Listado de Búsquedas Guardadas

Si ocurre que la búsqueda que guardó tenía algún atributo puesto como "Variable", entonces al seleccionarla, el sistema le pedirá al usuario que ingrese el valor del(los) atributo(s) seteados como variables, tal como se muestra en la Figura 22.

Las búsquedas guardadas también se pueden editar. Esta característica es muy útil porque facilita el poder realizar las búsquedas encadenadas, ya que sin la edición, para poder hacer el encadenamiento, necesariamente el usuario debe saber qué tipo de búsquedas necesita y crearlas en orden inverso, es decir que si por ejemplo la cadena de búsquedas es 1 - 2 - 3 - 4 - 5, por obligación primero debe crear y guardar la búsqueda 5, ya que la 4 tendrá asociada la 5 y si ésta no existe entonces no aparecerá en el listado de búsquedas disponibles. Esto limita bastante al usuario, pero con la edición se facilita todo pues si, por ejemplo, el usuario tiene una cadena de búsquedas 1 - 2 - 3 - 4 y desea agregar una quinta búsqueda a la cadena, basta con que cree la búsqueda 5 y luego edite la búsqueda 4 para que ahora asocie algún atributo a la recientemente creada. Sin la edición esto claramente no hubiera sido posible y el usuario hubiera tenido que comenzar todo de nuevo para tener la cadena deseada.

The screenshot shows the SMOT (USER) interface. At the top, there is a navigation bar with the following links: < Home >, < MOTs >, < BD externas >, < Registros >, < Usuarios >, < Roles >, < Búsqueda >, < ONS Queries >, and < Logout >. Below the navigation bar, there is a sidebar menu with the following items: Por Valor en Registro, Por Código de MOT, Encapsulada SMOTs, Encapsulada Externa, Búsquedas Internas Guardadas, and Búsquedas Externas Guardadas. The main content area is titled "Búsqueda: BUSQUEDA DE PRUEBA" and contains a search form with the following structure:

Atributo	Operador	Valor
ATRIBUTO UNO	=	<input type="text"/>

Below the form is a "Buscar" button. At the bottom right of the page, there is a footer that reads "SMOT - Sistema de Modelos de Objetos Trazables".

Figura 22: Búsqueda Guardada - Atributo Variable

C .5.6. Consulta ONS

La otra gran funcionalidad presente en el sistema y que hace frente a uno de los problemas planteados en un comienzo, es la de integrar el sistema de trazabilidad con la tecnología RFID y con ONS. Esta integración se puede ver por dos lados: Por uno lado, el usuario utiliza SMOT para realizar la consulta ONS y se le entrega una lista de servicios como resultado y en esa lista se entrega una URL a un módulo de SMOT que entrega la trazabilidad del producto asociado al EPC. Y por el otro lado, el usuario puede realizar su consulta ONS por otros medios y dentro de los resultados obtenidos, el usuario puede seleccionar la URL de SMOT, entregándole la trazabilidad del producto.

Para el primer caso antes es necesario contar con ciertos elementos para realizar a consulta ONS. Estos corresponden a un RFID y un lector de RFID. Tal como se describió al comienzo del documento, un RFID es un sistema de almacenamiento y recuperación de datos remoto. Para nuestro caso usaremos los *tags*. Estos existen en diversos tamaños y formas. La característica que tienen es que son programables, es decir que se les puede agregar valores que posteriormente serán leídos por un lector de RFID. En nuestro caso, a cada tag se le inserta un EPC y es asociado a un producto en particular. La manera de asociarlos típicamente es adheriéndolos al producto (el tipo de tag a usar dependerá de los recursos y necesidades de la empresa). Una vez que todos los productos tienen su tag asociado, se utiliza el lector RFID para obtener los EPC's asociados a cada uno de los productos. Aquí es donde entra en acción SMOT, ya que con el valor rescatado del tag, se ingresa el EPC en el sistema, luego SMOT realiza la consulta ONS y entrega como resultado una lista de servicios. Entre estos servicios se encuentra una URL a SMOT. Si el usuario selecciona esa URL, SMOT le entregará la historia completa de ese producto. Como en esta oportunidad no es posible establecer condiciones de búsqueda, el sistema entregará todos los registros que se encuentren almacenados en la base de datos.

El segundo caso expuesto es un subconjunto del primero, ya que en esta ocasión el usuario no utiliza SMOT para realizar la consulta ONS, pero en la lista de servicios si aparece la URL a SMOT, entregándole todos los registros del producto.

En vista de que el segundo caso es un subconjunto del primero, a continuación solo mos-

traremos un ejemplo de uso para el primer caso. Parte del ejemplo se aplica también para el segundo caso.

La Figura 23 muestra como el usuario debe colocar el código EPC para poder realizar la consulta. En caso de que intente realizar la consulta dejando en blanco el campo requerido, el sistema lanza un aviso exigiéndole que ingrese un valor.

The screenshot shows the SMOT web application interface. At the top, there is a blue header with the SMOT logo and the text 'HOME :: ONS QUERY :: Consulta ONS'. Below the header is a navigation menu with links: '< Home >', '< MOTs >', '< BD externas >', '< Registros >', '< Usuarios >', '< Roles >', '< Búsqueda >', '< ONS Queries >', and '< Logout >'. On the left side, there is a blue sidebar with a menu item '• Nueva Consulta ONS'. The main content area is white and titled 'Consulta ONS'. It features a form with a text input field labeled 'Ingrese el código EPC a consultar' and a 'Hacer Consulta' button. At the bottom right of the page, there is a footer with the text 'SMOT - Sistema de Modelos de Objetos Trazables'.

Figura 23: Consulta ONS

Si es que ocurriese que el código EPC no está asociado a ningún producto, la consulta ONS no mostrará ningún servicio para ese código. De no ser así, el sistema mostrará los servicios disponibles para dicho producto. En la Figura 24 se muestra el listado de servicios que entrega la consulta. Solamente para efectos ilustrativos, en la figura se muestran dos tablas de resultados. La primera muestra los resultados de la consulta ONS tal cual ésta los entrega, tal como se mostró anteriormente (ver *Antecedentes - ONS*). La segunda tabla muestra como SMOT entrega la lista de servicios obtenida de la consulta ONS. La razón por la que se muestra esto en la figura es para que el usuario vea que SMOT entrega los resultados

de una forma más amigable, pues de esta manera el usuario solo debe hacer click en el link entregado y se mostrarán los registros existentes para ese producto. En la actualidad SMOT solo muestra la segunda tabla.

SMOT
HOME :: ONS QUERY :: Consulta ONS :: Resultado Consulta ONS

[< Home >](#) [< MOTs >](#) [< BD externas >](#) [< Registros >](#) [< Usuarios >](#) [< Roles >](#) [< Búsqueda >](#) [< ONS Queries >](#) [< Logout >](#)

- Nueva Consulta ONS

Resultado Consulta ONS

Order	Pref	Flags	Service	Regexp	Replacement
0	0	u	EPC+html	!^.*\$!http://201.214.63.157/smot/resultado_ons/mot_de_prueba!	.

Preferencia	Tipo Servicio	Servicio
0	HTML	Trazabilidad de MOT DE PRUEBA

SMOT - Sistema de Modelos de Objetos Trazables

Figura 24: Respuesta Consulta ONS

D . Validación de la Solución

Para verificar que nuestra solución propuesta es válida y resuelve los problemas planteados, a continuación se mostrarán dos casos de uso para diferentes modelos de negocio, y se verificará que SMOT cumpla entregando la trazabilidad deseada para ambos casos.

Para preparar la solución se tomaron a modo de ejemplo dos modelos de negocio. El primero es el de un taller automotriz y el segundo el de un empresa que vende productos congelados a distintos centros de venta. Cabe mencionar que los ejemplos acá mostrados se encuentran simplificados para efectos de la validación de la solución. El modelo de negocios para los dos ejemplos son más complejos de lo que acá se muestra.

D .1. Ejemplo Taller Mecánico

El primer ejemplo es el de un taller mecánico. Para el taller se han creado los siguientes MOT's:

- Cliente: Con atributos Nombre, RUT y Dirección.
- Mecánico: Con atributos Nombre.
- Vehículo: Con atributos Patente, Cliente, Nombre Mecánico y Mantenimiento Realizada.
- Trabajos: Con atributos Mantenimiento.

La idea de este ejemplo es mostrar una búsqueda encadenada simple y corta en la que, a partir del nombre de un mecánico, puedo averiguar todos los trabajos que realizó y a quien le realizó esos trabajos, y a partir de ahí, obtener la dirección de algún cliente para avisarle, por ejemplo, que su mecánico habitual se encuentra de vacaciones y no estará disponible para atenderlo. Con esto podemos ver como SMOT sirve no solo para ver la trazabilidad (en este caso de los trabajos realizados por un mecánico) si no que también para otros fines, todo a partir de una búsqueda.

En la Figura 25 se observa que se realizó una búsqueda encadenada, en la que se asoció al atributo "Nombre" la búsqueda "Vehiculos del Mecanico", la cual es una búsqueda guardada

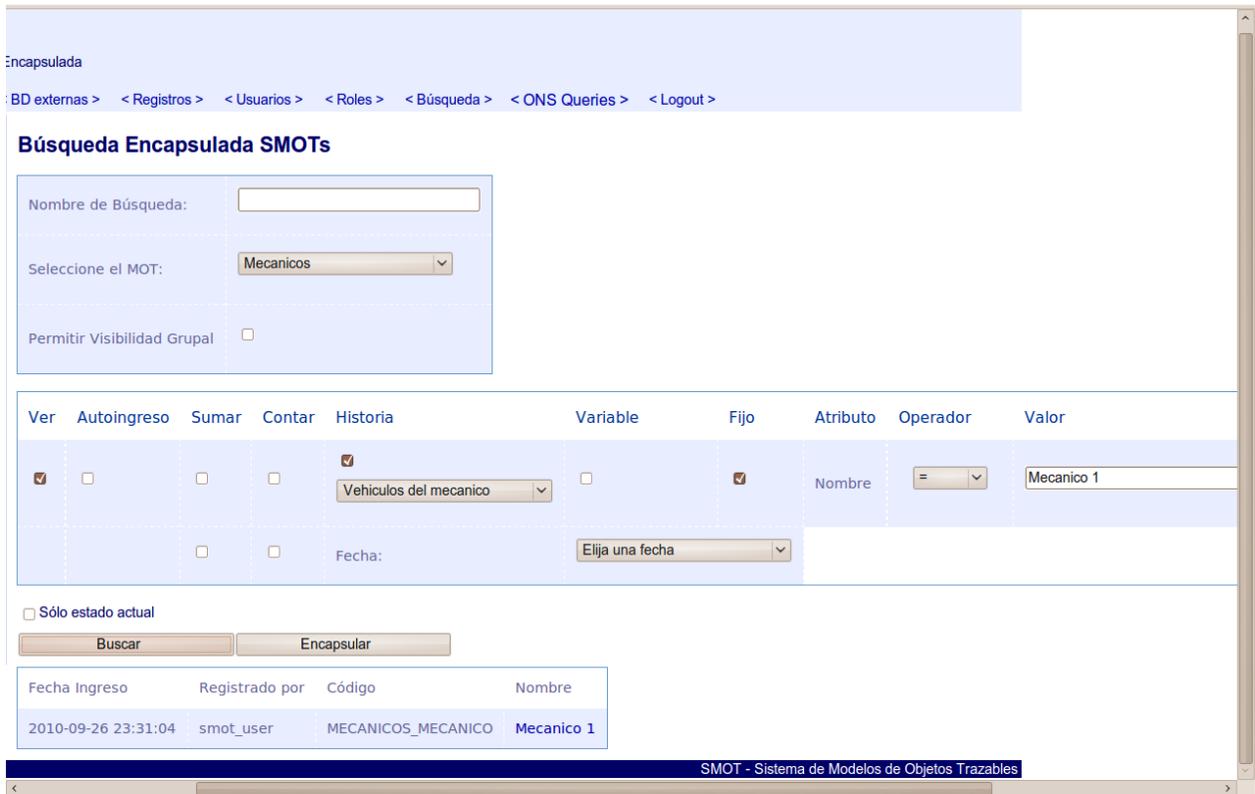


Figura 25: Ejemplo taller mecánico - Primera Consulta

en el sistema y que entrega todos los vehiculos que fueron revisados por un mecánico, donde el nombre del mecánico es un atributo variable. Del resultado de la búsqueda, en vista de que se asoció una búsqueda al atributo "Nombre", todos los valores de la columna "Nombre" en la tabla de resultados serán un link a la búsqueda asociada, en este caso "Vehiculos del Mecanico", y el valor que ahí aparece será el valor entregado para el atributo variable mencionado anteriormente. La Figura 26 muestra el resultado obtenido al hacer hacer click sobre el link.

Como se puede observar del resultado, el mecanico "Mecanico 1" ha trabajado en el vehículo de dos clientes, el "Cliente 1" y el "Cliente 2", para el primero le ha trabajado en 2 automóviles y para el segundo solo en uno. Nuevamente se presenta una búsqueda encadenada, ya que los valores de la columna "Cliente" son links que llevan al usuario a una última consulta, que corresponde a los datos del cliente. Esta consulta tambien fue previamente guardada. Finalmente la Figura 27 muestra el resultado final de esta búsqueda encadenada, entregando los datos del cliente.

SMOT (USER)

HOME :: Búsqueda :: Búsqueda Encapsulada :: Encadenamiento

[< Home >](#) [< MOTs >](#) [< BD externas >](#) [< Registros >](#) [< Usuarios >](#) [< Roles >](#) [< Búsqueda >](#) [< ONS Queries >](#) [< Logout >](#)

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Resultado Búsqueda Encadenada

Fecha Ingreso	Registrado por	Código	Patente	Cliente	Nombre Mecanico	Mantenion Realizada
2010-09-26 23:29:21	smot_user	VEHICULO_MECANICO	44-44-DD	Cliente 1	Mecanico 1	Motor
2010-09-26 23:29:21	smot_user	VEHICULO_MECANICO	22-22-BB	Cliente 2	Mecanico 1	Neumaticos
2010-09-26 23:29:21	smot_user	VEHICULO_MECANICO	11-11-AA	Cliente 1	Mecanico 1	Luces

SMOT - Sistema de Modelos de Objetos Trazables

Figura 26: Ejemplo taller mecánico - Segunda Consulta

Finalmente la Figura 27 muestra el resultado final de esta búsqueda encadenada, entregando los datos del cliente

SMOT (USER)
HOME :: Búsqueda :: Búsqueda Encapsulada :: Encadenamiento

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Resultado Busqueda Encadenada

Fecha Ingreso	Registrado por	Código	Nombre	RUT	Direccion
2010-09-26 23:46:34	smot_user	CLIENTE_MECANICO	Cliente 1	11.111.111-1	Direccion 1

SMOT - Sistema de Modelos de Objetos Trazables

Figura 27: Ejemplo taller mecánico - Tercera Consulta

D .2. Ejemplo Productos Congelados

En este ejemplo también se realizará una búsqueda encadenada en la que la situación es la siguiente: Se tiene un producto "PROD A" que está defectuoso debido a que no estuvo sujeto a las temperaturas adecuadas en la cámara de frío, por lo tanto es posible que todos los productos que se encontraban dentro de la cámara junto con el producto "PROD A" también presenten problemas, por lo que la empresa decide consultar a SMOT para saber que productos estaban en la cámara y a quienes se le vendieron dichos productos, para avisarles de la posibilidad de que deban retirar el producto de la venta.

Las siguientes 6 imagenes muestran el flujo que comienza con una búsqueda por el objeto PRODUCTOS, consultando por el Producto en mal estado. Con el valor del id de la cámara de frío se obtiene que cámara es la que albergó al producto. Con ese resultado, y utilizando el id de la cámara, se obtienen todos los productos que estuvieron en esa cámara. Con ese resultado se selecciona algún producto y se obtiene el identificador de a quien se le vendió. Con ese identificador se obtiene la información completa del comprador. Para efectos de una

mejor visualización en el documento, los valores que se encuentran como links se marcaron con un círculo rojo.

SMOT (USER)
HOME :: Búsqueda :: Búsqueda Encapsulada :: Resultados

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Búsqueda:

Fecha Ingreso	Registrado por	Código	id_camara	id_producto	lote	fecha	estado
2010-05-01 18:34:29	smot_user	REG_CAMARA_8	1	6	8	2010-05-01	O

SMOT - Sistema de Modelos de Objetos Trazables

Figura 28: Ejemplo Cámara de Frío - Primera Consulta

SMOT (USER)

HOME :: Búsqueda :: Búsqueda Encapsulada :: Encadenamiento

[< Home >](#) [< MOTs >](#) [< BD externas >](#) [< Registros >](#) [< Usuarios >](#) [< Roles >](#) [< Búsqueda >](#) [< ONS Queries >](#) [< Logout >](#)

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Resultado Busqueda Encadenada

Fecha Ingreso	Registrado por	Código	id_camara	ubicacion	responsable
2010-06-10 18:31:30	smot_user	CAMARA_1	1	Camara Secundaria	Encargado 1

SMOT - Sistema de Modelos de Objetos Trazables

Figura 29: Ejemplo Cámara de Frío - Segunda Consulta

SMOT (USER)

HOME :: Búsqueda :: Búsqueda Encapsulada :: Encadenamiento

[< Home >](#) [< MOTs >](#) [< BD externas >](#) [< Registros >](#) [< Usuarios >](#) [< Roles >](#) [< Búsqueda >](#) [< ONS Queries >](#) [< Logout >](#)

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Resultado Busqueda Encadenada

Fecha Ingreso	Registrado por	Código	id_camara	id_producto	lote	fecha	estado
2010-05-01 18:34:29	smot_user	REG_CAMARA_8	1	6	8	2010-05-01	O
2010-04-15 01:13:15	smot_user	REG_CAMARA_7	1	22	7	2010-04-15	I
2010-04-05 12:02:28	smot_user	REG_CAMARA_0	1	36	---	2010-04-05	I
2010-02-21 18:18:39	smot_user	REG_CAMARA_5	1	57	5	2010-02-21	I
2010-01-28 17:01:16	smot_user	REG_CAMARA_2	1	39	2	2010-01-28	I

SMOT - Sistema de Modelos de Objetos Trazables

Figura 30: Ejemplo Cámara de Frío - Tercera Consulta

SMOT (USER)

HOME :: Búsqueda :: Búsqueda Encapsulada :: Encadenamiento

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Resultado Busqueda Encadenada

Fecha Ingreso	Registrado por	Código	id_proveedor	id_producto	fecha_venta	nombre_proveedor
2010-01-19 02:38:06	smot_user	REG_VENTA_0	7	6	2010-01-19	proveedor_7

SMOT - Sistema de Modelos de Objetos Trazables

Figura 31: Ejemplo Cámara de Frío - Cuarta Consulta

SMOT (USER)

HOME :: Búsqueda :: Búsqueda Encapsulada :: Encadenamiento

[< Home >](#) [< MOTs >](#) [< BD externas >](#) [< Registros >](#) [< Usuarios >](#) [< Roles >](#) [< Búsqueda >](#) [< ONS Queries >](#) [< Logout >](#)

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Resultado Busqueda Encadenada

Fecha Ingreso	Registrado por	Código	id_proveedor	nombre_proveedor	direccion_proveedor
2010-04-01 03:21:02	smot_user	PROVEEDOR_7	7	proveedor_7	direccion_7

SMOT - Sistema de Modelos de Objetos Trazables

Figura 32: Ejemplo Cámara de Frío - Quinta Consulta

D .3. Ejemplo Validación Consulta ONS

Para mostrar la validación de la integración de fue necesario simular que se tenía un código EPC valido y por lo tanto se simuló que dentro de los servicios disponibles, al momento de obtener una respuesta de la consulta ONS, se encontraba la URL de la trazabilidad del producto. Las siguientes imagenes muestran el flujo que se siguió para obtener la trazabilidad de un producto a través de su EPC:

The screenshot displays the SMOT web application interface. At the top left, the logo 'SMOT' is visible, followed by the breadcrumb 'HOME :: ONS QUERY :: Consulta ONS'. A navigation menu includes links for '< Home >', '< MOTs >', '< BD externas >', '< Registros >', '< Usuarios >', '< Roles >', '< Búsqueda >', '< ONS Queries >', and '< Logout >'. On the left sidebar, there is a menu item '• Nueva Consulta ONS'. The main heading is 'Consulta ONS'. The form area contains a label 'Ingrese el código EPC a consultar' and a text input field with the value 'c:id:sgtin:0614141.011015.583865'. Below the input field is a button labeled 'Hacer Consulta'. The footer of the page reads 'SMOT - Sistema de Modelos de Objetos Trazables'.

Figura 33: Ejemplo Validación Consulta ONS - Ingreso EPC

En este ejemplo se utilizó el EPC: urn:epc:id:sgtin:0614141.011015.583865 para simular un código EPC válido y utilizarlo para hacer la consulta ONS

SMOT
HOME :: ONS QUERY :: Consulta ONS :: Resultado Consulta ONS

[< Home >](#) [< MOTs >](#) [< BD externas >](#) [< Registros >](#) [< Usuarios >](#) [< Roles >](#) [< Búsqueda >](#) [< ONS Queries >](#) [< Logout >](#)

- Nueva Consulta ONS

Resultado Consulta ONS

Preferencia	Tipo Servicio	Servicio
0	HTML	Trazabilidad de MOT DE PRUEBA

SMOT - Sistema de Modelos de Objetos Trazables

Figura 34: Ejemplo Validación Consulta ONS - Lista de Servicios

Para esta simulación solo se agregó un servicio, que en este caso resulta ser la URL a SMOT que le entrega la trazabilidad del producto MOT DE PRUEBA. Pueden existir más servicios para este producto, pero para efectos de la demostración sólo se utilizó uno

SMOT (USER)
HOME :: Búsqueda :: Búsqueda Encapsulada :: Resultados

< Home > < MOTs > < BD externas > < Registros > < Usuarios > < Roles > < Búsqueda > < ONS Queries > < Logout >

- Por Valor en Registro
- Por Código de MOT
- Encapsulada SMOTs
- Encapsulada Externa
- Búsquedas Internas Guardadas
- Búsquedas Externas Guardadas

Búsqueda:

Fecha Ingreso	Registrado por	Código	ATRIBUTO UNO	ATRIBUTO DOS
2010-09-27 06:12:39	smot_user	Código Prueba	valor uno v2	valor dos
2010-09-27 06:12:39	smot_user	Codigo Prueba	valor uno	valor dos

SMOT - Sistema de Modelos de Objetos Trazables

Figura 35: Ejemplo Validación Consulta ONS - Trazabilidad Producto

La razón por la que fue necesario simular los datos es por que al momento de escribir este documento, no se tuvo acceso a información sobre como asignar un EPC válido, de manera oficial, a un producto de manera tal que se pudiera hacer una consulta ONS válida y que entregara servicios asociados a ese EPC. Tampoco se obtuvo información de cómo agregar un servicio de manera oficial (en nuestro caso la URL a SMOT). Pero esto no es lapidario a la hora de evaluar la validez del sistema en cuanto a su integración con ONS, ya que estos dos puntos no tienen que ver con su funcionamiento, pues ya se sabe qué es lo que retorna una consulta ONS sobre un código EPC válido y ya se sabe la forma en que ONS entrega dichos resultados a través de la lista de servicios. Luego nos es posible simular un caso de uso con valores ficticios.

Queda como trabajo a futuro lograr tener un EPC válido para realizar las pruebas pertinentes con SMOT.

D .4. Eficiencia en consultas

Además de que el sistema sea capaz de llevar la trazabilidad de los objetos, también es importante saber el tiempo de respuesta que tiene SMOT frente a una consulta. Saber hasta que cantidad de registros soporta el sistema para que aún entregue resultados dentro de un tiempo razonable es muy importante, pues no sirve de nada poder hacer lo que hace SMOT si es que se demora 30 minutos en responder a una consulta.

Se realizaron pruebas con diferentes cantidades de registros para ver hasta que punto se obtenían resultados aceptables. A continuación se dan a conocer dichas pruebas simulando el ingreso de registros en 1 año, 5 años y 10 años para una empresa mediana/grande:

1. 1 año simulado: Se obtuvo que el inventario de una base de datos con 100.000 registros se demoró 36,47 segundos
2. 5 años simulado: Se obtuvo que el inventario de una base de datos con 400.000 registros se demoró 49,06 segundos
3. 10 años simulado: Se obtuvo que el inventario de una base de datos con 800.000 registros se demoró 80,93 segundos

Independientemente de si el cliente queda satisfecho con estos tiempos de respuesta, queda como trabajo a futuro el optimizar las búsquedas de manera de disminuir cada uno de estos tiempos.

Conclusión

Durante el desarrollo de esta tesis se abordaron diversos problemas, todos con un denominador común, la **trazabilidad**. El como desarrollar un sistema que fuera capaz de llevar la trazabilidad de una empresa pero que además fuera capaz de adaptarse a cualquier modelo de negocio se convirtió en un desafío, al igual que el aprovechar las nuevas tecnologías e integrarlas a este nuevo sistema.

De los objetivos planteados se logró cumplir casi con la totalidad de ellos. Es decir, se logró construir un sistema que es capaz no solo de llevar la trazabilidad de los productos de una empresa, si no que además es capaz de llevar la trazabilidad de prácticamente cualquier cosa (personal, acciones, etc), no solo de productos. También es un sistema adaptable a cualquier escenario, desde el más simple al más complejo. Es suficiente con sentarse a conversar cual es el modelo de negocio que posee la empresa y luego modelarlo con el sistema (a través de la creación de objetos, búsquedas, etc).

Se dijo que casi se logró cumplir con la totalidad de los objetivos planteados, ya que, si bien se integró el uso de RFID y ONS al sistema, no se logró hacerlo con datos reales, ya que no se contó con la información sobre como crear un propio EPC válido ni como agregar a los servicios entregados por la consulta ONS la URL hacia SMOT. Solo fue posible simular la situación con valores ficticios. Queda como trabajo a futuro el lograr una integración completa utilizando datos reales.

Durante gran parte del documento se recalcó la importancia de contar con un sistema de estas características. Este tipo de sistemas son costosos de hacer, requieren de un proceso largo de entendimiento del negocio, para luego recién comenzar con la implementación y esperar a que cumpla con todos los requisitos de la empresa, pero con SMOT solo basta

con sentarse a conversar el modelo de negocio y construir los objetos en base a eso y es suficiente. Claramente el ahorro de recursos, especialmente de dinero y tiempo, que se logra con SMOT es de gran relevancia y lo convierte en un sistema a tomar en cuenta. Además, con su estructura modular, se deja espacio para agregarle nuevas funcionalidades, lo que expande aún mas el rango de posibilidades de aplicación. También, por último, cabe mencionar que en este sistema ningún dato es eliminado, todo se mantiene en la base de datos, ya que de esta forma es posible asegurar una trazabilidad como corresponde.

Referencias

[1] Implementation and Testing of an EPCglobal-aware Discovery Service for Item-level Traceability, Ugo Barchetti, Alberto Bucciero, Mario De Blasi, Luca Mainetti and Luigi Patrono. Department of Innovation Engineering, University of Salento. Lecce, Italy.

[2] Discovery Architecture for the Tracing of Products in the EPCglobal Network, Gyeongtaek Lee, Jonghun Shin, Daewon Park, Hyukchul Kwon. Department of Computer Science And Engineering, Pusan National University, Korea.

[3] A Fair Solution to DNS Amplification Attacks, Georgios Kambourakis, Tassos Moschos, Dimitris Geneiatakis and Stefanos Gritzalis. Laboratory of Information and Communication Systems Security, Department of Information and Communication System Engineering, University of the Aegean, Karlovassi.

[4] WSEC DNS: Protecting Recursive DNS Resolvers from Poisoning Attacks, Roberto Perdisci, Manos Antonakakis, Xiapu Luo and Wenke Lee. College of Computing, Georgia Institute of Technology, Atlanta, USA.

[5] Consideration for Efficient RFID Information Retrieval in Traceability System, Rei Itsuki and Atsushi Fujita, Hiroshima International University and Computer System Engineering Co., Ltda. Japan.

[6] RFID: A Technical Overview and Its Application to the Enterprise, Ron Weinstein, IEEE Journals, Pages 27- 33, 2005.

[7] ONS Standard v.1.0.1

[8] Network Information Center (NIC), www.nic.cl