

2. Marco Teórico

En este capítulo analizaremos los cuatro diferentes métodos para obtener la información, para que en base a los resultados de este análisis, poder seleccionar la plataforma de diseño adecuada, el lenguaje a utilizar en cada una de las tecnologías, así como la base de datos para este sistema.

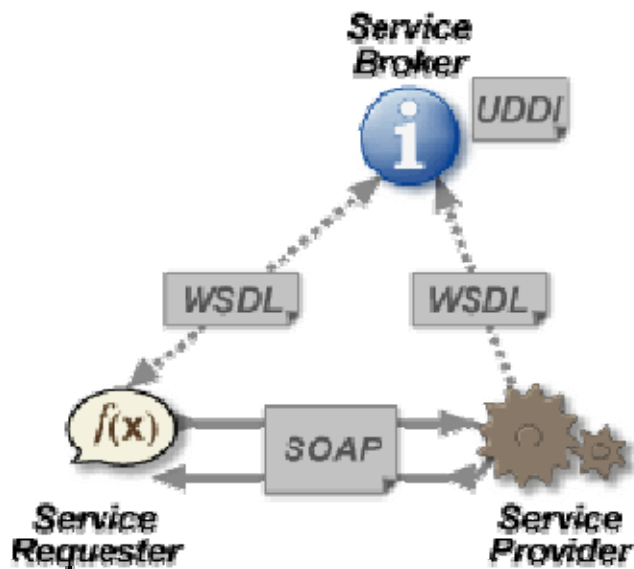
2.1. Web Services

En esta sección explicaremos conceptos relacionados a los Web Services, veremos el funcionamiento general de los Web Services, y específicamente hablaremos del servicio que estaremos utilizando para obtener las reservaciones basadas en las especificaciones *OTA_HotelResNotifRQ* de OTA.

Un Web Service es un servicio disponible en Internet que utiliza mensajes XML para la transmisión de información. Una de las principales ventajas de los Web Services es la interoperabilidad en diferentes sistemas. Estos servicios se crearon debido al problema que se enfrentaron las compañías de compartir información, es por eso que se crean los Web Services [Keith Ballinger, 2003]. Un Web Service debe contener un WSDL (Web Service Descriptor Language), el cual muestra una descripción publica de la interfase. Los tres estilos mas utilizados para el uso de Web Services son:

- RPC (Remote Procedure Call): Representa una llamada a una función o método. La unidad básica de los Web Services RPC son los WSDL.
- SOA (Service-Oriented Architecture): Se pueden utilizar para implementar una arquitectura de acuerdo a los conceptos SOA, donde la unidad básica de funcionamiento es únicamente el mensaje.
- REST (Representational State Transfer): Describe arquitecturas utilizando protocolos similares a http limitando la interfase a operaciones como GET, POST, PUT, DELETE)

Nosotros estaremos enfocándonos al RPC, utilizando SOAP para encapsular los mensajes. El siguiente diagrama muestra la arquitectura general de los Web Services



Gráfica 2.1. Web Services, Wikipedia

2.1.1. XML

Por sus siglas en inglés eXtensive Markup Language podemos decir que son documentos electrónicos que siguen un conjunto de reglas donde definen una sintaxis genérica utilizada para localizar datos con etiquetas que pueden ser leídas por un humano [Elliott Rusty Harold & W. Scott Means, 2004]. El objetivo del diseño de los XML fue la simplicidad, generabilidad y usabilidad en Internet. Aunque este diseño se enfocó en documentos en un principio, hoy en día es utilizado para representar estructuras de datos en Web Services. A continuación veremos un ejemplo básico de una cancelación en un documento XML.

```
<?xml version="1.0" encoding="utf-8"?>
<OTA_HotelResNotifRS xmlns="http://www.opentravel.org/OTA/2003/05"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05
OTA_HotelResNotifRS.xsd"
  EchoToken="354"
  TimeStamp="2010-08-24T23:55:52-05:00"
  Target="Production"
  Version="1.000"
  ResResponseType="Canceled">
  <Success />
  <HotelReservations>
    <HotelReservation>
      <UniqueID Type="14" ID="ABE125" />
      <ResGlobalInfo>
        <HotelReservationIDs>
          <HotelReservationID ResID_Type="10" ResID_Value="RRLTKOAO" />
        </HotelReservationIDs>
      </ResGlobalInfo>
    </HotelReservation>
  </HotelReservations>
</OTA_HotelResNotifRS>
```

2.1.2. OTA

Open Travel Alliance [Open Travel, 2010] es una asociación que ha puesto un estándar en la transmisión de información en el campo turístico a través de archivos XML. Esta asociación no lucrativa certifica compañías con el uso de sus mensajes. Debido a que hoy en día, OTA se ha convertido en un estándar, se desarrollara la de-codificación de las reservaciones basándonos en los mensajes de reservaciones correspondientes de OTA. Para las reservaciones existe un tipo de mensaje el cual contiene información para una nueva, modificada o cancelada. En el Anexo A, podemos encontrar tres ejemplos de reservaciones, una nueva, una modificada, y una cancelada. Estos documentos los tomaremos de ejemplo para el funcionamiento de este sistema.

El objetivo de nuestro Web Service es decodificar la información en las reservaciones provenientes bajo las especificaciones de OTA para estas insertarlas en una base de datos. Los campos básicos para una reservación son el numero de la reservación, fecha de generación, tipo de movimiento (nueva, modificada, cancelada), tipo de habitación, tipo de tarifa, fecha de llegada, fecha de salida, cantidad de adultos, cantidad de niños, nombre de pasajeros, y la información de la llegada, tal como es el vuelo, y la hora, en caso de existir. En un envío, pueden venir conjuntos de nuevas, modificadas o canceladas, pero nunca diferentes movimientos en un solo envío.

2.2. Correo Electrónico

La mayoría de los Tour Operadores, envían sus reservaciones a los hoteles vía email cada vez que se genera algún movimiento (una reservación nueva, modificada o cancelada) en su sistema. Para poder hacer la decodificación de los correos electrónicos, lo primero será generar un sistema que se conecte al servidor de email, busque nuevas reservaciones, y en caso de existir nuevas, estas bajarlas a un directorio como archivo de texto, para que otro proceso las decodifique y las inserte a la base de datos. Para el propósito de esta tesis se ha creado este correo electrónico tesis.reservas@gmail.com, al cual se le estarán enviando las reservaciones. En este caso hemos seleccionado a Cheap Caribbean como nuestro Tour Operador. La metodología a utilizar para la decodificación de este tipo de reservaciones será con lenguaje supervisado, donde primero recolectaremos por lo menos 500 diferentes correos electrónicos para estos clasificarlos [Bing Liu, 1998].

2.2.1. Cheap Caribbean

Cualquier persona puede entrar a <http://www.cheapcaribbean.com> y seleccionar un paquete para sus vacaciones, el cual incluye el vuelo, o únicamente seleccionar el hotel. Para poder hacer la reservación, el cliente tiene que seleccionar un destino, la fecha de salida, y de su regreso, el hotel, el tipo de habitación y tarifa, llenar la información requerida, como es su nombre, dirección, datos de su tarjeta de crédito, y nombre de pasajeros, entre otros. Al finalizar esta reservación, el cliente debe recibir una confirmación

en su pantalla. En este momento, Cheap Caribbean envía un email al hotel seleccionado por el cliente, confirmándoles la próxima llegada de estos pasajeros. En el Apéndice B podemos ver los dos tipos de movimientos que envía Cheap Caribbean, los cuales son reservaciones nuevas, y reservaciones modificadas.

2.2.2. Decodificación de Cheap Caribbean

La información que deberemos decodificar de una reservación de Cheap Caribbean es, el movimiento de la reservación (nueva o cancelada), numero de reservación, información de vuelos de llegada en caso de existir, fecha de generación, nombre de hotel, total de pasajeros, nombre de pasajeros, fecha de llegada, fecha de salida, tipo de habitación, tipo de tarifa, y comentarios. Debido a que la única información que cambia en cada correo electrónico, es la que acabamos de mencionar, será muy fácil poder extraer esta misma, ya que el sistema se basara en búsquedas de etiquetas en el HTML para poder saber la posición donde comienza y donde termina nuestro campo a buscar. Para poder realizar esta decodificación, se necesitaran previamente recolectar por lo menos 500 diferentes movimientos de reservaciones, para estas analizarlas y poder descubrir un patrón y poder convertirlo a reglas [Anthony Scime, 2005] donde fácilmente podremos encontrar la información que buscamos.

2.3. Flat File

Hoy en día, aun existen Tour Operadores que envían información en “Flat Files”, estos pueden ser en archivos separados por coma, o en archivos de texto con una cierta estructura diseñada por estos Tour Operadores. La mayoría de los Tour Operadores que utilizan esta tecnología, generan un archivo al final del día con todas las reservaciones que se generaron para un cierto hotel, para entonces, transmitir al hotel este archivo para ser decodificado. En este caso hemos seleccionado a The Mark Travel Corporation como ejemplo para esta tecnología.

2.3.1. The Mark Travel Corporation

Para este Tour Operador, también cualquier persona puede abrir un navegador y abrir <http://www.marktravel.com/vacation/index.asp> para poder reservar sus próximas vacaciones. El cliente puede seleccionar un paquete, o únicamente el hotel destino, donde si decide realizar la reservación, deberá llenar la información básica de su reservación, como es su nombre, además de los nombres de todos los pasajeros, destino, fecha de salida, fecha de regreso, numero de pasajeros, tipo de habitación, tipo de tarifa. Este Operador va almacenando todos los movimientos generados durante el día, para que al momento del corte, generen un archivo basado en sus propias especificaciones, el cual contiene todas las reservaciones generadas durante ese día.

2.4. Web Site (Extranet)

Algunos operadores presentan sus reservaciones en portales o intranet. El hotel debe tener un usuario y password que le den acceso a este portal. Una vez dentro de este portal, el hotel podrá hacer diferentes búsquedas para sus reservaciones, estas dependerán de las opciones que cada Tour Operador ofrece. El hotel deberá entrar por lo menos una vez al día para ver los movimientos realizados, y capturarlos en su *SGP*.

2.4.1. Thomson UK

Un ejemplo para estos operadores seria Thomson UK, donde un usuario puede ir a una agencia de viajes, o ir directamente a <http://www.thomson.co.uk> y realizar su reservación. Una vez que esta reservación se confirma al cliente, esta se va a la base de datos central, donde estará disponible en el portal para que el hotel pueda recuperarla.

2.5. Uso de la base de datos

Para el almacenamiento de las reservaciones utilizaremos una base de datos MySQL, donde tendremos dos tablas principales para las reservaciones, la tabla para las reservaciones, y una tabla para los pasajeros de una reservación. En la tabla de las reservaciones, tendremos la información básica de una reservación, como por ejemplo, el número de reservación, la fecha de generación, procedencia, fecha de llegada y de salida,

numero de pasajeros, tipo de tarifa, tipo de habitación, y comentarios entre otros. Y en la tabla de los nombres, tendremos todos los nombres y apellidos de los pasajeros asociados a un número de reservación. También tendremos catálogos para los hoteles, donde tendremos el nombre del hotel, un código asociado a este, dirección, ciudad, y país entre otros.

2.6. Descripción del sistema

En base a lo mencionado anteriormente, se pretende crear un sistema que estará compuesto por cuatro principales módulos independientes, con el propósito de ir aumentando mas para la decodificación de otros Tour Operadores, y para la generación de diferentes formatos para otros SGP. Este sistema estará corriendo bajo Microsoft Windows XP, y se instalara una base de datos MySQL donde almacenaremos las reservaciones. A continuación se describe el funcionamiento de los cuatro módulos principales, además de el sistema que utilizaremos como simulador de estos.

2.6.1. OTA (Web Service)

Este modulo se encargara de procesar las reservaciones provenientes con formato OTA. Una vez que se llama este Web Service, este decodificara la reservación proveniente, y la insertara a una tabla de transmisiones en la base de datos, donde un trigger, verificara por la existencia de esta reservación en la tabla principal de las reservaciones para en caso de no existir esta, entonces insertarla en la tabla principal. Para el funcionamiento, se

necesitara configurar el IIS (Internet Information Server) de Microsoft con las extensiones .Net Framework correspondientes para la ejecución de este.

2.6.2. Cheap Caribbean (Correo Electrónico)

El modulo para la decodificación de los correos electrónicos estará dividido en dos aplicaciones desarrolladas en Java donde la tarea de la primer aplicación será comprobar la existencia de nuevos correos electrónicos, para que en caso de existir nuevos, bajarlos a archivos de texto en un directorio pendiente para procesar, y la segunda aplicación estará decodificando los nuevos archivos pendientes, e insertando las reservaciones a la tabla de transmisiones en la base de datos, para que el mismo trigger verifique la existencia de esta reservación, y la inserte en caso de no existir. Para el funcionamiento de este método, se requiere crear una tarea en Microsoft Schedule Tasks ejecutándose cada hora, para que mande llamar la aplicación principal.

2.6.3. The Mark Travel Corporation (Flat Files)

Los archivos provenientes de este Tour Operador, serán procesados por este modulo, el cual estará desarrollado en Microsoft Visual Basic .Net, y básicamente se estará ejecutando automáticamente una vez al día, buscando en el directorio de entradas los archivos de los hoteles con reservaciones en el sistema, para procesar estos archivos basándonos en las especificaciones que utiliza este Tour Operador para su generación, y

después insertarlas en la tabla de transmisiones en la base de datos, para que el trigger verifique si esta reservación deberá ser insertada o no.

2.6.4. Thomson (Portal Web)

Para la decodificación de estos sistemas, se desarrollara un modulo donde hará llamadas http al portal, donde entrara con el usuario y password del hotel para primero buscar todas las reservaciones pendientes, y luego hará llamadas http para cada reservación individual con el propósito de ver los detalles de las reservaciones y estos insertarlos en la tabla de transmisiones en la base de datos, para que el trigger verifique si esta reservación deberá ser insertada o no.

2.6.5. Simulador

Para poder demostrar el funcionamiento de este sistema, se requiere de crear un simulador de los cuatro métodos mencionados anteriormente, donde el desarrollo de este sistema será en Microsoft Visual Basic .Net, y generara las reservaciones buscando estas en la base de datos principal de nuestro sistema. Este simulador generara un archivo XML basándose en las especificaciones OTA, un archivo Flat File basándose en las especificaciones de The Mark Travel Corporation, generara un email basándose en el formato utilizado por Cheap Caribbean, y generara un Sitio Web con las mismas características que utiliza el Tour Operador Thomson.

2.7. Conclusiones

Para el funcionamiento de este sistema, se necesitaran de cuatro módulos independientes, donde cada uno de ellos estará procesando información dependiendo del origen, y compartiendo una misma base de datos, donde estaremos almacenando las reservaciones provenientes de diferentes portales. Para el propósito de esta tesis, se implementaran únicamente cuatro módulos, representando las cuatro principales tecnologías que utilizan los Tour Operadores, dejando la posibilidad de ir aumentando mas módulos conforme se requiera de más Tour Operadores.