Universidade de Aveiro
2010

Departamento de Electrónica, Telecomunicações e Informática

Cesário Pinho Lucas

Sistema de Informação de suporte ao E-Turismo numa Região

E-Tourism Information System for a Specific Region

Universidade de Aveiro
2010

Departamento de Electrónica, Telecomunicações e Informática

Cesário Pinho Lucas

Sistema de Informação de suporte ao E-Turismo numa Região

E-Tourism Information System for a Specific Region

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia de Computadores e Telemática (M.I.E.C.T.), realizada sob a orientação científica do Professor Dr. Osvaldo Manuel da Rocha Pacheco, Professor Auxiliar do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

Dedico este trabalho aos meus pais, à minha irmã, ao Henrique e à Dália.

o júri

presidente

Prof. Doutora Maria Beatriz Alves Sousa Santos
Professora Associada do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

Prof. Doutor João Agostinho Batista de Lacerda Pavão
Professor Auxiliar do Departamento de Engenharias da Universidade de Trás-os-Montes e Alto Douro

Prof. Doutor Osvaldo Manuel da Rocha Pacheco
Professor Auxiliar do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

agradecimentos

Gostaria de agradecer às pessoas que nunca me ajudaram, pois foram essas mesmas que me deram ainda mais vontade para continuar a lutar e vencer.

palavras-chave

E-Business; Empacotamento Dinâmico; Sistemas de Gestão de Destinos Turísticos; Web Services; Mineração de Dados

Resumo

No Turismo e concretamente para os Destinos Turísticos, o desenvolvimento de uma estratégia de E-Business permite reduzir a dependência de intermediários para a distribuição de produtos turísticos. Neste âmbito, o conceito de sistemas de gestão de destinos tem sido utilizado nos últimos anos para descrever a infra-estrutura em termos de tecnologias de informação e comunicação de uma organização de gestão de destinos.

Por sua vez, a integração da Internet, Turismo e das Tecnologias de Informação e Comunicação associadas ao comércio provocaram alterações no comportamento e na atitude do turista, uma vez que vieram permitir aos mesmos construírem a sua própria viagem, à sua medida, abandonando os pacotes tradicionais

Tendo isto em vista, um Sistema de Informação de Suporte ao E-Business no turismo de uma região deverá permitir duma forma concisa divulgar informação turística relativa a uma dada área de interesse. Efectuar reservas e contratualizar actividades tais como alojamento, transportes e animação devem ser temas centrais. Permitir que o e-turista elabore o seu próprio pacote dinâmico com diferentes componentes à escolha e com apresentação do preço em tempo real é uma mais-valia preciosa.

No âmbito deste trabalho, pretende-se numa primeira fase estudar os conceitos ligados às diversas áreas de interesse e enquadrar o projecto nestas áreas. Numa segunda fase, é esperada a modelação dum sistema que responda às questões colocadas acima de modo integrado e que permita conhecer as bases de um sistema inovador de suporte ao E-Turismo. A implementação dum protótipo exploratório deverá validar os conceitos desenvolvidos.

keywords

E-Business; Dynamic Packaging; Destination Managment System; Web Services; Data Minig

abstract

In Tourism and specifically for Tourism Destinations, the development of an E-Business strategy allows the reduction of the intermediary dependency for the distribution of tourism products. Following this, the concept of Destination Management System has been used to describe an infrastructure in terms of information and communication technology for a Destination Management Organization.

The integration of Internet, Tourism and Information and Communication Technologies associated with business brought changes in the behavior and attitude of tourists, thanks to the possibility for them to create their own trip, based on their likings, in contrast with the traditional static packages.

Having this in mind, an E-Business Information System for supporting Tourism in a region should be able to provide touristic information about a select area of interest. Making reservations and booking activities such as lodging, transports and cultural events should be fundamental procedures. Allowing the e-tourist to build his own dynamic package with different components while the price is automatically updated is a valuable bonus.

In a first phase of this project, a general study of concepts linked to the various areas of interest is expected along with the insertion of these concepts in the general objectives. In a second phase, the modulation of a system that responds to these objectives in an innovative and functional manner is highly desired. The implementation of an exploratory prototype should enable the validation of the concepts developed.

Contents

1- Introduction 1
1.1- Motivation and Context 1
1.2- Objectives 2
1.3- Dissertation Structure 3
2- State of the Art 7
2.1- The Internet 7
2.1.1- Evolution of the Internet 7
2.1.2- Internet Future Expectations 8
2.2- Travel and Tourism 9
2.2.1- Travel and Tourism Expectations 11
2.3- Supply Chains and Suppliers 12
2.3.1- Intermediaries 14
2.3.2 Consumers 16
2.4- E-Business 17
2.4.1- Traditional Client-Server Architecture in E-Business 20
2.5- E-Tourism, a Subset of E-Business 22
2.5.1- The E-Tourist Profile 23
3- Supporting Technologies and Tools 25
3.1 – Overview 25
3.2 – HyperText Markup Language 25
3.3- XML 26
3.4- SOAP 28
3.5- Web Services 30
3.5.1-WSDL 31
3.5.2-UDDI 32
3.6- Enterprise Service Bus 32
4- OpenTravel 33
4.1- OpenTravel Alliance 33
4.2- OpenTravel Specification 36
4.2.1- OpenTravel Specification Life Cycle 38
4.3- OpenTravel Schema 39
4.3.1- Schema-based XML Example 40

5- System Architecture 47
5.1 – Multi-tier Architecture 47
5.1.1 – The Hybrid Architecture 49
5.2 – System Packages 55
5.2.1- System Coordination Use-Case 55
5.2.2- Interface Use-Case 57
5.2.3- Service Use-Case 58
5.2.4 – Bus Use-Case 59
5.2.5 – External Web Services Use-Case 61
5.3- System Components 62
6 - System Operation and Concept Validation 67
6.1- System Operation 67
6.1.1- Login and Logout 68
6.1.2- Query 68
6.1.3- Purchase Confirmation 69
6.2- Concept Validation 71
6.2.1- Development Environment 71
6.2.2- Development Procedure and Accomplishments 72
6.3- System Response Time 74
7 – Final Statements and Future Work 75
7.1- Final Statements 75
7.2- Future Work 77

List of Figures

Figure 2.1- Internet Usage In The World 8
Figure 2.2- World and U.S.A. Travel Expenditure 11
Figure 2.3- Traditional Supply Chain Architecture 13
Figure 2.4- Tourism Supply Chain 14
Figure 2.5- Business-to-Consumer Process 18
Figure 2.6- Tourism Supply Chain Links 19
Figure 2.7- E-Business Architecture 21

Figure 3.1- Simple XML Example 27
Figure 3.2- Simple XML Schema Example 277
Figure 3.3- SOAP Operation Scenario 28
Figure 3.4- SOAP Message Structure 29
Figure 3.5- Web Service Usage Diagram 30
Figure 3.6- Enterprise Service Bus Diagram 32

Figure 4.1- OpenTravel Organizational Structure 34
Figure 4.2- OpenTravel Specification Life Cycle 38
Figure 4.3- OpenTravel Schema Architecture 39
Figure 4.4- OTA_PingRQ.xsd (abbreviated) 41
Figure 4.5- Possible XML Ping Request 42
Figure 4.6- Possible XML Ping Response 42
Figure 4.7- OTA_AirFlifoRQ.xsd (abbreviated) 43
Figure 4.8- XML file of a possible OTA Air Flight Information Response 44
Figure 4.9- XML file of a possible OTA Air Flight Information Request 45

Figure 5.1- Basic 3-Tier Architecture Diagram 47
Figure 5.2- Hybrid Architecture Diagram 49
Figure 5.3- Hybrid Architecture Diagram with Bus Tier Replication 52
Figure 5.4- Round-trip and Processing Time Calculation 54
Figure 5.5- General System Package Diagram 55
Figure 5.6- System Coordination Use-Case Diagram 56
Figure 5.7- Interface Use-Case Diagram 58
Figure 5.8- Service Use-Case Diagram 59
Figure 5.9- Bus Use-Case Diagram 60

Figure 5.10- Web Service Use-Case Diagram 61
Figure 5.11- Main System Class Diagram 62
Figure 5.12- Reservation Class Diagram 63
Figure 5.13- Cruise Reservation Class Diagram 65

Figure 6.1- Visual System Diagram 70
Figure 6.2- Simple Java Class for Web Service Implementation 73

List of Tables

Table 2.5- Types of Packaging 23
Table 4.1- Members of the OpenTravel Alliance (not complete) 35
Table 4.2- Dynamic Package Type Component 37

List of Acronyms

ARPANET……...Advanced Research Projects Agency Network

B2B……...…Business-to-Business

B2C…….…… Business-to-Consumer

BI.....Business Intelligence

CRM…….…....Customer relationship management

CRS……...…Computer Reservation system

CT…….…Collaboration Technologies

D.S.L…….…….…...Digital Subscriber Line

DCOM……. Distributed Component Object Model

DMO…….....….Destination Management Organization

DMS…….……...….Destination Management System

DTD…….. Document Type Definitions

ESB……Enterprise Service Bus

GDP…….……Gross Domestic Product

GDS……...…...Global Distribution System

HTML……...HyperText Markup Language

I.S.P…….……..Internet Service Provider

IDE……Integrated Development Environment

IO…….....…Interoperability Committee

JRMP…….Java Remote Method Protocol

KM…….….…..Knowledge Management

LTO……...Local Tourism organization

M.I.T…….Massachusetts Institute of Technology

NTO……..National Tourism Organization

ORPC……Object Remote Procedure Call

OTA……..Open Travel Alliance

PMS……...Property Management System

POS……...Point of Sale

RMI……Remote Method Invocation

RTO……..Regional Tourism Organization

RTT……...……....Round-Trip Time

SC……..Supply Chain

SCM…….….Supply Chain Management

SCP……Supply Chain Planning

SGML……Standard Generalized Markup Language

SOA…….….Service Oriented Architecture

TSC……...……Tourism Supply Chain

UDDI……Universal Description, Discovery and Integration

URL……..…Uniform Resource Locator

W3C…….World Wide Web Consortium

WSDL……...Web Services Description Language

WWW…..……..……World Wide Web

XML…….eXtensible Markup Language

XSD……. XML Schema Definition

Glossary

Accommodation capacity- measure of accommodation stock at a defined destination. May be given by various different measures: e.g. number of establishments; number of main units within an establishment (e.g. rooms, caravan stances); capacity in terms of residents (e.g. bed spaces)

Agent- one who acts or has the power to act as the representative of another. Most frequently in travel anyone other than a principal, such as a retail travel agent, receiving agent, ticket agent, local operator or wholesaler

Airline Fare- price charged for an airline ticket. Some of the categories are as follows: advance purchase excursion (APEX): heavily discounted excursion fare available on many international routes. Reservations and payment will be required well in advance of departure, with varying penalizes for cancellation; excursion: individual fares that require a round-trip within time limits, discounted from coach fare, limited availability; group: discounts from regular fares for groups; and regular or normal: any unrestricted fare

Business Travel- travel for commercial rather than leisure purposes. Business travel is sometimes used as a cover-all to include what are sometimes referred to as the "MICE" markets – meetings, incentives, conferences and exhibitions

Chain of distribution- the means by which products (package holidays in this instance) are distributed from producers (principals) to consumers (tourists), often via wholesalers and retailers (tour operators and travel agents).

Company- Any entity engaging in business, such as a proprietorship, partnership or corporation

Computer reservation systems- Computerized Reservation Systems used for inventory management by airlines, hotels and other facilities. CRSs can allow direct access through terminals for intermediaries to check availability, make reservations and print tickets

Confirmed Reservation- an oral or written agreement by a supplier that he has received and will honor a reservation. Oral confirmations have no legal weight. Even written or telegraphed confirmations have specified or implied limitations. e.g.: a hotel not honoring a reservation after 6 pm., unless late arrival has been guaranteed in some manner

Consortium- a loosely knit group of independently owned and managed companies such as travel agencies, tour operators, hotels, or other suppliers, with a joint marketing distribution process

Destination- the place to which a traveler is going. In the travel industry, any city, area, or country which can be marketed as a single entity for tourists

Development Environment- the set of processes and programming tools used to create the program or software product

Distribution- the process employed to provide customers access to the product. For travel products distribution focuses largely on the ways in which the customer can reserve or purchase the product

Domestic tourism- travel within the country of residence

Dynamic Packaging- The act or process of creating a tour product to meet client needs in real time using a web-based software program or "dynamic packaging engine

E-Business- application of communication technologies over the Internet in support of all the activities of a given business, not only buying and selling but also servicing customers and collaborating with business partners. Business's data processing systems are also more efficiently linked together in order to better satisfy the needs and expectations of their clients

e-Commerce- Internet facilitated commerce, using electronic means for promoting, selling, distributing, and servicing product

E-Marketing- the act of trying to achieve marketing objectives through electronic or digital means. E-Marketing campaigns are performed through the use of internet-based content, such as newsletters and emails

End User- Person or organization that actually uses a product, as opposed to the person or organization that authorizes, orders, procures, or pays for it

E-Tourism- application of Information and Communication Technologies in the Tourism sector. Electronic Tourists can book hotels, tours, flights and other services on the Internet

GDP- The total market value of all final goods and services produced in a country in a given year, equal to total consumer, investment and government spending, plus the value of exports, minus the value of imports

Information systems- systems that use information technology to capture, transmit, store, retrieve, manipulate, or display information

Intermediary- an organization within the chain of distribution whose function is to facilitate the supply of a given product from producers to consumers. In the travel industry examples are travel agencies and tourism information offices

Leisure travel- travel undertaken for pleasure and unrelated to paid work time

Open Source- describes a program whose source code is made available for use or modification as users or other developers see fit

Organization- a deliberate arrangement of people to achieve a particular purpose

Package- pre-arranged combination of elements such as air, hotel, sightseeing, and social events put together and sold at an all-inclusive package price; to package, meaning to combine elements as above into an all-inclusive package product

Reservation Systems- computerized systems leased to travel agencies offering airline, hotel, car rental and selected tour availability and bookings. Systems are affiliated with major carriers and feature flight schedules of the sponsoring and other carriers, plus additional travel products

Service Provider- a person or company that supplies a particular service

SOAP- simple XML-based protocol to let applications exchange information over HTTP. SOAP stands for Simple Object Access Protocol and is language and platform independent and a W3C recommendation

Suppliers- individuals, companies or other organizations which provide goods or services to a recognizable customer, consumer, client or supply chain

Sustainable tourism- according to the World Tourism Organization, this is "envisaged as leading to management of all resources in such a way that economic, social and aesthetic needs can be fulfilled with maintaining cultural integrity, essential ecological processes, biological diversity, and life support systems"

Three-tier application- a program that is organized into three major parts: the workstation or presentation interface; the business logic; and the database and related programming. Each of these is distributed to one or more separate places on a network

Tour- any pre-arranged journey to one or more destinations

Tour Operator- develops, markets and operates group travel programs that provide a complete travel experience for one price and includes transportation (airline, rail, and/or ship), accommodations, sightseeing, selected meals and an escort. Tour operators market directly to the consumer, through travel agents and are beginning to be listed on computerized reservation systems

Tourism Industry- a group of businesses that provide services and facilities for consumption by tourists

Tourism Product- different things to the various members of the tourism industry. To the hotel it is "guest- nights". To the airline it is the "seats flown" and the "passenger miles". To the museum, art gallery or archaeological site, the product is measured in terms of the number of visitors. For the tourist the product is the complete experience resulting from the package tour or travel facility purchased, from the time they leave home until their return

Tourism- the all-embracing term for the movement of people to destinations away from their place of residence for any reason other than following an occupation, remunerated from within the country visited, for a period of 24 hours or more

Tourist- someone who travels out of their usual environment for less than one year. They travel either to another country or place, for leisure, recreation, or business purposes

Travel Agent - an individual who arranges travel for individuals or groups. Travel agents may be generalists or specialists (cruises, adventure travel, conventions and meetings.) The agents receive a 10 to 15% commission from accommodations, transportation companies and attractions for coordinating the booking of travel. They typically coordinate travel for their customers at the same or lower cost than if the customer booked the travel on his/her own.

User Interface- everything designed into an information device with which a human being may interact -- including display screen, keyboard, mouse, light pen, the appearance of a desktop, illuminated characters, help messages, and how an application program or a Web site invites interaction and responds to it

Visitors- a broader category than 'tourist', includes tourists and same-day visitors

1- Introduction

1.1- Motivation and Context

In the past years, there has been an increasing interest in the usage of technology to achieve a better distribution of the Tourism Information available to the average user [1]. Having this in mind, and with the gradual evolution of the Internet, the necessary conditions were met to allow an explosion of information circulating on the World Wide Web (WWW). With all the information in existence, there have been several different approaches to representing it internally, distributing it and finally presenting it to the general public. Different programming languages have been used, different types of database structures, different interface typologies, web services, etc, but the final objective has always been the same, the representation of information regarding Tourism of any given destination.

The junction of several of these different approaches ends up giving a software developer a solution to the global problem of Tourism Information representation. This solution is often known as a Destination Management System (DMS). Unfortunately, all of the existing DMS's have been developed in an isolated form, depending on the size, type or even global interests of the company or organization in question, and have very little correlation between already existing systems. These isolated solutions raise new problems that are more complex than ever in a totally different way.

Imagine a tourist wants to visit a specific destination and is interested in incurring in several different activities and visits during his stay. In order to be able to prepare for these activities, the tourism might have to visit several different websites to get to know the activities available in the region and several other websites to make the reservations. This may lead to waste of precious time and some inconvenience for the tourist. The tourist might be asked to insert the same information several times, the payment methods might be different from system to system and repeated information will most certainly appear. If we add usernames

and passwords that must be remembered along with specific website names and supported company services and products, this can lead the tourist to utter confusion and frustration…

1.2- Objectives

In Tourism, and more specifically for Tourism Destinations, the development of an e-business strategy allows the decrease of mediator dependency for the distribution of tourism products. DMS's have been used in recent years and describe the technological information and communication infrastructure of a Destination Management Organization (DMO).

For a DMO to entirely fulfill its' duty by the usage of a DMS, the DMS should be able to offer a large amount of information regarding a specific destination in the easiest way possible. A DMS should also allow a seamless method for using reservation and contracting services. These services can belong to a very wide range of areas, going from accommodation and transports to animation and local attractions.

Nowadays, the characteristics shown above are already available in some if not all of the DMSs in existence. The true innovation that is expected with this Dissertation is intertwined with the concept of Dynamic Packaging. Dynamic Packaging is understood as the combination of different components of a trip bundled together and whose price is calculated in real time. This dynamic package is created in response to the tourists' own choices and can originate from several different supplier sources. This way, the DMS project that will be conceived should have, along with all the other characteristics already mentioned, the capability of allowing the tourist to enter an enhanced experience where he is free to make his own choices of all the available services a destination has to offer. The integration of Internet, Tourism, and Information and Communication Technologies in one single bundle adds to this in the way that the tourist (client in the commercial point of view) can build his own personalized visit, depending on his own tastes and wishes. Once again this is a very unique experience compared to the traditional one given by ordinary travel agencies.

As a secondary objective for this project, after studying and defining all the concepts behind the modulation of an E-Tourism Information System with support for dynamic packaging, it is also expected that the requirements for this system be made. These requirements should later enable a prototype to be built in order to validate the ideas developed along the way. Polling and automatic learning of the preferences of tourists are other interesting areas that can later be

exploited and easily integrated to the system in order to improve the tourists´ experience during all the phases of travel. Although these ideas will be less studied and worked upon, support for them will exist in the E-Tourism Information System. It is also important to mention that the objectives presented here have been accordingly thought to be used in a small to medium-sized region or destination.

To sum things up, imagine it was possible to have everything you ever needed in one place and that it was easy to access in the sense that anyone could use it, young or old, fat or thin, tall or short. Now transport this idea to the Tourism Industry and add the Internet and we have the main elements necessary to begin understanding the needs that are behind the work that will be developed during the course of this Dissertation.

1.3- Dissertation Structure

This Dissertation is divided into seven different chapters having in account the general ideas present in each. Each chapter was created for addressing specific needs based on one main concept. The names and content of each of the chapters of this dissertation are as follows:

Chapter 1- Introduction

This Chapter is focused on giving a global vision of this project. The context in which this project is inserted is explained in detail along with the motivation that drives it. The main objectives and goals for the completion of this project are also enumerated and a small introduction to important concepts of Tourism is given. Finally, the main structure of this project is stated and summarized by chapters.

Chapter 2- State of The Art

In Chapter 2, a general description of basic information related to the project is made. Important topics that are crucial elements for the understanding of future concepts are detailed and explained in order for the reader to grasp the main idea. Some of the topics included are Travel, Tourism and Expectations for the growth of these sectors in the following years. Supply chains are also described with an emphasis on Tourism Supply Chains so that the entire process can be fully understood and strategies can be built upon this knowledge. Next, E-Business is explained, from the main concept to the more specific details involving it. E-Tourism is also mentioned as being a subset of E-Business looking towards the Tourism sector. Finally, the E-Tourist profile is referred to, showing the needs and trends of this type of cyber-tourist.

Chapter 3- Supporting Technologies and Protocols

Chapter 3 is reserved for the presentation of all the major technologies and protocols that were used for the accomplishment of the tasks included in this project. Scripting languages such as HTML and XML are referenced and some simple examples are shown. The SOAP protocol is also mentioned along with WSDL and UDDI. Together, these protocols and languages allow for the usage of Web Services in a totally platform-independent manner, enabling standardization and interoperability. In the end of this chapter, the Enterprise Service Bus is exploited. This software architecture for middleware gives shape to the E-Tourism Information System and joins together all of the technologies and protocols mentioned before into one simple and unified tier of abstraction.

Chapter 4- OpenTravel

In Chapter 4, the OpenTravel Alliance is discussed. First off, a description of the Alliance is made. The origins of the Alliance along with the OpenTravel Organizational structure are described. Also, the main elements of the OpenTravel Alliance staff, including work groups and project teams are enumerated and the main tasks they are in charge of are also mentioned. Also included in Chapter 4 are the names of key members of the OpenTravel Alliance and their importance to the organization. After this, the OpenTravel specification is explained. Major functionalities, such as reservation booking or product availability are mentioned. The OpenTravel specification life-cycle is announced and the way versions of the specification are created and deployed is also explained. Next, the OpenTravel Schemas are introduced as being a representation of a model for describing the structure of information based on a set of rules previously agreed upon by the OpenTravel Committee. At the end of this chapter, examples of relevant schemas based on the specification are given along with possible XML files that are based on the schemas chosen. The examples are simple in nature but fully functional and represent the guidelines needed for the creation of the E-Tourism Information System.

Chapter 5- System Architecture

This Chapter includes most of the modulation work performed for this project. It is in this Chapter that the E-Tourism Information System takes shape. The architecture of the system is explained in detail, having as a starting point the multi-tier architecture used in many other systems already in use. Next, the Hybrid Architecture is announced and concepts, innovations and ideas are clarified. The four different tiers composing the system using the Hybrid Architecture are also fully examined. After this, the system packages are shown. The system coordination package, the interface package, the service package, the

bus package and the external web services package are the five packages identified and explained. These packages are closely tied to the Hybrid Architecture in use. Various use-case diagrams were added along the way in order to better explain the relations that exist between the actors and the operations that these perform. Lastly in this chapter, system components are described. These components are the key elements for the system. A simple reservation can be composed of one of various components such as a vehicle component, package component, air component, cruise component, global reservation component, golf component, hotel component and package component. In the case of system components, class diagrams were used for better describing their structure.

Chapter 6- System Operation and Concept Validation

In Chapter 6, system operation is discussed by referring to the main procedures that are executed by the system. The most important procedures such as performing login, logout, a query or a purchase confirmation are thoroughly commented. Next, the concepts that were suggested earlier on are validated. This validation occurs through the usage of an experimental prototype of the E-Tourism System. The development environment used for the creation of the prototype is described along with the accomplishments that were made. Development procedures were also included to give a general idea of what was done. At the end of this chapter, the concept of response time is explained. This represents the average time the system takes to perform a given task.

Chapter 7- Final Statements and Future Work

Chapter 7 is the last chapter in this document. This chapter serves as a conclusion to the work developed during the course of this project. A brief description of what was done and in which way is included along with the methodologies used to achieve the end result. Possible future work that can be developed in order to enhance the quality of the project is also suggested.

2- State of the Art

2.1- The Internet

As of June 2009, about 1.67 billion people worldwide where using the Internet on a regular basis [2]. The Internet is widely visited by a large sum of people for many different reasons. These reasons can be work, fun, communicating, pleasure, time consumption or particularly, and in our case most importantly, for obtaining information. Let's not forget that information is the key element of Tourism. But not always has the Internet been this well-know, widely distributed nor as accessible as it is today to the common user. These are the main reasons that allow it to be explored in several new frontiers and integrated with Tourism in new and exciting ways.

2.1.1- Evolution of the Internet

The Internet has been around since the early 1960's but initially it did not have the shape nor the functionalities and purposes it has today. The Internet has come a long way. It has been and still is in constant evolution.

In 1962, as a military effort, some people found a great potential in the sharing of information between computers. Joseph Carl Licklider of M.I.T. was the first person to suggest a global computer network [3]. Leonard Kleinrock, also of M.I.T and later U.C.L.A., developed the mathematical theory of packet switching networks [4]. Later Lawrence Roberts was able to connect two computers together using the developed theory over dial-up telephone lines [5]. Joining these ideas together, the building blocks for the development of the Internet, called ARPANET in those days, where ready to be put in practice.

Early Internet was used mainly by computer technicians, informatics experts and a small handful of fanatics. It was not at all straightforward or user-friendly. This has changed in recent years. Nowadays, almost anybody can access and use the

Internet having only basic computer skills. There are many ways to access the Internet. Among Internet Service Providers (I.S.P.) and Digital Subscriber Lines (D.S.L.), the simplest of all is by just connecting to a network in any given location where access is available. These networks with limited coverage are often called hotspots and are available in public locations such as airports, shopping centers or hotel lobbies by means of wireless technology. At home or on the road, anyone, including a potential tourist, is able to be constantly wired to the network and connected to the Internet.

2.1.2- Internet Future Expectations

The Internet has been growing exponentially for the last two decades and is doubling in size every year. All around the world the penetration rate of the Internet is rising and it is estimated that by the year 2015, 50% of the world will be connected to the Internet [6][29]. These are amazing values and show the importance that Internet will have in future generations to come.

As we can see in figure 2.1, Internet usage has been increasing rapidly and is expected to continue on to 2011. We can also see that a great amount of the traffic that circulates on the Internet is related to information and personal services. It is thought that by 2011, this amount will have exceeded 25% of global traffic. This shows once again that the Internet can be seen as a powerful tool and with great potential if used correctly, namely in E-Business.

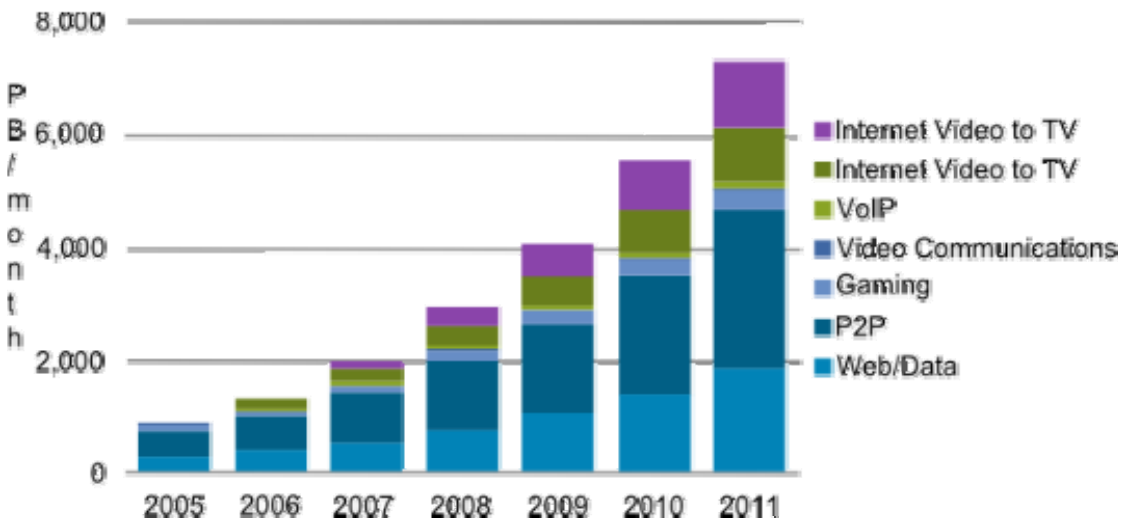

Figure 2.1- Internet Usage in The World
(Source: http://homepages.cwi.nl/~troncy/Talks/2009-03-06-mozcamp)

2.2- Travel and Tourism

In the eyes of the World Tourism Organization, Tourism can be seen as the act of "people who travel to and stay in places outside their usual environment for more than twenty-four hours and not more than one consecutive year for leisure, business and other purposes not related to the exercise of an activity remunerated from within the place visited [24]". But taking into account all the different factors that are involved and related to Tourism and Travel, it is much more than just this.

The Travel and Tourism Industry can be divided into five different sectors [25]. These sectors are:

- **Accommodation**
- **Food and Beverage Services**
- **Recreation and Entertainment**
- **Transportation**
- **Travel Services**

Accommodation includes all the services that are related to lodging a person in a facility that provides shelter and protection. These facilities can be hotels, guest houses, motels, hostels, bed & breakfast and all other types of housing offers that exist and fit the needs of travelers.

Food and Beverage Services, as the name indicates, include all the areas dedicated in providing the tourist with nourishment such as food and drinks during his stay. The different Food and Beverage Services can allow the tourist to experience the gastronomy of the region being visited and can be a way of better understanding the local culture and customs.

Recreation and Entertainment can be seen as a complement to a visit to a destination. During a visit, it is normal to look for alterative things to do to fill in the gaps that exist between the main events of the visit. For example, if a visitor intends to visit Algarve, in Portugal, his main objective might be sightseeing but for recreation and entertainment be might want to try snorkeling for the first time or attending the Seafood Festival. Participating in these events is normally not planned ahead. These are more of "impulse events" where a person has knowledge of an activity going on and reacts to it. Social activities, local parties, exhibits, museums, nightclubs and many other events also belong to the Recreation and Entertainment sector of Tourism.

Transportation is related to providing a person with the necessary means to move around from one place to another. If in addition a company is able to make the trip as pleasurable as possible, this makes the travel experience increase in value and

satisfaction for the client. Depending on the place you are visiting there are several different means of transportation such as air (such as an airplane flight, helicopter tours), ground (bus rides, taxis, train, metro) and water (including ferry boats, cruises, yacht trips and many others).

Finally Travel Services can be seen as one of the broadest sectors of the Travel and Tourism Industry. Travel Services include everything that a person might need in terms of organizational concepts and services. Travel planning and conference organizing companies along with associations and agencies, either private or public, play a key role in this sector. Travel Trade also falls into this sector being responsible for the "packaging, booking and ticketing of travel products", usually provided by the travel companies [10].

The junction of these five sectors is so broad and includes so many other sectors, organizations and people that Travel and Tourism can be truly understood as a global industry.

Depending on the tourist's destination, Tourism can be classified as Domestic or International [28]. Domestic Tourism is all the tourism that is practiced inside the tourist's country of origin. An example of this is a Portuguese Tourist visiting the South Coast of Portugal. International Tourism, on the other hand, is all the other tourism that is practiced in countries that are not the same as the country of origin of the tourist. An example of this is a Portuguese Tourist traveling to the United States in order to visit New York.

International Tourism itself can be divided into inbound or outbound depending if the traveler is entering the country, or leaving it to visit another country respectively [26][27]. These terms are often used for statistics. Domestic Tourism does not use these terms because the visitor does not enter nor does he exit a country, he only moves inside one were he already lives.

Inbound Tourism has many advantages for the county it is performed in, the most important of course being income. Other advantages include a possible steady number of tourists all year round eliminating "low seasons", a market that is less dependent of local visitors, increased sales due to foreign visitors usually spending more money than locals and providing cultural exchange for everyone. The disadvantages are less likely to happen except when the intense tourist activity ends up altering or endangering the local environment as the case of Machu Picchu [30]. Lack of a sustainable growth plan or the superior interests of the population are some of the reasons this might happen.

Another way of classifying Travel and Tourism is based on its purpose. A trip to a certain place can be seen, depending on its purpose, as Leisure Travel or Business Travel [26]. These are the two main types of Travel and Tourism that exist and because of their nature, they have different characteristics.

Leisure Travel is focused on creating an experience for the tourist that involves having fun and discovering new places and things usually during a holiday season or vacation. Since Leisure Travel is for someone who is traveling for pleasure (either alone or accompanied by a friend of even several family members), it is expected that the experience be more casual and relaxed. The well-know R&R (Rest and Relaxation) is the goal of the trip.

When it comes to Business Travel, there are many reasons that justify a trip. People travel on business because of sales conferences, business presentations, meetings, training courses and even product launches. Theses trips tend to last less time than Leisure trips and usually the traveler goes alone or in a small group with work colleagues. These tourists usually carry little baggage and are more formal in the way they act and dress.

In this project, the DMS that is intended to be developed should have the capability to attract both leisure tourists and business tourists. This adds a greater value to the final result because it increases the portion of market that is captured in the process.

2.2.1- Travel and Tourism Expectations

Since the 1950's up until 2005, International Tourism had been growing at an average annual rate of more than 6%. This translated into a growth from 25 million travelers to more than 806 million travelers [32].

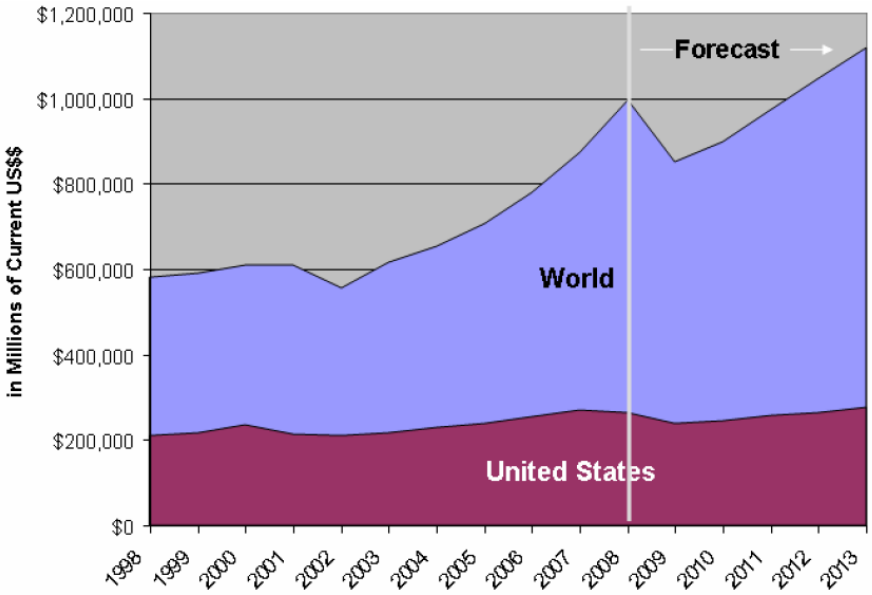

Figure 2.2- World and U.S.A. Travel Expenditure
(Source: http://www.egencia.com/docs/global_business_travel_forecast_2010.pdf)

But due to the global crisis that has been affecting the majority of developed countries and also many countries in development for the past couple of years, there was a sudden inversion of growth in the second half of 2008. By 2009, international tourist arrivals where decreasing an average of 4% to roughly 880 million travelers [31].

Despite what has been said, in 2008 International Tourism generated close to 1 Trillion US dollars in earnings worldwide [33]. This is by itself an impressive number. This value adds up to more than 9% of the global Gross Domestic Product (GDP) employing about 220 million people worldwide [34]. And by 2020, the number of international travelers is expected to be higher than 1.5 billion individuals per year [32]. This gives a general idea of how important and influent the Tourism Industry is. This shows that the sector is alive and recovering rapidly from the crisis with perspectives of growth in the near future. This is another reason that gives purpose to the work I am developing in this project.

2.3- Supply Chains and Suppliers

With so many visitors traveling during the course of a year and such a large demand, one could wonder where the offer comes from. The answer lies in suppliers. In general, suppliers are the people or companies that provide services or goods to a chain that ultimately produces a final product or service for the customer. Suppliers often have little or no contact with the consumer. On the other hand, they usually sign long-term contracts with intermediaries in order to maintain a steady work flow. Intermediaries are the entities that establish a connection between the suppliers and the final consumer.

For goods or services to get to the final consumer, a Supply Chain (SC) must be established [36]. A SC is created by virtually linking different entities together through the relations they have between each other. A traditional SC architecture normally has one or more suppliers, manufactures, distributors, retailers and end-customers [37]. The product or service "travels" through the chain until it is sold to the customer. The item can also travel in the opposite direction of the chain in some special situations such as an item return or if the product has a defect.

Depending on the type of SC that is necessary for a given sector, there might or might not be a place for all the entities mentioned. Sometimes a single company or organization can take on the job of two entities at once. This eliminates intermediaries and the final cost can decrease thanks to less expenditure on transportation, handing or other tasks that involve external services or fees.

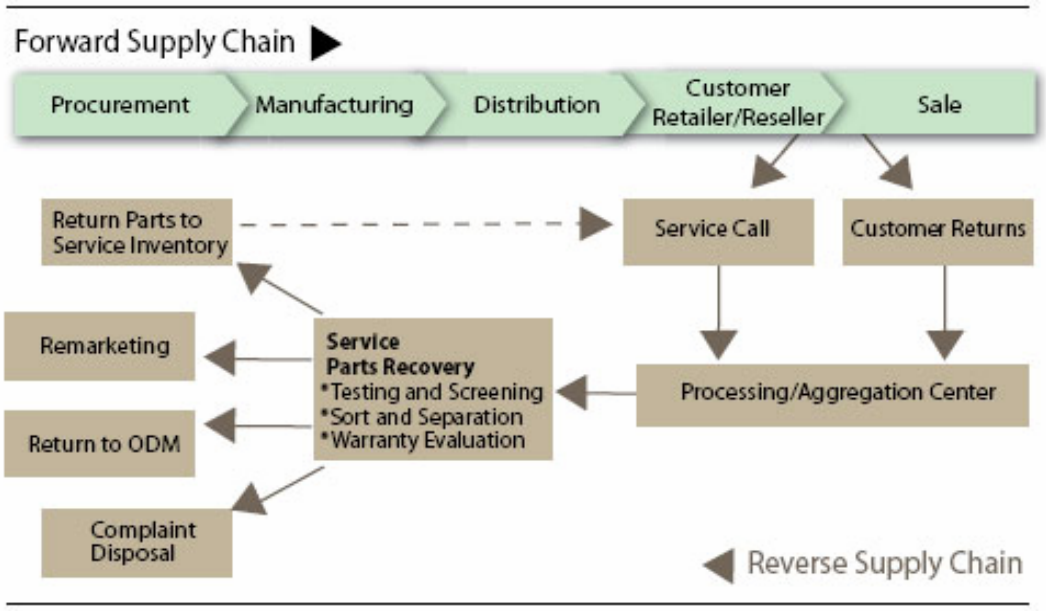

Figure 2.3- Traditional Supply Chain Architecture
(Source: http://www.rlmagazine.com/edition16p12.php)

Taking into account the particular features that the Tourism Industry has to offer, it makes lots of sense adapting the traditional SC architecture to Tourism. Having this in mind, a Tourism Supply Chain (TSC) can emerge. The main difference between a TSC and a general SC is that while traditional supply chain customers usually buy a product, tourists travel to the product they buy [19]. This induces a strong service component to the chain. Although this might sound more complex, it allows a simpler supply chain structure to be used. From the traditional SC architecture we move on to one that, when used along with E-Business, can lower costs and increase reliability. This is a key element for the success of this project.

When talking about TSC's, there is also a great concern in creating a TSC that is sustainable [21]. Sustainability can be seen from two different points of view. The first one is related to the reliability of suppliers and players that are part of the chain. It is important to build a reliable chain, with alternative suppliers and service providers, making it as "strong" as possible. This allows, in the event of a failure from one of the elements, for products to still be delivered independently of the failure that might have occurred. Some strategies like creating long-term partnerships with suppliers, having a consistent volume of operations or even fair-pricing can help achieve this goal. The second point of view on a sustainable TSC is related to environmental issues of a region. This type of sustainability is more critical in developing regions where Tourism is practiced [20]. The local community must be integrated and aware of the changes due to Tourism and should also benefit from them [22]. The TSC must be thought in a way that the impact on a specific region is the least possible while still being able to create wealth and employment for the

local. This type of sustainability is far broader than explained here but not as relevant to the project. Despite this, as we will see later, TSC will be very useful and play a key role in determining the architecture that will be used for the development of the DMS.

2.3.1- Intermediaries

Intermediaries, as we have seen, can be defined as the entities that establish a connection between the suppliers and the final consumer. Specifically in Tourism, intermediaries are the organizations, companies, resources or people that connect the primary service suppliers, airlines and other transportation to the tourist that is going to consume the product [40].

Tourism evolves around consumption of airline fares, services, visits, transportation tickets or in short, services. To get these services and especially the information to the tourist there has been a great advance in the creation of custom-made organizations, strategies and even technological solutions [38][39].

The solution in getting Tourism services from the primary supplier to the tourist often depends on the types of intermediaries and on the service we are handling. The most evident way of connecting suppliers to tourists is directly. Although this sounds simple, this rarely takes place due to lack of information from the tourist and lack of knowledge from the supplier. In alternative, when we talk about a primary service such as a hotel or conference booking, several different paths can be taken.

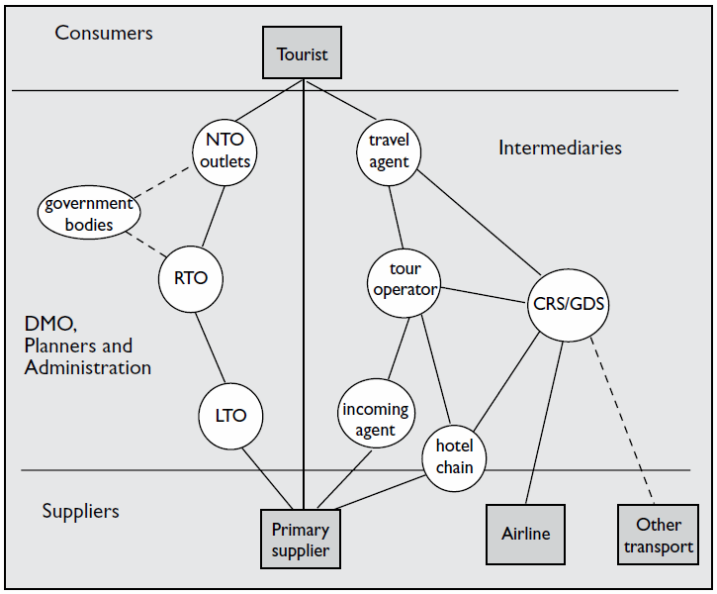

Figure 2.4- Tourism Supply Chain
(Source:E-Tourism and Commerce [1])

Tourism Organizations (TO) can be used as powerful tools by tourists to obtain information and make bookings online. There are different kinds of TO's depending on there scope. Normally these can be divided into three groups, Local TO's (LTO), Region TO's (RTO) and National TO's (NTO). As their names indicate, each type of organization operates at a different level in society. It is also known that people look for information and help regarding tourism operations in different ways based on the different TO's available [41]. Broader information about a certain destination or spot is more likely to appear in a higher level organization such as a NTO [42]. Knowing this, people usually look for information related to history, culture and geography in NTO's. As for RTO's, they often offer more services and information related to tourism bureaus. Tourists use these for location identifying, obtaining map information, getting to know regional products and gastronomy and also integrating tourism programs that might exist. Finally, LTO's have a bigger emphasis on local activities like special events taking place, shopping and restaurants and night-clubs that exist in the surroundings. Although distinction between types of TO's exist, many times it is difficult to truly separate them and tell which type one belongs to because of the diversity of information and services it might have and provide.

Another way of a tourist being able to book a service is through a travel agent. A travel agent is commonly known for helping the traveler in deciding with travel option is best suited for him. After this, the travel agent will make the necessary arrangements for the traveler to pay for the service and go on his way. For the traveler this is quite transparent and simple. But behind the travel agent usually lies many other intermediaries before he can access the supplier [43]. Tour operators are often used by travel agents to obtain products they otherwise would not have available. The main function of tour operators is to combine travel components and services in order to create a travel package that will usually have a reduced price compared to buying the different components separately.

Yet another way of travel agents connecting to suppliers is through Computer Reservation Systems (CRS) and Global Distribution Systems (GDS). A CRS is traditionally used by airlines and other transport companies but can be used by other entities as well [23]. A CRS is a system that allows the control of a company's inventory and ticketing actions. It holds the companies fares, reservations, ticket records and updated schedules. All this information can usually be accessed online although some level of protection exists to avoid non-authorized personal accessing the system. A GDS, on the other hand, can be seen as a switch that connects to several CRS's. These connections enable travel providers and travel agencies to communicate through a platform in a quick and efficient way, interfacing different CRS's. There are four main GDS's in existence, Amadeus, Galileo, Sabre and WorldSpan [44]. Amadeus is the biggest of the four currently having 35% of the global market. It is most used in Europe and Asia. The second

largest is Sabre with 25% of the global market and is most used in the United States. The other two GDS's account for the remaining 40% of the global market. All of these GDS's where formed thanks to partnerships between airline companies and other companies related and with an interest in Tourism. In short, these systems are of very important relevance being used not only by travel agents but also tour operators, hotel chains, rent-a-car operators and many other entities.

To conclude, the term distribution is often heard about in Tourism and when intermediaries are spoken of, but it is a concept that might not be evident at a first glance. Distribution, with the aid of intermediaries, makes a lot of sense when products are involved. If an item is produced it is only logical that it should be transported to the place where it will be sold. But if we think about this in terms of services in Tourism, distribution doesn't quite fit. There is nothing to be transported or physically delivered to the tourist. For example, it doesn't make sense to transport plane tickets or cruises to the tourist. Although this is true, distribution in Tourism is even more important and critical than in other sectors. Therefore, distribution must be seen with different eyes. Since Tourism is such a particular sector, distribution for its services has suffered adaptations. Intermediaries have played an important role in these adaptations and new entities have been created in order to face these challenges. Some of the technologies above mentioned are included in the solutions found since the main item being transported in Tourism is information and this is easily accomplished with the help of computers.

2.3.2 Consumers

The consumer might as well be the most important player in a SC. He is the one that buys the products or services available and he sets the trends. A consumer is known by many names. He can be called consumer, buyer, customer, client, purchaser or many other names. For the interest of Tourism, the consumer is called tourist. A tourist is a person that is practicing any form of tourism. The service that is being consumed has been bought or acquired before his journey to the given destination.

In order to captivate a potential tourist, some things must be taken into account. First of all, Tourism customers will be buying a service, not a good. A service is performed and a tourist earns the right to use it at a certain time and place. Tourists must also travel to the place where the service will be performed. This makes Tourism something that is intangible [26]. It can not be seen, only experienced and this usually only happens after it has been paid for.

Therefore, when travel agencies or other promoters think about selling Tourism services of any sort to a potential buyer, from simple airfares to full holiday

packages, they must focus mainly on the people who are most likely going to buy their services [45]. They should also design their services in a matter that meet the needs of the buyer and should then find an effective form of letting the target market know of the existence of the same service. Pricing is also very important and sometimes crucial not only for travel agents but even entire countries. Many nations worldwide are reliant on tourism as a main income for their economies [46].

The ideas here presented about consumers are intended to focus on the traditional form of Tourism, where potential tourists visit a TO or a Travel Agency and purchase a service. I will explain in a bit more detail later on in this chapter the concept of a Tourist as a consumer, namely the E-Tourist profile, which is related closely to this project.

2.4- E-Business

E-business is a general concept that is heard of very frequently. One thing is for sure, it is not limited to the interfaces that the common user often sees. In most cases, this portion is actually the only one the client ever sees. There is a lot more to E-business than just that.

E-Business is a term used to describe all business processes that are made electronically, usually with support of the Internet. The addition of the Internet to the Information and Communication Technologies allowed the rise of E-Business. As Daniel Amor of Hewlett Packard would say, "E-Business is about using the convenience, availability and world-wide reach to enhance existing businesses or creating new virtual business [18]."

E-Business is made up of many components. It is important to refer that not all DMSs use the existing components of E-Business in the same way. This is one of the reasons why DMSs are usually custom-made for a given company. Each company has different needs. Some of the most important components of E-Business and relevant to this project are E-Commerce, Customer Relationship Management (CRM), Knowledge Management (KM), Collaboration Technologies(CT) and Supply Chain Management (SCM) [11].

E-Commerce is closely related to E-Business. E-Business can be thought as the structure behind E-Commerce and E-Commerce can be seen as the propeller of E-Business. E-Commerce represents the trading of goods and services over the Internet and also includes a part of Electronic Marketing. There are several different types of E-Commerce. These types can be classified having in mind the role played by the entities in the transactions. The two main types of E-Commerce

relevant to this project are Business-to-Business (B2B) and Business-to-Consumer (B2C).

B2B E-Commerce can be understood as the trading of goods between different companies (businesses). These trades don't involve the individual consumer. This form of E-Commerce is the largest in size, involving almost 94% of all E-Commerce transactions and large sums of money, towering trillions of dollars every year [12][13].

B2C is the second largest form of E-Commerce and involves the trading of goods between businesses and the final consumer [14]. This is the normal procedure that is made by a consumer (customer) that uses the Internet for buying products of all sorts, including tourism packages, from companies that are represented on-line.

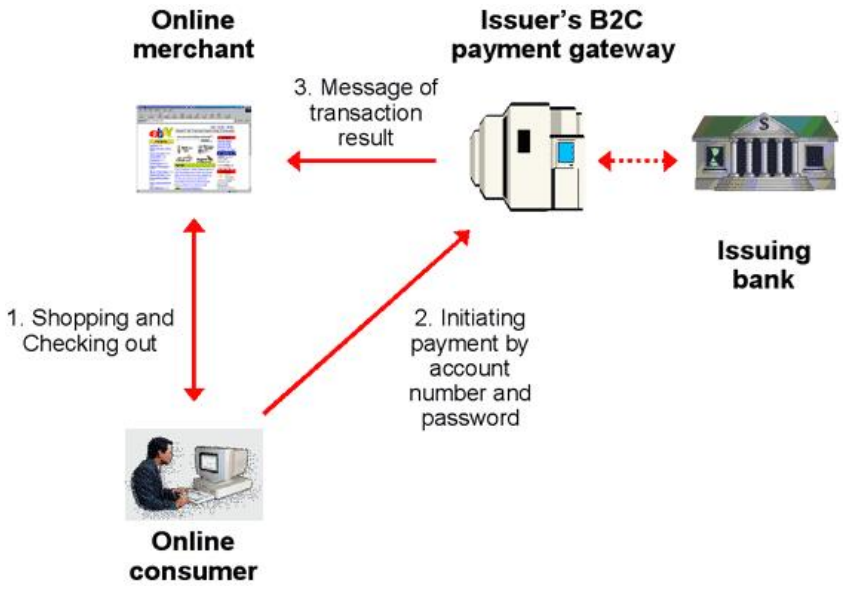

Figure 2.5- Business-to-Consumer Process
(Source: http://www.gtnews.com/article/6547/figure1.gif)

Looking back on E-Business components, Knowledge Management is another term that mustn't be forgotten due to its key role. Knowledge, which is commonly understood as information in Business, is a fundamental part of Tourism. In E-Business, products can only be consumed if first information about them is viewed. The information provided about a product must always be up-to-date and reflect the correct quantities available. The capability of a company using information correctly and to its benefit is usually the difference between success and failure. The ability to organize and manage information is even more important when we talk about systems that need to retrieve and present information regarding several different individual sources, as is the case of the DMS that is proposed to be developed during this Dissertation.

Another E-Business component is Customer Relationship Management. CRM can be seen as the usage of all the necessary tools and methodologies to control and organize customer relations [16]. CRM is somewhat a challenge in E-Business for one simple reason. There is a dramatic change in traditional methods that are used with the new "electronic customer". From the individualized and formal deal that is made in the presence of a representative (in the Tourism Industry it usually takes place in a Travel Agency), to a public and virtual interface that is intended to be used by a wide range of individuals, many changes must be made. CRM is also related to identifying consumer trends, preferences and trying to increase customer satisfaction. This can be achieved by using polling and automatic learning abilities. These actions have already been mentioned before as being valuable and highly wanted in a high-end E-Business solution.

SCM is used in planning and managing all the steps that involve making available the final product that will be offered to the end-consumer [17]. But before the product can be available to the consumer, it must first make its way from supplier to manufacturer to retailer to distributor. This flow is often the source of problems because even if only one actor fails, the entire flow will stop in its tracks. SCM, just as CRM before, must be revised and adapted from its original form to fit the needs of a DMS for Tourism. Since the concept of tourism-oriented SCM is a lot broader because it involves so many different areas, more people are necessary along with more coordination to obtain the final destination product. In the end, the main goal is to integrate different business processes in order to carry information, products, services and funds through the supply chain reducing time, effort and costs [35]. This will later allow a better retain price of the final product or service to the customer.

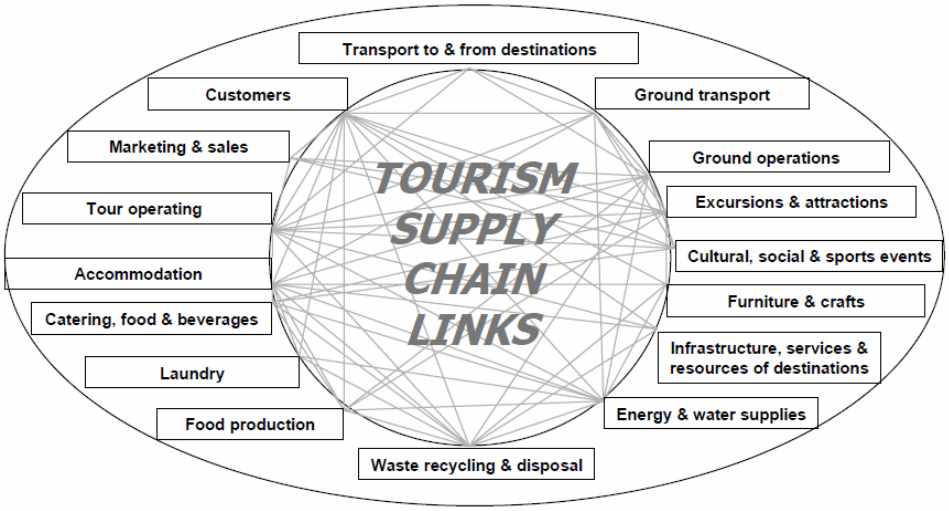

Figure 2.6- Tourism Supply Chain Links
(Source: http://www.icrtourism.org/documents/TourismSupplyChainsfinalreport31January2004.pdf)

Finally, looking back on E-Business components, CT is all about using technology to enable better communication, coordination and cooperation in a company [15]. By extending this concept, we are able to benefit from the collaboration of different companies that work together to ensure more notability, increased sales, and ultimately, added features and better prices for the final consumer. Instead of working against each other, companies work with each other, growing equally and leaving other competition behind. Collaboration has been used in many areas other than Tourism, but has brought success to almost all.

Thanks to all of the components mentioned earlier, E-Business offers a great support to the creation of systems that interconnect different companies together and allow smoother communication between them. It also allows buying and selling orders to take place in a much easier way. Companies can now automatically ask other companies for new products and refill their stocks with greater ease. This makes the whole process of fabrication, availableness and demand a lot better. There are many other advantages to the use of E-Business as an enterprise solution:

- **Reduction of Logistic Costs:** this can be accomplished by improving shipping and receiving, reducing paperwork, auditing freight costs, etc.
- **Increase of Overall Efficiency**: this can be achieved by automating processes, centralizing information and information consistency
- **Increases Target Clients and the Company's Potential to grow:** if a company only has local visibility it is natural that is will only have local clients
- **Decreases Production Time:** less paperwork, less processes to follow, less time wasted
- **Increases Marketing by exploring new frontiers:** having a company represented internationally though E-Business is half way to success

E-Business is a win-win situation in many aspects. But although this is true, a great amount of care must be present when developing an E-Business solution and all factors and company needs must be taken in account. If the job is done correctly, the advantages are overwhelming.

2.4.1- Traditional Client-Server Architecture in E-Business

It isn't a surprise that E-Business has had such a wide acceptance in the past few years. Investors dedicate time and money in information technology and systems because they provide economic value to an entity and help lower costs and save money [47]. It is seen more as an investment that will eventually pay for itself than

as an expense. Having also in consideration that E-Business is a fairly easy solution to implement, its' continuing growth is natural.

In a simple Client-Server architecture oriented for the communication between a business and the consumer, the consumer (client) uses an interface, usually a web browser to send requests to a web commerce server. The web commerce server then establishes a connection with a database server and usually a payment gateway that leads to another server, this time oriented for executing payments by connecting directly to bank systems. When the database server processes the consumer's request, it sends back the reply to the web commerce server. The web commerce server then terminates the connection and after that sends the request with the information back to the consumer so he can read it [48].

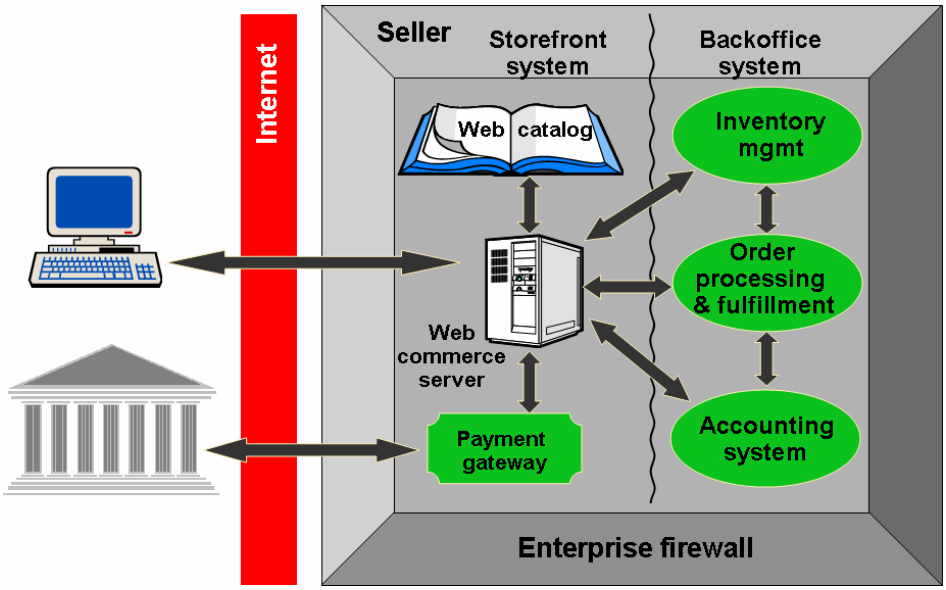

Figure 2.7 - E-Business Architecture
(Source: http://www.educause.edu/Resources/EBusinessArchitectureOverviewa/155188)

As we can see, using such a system is very attractive due to its simplicity to the common Internet user. It is also very convenient since the consumer can, in doubt, request more information about the service he is buying and feels no pressure in making a decision. Payment is also not a problem since there are usually alternative payment methods and all insure enhanced security for the customers' sensitive data.

This ideology will be the starting-point for the implementation of the desired E-Tourism Information System that this project demands.

2.5- E-Tourism, a Subset of E-Business

Just like E-Business, the integration of Information and Communication Technologies with Tourism gave way to E-Tourism. The idea of E-Tourism is very similar to E-Business. The only difference is that the type of DMS necessary to display and sell services is specifically orientated to be used by the Tourism Industry. During the general explanation of E-Business most of the general factors of E-Tourism were analyzed and commented. Just to complete some thoughts it is important to refer the concept of Dynamic Packaging.

To fully understand Dynamic Packaging, one must first understand how a tourist buys all the different services and items he needs. Considering this is done online, a tourist must visit several different sites to purchase different services. A tourist might have to visit a rent-a-car website for renting a car, a hotel booking website for room reservation, an airline website for plane tickets and so on. This is very time consuming and posts a problem when all these different sites have different credentials for logging in the same customer. Payment information must also be replicated to several different websites by the user, hoping the user has at least one available payment method for each website. These problems appear due to the inexistence of neither interaction nor connection between these different websites. Packaging tries to eliminate some of these problems thanks to the combination of different travel components in one single place [7]. These components are viewed as a single item and the price is presented in real-time and also unique.

There are basically three different types of Packaging. The first is Static Packaging. This Packaging is the simplest and less flexible. Although this is true, it is still a big improvement compared to not having packaging at all. Static Packaging allows tourists to pick packages built up of several components that have been joined together. The package cannot be dismembered nor has any options for customization but can be paid as only one item and in one place. Semi-Dynamic Packaging is one step ahead allowing customers to add or eliminate one or more components to a package already in existence. It is important to refer that the price of the total package alters in function of the components that are added or eliminated but the price itself is not dynamic because each component has a price that individually doesn't change [50]. Dynamic Packaging is the ultimate experience where a customer has total flexibility in choosing the components he desires. The innovation also resides in the way the price is calculated. Along with depending of the components that he selected, the price is also affected by rules that might exist between different components [8][49]. If for example, a tourist buys a plane ticket and reserves a hotel belonging to the same company, he will be able to enjoy a discount for this.

Types of Packaging	Static Package	Semi-Dynamic Package	Dynamic Package
Component	Fixed	Changeable	Changeable
Component Rules	Individual, Simple	Individual, Simple	Interdependent, Complex
Component Price	Fixed	Fixed	Dynamic depending on rules
Package Price	Fixed	Changes	Changes

Table 2.5- Types of Packaging
(Adapted from: http://aiglobal.org/....DynamicPackaging09Nov2006.pdf [50])

In short, tourists are now able to choose more destinations, hotels, flight options, car rentals and other tourism services with great ease, personalization and with better prices. Dynamic Packaging has brought a level of flexibility and choice to the tourist greater than ever before.

2.5.1- The E-Tourist Profile

An E-Tourist can be seen as a fusion between a Tourism Client and a normal E-Business Consumer. E-Tourists haven't yet been fully studied or analyzed as one would wish mainly due to the fact that E-Business and particularly E-Tourism is a relatively recent activity [42]. In one study, it is said that "The Web is changing the needs of consumers, who are increasingly less loyal, take more frequent vacations of shorter duration and take less time between choosing and consuming a tourism product"[1]. This may be true but one thing is certain, every E-Tourist is surely looking for a destination with quality and trip value with his purchase.

But the tourism product in not only related to business and marketing but also has a big role when it comes to personal experiences. Sometimes how someone is selling a service is as important as what is being sold. Some components of a trip might not be as attractive as others but if integrated with other activities or components they can show to be a success. This all depends on the experience that the tourist will engage in. Another study shows that "certain physical, social, cultural, technological, political and economic characteristics of a destination

develop an environment effect that directly influences tourist perceptions and experiences"[51] of the destination he visits.

One thing is for sure, in a more and more global world, where Internet is taking over along with more and more powerful computer technologies allied with Tourism, in a near future, the E-Tourist profile will fit a vast majority of the population.

3- Supporting Technologies and Tools

3.1 – Overview

Over the past few years there has been a substantial amount of technologies and virtual tools that have emerged and made possible the appearance of a new era for the IT industry as well as the Tourism Industry. With the passing of time, the use of IT technologies in Tourism has had a major impact on the way companies collect, process, store, manage and display information. With the addition of more and more novelties, new ideas have also emerged and systems have been engineered that are able to perform new tasks and provide users with a unique new experience. In this chapter, we will be addressing the technologies and tools that will be used for the development of the proposed E-Tourism Information System.

3.2 – HyperText Markup Language

HyperText Markup Language (HTML) is probably one of the oldest technologies being used in the context of this thesis proposal. Even so, HTML only goes back to the beginning of the 1990's, when it was officially created [71]. At that time, Tim Berners-Lee was busy creating a proposal for what later would be known as the World Wide Web (WWW) [72]. This proposal was for the creation of a hypertext document system that was to be used by the CERN community located in Switzerland. While this proposal was being made, along with the need for an identification scheme (Uniform Resource Locator addressing) and a transport mechanism (HyperText Transfer Protocol), the necessity for a formatting language also arose. HTML was born. Since then, HTML has suffered many updates to its

original format. HTML is the direct result of the Standard Generalized Markup Language (SGML). SGML is known to be a technical specification that is composed of well-defined vocabulary and syntax for defining markup languages for documents. Due to the complexity of SGML, HTML was derived from it to help users that were not familiar with the first standard to be able to publish the scientific documents and other technical information in an easier way.

HTML contains labels for elements and attributes along with grammar defining the hierarchy of the different elements that compose the language. Other features such as the capability of creating electronic links to documents (hyperlinks) helped HTML become the predominant markup language for web pages. As other markup languages, the HTML elements used are not shown but rather interpreted by the web browser in order to display the information in the intended way. The HTML standard is currently being developed by the World Wide Web Consortium (W3C) [70].

The use of HTML for this project is in great part due to its simplicity. Since it is basically composed of headers and content, there is very little where mistakes can be made. Creating a system that uses HTML also enables it to be easily ported online without much hassle as is the objective of this project. Considering HTML also allows implementation of requests and responses from text messages, it is the perfect choice for use with the OpenTravel specification. In addition to this, the option to use authentication and SSL/HTTPS encryption increases overall security of the system that is to be created.

3.3- XML

Extensible Markup Language, or just XML, is a markup language that was published by the W3C for documents containing structured information. XML became a W3C Recommendation on the 10th of February, 1998. Although XML is a markup language just like HTML, one does not substitute the other. Unlike HTML that was designed for displaying data, XML was designed for transporting and storing data in a simplistic way. Another difference is that HTML has a fixed tag set and semantics [77][78]. XML on the other, allows the user to define himself different tags and also the structure of the elements that compose the vocabulary of the XML file. In part, XML is much more flexible to use than HTML. Even so, in order to use XML, some rules must be obeyed literally. Some of the basic rules are that all XML elements must have an opening and a closing tag. These elements must also be correctly nested within each other. The user must also be aware that XML tags are case sensitive. Also important to have in mind is that every XML document must have a root element. This element is seen as a parent to all other

elements of the document. One last basic rule is that each attribute value must be correctly quoted to be recognized as such.

```
<?xml version="1.0" encoding="utf-8" ?>
<message>
        <to>You</to>
        <from>Me</from>
        <title>Secret</title>
        <text>You are great</text>

        <number>12</number>
        <price>Secret</price>
</message>
```

Figure 3.1- Simple XML Example

In order to define and validate the elements and structure of a XML document, two different methods can be used. These actions can been performed either by using Document Type Definitions (DTD) or with XML schemas. DTD define and validate the structure of a document by using a list of legal elements and attributes. One of the main advantages that DTD have over XML schemas is the DTD can be defined inside an XML document. This allows each XML file to carry a description of its own format with it enhancing security. There is also the possibility to use an external reference to the DTD just like with XML schemas. XML schemas are more important to this project that DTD because further ahead we will be using a specification that is defined using only XML schemas.

```
<?xml version="1.0" encoding="utf-8" ?>
   <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
              elementFormDefault="qualified" >

<xs:element name="message">
 <xs:complexType>

 <xs:sequence>
    <xs:element name="to" type="xs:string"/>
    <xs:element name="from" type="xs:string"/>
    <xs:element name="title" type="xs:string"/>
    <xs:element name="text" type="xs:string"/>
    <xs:element name="number" type="xs:int"/>
    <xs:element name="price" type="xs:float"/
  </xs:sequence>

</xs:complexType>
</xs:element>

</xs:schema>
```

Figure 3.2- Simple XML Schema Example

XML schemas are written in files with .XSD extension. XSD stands for XML Schema Definition. XSD is more recent than DTD and therefore is more complete. Since XSD is written in XML, there is no need to learn new syntax unlike DTD. More than just defining and validating the structure of XML tags, XSD also allows:

- defining data types of the different elements used, such as xs:boolean, xs:string, xs:byte, xs:date, xs:int, xs:float, xs:time
- creating specific and complex data types based on the user's needs from the already existing types
- defining the number of occurrences of different elements by using "minOccurs" and "maxOccurs"
- using inheritance between schemas
- specifying element contents as being unique

3.4- SOAP

SOAP provides a unified way for invoking remote procedures, activating applications and manipulating objects over different types of networks and platforms that might not be directly compatible with each other. SOAP stands for Simple Object Access Protocol and is completely platform-independent. SOAP is similar to Object Remote Procedure Call (ORPC) from the Distributed Component Object Model (DCOM) or to the Java Remote Method Protocol (JRMP) for Java Remote Method Invocation (RMI). In comparison to JRMP and ORPC, SOAP has an important advantage because it is a text-based protocol. This allows it to nest into XML files and perform remote tasks in an easier way. SOAP is also seen as a mere message-based protocol and therefore does not have a specific transport protocol associated to it. HTTP along with FTP, SMTP and other transport protocols can all be used with SOAP although HTTP is one of the preferred [79].

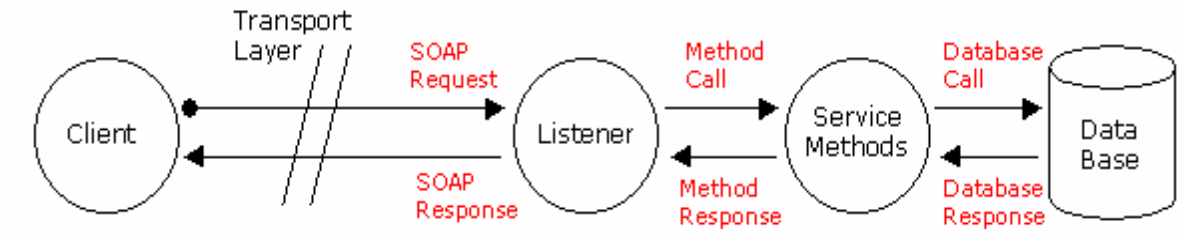

Figure 3.3- SOAP Operation Scenario
(Source: http://www.soapuser.com/basics1.html)

SOAP is a message protocol which uses request and response messages to communicate with services on the other end of the transport layer. SOAP requests are sent to a listener that is in charge of later performing an internal method call to an application or server that may be running in the background. This application accesses the database in order to retrieve the information that has been asked for. The database then sends the information to the application running on the server and afterwards the server executes a method to respond to the listener. When the listener receives the response, it resends it in a SOAP request message back to the client that first requested the information.

A SOAP message is usually composed of three distinct elements, a SOAP Envelope, a SOAP Header and a SOAP Body [80]. The SOAP Envelope is the root element of a SOAP message. This element is mandatory and must be unique. It is defined by the "ENV" namespace prefix and the "Envelope" element. Inside the SOAP Envelope can be the SOAP Header and the SOAP Body. The SOAP Header contains metadata about the SOAP message. This can include information about routing, security, data control or even digital signatures. The header element is not mandatory in SOAP messages although more than one header can be defined. Last but not least, the SOAP Body is a mandatory element that contains all the XML data that was defined by the application and is being exchanged with the SOAP message. The operation name wanting to be executed along with the necessary parameter names are all included inside the body. The SOAP Body as well as the SOAP Header are children of the SOAP Envelope element. As a side note, inside the SOAP Body, a SOAP Fault parameter can also be carried. This parameter is not mandatory but can be used in SOAP responses when errors occur during the message processing. Specific information is sent in this parameter about the error type, usually identified through a code number.

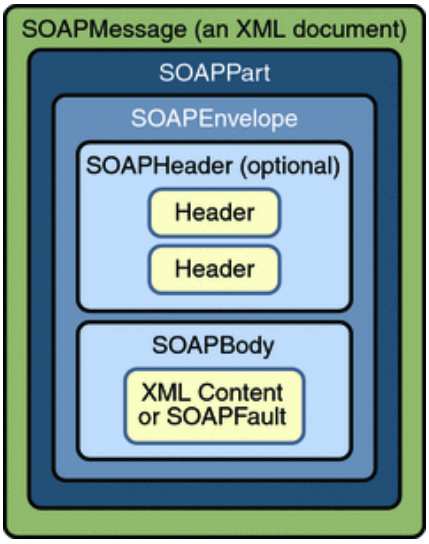

Figure 3.4- SOAP Message Structure

3.5- Web Services

Web Services are software application components available on the web that are able to exchange information with other applications that are also online. This means that web services act as units of code that can be remotely activated on request. In comparison to other technologies, web services are on the upper end because they support interoperability between different programming languages, operating systems and application platforms. This is thanks to the usage of open source protocols by the web services. Open source protocols encourage all major companies to join in on this solution. Since no single company has control over any open source technology or specification used, these can grow and mature based on global interests for the industry in general. XML, as seen earlier, is an open source language for storing and structuring data. In the case of web services, it is used to encode data that is intended to be sent and received by a web service. HTTP is also widely used with web services. HTTP allows the exchange of data over the transport layer in XML format. Another advantage of web services over others technologies is since it is a text-based solution due to XML, bypassing firewalls and other security measures is a lot easier.

Web Services are usually composed of three different platform elements:

- SOAP (Simple Object Access Protocol)
- WSDL (Web Services Description Language)
- UDDI (Universal Description, Discovery and Integration)

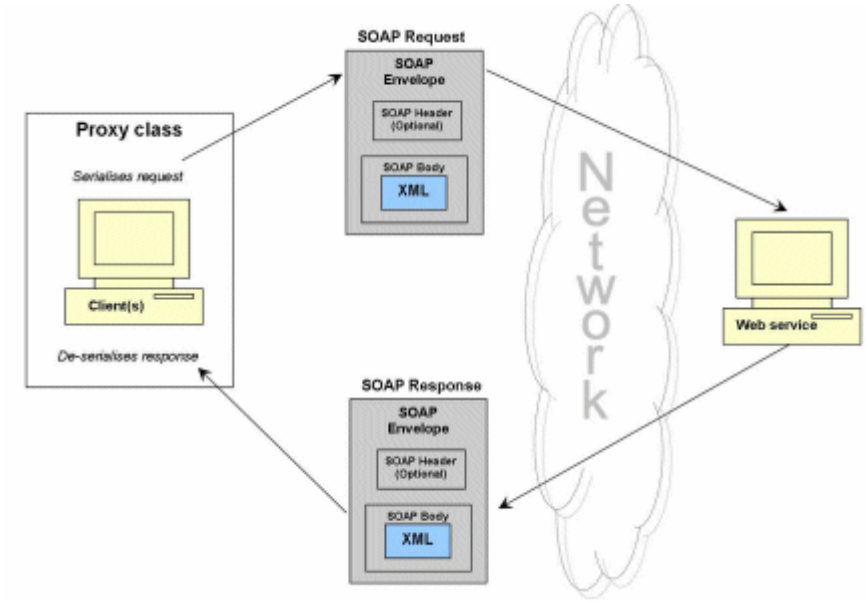

Figure 3.5- Web Service Usage Diagram

(Source: http://www.simple-talk.com/dotnet/performance/understanding-xml-web-services-for-testers)

SOAP is the first element that makes up a web service. As we have learned before, it is a very powerful tool for making remote method invocations. In figure 3.5, we can see a typical example of a call to a web service. First, a SOAP request is made by the client to the web service. The request is serialized and the message is encapsulated with a SOAP envelope. The web service then replies with a SOAP request where the information is again inside the SOAP envelope. When the client receives the response, it is de-serialized for viewing.

Next we will learn how WSDL and UDDI also play a part in the operation of web services.

3.5.1-WSDL

WSDL stands for Web Service Description Language. WSDL is used to describe the interface of a web service. Available operations and messages of a web service are described in an abstract way. This allows a user to create a SOAP message in a way that when the request is sent to the web service, it will recognize the syntax and respond accordingly. Details are not important, just what you can do with the web service. WSDL can also be used for locating web services.

Web Service Description Language is a bit controversial because it had versioning problems during its development. The version 1.1 of WSDL was not accredited by W3C. It was only seen as a W3C note. Later, version 1.2 was abandoned for not being considered viable. Then, only in 2002 was WSDL 2.0 seen as a W3C recommendation. This has lead the vast majority of web services to be based on version WSDL 1.1. Although WSDL 2.0 is more complete, modern and robust, programmers are not tempted to upgrade due to all the differences that exist between one version and the next.

WSDL is composed of four different elements [81]:

- Message– an abstract, typed definition of the data being communicated.
- Types– a container for data type definitions using some type system (such as XSD)
- Port Type–an abstract set of operations supported by one or more endpoints.
- Binding– a concrete protocol and data format specification for a particular port type.

WSDL will not be discussed in greater detail for it is not a crucial element in this project. Eventual files containing WSDL that might appear later on will not be created for this project but only used in it.

3.5.2-UDDI

UDDI stands for Universal Description, Discovery and Integration. UDDI is a registry which can be used to look for WSDL files. It stores information about existing web services made available by different entities. UDDI is to web services as the phone book is to residential phone numbers. Some problems UDDI solves is the incapability to expand market reach, remove obstacles for faster recognition on the web and the ability to describe web services in a simple and secure way.

3.6- Enterprise Service Bus

An Enterprise Service Bus (ESB) can neither be seen as a technology nor a tool. It is more of a software infrastructure that can be used between a client application and various different services. It acts as a middleware for implementing a service-oriented system. Complex details of specific services and programming languages are handled in the background of the ESB, creating a type of mask for the client application. Thanks to this architecture, interoperability between distinct software architectures and network protocols can be achieved and distributed applications can come together in a single location. This concept is fundamental for the development of this project. The ability to connect different web services that were created in completely different environments in a transparent tier is key. Along with interoperability, an ESB can perform many other functions. Distributing information across a domain in a secure and efficient way by ensuring message delivery and routing is also important in a wide-scale distributed system as the one intended to be modelled.

There are several different types of ESB's that can be used in the development of an enterprise system, either commercial or open source. OpenESB from Sun Microsystems, Mule from MuleSoft and ServiceMix from Apache are some of the most used and well-known open source ESB solutions available. As for commercial versions, JBOSS from Red Hat, Weblogic from Oracle and WebSphere from IBM are also worthy alternatives.

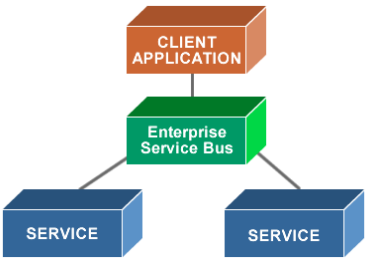

Figure 3.6- Enterprise Service Bus Diagram

4- OpenTravel

"OpenTravel provides a community where companies in the electronic distribution supply chain work together to create an accepted structure for electronic messages, enabling suppliers and distributors to speak the same interoperability language, trading partner to trading partner [63]".

4.1- OpenTravel Alliance

The OpenTravel Alliance is a non-profit organization that was founded in 1999 by various different travel companies. The main goal of this alliance was and still is to build a data transmission specification that is accessible to any tourism-related company interested in it. This specification should allow the exchange of electronic business information between the various elements of the travel industry in a seamless way [65]. In the first few years of operation, the main tasks of the alliance where based on structuring the organization and building a general strategy for it. The first specification was only released more than two years after the alliance was officially started. Nowadays, after building a solid base, tasks are more focused on finding new and existing necessities that must be added to the specification. Implementing these additions and ordering these by importance are also key tasks.

As the name indicates, OpenTravel makes its code and different information files available to the community for free, this means the code is open to everyone that is willing to use it. Any company interested in becoming an official member of the OpenTravel Alliance, must pay a small fee based on the company's annual revenue [66]. This fee is used to sponsor the alliance and allows the new member to be able to participate in activities such as influencing the structure of the specification by adding new ideas and proposals, gaining credibility in the tourism market and having access to privileged information about the specification and its implementation.

OpenTravel controls all operations, modifications and decisions that are made and that are related to the alliance. OpenTravel also controls the source code that is made available to the community. There is no company that solely influences the standard or runs the alliance. A very well-defined structure exists that is in charge of taking care of major issues. This structure is made up of different elements depending on the area of influence of each one [64]. In short, the Board of Directors is in charge of making strategic decisions and ensuring that all activities and projects are going according to plan. The Executive Director organizes meeting quorum requirements to make sure that the necessary number of directors and members of the Board of Directors are present at any given meeting. The Communications Committee, as the name indicates, is in charge of communicating with the exterior in order to make OpenTravel better known to the public, the tourism industry and potential members of the organization. It is in charge of all marketing activities of OpenTravel. The Interoperability Committee (IO) is responsible for reviewing and approving proposals from different teams that are working on different projects. The IO is also responsible for determining specific content for the specification, determining necessary documentation and distributing the specification for review to the members of the alliance and ultimately to the Board of Directors for final approval and publication. The IO Subcommittee is part of the IO and ensures the quality of the specification the IO reviews and approves. The specification is also compared to other industry data elements by the IO Subcommittee in order to make the specification as neutral and, in some cases, as interoperable with other specifications as possible. The Work Groups are presently divided into four sub-groups, Transport, Architecture, Hospitality and Travel Integration. Each one of these sub-groups is in charge of creating new schemas and modifying existing ones, relating these to each of the four areas of influence.

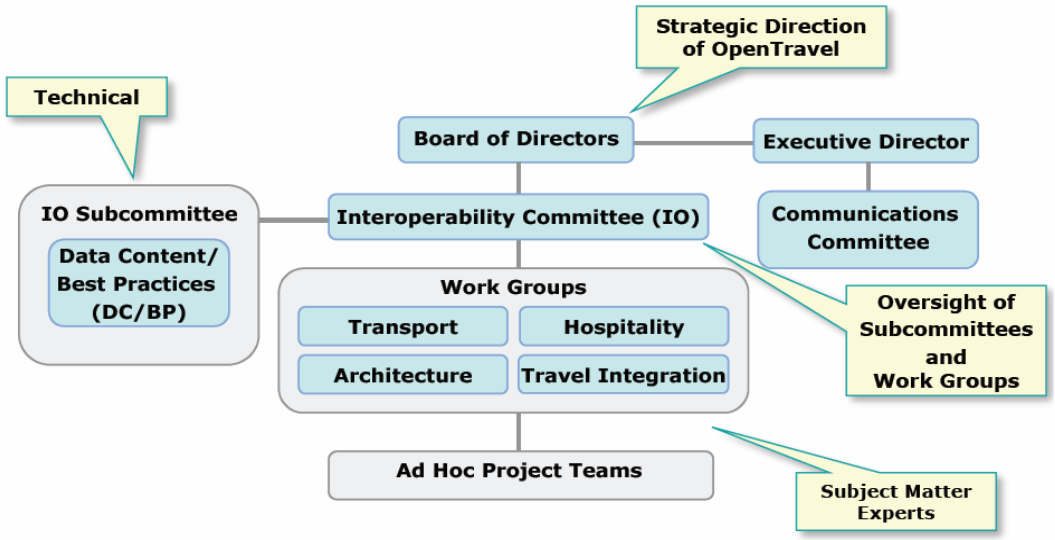

Figure 4.1- OpenTravel Organizational Structure
(Source: http://www.opentravel.org/AboutUs/Structure.aspx)

Finally the Ad Hoc Project Teams are periodically created to address specific problems or topics that might emerge. These teams are generally made up of members of the alliance that are seeking a solution to a specific problem that has not yet been addressed and also want to make their contribution to the specification. Their main objective is to create a proposal that later is integrated to one of the area-related sub-Work Groups. The proposal is later evaluated and if relevant, it is worked on by other committees until it is integrated to the current specification.

The OpenTravel Alliance as a whole is composed of many different independent members that are committed in giving their contribution to the organization. These members represent different companies located all over the world that are some how related to the tourism industry. Car rental firms, airline companies, cruise lines, hotel chains, leisure suppliers, tour operators, travel agencies, service providers and technology/solutions providers are all potential members of the OpenTravel Alliance. Currently OpenTravel has a wide range of members that include some of the most important companies in the tourism, technology, distribution and even educational industry [67]. Select representatives of each of these companies are chosen, based on elections, to occupy seats available in the alliance's structure. As an example, presently, the Board of Directors is composed of members from companies such as Avis Budget, Hotel Booking Solutions Inc., Continental Airlines, Best Western International, Marriot International Inc., Sabre, Amadeus and the Hertz Corporation. Currently, Valyn Perini is the only member of the Board of Directors that is exclusively working for the OpenTravel Alliance.

American Airlines	US Airways	Northrop Grumman
TRX	Aeroplan	OAG
Hewlett Packard	Amadeus	OpenJaw
ITA	ARINC	Sabre
Travelport	Datalex	United Airlines
Delta Airlines	EDS	Transat
Worldspan	IBM	Continental Airlines
TWA	British Airways	Accor
Hilton Corporation	AMTRAK	VisualForma
UTS	Microsoft	Rubicon
SITA	Lufthansa Systems	Northwest

Table 4.1- Members of the OpenTravel Alliance (not complete)
(Adapted from: http://www.opentravel.org/Resources/uploads/OpenTravel_OAG.pdf)

4.2- OpenTravel Specification

The OpenTravel Specification is the definition of a standard that is being used across the tourism industry for the harmonization of message exchange, registration of business processes and partnership trading based on common terms. The specification was built and is continuously being updated having in mind the needs of the members of OpenTravel. As mentioned earlier on, the first specification version was released in early 2001. The main objectives for this first specification were achieved by ensuring the openness of the standard, flexibility, the capacity for extensibility, security and platform independence. Recently, other objectives have tried to be met such as achieving interoperability with other systems, increasing simplicity of use, making greater the performance of the standard and most importantly increasing the acceptance of the specification on a global term [68]. As a note, the OpenTravel Specification is recognized and credited by the World Wide Web Consortium (W3C). W3C is an international community that ensures the long-term growth of the Web by developing protocols and guidelines to be followed by other consortiums, alliances and organizations [69].

The OpenTravel Specification itself is large in size. Currently, the OpenTravel Specification 2010A is made up of 272 ".xsd" files that define it. Each of these files is usually made up of several hundred lines of code that allow users to perform various different types of functions usually related to each other. There are also many optional data fields in each file. As for code errors, there are more than 850 in the 2010A code error list. These fields and codes can be used only when necessary in order to minimize implementation time and cost. Internal documentation can later be created to support the developed project and implementation.

Some of the main functionalities of the OpenTravel Specification are:

- Search & Product Availability
- Reservation Booking
- Reservation Modification
- Passenger/Client Check-in & Check-out
- Ticket Fulfilment
- Schedules
- Descriptive Information of Operation
- Booking Queue Management

- Dynamic Packaging (parcially)
- Message Retransmission
- Fare Search & Display
- Fare Availability
- Fare Pricing
- Error Management
- Seat Availability & Information

As seen above, the OpenTravel Specification is complete in many ways. There is a wide variety of solutions for reservation management of nearly all areas related with tourism. Car, hotel, cruise, air, golf, travel insurance, rail, tour and shorex are all possible to address with the specification. Other additions include the possibility to define a user profile, loyalty certificates for an e-tourist and the creation of dynamic packages. Unfortunately not all types of reservations are supported by the dynamic packaging model yet. Currently, air components, car components, hotel components and actual package option components are the only components that can be added to an OpenTravel Alliance (OTA) dynamic package. In the near future, it is expected that other components such as cruise component, insurance component and shorex component be added to the dynamic package type. Finally, error handling and message retransmission are also supported natively by the specification. These additional features as well as others not mentioned but included in the specification help in making a possible system based on it more complete, robust and ultimately secure.

Air Component	An air trip that is part of the dynamic package. This can include the flight reservation number and additional information such as flight seating, flight number and legs.
Car Component	A rental car that is part of the dynamic package. Used to provide Property Management System and/or Computer Reservation System identifiers.
Hotel Component	A hotel room stay that is part of the dynamic package. Specifies the action being requested and/or the status of the hotel component as well as a reference number for a query and the length of time starting at the create time for which the component will be held.
Package Option Component	An additional element that is part of the dynamic package. Used to provide PMS and/or CRS identifiers.

Table 4.2- Dynamic Package Type Component

4.2.1- OpenTravel Specification Life Cycle

Since the year 2001, the OpenTravel Specification has been constantly updated at a biannual rate [73]. There are different versions of the OpenTravel specification available on the OpenTravel website depending on the year and the semester of creation. These versions can be divided into three different types, current version, in-development version and past versions.

Past versions of the specification are versions that are completely finished and will not be changed in anyway in the future. These version types are available for download to anyone interested and can be seen as legacy versions. To become a past version, a given specification must first be an in-development version and later a current version. The current version on the other hand is the most recent version of the specification available for download and therefore is the most complete version as well. Any member of the OTA can submit comments for the current version of the specification. These comments, if relevant, may later influence the in-development version of the specification. Finally, the in-development version type is the prototype of the next available current version. The in-development version of the specification is a "work-in-progress" version where comments of the current version are being added to enhancements and modifications made by the members of the OTA.

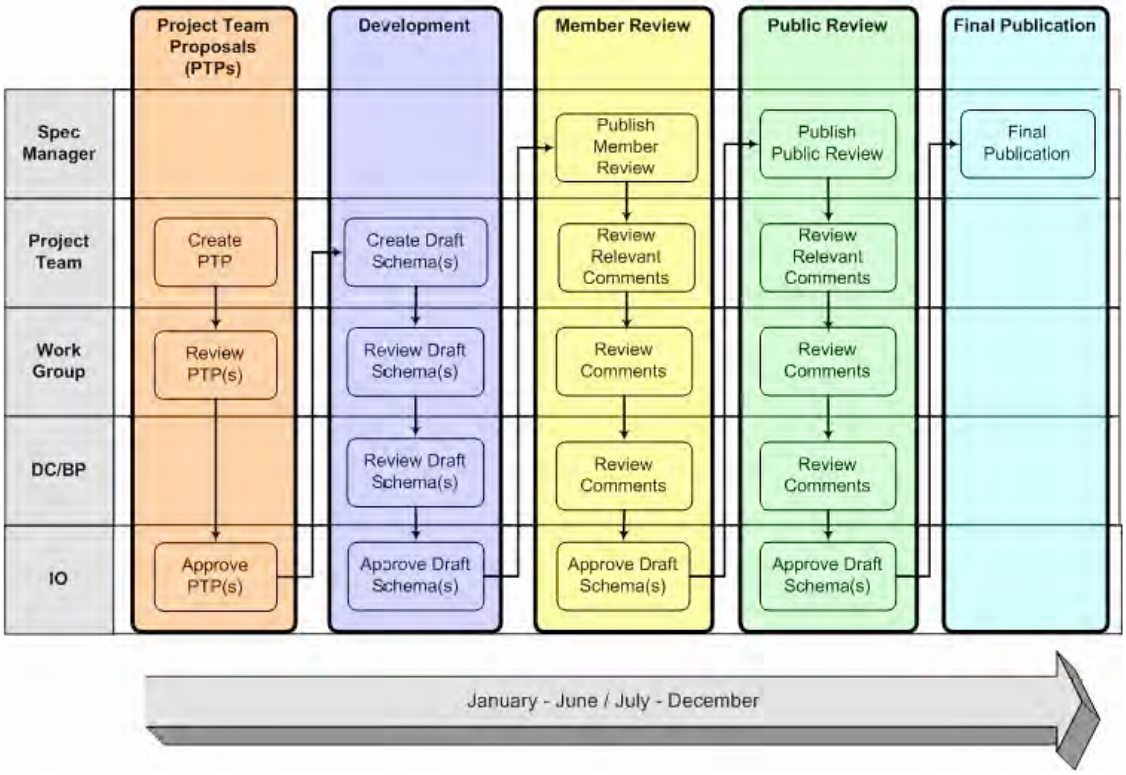

Figure 4.2- OpenTravel Specification Life Cycle

(Source: http://www.opentravel.org/Resources/Uploads/PDF/OpenTravelProcManv6May2008.pdf)

After this first process is complete, a member review followed by a public review of the specification open to all is made. This cycle eventually leads to the transformation of the in-development version into the next current version and the current version transforming into a past version. As a note, current versions along with past versions are specifications that have already been published on the OpenTravel domain and are available for download. These versions are official and can be freely used in projects and applications with no restriction. The in-development version is not recommended to be used in projects due to the possibility of it suffering changes as well as the fact that it is not accessible to the general public.

4.3- OpenTravel Schema

A schema in general is a model for describing the structure of information based on a set of rules and vocabulary. As for a XML Schema, it defines the entire structure of an XML document. Existing elements and attributes are defined in an XML schema along with the order and number of child elements that might exist. Other information such as data types and default values are also included in the schema. The schema acts as a guideline for the creation of the actual ".xml" file. The schema architecture of the OpenTravel Specification can be defined as being "a hierarchical collection of schemas that build from reusable simple structures into business messages [74]".

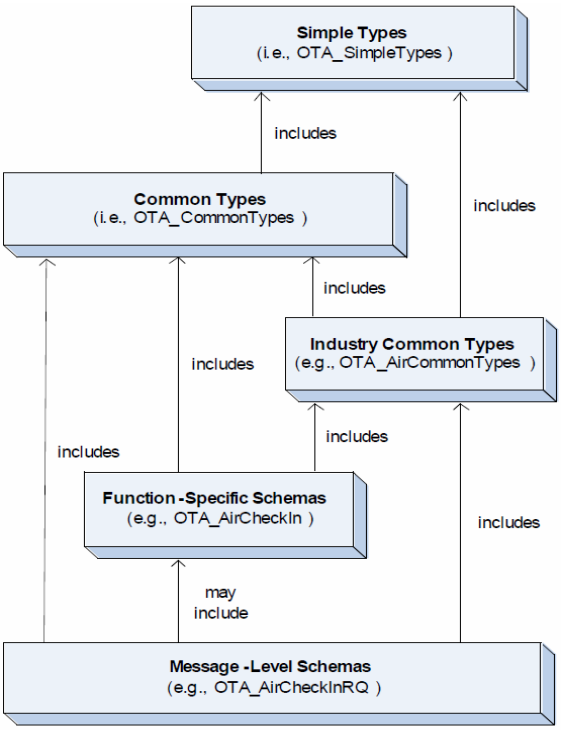

Figure 4.3- OpenTravel Schema Architecture

In the case of OTA, the schema architecture is currently composed of 272 ".xsd" files that make up the specification. A hierarchy divides this architecture into five different OpenTravel types. These types can be simple types, common types, industry common types, function-specific types and message-level types [75].

Simple types contain simple data types that are compatible with already existing W3C XML data types. It is important not to forget that interoperability in the OpenTravel specification is one of its main goals. An attribute of simple type is made up of a string with length between 1 and 32 characters. An example of a simple type file could be the "OTA_SimpleTypes.xsd". Common types contain common attributes that are used in message-level types, function-specific types and also the industry-common types. An example of a common type file could be the "OTA_CommonTypes.xsd". Industry-common types contain complex attributes used in specific industry areas. Examples of industry-common type files could be "OTA_AirCommonTypes.xsd", "OTA_HotelCommonTypes.xsd" and "OTA_RailCommonTypes.xsd". Function-related types are made up of reusable structures used across messages of similar functionality. Some message-level schemas can include attributes of a function-specific schema. Examples of function-related type files could be "OTA_DestinationActivity.xsd" or "OTA_AirCheckin.xsd". Lastly, message-level types usually define a business transaction and have a pair of request-response messages. These are top-level messages specifically created to address a particular issue. Examples of message-level type files could be "OTA_CruiseSailAvailRQ.xsd", "OTA_CruiseSailAvailRS.xsd", "OTA_VehRateRuleRQ.xsd" along with "OTA_TourSearchRS.xsd" and "OTA_VehResRQ.xsd". These are the most common types in the OpenTravel schema architecture.

Along with these five different OpenTravel types, the architecture of the schema is also composed of other elements. A namespace avoids conflicts between element names that are the same when more than one specification is being used. The most commonly used namespace for the OpenTravel specification is "http://www.opentravel.org/OTA/2003/05" [76]. Codelists and error messages are also part of the general architecture of OpenTravel. An official list of these codes and errors is available in the OTA website.

4.3.1- Schema-based XML Example

When we look at an OpenTravel XML schema, it is only natural to think about the ".xml" file that will be created based on this schema. This is the file that will be closely related to the website where the application will eventually be running. Promptly, two basic examples will be shown of the application of the OpenTravel schema on an ".xml"file. It is important to understand that these examples are

functional but also academic due to their simple nature. In real environments, examples tend to be a bit more complex, where other options and attributes can be added to meet industry requirements.

For the first example, "OTA_Ping" messages will be used. The OTA Ping Request and Response messages are used just like in the networking industry to test the connectivity of an application. The OTA Ping Request sends a string to the server and later the server responds to this request with a Ping Response echoing back the string that was initially sent. As we can see in the "OTA_PingRQ.xsd" schema, the default namespace is defined as being "http://www.opentravel.org/OTA/2003/05". The version of the OTA specification being used is also defined, in this case being the most recent available (OTA2010A). The OTA_PingRQ message-level schema inherits attributes from the common-types schema. Next, the OTA_PingRQ message is identified and its attributes are shown and specified. In this case, the "EchoData" is the only attribute present and is of type string. This is the attribute that will "carry" the string to the server as mentioned earlier.

The OTA_PingRS message schema is very similar to the OTA_PingRQ message schema. The OTA_PingRS schema also carries the "EchoData" attribute in order to send back the response to the client that initially executed it. Along with this attribute, others are included such as "Success", "Warnings" and "Errors". These attributes are used for debugging the application when something goes wrong such as when the server does not answer to a ping. When a ping request is successfully received by the server, along with the string, the <Success> attribute is also send back to the client to signal the success of the operation.

```
-<xs:schema targetNamespace="http://www.opentravel.org/OTA/2003/05" elementFormDefault="qualified" version="2.000" id="OTA2010A">
  <xs:include schemaLocation="OTA_CommonTypes.xsd"/>
 -<xs:annotation>
   +<xs:documentation xml:lang="en"></xs:documentation>
  </xs:annotation>
 -<xs:element name="OTA_PingRQ">
  -<xs:annotation>
    +<xs:documentation xml:lang="en"></xs:documentation>
   </xs:annotation>
  -<xs:complexType>
   -<xs:sequence>
    -<xs:element name="EchoData" type="xs:string">
      +<xs:annotation></xs:annotation>
     </xs:element>
    </xs:sequence>
    <xs:attributeGroup ref="OTA_PayloadStdAttributes"/>
   </xs:complexType>
  </xs:element>
</xs:schema>
```

Figure 4.4- OTA_PingRQ.xsd (abbreviated)

Below is the transcription of possible ".xml" files for the request and response of an OTA_Ping message. These are actual files that could be used in a functional application. The OTA schemas are merely informative and are never directly used in an application. As a note, in the ".xml" files, the string used for the test is "This is a test".

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap=http://schemas.xmlsoap.org/soap/envelope/">
        <soap:Header>
                <!-- Routing, security or other control data. -->
        </soap:Header>

        <soap:Body>

        <OTA_PingRQ 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_PingRQ.xsd" TimeStamp="2010-03-17T11:09:49-05:00"
        Target="Production" Version="1.001" SequenceNmbr="1">
                <EchoData> This is a test </EchoData>
        </OTA_PingRQ>

</soap:Body>
</soap:Envelope>
```

Figure 4.5- Possible XML Ping Request

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap=http://schemas.xmlsoap.org/soap/envelope/">
        <soap:Header>
                <!-- Routing, security or other control data. -->
        </soap:Header>

        <soap:Body>

        <OTA_PingRS 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_PingRS.xsd" TimeStamp="2010-03-17T11:09:49-05:00"
        Target="Production" Version="1.00" SequenceNmbr="1">
                <Success/>
                <EchoData>This is a test</EchoData>
        </OTA_PingRS>

        </soap:Body>
</soap:Envelope>
```

Figure 4.6- Possible XML Ping Response

For a second example, the "OTA_AirFlifo" messages will be used. The OTA_AirFlifoRQ message allows the request of real-time flight departure, arrival and gate information for a particular flight, including actual as well as scheduled departure and arrival times. On the other hand, the OTA_AirFlifoRS message responds to the OTA_AirFlifoRQ message with the actual information. As we can see in the "OTA_AirFlifoRQ.xsd" schema, the default namespace is also defined as being "http://www.opentravel.org/OTA/2003/05". The version of the OTA specification being used is the most recent available (OTA2010A). The OTA_AirFlifoRQ message-level schema inherits attributes from the industry common schema "OTA_AirCommonTypes". Next, the OTA_AirFlifoRQ message is identified and its attributes are shown and specified. In this case, "POS", "Airline", "FlightNumber", "DepartureDate", "DepartureAirport", "ArrivalAirport" and "FlightSegment" are the available attributes for use. The "POS" attribute is of POS type. "POS" attribute stands for "Point of sale". The Point of Sale identifies the party or connection channel making the request. The "Airline" attribute is of CompanyNameType type. Airline specifies the airline to request FLIFO details. It also identifies a company by name.The "FlightNumber" attribute is of FlightNumberType type. FlightNumber specifies a flight to request details for. A flight number can be either composed of 1 to 4 numbers followed by optional uppercase A – Z characters or simply OPEN or ARNK.

```
-<xs:schema targetNamespace="http://www.opentravel.org/OTA/2003/05" elementFormDefault="qualified" version="2.000" id="OTA2010A">
  <xs:include schemaLocation="OTA_AirCommonTypes.xsd"/>
 -<xs:annotation>
  +<xs:documentation xml:lang="en"></xs:documentation>
  </xs:annotation>
 -<xs:element name="OTA_AirFlifoRQ">
  -<xs:annotation>
   +<xs:documentation xml:lang="en"></xs:documentation>
   </xs:annotation>
  -<xs:complexType>
   -<xs:sequence>
    +<xs:element name="POS" type="POS_Type" minOccurs="0"></xs:element>
    +<xs:element name="Airline" type="CompanyNameType" minOccurs="0"></xs:element>
    +<xs:element name="FlightNumber" type="FlightNumberType" minOccurs="0"></xs:element>
    +<xs:element name="DepartureDate" minOccurs="0"></xs:element>
    +<xs:element name="DepartureAirport" type="LocationType" minOccurs="0"></xs:element>
    +<xs:element name="ArrivalAirport" type="LocationType" minOccurs="0"></xs:element>
    -<xs:element name="FlightSegment" minOccurs="0" maxOccurs="10">
     -<xs:annotation>
      +<xs:documentation xml:lang="en"></xs:documentation>
      </xs:annotation>
     +<xs:complexType></xs:complexType>
     </xs:element>
    </xs:sequence>
    <xs:attributeGroup ref="OTA_PayloadStdAttributes"/>
   </xs:complexType>
  </xs:element>
 </xs:schema>
```

Figure 4.7- OTA_AirFlifoRQ.xsd (abbreviated)

"DepartureDate" attribute is of Date type. DepartureDate, as the name indicates, is the date a certain flight departures an airport. "DepartureAirport" attribute is of LocationType type as well as the "ArrivalAirport" attribute. DepartureAirport and ArrivalAirport are the names of the airports the flight will be parting from and arriving at. Finally the "FlightSegment" attribute is a complex type. It indicates the flight leg for which FLIFO information is being requested.

Below is the transcription of possible ".xml" files for the request and response of an OTA_AirFlifo message. These are actual files that could be used in a functional application. As a note, in the ".xml" files a request is being made for the flight status of a TAP flight 6677 from Oporto to Newark.

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap=http://schemas.xmlsoap.org/soap/envelope/">

<soap:Header>
                <!-- Routing, security or other control data. -->
</soap:Header>

<soap:Body>
<OTA_AirFlifoRS EchoToken="566732" TimeStamp ="2010-11-01T010:47:34-06:00"
Target=" Production"
Version="1.000" SequenceNmbr="1" xmlns ="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05OTA_AirFlifoRS.xsd">

    <Success/>
    <FlightInfoDetails TotalFlightTime="PT6H45M" TotalMiles="4423">
    <FlightLegInfo FlightNumber="6677" JourneyDuration="PT7H45M"
LegDistance="4423">
    <DepartureAirport LocationCode="OPO"/>
    <ArrivalAirport LocationCode="EWR" Diversion="0"/>
    <MarketingAirline CompanyShortName=" Tap" Code="TP">Transportadora Aerea
Portuguesa
    </MarketingAirline>
    <Equipment AirEquipType="747">Boeing 747-800</Equipment>
    <DepartureDateTime Scheduled="2010-11-01T11:00:00"/>
    <ArrivalDateTime    Scheduled="2010-11-01T13:45:00"    Estimated="2010-11-
01T13:45:00"/>

<OTA_AirFlifoRS>

</soap:Body>


</soap:Envelope>
```

Figure 4.8- XML file of a possible OTA Air Flight Information Response

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap=http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Header>
                <!-- Routing, security or other control data. -->
   </soap:Header>
   <soap:Body>

<OTA_AirFlifoRQ 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_AirFlifoRQ.xsd" EchoToken="566732" TimeStamp="2010-11-01T11:23:34-
03:25" Target="Production"
Version="1.000" SequenceNmbr="1">

   <Airline Code="TP"/>
   <FlightNumber>6677</FlightNumber>
   <DepartureDate>2010-11-01</DepartureDate>
   <DepartureAirport LocationCode="OPO"/>
   <ArrivalAirport LocationCode="EWR "/>
</OTA_AirFlifoRQ>

</soap:Body>

</soap:Envelope>
```

Figure 4.9- XML file of a possible OTA Air Flight Information Request

5- System Architecture

5.1 – Multi-tier Architecture

There are many different ways to develop a complex information system. Depending on the type of system that is expected, there are different architectures that can be used. These architectures can range from the most basic client-server architecture to distributed, peer-to-peer, service-oriented or event-driven [52]. Inside the client-server architecture model, there are other more specific architecture subtypes such as the n-tier model and namely the 3-tier model.

The 3-tier architecture is one of the most widely used models for the development of software systems on the Internet today. Also due to its characteristics, it is a great starting point for the E-tourism information system that is here presented.

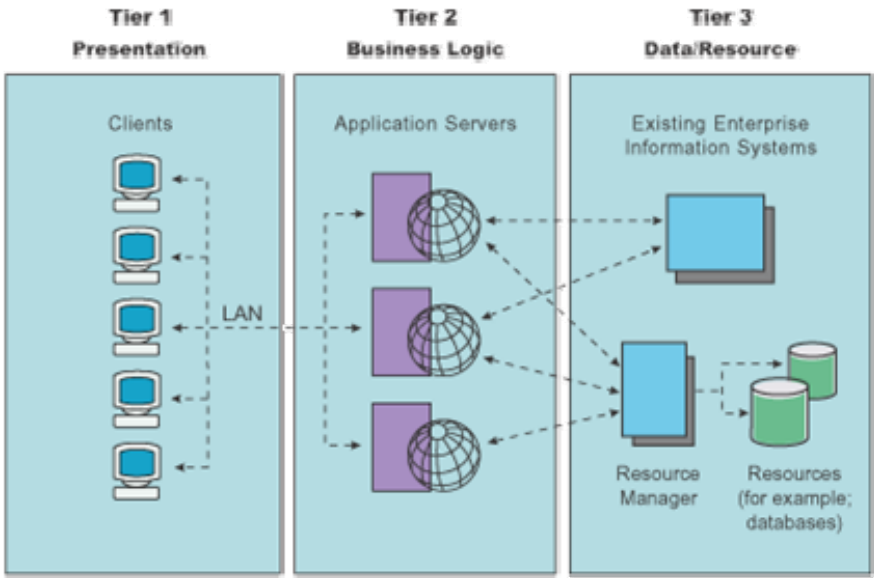

Figure 5.1- Basic 3-Tier Architecture Diagram

(Source: http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/index.jsp?topic =/com.ibm.websphere.base.doc/info/aes/ae/covr_3-tier.html)

The 3-tier model is basically divided into three different tiers of abstraction that separate application functionality throughout the system. These three tiers are usually called the Presentation Tier, the Business Logic Tier and finally the Data Tier. These tiers of abstraction do not necessarily need to be running on the same server. Ahead we will see this is very important to us, having in mind that the web services that we will be using are platform independent and are spread across the Internet.

The first tier that makes up the 3-tier model is the Presentation Tier. This tier can be considered one of the most simple to implement due to the abundant existence of interface development tools and their ease of use. Although this may be true, it is as important as the other tiers because it is responsible for showing all the data to the user and creating an interface with the Business Logic tier. For a Presentation tier to be successful it must be simple and provide a clear view of the system and its functionalities [53]. Browser compatibility should also be taken into account

The second tier, the Business Logic Tier, is responsible for the validation of all the data that is input to the system before methods are called from the Data Tier. When data is received from the following tier, it is also checked and organized so that is can be formatted and sent to the Presentation Tier. The Business Logic Tier serves as a bridge between the Presentation and Data Tier and usually is composed of as much logic as possible. This logic includes business rules, calculations and all other actions that need to be performed to successfully unite the system together.

The Data Tier is responsible for providing the information from the existing databases to the business tier and ultimately to the user interface in the presentation tier. It holds all information regarding the system and by not being directly linked to the end-user allows a better level of security than the traditional client-server models. This separation also allows for the database location to be changed without any necessary intervention on the client-side [54][55].

As we can see, separating the system into different tiers has various advantages. One of the most obvious is that these tiers can be developed and tested independently. This allows saving time and money because different modules can be developed at the same time and then put together to create the final product. Another enormous advantage is related to maintenance. When a system is complex and large in size, it is exponentially more difficult to debug and maintain. By dividing such a system into smaller pieces it enables the source of a possible problem to be reduced to a smaller area of influence. Also when changes need to be made, these can be executed faster and in an easier manner.

Of course, no architecture is perfect and there are some downsides to the 3-tier model, for it has a more complex structure. While developing a system with this

architecture, one must be aware of the possible need to one day expand the existing system. This is critical in companies that are growing fast [56]. Another problem and one of the most relevant to this project is regarding the physical separation between the different tiers. This separation, especially between business logic and data servers, may affect overall performance in a very negative way. Delay is one factor that can lead to dissatisfaction from the end-user. Waiting too long for a response from a system can be the difference between its success and failure. The only thing worse than waiting too long for a service, is not having a service to count on. This can also occur due to the physical separation mentioned before. The failure of one entity from a given tier can cause the failure of the entire system since the tiers are closely linked together and dependent of each other.

5.1.1 – The Hybrid Architecture

Although the 3-tier architecture can be useful in a wide variety of situations, having in consideration the needs of the E-Tourism Information System and the existing flaws of the 3-tier model, I decided to adapt this architecture. I found that not all issues that might occur are covered or at least minimized with the original model. The way I like to classify the adapted architecture I used is by saying it is a hybrid between the 3-tier architecture and a Service Oriented Architecture (SOA).

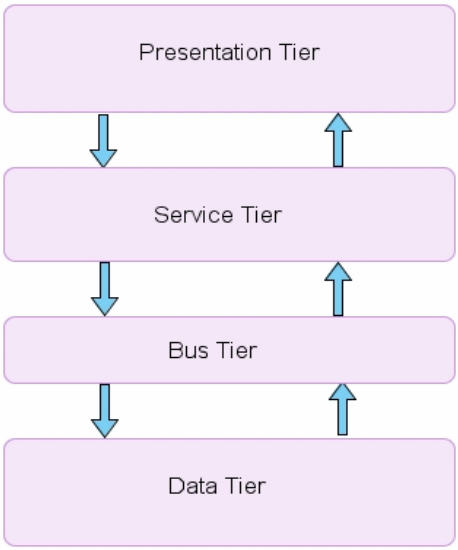

Figure 5.2- Hybrid Architecture Diagram

As we know, a SOA is an architecture that is focused on the usage of services that are available on the Internet either publicly or through agreements with other companies. The main objective is to join together and use these services instead of reusing the code that they are composed of [57]. This also allows the abstraction of the development of the service along with its implementation. Now different code

can be used in a transparent way without the worry or the necessity of altering methods or converting code from one programming language to another. All this put together results in a better alignment of the system with the business needs that exist.

By combining the best of both worlds, I came up with a solution that divided the original Business Logic Tier into two different tiers, the Service Tier and the Bus Tier, as seen in figure 5.2. The Presentation Tier along with the Data Tier remained in essence the same.

5.1.1.1- Presentation Tier

Just like the original presentation tier from the 3-tier architecture model, this hybrid presentation tier also displays the information available in our system. The only difference now is that the information that is displayed is not coming from one centralized source but from various different sources that are remotely located and spread across the network. This complexity is completely transparent to the user and his experience while using the interface should be the same as if he were using a local database to retrieve information.

5.1.1.2- Service Tier

The Service Tier is one of the most challenging tiers I developed in the hybrid system. There are several different events that are happening in simultaneous or waiting for others at any given time. One of the main reasons that influenced me in dividing the original 3- tier Business Logic Tier into two distinct tiers was the initial complexity it presented. Along with the changes I added, the original tier would have been even more complex and difficult to manage. Due to this, I decided to transfer as much logic as possible to the Bus Tier and give it other functions we will see just ahead.

The Service Tier is responsible for the dispatch of all the requests and responses coming from and to the interface. Although the interface can be composed of Flash elements, CSS or other technologies, HTML will be the main carrier of tourism-related information in both ways. This will be possible due to the input that will be placed in the interface either by the means of forms, radio buttons, check boxes or text boxes. The information inserted in forms at the interface is sent over to the Service Tier where it will be processed. The forms are then parsed in order to extract their content and requests are categorized by location.

If the data requested is local, a query to the local database is requested. The local database will store information such as profile data, user settings, history of

transactions and in addition it can also store personalized publicity or search-related offers.

Although this topic is not directly linked to the system, increasing the e-tourist's integration with the system is an interesting area that can be later developed. By learning about the e-tourists' tastes, in a non-invasive way, the system can increase the interactivity with the client and ultimately increase sales. Offering personalized publicity, such as when a person looking for a hiking adventure is shown hiking boots, can also enhance the earning of the system indirectly by capitalizing in protocols with external companies Another way to attract more clients, is by offering the ability for experience exchange between users. Writing testimonials and sharing pictures can be included for e-tourists that are unsure about a trip. The data and pictures related to this can also be accommodated in the local database.

Returning to the nature of the requested information, if the data that is requested is stored in a remote location, a SOAP Request message must be created. This message will be created having in consideration the queries that were performed in the interface. The structure of the SOAP Request will use the OpenTravel specification as a guideline. This request will be sent to the Bus Tier that will later handle it. Some time after that, the Bus Tier will respond with a SOAP response also using the specification mentioned before. In order to send the requested data back to the interface, the Service Tier will join the remote data with the local data, format the data according to the web interface and send it to the user's browser.

The innate capability for SOAP to serialize objects makes it the perfect choice to send objects from one tier domain to another. It also allows the persistence of an object [58]. This means the object can be retrieved and restored to its original state after being on "hold" for some time. Serialization and deserialization play a key role for requests and responses not to be lost or mismatched during the process of information retrieval.

Looking back at the database, I decided to make it local to the system as mentioned before. Since the proposal developed here is for use in a small to medium-sized region, I did not find the need to use an enterprise solution for the database. Although this may be true, it the event of an unexpected growth of sales, a more scalable solution can be easily adapted to the system, such as distributed database architecture by Oracle to introduce redundancy.

5.1.1.3- Bus Tier

The main reason for the creation of this new tier was to eliminate as much logic as possible from the Service Tier described above. The Service Tier by itself is complex enough. The other reason is related to the maintainability of the new Bus

Tier. By being an independent tier, changes can be executed more easily and without disturbing the other tiers in any way. Another advantage of creating this additional tier was having the possibility to replicate it. By replicating this tier, not only are we able to support more users (as long as the web services are able to keep up), we also introduce redundancy to the system. Requests coming from the Service Tier can be easily sent to different bus tiers and load-balancing becomes a reality.

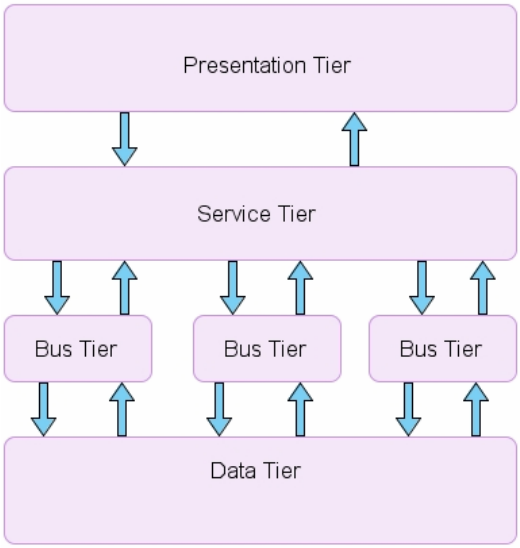

Figure 5.3- Hybrid Architecture Diagram with Bus Tier Replication

As we can see, this tier serves as a physical implementation of the Enterprise Service Bus that was referred earlier in chapter 3. It allows the orchestration of all the different actions that are happening in this tier. Internally, the Bus Tier receives a SOAP request in OpenTravel format and then is responsible for "translating" that request into multiple SOAP requests depending on the web services that are needed. This happens because, unfortunately, some of the major tourism operators do not use the OpenTravel specification yet. The methods that will be used and invoked in their web service might even do the exact same thing as the ones of OpenTravel, but just having a different name is enough to introduce this issue. Despite this being the only way to remotely access the data, in many cases the translation is almost transparent. This issue also actually brings some advantages to the system itself. Filtering is the most significant. For example, if a user only requests information about a flight to Pakistan, it makes no sense for the Bus Tier to send a request to the Hertz, Hilton or even TAP web service. Companies that are not related to aviation or do not offer flights to specific destinations should not be requested. This saves resources along the network and decreases the amount of data that needs to be processed by this tier. The time needed to send and receive data from the web services is also greatly reduced. This is thanks to the ability for the requests to be sent in parallel to all the different web services. Instead of sending a sequence of ten consecutive requests to ten

different web services and waiting for the response one by one, the bus is able to send those ten requests at the same time and then wait for their response as if it were one. In 5.1.1.4, this is explained in greater detail.

Aside from message transformation, other actions are provided by the Bus Tier. Actions such as message validation and message routing are essential for the correct information being sent to the right place every time. Exception treatment is also important when there is a problem retrieving information or accessing a service. OpenTravel currently has 849 error codes that allow a quick way of understanding and fixing a problem when an exception occurs. Although I decided to name this tier the Bus tier, as it acts as a hub or bus for web services, it can almost be confused with an oversized web service itself that calls other web services located elsewhere. Globally speaking, it basically just receives SOAP request and sends SOAP responses. The ability to request different information from different web services at the same time is what sets it apart from others.

5.1.1.4- Data Tier

The Data Tier retains its initial function as in the original 3-tier architecture for it continues to hold the system's information but now in an abstract and distributed way through the usage of web services. Instead of using a database to store information in a conventional way, the projected system uses web services that provide all the information necessary and also the means to alter and use this information.

Although the scattered information is ultimately stored in databases, the e-tourism information system is not responsible for the maintenance of these databases. Each company that provides the web service is the one responsible for taking care of these databases and the frameworks that are behind the web services. This allows the company to have better protection over its data and code since the final user is not in direct contact with there internal framework. By doing this, the company also insures that its information is always obtainable and up-to-date and most importantly, it is delivered in an acceptable amount of time to the end-user. It is in a company's best interest to provide an efficient and reliable web service to our system.

As we can see in figure 5.4, the time the bus tier takes to request and receive information from the data tier is the direct result of the web services used. The round-trip time by definition is the time required for a signal pulse to travel from a specific source to a specific destination and back again [60]. In our case, this time is the time that a Soap Request message takes to be sent to a specific web service and later a Soap Response message to be sent back to the bus tier. Over 90% of round-trip times (RTT) to any given destination on the Internet are smaller than 1s

[62]. The RTT must be added to the time the web service takes to access and prepare the information to be sent back to the bus tier. Although there is very little information about the mean value of processing time related to web services, 96,15% of most queries to an Oracle database take less than 1 second [61]. Having this in consideration, the processing time has been estimated in 2 seconds although it is thought to be a very inflated value. This gives us a general idea of how long a query to a web service might take. Thinking in terms of simple tourism web sites that already exist, normal queries rarely take more than a few seconds to process and display. Considering that a web service for our system needs not to render to an interface at this phase of the system, we can imagine that the information will take very little time to process as it is basically text.

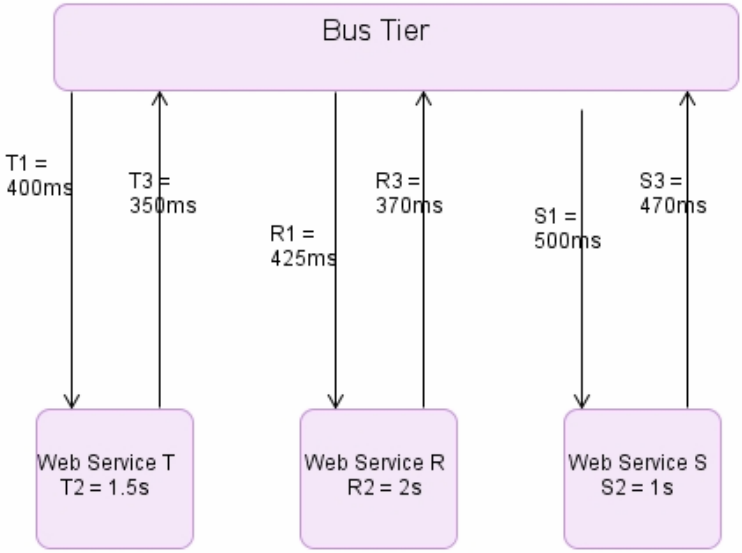

Figure 5.4- Round-trip and Processing Time Calculation

As said earlier, 2 seconds seams to be an inflated value. But by looking in parallel to the web services, we see that the amount of time that the bus tier will need to wait until it receives a Soap Response will be the biggest of the sum of the round-trip time added to the processing time of all web services. Adding 2 seconds from the processing time to the average round-trip time of 1 second gives us a comfortable value of 3 seconds.

With the usage of a time-out at the bus tier, we are able to cancel a pending operation and abort its resource allocation if it takes to long to process. An example of this scenario is a specific web service not responding to requests while others are working correctly. This should not affect in any way the other web services that are providing a working service. The time-out considered should be slightly larger than 3 seconds. Later this value can be decreased for optimization as long as it is never smaller than the value of the slowest processing time of all web services plus one second for the RTT.

5.2 – System Packages

In order to organize and have a better view of the system, use-cases and use-case actors, I divided the model environment into packages. These packages separate physical as well as virtual areas of influence of different actors along the system's scope. I identified five different packages according to this. The five packages identified were the Interface Package, the Service Package, the Bus Package, the Web Services Package and finally the System Coordination Package. As we can see from figure 5.3, these five packages are inline with the hybrid architecture model mentioned earlier. Despite the existence of these packages in different areas of influence, this does not mean they are isolated in anyway from each other. In fact, they are mostly very strongly linked together.

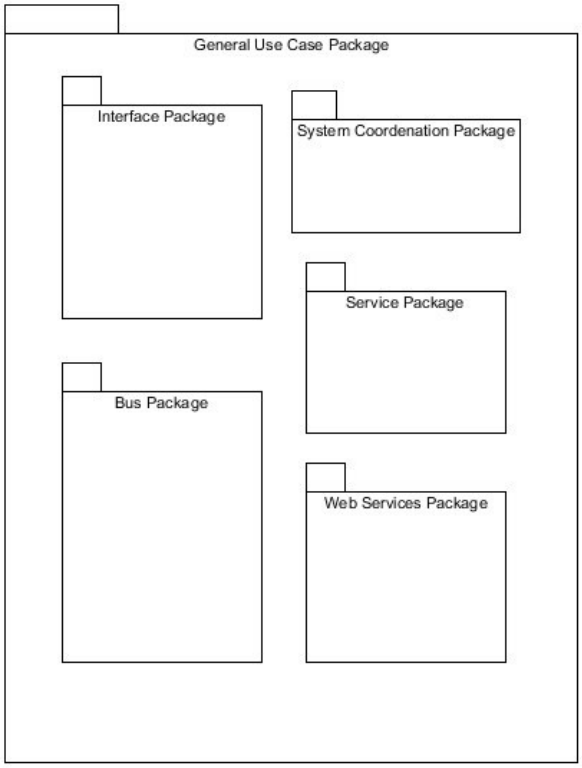

Figure 5.5- General System Package Diagram

5.2.1- System Coordination Use-Case

In order to run a successful business, it is important to have control over the operations that are going on throughout the system. High-end operations are the base for the entire system and act as guidelines for the rest of the actions that will take place. The high-end level operations that are most common in an E-tourism

Information System are represented in figure 5.4 where we can see the main roles of the actors.

The actor President/CEO is in charge of defining global business strategies for the company he is responsible for. Global management of the organization is also linked to him. This actor should have some experience in the field and have a very wide perspective of tourism. He is responsible for making key decisions regarding the system. He should also organize reunions with the Board of Directors in order to discuss and inform these of decisions that are made. Good communication between him and the Board of Directors allows changes to happen more rapidly and with less effort.

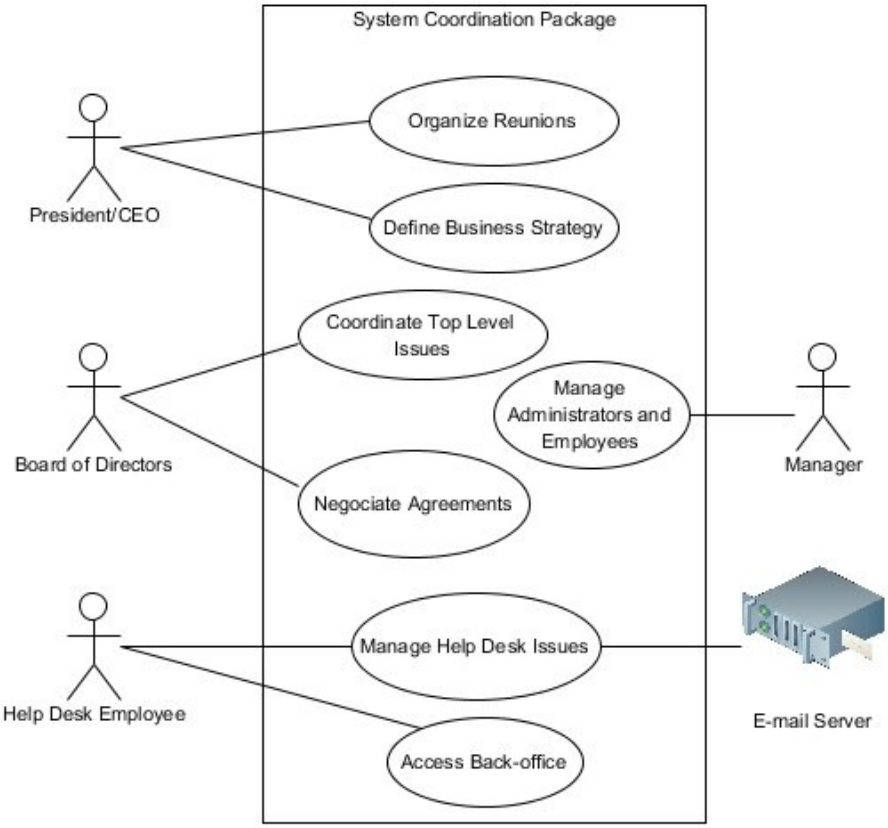

Figure 5.6- System Coordination Use-Case Diagram

The Board of Directors is an actor that is composed of several different members. These members can be in different hierarchies and managed by other members. This depends on the jobs they will be performing or assisting and the importance of their role to the system. All in all, there main objective is to coordinate top level issues that are in vogue. They get the main idea from the President/CEO and together discuss the best way to perform or implement the idea. This actor can also be responsible for the negotiation of agreements with other companies. In our system, a possible agreement could be linked with usage and access to a specific web service or the negotiation of profit for the usage of the same web service.

A Manager can be a very broad actor. In our case, he is specifically intended to work with the system's administrators and employees and transmit the actions that are requested by the Board of Directors. He should provide guidance to the administrators and employees and participate as much as possible in order to make sure the requests of the Board of Directors are fulfilled. He is also responsible in some cases for informing administrators of agreements or specific changes to the system, either verbally or with written documents. The Manager will usually get this information from the Board of Directors.

Finally the actor Help Desk Employee performs the well-know job of assisting the E-tourists that will be using the system. All doubts, problems, interface issues, password recovery and questions should be handled by this actor. There should be a written protocol that he should follow based on a Terms of Utilization document of the company. In order to communicate with the E-tourists and other entities, the Help Desk Employee will use the e-mail server provided by the system along with other traditional forms of communication. The Help Desk employee should also have access to the back-office of the system where he can perform limited modifications to the database. In case of the inability to solve the problem, he should contact the manager for an answer. In the event of a more significant issue, the manager can contact an administrator or a member of the board of directors and so on until he might ultimately reach the top of the hierarchy, the President himself.

5.2.2- Interface Use-Case

The interface is the only portion of the system that will be viewed and used consciously by the client of the system. The client is called E-tourist as mentioned earlier in Chapter 2. The rest of the system along with its operation should be completely transparent to him. The E-tourist will be able to perform different tasks depending or not if he is registered in the system.

A simple E-tourist that accesses the system should be able to browse information regarding reservations, availability and prices and of course query for this information. If an E-tourist is pleased with the general concept of the system he can also register in order to perform other tasks. A simple E-tourist has the frequent behavior of a person that is just browsing for information and not so much interested in details. He performs this action to get a general idea of a possible trip or package he might intend to purchase later on.

If an E-tourist logs into his account, he is promoted to a registered E-tourist and is able to perform tasks such as purchasing a tourism package, viewing personalized information and logging out. Needless to say he is also able to perform any task a simple E-tourist is capable of performing.

Just to complete the idea behind personalized information, this can be linked to personal data located in the user profile. Address, phone number, age or payment methods fit into this category. On the other hand, personalized information can also be linked to the capacity of the system learning about the e-tourists tendencies. This issue will be briefly explained further ahead.

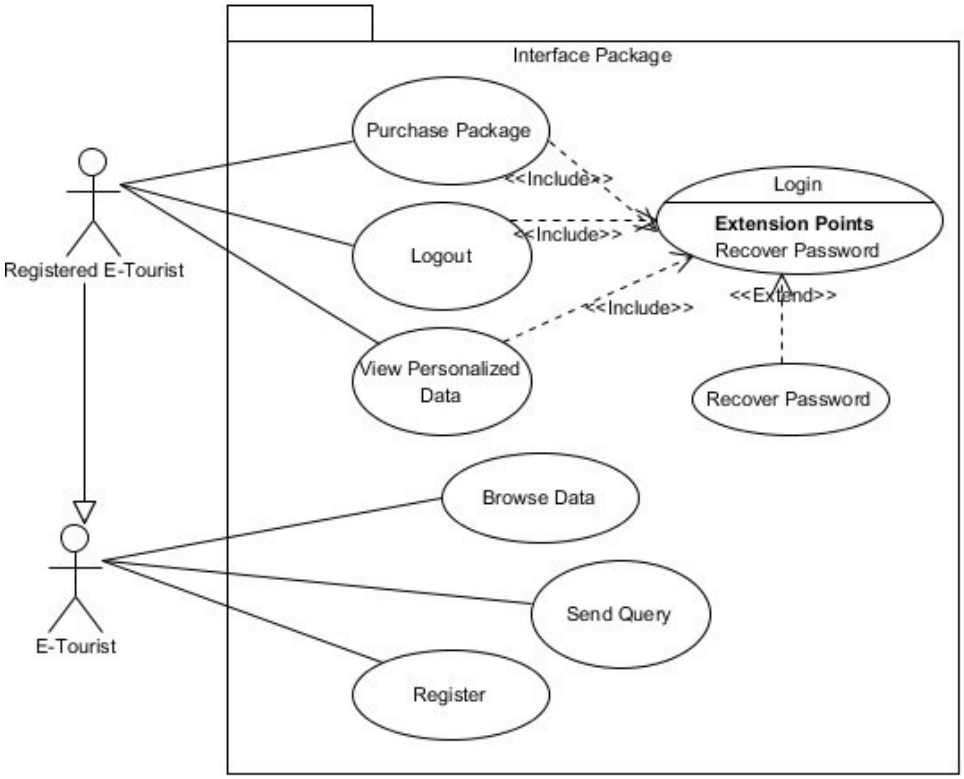

Figure 5.7- Interface Use-Case Diagram

5.2.3- Service Use-Case

In the Service Use-case diagram, we have two main actors, the server administrator and the database administrator. The server administrator will be responsible for updating the user interface displayed to the users in general. This interface is directly connected to the web server and it is this web server that is responsible for attending requests and sending responses back. The server administrator is also in change of updating the OpenTravel specification that is used in the SOAP messages that are sent between the Service Tier and the Bus Tier. The Open Travel specification is usually updated twice every year and backward compatible. The server administrator is logically also responsible for maintaining the server functional and promoting little or no down-time to the system.

The database administrator on the other hand is responsible for maintaining and optimizing the local database and coping with expected growth in the size of data stored. He is also responsible for the backup of the database so that in the case of

an unexpected failure he can easily recuperate the initial state of the database minimizing data loss significantly. Finding strategies for reducing the time the database takes to respond to queries is a definite plus. Needless to say that the database administrator will be working closely with the database server and must be responsible for its overall functionality.

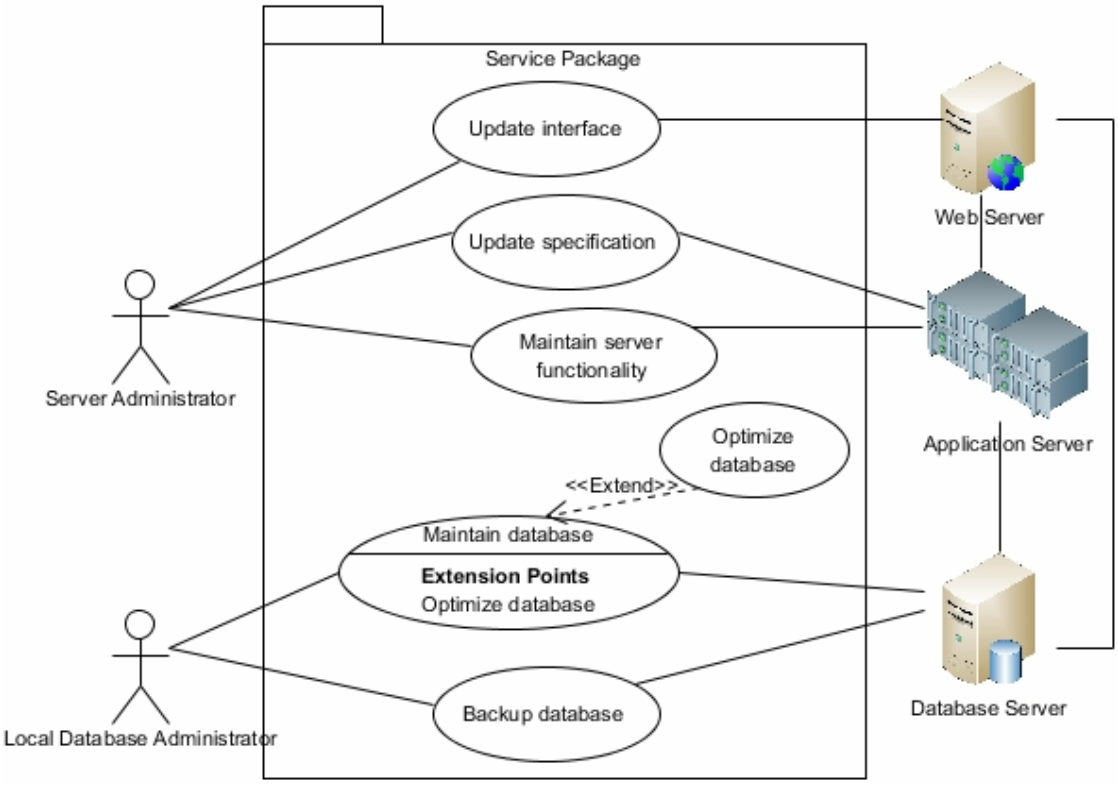

Figure 5.8- Service Use-Case Diagram

5.2.4 – Bus Use-Case

The bus administrator will be the entity responsible for maintaining the bus tier functional. In order to do this, this entity must be able to use the methods of all the different web services with the help of the WSDL files. As we know, the WSDL file of each web service should have all the information needed for the complete description of the service itself. Maintaining the WSDL specifications updated internally in the Bus Tier is one of the tasks the bus administrator should have present. Although these specifications do not change very often because companies are very reluctant to change, any change should be rapidly assimilated in the Bus. Adding, updating or deleting old specifications are the standard actions.

Another task for the bus administrator is the management of agreements between different companies inside the Bus. Although the Bus administrator will not take any part in the negotiation of the agreements, he must be responsible for adding these agreements to the system. He will be aware of new agreements either

though the usage of the WSDL (if this action is supported by the web service), or in written from the Manager that is responsible for transmitting it to the administrator as mentioned in 5.2.1. The system should then automatically include the possible agreements that translate into a discount or offer to the request that was made by the client. Only relevant agreements should be used in a specific request. In other words, if a company has a specific agreement with another company, by combining two different products from both companies, the E-tourist can benefit from an additional discount due to this. An example for this is when a client is searching for a car rental service and a flight to Chicago and there is a 20% discount for anyone who buys a combined flight and rental with American Airlines and Hertz, due to an existing agreement between the two companies. When this is teamed up with dynamic packaging, a once static system becomes highly dynamic but this also implies new development challenges as we have seen.

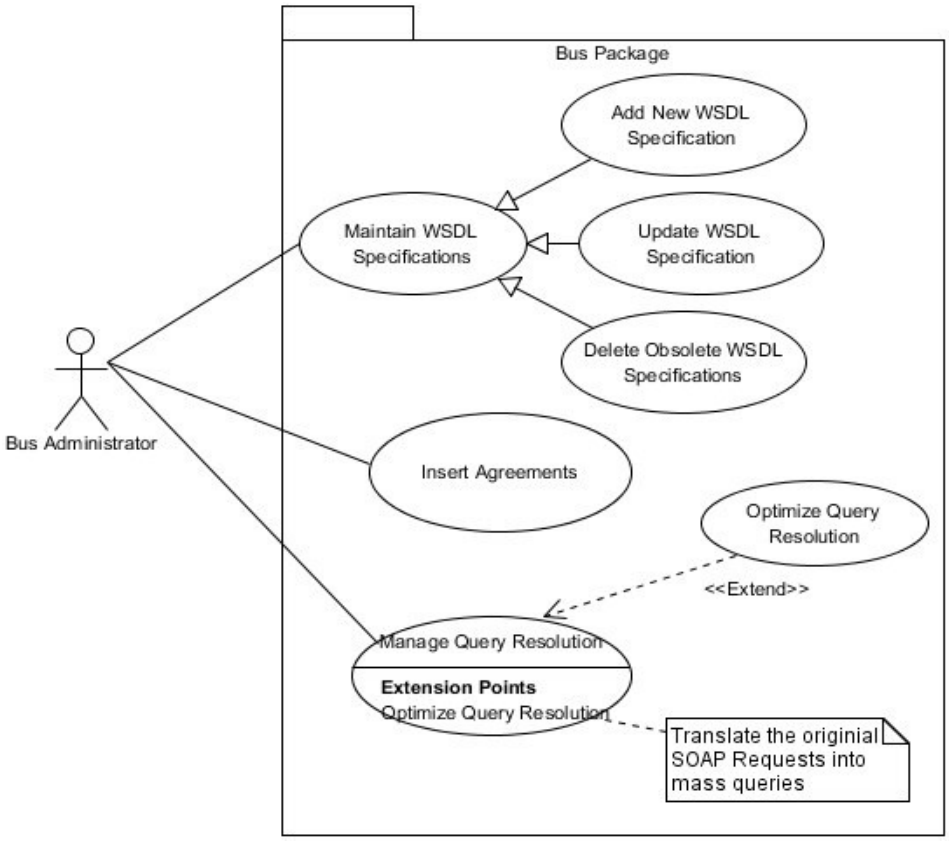

Figure 5.9- Bus Use-Case Diagram

One of the final tasks the Bus administrator is in charge of is the management of query resolutions. Query resolutions can be seen as the translation of the OpenTravel Soap requests into web service Soap requests. Depending on each specific request, the Bus must translate it into multiple requests to the web services that are relevant to the query. As seen in 5.2.1.3, calling these web services should

be done in an orderly way to promote optimization. An example of optimization is only calling web services that offer solutions that are desired. Calling a car rental web service when we are only looking for a hotel is not a good strategy. Another example of optimization is when, in order to obtain necessary information, instead of calling two different methods from the same web service, a new method is discovered that retrieves the same information in less time or with less bandwidth usage.

5.2.5 – External Web Services Use-Case

Although the external web services are not administrated or responsibility of the E-tourism Information System, I decided to include a general idea of their operation. Everything that is "hidden behind a company's web service" can not be altered or managed without the company's consent. Internally, the company must have an entity responsible for the implementation and maintenance of the web service along with all the other components that makeup the framework.

Besides maintaining operability, another important task is the update of the WSDL file since this is the only way the Bus Administrator can be aware of the operations he can perform with the web service. When changes are to be performed, the Bus Administrator can be informed before the change is rolled-out giving him time to prepare and avoiding unavailability of the new or updated service. It is in the interest of the company to inform the E-tourism system of any modification in advance. A service that is not working is a service that is not generating money for the company.

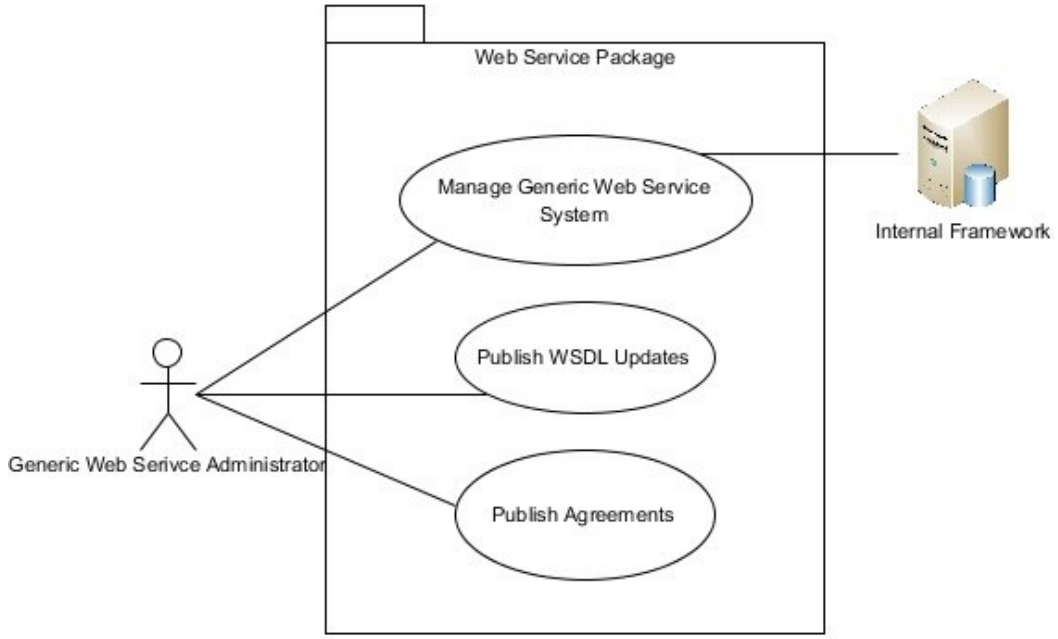

Figure 5.10- Web Service Use-Case Diagram

In the same way, new agreements can be published through the WSDL and added by the Bus administrator to the system. This is the most secure way of transmitting this type of information because the data is being made available directly by the company. Possible misinterpretations or loss of details are less likely to happen when using this method of communication. Changes regarding agreements are expected to happen more often than changes regarding the internal methods of a web service. This happens because sales, offers and specials are more likely to vary, depending on season, internal strategies, product availability and even speculation of markets

5.3- System Components

Using the OpenTravel specification as a guideline, we can identify the main components that make up the E-tourism Information system. This allows us to have a better understanding of the part played by each of these entities and their influence on the system. Due to the sheer size of the specification, only the most important entities will be analyzed.

The e-tourist is one of the most important entities of this project. The whole system revolves around him. When an e-tourist registers with the system, a profile is created for him. The e-tourist profile can only exist when it is associated to an e-tourist for it makes no sense existing on its own. The profile contains important information about the user of the system and allows interactivity with the interface.

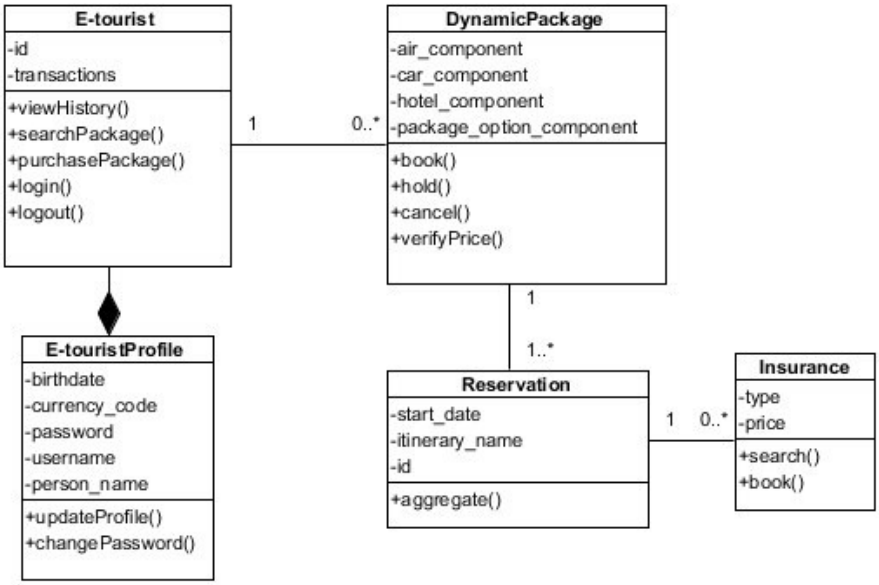

Figure 5.11- Main System Class Diagram

Also associated with the e-tourist are the dynamic packages he is able to create. Up until the OpenTravel 2008B specification, support for dynamic packaging did not exist. It is only comprehensible that a lot is still missing from the specification to support all the different types of reservations that exist since it is so recent. Still, OpenTravel already has support for air, car, hotel and package option components. I imagine that the next release might include some updates for the dynamic package container, namely for cruise, golf and static package components. In order to create dynamic packages, various reservations must be joined together. Each dynamic package must have at least one reservation associated with it. Although this is true and possible, the real advantage of dynamic packaging comes when multiple reservations are called upon. Reservations can also have different types of insurance related to them.

As said before, there are various different types of reservations that an e-tourist can access and book. All of these reservations inherit characteristics from the main class "Reservation". The Vehicle reservation class is related with car rental services. Important information such as pick-up location, the number of rental days, vehicle type and rates are essential although there are many more attributes that exist and are possible to include.

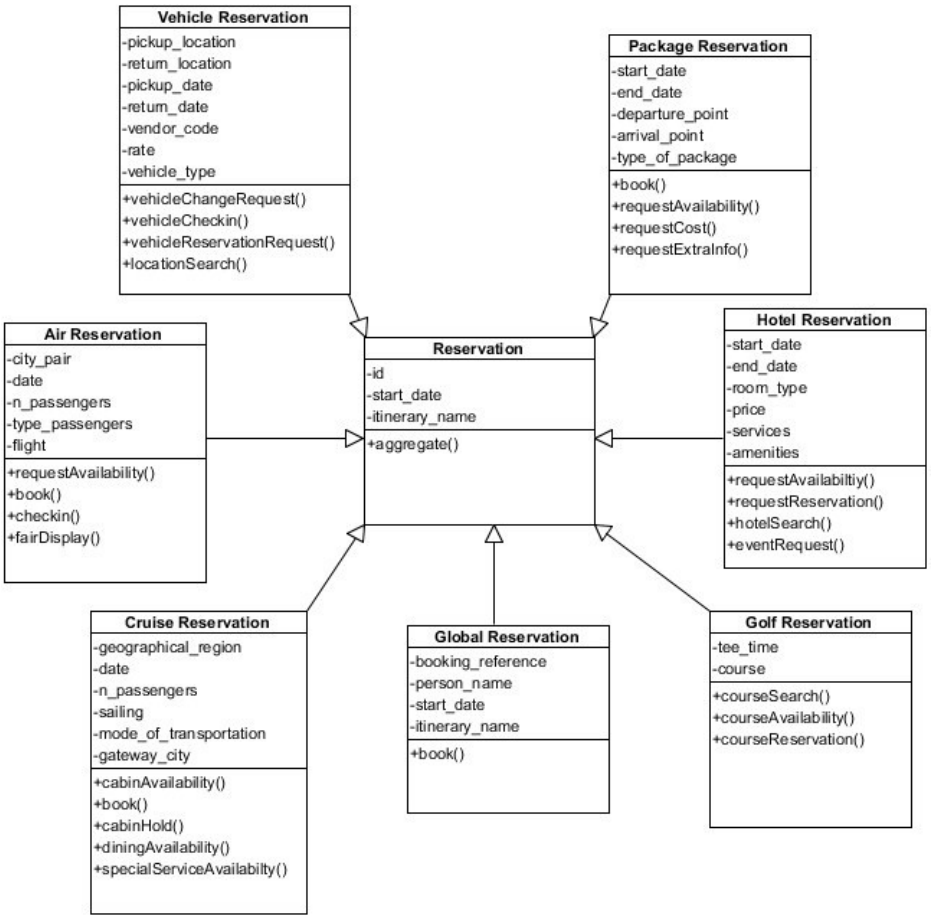

Figure 5.12- Reservation Class Diagram

Air reservation is dedicated to flight programs. The city origin and destination along with date, number of passengers and flight number are the main attributes for this class. Needless to say that all reservation classes allow requisition of availability, booking, searching and other normal operations associated with travel. The Hotel reservation class is destined for the booking of hotel rooms and services. Again, attributes as start date, number of nights, room type and price pop up as well as services and amenities. These last two are generic and can include cable service, mini-bar expenses, wi-fi access or special requests. The Golf reservation class is a smaller size class that includes attributes such as the time the tee is intended to be used and the desired course. The Package reservation class should not be confused with the dynamic package class. The Package reservation class refers to a static package that is already made available by a tourism service provider. The dynamic package class is made by the e-tourist inside the E-tourism Information system.

Just to clarify, acquiring a package from an external service though the system such as an "all-in-one, everything included trip" is considered a static package. By joining this trip with a car rental, the e-tourist himself is creating a package that is dynamic since he is the one making the decisions and choices. The bundle of the two reservations is the dynamic package although there is a static package inside.

Coming back to reservation types, we have the Cruise reservation class. Since this class is a bit more complex, I decided to add figure 5.11 to further explain its contents. A cruise itself is a complex service that is provided to an e-tourist. There are many different things that can make up a cruise experience. Starting with the cruise cabin, this class is destined to ordain the cabin inside the ship that will be assigned to the e-tourist. The number of guests that will be using the cabin and the selected fare also play a role here. There can be more than one cabin assigned to a single e-tourist if this is his wish. The cruise fare is responsible for contained the methods for calculating the final price of the cruise with all the services included. Cruise dining is for selecting the type of cuisine service that will be available during the voyage. The dining room used can be selected and the guest names, number and types can be selected. Other options can include services such as open-bar or Captain's Dinner. As for the Cruise Special Service class, as the name indicates, it is used for special services that can be provided during the cruise event. These services are the ones that can not be fit into the other existing classes. The Cruise Shorex class is specialized in tours, trips, excursions and other activities that are performed outside of the ship, usually ashore. These are usually expected to happen when the ship moors. These services are traditionally purchased on location but with the OpenTravel specification, now they can be easily researched and purchased beforehand. Lastly, the Cruise category is the class that selects the rights and privileges that the e-tourist will have during the sailing. It is closely tied to the fare that is paid.

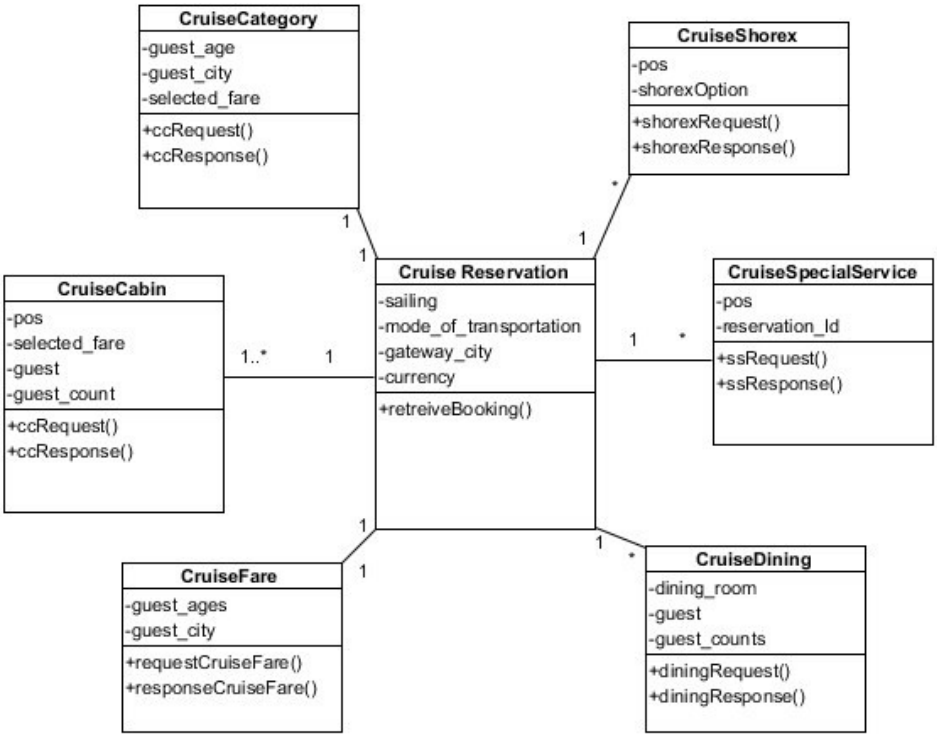

Figure 5.13- Cruise Reservation Class Diagram

Finally, the last reservation type available is the Global reservation. This class enables the reservation of any service that does not fit into the other types of reservations available. It is a very generic class in order to accept various attributes and methods for a large selection of less structured services. For example, a visit to a specific place or an unconventional activity can be easily fit into this class. In sum, all of the available reservation classes allow the specification to be able to accommodate almost any tourism service in existence.

6 - System Operation and Concept Validation

6.1- System Operation

Despite the complexity of the E-tourism Information system, there are relatively little procedures that will be frequently performed on the system. Some of the most relevant are the login, the query, the purchase confirmation and the logout. Other procedures exist such as registering, adding testimonials, updating profile or viewing past transactions but these are less relevant and are ultimately contained inside others in one way or another. The small number of procedures that can be executed might be surprising compared to the complexity of the system but this happens for a reason. This provides a simple, clean interface for the user with all the necessary functionalities. Little knowledge about the system is needed in order for the E-tourist to use the interface and confusion is widely avoided. Only vital information is asked for and only the required level of information detail is sent back to the buyer.

In relation to procedures, each and every action that is performed will not be describing in detail for each procedure in each tier of the system. This would imply the election of a specific technology or solution for each of theses, going against the modulation nature of this project. Although this is true, for the development of a simple prototype of the E-tourism system, and in particular for the Bus and Web Service tier, the GlassFish Application Server project was used which includes the OpenESB solution as well. This choice was based on the amount of time available for the production of the prototype, allowing quick learning of the ideologies behind it and a simple and fast implementation. Oracle Netbeans was the integrated development environment chosen due to previous experience of the interface.

6.1.1- Login and Logout

Logging in is one of the first actions that an E-tourist executes when he or she is committed to search information about an e-tourism package. The login procedure starts when the user requests to login to the system through the interface. The web server responds and displays the login page. The user then inserts the username and password and requests validation. The login information is sent to the web server where the data is stripped from the HTML code and sent to the application server. The application server is then responsible for performing a query to the local database for validation of the password. Necessary encryption and decryption are also performed here. If the password present at the database matches the one in the application server, the response it sent back to the web server that will then integrate the affirmative response into the interface. If the password does not match, the web server will send back a new page asking the user to retype the password or to request password retrieval. As we can see, the Bus along with the Data tiers where not used at all in this procedure. The ability to distinguish the internal data requests (local database) from remote data requests (web services) should be included in the application server.

The logout procedure ends up being even simpler than the login. The user clicks the logout button and the connection is promptly terminated. Access to the database in this case is only necessary for registering the time and date the user left the system (log). The web server can respond immediately with a new page confirming the logout without having to wait for any response from the database. The connection between the user and the system can also be automatically terminated after a certain timeout is reached.

6.1.2- Query

In order to perform a query to the E-tourism Information System, one must navigate to the query page. This page should have the capability for an e-tourist to immediately pick the types of components he desires to have in a package he builds. Other suggestions can be later given to the user to add to the package. The request is inserted in the interface and sent to the service tier just like what happens with the login procedure. The webpage is parsed and then remote information requests are separated from local requests. In the case of local requests, the local database is used in the same way as seen before with the login. When the database retrieves the information, it is set on hold until later on in the process. For the remotes requests, a SOAP request message is created using the OpenTravel standards. This message is then promptly sent to the Bus Tier for processing.

In the Bus Tier, the OpenTravel request is received and is then converted into multiple SOAP requests. Each new SOAP request will be created having in mind the web service it is intended for. The new SOAP requests are sent to the right web service and the Bus Tier then waits for a response from each web service. As said earlier on in the chapter, this is done in parallel to save time. When all web services respond, the information is joined together and converted into the OpenTravel format again. Agreements, discounts or special offers are also joined in order to be formatted accordingly in the Service Tier where all the data is sent to. In case that one or more web services fail, there is the possibility for requests to be resent to the web services that initially failed. If this happens a specific number of times, the web service that continuously failed is discarded and information is sent anyway. Every time a request is performed, a time-out should also be used so that the Bus does not wait for an unexpected amount of time for a response from any web service.

The Data Tier for this project does not need to be known in detail in order to use it. Whether the framework behind the web service is Microsoft .NET, Spring or any other, we can use their web services in the same way. Technologies, different programming languages, databases and web servers are also completely transparent to our system due to the fact that we will not be using them in a direct way. In sum, a web service provider can virtually use anything to build a web service, as long as it works and is reliable. Back at the Service Tier, the OpenTravel SOAP response is received and the data contained within it is added to the data that had previously been retrieved from the local database and set on hold. The data from both locations is formatted and sent to the Presentation Tier where it is shown by the browser. The user then views the information unknowing the complex process that occurred in the background.

6.1.3- Purchase Confirmation

The purchase confirmation procedure is in many ways similar to the query procedure. The main differences occur in the Bus Tier. The Bus Tier receives the SOAP request as before and once again converts the OpenTravel specification into multiple SOAP requests. In the case of a purchase, the first SOAP messages sent to the web services must be to confirm the intention to purchase a service. All web services must respond in an affirmative way. This eliminates the possibility of booking a service and finding out that due to the impossibility of booking another service the first one must be cancelled. The Bus Tier can resend requests in order to achieve availability of all services. This ends up being a sort of “hold state” where the web service providers temporarily reserve the service until a certain time passes and the service is “unreserved” or, on the other hand, the purchase is

effectively confirmed. When all web services respond affirmatively to the Bus Tier of the initial "hold", then another SOAP message can be sent from the Bus to each web service effectively confirming the purchase of the package. Only when the Bus Tier receives new SOAP responses from every web service confirming the final reservation, is the reservation official. The Bus tier then notifies the Service Tier and so on until the system user ultimately views the purchase confirmation.

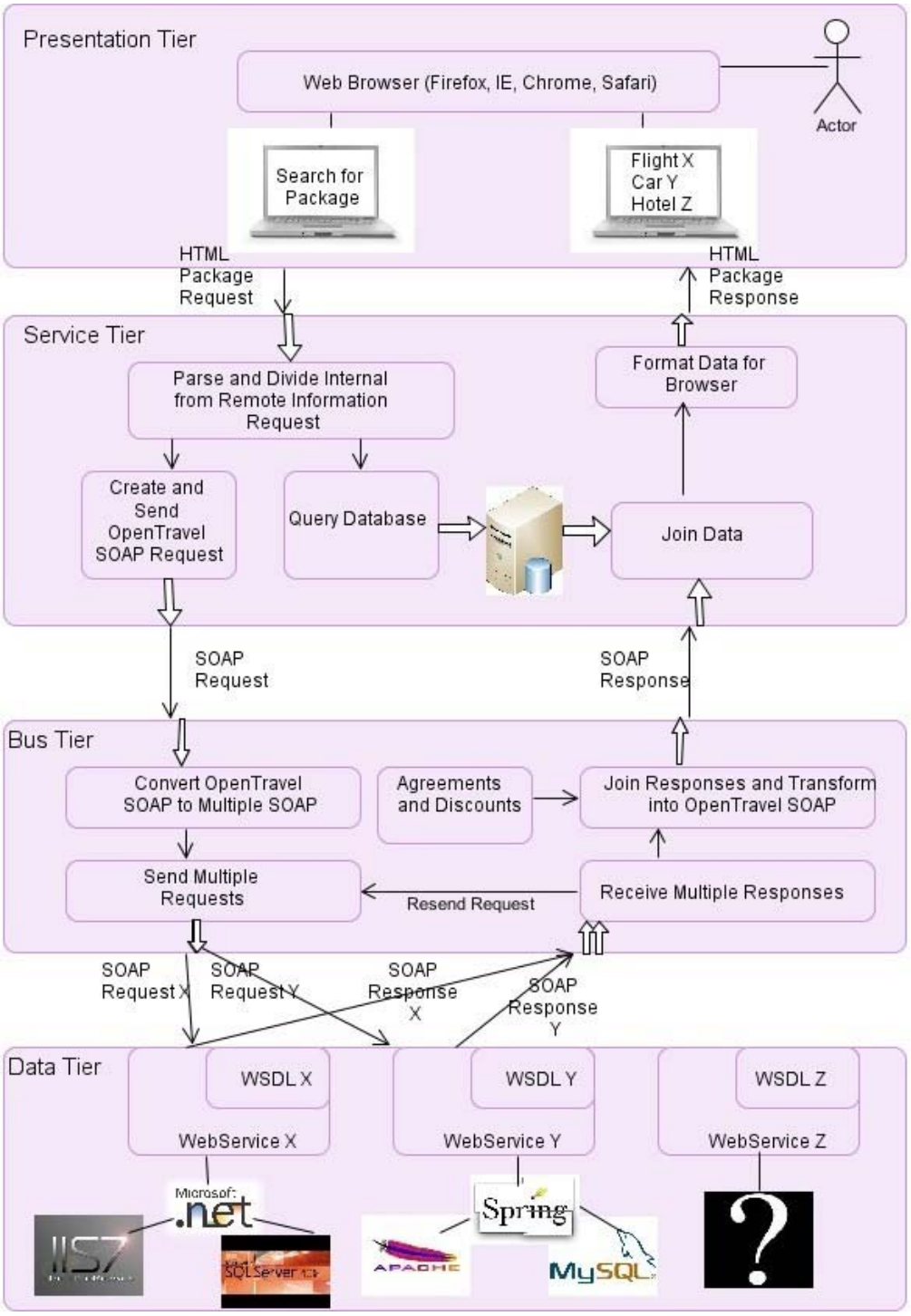

Figure 6.1- Visual System Diagram

6.2- Concept Validation

In order to develop a web application with the magnitude of the E-Tourism Information System, many resources would be needed. To accomplish this feat, the most important resource would be tied to manual labour. The software development would be time-consuming and in some cases repetitive. The sheer size of the OpenTravel specification alone would be enough to create the possibility for many lines of code to be programmed.

Since the system is projected to be an enterprise solution, it would only be fair to have a taskforce committed to its development. Unfortunately this is not possible nor is it the objective of this project. In order to validate some of the concepts and ideas behind this document, and based on a limited time frame, tests were made with an experimental model created for this purpose. It is important to have in mind that this experimental model is in no way related to the original model intended for use by the final consumer. Still, the experimental model is useful for implementing theoretical concepts that have not yet been explored. The usage of this technique enables us to demonstrate the viability of the project.

6.2.1- Development Environment

In order to create an experimental model of the E-Tourism Information System, several different technologies, programming languages and development environments where taken into account. Having in mind the demonstration nature of the model, various other options where ruled out given their complexity and learning curve. An optimal solution would be a platform framework where services were totally integrated in a single location. Fast development along with functionality was preferred over other characteristics. After all these considerations, the decision laid upon NetBeans IDE from Oracle. NetBeans IDE is a free, open-source integrated development environment (IDE) for software developers. It provides a generic framework for creating Swing applications. NetBeans IDE supports a wide range of platforms and languages such as JavaFX, Ruby, Ruby on Rails, Groovy, Grails, PHP, JavaScript, Fortran, C and C++. Other features such as fast directory-based deployment, both local or remote, XML code validation and the possibility for viewing console output from the local server also helped in developing the system in a more efficient manner.

NetBeans includes some of the most important features necessary to this project: support for Web and Enterprise Applications. Web applications can be easily built with JavaScript, PHP, JavaServer Faces and JavaServer Pages. Enterprise Applications allow the development of secure server-side Java applications that

are easily scalable and robust. These applications can later be accessed through a web browser for increased performance. Also included with NetBeans IDE is support for web services. Thanks to the web service wizards, a software developer can quickly create different web services either from existing Java classes or from WSDL files. As for one last feature, the Glassfish Application Server comes in handy. Glassfish from Sun Microsystems is for JavaEE platforms just like the one present in NetBeans. For this reason, Glassfish allows a seemless integration with NetBeans IDE and includes a subset of functionalities from OpenESB. Considering one of the E-Tourism Information Systems is based on a ESB, Glassfish also comes in handy acting as an intermediary tier of middleware to communicates between web services.

6.2.2- Development Procedure and Accomplishments

For the development of the prototype, first a web application was created to encompass the entire project. This task is quite simple but defines the ground on which the entire project will function. Although this environment is local because it is running on the same computer, the transactions that occur between the different elements contained within it can be seen as being remote, simulating the usage of services spread across a global network. This allows more realistic experiments to be made.

Inside the web application, various elements were added. The first elements created were the web services. In order to successfully create web services, NetBeans offers two approaches. The first approach is based on WSDL files. Starting from the WSDL file that specifies all the possible operations of the web service and how to access them, NetBeans is capable of automatically creating a web service that mimics these operations. This is very useful when the web services are already defined because it saves implementation time and also allows for multiple web services to be used almost transparently in the environment. These services can equally be located in a local hard drive or retrieved from a network address. Unfortunately, no WSDL file was promptly available for use in this project because these files are only made available to select entities that usually have agreements with the tourism-related companies that provide the services. To work around this , the second possible approach for building a web service was used. The second approach can be accomplished by using a java class to create the web service. This is considered to be a “bottom-up” approach to the creation of web services. This approach is suggested for creating web services from scratch as is the case. The endpoint in this case is coded in java and unfortunately is more prone to errors.

```
@WebService()
public class Tap {

    /**
     * Web service operation
     */
    @WebMethod(operationName = "getFlight")
    public String getFlight(@WebParam(name = "inputData")
                    AirCompany inputData)
{

        return ":" + inputData.getAirline()+ inputData.getArrivalAirport();
    }
}
```

Figure 6.2- Simple Java Class for Web Service Implementation

Due to the size of some procedures, it was impossible to include all of the attributes of some classes in a web service. The solution was to use only the most important attributes. Thankfully, the OpenTravel specification supports this because most attributes are not mandatory. In terms of concept validation, the number of attributes used is not as important as knowing that the specification can be freely used in terms of attributes and has the flexibility to do so.

The technique of building java classes was repeated in order to create various different web services. These web services were to be later used in experimental queries as we will see. It is obvious that additional operations can also be added to the web services created by simply editing the java classes and rebuilding the web services again. In order to make the web service fully functional, it had to be deployed in order to be available on the development environment. To deploy a web service, two containers can be chosen to "insert" it in. Depending on the choice of web service implementation, either a web container or a EJB container can be used. In our case, a web container was used. As a note, once the web service is created, a WSDL file is automatically created with it. This means that from now on the WSDL file can be used for creating a new web service for the first or for when creating the application client.

From the time the web services were "online", it was essential to connect these together. For this, we counted on the help of the Glassfish server that provided an intermediate tier. Glassfish server also provided test clients for ensuring the correct functionality of the different web services. For the orchestration of the different web services, simple logic was used for the validation of the bus tier of the E-Tourism Information System. In a corporate solution, other optimization solutions can be taken into account to increase performance. In our case, the optimization was performed based on the query made. In case a query for flight-related information was made, web services related to car rentals and other areas not linked to aviation were not called upon.

In order to make use of the different methods included in the various web services, a web service client application was also necessary to be created to perform the requested queries. When creating a client application with the presence of the WSDL file from the web service, implementation of the client could run a lot smoother. This is due to NetBeans automatically creating the skeleton of the code necessary for the client, once again saving implementation time. In a structural point of view, this web service client acts as a simple service tier in the E-Tourism Information System,. In the case of NetBeans, three different types of web service clients can be created and used. These clients can be a java class in a JavaSE application, a simple servlet or a JSP page in a web application. Considering we already had the web application created and at hand, the JSP page client was chosen. To use the JSP page, we had to copy the operation for testing to the index.jsp page of the client application. After this we manually inserted the values of the input parameters of the operation. Next, be started the Glassfish server and the application client was built and deployed. In the browser, the results of the request to the different web services were displayed. As we can conclude, the client application enabled us to see the results of different queries to the web services. Unfortunately a realistic user interface was not possible to develop due to time restrictions. Another element not implemented was the internal database in the service tier. Both of these elements have already been exhaustively recreated by others and therefore are not a priority.

6.3- System Response Time

It is very hard to estimate the average time the system can take to perform a specific task like a query or a purchase confirmation. Even in systems that are operational this value is hard to calculate due to all the external factors that exist. Ten seconds is the reasonable amount of time that a user should have to wait for a response from a web application before losing interest in this task [59]. For a response time greater than this, a countdown timer or a feedback indicator should be added to the web interface in order for the user to have an idea of the amount of time that he will have to wait.

Although most of the time values that where mentioned in 5.1.1.4 were estimations based on related data or the experimental implementation of the system created for this project, only by fully implementing the system can we have more accurate values of the total time needed to perform an entire package request. Having all these factors in mind, it is reasonable to believe that a functional implementation of the E-tourism Information System would take no more than 10 seconds to respond to the most complex requests such as purchase confirmations and queries. Other operations are expected to last considerably less time due to their nature.

7 – Final Statements and Future Work

This chapter is destined in providing a general conclusion to the work developed during the course of this project. Possible additions to this project are also proposed along with comments on future work.

7.1- Final Statements

This Project was thought, published and worked upon so that an E-Tourism Information System could gradually evolve and at the end of its' lifecycle would ultimately represent an integrated solution to the paradigm of dynamic packaging in the E-Tourism industry.

In the First Chapter, a brief introduction to the general concepts of the project was made. These concepts provided a layout for the tasks necessary to be accomplished and gave a better understanding of the objectives that needed to be fulfilled. This chapter also acted as a guideline to give meaning to the development of the E-Tourism Information System.

As for the Second Chapter, it allowed a better viewing of the environment surrounding this project. The Internet was seen as a powerful tool for exposure and Travel and Tourism were both studied in detail to understand the sector in which the system was based upon. Supply chains provided an insight view of the lifecycle of a tourism destination product and how it was seen by different players such as intermediaries or consumers. Later on, E-business was analyzed followed by E-tourism. Also in this chapter a crucial concept for the project was studied, being called dynamic package. Lastly, the e-tourist profile was analyzed and brought into account. Having all these concepts from so many different areas put together in one place gave the idea that joining them in one system would not be trivial.

In the Third Chapter, different technologies, protocols and virtual tools were analyzed for the creation of the desired E-Tourism Information System. Having in consideration the nature and the requirements of the system, decisions were made. Some limitations brought by the system also took part in the choice. Considering the system to be part of the of the WWW, HTML for presentation was the nature choice. For the transport and storage of information in a simple and efficient manner without much bandwidth allocation, XML was picked. SOAP allowed remote procedures to be invoked in a unified way over the network. Web Services allowed the standardization of the method of deploying and granting access to the relevant information related to the tourism products by different service providers. WSDL and UDDI along with the already mentioned SOAP closed the web service circle. Finally, an Enterprise Service Bus "glued" the different web services together in order for system users to use them with an interface in a clean and transparent way. In terms of learning experience, these existing and more or less known technologies were used in innovative ways in order to create the desired effect.

In Chapter 4, the OpenTravel Alliance was dissected. Information ranging from its origin, organizational structure, team members and allies were all included. More relevant to the project implementation, the OpenTravel Specification was studied. Understanding the lifecycle of the specification allowed a better judgement of how to develop a reusable and updatable system. Along with the main functionalities the specification was capable of performing, the structure and content of the different schemas files were also viewed and organized according to type. In the end of the chapter, XML files using the schemas were displayed and later integrated into the prototype used for validation of concepts.

Chapter 5 was the chapter were more accomplishments were made throughout its course. Using the original and well-known multi-tier architecture as a starting point, an additional tier of abstraction was added to the already existing Presentation , Service and Data Tier. This new tier was called Bus Tier and would be in charge of coordinating the logic between the Service Tier and the Data Tier. The newly created Bus Tier brought new challenges to the system but also provided innovative forms of connecting different web services and allowed a uniform integration with the other tiers. After this, the system was divided into packages as a form of better organizing the actors and operations that would be present in each of these domains. This allowed a global understanding of the scope of the system. Lastly, by using the OpenTravel Specification as support, the system components were identified and then organized. The system components shined a light on the parts played by each of these entities and their influence on the system. By embedding the different components into the existing packages, the system came together as what we like to call "The E-Tourism Information System for a Specific Region".

The Sixth Chapter initially provided a general idea of the system operation. The most important and complex procedures that the system supports where defined and explained with some detail. Once again it is important to have in mind that these procedures where not based on a specific technology or implementation. They should rather serve as a guideline to what these procedures should be able to do in a system like the one that was modelled. Further ahead in the chapter, the concept validation was performed by means of an exploratory prototype created for this purpose. Development procedures where explained for reference as well as the accomplishments made. At the end, an annotation was made about the system response time.

The Final Chapter, in which this text is also included, provided a brief summary of the work developed during this project and will serve as a future reference to the reader.

As a resume of what was done, in a first phase of project development, the study of concepts inherit to a destination management system was performed. The importance and role of these concepts in a DMS allowed for the conceptual structure of the desired system to be modelled. After this, the system requirements were shaped and the support frame was built. Finally, an exploratory prototype of the E-Tourism Information System was fabricated and implemented in order to serve as a demonstration. As a personal note, the most important accomplishment in this project was not only the modulation of the entire system but also the possibility to create and implement the Bus Tier. The construction and implementation of the Bus Tier allowed us to verify the validity of the whole concept described as the solution for the E-Tourism Information System and gave significance to everything that was studied, analysed and developed. All in all, relevant work was performed and it was a pity that there was not more time available to add more functionality to the system.

7.2- Future Work

Although much has been accomplished with this project, many other aspects and functionalities can be added to it. This is in part due to the nature of it, encompassing so many different areas of interest such as Travel, Tourism, Software Engineering, Information Technologies, Business Management and even Marketing.

As major additions to the system, an interface would be greatly appreciated for the users in general. Even the best application in existence loses credibility if it is unable to show its' features in a clear and concise manner. Usability should be

taken into account when building the interface. This is an important aspect because like all new ideas that arise, if they are not properly presented, preconceived opinions can emerge tainting future acceptance of the system and of the concept in general.

Another area to explore could be the internal database located in the Service Tier. Along with holding information regarding the user profile, the database and surplus resources of the system could be used for data mining. By collecting information about searches and actions performed, the system could automatically learn about user preferences and store these in the database. Later on, this information could be used for giving suggestions about tourism packages the e-tourist might be interested in or even serving as a technique for personalized publicity, adding another source of income to the system.

As one last proposal, alliances with tour operators could be made in order to enjoy access to their web services and all the information related to their business. The usage of real web services with the E-Tourism Information System would really be a great addition and help this platform take a step up into a more corporate and commercial version.

References

[1] H. Werthner, F. Ricci, *"E-Commerce and Tourism"*, Communications of the ACM, December 2004, Vol. 14, pp. 102-105.

[2] *"Internet Usage Statistics - World Internet Users and Population Stats"*, http://www.internetworldstats.com/stats.htm, [accessed November 26, 2010].

[3] H. Mowlana, *"Global Information and World Communication"*, International Communication as a Field of Study, Second Edition, October 1997, Sage Publications Ltd., pp. 11-14, 47-49.

[4] L. Kleinrock, *"Leonar Kleinrock's Home Page"*, Brief Personal Profile, http://www.lk.cs.ucla.edu, [accessed November 26, 2010].

[5] L. G. Roberts, *"Dr. Lawrence G. Roberts"*, Papers by Lawrence G. Roberts, http://www.ziplink.net/users/lroberts/, October 24 1999, [accessed November 26, 2010].

[6] D. Comer, *"Computer Networks and Internets with Internet Applications"* Fourth Edition, 2004, Prentice Hall- Upper Saddle River, pp. 18-21.

[7] C. R. Goeldner, J. R. Ritchie, *"Tourism, Principles, Practices and Philosophies"*, Ninth Edition, September 20 2002, Wiley Publisher.

[8] L. Cunha, *"Introdução ao Turismo"*, Editora Verbo, Lisboa, 2001, pp. 201.

[9] R. Carter, F. Bedard, *"E-Business for Tourism- Practical Guidelines for Tourism Destinations and Businesses"*, September 2001, World Tourism Organization, pp. 16-18.

[10] Tourism HR Society, *"Travel Services"*, http://www.go2hr.ca/BCsTourism brIndustry/TourismSectors/TravelServices/tabid/216/Default.aspx, The Resource for People in Tourism, [accessed November 26, 2010].

[11] J. T. Cobb (University of Florida), D. Whitten (Purdue University), *"E-Business: Architecture Overview and Some Issues for Higher Education"*, 10 October 2001, EDUCAUSE Annual Conferences.

[12] Sial, Shailendra, "*Types of E-Commerce*", http://ezinearticles.com/?Types-of-E-Commerce&id=2002490, February 16 2009 [accessed November 26, 2010].

[13] J. R. Andam, "*E-commerce and E-Business*", May 2003, How big is the SME e-Business Market, e-Asean Task Force, pp. 27-31.

[14] R. Möller, *"E-Commerce: Concepts & Technologies"*, November 2004, E-Commerce Reference Model, pp. 16-23.

[15] P. Allen, *"Realizing e-Business with Components*", First Edition, Chapter 7, October 25 2000, Addison-Wesley Professional.

[16] M. Fux, D. S. Mathieu, F. Go, *"Integrated Electronic Customer Relationship Management for Alpine Tourist Destinations: Drivers and an Adoption Framework "*, May 2004, Institute of Information Systems- University of Berne Switzerland.

[17] J. T. Mentzer, *"Defining Supply Chain Management",* 2001, Journal of Business Logistics, Volume 22, Nr^{o}2, pp. 1-25.

[18] D. Amor, *"The E-Business (R)evolution- living and working in an interconnected world",* First Edition, September 15 2001, Sub-chapter 1.2: Defining E-Business, Prentice Hall.

[19] R. Tapper (Environment Business & Development Group), X. Font (Leeds Metropolitan University), *"Tourism Supply Chains- Report of a Desk Research Project for The Travel Foundation",* January 31 2004, Leeds Metropolitan University, pp. 5-11.

[20] T. Selänniemi, J. Sweeting, *"Supply Chain Management for Tour Operators- A handbook on integrating sustainability into tour operators' supply chain systems"*, Final Version, 24 November 2003, Tour Operators' Initiative for Sustainable Tourism Development, pp. 10-12.

[21] Rainforest Alliance, *"Sustainable Tourism- Be a Sustainable Traveler",* 2010, http://www.sustainabletrip.org/tourism, [accessed November 26, 2010].

[22] H. Srinivas, *"The Sustainable Tourism Gateway- What is Sustainable Toursim?"*, http://www.gdrc.org/uem/eco-tour/st-whatis.html, [accessed November 26, 2010].

[23] G. Inkpen, *"Information Technology for Travel and Tourism"*, Second Edition, July 30 1998, Longman Publisher.

[24] World Tourism Organization, *"Tourism – an Economic and Social Phenomenon",* http://www.unwto.org/aboutwto/why/en/why.php?op=1, [accessed November 26, 2010].

[25] GO2 Resource for People In Tourism, *"Tourism Industry Overview"*, 2010 http://www.go2hr.ca/BCsTourismbrIndustry/WhatisTourism/tabid/71/Default.aspx, [accessed November 26, 2010].

[26] V. T. C. Middleton, A. Fyall, M.Morgan, A. Ranchhod, *"Marketing in Travel and Tourism"*, Fouth Edition, 2009, Butterworth-Heinemann, pp. 2-18, 20-24, 336-354.

[27] World Tourism Organization, *"Collection of Tourism Expenditure Statistics"*, 2007, pp. 3-5.

[28] Capricorn Tourism and Economic Development, *"Inbound Vs. Domestic Markets- What is Inbound Tourism?"*, http://www.capricorntourism.com.au/industry-development/starting-a-tourism-business/inbound-vs-domestic-markets/, [accessed November 26, 2010].

[29] M. Lasar, *"The World gets a National Broadband Plan"*, September 2010, http://arstechnica.com/telecom/news/2010/09/the-world-gets-a-national-broadband-plan.ars, [accessed November 26, 2010].

[30] The Associated Press, *"Tourist Activity Endangering Machu Picchu"*, July 2 2008, http://www.cbsnews.com/stories/2008/07/02/travel/main4227137.shtml, [accessed November 26, 2010].

[31] World Tourism Organization, *"UNWTO World Tourism Barometer- Committed to Tourism, Travel and the Millennium Development Goals"*, November 2010, Volume 8, Issue 3, UNWTO Publisher.

[32] Tour Operator Initiative, *"Sustainable Development of Tourism- Conceptual Definition"*, June 23 2010, http://www.toinitiative.org/index.php?id=82, [accessed November 26, 2010].

[33] Business Corporate Travel Intelligence, *"NBTA Report Puts Global Business Travel Spending At Nearly $1 Trillion"*, July 24 2009, http://www.btnonline.com/businesstravelnews/headlines/article_display.jsp?vnu_content_id=1003997165, [accessed November 26, 2010].

[34] World Travel and Tourism Council, *"Recovery Stronger than Expected but Likely to Slow Down in 2011"*, 9 November 2010, http://www.wttc.org/, [accessed November 26, 2010].

[35] K.C. Laudon, J.P. Laudon, *"Management Information Systems: Managing the Digital Firm"*, Tenth Edition, 2007, Prentice Hall.

[36] B. Farbey, A. Finklestein, "Exploiting Software Supply Chain Business Architecture: a Research Agenda", University College London – Department of Computer Science.

[37] J. Borrelli, *"The Intelligent Supply Chain"*, Reverse Logistics Magazine, Edition 16- Jan/Feb 2009.

[38] Jean-François Baylocq, *"Meeting on the Trade and Development Implications of Tourism Services for Developing Countries"*, 19-20 November 2007, UNCTAD XII, Geneva.

[39] C. Kim,*"E-Tourism: an Innovative Approach for the Small and Medium-Sized Tourism Enterprises"*, 2004, Kyunghee University, Korea.

[40] A. Shikida, M. Yoda, A. Kino, M. Morishige, *"Tourism Relationship Model and Intermediary for Sustainable Tourism Management- The Importance of Values in Sustainable Tourism"*, Hokkaido University, Japan.

[41] Char-lee Mclennan, L. Ruhanen, *"Analysis of National, State, Regional and Local Tourism Strategies and Plans- Identification of Key Strategic Issues"*, Austalian Government Initiative.

[42] S. Choi, X. Y. Lehto, J. T. Oleary, *"What Does the Consumer Want from a DMO Website? A Study of US and Canadian Tourists' Perspectives"*, 2007, International Journal of Tourism Research, Issue 9, pp.59-72.

[43] Federation of Tour Operators, *"Tour Operators and Travel Agents- An explanation of the roles and functions of tour operators and travel agents", 2006,* http://www.fto.co.uk/operators-factfile/tour-operators/,[accessed November 26, 2010].

[44] S. Das (Consulting & Valuation Analysis), *"Global Distribution Systems in Present Times- Four Major GDS Systems"*, October 2002, HVS International.

[45] A. Fyall,B. Garrod, *"Tourism Marketing: a Collaborative Approach"*, Aspects of Tourism, 22 Dec 2004, Channel View Publications, Illustrated Edition, pp.43-45.

[46] R. Brau ,A. Lanza, F. Pigliaru, ,*"How Fast Are the Tourism Countries Growing? The Cross-country Evidence"*, September 2003, FEEM Working Paper No. 85.

[47] Y. Malhotra,*"Enabling Next Generation E-Business Architectures: Balancing Integration and Flexibility for Managing Business Transformation"*, June 2001, Intel E-Business Strategy White Paper, BRINT Institute.

[48] Sun Microsystems, *"The Banking Application- Distributed Application Architecture"*, Chapter 7, June 2006.

[49] Trisept Solutions, *"Making Sense out of Dynamic Packaging"*, http://www.triseptsolutions.com/Dynamic_Packaging.pdf, [accessed November 26, 2010].

[50] C. Eite, *"Dynamic Packaging"*, e-Conference, November 9 2006, Muscat, Oman, UniSYS.

[51] P. Murphy, M.P. Pritchard, B. Smith, *"The Destination Product and its Impact on Traveler Perception"*, 1999, Tourism Management 21, Elsevier Science Ltd.

[52] D. Garlan, M. Shaw, *"An Introduction to Software Architecture- "*, January 1994, School of Computer Science- Carnegie Mellon University, pp. 4-11.

[53] W. W. Eckerson, "*Three Tier Client/Server Architecture: Achieving Scalability, Performance, and Efficiency in Client Server Applications*", January 1 1995, Open Information Systems.

[54] A. Dickman, "*Two-Tier Versus Three-Tier Apps*", November 13 1995, Information Week 553, pp. 74-80.

[55] A. O. Ramirez, M. Richardson, *"The Three-Tier Architecture- An overview of a Mainframe System"*, Jul 01, 2000, The Linux Journal, Issue nr° 75

[56] Business-Software, *"Why Business Intelligence Software Is Essential to Business Performance Management"*, 2010, http://www.business-software.com /article/business-performance/why-business-intelligence-is-essential-for-bpm, [accessed November 26, 2010].

[57] D. Sprott and L. Wilkes, *"Understanding Service-Oriented Architecture"*, January 2004, http://msdn.microsoft.com/en-us/library/aa480021.aspx, Microsoft CBDI Forum, [accessed November 26, 2010].

[58] P. Obermeyer, J. Hawkins, *"Object Serialization in the .NET Framework"*, August 2001- Updated March 2002, http://msdn.microsoft.com/en-us/library/ms973893.aspx, Microsoft Corporation, [accessed November 26, 2010].

[59] J. Nielsen, *"Usability Engineering"*, Chapter 5, published by Morgan Kaufmann, 1993, San Francisco.

[60] S. Biaz, N. H. Vaidya, *"Is the Roundtrip Time Correlated with the Number of Packets in Flight ?"*, 2003, Dept. of Electrical and Computer Engineering, and Coordinated Science Laboratory- University of Illinois.

[61] Burleson Consulting, *"Measuring Oracle Response Time- Oracle Response Time Instrumentation"*, Oracle Tips, December 29 2008, http://www.dba-oracle.com/shadow_oracle_formspack.htm, [accessed November 26, 2010].

[62] J. Aikat, J. Kaur, F. Donelson Smith, K. Jeffay, *"Variability in TCP Roundtrip Times"*, National Science Foundation, September 23 2003, Department of Computer Science-University of North Carolina.

[63] Open Travel Alliance, *"OpenTravel- Setting the Standard"*, 2010, http://www.opentravel.org/, [accessed November 26, 2010]

[64] Open Travel Alliance, *"OpenTravel Alliance Procedures Manual"*, June 2008, Version 6.0.

[65] Open Travel Alliance, *"About the OpenTravel Alliance"*, 2010, http://www.opentravel.org/AboutUs/Default.aspx, [accessed November 26, 2010].

[66] V. Perini- Executive Director of Open Travel Alliance, *"Why Join?"*, 2010, http://www.opentravel.org/Join/Default.aspx, [accessed November 26, 2010].

[67] Open Travel Alliance, *"Member List"*, 2010, http://www.opentravel.org/Join/Default.aspx, [accessed November 26, 2010].

[68] D. Morley , S. Adkin, J. Lambe, K. Camenzuli, A. Leveillé, *"OTA Architecture Review"*, Advisory Forum, 2006, Orchestrating the Total Travel Solution, pp. 8-25.

[69] World Wide Web Consortium, *"W3C Mission- The Principles"*, 2009, http://www.w3.org/Consortium/mission.html, [accessed November 26, 2010].

[70] World Wide Web Consortium, *"HTML 4.01 Specification"* 24 December 1999, http://www.w3.org/TR/html4/, W3C Recommendation, [accessed November 26, 2010].

[71] D. Connolly, "*HyperText Markup Language Specification -- 2.0*", September 28 1999, http://www.w3.org/MarkUp/html-spec/index.html, [accessed November 26, 2010].

[72] T. Berners-Lee, *"Tim Berners-Lee"*, May 30 2010, http://www.w3.org/People/Berners-Lee/Longer.html, [accessed November 26, 2010].

[73] Open Travel Alliance, "*Online Schema*", 2010, http://www.opentravel.org/Specifications/OnlineXmlSchema.aspx, [accessed November 26, 2010].

[74] J. Lambe (OpenJaw Technologies), *"OpenTravel Schemas- Getting Started"*, February 5 2008, OpenTravel European Conference.

[75] J. Turato (OpenTravel Board of Directors Vice-President), *"Welcome"*, February 5 2008, OpenTravel European Conference.

[76] OpenTravel Alliance ,*"OpenTravel Alliance Revised Namespace and Versioning Specification"*, Version 1.0, May 21 2003, http://xml.coverpages.org/OTA-NamespaceAndVersioning2003A.pdf, [accessed November 26, 2010].

[77] E. R. Harold, W. S. Means , *"XML in a Nutshell : A Desktop Quick Reference (Nutshell Handbook)"*, January 15 2001 , Chapter 2 and 5, O'Reilly Media, 1st edition.

[78] B. Marchal, *"XML by Example - Building E-Commerce Applications"*, 1st Edition, December 14 1999, QueCorp Publisher.

[79] N. Quaine, *"SOAP Basics- What is SOAP?"*, 2007, http://www.soapuser.com/basics1.html, [accessed November 26, 2010].

[80] E. Wilde, *"Web Services Part I - Web-Based Services"*, October 4 2006, UC Berkeley iSchool, http://dret.net/lectures/services-fall06/webservices1, INFOSYS 290-3, [accessed November 26, 2010].

[81] E. Christensen, F. Curbera, G. Meredith, S. Weerawarana, *"Web Services Description Language (WSDL) 1.1"*, W3C Note, March 15 2001, http://www.w3.org/TR/wsdl, World Wide Web Consortium, [accessed November 26, 2010].