Integrating data in a Web and mobile Web application from Movie Theaters' schedules

By: Alberto Verza Salomón

Mentor: Ana María Iglesias Maqueda

Introduction

This work studies the main methods and techniques for data integration [1]. Specifically, how to extract data coming from different and heterogeneous external sources is studied. Moreover, how to integrate and show it in a Web application is analyzed.

Then, a proof of concept concerning these methods and techniques will be materialized into a web site application where different kinds of data sources are used in order to achieve the following goal: to provide to the end user an easy and integrated way to access the heterogeneous data coming from different sources in a transparent manner. Note that these data could have different original location and format.

The Web application domain for the proof of concept is focused in *Movie Theaters' schedules* in Spain. This domain specially fits for the proof of concept of this study because there is a lot of scattered data around the World Wide Web about it: performances, cinema locations, films' details, user opinions, etc. Then, the user sometimes is not able to find the complete schedule, or to locate the physical location of the theaters, etc. Therefore, to provide an application for summarizing all these data coming for different Web services, Web pages or Databases, among others, it is really useful for the final users.The Web application will be compatible with both PC and mobile devices. The functionality will be the same between these two versions, but their interfaces will be adapted to their respective devices.

Goals

The following goals are set in this Project to be reached:

- **Take data from different sources and show them under the same interface:** this is the main goal of the Project. The application must extract information from multiple data sources, heterogeneous, with different nature and format: web pages, JavaScript files, plaintext files, databases, Web Services, APIs, etc.. This data should be integrated in a local storage in order to make the application more efficient, applying transformations as needed, in order to display all this information under the same homogeneous interface: CineSpain web site.

- **Develop an application adapted to PCs and mobile devices:** thanks to recent web technologies, it is possible to make the application to distinguish between PCs and mobile devices, and serve a PC or mobile version of the website accordingly. This level of compatibility must be reached, taking into account the big increase that mobile devices expansion is suffering in these last years.

As an optional goal, **reach an acceptable level of usability for the users:** Usability [1] is a desirable property which depends in great measure on user opinions. The evaluations are very subjective, and it is tremendously difficult to satisfy all the users. That is why in order to obtain information about the satisfaction level concerning the website, some users with different navigation skill profiles have been surveyed.

State of the Art

Data integration [2] is a technique employed to build systems which are capable of sharing and accessing to information provided by multiple data sources.

The main goal of this discipline is to make data integration systems able to offer homogeneous access to data, where data is coming from a set of heterogeneous sources. Key aspects like nature, quantity of sources, heterogeneity and system autonomy must be considered, explained and justified in this kind of systems.

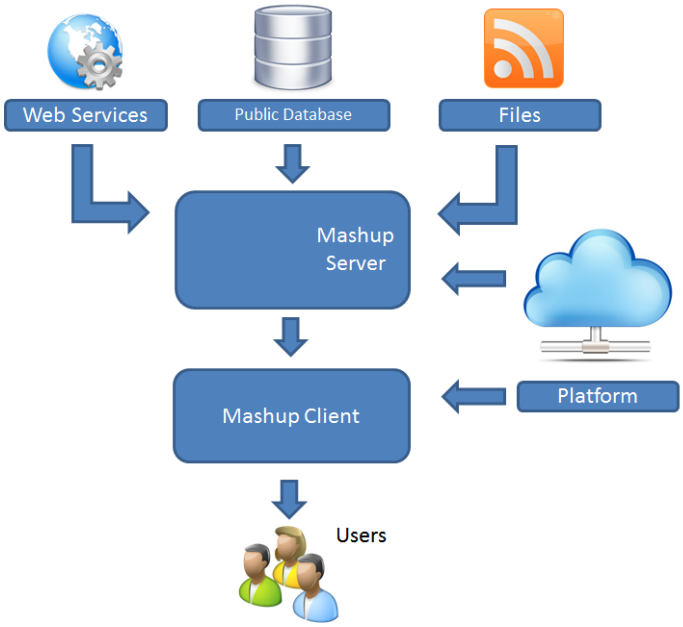
Integration architectures

The most prominent architectures are known as *warehousing* and *virtual integration [3].*

In **warehousing** architecture, data coming from sources is loaded, transformed and stored in a physical database for its next use. The transformation step can be taken before storing or after storing (in this last case, transformation functions are performed by the database engine).

In **virtual integration** architecture, only needed data is recovered from the sources, so petitions are covered by real time responses.

Integrating data in the Web



Integrating data in the Web gives as a result the so called "Mashups", which are applications that show integrated data unified in a Web interface. The typical Mashup architecture is compounded by: the data sources, the server, the client, and the platform which contains the service (see Figure 1).

Figure 1. The typical Mashup architecture.

Prior to homogenize all the employed sources' formats, most Mashups insert information as registers into a local database, whose structure is the same for all the inserted data. By this way, the Mashup's server takes data from the local database and displays them through the client side.

The challenge that suppose integrating data in the Web resides in the Wrappers construction. There are four main categories to solve this problem: manual, learning, automatic and interactive.

Data integration applications for general purposes

Data integration applications make code generation automatically, so they are a great way to save time and effort when creating integration systems. According to their approach, they are classified as **ETL** (Extract, Transform, Load), **ELT** (Extract, Load Transform), and **EII** (Enterprise Information Integration) tools [3].

While ETL and ELT are warehousing oriented procedures, EII is ready for the virtual integration architecture. This fact gives, for the one hand, it the benefit of offering real time information and the suppression of data redundancy management. For the other hand, it has some disadvantages like slower response times, lesser data transformation capabilities and littler data volume (because of traffic limits).

As data integration Software for general purposes, this work highlights *Oracle Warehouse Builder*, a commercial ETL tool; *Talend Open Studio*, open source which supports ELT and ETL; and *InfoSphere Federation Server*, a commercial module developed by IBM for EII data integration.

Data integration tools for Mashups

There are a wide range of tools focused on making Mashup [4, 5] applications, wich are known as CASE (Computer-Aided Software Engineering) tools, because they automate some process steps related to information systems.

More details are given about the following tools:

- **Yahoo Pipes:** it is a free web tool offered by *Yahoo!* whose objective is to aggregate, to manipulate, and to recollect content related to the Web.
- **Microsoft Popfly:** it was a web based tool used to make and share various kinds of applications like web pages, Mashups and casual games.
- **IBM Mashup Center:** it is a platform for Mashup creation using management, configuration and security options.

Discussion and tools comparative

The following chart shows a comparative among all the mentioned platforms in order to study which of them is the most suitable for CineSpain project.

		Oracle Warehouse Builder	Talend Open Studio	InfoSphere Federation Server	Yahoo Pipes	IBM Mashup Center
Procedure		ELT	ETL	EII	Mashup	Mashup
Compatible data sources	**Plaintext**	Specific	All	All	Structured	Structured
	Databases	Oracle, compatibles	Wide compat.	Wide compat.	No	Wide compat.
	Web Services	SOAP + REST	Based on WSDL	Basados on WSDL	REST	Based on WSDL
Customization options		Yes	Yes	Yes	No	No
Difficulty		High	High	Very High	Low	Medium

Chart 1. Data integration tools comparative.

CineSpain requires compatibility with data sources like plaintext (structured and unstructured), REST Web Services, and MySQL databases.

Regarding this information, the best suitable platform for our application's needs would be *Oracle Warehouse Builder*, because it covers all the mentioned requirements for the actual developing application environment. However, for this work, the implementation process will be manual due to the fact that it is one of the users' requisites, so data integration concepts will be tightly assumed and learning time for these integration tools won't be needed.

Analysis

The client needs to offer an online service which shows the movie theaters' billboards from Spain, and which locate these buildings in a geographic map. Although there are multiple services to check the schedule in the Web nowadays, none of them offers the possibility of exploring cinemas inside all over the country at the same time (usually they require a previous province selection, and only those ones are shown).

The application will display search results after completing a step-by-step guided process. Then the user will have to choose one film from the result set. After that, the Web will show user's comments, relevant information, geographic building location and the schedules, all related with the selected film.

Users will be able to register as application users if they want to. Registered and identified users will have a set of benefits. In this first stage of the project, they will have access to their consulted films history and to archived films which are no longer shown.

This service will be compatible with PC and mobile devices by adapting contents to this platforms according to their specifications. The name assigned to the application and to the whole project is *CineSpain*.

Acceptable alternatives

In order to find acceptable alternatives for the application, four attributes which should be covered must be taken into consideration:

- **Range:** kind of devices which can access to our application (PCs, mobiles, Tablets…)
- **Compatibility:** once that a kind of device is selected, the number of systems that are compatible with our application.
- **Installation:** easiness of installation and configuration by the user.
- **Costs:** financial value that this alternative development supposes relative to the others.

The three analyzed alternatives were:

1) PC application and mobile application both with maximum compatibility.
2) PC application and mobile application both with basic compatibility.
3) Web application.

The **Web application** obtained the highest rank at the end of the comparative, so this is the one to be developed.

Design

The design process begins with the definition of the **use cases:** find showing films, new user registration, and search history consult.

The first designed part is the database, which must contain information about users, historical data, films, movie theaters and movie theaters' times: *User, History, Film, Archived film, Showed film, Movie Theater* and Schedule.

On the other hand, the web site should contain the following web pages:

- **Index.php:** this page appears just when one accesses to the application. It contains search criteria fields (film title, genre, director and cinemas) and the search button into the form.
- **Paso2.php:** this webpage is related to the second step in the search process. It shows the list with the matching results according to specified criteria in the previous step.
- **Paso3.php:** this webpage is related to the third step in the search process. It shows details about the film selected in step 2. It also shows user comments from Twitter and a geographic map. Both elements appear in this page when the active version is the PC version. In the mobile version, specific pages have been enabled to see this information (**cines.php** and **tweets.php,** respectively).
- **Entrar.php:** it contains the form with the user and password fields. Users must enter their data here if they want to identify themselves as registered users. This page also contains an option to go to the register form.
- **Registrar.php:** it shows the required fields to complete the register process and to create the user account.
- **Confirmado.php:** it shows the user provided data in the registration form, which now are linked to its new account.
- **Panel.php:** it is the user area, also called 'Mi cuenta' inside the application. Users can check their linked data here, and also they can consult their history directly in the PC version. In the mobile version, an option replaces the history panel that redirects the user to **historial.php**.
- **Results2.php:** this page shows information about archived films. These ones are no longer shown by cinemas.

Related to the data integrated in the Web or Mobile-Web application, different data sources were selected. The first one is the **web page** *http://entradas.com*, from where all information about the films, the cinemas and the schedule is extracted. The data format doesn't come only as structured files, but also as unstructured files (which need a bit more of work while processing).

The second group of sources are **REST Web Services,** in particular the geographic location service from the *Google Maps API v3*, and the user comments recovery from the *Twitter API REST v1.1*.

The third group are the **databases**. CineSpain works with two of them: an external database called *Sábado* supported by CESyA ("*Centro Español de Subtitulado y Audiodescripción*), and a local database. The former is used to extract details about archived films. The latter is the store where data is homogenized prior to make queries and displaying obtained results in the website.

At last, the application needs another functions which don't depend directly on the data sources. These functions are **film recommendations** that shows a recommended film based on the user's history at step 2, and **device detection**, that gives the application a functionality to distinguish accesses from PCs and mobile devices.

Evaluation

The evaluation process have been divided in two different sections. The first one is the *system functionality evaluation*, designated to check critical errors while navigating and displaying information. To make the necessary test cases, the tool **Selenium IDE** is used for this task. It is a Mozilla Firefox plugin that brings its own development environment.

Operations in Selenium IDE are very simple: test cases are programmed using a helpful command called 'Record', which memorizes manual navigation sequences from the browser. However, it is not complete enough: web content checks must be added manually since the Record function cannot do it by itself. Once the tests are made, they must be executed to fix possible imperfections. The test cases that have been passed in this application are: *search without any specified criteria, search by title, genre and director, search by cinema, user registration, sign in and sign out, history access, and archived films access.*

The second part in the evaluation process is the *user evaluation*. A usability test has been created for both PC and mobile Web versions. Users with different navigation skill

levels (*unexperienced*, *medium* and *expert*) have been surveyed. For each question a score from 1 to 5 can be assigned using the Likert scale [6].

In general, the users surveyed were satisfied with the applications. An important limitation in the user evaluation is that only three persons answered the survey (one of each skill level). Therefore, non-significant results could be extracted from the evaluation except the open-questions were the users could recommend us some application aspects to be improved, like including more selection criteria or re-organizing the history panel in the PC version. Then, to carry out a user evaluation with more users in order to extract significant data from it is proposed as future work.

Conclusions

After finalizing this project, the acquired level in the use of the multiple employed technologies has raised noticeably. Treating with application problems to find the best solutions has been the main reason.

The aims of the project have been accomplished, as it's explained next:

- **Take data from different sources and show them under the same interface:** the application is capable of extracting data from web pages, JavaScript files, plaintext files, databases, Web Services and third-party APIs, displaying all this information in a single and homogeneous interface.

- **Develop an application adapted to PCs and mobile devices:** compatibility between these two devices has been reached by developing two different interfaces, each one adapted to its respective device.

Future works

CineSpain application could be improved by adding more data sources and functions. Some actions that could be taken in order to make the application better are described below:

- **Comments cache:** with this improvement, the application needs a database restructuration to store recovered comments from Twitter, and thereby avoiding restrictions like the '180 petitions every 15 minutes' one.
- **Increasing the number of sources:** new data origins could be added so that complementary information would be displayed. Data redundancy is also another option to make.
- **Getting a better recommendation system:** in the actual state, recommendations are based in the user's history exclusively. This system could be improved if histories with similar profile are merged when recommending.
- **Administration Back-office:** a separated private access application for administration purposes could be made to manage CineSpain website.
- **Website Accessibility [7]:** it is important to make the navigation easier for all users. If accessibility rules from the *Web Accessibility Initiative* are implemented, there will be more users that browse through the web without difficulties.
- **Website Usability:** starting from the survey that was presented in the Usability evaluation, a 100 people sample must be taken to re-evaluate this quality attribute in a more precise and serious way. Data analysis should be done by using statistic models like variance analysis over data intervals.

References

[1] Jakob Nielsen. Usability 101: Introduction to Usability. Nielsen Norman Group. Disponible en: http://www.nngroup.com/articles/usability-101-introduction-to-usability/[2] AnHai Doan y Alon Halevy, “What Is Data Integration?”, “Data Integration Architectures”, en *Principles of Data Integration,* Ed. Morgan Kaufmann, Massachusetts: Elsevier, 2012.[3] Matt Casters y Roland Bouman, “ETL Primer”, en *Pentaho® Kettle Solutions: Building Open Source ETL Solutions with Pentaho Data Integration*, Ed. Wiley, Indiana: Wiley Publishing, Inc., 2010.

[4] Oswald Campesato y Kevin Nilson, "Mash-Ups and Search Technology," en *Web 2.0 Fundamentals*, Ed. Sudbury, Massachusetts: Jones and Bartlett Publishers, 2011, pp. 305-321.

[5] “Enterprise Mashups: The New Face of Your SOA". SOA World Magazine. Available at: http://soa.sys-con.com/node/719917

[6] Allen, Elaine and Seaman, Christopher (2007). "Likert Scales and Data Analyses". Quality Progress, 2007, pp. 64-65.

[7] “Guía breve de Accesibilidad Web”. World Wide Web Consortium (W3C). Available at: http://www.w3c.es/Divulgacion/GuiasBreves/Accesibilidad