# AC52048: ETL Project

## Overview

The purpose of this project is to read Internet log files into a data warehouse and be able to point a data visualizer at it to produce analytical graphs. You should use Microsoft Integration Services (or any other tool of your choice) to load the data from the files into a set of dimension tables and one (or more) fact table(s) that are held in a relational database engine of your choice. From there you may, if you so desire, move the data into any other structure and storage engine that you like.

Whatever you do with the data, you should end up with a system in which you can point a visualisation tool at the data and allow a used to plot the measures against the dimensions.

So you might, for example, move the data to SSAS, create a MOLAP cube and point Tableau at that. Of you might leave the data as fact and dimension tables in the relational engine and simply point a visualisation tool at that.

You must draw a Sun model of the design, that should be handed in on 7th March.

You will give a 15 minute demo of the project in the last week of March..

## More Detail

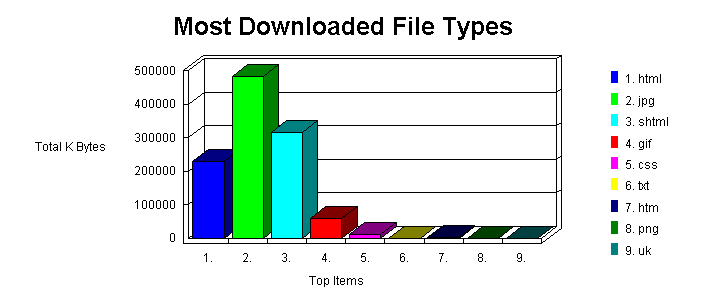
You are provided with log files from Internet Information Services (IIS) which a company needs to analyze. Currently they are using two different applications that read the log files and generate graphs. However this offers them no flexibility in their analyses (and it doesn’t help that the packages are no longer supported). They have decided that they need a data warehouse to do the analysis, the log files they have and are still generating will be read into the data warehouse. You need to Extract and Transform the data you have been given and then load it into Microsoft analysis services or the analytical engine of your choice.

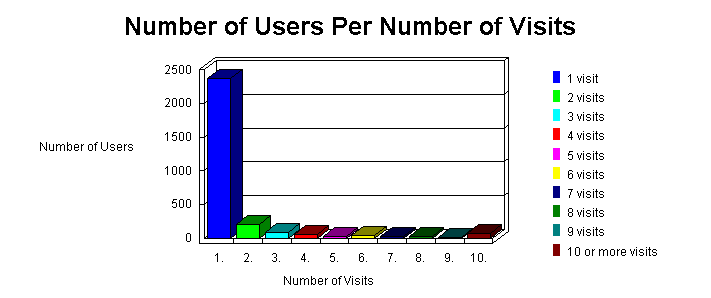
The company has provided you with a current set of graphs, charts and reports from their current applications which your warehouse will need to support. In addition they have provided you with a small wish list of requirements your warehouse will need to support:

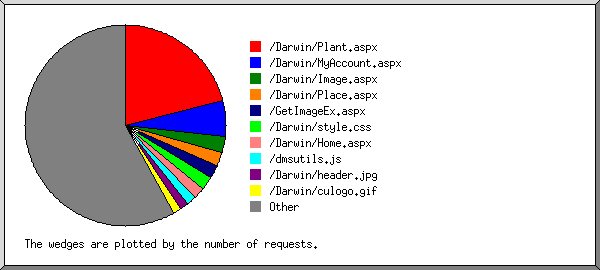
1. Turn source ip of the client (cs-username in the text file) into Geolocation, extract the postcode and find the city, state (area) and country. You can do this using online IP to geolocation services such as: <http://www.ip2location.com/> and Yahoo geo code (<http://developer.yahoo.com/maps/rest/V1/geocode.html>) or the newer http://developer.yahoo.com/geo/placefinder/.
2. The log files also contain visits from web crawlers such as Google and Bing. However the company needs to split their analysis into real visits and visits from crawlers. Fortunately, all well behaved web crawlers will begin searching your site by asking for file called robots.txt (you’ll find examples in the log files provided). You can assume that if an IP requests the file robots.txt then all other requests from that IP are from the same crawler.

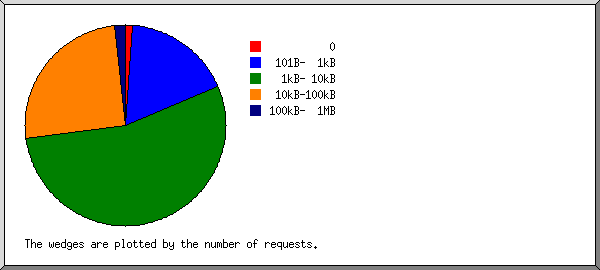
**Note** The following graphs are indicative, you don’t need to generate graphs that look like these, but you should be able to point Spotify or something similar at your cube and generate these graphs.Charts, Graphs and reports:

### File Access





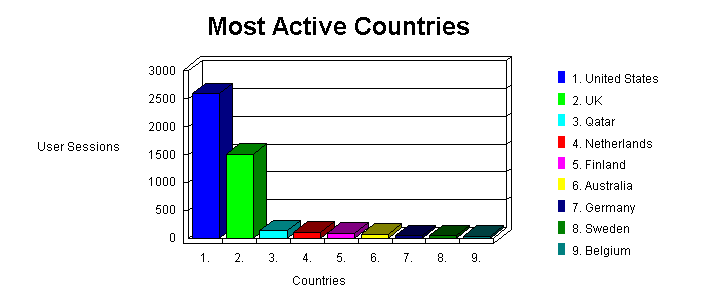
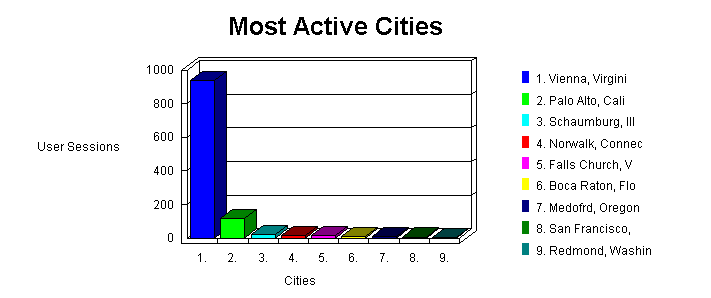




File type report and bytes served.

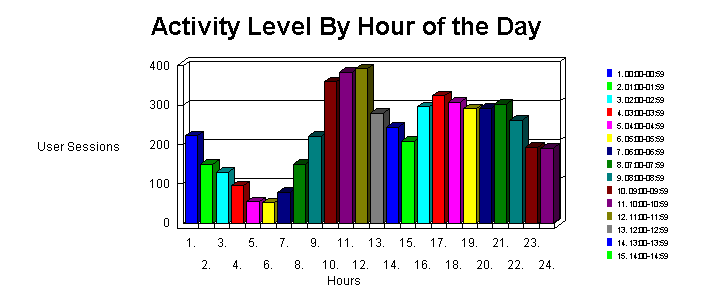
| **reqs** | **%bytes** | **extension** |
| --- | --- | --- |
| 35705 | 81.15% | .aspx |
| 30654 | 14.06% | .jpeg [JPEG graphics] |
| 2526 | 1.93% | .jpg [JPEG graphics] |
| 2170 | 1.43% | .css [Cascading Style Sheets] |
| 8336 | 0.49% | .gif [GIF graphics] |
| 2001 | 0.40% | .js [JavaScript code] |
| 1906 | 0.29% | [directories] |
| 2562 | 0.24% | .png [PNG graphics] |
| 2 |  | [not listed: 1 extension] |

### Geolocation

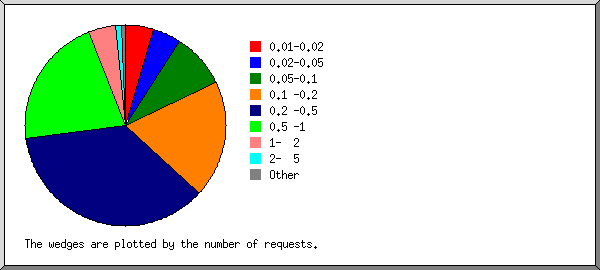
 

### Time and Date

### 



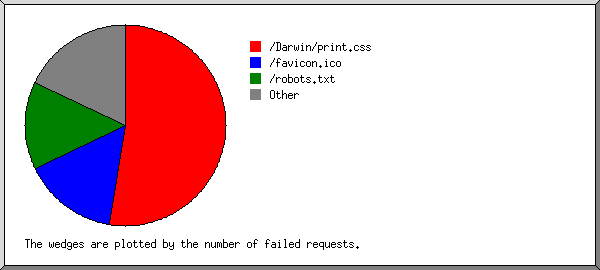
Processing time report (time-taken in table)



### Errors

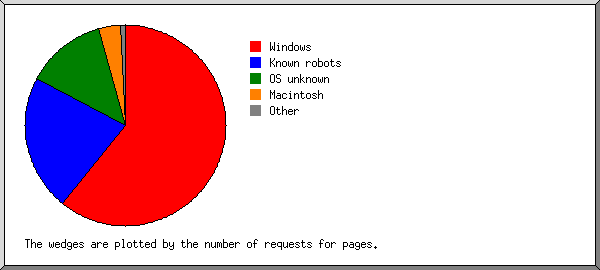
### 

Failure report: *This report lists the files that caused failures, for example files not found.*

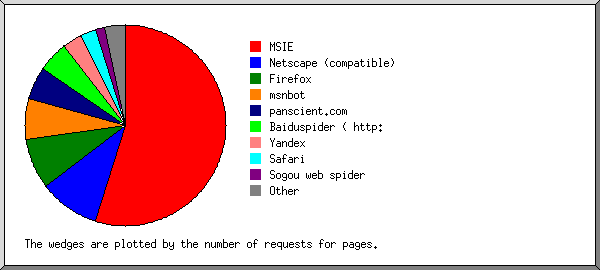


### Client information:

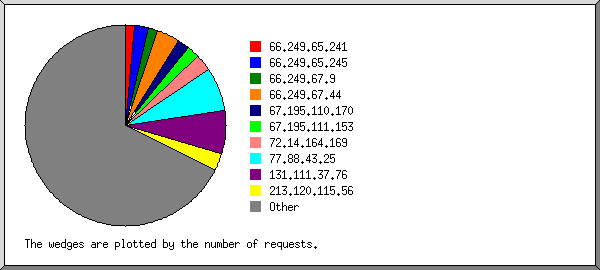
Client OS



Browser summary



Hosts report.



Referrer report, where our visitors came from

