Sessionizing

Overview

Write a program that is initialized with data about page views of visitors to web sites and provides info about visitor sessions.
You can select which programming language to use.

Session and Session Length
Page view is a single page of a site that was visited by a visitor at some timestamp.
Session is a group of page views of a single visitor to a single site such that the time between every two successive page views is not longer than 30 minutes.
Session Length is the time difference between the first and last page views of a Session.

Note: A Session may contain only a single page view. In this case the Session Length is zero.

Input

The program is initialized with data about page views. The data is provided as CSV files, where each line has the following fields of a single page view:
visitor_id - a unique identifier of the Visitor
site_url - the main URL of the visited web site
page_view_url - the URL of the visited page
timestamp - the timestamp of the page view in Epoch time (seconds since 1/1/1970).

E.g.
visitor_6673,www.site_4.com,www.site_4.com/page_1,1347844440

The data in each CSV file is sorted by the timestamp field, but there may be overlaps between the times in separate files.

Supported Queries

The program should support the following queries:

Num_sessions
Input:Site_url
Output: number of Sessions for the given site

Median_session_length

Input:Site_url
Output: median of Sessions length (in seconds) for the given site
Num_unique_visited_sites
Input: Visitor_id
Output: number of unique visited sites by the given visitor

Calling the program

The program should provide a way to query it.
For example you can implement a program that receives input as arguments in the command line or as a web service that provides a REST API.

Scale support

The submitted service can be implemented using an in-memory solution (i.e. you can assume the size of the input data can be managed in a single PC memory).

However, you should describe what changes are required so that the service would support large scale input.
You can provide the description inside the code (as comments in the code on what you would change for scale) or in a separate text.

Submission
You should submit the following:

- A working solution that initializes itself from the input files when it starts and can be called with multiple queries.
- There is no deadline for this, but please tell us how much time you spent and on what when you submit.
- A text file with:
  - A description of the solution
  - Clear instructions on how to set up, run and call your program
  - How you would support scale
  - Space and time complexity
  - How you tested your code