

Algoviz Graph Creator

Algoviz Graph Creator tool is based on:

- PHP v5.0.4
- JpGraph v2.1.4
- SQLite v3.0

JpGraph (PHP Graph creating library)

JpGraph is an Object-Oriented Graph creating library for PHP. The library is entirely written in PHP and ready to be used in any PHP script. The library can be used to create numerous types of graphs either on-line or written to a file. JpGraph makes it easy to draw very complex graphs which require a very fine grained control. The library assigns context sensitive default values for most of the parameters which minimizes the learning curve. The features are there when you need them - not as an obstacle to overcome.

Features of JpGraph

The following is a shortlist of available features within the library.

- Web-friendly, average image size for a 300*200 image is around 2K and images are seldom bigger than 4-5K
- Support for both GD1 and GD2. The library will auto detect which library is installed.
- Automatic generation of client side image maps to make it possible to generate drill-down graphs.
- Advanced interpolation with cubic splines to get smooth curves from just a few data points.
- Supports several plot types, spider-plots, pie-charts (both 2d and 3d) , scatter-plots, line-plots, filled line-plots, accumulated line-plots, bar plots, accumulated bar plots, grouped bar plots, error plots, line error plots, box plots, stock plots.
- Support for alpha blending
- Has over 200+ built in Country flags
- Supports advanced Gantt-charts (ex1, ex2)
- Flexible scales, supports integer, linear, logarithmic, text (counting) scales and any combination thereof on x/y axis
- Supports multiple Y-axes
- Support various layout with a background image behind the plot
- More than 400 named colors
- Extensive documentation with both a > 150 pages tutorial and an extensive class reference.
- Supports internal caching (with timeout) of generated graphs to lessen burden of a HTTP server.

SQLite

SQLite is an ACID-compliant relational database management system contained in a relatively small C programming library.

Unlike client-server database management systems, the SQLite engine is not a standalone process with which the program communicates. Instead, the SQLite library is linked in and thus becomes an integral part of the program. The program uses SQLite's functionality through simple function calls. This reduces latency in database access because function calls are more efficient than inter-process communication. The entire database (definitions, tables, indices, and the data itself) is stored as a single cross-platform file on a host machine. This simple design is achieved by locking the entire database file at the beginning of a transaction.

SQLite was created by D. Richard Hipp, who sells training, direct technical support contracts and add-ons such as compression and encryption. The source code for SQLite is in the public domain.

Features

SQLite implements most of the SQL-92 standard, including database transactions that are atomic, consistent, isolated, and durable (ACID). SQLite supports triggers and most complex queries.

SQLite uses an unusual type system. Instead of assigning a type to a column as in most SQL database systems, types are assigned to individual values. For example, one can insert a string into an integer column (although SQLite will try to convert the string to an integer first, if the column's preferred type is integer). This adds flexibility to columns, especially when bound to a dynamically typed scripting language. However, the technique is not portable to other SQL databases. The inability to have strictly typed columns, as in typical databases, is a common criticism. However, there are plans to add this capability in the future. SQLite's developers call this feature "strict affinity".

Several computer processes or threads may access the same database without problems. Several read accesses can be satisfied in parallel. A write access can only be satisfied if no other accesses are currently being serviced; otherwise the write access fails with an error code (or can automatically be retried until a configurable timeout expires). This concurrent access situation would change when dealing with temporary tables.

A standalone program called `sqlite3` is provided which can be used to create a database, define tables within it, insert and change rows, run queries and manage a SQLite database file. This program is a single executable file on the host machine. It also serves as an example for writing applications that use the SQLite library.

SQLite also has bindings for a large number of programming languages, including BASIC, C, [[C++]], Java, Delphi, Tcl, R, PHP, Perl, Ruby, Python, newLisp and Smalltalk.[2]

Tool Description

Algoviz Graph Creator tool has been designed to aid user in understanding and analyzing data present in Algoviz wiki. Data in Algoviz wiki is organized in terms of various fields that describe features related to each algorithm visualization tool. Algoviz Grapher has been designed to use the fields that are associated with each algorithm visualization tool, like Author, Institution that submitted the AV and present the aggregate data based on those fields to user in more intuitive fashion in forms of graphs and tables. For example, if a user wants to see how many and what all AVs have been submitted by say, Virginia Tech, he/she will simply choose “Institution” as the classification criterion in Step 2 i.e. “Select Fields and Graph Labels” and , and press “Render Graph”. The resulting graph shall show all the count of AVs submitted by various institutions in graph format. We intend to enhance the existing functionality by providing a hyperlink to each of the plots in Histogram and each entry in table that shall redirect to the detailed info about that count, for e.g. on clicking say count “35” in front of Virginia Tech, shall open up a new page that contains Algoviz Wiki links for various AVs submitted by people associated with “Virginia Tech”.

The Algoviz Graph consists of scraper that goes through entire Algoviz wiki and gets all the URLs present in wiki and the corresponding content in those pages and creates a .csv file. The CSV file name is same as the current time stamp. The data in csv file is then inserted into a table of SQLite database. Tool allows users to generate new csv file so that data that would be inserted into table would be the most recent data.

Currently, tool supports two types of graph: histogram and pie chart, tool also supports table view. Major motivation behind providing different types of graph and table view is that it would help users in understanding and analyzing data.

Tool help

Tool has been basically divided into four steps that allow users to download and make use of a new .csv file (based upon the current data in Wiki), select Categorization Field (i.e. x-axis) to be used while generating the graph. This is followed by some filtering and data organization options like choosing an option to see data in ascending or descending format and finally, the graph generation.

In step1, users are given an option of downloading and making use of the new csv file, based upon current Wiki data or making use of an existing csv file. Besides that there is an option to select type of graph they want to view, like Histogram or Pie chart or table view.

In step 2, users have an option to select a single field out of total 26 fields. This selected field would be the field on which graph would be generated so the user selected would represent x-axis of the graph while y-axis would be the count number associated with that selected field. Also users shall have an option to give the title of the graph as well as title of x-axis and y-axis.

In step 3, the users are given various miscellaneous and filtering options like if they want graph to contain data only for sorting algorithm, and if they want counts to be ascending or descending. Further, they can choose if they want column names to be ascending or descending.

In step 4, the user are prompted to click hyper link in order to view the graph and graph would be displayed in new window, also authors have an option to view csv file or generate csv file depending on their need. Further, to improve user experience, a drop down is provided which includes various types of graphs that user can see and upon selection, the graph type is changed dynamically.

Source Code Details

Source code basically consists of two parts:

- a) Scraper code: It is written entirely in PHP scripting language. It scrapes through the wiki and produces a CSV file with name as the current timestamp.
- b) Code for web based interface for the toolkits that uses PHP along with JpGraph and SQLite as back end. The web interface code primarily comprise in `algovizGrapher.php` that provides the interface for all the four steps that user has to perform in order to generate graph. Graph generation code is present in the file `graph.php`. The properties and dimension settings of graph are present in `graphproperties.php` and `graphtools.php`. The generated csv files are present in the dataset directory and SQLite files are present in the same directory. `Datatools.php` comprises of code that handle database. Apart from this, there is `jpgraph` library code, which is available at <http://hem.bredband.net/jpgraph/jpgraph-1.22.1.tar.gz> for download.