

Functional Specification



November 29, 2012

Functional Specification	Alexander Noe
Design	Jonathan Klawitter
Implementation	Anas Saber
QA / Testing	Nikolaos Alexandros Kurt Moraitakis
Final	Lukas Ehnle

About this document

General Information

PSE is a mandatory course for students of computer science bachelor at the Karlsruhe Institute of Technology (KIT). Therefor groups of five to six students are formed to write programs of 'medium size'. The task given in this course consists of analyzing and visualizing the server log of a database.

This specification is the one of PSE Group #10 in the winter semester 2012/13.

What this specification does

The purpose of this document is an outline of the functional specifications and requirements for the WHAT application. It tries to give a complete and exact model of the future system. In the course of this it wants to give the developers answers to all possible question concerning what should be implemented.

What this specification does not

This specification does not give any information on how the system should be implemented. Also it doesn't contain any project planning or deadlines.

A 'Living Document'

Software development is a dynamic process. So with proceeding development the content of this specification will change continuously.

Skyserver

The concept of the system described in this specification should work with any database storing it's query-log. The SkyServer will function as an example and testing reference for this system. SkyServer is one of the biggest databases for astronomical data.

Questions and comments

If you have any questions or comments regarding this document feel free to send us an E-Mail at pse10-group14-ws12@ira.uni-karlsruhe.de.

Contents

1	Goals	5
1.1	Web page	5
1.1.1	Core criteria	5
1.1.2	Optional criteria	5
1.1.3	Exclusion criteria	5
1.2	Parser	6
1.2.1	Core criteria	6
1.2.2	Exclusion criteria	6
1.3	Analyzer	6
1.3.1	Core criteria	6
1.3.2	Optional criteria	6
2	Usage	7
2.1	Area of Application	7
2.2	Target groups	7
2.3	Operating conditions	7
3	Operating environment	8
3.1	Software	8
3.2	Hardware	8
3.3	Orgware	8
4	Functional requirements	9
4.1	Main functions	9
4.2	Extending functions	10
5	Data	11
5.1	Static data	11
5.2	Dynamic data	11
5.3	Data-Warehouse data	12
6	Nonfunctional requirements	13
6.1	Usability	13
6.2	Swiftness	13
6.3	Maintainability	13
6.4	Safety	13
6.5	Other	13
7	Test Cases	14
7.1	General	14
7.2	Administrator specific	14
7.3	Parser specific	15
7.4	Analyzer specific	15

8 Models	16
8.1 Overview	16
8.2 Dynamic models	17
8.3 Web interfaces	18
9 Development Environment	20
10 Glossary	21

1 Goals

With this program the user should be put in the position to visualize the prepared data of queries, made into his database.

To structure the criteria the system has to fulfill, it is divided into three parts for this and other sections.

- Web page
- parser
- Analyzer

See 8.1 for an overview of these parts and their relationships. In the following section their specific goals and criteria are described.

1.1 Web page

1.1.1 Core criteria

- The web page is the graphical user interface (GUI) for users and administrators.
- Users can choose the charts they want to see and also choose and filter the dimensions and measures for those. It has to support at least scatter plots, bar charts and bubble charts.
- The web page provides help for how to use the diagrams and variables (dimensions and measures). Also the navigation on the page is easy to use and straight forward.
- An admin-login provides administrative rights, which are needed to use the parser.

1.1.2 Optional criteria

- The language of the web page can be changed (e.g. German).
- The administrator gets the ability to handle the data warehouse and load new log files.
- A little history of the last requested charts is stored and viewable.
- You can download a picture of a created charts.

1.1.3 Exclusion criteria

- The web page does not allow normal users to load new data into the data warehouse.
- The web page does not provide statistical tables.

1.2 Parser

1.2.1 Core criteria

- The parser is able to perform the ETL-process on .csv log files (from SkyServer). This means it extracts the data it needs from the files, transforms them and loads them into the data warehouse. dimension and measures are specified in 5.3.
- The parser will recognize invalid logs and won't add them to the data warehouse. Every log with a mistake won't be accepted, because an error-message is not as bad as a corrupted data warehouse.
- The parser will be fed with log files from the administrator via web page.

1.2.2 Exclusion criteria

- This parser is only able to operate on .csv log files from SkyServer. It can neither read logs in another formats nor logs from another source.
- There is no way to avoid using this parser when adding data to the data warehouse. This prevents corrupting the data warehouse and guarantees correct data in it.
- The parser doesn't correct mistakes in the log files.

1.3 Analyzer

1.3.1 Core criteria

- The analyzer is the gate to the data warehouse. It extracts the specific data, needed for the diagrams, passing them to the java-script front-end.
- The analyzer can take information filtered by user-selected criteria, to use only certain data for the charts.
- The charts that are supported are at least:
 - scatter plots
 - bar charts
 - bubble charts

1.3.2 Optional criteria

- The analyzer will support more chart-types, for example the combination of a bar chart and a scatter plot.
- The analyzer does a little bit of data mining, and presents some potentially interesting information.

2 Usage

2.1 Area of Application

The application area of this product is scientific databases that need to be analysed, understood and visualised. The program has been designed with the SkyServer in mind, so that is the primary application area. However, as the only major requirement is the existence of correct access logs of the database. The software may also be used with many other scientific databases, although minor modifications may be needed.

2.2 Target groups

The target group of this application is people that want to analyse and visualize queries made against a database. Specifically, in our case, the SkyServer.

This includes

- the owners of the database who want to optimize the access to their data,
- people that are interested in what others use the database for,
- people new to the database, who want to understand it or discover interesting information.

To summarize, WHAT will be of interest for many people, whether they want to see how their database is used, analyse current trends, or just love statistics and diagrams.

We assume a rather technical audience.

2.3 Operating conditions

The program is mainly used as a website, with the primary difference being that the server has to be started, if the capacity for it to run all the time on a dedicated machine doesn't exist. The program needs a server to run.

If a dedicated server exists, the program can be used from anywhere with a decent network connection to it.

If not, the program can still be run on the same computer as the server (on localhost), but the server will have to be started first.

3 Operating environment

Whereas the program and the data warehouse run on a server, the access to it will be via a web browser on the users computer. This implies that the webpage will need to be hosted on a webserver. However, a dedicated server machine is not required, as the clients computer can host both the client and the webserver, and then access the webpage locally.

3.1 Software

- The server hosting the data warehouse needs SQL support.
- The server hosting the web server needs a recent version of the Java VM

Following software is required on the users computer:

- Latest version of a modern browser
 - Though Chrome and Firefox will be officially supported, other browsers might work as well.
- JavaScript (enabled)

3.2 Hardware

The server has to be fast enough to support all clients. This depends on the expected number of clients. Most computations will be done on the server.

The client needs to be fast enough to visualize the data received from the server, as that's almost all it does. The required hardware thus depends greatly on the amount of data that needs to be visualized. That said, any recent computer, with for example, an Intel® Core™2 Duo CPU E8400 @ 3.00GHz × 2 processor, 4GB(2x2GB) of 667Mhz DDR2 SDRAM, running Ubuntu Linux 12.10 Quantal Quetzal, as a point of reference, should be able to visualize about 50.000 data points on a scatter plot with ease.

3.3 Orgware

The server needs to be able to connect to the client with a reasonable latency. Also it has to be set up before the client is able to connect to it.

To fulfill optional requirements, the program must be able to run queries against the origin database.

4 Functional requirements

4.1 Main functions

This functions are required to fulfill the core criteria.

General

- /F010/ Provide access via web page
- /F020/ Provide option for chart types
- /F030/ Show bar charts
- /F040/ Provide option to select and filter variables for bar charts
- /F050/ Show scatter plots
- /F060/ Provide option to select and filter variables for scatter plots
- /F070/ Show bubble charts
- /F080/ Provide option to select and filter variables for bubble charts
- /F090/ Show information about chart types
- /F100/ Show information about selectable variables

Administrator specific

This functions are required for the administrative business.

- /F110/ Provide access via web page with administrator rights
- /F120/ Provide the opportunity to pass log files to the parser

Parser specific

The functions required from the parser are mainly to execute the ETL-process.

- /F130/ Extract specific data from the log files
- /F140/ Extract access database, access time and user information (dimensions)
- /F150/ Extract number of rows, elapsed time, busy time (measures)
- /F160/ Extract type of data requested from the WHERE-part
- /F170/ Transform the data to fit into the data warehouse schema
- /F180/ Load the data into the data warehouse

Analyzer specific

/F190/ Run specific queries against data warehouse

/F200/ Transform received data serving the web page

4.2 Extending functions

To fulfill the optional goals the following functions are required.

General

/F210/ Select language on web page

/F220/ Show bubble map

/F230/ Show histogram

/F240/ Show combination of bar chart and scatter plot

/F250/ Show other charts

/F260/ Show history of the 5 last requested charts

/F270/ Provide option to download a computed chart as picture.

/F280/ Support zooming on charts

/F290/ Show interesting attribute combinations

/F300/ Show clusters in the data

/F310/ Add background clustering

Administrator specific

/F320/ Provide the opportunity to initialize the data warehouse

/F330/ Provide the opportunity to request new log file for specific time intervals from the database

/F340/ Provide the opportunity to clean the data warehouse

5 Data

5.1 Static data

/D10/ Language files

/D20/ Manual

/D30/ Source Code

/D40/ Documentation

/D50/ Graphics for GUI

/D60/ HTML backbone

/D70/ Javascript files

/D80/ Stylesheet files

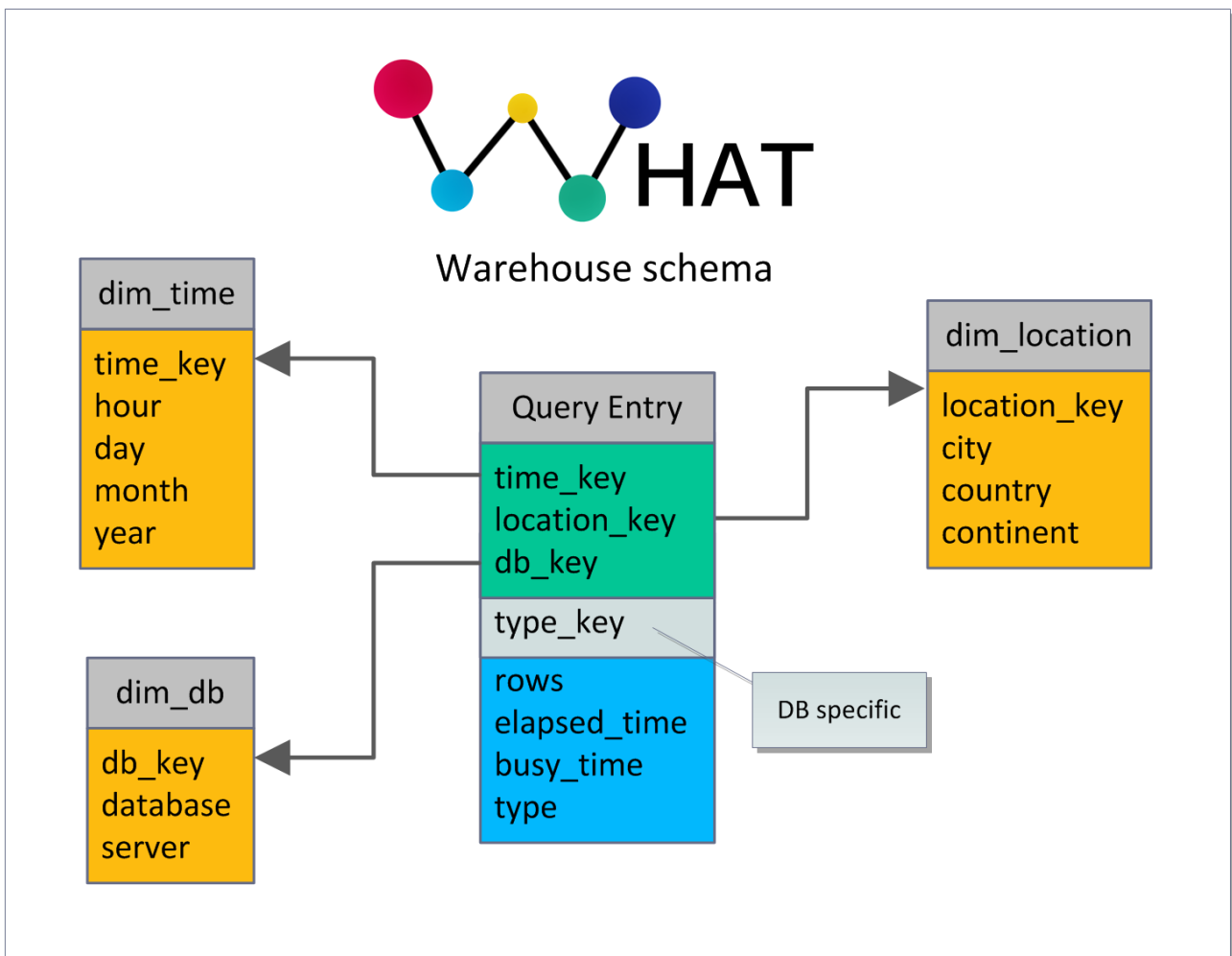
5.2 Dynamic data

/D90/ Administrator login data

5.3 Data-Warehouse data

The data warehouse stores all the data, coming from from the databases (SkyServer) server logs. They were subject of the ETL-process in the parser. If optional criteria are fulfilled they can be replaced by clearing the data warehouse and loading new data via web page as administrator.

The data warehouse may use a star schema.



The column 'type_key' can either be just a measure or refer to a new dimension. This depends on the database on which is operated.

See rows, elapsed time and busy time in the glossary (??) for descriptions.

6 Nonfunctional requirements

6.1 Usability

- /N010/ Minimalistic GUI design: Not more than 8 clicks needed to see a chart.
- /N020/ Responsive GUI
- /N030/ GUI which you will be able to work with not later than trying for 3 minutes.
- /N040/ The data warehouse can store more than 1000000 entries

6.2 Swiftness

- /N050/ Parse 1000 log rows entries in 10 seconds
- /N060/ Visualize scatter plot of 50000 points in 10 seconds

6.3 Maintainability

- /N070/ Languages files for easy adding of new languages
- /N080/ Reasonably commented, clean code

6.4 Safety

- /N090/ Protection against DoS attacks.

6.5 Other

- /N100/ Java is the programming language for the main program
- /N110/ MySQL or Oracle SQL is used for the data warehouse

7 Test Cases

7.1 General

These test-cases are necessary for the web page to work properly and succeed in serving basic requests.

/T010/ Access web page (/F010/)

 /T012/ via Google Chrome,

 /T014/ via Mozilla Firefox.

/T020/ Navigate swiftly through the web page. (/N030/)

/T030/ Open page for scatter plot-creation and create an example with x-Axis date and y-Axis number of rows per request. (/F020/, /F050/, /F060/)

/T040/ Open page for bar chart-creation and create an example with x-Axis year and y-Axis number of requests. (/F020/, /F030/, /F040/)

/T050/ Open page for bubble chart-creation and create an example with x-Axis year, y-Axis Country and size number of requests. (/F020/, /F070/, /F080/)

/T060/ Call the information about chart types. (/F090/)

/T070/ Request information about selectable variables. (/F100/)

7.2 Administrator specific

These test cases check the correct admin-login.

/T080/ Enter loginname and password for admin-login. (/F110/)

/T090/ Log-in and get administrative rights. (/F110/)

/T100/ Try to log in with wrong loginname. (/F110/)

/T110/ Try to log in with wrong password. (/F110/)

/T120/ Pass example log file to the parser. (/F120/)

7.3 Parser specific

The following test cases ensure the correct work of our parser.

/T130/ Parse example log file and extract valuable information correctly. (/F130/)

More specifically:

/T132/ Extract correct database, time, user information and type, (/F140/)

/T134/ Extract correct number of rows, elapsed time, busy time from the log file, (/F150/)

/T136/ Extract the type requested. (/F160/)

/T140/ Load the extracted information into the data warehouse correctly. (/F170/, /F180/)

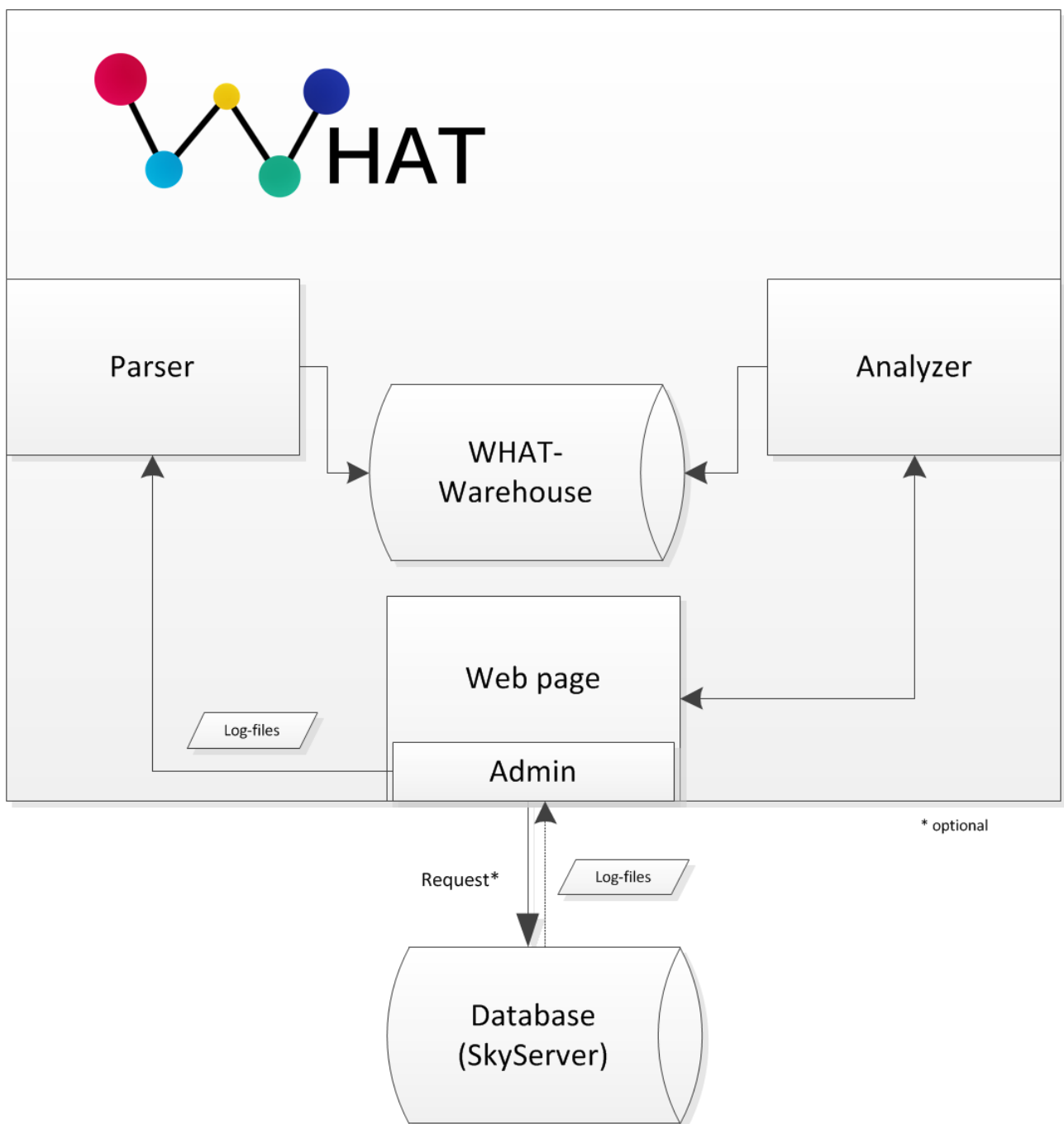
7.4 Analyzer specific

The functions /F190/ and /F200/ of the analyzer are already tested in the general test-cases /T030/ to /T050/.

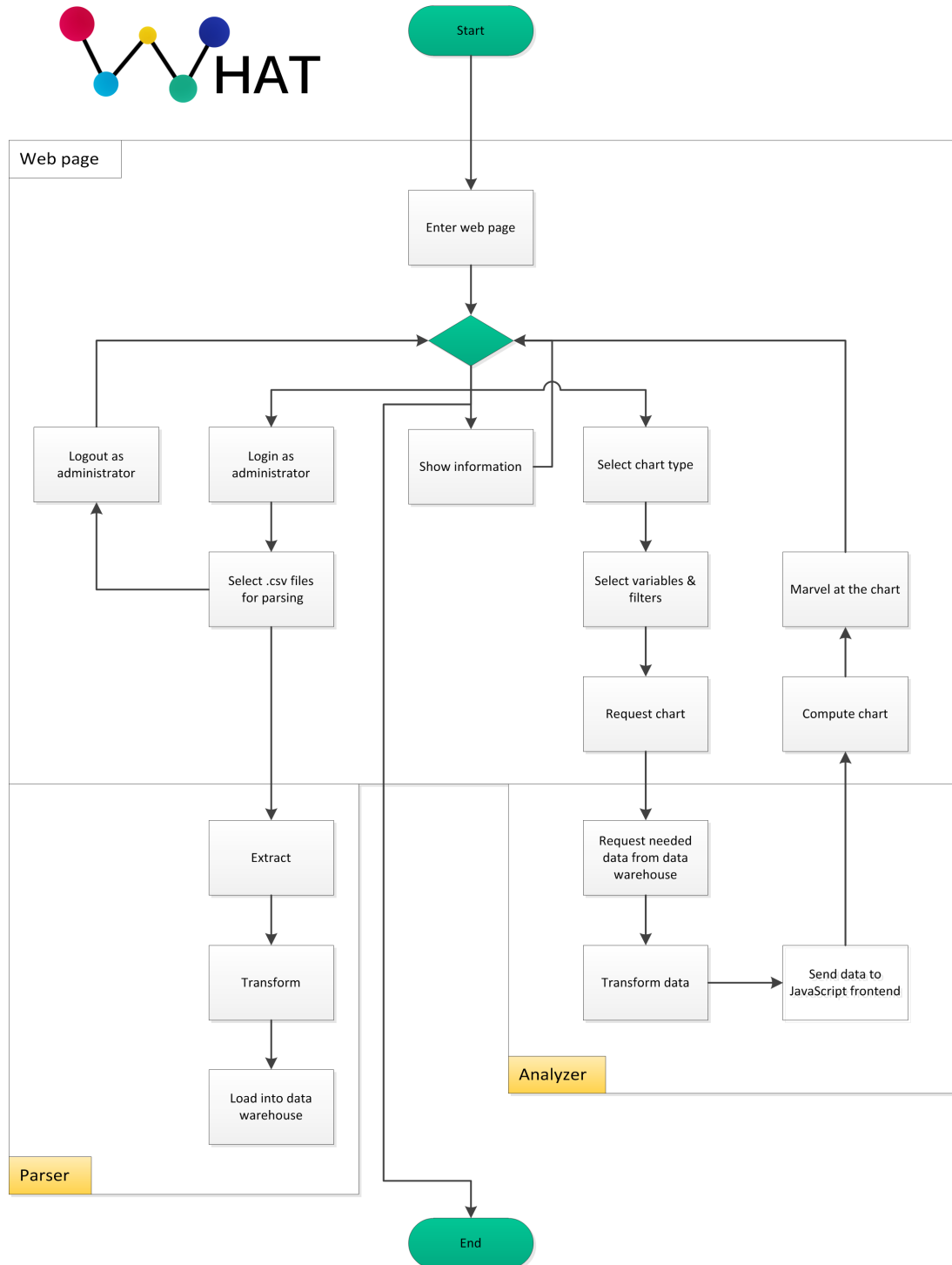
8 Models

8.1 Overview

This picture describes the general concept of WHAT.

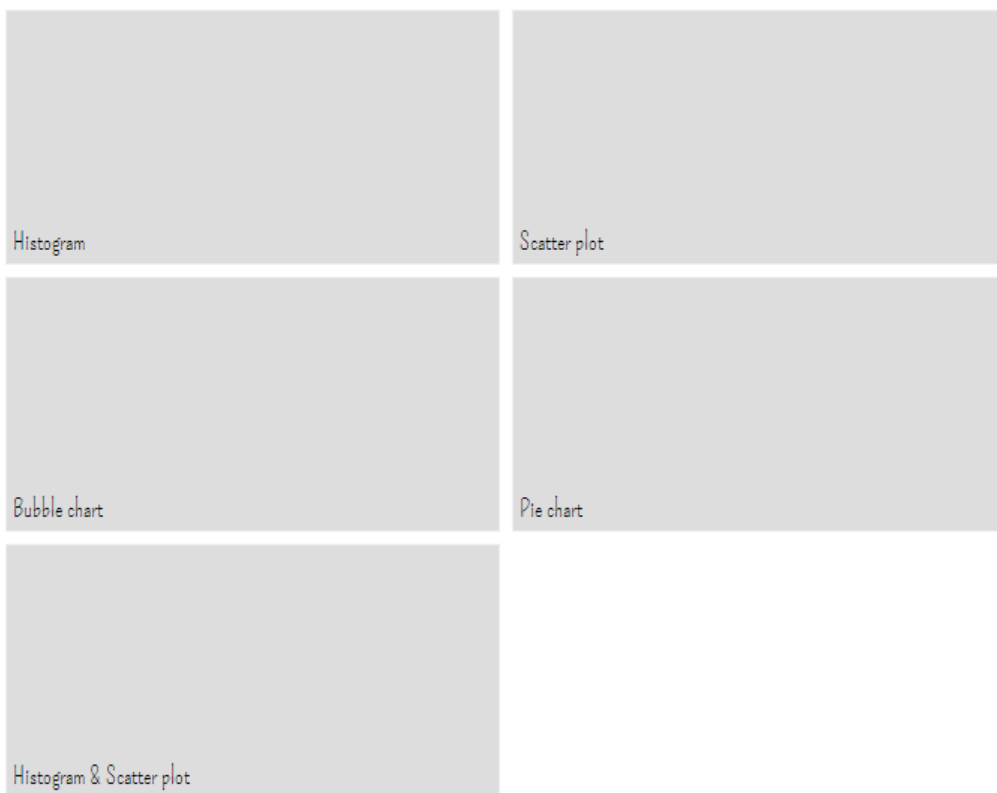


8.2 Dynamic models



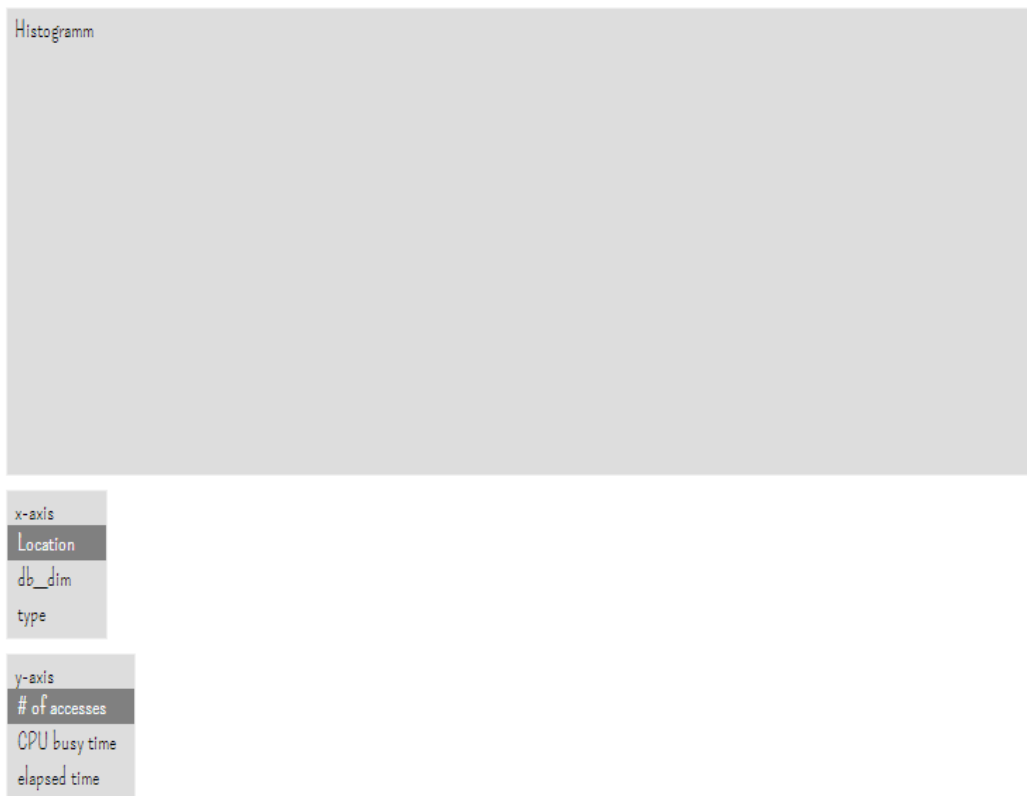
8.3 Web interfaces

The start page of the web page. Here you can choose directly the type of chart you want.



PSE Project by Alex, Anas, Jo, Lukas & Niko

An example for a chart page. Below the chart you can choose the variables and filters.



PSE Project by Alex, Anas, Jo, Lukas & Niko

9 Development Environment

Operating System

- Windows 7
- Fedora 17
- Ubuntu 12.10

IDE

- Eclipse

Object Models

- Microsoft Visio

Version Control

- Git

Miscellaneous

- Github (Hosting of Repository, Issue Tracking)
- Travis (Integration)
- \LaTeX (Documents)
- Gradle (Building)



10 Glossary