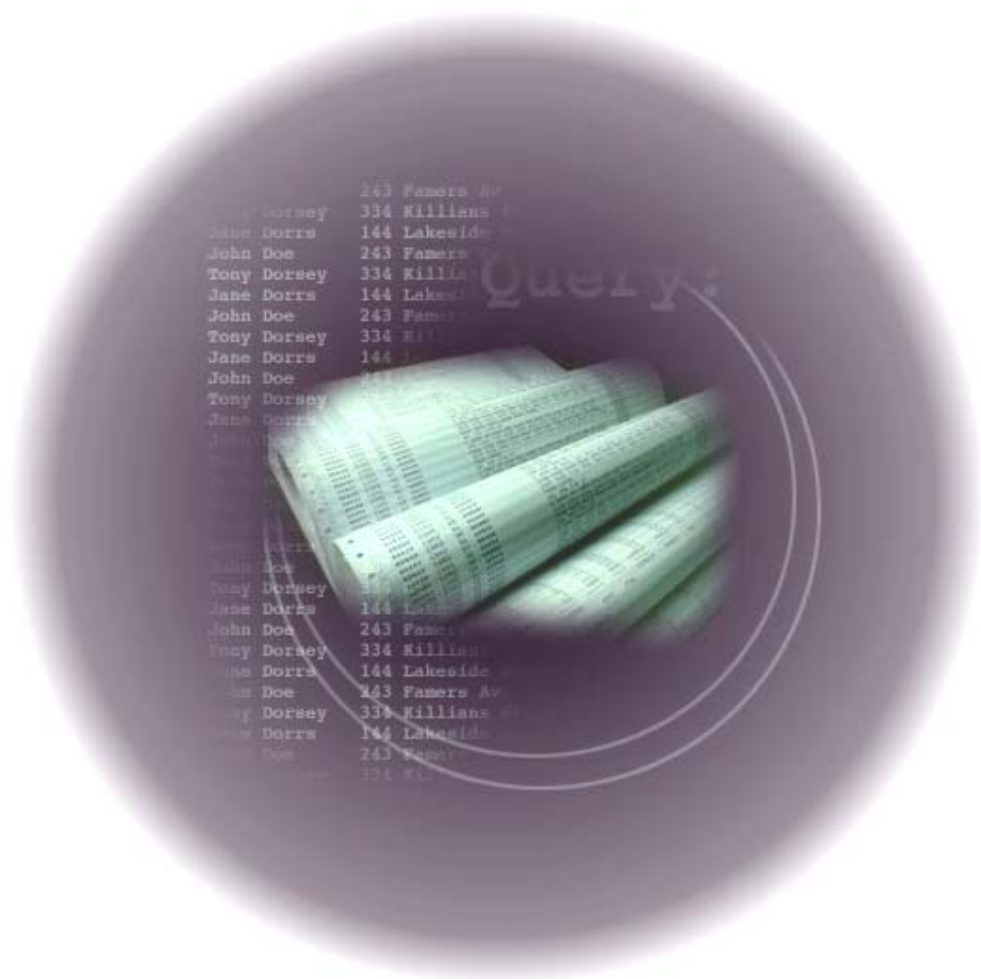


Bachelor Thesis
Course „Project 2“

Concept Of A Database Prototype

(GOCE Mission)



Daniel Gander

Graz, 2003

Index Of Contents

Index Of Contents	2
1 Introduction	3
1.1 Project definition.....	3
1.2 Project Objectives	3
2 Used Tools	4
2.1 Short Introduction to the Open Source Philosophy	4
2.2 Operating System RedHat Linux 7.3	5
2.3 Webserver Apache HTTP Server Version 2.0	6
2.4 Database Management System MySQL 3.23.10 – alpha	6
2.5 Middleware PHP 4.0.3pl1	7
2.6 Administration Tool PHPMyAdmin 2.4.0.....	7
3 Documentation	9
3.1 Structure Of The Database.....	9
3.1.1 Levels	9
3.1.2 Categories.....	9
3.1.3 Product	10
3.1.4 Data	10
3.2 User administration.....	10
3.2.1 “user”-Table	10
3.2.2 “host”-Table	13
3.2.3 “db”-Table.....	13
3.2.4 “tables_priv”-Table	15
3.2.5 “columns_priv”-Table.....	16
3.2.6 “user_sessions”-Table	16
3.2.7 “info”-Table	18
3.2.8 Example: “Add a user”.....	18
3.3 Search System.....	21
3.3.1 Search data	21
3.3.2 “Shopping basket”.....	24
3.3.3 Send request	25
3.3.4 Background jobs	26
4 Conclusion and future aspects	27
5 Indexes	28
5.1 Table Of Figures	28
6 Bibliography	29
6.1 Online sources.....	29

1 Introduction

1.1 Project definition

This project describes the realisation of an international, relational database prototype in cooperation with Dipl. Ing. Gerhard Schrotter. The system has to be save for abuse by hackers or like. The system has to be automated in highest grade and the total costs must be kept as down as possible.

1.2 Project Objectives

Aim of the project is to develop a database prototype. Another main aspect of the project is to test the interface to the database system till June 2003. To minimize total costs General Public License (GPL) Software has been used.

In the hole test-phase we used Champ Data, because GOCE data were not available yet (Link: <http://isdc.gfz-potsdam.de/champ/> [10.06.2003]).

2 Used Tools

2.1 Short Introduction to the Open Source Philosophy

All software is built with source code. Open source means that the code can be seen and changed. With this power comes control.

In the proprietary model, development occurs within one company. Programmers write code, hide it behind binaries, charge customers to use the software and then charge them more to fix it when it breaks. No one ever has to know how bad the software really is. Bad software costs companies \$78 billion per year, according to CIO Magazine.

The problem worsens when you become tied to a company's protocols and file formats. Then you're hooked. Bruce Perens calls this the “addiction model of software procurement.” Any model that puts customers at such a fundamental disadvantage is conceptually broken.

That's why open source is inevitable. It returns control to the customer. The code is open and you can see it, change it, and learn from it. Bugs are more quickly found and fixed. And when customers don't like how one vendor is serving them, they can choose another without overhauling their infrastructure. That means: No more arbitrary pricing. No more technology lock in. No more monopolies.

And we believe open source simply creates better software. It multiplies one company's development capacity many times over. Everybody collaborates, the best software wins. Not just within one company, but among an Internet-connected, worldwide community. It's no coincidence that the rise of open source closely followed the rise of the Internet. The perfect breeding ground for collaboration, the Internet moves ideas and code around the world in an instant.

But it doesn't just happen by accident. Larger open source projects have a complex leadership structure. These leaders oversee a process that collects software submitted by the open source community, then integrate the software, test it, and decide whether or not to include it in an upcoming release.

Red Hat then assembles the kernel with other elements of the operating system, follows a complex quality assurance process, and certifies it for use with the top enterprise hardware and software vendors. This creates a standard platform; a key factor in enterprise adoption.

Widespread support provides the necessary stability to take Linux into mission-critical deployments.

The most widely known open source software is the Linux operating system. Linux made up 25% of all server operating systems sold in 2001--#2 in the market--according to IDC. And the open source Apache web server is the #1 web server, holding 67% of the market, according to E-Soft.

The concept behind open source is not new. For centuries, universities and other research communities have shared their work. Monks copied books by hand. Scientists publish new discoveries in journals. Imagine if we were unable to build on this past knowledge. Yet this is the mentality on which the proprietary software model depends.

In the same way shared knowledge propels the whole of society forward, open technology development can drive innovation for an entire industry.¹

2.2 Operating System RedHat Linux 7.3

The used operating system is Redhat Linux 7.3. All Linux distributions are under Open Source. In 1991 Finnish college student Linus Torvalds created an operating system for his own use that used cheap PC hardware but provided the same capabilities as the expensive Unix systems he used at college. He made the new OS available on the Internet with source code, and protected it with a special license that allowed others to improve the software as long as the code remained visible and modifiable. In other words, open. As the word got out about Linux and demand grew, Red Hat and other companies began selling a packaged version of the OS, including documentation and support. Red Hat Linux often found its way into server rooms because it was less expensive, but it soon gained widespread popularity more for its reputation as a stable, reliable operating system. Now Linux rises on a wave of inevitability. It has become the world's fastest growing server OS, according to research firm IDC. And it continues to achieve adoption among the largest enterprises in mission-critical roles. In 2002 a Merrill Lynch study found that a quarter of CIOs said that Linux is strategic to their enterprise. Red Hat's dominant position in the Linux market has enabled it to assemble the world's most talented group of open source engineers. They assemble the Linux kernel and other elements of the Linux operating system, compile it, and test it for performance and reliability. Then they add new features, test for

¹ Source: <http://www.redhat.com/about/mission/opensource.html> [10.06.2003]

compatibility; all while sharing the software with customers, partners and software vendors, and members of the open source community in a structured feedback cycle. No other Linux company has a process this complete or meticulous. Red Hat also provides a voice for our customers in the open source community. We originate and support many projects that our customers feel are critical to their business. That's why Red Hat Linux become the market-leading, most trusted Linux distribution. This is the reason why so many companies take advantage of the expertise for services like support, training, and consulting.²

2.3 Webserver Apache HTTP Server Version 2.0

The Apache Project is a collaborative software development effort aimed at creating a robust, commercial-grade, feature-full, and freely-available source code implementation of an HTTP (Web) server. The project is jointly managed by a group of volunteers located around the world, using the Internet and the Web to communicate, plan, and develop the server and its related documentation. These volunteers are known as the Apache Group. In addition, hundreds of users have contributed ideas, code, and documentation to the project. This file is intended to briefly describe the history of the Apache Group and recognize the many contributors.³

2.4 Database Management System MySQL 3.23.10 – alpha

The MySQL database server is the world's most popular open source database. Its architecture makes it extremely fast and easy to customize. Extensive reuse of code within the software and a minimalistic approach to producing functionally-rich features has resulted in a database management system unmatched in speed, compactness, stability and ease of deployment. The unique separation of the core server from the storage engine makes it possible to run with strict transaction control or with ultra-fast transactionless disk access, whichever is most appropriate for the situation.

² Source: <http://www.redhat.com/about/mission/linux.html> [10.06.2003]

³ Source: http://httpd.apache.org/ABOUT_APACHE.html [10.06.2003]

The MySQL database server is available without a license fee under the GNU General Public License (GPL).⁴

2.5 Middleware PHP 4.0.3pl1

PHP (PHP Hypertext Preprocessor) is an HTML-embedded scripting language. Much of its syntax is borrowed from C, Java and Perl with a couple of unique PHP-specific features thrown in. The goal of the language is to allow web developers to write dynamically generated pages quickly.

2.6 Administration Tool PHPMyAdmin 2.4.0

PhpMyAdmin can manage a whole MySQL server (needs a super-user) as well as a single database. To accomplish the latter you'll need a properly set up MySQL user who can read/write only the desired database. It's up to you to look up the appropriate part in the MySQL manual.

Currently phpMyAdmin can:

- create and drop databases
- create, copy, drop, rename and alter tables
- do table maintenance
- delete, edit and add fields
- execute any SQL-statement, even batch-queries
- manage keys on fields
- load text files into tables
- create and read dumps of tables
- export data to CSV, XML and Latex formats
- administer multiple servers
- manage MySQL users and privileges
- check referential integrity
- using Query-by-example (QBE), create complex queries automatically connecting required tables
- create PDF graphics of your Database layout

⁴ Source: <http://www.mysql.com/products/mysql/index.html> [10.06.2003]

- search globally in a database or a subset of it
- transform stored data into any format using a set of predefined functions, like displaying BLOB-data as image or download-link or ...
- communicate in 42 different languages⁵

⁵ Source: <http://www.phpmyadmin.net/documentation/#faq> [10.06.2003]

3 Documentation

3.1 Structure Of The Database

The structure of the database is quite simple. The structure of the folder tree on the system is the same like in the MySQL database. The directory structure in meaning of numbers of subdirectories is fixed. The system was generated in this way, because it should be flexible and easy to adopt for new products, categories, and data. The first folder and also the first table in the GOCE database is called "LEVEL".

3.1.1 Levels

The first table in the GOCE database is the "Level-table". It has two columns - the id and the Level column. The fields of the Level column are the names of the following tables. This strategy of finding the way through the structure of the database is normally not the way of thinking "relational". But this system is very efficient, because it is very flexible for any kind of changing of the structure.

id	Level
1	LEVEL1
2	LEVEL2
3	LEVEL3
4	LEVEL4

Figure 1: Levels

3.1.2 Categories

This table has also two columns, the id and the Category column. The Category column contains all the possible products of the chosen level.

id	Category
1	LEVEL3__ORBIT

Figure 2: Level 3 - category

3.1.3 Product

The Product table has the same structure as the tables before. As an example the LEVEL3 Ultrarapid Science Orbit and LEVEL3 Rapid Science Orbit is shown.

id	Product
1	LEVEL3__ORBIT__RSO
2	LEVEL3__ORBIT__USO

Figure 3: Level 3 - products

3.1.4 Data

The last table class contains the data files. It has five columns and as much rows as files are in the database. The columns are the size of the file, the time stamp of last change of the file, the time stamp of the insert and of course the filename and the name of the metafile that contains information about the file.

Size	Stamp_LC	Stamp_IN
202548	20030409093025	20030530123001
203626	20030409093025	20030530123001

File	Metafile
CH-OG-3-RSO+CTS-CHA_2001_182_10.dat	CH-OG-3-RSO+CTS-CHA_2001_182_10.dif
CH-OG-3-RSO+CTS-CHA_2001_182_22.dat	CH-OG-3-RSO+CTS-CHA_2001_182_22.dif

3.2 User administration

All the user administration can be made with MySQL. The corresponding database with the rights of the users is named mysql-database.

3.2.1 “user”-Table

The user-table is a standard table of MySQL. All users that read or write data in a database must have an user account. All global rights (for example “select”, “insert” or

“create”) for all databases are saved in the user table. It is possible to deny/allow the access on a database, a table or even a column in a table.

Field	Type	Attributes	Null	Default	Extra
Host	char(60)	BINARY	No		
User	char(100)	BINARY	No		
Password	char(16)	BINARY	No		
Select_priv	enum('N', 'Y')		No	N	
Insert_priv	enum('N', 'Y')		No	N	
Update_priv	enum('N', 'Y')		No	N	
Delete_priv	enum('N', 'Y')		No	N	
Create_priv	enum('N', 'Y')		No	N	
Drop_priv	enum('N', 'Y')		No	N	
Reload_priv	enum('N', 'Y')		No	N	
Shutdown_priv	enum('N', 'Y')		No	N	
Process_priv	enum('N', 'Y')		No	N	
File_priv	enum('N', 'Y')		No	N	
Grant_priv	enum('N', 'Y')		No	N	
References_priv	enum('N', 'Y')		No	N	
Index_priv	enum('N', 'Y')		No	N	
Alter_priv	enum('N', 'Y')		No	N	

Figure 4: Structure of "user"-Table

This is an example of the user table in the development-phase of the project. There is a superuser (su), who is needed to access the mysql database for login information. If the user is present the login is possible, else it isn't. After the user is recognized by the system, he/she uses its own username and password to access to the databases. The username is the IP-address of his computer. The password is coded. The host must be the IP-address of the server because MySQL is running on the same server.

Host	User	Password	Select_priv	Insert_priv
129.27.91.65	su	077fe1f649259428	Y	Y
129.27.91.65	info	770491ea58a891a4	N	N
129.27.91.65	129.27.91.39	43e9a4ab75570f5b	N	N
129.27.91.65	session	033fe6de53e0212c	N	N
129.27.91.65	129.27.91.65	171f5d1d71f84332	N	N
fmgeolx05	pftp	171f5d1d71f84332	N	N

Update_priv	Delete_priv	Create_priv	Drop_priv	Reload_priv
Y	Y	Y	Y	Y
N	N	N	N	N
N	N	N	N	N
N	N	N	N	N
N	N	N	N	N
N	N	N	N	N

Shutdown_priv	Process_priv	File_priv	Grant_priv	References_priv
Y	Y	Y	Y	Y
N	N	N	N	N
N	N	N	N	N
N	N	N	N	N
N	N	N	N	N
N	N	N	N	N

Index_priv	Alter_priv
Y	Y
N	N
N	N
N	N
N	N
N	N

Figure 5: Example of "user"-table

3.2.2 “host”-Table

With the host table it is possible to allow or to deny access to the databases from several hosts. This function is not used in the database concept.

Field	Type	Attributes	Null	Default	Extra
Host	char(60)	BINARY	No		
Db	char(64)	BINARY	No		
Select_priv	enum('N', 'Y')		No	N	
Insert_priv	enum('N', 'Y')		No	N	
Update_priv	enum('N', 'Y')		No	N	
Delete_priv	enum('N', 'Y')		No	N	
Create_priv	enum('N', 'Y')		No	N	
Drop_priv	enum('N', 'Y')		No	N	
Grant_priv	enum('N', 'Y')		No	N	
References_priv	enum('N', 'Y')		No	N	
Index_priv	enum('N', 'Y')		No	N	
Alter_priv	enum('N', 'Y')		No	N	

Figure 6: Structure of "host"-table

3.2.3 “db”-Table

The db table gives rights for users to access to the database. If in the user database the global right is set, this table has no influence on the right behaviour anymore. It is also possible to set several rights like “insert”, “create” or “delete”.

Field	Type	Attributes	Null	Default	Extra
Host	char(60)	BINARY	No		
Db	char(64)	BINARY	No		
User	char(16)	BINARY	No		
Select_priv	enum('N', 'Y')		No	N	
Insert_priv	enum('N', 'Y')		No	N	
Update_priv	enum('N', 'Y')		No	N	
Delete_priv	enum('N', 'Y')		No	N	
Create_priv	enum('N', 'Y')		No	N	
Drop_priv	enum('N', 'Y')		No	N	
Grant_priv	enum('N', 'Y')		No	N	
References_priv	enum('N', 'Y')		No	N	
Index_priv	enum('N', 'Y')		No	N	
Alter_priv	enum('N', 'Y')		No	N	

Figure 7: Structure of "db"-table

In the db-table the “select”-right to the GOCE database is given, the others are denied. Every user has a personal user database (e.g. user_129_27_91_39). Every user has the right to “select”, “insert” and “delete” data.

Host	Db	User	Select_priv	Insert_priv
129.27.91.65	GOCE	129.27.91.39	Y	N
129.27.91.65	user_129_27_91_39	129.27.91.39	Y	Y
129.27.91.65	user_129_27_91_65	129.27.91.65	Y	Y
129.27.91.65	GOCE	129.27.91.65	Y	N

Update_priv	Delete_priv	Create_priv	Drop_priv	Grant_priv
N	N	N	N	N
N	Y	N	N	N
N	Y	N	N	N
N	N	N	N	N

References_priv	Index_priv	Alter_priv
N	N	N
N	N	N
N	N	N
N	N	N

Figure 8: Example of "db"-table

3.2.4 "tables_priv"-Table

With the tables_priv-table its possible to allow or to deny rights for tables in a database.

Field	Type
<u>Host</u>	char(60)
<u>Db</u>	char(64)
<u>User</u>	char(16)
<u>Table_name</u>	char(60)
Grantor	char(77)
Timestamp	timestamp(14)
Table_priv	set('Select', 'Insert', 'Update', 'Delete', 'Create', 'Drop', 'Grant', 'References', 'Index', 'Alter')
Column_priv	set('Select', 'Insert', 'Update', 'References')

Attributes	Null	Default	Extra
BINARY	No		
BINARY	No		
BINARY	No		
BINARY	No		
	No		
	Yes	NULL	
	No		
	No		

Figure 9: Structure of "tables_priv"-table

The user "info" reads the email-address from the "info"-table. This means that the user "info" must have the right to use the "select" command. The user "session" reads and

writes in the “user_session”-table. So he/she must have the rights to select, insert, delete and update this table.

Host	Db	User	Table_name	Grantor
129.27.91.65	mysql	info	info	129.27.91.65@fmgeobx05
129.27.91.65	mysql	session	user_sessions	129.27.91.65@fmgeobx05

Timestamp	Table_priv	Column_priv
20030603113040	Select	
20030603112835	Select,Insert,Update,Delete	

Figure 10: Example of "tables_priv"-table

3.2.5 “columns_priv”-Table

With the columns_priv-table its possible to allow or to deny rights on several columns in a table. This function was not used in the project.

Field	Type	Attributes	Null	Default	Extra
<u>Host</u>	char(60)	BINARY	No		
<u>Db</u>	char(64)	BINARY	No		
<u>User</u>	char(16)	BINARY	No		
<u>Table_name</u>	char(64)	BINARY	No		
<u>Column_name</u>	char(64)	BINARY	No		
Timestamp	timestamp(14)		Yes	NULL	
Column_priv	set('Select', 'Insert', 'Update', 'References')		No		

Figure 11: Structure of "columns_priv"-table

3.2.6 “user_sessions”-Table

The user_session-table is not a standard table of MySQL. It was added primarily to save variables during a session of a user on the system but secondarily also to make a history log of all access activities on the system.

Field	Type	Attributes	Null	Default	Extra
session_id	varchar(32)		No	"	
session_created	varchar(14)		No	0	
session_active	varchar(14)		No	0	
session_counter	int(11)		No	1	
session_remote_address	varchar(128)		No	"	
session_data	longtext		No		

Figure 12: Structure of "user_session"-table

The table consists of 6 columns. The first column shows the PHP session-ID during a session. The second shows date and time of the creation of the session. The third column shows date and time of the last update of the session. The fourth column shows the numbers of updates of the session. In the fifth column there are IP-address and hostname of the logged user saved, and in the last one the variable data of the session is stored.

session_id	session_created	session_active
080b548576c6b1d5fce43831b001cea3	20030604081653	20030604133536
9eb27db3bbac9d76784f4f58dfe982ab	20030604082523	20030604082726
d1353b35db95e0085b66c0e3350e693f	20030604123327	20030604123550
99238c34ec5368eccc83c3937a2199d3	20030605112729	20030605125656
bef30ff52078ba790d2aa88efeb67d84	20030605155526	20030606103602

session_counter	session_remote_address
124	129.27.91.65 - fmgeolx05
52	129.27.91.19 - fmgeopc31.tu-graz.ac.at
23	129.27.41.97 - subzi097.tu-graz.ac.at
49	129.27.91.65 - fmgeolx05
985	193.171.243.33 - d-33.vc-graz.ac.at

session_data
password s:8:"YmF0bWFu";email s:44:"c2Nocm90dGVyQG...
user s:12:"129.27.91.39";password s:8:"YWRtaW4=";e...
password s:8:"YWRtaW4=";email s:40:"cGxhbmtAZ2VvbW...
password s:8:"YWRtaW4=";email s:40:"cGxhbmtAZ2VvbW...
password s:8:"YWRtaW4=";email s:40:"cGxhbmtAZ2VvbW...

Figure 13: Example of "user_session"-table

3.2.7 “info”-Table

The info-table is also not a standard table of MySQL. It was added to save the email-addresses of all users to contact them automatically.

Field	Type	Attributes	Null	Default	Extra
<u>ipaddress</u>	varchar(15)		No		
email	varchar(100)		No		

Figure 14: Structure of "info"-table

ipaddress	email
129.27.91.65	schrotter@geomatics.tu-graz.ac.at
129.27.91.39	plank@geomatics.tu-graz.ac.at
193.171.243.33	gander@sbox.tugraz.at

Figure 15: Example of "info"-table

3.2.8 Example: “Add a user”

If a new “normal”-user is added to the database, the following steps are to do. The user has an IP-address “193.171.225.220”, his/her password is “hallo” and his/her email-address is test@user.at.

- a) Add the user to the /mysql/user table:

Host: 129.27.91.65

⇒ *It's the hostname of the server.*

User: 193.171.225.220

⇒ *It's the IP-address of the computer of the user.*

Password: hallo

⇒ *It's his/her password.*

Select_priv, Insert_priv, Update_priv, Delete_priv, Create_priv, Drop_priv,
Reload_priv, Shutdown_priv, Process_priv, File_priv, Grant_priv, References_priv,
Index_priv, Alter_priv: N

⇒ *All rights must be denied.*

b) Add the user to the /mysql/db table for access to the GOCE database:

Host: 129.27.91.65

⇒ *It's the hostname of the server.*

db: GOCE

⇒ *It's the name of the database, witch the users has rights to access.*

User: 193.171.225.220

⇒ *It's the IP-address of the computer of the user.*

Select_priv: Y

⇒ *The user has the right to select from the database "GOCE"*

Insert_priv, Update_priv, Delete_priv, Create_priv, Drop_priv, Reload_priv,
Grant_priv, References_priv, Index_priv, Alter_priv: N

⇒ *This rights must be denied.*

c) Add the user to the /mysql/db table for access of his private database:

Host: 129.27.91.65

⇒ *It's the hostname of the server.*

db: user_193_171_225_220

⇒ *It's the name of the database, that the users has rights to access.*

User: 193.171.225.220

⇒ *It's the IP-address of the user's computer.*

Select_priv, Insert_priv, Delete_priv: Y

⇒ *The user has the right to select, to insert and to delete from his/her private database.*

Update_priv, Create_priv, Drop_priv, Reload_priv, Grant_priv, References_priv,
Index_priv, Alter_priv: N

⇒ *This rights must be denied.*

d) Add the user to the /mysql/info table:

IP address: 193.170.225.220

⇒ *It's the IP-address of the computer of the user.*

email: test@user.at

⇒ *It's his email-address.*

e) Make a new database called: user_193_170_225_220

f) Make two new tables in the new database with the following attributes:

- Name: download_193_170_225_220

Columns:

Field	Type	Attributes	Null	Default	Extra
<u>File</u>	varchar(40)		No		
Size	varchar(10)		No		
Dir	varchar(200)		No		

Figure 16: Structure of "download"-table

⇒ *The "shopping-basket" of the user.*

- Name: downloadhistory_193_170_225_220

Columns:

Field	Type	Attributes	Null	Default	Extra
Stamp	timestamp(14)		Yes	NULL	
<u>File</u>	varchar(40)		No		
Size	varchar(10)		No		
Dir	varchar(200)		No		

Figure 17: Structure of "downloadhistory"-table

⇒ *All downloaded files of the user.*

3.3 Search System

The search function is composed quite simple. It's possible to search data, to put data in a kind of shopping basket and to send a request to the system to copy the selected files in the personal download directory of the logged in user.

3.3.1 Search data

By a click on "Search" the database master window will be opened and a window with the selection frame on the left side and the "shopping-basket" on the right side will appear. If something was putted in the shopping basket during the session, it will be shown.



Figure 18: Search window 1

The first step is to select a level, after that the category, and as last step the product.



Figure 19: Search window 2

It is also possible to choose the day of the orbit data and the length of the observed time in days.



Figure 20: Search window 3

After that the systems shows all files with the chosen parameters. Then it is possible to select all or some of them and to put them by clicking on the “choose files”-button in the “shopping-basket”. It is also possible to sort them by size ascending/descending or by file ascending/descending or to select all or none of them by clicking on the corresponding button.



Figure 21: Search window 4

3.3.2 “Shopping basket”

The system puts the chosen files in the “shopping-basket”. Now it’s possible again to sort them or to delete files from it. The files are stored in the shopping basket until they will be deleted. If the user logged out from the system, the chosen files remain in his shopping-basket. Next time he logs in, he can delete them or send a request.

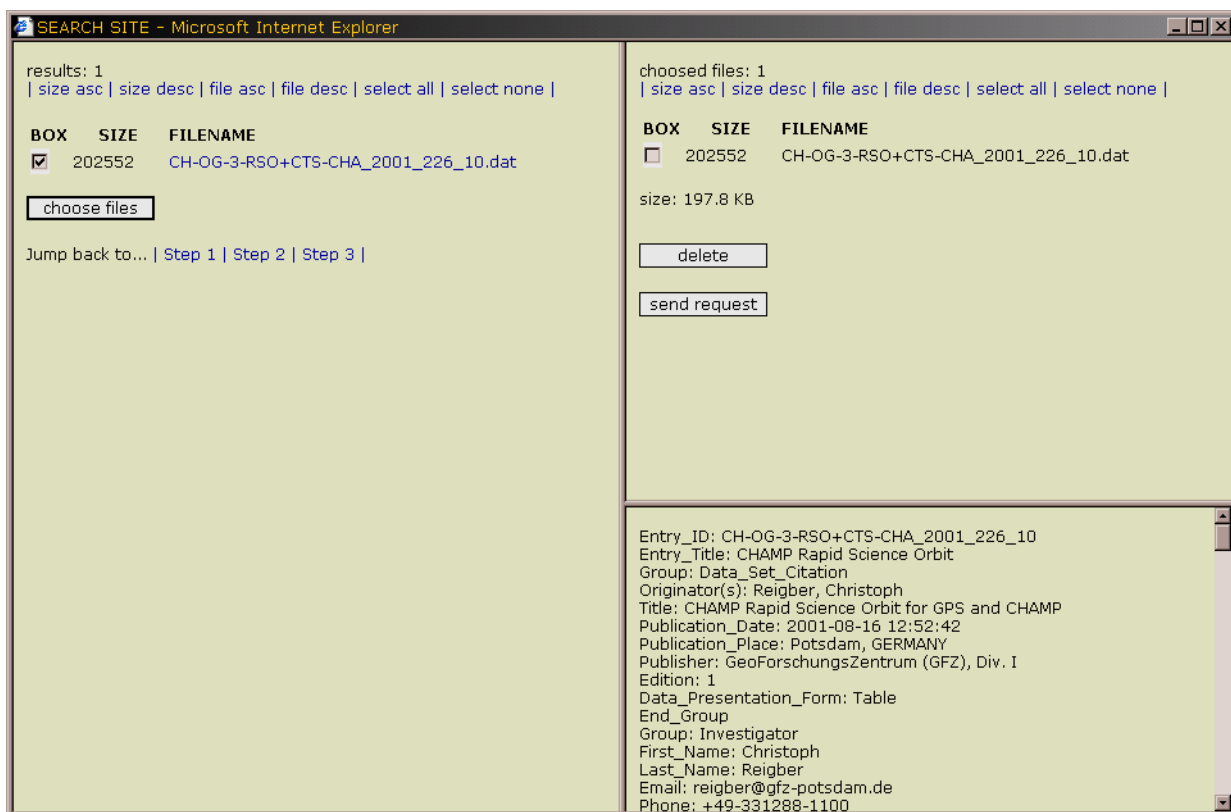


Figure 22: Search window 5

3.3.3 Send request

By clicking on the “send request”-button the system puts the files in an auto job process. The system has no real time copy function. The auto job process will be executed at fixed times. This process copies all chosen files of all users into his/her download-directory.

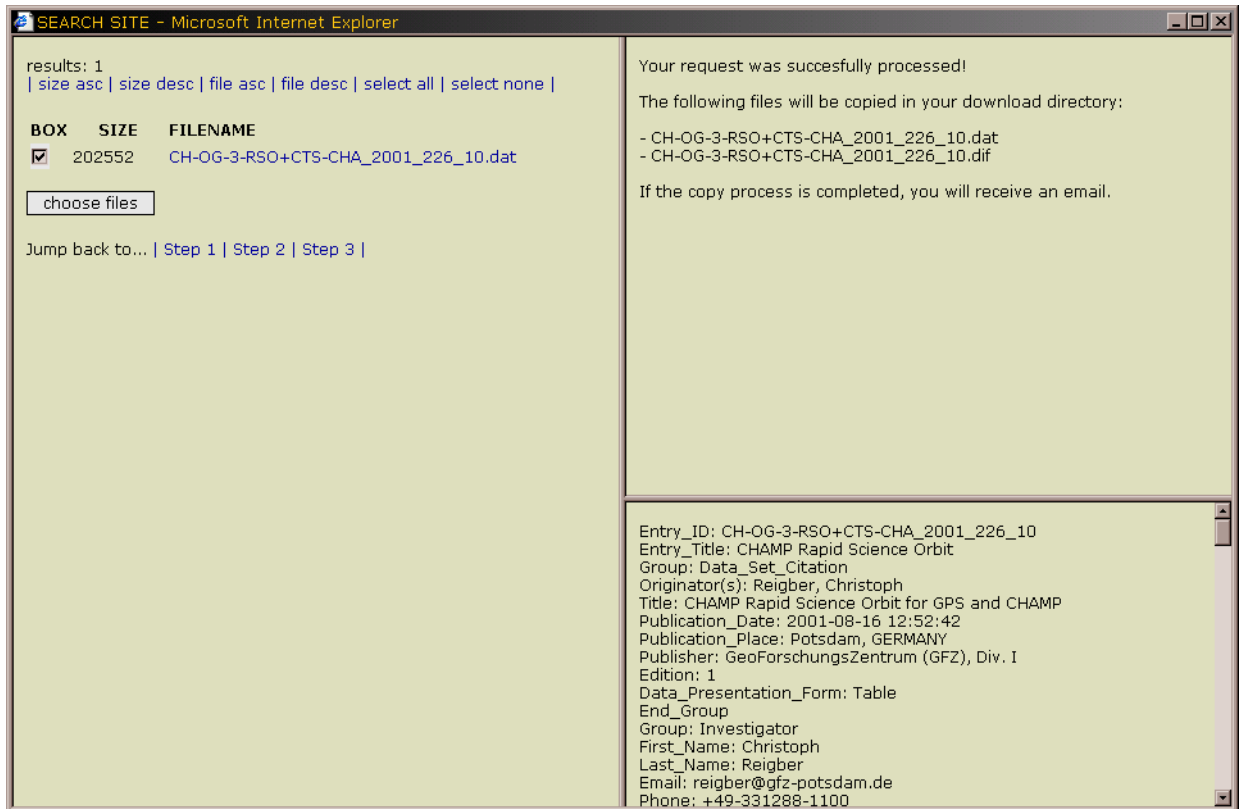


Figure 23: Search window 6

3.3.4 Background jobs

The user receives an email if the copy process is done and he can login via ftp in his directory and download the files. All copied files will be saved in a history log table in the user's database.

4 Conclusion and future aspects

The main goal of the project is reached. This means that a web based prototype of the database is running; no costs of software incurred. The system is as save as possible.

Future aspects could be:

- Automatisation of adding users (script based)
- Optimisation of the source-code
- Changing from a FTP server to a SFTP server (with secure socket layer)
- Improvement of the search-algorithm (adding search features, working with cached search results)
- Testing of the system with different users and more workload (stable or not?)

5 Indexes

5.1 Table Of Figures

Figure 1: Levels.....	9
Figure 2: Level 3 - category	9
Figure 3: Level 3 - products	10
Figure 4: Structure of "user"-Table	11
Figure 5: Example of "user"-table	13
Figure 6: Structure of "host"-table	13
Figure 7: Structure of "db"-table	14
Figure 8: Example of "db"-table.....	15
Figure 9: Structure of "tables_priv"-table	15
Figure 10: Example of "tables_priv"-table.....	16
Figure 11: Structure of "columns_priv"-table	16
Figure 12: Structure of "user_session"-table	17
Figure 13: Example of "user_session"-table	18
Figure 14: Structure of "info"-table.....	18
Figure 15: Example of "info"-table	18
Figure 16: Structure of "download"-table	20
Figure 17: Structure of "downloadhistory"-table	20
Figure 18: Search window 1.....	21
Figure 19: Search window 2.....	22
Figure 20: Search window 3.....	23
Figure 21: Search window 4.....	24
Figure 22: Search window 5.....	25
Figure 23: Search window 6.....	26

6 Bibliography

6.1 Online sources

- <http://www.php.net/manual/de/ref.session.php> [10.06.2003]
- <http://keasyphp.org/php/Sessions/phpSecureSessions.inc.phps> [10.06.2003]
- <http://www.mazenphp.de/pipermail/ugffm/2002-June/000131.html> [10.06.2003]
- <http://hr.uoregon.edu/davidrl/lamp/> [10.06.2003]
- <http://www.netzmafia.de/skripten/unix/unix8.html> [10.06.2003]
- http://www.zdnet.de/internet/artikel/java/200202/phpemails_03-wc.html [10.06.2003]
- <http://www.dclp-faq.de/search.php?q=php.ini&l=20> [10.06.2003]
- <http://www.brtnet.org/linux/lamp.htm> [10.06.2003]
- http://www.phpforum.de/redesign/links/anzeigen.php?rubrik_id2=7&PHPSESSID=
[10.06.2003]
- <http://www.phpwelt.de/tutorials/tutorials.php?tid=102&PHPSESSID=bd1362ee533bc77382444ade0754f85e> [10.06.2003]
- <http://www.phpwelt.de/tutorials/tutorials.php?tid=78&PHPSESSID=> [10.06.2003]
- <http://www.phpbuilder.com/columns/moon19990716.php3> [10.06.2003]
- http://www.php-center.de/beitraege/detail.php?a_id=184 [10.06.2003]
- <http://www.selfphp.info/> [10.06.2003]
- <http://www.washington.edu/computing/web/publishing/php-ini.html> [10.06.2003]