

CRAF RFI Database: a MySQL implementation

Sergio Poppi

INAF- Cagliari Astronomical Observatory

In collaboration with

CRAF

Presentation Contents

- Description of the CRAF EMI and Spectrum Occupancy Database (CRAF-01-02-rev.2)
- MySQL databases.
- Implementation CRAF EMI and Spectrum Occupancy database using a MySQL engine and web based clients.

CRAF EMI & Spectrum occ. DB

- CRAF EMI and Spectrum Occupancy Database: CRAF-01-02-rev.2 provide an Interference Report, which electronically is sent to the CRAF clearing house.
- The file format for the CRAF database was adopted in the 18th CRAF meeting (Grenoble, 10-11 April 1995) and revised in 23rd the 23 CRAF meeting (St.Petersburg, 24-25 April 1997);
- The input to the database file is a simple ASCII file with a record-length of 80 characters.
- The transfer of data files to CRAF is done via the ftp facility.

Database fields

10	INT_UNIT	Character	2	the number 999999 should be specified. Intensity unit: KE = Kelvin, JY = Jansky Note: the intensity is uncalibrated: it can only be calibrated if it is known where in the antenna pattern the interfering transmission is received and the antenna pattern is known.
11	RFI_AZ	Character	3	Azimuth of EMI source in degrees (if available); 'AAA' if azimuth is not defined.
12	RFI_EL	Character	2	Elevation of EMI source in degrees (if available); 'EE' if elevation is not defined.
13	TYPE	Character	2	kind of observation: BR = broadband SP = spectral
14	ANT_AZ	Character	3	Azimuth of observation (in degrees); 'AAA' if azimuth is not defined
15	ANT_EL	Character	2	Elevation of observation (in degrees); 'EE' if elevation is not defined.
16	DEG	Character	3	degree of degradation in percent. For spectrum occupancy data: '000'.
20	EOR	bytes	1	specify '=' to identify end-of-record
** Total **			80 bytes	

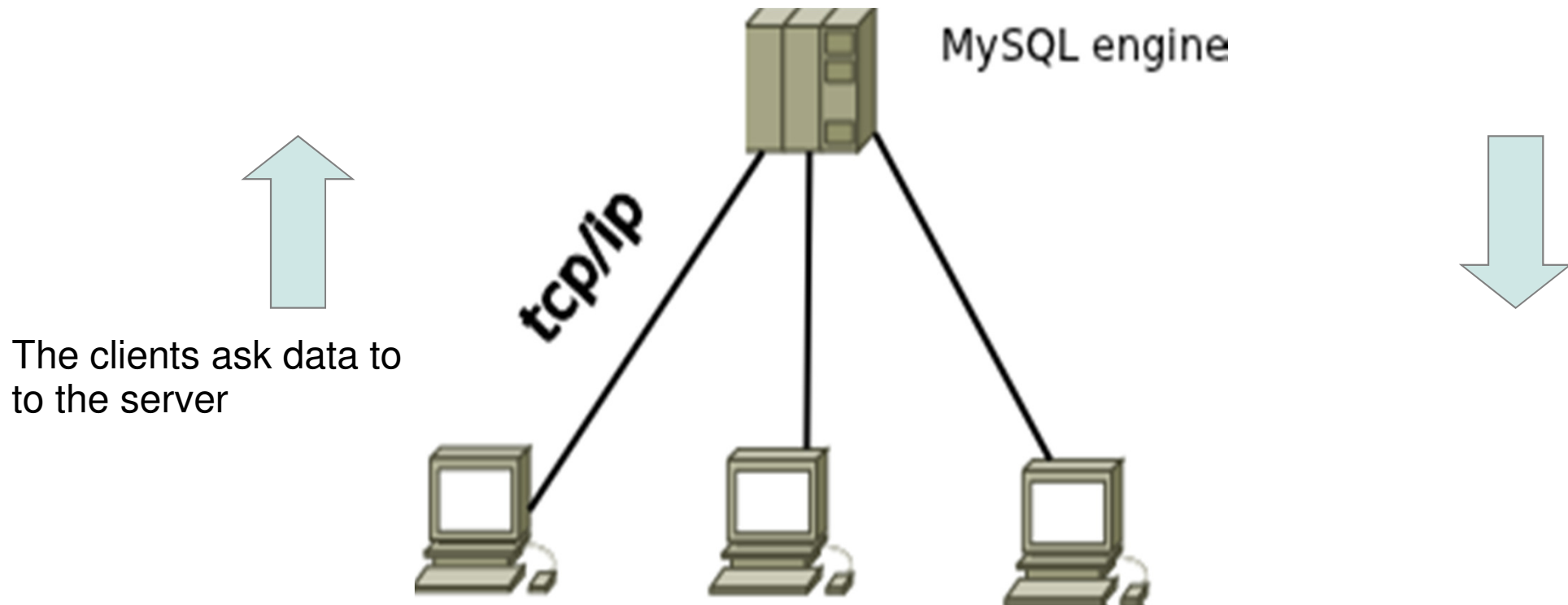
Database features

- Data stored in a rigid structure;
- Operator must format data in suitable form (80 chars ascii);
- Data analysis made with server side scripts;

Client/server architecture

A new database

The server sends to the clients the requested data



MySQL

- **MySQL** is the world's most widely used open source relational database management system:
 - A **relational database** is essentially a group of **tables** (entities).
 - **Tables** are made up of columns (attributes) and rows (tuples).
 - **Tables** have constraints, and relationships are defined between them.
 - Relational databases are queried using **SQL** , and result sets are produced from **queries** that access data from one or more tables.

Table "Books"

id	author	title	Editor	price
1	Alighieri,D.	La Divina Commedia	Mondadori	18 €
2	Kraus, J.D.	Radioastronomy	Cyg. Quasar	50€

SQL - Structured Query Language

- SQL is a special-purpose programming language designed for managing data, held in a relational database management systems (RDBMS)

- Queries

```
SELECT *  
FROM Book  
WHERE price > 10.00  
ORDER BY title;
```

- Database Management

```
CREATE TABLE My_table(  
  my_field1 INT,  
  my_field2 VARCHAR(50),  
  my_field3 DATE NOT NULL,  
  PRIMARY KEY (my_field1, my_field2)  
);
```


Database maintenance

- The SQL strings allow the maintenance of the database:
 - Create tables
 - Add columns
 - Grant users
 - Add new users
 - Give permissions to users
 - Specialize the users' privileges

The MySQL implementation

- Based on CRAF EMI and Spectrum Occupancy Database (CRAF-01-02-rev.2)
- Architecture: client and server
- Scalable (its ability to be enlarged to accommodate the growth)
- Server made with open source software;
- Clients can be written with several programming languages

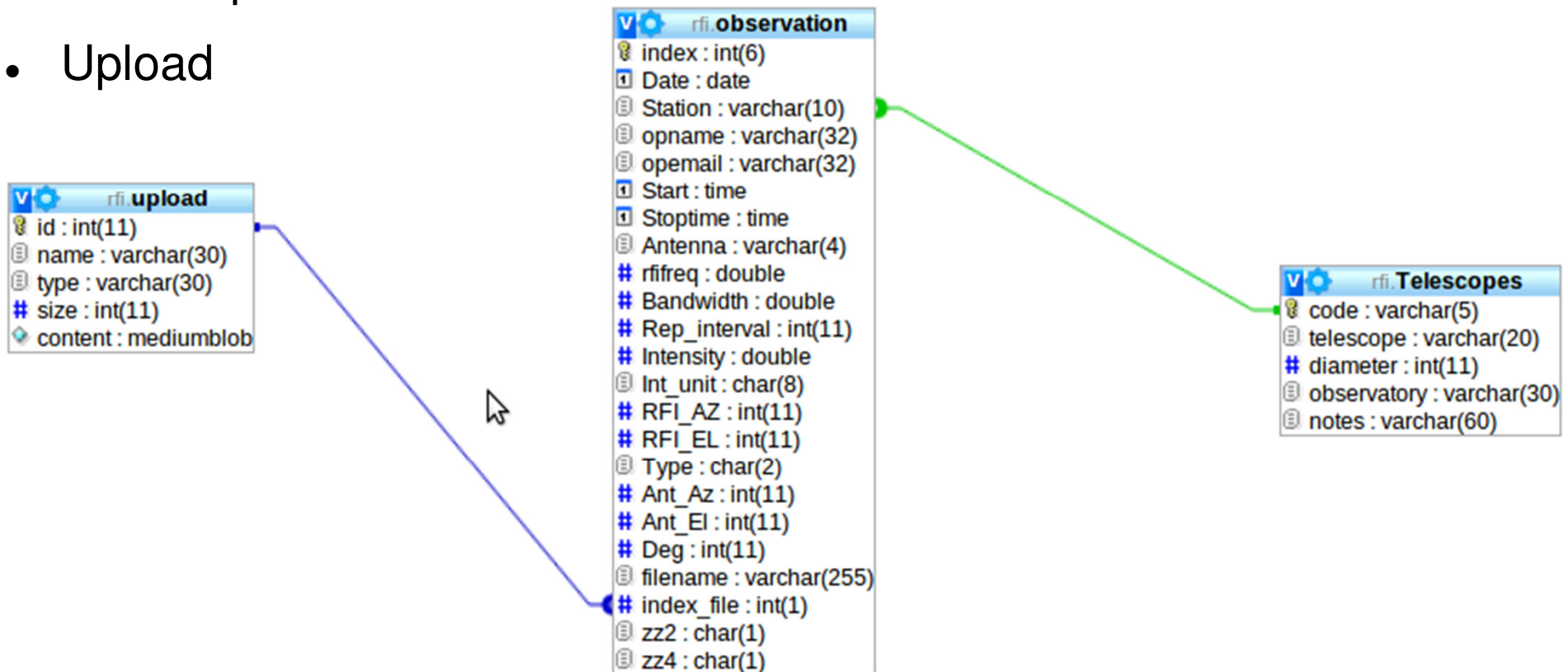
CRAF DB: MySQL implementation

- On the server side:
 - MySQL Server;
 - Apache web server, with php module
- On the client side:
 - Web browser

CRAF DB: MySQL implementation

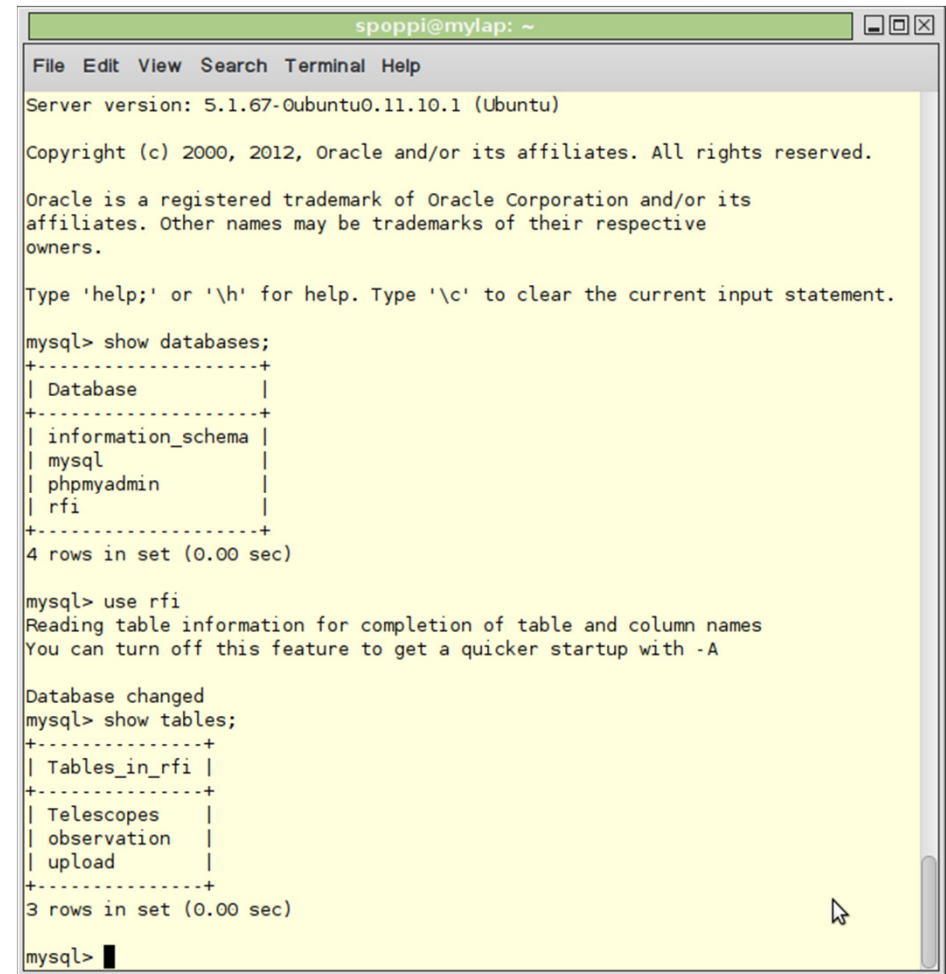
Designed in 3 tables

- Observation
- Telescope
- Upload



CRAF MySQL DB administration

- DB Administration with command line interface or through web interface (phpMyAdmin);



The screenshot shows a terminal window titled 'spoppi@mylap: ~'. The terminal displays the MySQL command-line interface. The user has entered 'mysql> show databases;', which returns a list of four databases: information_schema, mysql, phpmyadmin, and rfi. The user then enters 'mysql> use rfi', which changes the current database to 'rfi'. Finally, the user enters 'mysql> show tables;', which returns a list of three tables: Telescopes, observation, and upload. The terminal output is as follows:

```
spoppi@mylap: ~
File Edit View Search Terminal Help
Server version: 5.1.67-0ubuntu0.11.10.1 (Ubuntu)
Copyright (c) 2000, 2012, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| phpmyadmin |
| rfi |
+-----+
4 rows in set (0.00 sec)

mysql> use rfi
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_rfi |
+-----+
| Telescopes |
| observation |
| upload |
+-----+
3 rows in set (0.00 sec)

mysql>
```

phpMyAdmin

localhost / localhost / rfi | phpMyAdmin 3.4.5deb1 - Google Chrome

localhost / localhost / rfi | phpMyAdmin 3.4.5deb1 - Google Chrome

localhost/phpmyadmin/index.php?db=mysql&token=3e78fc8abe782450f6b6b4e6b5804ace

Getting Start... Facebook | Ser... Internet Mobile ... Beckhoff Inform... ENGLISH PAG... www.aoc.nrao... 4ce3c88bf70a4...

phpMyAdmin

localhost ▶ rfi

Structure SQL Search Query Export Import Operations More

Table	Action	Rows	Type	Collation	Size	Overhead
observation	Browse Structure Search Insert Empty Drop	95	MyISAM	latin1_swedish_ci	12.6 KiB	-
Telescopes	Browse Structure Search Insert Empty Drop	21	MyISAM	latin1_swedish_ci	2.8 KiB	-
upload	Browse Structure Search Insert Empty Drop	83	MyISAM	latin1_swedish_ci	9.5 MiB	-
3 tables	Sum	199	MyISAM	latin1_swedish_ci	9.5 MiB	0 B

Check All / Uncheck All With selected:

Print view Data Dictionary

Create table on database rfi

Name: Number of columns:

Network Disconnected - you are now offline Go

Clients

- Clients can be implemented with several languages:
 - C++
 - Php
 - java
 - python
 - Labview
 - etc.. etc...
- Web based clients can communicate with the server using web browsers

Web based clients

- The clients allow to
 - report the RFI measure with a browser;
 - retrieve the archived measures;
- The web pages are dynamically generated using php;
- Php scripts handle the connection with the database

Operator interface:report a RFI

RFI Database - Google Chrome

localhost/rfi/rfidb/

Getting Start... Facebook | Ser... Internet Mobile ... Beckhoff Inform... ENGLISH PAG... www.aoc.nrao... 4ce3c88bf70a4...

data submitted:

Date: 2013-04-05

Station: srt64

Operator's name: Sergio Poppi

Operator's email: spoppi@oa-cagliari.inaf.int

Start time: 12:33:04

Stop time: 14:00:00

Rfi frequency: 2000

bandwidth:

Repetition Interval:

Intensity: 10

Intensity unit: Jy

RFI azimuth :

RFI elevation:

Type:

Antenna azimuth:

Antenna elevation:

Degradation:

Query RFI information (beta version)

Get from data from **Start date** to **Stop date**.
Start date blank, gives data untill **Stop date**
If **Stop date** is let blank, all the database entries from **Start date** are shown.
If both the form field are left empty, all the database is shown.

The fields start and stop freq act in the same way of the as the date fields. If the freq. fields are blank, the query is done by date.
Otherwise, if a query by **frequency** is needed, the date fields must be blank.
If all the fields are filled, the query is a logical AND between date and frequency.

Start Date (yyyy-mm-dd): ...

Stop Date (yyyy-mm-dd): ...

Start Freq(MHz):

Stop Freq(MHz)

Station

Delete Entry

Obs. Index

Training Pres....pptx

Show all downloads...

Date helper

my Technical Forum - April, 8th – 12th 2013, Bonn

Retrieve data from the archive

Activities Google Chrome submit data - Google Chrome

localhost/rfi/rfidb/sendquery.php

Getting Start... Facebook | Ser... Internet Mobile ... Beckhoff Inform... ENGLISH PAG... www.aoc.nrao... 4ce3c88bf70a4... Other Bookmarks

Index	Date	Station	Operator	Start Time	End Time	Antenna	RFI Freq.	Bandwidth	Rep. Interval	Intensity	Intensity Unit	RFI Azimuth	RFI Elevation	Type	Antenna Azimuth	Antenna Elevation	Degradation	File Name
74	2010-04-05	srt64	Pietro Bolli	22:10:00	22:20:00	64	1500	10	-1	0		120	10	sp	0	0	10	
73	2010-04-07	srt64	SP	22:00:00	23:00:00	150	22000	200	0	0		0	0	sp	0	0	0	
83	2012-11-05	srt64	Pietro Bolli	19:00:00	20:00:00	64	1453	10	0	250	K	234	45	sp	240	42	0	800_1300_polV_est_ampl_maxhold.pdf
82	2012-11-05	srt64	Pietro Bolli	19:00:00	20:00:00	64	1453	10	0	250	K	234	45	sp	240	42	0	
81	2012-11-14	srt64	Pietro Bolli	23:30:00	23:40:00	64	1421	2	-1	100	K	245	45	br	275	50	0	
80	2012-11-14	srt64	Pietro Bolli	22:30:00	22:45:00	64	1420	2	-1	100	K	245	45	br	280	50	0	
87	2012-11-26	srt64	Sergio Poppi	12:00:00	23:00:00	32	24000	0	0	0	?	0	0	?	0	0	0	1620.pdf
160	2013-04-05	srt64	Sergio Poppi	12:33:04	14:00:00	64	2000	0	0	10	Jy	0	0		0	0	0	
161	2013-04-05	srt64	Sergio Poppi	12:33:04	14:00:00	64	2000	0	0	10	Jy	0	0		0	0	0	

```
// Example of SQL query
if ($station == 'all')
{
    $query = 'SELECT * FROM observation ORDER BY `Date`
ASC;';
} else
...
$results= mysql_query($query, $db)
```

Improvements

- Define new attributes to better describe the observation to be reported;
- Make the data report more effective;
- Write new clients to be suitable for unmanned data recording **for automatic monitoring** equipments (labview client for lab devices?)

Conclusions

- Based on CRAF EMI and Spectrum Occupancy Database (CRAF-01-02-rev.2)
- The structure of the database is flexible (updates possible with SQL commands)
- Multi users, the access privileges can be specialized.

Conclusions

- Web based clients allow to send and retrieve RFI reports to the database through a web browser;
- New clients can be developed using the most suitable programming language;
- Made with open source software!

Contact

- mailto: spoppi at oa-cagliari.inaf.it
- DB link: <http://srt-wp4.oa-cagliari.inaf.it/~rfi/rfidb/>