RD50 DB

A first prototype

D. M. Creanza, S. My

Dipartimento Interateneo di Fisica & INFN - Bari

First Tables

```
mysql> use RD50
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_RD50 |
+----------------+
| MeasConditions |
| MeasType       |
| Measurement    |
+----------------+
3 rows in set (0.00 sec)
```

Of course this is a minimal set of tables, just to start testing the scripts.
The next important table to define will be the Device table.
Also the definition (the structure, the content etc.) of these tables (see next slide) will be modified according to the occurring needs, as the construction of the DB evolves.

Description of the Tables

```
mysql> describe MeasConditions;
+------------+---------------+------+-----+---------+-------+
| Field      | Type          | Null | Key | Default | Extra |
+------------+---------------+------+-----+---------+-------+
| ID         | int(11)       | NO   | PRI | 0       |       |
| MeasID     | int(11)       | NO   | PRI | 0       |       |
| MeasName   | varchar(50)   | YES  |     | NULL    |       |
| Comment    | varchar(1000) | YES  |     | NULL    |       |
| Temp       | float(5,2)    | YES  |     | NULL    |       |
| Fluence    | varchar(30)   | YES  |     | NULL    |       |
| CondName   | varchar(500)  | YES  |     | NULL    |       |
| CondValues | varchar(500)  | YES  |     | NULL    |       |
+------------+---------------+------+-----+---------+-------+
8 rows in set (0.00 sec)

mysql> describe MeasType;
+----------------+--------------+------+-----+---------+----------------+
| Field          | Type         | Null | Key | Default | Extra          |
+----------------+--------------+------+-----+---------+----------------+
| ID             | int(11)      | NO   | PRI | NULL    | auto_increment |
| MeasName       | varchar(50)  | YES  |     | NULL    |                |
| SourceVarName  | varchar(50)  | YES  |     | NULL    |                |
| MonitorVarName | varchar(50)  | YES  |     | NULL    |                |
| Comment        | varchar(700) | YES  |     | NULL    |                |
| Legend         | blob         | YES  |     | NULL    |                |
+----------------+--------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)

mysql> describe Measurement;
+-----------+--------------+------+-----+-------------------+-----------------------------+
| Field     | Type         | Null | Key | Default           | Extra                       |
+-----------+--------------+------+-----+-------------------+-----------------------------+
| ID        | int(11)      | NO   | PRI | NULL              | auto_increment              |
| DeviceID  | int(11)      | YES  |     | NULL              |                             |
| MeasName  | varchar(50)  | YES  |     | NULL              |                             |
| MeasTime  | varchar(40)  | YES  |     | NULL              |                             |
| Time      | timestamp    | NO   |     | CURRENT_TIMESTAMP | on update CURRENT_TIMESTAMP |
| VarName   | varchar(400) | YES  |     | NULL              |                             |
| VarValues | text         | YES  |     | NULL              |                             |
+-----------+--------------+------+-----+-------------------+-----------------------------+
7 rows in set (0.00 sec)
```

Source/Monitor Variables

```
mysql> select MeasName, SourceVarName, MonitorVarName from MeasType;
+----------+------------------+------------------+
| MeasName | SourceVarName    | MonitorVarName   |
+----------+------------------+------------------+
| Diode IV | Vback;Vfront;Vgr | Iback;Idiode;Igr |
| Diode CV | V1;V2            | I2;I2            |
| Strip IV | a                | b                |
| FFF      | D                | w;t              |
| GGGG     | D                | w;t              |
| LLL      | a                | v                |
+----------+------------------+------------------+
6 rows in set (0.00 sec)
```

Reasonable…

Fake (not correctly defined): just for development test

Web Interface

At the moment, we have working routines for:

- reading measurement files
- writing measurement results
  - to already existing meas-type DB tables

  OR
  - to new meas-type DB tables (defining them just before)

The nice feature is that this is accomplished automatically by a user-friendly web page, with selection menus and buttons, and requires no particular effort (or knowledge) to the operator. The only prescription is that the measurement file MUST be a text file with these characteristics:

- it contains several columns (one per measurement parameter, that can be a SOURCE or MONITOR parameter) separated by whatever number of blanks
- each raw represents one measurement point
- the first raw is an HEADER containing the names of each column (i.e. of each meas. parameter)

For example:

Vback	Vfront	Vgr	Itot	Idiode	Igr
0.0	0.0	0.0	2.5E-12	-1.0E-12	-1.5E-12
5.0	0.0	0.0	4.0E-11	-3.0E-11	-1.0E-11
10.0	0.0	0.0	5.0E-10	-3.5E-10	-1.5E-10

.....

How to proceed

Such a file shouldn't be difficult to produce for simple measurement (IV, CV etc.), whatever measurement setup you may have... at least we hope so.

Anyhow, not *all* measurements can be reduced to such a simplified format (for example: family of curves as a function of a parameter, etc.)

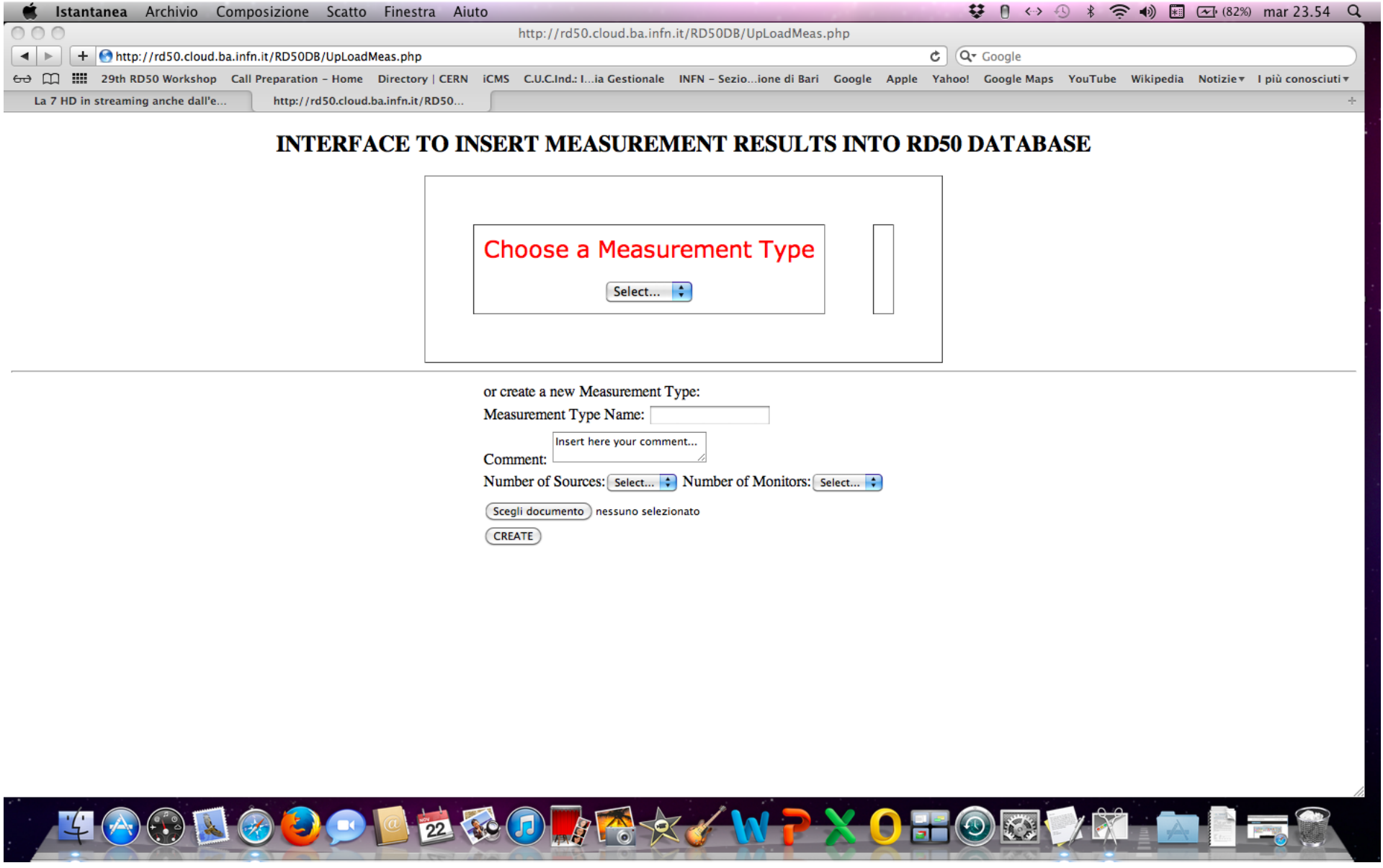
What we need to do now is to sit around a table and define the "standard" format for each individual kind of measurements, deciding also what is to be recorded in the DB as a *measurement value* and what can, instead, go into the *"comment"* field.

Once this issue is set, we can write the scripts that do the job.

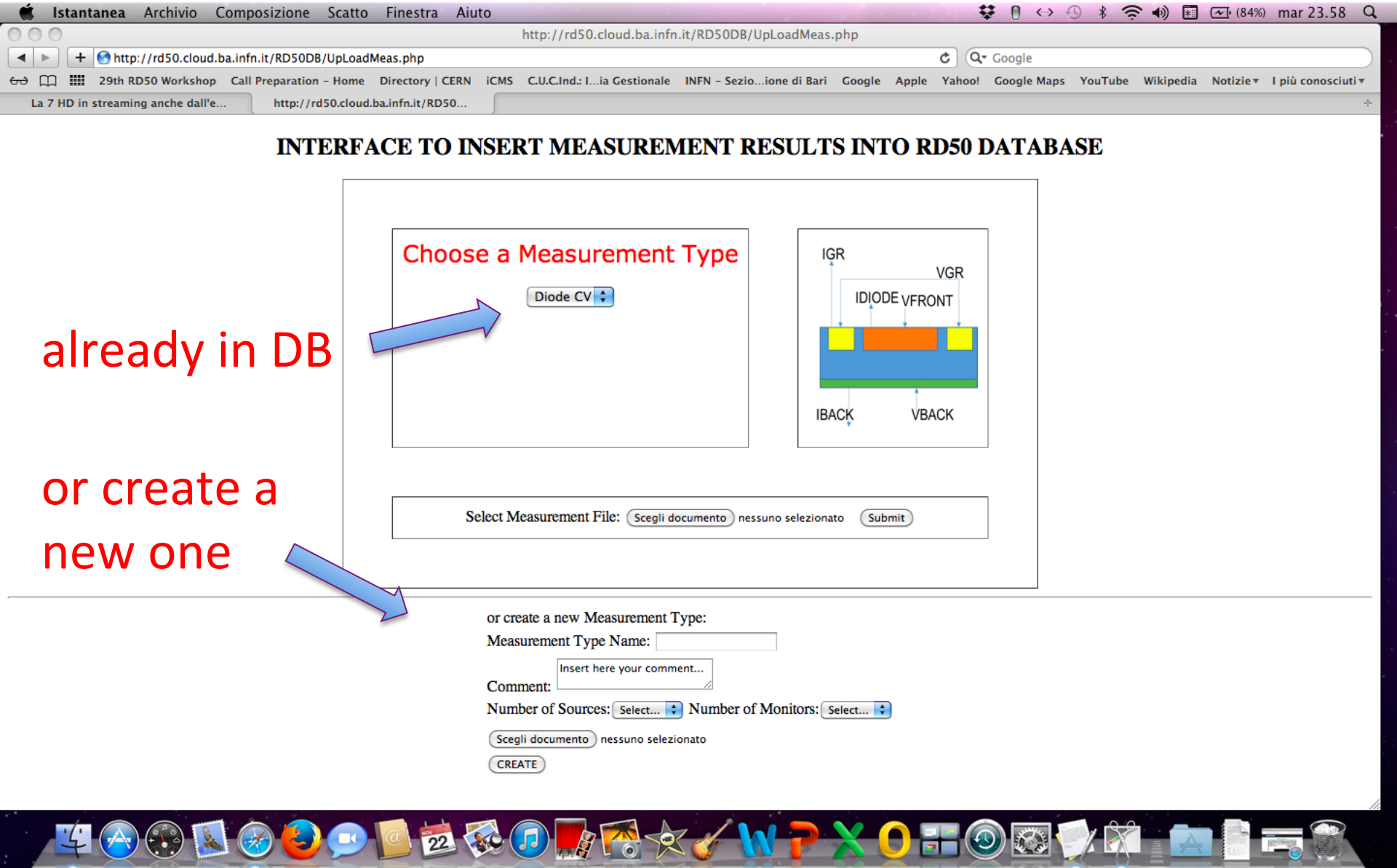
Few screenshots from the web site follow in next slides, for illustration purposes (don't look at graphic aspects, they will improve!).

P:S: Please, don't use the web site now, we are working on it and some functions are disabled.

Connect to the web site



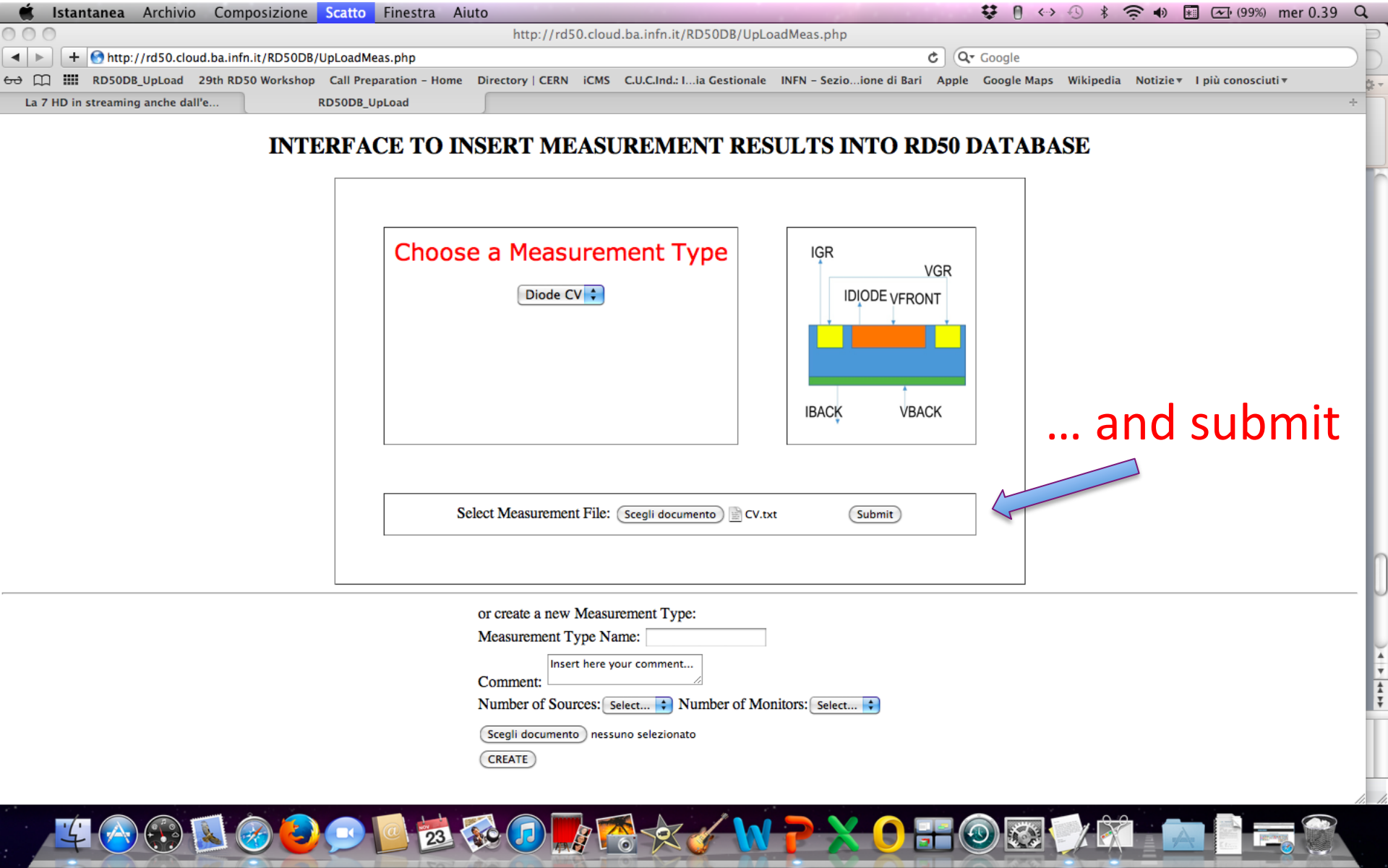
Choose a measurement type



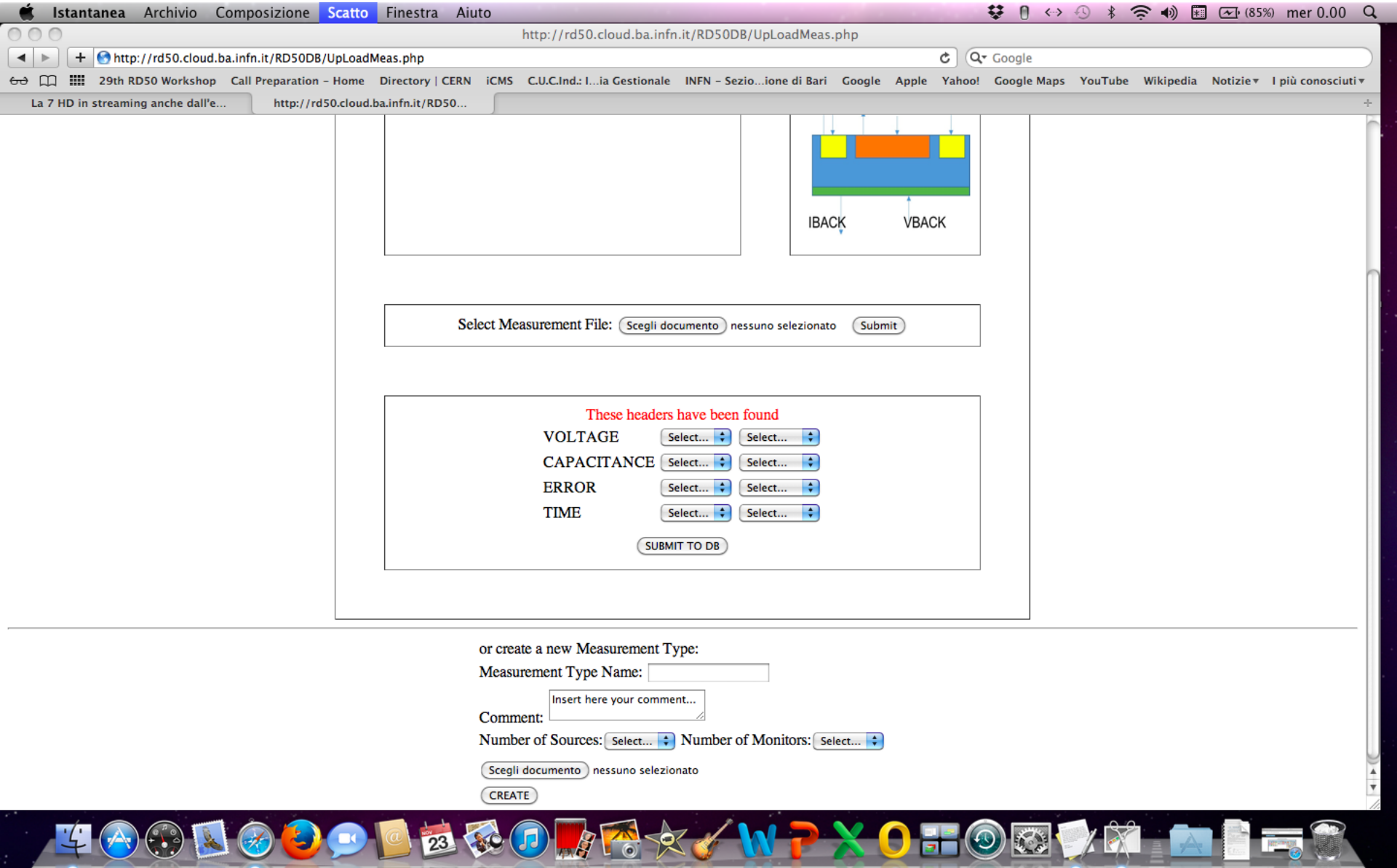
already in DB

or create a new one

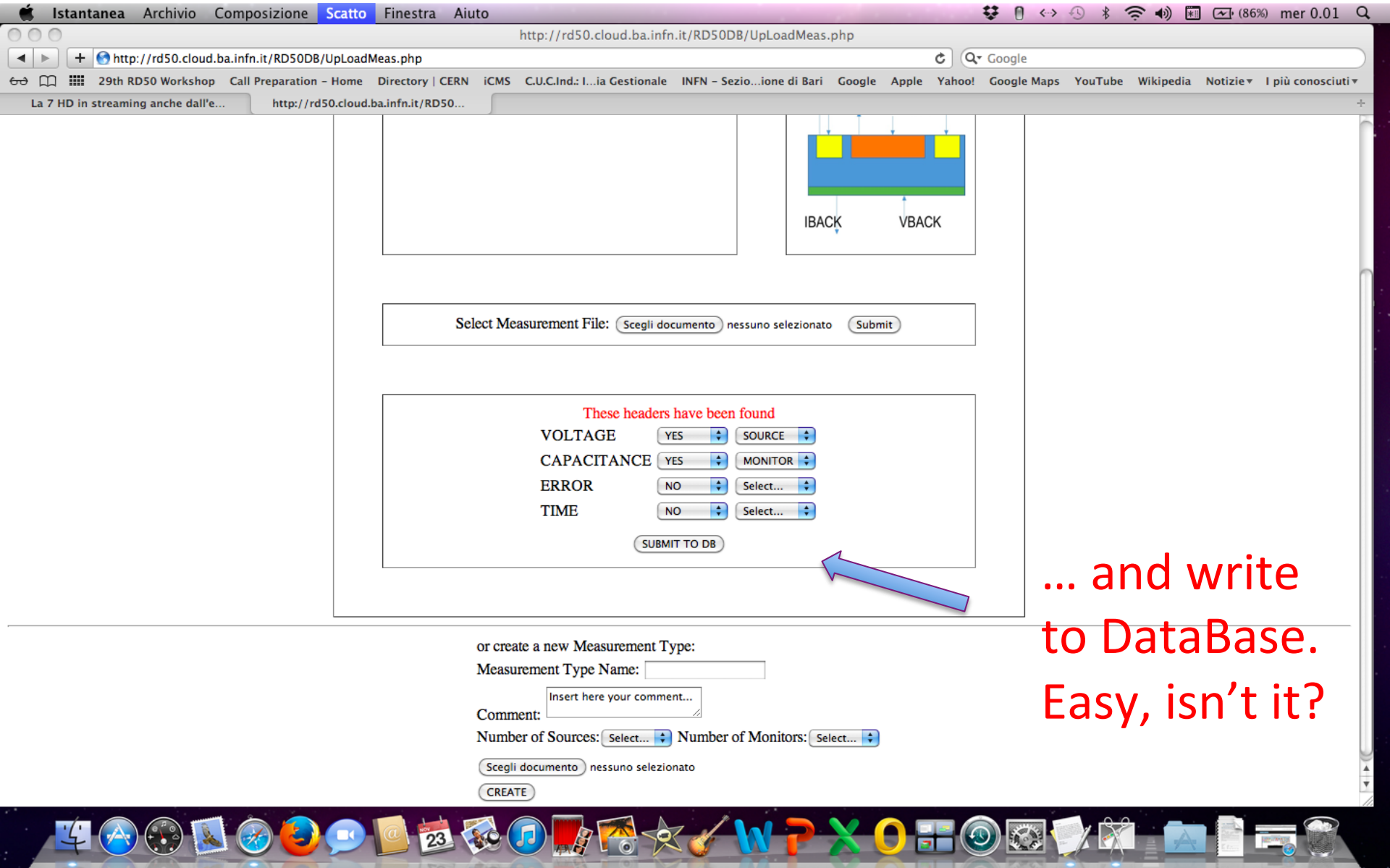
Upload your measurement file…



The variables in your header will be shown



Decide what to write in DB, define *S* & *M*…



… and write to DataBase. Easy, isn't it?