

Add new column (field) to the database

This guide shows how to add a new column/field to the database and update the package accordingly. The first part dedicated to the database applies to all tables in the database. The second part, where you are shown how to modify the package, is only necessary if you modify the Observation or Measurement tables in the database.

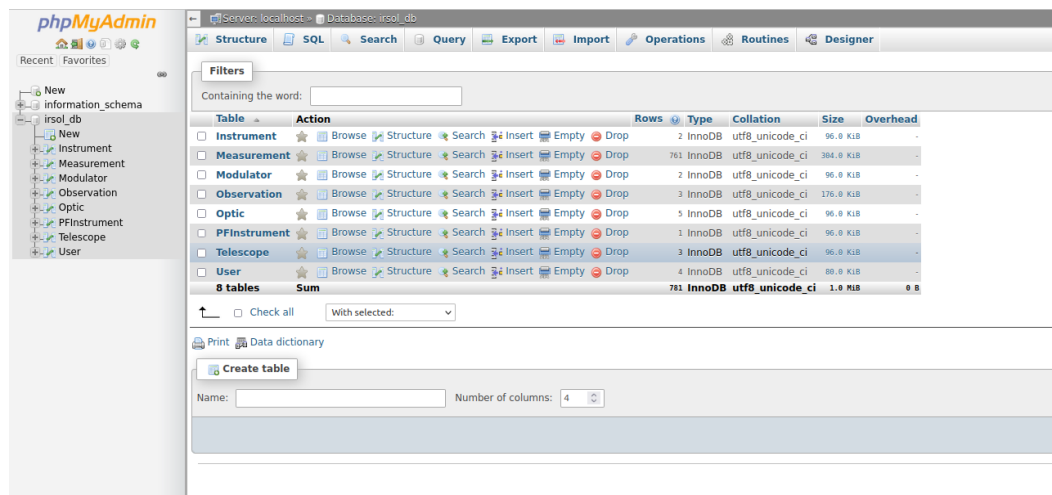
1. Database

The first thing to do is to access the interface for managing the database. To do so, open the following link: https://piombo7.ti-edu.ch/mysqladmin/db_structure.php?server=1&db=irsol_db You must log in to access the database. You can use the following credentials to do this:

Username: irsol_db

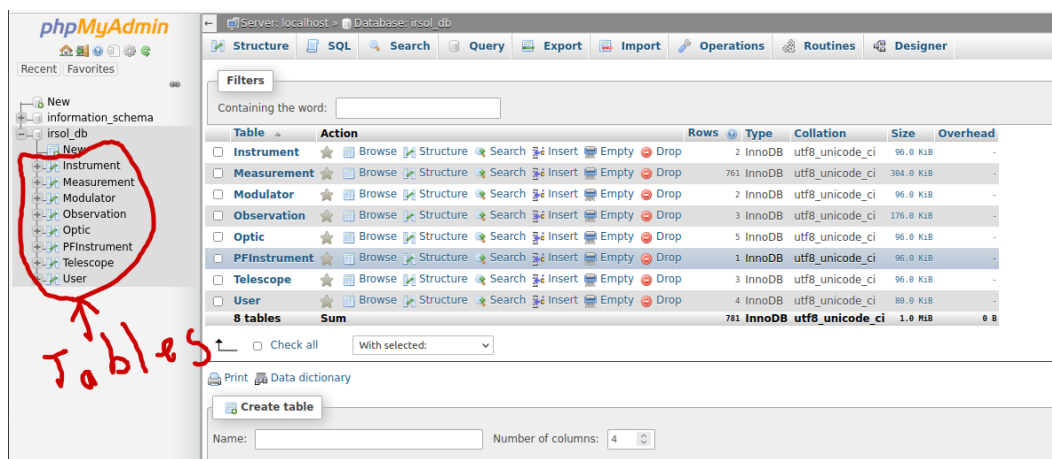
Password: SxHrxspjj8BgdjQBy

After logging in, you should find yourself on the screen shown in the following image.



The following operations are general for all tables. In this example we will take the table Measurement as a reference.

First of all, we select the table to which we wish to add a new field/column from the menu on the right-hand side of the screen. As shown in the following image.



In our example we will take the table Measurement. So press the table name. Once you have pressed the name of the table to which you wish to add the new field/column, the screen shown in the following image should appear.

Showing rows 0 - 24 (761 total, Query took 0.0111 seconds.)

SELECT * FROM `Measurement`

Number of rows: 25 Filter rows: Search this table Sort by key: None

	id_measurement	name	observer	cam_it	cam_tc	start_datetime	end_datetime
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	5355_mu0.10_m1sigU1.fits.gz	NULL	0.75000000	-14.92883515	2014-03-20 08:57:40	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	5355_mu0.40_m4sigU1.fits.gz	NULL	0.37500000	-14.97028154	2014-03-20 12:27:10	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	6045_center_m48sigU1.fits.gz	NULL	0.25000000	-14.98003853	2014-03-20 15:05:05	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	6045_mu0.40_m5sigU1.fits.gz	NULL	0.37500000	-14.98471217	2014-03-20 14:09:17	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	6045_center_m52sigU1.fits.gz	NULL	0.25000000	-14.98282801	2014-03-20 15:37:23	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	5355_center_m3sigQ1.fits.gz	NULL	0.25000000	-15.11488181	2014-03-20 09:07:32	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	5355_center_m5sigQ1.fits.gz	NULL	0.25000000	-15.10287201	2014-03-20 09:19:58	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	5355_center_m15sigQ1.fits.gz	NULL	0.25000000	-15.03948649	2014-03-20 10:16:34	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	9	5355_mu0.15_m2sigQ1.fits.gz	NULL	0.62500000	-14.93447314	2014-03-20 09:16:00	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	5355_center_m30sigQ1.fits.gz	NULL	0.25000000	-15.01290316	2014-03-20 12:39:06	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	11	5355_mu0.30_m3sigQ1.fits.gz	NULL	0.50000000	-14.96941612	2014-03-20 12:09:55	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	12	6045_mu0.20_m7sigU1.fits.gz	NULL	0.62500000	-14.97697851	2014-03-20 15:23:55	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	13	5355_center_m20sigU1.fits.gz	NULL	0.25000000	-15.07463925	2014-03-20 11:46:28	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	14	6045_mu0.25_m8sigQ1.fits.gz	NULL	0.50000000	-14.95150599	2014-03-20 15:47:02	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	15	5355_mu0.50_m4sigU1.fits.gz	NULL	0.37500000	-14.98912021	2014-03-20 12:36:42	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	16	5355_center_m13sigU1.fits.gz	NULL	0.25000000	-15.02430262	2014-03-20 10:06:28	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	17	5355_center_m23sigQ1.fits.gz	NULL	0.25000000	-15.05981620	2014-03-20 12:02:45	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	18	6045_center_m55sigQ1.fits.gz	NULL	0.25000000	-14.99718033	2014-03-20 15:58:33	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	19	5355_mu0.15_m1sigU1.fits.gz	NULL	0.62500000	-14.93095191	2014-03-20 09:10:32	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	20	6045_center_m42sigU1.fits.gz	NULL	0.25000000	-15.01220527	2014-03-20 14:13:34	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	21	6045_mu0.40_m6sigQ1.fits.gz	NULL	0.37500000	-14.97774799	2014-03-20 14:15:29	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	22	5355_refimg_sjc_m1.fits.gz	NULL	12.00300000	NULL	2014-03-20 07:14:53	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	23	5355_mu0.40_m1sigQ1.fits.gz	NULL	0.37500000	-14.99038523	2014-03-20 09:56:54	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	24	6045_mu0.10_m8sigU1.fits.gz	NULL	0.75000000	-14.96242079	2014-03-20 14:58:31	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	25	5355_mu0.40_m2sigU1.fits.gz	NULL	0.37500000	-14.98793778	2014-03-20 10:03:10	NULL

Check all With selected: Edit Copy Delete Export

In this view, press the "Structure" tab in the menu at the top of the screen.

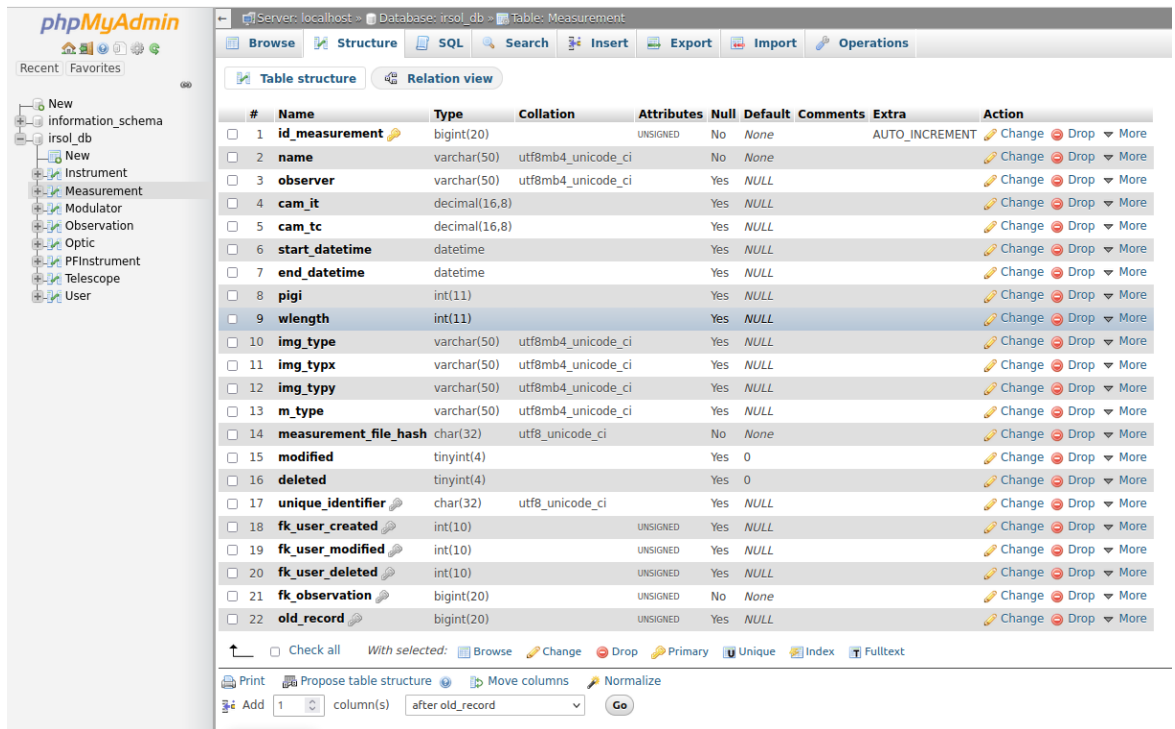
Showing rows 0 - 24 (761 total, Query took 0.0111 seconds.)

SELECT * FROM `Measurement`

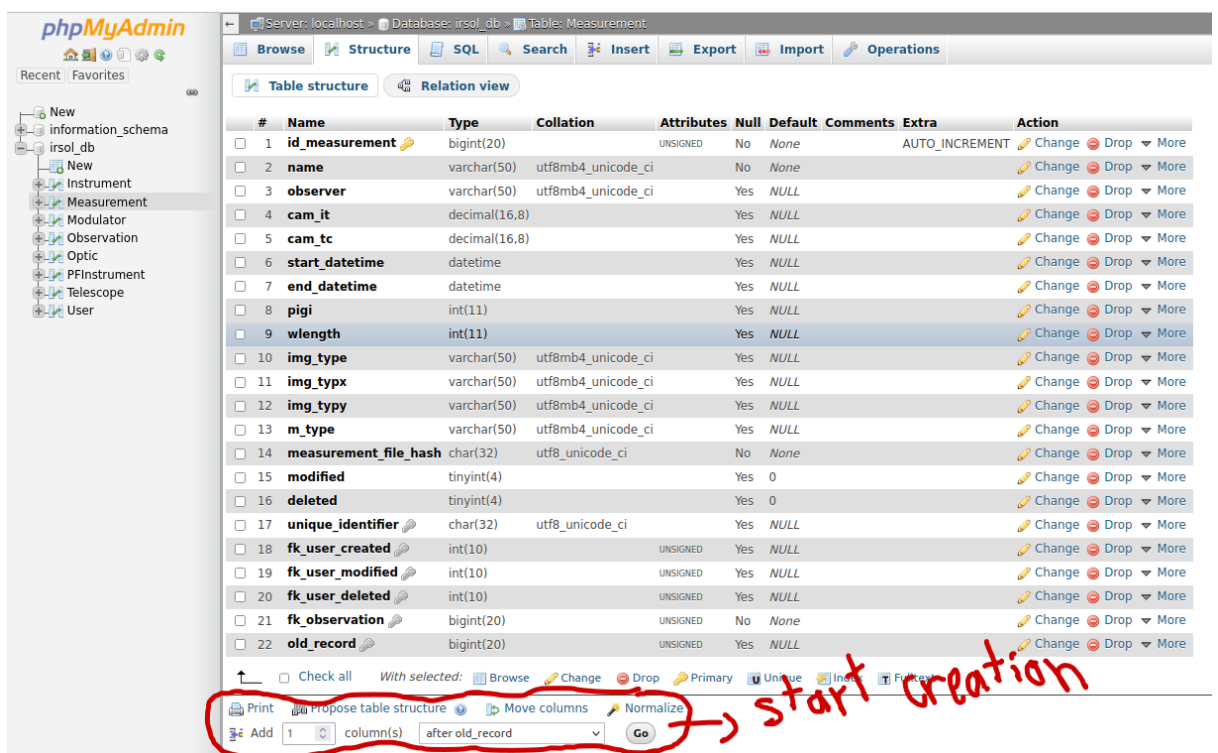
Number of rows: 25 Filter rows: Search this table Sort by key: None

	id_measurement	name	observer	cam_it	cam_tc	start_datetime	end_datetime
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	5355_mu0.10_m1sigU1.fits.gz	NULL	0.75000000	-14.92883515	2014-03-20 08:57:40	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	5355_mu0.40_m4sigU1.fits.gz	NULL	0.37500000	-14.97028154	2014-03-20 12:27:10	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	6045_center_m48sigU1.fits.gz	NULL	0.25000000	-14.98003853	2014-03-20 15:05:05	NULL

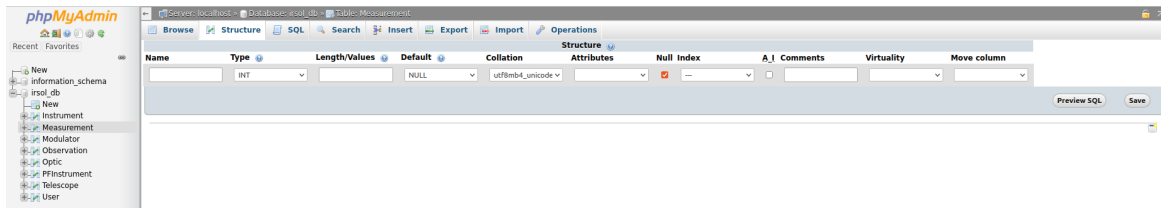
Once you click on the "Structure" tab, the screen shown in the following image should appear.



To add a column to the database, it is necessary to define after which column the new column is added. By default the best choice is to create it after the last existing column. In our example we add a column after the column "old_record". Once this configuration is set up you can press the "Go" button to start creating the new column. All this can be seen in the following image.



Once you press the "Go" button, you should see the screen shown in the following image.

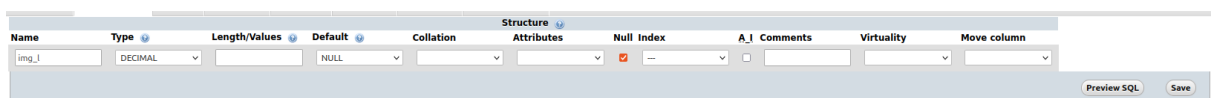


In this screen you are asked to define the fields that will characterize the new column to be added. The fields we fill in with information are as follows:

- **Name** => This field defines the name of the new column. For the name I recommend using all lowercase letters and separating words with underscores. In our example, the name of the new column will be "img_l". This name refers to the IMG_L header in the .z3db and .fits files.
- **Type** => This field defines the type of data that the column will contain. If the column contains integers, the type to use is INT. In the case of text, the type to use is VARCHAR(X), where X represents the maximum number of characters. X must be set in the next field, i.e. "Length/Values". In the case of a float, the type to be used is DECIMAL(M,D), where M is the maximum number of digits and D is the maximum number of decimals. M,D shall be set in the next field, 'Length/Values'. To be compatible with the search M must be equal to 32 and D to 16. Let's see them as default parameters. In our example, the values to be entered in the column are all integers. Therefore, the type chosen is INT.
- **Length/Values** => This field is used to specify the type of data contained by the column. As already explained above, if the type chosen is VARCHAR, in this field it is possible to specify the maximum number of characters that a value can contain. If, on the other hand, the type is DECIMAL, it is possible to define the maximum number of digits and decimals. Default value 32,16 . In our example this field is empty.
- **Default** => This field is used to define the default value of a column if the value for this column is not present during the insertion. It is also used at column creation time to assign a value to the column for all records already present in the database. By default you are advised to keep the value *NULL*. In our example the default value will be *NULL*.
- **Collation** => A collation is a set of rules that defines how to compare and sort character strings. This field is only to be filled if the column type is VARCHAR. By choice, the value for all columns in the database that contain a string is utf8mb4_unicode_ci (before the last value in the choice). In our example, the field remains empty.
- **Comments** => This field allows you to add a comment to the columns to understand what they represent.

For the creation of a simple column all other fields are not necessary. However, if you wish to have more complex columns, it is advisable to learn what the other fields are for.

The following image shows the value of the fields in our example.



Once all the values of the fields of interest have been set, click on the "Save" button to add the new column to the selected table. If all goes well, you should see the new column below

the last column of the table. As shown in the following image. To view the table again, click on the "Structure" tab in the menu at the top of the screen.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_measurement	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	name	varchar(50)	utf8mb4_unicode_ci		No	None			Change Drop More
3	observer	varchar(50)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
4	cam_it	decimal(16,8)			Yes	NULL			Change Drop More
5	cam_tc	decimal(16,8)			Yes	NULL			Change Drop More
6	start_datetime	datetime			Yes	NULL			Change Drop More
7	end_datetime	datetime			Yes	NULL			Change Drop More
8	pigi	int(11)			Yes	NULL			Change Drop More
9	wlength	int(11)			Yes	NULL			Change Drop More
10	img_type	varchar(50)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
11	img_tpx	varchar(50)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
12	img_tpy	varchar(50)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
13	m_type	varchar(50)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
14	measurement_file_hash	char(32)	utf8_unicode_ci		No	None			Change Drop More
15	modified	tinyint(4)			Yes	0			Change Drop More
16	deleted	tinyint(4)			Yes	0			Change Drop More
17	unique_identifier	char(32)	utf8_unicode_ci		Yes	NULL			Change Drop More
18	fk_user_created	int(10)		UNSIGNED	Yes	NULL			Change Drop More
19	fk_user_modified	int(10)		UNSIGNED	Yes	NULL			Change Drop More
20	fk_user_deleted	int(10)		UNSIGNED	Yes	NULL			Change Drop More
21	fk_observation	bigint(20)		UNSIGNED	No	None			Change Drop More
22	old_record	bigint(20)		UNSIGNED	Yes	NULL			Change Drop More
23	img_l	int(11)			Yes	NULL			Change Drop More

That's it for the database part.

2. Python Package

The following explains how to modify the python package to enable it to process the new columns. This part of the guide is only necessary if you have added columns to the Measurement or Observation table.

- **Measurement**

To enable the package to read a new header from the .z3db or .fits files, the package settings must be changed. First we need to access the settings.py file inside the package. It contains all settings. The file is located within the package at the following location "db_irsol_client/db_irsol/settings.py". Once the file is open, we need to locate the two constant variables `FIELDS_NAMES_Z3BD_FILE_TO_DATABASE_COLUMNS_NAMES` and `FIELDS_NAMES_FITS_FILE_TO_DATABASE_COLUMNS_NAMES`, as shown in the image below.

These two variables are dictionaries. Their task is to link the names of the headers in the .z3db and .fits files with the names of the columns in the database. In both dictionaries, the key of a record represents the name of a header, while its value represents the name of the relative column.

In our example above, we created the column `img_l` within the Measurement table. The `img_l` column refers to the `IMG_L` header. Therefore the record shown in the following image must be added to both dictionaries.

```
'IMG_L': 'img_l'
```

In the following image you can see the two updated dictionaries.

```

FIELDS_NAMES_Z3BD_FILE_TO_DATABASE_COLUMNS_NAMES = {
    'CAM_IT': 'cam_it',
    'CAM_TC': 'cam_tc',
    'DATE': 'start_datetime',
    'DATE_END': 'end_datetime',
    'PIGI': 'pigi',
    'WLENGTH': 'wlength',
    'IMG_TYPE': 'img_type',
    'IMG_TY PX': 'img_typx',
    'IMG_TY PY': 'img_typy',
    'M_TYPE': 'm_type',
    'OBSERVER': 'observer',
    'TELESCOP': 'fk_telescope-name',
    'IMG_L': 'img_l'
}

FIELDS_NAMES_FITS_FILE_TO_DATABASE_COLUMNS_NAMES = {
    'CAM_IT': 'cam_it',
    'IT': 'cam_it',
    'CAM_TC': 'cam_tc',
    'TC': 'cam_tc',
    'DATE': 'start_datetime',
    'PIGI': 'pigi',
    'WLENGTH': 'wlength',
    'IMG_T': 'img_type',
    'M_TYPE': 'm_type',
    'OBSERVER': 'observer',
    'TELESCOP': 'fk_telescope-name',
    'IMG_L': 'img_l'
}

```

The thing to keep in mind during this procedure is that the headers of .z3db files may have a different name from the headers of .fits files. Two headers with different names can show the same data that needs to be entered in one column. This is why there are two dictionaries and not just one. The following image shows the situation just described in red.

```

FIELDS_NAMES_Z3BD_FILE_TO_DATABASE_COLUMNS_NAMES = {
    'CAM_IT': 'cam_it',
    'CAM_TC': 'cam_tc',
    'DATE': 'start_datetime',
    'DATE_END': 'end_datetime',
    'PIGI': 'pigi',
    'IMG_LENGTH': 'img_length',
    'IMG_TYPE': 'img_type',
    'IMG_TYPIX': 'img_typx',
    'IMG_TYPTY': 'img_typy',
    'M_TYPE': 'm_type',
    'OBSERVER': 'observer',
    'TELESCOP': 'fk_telescope-name',
    'IMG_L': 'img_l'
}

FIELDS_NAMES_FITS_FILE_TO_DATABASE_COLUMNS_NAMES = {
    'CAM_IT': 'cam_it',
    'IT': 'cam_it',
    'CAM_TC': 'cam_tc',
    'TC': 'cam_tc',
    'DATE': 'start_datetime',
    'PIGI': 'pigi',
    'IMG_LENGTH': 'img_length',
    'IMG_T': 'img_type',
    'M_TYPE': 'm_type',
    'OBSERVER': 'observer',
    'TELESCOP': 'fk_telescope-name',
    'IMG_L': 'img_l'
}

```

Another detail to pay attention to is that the names of the headers have changed over the years. So two .fits files from different years may have two differently named headers describing the same data. In this case, both headers must be added to the dictionary. So the dictionary will have two new records with different keys but same value. The following image shows the above situation in red.


```

FIELDS_NAMES, FITS_FILE_TO_DATABASE_COLUMNS_NAMES = {
    'CAM_IT': 'cam_it',
    'IT': 'cam_it',
    'CAM_TC': 'cam_tc',
    'TC': 'cam_tc',
    'DATE': 'start_datetime',
    'PIGI': 'pigi',
    'WLENGTH': 'wlength',
    'IMG_T': 'img_type',
    'M_TYPE': 'm_type',
    'OBSERVER': 'observer',
    'TELESCOP': 'fk_telescope-name',
    'IMG_L': 'img_l'
}

```

- **Observation**

In case a new column is added to the observations table, it is necessary to modify the Python file that contains the functions to parse the observations log files. If you do not modify the file, the value of the added column will always be the default value for all records. The file to be modified is located in the python package folder at the following path “*db_irsol_client/db_irsol/parsers/log_parser.py*”.

The file contains the LogParser class and a single method. The modification is to be made within the method. The modification is nothing more than adding the new column to the parameter dictionary. The value of the new column must be extracted from the observation log file. So it is necessary to write your own extraction algorithm. The following image shows how the project name is extracted from the observation log file. For a new column, you have to do the same thing but change the column name (i.e. change the name of the key in the dictionary) and the algorithm for extracting the data to the right of the equal symbol.

```

def parse_log_file_first(log_file_path):
    parameters = {}
    # Open file and read all lines
    log = open(log_file_path, 'r')
    lines = log.readlines()

    # Get project
    parameters["project"] = lines[0].replace('<TITLE>', '').strip()

```

Replace with the name of the new added column

Extraction algorithm