

Some considerations about analysis software framework for EUDET pixel beam telescope

JRA1 Analysis Software meeting, Geneva, 26 January 2007.

Tatsiana Klimkovich
DESY

Contents

- ✓ CVS repository
- ✓ Software tools: input from ILC community
- ✓ Detector geometry description



CVS repository

- ✓ Created within ILC software repository (thanks to Harald Vogt)



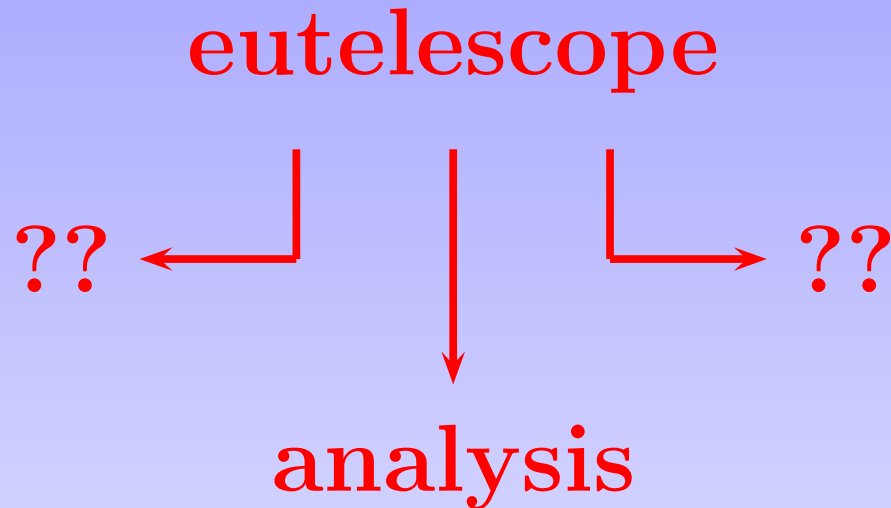
CVS repository

- ✓ Created within ILC software repository (thanks to Harald Vogt)

eutelescope

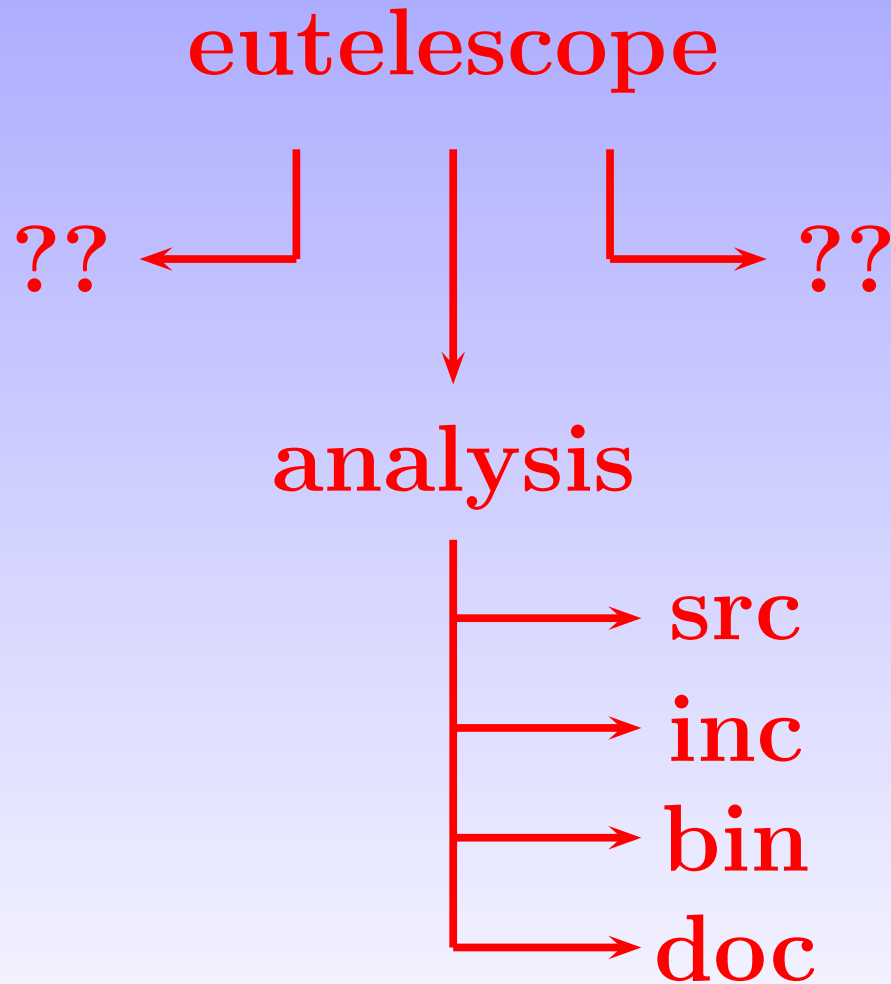
CVS repository

- ✓ Created within ILC software repository (thanks to Harald Vogt)



CVS repository

- ✓ Created within ILC software repository (thanks to Harald Vogt)



CVS repository

- ✓ Web interface will be available as soon as we have something to put there (for reading the last version of the package). See example of TPC repository “marlintpc”: <http://www-zeuthen.desy.de/lc-cgi-bin/cvsweb.cgi/MarlinTPC/?cvsroot=marlintpc>
- ✓ List of people developing the code:
 - ★ Aleksander Filip Zarnecki
 - ★ Antonio Bulgheroni
 - ★ Tatsiana Klimkovich
 - ★ ???



CVS repository: How to start?

- ✓ Ask me to add your name to developer list
- ✓ After it is ready type:

```
export CVS_RSH=ccvssh
export CVSROOT=:ext:username@cvssrv.ifh.de:/eutelescope
cvs co analysis
```

You will get all directories in “analysis”

- ✓ When somebody commits changes to the code all other people in the list receive emails
- ✓ We can tag different versions of our code and make for them web interface

CVS repository: How to start?

Main cvs commands:

- ✓ If you want to commit a change to the file:

```
cvcs ci -m "your comment" filename
```

If this is without -m then cvs will start up your default editor for you to type a log message

- ✓ If you want to add or delete file:

```
cvcs add filename
```

```
cvcs delete filename
```

The change will happen on CVS only if after that you type:

```
cvcs ci
```

- ✓ To update the working directory type in that directory:

```
cvcs update or cvcs up
```

Look at the comments:

M: modified, will be committed

U: file has been updated since you last checked it out

A: added, will be added on commit

D: deleted, will be deleted on commit

?: egregious file not part of CVS record for this dir



CVS repository: How to start?

- ✓ **To look at the log file:**

```
cvls log filename
```

- ✓ **To look at what you have changed to the file:**

```
cvls diff filename
```

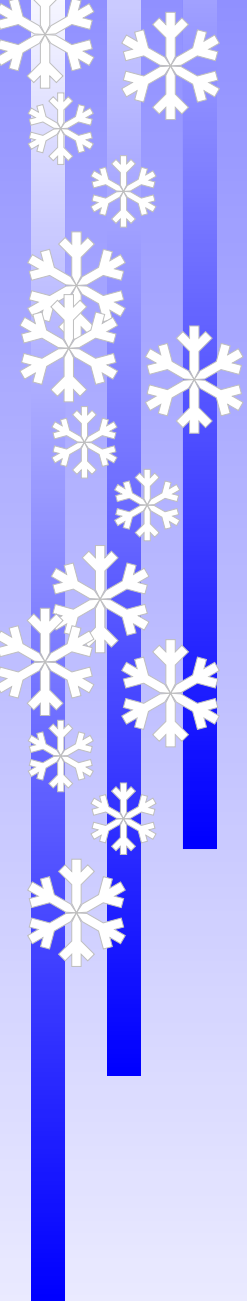
Why should we use ILC software?

- ✓ Useful experience for vertex detector R&D
- ✓ Stick to common standards
- ✓ Convenient for other users
- ✓ Easy to combine with Mokka simulation
- ✓ TPC and Calice are already using this software
- ✓ Some experience among vertex community already exists (Marco Battaglia)
- ✓ Should go one step further in comparison with Calice:
 - ★ use of GEAR or LCGO for geometry description
 - ★ get LCIO format data directly from DAQ (Calice converts binary files to LCIO)



Software tools: input from ILC community

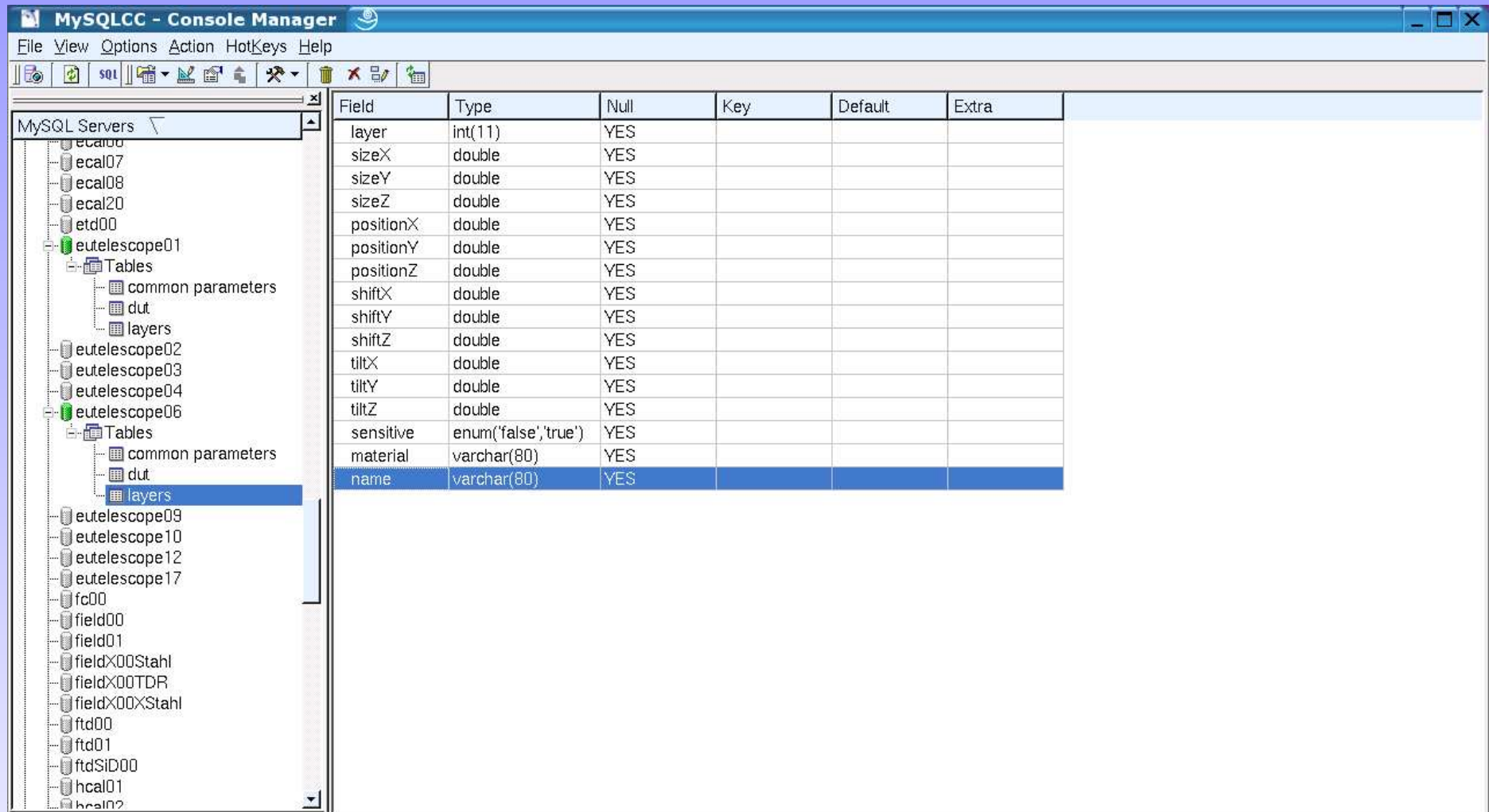
- ✓ Collaborate to ILC software developers (Frank Gaede is a leader of ILC software development at DESY)
- ✓ Use of experience of Calice and TPC as much as possible (e.g. Roman Pöschl and Sebastian Schmidt from Calice)
- ✓ Sebastian Schmidt can provide the tools for easy installation of ILC software packages
- ✓ ILC does not recommend to use self-defined LCIO classes (Generic Object classes) because they are slow. Should use plain LCIO classes



Detector geometry description in simulation

- ✓ Geometry parameters are in MySQL database
- ✓ Graphical interface exist

Detector geometry description in simulation



The screenshot shows the MySQLCC - Console Manager interface. On the left, a tree view displays a hierarchy of MySQL servers and databases. The 'eutelescope06' database is expanded, showing a 'Tables' folder containing 'common parameters', 'dut', and 'layers'. The 'layers' table is selected, and its field definitions are shown in the main pane.

Field	Type	Null	Key	Default	Extra
layer	int(11)	YES			
sizeX	double	YES			
sizeY	double	YES			
sizeZ	double	YES			
positionX	double	YES			
positionY	double	YES			
positionZ	double	YES			
shiftX	double	YES			
shiftY	double	YES			
shiftZ	double	YES			
tiltX	double	YES			
tiltY	double	YES			
tiltZ	double	YES			
sensitive	enum('false','true')	YES			
material	varchar(80)	YES			
name	varchar(80)	YES			

```
[Tatsiana] Querying MySQL Server for Database Information.  
[Tatsiana] Querying MySQL Server for Table information in database: models03  
[Tatsiana] Querying MySQL Server for Table information in database: mysql  
[consult@local] Querying MySQL Server for Database Information.  
[root@localhost] Querying MySQL Server for Database Information.  
[root@localhost] Querying MySQL Server for Table information in database: eutelescope01  
[root@localhost] Querying MySQL Server for Table information in database: eutelescope06
```

Detector geometry description in simulation

MySQLCC - [root@localhost] Query Window

File Edit View Query Options HotKeys

	layer	sizeX	sizeY	sizeZ	posit	positi	positionZ	shiftX	shift	sh	tiltX	tiltY	tiltZ	sensitive	material	name
1	1001	10	10	2.5	0	0	-100	0	0	0	0	0	0	false	polystyrene	Scintillator 1
2	2001	10	10	0.03	0	0	-55	0	0	0	0	0	0	false	aluminium	Al wall 1
3	1	7.7	7.7	0.5	0	0	-50	0	0	0	0	0	0	true	silicon_2.33gcc	Sensor 1
4	2	7.7	7.7	0.5	0	0	-30	0	0	0	0	0	0	true	silicon_2.33gcc	Sensor 2
5	3	7.7	7.7	0.5	0	0	-10	0	0	0	0	0	0	true	silicon_2.33gcc	Sensor 3
6	4	7.7	7.7	0.5	0	0	10	0	0	0	0	0	0	true	silicon_2.33gcc	Sensor 4
7	5	7.7	7.7	0.5	0	0	30	0	0	0	0	0	0	true	silicon_2.33gcc	Sensor 5
8	6	7.7	7.7	0.5	0	0	50	0	0	0	0	0	0	true	silicon_2.33gcc	Sensor 6
9	2002	10	10	0.03	0	0	55	0	0	0	0	0	0	false	aluminium	Al wall 2
10	1002	10	10	2.5	0	0	100	0	0	0	0	0	0	false	polystyrene	Scintillator 2

Columns

- layer
- material
- name
- positionX
- positionY
- positionZ
- sensitive
- shiftX
- shiftY
- shiftZ
- sizeX
- sizeY
- sizeZ
- tiltX
- tiltY
- tiltZ

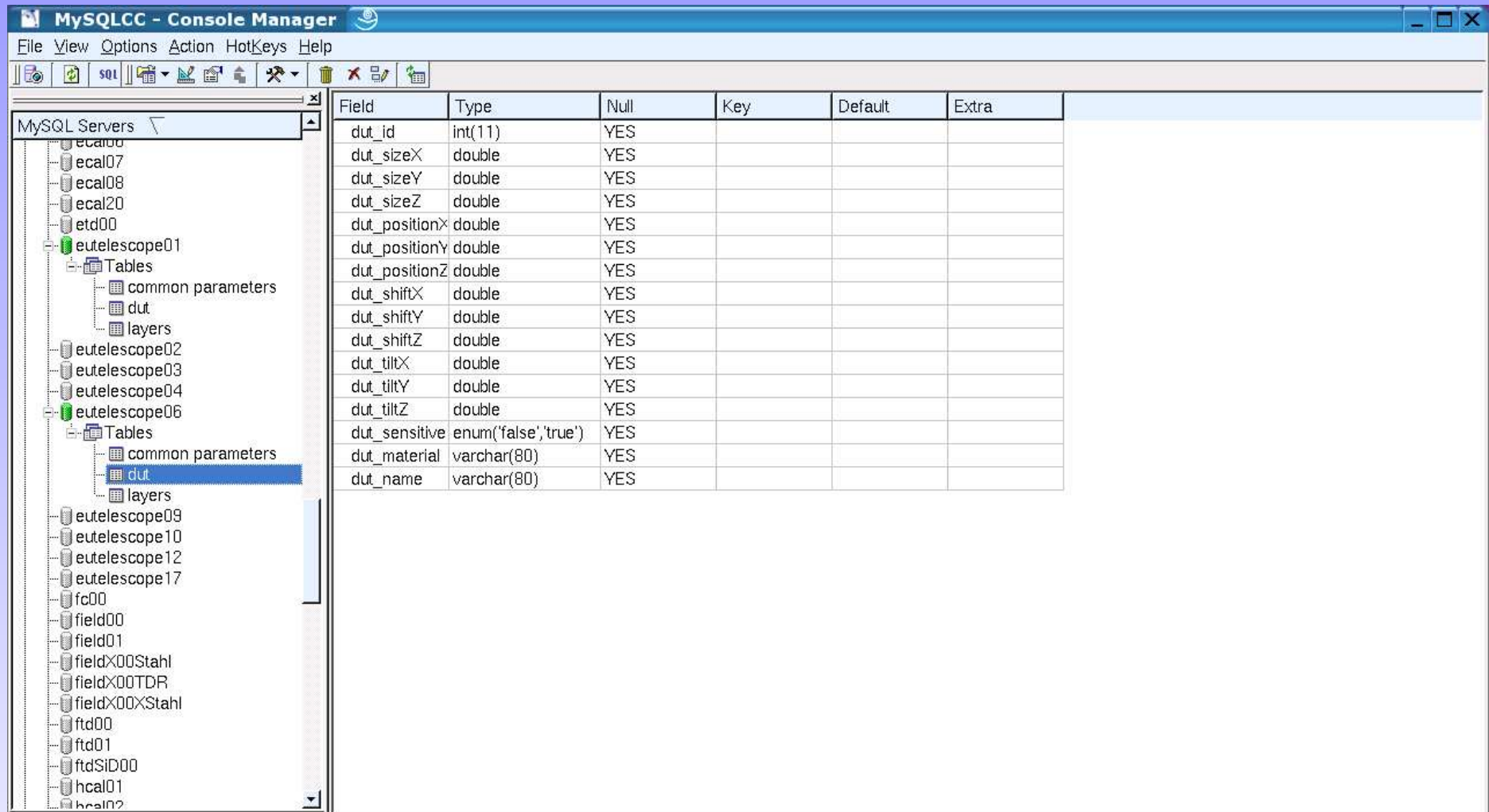
Result 1

Could not load Syntax File: '/afs/desy.de/user/t/klmk/syntax.txt'

10 rows in set (0.03) sec

Messages History Explain

Detector geometry description in simulation

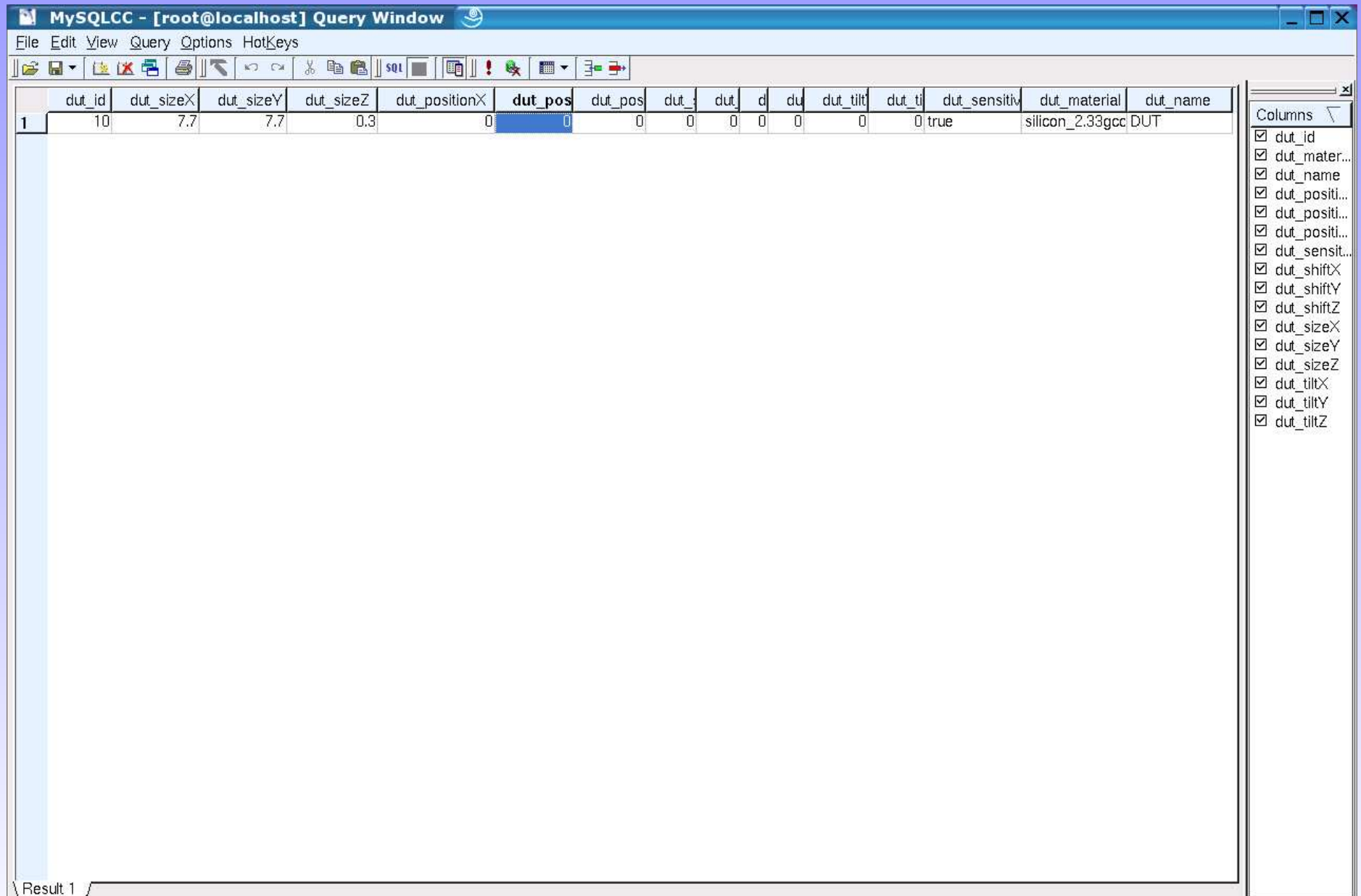


The screenshot shows the MySQLCC - Console Manager interface. On the left, a tree view displays a hierarchy of MySQL servers and databases. The 'eutelescope06' database is expanded, showing a 'Tables' folder containing 'common parameters', 'dut', and 'layers'. The 'dut' table is selected, and its field definitions are displayed in a table on the right.

Field	Type	Null	Key	Default	Extra
dut_id	int(11)	YES			
dut_sizeX	double	YES			
dut_sizeY	double	YES			
dut_sizeZ	double	YES			
dut_positionX	double	YES			
dut_positionY	double	YES			
dut_positionZ	double	YES			
dut_shiftX	double	YES			
dut_shiftY	double	YES			
dut_shiftZ	double	YES			
dut_tiltX	double	YES			
dut_tiltY	double	YES			
dut_tiltZ	double	YES			
dut_sensitive	enum('false','true')	YES			
dut_material	varchar(80)	YES			
dut_name	varchar(80)	YES			

```
[Tatsiana] Querying MySQL Server for Database Information.  
[Tatsiana] Querying MySQL Server for Table information in database: models03  
[Tatsiana] Querying MySQL Server for Table information in database: mysql  
[consult@local] Querying MySQL Server for Database Information.  
[root@localhost] Querying MySQL Server for Database Information.  
[root@localhost] Querying MySQL Server for Table information in database: eutelescope01  
[root@localhost] Querying MySQL Server for Table information in database: eutelescope06
```


Detector geometry description in simulation



The image shows a MySQLCC Query Window displaying a table with detector geometry parameters. The table has 17 columns and 1 row of data. The columns are: dut_id, dut_sizeX, dut_sizeY, dut_sizeZ, dut_positionX, dut_pos, dut_pos, dut, dut, d, du, dut_tilt, dut_til, dut_sensitiv, dut_material, and dut_name. The values for the first row are: 1, 10, 7.7, 7.7, 0.3, 0, 0, 0, 0, 0, 0, 0, 0, true, silicon_2.33gcc, and DUT. The 'dut_pos' column is highlighted in blue. A 'Columns' panel on the right lists all columns with checkboxes, all of which are checked.

	dut_id	dut_sizeX	dut_sizeY	dut_sizeZ	dut_positionX	dut_pos	dut_pos	dut	dut	d	du	dut_tilt	dut_til	dut_sensitiv	dut_material	dut_name
1	10	7.7	7.7	0.3	0	0	0	0	0	0	0	0	0	true	silicon_2.33gcc	DUT

Result 1

Could not load Syntax File: '/afs/desy.de/user/t/tklimk/syntax.txt'
1 row in set (0.04) sec

Detector geometry description in simulation

- ✓ To fill the database: use special script (dumpfile), also can do using graphical interface
- ✓ Use as few as possible hard-coded numbers
- ✓ The data from database are read out by Mokka telescope driver code, e.g. for DUT:

```
db->exec("SELECT * FROM `dut`");  
db->getTuple();  
const G4int    dut_ID      = db->fetchInt("dut_id");  
const G4double dut_sizeX   = db->fetchDouble("dut_sizeX") * mm;
```