PITZ graphical user interface and jddd

More than fifteen years worked for the PITZ gui - what remains?

Bert Schöneich / Winfried Köhler
(abgestimmt mit Elke Sombrowski, DESY HH)
PITZ-Betriebsseminar
Gohrisch, 19.-22. June 2017

structure

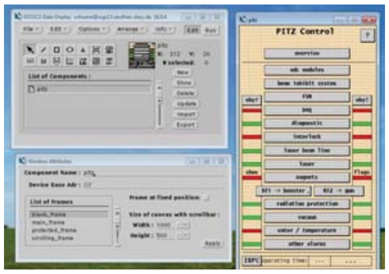
history - 1

graphical user interface for PITZ

– started fifteen years ago:
  first gui development tool:

  ddd

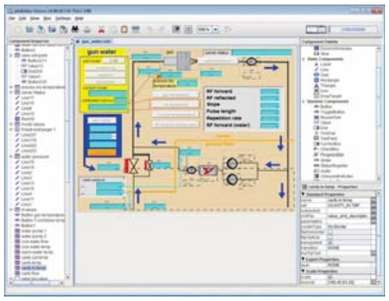
  - “DOOCS Data Display”
  - results stored in CAF-files



– since 8 years:
  second gui development tool:

  jddd

  - “Java DOOCS Data Display”
  - results stored in xml-files

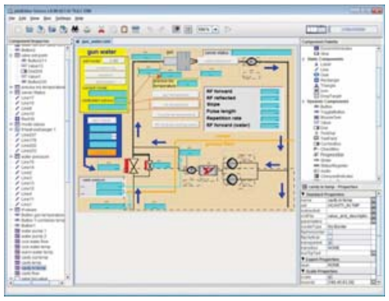


history - 2

from old to new

ddd (CAF)

- first step: partially automated, which was 85% successful
- with tools from DESY Hamburg and David Melkumyan
- afterwards all 400 files in three revision cycles worked by hand
- needs nearly 2 years



jddd (xml)

jddd - 1

status

(May 2017)

- PITZ gui:
  - 824 xml-files
  - 145 folders
  - 18,9 MB

- PITZ gui svn:
  - 4.440 files
  - 412 folders
  - 104 MB

- old CAF files:
  - 372 CAF-files

jddd - 2

highlights - 1

1. good combination of graphical and/or numerical gui programming (placing, moving or changing size of objects)

2. object oriented graphical programming
   - sub windows, callable from many parent windows
   - sub components, useable in many different windows with different DOOCS addresses

3. result of programming is a read- and writeable xml-file
   - can be changed by a normal text editor like emacs
   - search and replace in the code is possible
   - code can be changed with perl scripts

jddd - 3

highlights - 2

4. many components available
   (pane, static, dynamic, logic, plot)

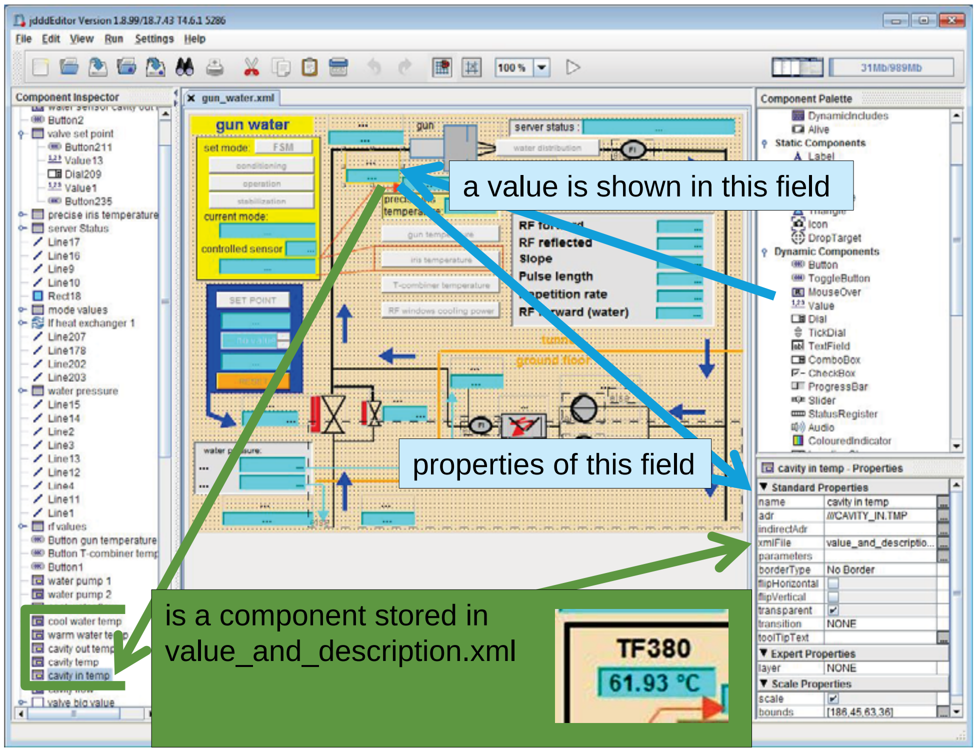
5. gui window components visible as graphic and as list simultaneously
   during programming

6. well defined and useful directory structure for the xml files

7. open from local computer or remote from browser

jddd - 4

highlights - 3



a value is shown in this field

properties of this field

is a component stored in value_and_description.xml

cavity in temp - Properties

▼ Standard Properties

name	cavity in temp
adr	///CAVITY_IN.TMP
indirectAdr	
xmlFile	value_and_descriptio...
parameters	
borderType	No Border
flipHorizontal	☐
flipVertical	☐
transparent	✓
transition	NONE
toolTipText	

▼ Expert Properties

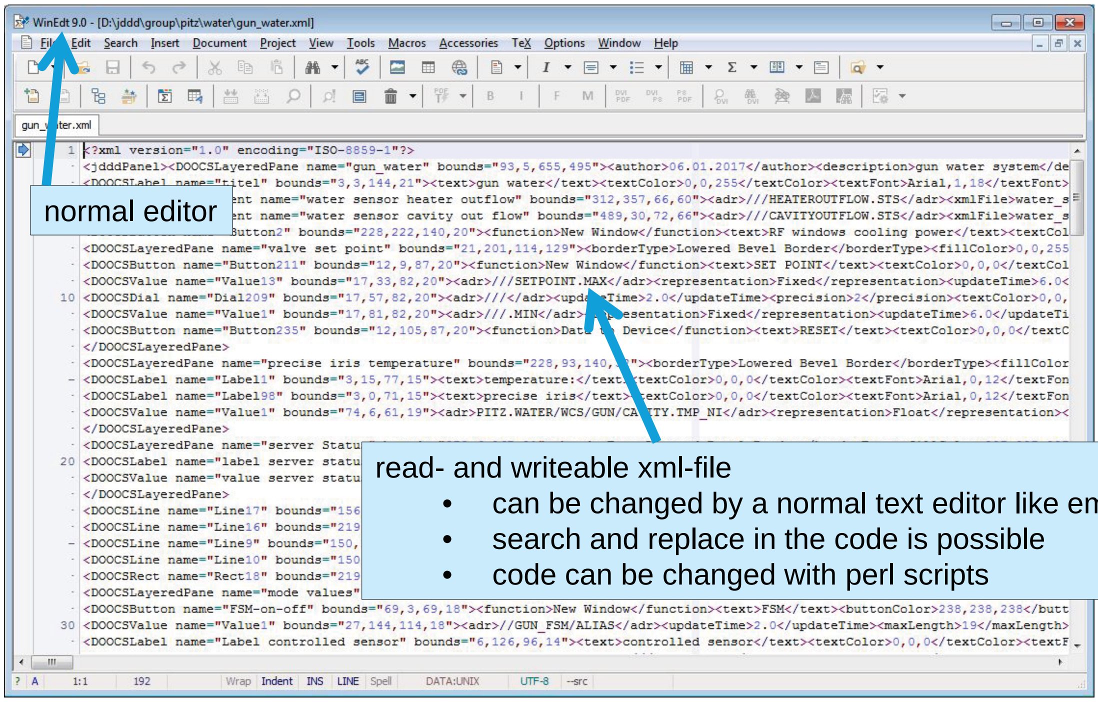
layer	NONE

▼ Scale Properties

scale	✓
bounds	[186,45,63,36]

jddd - 5

highlights - 4


WinEdt 9.0 - [D:\jddd\group\pitz\water\gun_water.xml]
File Edit Search Insert Document Project View Tools Macros Accessories TeX Options Window Help
gun_water.xml
? A 1:1 192 Wrap Indent INS LINE Spell DATA:UNIX UTF-8 --src

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<jdddPanel><DOOCSLayeredPane name="gun_water" bounds="93,5,655,495"><author>06.01.2017</author><description>gun water system</de
<DOOCSLabel name="titel" bounds="3,3,144,21"><text>gun water</text><textColor>0,0,255</textColor><textFont>Arial,1,18</textFont>
ent name="water sensor heater outflow" bounds="312,357,66,60"><adr>///HEATEROUTFLOW.STS</adr><xmlFile>water_s
ent name="water sensor cavity out flow" bounds="489,30,72,66"><adr>///CAVITYOUTFLOW.STS</adr><xmlFile>water_s
Button2" bounds="228,222,140,20"><function>New Window</function><text>RF windows cooling power</text><textCol
<DOOCSLayeredPane name="valve set point" bounds="21,201,114,129"><borderType>Lowered Bevel Border</borderType><fillColor>0,0,255
<DOOCSButton name="Button211" bounds="12,9,87,20"><function>New Window</function><text>SET POINT</text><textColor>0,0,0</textCol
<DOOCSValue name="Value13" bounds="17,33,82,20"><adr>///SETPOINT.MAX</adr><representation>Fixed</representation><updateTime>6.0<
<DOOCSDial name="Dial209" bounds="17,57,82,20"><adr>///</adr><upd  Time>2.0</updateTime><precision>2</precision><textColor>0,0,
<DOOCSValue name="Value1" bounds="17,81,82,20"><adr>///.MIN</adr>  esentation>Fixed</representation><updateTime>6.0</updateTi
<DOOCSButton name="Button235" bounds="12,105,87,20"><function>Dat   Device</function><text>RESET</text><textColor>0,0,0</textC
</DOOCSLayeredPane>
<DOOCSLayeredPane name="precise iris temperature" bounds="228,93,140, "><borderType>Lowered Bevel Border</borderType><fillColor
<DOOCSLabel name="Label1" bounds="3,15,77,15"><text>temperature:</text textColor>0,0,0</textColor><textFont>Arial,0,12</textFon
<DOOCSLabel name="Label98" bounds="3,0,71,15"><text>precise iris</text> extColor>0,0,0</textColor><textFont>Arial,0,12</textFon
<DOOCSValue name="Value1" bounds="74,6,61,19"><adr>PITZ.WATER/WCS/GUN/CA ITY.TMP_NI</adr><representation>Float</representation><
</DOOCSLayeredPane>
<DOOCSLayeredPane name="server Statu
<DOOCSLabel name="label server statu
<DOOCSValue name="value server statu
</DOOCSLayeredPane>
<DOOCSLine name="Line17" bounds="156
<DOOCSLine name="Line16" bounds="219
<DOOCSLine name="Line9" bounds="150,
<DOOCSLine name="Line10" bounds="150
<DOOCSRect name="Rect18" bounds="219
<DOOCSLayeredPane name="mode values"
<DOOCSButton name="FSM-on-off" bounds="69,3,69,18"><function>New Window</function><text>FSM</text><buttonColor>238,238,238</butt
<DOOCSValue name="Value1" bounds="27,144,114,18"><adr>//GUN_FSM/ALIAS</adr><updateTime>2.0</updateTime><maxLength>19</maxLength>
<DOOCSLabel name="Label controlled sensor" bounds="6,126,96,14"><text>controlled sensor</text><textColor>0,0,0</textColor><textF
```

normal editor

read- and writeable xml-file

- can be changed by a normal text editor like emacs
- search and replace in the code is possible
- code can be changed with perl scripts

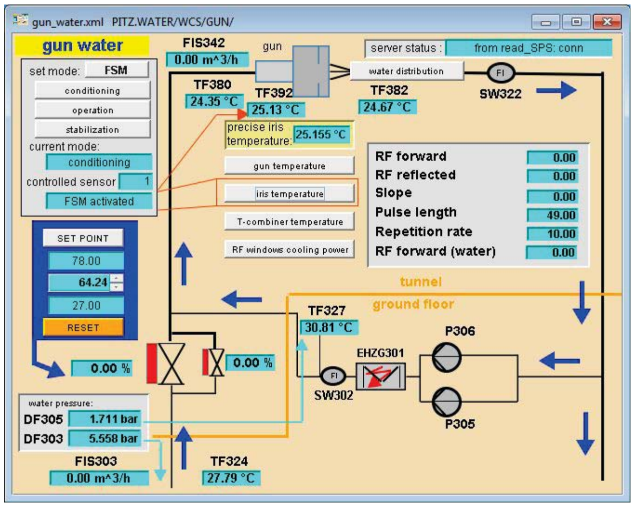
jddd - 6

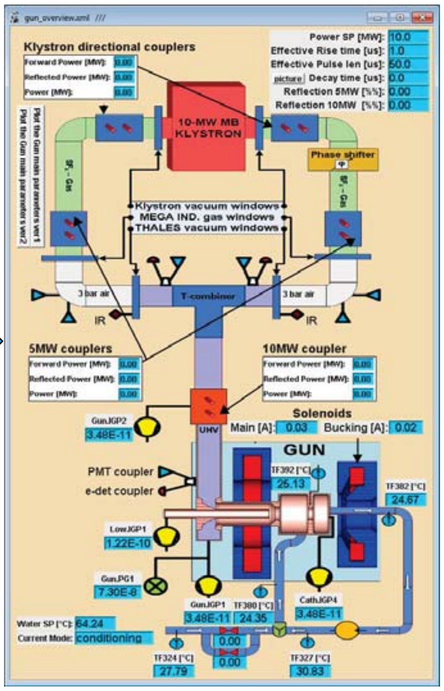
lowlights

1. During development, real (!) Data is being worked on.
2. stupid gui
   - no gui internal memory is available, even temporarily
3. slow speed during opening guis
4. No real documentation of jddd is available, but training.
5. Sometimes it is difficult to catch with the mouse pointer an graphical object.
6. Be carefully by combining components in “Groups”.
   A group concealed invisibly behind her lying components.
   For example, a visible (!) button is no longer operable if it is behind a group.

type of programming - 1

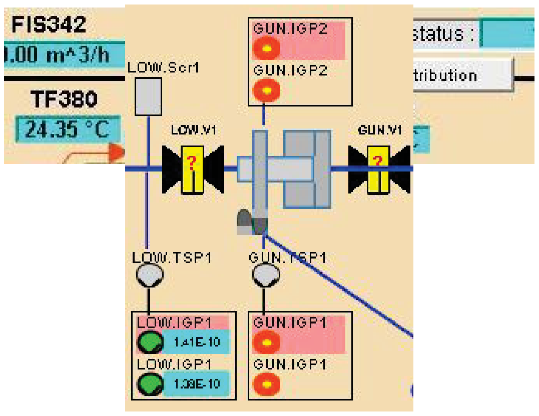
symbolic versus reality

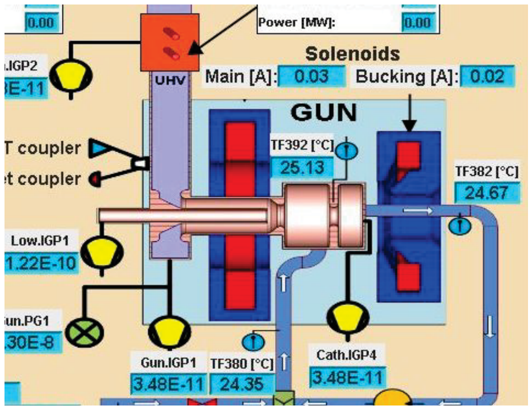




type of programming - 2

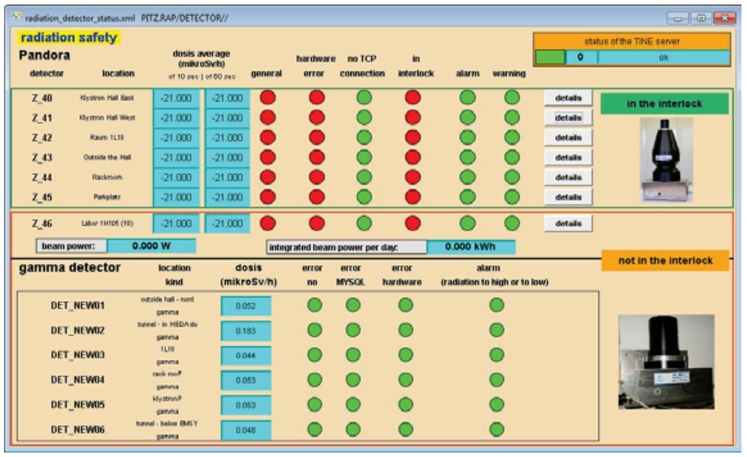
symbolic versus reality





type of programming - 3

- reality if really necessary
  (just as real as necessary)
- reality if nice without disturbing



radiation_detector_status.xml PITZ.RAP/DETECTOR//

radiation safety

status of the TINE server: 0 ok

Pandora

detector	location	dosis average (mikroSv/h) of 10 sec	of 60 sec	general	hardware error	no TCP connection	in interlock	alarm	warning	
Z_40	Klystron Hall East	-21.000	-21.000	🔴	🔴	🟢	🔴	🟢	🟢	details
Z_41	Klystron Hall West	-21.000	-21.000	🔴	🔴	🟢	🔴	🟢	🟢	details
Z_42	Raum 1L18	-21.000	-21.000	🔴	🔴	🟢	🔴	🟢	🟢	details
Z_43	Outside the Hall	-21.000	-21.000	🔴	🔴	🟢	🔴	🟢	🟢	details
Z_44	Rackroom	-21.000	-21.000	🔴	🔴	🟢	🔴	🟢	🟢	details
Z_45	Parkplatz	-21.000	-21.000	🔴	🔴	🟢	🔴	🟢	🟢	details
Z_46	Labor 1H105 (18)	-21.000	-21.000	🔴	🔴	🟢	🔴	🟢	🟢	details

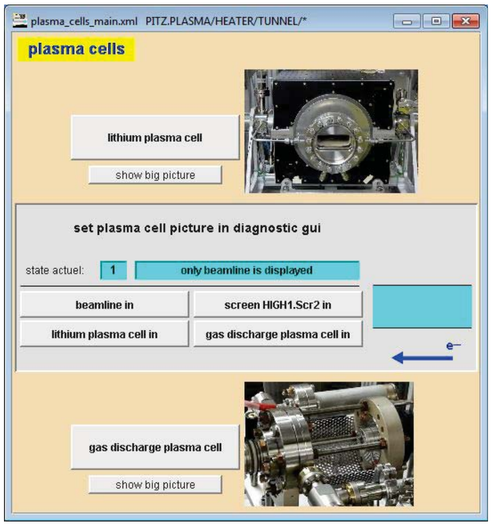
in the interlock

beam power: 0.000 W

integrated beam power per day: 0.000 kWh

not in the interlock

gamma detector	location kind	dosis (mikroSv/h)	error no	error MYSQL	error hardware	alarm (radiation to high or to low)
DET_NEW01	outside hall - nord gamma	0.052	🟢	🟢	🟢	🟢
DET_NEW02	tunnel - in HEDA du gamma	0.183	🟢	🟢	🟢	🟢
DET_NEW03	1L10 gamma	0.044	🟢	🟢	🟢	🟢
DET_NEW04	rack roof gamma	0.053	🟢	🟢	🟢	🟢
DET_NEW05	klystron/F gamma	0.053	🟢	🟢	🟢	🟢
DET_NEW06	tunnel - below EMSY gamma	0.048	🟢	🟢	🟢	🟢


plasma_cells_main.xml PITZ.PLASMA/HEATER/TUNNEL/*
plasma cells
lithium plasma cell
show big picture
set plasma cell picture in diagnostic gui
state actuel: 1
only beamline is displayed
beamline in
screen HIGH1.Scr2 in
lithium plasma cell in
gas discharge plasma cell in
e⁻
gas discharge plasma cell
show big picture

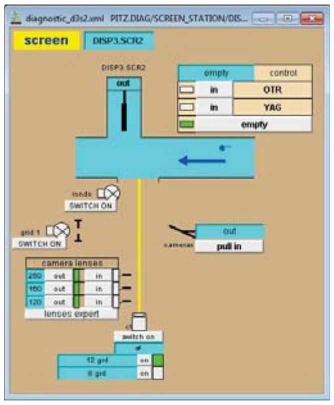
type of programming - 4

Standardization (with Bagrat and Jörg)

nice for the operators:
- easier to recognition for the operator
- less errors during handling

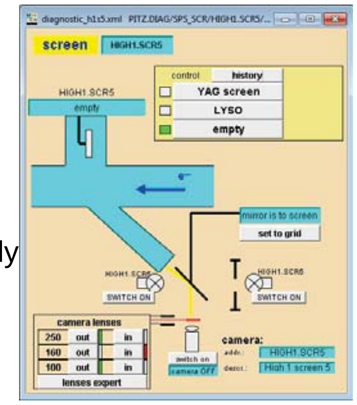
nice for the developers:
- faster to create
- less errors during creation

but:
- standardization starts at hardware (!)
- All intermediate steps from the hardware to the gui must use standardization.
- Just show the necessary things for the operator.
- simplify will be successfully



standardized

differently



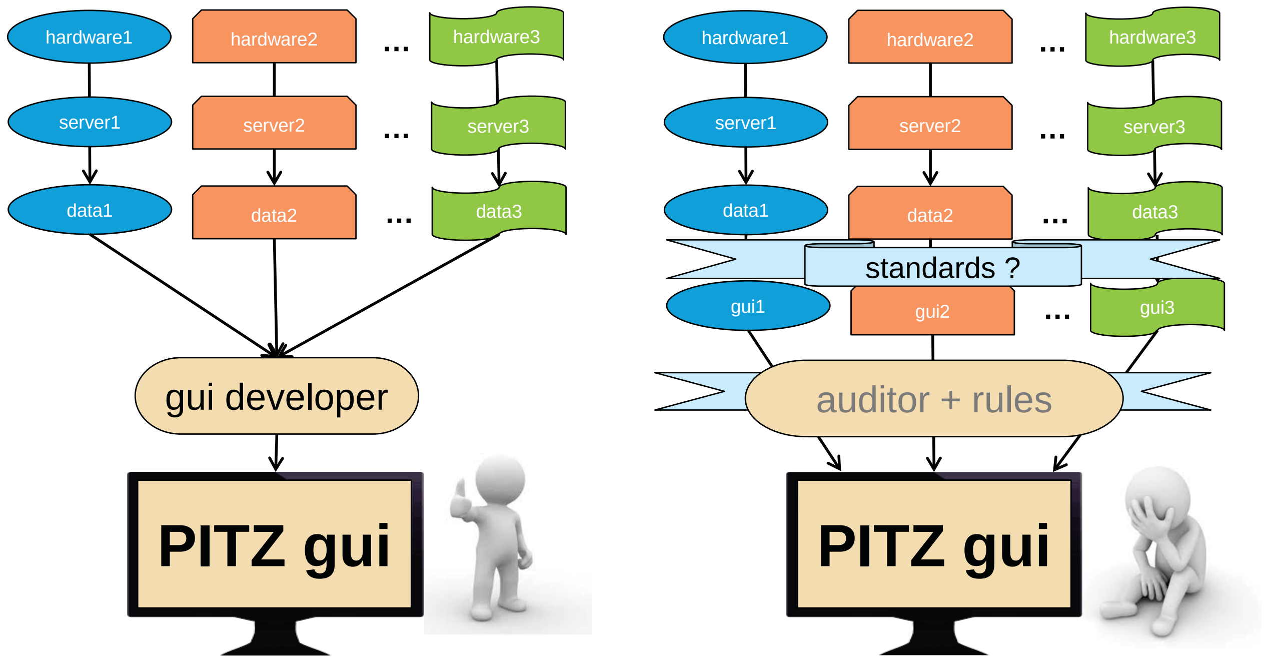
Who programmed the gui? - 1

generalist versus specialist: “move the YAG-screen in the beam”

specialists for	creates *his* button in *his* gui
mechanic device	actuator 20 mm forward
electronic hardware	2 mA for 3 sec on pin 5
SPS	gate 4 on high
control software	increment bit 5 of variable 2
gui software	set DOOCS variable 3 on high
operator wants and needs	a completely transparent command
operator have to have	button “move YAG in”

Who programmed the gui? - 2

one developer or more developers

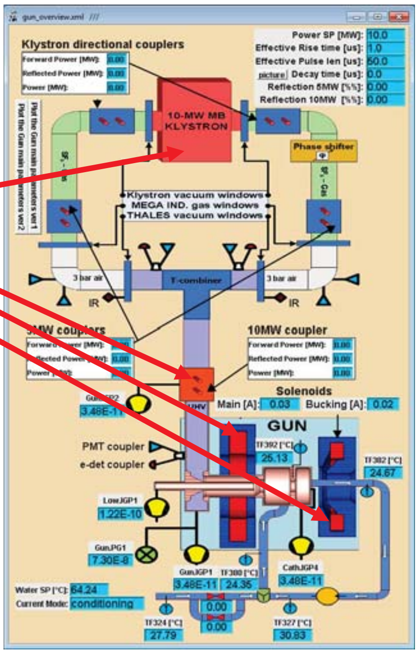


Who programmed the gui? - 3

for example a first rule:

Nothing is **red**, except for a mistake or a danger.

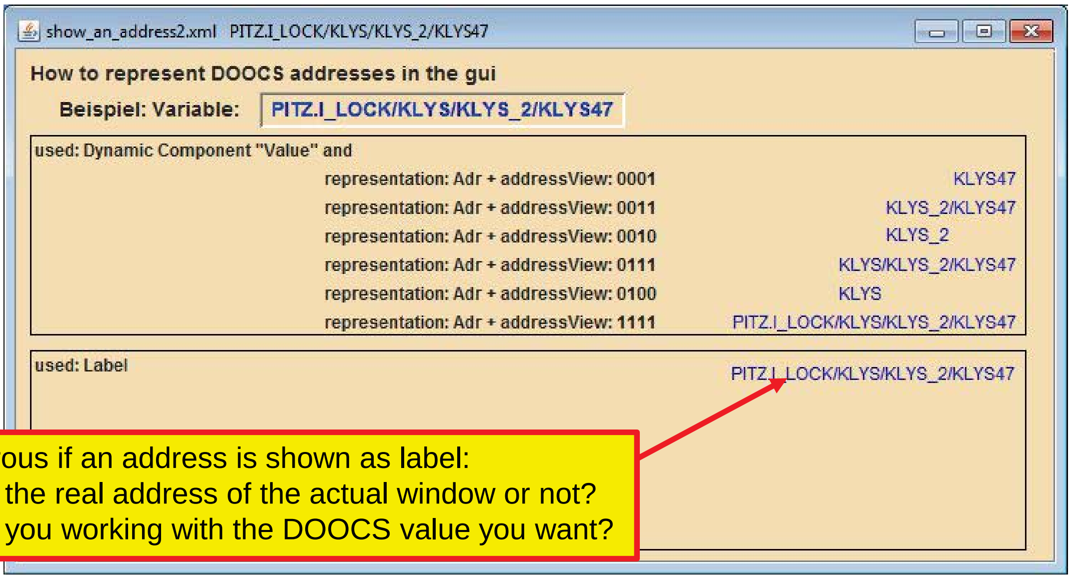
but



Who programmed the gui? - 4

for example a next rule:

If an address is to be displayed, then as a readout value not as a label.



show_an_address2.xml PITZ.I_LOCK/KLYS/KLYS_2/KLYS47

How to represent DOOCS addresses in the gui

Beispiel: Variable: PITZ.I_LOCK/KLYS/KLYS_2/KLYS47

used: Dynamic Component "Value" and

representation: Adr + addressView: 0001	KLYS47
representation: Adr + addressView: 0011	KLYS_2/KLYS47
representation: Adr + addressView: 0010	KLYS_2
representation: Adr + addressView: 0111	KLYS/KLYS_2/KLYS47
representation: Adr + addressView: 0100	KLYS
representation: Adr + addressView: 1111	PITZ.I_LOCK/KLYS/KLYS_2/KLYS47

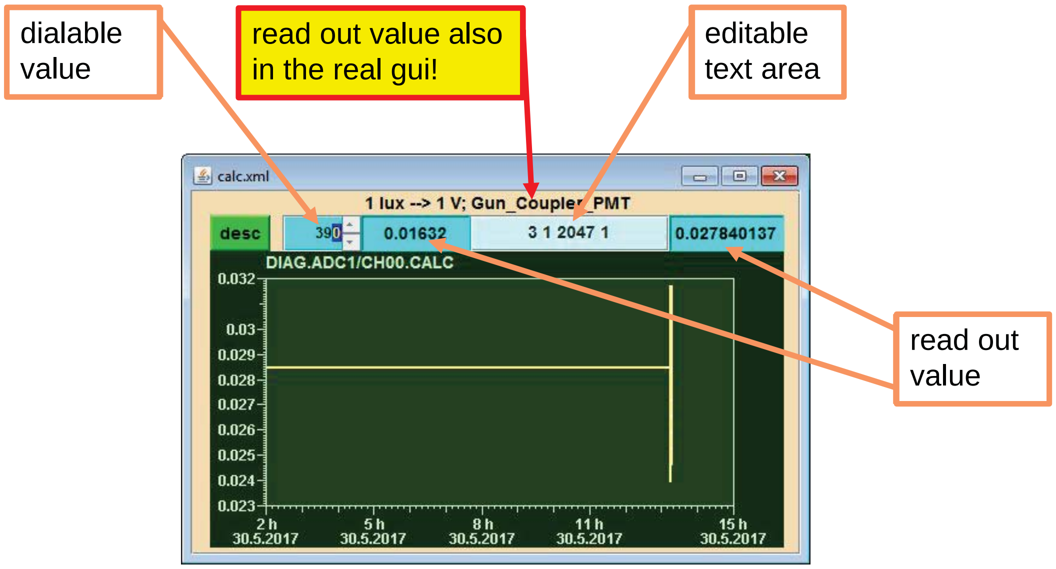
used: Label PITZ.I_LOCK/KLYS/KLYS_2/KLYS47

dangerous if an address is shown as label:
- Is it the real address of the actual window or not?
- Are you working with the DOOCS value you want?

Who programmed the gui? - 5

for example a last rule:

The type of a displayed value must be clearly recognizable graphically.
This must be standard for the whole gui.



open things

open

- standardization of, for example, the diagnostic screens
(Many of them are, but not all.)

- documentation (necessary?)

jddd - Wiki at DESY Zeuthen - 1

the wiki

1. https://wiki-zeuthen.desy.de/JDDD

2. created by Winfried Köhler

3. filled by Winfried Köhler and Bert Schöneich

4. content
   - general notes on the use of JDDD
   - tips und tricks
   - known JDDD bugs and workarounds

5. to give to the developer
   - practical tips
   - experience
   - „standards“ for Zeuthen
   - technics
   - bugs, errors, workarounds

jddd - Wiki at DESY Zeuthen - 2

other helpful websites

1. https://jddd.desy.de/
   - introduction
   - special features
   - screenshots
   - demo panels
   - help
   - faq

2. http://tesla.desy.de/doocs/doocs.html

3. https://www-zeuthen.desy.de/pitz/apps/
   - PITZ java applications
   - start applications

4. http://pitz.desy.de/

end

1st March 2018

Who will do this job now?

(The answer is not 42.)