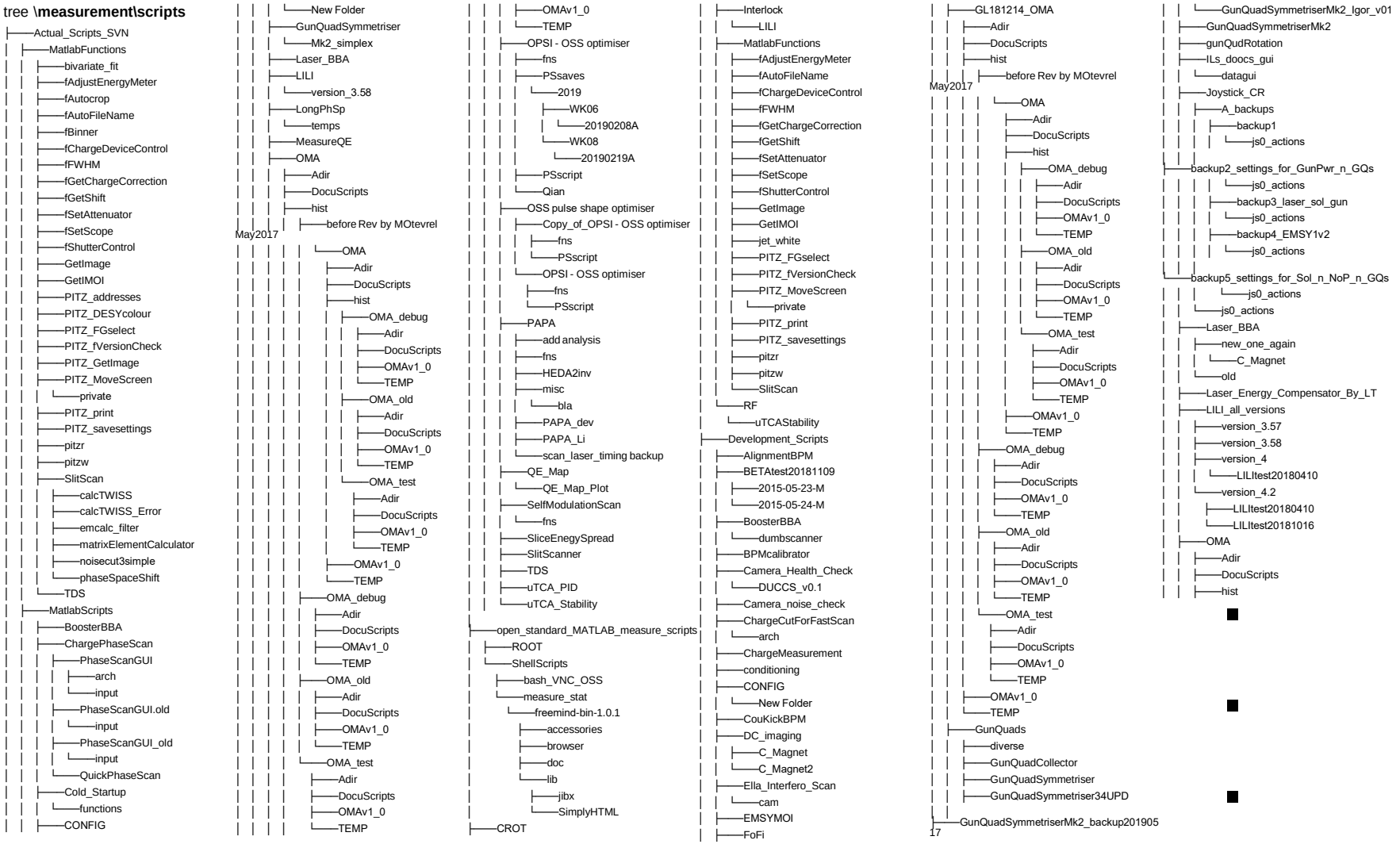
Software Management at PITZ

Introducing new git structure and declaring some good intents (again)

HELMHOLTZ ASSOCIATION

DESY.

Motivation:


tree \measurement\scripts
Actual_Scripts_SVN
MatlabFunctions
bivariate_fit
fAdjustEnergyMeter
fAutocrop
fAutoFileName
fBinner
fChargeDeviceControl
fFWHM
fGetChargeCorrection
fGetShift
fSetAttenuator
fSetScope
fShutterControl
GetImage
GetIMOI
PITZ_addresses
PITZ_DESYcolour
PITZ_FGselect
PITZ_fVersionCheck
PITZ_GetImage
PITZ_MoveScreen
private
PITZ_print
PITZ_savesettings
pitzr
pitzw
SlitScan
calcTWISS
calcTWISS_Error
emcalc_filter
matrixElementCalculator
noisecut3simple
phaseSpaceShift
TDS
MatlabScripts
BoosterBBA
ChargePhaseScan
PhaseScanGUI
arch
input
PhaseScanGUI.old
input
PhaseScanGUI_old
input
QuickPhaseScan
Cold_Startup
functions
CONFIG
New Folder
GunQuadSymmetriser
Mk2_simplex
Laser_BBA
LILI
version_3.58
LongPhSp
temps
MeasureQE
OMA
Adir
DocuScripts
hist
before Rev by MOtevrel
May2017
OMA
Adir
DocuScripts
hist
OMA_debug
Adir
DocuScripts
OMAv1_0
TEMP
OMA_old
Adir
DocuScripts
OMAv1_0
TEMP
OMA_test
Adir
DocuScripts
OMAv1_0
TEMP
OMAv1_0
TEMP
OMA_debug
Adir
DocuScripts
OMAv1_0
TEMP
OMA_old
Adir
DocuScripts
OMAv1_0
TEMP
OMA_test
Adir
DocuScripts
OMAv1_0
TEMP
OMAv1_0
TEMP
OPSI - OSS optimiser
fns
PSsaves
2019
WK06
20190208A
WK08
20190219A
PSscript
Qian
OSS pulse shape optimiser
Copy_of_OPSI - OSS optimiser
fns
PSscript
OPSI - OSS optimiser
fns
PSscript
PAPA
add analysis
fns
HEDA2inv
misc
bla
PAPA_dev
PAPA_Li
scan_laser_timing backup
QE_Map
QE_Map_Plot
SelfModulationScan
fns
SliceEnegySpread
SlitScanner
TDS
uTCA_PID
uTCA_Stability
open_standard_MATLAB_measure_scripts
ROOT
ShellScripts
bash_VNC_OSS
measure_stat
freemind-bin-1.0.1
accessories
browser
doc
lib
jibx
SimplyHTML
CROT
Interlock
LILI
MatlabFunctions
fAdjustEnergyMeter
fAutoFileName
fChargeDeviceControl
fFWHM
fGetChargeCorrection
fGetShift
fSetAttenuator
fSetScope
fShutterControl
GetImage
GetIMOI
jet_white
PITZ_FGselect
PITZ_fVersionCheck
PITZ_MoveScreen
private
PITZ_print
PITZ_savesettings
pitzr
pitzw
SlitScan
RF
uTCAStability
Development_Scripts
AlignmentBPM
BETAtest20181109
2015-05-23-M
2015-05-24-M
BoosterBBA
dumbscanner
BPMcalibrator
Camera_Health_Check
DUCCS_v0.1
Camera_noise_check
ChargeCutForFastScan
arch
ChargeMeasurement
conditioning
CONFIG
New Folder
CouKickBPM
DC_imaging
C_Magnet
C_Magnet2
Ella_Interfero_Scan
cam
EMSYMOI
FoFi
GL181214_OMA
Adir
DocuScripts
hist
before Rev by MOtevrel
May2017
OMA
Adir
DocuScripts
hist
OMA_debug
Adir
DocuScripts
OMAv1_0
TEMP
OMA_old
Adir
DocuScripts
OMAv1_0
TEMP
OMA_test
Adir
DocuScripts
OMAv1_0
TEMP
OMAv1_0
TEMP
OMA_debug
Adir
DocuScripts
OMAv1_0
TEMP
OMA_old
Adir
DocuScripts
OMAv1_0
TEMP
OMA_test
Adir
DocuScripts
OMAv1_0
TEMP
OMAv1_0
TEMP
GunQuads
diverse
GunQuadCollector
GunQuadSymmetriser
GunQuadSymmetriser34UPD
GunQuadSymmetrisMk2_backup201905
17
GunQuadSymmetrisMk2_Igor_v01
GunQuadSymmetrisMk2
gunQudRotation
ILs_doocs_gui
datagui
Joystick_CR
A_backups
backup1
js0_actions
backup2_settings_for_GunPwr_n_GQs
js0_actions
backup3_laser_sol_gun
js0_actions
backup4_EMSY1v2
js0_actions
backup5_settings_for_Sol_n_NoP_n_GQs
js0_actions
js0_actions
Laser_BBA
new_one_again
C_Magnet
old
Laser_Energy_Compensator_By_LT
LILI_all_versions
version_3.57
version_3.58
version_4
LILItest20180410
version_4.2
LILItest20180410
LILItest20181016
OMA
Adir
DocuScripts
hist

The Git (stash) structure

Stash root structure

PITZ

- Hardware
- Libraries
- Operation Tools
- Simulation
- Tools

- Operation Tools: scripts for shifts that call any number of dependencies

---

- Hardware: device specific objects, functions, & tools.
- Libraries: general hardware independent dependencies
- Simulation: general tools for running and processing simulations
- Tools: machine related, non-shift scripts that call any number of dependencies for <> or data analysis

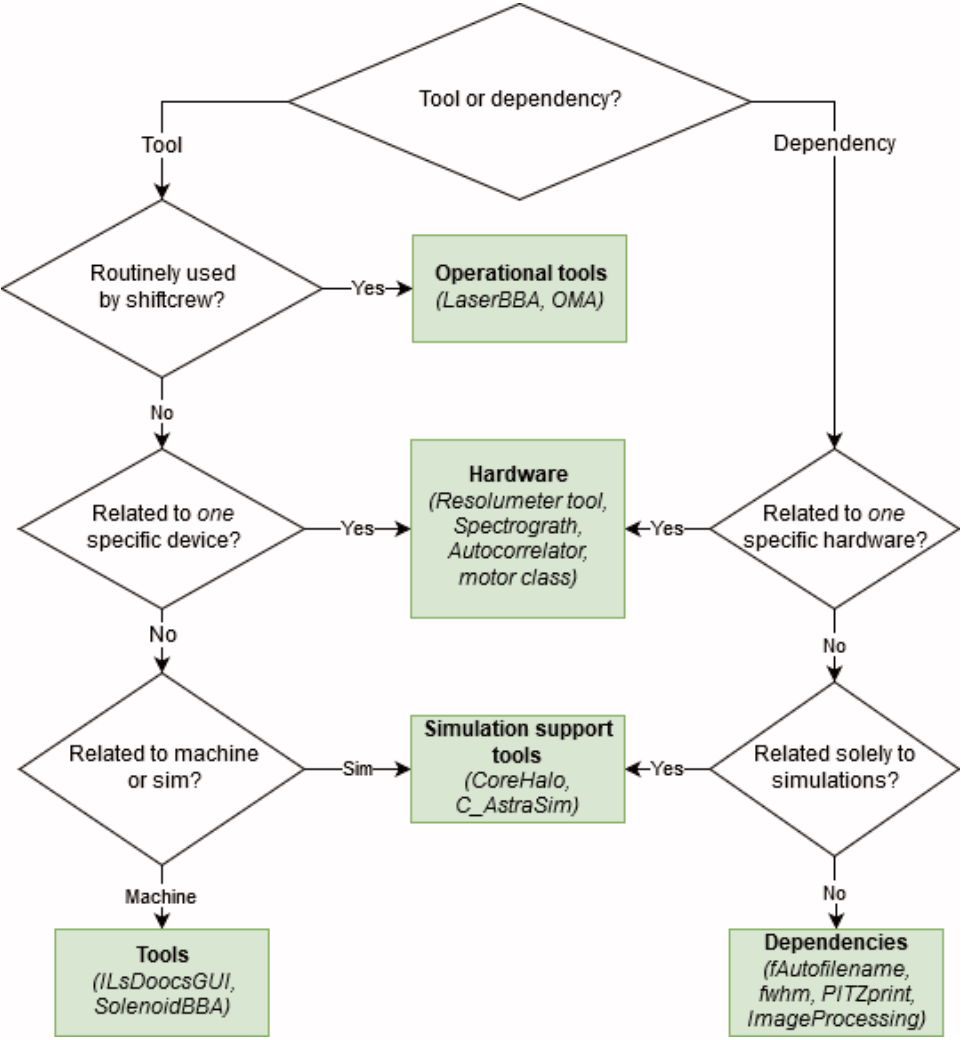
Sample

```
PITZ
├──Hardware
│   ├──BPMs
│   ├──Camera
│   │   ├──DUCCS
│   │   ├──ResoMeter
│   │   ├──_Class
│   │   └──_Function
│   ├──Charge
│   ├──Laser
│   │   └──_Class
│   ├──RF
│   │   └──uTCA stability
│   ├──Scope
│   ├──Shutter
│   │   └──function
│   ├──Spectrograph
│   └──TLmotors
├──Libraries
│   ├──_class
│   │   ├──GeneralVolumeAnalyzer
│   │   └──ImageProcessing
│   │          ImageProccessing.m
│   └──_function
│       ├──bivariate_fit
│       ├──fFWHM
│       ├──fGetShift
│       └──PITZ_addresses
├──Operation Tools
│   ├──DC measurement
│   ├──LaserBBA
│   ├──LTscan
│   ├──Mirror56
│   ├──OMA
│   ├──Phasescan
│   └──QEmap
├──Simulation
│   ├──ASTRA
│   │   └──ASTRA Analyzer
│   ├──Krack3
│   └──NLO
│       └──Chi3D
└──Tools
    ├──ILs_doocs_gui
    ├──LaserCharacterizer
    ├──SolenoidBBA
    └──Spectrograph Analyzer
```

stash.desy.de/projects/PITZ/repos/pitz/

The Git (stash) structure

Stash root structure



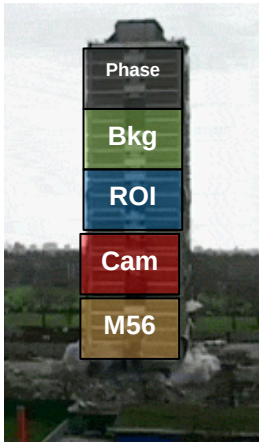
Sample

```
PITZ
├──Hardware
│   ├──BPMs
│   ├──Camera
│   │   ├──DUCCS
│   │   ├──ResoMeter
│   │   ├──_Class
│   │   └──_Function
│   ├──Charge
│   ├──Laser
│   │   └──_Class
│   ├──RF
│   │   └──uTCA stability
│   ├──Scope
│   ├──Shutter
│   │   └──function
│   ├──Spectrograph
│   └──TLmotors
├──Libraries
│   ├──_class
│   │   ├──GeneralVolumeAnalyzer
│   │   └──ImageProcessing
│   │          ImageProccessing.m
│   └──_function
│       ├──bivariate_fit
│       ├──fFWHM
│       ├──fGetShift
│       └──PITZ_addresses
├──Operation Tools
│   ├──DC measurement
│   ├──LaserBBA
│   ├──LTscan
│   ├──Mirror56
│   ├──OMA
│   ├──Phasescan
│   └──QEmap
├──Simulation
│   ├──ASTRA
│   │   └──ASTRA Analyzer
│   ├──Krack3
│   └──NLO
│       └──Chi3D
└──Tools
    ├──ILs_doocs_gui
    ├──LaserCharacterizer
    ├──SolenoidBBA
    └──Spectrograph Analyzer
```

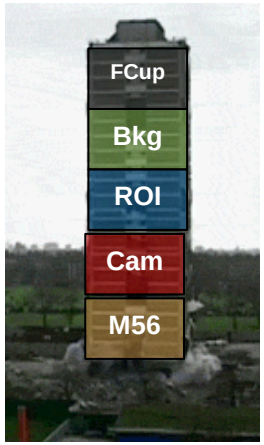
stash.desy.de/projects/PITZ/repos/pitz/

A programming Framework for PITZ

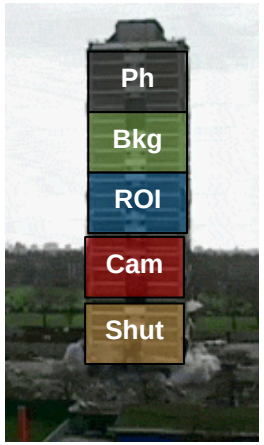
LaserBBA



QEmap



TDS

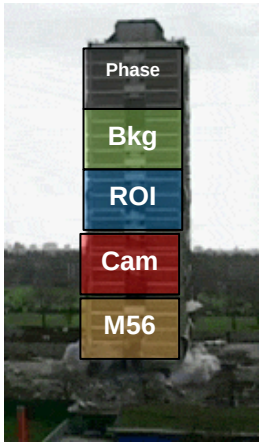


OMA

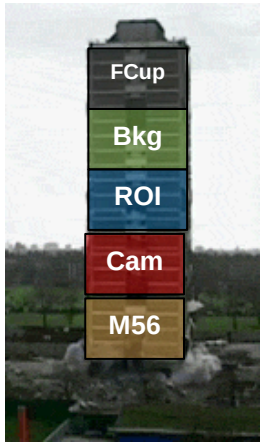
- Problems should be solved once and not again for each script (→ Background)
- Programm blocks should be interchangable
- Blocks should have clear capsulation

Encapsulation of program Units

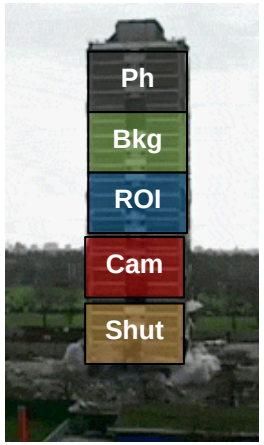
LaserBBA



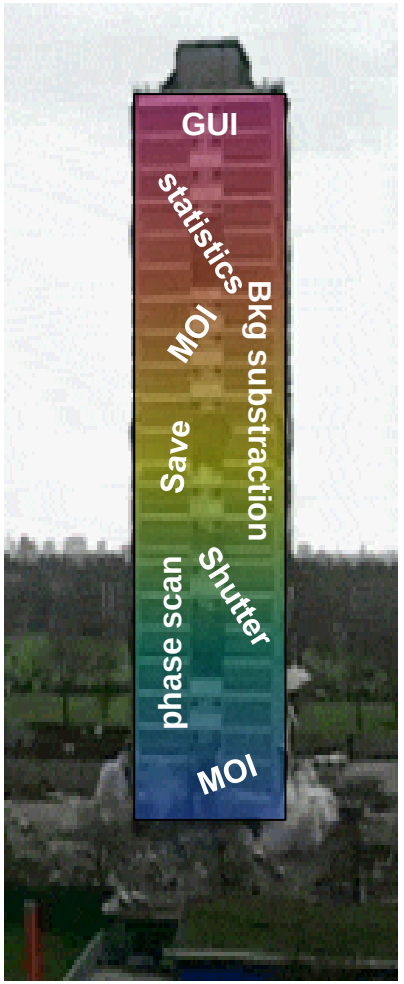
QEmap



TDS



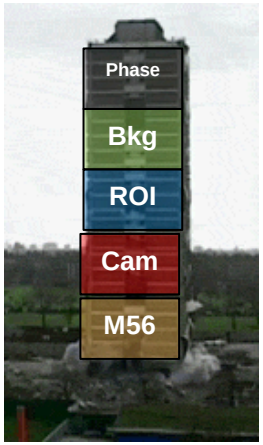
OMA



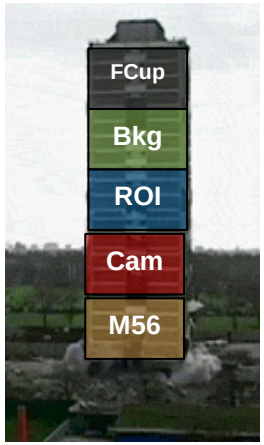
- Problems should be solved once and not again for each script (→ Background)
- Programm blocks should be interchangable
- Blocks should have clear capsulation !!!!!

Small Changes often result in Rewrite

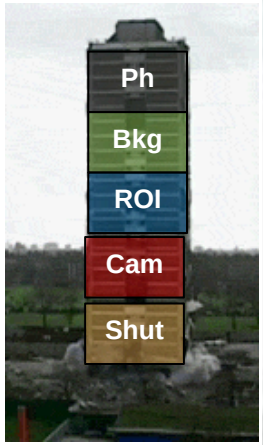
LaserBBA



QEmap



TDS

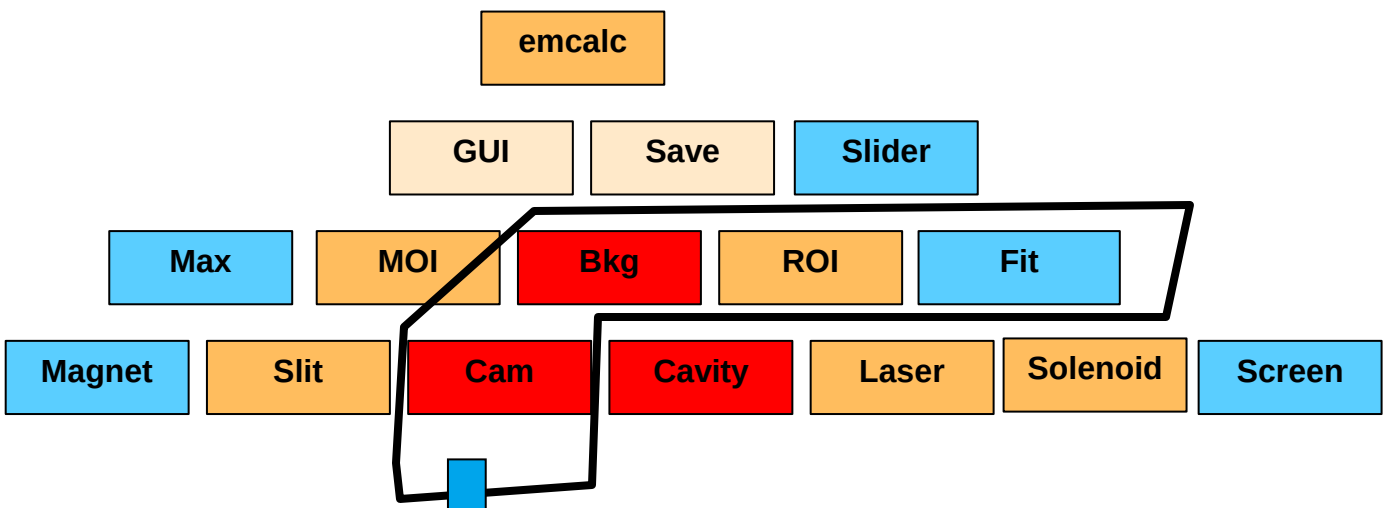


OMA

- Problems should be solved once and not again for each script (→ Background)
- Programm blocks should be interchangable
- Blocks should have clear capsulation !!!

Call for complete rewrite, because the code is so „horrible“ and won‘t be touched (OMA, FastScan etc)

Block Responsibility



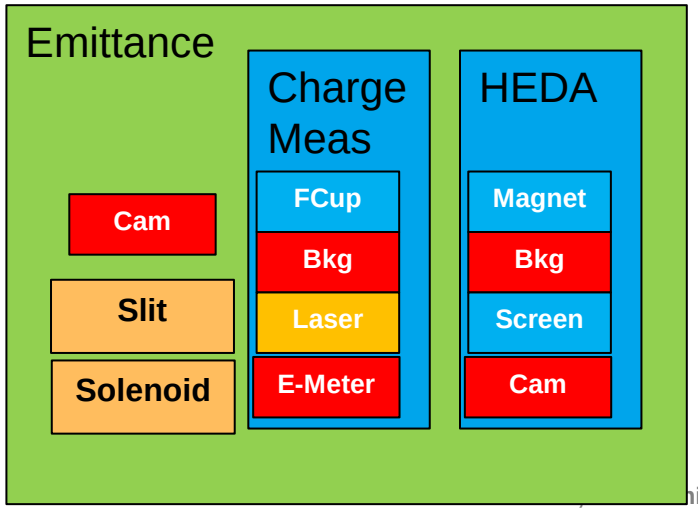
Students Beginners

Advanced Developers

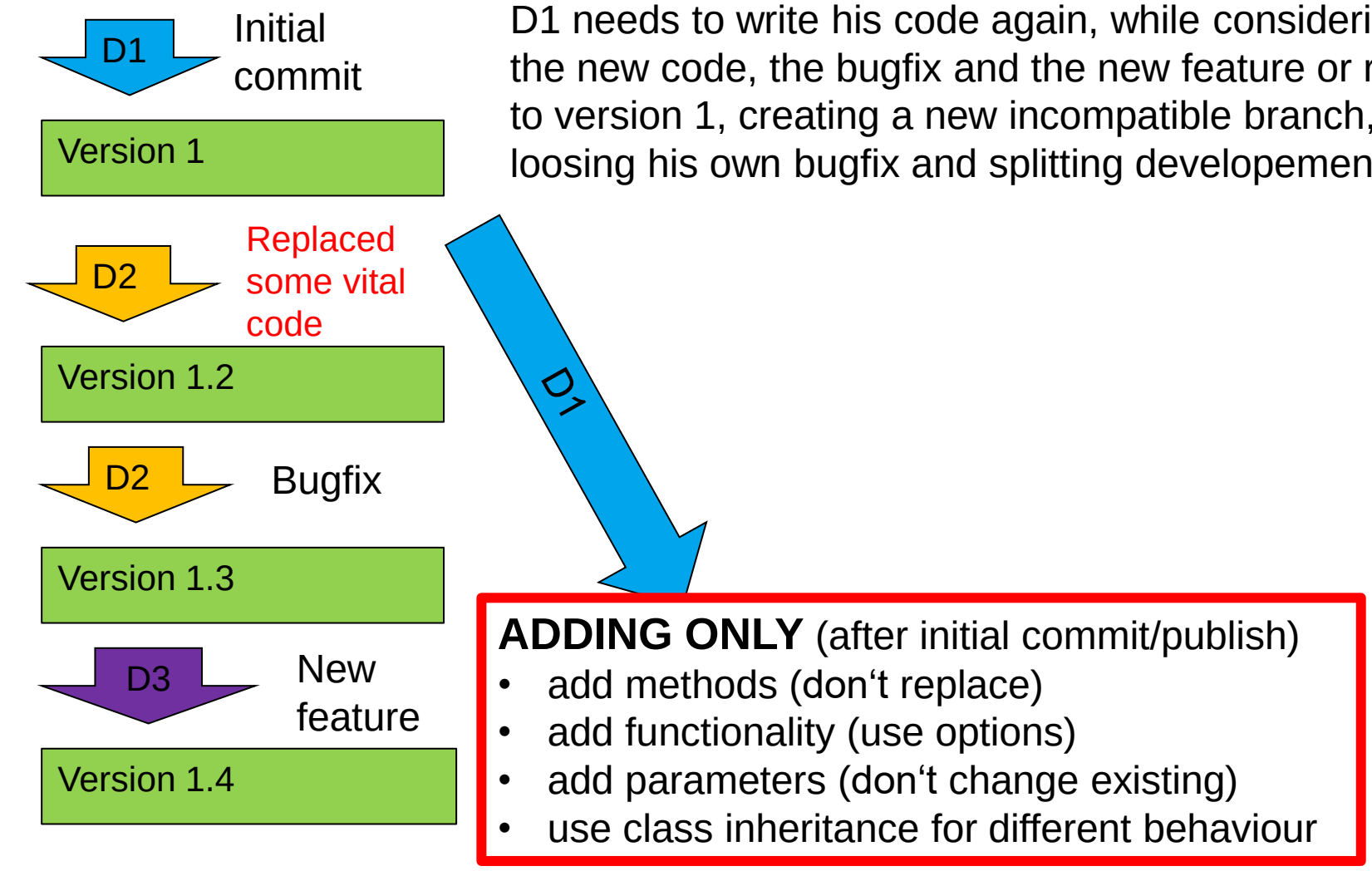
Experts Only

Build blocks out of blocks

out of blocks



Cooperative Developement → Adding ONLY



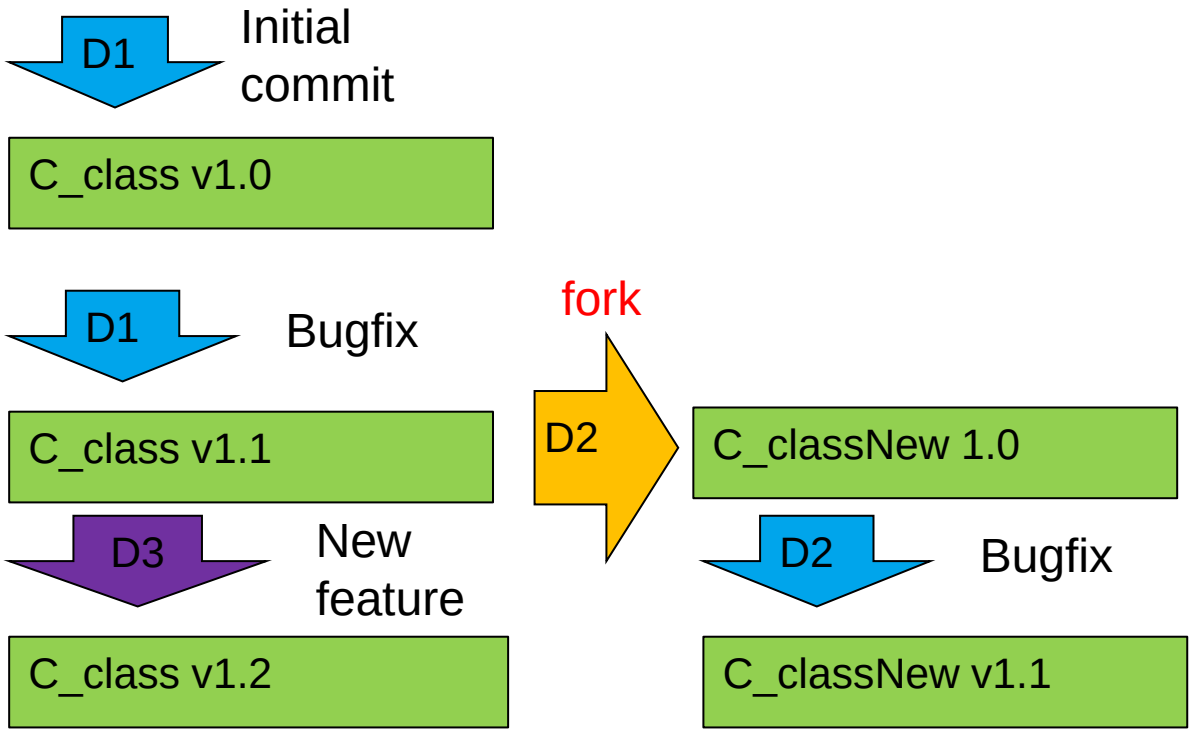
D1 needs to write his code again, while considering the new code, the bugfix and the new feature or revert to version 1, creating a new incompatible branch, loosing his own bugfix and splitting developement.

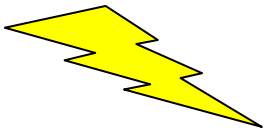
ADDING ONLY (after initial commit/publish)

- add methods (don‘t replace)
- add functionality (use options)
- add parameters (don‘t change existing)
- use class inheritance for different behaviour

Conflict!
Developer 1 discovers his code has been replaced
and does not the same as before

Cooperative Developement → Alternatively: fork





Workflow unbroken and development proceeds in parallel

Consistent Effort to prevent decay

Cooperate
- GIT
- Dev Meeting
- Bug Report /Feedback

Teach
- Student Training
- Documentation
- OOP Teaching

Protect
- Peer Review (initial) Commits
- Enforce proper capsulation
- Resolve Branching

Prioritize
- Lead developer
- Machine time for test
- Group leader support

Further discussion

- What changes for an operator?
  - Short term: nothing, same “open Matlab with standard scripts”
  - Long term: better user experience

    Less duplication of effort
    Reliability

- Problem: AFS is incompatible with version control
  Solution: move to NFS …\zn_pitz\NFS\Measure\scripts\
  make AFS …\measure\scripts\ **read-only**

  Ideally: shift crew commit changes as they go
  Realistically: changes are committed at the end of run period

Final Notes

- Utilize personal sandbox for development and duplicate into master PITZ repo

- PITZ Matlab guidelines:

https://confluence.desy.de/display/PITZ/Matlab+guidelines

- Scripts meetings: **next on Mon Dec. 2 @13:00**
  - Mailing list: pitz-dev