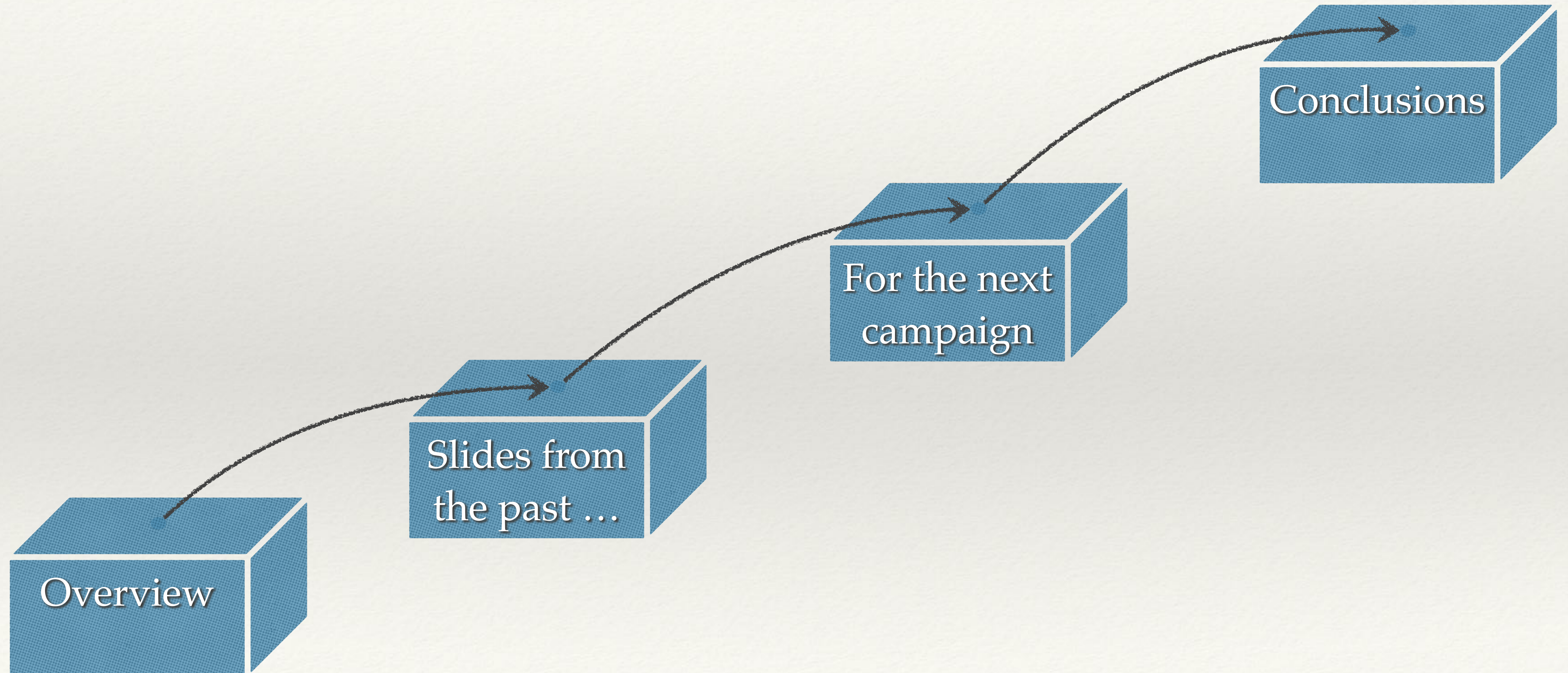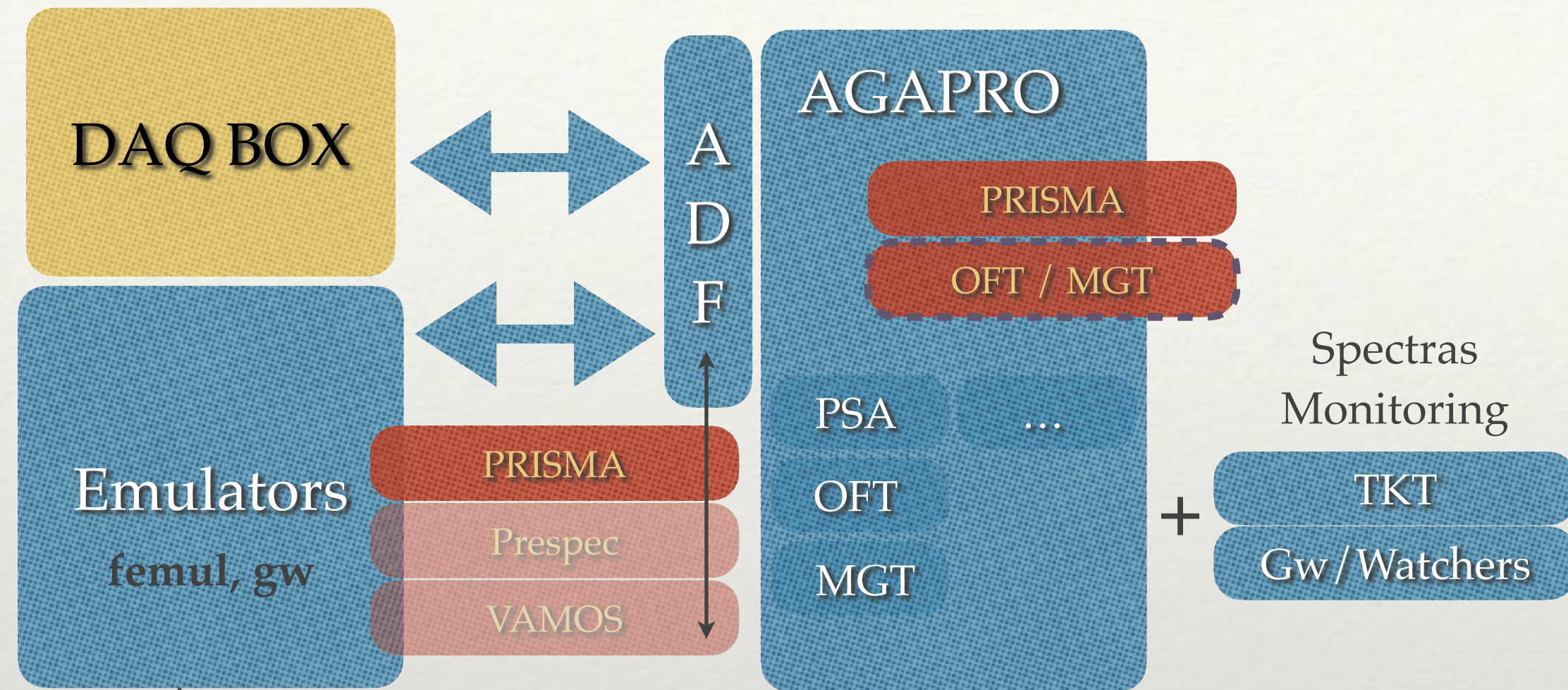# Data Analysis
# of future experiments
with AGATA & PRISMA

O. Stézowski

➥ From Legnaro around 2010 back to Legnaro about ten years later !

Outline
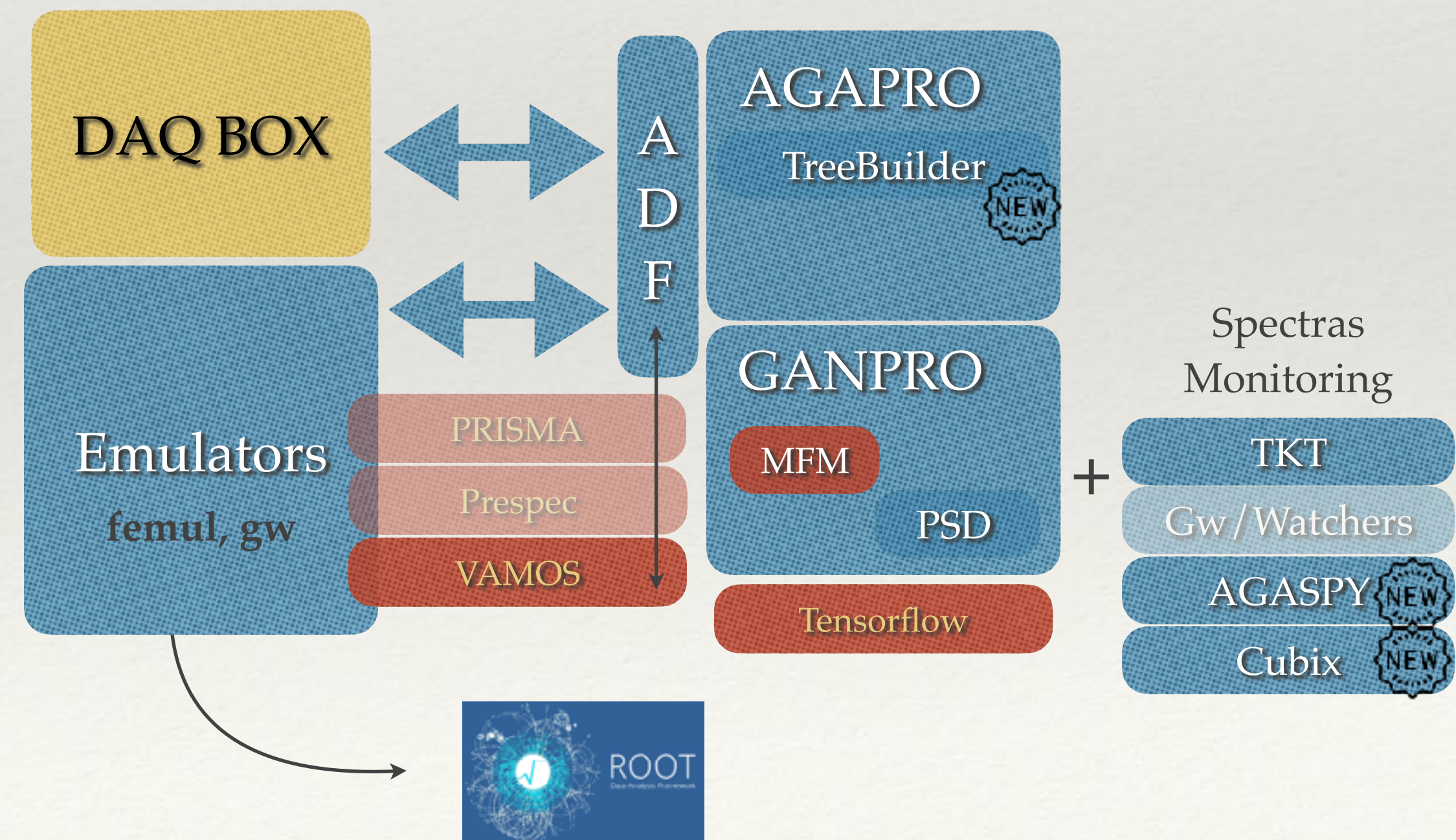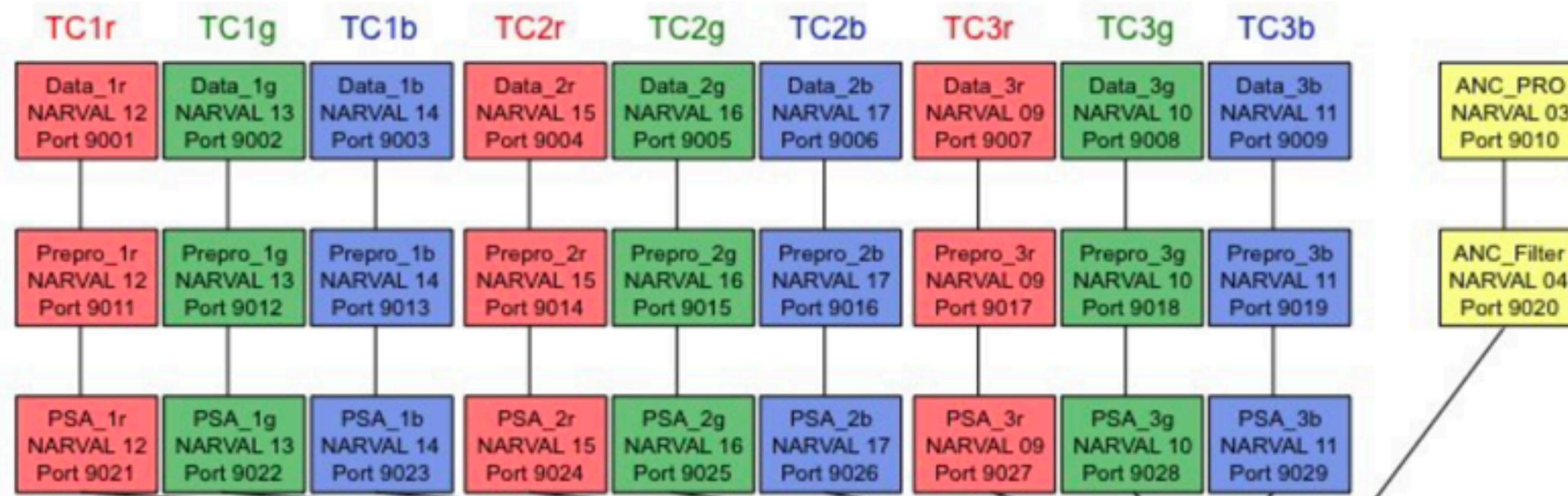
Overview : from right side back to left side …
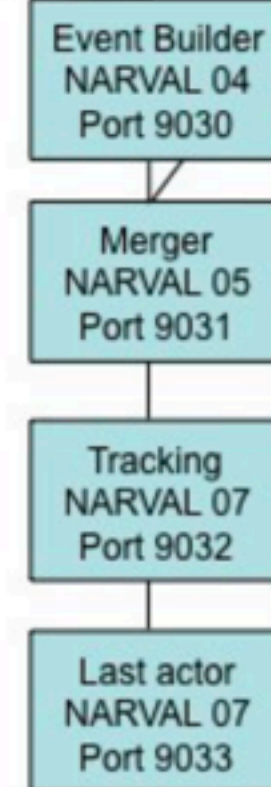
Slides from the past …
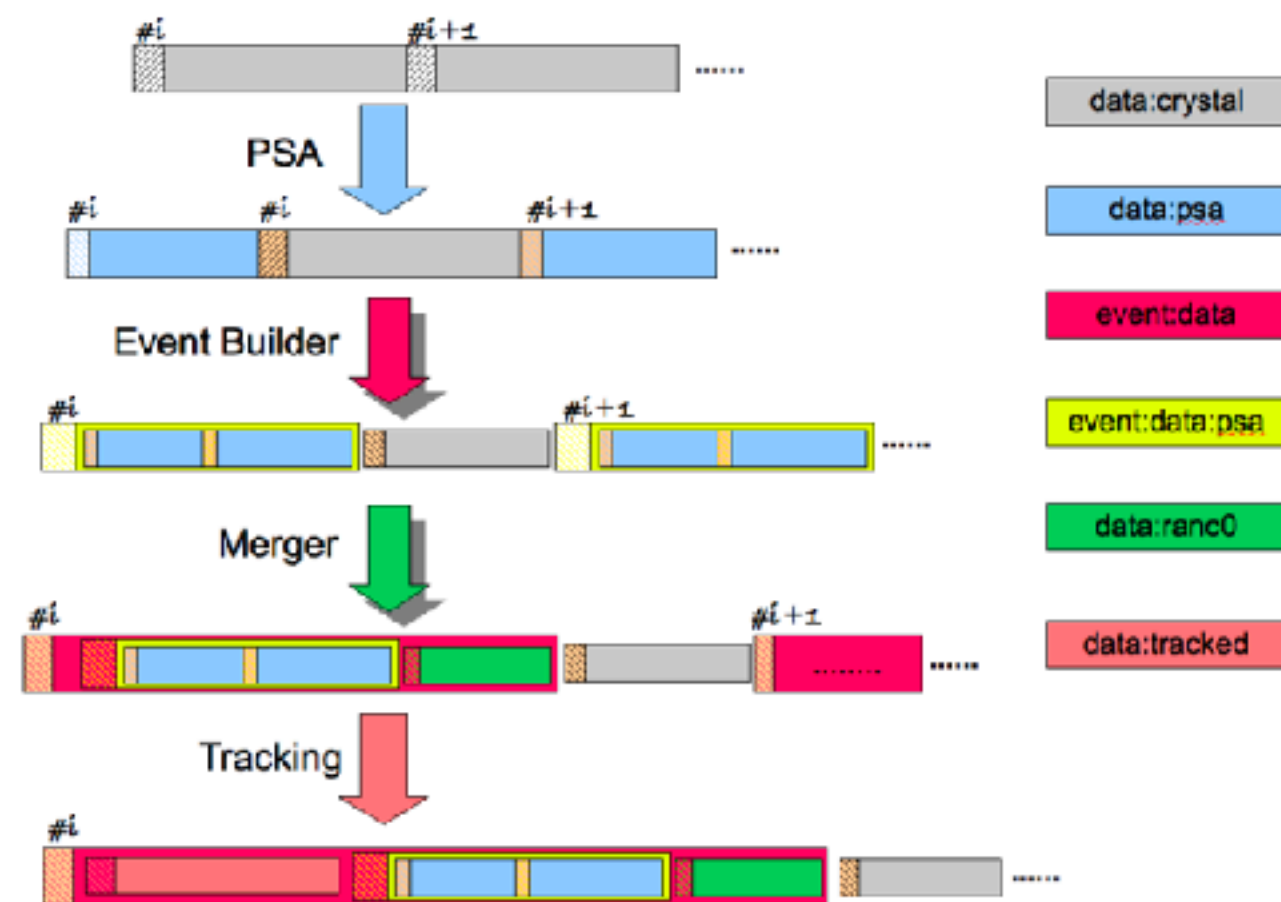
Slides from the past …

Topology at the beginning of AGATA … **We have much more Ge now !**
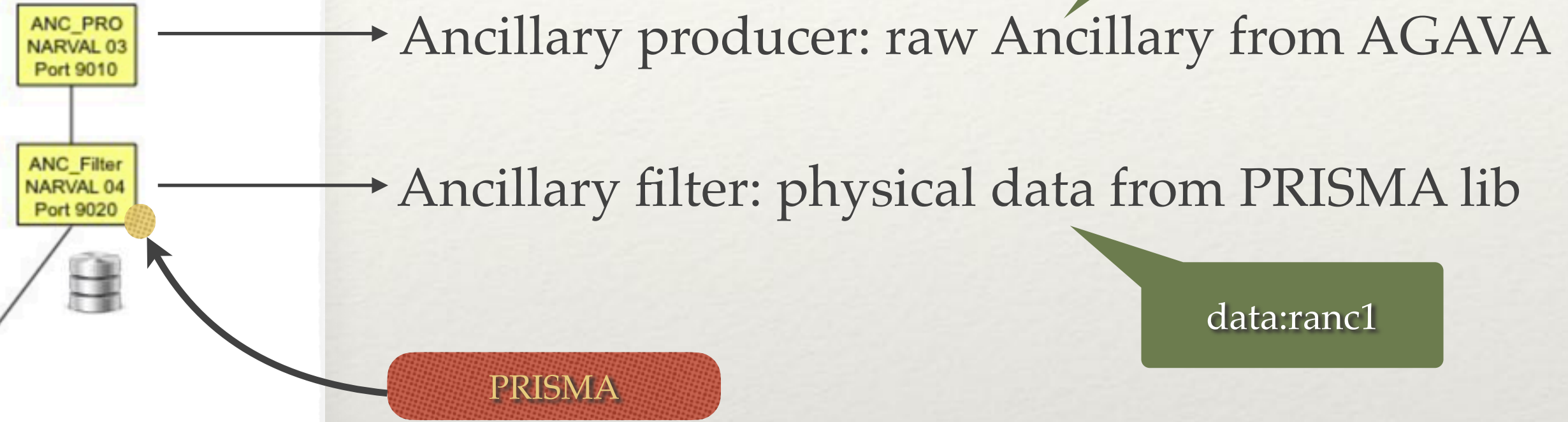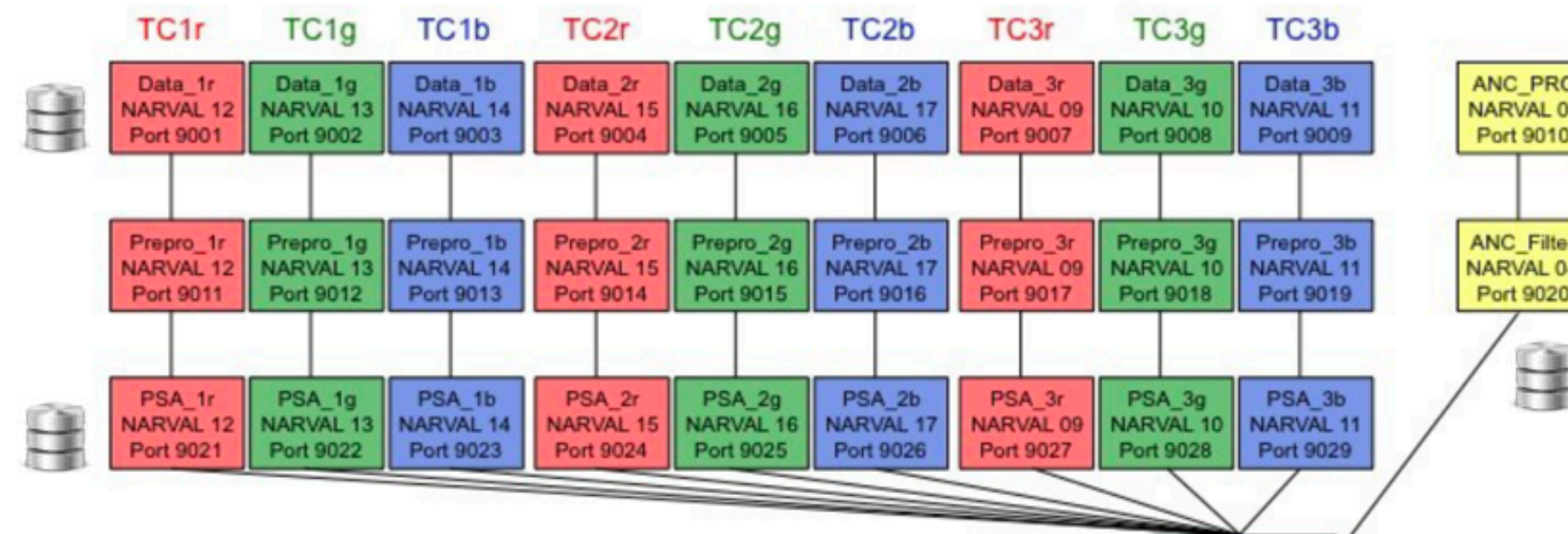
data:ranc0

Ancillary producer: raw Ancillary from AGAVA

Ancillary filter: physical data from PRISMA lib

data:ranc1

PRISMA

Merged at the same level

RunXXX.adf

Watcher

Data flow structure almost unchanged !

Watchers : tasks to specific frames

Slides from the past ...
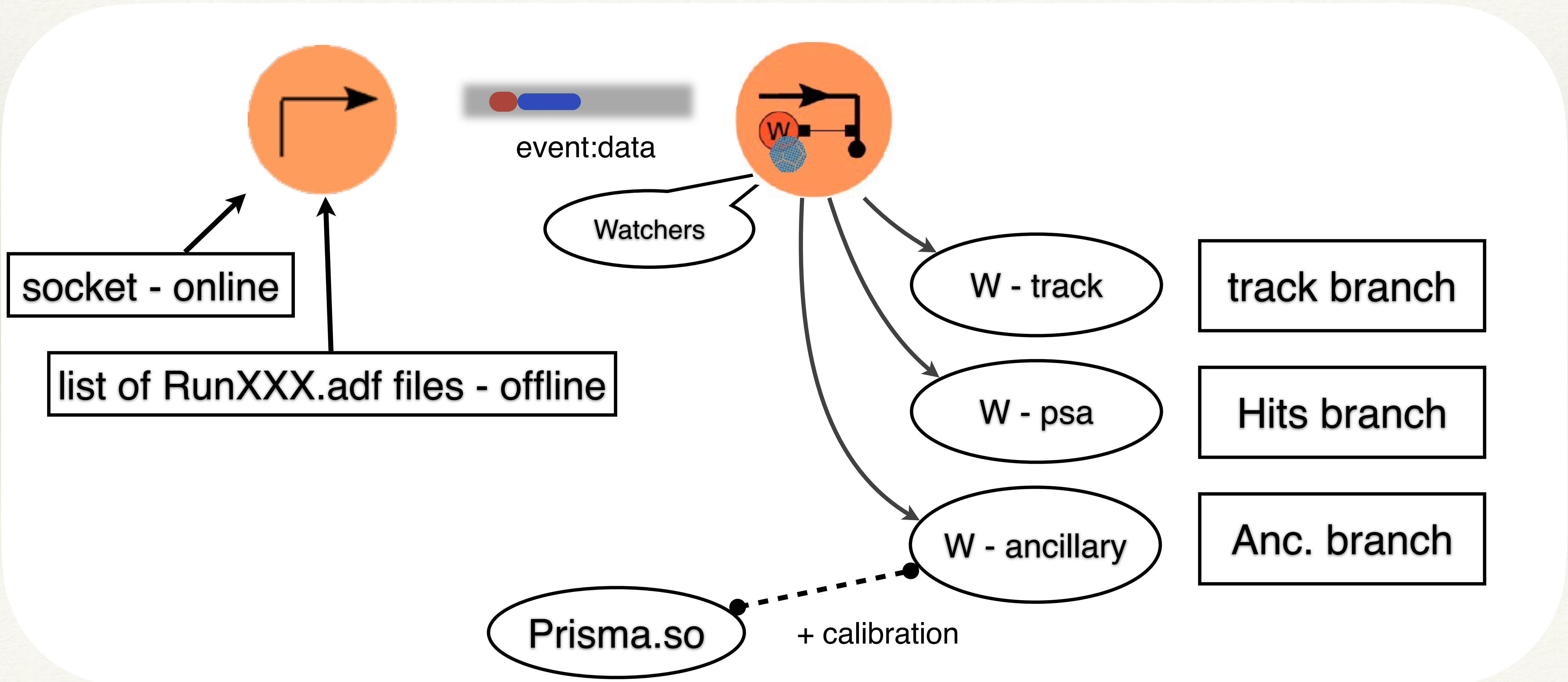
Slides from the past …
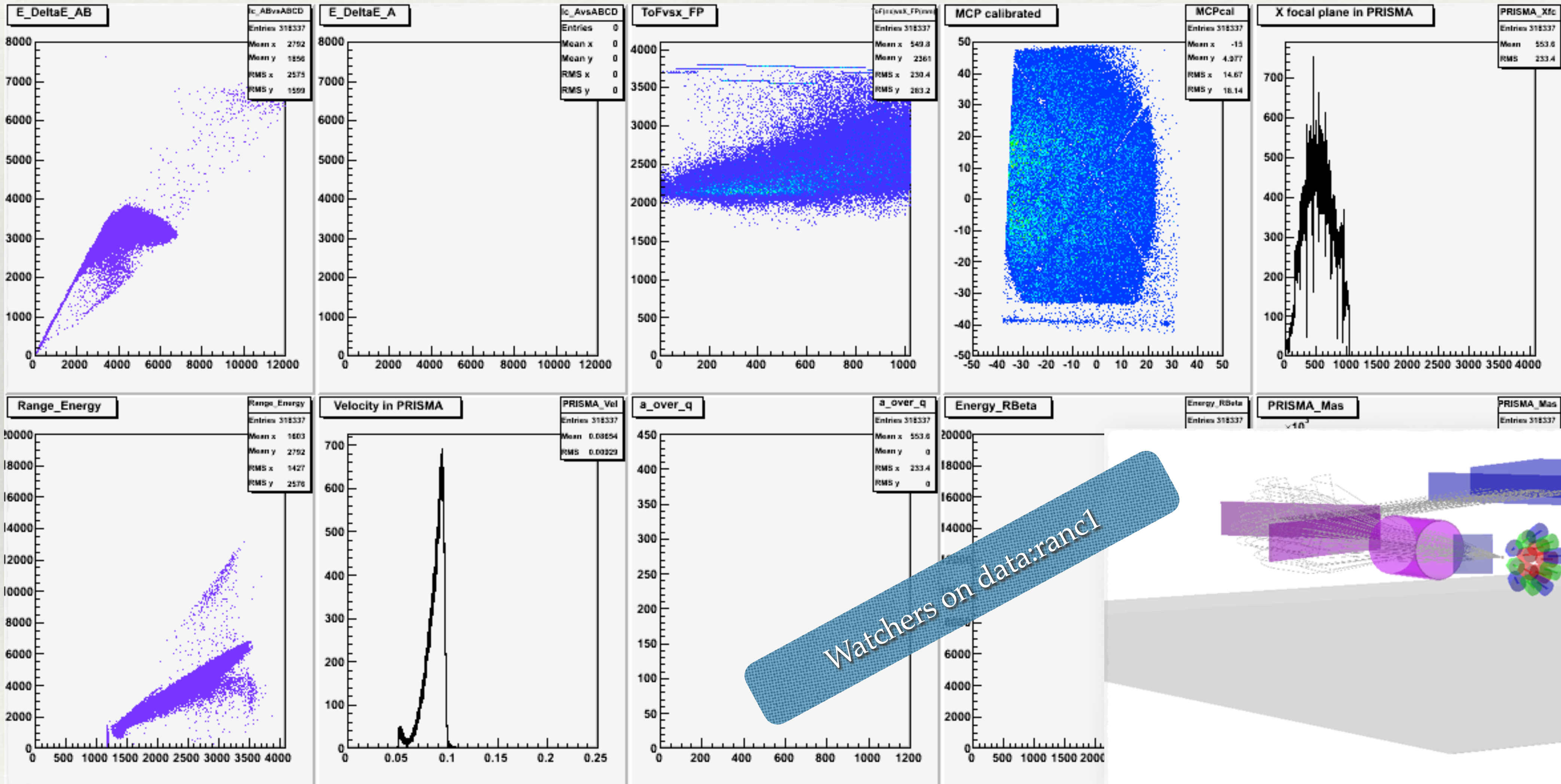
Watchers on data:ranc1
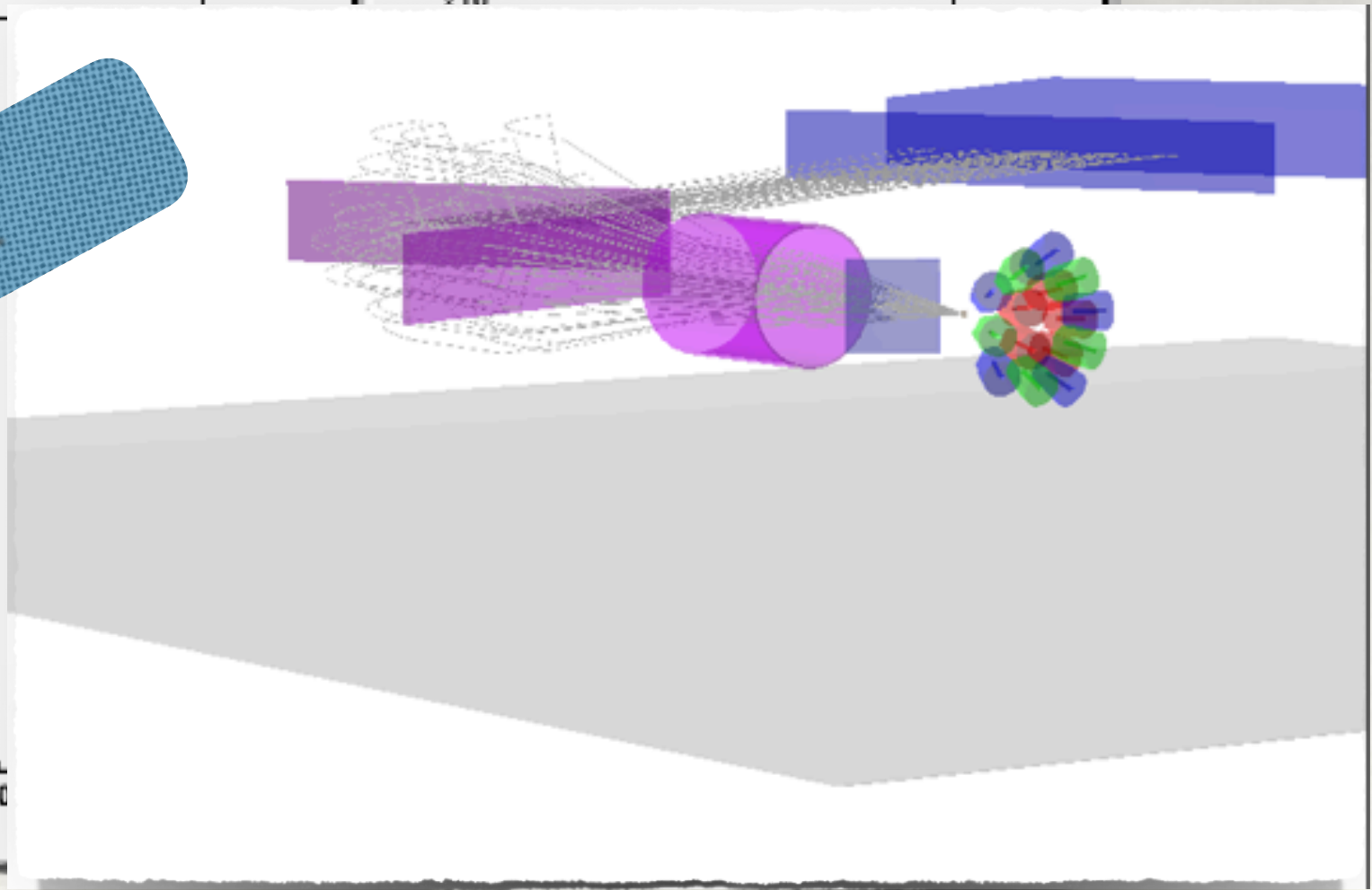
For the next campaign

More AGATA Local Level
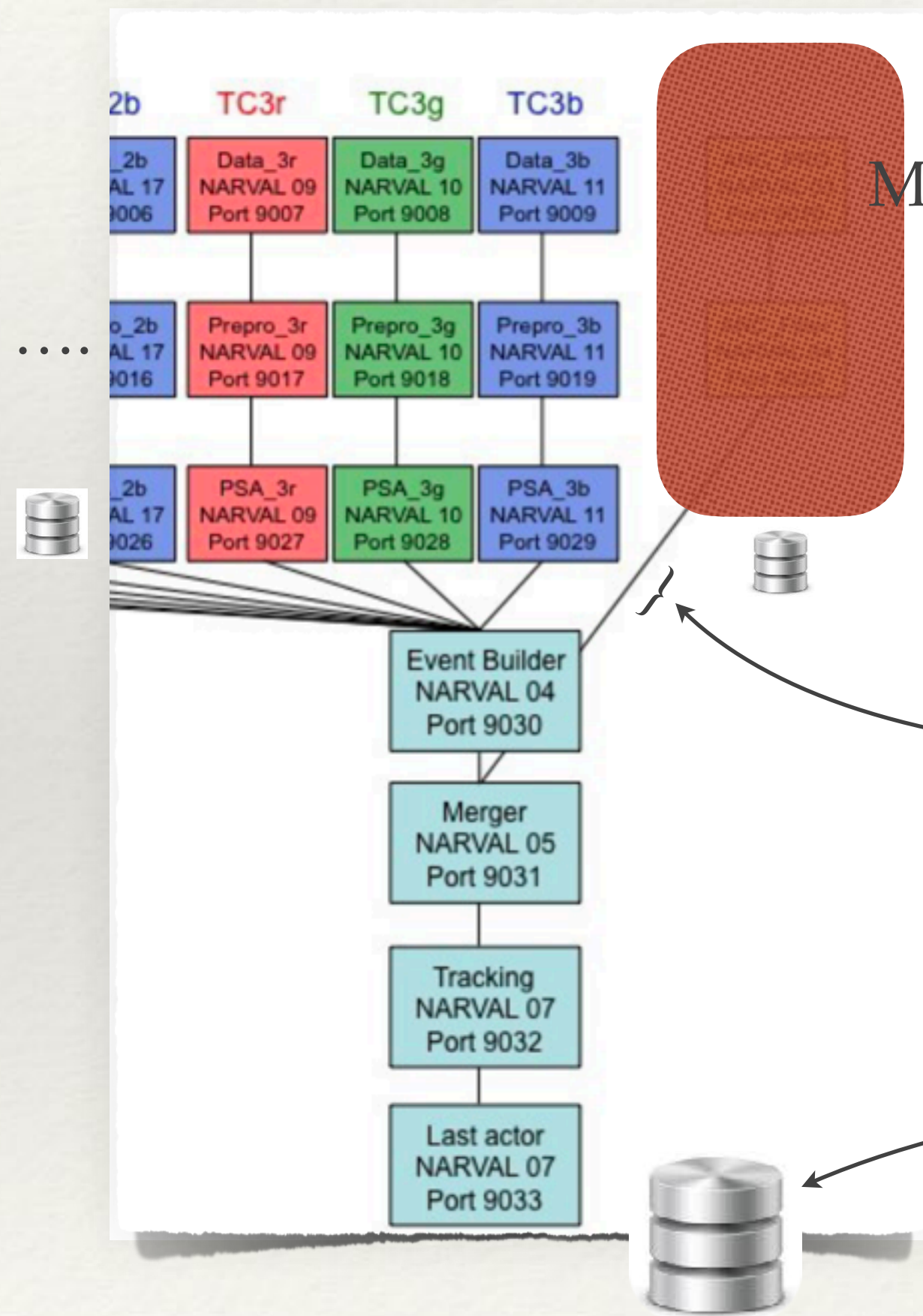
More complex … this part is managed by the LNL DAQ box - XDAQ

NOTE: LNL xdaq data is ADF compliant

In the AGATA world, again :
data:ranc0 - raw data as produced by the AGAVA card
data:ranc1 - data out of the PRISMA library

Direct storage in ROOT TTrees !

For the next campaign

PRISMA lib 2010
- Distributed tarball
- Makefile to build

→

PRISMA lib 2020-
- In a git repository https://baltig.infn.it/prisma/prisma_library
- Cmake (almost done)

Move code
from Watchers to TreeBuilder

Move code
from Watchers to AGASPY

To be done … should not be time consuming …

<u>Distributions:</u>

Data will be on the GRID

GridDataSync.py script to get them in https://gitlab.in2p3.fr/IPNL_GAMMA/scripts

Software on git

gRaySoftware.py script to get them in <u>https://gitlab.in2p3.fr/IPNL_GAMMA/scripts</u>
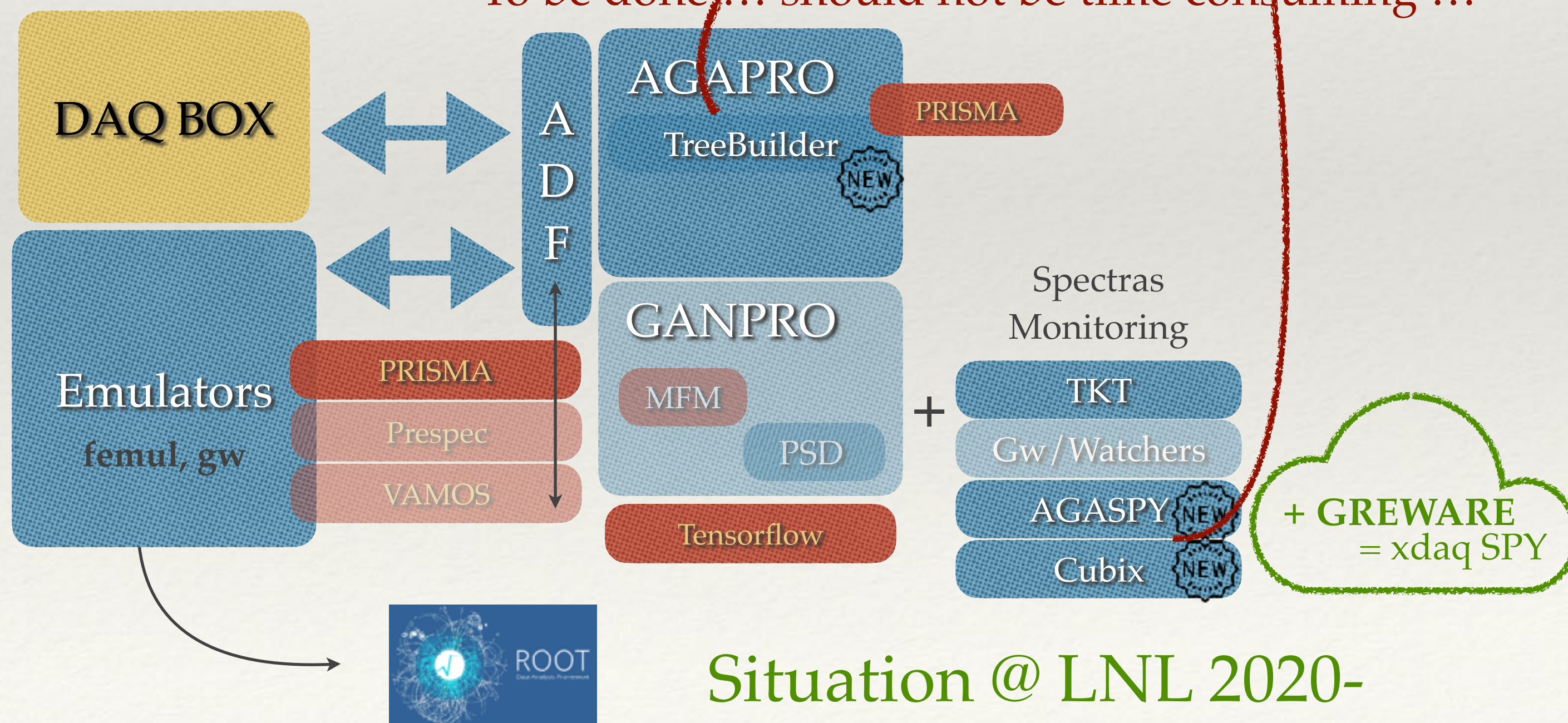—> <u>libPRISMA to be added</u> … but almost trivial if git/cmake

+ container with « all » the software installed in it !
<u>libPRISMA to be added …</u>

DAQ BOX

A D F

AGAPRO
TreeBuilder
PRISMA
NEW

Emulators
femul, gw

PRISMA
Prespec
VAMOS

GANPRO
MFM
PSD

Tensorflow

Spectras
Monitoring

TKT
Gw/Watchers
AGASPY NEW
Cubix NEW

+

+ GREWARE
= xdaq SPY

ROOT

Situation @ LNL 2020-

For the next campaign ... few words about containers

Containers are 'light' virtual machines
- Applications come with there dependencies ☞ no need to worry about missing libraires, versions issues etc ...

- On LINUX machines, no CPU overhead

It requires to have docker or singularity installed on the machine
- Singularity might be 'safer'. Contrary to docker, the user is not root in the container
- One can run a docker in singularity ...

In gitlab, docker are used to test automatically compilations (=continuous integration)
A full docker is produced and available here *gitlab-registry.in2p3.fr/ipnl_gamma/docker_gamma:prod*

Example

3 - link the directory with all the AGATA data

```
sudo docker run -e "HOME=/home/developper" -v $HOME:/home/developper --rm -it --user $(id -u):$(id -g) gitlab-registry.in2p3.fr/ipnl_gamma/docker_gamma:prod
Unable to find image 'gitlab-registry.in2p3.fr/ipnl_gamma/docker_gamma:prod' locally
prod: Pulling from ipnl_gamma/docker_gamma
f22ccc0b8772: Pull complete
3cf8fb62ba5f: Pull complete
e80c964ece6a: Pull complete
9bed514c2952: Downloading [===


I have no name!@9ff0b12a56d4:/$ ls /opt
AgataSoftware  data  install_agata.sh  root  tensorflow
I have no name!@9ff0b12a56d4:/$ls /opt/AgataSoftware/bin/
ListFrames  MFMtest  RecalEnergy  SortCapsule  SortFFTSortPsaHits  femul  xTalkInvert  xTalkMake  xTalkSort

exit
```

1 - Downloaded first time

2 - you are in the container .... It looks like standard shell

4 - exit to stop the container

Conclusions

Goals :

Start the second LNL campaign avoiding new developments: only what is mandatory to run
« easier » because of the first campaign and because the main ingredients are still used

Still some work to be done for the first weeks but it should not required huge steps
(I might be wrong ! Hopefully not …)

☞ container technologies into the game, even if not mandatory
☞ However, probably better to get use to …

Future :

Important developments are foreseen to face the future …
  … to be integrated once the system is operational at Legnaro !