PHYSH

Giulio Eulisse
Northeastern University of Boston

Outline

- What is PHYSH?
- Demo
- Some details on architecture and implementation.
- Questions? Discussion.

What is PHYSH?

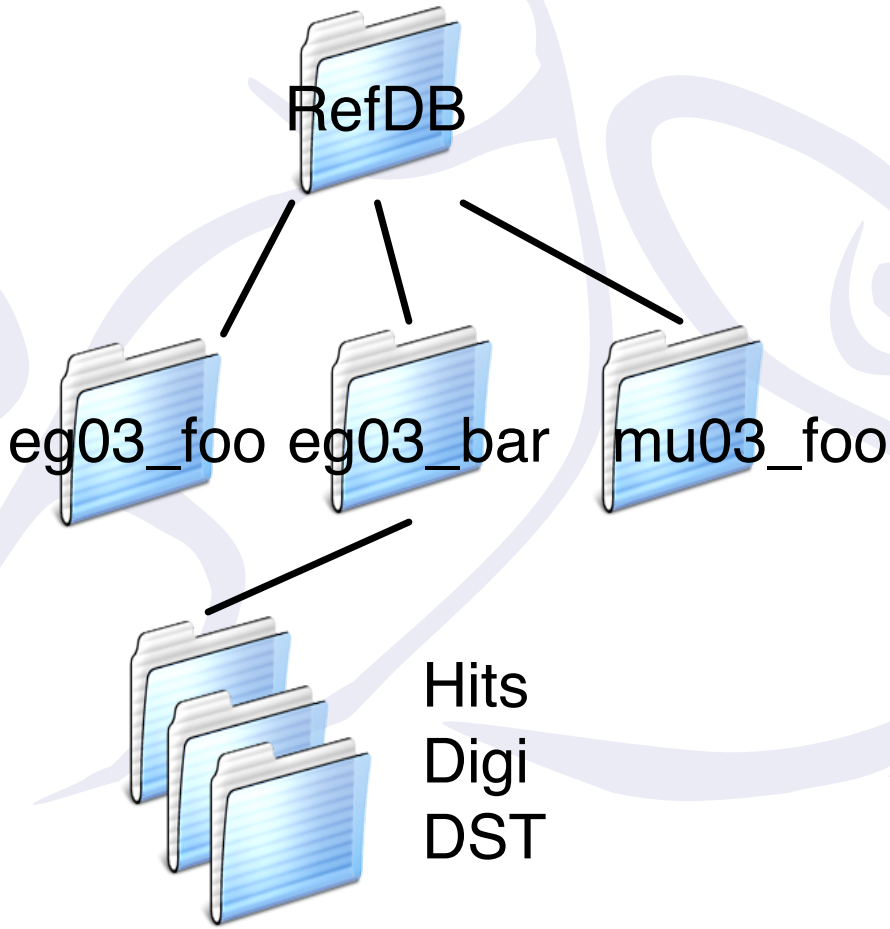
PHYSH: what is it all about?

- PHYSH is thought to be the end user PHYsicists SHell.

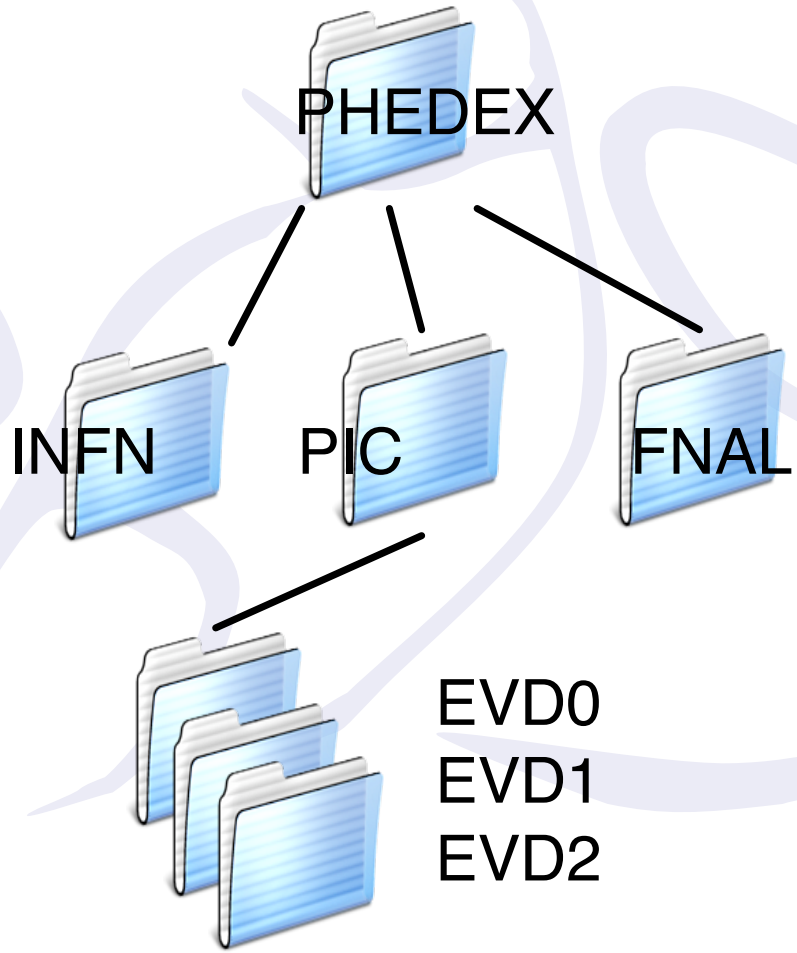
  1. It is an extendible glue interface among different services (already present or to be coded).

  2. The interface to the user is modelled as a virtual filesystem interface.

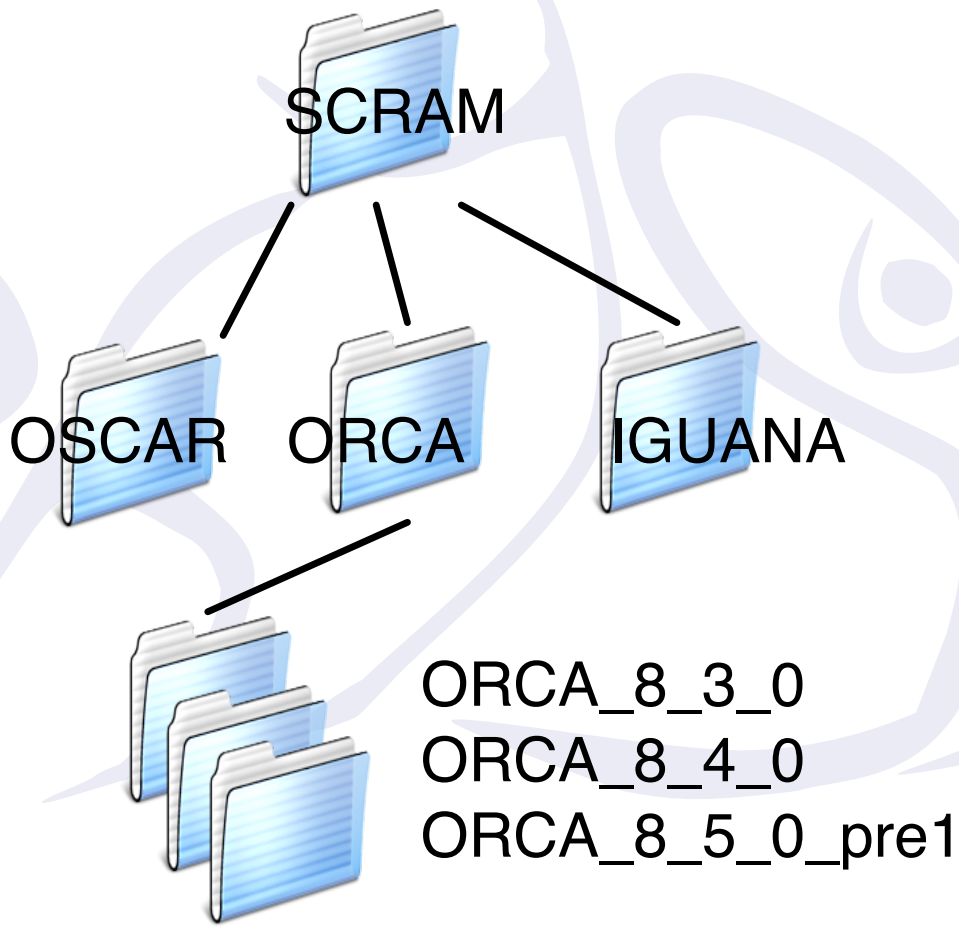
Hierarchical organization(1)



Hierarchical organization(II)

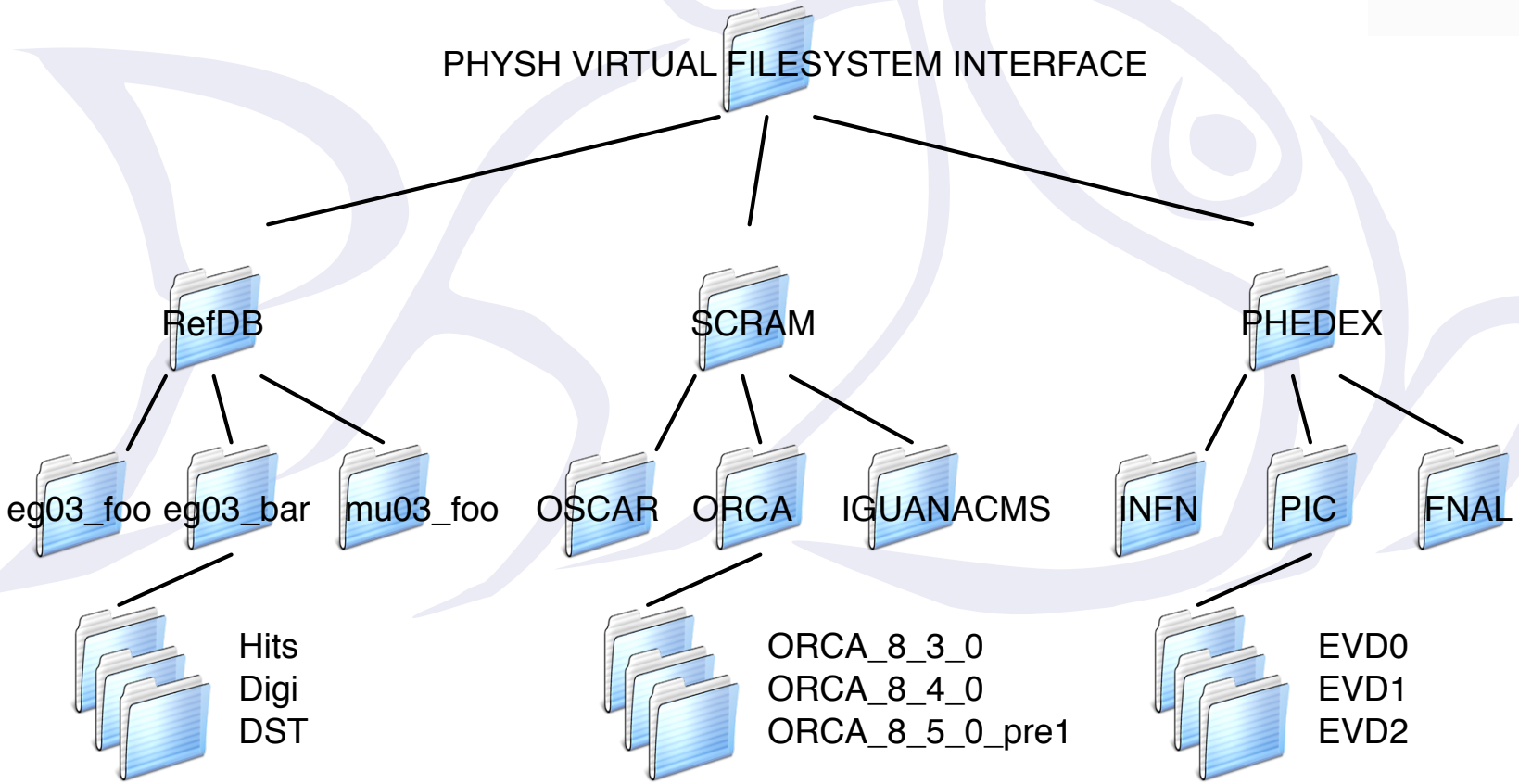


Hierarchical organization(III)



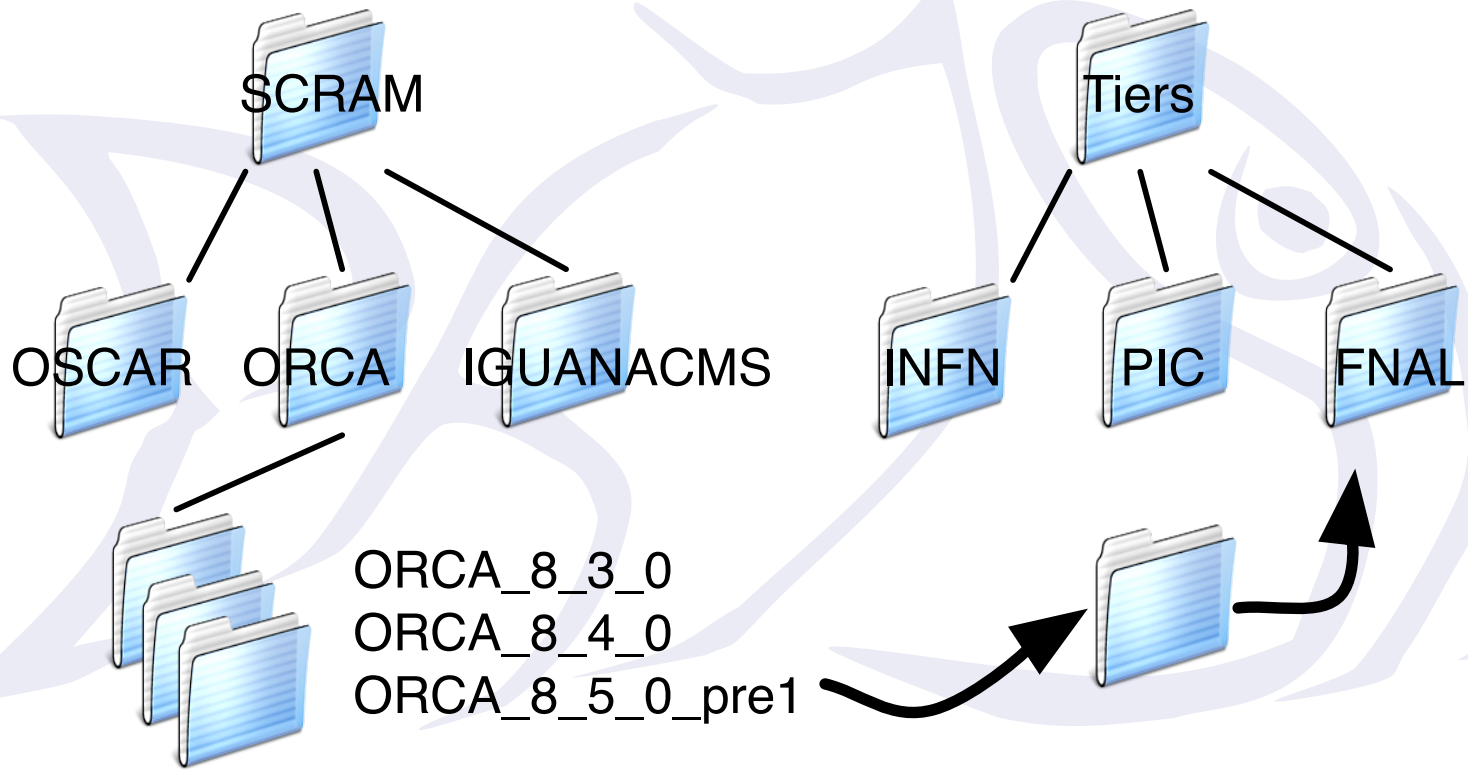
Do you see a pattern???

I bet you do...



File handling like interface for all actions

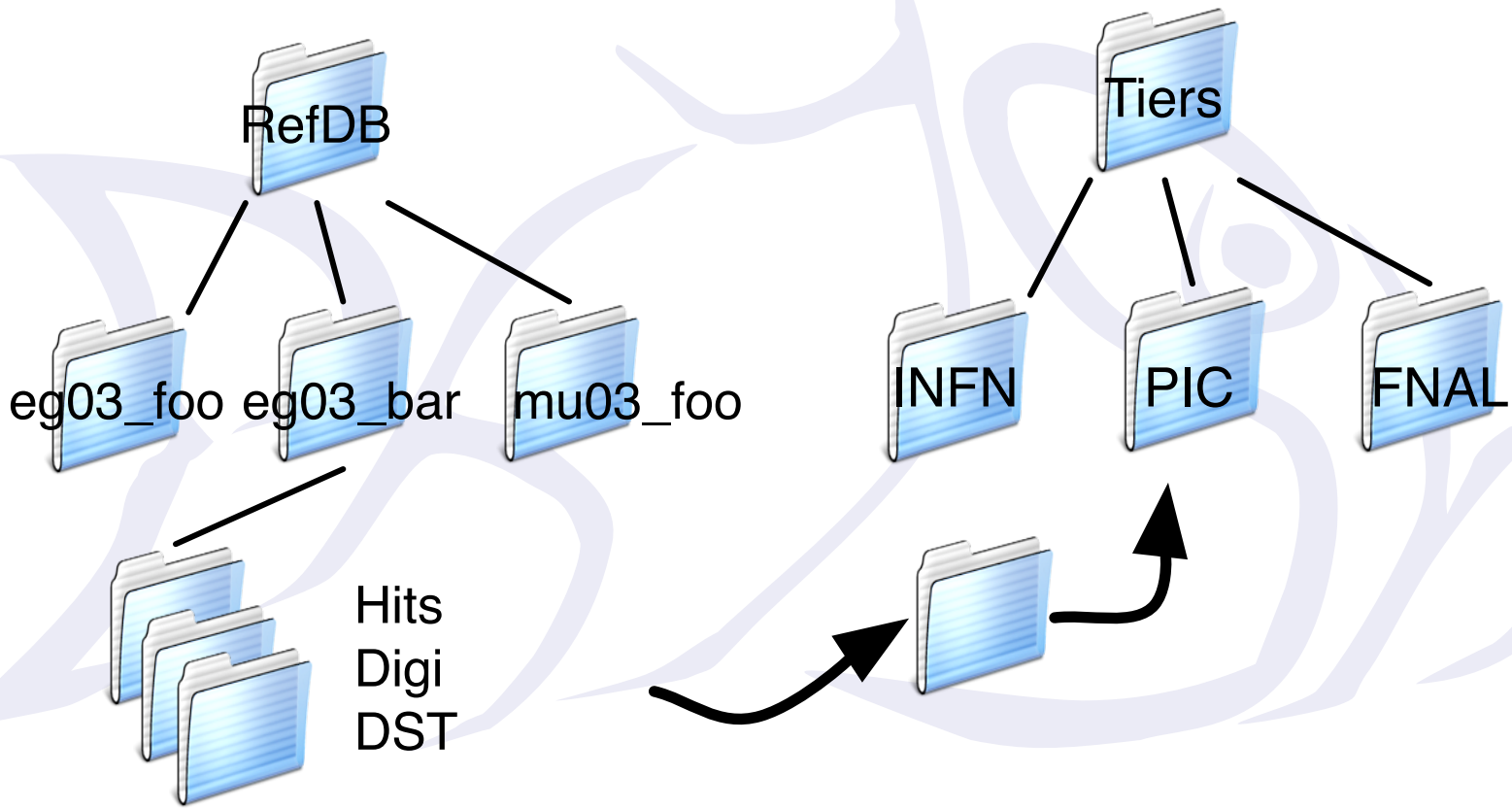
cp /Applications/ORCA/ORCA_8_5_0_pre1 /Tiers/FNAL/



installs ORCA_8_5_0_pre1 at FNAL

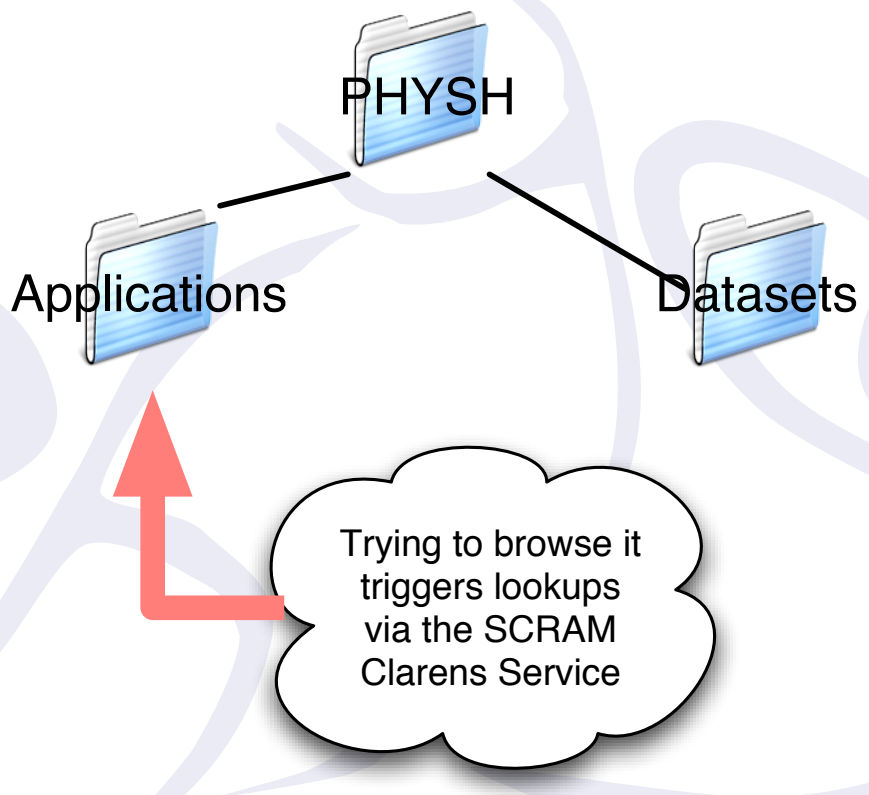
File handling like interface for all actions

cp /RefDB/eg03_foo/DST /Tiers/PIC

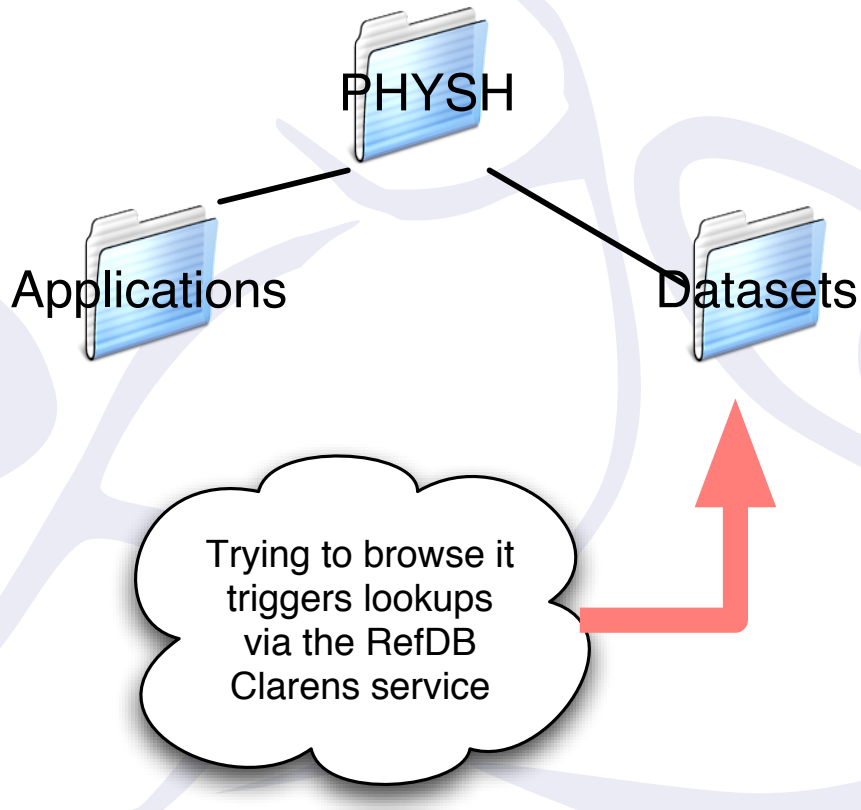


copies a certain collection to PIC

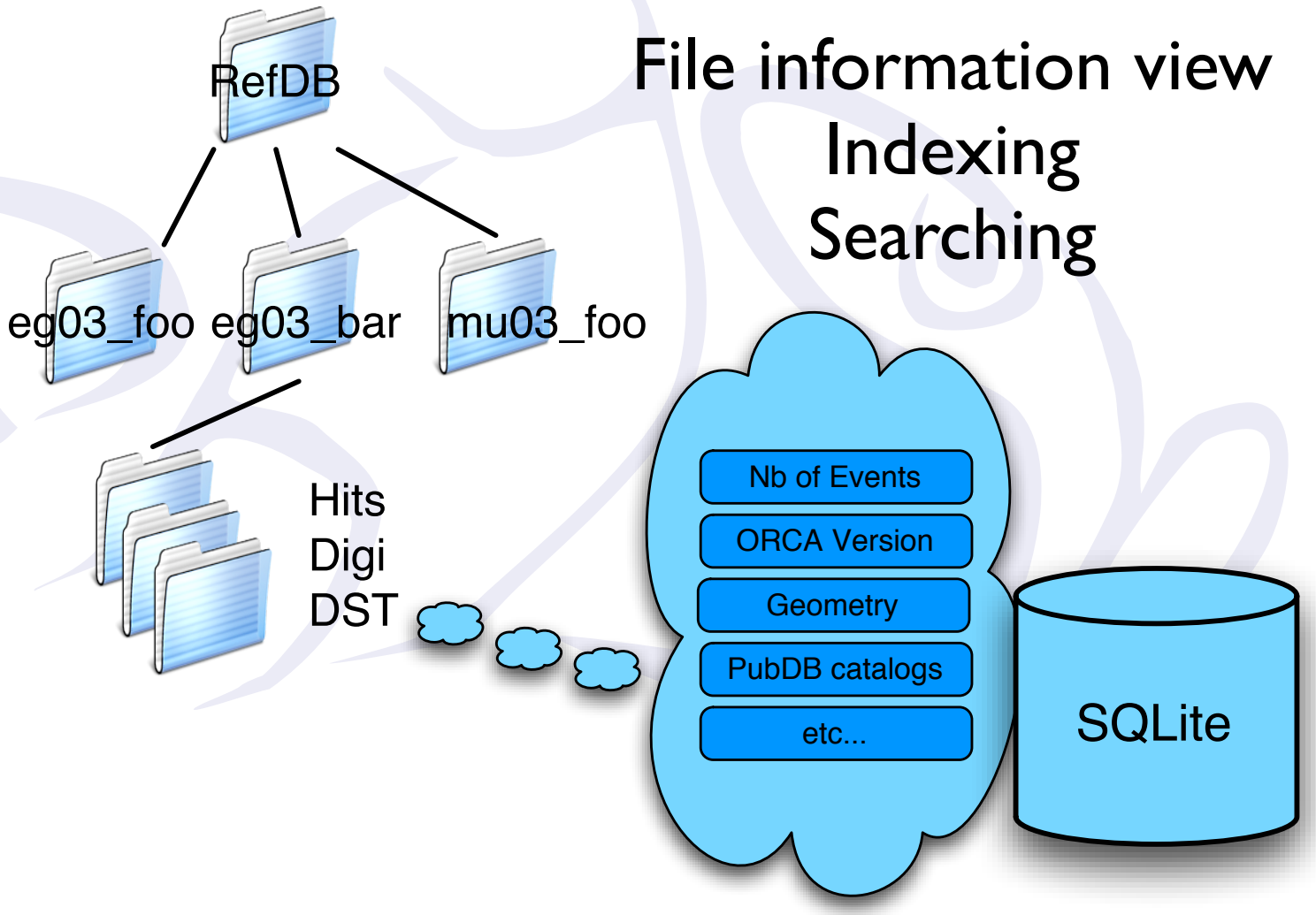
Files can be of different types



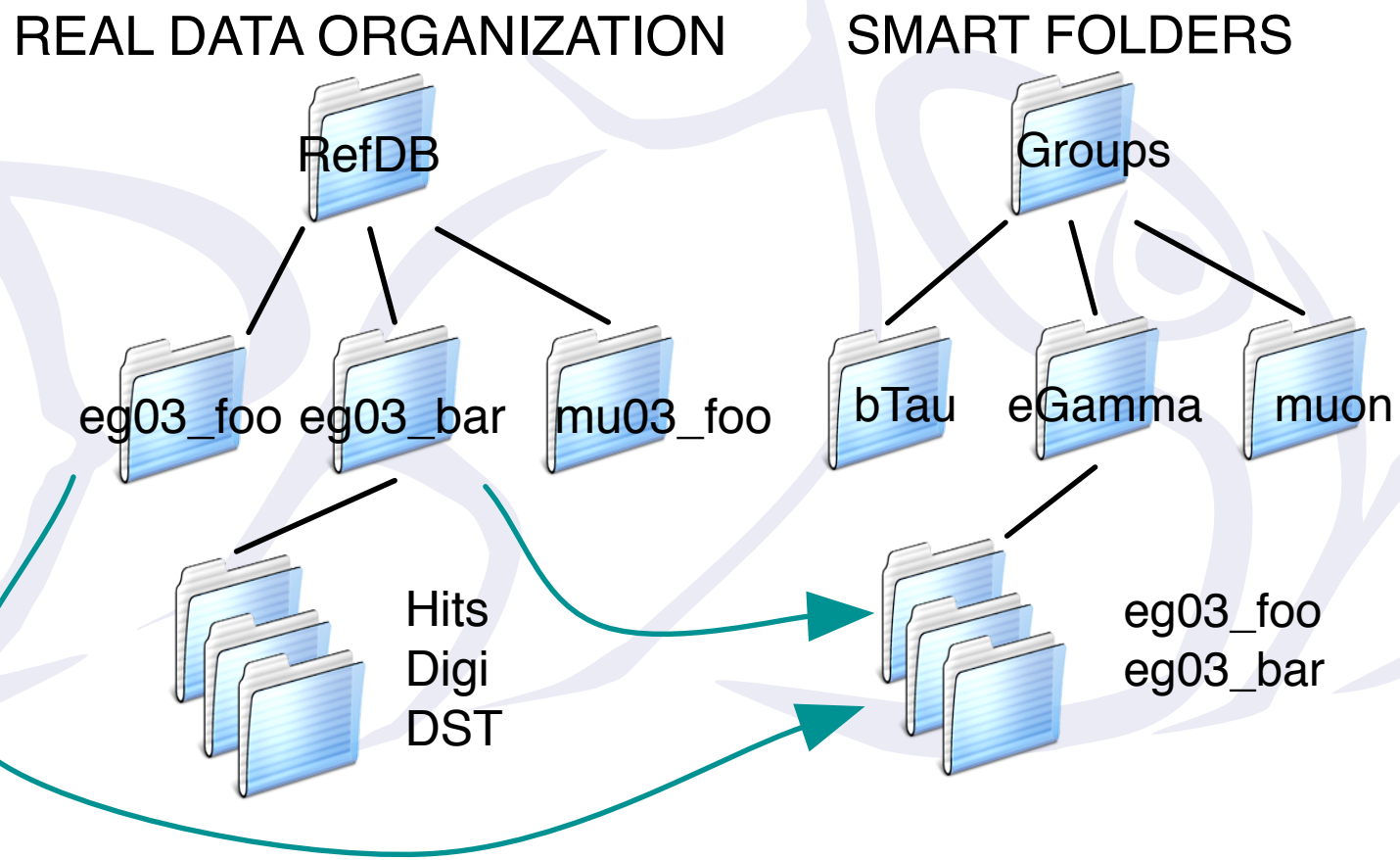
Files can be of different types



Files have metadata



Smart folders

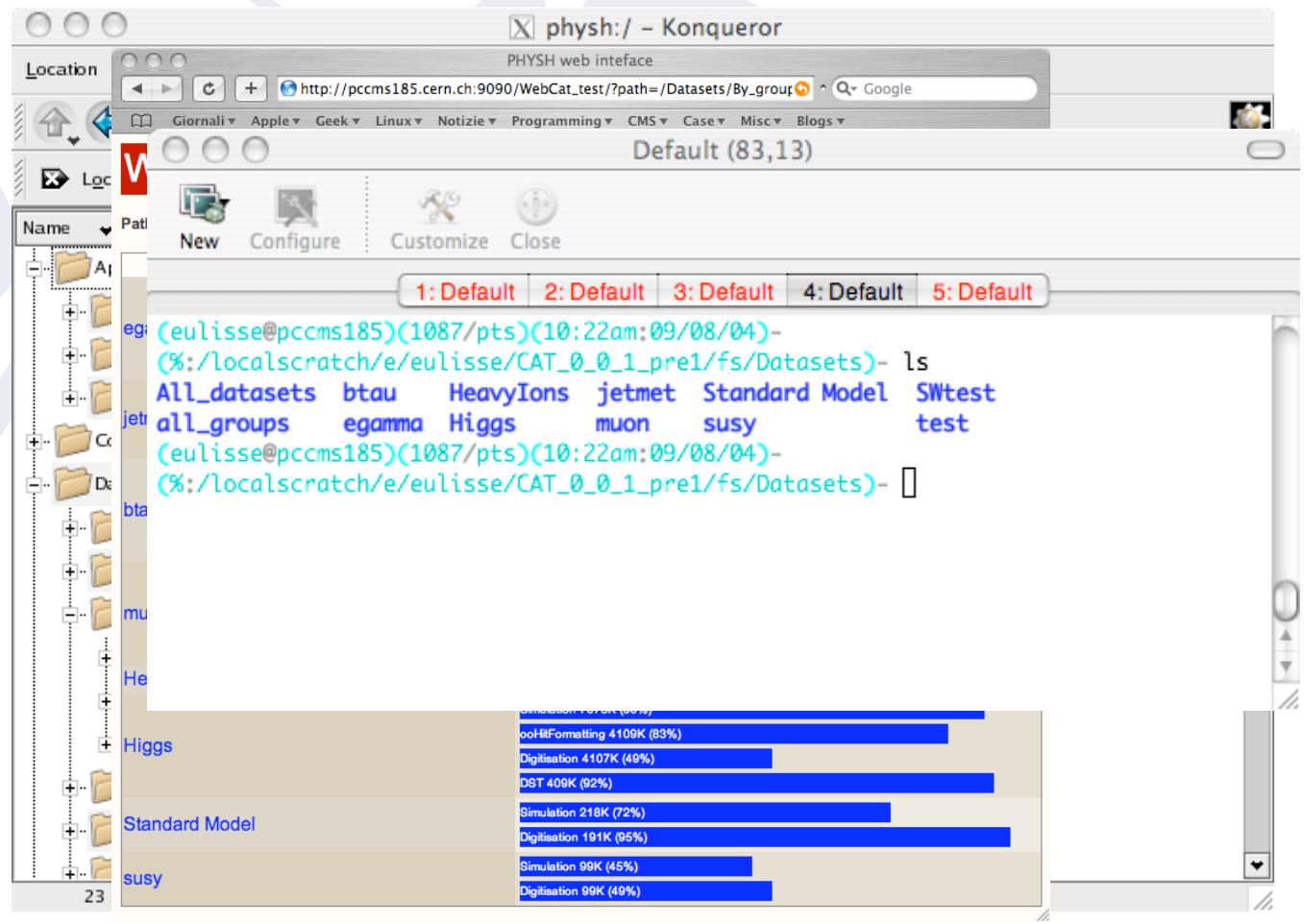


VFS interface

Filesystem interafaces is what most people are common with when dealing with their data.

Data is organanized as files in a directory structure.
Actions are expressed in terms of copying, linking, creating files.

Multiple clients satisfy different needs.



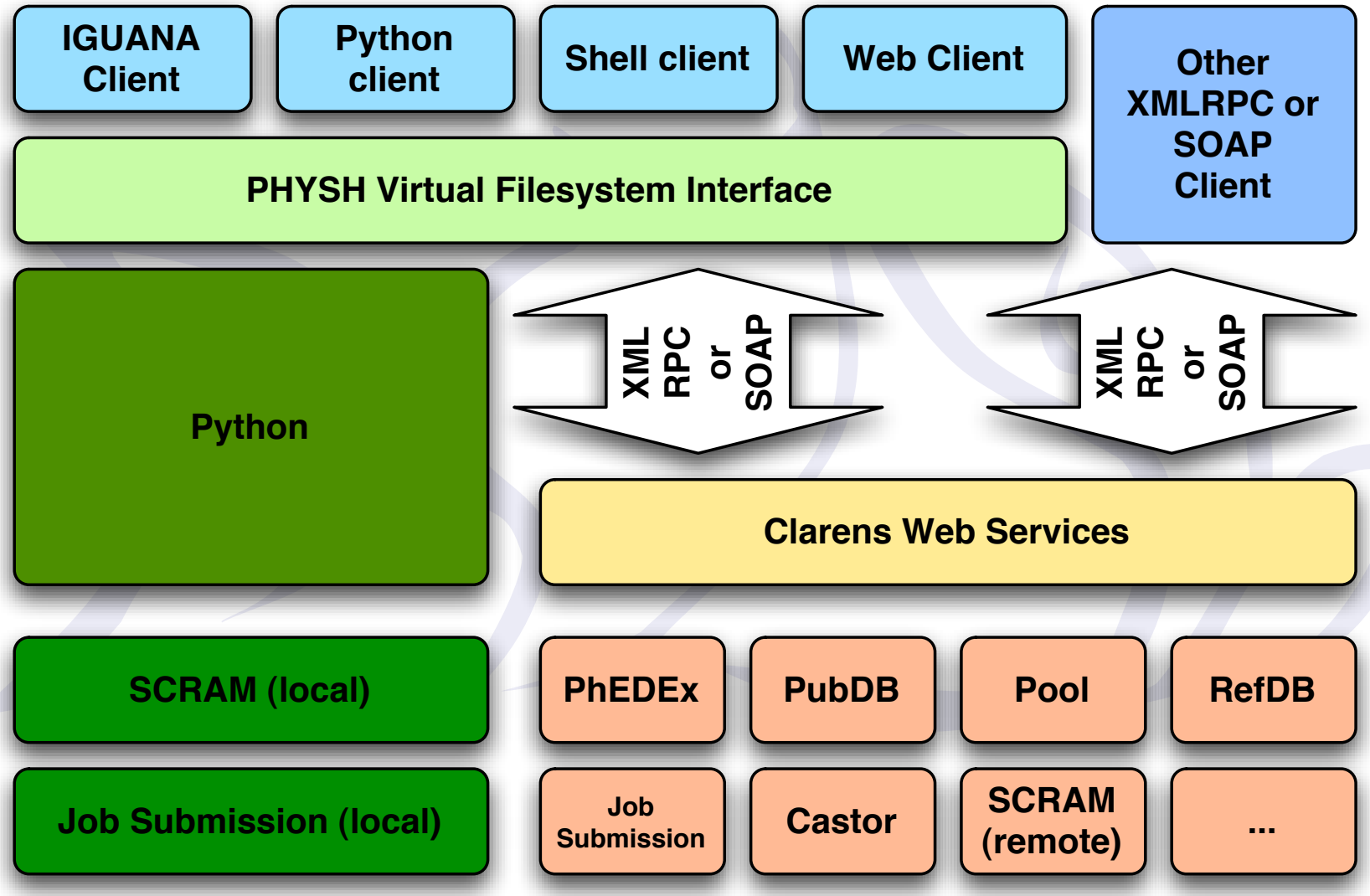
Example PHYSH script

```
#! physh

#This script executes ExDSTStatistics on all the datasets with 8_7_1
#DST of susy group.

#Required until we get proper population support
ls Applications/ORCA/ORCA_8_7_1
ls ComputingElements

for x in `ls Datasets/susy/DST/su_DST871_2x1033PU_g133_OSC`
do
    echo $x
    JOBDIR=Workspace/MyJobFor$x
    mkdir $JOBDIR
    ln Applications/ORCA/ORCA_8_7_1/ExDSTStatistics $JOBDIR
    cp Datasets/susy/DST/su_DST871_2x1033PU_g133_OSC/$x $JOBDIR
    cp $JOBDIR ComputingElements/Grid
done
```

Demo

Some details on architecture and implementation

WebService based architecture



Why WebServices

- The reason for using WebServices:
  - To insulate the client from the backend.
  - To provide authentication/permission to the various actions.
  - To have active behaviour of the service.

Clarens

- Nice WebServices framework from Caltech, part of GAE.
- Why Clarens:
  - Uses python (COBRA and IGUANA already use python as scripting language).
  - Easy client setup (one python file).
  - Easy to write service for: my summer student, Leonardo Sala, wrote a nice working PHEDEX monitoring service from scratch (he did not even know python) in less then a couple of weeks.
  - It was there.
- One note: I'm not involved in the official Clarens development (apart from bug-fixing ;-)), just a mere user (who happens to like it).

Clarens features

- Certificate based authentication (based on apache).
- ZeroConf/Monalisa service discovery.
- *ACLs* for API

Clarens features

- API *ACLs*.

  Based on the certificate used to authenticate to the server, the client gets access (or not) to some API.
  Usecase: in PHEDEX Service you could allow "everyone" to call *submitTransferRequest*, but only few key people to access the *authorizeTransfer* method.

PHYSH additions on top of Clarens

- RefDB, PubDB, DLI, PHEDEX (Leonardo Sala @ unimib.it) prototype services for allowing the PHYSH client to work. These are mainly SQL query proxies.
- Configuration Service.

RefDB, PubDB services

- Provide SQL proxy to the database.
- Provide some canned queries.
- Somewhat insulate the client from the database backend.
- No proper design behind them but more like: ”Let’s have the client running”.

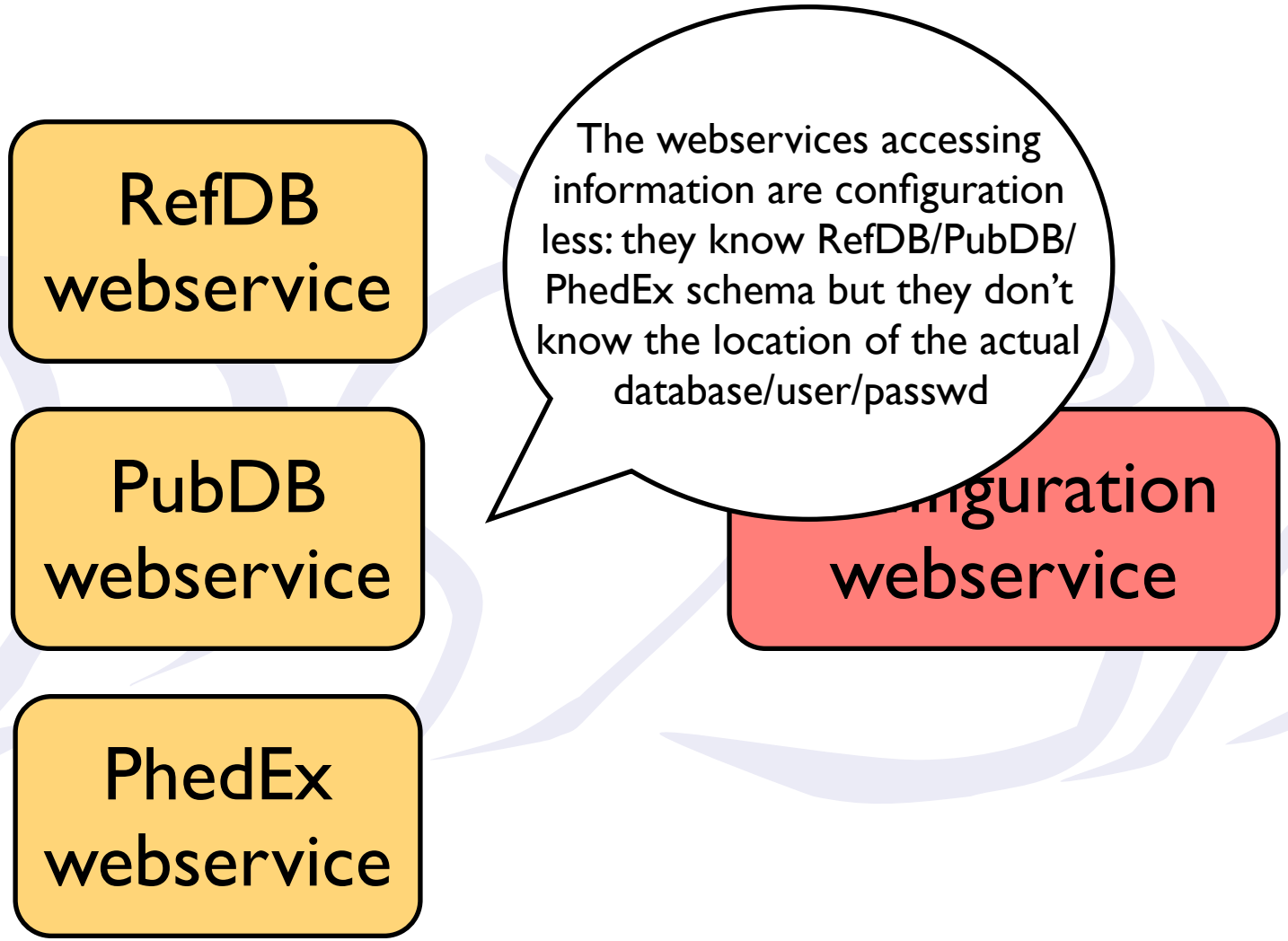
DLI service

- In PHYSH repository for historical reasons.
- Provides a "listReplicas method" that is understood by the EDG RB and allows it to take decisions according to the position of the data.
- See the work done by Heinz Stockinger for CMS as a reference.

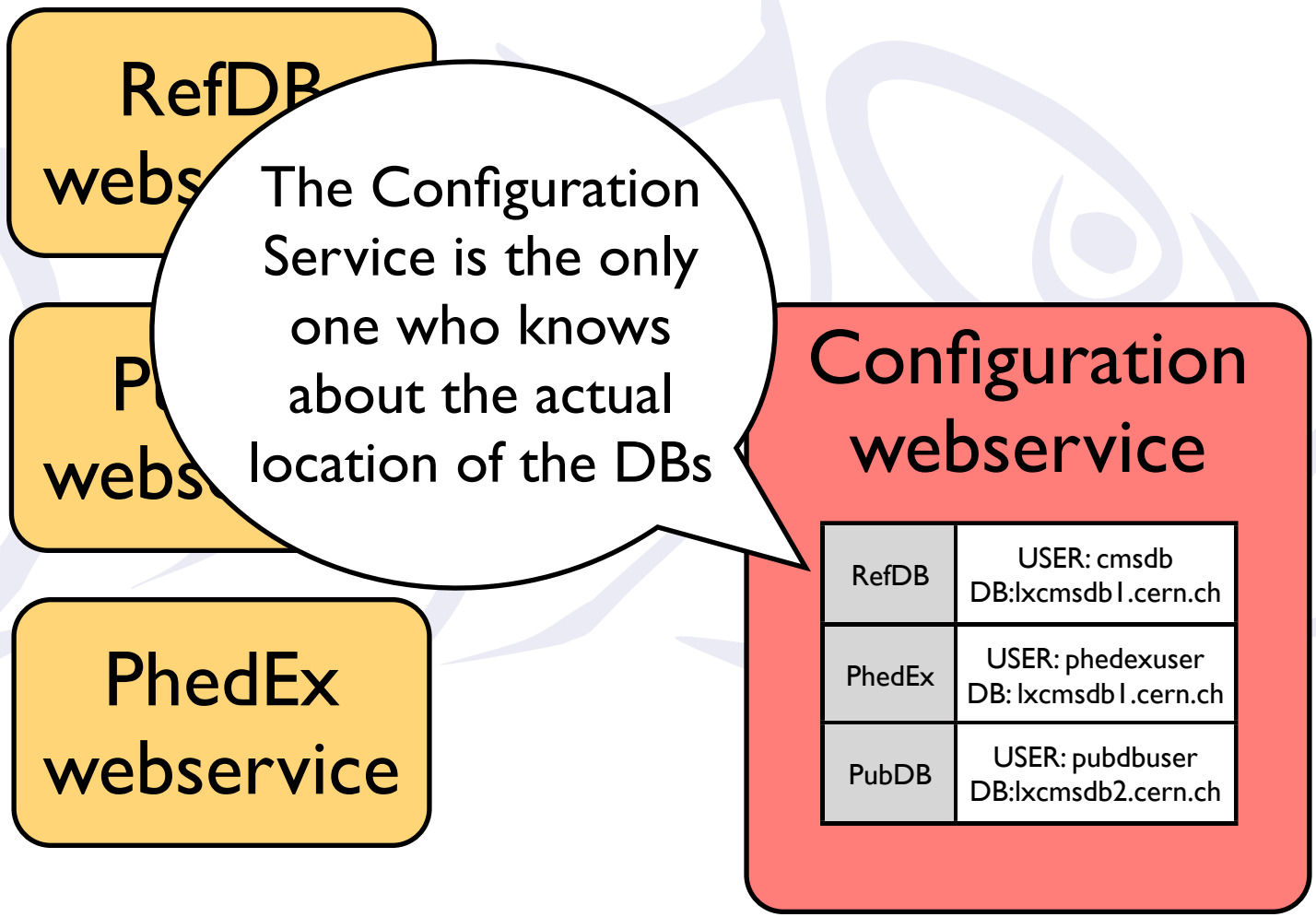
PHEDEX service

- Webservice wrapper around PHEDEX (CMS transfer manager software).
- Allows user to request the transfer of a dataset from one tier to another.
- Allows “administrators” to validate and submit the transfer.

Configuration Service

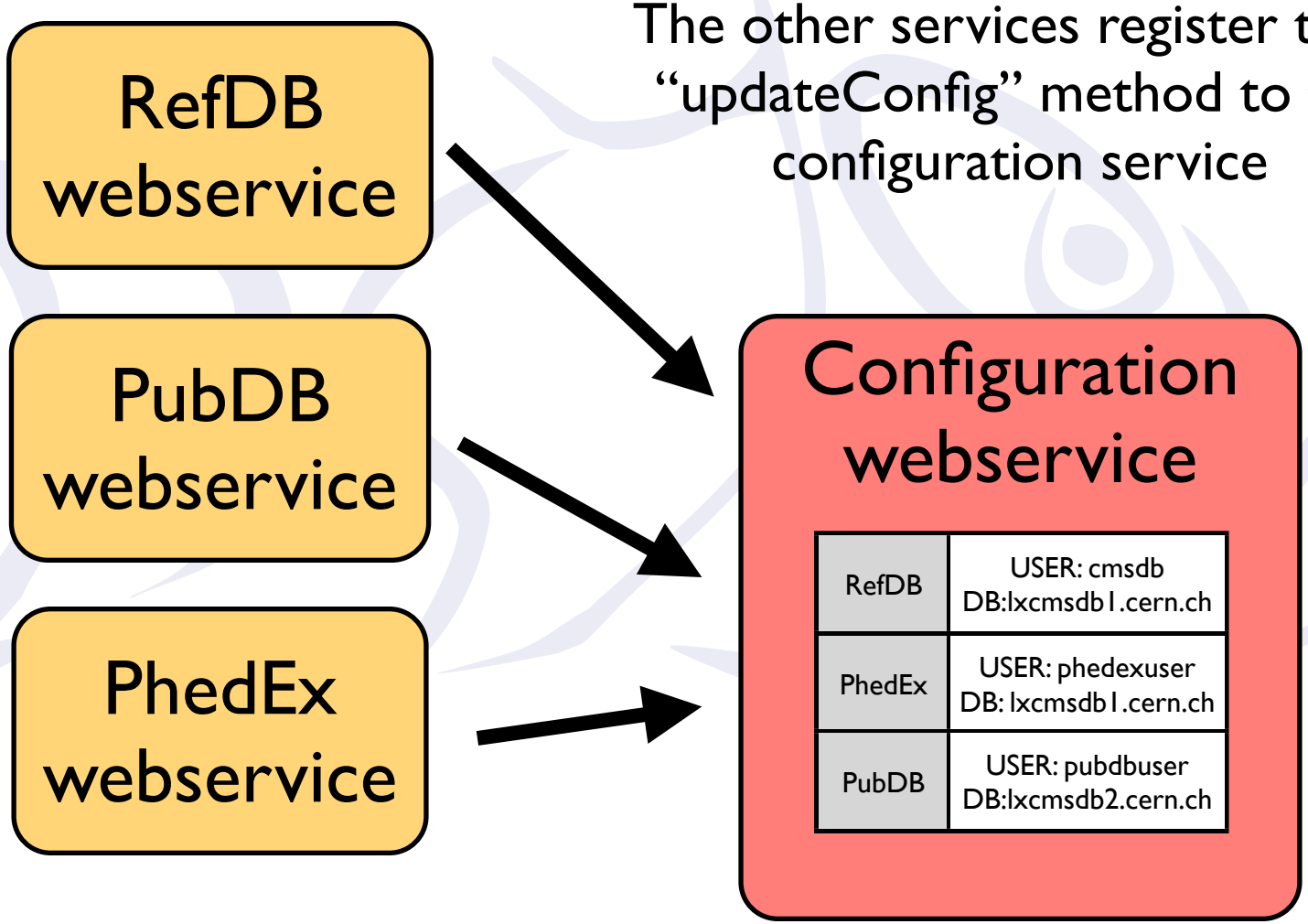


Configuration Service



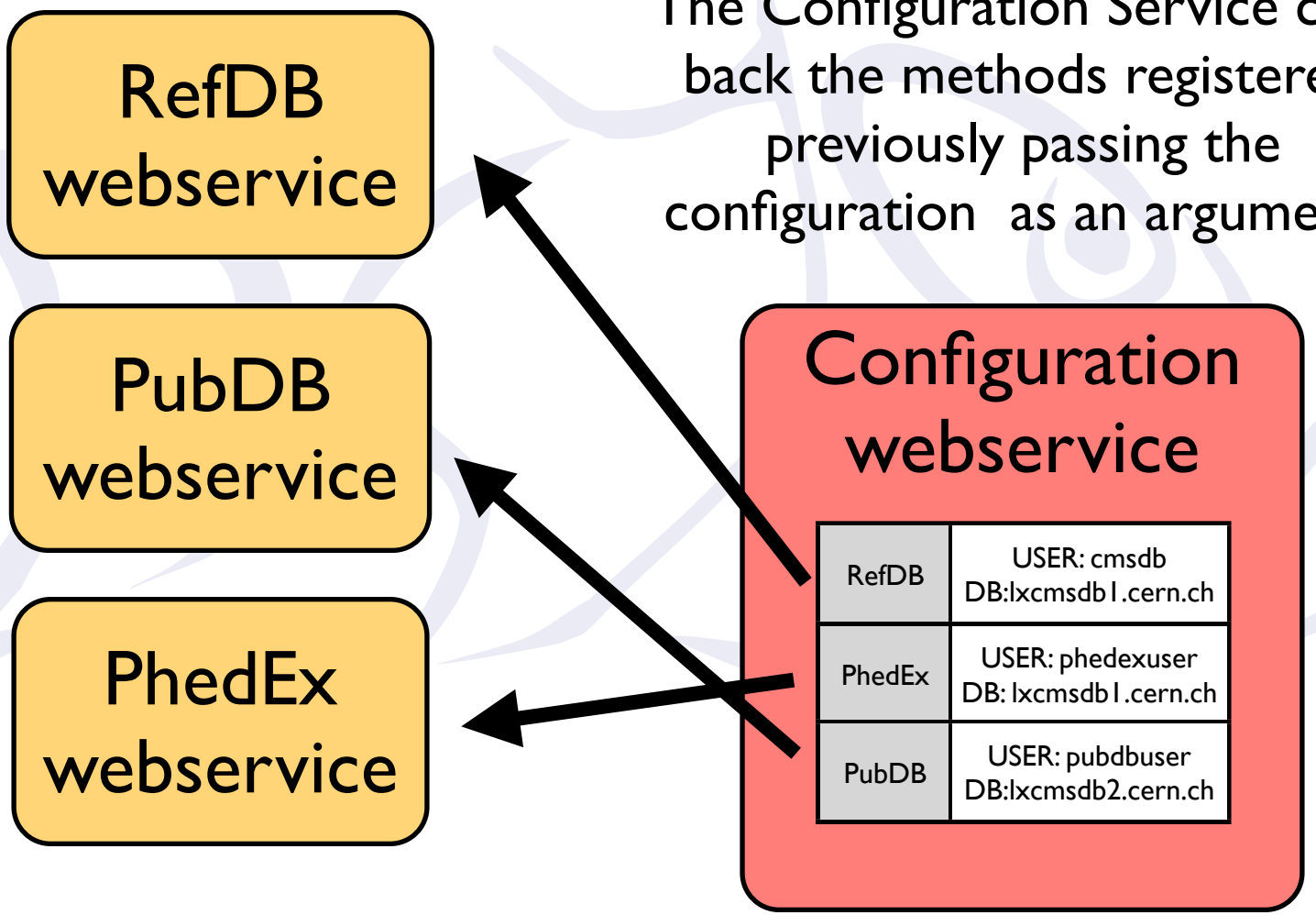
Configuration Service

The other services register their "updateConfig" method to the configuration service



Configuration Service

The Configuration Service calls back the methods registered previously passing the configuration as an argument.



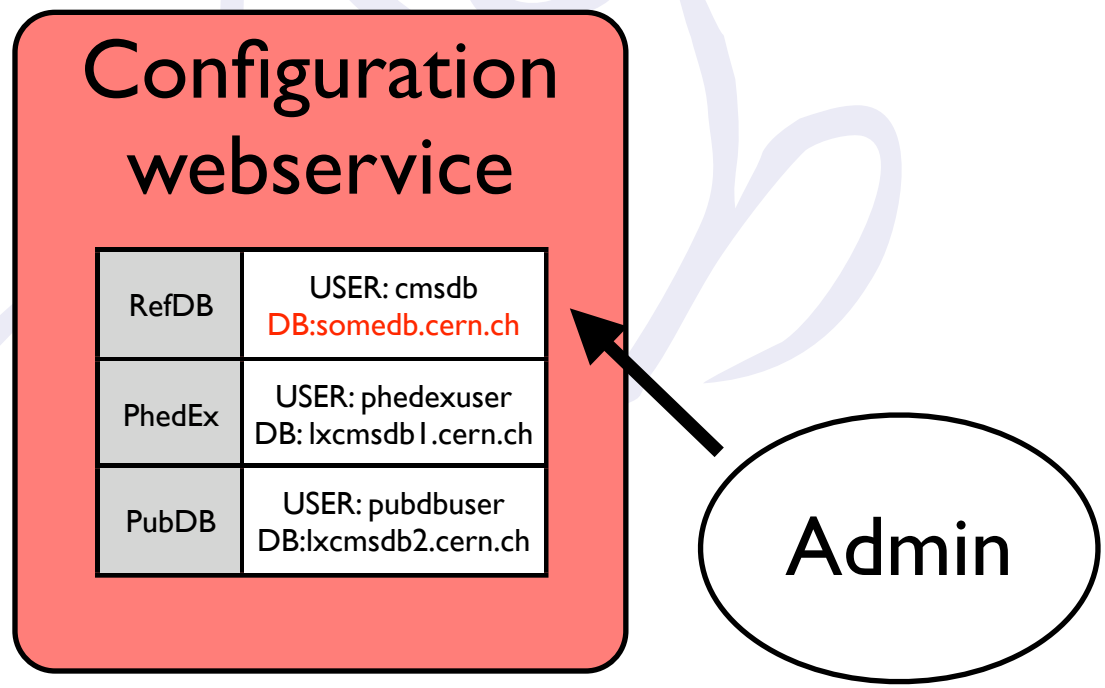
Configuration Service

RefDB webservice

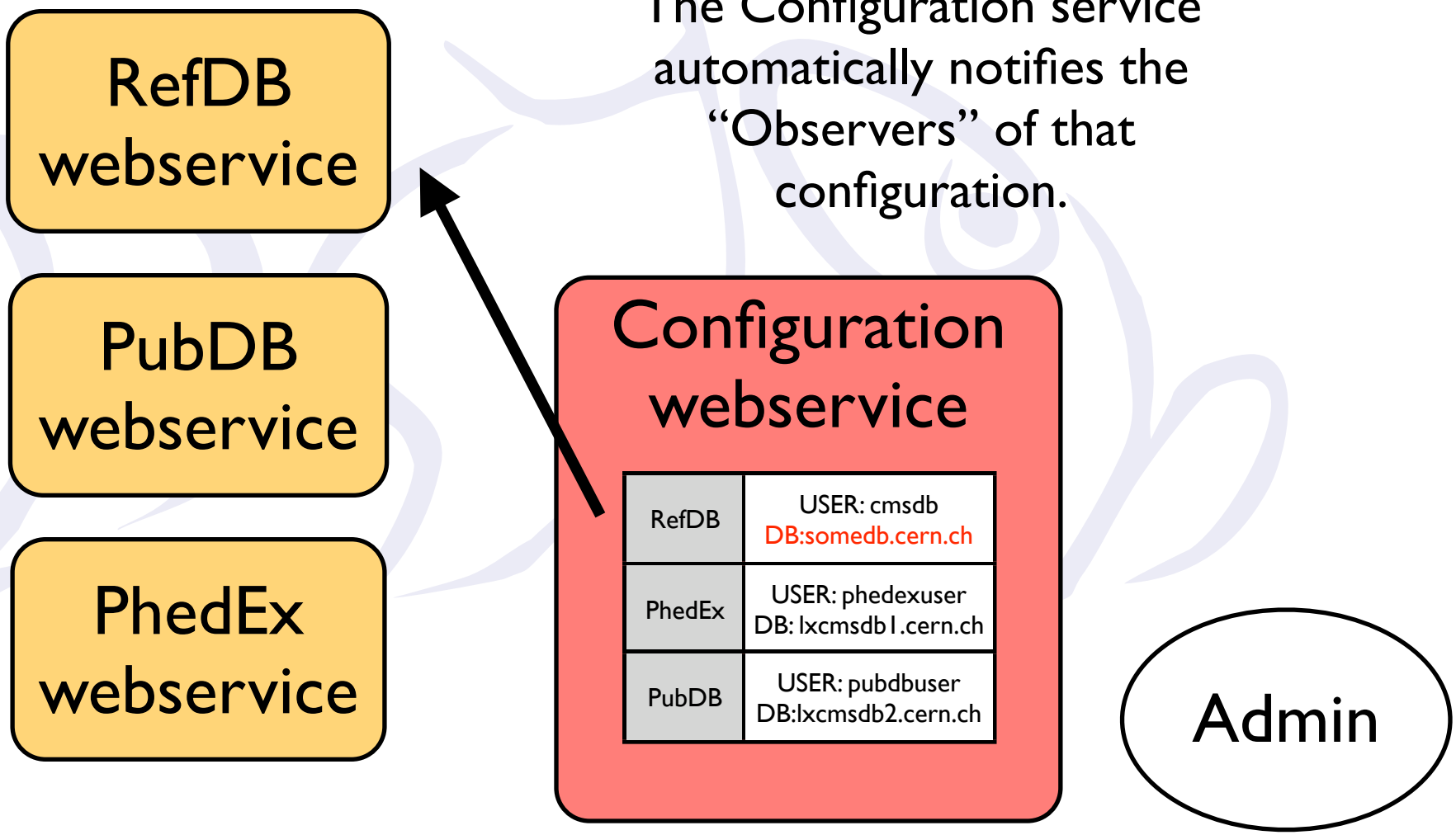
PubDB webservice

PhedEx webservice

The admin changes some on the Configurations



Configuration Service



References

- http://cmsdoc.cern.ch/cms/aprom/physh/

Thanks to:

- Vipin Bhatnagar
- Conrad Steenberg
- Leonardo Sala
- All the people who provided feedback

Questions?