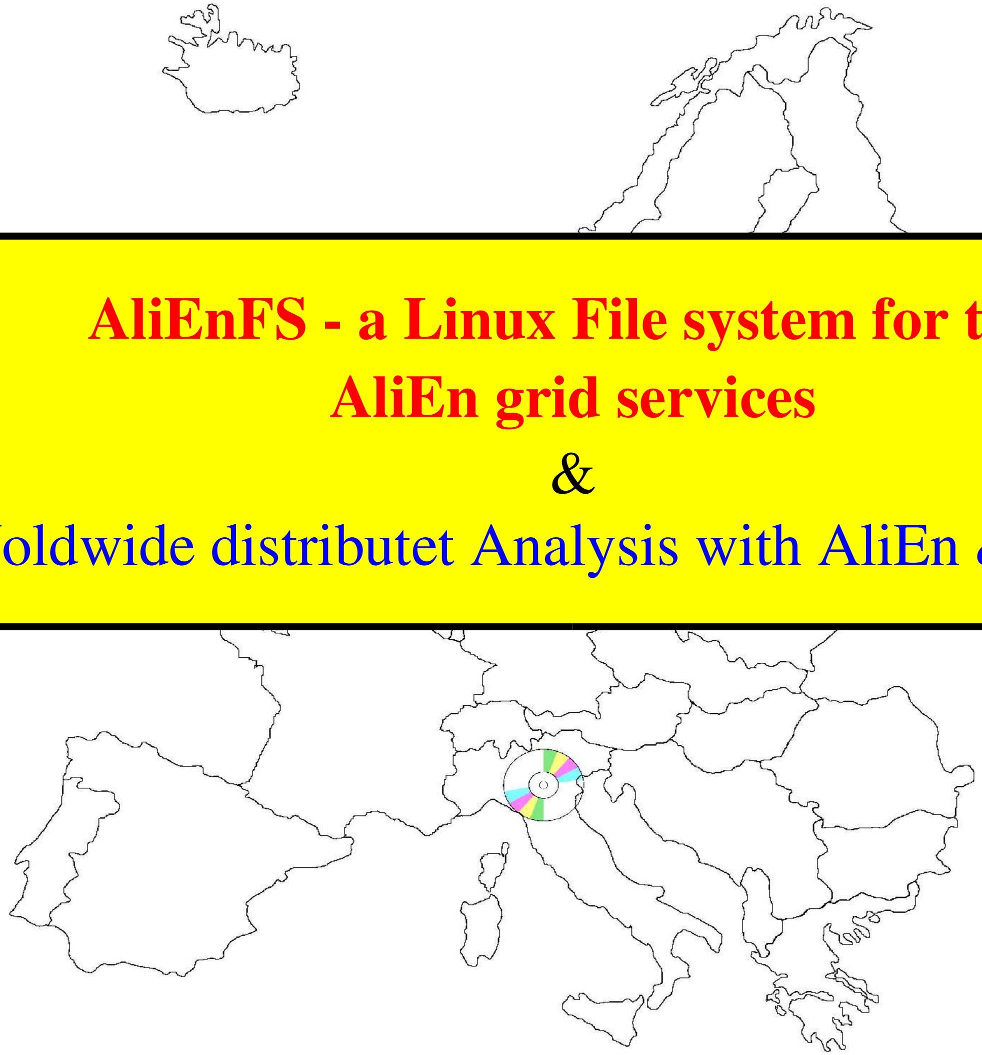
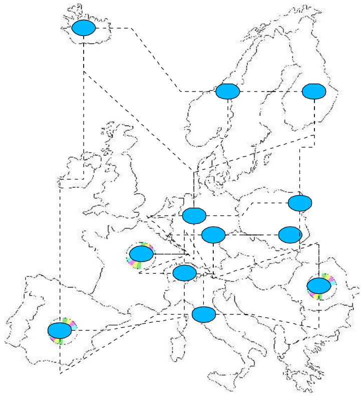
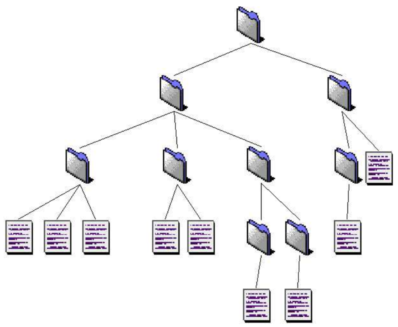


AliEnFS - a Linux File system for the AliEn grid services

&

Woldwide distributet Analysis with AliEn & ROOT





P.Buncic, A.Peters, P.Saiz for the ALICE Collaboration

Overview

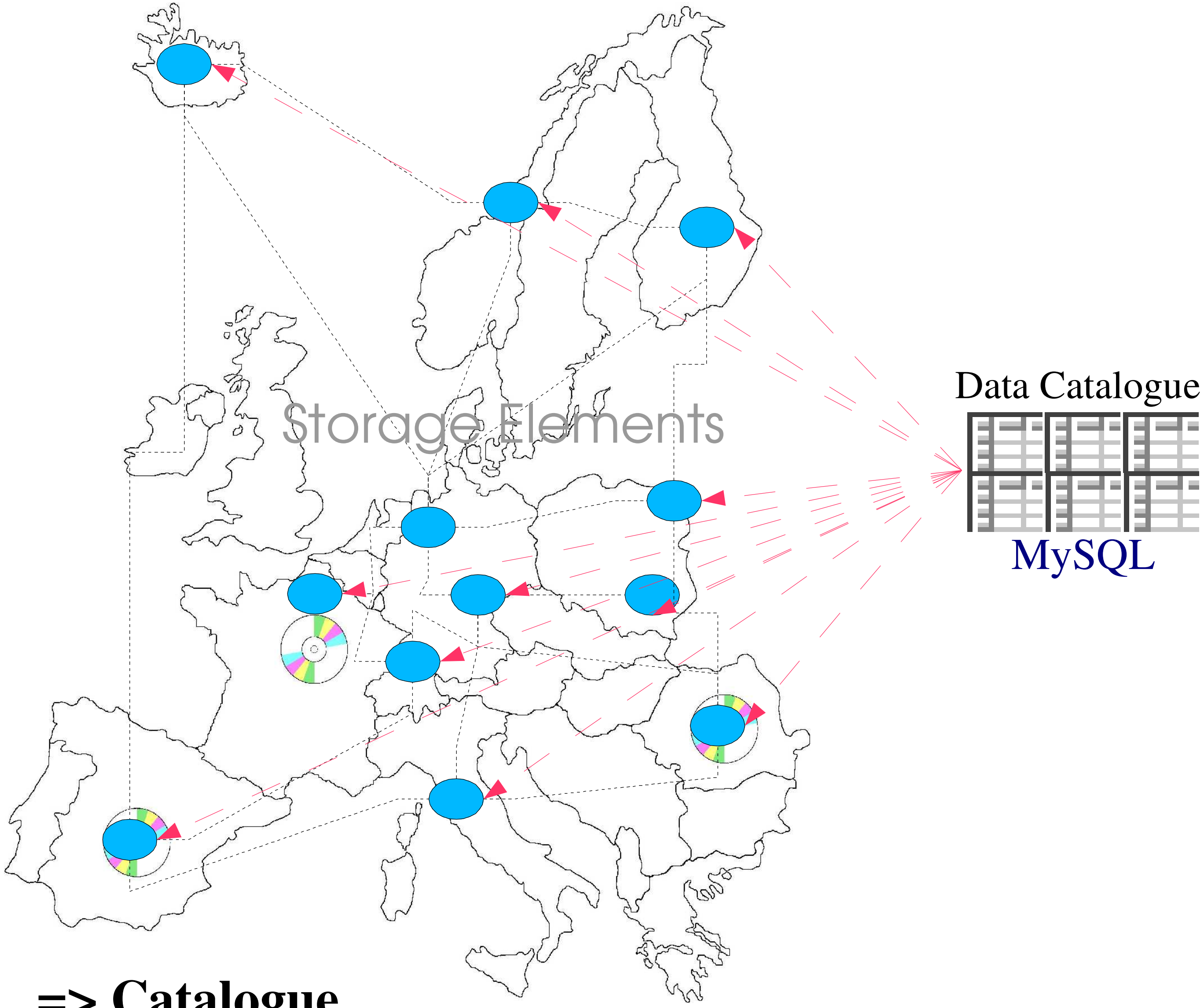
I A Linux FS for the AliEn GRID services

- **Application File Access** in AliEn @GRID
- **Implementation** of the AliEn Global Grid File System
  - VFS – LUFS - AliEnFS Modul

II Worldwide distributed Analysis with AliEn + ROOT

- ROOT goes AliEn @GRID
  - new ROOT classes for parallel Grid Analysis@AliEn @GRID on distributet data sets

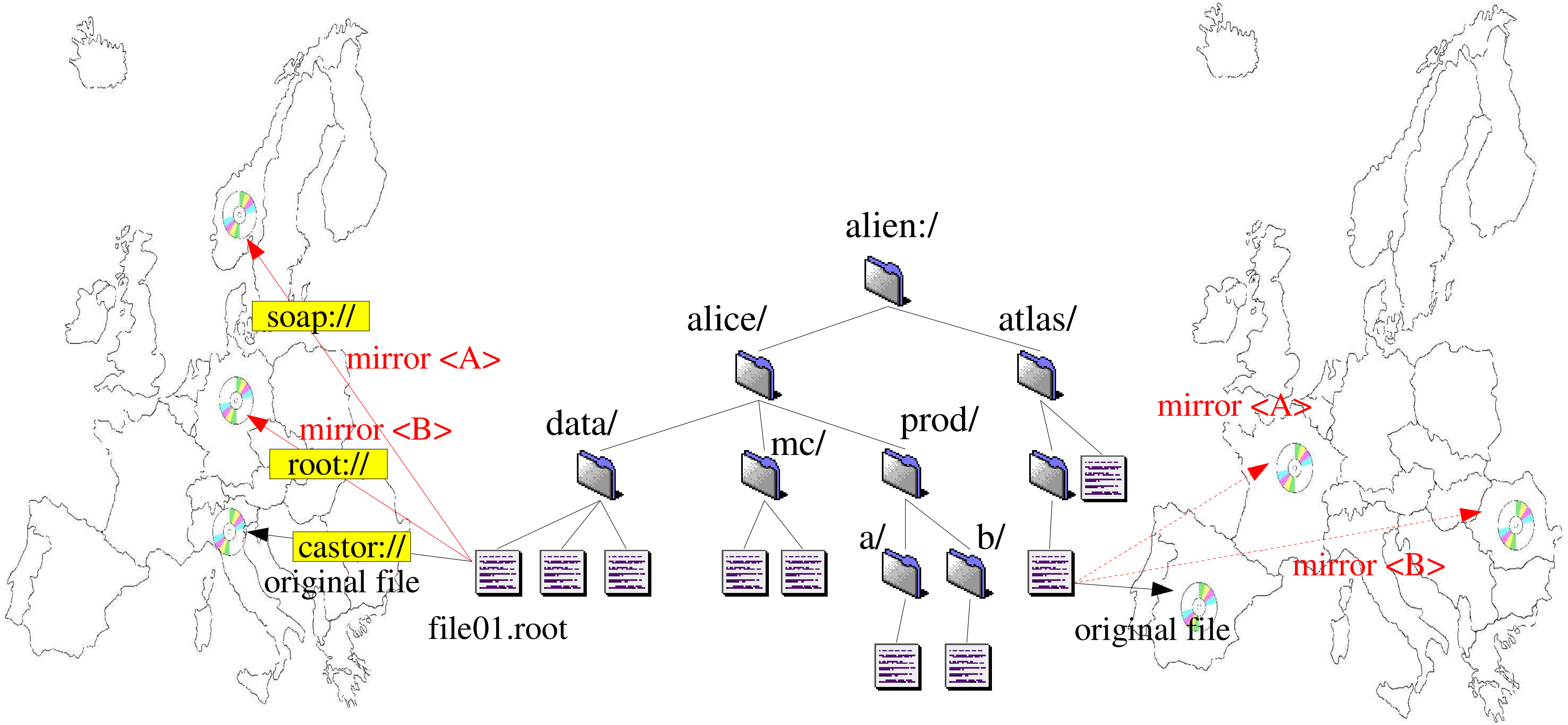
The AliEn File System



needs

- **Browsing** **=> Catalogue**
- **Access** **=> Services / APIs**

AliEn Catalogue Hierarchy

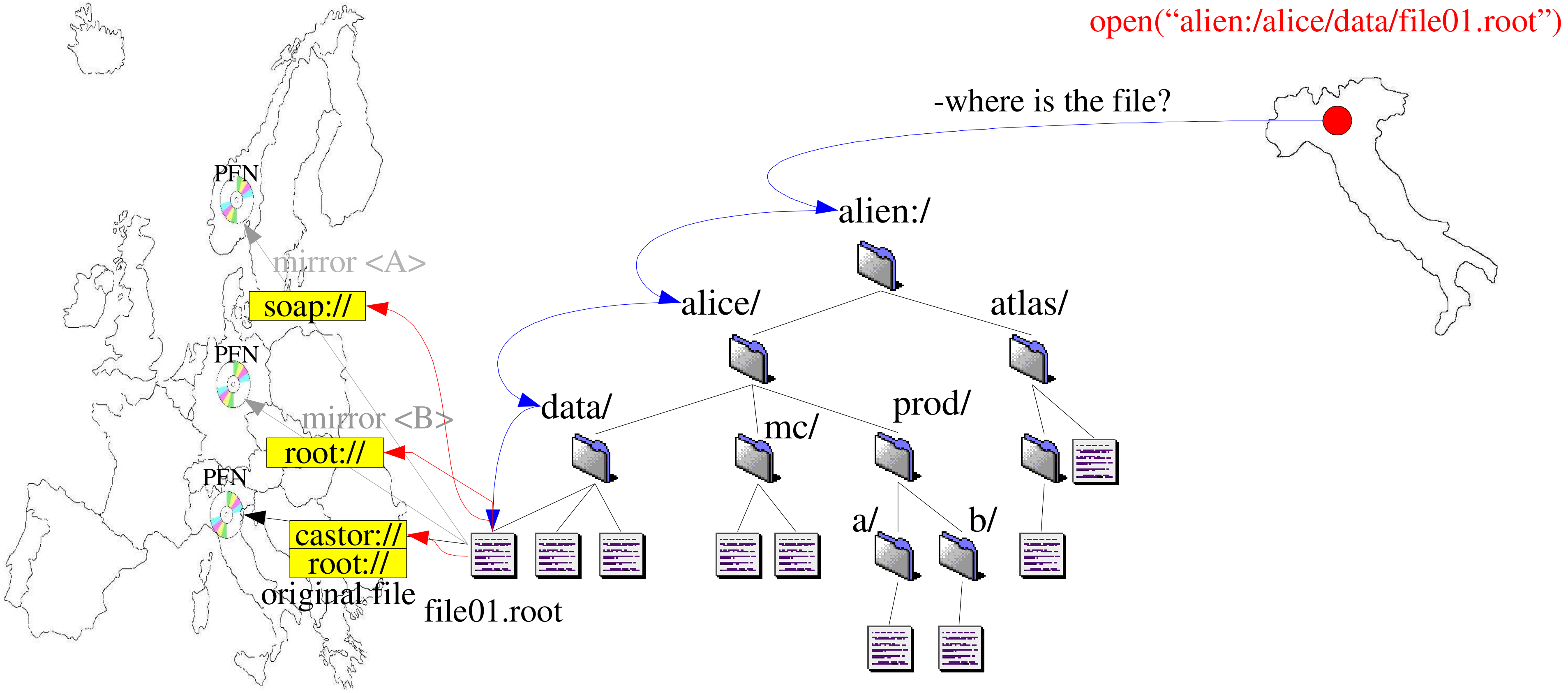


-Catalogue is based on MySQL databases (tree structure)

-Folders are linked database tables

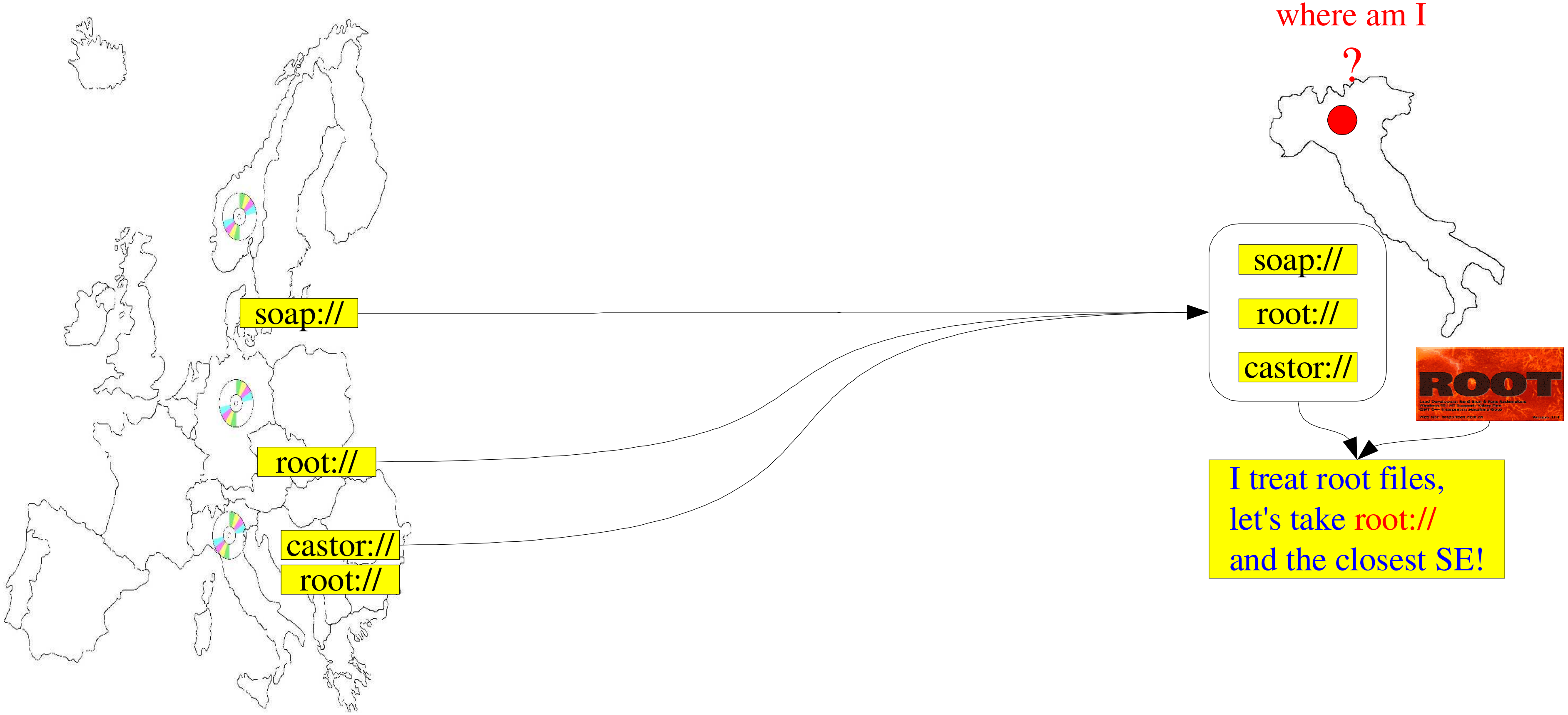
-Files are table entries with location pointers

(offsite)Application File Access



AliEn Application File Access

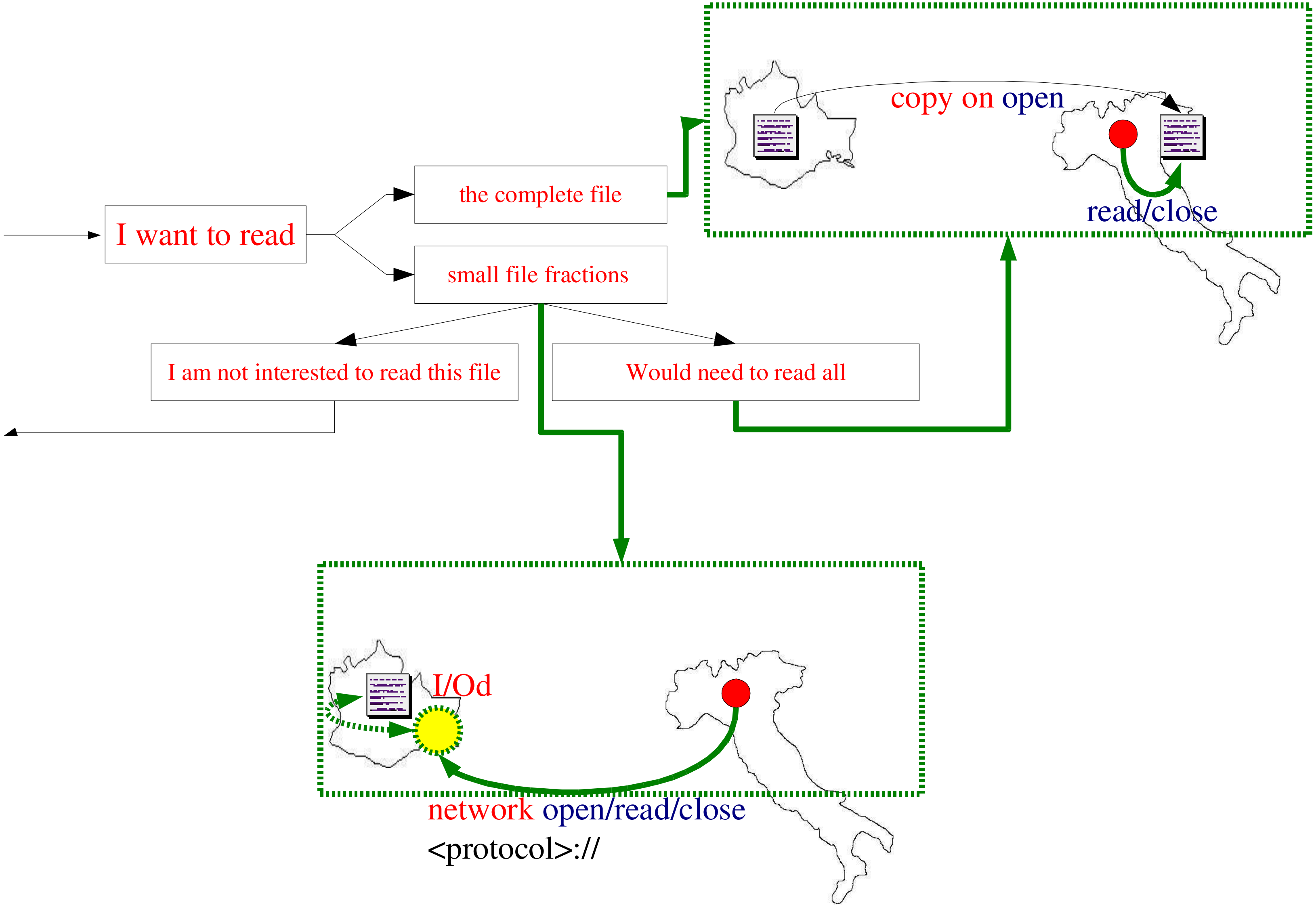
which protocol do I know,
where am I
?



I treat root files,
let's take root://
and the closest SE!

AliEn Application File Access

READ



I want to read

the complete file

small file fractions

I am not interested to read this file

Would need to read all

copy on open

read/close

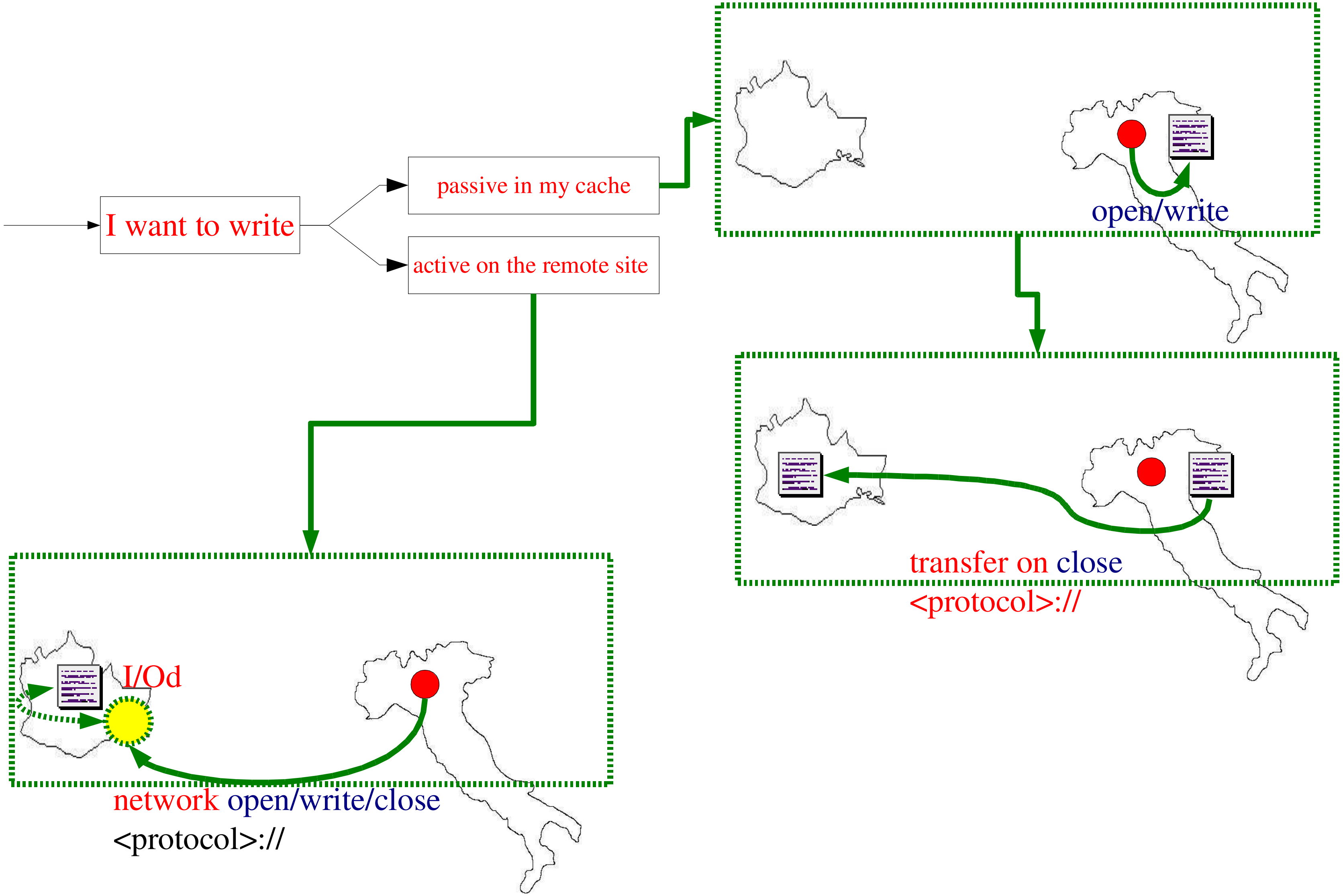
I/Od

network open/read/close

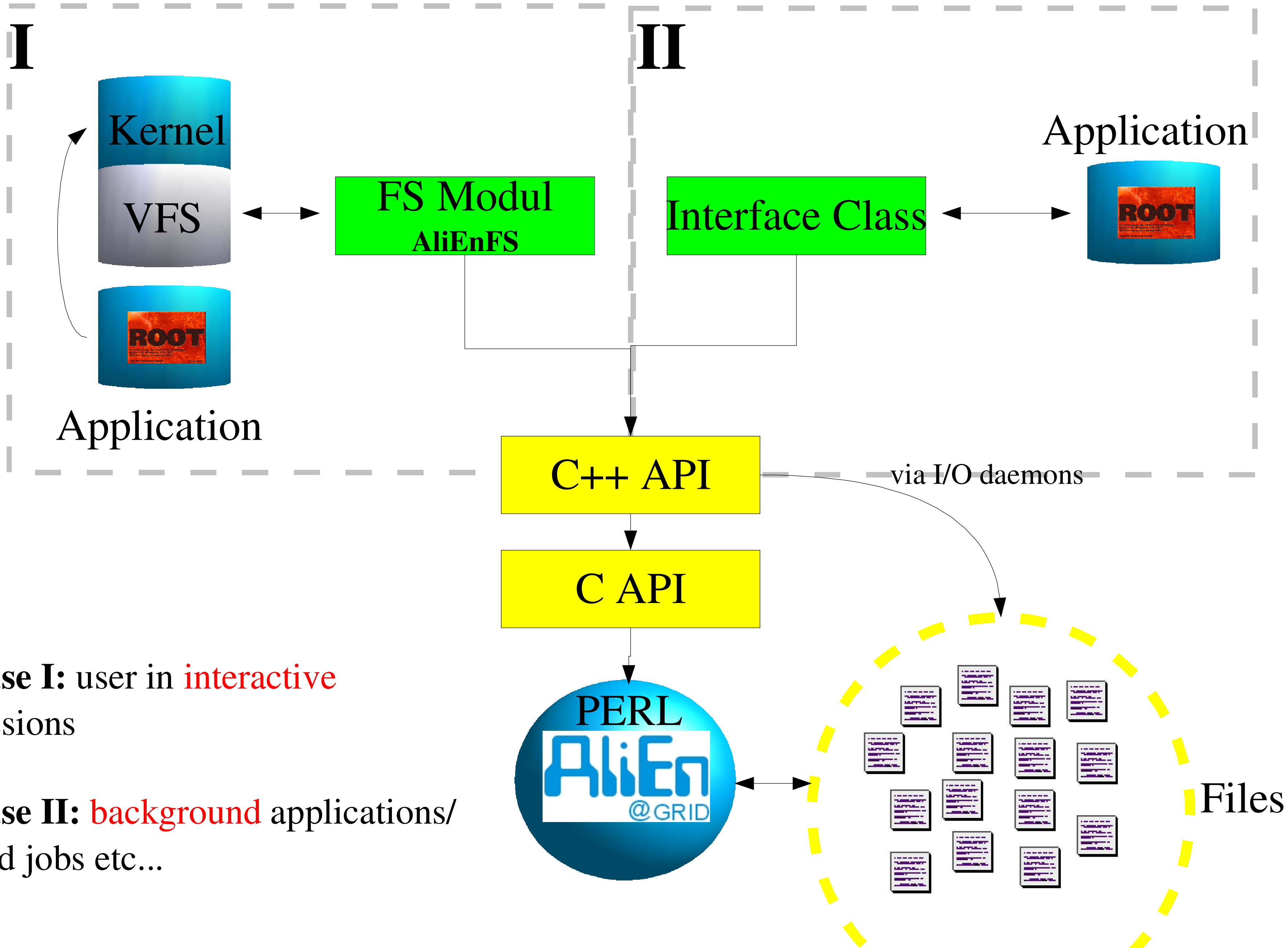
<protocol>://

AliEn Application File Access

WRITE



From Files to a File System



Case I: user in interactive sessions

Case II: background applications/ grid jobs etc...

Implementation of AliEnFS

AliEnFS is written as a module for **LUFS** Linux Userland File System
Open Source Project @ **http://lufs.sourceforge.net/**

-

LUFS

-LUFS runs a kernel module, which delegates VFS calls to various FS daemons, which run in user space. Supports directory caching.

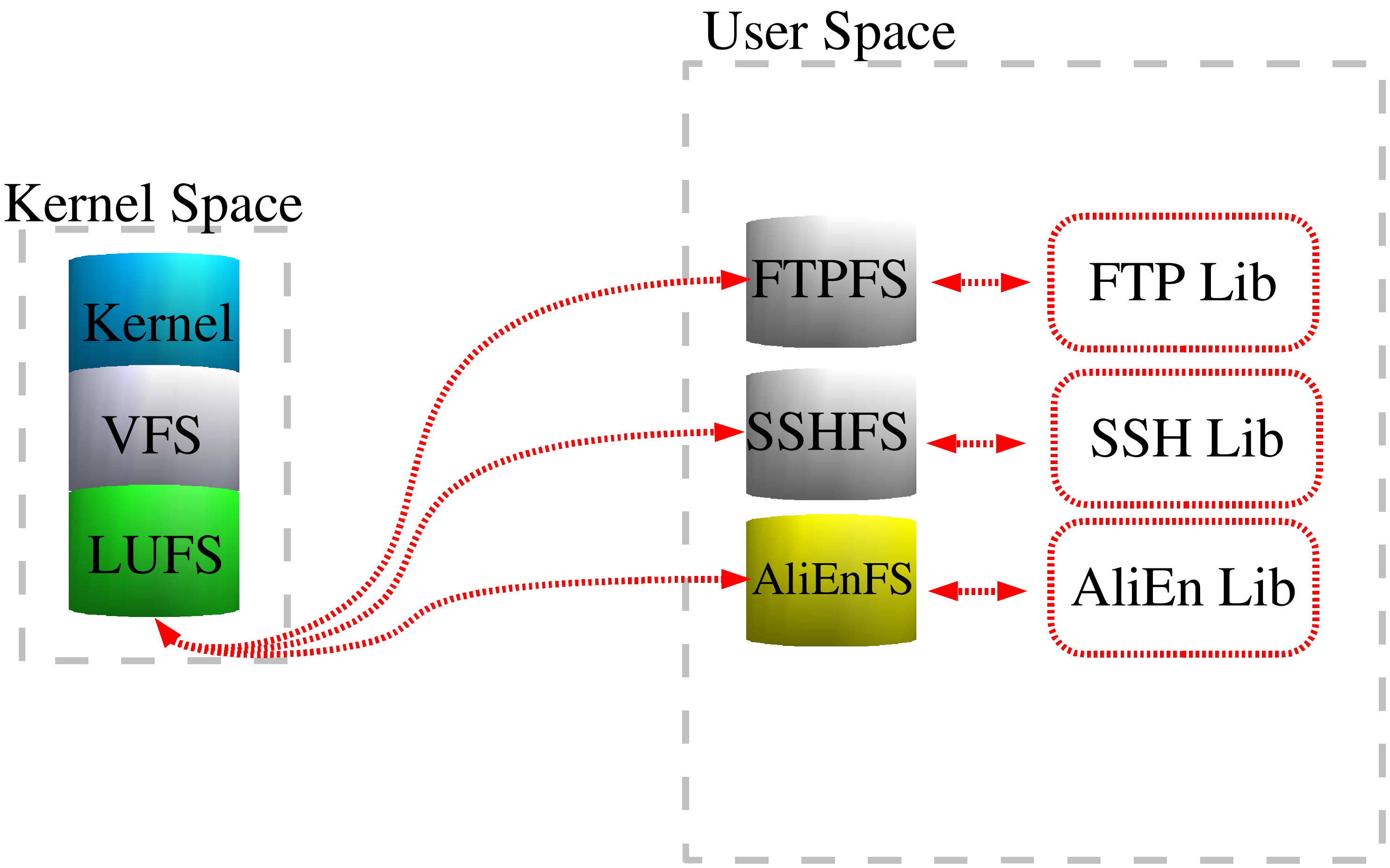
-user space daemons allow easy use of existing cryptographic libraries f.e. for OpenSSH **and** the AliEn API!

-communication via kernel module + FS daemons is done via UNIX domain sockets

Some existing modules: **localfs, ftpfs, sshfs**

f.e. to build your own FS, compile localfs with a redefinition of POSIX commands =>f.e. **rfiofs**

LUFS & AliEnFS IPC



Kernel Space

Kernel

VFS

LUFS

User Space

FTPFS

SSHFS

AliEnFS

FTP Lib

SSH Lib

AliEn Lib

AliEnFS/Mount Authentication

mount has to be executed by each user for authentication reasons!

AliEnFS

```
> lufsmount alienfs://<user>@  /home/user/alien
```

- on **mount**, authentication to the catalogue is performed – prefereable with the installed user specific SSL key pair.

- several *I/O threads* handle the FS operations
    - AliEn connections are shared => AliEn C++ API is thread safe!

- reliability depends only on the API implementation (handling of connection errors, I/O errors etc.)

- AliEnFS uses prefereably active access methods with I/O daemons (to work with applications like the KDE file browser which open each file in a directory, to understand the type)

Status of AliEnFS

working:

- browsing, user+group translation ✓
- location modus to show orginal location as file links ✓
- generic POSIX commands open/read/write/sync/close implemented to allow usual shell commands:
  cp,rm,du, find, ls ✓

in preparation:

- automount
- adding files via <ln -s> to the catalogue
- displaying tag values as .tag files in the file system

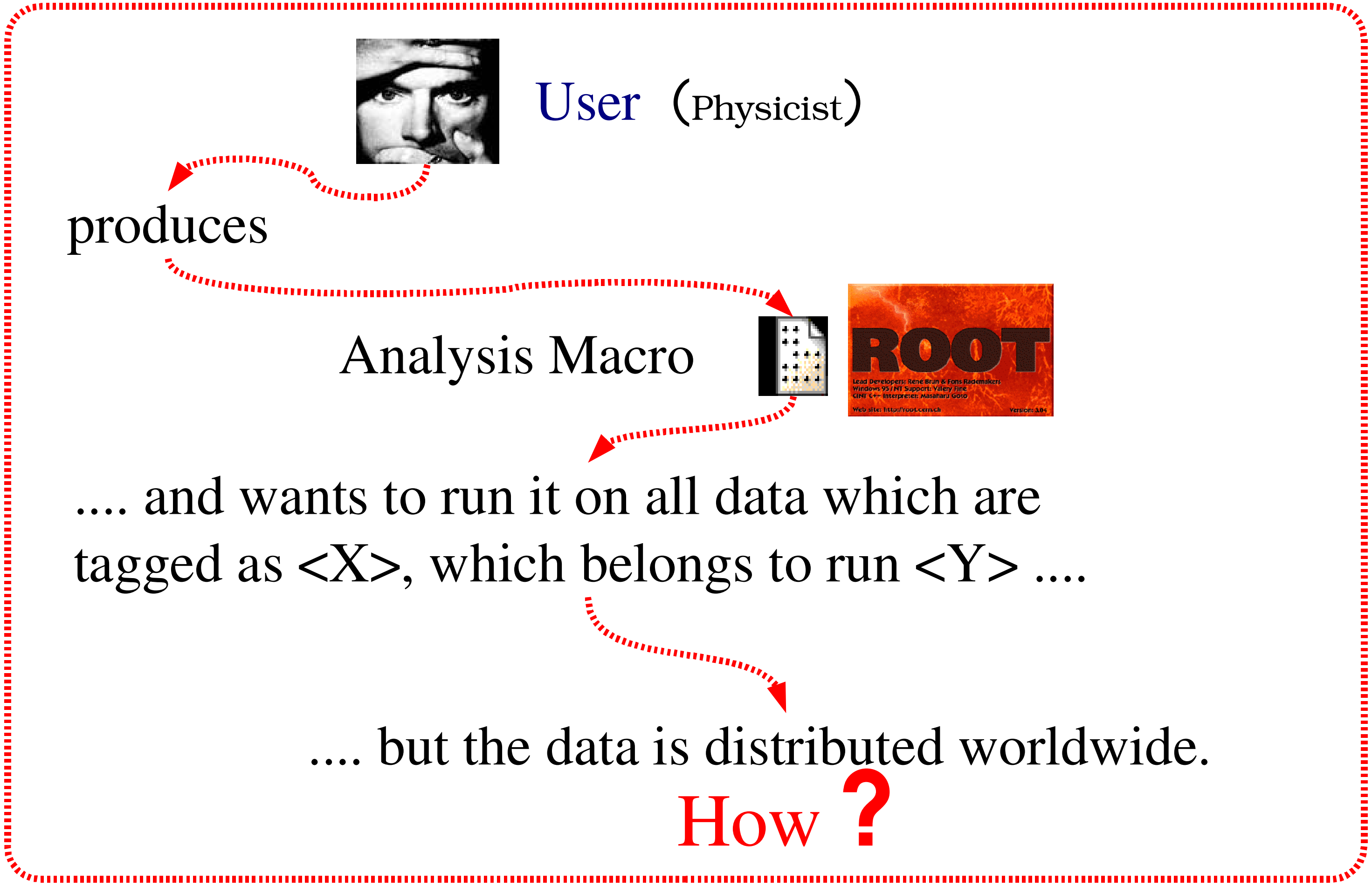
functionality through LUFS/AliEn:

- re-exporting of mounted dirs with SAMBA,SSHFS... ✓
- files are cached locally by AliEn ✓

Worldwide distributed Analysis

AliEn + ROOT

The task:


ROOT
Lead Developers: René Brun & Fons Rademakers
Windows 95 / NT Support: Valery Fine
CINT C++ Interpreter: Masaharu Goto
Web site: http://root.cern.ch
Version: 3.04

User (Physicist)

produces

Analysis Macro

.... and wants to run it on all data which are tagged as <X>, which belongs to run <Y> ....

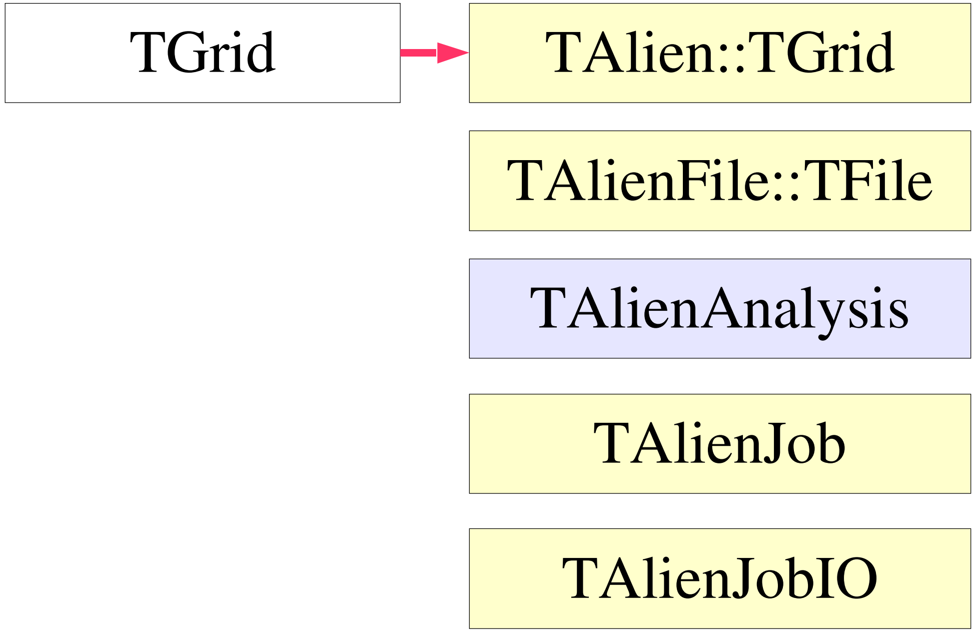
.... but the data is distributed worldwide.

How ?

The task can be done already now with ROOT + AliEN ....

Worldwide distributed Analysis

AliEn + ROOT Classes: Class Tree



TGrid →	TAlien::TGrid	Authentication + Catalogue Browsing
	TAlienFile::TFile	ROOT File access via AliEn
	TAlienAnalysis	Parallel Grid Analysis Object
	TAlienJob	AliEn Job (belonging to TAlienAnalysis Object)
	TAlienJobIO	Managing File I/O for a specific Ana. Job

The Analysis Object:

TAlienAnalysis

Each Analysis Object is stored with unique names in the user directory
- contains corresponding TAnalysisJob Objects, if Jobs are submitted
- can be reopened anytime from a ROOT session

Worldwide distributed Analysis

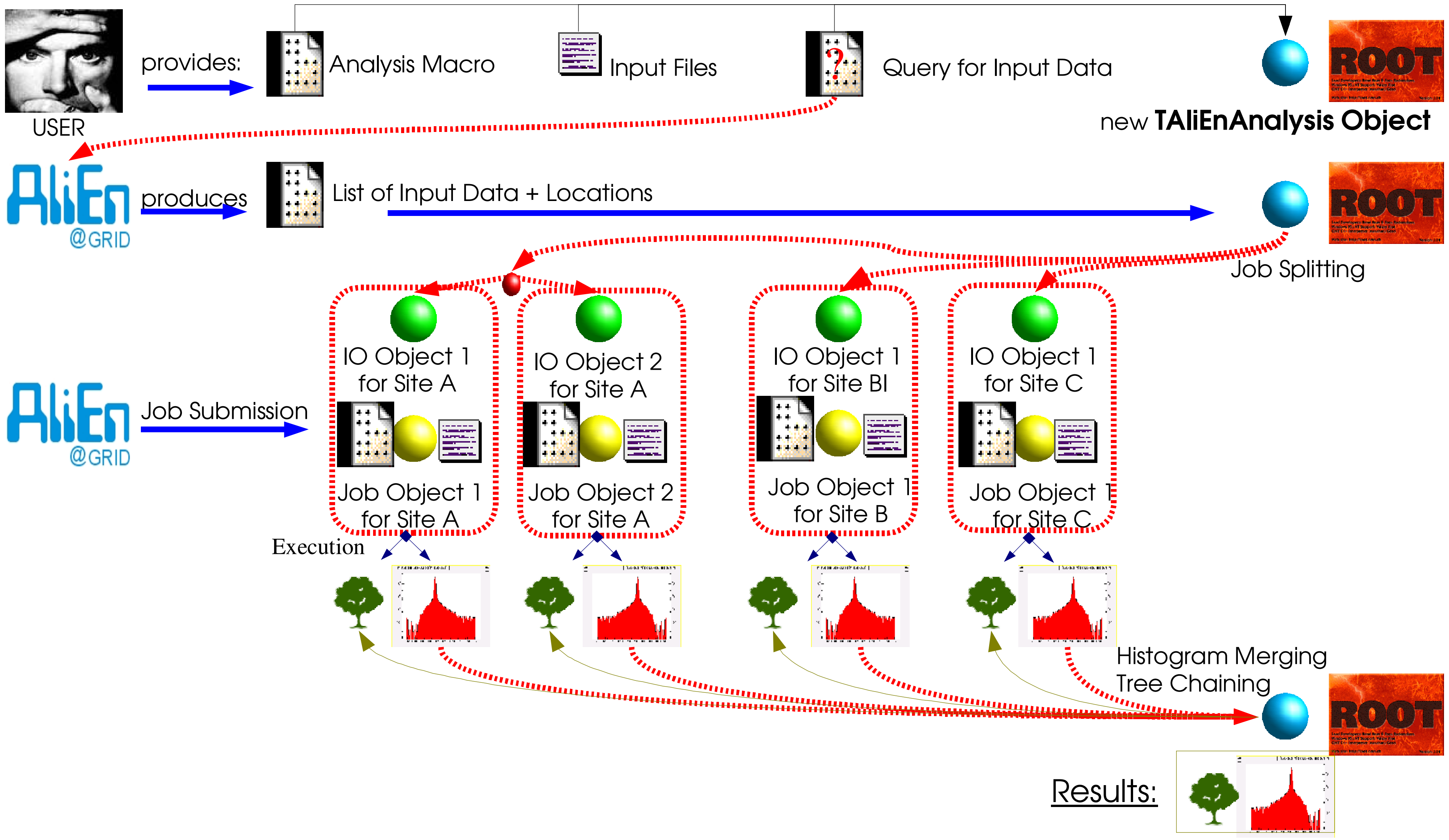
AliEn + ROOT Classes: Example Session

```
// connect + authenticate to the GRID Service alien as “user”
TGrid *alien = TGrid::Connect("alien",”user”,"","");
// create a new analysis Object ( <unique ID>, <title>, #subjobs)
TAlienAnalysis* newanalysis = new TAlienAnalysis("run001","analysis",10);
// set the program, which executes the Analysis Macro/Script
newanalysis->AnalysisScriptExecuter("AliRoot.sh");
newanalysis->AnalysisScript("file:/home/peters/test.C"); // script to execute
newanalysis->RootOutputFileAutoMerge(true); // merge all produced .root files
newanalysis->AnalysisQuery("2002-10/V3.08.Rev.04/00110/%galice.root");
newanalysis->PrepareJobSplitting(); // split the task in subjobs
newanalysis->Submit(); // submit all subjobs to the AliEn queue
newanalysis->GetResults(); // download partial/final results and merge them
newanalysis->DumpJobInfo(); // display job information
```

Andreas J. Peters CERN/Geneva @ CHEP 2003 – San Diego

Worldwide distributed Analysis

AliEn + ROOT: Analysis Session Flow Diagram



USER

provides:

Analysis Macro

Input Files

Query for Input Data

new **TAliEnAnalysis Object**

produces

List of Input Data + Locations

Job Splitting

Job Submission

IO Object 1 for Site A

Job Object 1 for Site A

IO Object 2 for Site A

Job Object 2 for Site A

IO Object 1 for Site BI

Job Object 1 for Site B

IO Object 1 for Site C

Job Object 1 for Site C

Execution

Histogram Merging
Tree Chaining

Results:

Summary

•C++ application interface enables active/partial file access in the AliEn GRID environment, which is essential for qnalysis needs

•AliEnFS module allows to mount the AliEn Catalogue + MSSs as a "normal" Linux file system for interactive work

•Analysis of worldwide distributed datasets can be done with
•ROOT + AliEn specific extension classes

Outlook

•AliEnFS + AliEn Grid Analysis with ROOT will be tested by Alice collaborators if they satisfy the user requirements.

•the analysis framework can also be used for large scale MC productions and to use PROOF on AliEn.