The Alpes Lasers use case

ALDIRAC

May 25, 2014

Outline

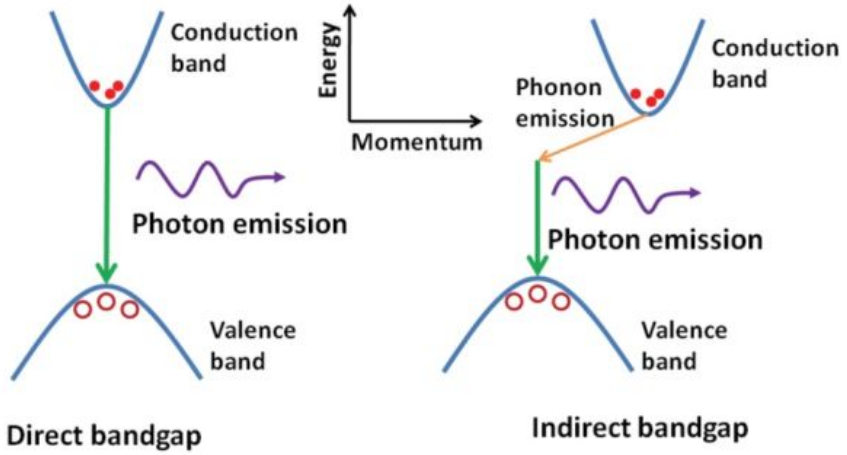
- Context
- Use case
- Work model
- Application workflow
- User Interface
- Dedicated system

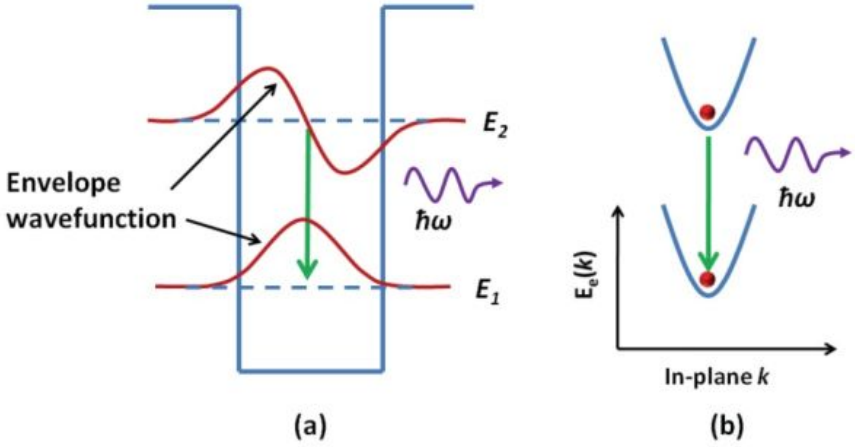
Who is Alpes Lasers?

- Specialized in Quantum Cascade Lasers (QCL) in mid infrared
- In Neuchâtel, Switzerland
- 20 people
- World leader: many experts of the field work here
- Many different types of clients, from Uni to private sector
- Mostly research and development activities


ALPES LASERS
KLA-Tencor
Accelerating Yield
EADS
CASCADE TECHNOLOGIES
DAYLIGHT SOLUTIONS
EPFL
BOSTON UNIVERSITY
ETH
PUC RIO
NORTHWESTERN UNIVERSITY
IPM
Institut Physikalische Messtechnik
LGR
TECHNISCHE UNIVERSITÄT DARMSTADT
INEM
TU WIEN
Georgia Institute of Technology
biotix
SCUOLA NORMALE SUPERIORE
NASA
PSI
THALES
Pacific Northwest National Laboratory
...delivering breakthrough science and technology
Areté Associates
Tucson Operations
TELOPS
Southwest Sciences
Ball
PHILIPS
ESL DEFENCE
LUND UNIVERSITY

What are QCLs?



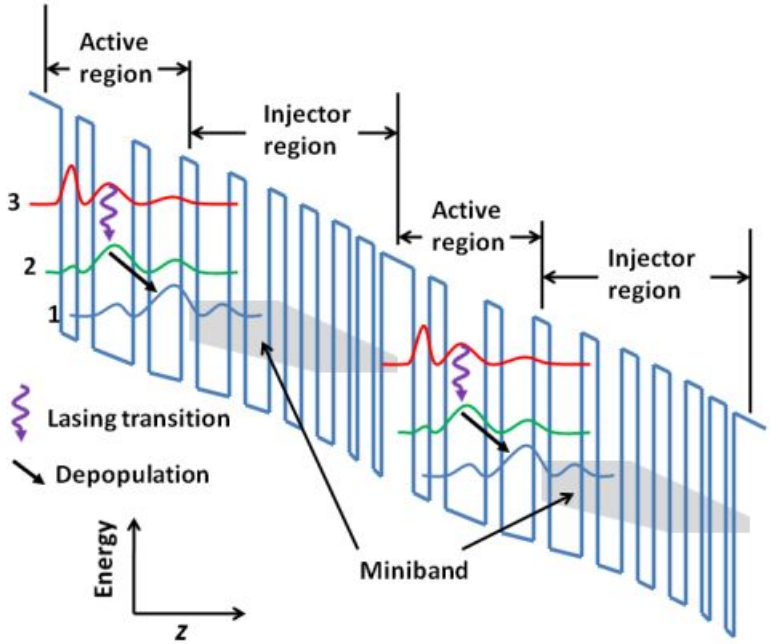


Interband

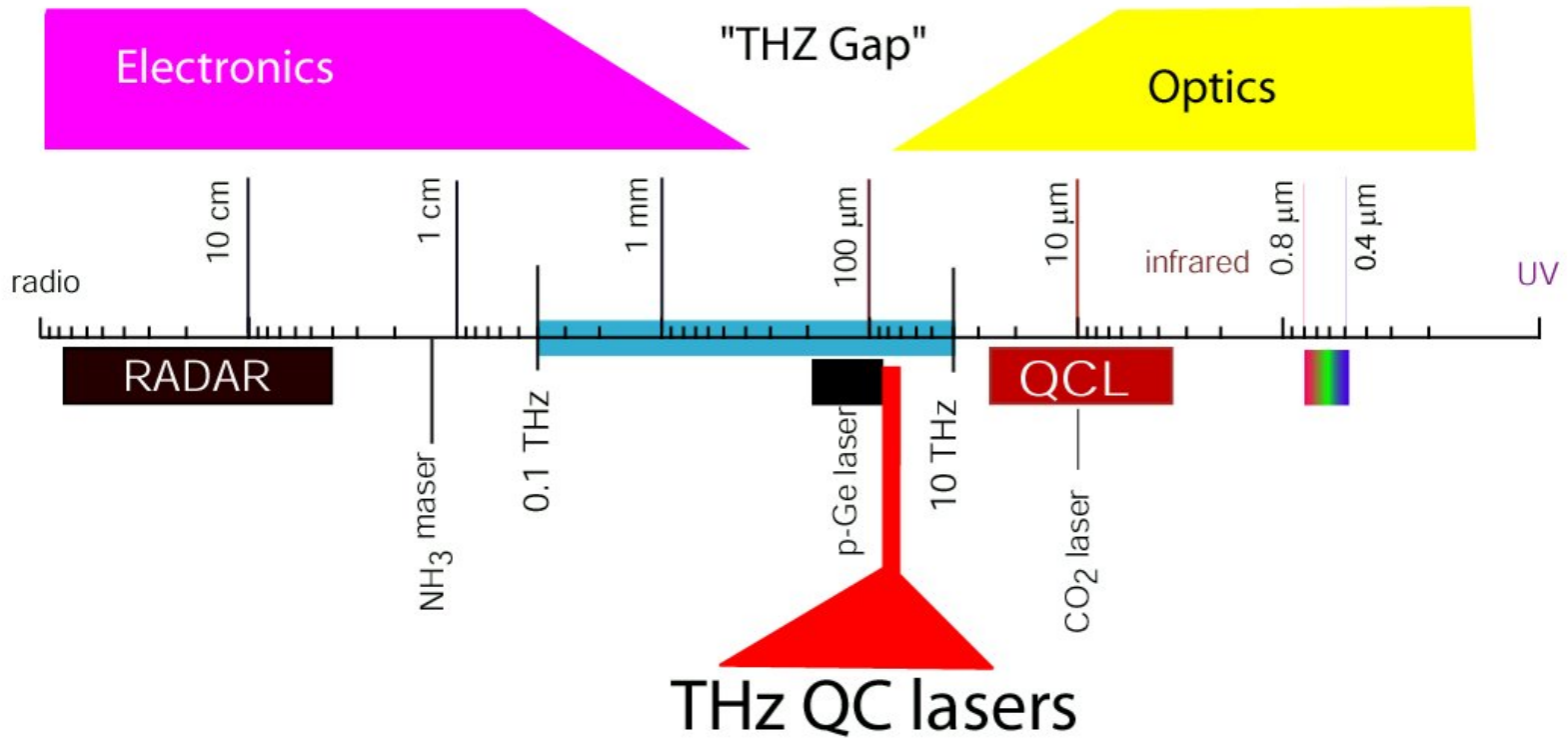
Intersubband (QCLs)

One electron yields $>$ 1 photon

Possible to tune the wavelength, operate at room temperature!



The optical domains



Obtaining THz emission usually done using FEL

What for?

Spectroscopy:

- Trace gas detection
- Remote sensing
- Environmental monitoring
- Quality analysis

TRACE GAS	cm^{-1}	*1 s RMS ppb 76 m path*	*LoD ppb 100 s*
NH_3	967	0.2	0.06
C_2H_4	960	1	0.5
O_3	1050	1.5	0.6
CH_4	1270	1	0.4
N_2O	1270	0.4	0.2
H_2O_2	1267	3	1
SO_2	1370	1	0.5
NO_2	1600	0.2	0.1
HONO	1700	0.6	0.3
HNO_3	1723	0.6	0.3
HCHO	1765	0.3	0.15
HCOOH	1765	0.3	0.15
NO	1900	0.6	0.3
OCS	2071	0.06	0.03
CO	2190	0.4	0.2
N_2O	2240	0.2	0.1
$^{13}CO_2/\ ^{12}CO_2$	2311	0.5 ‰	0.1 ‰

RED: OBSERVED
BLACK: ANTICIPATED

Same material system gives access to wide range of wavelengths, small device size compared to traditional devices

Why DIRAC? (1)

- QCL tech. is recent, only 20 years old
- No widely established tools
- Very hard to build devices, no complete theoretical models to predict a structure's behavior.

Why DIRAC? (2)

- Some software exist (in-house) to predict some of the devices' behavior, need validation against built structures. Intellectual Property is involved.
- Use of Open Source software dominates
- Do not want to spend time reinventing the wheel and DIRAC seems complete.
- Can have an expert in house (S. Poss)

It's a reasonnable, attractive solution

Our work model

- Do not have a local CE, use Amazon EC2 with VMDIRAC
- No Storage Element, need to be careful with Output Data
- Deal with the Intellectual Property
- Define and implement dedicated system to interact with the in-house database (PSQL): SimuDB

Amazon EC2

- No contextualization but 2 types of machines pre-installed/configured:

  Small 1 core machine for test jobs

  Large Many core machines: Utility to start as many JobAgents as we want

- EC2 provides 32 core machines for $1.9/hour, could run up to 640 jobs concurrently (20 machines)
- Security group: dedicated ip/ports only

Available resources in our office

- One central DIRAC server hosted on VM
  4 cores, 8GB RAM
- SSHBatchCE configured and used for tests
  (2 hosts, 4 cores)
- 20Mbps download / 2Mbps upload link

Deal with the existing resources

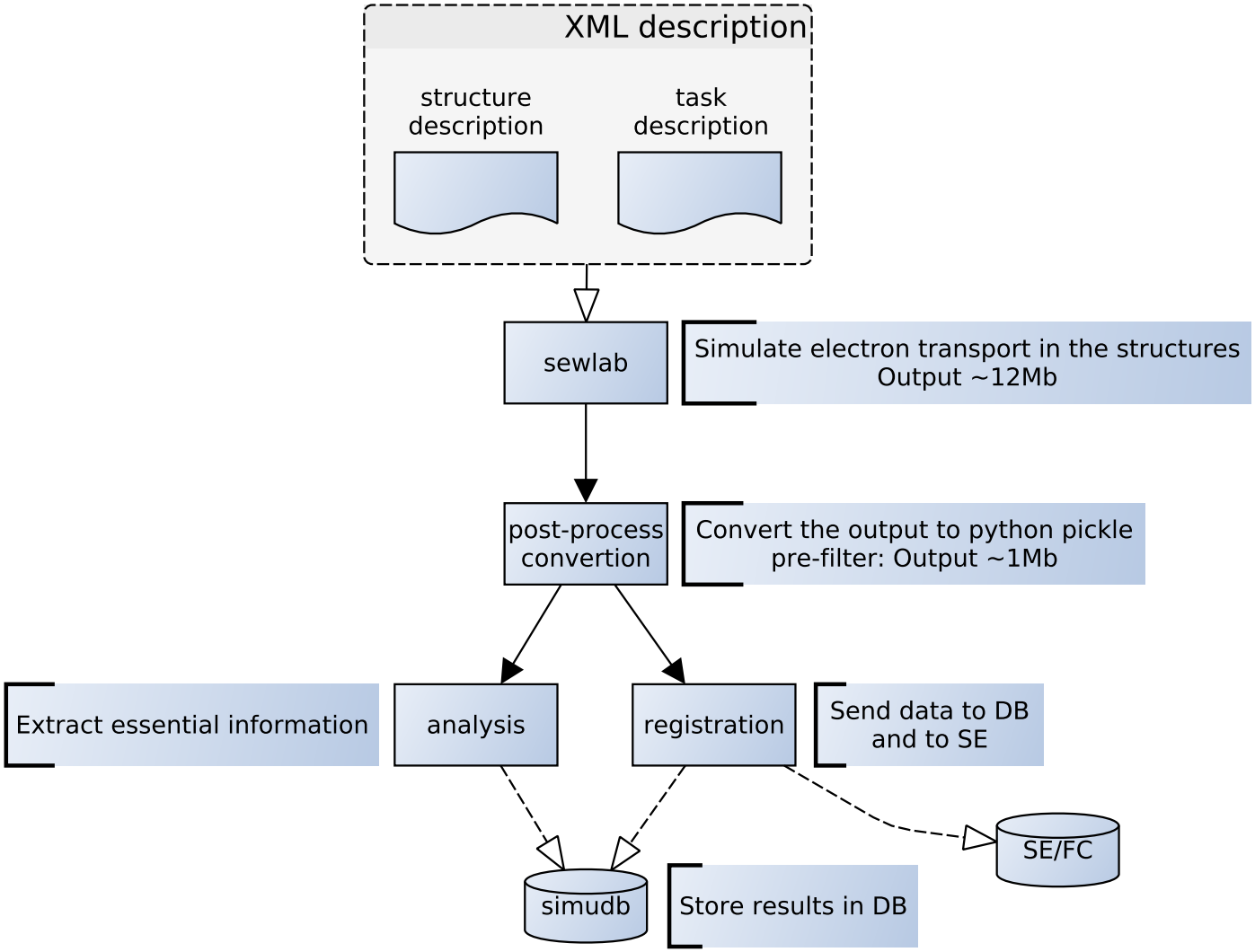
Software Management

Account for IP: strict access control

- Software only hosted in our machines
- Packaging done using our tools, based on python packaging tools (i.e. *pip*)
- Use dependency description (à la ILCDIRAC) to resolve all/only soft. bits needed for a job
- Use `rsync+ssh` to collect only software differences: reduced software transfer footprint

CVMFS could be an alternative, if needed.

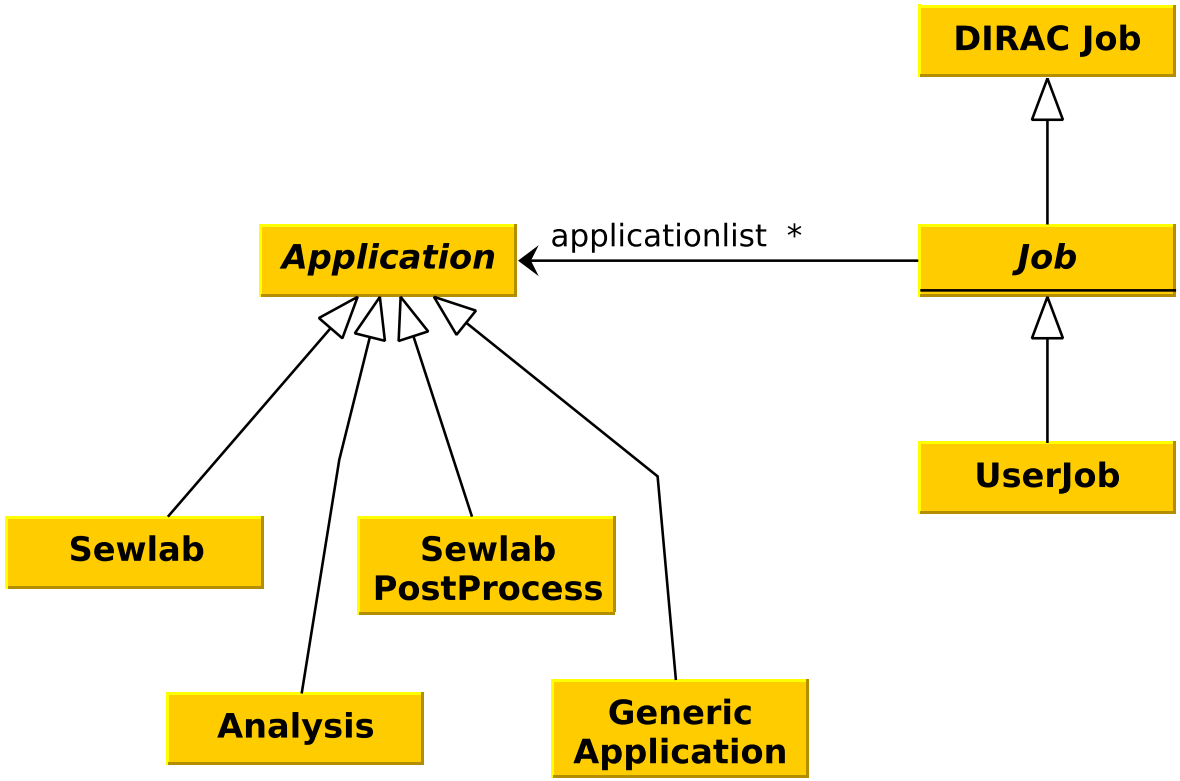
Application workflow



User Interface (1)

Based on ILCDIRAC:

- Uses DIRAC's *Workflow*
- Decoupling of Applications and Jobs



ALPES LASERS

User Interface (2)

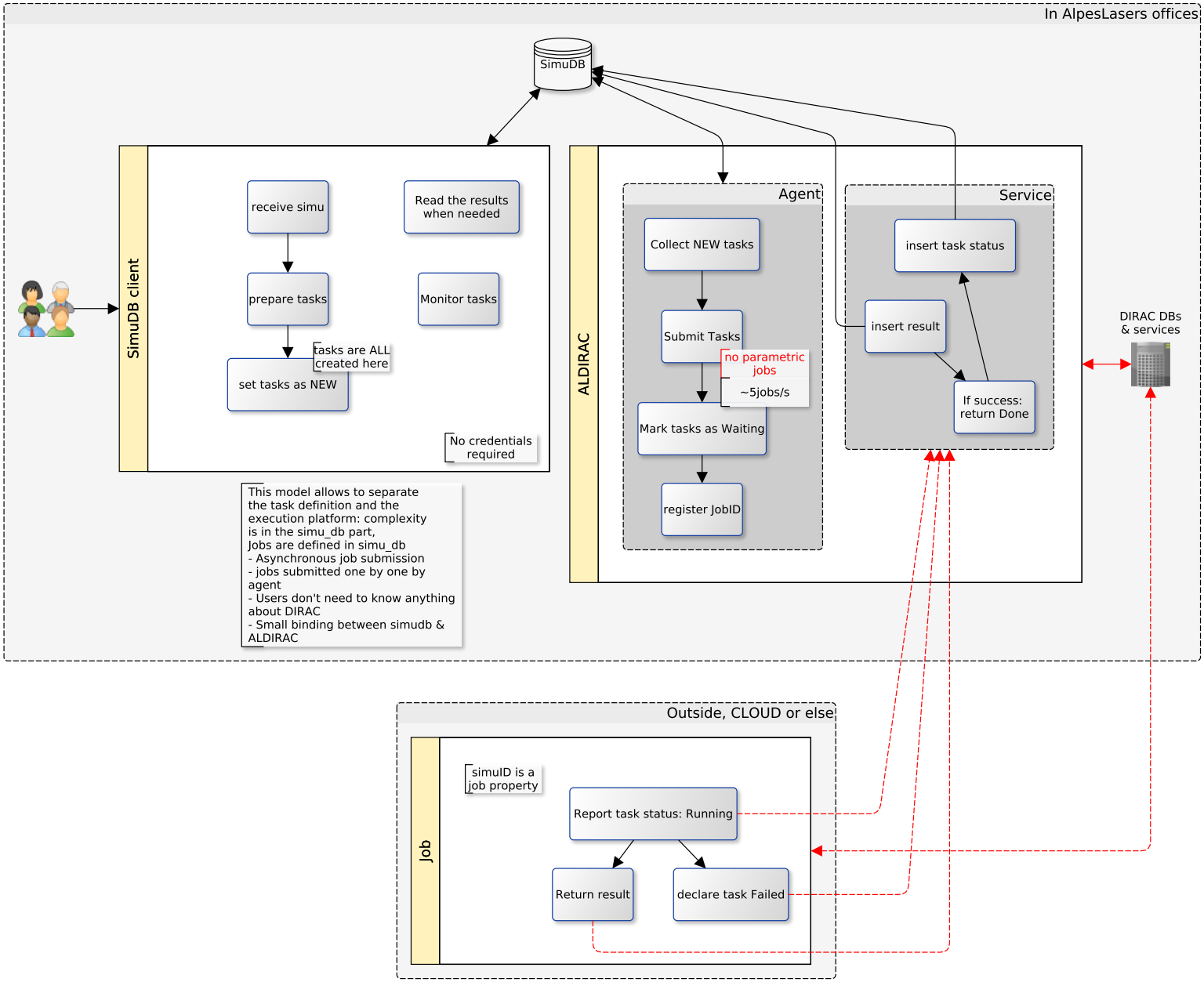
- Uses Template Method pattern
  - define algorithms in super class, let subclasses overwrite the steps
- Took an afternoon to get the UI working
- Adding a new application is fast

Dedicated system: SimuDB

Decouple ALDIRAC from Alpes Lasers' data source (PostgreSQL DB)

- Tasks created in central DB by external client
- Submitted by dedicated agent
- Tasks' statuses reported by jobs through dedicated service
- Results inserted in DB directly by job through service

Advantage: loose coupling, ease of change.


In AlpesLasers offices
SimuDB
SimuDB client
receive simu
Read the results when needed
prepare tasks
Monitor tasks
tasks are ALL created here
set tasks as NEW
No credentials required
This model allows to separate the task definition and the execution platform: complexity is in the simu_db part,
Jobs are defined in simu_db
- Asynchronous job submission
- jobs submitted one by one by agent
- Users don't need to know anything about DIRAC
- Small binding between simudb & ALDIRAC
ALDIRAC
Agent
Collect NEW tasks
Submit Tasks
no parametric jobs
~5jobs/s
Mark tasks as Waiting
register JobID
Service
insert task status
insert result
If success: return Done
DIRAC DBs & services
Outside, CLOUD or else
Job
simuID is a job property
Report task status: Running
Return result
declare task Failed

Conclusions and Prospects

Conclusions:

- Ran successfully few thousand jobs

Prospects:

- Simulate all existing devices
- Compare results with measured values
- Determine quality of simulation tools
- Add more tools to ALDIRAC
- Produce a web front end to our workflows to allow clients to interact