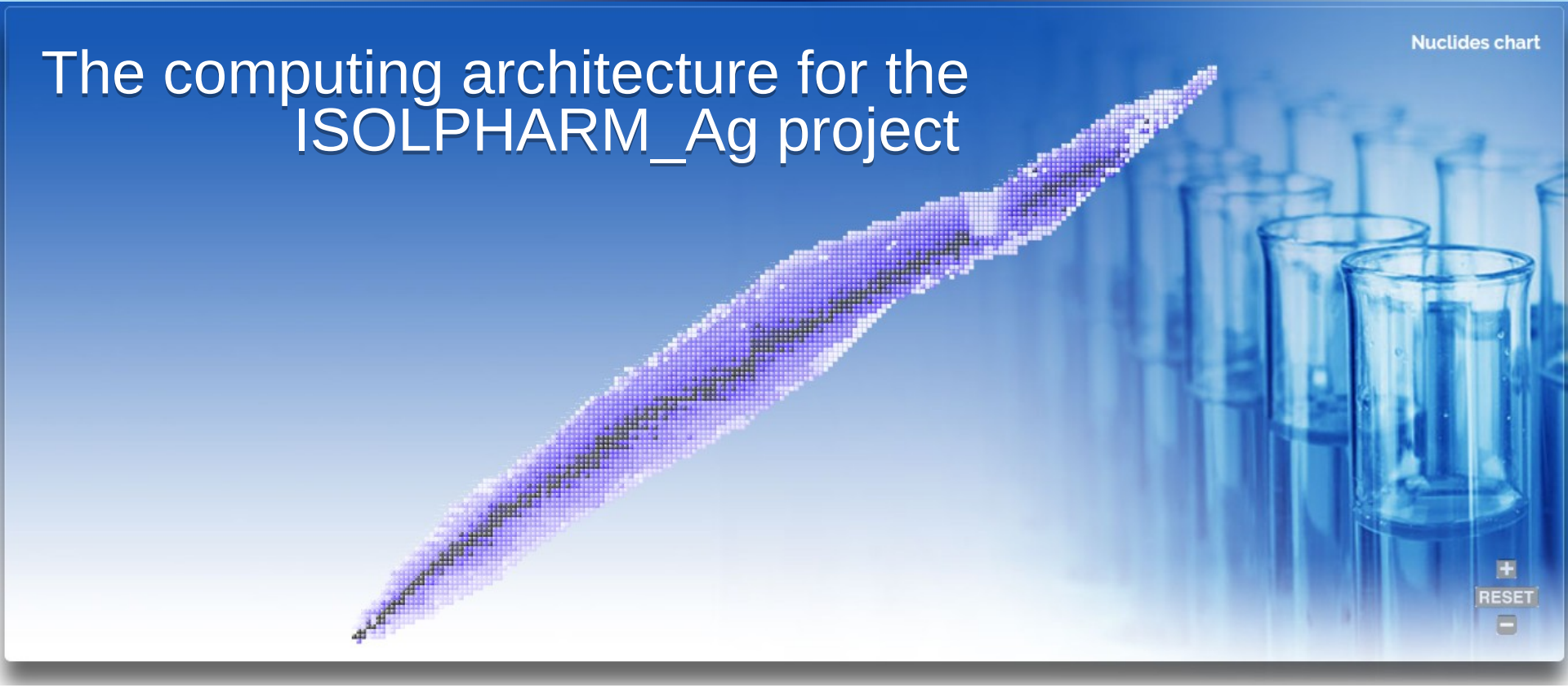
The computing architecture for the ISOLPHARM_Ag project

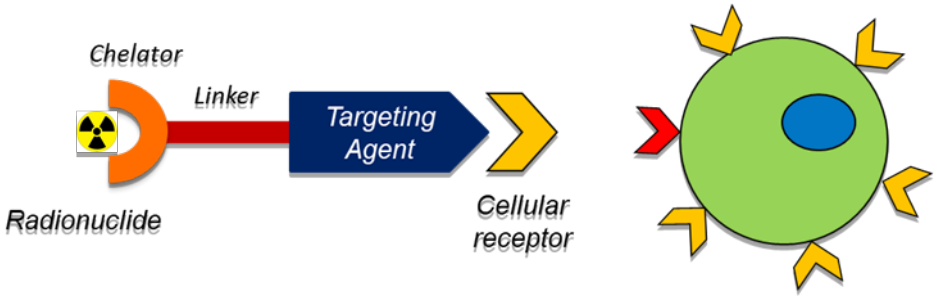


Lisa Zangrando
INFN-PD

The ISOLPHARM project

ISOLPHARM is an INFN project devoted to the discovery and development of high purity *radiopharmaceuticals* exploiting the radionuclides producible with the INFN international (EU, USA, CANADA) patented methodology by means of the ISOL technique

Radiopharmaceuticals are drugs which deliver a *radionuclide* to a specific cancer cell allowing diagnostic or therapeutic procedures

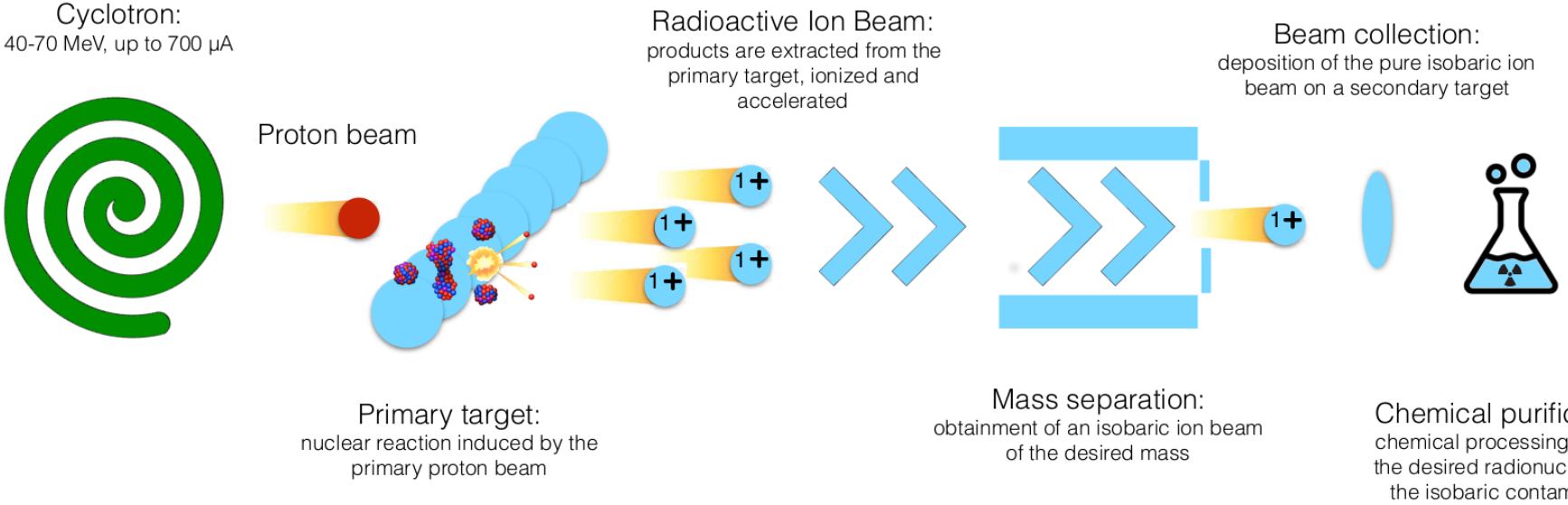


Radionuclides will be produced by using the SPES facility at INFN-LNL

ISOLPHARM_Ag is a two-year INFN activity to study and demonstrate the production and use of a particularly promising silver isotope (111Ag)

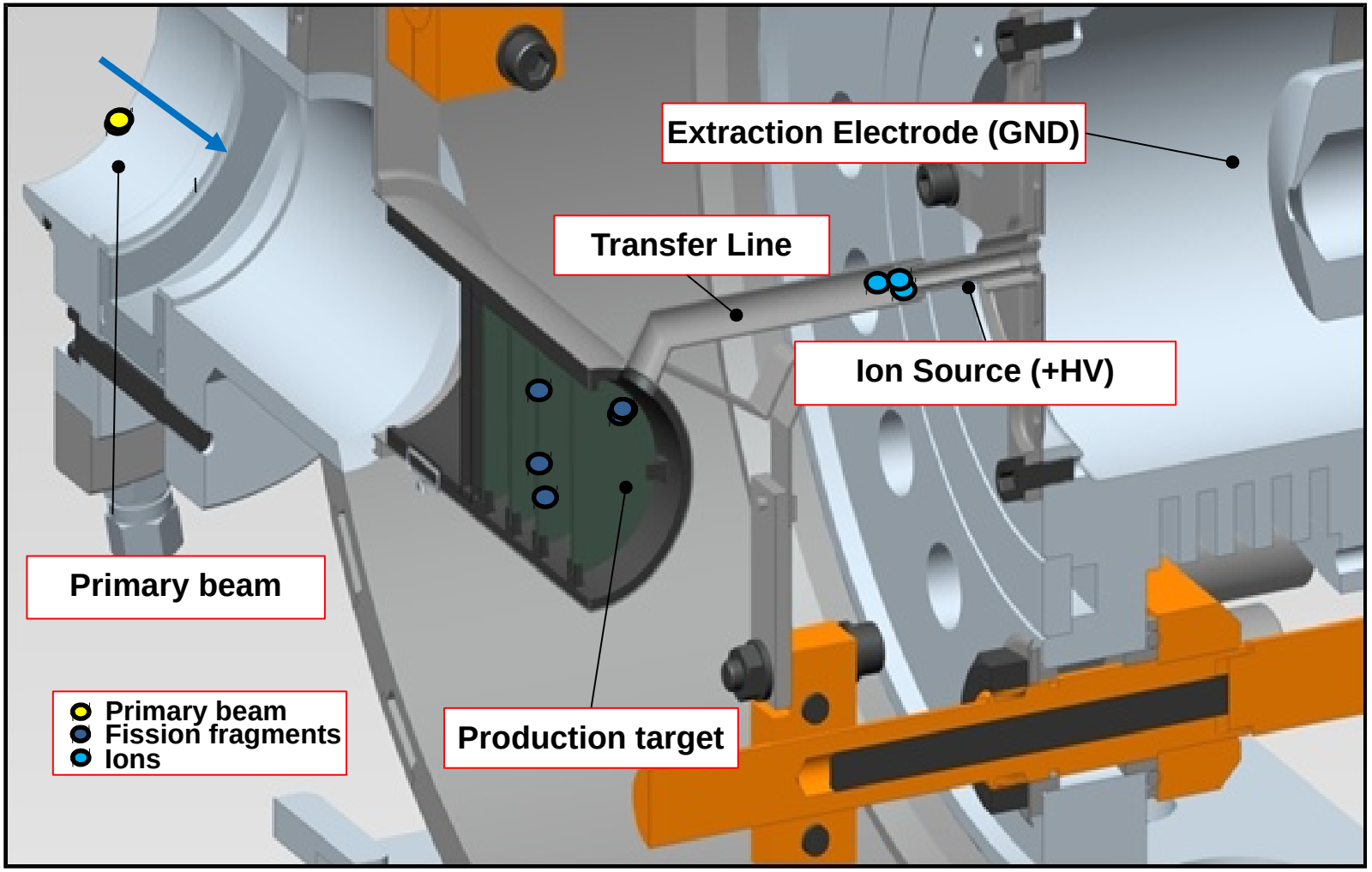
The ISOLPHARM methodology

The ISOLPHARM method is capable of selecting and isolating a single radio-isotope



With the ISOL technique, we can produce high purity radiopharmaceuticals with lower costs compared to traditional techniques and with a reduced environmental impact.

The production target



The computing needs

It is extremely important to study the behavior of the production target

We need to predict the production and release of 111Ag from the primary target

We achieve it by executing Monte Carlo simulations based on FLUKA and Geant4

- computing intensive
- computing tasks parallelizable

An IT infrastructure has been designed to meet the project computing needs

- execution of independent and parallel Monte Carlo simulations
- analysis of the produced data through different tools including graphical and Big Data facilities
- Web based portal to allow users to submit their Monte Carlo simulations and then collect/view the produced data

ISOLPHARM relies on the CloudVeneto infrastructure for the provisioning of the required computing resources

CloudVeneto

CloudVeneto is shared by INFN-PD, INFN-LNL and 10 departments of the University of Padova

	Compute nodes	Cores (HT)	RAM (GB)	Storage (TB)
INFN PD	18	808	2656	133
INFN LNL	13	416	1472	
UNIPD	12	480	1920	90
total	**43**	**1704**	**6048**	**223**

NB: storage based on Ceph (S3 interface on-going)

The computing architecture

The computing architecture should adopt modern thecnologies with support for:

- fast application deployment (even in Cloud)
- scalability and high availability
- security

Application deployment: Linux containers

- small, fast and portable runtime environments
- use the OS-level virtualization
- isolated from each other and from the host
- microservice architecture

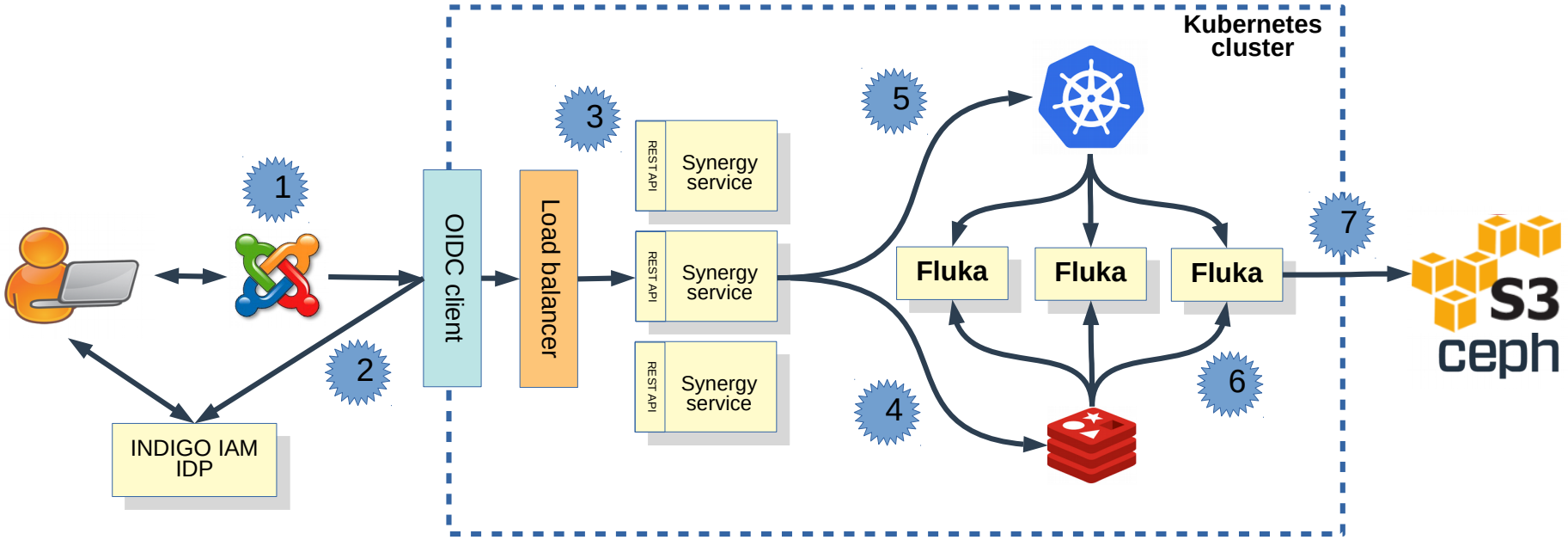
Container orchestrators: Apache Mesos, Docker swarm, Kubernetes

- open-source systems that manage large computing cluster
- main features: automating deployment, auto-scaling, high availability, monitoring and management of containerized applications (i.e. orchestration).

Tested Mesos and Kubernetes

Chosen Kubernetes

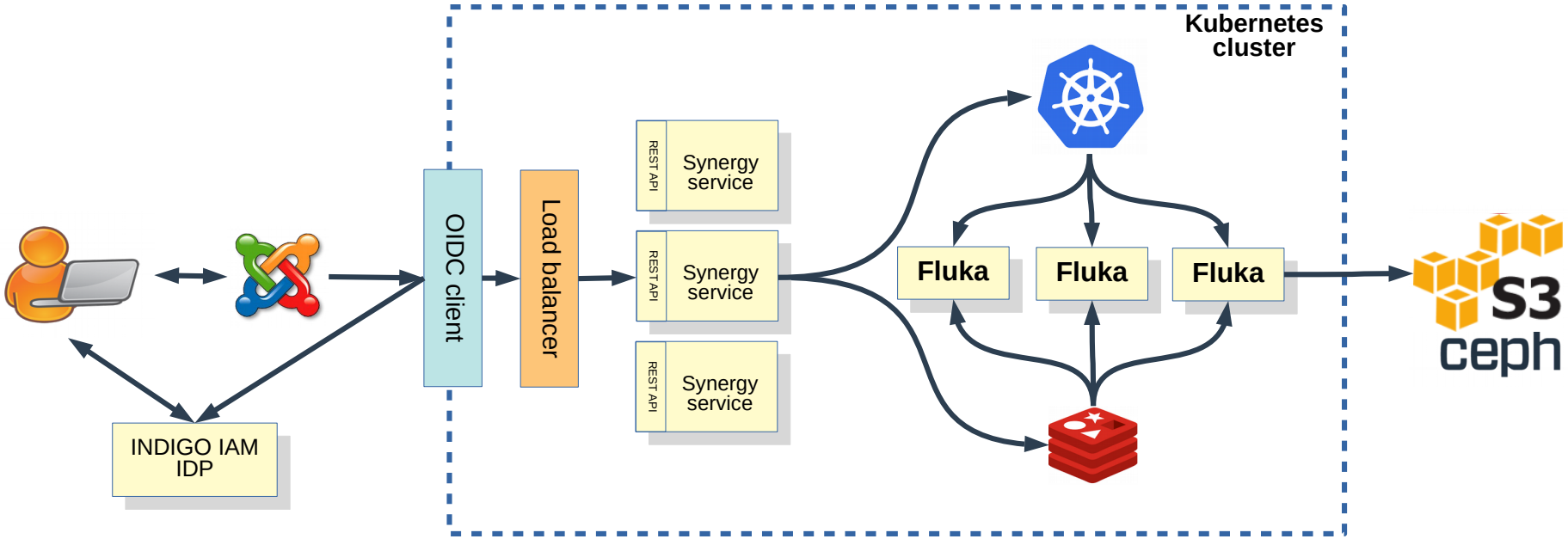
The computing architecture



1) A user submits a job
2) Security: AuthN / AuthZ (INDIGO IAM)
3) LB forwards the job to Synergy (3 replicas in HA)
4) Synergy validates the user job request, creates a new queue in Redis and inserts the task parameters
5) Synergy creates a new parallel job (Fluka or Geant4) in Kubernetes
6) Each task takes parameters from the queue and populates its template
7) The output files produced by the task execution are stored into a S3 storage cluster

The security

- Kubernetes supports several AuthN technologies (OIDC, SAML, etc)
  - just configuration (NOT code)
- Security based on INDIGO IAM (Identity and Access Management)
  - OpenId Connect (OIDC)
- AuthZ based on Kubernetes RBAC (Role-Based Access Control)
  - local policies



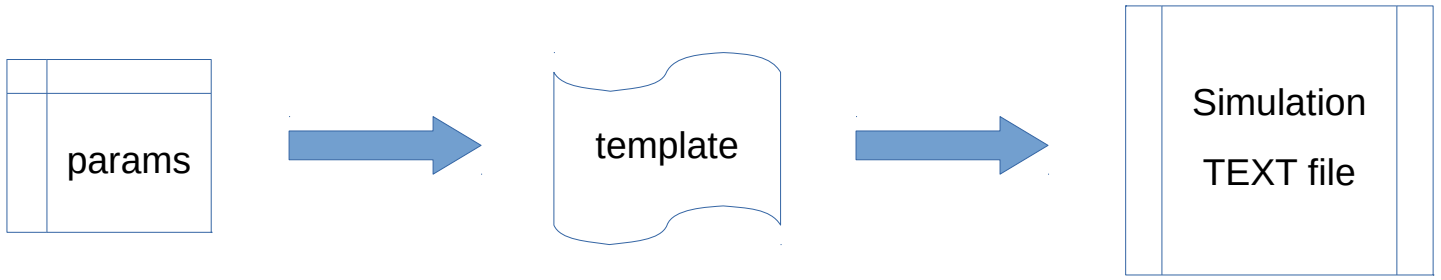
Abstract input parameters

Setting up a simulation is a very complex and error prone task

The complexity increases if you consider that FLUKA and Geant4 have different simulation schema and input parameters

To simplify the simulation set-up for normal (i.e. not advanced) users, we defined a set of abstract input parameters and ad-hoc simulation templates for FLUKA and Geant4

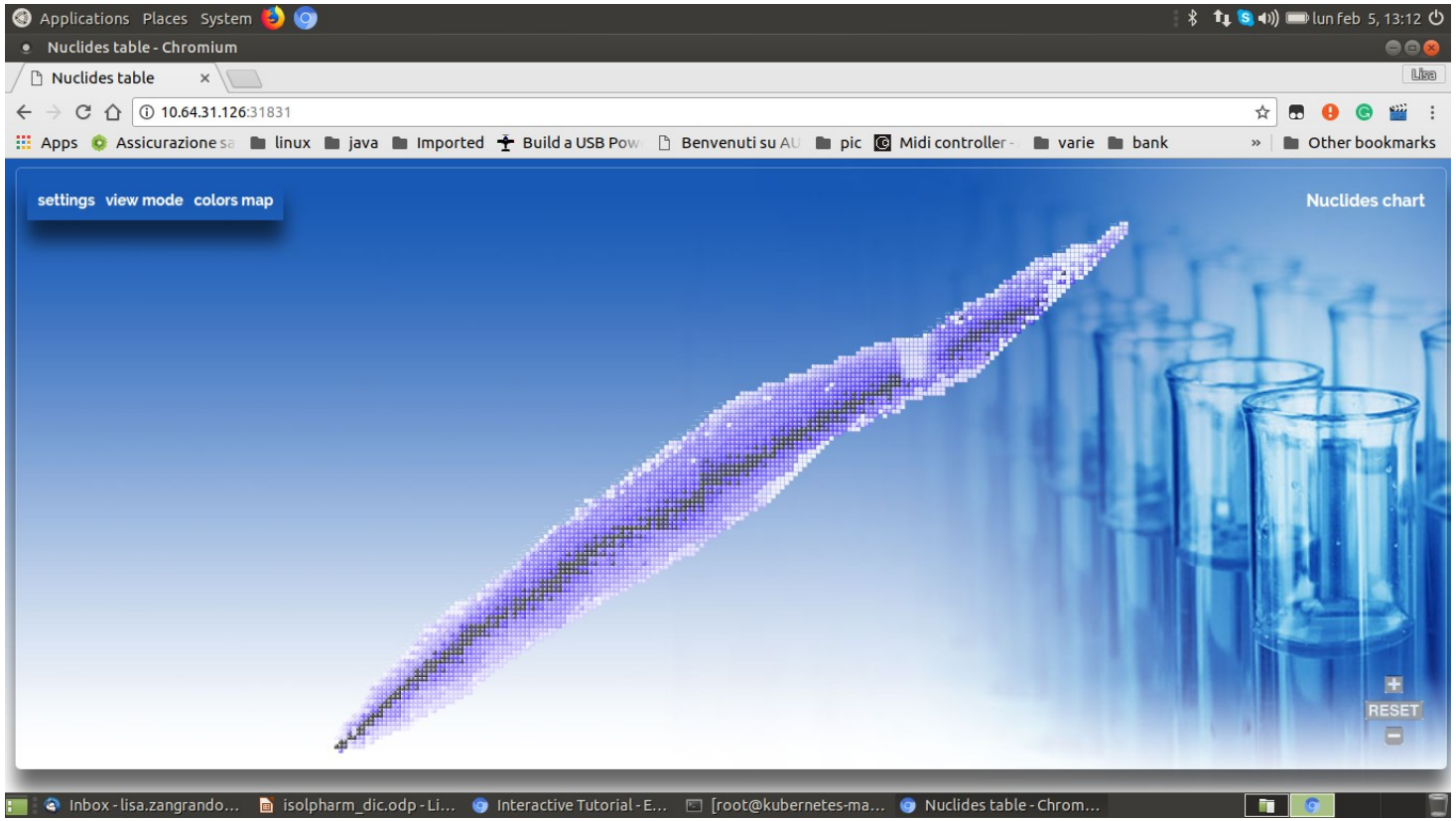
Such parameters will be specialized for FLUKA or Geant4 and applied to the related simulation template at runtime



The Web portal

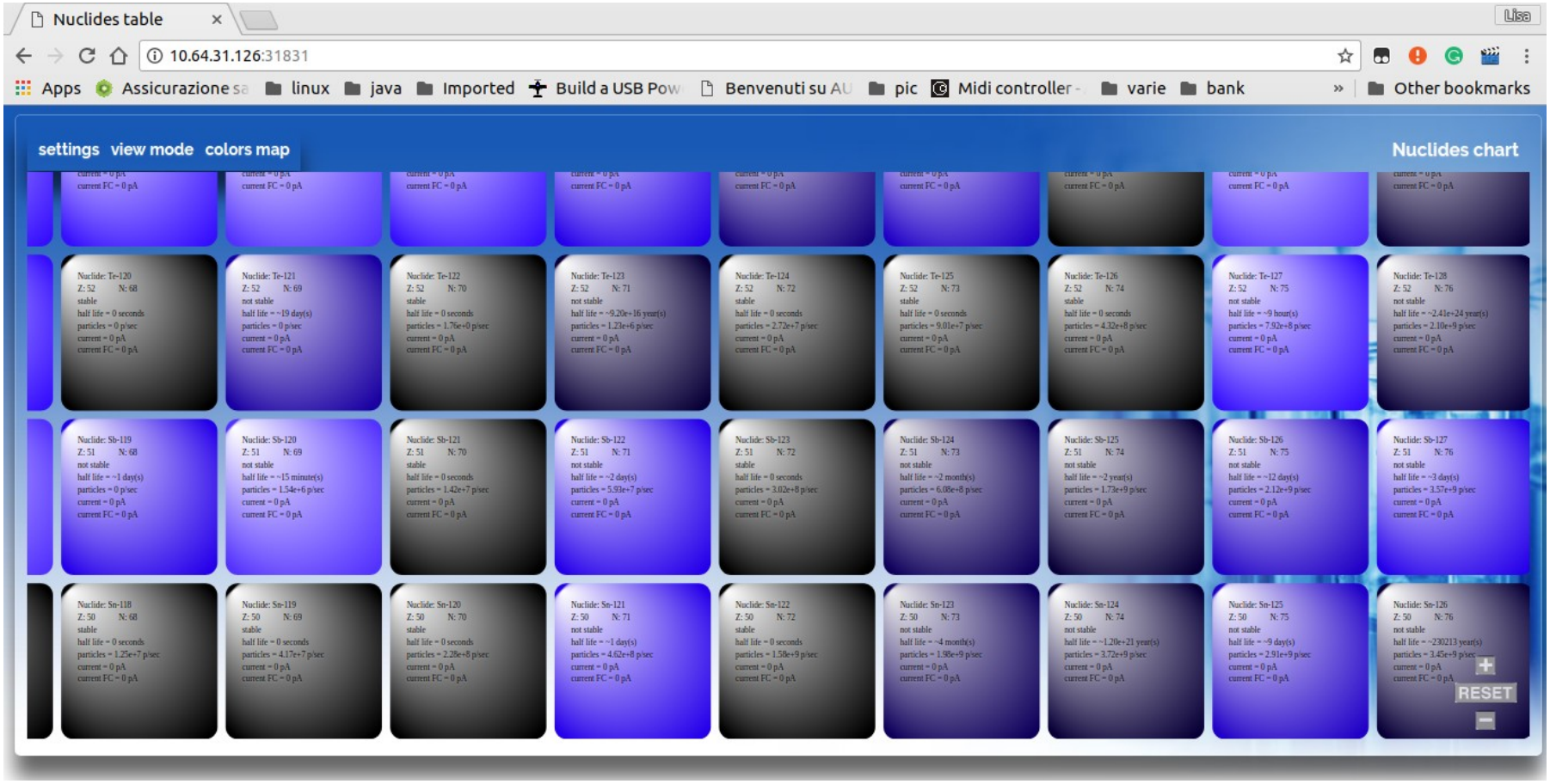
We take advantage of **web.infn.it** which provides Joomla!, a free and open-source Content Management System (CMS) for publishing Web content

Implemented first prototype for graphically display the geant4 and FLUKA simulation results (e.g. particles productions) mapped on the nuclides chart



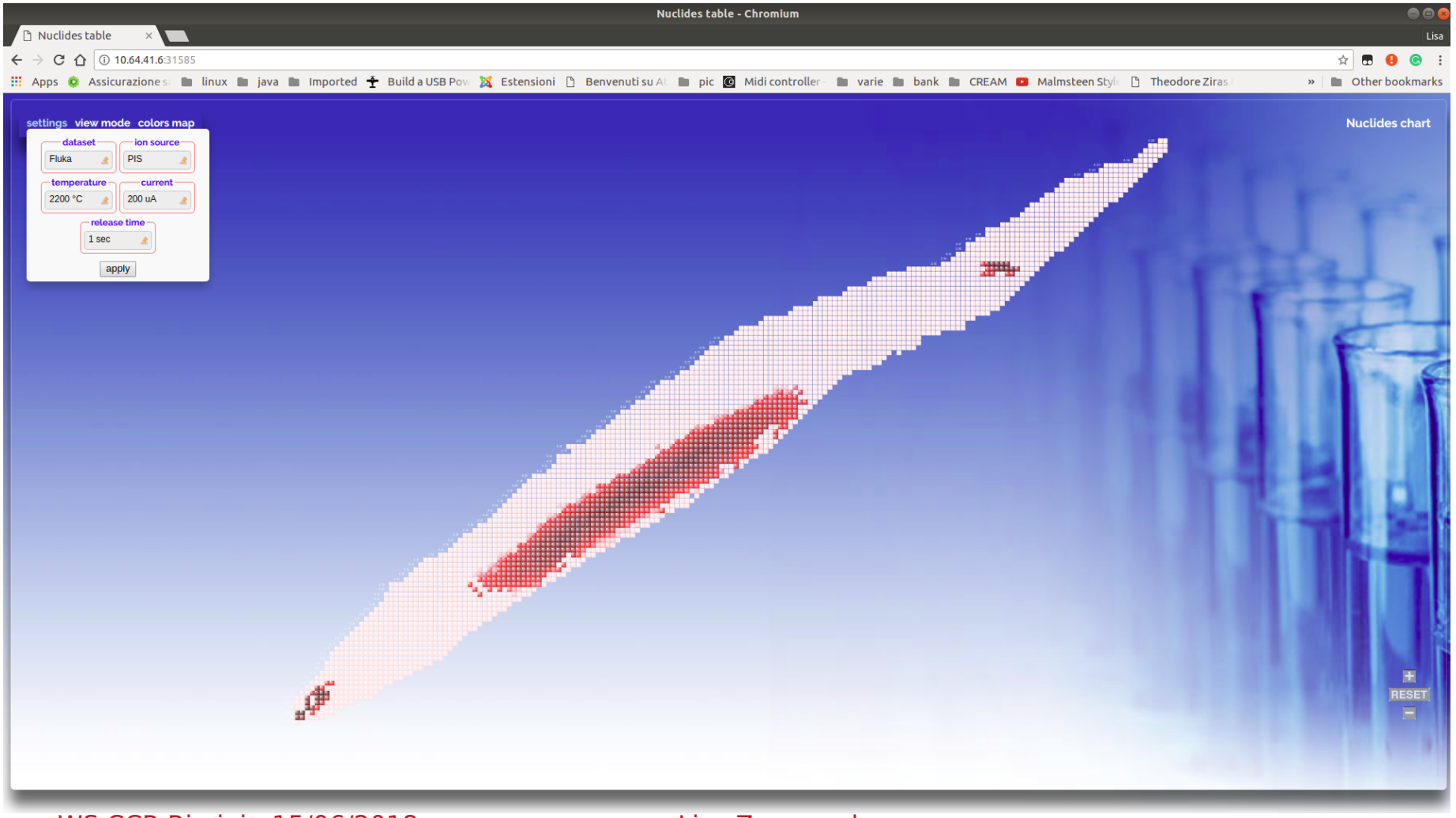
The Web portal: the Nuclides Chart

The chart shows the detailed data about the produced nuclides



The Web portal: the Nuclides Chart

Several view modes: by half life, particles, current (FC)



The status

- Overall architecture validated
  - building block for Big Data analytics
- Synergy:
  - no changes required for the core service
  - implementation of two new managers for interacting with Redis and K8s master: 70% done
- FLUKA and Geant4:
  - docker containers: done
  - common set of input parameters: done
  - simulation templates: FLUKA done, Geant4 on-going
- S3 support in Ceph: to be finalized
- Web Portal based on Joomla: just evaluated
- data analysis tools
  - first prototype of Nuclides chart to be ported in Joomla
  - some others to be implemented

Questions?