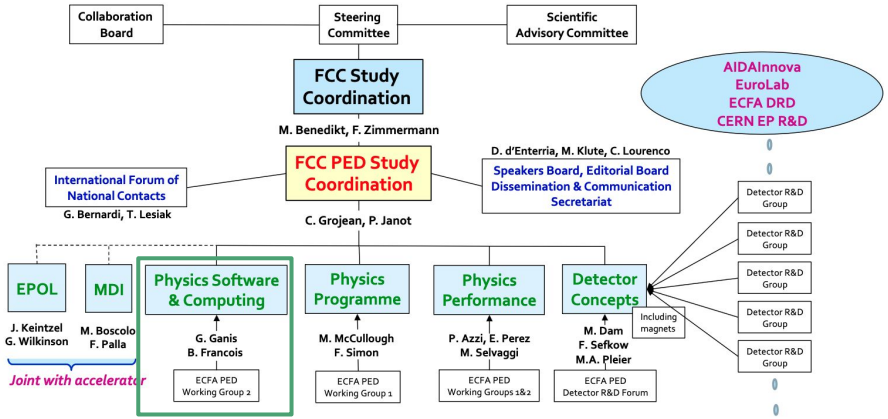
FUTURE
CIRCULAR
COLLIDER

FCC Physics Software General Status

FCC Week 2025
Vienna, Austria

May 21st, 2025
Gerardo Ganis, **Brieuc François** (CERN)

Introduction



- **Software is omnipresent in modern high energy physics**
  - As a backbone of the field, software can have **big positive impact if well designed**
    - A lot of care has to be taken in its preparation/implementation!
- Initial mandate of the **Physics Software and Computing group**: support the **software and computing needs of the FCC Feasibility Study**
- Recently extended to include the pre-TDR phase (2025-2027) with the objective to **leave a *usable* system for proto-collaborations to start with**
  - But let's try to do more!

FCC Software Developments Challenges

Main **challenges** for the FCC software developments

- A **two-fold mission** with some tensions: “Write software for the future” VS “Deliver working solution in a short time with little person power”
- FCC is a very **long term** project and we have to provide stability
  - Maintain expertise over long periods, deal with technology discontinuation, remain open to big paradigm changes, …
- A lot of **moving targets** ➔ great need for flexibility
  - E.g. detector geometry change impacts all subsequent steps (DIGI-RECO-ANA)
- Want to use software for **optimization** ➔ run whole chains under many scenarios
  - E.g. choice of optimal detector designs based on physics analyses results (instead of single, isolated, detector performance metrics)
- …

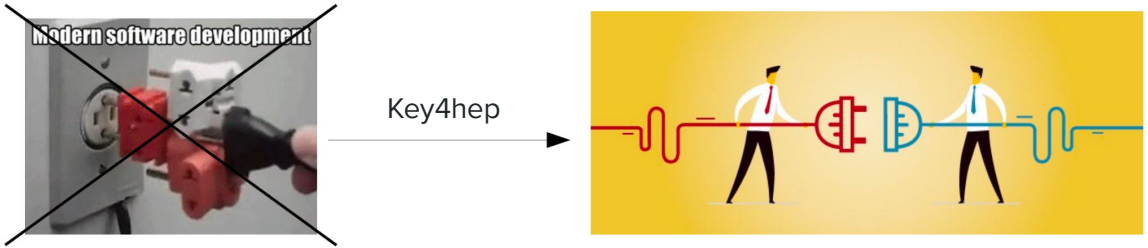
FSR and Software Notes Released

FCC FSR: Volume 1, PED DOI:10.17181/CERN.9DKX.TDH9

The FCC software for PED studies: [DOI:10.17181/8k0c4-nkr70](DOI:10.17181/8k0c4-nkr70)

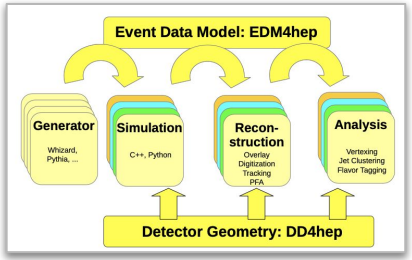
Common Software

- HEP experiments have largely overlapping software needs, and commonalities are bigger than differences
  - Adopting a **common software ecosystem** is very appealing and could maximize the scientific outcome of the involved projects
    - Exploit synergies, profit from centrally provided solutions (avoid reinventing the wheel), large pool of experts to get support from, higher chances for long term availability of the solutions, …
- The **Key4hep** software project as the answer
  - Aims at providing complete solutions (generators to analyses) **for all future colliders**
  - Matured a lot recently, already used in production
  - Success: adopted and developed by FCC, ILC, Muon Collider, EIC, …



Key4hep

- Provides policies to ensure interoperability and enable synergies
  - Main ingredients as of today: **DD4hep** (detector geometry and Geant4 simulation), **EDM4hep** (data model built by PODIO) and **Gaudi** (orchestration framework)
- Provides a stack of consistently built packages covering many various applications
  - On AlmaLinux9: `source /cvmfs/sw.hsf.org/key4hep/setup.sh`
- Interfaces software from linear collider (ILCSoft, based on Marlin and lcio)
  - Migration of these algorithms ongoing to get rid of the wrappers/converters (and go multi-threaded)
- And much more: template to easily start new GitHub projects, testing and physics based validation facilities, debug builds, …



Detector Geometries

- FCC (and other future colliders) detector geometries hosted in k4geo package
  - **Complete CLD**, including variants e.g. CLD + ARC for PID
  - **ALLEGRO ECAL and HCAL** barrel + endcaps
    - Completed with IDEA tracking system and muon tagger place-holder
  - **Complete IDEA former baseline**, version with crystal almost ready
  - **ILD adapted to the FCC-ee** MDI

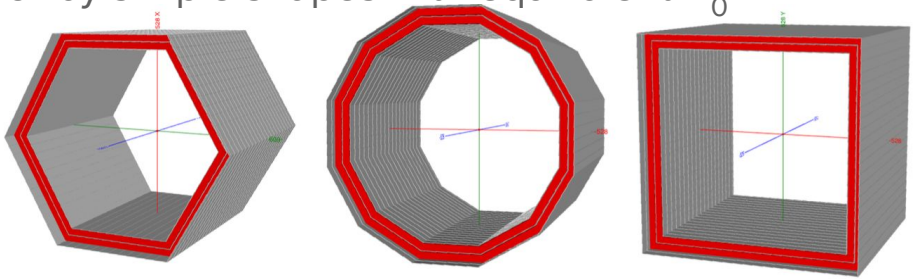
```
<include ref="Vertex_o4_v07_smallBP.xml" />

<include ref="InnerTracker_o3_v08.xml" />
<include ref="OuterTracker_o3_v08.xml" />

<include ref="ARC_o1_v01.xml"/>

<include ref="ECalBarrel_o2_v01_03.xml" />
<include ref="ECalEndcap_o2_v01_03.xml" />
```

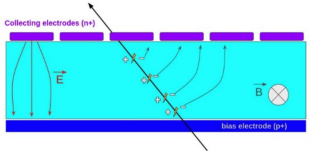
- DD4hep plug-and-play: **sub-detectors integrated in full concept with one xml line**
  - If dimensions have to be adapted, only easy with flexible sub-detector builders
- **At this stage of the project, important to keep detector builders simple/flexible!**
  - Services and support can be accounted for by simple shapes with equivalent X_0
  - Det. layouts are and will keep changing
- More details in Alvaro's talk

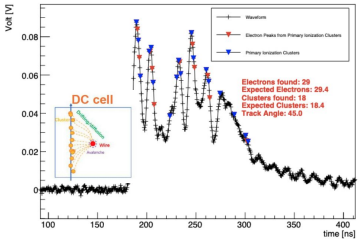


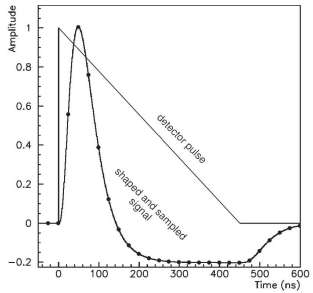
IDEA muon system flexibility, Mahmoud Ali

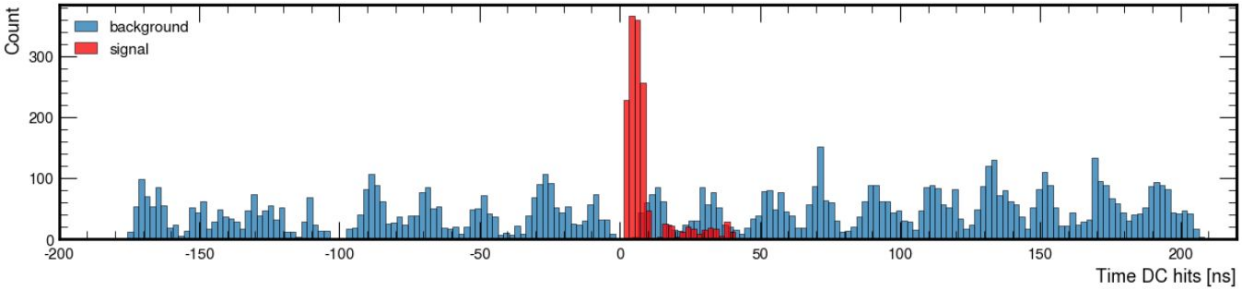
Digitization

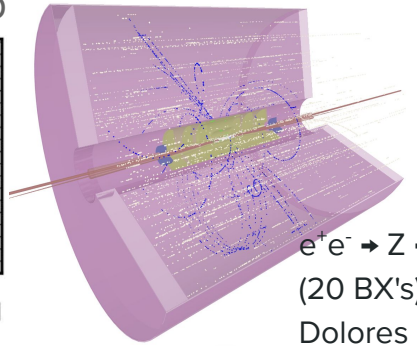
- Simple ‘digitizer’ available for all detectors
  - Gaussian smearer for tracking like detectors
  - Hit energy sum, calibration, noise and cross-talk for calorimeters
- Need to assess effect of **beam induced background**!
  - **Overlay** tool ready: mixing SIM hits of pre-generated bkg ‘events’
- **Detailed digitizer** available for SiPM’s, others being prepared
  - Only way to properly take bkg into account + will help to define electronics specifications
  - Generic silicon, drift chamber and Noble Liquid ECAL are ongoing
  - Strategy: try to treat everything than can be after the SIM step











$e^+e^- \rightarrow Z \rightarrow qq$ + IPC (20 BX's) in IDEA
Dolores Garcia

Reconstruction

- A lot already exists, but far from complete (most active area at the moment)
- **Tracking**: CLD conformal tracking, ILD (need tuning for the FCC-ee version), IDEA ongoing (see Andrea's talk), ACTS and Key4hep compatibility being improved (muon collider effort), highly displaced non pointing tracking ongoing (Mahmoud)
  - Tracks from GenParticles available as a work around for dev. purposes
- **Calorimeter clusterings**: sliding window and topological (ALLEGRO, generalized → now being applied to IDEA dual readout), CLUE
- **Particle Flow**: PandoraPFA used in CLD and being adapted for ALLEGRO, promising ML based approaches ongoing (see Farouk's and Lena's talk)
- **ParticleID**: ARC reco (Serena), ML flavor tagging (Sara), t.o.f. (ILCSoft, not used)
- **Let's try to design generic algorithms (detector agnostic as much as we can)**
  - We want to study a lot of different detector designs and can not maintain dedicated RECO chains for all of them!
  - When not fully possible, factoring out the detector specific part should be done
- More details on digitization and reconstruction in Sanghyun's talk

Analysis and Resources

- Parametrized and Full Sim outputs are in **EDM4hep format, ROOT based**
  - Primarily meant to be manipulated through the PODIO layer (C++ or Python bindings)
  - Also possible to analyze with ‘plain’ (Py)ROOT (but less user friendly, less stable)

While everyone can use their favourite **analysis framework**, FCCSW provides a **solution**: FCCAnalyses

- **RDataFrame** (RDF) + **RDataSource** to keep EDM4hep/PODIO functionalities (slow at the moment)
- Automatized **central samples management**
- Skimming, filtering, histogramming, plotting, CERN HTCondor job submission
- Helper tools: jet clustering, **ML support** (ONNX), ...
- Used by most FCC analyzers so far!
- More details in Juraj’s talk

What is ahead of us?

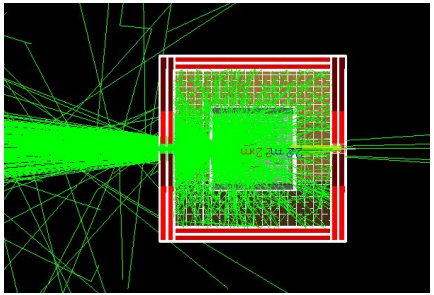
- Improve Python based analysis interface
- Facility for Gaudi based n-tuple producers
- **More analyses with Full Sim**
- Increase available computing resources
  - Everything done so far with 700 TB and 9000 HS23 from CERN (**1 ‰ of LHC resources**)
- GRID submission tools (ILCDirac) ready ➜ **need to attach more external resources to the FCC VO**
- Adapt for distributed samples, Rucio?
- **LEP data being migrated to EDM4hep!!**

Considering the FCC project's timeline, **complete paradigm shifts can also be explored!**

- **EDM4hep** files can now also be **analyzed** with the **Julia language**
- Part of JuliaHEP, a general effort proposing a **solution to the “two-language problem”** (C++/Python)

Additional Remarks

- Delphes parametrized simulation (not discussed here) is integrated in Key4hep and will be kept maintained
  - Will remain a key player e.g. for detector parameter sweep scans
- A chain from beam induced background to detector simulation is in place
  - Let's start playing with **MDI designs <--> occupancy studies**
  - Would profit from having **accurate and flexible MDI designs in Full Sim** (currently from CAD, not easy to change for us)



- Strong links between software development and DRD activities is crucial
  - Should **encourage adoption of Key4hep by DRD's** (where it makes sense)
    - DRD software development directly useable in the more global context of future collider studies (DRD's direct customer, all using Key4hep already)
    - DRD activities will greatly profit from the existing tools used in future collider studies

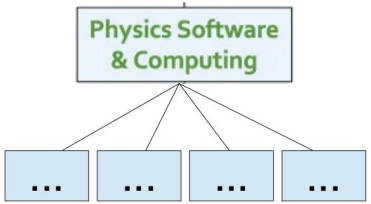
Documentation

- Trying to maintain up to date documentation/tutorials for key components
  - FCC Software landing page
  - FCC Software Forum
  - Key4hep
  - DD4hep and its Doxygen
    - DD4hep tutorials: DRD6, US FCC workshop, EIC Youtube videos
  - PODIO
  - EDM4hep Doxygen
  - FCC Software tutorials
  - FCCAnalyses
  - FCC FSR and associated software note

Please give feedback and report broken/outdated parts!
FCC-PED-SoftwareAndComputing-DocumentationATcern.ch

Conclusions

- Good progress on FCC software made during the Feasibility Study phase
- **Consolidation and advancements** in all directions has to occur for the **pre-TDR**
  - More detailed status and needs talk given at the 8th FCC Physics Workshop
  - Go through it to know where to contribute
- The **link between FCC Software/Key4hep and DRD's** should be strengthened
- Don't write code for yourself, make it generic and/or factorized!
- The FCC software team has grown a lot
  - 31 editors of the FCC FSR software note, thanks!
- The ball is rolling, let's keep the momentum (and acceleration)
  - Requires more people taking leading roles
  - Preparing a better **structured organization** with more official **responsibilities**



The FCC software for PED studies

Appendix to the FCC Feasibility Study Report

March, 2025

Principal editors
Brieuc François[1], Gerardo Ganis[1]

Chapter editors
Juan Miguel Carceller[1], Jan Eysermans[2], Thomas Madlener[3], André Sailer[1], Michele Selvaggi[1], Juraj Smiesko[1], Alvaro Tolosa-Delgado[1]

Contributors
Mahmoud Al-Thakeel[1,4], Andrea Ciarma[4], Wonyong Chung[5], Andrea De Vita[1,4], Archil Durglishvili[6], Dolores Garcia[1], Armin Ilg[7], Sungwon Kim[8], Sanghyun Ko[9], Andreas Loeschcke Centeno[10], Giovanni Marchiori[11], Michaela Mlynarikova[1], Lorenzo Pezzotti[1], Louis Portales[12], Edoardo Proserpio[13], Braulio Rivas[14], Jennifer Roloff[15], Romualdo Santoro[13], Swathi Sasikumar[1], Birgit Stapf[1], Zhibo Wu[16], Anna Zaborowska[1]

[1] CERN
[2] Massachusetts Institute of Technology
[3] DESY
[4] INFN
[5] Princeton University
[6] Ivane Javakhishvili Tbilisi State University
[7] University of Zurich
[8] Yonsei University
[9] Seoul National University
[10] University of Sussex
[11] APC, CNRS/IN2P3, Université Paris Cité
[12] CEA Paris-Saclay, IRFU, DPhP
[13] University of Insubria
[14] Escuela Superior Politécnica del Litoral
[15] Brown University
[16] LAPP, CNRS

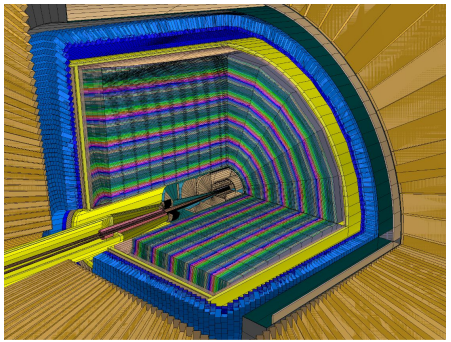
Thank you!

Additional Material

Visualisation

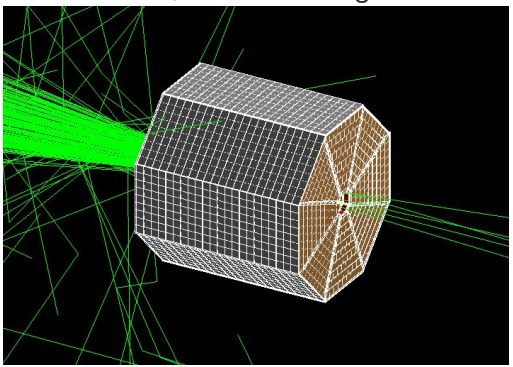
- Having a **visual rendering** of software's processes and outputs helps in
  - Debugging, validation, physics interpretation, outreach, …
- Several visualisation tools are available for FCC studies
  - DD4hep geoDisplay, TGeo + JSROOT, Geant4 Qt through ddsim
  - Phoenix based event displays
    - Customized for FCC (EDM4hep aware)
  - EDM4hep Event Data Explorer (EEDE): object properties and relations

GeoDisplay, W. Chung



IDEA with crystals implementation

G4 Qt, A. Tolosa Delgado



Synchrotron radiation in IDEA

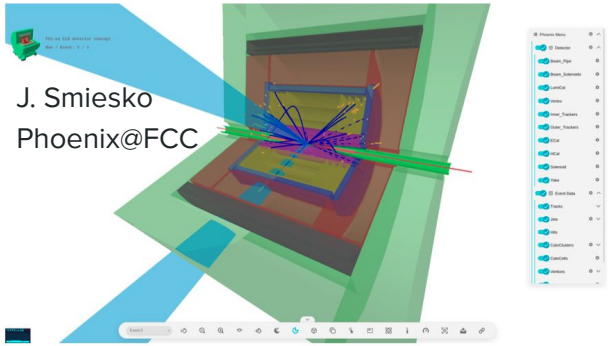
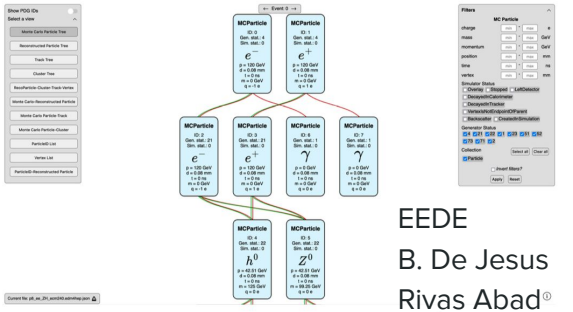


Fig. 128: Visualisation of a $t\bar{t}$ event within the CLD detector, displayed using the `Phoenix@FCC` web-based event display.



EEDE
B. De Jesus
Rivas Abad

Fig. 129: An example visualisation of the Monte Carlo particle tree for an $e^+e^- \to ZH$ event, generated using the `EDM4hep` Event Data Explorer [770].

Monte Carlo Generators

Relevant generators available through Key4hep, including LEP ones (still state-of-art in some cases)

- ❏ KKMCee (v5), Whizard, Sherpa, Herwig3, BabaYaga, MadGraph5, Pythia8, …
- ❏ Some ported from **Fortran to C++** (e.g. KKMC, S. Jadach, FCCW 2022)

Interface to Key4hep processing chains (k4Gen)

- ❏ Readers for the relevant formats (e.g. hepmc3)
- ❏ Integration of **MC generation directly in Gaudi**
  - ❏ GEN(-SIM-)DIGI-RECO in one config
  - ❏ Only Pythia8 so far

Python based module for **automatic MC configuration file generation**: k4GeneratorsConfig

- ❏ Sherpa, Whizard, MadGraph, KKMC, Pythia

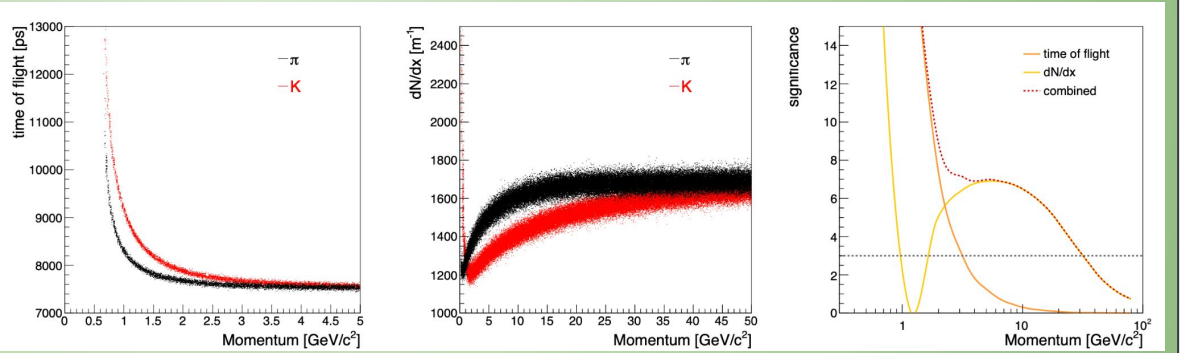
A lot of work needed in the long run…

- ❏ Higher order corrections integration, improved Beam energy spread (BES), ISR, beamstrahlung treatments (FCC-ee. EPJ. Plus 136, 911 (2021))
- ❏ MC Generators should be provided as **'modern', open software packages**
  - ❏ Versioning, building, automatic testing, with documentation
  - ❏ Some generator code modernization needed
- ❏ Validation of 'LHC' generators for e^+e^- (e.g. MadGraph)
- ❏ Key4hep specific
  - ❏ Generic, a posteriori, treatment of BES for generators not providing it
  - ❏ Extend the list of generators supported by k4Gen

Parametrised Simulation

- Parametrised simulation was used to produce first physics reach estimates and set detector requirements
- **Delphes** was i**ncorporated into Key4hep and enhanced** to meet the needs of the FCC Feasibility Study
  - k4SimDelphes: wraps core Delphes components and adds functionality for EDM4hep output
    - Standalone command-line tools (`Delphes*_EDM4HEP`) + Gaudi algorithm
  - Pythia8/EvtGen interface
  - Track covariance matrix
    - Detector resolution + multiple scattering for arbitrary geometries
  - Time of flight
  - dN/dx

M. Selvaggi



- Primary vertex smearing
- Further tune the existing detector parametrisation cards based on Full Sim
- Streamlined card production from Full Sim workflows

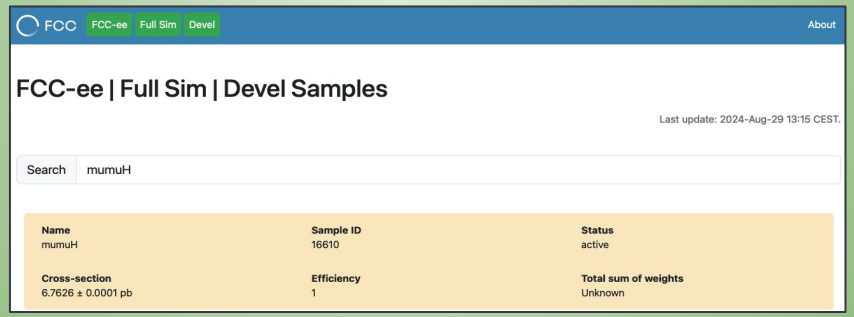
FCC-ee and the LEP data

- FCC-ee and LEP share some center of mass energies
- Data from ALEPH, DELPHI and OPAL are still available but difficult to manipulate
- **Migration of LEP data to EDM4hep would**
  - Bring a **great added value** to
    - FCC-ee: get **access to still useful data**
    - LEP: get **better data preservation perspectives**
  - Enable the application of the **same algorithms on simulated and real data**
    - More realistic evaluations of the techniques under study
  - Provide opportunities for future FCC **PhD students to publish with real data**
- Ongoing project to investigate **migration of ALEPH data to EDM4hep**
  - Extract data in ASCII form, running ALEPH code on SLC4/SLC6 (Singularity + CernVM-FS)
  - Injection in EDM4hep, **analysis with *standard* FCC analysis techniques**
  - Main **difficulty**: recover detailed information about the data, **not fully documented**
    - **Opportunity** for **LEP experts** to help!
  - **More details in Jacopo's talk**

Current and Pre-TDR Resources

Centrally available samples

- Many Delphes samples available
  - Produced with CERN HTCondor
- **Grid submission** tool ready: FCC VO, ILCDirac
  - Used to produce **first Full Sim samples** with CLD (Higgs recoil mass, couplings, …)
- Web based samples browsing



Everything done so far with **O(1 ‰) of LHC resources!**

- 700 TB, 9000 HS06, 3 GPUs (CERN SME quotas)

Plans for the pre-TDR phase

- Full Sim getting ready → resources allocated to FCC have to scale up (**storage** and CPU)
  - Discussions ongoing to have **FCC integrated in WLCG**
  - **Calls** for integrating **more national resources into the FCC VO** (storage!)
  - Investigating EuroHPC resources for "burst" CPU needs (e.g. MC campaigns)
- A minimal '**100*LEP@Z + nominal int. lumi. @ W/HZ/Top'** affordable a few times for 4 detectors throughout the pre-TDR
- Set-up MC production campaign procedures
  - Pre-production, validation, …
  - Content of GEN-SIM-DIGI-RECO-?NTUPLE?

Longer Term Storage Needs (FCC-ee)

- Estimated FCC-ee **storage** need during the **Z run, for data only**, is **comparable** to what will be needed by the end of **HL-LHC**
- To fit the MC budget, an **increase of the HL-LHC resources** will likely be **needed**
  - +20% each year after HL-LHC would likely be required to simulate ~10 times the int. lumi.
- We **need to understand the statistical power that will be required** for the analyses and plan accordingly

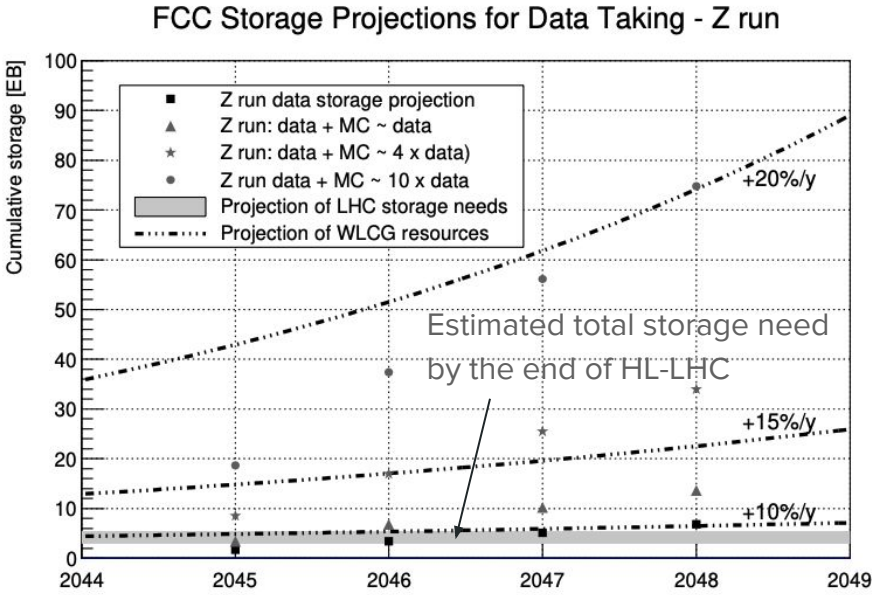


Fig. 131: Projection of the current resources to Z run based on the assumptions of four experiments, four equal runs in 2045, 2046, 2047, and 2048, and varying amounts of Monte Carlo data. The figure also includes the projected resource needs of the LHC and a hypothetical evolution of WLCG resources under different scenarios for sustained annual budget increases. See text for details.

More details in this talk
and in the FSR Sec. 8.11

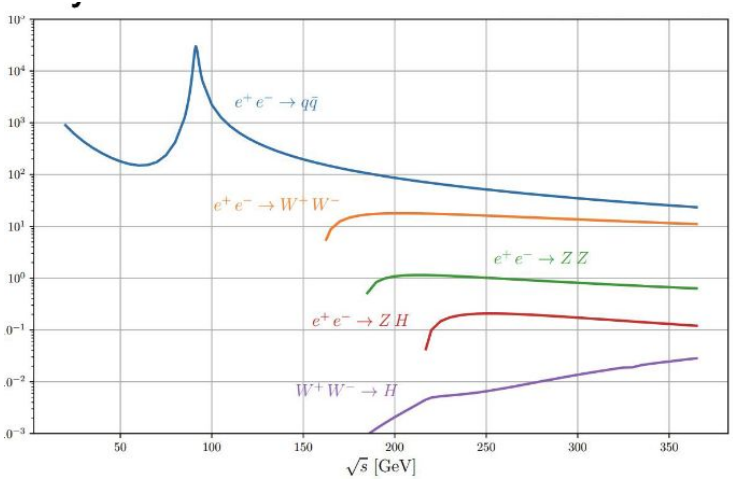
Computing needs per event

Process $e^+e^- \rightarrow$	E_{CMS} (GeV)	Event Sizes /evt		Processing time /evt	
		`Delphes` (kB)	`Full` (MB)	`Delphes` (ms)	`Full` (s)
$Z \rightarrow \bar{q}q, \ell^+\ell^-$	91.18	8.3, 1.2	1.1, 0.16	14, 0.5	11, 1.6
$W^+W^- \rightarrow$ all, $\nu\bar{\nu}\ell^+\ell^-$	157-163	9.5, 1.2	1.3, 0.16	16, 0.5	13, 1.6
$HZ \rightarrow \nu\bar{\nu}b\bar{b}$, $b\bar{b}b\bar{b}$	240	8.9, 13	1.2, 1.8	15, 23	12, 18
$ZZ \rightarrow$ all	240	10	1.4	17	13
$\bar{t}t \rightarrow$ all	365	18	2.3	30	23

Computing needs for 100xLEP

Table 23: Baseline projected needs per detector concept for the scenario with nominal integrated luminosity samples for the W, HZ and top runs, and event samples 100 times larger than the LEP samples for the Z run (see text for details). The total corresponds to 4 experiments requiring the same resources. Amount of HS06 is shown for a reference period of 3 years, i.e., roughly the pre-TDR duration. In bold are the figures beyond today's availability.

Run	Process	N evts	Delphes		Full	
			Storage (TB)	CPU (HS06)	Storage (TB)	CPU (HS06)
Z (100xLEP)	$\bar{q}q$	400 M	3.25	~ 1	440	475
	$\ell^+\ell^-$	42.5 M	0.05		6.5	7
W	W^+W^-	60 M	0.6		75	72
HZ	HZ	500 k	0.0065		1	~ 1
	VBFH	16 k	$\sim$0.001		0.25	
Top	$\bar{t}t$	500 k	0.009		9	~ 1
	HZ	90 k	$\sim$0.001		0.2	
	VBFH	23 k	$\sim$0.001		0.25	
Total		500 M	4	~ 1	**530**	**550**
4 exp		2000 M	16	~ 4	**2100**	**2200**



Longer Term Resource Needs (FCC-ee)

Table 22: Baseline needs for the nominal integrated luminosity and per detector concept. Amount of HS06 is shown for a reference period of 3 years, i.e., roughly the pre-TDR duration.

Run	Process	N evts	Delphes		Full	
			Storage (PB)	CPU (HS06)	Storage (PB)	CPU (HS06)
Z	$\bar{q}q$	1500 G	12.5	2.2 k	1650	2 M
	$\ell^+\ell^-$	225 G	0.275	12	40	40 k
W	W^+W^-	60 M	$\sim 10^{-3}$		0.075	72
HZ	HZ	500 k	$\sim 10^{-5}$		$\sim 10^{-3}$	~ 1
	VBFH	16 k	$\sim 10^{-6}$		$\leq 10^{-3}$	
Top	$\bar{t}t$	500 k	$\sim 10^{-5}$		$\sim 10^{-2}$	~ 1
	HZ	90 k	$\sim 10^{-6}$		$\leq 10^{-3}$	
	VBFH	23 k	$\sim 10^{-6}$		$\leq 10^{-3}$	
Total		1725 G	**13**	2.2	**1690**	**2 M**