



Offline computing resources for FCC-ee and related challenges

Clément Helsens^a , Gerardo Ganis

CERN, Meyrin, Switzerland

Received: 7 July 2021 / Accepted: 15 November 2021
© The Author(s) 2021

Abstract The international Future Circular Collider (FCC) study aims at designing pp , e^+e^- , e^+p colliders to be built in a new 100-km tunnel in the Geneva region. The electroweak, Higgs and top factory (FCC-ee) is designed to provide collisions at a centre-of-mass energy range between 90 (Z-pole) and 365 GeV ($t\bar{t}$) and unprecedented integrated luminosities, producing huge amounts of data which will pose significant challenges to data processing. In this study, we discuss the needs in terms of storage and CPU for the diverse phases of the project, and the possible solutions mostly based on the models developed for HL-LHC.

1 Introduction

The FCC-ee, the first stage of the Future Circular Collider (FCC) integrated programme [1], plans to collide e^+e^- at various centre-of-mass energies. The nominal run plan for expected instantaneous and integrated luminosities and relevant events statistics for the different physics runs is reported in Table 1.

FCC-ee is planned to start operation after the high-luminosity stage of LHC (HL-LHC) is completed, i.e. around 2040. Offline computing at FCC-ee will therefore take advantage of the HL-LHC computing model and achievements. The computing needs for FCC-ee are driven by the Z run and are usually considered comfortable, in particular considering that no or negligible pile-up is expected for an e^+e^- collider.¹ The exercise we are discussing in the study relates to the preparation of the Feasibility Study Report, FSR, to be submitted to the next European Strategy Update. We assume the bulk of the studies, driven by the Physics Performance group [2], will be run during the three years 2022–2024. We also need assumptions for the number of detector concepts to be evaluated. This is more complicated, and the only possible approach is to estimate the resources needed as a function of the number of detector variations to be evaluated.

The study is structured as follows. After presenting the typical workflows which we consider relevant for this study in Sect. 2, in Sects. 3 and 4 we estimate the needs in terms of storage and computing, for the diverse phases of the project, namely Monte Carlo generation,

¹ Machine–detector–interface-induced backgrounds can potentially be important at FCC-ee; they are the subject of ongoing detailed studies and the current results show that they should not significantly affect the size of the data samples [3].

^a e-mail: clement.helsens@cern.ch (corresponding author)

Table 1 Run plan for FCC-ee. Source [1]

Run	\sqrt{s} (GeV)	\mathcal{L} ($10^{34}/\text{cm}^2\text{s}^{-1}$)	$\int \mathcal{L}$ (ab^{-1})	Statistics
Z	91.2	230	150	3×10^{12} Z decays (visible)
WW	160	28	12	10^8 W^+W^- events
ZH	240	8.5	5	10^6 ZH events
$t\bar{t}$	350, 365	1.8, 1.55	0.2, 1.5	10^6 $t\bar{t}$ events

detector simulation, event reconstruction and data analysis. In Sect. 5, we discuss the estimates and outline the main areas which we consider challenging. In Sect. 6, we sum up and conclude.

2 Typical workflows

The resource requirements obviously depend on the objectives to be pursued, which, in turn, determine the workflow to be followed. We can initially distinguish the following general cases:

1. Collision data reconstruction and analysis. This concerns running experiments or test-beam data processing. The reconstruction part is typically well defined and run only a few times (for re-calibration purposes or improved reconstruction algorithms). The analysis part is by nature chaotic and less standard, although it may contain well-defined phases, for example, the preparation of tuples for the final selections and fits.²
2. Full/fast Monte Carlo simulations, including digitisation, followed by reconstruction and analysis. This concerns current experiments and experiments being designed. Interpretation of test-beam data may also require simulation, at least to some extent. For running experiments, the reconstruction and analysis phase are the same as for collision data. The amount of simulated data required depends on the use case. For running or test-beam experiments, the amount of simulated data should be enough to make the associated statistical uncertainty component negligible. At LEP, a rule of thumb of ten times more Monte Carlo data than collision data was often used. In general, this would not be applicable at LHC, given the amount of resources taken by the simulation, except perhaps for studying specific background features in reduced portions of the phase space.
3. Parameterised Monte Carlo simulations, possibly followed by the analysis. This typically concerns experiments in the design stage, although the use of these techniques to interpolate between full simulation parameter points is not uncommon for collision data analysis, especially in searches for new physics. As in the previous case, the amount of parameterised simulated data should be enough to make the associated statistical uncertainty component negligible.

For each of these cases, we have to consider a number of variations resulting from the physics studies—or several detector concepts, for experiments being designed—with differ-

² The preparations of the analysis tuples by individuals or working groups have acquired an increasingly important role in HEP experiments in an attempt not only to homogenise the set of high-level variables to work with, but also to optimise the use of resources. In particular, optimisation of data I/O, a known bottleneck, may possibly be achieved by exploiting the experiment's own infrastructure. An example of this is the case of the ALICE analysis trains [4], which minimise I/O by applying a set of registered algorithms to the same event readout only once.

ent resource requirements. Good organisation of the different activities can certainly optimise the requirements, in particular by eliminating duplications.

3 Storage

The storage needs depend on the data format, and the persistency and redundancy requirements for the data. For a project at the design stage, redundancy is mostly connected to the optimisation of other resources, network and CPU (it might be, for example, more efficient to duplicate some data locally than to access them remotely); it will depend on the resources finally made available and their geographical distribution. Data persistency is also connected with the available resources and with the trade-off between the cost of recreating the data and the cost of storing them. For example, there is the tendency to keep the data used for a publication as a reference, although what is strictly needed is the recipe to reproduce them. Efficient bookkeeping of the configuration settings and software environments used for creating a data set would certainly allow the needs to be better balanced.

The data format depends on the *event data model*; different phases of the experiment will require different levels of detail and, in the initial steps, possibly different data structures. However, as soon as we come to describing the *physics content* of an HEP event, a set of standard observables can be defined.

In the computing model being set up for FCC, based on the FCCSW[5,6] framework, data at any level are described by the data structures provided by EDM4hep[7], a common event data model developed for future HEP experiments. This means that

- Full/fast simulation generates an EDM4hep output;
- Reconstruction algorithms understand EDM4hep input and write EDM4hep output;
- Parameterised simulation produces an EDM4hep output where the quantities have the same meaning as those from reconstruction. High-level reconstruction algorithms, such as vertex finding, should be applicable both to reconstruction output and parameterised simulation output;
- Analysis is run on EDM4hep files; in particular, the same analysis algorithms should be applicable to fully simulated and reconstructed events and to parameterised events resulting from parameterised simulation.

In FCCSW, full and fast simulation refers to the full and fast mode of Geant4[8] and includes also digitisation. The reconstruction algorithms are FCC specific or taken from key4hep[9,10]. Parameterised simulation is obtained with DELPHES[11].

3.1 RAW event sizes

The RAW format is the event format used to describe collision data and fully simulated data. The exact format is only available once the detector design choices are frozen. To estimate the RAW event sizes for experiments in the design phase, a baseline solution for a typical detector is needed. For FCC-ee, there are currently two such baseline solutions under study: CLD, an adaptation of the CLIC baseline detector, and IDEA, a new innovative detector concept for e^+e^- colliders. These two detector concepts are being studied in some details. Table 2 summarises the understanding at the time of writing based on [12].

It is evident from Table 2 that the technology choice can make a difference and more refined/innovative technologies may result in a very large amount of data. Ongoing software developments indicate that this is potentially problematic not only for the storage, but also

Table 2 Typical RAW event sizes in kB for the Z run for the two baseline detector solutions [12] and the ALEPH detector [13]; the contribution of the final states originating from the Z exchange (Z decays) is singled out from the expected total (all events)

Readout channels	CLD		IDEA		ALEPH	
	1.9 G		2.8 G		1 M	
Sub-detector	Z decays (kB)	All events (kB)	Z decays (kB)	All events (kB)	Z decays (kB)	All events (kB)
Vertex	1.3	62	1.3	62		
Tracker	1.4	102	500	595		
Calorimeter	230	920*	500	2000*		
Muon	0.03	0.75	0.03	0.75		
Total	233	1085	1001	2658	120	550

*For the calorimeters, reference [12] does specify numbers for the all events case, only for the IDEA pre-shower; the numbers are obtained by applying the same factor 4 expected for the IDEA pre-shower to all the calorimeters

for the computing needs of simulation and/or reconstruction. The numbers for the IDEA tracker, a high granularity stereo drift chamber, already include optimisation based on the use of FPGA to reduce the data sample by a factor 15 [12]. While there is a general belief that there is still room for improvement, if only by applying standard techniques such zero suppression, a range of 1–2 MB seems appropriate for the study at hand.

3.2 AOD event sizes

Analysis data objects format, or AOD, is the format used for analysis and therefore the output created by the reconstruction phase or by parameterised simulation. Table 3 shows the typical event sizes for different types of events processed through DELPHES in EDM4hep format. This is expected to be a good estimation of the typical event sizes after reconstruction of collision data or fully simulated events. From this table, a range of 5–10 kB per event seems appropriate for this study.

3.3 Storage requirements

The rough estimates for the event sizes provided in the previous sections can be used to estimate the amount of storage required at various stages.

3.3.1 RAW data and the event format for full simulation

The amount of RAW data expected for the FCC-ee runs based on the estimates discussed in Sect. 3.1 are given in Table 4.

As expected, the Z run is by far the most demanding in terms of storage and will be used as a reference in the following. It can be seen from the table that the Z run values are in the range of a few EB, i.e. of the order of the values expected for HL-LHC [16]. By the time FCC-ee is brought to operation, not before 2040, the amount of RAW experiment collision data should not therefore present a challenge and should be manageable with a simple evolution of the HL-LHC model.

The picture is somehow different when the full simulation needs are considered. To understand whether a detector choice has the potential to match the FCC-ee requirements in terms

Table 3 Typical FCC-ee event sizes, in kB, for different types of events processed through DELPHES in EDM4hep format (using the IDEA detector concept card with track covariance [14])

Process	\sqrt{s} (GeV)	Average size (kB)
$Z \rightarrow u\bar{u}, d\bar{d}, s\bar{s}$	91.2	4.9
$Z \rightarrow c\bar{c}$	91.2	5.2
$Z \rightarrow b\bar{b}$	91.2	5.5
$Z \rightarrow \tau^+\tau^-$	91.2	1.2
Z decays, ALEPH, AOD	91.2	9.0
Z decays, ALEPH, MINI	91.2	2.2
ZH inclusive	240	8.9
ZZ inclusive	240	6.6
W^+W^- inclusive	240	6.4
$t\bar{t}$	350	13

The values measured in ALEPH are taken from Ref. [13] and shown for comparison

The ALEPH MINI format only contained the information need for end-user analysis [15], in compressed form

Table 4 RAW data estimates for FCC-ee

Run	\sqrt{s} (GeV)	Statistics	RAW data
Z	91.2	3×10^{12} Z decays (visible)	3–6 EB
WW	160	10^8 W^+W^- events	0.1–0.2 PB
ZH	240	10^6 ZH events	1–2 TB
$t\bar{t}$	350, 365	10^6 $t\bar{t}$ events	1–2 TB

of systematic uncertainty control, very large samples of simulated data might be required, and this for several diverse detector solutions. However, as mentioned at the beginning of Sect. 3, the persistency requirements of the simulated data are different to those of the collision data, availability of the simulated data being strictly needed only for the time required by reconstruction runs. So, if the storage of RAW simulated data is potentially a challenge for the FSR preparation phase, an efficient strategy to optimise the storage needs over time will provide a means to mitigate the impact on resource requirements. This strategy should allow for the interplay between fast and full simulation.

3.3.2 AOD data samples

Based on the estimates discussed in Sect. 3.2, the amount of AOD data expected for the FCC-ee runs is given in Table 5. The amount of data expected for the **Z** run is of the order of tens of PB, which represents a considerable amount during the FSR preparation phase, requiring a dedicated strategy and resource management.

4 Computing resources

Estimating computing resources is more complicated because more unknowns, such as the evolution of the efficiency of the various codes, enter the game.

Table 5 AOD data estimates for FCC-ee

Run	\sqrt{s} (GeV)	Statistics	AOD data
Z	91.2	3×10^{12} Z decays (visible)	15–30 PB
WW	160	10^8 W^+W^- events	0.5–1 TB
ZH	240	10^6 ZH events	5–10 GB
t\bar{t}	350, 365	10^6 $t\bar{t}$ events	5–10 GB

Table 6 Time estimated to generate $q\bar{q}$, $\tau^+\tau^-$ and $\mu^+\mu^-$ events at the Z peak

Generator	Process	100k/core (s)	Z sample/core	Z sample/9000 HS06 (days)
Pythia8	$q\bar{q}$	148	4×10^9 s \sim 126 y	50–75
KKMCee	$q\bar{q}$	151	4×10^9 s \sim 126 y	50–75
KKMCee	$\tau^+\tau^-$	195	0.25×10^9 s \sim 8 y	3–4.5
KKMCee	$\mu^+\mu^-$	334	0.44×10^9 s \sim 14 y	5–7.7

The metrics for computing resources that is generally accepted is HEPSPec. Exact numbers of HEPSPec provided by a computing element (CE) depend on the detailed hardware configuration, which is impossible to know at this stage. In the following, we will assume that one core of today's CERN OpenStack CE brings 10–15 HEPSPec. The current computing resources assigned to FCC at CERN amount to 9000 HEPSPec which we will also refer to as a *computing unit*.

The $q\bar{q}$ events at the **Z** run centre-of-mass energies are typically the most demanding case and will be used as a reference, together with the only currently published example of an FCC-ee case study full analysis involving rare b-mesons decays [17]. This can offer indications of the impact of the reconstruction and/or the analysis phases on the computing resources required.

The numbers shown in the remainder of the study have been obtained by running benchmark codes on a CERN OpenStack machine with 16 cores, 32 GB RAM [18].

4.1 Monte Carlo generation

The real time taken for the generation of 100k $q\bar{q}$ events with Pythia8[19] and with KKMCee[20] at the **Z** run centre-of-mass energies is shown in Table 6; the two reference generators give similar results. For comparison, the table also shows numbers for the generation of $\tau^+\tau^-$ and $\mu^+\mu^-$ events with KKMCee, which, per event, are up to a factor two larger than those for $q\bar{q}$.

Table 6 also shows the time required to generate a sample equivalent to that expected from the **Z** run on a single CERN core and using the computing resources currently assigned to FCC at CERN. We have already seen that this step is challenging and full-scale production requires an optimised use of resources. Of course, an efficiency optimisation of the Monte Carlo codes is also a possibility to be taken into account.

One additional comment relates to the use of a dedicated generator for the decay of heavy quarks, such as EvtGen [21], which was used for the analysis in Ref. [17]. These analyses require exclusive samples, currently obtained by filtering away unwanted decays. Since the number of events to be generated is not large, the inefficiency of this technique is not currently

Table 7 Time estimated to simulate $q\bar{q}$ events at the Z peak

Process	1k/core	Z sample/core	Z sample/9000 HS06
$q\bar{q}$	20k s = 5 h 33 min	6×10^{13} s $\sim 1.9 \times 10^6$ y	$2.1\text{--}3.2 \times 10^3$ y

a limitation, but it represents a waste of CPU usage for rare hadrons (such as B_c) as most of the events generated are skipped. Improvements in the efficiency of the filtering technique are open areas of work.

4.2 Detector simulation

The full simulation time per event in the CLD detector is approximately 20 seconds per hadronically decaying Z boson; the same number for IDEA cannot be derived yet. The CLD number is similar to the time required to simulate $t\bar{t}$ events at ATLAS or CMS once the average multiplicity scaling is taken into account, so it can be considered a realistic estimation of what can be done with current simulation techniques.³ Table 7 shows the projected integral time estimates. Considering the full statistics at the Z it becomes clear that the current computing resources are largely insufficient as it would take 2 to 3 thousand years to simulate it.

As can be seen in the table, the computing resources to simulate a full statistics equivalent of the Z data sample in 2–3 years (the expected duration of Z and also the preparation time for the FSR) are about 10 million HEPspec, which is of the order of the resources available to the LHC experiments today [22].

This can be seen in two ways. On the one side, producing the full statistic samples during the FCC-ee operations, although resource demanding, is not likely to be an issue; the computing resources available to FCC-ee will be at least equivalent to those available for HL-LHC. On the other side, for the FSR, it will certainly be impossible to have full statistics samples in full simulation of all the detector concepts.

When the fast simulation option is enabled in Geant4, the response of the sub-detectors is parameterised and the particle transport simplified. CMS has shown that applying these techniques to the calorimeter system, an acceptable precision in the description of the detector response can be kept with an overall speed-up by a factor of about 10. This is still not enough for full statistics samples for the FSR, but it goes in the right direction and the FCC community is certainly looking with interest at the Geant4 team efforts to improve the quality and speed of the fast simulation option.

Alternative approaches to reduce the impact of detector simulation on the overall simulated event processing budget include methodologies of partial detector simulation, such the one adopted by LHCb, e.g. not to simulate Cherenkov processes when the physics channel studies do not use that information, or new approaches to detector simulation, such as those based on deep machine learning technologies, which start to have a role at the LHC [23], with promising results.

4.3 Reconstruction

The event reconstruction is expected to be less busy at FCC-ee than at FCC-hh and (HL-)LHC, particularly for the tracks as the multiplicity is orders of magnitude smaller than

³ Any improvement in the speed of simulation coming from the current extensive fast simulation R&D is extremely welcome, although unlikely to change these numbers significantly in the timescale of the FSR.

Table 8 Time estimated to simulate $q\bar{q}$ events at the Z peak with DELPHES

Process	100k/core	Z sample/core	Z sample/9000 HS06
$q\bar{q}$	212 s	6.4×10^9 s \sim 202 y	0.22–0.34 y

Table 9 Number of $q\bar{q}$ events that can be produced per day with one computing unit and with the equivalent of the ATLAS computing resources

	Generation	Simulation	Reconstruction	DELPHES
Computing unit	$3.5\text{--}5.2 \times 10^{10}$	$2.6\text{--}3.9 \times 10^6$	$5.2\text{--}7.8 \times 10^6$	$2.4\text{--}3.6 \times 10^{10}$
ATLAS equivalent	$3.5\text{--}5.2 \times 10^{13}$	$2.6\text{--}3.9 \times 10^9$	$5.2\text{--}7.8 \times 10^9$	$2.4\text{--}3.6 \times 10^{13}$

at the LHC, thus greatly simplifying the pattern recognition. For comparison, at ALEPH the reconstruction step took about 10–15% of the simulation time [13], while at Belle-II, it accounts for about a third of the total processing time [24] and is dominated by tracking and depends on the amount of background. For FCC-ee, it is therefore reasonable to assume that the reconstruction time could potentially lead to a maximum comparable to half the simulation time discussed in Sect. 4.2.

4.4 Detector parameterisation

FCC studies use DELPHES for a fast parameterised simulation of the detector concepts. The DELPHES processing of 100k $q\bar{q}$ events generated with Pythia8 at the Z takes 212 seconds on a single-core CERN machine. Table 8 shows that using the CERN computing unit, between 2.5 and 4 months would be needed to produce the full Z statistics.

4.5 Analysis

Quantifying the needs for physics analyses depends on the use case. To illustrate it, we will focus on one recently published analysis using all the common tools [17]. This example focuses on precisely measuring a rare b-hadron decay, and tight cuts need to be applied to achieve excellent signal purity. In order to achieve an accurate estimation of the backgrounds, it was not possible to generate the expected inclusive data statistics. About 10 billion events, including exclusive decays with larger acceptance, were generated and reconstructed with DELPHES in the EDM4hep data format occupying approximately 50 TB of disk space. Analysing these events to create small ntuples with all the heavy calculations done (vertexing, candidate building, etc.) with the current CERN FCC batch resources takes half a day. The second step with the final selection can be achieved locally within less than an hour.

5 Ways ahead

Table 9 summarises the number of events which could be produced per day with one computing unit and with the equivalent of the current ATLAS computing resources.

In projecting the numbers summarised in Table 9, we have to consider the two cases of the FCC-ee operation and of the preparation of the FSR separately, as already done in some cases above. Since the FCC-ee operation will come after the HL-LHC experience, there is

little doubt that the resources required, in terms of both storage and computing, should be affordable.

The situation is different concerning the studies for the FSR. At the time of writing, the Physics Performance group has 33 case studies to be addressed. Projecting the resources needed by one of these cases [17], and assuming that all that can be shared between case studies is effectively shared, i.e. that all removable duplications are removed, it is probably safe to say that a close-to-full expected statistic is possible at the parameterised simulation level. However, a full simulation for each detector concept would not be possible.

Based on these considerations, we see that the following main areas should be addressed: improvement of the parameterised simulation; the interplay of full/fast/parameterised simulations; the minimal needs in terms of simulation statistics.

5.1 Improving the parameterised simulation

The tracking description of the parameterised simulation with DELPHES has been considerably improved during the studies following the publication of the FCC CDR. A detailed fast simulation of the tracks, including track covariance, allows much more detailed and realistic studies of tracking algorithms and therefore of observables related to tracking, such as vertexing or heavy flavour tagging. This has required the introduction of geometrical concepts in DELPHES, though possibly in a simplified version. The question is whether the same kind of approach could be used for other parts of the detector. The obvious first candidate is the calorimeter, where full simulation results, including spatial development of showers, could be parameterised and applied directly at DELPHES level. Similar approaches could be envisaged for other detectors, such as muon chambers, Cherenkov detectors or inner detectors. The separation barrel/forward and insensitive region (cracks) simulation could potentially be improved without a large impact on the processing time. Parameterised simulation of multiple scattering (for charged particle propagation to the calorimeters), at least in the inner detectors could also be implemented, as well as parameterisation for detached vertexes.

5.2 Interplay of full/fast/parameterised simulations

Somehow connected with the improvements of the parameterised simulation discussed in Sect. 5.1 is the interplay between the different levels of simulation. If resources are only available for limited studies in full simulation, it is important at best to use these studies to feed the better understanding back to parameterised simulation, or to derive the results of the studies. These techniques were already used for the CDR to understand the needs in terms of detector performance for a given measurement: the relevant detector response was studied in full simulation and translated into the impact on the result with parameterised simulation [25].

5.3 Minimal needs in terms of simulation statistics

In Sect. 2, it is mentioned that a rule of thumb for the requirements of the case studies in terms of the simulation statistics was *at least equivalent to the expected data sample*. We saw in the previous sections how difficult is to have samples satisfying this requirement for detailed simulations. There is therefore the need to go beyond the rule of thumb and develop systematic evaluation technologies which could be statistically more powerful, thereby reducing the number of events required. Developing these could also be very useful for the analysis of the collision data, when they come.

6 Conclusions and outlook

In this study, we have started investigating the computing needs of FCC-ee in view of the next phases of the project and operation. We have shown how, probably without surprise, these requirements are dominated by the **Z** run. We have also shown that despite being large, the requirements for storage and computing resources should not pose problems during the operation of the machine, planned to start after HL-LHC. Given this timescale, there is the possibility to benefit from all the advances, developments and findings of HL-LHC, including the resource sustainability aspects. Finally, we have shown how the preparation of the FSR for the next European Strategy Upgrade is potentially challenging, and will require the experiment groups to develop ways to optimally use and manage the data samples available, *de facto* increasing their statistical power and reducing the effective resource needs. After all, scarcity of resources is often behind the birth of brilliant ideas.

Funding Open access funding provided by CERN (European Organization for Nuclear Research).

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. FCC Collaboration, FCC physics opportunities. *Eur. Phys. J. C* **79**, 474 (2019), Table 2.1
2. The FCC Physics and Performance GitHub repository. <https://github.com/HEP-FCC/FCCEePhysicsPerformance>
3. Y. Voutsinas et al., *FCC-ee Interaction Region Backgrounds*, talk presented at 4th FCC Physics and Experiments workshop, November 2020. <http://indico.cern.ch/event/932973/contributions/4075874/>
4. M. Zimmermann, The ALICE analysis train system. *J. Phys. Conf. Ser.* **608**(1), 012019 (2015). <https://doi.org/10.1088/1742-6596/608/1/012019>
5. FCCSW, A software framework for FCC Physics and Detector studies. <https://github.com/HEP-FCC/FCCSW>
6. J. Cervantes et al., *EPJ Web Conf.* **245**, 05018 (2020). <https://doi.org/10.1051/epjconf/202024505018>
7. The EDM4hep *Common Event Data Model* Project. <https://github.com/key4hep/EDM4hep>
8. J. Allison et al., *Nucl. Instrum. Meth. A* **835**, 186–225 (2016). <https://doi.org/10.1016/j.nima.2016.06.125>
9. The key4hep Project. <https://github.com/key4hep/>
10. A. Sailer, G. Ganis, P. Mato, M. Petrič, G.A. Stewart, *EPJ Web Conf.* **245**, 10002 (2020). <https://doi.org/10.1051/epjconf/202024510002>
11. J. de Favereau et al., *DELPHES 3*. *JHEP* **02**, 057 (2014). [https://doi.org/10.1007/JHEP02\(2014\)057](https://doi.org/10.1007/JHEP02(2014)057)
12. F. Grancagnolo, *Event Rates at Z-pole*, talk presented at 4th FCC Physics and Experiments workshop, November 2020. <http://indico.cern.ch/event/932973/contributions/4099063/>
13. D. Schlatter et al., *ALEPH in Numbers*, internal ALEPH note (1996)
14. Available at https://github.com/delphes/delphes/blob/master/cards/delphes_card_IDEA.tcl
15. R. Hagelberg, *Data Storage and Network in Offline analysis of ALEPH*, ALEPH note 92–164 (1992)
16. T. Boccali, Computing models in high energy physics. *Rev. Phys.* **4**, 100034 (2019). <https://doi.org/10.1016/j.revip.2019.100034>
17. Y. Amhis, C. Hulsens, D. Hill, O. Sumensari. [arXiv:2105.13330](https://arxiv.org/abs/2105.13330)
18. The CERN Openstack Private Cloud. <https://clouddocs.web.cern.ch/>
19. T. Sjöstrand, S. Ask, J.R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C.O. Rasmussen, P.Z. Skands, *Comput. Phys. Commun.* **191**, 159–177 (2015). <https://doi.org/10.1016/j.cpc.2015.01.024>

20. A. Arbuzov, S. Jadach, Z. Wąs, B.F.L. Ward, S.A. Yost, *Comput. Phys. Commun.* (2021). <https://doi.org/10.1016/j.cpc.2020.107734>
21. D.J. Lange, *Nucl. Instrum. Meth. A* **462**, 152–155 (2001). [https://doi.org/10.1016/S0168-9002\(01\)00089-4](https://doi.org/10.1016/S0168-9002(01)00089-4)
22. ATLAS Computing and Software—Public Results. <https://twiki.cern.ch/twiki/bin/view/AtlasPublic/ComputingandSoftwarePublicResults>
23. See, for example. <https://cds.cern.ch/record/2774959/files/ATL-SOFT-SLIDE-2021-314.pdf>
24. Thomas Kuhr, private communication
25. M. Aleksa et al. [arXiv:1912.09962](https://arxiv.org/abs/1912.09962)