Belle II Computing

Some more details

Ueda I.

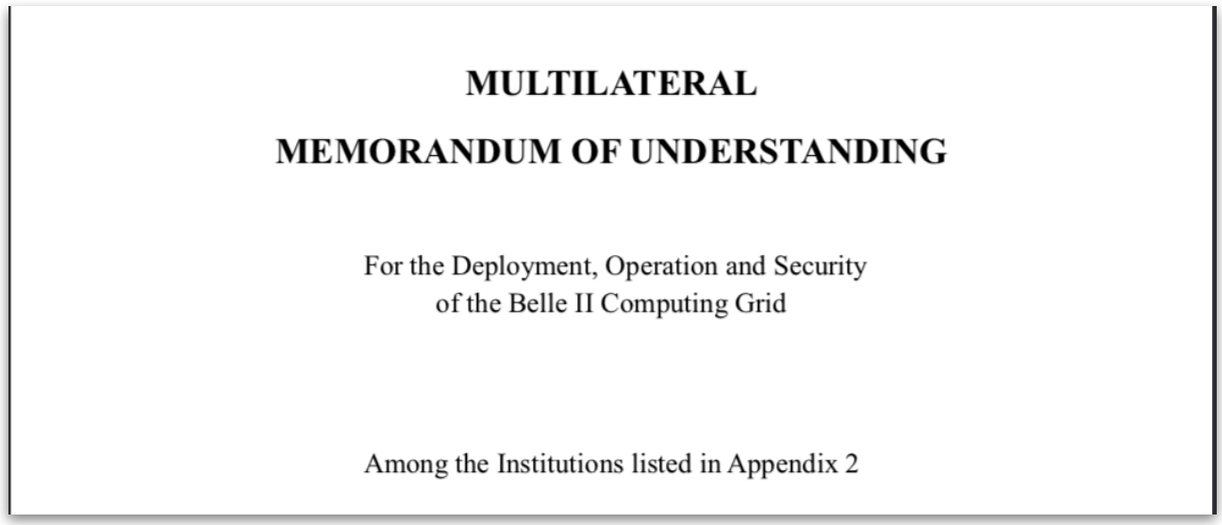
2020.Jan.15. Belle II France Computing Workshop

Political Matters

Belle II Computing MoUs

Service Level

- Multilateral Memorandum of Understanding for the Deployment, Operation and Security of the Belle II Computing Grid
- Managed by the KEK Computing Research Center (T. Nakamura)
- `https://wiki.kek.jp/display/belle2grid/MoU`



MULTILATERAL

MEMORANDUM OF UNDERSTANDING

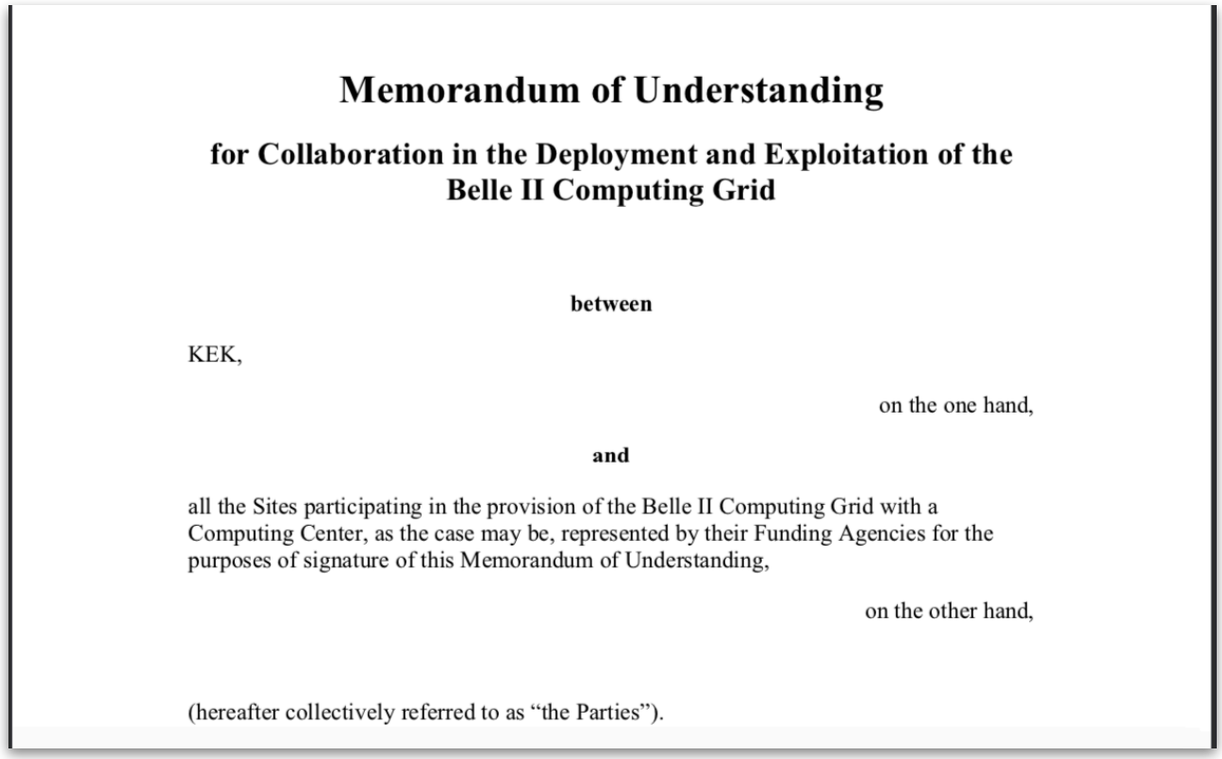
For the Deployment, Operation and Security
of the Belle II Computing Grid

Among the Institutions listed in Appendix 2

Computing Resources

- Memorandum of Understanding for Collaboration in the Deployment and Exploitation of the Belle II Computing Grid
  - between KEK and the individual sites
- Managed by the Belle II Computing Steering Group (F. Bianchi)
- `https://confluence.desy.de/display/BI/MoU+on+Computing+Resources`

Both very similar to the (parts of) WLCG MoU



Memorandum of Understanding

for Collaboration in the Deployment and Exploitation of the
Belle II Computing Grid

between

KEK,

on the one hand,

and

all the Sites participating in the provision of the Belle II Computing Grid with a Computing Center, as the case may be, represented by their Funding Agencies for the purposes of signature of this Memorandum of Understanding,

on the other hand,

(hereafter collectively referred to as "the Parties").

Belle II Computing Centers

- In Belle II we do not use the term "tier" in general (except in the Service Level MoU)

KEK - the host laboratory

- raw data and other forms of data on permanent mass storage
- for reconstruction (processing of raw data)
- also for Monte Carlo production and end-user analysis

Raw Data Centers

- raw and reconstructed data permanent storage
- for data-intensive analysis, re-processing of raw data
- also for Monte Carlo production and end-user analysis

Regional Data Centers

- partial copy of the processed data
- end-user analysis
- also for Monte Carlo production

Monte Carlo Production Centers

- Monte Carlo production and optionally end-user analysis

A single site may play multiple roles, depending on its size and resources availability

Some Technical Details on Storage Elements

Belle II Usage of Grid Storage Elements

Belle II uses DIRAC

- to manage its distributed computing resources

- Storage Elements can be accessed with
  - SRM, HTTPS/DAVS, or xrootd via GFAL2
  - Currently, mainly SRM (historical)
  - Some special SEs with HTTPS/DAVS, or GSIFTP

DIRAC requires space accounting

- SRM space token (historical)
- WLCG storage space accounting JSON

Belle II has been testing Dynafed

- HTTPS/DAVS has been requested and available at most sites
  - may become a requirement in near future (to replace SRM)
-

Belle II Usage of Grid Storage Elements

Multiple "endpoints" per SE

Or, sites may provide multiple SEs

- **`SiteName-DATA-SE`** for data distribution
  - `srm://storage.element.host:port/srm/managerv2?SFN=/base-path/`**`DATA`**
- **`SiteName-TMP-SE`** for output/temporary files
  - `srm://storage.element.host:port/srm/managerv2?SFN=/base-path/`**`TMP`**
- **`SiteName-TAPE-SE`** for raw data (only at the Raw Data Centers)
  - `srm://storage.element.host:port/srm/managerv2?SFN=/`**`another-base-path`**
- Single space token 'BELLE' for the both DATA-SE and TMP-SE
  - a separate space token for TAPE-SE
- WLCG JSON can be per base path but better be per endpoint

Storage URL = basepath + LFN

- Logical File Name (LFN) starts with /belle/
  - `/belle/MC/..., /belle/Data/..., ...`
- SURL
  - `srm://dcblsrm.sdcc.bnl.gov:port/srm/managerv2?SFN=/pnfs/sdcc.bnl.gov/data/belle/diskdata/DATA/belle/MC/...`
  - `srm://dcblsrm.sdcc.bnl.gov:port/srm/managerv2?SFN=/pnfs/sdcc.bnl.gov/data/belle/diskdata/TMP/belle/Data/...`
  - `srm://dcblsrm.sdcc.bnl.gov:port/srm/managerv2?SFN=/pnfs/sdcc.bnl.gov/tape/belle/Raw/...`

Belle II Usage of Grid Storage Elements

"Local SEs"

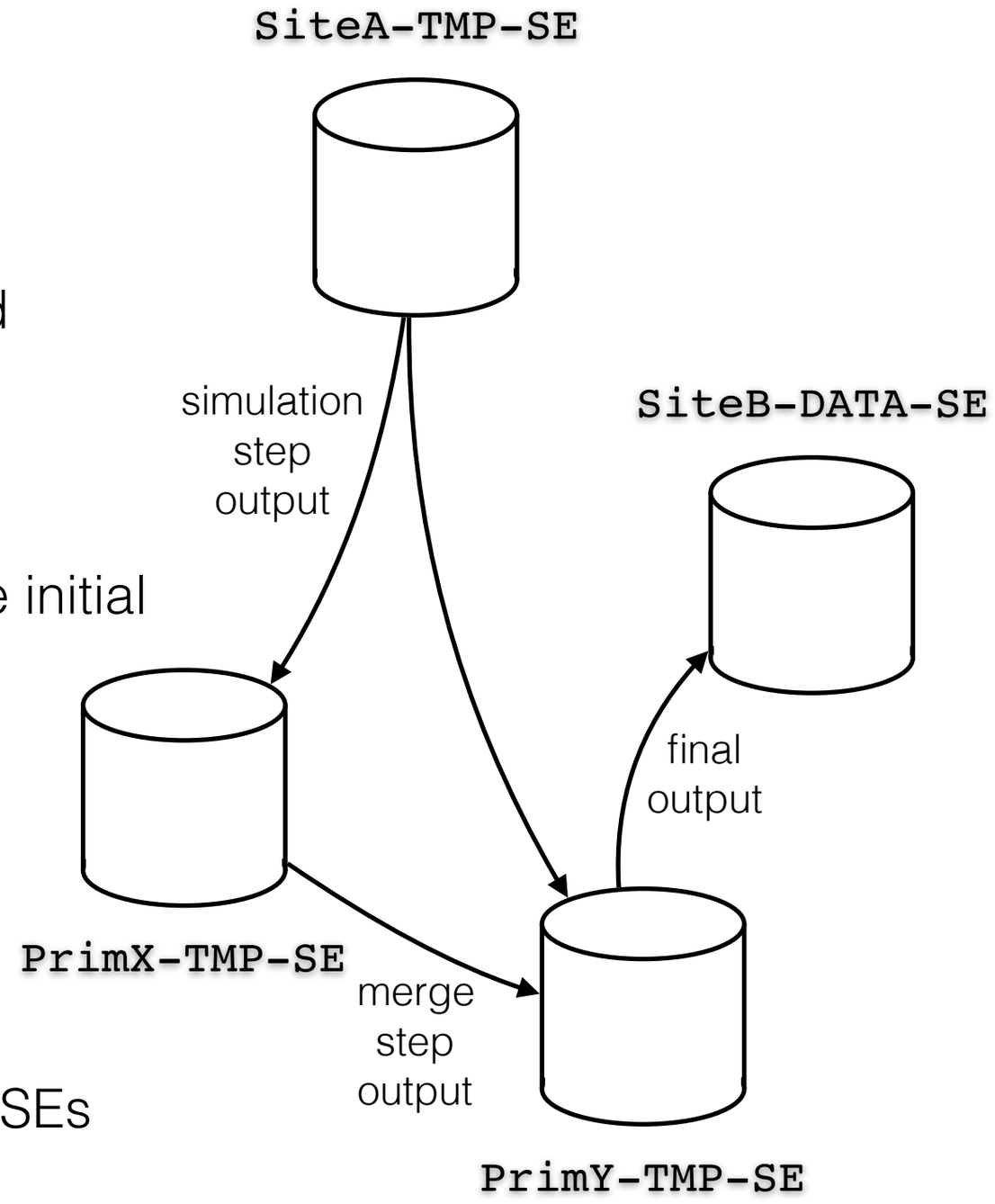
- Each site has an associated SE
  - Input data are taken from its DATA-SE
  - Output data are uploaded to its TMP-SE
- A non-grid site would use a near-by (and rather large) SE

"Primary SEs"

- A set of "large" and "stable" sites host the initial replicas (cf. Regional Data Centers)
  - on its TMP-SE
- "Merge" jobs to run at those large sites

Data distribution

- The "final" output (after merge steps) are replicated from the primary SEs to other SEs



Raw Data Management

Raw Data from Online to Grid

Online - DAQ/HLT

- stores raw data on the online storage in a custom temporary format (SROOT)
- releases a set of raw data files once in a while
  - after the end of each run, or when the online storage partition gets full

Offline - FrontEnd servers (aka Core Computing)

- copies raw data files from online disks, once they are released
- converts the raw data files (in the temporary DAQ format) into the permanent format (ROOT)
- registers the converted (permanent) raw data files to BelleRawDIRAC

Grid - resources/services managed by Belle II DIRAC

- BelleRawDIRAC uploads raw data files onto the grid Storage Element at KEK
- and registers uploaded files into the file catalogs (replica location and metadata)
- Belle II Distributed Data Management system (DDM) replicates raw data files to the Raw Data Center(s) using FTS

Raw Data Flow in bunches

Online-offline copies/conversion in bunches

- All the files of a run come only after the end of the run
- Many files to be put into FTS in a short time

Raw data files are grouped in "data blocks"

- Each data block replicated to a Raw Data Center
- A single Raw Data Center receives all the files in a time window
  - others will be waiting for next data blocks to come in the next time windows

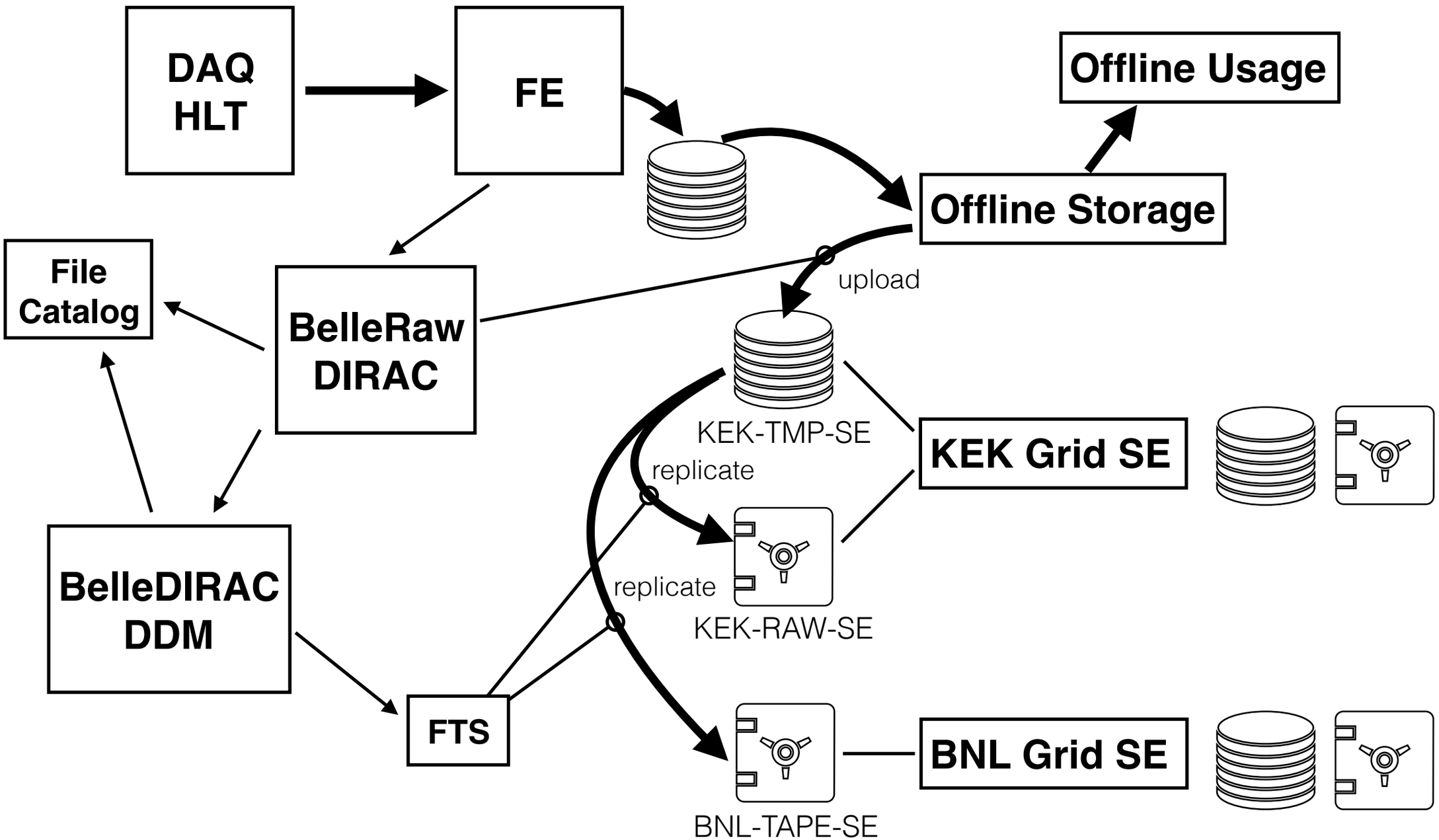
that means

Transfers of raw data files in bunches

- Data flow with peaks and intervals, rather than a constant flow
  - With the full luminosity, the export from KEK may be a constant flow,
  - and multiple Raw Data Centers may receive raw data (of different data blocks) at the same time, but we should not have all the links constantly busy all the time

Raw Data from Online to Grid

Now



Raw Data staging for Reprocessing

Raw data files stored on TAPE

- Files are to be staged from TAPE
- Job-by-job staging to be avoided
- Use of files on the buffer disk (in front of tape system) to be avoided
  - unless the "buffer" space is huge -- that is the case in KEK

Raw data files to be copied onto DISK for processing

- Raw data processing tasks defined for all the files of all the runs
- Raw data files copied from TAPE (`TAPE-SE`) to DISK (`TMP-SE`)
  - not all-in-one-go, but run-by-run (datablock-by-datablock)
- Processing jobs to be released for the datablocks copied onto DISK

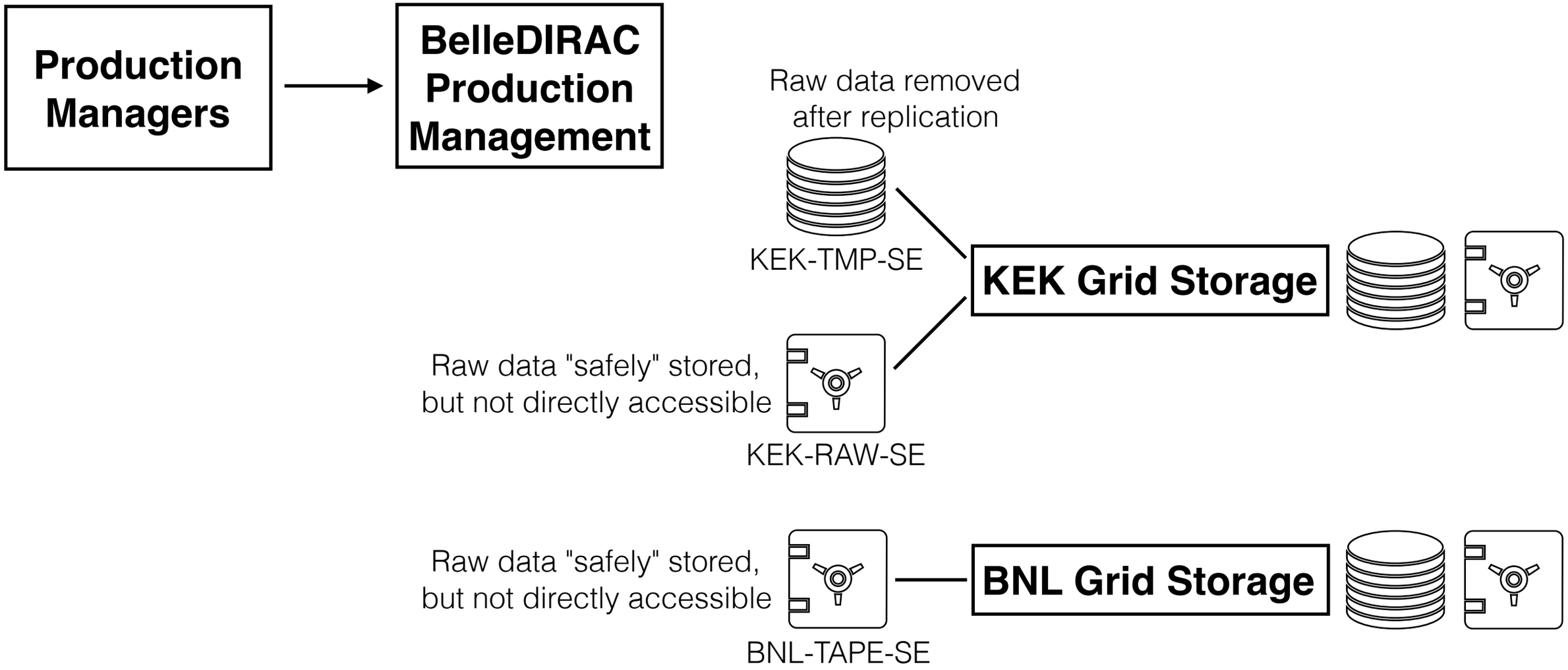
Raw data files to be removed from DISK after processing

- Raw data files to be removed from DISK, so that the files from the other runs yet to be processed can be staged
  - unless there is enough space to store all the files from all the runs

Raw Data for Processing Productions

Raw Data Processing

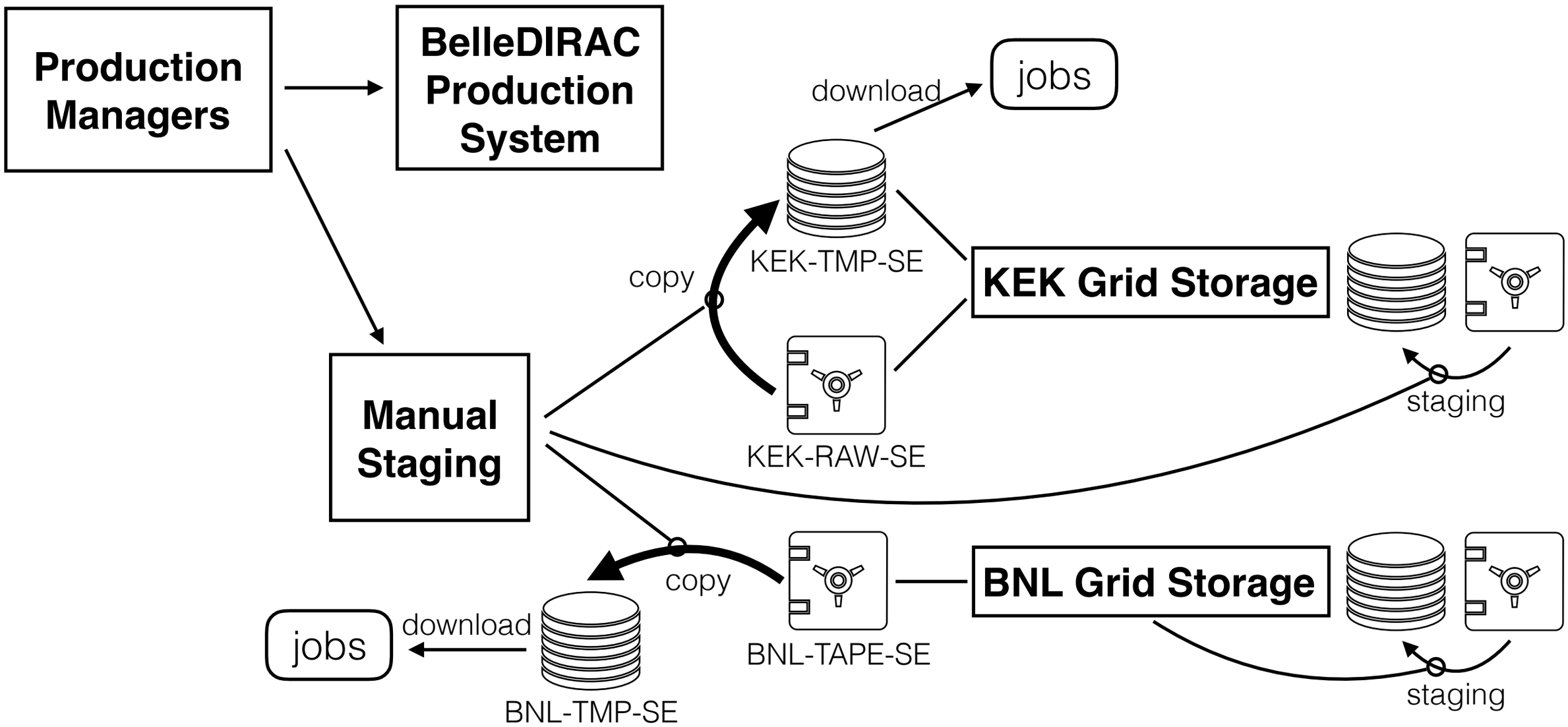
- **to be launched only when data are available on DISK**



Raw Data for Processing Productions

Raw Data Processing

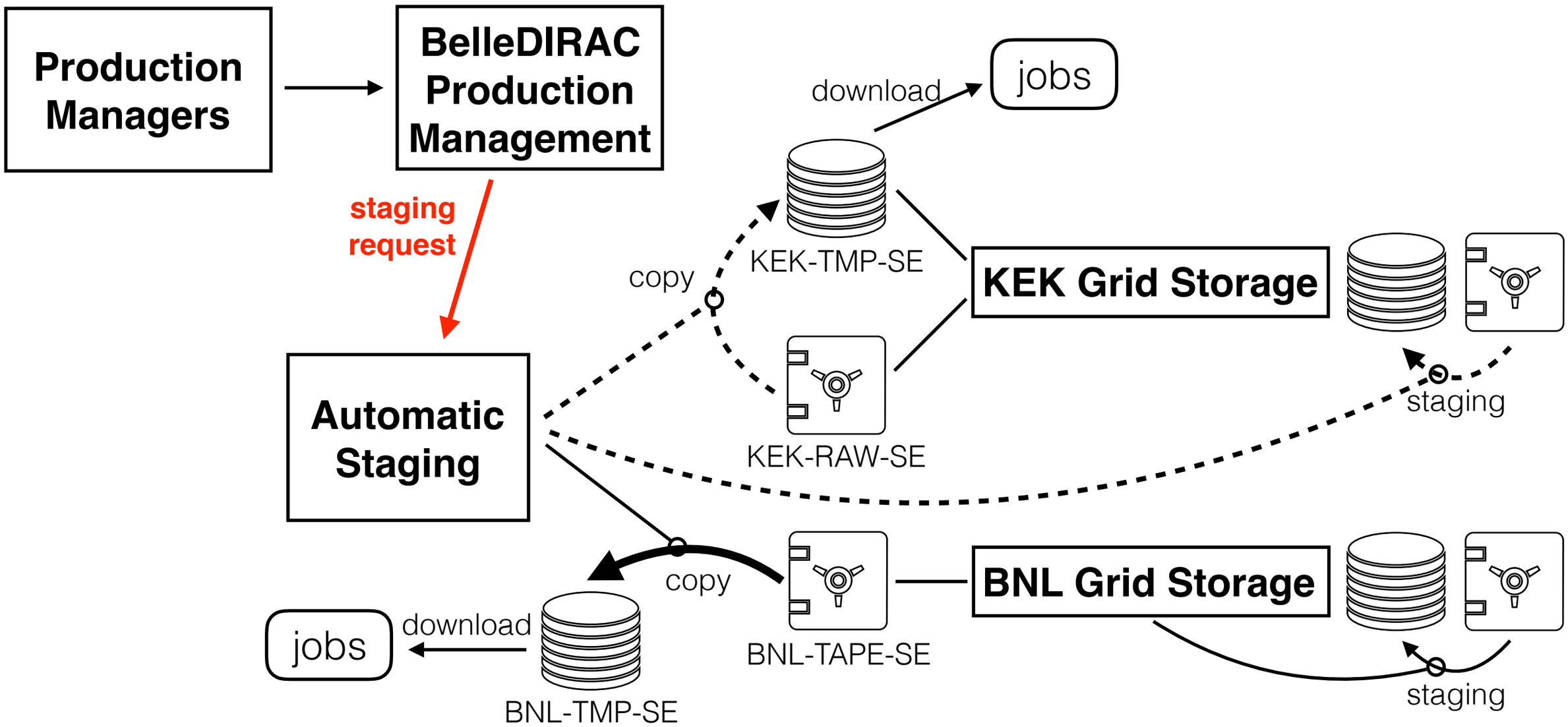
- to be launched only when data are available on DISK
- **"staging" need to be done run-by-run == not "all in one go"**



Raw Data for Processing Productions

Raw Data Processing in future (still a wish)

- Production Managers launch productions without waiting for "staging"
- **ProductionManagement system triggers staging**



Containerization of DIRAC jobs

Background for Containerization

- **OS upgrade**: the schedules for the sites and that for Belle II may differ
  - Some sites may need to upgrade before Belle II is ready
  - Other sites may need to run OSes that are obsolete for, or not supported by, Belle II
- **OS libraries**: DIRAC jobs expect pre-installed libraries
  - Sites may not have them all installed on WNs
- **End-user environment**: The current working model requires each user to install the Belle II grid client tools (gBasf2)
  - Some end-users face difficulties in installing gBasf2
  - Their platforms may not be supported by gBasf2
- **DIRAC** can...
  - DIRAC pilot jobs check the platform on the WNs, and DIRAC can assign adequate payload jobs to them,
    - but cannot do anything when the platform is not supported
  - DIRAC can send a wrapper job to launch Singularity and run a pilot job in it
    - `https://github.com/DIRACGrid/DIRAC/blob/integration/Resources/Computing/SingularityComputingElement.py`

Containerization of DIRAC jobs

- A container image with a selected OS and a set of required libraries can solve some issues
  - Belle II grid jobs can run on a selected platform
    - independent to the platform of the WNs
  - We can decide when we switch the OS on which we run our grid jobs.
- A container image with a proper gBasf2 installation may ease end-users
  - Some end-users may find it easier to launch singularity than installing gBasf2
    - NB. another idea is to prepare gBasf2 installation on cvmfs as the other experiments do
  - Some end-users may be happy to use their preferred platform, rather than being forced to use the supported OSes
    - and launch singularity to run gBasf2 client tools

Containerization - Possible Tasks

DIRAC SingularityComputingElement

- The current implementation does not expect the container image to include DIRAC installation.
  - It makes it possible to run a newer DIRAC release without renewing the container image, I suppose.
  - Although it is kind of wasting time in installing DIRAC every time.
- Task 1: to prepare a container image with CC7 (without DIRAC installation) and try utilizing SingularityComputingElement
- Task 2: to study usefulness of a container image with DIRAC installed

End-user environment

- An official Belle II grid environment in a container
- Task 3: to prepare a container image with gBasf2 installation and validate it as an end-user environment
- Task 4: to study feasibility of utilizing gBasf2 installation on cvmfs (if prepared)