Offline Interface

S. Y. Suzuki (KEK CRC)

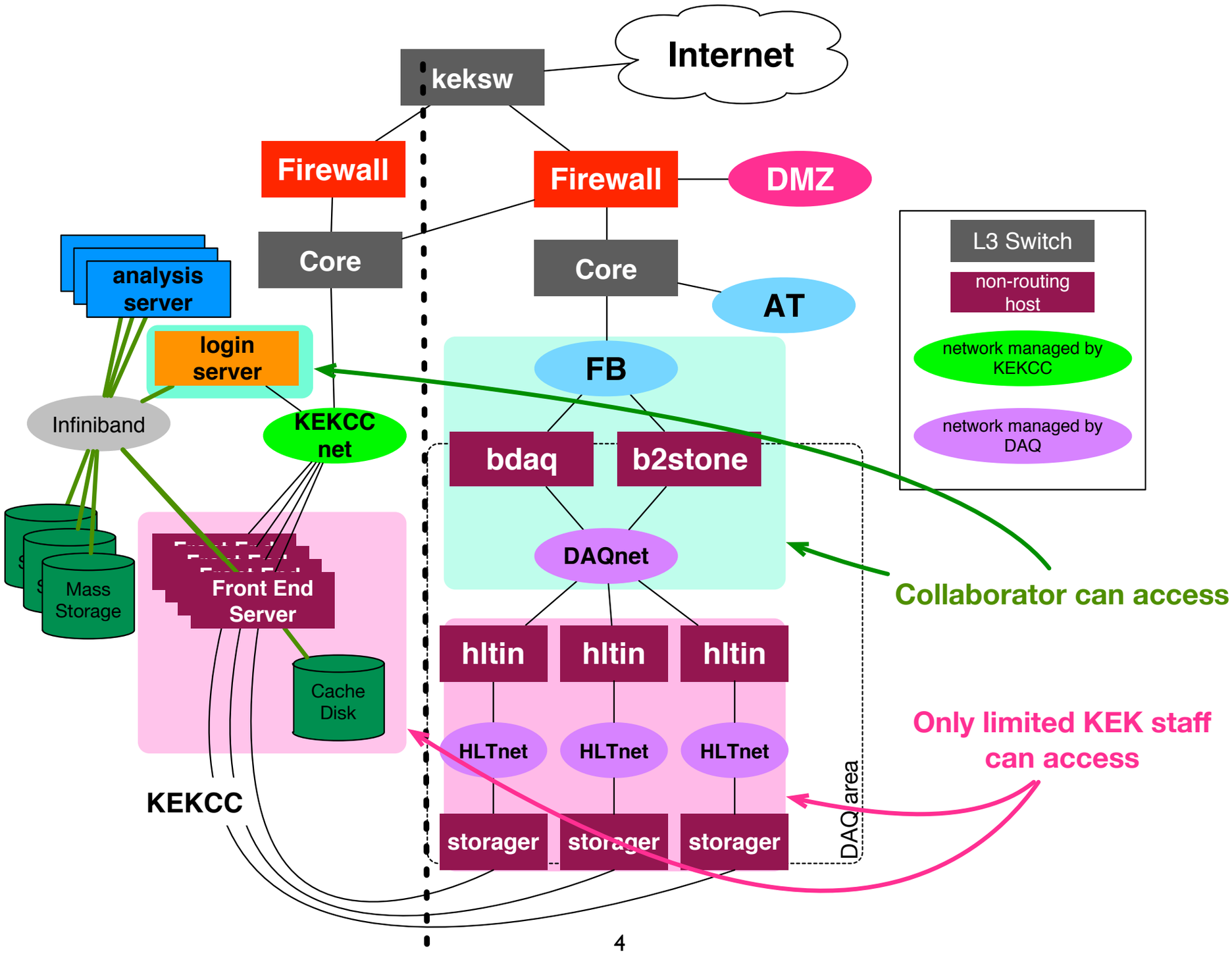
Preface

- Please look at slides for the last BPAC by Konno-san and by T.Hara-san

Status

- Finally official data transfer is started
- But it doesn't care data recorded without HLT

Rawdata from DAQ to KEKCC



Precondition

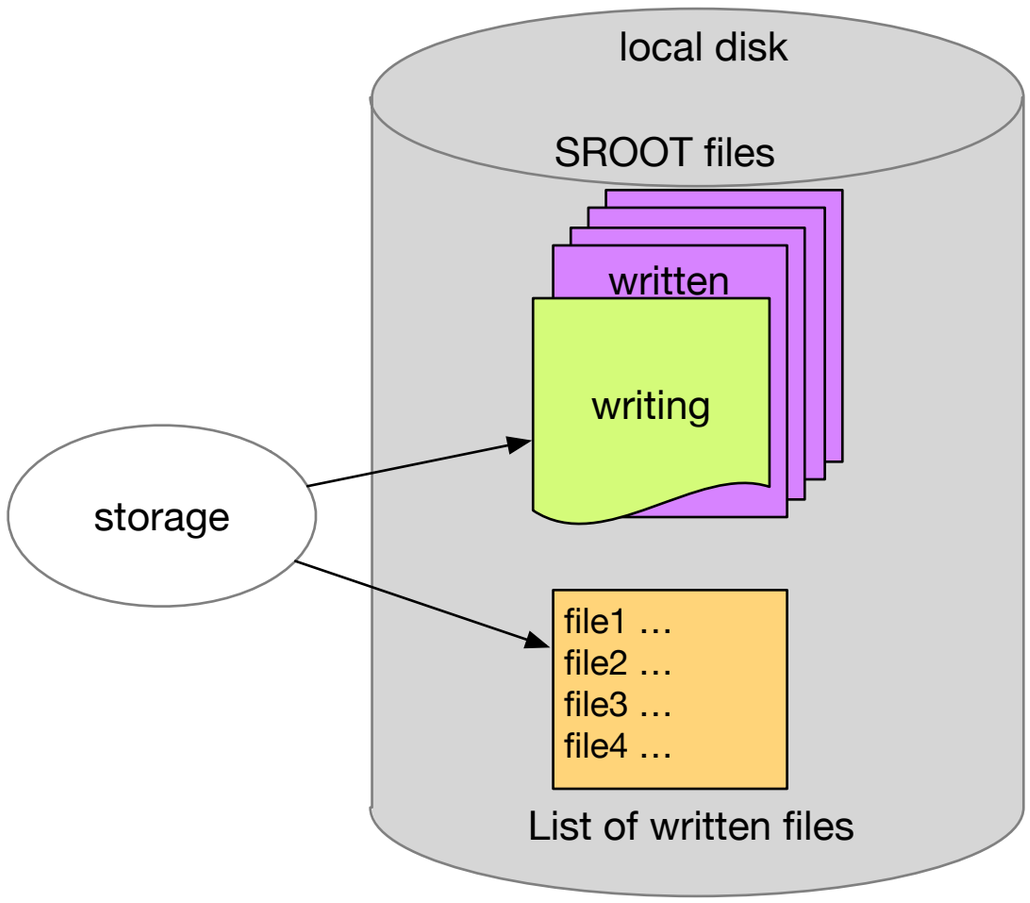
- KEK Security committee approves the direct path from Belle2DAQ to KEKCC, but SSH must not use the path.

B.F.E.

- The front-end servers in KEKCC for rawdata transfer are named as "bfe0X"
  - may mean Belle2 Front End server
- BFE in following slides points it
- currently KEKCC has 5 BFE

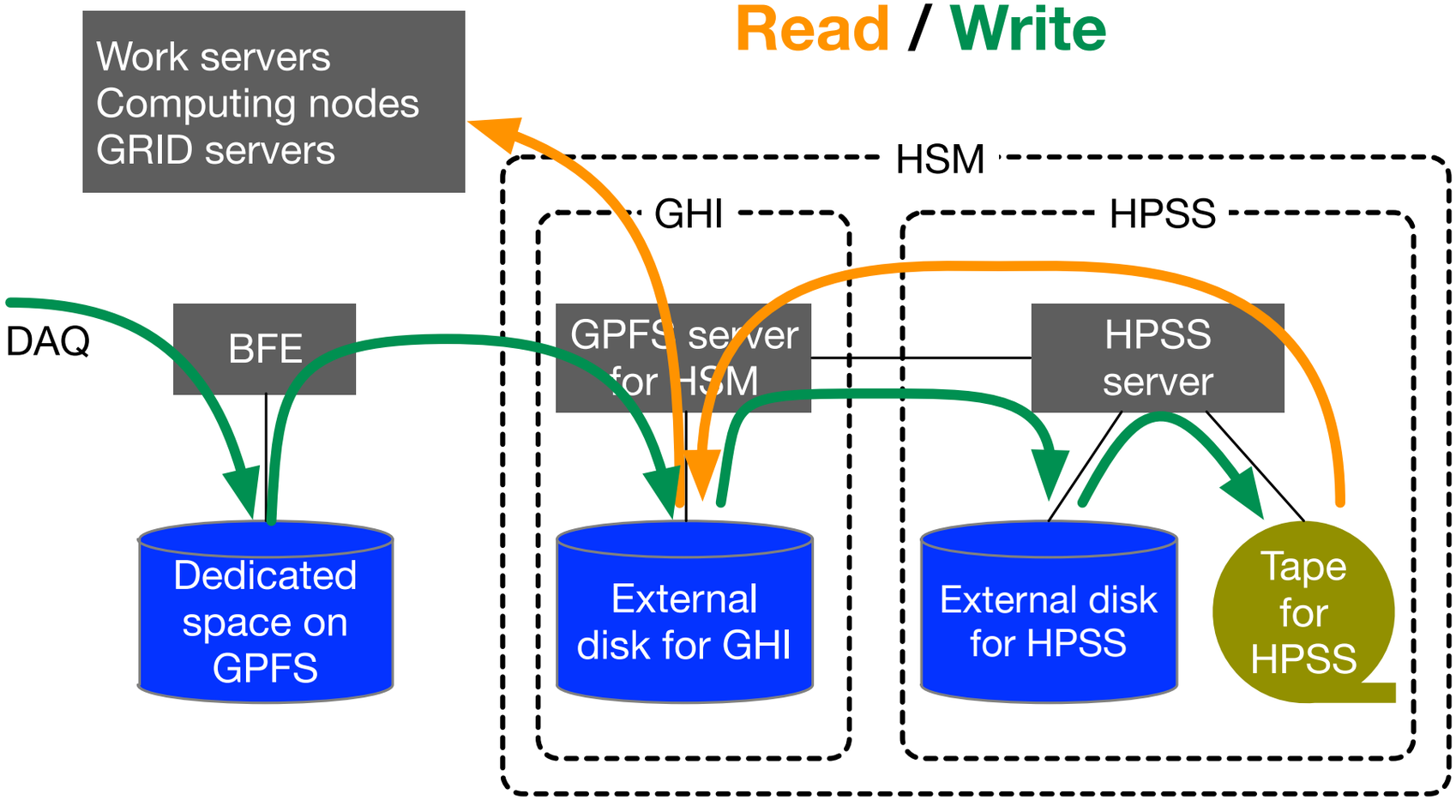
Files on HLT

- SROOT files
- file list
- maintained by storage program



Files on KEKCC

- Files in dedicated space are not accessible from other servers



Sequence on HLT

1. During writing a SROOT file,
    - update # of events, adler32 checksum
2. When it closes the file
    - records the filename and so on
3. When a partition is fully filled by data
    - exports the records as the list file
4. When HLT tells all files are successfully stored
    - removes the transferred files

Sequence on BFE

1. get the list of the SROOT files
2. copy files to the temporary area in GPFS
3. convert them to regular ROOT files.
   In the beginning stage, SROOT may be kept.
4. copy them to HPSS via GHI **(may not by BFE)**
5. confirm they are on tapes of HPSS **(may not by BFE)**
6. upload the list of the copied files to DAQ

Transfer by rsync

- DAQ prefers rsync for the file transfer to minimize the installation procedure.
  - without using ssh, bare protocol
  - achieves 240MB/s ~ 300MB/s
    - limited by internal checksum calculation (md5 + adler32 like)
- Any other?

```
top - 10:59:43 up 29 days, 19:07, 20 users,  load average: 1.51, 0.81, 0.62
Tasks: 846 total,   1 running, 844 sleeping,   1 stopped,   0 zombie
Cpu(s):  0.3%us,  2.2%sy,  0.0%ni, 97.1%id,  0.2%wa,  0.0%hi,  0.2%si,  0.0%st
Mem:  66072600k total, 65814352k used,   258248k free,    66384k buffers
Swap:        0k total,        0k used,        0k free, 38822952k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 5041 nobody    20   0 12256 1744 1080 D 71.6  0.0   0:51.27 rsync
  451 root      20   0     0    0    0 S  2.9  0.0  47:24.20 kswapd0
  452 root      20   0     0    0    0 S  2.9  0.0  52:31.14 kswapd1
```

```
cdc.0000.000934.sroot-17
  2,147,486,444 100%  285.20MB/s    0:00:07 (xfr#9, to-chk=965/990)
cdc.0000.000934.sroot-18
  2,147,492,720 100%  316.25MB/s    0:00:06 (xfr#10, to-chk=964/990)
cdc.0000.000934.sroot-19
  2,147,493,132 100%  304.04MB/s    0:00:06 (xfr#11, to-chk=963/990)
cdc.0000.000934.sroot-2
  2,147,489,076 100%  290.91MB/s    0:00:07 (xfr#12, to-chk=962/990)
cdc.0000.000934.sroot-20
  2,147,489,337 100%  317.82MB/s    0:00:06 (xfr#13, to-chk=961/990)
cdc.0000.000934.sroot-21
  2,147,489,735 100%  306.86MB/s    0:00:06 (xfr#14, to-chk=960/990)
cdc.0000.000934.sroot-22
  2,147,489,588 100%  296.38MB/s    0:00:06 (xfr#15, to-chk=959/990)
cdc.0000.000934.sroot-23
  2,147,492,830 100%  285.91MB/s    0:00:07 (xfr#16, to-chk=958/990)
cdc.0000.000934.sroot-24
  2,147,483,977 100%  316.00MB/s    0:00:06 (xfr#17, to-chk=957/990)
cdc.0000.000934.sroot-25
  2,147,487,697 100%  302.02MB/s    0:00:06 (xfr#18, to-chk=956/990)
cdc.0000.000934.sroot-26
  2,147,492,509 100%  291.57MB/s    0:00:07 (xfr#19, to-chk=955/990)
```

WDT by Facebook

- https://github.com/facebook/wdt
- difficult to build, dislike RHEL6
- dynamic high-port use, dislike FW
- transfer multiple files simultaneously, dislike slow seek HDD
- Not for our case

GridFTP

- widely used in offline field
- Unfamiliar for online experts in KEK
- Just skip it

GridFTP-Lite

- easy installation via yum
- based on SSH authentication
- doesn't provide any access restriction
- files must be explicitly specified
- dynamic high-port usage
- Possible, but enabling SSH is dangerous

bbcp

- developed by SLAC
- easy installation
- not updated recently
- needs remote command execution like SSH
- dynamic high-port use
- no access restriction
- similarly dangerous, but achieves 1GB/s

```
bbcp: Creating ./disk01/storage/ecl.0000.000567.sroot-382
bbcp: 170429 00:01:23  65% done; 1.1 GB/s, avg 1.1 GB/s
File ./disk01/storage/ecl.0000.000567.sroot-382 created; 2147489157 bytes at 1.1 GB/s
bbcp: Creating ./disk01/storage/ecl.0000.000567.sroot-18
bbcp: 170429 00:01:25  71% done; 1.1 GB/s, avg 1.1 GB/s
File ./disk01/storage/ecl.0000.000567.sroot-18 created; 2147496249 bytes at 1.1 GB/s
bbcp: Creating ./disk01/storage/ecl.0000.000518.sroot-79
bbcp: 170429 00:01:27  73% done; 1.0 GB/s, avg 1.0 GB/s
File ./disk01/storage/ecl.0000.000518.sroot-79 created; 2147510043 bytes at 1.0 GB/s
bbcp: Creating ./disk01/storage/ecl.0000.000544.sroot-704
bbcp: 170429 00:01:29  76% done; 1.0 GB/s, avg 1.0 GB/s
File ./disk01/storage/ecl.0000.000544.sroot-704 created; 2147517359 bytes at 1.0 GB/s
bbcp: Creating ./disk01/storage/ecl.0000.000544.sroot-1099
bbcp: 170429 00:01:31  76% done; 1.0 GB/s, avg 1.0 GB/s
File ./disk01/storage/ecl.0000.000544.sroot-1099 created; 2147508819 bytes at 1.0 GB/s
bbcp: Creating ./disk01/storage/ecl.0000.000518.sroot-250
bbcp: 170429 00:01:33  76% done; 1001.2 MB/s, avg 1001.2 MB/s
File ./disk01/storage/ecl.0000.000518.sroot-250 created; 2147515535 bytes at 1.0 GB/s
bbcp: Creating ./disk01/storage/ecl.0000.000544.sroot-133
bbcp: 170429 00:01:35  81% done; 1.0 GB/s, avg 1.0 GB/s
File ./disk01/storage/ecl.0000.000544.sroot-133 created; 2147488808 bytes at 1.1 GB/s
bbcp: Creating ./disk01/storage/ecl.0000.000518.sroot-299
bbcp: 170429 00:01:37  80% done; 1004.2 MB/s, avg 1004.2 MB/s
File ./disk01/storage/ecl.0000.000518.sroot-299 created; 2147510920 bytes at 1021.6 MB/s
bbcp: Creating ./disk01/storage/ecl.0000.000490.sroot-48
bbcp: 170429 00:01:39  85% done; 1.0 GB/s, avg 1.0 GB/s
```

Hence...

- Still we are using rsync without SSH and access restriction by rsync itself.
- 240 ~ 300MB/s is enough for single HLT.
- When the input is faster, we will enable multiple HLT operation

Summary

- Official rawdata transfer mechanism is ready.