Belle II production system

Rafal Grzymkowski (Institute of Nuclear Physics PAN)
Radek Ludacka (Charles University)
Hideki Miyake (KEK)
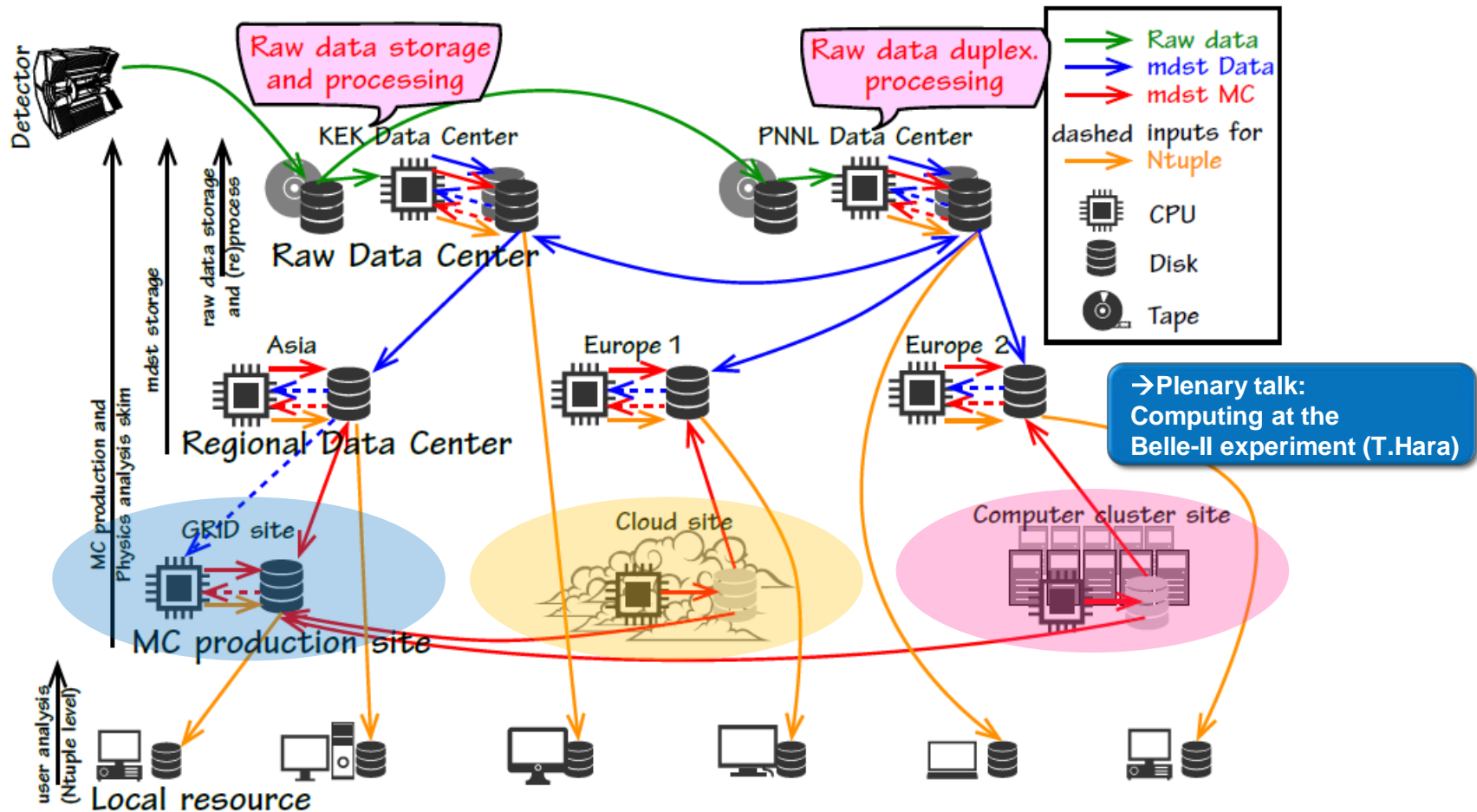Malachi Schram (PNNL)
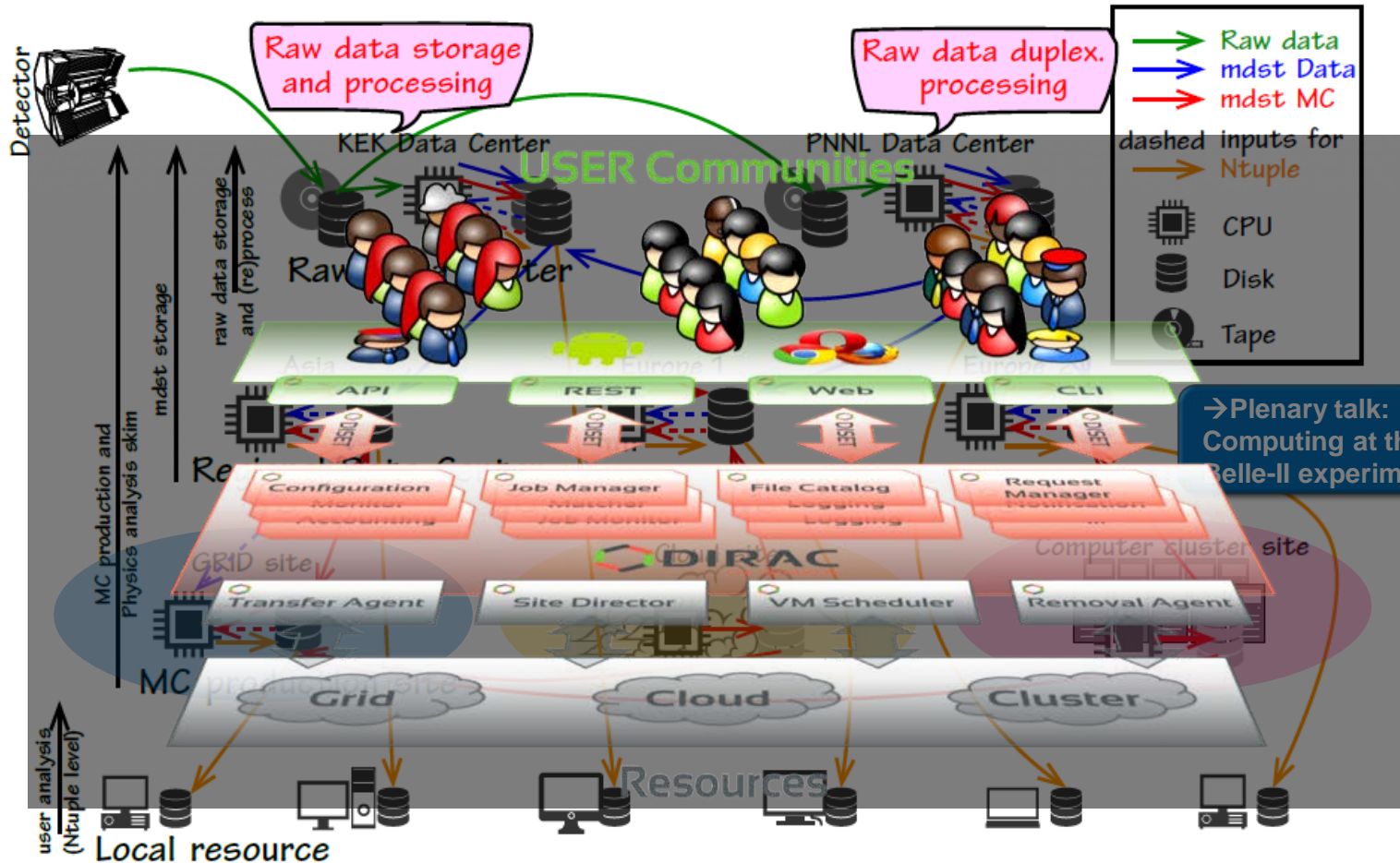
for Belle II collaboration

2015, April 13th

CHEP2015, OIST, Okinawa, Japan

Belle II computing model

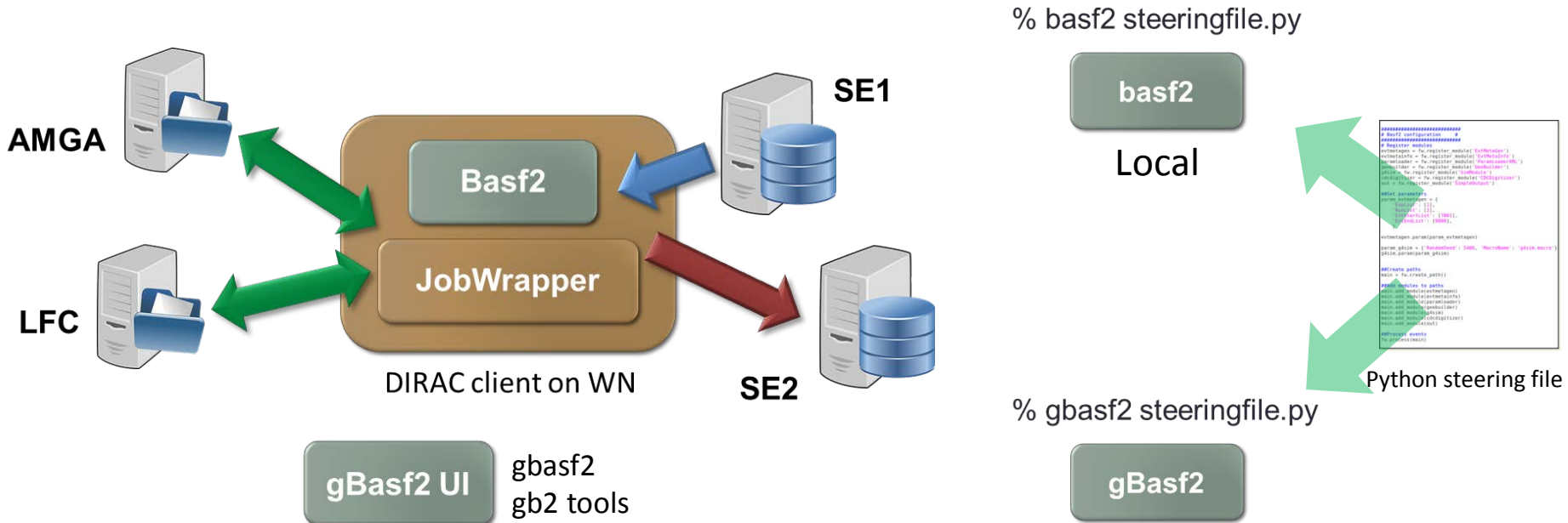- Variety types of resources: EMI, OSG, Cloud, local cluster…

Belle II computing model

- Variety types of resources: EMI, OSG, Cloud, local cluster…
  - → **DIRAC solution**

Basf2 and gBasf2

- Basf2 is our analysis software framework →**Poster session A233 (T.Kuhr)**

- Interface to distributed computing is given by gBasf2 (grid Basf2)
  - Provide transparent job execution on DC environment
    - Data input/output, file catalog/metadata registration...
  - Provide also collection of tools to handle job and data through DIRAC API (gb2 tools)

% basf2 steeringfile.py

basf2

Local

Python steering file

% gbasf2 steeringfile.py

AMGA

SE1

Basf2

JobWrapper

LFC

DIRAC client on WN

SE2

gBasf2 UI   gbasf2
gb2 tools

gBasf2

GRID

BelleDIRAC

- The extension module of DIRAC for Belle II
  - Not only simple extension but additional functionalities

DIRAC extension Extensions of base DIRAC (WMS, RMS, RSS, TMS...)
New: VMDIRAC, WebApp
→Poster session B342 (R.Grzymkowski)

gBasf2 Job submitter, job wrapper, command line tools

AMGA API, proxy service, access monitor
→Poster session B313 (G.Park)
→Poster session B466 (J.Kwak)

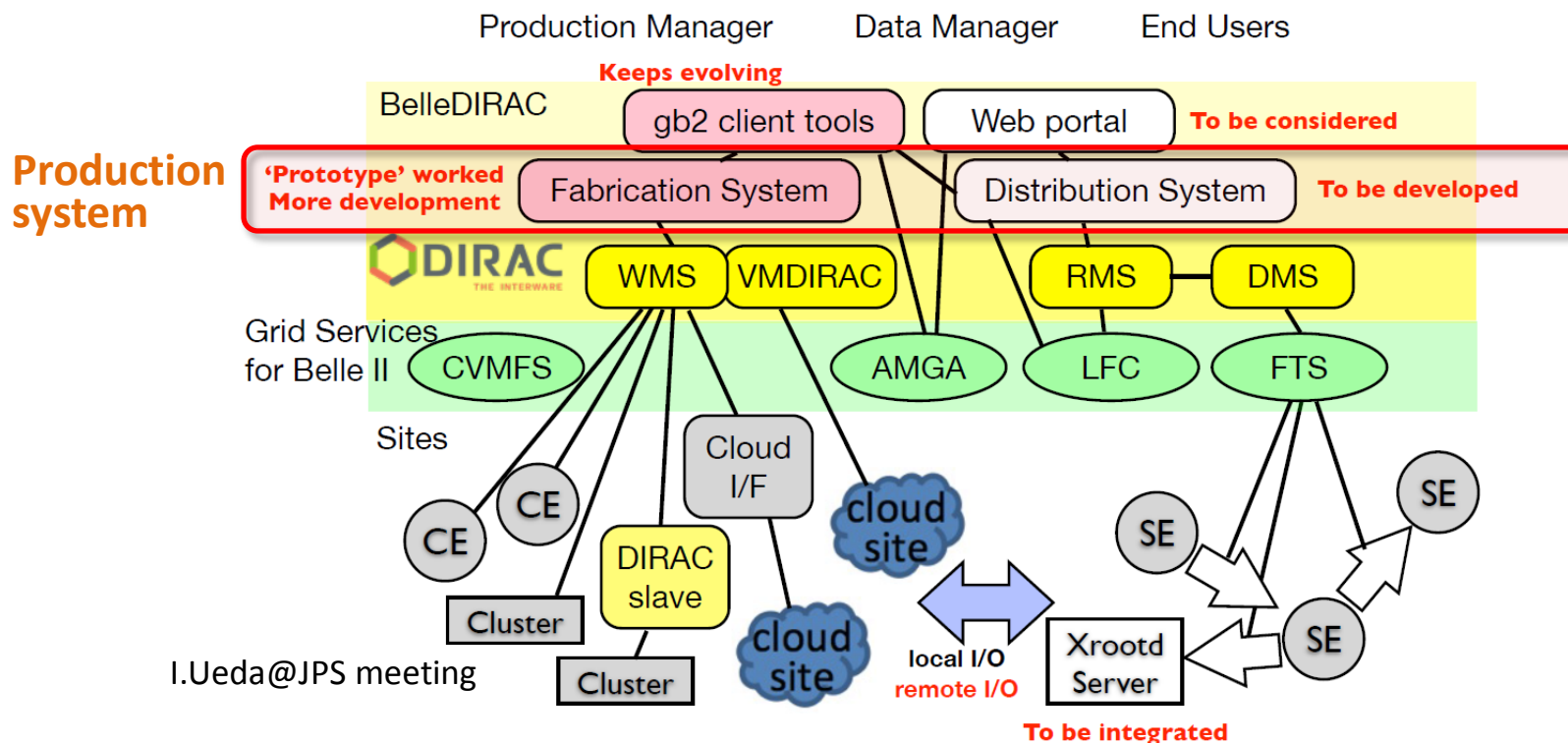**Monitoring** qstat wrapper, pilot submission, SE access
→Poster session A337 (Y.Kato)
→Poster session B314 (K.Hayasaka)

Production system

Big picture

- We should continuously handle many kinds of jobs
  - Key words: scalable, flexible, automatized
- Need an application layer for limited number of people to control the complicated workflow

Production system

- Based on DIRAC Transformation system (data-driven processing system)
- Consist of several subsystems

Management — Manage production workflows (controlled by UI and monitoring system)
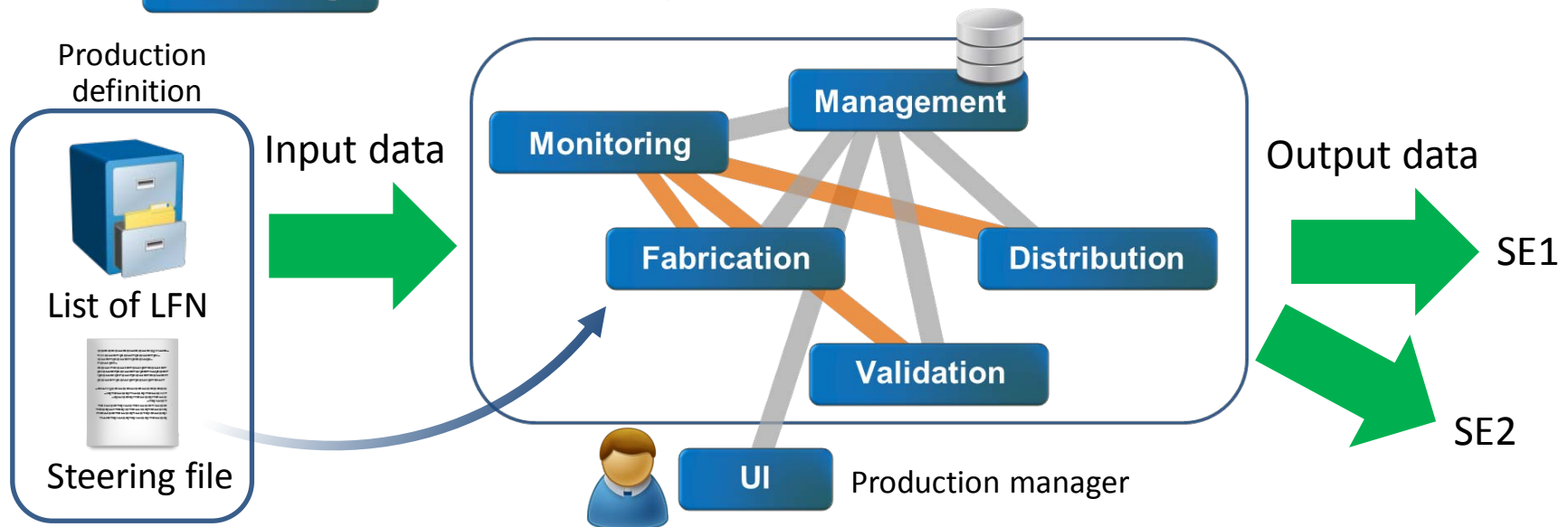
Fabrication — Handle steering file to produce output events

Validation — Evaluate produced events

Distribution — Determine adequate output storage and transfer the output

} A production

Monitoring — Check each subsystem status as well as distributed resources (CE, SE…)

Production definition

List of LFN

Steering file

Input data

Monitoring

Management

Fabrication

Distribution

Validation

UI — Production manager

Output data

SE1

SE2

Prototype implementation

- Still we are in the beginning of the development
- To evaluate our concept we have prepared initial test bench
- Dedicated for event fabrication (workload management)

- Features
  - **Production Management**
    - Registration with CUI (no bookkeeping yet)
    - Logging by summarized information

  - **Automatic validation of production definition**
    - Run small number of jobs with high priority (scout job)

  - **Automatic job submission**
    - Can choose each own submission rate based on the production priority

  - **Automatic resubmission of failed jobs**
    - Diagnose and log the errors
      → summarize statistical information
    - Cleanup failed jobs (remnant on SE, catalog, or request)

Production monitor

| Total Created	Submitted	Waiting	Running	Done	Completed	Failed	Stalled
11010 (+2000)	0 (-270)	3267 (-475)	3408 (+638)	4226 (+2052)	10 (+4)	0 (-23)	0 (-25)
11010 (+2000)	0 (-482)	3678 (+283)	762 (-67)	6537 (+2553)	10 (+7)	2 (+1)	2 (-18)
12010 (+2000)	0 (-532)	3361 (-97)	1507 (-189)	7054 (+2800)	18 (+16)	1 (-2)	1 (-4)

Failed jobs are automatically handled

```
Miyake@hpz800[8]% ./prod_analyser_site.py 556
Transformation          TransID  Site          Pilot/ Apps/Stall/Catlg/Input/Outpt/Other
mc46_test_new1_Touschek_HER  00556  ANY          26    74   207    0    1    8   86
                                    CLOUD.AWS_Singapore.  0    3   92    0    1    0    1
                                    CLOUD.AWS_Sydney.au   6   43 1356    0    1    0   10
                                    CLOUD.AWS_Tokyo.jp    0    0   70    0    0    0    0
                                    CLOUD.CC1_Krakow.pl   0   67   36    0    0   20    8
                                    DIRAC.BINP.ru         0  957 5158    0    0   21   94
                                    DIRAC.Beihang.cn      0   64    0    0    0    0    0
                                    DIRAC.Nara-WU.jp      0   11    3    0    0    1    1
                                    DIRAC.Niigata.jp      0   25    2    0    0    2    4
                                    DIRAC.Osaka-CU.jp     0   13    3    0    0    0    0
                                    DIRAC.PNNL.us         0   94   34    0    0   20   19
                                    DIRAC.Tokyo.jp        0    3    1    0    0    0    0
                                    DIRAC.UVic.ca         1 1761 4899    0    0  186  230
                                    DIRAC.Yamagata.jp     0    3    0    0    0    0    0
```
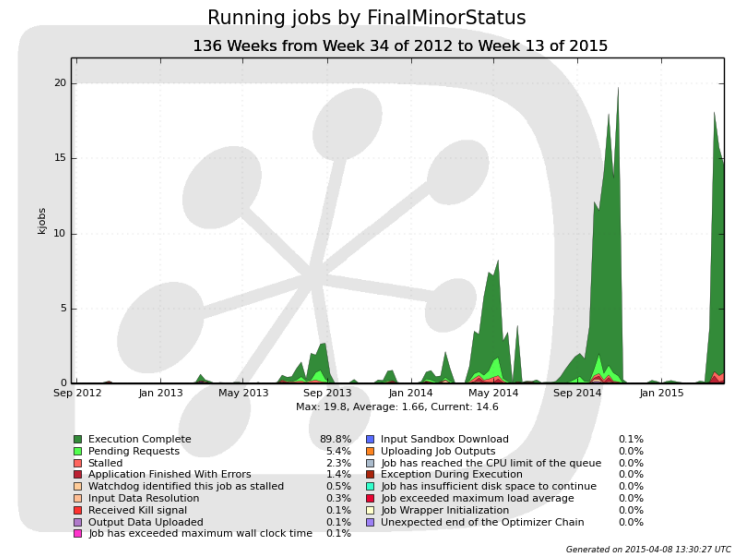
Production job failure reason per site

Performance

- ## Typically managing 15-25K jobs (about 150K HS06)
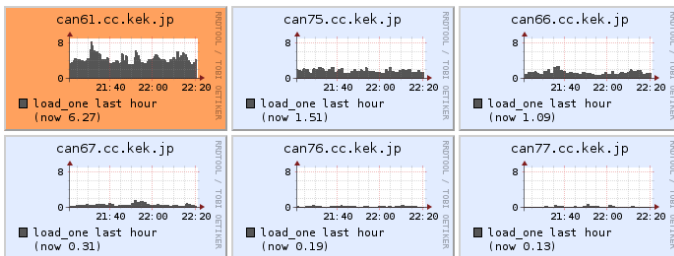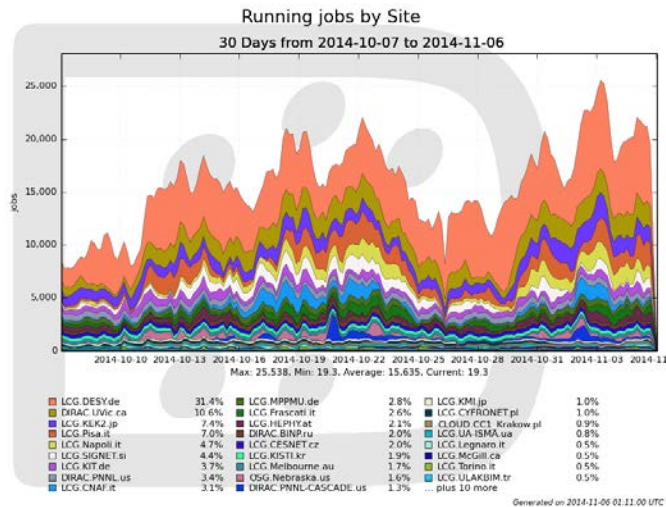
- "Campaign" driven operation
- Should move on continuous production service
- Need further manpower and/or automation


System: tolerable for more jobs

Toward full production system: 1

Management

- Bookkeeping using GUI
- Replica catalog

Fabrication

- Fine grain logging
  - e.g. Keep log files and timestamp of all failed jobs
- Log archiving
  - Keep production logs on SE
- Query input data using AMGA metadata
- Reprocessing of jobs requested by other subsystems (validation, distribution)
- Output merging
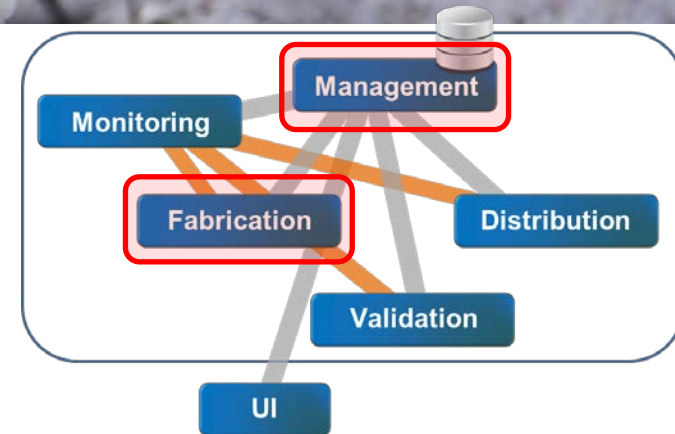
Toward full production system: 2

Distribution

- Smart data distribution
- Consider resource status and usage
  - SE capacity and free space
  - Network (PerfSONAR mesh)
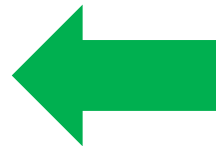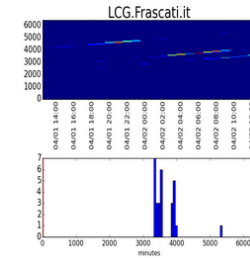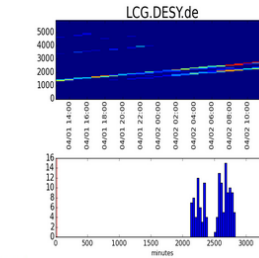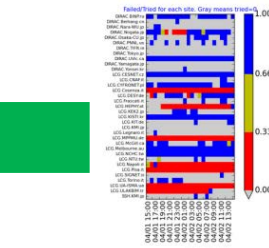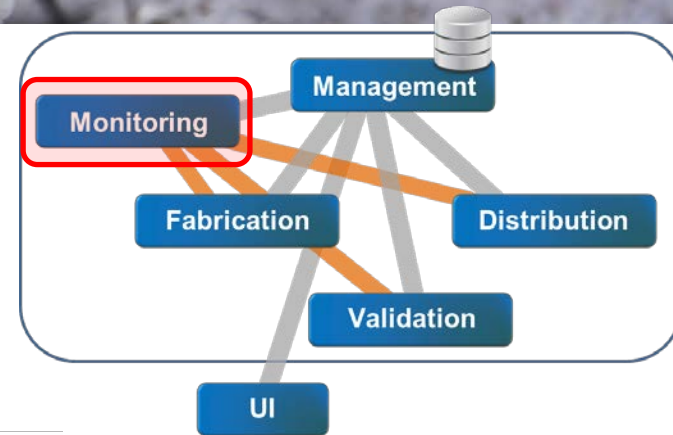 - Area based resource distribution plan
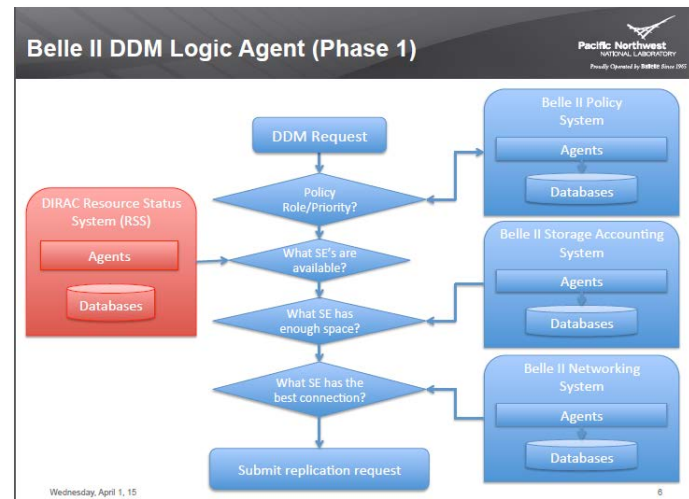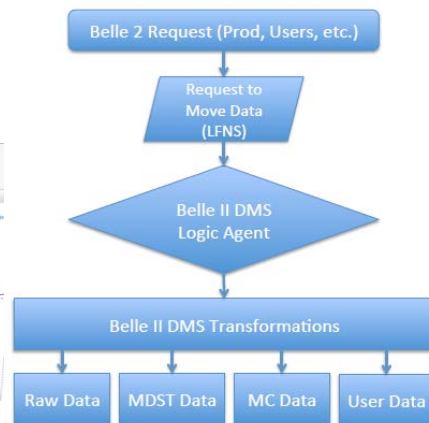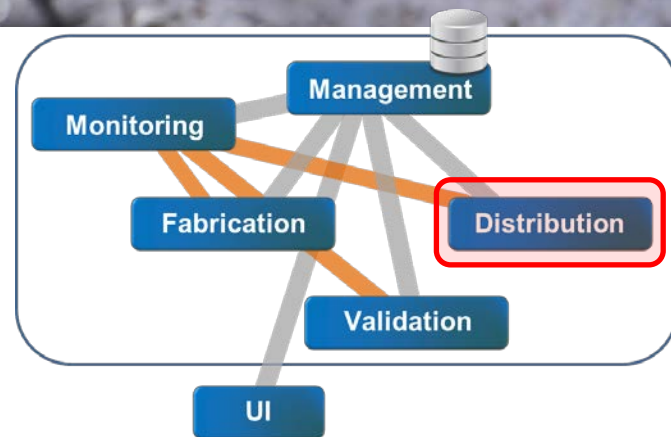
M. Schram@Belle II General Meeting

Toward full production system: 3

Monitoring

- System control by monitoring information
  - Currently all of managements (e.g. resource disabling) is performed by expert
  - Move on automatic resource control
- Submit GGUS ticket for trivial failure

DIRAC Resource Status System

Production Management System

→**Poster session A337 (Y.Kato)**

→**Poster session B314 (K.Hayasaka)**

Monitoring data
summarized in HappyFace

Hideki MIYAKE, Belle II production system, CHEP2015

Conclusions

- Belle II exploits distributed computing model

- Belle II analysis software framework runs on GRID using job wrapper based on DIRAC API

- The production system is aiming at autonomous operation in both the workload and the data management

- Prototype production system is working for Belle II MC mass production

- Full production system development is undergoing:
  - More realistic workload management
  - Smart data distribution
  - Automatic system controller

  based on monitoring information

backup

Belle II sites

- Resources spread around the world

EMI
OSG
Cloud
Local cluster
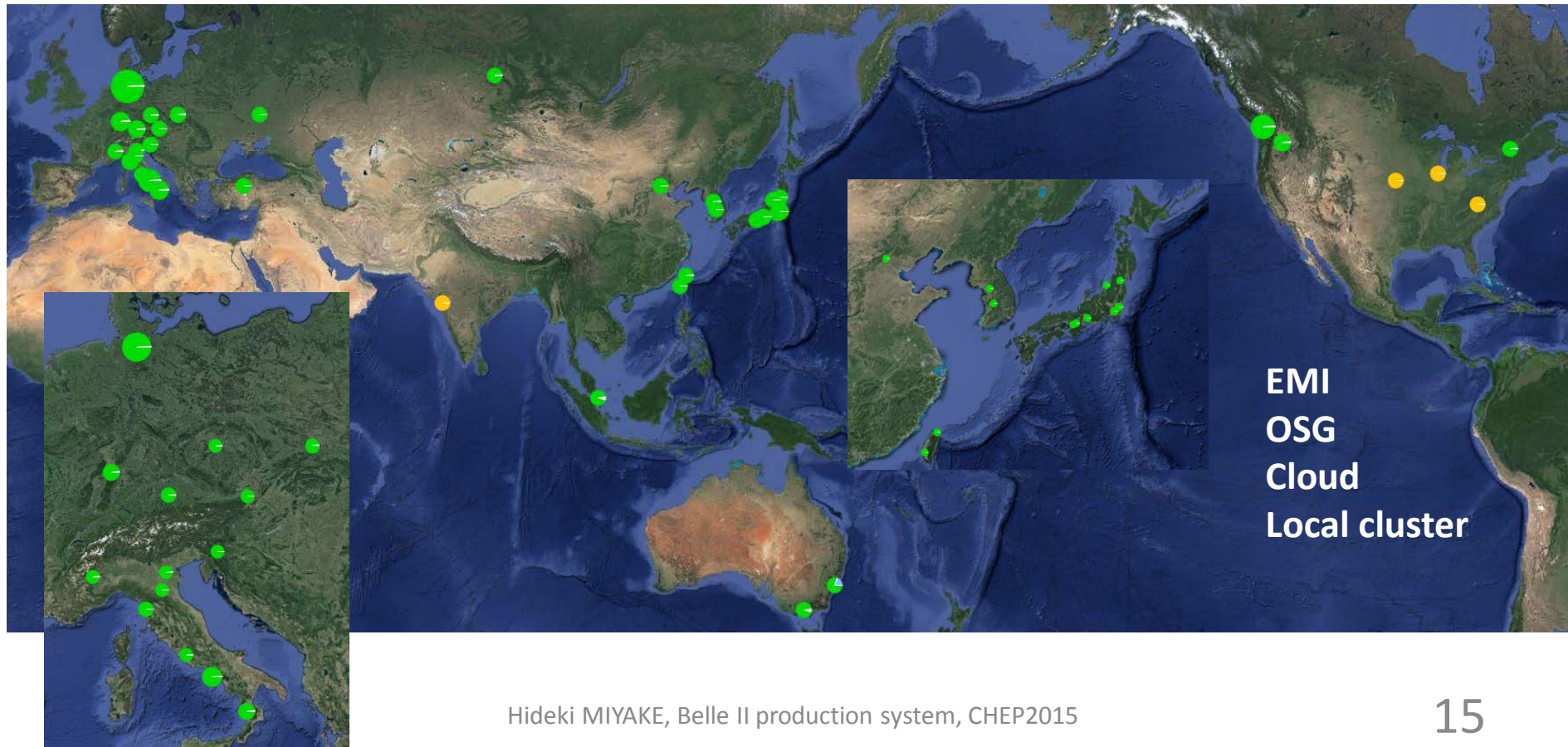
Belle II sites

- Resources spread around the world
- Contribution from small sites are NOT small!

CPU days used by Site

30 Days from 2015-03-02 to 2015-04-01

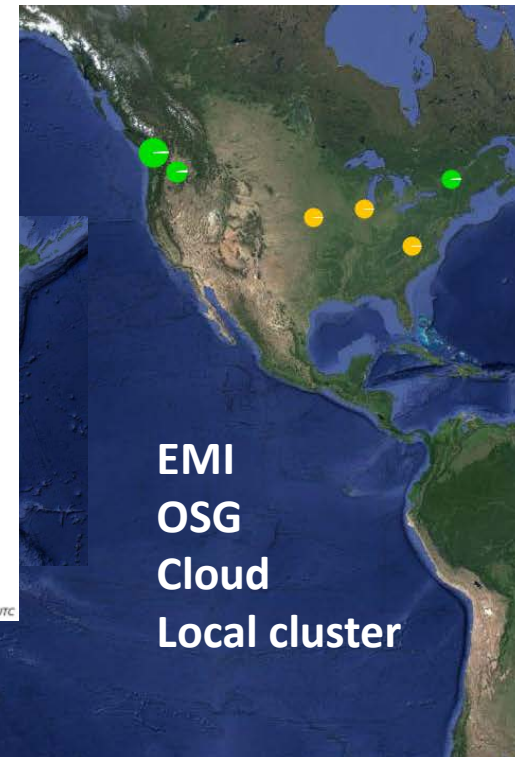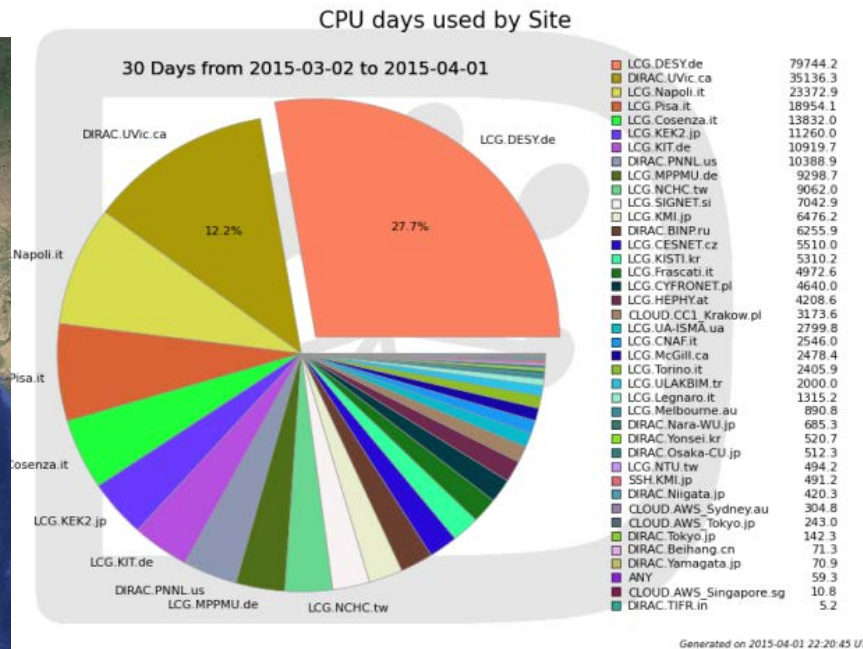| Site	CPU days
LCG.DESY.de	79744.2
DIRAC.UVic.ca	35136.3
LCG.Napoli.it	23372.9
LCG.Pisa.it	18954.1
LCG.Cosenza.it	13832.0
LCG.KEK2.jp	11260.0
LCG.KIT.de	10919.7
DIRAC.PNNL.us	10388.9
LCG.MPPMU.de	9298.7
LCG.NCHC.tw	9062.0
LCG.SIGNET.si	7042.9
LCG.KMI.jp	6476.2
DIRAC.BINP.ru	6255.9
LCG.CESNET.cz	5510.0
LCG.KISTI.kr	5310.2
LCG.Frascati.it	4972.6
LCG.CYFRONET.pl	4640.0
LCG.HEPHY.at	4208.6
CLOUD.CC1_Krakow.pl	3173.6
LCG.UA-ISMA.ua	2799.8
LCG.CNAF.it	2546.0
LCG.McGill.ca	2478.4
LCG.Torino.it	2405.9
LCG.ULAKBIM.tr	2000.0
LCG.Legnaro.it	1315.2
LCG.Melbourne.au	890.8
DIRAC.Nara-WU.jp	685.3
DIRAC.Yonsei.kr	520.7
DIRAC.Osaka-CU.jp	512.3
LCG.NTU.tw	494.2
SSH.KMI.jp	491.2
DIRAC.Niigata.jp	420.3
CLOUD.AWS_Sydney.au	304.8
CLOUD.AWS_Tokyo.jp	243.0
DIRAC.Tokyo.jp	142.3
DIRAC.Beihang.cn	71.3
DIRAC.Yamagata.jp	70.9
ANY	59.3
CLOUD.AWS_Singapore.sg	10.8
DIRAC.TIFR.in	5.2

Generated on 2015-04-01 22:20:45 UTC

**EMI
OSG
Cloud
Local cluster**

Central system
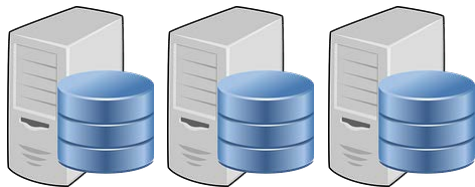
- ## KEK DIRAC cluster

  Workload, data management, web portal and other services

  AMGA   LFC

- ## Remote DIRAC slaves (controlled by master server)

  Plus some development serves
  (Krakow, PNNL, KMI)

  Some institutes manage own DIRAC slaves
  for job submission and monitoring
  - PNNL, KMI, UVic

  CVMFS
  Strutum0: CERN
  Strutum1: CERN, GridKa, ASGC