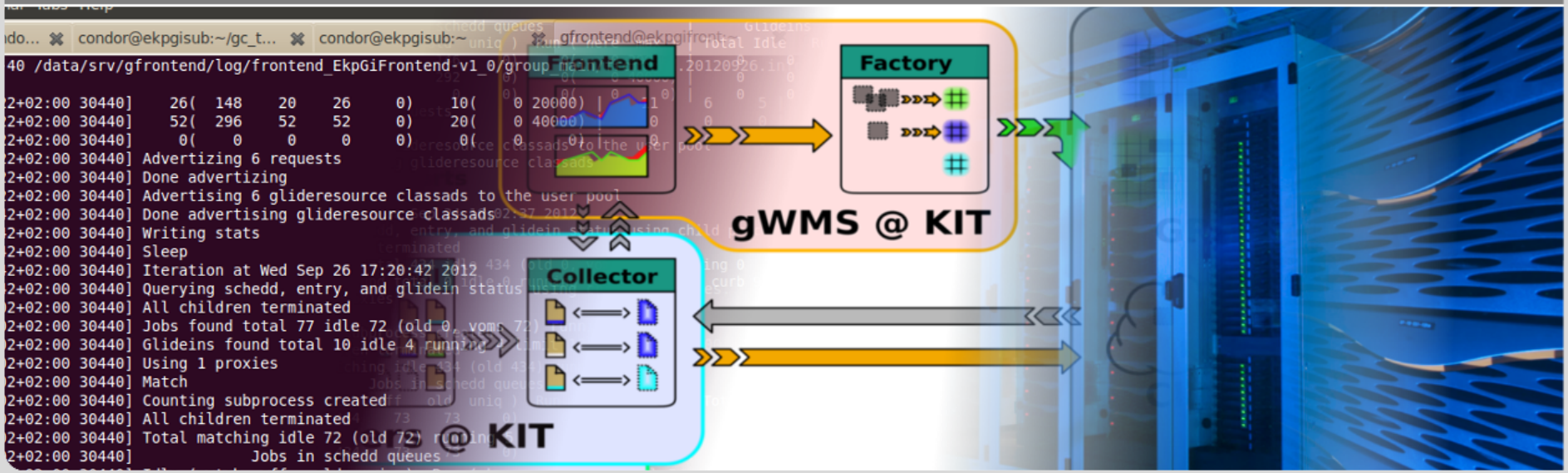
Karlsruhe Institute of Technology

# Realization of “Single Sign-In” User Centered Computing at KIT-EKP: GlideinWMS

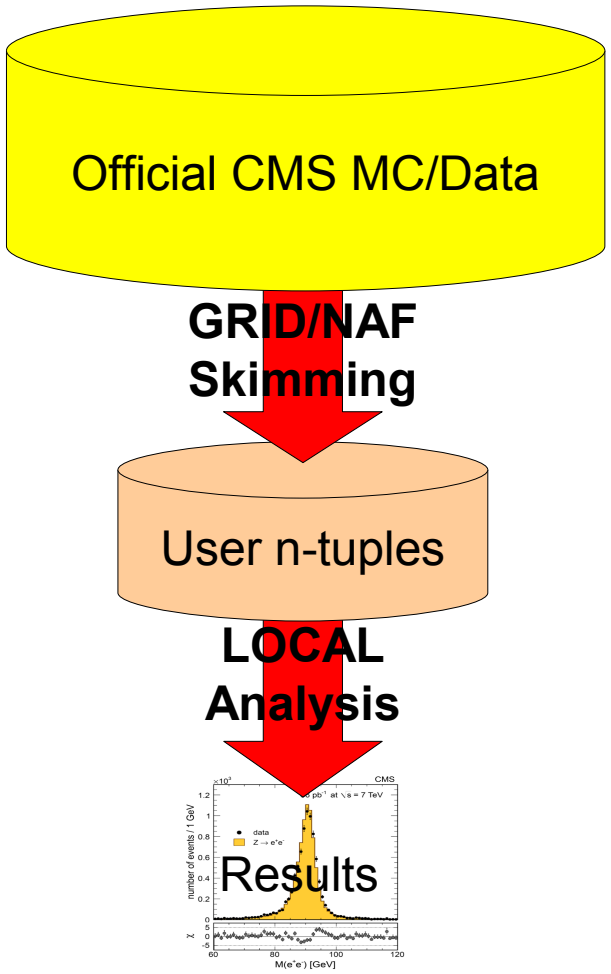
**Max Fischer**, Oliver Oberst, Günter Quast, Marian Zvada

INSTITUT FÜR EXPERIMENTELLE KERNPHYSIK (EKP) · FAKULTÄT FÜR PHYSIK



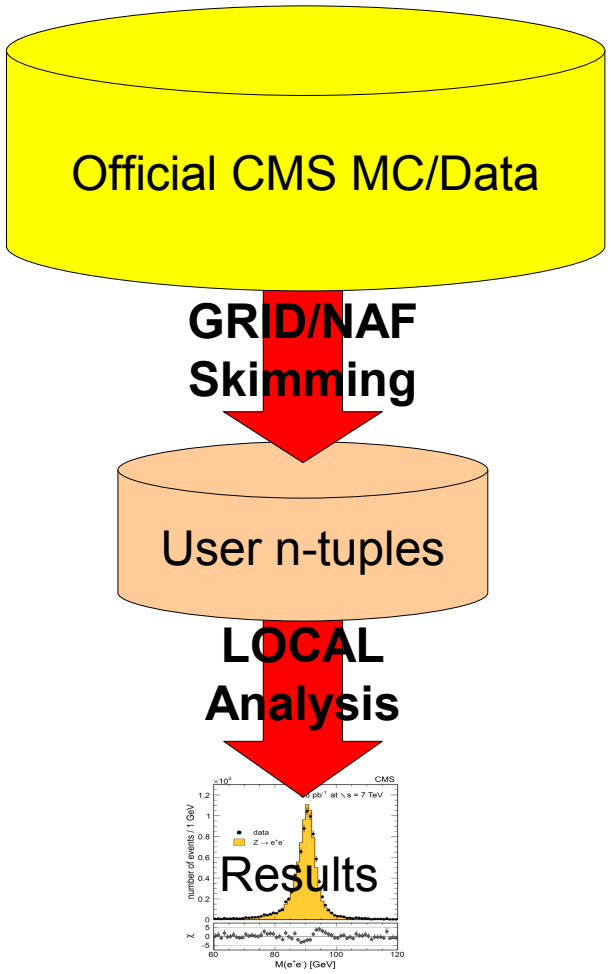
# Situation of DCMS Computing in 2011

- 2011 DCMS computing survey
- Prefered resource usage:
  - Skimming: Grid
    - automated processing
  - Analysis: Local
    - little overhead
    - simple debugging



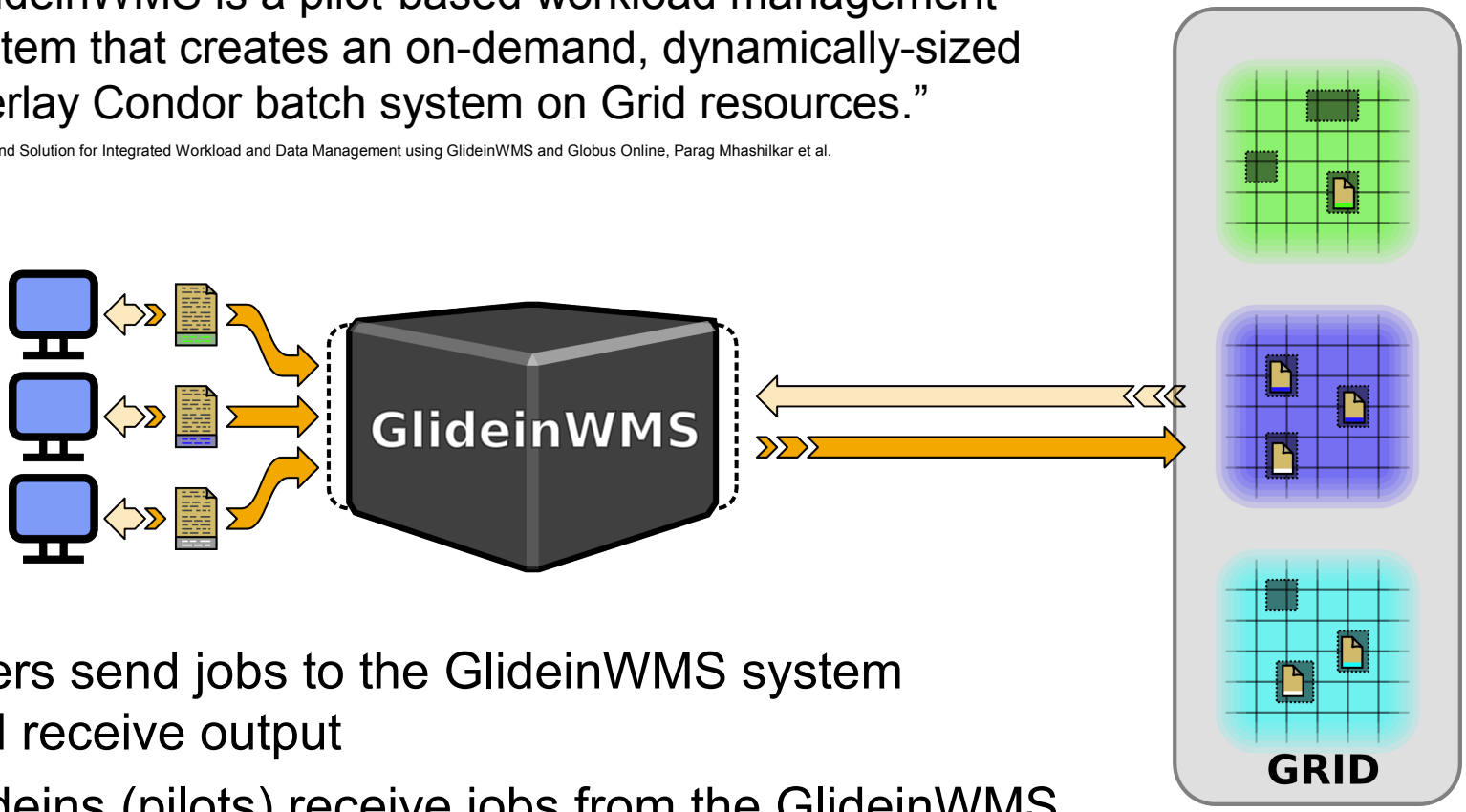
# Situation of DCMS Computing in 2011

- 2011 DCMS computing survey
- Prefered resource usage:
  - Skimming: Grid
    - automated processing
  - Analysis: Local
    - little overhead
    - simple debugging
- Key issues for computing:
  - Inadequate resource balancing
    - Bottlenecks at popular resources
    - Manual, individual workarounds
  - Unattractive high-throughput resources
    - Large overhead in administration
    - Site specific requirements



# Possible Solution: GlideinWMS

> “GlideinWMS is a pilot-based workload management system that creates an on-demand, dynamically-sized overlay Condor batch system on Grid resources.”
>
> End-To-End Solution for Integrated Workload and Data Management using GlideinWMS and Globus Online, Parag Mhashilkar et al.




- Users send jobs to the GlideinWMS system and receive output
- Glideins (pilots) receive jobs from the GlideinWMS system, execute them and return output
- User↔Glidein interaction is possible but not required

# Possible Solution: GlideinWMS II

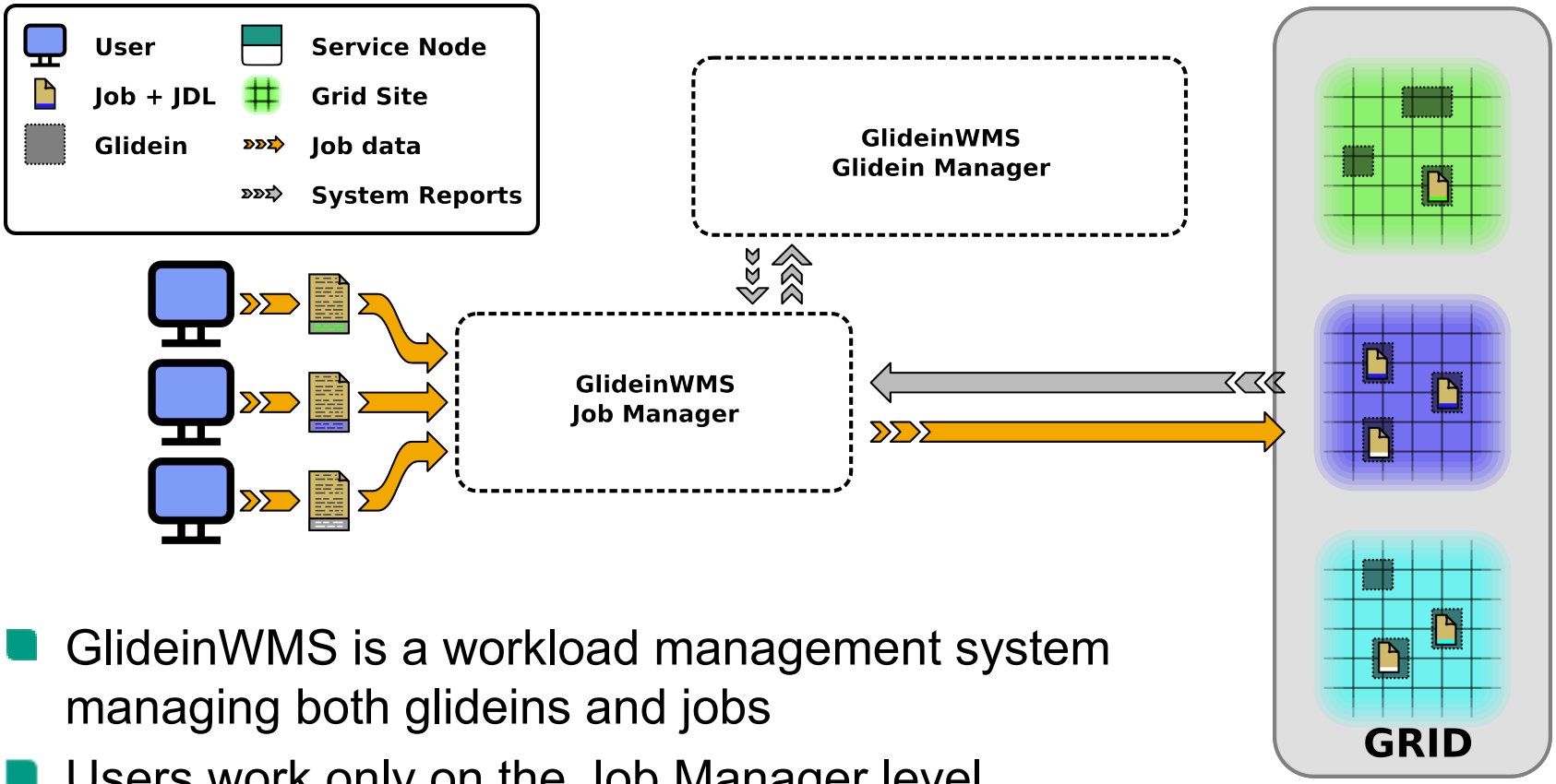
- Benefits for users
  - Easy usability for default use
  - Options for advanced use
  - Uniform access environment
  - Extensible via job management tools

# Possible Solution: GlideinWMS II

- Benefits for users
  - Easy usability for default use
  - Options for advanced use
  - Uniform access environment
  - Extensible via job management tools

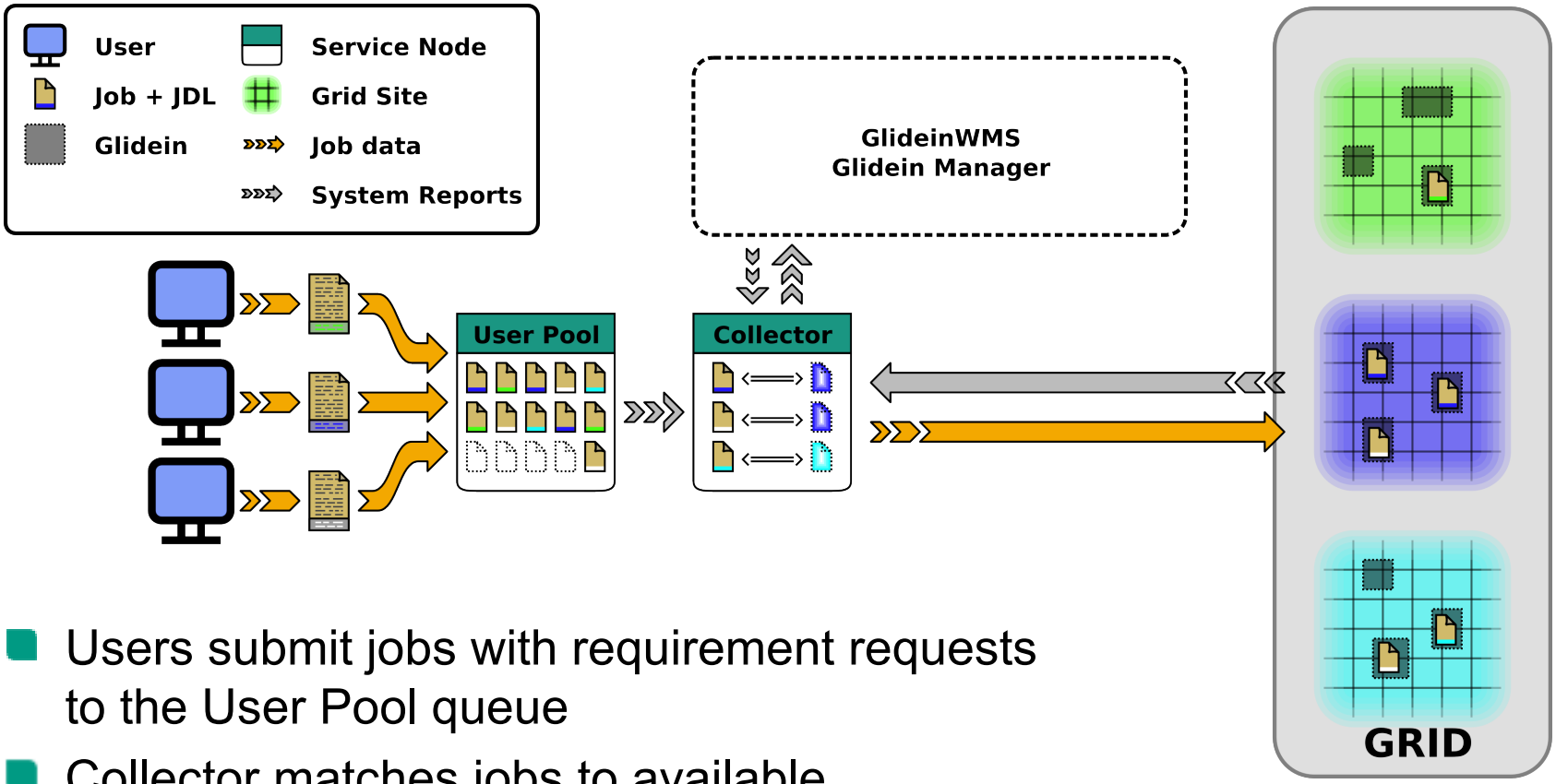
- Benefits for administrators
  - Resistant to temporary unavailability of resources
  - Centralized status monitoring
  - Uniform architecture for all users

# GlideinWMS – concept



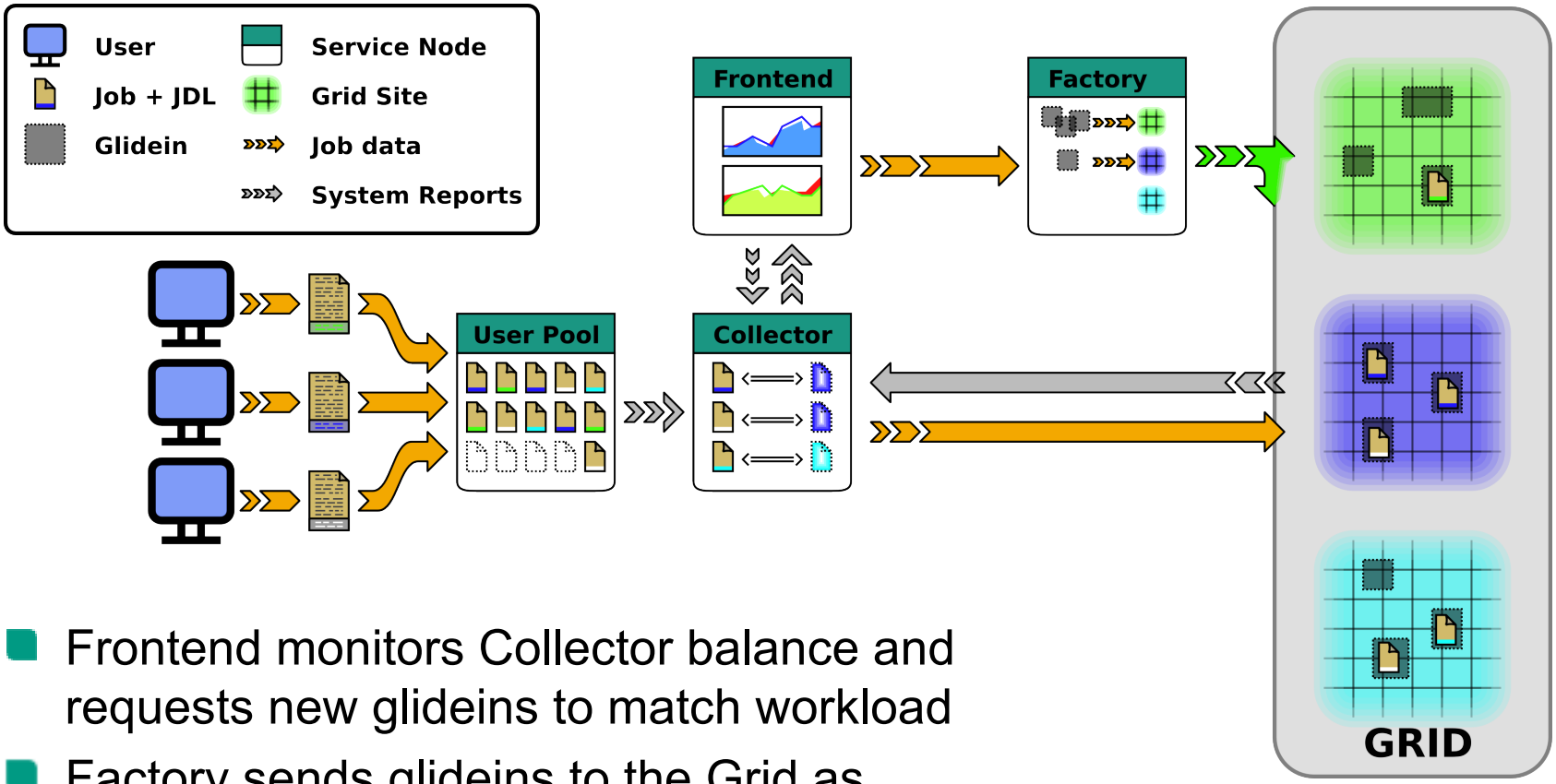

- GlideinWMS is a workload management system managing both glideins and jobs
- Users work only on the Job Manager level
- Glideins reserve resources on the Grid, validate sites and execute jobs

# GlideinWMS – concept II



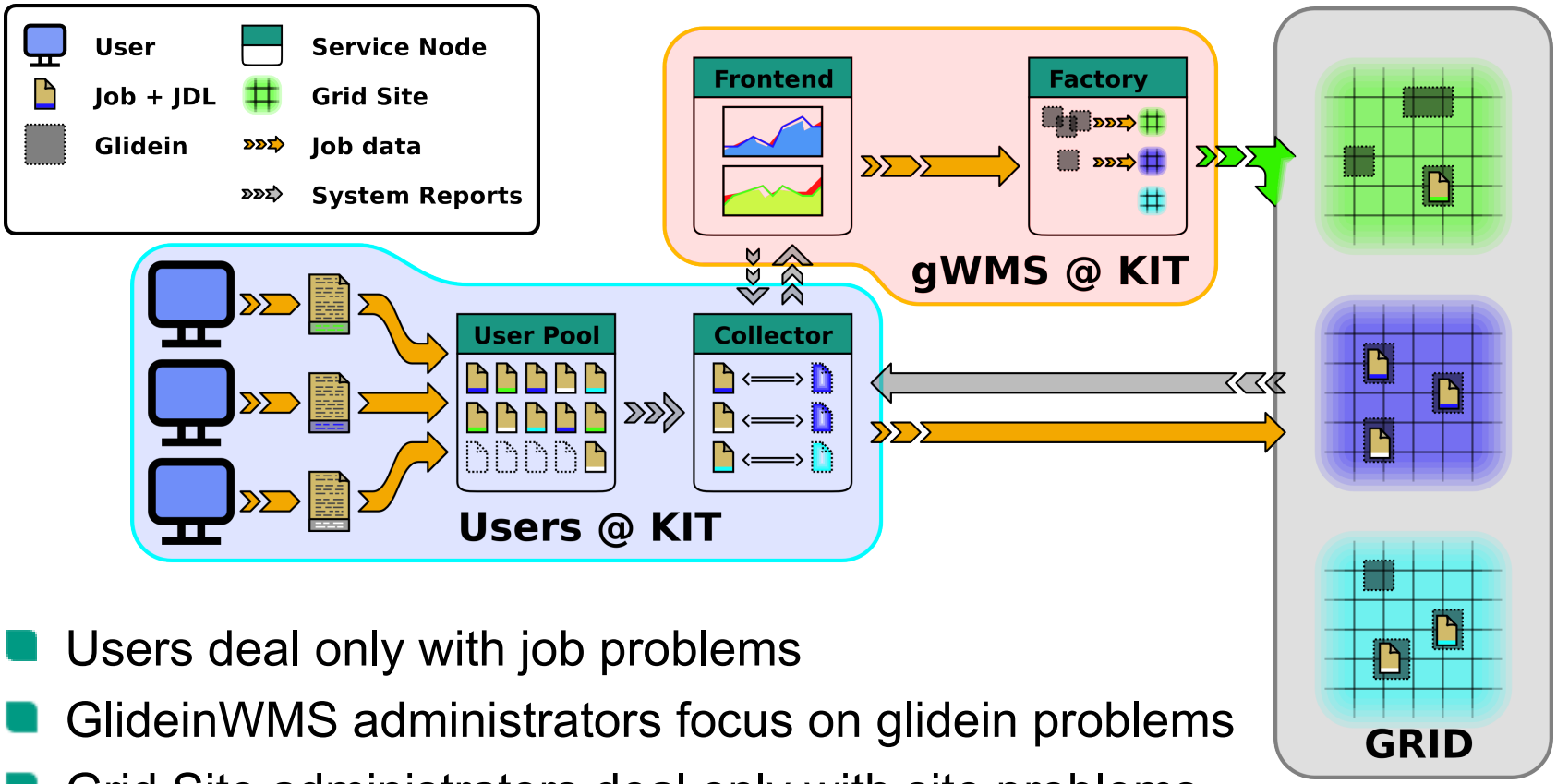

- Users submit jobs with requirement requests to the User Pool queue
- Collector matches jobs to available glidein computing resources
- User Pool shields User from complexity of outside resources

# GlideinWMS – concept III




- Frontend monitors Collector balance and requests new glideins to match workload
- Factory sends glideins to the Grid as requested by Frontend

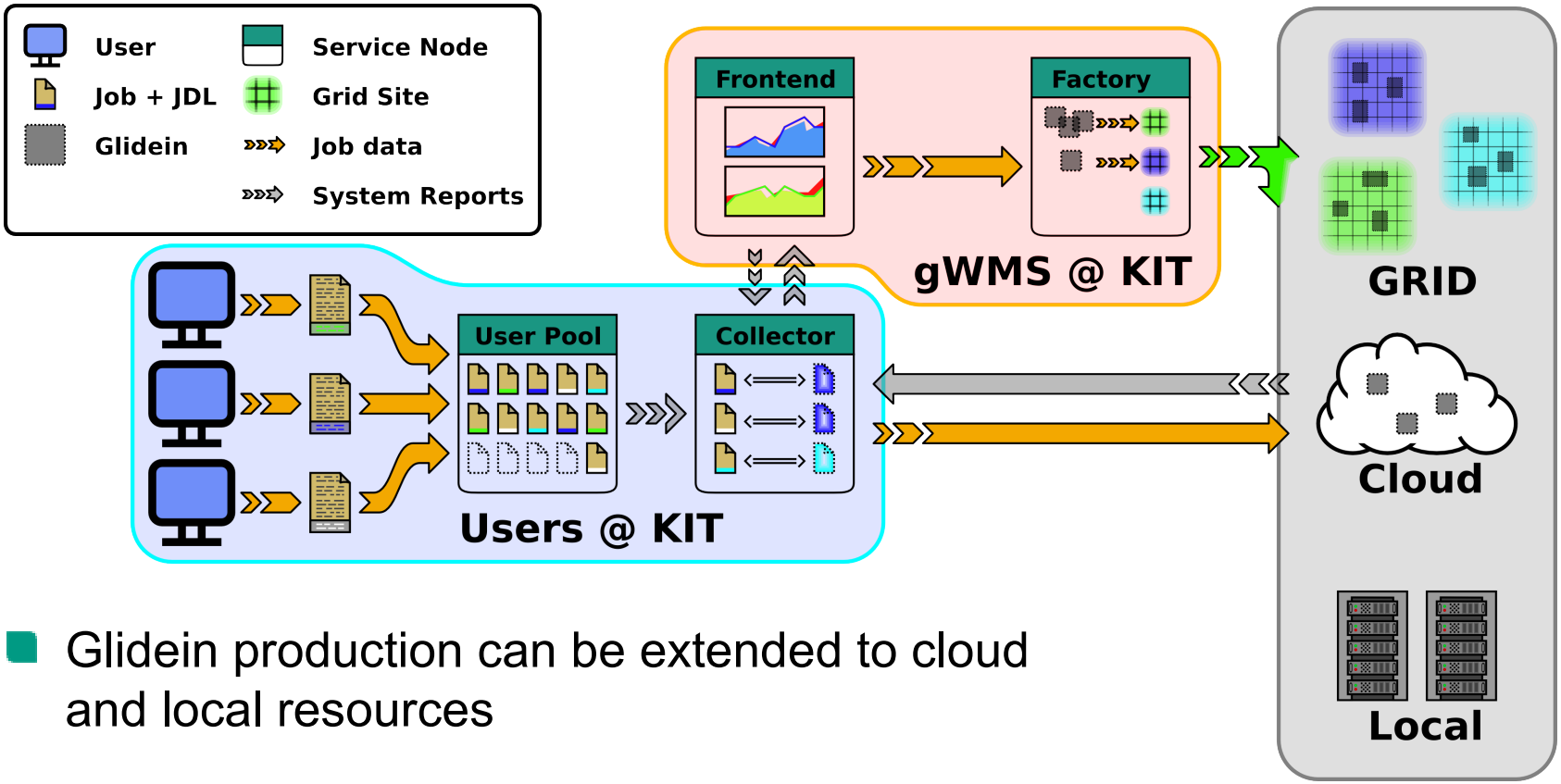
# GlideinWMS – concept IV




- Users deal only with job problems
- GlideinWMS administrators focus on glidein problems
- Grid Site administrators deal only with site problems

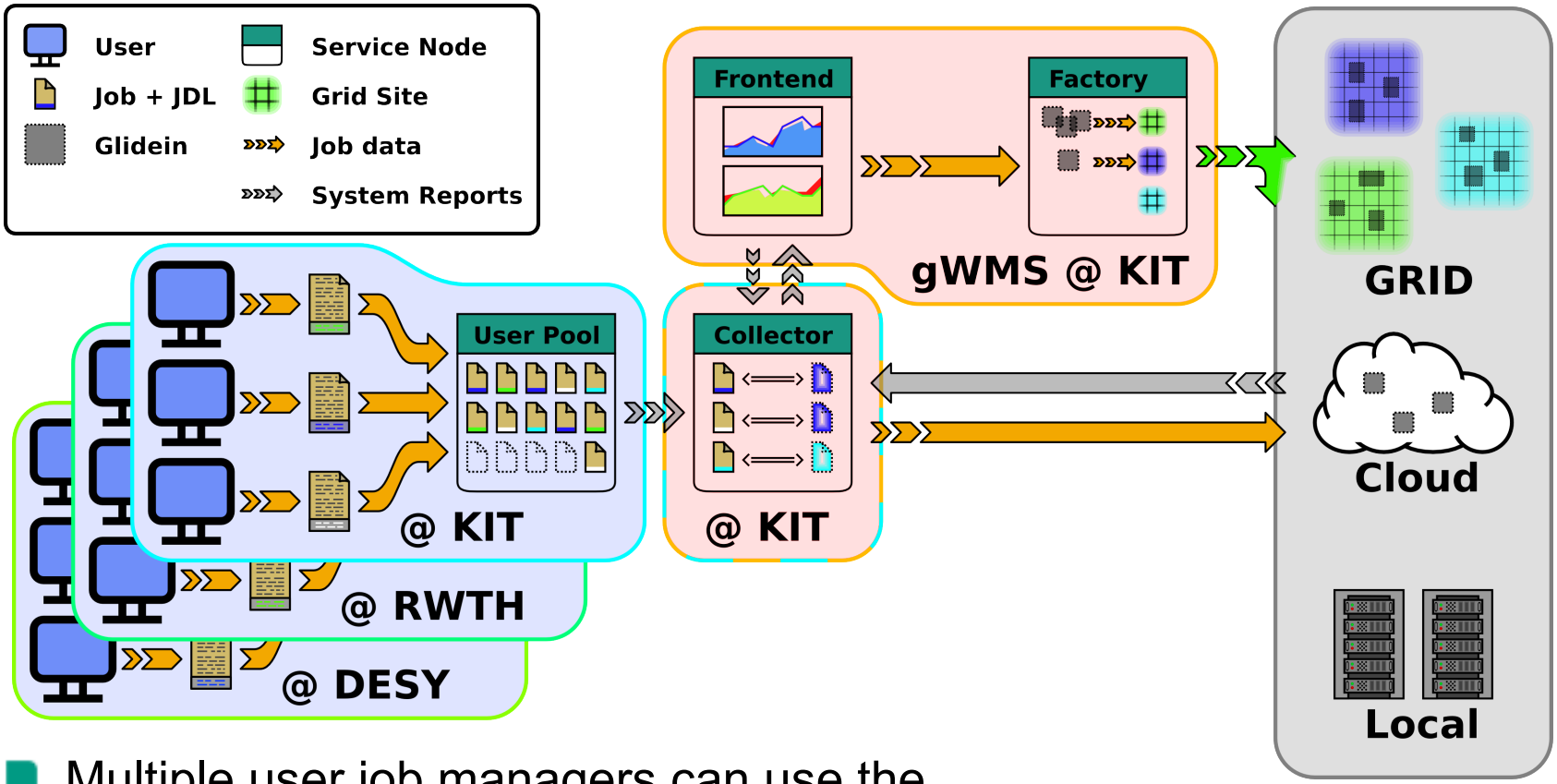
→ clear separation of responsibilities

# GlideinWMS – perspective




- Glidein production can be extended to cloud and local resources

# GlideinWMS – perspective II



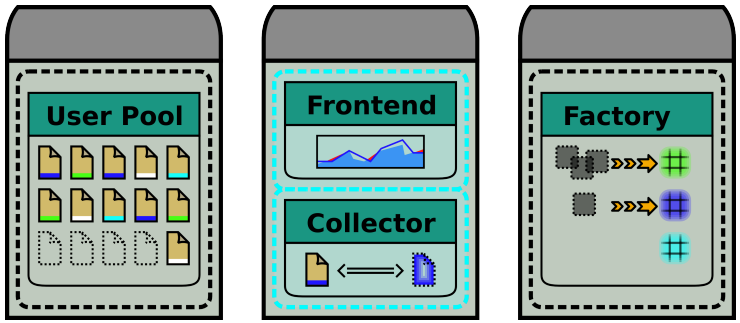

- Multiple user job managers can use the existing GlideinWMS infrastructure

# GlideinWMS: Status at EKP

- Supervision by CMS GlideinWMS expert Marian Zvada

# GlideinWMS: Status at EKP

- Supervision by CMS GlideinWMS expert Marian Zvada
- Operation of dedicated GlideinWMS infrastructure
  - 3 high performance blades
    - 24 cores @ 2.67 GHz
    - 48 GB RAM
    - 1.8 TB HDD
  - 2 virtual machines for 4 nodes total
    - One node per service




**Sufficient for 25000 simultaneously running jobs**

# GlideinWMS: Status at EKP

- Supervision by CMS GlideinWMS expert Marian Zvada
- Operation of dedicated GlideinWMS infrastructure
- Production of Glideins since Summer 2012
  - Grid site: GridKA
  - 1st run: 5 jobs `$ condor_submit submit-to-KIT.jdl`
  - 2nd run: 1000 jobs `$ condor_submit submit-to-KITmany.jdl`

```
Universe   = vanilla
Executable = toyMC.sh
Arguments  = 10000000 100
Log        = toyMC.log
Output     = toyMC.out.$(Cluster).$(Process)
Error      = toyMC.err.$(Cluster).$(Process)
+DESIRED_Sites = "T1_DE_KIT"
requirements = stringListMember(GLIDEIN_CMSSite,DESIRED_Sites)
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
Queue 1000
```

# GlideinWMS: Status at EKP

- Supervision by CMS GlideinWMS expert Marian Zvada
- Operation of dedicated GlideinWMS infrastructure
- Production of Glideins since Summer 2012
  - Grid site: GridKA
  - 1st run: 5 jobs
  - 2nd run: 1000 jobs

  - To date:
    - Over 50k test jobs
    - Over 10k CMSSW Jobs – skims and analysis

# GlideinWMS: Status at EKP

- Supervision by CMS GlideinWMS expert Marian Zvada
- Operation of dedicated GlideinWMS infrastructure
- Production of Glideins since Summer 2012
- Transition to analysis usage at EKP
  - Module for Grid-Control job manager ✓
    - Parallel development for CMS CRAB
  - Remote submission from all EKP login pools ✓
  - Integration of DCMS Grid Resources ✓
  - Grid user identification via proxy/gLExec

# GlideinWMS: Status at EKP

- Supervision by CMS GlideinWMS expert Marian Zvada
- Operation of dedicated GlideinWMS infrastructure
- Production of Glideins since Summer 2012
- Transition to analysis usage at EKP
- Plans for long term usage
  - Integration of local resources at EKP
  - Integration of cloud resources at CN of KIT (OpenNebula)
  - Integration of other DCMS resources
  - Access for all DCMS institutes/users

# Summary and Outlook

- Summary
  - GlideinWMS improves resource balancing
  - Users gain straightforward access to remote computing resources
  - Administrators do not have to micromanage users

  - Implementation of GlideinWMS at KIT
    - Expert experience from GlideinWMS at CERN/FNAL/UCSD
    - Dedicated hardware infrastructure exists
    - Successful operation and testing since August 2012
    - Ongoing service development

- Outlook
  - Launch as primary CMS computing service at EKP
  - Integration of additional computing resources
  - To be tested:

    Access to EKP Glidein infrastructure for interested DCMS institutes

# Summary and Outlook

- Summary
  - GlideinWMS improves resource balancing
  - Users gain straightforward access to remote computing resources
  - Administrators do not have to micromanage users

  - Implementation of GlideinWMS at KIT
    - Expert experience from GlideinWMS at CERN/FNAL/UCSD
    - Dedicated hardware infrastructure exists
    - Successful operation and testing since August 2012
    - Ongoing service development

- Outlook
  - Launch as primary CMS computing service at EKP
  - Integration of additional computing resources
  - To be tested:

    Access to EKP Glidein infrastructure for interested DCMS institutes