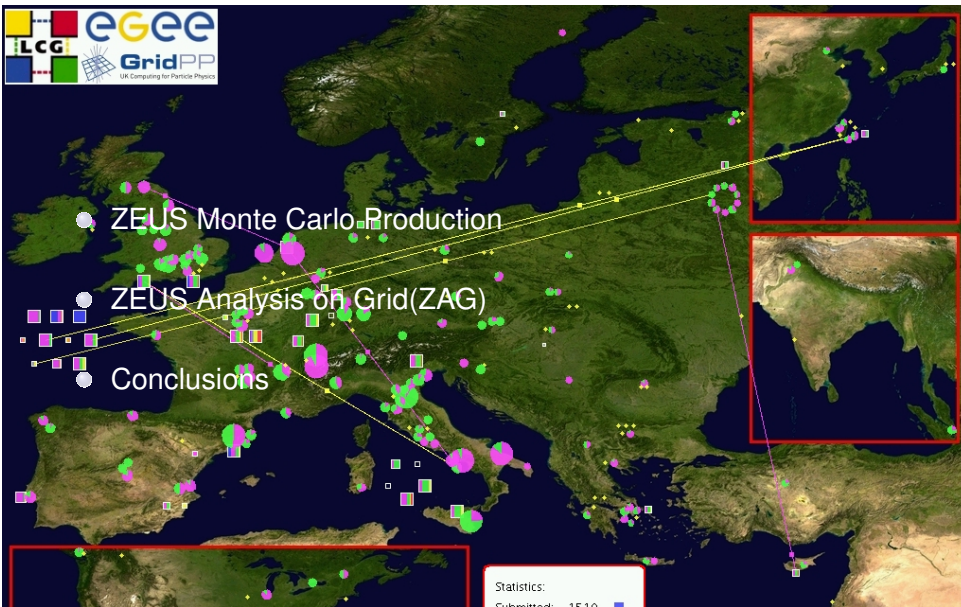


ZEUS Grid Usage: Monte Carlo Production and Data Analysis

Hartmut Stadie
Universität Hamburg



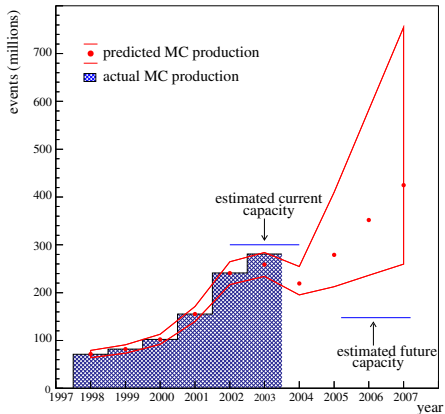
CHEP Conference
September, 4th 2007



Monte Carlo Demand

Projection in 2003:

Monte Carlo Production 1998 to 2007



Resources needed for Monte Carlo driven by

- integrated luminosity
- CPU time:

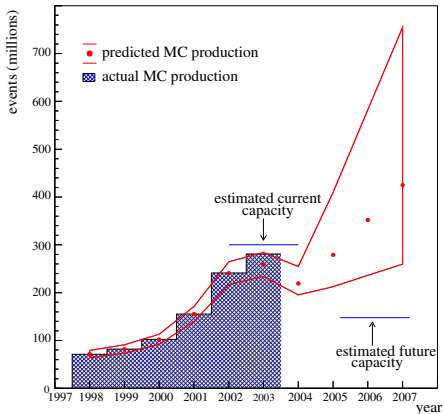
Executable	runtime increase
simulation	2.1 ×
trigger	1.7 ×
reconstruction	3.5 ×

- output files: 2.8 times larger per event

Monte Carlo Demand

Projection in 2003:

Monte Carlo Production 1998 to 2007



Resources needed for Monte Carlo driven by

- integrated luminosity
- CPU time:

Executable	runtime increase
simulation	2.1 ×
trigger	1.7 ×
reconstruction	3.5 ×

- output files: 2.8 times larger per event

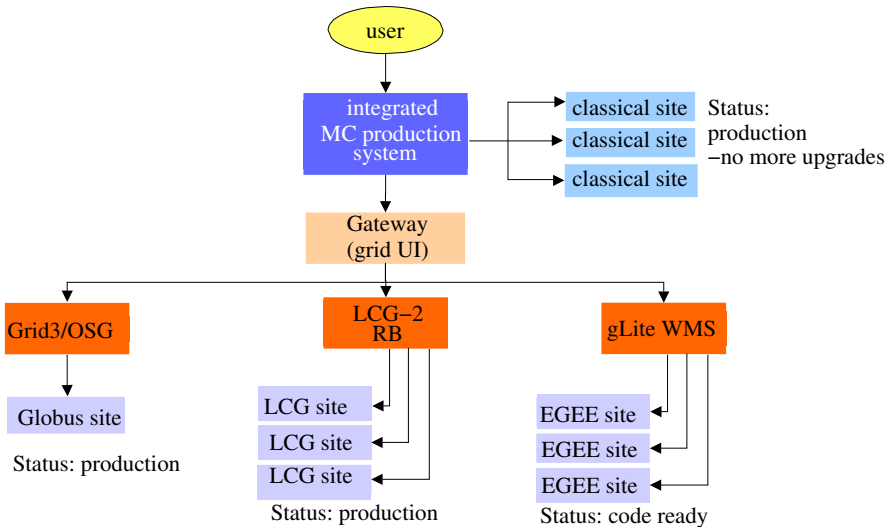
Use Grid resources at collaborating institutes!

Design Goals

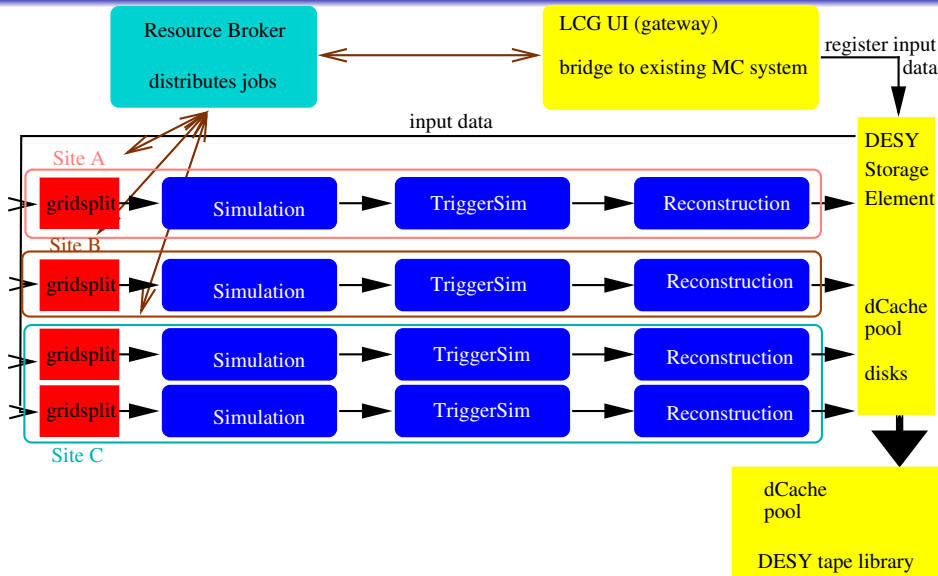
Goals

- use Grid resources opportunistically:
 - short jobs
 - few requirements
 - support different middlewares
- high efficiency (retries, time-outs)
- automatic submissions, checks, re-submission
- monitoring:
 - show job results
 - identify problematic sites
 - find bottlenecks

Bridge to the Grid



Work Flow of the MC Jobs

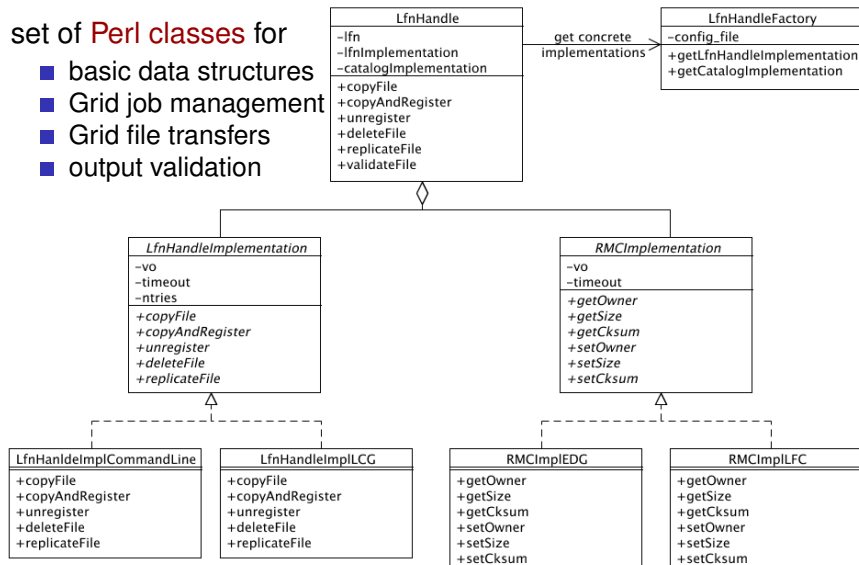


ZEUS Grid-Toolkit

- set of **Perl classes** for
 - basic data structures
 - Grid job management
 - Grid file transfers
 - output validation

ZEUS Grid-Toolkit

- set of **Perl classes** for
 - basic data structures
 - Grid job management
 - Grid file transfers
 - output validation



ZEUS Grid-Toolkit

- set of **Perl classes** for
 - basic data structures
 - Grid job management
 - Grid file transfers
 - output validation
- implementation using Grid tools encapsulated
- supports **different middleware**
- **fixes known deficiencies** of grid tools
- enforces **time-outs** on every command
- independent of the ZEUS software

LCG/EGEE Sites

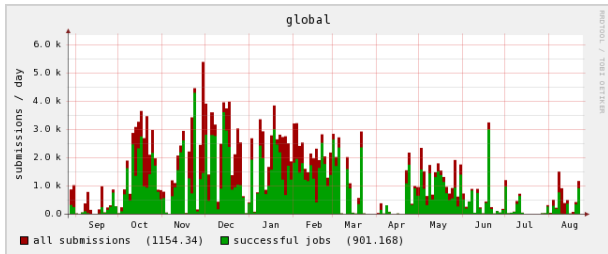
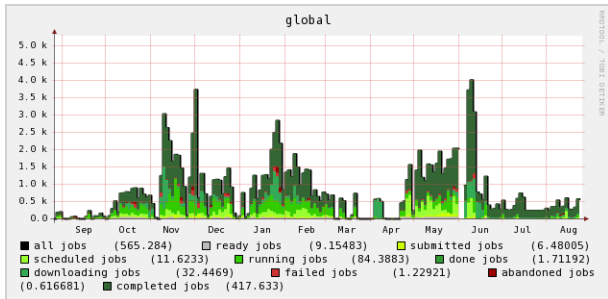
Status for Resource Broker CERN-CIC_lxn1188: Fri May 13 09:45:01 BST 2005



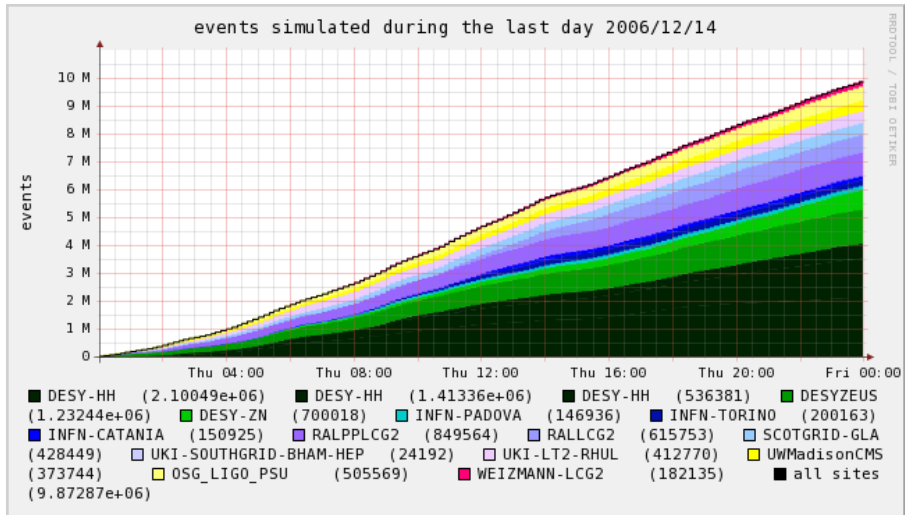
ZEUS Grid Job Monitoring

Monitoring

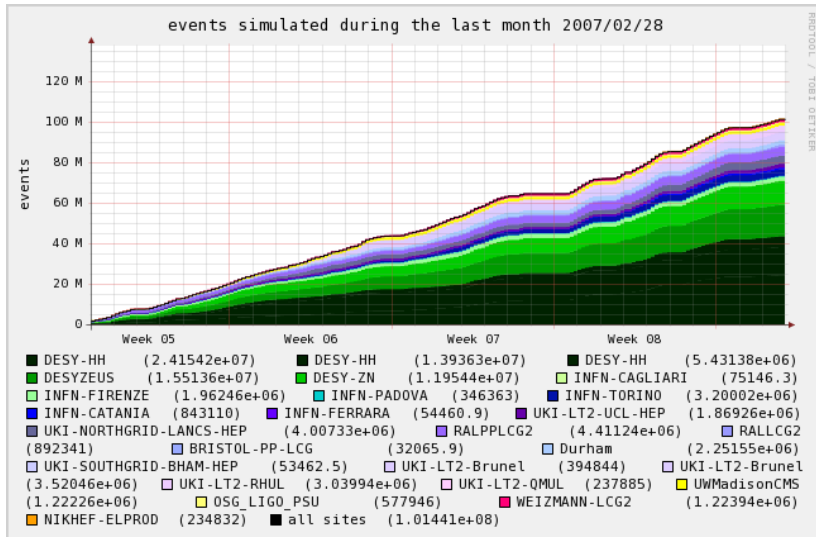
- overall and per site
- job states
- total number of (free) CPUs
- submissions
- efficiency
- error messages



December 14th 2006

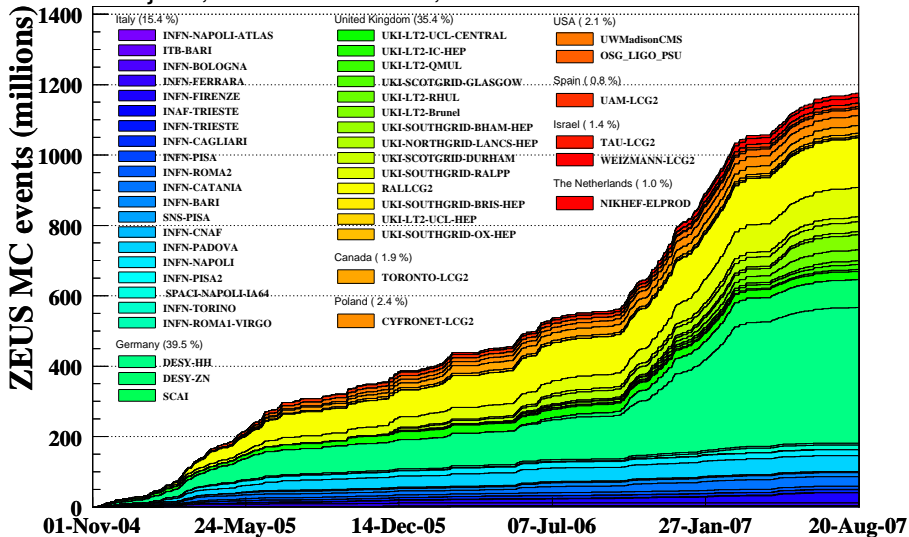


February 2007



Overall Production

1.15 M jobs, > 100 TB of data, **around 70 % of all ZEUS MC**



ZEUS Analysis on the Grid: Goals

Using the Grid should be similar to using central analysis farm(ZARAH)!

Requirements

- similar commands for submission, etc
- no Grid middleware knowledge required
- 99.999 % efficiency
- no changes needed for cards or executables
- same results!

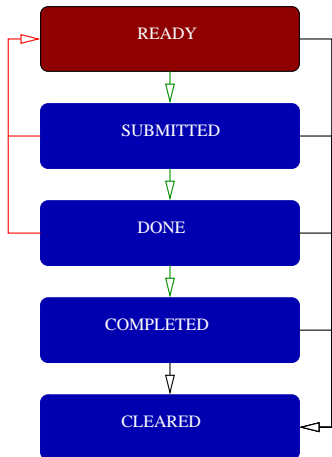
ZEUS Analysis on the Grid: Goals

Using the Grid should be similar to using central analysis farm(ZARAH)!

Requirements

- similar commands for submission, etc
submission framework
- no Grid middleware knowledge required
service for Grid submission, proxy storage
- 99.999 % efficiency
output validation, automatic re-submission
- no changes needed for cards or executables
data access using dcap protocol
- same results!
distribute constant databases, environment

Design: Finite State Machine



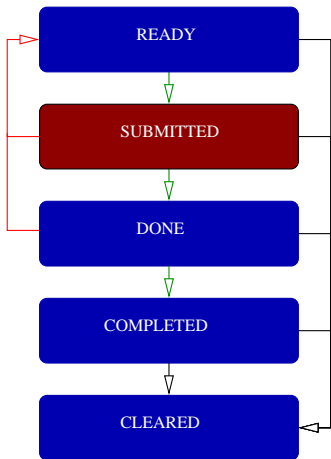
user commands **on any SL3 node**:

```
source ~zeusgrid/zeus/setup.csh
grid-proxy-init -valid 100:00
zag-upload-proxy
zag-submit-job eaze_run.sh \
  -f control.cards eaze_example.exe
```

READY State

- enter:
 - create archive with input files
 - upload archive to the Grid
- update:
 - OK
- leave:

Design: Finite State Machine

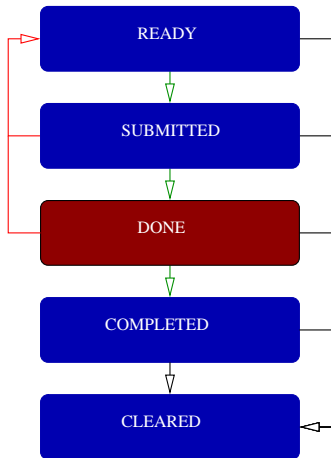


ZAG system automatically updates jobs in READY or SUBMITTED!

SUBMITTED State

- enter:
 - submit job to Grid
- update:
 - query Grid job status
 - **FAILED** if Grid job failed
 - **OK** if Grid job is done
- leave:
 - cancel running Grid job
 - get Grid job output

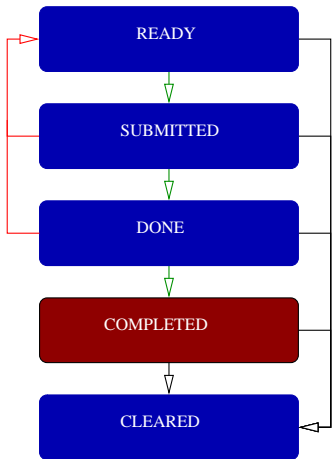
Design: Finite State Machine



DONE State

- enter:
 - check Grid job log files
 - check existence of output archive
 - **FAILED** if any test fails
- update:
 - copy archive from Grid to *JOBDIR*
 - uncompress the archive containing the output files
 - **OK** if this is done
- leave:

Design: Finite State Machine



user commands:

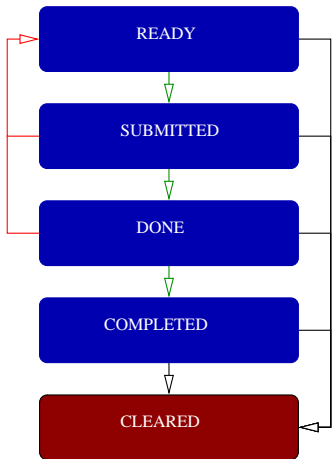
```

zag-job-ls 12345
zag-job-get 12345 -f orange.root
  
```

COMPLETED State

- enter:
- update:
- leave:

Design: Finite State Machine



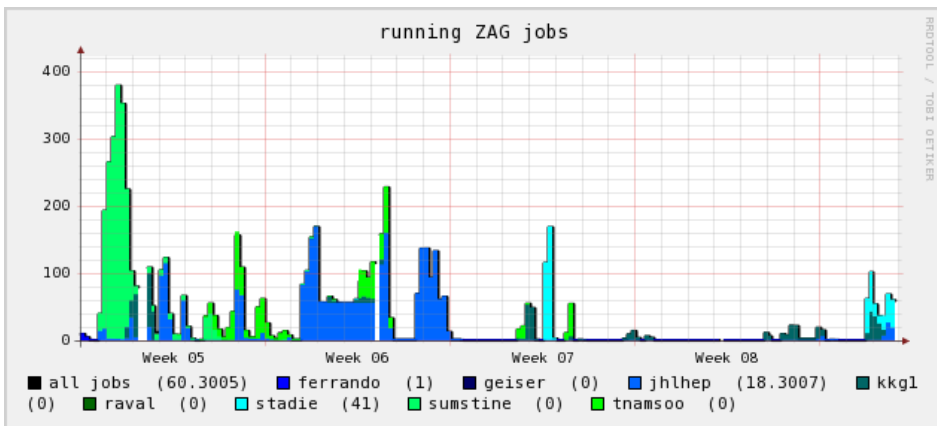
user command:

```
zag-purge-job 12345
```

CLEARED State

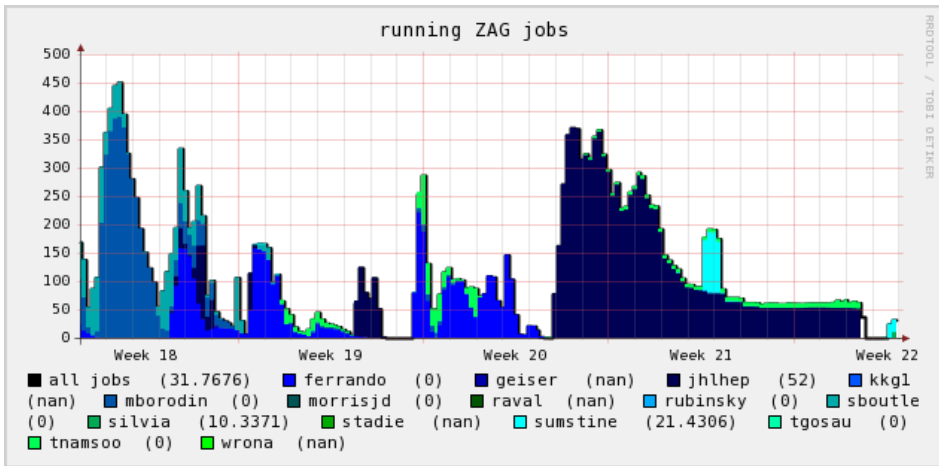
- enter:
 - remove archives from Grid
 - remove output directory
- update:
- leave:

Usage in February (DESY Tier2)



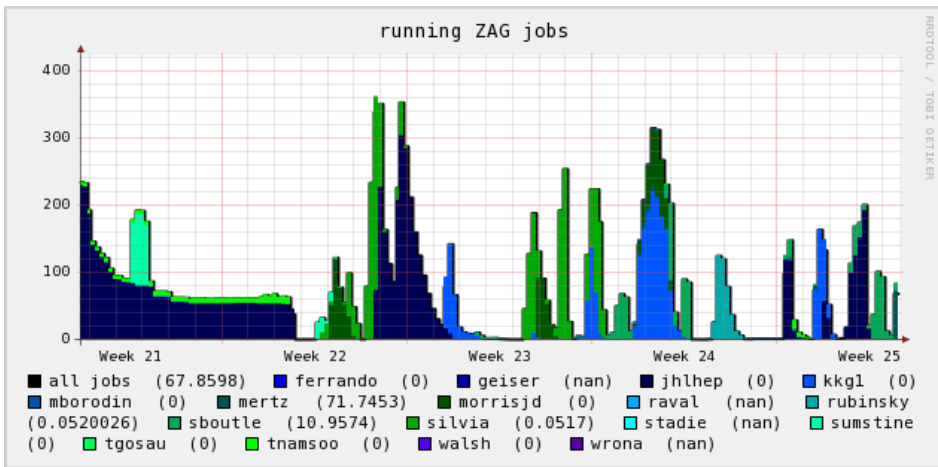
Eight different users
Up to 400 jobs running **in parallel**

Usage in May (DESY Tier2)



15 different users
<http://www-zeus.desy.de/grid/gridmon.php>

Usage in June (DESY Tier2)



17 different users

<http://www-zeus.desy.de/grid/gridmon.php>

Overall Usage in 2007

Stats of the **ZEUS** Grid Usestart date end date

Users	Completed Jobs	Grid Jobs Success Rate (%)	Runtime (Days)	Outputsize (MB)
alexch	6	31.57	0.04	16.60
ferrando	3276	63.89	1878.55	17073.87
geiser	1	100	0.00	0.00
jhlhep	18908	90.56	3299.85	1737629.59
kkgl	13677	97.06	927.73	46896.46
mborodin	2408	78.89	1008.19	283470.95
mertz	10088	85.78	928.60	775175.75
morrisjd	2135	99.76	276.52	127707.51
raval	56	98.24	0.61	85.63
roloff	0	0	0	No Return
rubinsky	2453	91.29	836.85	607420.43
sboutle	9091	97.38	1337.94	312076.18
silvia	7052	99.98	410.91	48038.30
stadie	833	50.06	216.94	40449.30
sumstine	1839	88.28	1385.86	379174.74
tgosau	1	100	0.12	0.00
tnamsoo	14163	91.39	2304.15	530778.87
walsh	1	100	0.81	0.01
wrona	224	56.28	171.51	37555.22
Total	86212	89.84	14985.18	4943549.41

Conclusions and Outlook

- Grid MC production is indispensable for ZEUS analyses
- 70 % of MC production done on Grid
- **normal ZEUS analysis jobs** can be run on Grid
- similar CLI as for ZEUS central analysis farm
- ZAG used by a growing number of ZEUS members
- heavily used by Grid **non-experts**
- peak usage exceeds analysis farm capacity by a factor of two

Thanks!

Thanks to DESY IT for their support and hosting of our central services, and to all site admins for their fast and competent replies!