



# Experience of running ARC on big sites as part of Atlas production



---

Gianfranco Sciacca

AEC - Laboratory for High Energy Physics, University of Bern, Switzerland

NorduGrid Conference 2015  
Tuesday 2 June 2015, University of Bern

*u<sup>b</sup>*



# ARC in ATLAS



## ATLAS Grid Monitor

2015-06-01 CEST 13:38:08



Processes: ■ Grid ■ Local

Country	Site	CPU	Load (processes: Grid+local)	Queueing
Denmark	Steno Tier 1 (DCSC/KU)	16232	344+0	589+0
Germany	LRZ-C2PAP	4072	2822+1263	539+6
	LRZ-LMU lcg-irz-ce0	1824	1454+0	486+13
	LRZ-LMU lcg-irz-ce3	1824	0+1454	13+485
	LRZ-LMU_MUC	9489	0+83	0+0
	RZG ATLAS HYDRA	167848	0+145342	151+0
	wuppertalprod	3684	2160+1141	84+160
Norway	Abel C1(UIO/USIT)	11600	1190+8683	621+541
	Abel C3(UIO/USIT)	11600	360+9569	481+670
Slovenia	Arnes	2164	1975+0	540+0
	SIGNET	4224	2975+209	642+77
Sweden	Abisko (HPC2N)	15936	0+14913	777+0
	Alarik (SweGrid, Luna>	3776	192+2194	411+20
	Triolith - Atlas (NSC)	25472	355+24579	415+3975
Switzerland	ATLAS BOINC	33929	4064+1526	1315+0
	Bern ce01 (UNIBE-LHEP)	1368	336+0	423+0
	Bern ce02 (UNIBE-LHEP)	776	303+0	431+1
	Bern LHEP HPC TEST	4208	0+3488 (queue inactive)	0+0
	Bern UBELIX T3	2472	528+980	433+465
	Geneva (UNIGE-DPNC)	604	105+6	5+0
	Lugano PHOENIX T2	4058	0+3936	0+17
UK	arc-ce01 (RAL-LCG2)	13694	2425+10935	456+0
	arc-ce02 (RAL-LCG2)	13694	2304+11048	572+0
	arc-ce03 (RAL-LCG2)	13694	2354+11048	573+0
	cetest01 (UKI-LT2-IC->	4	150+3790	120+193
	t2arc01 UKI-SOUTHGRID>	1200	231+735	181+0
<b>TOTAL</b>	<b>26 sites</b>	<b>369446</b>	<b>25835 + 256922</b>	<b>10258 + 6623</b>

da de en fi fr hu no ru sk sv uk

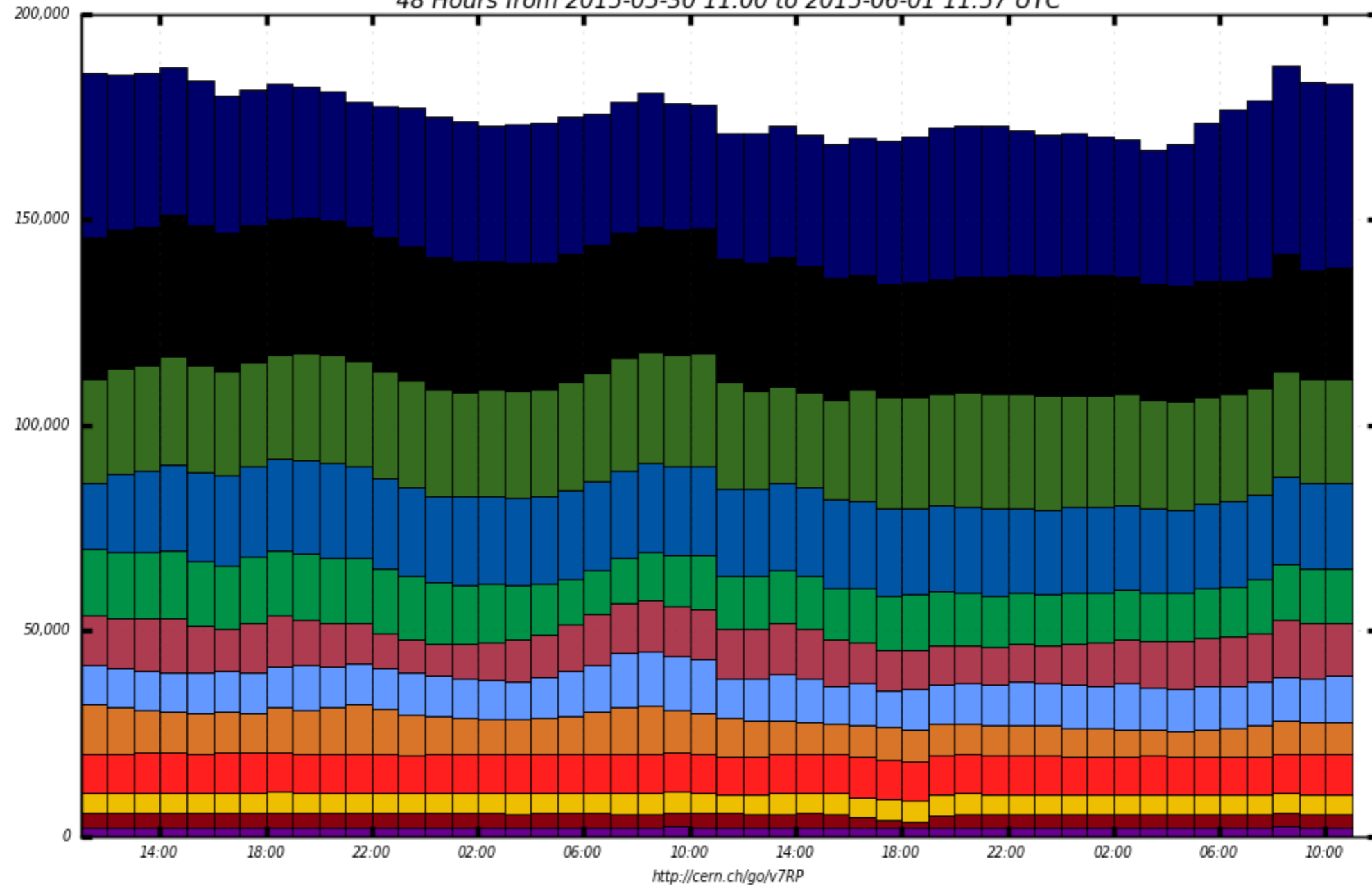


# ARC in ATLAS



## Slots of Running Jobs

48 Hours from 2015-05-30 11:00 to 2015-06-01 11:57 UTC



■ US  
■ CERN  
■ TW

■ DE  
■ ND  
■ RU

■ UK  
■ NL  
■ other

■ FR  
■ CA

■ IT  
■ ES

Maximum: 187,564 , Minimum: 166,875 , Average: 176,260 , Current: 183,134



# ARC in ATLAS: the ++



## Ease of deployment

modest hardware requirements

| metapackage+dependencies (all in e.g. `epe1`)

## (relative) Ease of configuration

| config file `arc.conf`

finely documented `arc.conf.reference`

no “configuration tool” required

## Ease of maintenance

Can upgrade on a live system (since `ARC 4.1.0-1`, at least for me)

| config file `arc.conf`

## Versatile

Not RH/SL only

Plenty of batch system support

Flexible configuration setup for cache and session dirs

## Non “invasive” setup

No middleware/services required on WNs

No outbound connectivity strictly required on WNs (with aCT)



# ARC in ATLAS: the ++



## Ease of deployment

modest hardware requirements

| metapackage+dependencies (all in e.g. `epe1`)

## (relative) Ease of configuration

| config file `arc.conf`

finely documented `arc.conf.reference`

no “configuration tool” required

**IT JUST WORKS!!**

## Versatile

Not RH/SL only

Plenty of batch system support

Flexible configuration setup for cache and session dirs

## Non “invasive” setup

No middleware/services required on WNs

No outbound connectivity strictly required on WNs (with aCT)



## a-rex crashes/stalls - plenty of scenarios

- `controldir` corrupted (`joblinks ...`)
- `rm -rf <controldir>/gm.fifo`
- `cache/session` under heavy load
- `gm-hearbeat` timeouts
- failed `DTRs`
- `logrotate`
- `glibmm-ERROR`
- "Maybe another instance of A-REX is already running"
- `<defunct>` processes (also `gridftpd`) (twice with ARC 4.2)
- `cache` full (related to the `joblinks` issue?) (`cachesize="40 25"`)

## UNIBE-LHEP: 17 recorded crashes since mid February - Same trend since ARC 4

### Infoproviders more stable since ARC 4 *(but see next slide)*

- before ARC 4, `CEinfo.p1` often slow/crashing
- cleanup `sessiondir` (cron)

### Jura

- Transition to Jura not trivial
- with `urbatch:1000`, many job summaries with >3k jobs - changed to 500



## Latest (since ARC 5)

- Job priority out of range for `gridengine` in `submit-sge-jobs`
- CE drops out of `infosys` regularly (all services running): add cron every 4h
- `grid-manager.log` exploding: full partitions, recovery is fiddly/time-consuming
- `cache-clean <defunct>` (also `arched` and `gridftpd`): all night to recover

```
[root@ce01 joblinks]# for i in `find /grid/lustre/cache/joblinks/ -maxdepth 1 -type d -mtime +14 -print`; do echo -e "Deleting directory $i";rm -rf $i; done
```



## Stability required - the pragmatic approach

ATLAS demands are quite challenging: new features, bugfixes, rushed upgrades

Bugs creep in regularly

Strong dependancy on the shared file-system (`cache/session`) performance

Service downtime/recovery is expensive (partitions full, systems jammed, re-boots, etc)

Dealing with system instabilities/failures requires a certain amount of “insider” help

`arc.conf` - some config complexities daunting for the entry-level admin

Fear of upgrades introducing new instabilities/fail modes

crons crons crons, hacks hacks hacks, is this the way?

`controldir cleanup (1/week)`

`sessiondir cleanup (1/week)`

`a-rex, gridftpd, nordugrid-arc-ldap-infosys clean/restart (6/day)`

`hacks to submit-<lrms>-job`

`hacks to SGEmod.pm, CEinfo.pl`





## Stability required - how to improve ?

ATLAS computing is a demanding and continuously evolving environment  
System/service stability is crucial for an effective management of the resources  
Use base is expanding and ARC faces new challenges in the face of it

## Ideas ?

Most cron workarounds ported to the services?

Script hacks as `arc.conf` parameters?

Resilience against file-system degradation?

Quality control/staged rollout? How expensive is an upgrade ...

Some/many bugs seem obscure or hardly reproducible on different systems: is the code base fully under control?

...