

PPS All sites Meeting: *Why it's so difficult to use PPS?*

A. Retico, N. Thackray (CERN)

SA1

EGEE 2007

PPS All sites Meeting: "PPS Users" session

Budapest, Hungary, 3rd October, 2007

- You are on your favourite UI
- Just upgraded to last version
- Your favourite command does not work
- You wonder ... ?



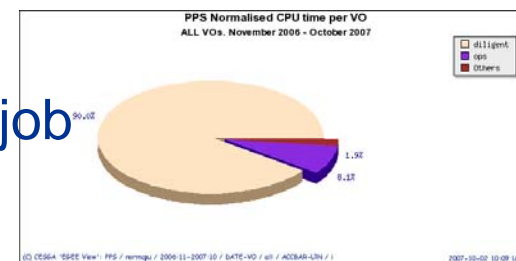
... “Why it was not found in PPS !?!!”

- **The answer proves that Users are human beings ...**
- **... but it is not always fair.**
- **Let's see why**

- **No regular nor continuous activity ever accounted in PPS to LCG VOs**
- **Three peaks of usage seen in 2007**
 - March: Test of SLC4 WNs
 - May: Deployment of SRMv2
 - May: Test of VOViews tag for Job Priority



- Diligent VO is using PPS as production infrastructure since Sep 2005
- E.g. Data challenge (July, August, September)
 - Extraction of features from pictures downloaded from flickr
 - > 38M images processed and received
 - ~500 jobs per day (through 2 WMSs)
 - 50 Mb of disk space and > 512 of RAM per job
 - 4000h of CPU time accounted
- Results of DC:
 - http://dlib-services.isti.cnr.it/datachallenge/log_count_dlib.html
- 90% of total “production” of PPS (the rest is OPS)
- No big deal compared to HEP VO but continuity, availability, reliability of PPS required



- **Production area (PROD) = PS + PPS**
- **~ 11% of the number of sites in production area are PPS**
 - Does not mean 11% of WLCG/EGEE infrastructure manpower dedicated to PPS
- **~19% of bugs in production area are found in PPS**
- **This even with PPS not being used**
 - Bugs in PPS mainly submitted by PPS site admins
- **So, yes, the (few) PPS people are *in general* good at catching bugs**

- **PPS is (of course) highly effective in finding bugs in installation/configuration area**
 - ~46% of issues in middleware installation and “basic” configuration
 - Hard to decode numbers for configuration bugs (include user-level configuration and services often deployed bypassing PPS)
- **PPS ~ OK for bugs found in WMS area (28%)**
 - “Usage” by SAM tests and COD monitoring helps
 - Bugs pertaining to “task-force” prod installations not accounted to PPS
- **Effectiveness of PPS “bug scouting” decreases for bugs in Data Management area**
 - 11% LFC, 9% FTS, 3% DPM
 - First cause: poorly used in PPS
 - Second cause: frequently deployed by-passing PPS (because poorly used in PPS)



PS: Production Service; PPS: Pre-Production Service

PROD Area:	PS + PPS	
#PPS sites:	30	(#sPP)
#PS sites:	241	(#sP)
PPS contrib.	11%	(#sPP/(#sP+#sPP))

Better than scale ratio

to be decoded

Numbers extracted on 28-Sep-07

from Savannah bug tracker

(start date: Oct 2004)

	category	#tot bugs	in PPS	in PS	% bugs in PROD (PS+PPS)	PPS contrib. (inPPS/inPROD)
	All	5470	250	1099	25%	19%
Admin. area	install + config	962	55	68	13%	45%
	install	93	16	28	47%	36%
	configuration	364	39	197	65%	17%
Functional area	WMS	641	36	91	20%	28%
	Data Management	590	15	111	21%	12%
	R-GMA	570	16	82	17%	16%
	LFC	41	4	33	90%	11%
	FTS	106	7	69	72%	9%
	DPM	51	1	39	78%	3%

Poor testing

Poor usage

- **PPS is there, operated as a service**
 - Fulfilled (installed, maintained, debugged)
 - Assured (continuity/availability/reliability cared)
 - Accounted
- **The service runs mostly unused**
- **This is a fact**

- **Skipping here PPS costs/benefits considerations ...**
 - These will be object of the SA1 Operations session
- **Poor usage of PPS has a negative impact also on one other key mission of PPS**
 - “**Improving the quality of the gLite Middleware**”
- ... and, in a word, on our performance

- **We asked the LCG VOs for input to understand why PPS is not used:**
 - Existing technical barriers
 - Needed improvements.
- **One written reply from LHCb**
- **One meeting with CMS**
- **Meeting with Atlas after EGEE07**
- **Alice present today**

- **Both LHCb and CMS agree on manpower as the main issue:**
 - A lot of effort needed by the VO to maintain and operate two parallel submission infrastructures in two “universes”
- **LHCb: Size of PPS “by definition” does not allow to spot problems**

- **Clients**

- Early distribution: as soon as built and module-tested by developers
- Always backward-compatible to be tested by the VO against production services

- **Services**

- Available in production BDII but “flagged” as PPS
- By default not used by other production services
- CEs and SEs to see the same Back-end resources as in production

- **Shares with LHCb the idea of deployment in production of “flagged” PPS services**
 - GlueStatus != ‘Production’
- **Staging of deployment to production**
- **“Task-force” usage model**
 - very focused and on-demand bursts of activity involving a limited number of PPS service instances
 - no strict need for service continuity out of these “peaks”
- **Proposal to make CMS test suites available in PPS**
 - Need for someone (in PPS) to run and check them

- **Questions after Patricia's talk, please**
- **Thanks**