

One year EBA

-

Proceedings on collaboration between EGEE and Platform Computing

Bernhard Schott, Platform Computing
EGEE'07 / 1st Oct 2007

- **EBA**
 - EGEE Business Associate Platform Computing
 - About Platform Computing
 - Target: Exploit LSF better
- **Milestones**
 - Install a test infrastructure at Platform labs
 - Investigate on potential improvements in LSF usage
 - The findings
- **Challenges**
 - Topics to work on
- **Conclusion**

EBA Platform Computing

- **The Collaboration**

- The collaboration is part of the EGEE Business Associate Program.

<http://www.eu-egee.org/egee-business-associates>

- The start of the collaboration has been announced at the EGEE'06 conference, September 2006, Geneva.

- **Joint basis & experience**

- Several Platform Computing customers participate in EGEE
- More than 30% of the total compute capacity is provided by Platform LSF sites.

- **Platform Computing**

- Founded 1992 by Prof. Songnian Zhou and his students (University Toronto)
- First 1000 node cluster in productive industry use: 1991
- Today, >500 people, ~200 developers
- Today, >2000 customers worldwide
- All verticals: EDA, Research, Automotive, Aerospace, Finance, Telco, BI, ...
- >1 million licenses sold & in support
 - *We do nothing but Grid computing*
- Platform in other EU-Projects:
 - Participating in FP6 project QosCosGrid
 - Co-Allocation project for DEISA

- **Platform LSF – the heavy workhorse of EGEE**
 - In use at CERN since mid '90s
 - Several other EGEE sites
- **Customer quality requirements**
 - Reliability (self-healing, recovery from incidents, policy driven proactive problem containment, **no** job loss during operation or in error condition, while reconfig or failover.
 - Performance (10millions jobs per day throughput with >90% job-slot utilization based on 15min job-runtime, max 5min for failover, >100 jobs/s submit rate)
 - Scalability (n*1000's users, n*1000's hosts, n*millions jobs in one logical cluster at any time, n*10millions jobs per day throughput, n*1000's way-parallel jobs)
- **Performance and Scalability translates into Reliability**
 - Reliability can be measured as “MTBF” –
Mean Transactions (=Jobs) Between Failure

- **10 million Jobs/day**
with a single LSF scheduler instance
... without any job loss.
- **Performance and Scalability translates into Reliability**
 - Reliability can be measured as “MTBF” –
Mean Transactions (=Jobs) Between Failure

- **Goal of the collaboration**

- The goal of the collaboration is to improve the utilization of LSF within EGEE through improving the gLite interface to LSF, the exploitation of advanced LSF features, coordinated testing efforts, and improved documentation.
- Resulting from initial discussions, we expect significant enhancement of capabilities for EGEE, both for users, new user groups as well as for resource providers, compute- and data-centers.

- **Continuous collaboration**

- Based on the progress made a joint project submission of Platform and EGEE partners within OMII-EU-II. Potential areas for such a project were clarified in course of the collaboration.
- Joint research with INFN, CERN, KTH, PSNC, FZJ

- **Gather experience with gLite by installing a test infrastructure**
 - at Platform labs
 - Onsite work at INFN

- Done
 - Open

- **Test installation tasks**
 - Create a Platform lab test environment for gLite for the full user to execution delivery path

- Failed

- **Investigate on potential improvements in LSF usage**
 - CE -> LSF interface
 - Utilisation of advanced LSF features

- New CE design
 - See next pages

- **Packaging of full gLite infrastructure**
 - gLite DVD **DONE ?**
 - “Resource provider package” (CE, ...) ?
 - “Grid Services” (WMS, AAA, ...) ?

- **Professional support**
 - Implementation & Operations support
 - for research users covered by ROCs
 - Offer for industry users? Security requirements!

- **Relevance for EGEE:**
 - growth beyond current structures
 - technology exploitation beyond THE EGEE Grid

- **Handling of heterogeneity in resources, usages, limitations, ...** (32bit/64bit, OS-version, RAM size, I/O type, ...)
- Use of LSF-Queues as organisational method.
 - Have as many queues as you like – it's for free!
- 1:1 mapping of CE to queue disables queues as organisational method.
- Options:
 - Use queues for
 - *prio & pre-emption*
 - *Resource mapping*
 - *Limit assignment*
 - bootstrapping LSF options – forward `-q <queue-name>` parameter
 - Esub ... to map VO internal roles to prio-sorted queues

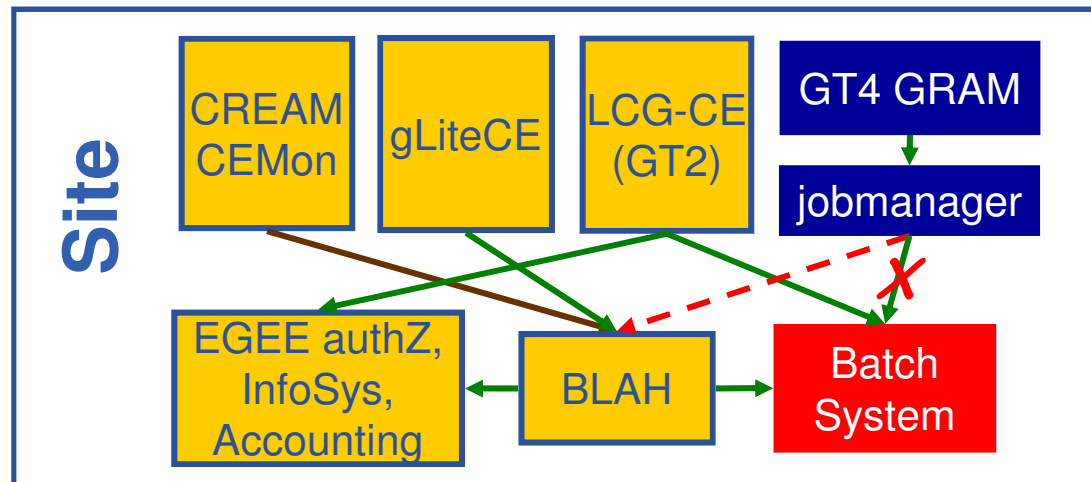
- **Handling of heterogeneity in resources, usages, limitations, ...**
 - Use of detailed resource descriptions / job specific resource requirements by bootstrapping LSF options
 - forward `–R <resource specifications> parameter`
 - Example:


```
bsub –R"select[type==SL4 && swap>=300 && \
                mem>500] order[swap:mem] \
                rusage[swap=300:mem=500]" job1
```
 - More options: `decay`, `span[hosts=n]` `span[ptile=n]`, `same`
 - Preferences: - “soft” scheduling advice
 - **HOSTS = hostGroupB+5 hostGroupA+2 others**
 - Syntax example given for queue configuration – for user cmd similar

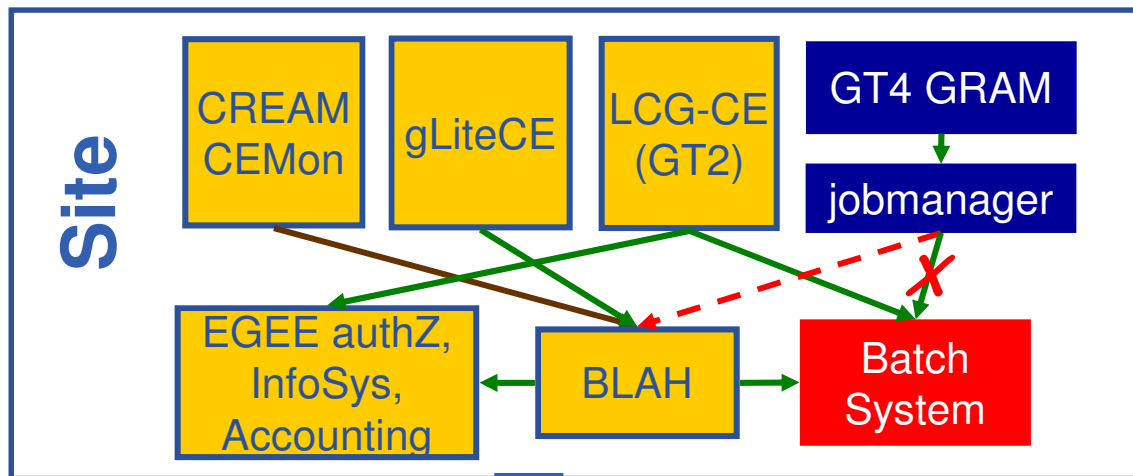
- **Performance improvement:**
 - offload WMS – “down-port” of compound job execution
- **DAG support (Direct Acyclic Graph)**
 - Down-port to LSF job dependencies
 - User selectable option
- **Parametric Jobs**
 - Translate to LSF Job-Arrays
 - User selectable option
- **Collections**
 - Map to LSF Job Groups
 - Enable advanced scheduling policies
 - User selectable options

- **CE speedup**

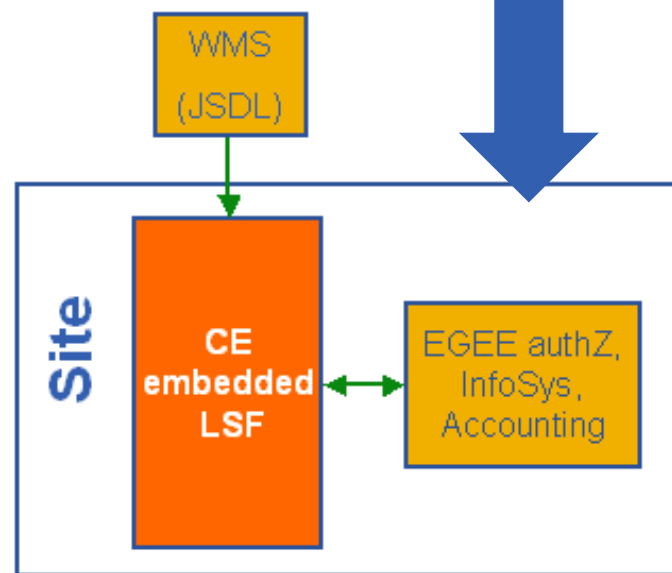
- LSF7 JSDL support
- New CE uses JSDL/BES
- How would interfacing to LSF look like?
 - → *CE would become smaller*
 - → *no BLAH*
 - → *reduced maintenance effort !*



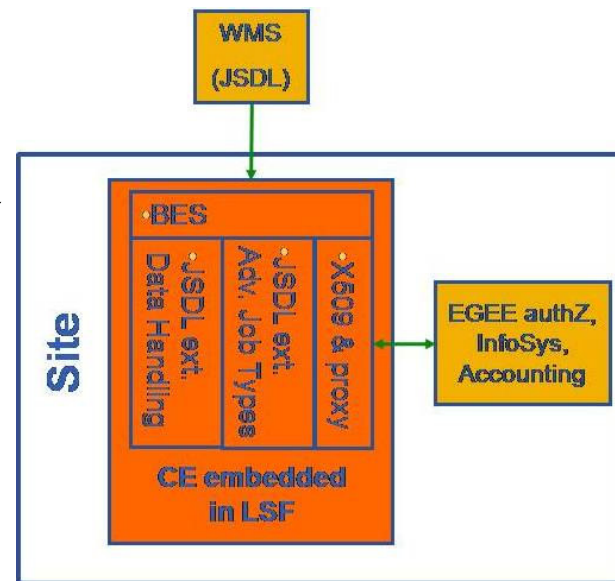
•From Claudio Grandi



- From Claudio Grandi
- Current CE-Implementations and planned or ongoing work



- **LSF-CE**
OGSA/BES JSDL



ndi

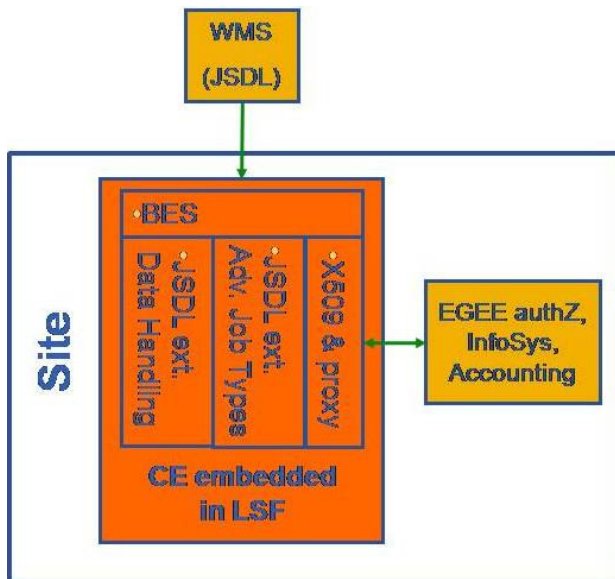
- **LSF-CE**

OGSA/BES JSDL

Still some work to do:

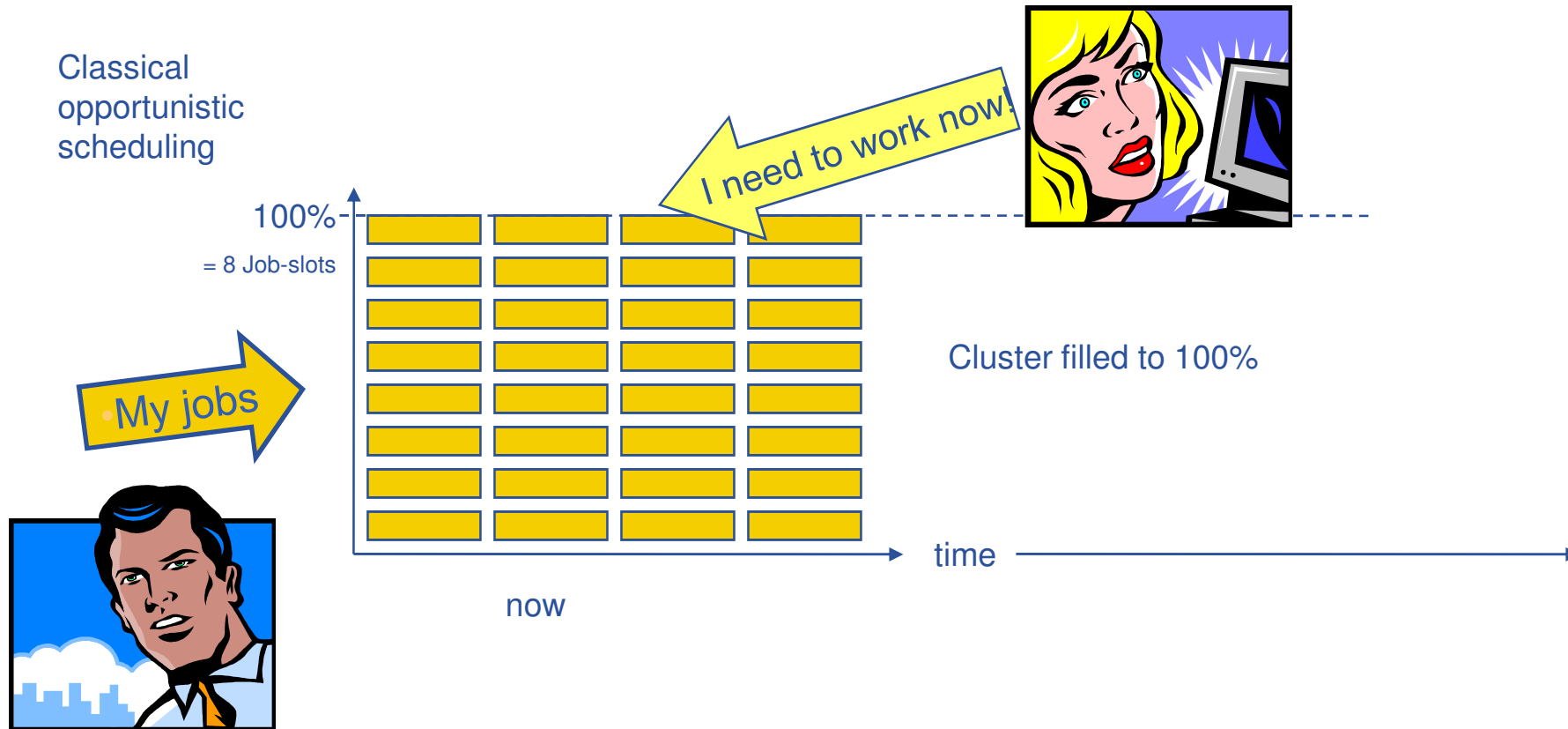
- BES/JSDL: Investigation on dynamic balanced performance and stability
- Multiple BES instances
 - Add performance
 - Share load
 - Failsafe
 - Service-Window
- JSDL extensions
 - Data handling
- Security Integration
 - Performance
 - Stability

• From Claudio Grandi

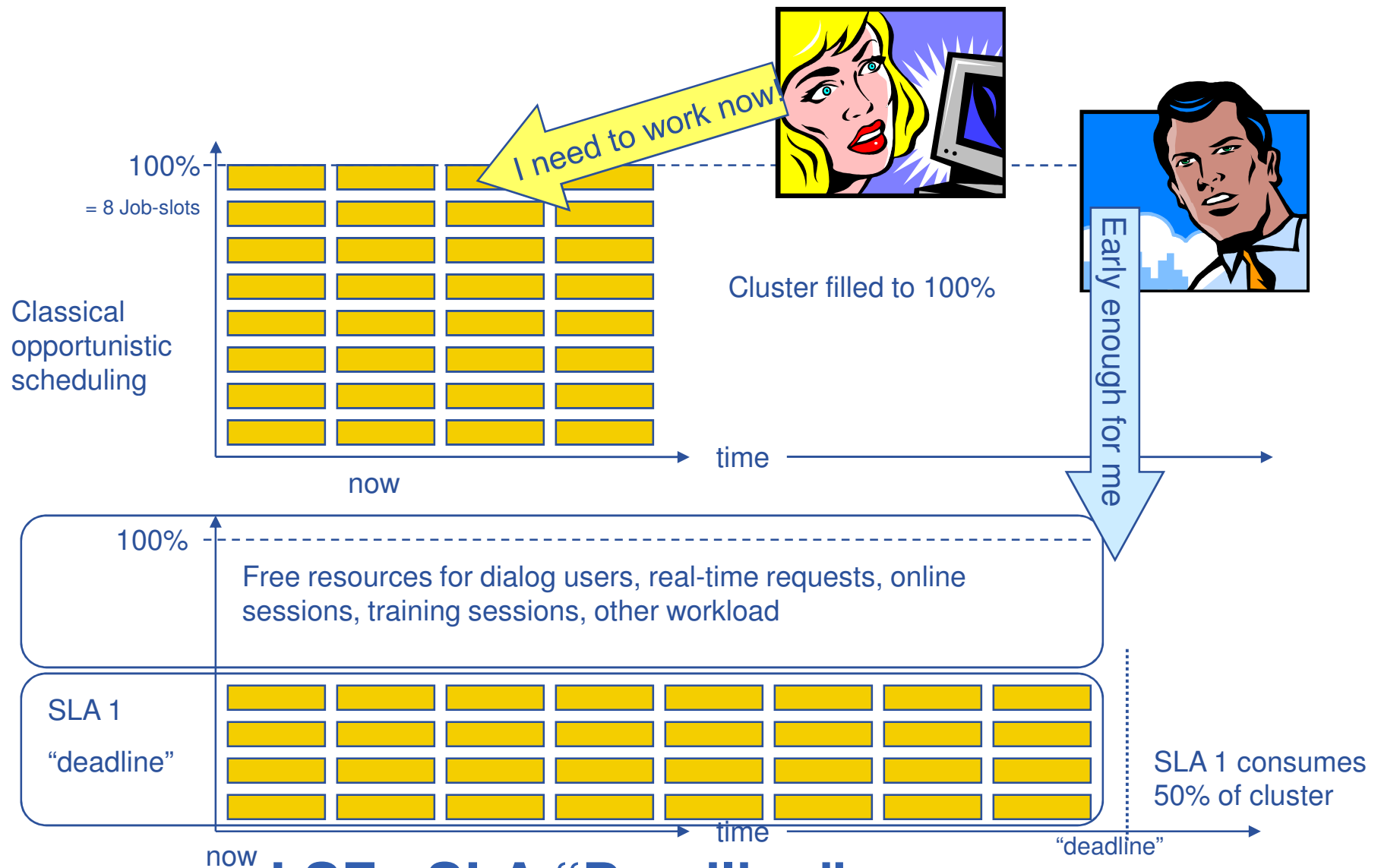


- **Enable SLA scheduling for Job Groups**
- **LSF implements SLAs for workload submitted to a “service class” (e.g. deadline, throughput, speed, ...)**
- **To resource providers, this translates into “least impact scheduling”**
 - A given set of resources (= a EGEE resource provider center) can serve more “happy” customer at the same time
 - = participate in more projects at the same time
 - = more funding sources
- **Difficult? Not on the resource side: its in LSF product since 2003 – “just switch it on”**

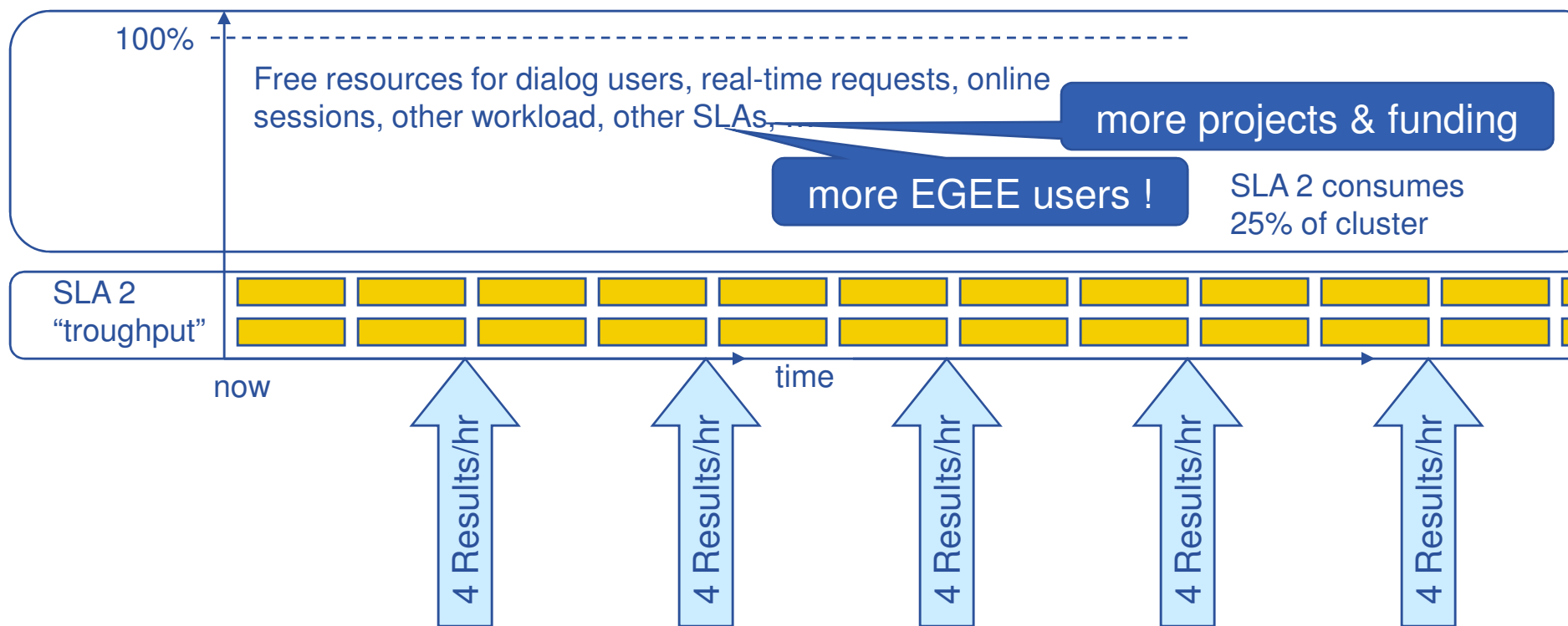
Classical opportunistic scheduling



LSF-SLAs for EGEE: SLA “Deadline”



LSF-SLAs for EGEE: SLA “Throughput”



I am a scientist, I need just as many results as I can process per time interval.

- **Conclusion**

- Under preparation:

- Standards based LSF-CE → EGEE performance boost
 - New job-types & policies → meeting user needs & boosting performance

- Not finished yet / slowing Industry usage or technology uptake

- Ease of use & ease of operations (in heterogeneous environments)
 - Low operations costs, stability against component failures
 - What about MS-Windows?
Hundreds of millions Windows boxes in Europe are waiting for EGEE
 - Execution site safety
 - *application containment, tight firewalls both inwards and outwards.*
 - *Predictable usage (no self-deployment, no pilot jobs).*

Thank you!

It's great working with you!

*Bernhard Schott
EU-Research Program Manager
Platform Computing
bschott@platform.com*