Contribution ID: **516** Type: **Oral**

Adjusting the fairshare policy to prevent computing power loss

Monday, October 10, 2016 3:30 PM (15 minutes)

On a typical WLCG site providing batch access to computing resources according to a fairshare policy, the idle timelapse after a job ends and before a new one begins on a given slot is negligible if compared to the duration of typical jobs. The overall amount of these intervals over a time window increases with the size of the cluster and the inverse of job duration and can be considered equivalent to an average number of unavailable slots over that time window. This value has been investigated for the Tier-1 at CNAF, and observed to occasionally grow and reach up to more than the 10% of the about 20,000 available computing slots. Analysis reveals that this happens when a sustained rate of short jobs is submitted to the cluster and dispatched by the batch system. Because of how the default fairshare policy works, it increases the dynamic priority of those users mostly submitting short jobs, since they are not accumulating runtime, and will dispatch more of their jobs at the next round, thus worsening the situation until the submission flow ends. To address this problem the default behaviour of the fairshare have been altered by adding a correcting term to the default formula for the dynamic priority. The LSF batch system, currently adopted at CNAF, provides a way to define its value by invoking a C function, which returns it for each user in the cluster. The correcting term works by rounding up to a minimum defined runtime the most recently done jobs. Doing so, each short job looks almost like a regular one and the dynamic priority value equilibrates to a proper value. The net effect is a reduction of the dispatching rate of short jobs and, consequently, the average number of available slots greatly improves. Furthermore, a potential starvation problem, actually observed at least once is also prevented. After describing short jobs and reporting about their impact on the cluster, possible workarounds are discussed and the selected solution is motivated. Details on the most critical aspects of the implementation are explained and the observed results are presented.

Tertiary Keyword (Optional)

Secondary Keyword (Optional)

Computing models

Primary Keyword (Mandatory)

Distributed workload management

Primary author: DAL PRA, Stefano (INFN)

Presenter: DAL PRA, Stefano (INFN)

Session Classification: Track 3: Distributed Computing

Track Classification: Track 3: Distributed Computing