SLAC preliminary comparison of LoadLeveler and LoadSharingFacility 10/10/94

Feature	LoadLeveler	Load Sharing Facility
Centralized scheduling	yes	yes
Balance batch load across the entire pool of machines, subject to scheduling definitions.	yes	yes
Balance interactive load across the pool of machines.	no	yes, if ls commands used for interactive
Make desktop machines available during idle periods with "owner" control over the availability conditions	yes	yes
Monitor system parameters including CPU load, paging rate, memory, I/O, logins, etc.	uses load average	yes
Job limits for various resources	yes (see next)	yes
Facility for parameters such as CPU time to be "normalized" across different CPU models so that time limits, accounting, etc. present the same units to the user regardless of where the job runs.	no	yes
System setup to be installed with shareable binaries, logs, etc. so that stuff installed on local machines is minimized.	yes	yes
Handles jobs accessing files in nfs, afs, and other appropriate file systems.	nfs/afs separate to be combined next release	appears to be separate for nfs/afs currently
GUI for configuration	no	supposedly but not in the version we have
GUI for job and machine status	yes	yes
GUI for job submission	yes	yes
Job flooding limits	limit on number	limit on number of jobs

	of jobs "in the queue" can be specified	per user per class. Can also limit by "group" but users in group must be defined in a table.
Schedule based on throng allocations	no, under consideration for next year	yes but rudimentary;
Sub-divide administrator authority to allow throng thanes to manage their own group of machines.	no	separate administrators can be defined for each machine "cluster"
Count "consumable" resources such as tape drives, software licenses, etc.	no	yes, but some may need local options based on locally-written script to determine usage
Automatic switch of the "master" machine if the master goes down.	supposedly in latest version	yes
Command to list defined "features" and job classes	no. available classes shown in gui	yes
Ability to define run-time priorities	no	yes, nice values settable by class
Scheduling time windows	by machine	by machine or class
Sked1/Sked2 concept of defining windows, etc. which can be different for weekdays/weekends.	yes	yes
Define differences for holidays	no	no
Ability to change job class after submission but before execution	no	yes
Automatic purging of old logs	yes	no, must be done by cron or manually

---

Ability to pre-empt long running jobs for a "secondary user" of a machine when a higher priority job for a "primary user"	no	no, although some of this can be done with careful definition of

enters the system.		sked parms
Notification to user job when it is about to be pre-empted, suspended, or killed.	no(?)	signal passed to program
Accounting data by job and machine.	some new stuff in latest version but limited info available	yes
Job status information after the job has completed execution.	no	yes
Job sequence dependencies	something has been added to new release	yes