

# HPC Interactive Job Licensing Checklist

An interactive job holds its licence for the length of the job slot, not the length of the work. Because the licence server sees one continuous checkout, none of that gap appears in a peak-concurrency figure or a utilisation report. **Twenty-two checks to find it, instrument it, and agree the rules before anything is reclaimed.**

## A Find what the slots are holding

- ☐ List every queue that accepts interactive submission (`bsub -I`, `-Is`, `-Ip`, `-XF`) and who is entitled to use it.
- ☐ For each queue, name the applications that check out a licence from inside the slot rather than from the workstation.
- ☐ Record the licence feature and vendor daemon exactly as the licence server names them — for Ansys under FlexNet, `ansyslmd`.
- ☐ Pull wall-clock duration per interactive job from scheduler accounting.
- ☐ Pull the licence-side duration to match: polled status gives hourly concurrency, the vendor daemon debug log gives per-session checkout and check-in events.
- ☐ **Subtract.** The difference between the job window and the work inside it is the block nobody has measured.

## C Agree the rules before anything is reclaimed

- ☐ Set `timeout` per application family rather than globally — a meshing session and an overnight solve do not idle alike.
- ☐ Set `irq`, the window in which the engineer is warned through the disabler interface first.
- ☐ Put protected work on `priority: total`: never disabled, inactivity logged and nothing else.
- ☐ Gate harvesting with a `license-use` rule so it engages only when the pool is genuinely under pressure.
- ☐ Run `method: log` for a full working cycle, then `standby`, and only then `suspend`.
- ☐ Decide `suspend-timeout` (default P6H) and its action — `resume`, `terminate` or `none` behave very differently at hour six.

## B Instrument the job slot

- ☐ Confirm the estate is **Unix** — the documented procedure for interactive LSF jobs is Unix only.
- ☐ Deploy `openit_job_starter.sh` from `/opt/openit/bin` so FreezeMonitor can identify which executables were launched inside the job slot.
- ☐ Activate collection on the execution hosts, not only the licence server. Activity is measured where the application runs.
- ☐ Verify all four activity signals are recorded: CPU utilisation, I/O operations, keyboard strokes, mouse interactions.
- ☐ Confirm the estate is on a baseline that includes metering for interactive jobs in LSF.

## D Prove it, then keep proving it

- ☐ Where a save-before-release is wanted, configure a `pre` custom action scoped with `exe=`, and check its `timeout` (default P5M) is long enough for your model sizes.
- ☐ Re-run the block A subtraction after one cycle — same queues, same features, same window length.
- ☐ Watch suspend-action counts per application. An unexpected jump means a threshold is mis-set, not that engineers changed their habits.
- ☐ Report concurrency per application version, so renewal is sized against the combined curve rather than a per-baseline seat count.
- ☐ Review month one with the engineering leads who own the queues, not only with the licence owner.

**The one number this checklist is chasing.** Held seat-hours minus worked seat-hours, per queue. Everything above exists to make that subtraction possible and to make the result defensible to the engineers whose jobs produced it. No recovery percentage is assumed here — on an HPC-heavy programme this is usually the largest single block of recoverable capacity, and it is also the one least visible in the reports you already have.

### SOURCES — OPEN IT ONLINE DOCS 10.2

Optimizing Interactive Jobs in LSF · [docs.openit.com/10.2/license\\_analyzer/level3/optimizing\\_interactive\\_lsf\\_jobs/](https://docs.openit.com/10.2/license_analyzer/level3/optimizing_interactive_lsf_jobs/)  
 Optimizing Application Usage — `cold_master.conf` parameters · [docs.openit.com/10.2/license\\_analyzer/level3/optimizing\\_application\\_usage/](https://docs.openit.com/10.2/license_analyzer/level3/optimizing_application_usage/)  
 LicenseAnalyzer™ Level 2 and Level 3 · [docs.openit.com/10.2/license\\_analyzer/level2/level3/](https://docs.openit.com/10.2/license_analyzer/level2/level3/) · FlexNet License Manager · [level1/flexnet/](https://docs.openit.com/10.2/license_analyzer/level1/flexnet/)

### NEXT STEP

**A utilisation snapshot runs the block A subtraction against your own queues.**

Talk to a licensing specialist · [openit.com](https://openit.com)