QUARTZ LIMS WHITE PAPER

Governing Capacity

Using Allocations and Limits to Make Throughput Predictable and Intake Fair

For laboratory managers and operations leaders in failure analysis and materials characterization labs

Executive Summary

Laboratory capacity is finite, but demand rarely is. Requestors submit work according to their own deadlines, not the lab's, and without a mechanism to govern intake, a lab absorbs whatever arrives. The result is familiar: queues lengthen, turnaround times become unpredictable, and the lab is judged on commitments it never had the capacity to meet.

This paper makes the case for governing intake at the sample level through allocations and limits, rather than managing demand informally or after the fact. It explains how a quota mechanism that consumes capacity when work is submitted and returns it when work is withdrawn gives a lab predictable throughput and a fair, transparent basis for setting expectations with the people it serves. It closes with how Quartz LIMS implements this model.

The Problem with Ungoverned Intake

When a lab accepts every request as it arrives, capacity planning becomes guesswork. There is no point at which the system says a group has reached its share, no automatic protection against a single requestor flooding the queue, and no record that distinguishes committed work from speculative work. Managers discover overload only when the backlog is already visible, which is the most expensive moment to discover it.

The informal alternatives are worse. Spreadsheets that track who has submitted what are out of date the moment they are saved. Verbal agreements about fair shares are unenforceable. And first-come, first-served intake quietly rewards whoever submits earliest rather than whoever has the most important work. None of these scales, and all of them push the cost of poor planning onto turnaround time, which is the metric customers feel most directly.

Allocations as a Capacity Currency

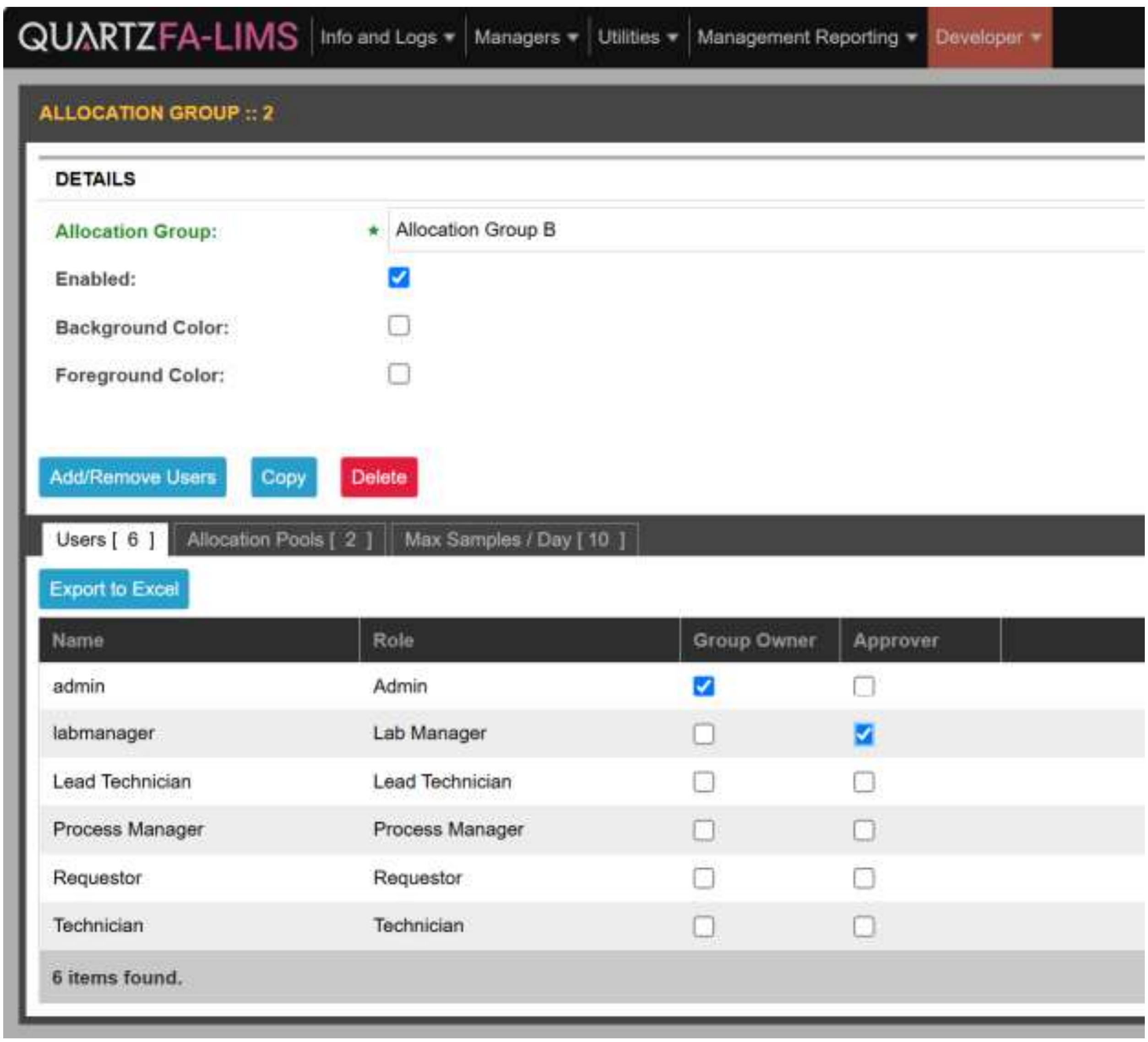
An allocation model treats capacity as a currency that is spent and reclaimed as work moves through the lab. Each requestor or group holds a balance, and submitting work draws against it. The key design decision is that allocations are consumed and returned at the sample level, because the sample is the true unit of work, and a job that carries many samples consumes more capacity than one that carries a few.

A well-designed allocation lifecycle behaves predictably at every step:

- Consumption on submission. When a job is submitted and sufficient allocations remain, the samples are accepted and the balance is drawn down. The lab knows, at the moment of intake, that the work fits within agreed capacity.
- Controlled handling of shortfalls. When a submission exceeds the remaining balance, the requestor is warned before committing. If the work proceeds without capacity, it is routed to a rejected state rather than silently joining the active queue, so the lab's working pipeline reflects only work it has agreed to perform.
- Return on withdrawal. When a sample is rejected or a job is withdrawn, its allocation is returned to the balance rather than lost. Capacity that is not consumed by real work becomes available again automatically.
- Adjustment on edit. When a submitted job is edited, its allocation is recalculated so the balance always reflects the actual committed workload.

The table below summarizes the lifecycle.

Event	Effect on capacity balance
Job submitted, capacity available	Allocations consumed for all samples; work enters the active pipeline.
Job submitted, capacity short	Requestor warned; if forced, job is rejected rather than queued.
Sample rejected or job withdrawn	Allocations returned to the balance for reuse.
Submitted job edited	Allocations recalculated to match the revised sample set.



Daily Limits and Priority Protection

Allocations govern the total volume a group may consume. Limits govern the rate at which work enters the lab on any given day, which is what protects daily throughput from sudden surges. Configurable per-job sample limits, set independently for different job types, prevent a single oversized submission from consuming a day's capacity, while a daily reset establishes a clean, predictable boundary between one operating day and the next.

Together, allocations and limits address two different risks. Allocations prevent any group from consuming more than its fair share over time. Limits prevent any single day from being overwhelmed regardless of balances. A lab that controls both has a defensible answer to the question every requestor eventually asks, which is why their work has not started yet.

Why This Belongs in the System, Not the Conversation

Capacity governance only works if it is automatic and visible. When the rules live in the system, the warning appears at submission, the balance updates without anyone maintaining it, and the record of what was committed is the same record the lab works from. A LIMS that serves as the single source of truth removes the gap between what was promised and what was planned, because both are derived from the same data.

This also changes the tone of the conversation with requestors. Instead of negotiating each request individually, the lab points to a transparent, consistently applied rule. Fair shares stop being a matter of who asks most insistently and become a matter of policy that the system enforces evenly.

When Limits Become Constraints

Capacity controls can be set too tightly. Allocations and limits that do not reflect real demand will push legitimate, urgent work into rejection, and a lab that frequently overrides its own rules teaches requestors that the rules are negotiable, which defeats their purpose. The mechanism is only as good as the balances and limits behind it, and those should be reviewed against actual demand rather than set once and forgotten.

It is also worth being clear about what allocations do and do not solve. They govern intake; they do not create capacity. If demand genuinely and persistently exceeds what the lab can perform, allocations will make that imbalance visible and fairly distributed, but the underlying answer is more capacity or fewer commitments, not a tighter quota. Used honestly, the model is a planning and fairness tool. Used to mask a structural shortfall, it simply relocates the frustration.

How Quartz LIMS Implements This

Quartz LIMS includes a quota and allocation mechanism designed for the multi-sample workflows of failure analysis and materials characterization labs. Allocations are consumed when work is submitted and returned when samples are rejected, per-job sample limits can be configured separately by job type, and a daily reset establishes a predictable intake boundary. Because the system is the lab's source of truth, the capacity balance and the working pipeline are always consistent.

As with every Quartz LIMS capability, these rules are configured to the lab's own structure and terminology rather than imposed as a fixed scheme, so the allocation groups, limits, and reset timing reflect how a given lab actually distributes its capacity. The result is predictable throughput and a fair, transparent basis for the commitments the lab makes.

To see how capacity governance would map to your lab, learn more at quartzlims.com or request a walkthrough with the Quartz team.

