The Need for Speed: Throughput Defines The Fastest Time To Results

YellowDog's ability to deliver high scale compute at hundreds-of- thousands-of-nodes scale is already giving global financial services and Gen AI firms a measurable competitive edge. As parallelisation deepens and task volumes explode, a new vector becomes decisive: **sustained task throughput**.

Developed in close collaboration with our financial services customers YellowDog is excited to announce the public release of High Throughput Scheduling (HTS), our latest innovation which now governs how quickly large-scale compute is converted into actionable results. This article lays out clear evidence showing why scheduler throughput combined with intelligently orchestrated compute determines faster completion times at scale.

Instant High-Scale Compute Infrastructure, Proven in Production

As compute-intensive workloads scale across quantitative finance, GenAI, and large-scale simulation, YellowDog has consistently demonstrated its ability to deliver at high scale and orchestrating tens, and hundreds of thousands of nodes to compress wall-clock time and accelerate insight.

As an example of this drive for faster results, the diagrams below show 2 aspects of a real production workload from a global hedge fund quant team running millions of tasks on YellowDog. In under an hour of wall-clock time, YellowDog orchestrated 61,998 compute nodes, using 32 distinct instance types, dynamically scaling capacity up and down as the workload progressed. The platform coordinated Intel, AMD, and Arm instances across multiple AWS EC2 regions, intelligently blending Spot and On-Demand capacity to sustain throughput at huge scale.

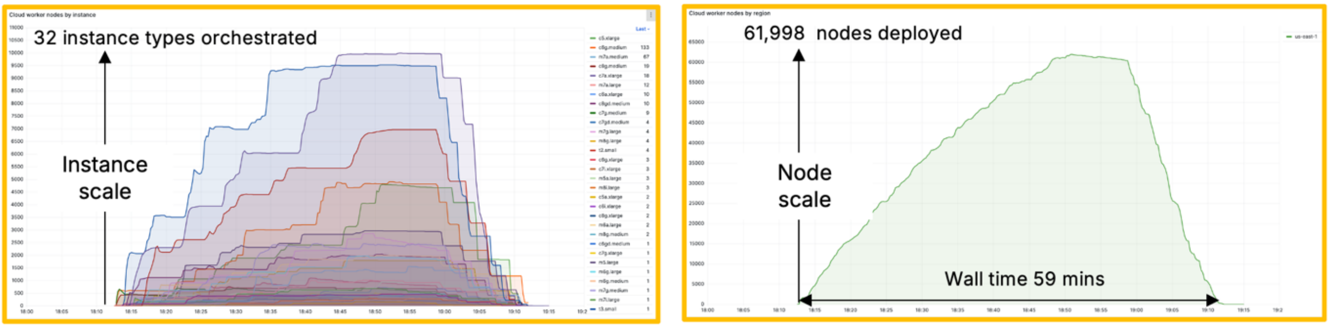

Fig 1. Global hedge fund quant team production workload on YellowDog

What makes this significant is not just the headline number of nodes, but the orchestration capability behind it. YellowDog continuously selected and provisioned the most appropriate capacity across instance families, absorbing Spot interruptions and infrastructure churn without collapsing throughput. Tens of thousands of nodes were brought online rapidly and retired cleanly at completion — compressing 41,436 compute hours into just 59 minutes of elapsed time. At this scale, only YellowDog can manage the multi-region cloud compute capacity needed to sustain these workloads and deliver your competitive advantage.

Introducing YellowDog High Throughput Scheduling (HTS)

While raw compute remains the primary currency of scale, its real value is defined by how efficiently it can be converted into completed work. This factor for improved compute efficiency becomes important when task volumes and parallelism increase, implying execution speed depends not just on access to capacity, but on the ability to continuously schedule and feed work into that compute at **sustained**, **high throughput.**

The need for sustained high throughput reflects a reality where billion-task workloads are becoming standard in Financial Services, particularly in Capital Markets. FRTB, RWA, and XVA runs already fan out into tens or hundreds of millions of independent tasks and are rapidly approaching — and exceeding — the billion-task mark. At that scale, speed is governed not by capacity alone, but by how efficiently work can be scheduled and completed across a massively parallel fleet, especially in intraday trading environments where time windows are tight and iteration drives advantage.

Built specifically to meet the demands of Financial Services customers, YellowDog is introducing **High Throughput Scheduling (HTS).** HTS is an integral component of the YellowDog platform, engineered to deliver faster workload execution at high scale using vastly superior throughput speeds when compared to traditional schedulers.

To demonstrate the real-world impact of High Throughput Scheduling, the charts below benchmark a real production run of **100 million one-second tasks**, executed on a **200-node / 40,000-worker cluster using YellowDog**:

- Sustained throughput at 40,000+ tasks per second
- >99% compute utilisation
- Multi-region execution
- 100% Spot capacity
- Heterogeneous Intel, AMD and Arm instances
- Total runtime: 41 minutes

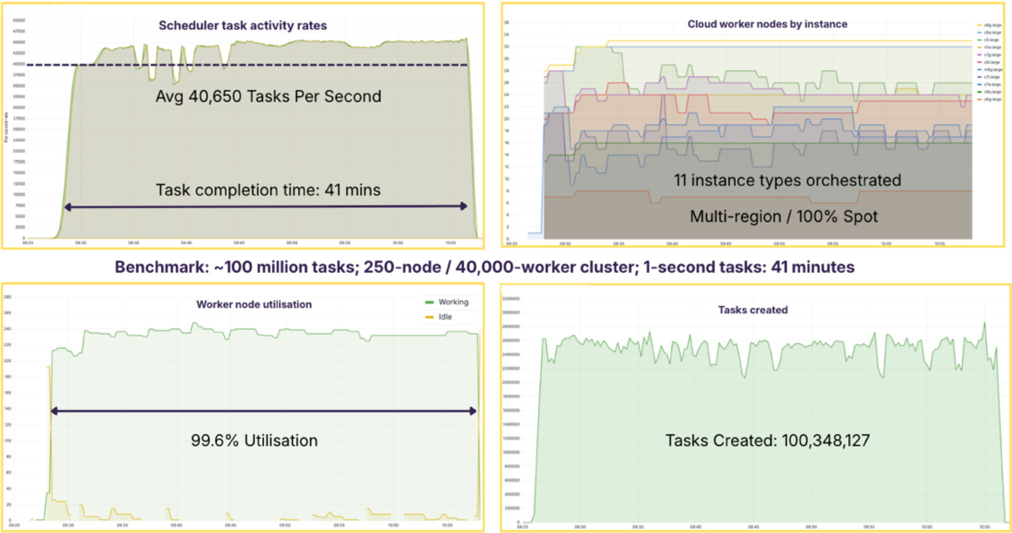

Fig 2. YellowDog High Speed Throughput benchmarked at 100 million tasks

What matters here is not only peak rate, but **sustained execution under real-world conditions**. Despite Spot pre-emption and node replacement, throughput remains bounded and workers stay saturated at close to 100%.

As concurrency scales into the hundreds of thousands, every stage of task allocation and execution must be optimised for sustained throughput. Without it, unallocated or queued work becomes the limiting factor on overall completion time. YellowDog removes this constraint by using intelligent allocation and absorbing infrastructure volatility at scale. This ensures continuous worker saturation, maximising compute utilisation, and preserving execution speed.

Why Throughput-Optimised Architectures Win at Scale

Throughput-optimised architectures are now the defining model for high-scale, heavily parallelised workloads. YellowDog uniquely combines large-scale compute orchestration with high throughput scheduling to give teams **predictive control over execution**. This allows Infrastructure teams to provision massive clusters as part of a normal workflow and use YellowDog to:

- benchmark workloads before execution
- predict completion time down to the minute
- model cost efficiency across instance types and architectures
- select optimal hardware mixes (CPU, GPU, ARM, x86)
- plan execution windows with confidence

By sustaining near-continuous utilisation across heterogeneous, multi-region fleets — including Spot capacity with advanced pre-emption handling, YellowDog converts raw scale into **repeatable, measurable outcomes**.

Are You Getting The Most From Your Compute Infrastructure?

If you want faster results: the question is do you need more compute or more throughput? The table below helps you identify how to actually accelerate execution.

Infrastructure Signal	More Compute	More Throughput
Clusters are full but queues keep growing	✓	
Completion times are unpredictable despite scaling out	✓	✓
Idle compute dominates cost at high scale		✓
Workloads are bursty, heterogeneous, or Spot-heavy	✓	✓
Adding cores no longer reduces wall-clock time		✓
Workload size is growing beyond current capacity limits	✓	
Trading deadlines mean workloads must complete by a specific time	✓	✓
Researchers need to run more iterations within the same window	✓	✓
You are hitting hard resource ceilings (vCPU/GPU limits)	✓	

Interpreting the signals:

- If most of your checks fall under **More Compute**, additional capacity may unlock faster results.
- If most fall under **More Throughput**, execution speed is likely governed by scheduling efficiency rather than raw capacity.
- If checks appear for **both**, your growth in scale may require a combined strategy — aligning elastic capacity with sustained throughput to fully convert compute into speed.

Whether your acceleration comes from scaling capacity, increasing throughput, or combining both, YellowDog is designed to deliver the orchestration and scheduling performance that translates your quant team's workloads directly into faster results.

Crucially, this does not require disruptive refactoring. YellowDog can be introduced retrospectively or incrementally and easily evaluated against your real workloads to quantify performance and benchmark for speed. This allows infrastructure teams to test at scale, validate improvements in completion time and utilisation, and move forward with confidence, without risk.

The Need for Speed: Is Lack of Throughput Costing You at Scale?

At the highest task volumes, throughput stops being incremental — it becomes decisive.

If your need for speed defines your competitive edge — it's time to talk about how YellowDog High Throughput Scheduling can help you deliver predictable, high-scale execution at the pace your workloads demand. Reach out to us here to start a conversation.

In a following article, we will compare the leading schedulers on sustained throughput performance and examine what those differences mean for completion time, hardware flexibility, and overall compute cost efficiency. The analysis will show how throughput translates directly into faster results, across any chosen mix of CPU or GPU instance types, and what that implies for infrastructure teams responsible for delivering performance at scale.