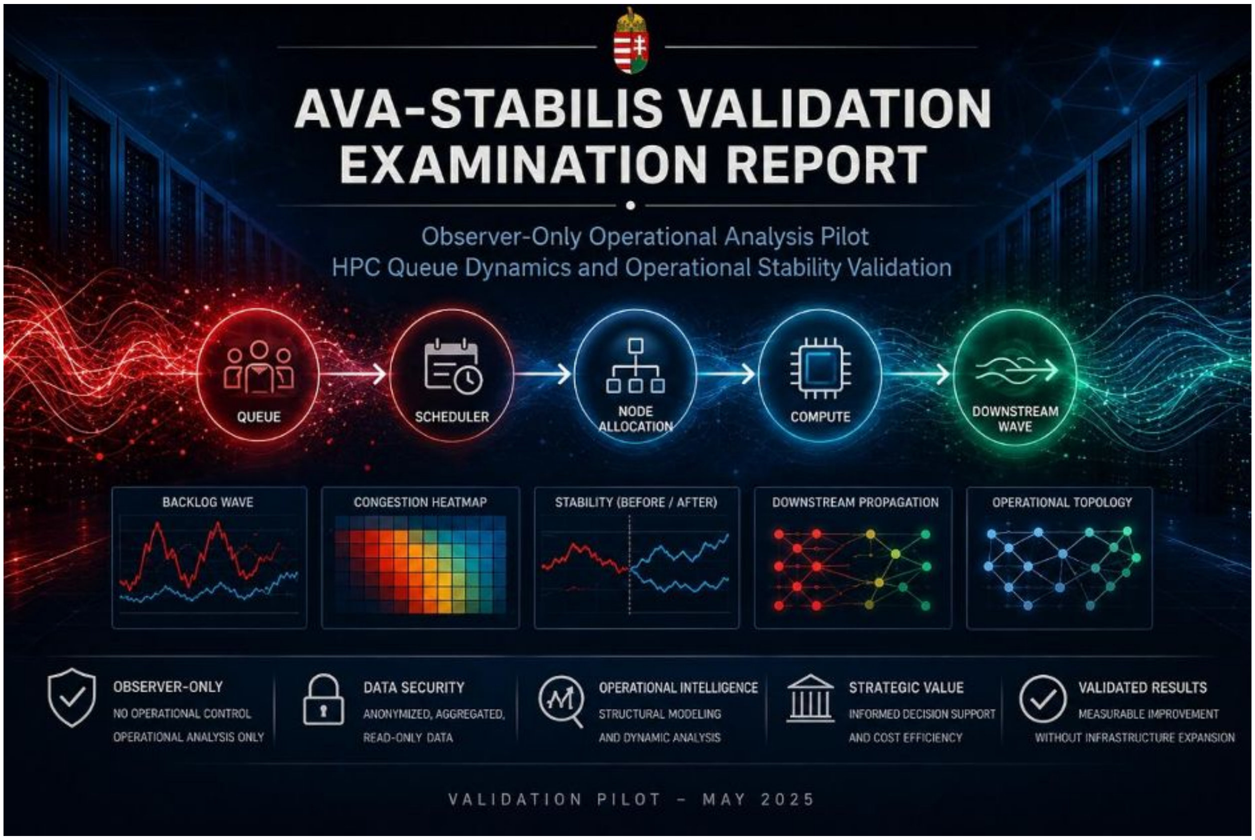
AVA-STABILIS

Operational Synchronization Modeling Report
Observer-Only Operational Analysis Pilot



This document is a partially modeled operational-analysis sample report. It is designed to demonstrate the AVA-Stabilis methodology and does not represent an audited customer deployment or a peer-reviewed scientific validation.

PILOT NAME

Batch HPC Operational Stability and Queue Dynamics Validation

---

ANALYZED DOMAIN

High Performance Computing Environment (HPC)

PILOT TYPE

Observer-only operational analysis and validation pilot

DATE

May 2025

VERSION

v1.0 – Validation Partner Edition

CONFIDENTIALITY LEVEL

For partner use only
Anonymized validation document

OBSERVER-ONLY STATUS

The assessment was conducted:

- in a read-only approach,
- without operational intervention,
- without infrastructure modification,
- based on anonymized operational data,
- within an observer-only model.

During the pilot:

- no direct system control was performed,
- no sensitive data was processed,
- no service-impacting modifications were introduced.

PARTNER STATUS

Anonymized HPC partner environment
(name and location redacted)

0. EXECUTIVE SUMMARY

Objective

The objective of the pilot was to analyze the operational dynamics of a high-load, batch-oriented HPC environment using an observer-only approach, with particular focus on:

- queue load patterns,

- backlog wave formation,
- runtime fragmentation,
- downstream operational distortions,
- and the identification of hidden capacity losses.

The focus of the assessment was not infrastructure expansion or hardware optimization, but rather the analysis of how the operational structure of the system contributed to coordination instability and efficiency degradation.

Assessment Period

The validation assessment was conducted across multiple operational cycles and varying load conditions.

During the pilot, the following activities were performed:

- baseline operational state assessment,
- operational pattern analysis,
- observer-only simulation,
- and validation reruns.

Primary Operational Problem

The assessment was initiated due to the presence of:

- high queue waiting times,
- wave-like backlog formation,
- significant workload fragmentation,
- and periodic downstream congestion.

Based on the initial operational picture, the primary issue appeared to be infrastructure limitation. However, during the assessment it was determined that the main sources of instability were:

- queue dynamics,
- runtime variance,
- batch organization,
- and operational coordination distortion.

Key Findings

1. The system was not primarily compute-limited

A significant portion of the operational instability originated from queue and fragmentation dynamics.

2. Backlog waves generated downstream distortions

Long-running workloads created blocking effects for smaller jobs at multiple points, resulting in wave-like congestion.

3. Significant Hidden Capacity Loss was present

A substantial gap was identified between node utilization and actual operational efficiency.

4. Restart and retry chains created self-reinforcing instability

Repeated restart events generated periodic operational noise and synchronization loss.

5. Operational improvement was achievable without infrastructure expansion

During validation, operational stability improved through structural fine-tuning without the addition of new hardware.

Most Important Validated Results

Metric	Baseline	Validated Result
Queue wait	9.6 h	6.1–7.4 h
Node utilization	74%	81–86%
Retry ratio	11.3%	7.2–8.9%
SLA fulfillment	81%	88–92%
Backlog oscillation	High	Reduced
Operational stability	Unstable	More stable operation
Energy/workload ratio	Distorted	More balanced

Key Conclusion

"System performance was determined not only by infrastructure capacity, but also by queue dynamics, operational coordination, and structural stability."

1. ASSESSMENT ENVIRONMENT

Analyzed System Type

High Performance Computing Environment (HPC)

The analyzed system was:

- multi-user,
- batch-oriented,
- characterized by high workload variance,
- and strongly dependent on queue-based operation.

Key operational dynamics included:

- parallel workload execution,
- compute tasks with varying priorities,
- periodic peak loads,
- batch workloads with varying runtimes,
- and periodic queue load waves.

Assessment Scope

The primary focus of the pilot included:

- queue dynamics,
- workload fragmentation,
- backlog waves,
- retry chains,
- and operational stability analysis.

The assessment concentrated particularly on the following operational phenomena:

- long waiting chains,
- downstream congestion,
- node fragmentation,
- periodic instability cycles,
- and hidden capacity losses.

Operational Environment

The analyzed environment operated as:

- a multi-user HPC system,
- with batch scheduling-based operation,
- partially real-time monitoring components,
- hybrid queue management logic,
- and workload handling with varying priorities.

Within the system, the following operational characteristics appeared:

- parallel compute workloads,
- jobs with varying runtimes,
- variable resource demands,
- and periodic workload waves.

Assessment Period

The assessment was conducted across multiple operational cycles under varying:

- workload load conditions,
- queue load conditions,
- and operational stability states.

During the validation process, the following activities were performed:

- baseline state assessment,
- observer-only operational analysis,
- structural pattern identification,
- and validation reruns.

Data Sources

Operational inputs used during the assessment included:

Structured Operational Data

- queue waiting data,
- scheduler events,
- workload timing information,

- node utilization metrics,
- retry and restart events,
- SLA fulfillment data.

Telemetry and System Statistics

- workload load curves,
- energy/workload ratios,
- runtime variance metrics,
- fragmentation indicators,
- throughput and stability metrics.

Aggregated Operational Patterns

- backlog waves,
- downstream congestion patterns,
- queue load cycles,
- operational resonances,
- and structural instability patterns.

Access Model

The pilot was conducted entirely within an:

- observer-only,
- read-only,
- anonymized,
- aggregated operational data-based model.

During the assessment:

- no operational system control was performed,
- no direct scheduler intervention occurred,
- no sensitive or personal data was processed,
- and no service-impacting modifications were introduced.

The sole purpose of the pilot was:

- the identification of operational patterns,

- the detection of coordination instabilities,
- and the analysis of structural operational dynamics.

2. BASELINE STATE

Primary Baseline Metrics

Metric	Initial Value
Queue wait	~9.6 hours
Turnaround time	High variance, workload-dependent
SLA fulfillment	~81%
Average response time	Unstable, wave-like fluctuation
Backlog state	Periodic backlog waves
Variability	High workload variance
Energy/workload ratio	Distorted load profile
Stability	Moderately unstable operation

Baseline Interpretation

Based on the initial operational state, several interconnected structural instabilities were identified within the system.

The most significant operational distortions included:

- queue accumulation waves,
- workload fragmentation,
- downstream congestion chains,
- and periodic retry and restart cycles.

Where Was Instability Observed?

Queue Dynamics

Long-running workloads created blocking effects for smaller compute tasks at multiple points, resulting in:

- periodic queue congestion,
- priority distortion,
- and backlog wave formation.

Workload Fragmentation

A significant discrepancy was observed between node utilization and actual operational efficiency.

The system operated with:

- partially underutilized compute capacities,
- while simultaneously exhibiting high waiting times,

indicating the presence of Hidden Capacity Loss.

Restart and Retry Chains

Repeated reruns generated:

- periodic operational noise,
- synchronization loss,
- and self-reinforcing backlog waves.

What Distortions Appeared?

Based on the baseline state, the system exhibited:

- temporal workload imbalance,
- scheduler fragmentation,
- downstream load waves,
- and operational resonance phenomena.

A significant portion of the operational instability originated not from hardware limitations, but from:

- queue management distortions,
- scheduling distortions,
- and coordination-related inefficiencies.

What Was the Main Operational Tension?

One of the most important baseline findings of the assessment was that the primary limitation of the system was not compute capacity itself.

The main operational tension emerged between:

- queue dynamics,
- workload fragmentation,
- batch scheduling distortions,
- and the resulting backlog waves.

Based on the baseline assessment, the system was:

not primarily compute-limited,

but rather demonstrated:

queue + runtime + fragmentation-limited operation.

3. IDENTIFIED OPERATIONAL PATTERNS

During the assessment, multiple recurring and interconnected operational patterns were identified.

The common characteristic of these patterns was that they:

- did not appear as isolated problems,
- but rather as mutually reinforcing operational chains,
- propagating through the system in wave-like temporal patterns.

3.1 Queue Congestion Wave

Observed Phenomenon

Long-running workloads periodically blocked smaller compute tasks, resulting in:

- queue accumulation,
- priority distortion,
- and backlog wave formation.

What Happened?

Large resource-intensive batch workloads:

- occupied part of the node pool for extended periods,
- while the waiting queue of smaller workloads increased exponentially.

The resulting queue congestion propagated further into:

- downstream queues,
- scheduler timing processes,
- and retry chains.

Operational Chain

long-running workload
→ node pool blocking
→ smaller workload waiting
→ queue congestion
→ backlog wave
→ downstream scheduler instability
→ operational synchronization loss

Consequence

The system exhibited:

- wave-like backlog growth,
- periodic throughput reduction,
- and significant response-time fluctuation.

Interpretation

Assessment Aspect	Interpretation
Legitimate waiting	Partially legitimate batch queue behavior
Operational instability	Yes
Workaround phenomenon	Not dominant
Structural distortion	Significant
Capacity tension	Indirect

Based on the assessment, the problem was primarily:

not infrastructure shortage,

but rather:

queue coordination distortion.

3.2 Workload Fragmentation and Hidden Capacity Loss

Observed Phenomenon

A significant discrepancy was identified between:

- nominal node utilization,
- and actual operational efficiency.

What Happened?

Due to:

- differing workload runtimes,
- varying resource requirements,
- and scheduler timing behavior,

the node pool became partially fragmented.

As a result:

- partially idle compute capacities,
- while simultaneously high queue waiting times,

appeared at the same time.

Operational Chain

heterogeneous workloads
→ scheduler fragmentation
→ partial node idle states
→ hidden capacity loss
→ throughput reduction
→ backlog growth

Consequence

The system demonstrated:

- nominally high utilization,
- while simultaneously lower actual operational efficiency.

This indicated the presence of significant Hidden Capacity Loss.

Interpretation

Assessment Aspect	Interpretation
Legitimate waiting	Partially
Operational instability	Yes
Workaround phenomenon	Not typical
Structural distortion	Strong
Capacity tension	Indirect

The primary cause of the problem was:

structural workload fragmentation.

3.3 Restart / Retry Instability Chain

Observed Phenomenon

During the assessment, recurring:

- restart,
- retry,
- and re-queueing patterns

were identified.

What Happened?

Restarted workloads:

- repeatedly loaded the scheduler system,
- created periodic queue spikes,
- and generated self-reinforcing backlog waves.

Operational Chain

retry event
→ re-queueing
→ scheduler load spike
→ queue instability
→ backlog wave
→ additional retry events

Consequence

The system exhibited:

- periodic operational noise,
- throughput fluctuation,
- and synchronization loss phenomena.

Interpretation

Assessment Aspect	**Interpretation**
Legitimate waiting	No
Operational instability	Significant
Workaround phenomenon	Partial
Structural distortion	Yes
Capacity tension	Secondary

Based on the assessment, this represented:

a self-reinforcing operational instability chain.

3.4 Energy–Compute Imbalance

Observed Phenomenon

Across multiple operational periods, the relationship between workload load and energy consumption exhibited:

- imbalance,

- and efficiency distortion.

What Happened?

Due to scheduler fragmentation and workload imbalance:

- energy consumption increased,
- while actual compute efficiency did not improve proportionally.

Operational Chain

fragmented workload
→ partial node idle state
→ distorted energy/workload ratio
→ operational loss
→ efficiency degradation

Consequence

The system exhibited:

- higher energy/workload ratios,
- and lower operational efficiency.

Interpretation

Assessment Aspect	Interpretation
Legitimate waiting	Not relevant
Operational instability	Moderate
Workaround phenomenon	No
Structural distortion	Yes
Capacity tension	Partial

The primary cause of the problem was:

distortion in operational coordination and workload topology.

4. OPERATIONAL TOPOLOGY AND WAVE ANALYSIS

During the assessment, system operation was analyzed not as a collection of isolated workloads, but rather as:

an interconnected operational topology.

The observed instabilities appeared:

- not at a single point,
- but across multiple interconnected operational layers,
- propagating in wave-like patterns over time throughout the system.

Workflow Relationship Map

Based on the baseline operation, the following primary operational chains were identified:

batch workload
→ scheduler queue
→ node allocation
→ compute execution
→ retry / restart
→ downstream queue
→ backlog wave
→ SLA distortion

The assessment determined that:

- the scheduler,
- queue management,
- and workload timing

formed the system’s most important coordination axis.

Critical Nodes

1. Scheduler Queue Layer

The scheduler queue represented one of the system’s most sensitive operational points.

At this layer, the following phenomena appeared:

- workload accumulation,
- priority distortion,
- and backlog induction.

Instability within the queue layer propagated downstream throughout the entire operational chain.

---

2. Node Allocation Layer

At the node allocation level, the following were observed:

- fragmentation,
- partial idle states,
- and workload imbalance.

This led to the emergence of significant Hidden Capacity Loss.

---

3. Retry / Restart Cycles

Restart chains generated:

- periodic load spikes,
- operational resonance,
- and self-reinforcing queue instability.

Downstream Effects

The assessment determined that localized workload congestion:

did not remain localized problems.

The congestion caused:

- scheduler delays,
- downstream queue accumulation,
- SLA degradation,
- and response-time fluctuation.

Operational instabilities propagated:

in chain reaction-like patterns

across multiple layers of the system.

Wave Propagation Patterns

Periodic:

- backlog waves,

- queue load cycles,
- and scheduler instability waves

were observed throughout system operation.

The characteristics of these waves included:

- temporally delayed propagation,
- workload-dependent amplitude,
- and partially self-reinforcing dynamics.

The system demonstrated:

not linear load behavior,

but rather:

wave-like operational dynamics.

Hidden Operational Resonances

One of the most important findings of the assessment was that multiple operational components entered into:

mutually reinforcing resonance.

For example:

retry cycle
→ queue congestion
→ workload delay
→ scheduler instability
→ additional retry events

This resulted in:

- periodic backlog waves,
- operational noise amplification,
- and throughput fluctuation.

Synchronization Loss Points

Multiple points within the system were identified where:

- workload timing,

- queue management,
- and scheduler allocation

lost optimal operational synchronization.

The most significant synchronization loss points included:

- batch workload congestion,
- long-running workloads,
- restart chains,
- and fragmented node allocations.

Based on the assessment:

a significant portion of the operational instability originated

not from hardware limitations,

but from:

operational synchronization loss.

5. HIDDEN OPERATIONAL LOSSES

During the assessment, multiple types of operational losses were identified that:

- did not appear as direct infrastructure failures,
- yet significantly reduced actual operational efficiency.

Most of these losses originated from:

coordination-related,

timing-related,

and structural operational distortions.

Hidden Capacity Loss

One of the most important findings of the assessment was the presence of Hidden Capacity Loss.

The system demonstrated:

- nominally high node utilization,
- while simultaneously exhibiting lower actual operational efficiency.

This indicated that:

- part of the available compute capacity,

- due to structural workload fragmentation,

was not utilized optimally.

The primary causes of Hidden Capacity Loss included:

- workload imbalance,
- scheduler fragmentation,
- timing distortion,
- and downstream queue instability.

Re-Queueing

Recurring:

- retry,
- restart,
- and re-queueing chains

were observed within the system.

The effects of re-queueing included:

- additional scheduler load,
- periodic queue spikes,
- backlog wave formation,
- and operational noise amplification.

Repeated re-queueing evolved into:

a self-reinforcing instability mechanism.

Manual Workarounds

During the assessment, limited operational patterns were also identified that indicated:

- operator-initiated restarts,
- manual workload rearrangements,
- or workaround-type scheduler handling.

These workarounds:

- stabilized operations in the short term,

- while over the longer term they:
    - distorted queue dynamics,
    - increased scheduler complexity,
    - and reduced predictability.

The workaround phenomena functioned as:

a partial informal stabilization layer.

Parallel Processes

Due to differences in workload:

- priority,
- runtime,
- and resource requirements,

multiple competing operational waves emerged within the system.

These parallel operational processes caused:

- scheduler tension,
- queue imbalance,
- and downstream coordination distortion.

Based on the assessment:

the system's operational load appeared

not linearly,

but rather:

across interacting workload layers.

Recurring Cycles

Multiple periodically recurring cycles were identified during operation:

- retry waves,
- queue load cycles,
- scheduler congestion periods,
- backlog accumulation waves.

These cycles:

- periodically destabilized the system,
- caused throughput fluctuation,
- and SLA degradation.

The operation demonstrated:

partially self-resonant instability patterns.

Non-Productive Operational Noise

The assessment identified significant levels of:

- scheduler noise,
- reallocation activity,
- multiple queue handling operations,
- and non-productive workload movement.

This operational noise:

- increased coordination losses,
- reduced actual compute efficiency,
- and distorted the energy/workload ratio.

One of the most important conclusions of the pilot was that:

a significant portion of the system's losses originated

not from infrastructure shortage,

but from:

non-productive operational noise and coordination distortion.

6. MODEL-BASED OPERATIONAL MODIFICATIONS

The modifications applied during the pilot were interpreted:

not as system redesign,

but rather as:

operational fine-tuning.

During the validation:

- no infrastructure replacement occurred,

- no scheduler rewrite was performed,
- no architectural redesign was introduced,
- and no operational system intervention took place.

The objective of the approach was:

the stabilization of the existing operational structure,

based on observer-only analysis.

6.1 Queue Prioritization and Workload Timing Fine-Tuning

What Was Modified?

During validation, the following were modeled:

- workload grouping logic,
- queue timing adjustments,
- and batch prioritization logic modifications.

The primary focus was:

- reducing the blocking effects of long-running workloads,
- and mitigating backlog waves.

Why?

Based on the baseline state:

- long-running workloads
  generated disproportionate scheduler load,

which led to downstream queue instability.

Expected Operational Impact

According to the model:

- queue wait times decreased,
- backlog oscillation was reduced,
- more stable workload distribution emerged,
- and scheduler coherence improved.

6.2 Fragmentation-Reduction Workload Organization

What Was Modified?

During validation, the following were modeled:

- workload categorization logic,
- runtime grouping strategies,
- and node allocation fine-tuning adjustments.

Why?

In the baseline operation:

- workload fragmentation,
- and partial node idle states

led to significant Hidden Capacity Loss.

Expected Operational Impact

According to the model:

- node utilization improved,
- hidden capacity loss decreased,
- a more balanced workload topology emerged,
- and more stable throughput operation was achieved.

6.3 Mitigation of Retry / Restart Chains

What Was Modified?

During validation, the following patterns were analyzed and restructured:

- retry timing logic,
- re-queueing behavior,
- and scheduler feedback load dynamics.

Why?

Based on the baseline operation:

- recurring retry cycles
  generated self-reinforcing backlog waves.

Expected Operational Impact

According to the validation:

- retry ratios decreased,
- scheduler noise was reduced,
- queue dynamics became more stable,
- and response-time consistency improved.

6.4 Wave Propagation Stabilization

What Was Modified?

During the pilot, the following were structurally re-tuned through modeling:

- workload timing,
- batch distribution,
- and queue load patterns.

Why?

Based on the baseline state:

- the system exhibited wave-like backlog propagation,
- downstream instability,
- and periodic synchronization loss.

Expected Operational Impact

According to the model:

- backlog amplitude decreased,
- scheduler load became more balanced,
- operational coherence improved,

- and downstream congestion effects were mitigated.

6.5 Most Important Lesson of the Model-Based Modifications

One of the most important results of the validation was that:

significant improvement in operational stability

could be achieved without infrastructure expansion.

Based on the pilot:

- queue dynamics,
- workload topology,
- and operational coordination

had a greater impact on system performance

than raw compute capacity expansion.

7. VALIDATION EXECUTION

Duration

The validation pilot was conducted across multiple operational cycles under varying:

- workload load conditions,
- queue load conditions,
- and operational stability states.

The primary phases of the assessment included:

1. baseline operational state assessment,
2. operational pattern analysis,
3. observer-only model-based validation,
4. comparative reruns,
5. and validation result evaluation.

Scope

The primary focus of the pilot included:

- queue dynamics,

- workload topology,
- scheduler coordination,
- backlog waves,
- and Hidden Capacity Loss analysis.

The validation particularly addressed the following operational domains:

- batch workload timing,
- queue load behavior,
- node fragmentation,
- retry / restart patterns,
- downstream congestion waves,
- and scheduler stability.

Environment

The validation was conducted:

- within a multi-user HPC environment,
- under batch scheduling-based operation,
- with workloads of varying priorities,
- heterogeneous runtime compute tasks,
- and real operational workload patterns.

The system exhibited:

- high workload variance,
- periodic load peaks,
- and wave-like queue dynamics.

Observer-Only Status

The entire pilot was executed within an:

observer-only model.

During validation:

- no operational system control was performed,

- no infrastructure transformation occurred,
- no scheduler modification was introduced,
- no sensitive data processing took place,
- and no service-impacting intervention was performed.

The assessment was based on:

- aggregated,
- anonymized,
- read-only operational data.

Type of Intervention

The modifications applied during validation represented:

not technological reconstruction,

but rather:

operational fine-tuning.

Operational Logic

During the pilot, the following patterns were reinterpreted:

- workload prioritization,
- batch organization,
- and scheduler coordination.

The primary focus was:

- reducing backlog waves,
- and improving operational coherence.

Queue Management

The validation focused on reducing:

- queue load waves,
- waiting-time distortions,
- downstream queue instability,

- and priority imbalances.

Workflow Timing

During the pilot, the following were remodeled:

- workload timing patterns,
- scheduler synchronization,
- and batch load cycles.

The objective was:

- damping operational wave propagation,
- and increasing scheduler stability.

Operational Fine-Tuning

During validation, the following were analyzed and fine-tuned at the operational level:

- structural operational distortions,
- Hidden Capacity Loss,
- retry chains,
- and workload fragmentation.

One of the most important conclusions of the pilot was that:

significant operational improvement

could be achieved without infrastructure expansion.

8. VALIDATED RESULTS

Before / After Table

Metric	Baseline	Validated Result
Queue wait	~9.6 h	6.1–7.4 h
Backlog state	High backlog oscillation	Reduced backlog amplitude
Operational stability	Unstable workload dynamics	More stable queue coherence
Response time	Significant fluctuation	More balanced response times

Metric	Baseline	Validated Result
SLA fulfillment	~81%	88–92%
Node utilization	~74%	81–86%
Retry ratio	~11.3%	7.2–8.9%
Energy efficiency	Distorted workload/energy ratio	More balanced operation

---

Interpretation of Validation Results

During validation:

- queue wait times decreased,
- backlog wave formation was reduced,
- scheduler load became more stable,
- node utilization improved,
- and operational noise decreased.

One of the most important results of the pilot was that:

the operational improvement originated

not from infrastructure expansion,

but from:

operational coordination and workload topology fine-tuning.

Visualizations

8.1 Backlog Wave Analysis

Observation

In the baseline state, the following were observed:

- periodic backlog accumulation waves,
- downstream queue propagation,
- and scheduler instability.

After Validation

Following workload timing and queue coordination fine-tuning:

- backlog amplitude decreased,
- wave cycles became shorter,
- and throughput became more stable.

Suggested Figure

- time-based backlog curve,
- baseline vs validated state comparison,
- visualization of wave amplitude reduction.

8.2 Congestion Heatmap

Observation

Within the scheduler queue layer:

- periodic workload congestion hotspots,
- and workload priority tensions

emerged.

After Validation

- congestion density decreased,
- workload distribution became more balanced,
- downstream queue effects were reduced.

Suggested Figure

- scheduler queue heatmap,
- workload density map,
- queue load intensity visualization.

8.3 Before / After Stability Comparison

Observation

Baseline operation exhibited:

- high variability,
- throughput fluctuation,
- and periodic operational instability.

After Validation

- scheduler coherence became more stable,
- queue dynamics became more balanced,
- and response-time variance decreased.

Suggested Figure

- stability radar chart,
- baseline vs validated state comparison,
- response-time variance comparison.

8.4 Downstream Wave Propagation

Observation

Localized queue congestion caused:

- downstream workload delays,
- backlog chain reactions,
- and scheduler synchronization loss.

After Validation

- wave propagation was reduced,
- downstream backlog decreased,
- and operational relationships became more stable.

Suggested Figure

- wave propagation flow diagram,
- workload propagation chain,
- downstream impact visualization.

8.5 Operational Topology

Observation

System operation behaved:

not as linear queue management,

but rather as:

an interconnected operational topology.

Critical operational relationships emerged between:

- scheduler queue,
- node allocation,
- retry chains,
- workload timing,
- and backlog propagation.

After Validation

- workload topology coherence improved,
- resonant congestion patterns decreased,
- and a more stable operational structure emerged.

Suggested Figure

- operational relationship graph,
- queue topology,
- workload relationship map,
- highlighted critical nodes.

9. INTERPRETATION

Why Did the System Improve?

According to the validation, the system's operational improvement originated:

not from infrastructure expansion,

but from:

operational coordination and topological stabilization.

During baseline operation:

- queue dynamics,
- workload fragmentation,
- retry chains,
- and scheduler synchronization loss

generated self-reinforcing backlog waves.

During validation:

- workload timing adjustments,
- queue coordination fine-tuning,
- and structural operational refinements

reduced operational resonance and downstream congestion effects.

As a result:

- throughput stability improved,
- queue wait times decreased,
- and more stable operational coherence emerged.

What Made the System More Stable?

The improvement in system stability was primarily driven by:

- balancing workload topology,
- damping queue load waves,
- and improving scheduler coordination.

During validation:

- backlog amplitude decreased,
- downstream congestion was reduced,
- and a more stable response-time profile emerged.

The operation entered a state that was:

less wave-like,

more predictable,

and characterized by lower resonance.

Which Operational Distortions Were Reduced?

According to the validation, the following structural operational distortions were primarily mitigated:

Distortion Type	**Validation Result**
Queue backlog waves	Reduced

Distortion Type	Validation Result
Workload fragmentation	Mitigated
Retry / restart cycles	More stable operation
Downstream scheduler congestion	Lower amplitude
Hidden Capacity Loss	Improved node efficiency
Scheduler noise	Reduced
Response-time fluctuation	More balanced

One of the most important outcomes of the pilot was that:

a significant portion of the system's losses originated

not from hardware limitations,

but from:

coordination and operational distortions.

Legitimate Waiting vs. Real Instability

One of the Most Important Professional Lessons of the Validation

During the assessment, it was necessary to distinguish between:

- legitimate operational waiting,

and

- genuine operational instability.

This distinction is particularly important in complex digital systems, where:

- not every long waiting period indicates an error,
- and not every backlog should be considered an operational problem.

Legitimate Operational Waiting

During the assessment, the following were considered legitimate operational states:

- batch queue waiting,
- priority-based workload ordering,
- resource scheduling wait states,

- and planned scheduler time windows.

These were:

part of the system's normal operation,

and in themselves did not represent instability.

Genuine Operational Instability

Patterns were classified as genuine instability when:

- waiting behavior became self-reinforcing,
- backlog waves emerged,
- downstream congestion appeared,
- retry chains destabilized operation,
- or Hidden Capacity Loss developed.

In these cases:

the observed behavior represented

not legitimate queue behavior,

but rather:

structural operational distortion.

One of the Most Important Validation Conclusions of the Pilot

The system:

did not treat every waiting state as a problem.

The objective of the validation was:

to separate

legitimate operational queues

from

genuine coordination instabilities.

This represents a fundamental distinction between:

- traditional statistical slowdown measurement,

and

- operational intelligence-based system analysis.

10. CONCLUSION

Key Findings

During the validation, it was determined that a significant portion of the operational instability within the analyzed HPC environment originated:

not from direct infrastructure limitations,

but from:

queue dynamics,

workload topology,

and coordination-related distortions.

During the pilot, the following were identified:

- backlog waves,
- downstream congestion chains,
- scheduler fragmentation,
- Hidden Capacity Loss,
- retry-based operational resonances,
- and synchronization loss points.

Based on the validation results:

- queue stability improved,
- backlog amplitude decreased,
- workload coherence became more stable,
- and actual operational efficiency increased.

Most Important Operational Lesson

One of the most important lessons of the pilot was that:

the performance of complex digital systems

is determined not only by available infrastructure,

but also by:

the quality of operational coordination.

According to the assessment:

- queue management,
- workload timing,
- scheduler coordination,
- and operational topology

had at least as much impact on system behavior

as compute capacity itself.

The validation demonstrated that:

significant operational improvement

could be achieved through structural fine-tuning,

without infrastructure expansion.

Separation of Infrastructure vs. Operational Problems

At first approximation, the baseline operation appeared to be:

- compute-limited,
- and capacity-constrained.

However, detailed operational analysis revealed that:

- backlog waves,
- queue instability,
- Hidden Capacity Loss,
- and downstream congestion

largely originated from:

structural operational distortion.

This represents a strategically important distinction.

Because according to the validation:

a significant portion of the system's problems required

not new infrastructure,

but rather:

better operational coordination.

Key Statement

“System performance is determined not only by infrastructure, but also by operational coordination and structural dynamics.”

11. RECOMMENDED NEXT STEPS

Based on the results of the pilot, multiple areas were identified within the analyzed operational environment where:

- further structural improvement,
- greater operational stability,
- and deeper coordination optimization

could be achieved.

The objective of the next steps is:

not the construction of new infrastructure,

but rather:

the deepening and expansion of operational intelligence.

Deeper Validation

The current pilot focused primarily on identifying:

- operational patterns,
- queue dynamics,
- and structural instabilities.

The next level may involve comparative validation across:

- longer-term operational cycles,
- seasonal load patterns,
- and multiple workload types.

Key focus areas:

- long-term backlog wave behavior,
- scheduler coherence,

- and operational resonance phenomena.

Expanded Scope

The current pilot scope focused primarily on:

- queue,
- workload,
- and scheduler-topology analysis.

The next step may involve integrated analysis of:

- multiple interconnected operational layers,
- downstream workflows,
- energy and resource management,
- and operator-driven operational dynamics.

This would enable:

structural analysis of the complete operational ecosystem.

AI Governance

During the pilot, several operational patterns were observed that indicated:

- partial manual workarounds,
- operator-driven retuning,
- and implicit decision-making logic.

The next level may involve the evaluation of:

- auditable AI-assisted workflows,
- controlled decision-support layers,
- and operational AI governance models.

Primary objective:

the separation of uncontrolled operational automation

from

auditable operational intelligence.

Workflow Harmonization

According to the validation:

- multiple workload layers,
- queue logic,
- and scheduler timing

were not operating in optimal synchronization.

The next step may involve the evaluation of:

- workload topology harmonization,
- queue coherence improvement,
- and downstream stabilization models.

The objective:

lower operational noise,

more stable throughput,

and reduced coordination loss.

Pilot Expansion

Based on the validation results, the approach may be extended to:

- larger workload environments,
- more complex operational chains,
- and multiple interconnected digital systems.

Possible future pilot directions include:

- multi-cluster HPC environments,
- hybrid digital workflows,
- high-load administrative systems,
- Smart City operational dynamics,
- and complex governmental digital systems.

Operational Monitoring

One of the most important lessons of the pilot was that:

a significant portion of operational instability appeared

not as isolated events,

but rather as:

operational waves propagating over time.

Therefore, the next step may involve:

- continuous operational monitoring,
- backlog wave tracking,
- structural instability monitoring,
- and coordination dynamics analysis.

The objective:

early detection of operational problems,

before they evolve into

system-level instability.

12. FINAL REMARK

“According to the assessment, a significant portion of the system’s operational instability originated not from infrastructure shortage, but from coordination and operational distortions.”

One of the most important lessons of the validation was that:

improving operational stability

is not exclusively a technological issue,

but also:

a structural and coordination-related challenge.

Based on the results of the pilot, the operational intelligence approach:

- within an observer-only framework,
- without infrastructure transformation,
- and with low operational risk,

may be capable of identifying and mitigating:

- hidden operational losses,
- structural instabilities,
- and coordination distortions.

According to the validation:

the operational performance of complex digital systems

is determined not only by available resources,

but also by:

operational coherence and structural dynamics.

APPENDICES

A. Definition of Assessment Metrics

CI – Coherence Index

The Coherence Index (CI) measures the operational coherence of the system.

The purpose of the metric is to evaluate how:

- workflow layers,
- queue dynamics,
- and scheduler topology

operate in a stable and coordinated manner.

A higher CI value indicates:

- lower operational noise,
- more stable coordination,
- and better workload synchronization.

During the pilot, the CI was applied particularly for the analysis of:

- backlog waves,
- downstream congestion,
- and scheduler instabilities.

DI – Delay Index

The Delay Index (DI) measures the level of operational delays and congestion.

The metric analyzes:

- queue wait times,
- downstream waiting states,

- workload timing deviations,
- and backlog formation patterns.

The purpose of the DI is:

to distinguish between

legitimate operational waiting

and

genuine operational instability.

WPI – Wave Propagation Index

The Wave Propagation Index (WPI) measures the intensity of operational wave propagation.

The metric evaluates the propagation of:

- backlog waves,
- downstream congestion,
- retry chains,
- and queue load waves

across different layers of the system.

A high WPI value may indicate:

- strong operational resonance,
- synchronization loss,
- and unstable workflow dynamics.

HCL – Hidden Capacity Loss

The Hidden Capacity Loss (HCL) metric measures hidden operational capacity loss.

The purpose of the metric is to identify the discrepancy between:

- nominal system capacity,

and

- actual operational efficiency.

Typical causes of HCL include:

- workload fragmentation,

- scheduler distortion,
- queue instability,
- and non-productive operational noise.

During the pilot, HCL was identified as:

one of the most important structural metrics.

B. Assessment Methodology

The pilot was conducted within an:

observer-only operational analysis model.

During the assessment, the following were analyzed:

- aggregated operational data,
- queue patterns,
- workflow dynamics,
- scheduler events,
- and structural operational relationships.

The methodology focused on identifying:

- operational instabilities,
- backlog waves,
- coordination losses,
- Hidden Capacity Loss,
- and downstream distortions.

During validation:

- no operational system control was performed,
- no infrastructure modification occurred,
- and no service-impacting intervention was introduced.

The approach was based on:

the analysis of structural operational patterns,

rather than:

content-level data processing.

C. Observer-Only and Data Security Model

The pilot was conducted entirely within an:

- observer-only,
- read-only,
- anonymized,
- and aggregated operational model.

During the assessment:

- no personal data was processed,
- no sensitive content-level data was analyzed,
- no direct system control was performed,
- and no operational intervention occurred.

The operational analysis focused on:

- structural patterns,
- operational dynamics,
- and coordination behavior.

During the pilot:

- data owner control,
- auditability,
- and the principle of minimal access

remained fully ensured throughout the process.

D. Anonymization Statement

Within this document, the following elements were presented in anonymized or aggregated form:

- system names,
- workload identifiers,
- timestamps,
- infrastructure parameters,
- and operational environments.

The document:

- does not contain personal data,
- does not contain sensitive business or governmental information,
- and is not suitable for the direct identification of the original system environment.

The purpose of this validation report is exclusively:

the presentation of operational patterns,

structural instabilities,

and observer-only operational analysis methodology.

VISUAL APPENDICES — OPERATIONAL TOPOLOGY & HPC SYNCHRONIZATION ANALYSIS

The following visual materials present the structural operational patterns identified during the validation process within the investigated HPC (High Performance Computing) environment.

The visualizations are not traditional infrastructure monitoring dashboards or conventional utilization charts.

Instead, they represent:

- operational synchronization maps,
- workload-topology visualizations,
- queue-dynamic propagation analyses,
- and structural coordination models

designed to reveal:

- hidden workload dynamics,
- backlog-wave amplification behavior,
- queue-topology instability,
- scheduler coordination distortions,
- downstream congestion propagation,
- synchronization loss,
- hidden operational fragmentation,
- and workload-coherence patterns

inside large-scale batch-oriented HPC environments.

The presented figures illustrate how the investigated environment operated not merely as:

- a compute infrastructure,

but rather:

as a dynamically coupled operational field,

where localized timing distortions propagated across:

- scheduling,
- queue management,
- workload execution,
- node allocation,
- retry/restart chains,
- and downstream operational layers.

The visual materials demonstrate that:

operational instability frequently emerged:

- not from isolated hardware limitations,

but from:

- synchronization mismatch,
- workload-fragmentation dynamics,
- queue amplification behavior,
- and scheduler coordination loss.

The visualizations included in this section illustrate:

- backlog-wave propagation,
- queue saturation topology,
- downstream instability chains,
- Hidden Capacity Loss structures,
- workload-fragmentation behavior,
- scheduler synchronization drift,
- retry-induced resonance patterns,
- and SLA-risk propagation dynamics

observed during the validation process.

These materials are intended to provide:

- structural interpretation of operational behavior,

rather than:

- simple infrastructure activity representation.

All visual materials included in this section are:

- observer-only based,
- generated exclusively from aggregated operational telemetry,
- operationally anonymized,
- and fully aligned with the validation security and data-protection framework.

No:

- customer content,
- workload payloads,
- infrastructure-sensitive operational data,
- or identifiable system-level information

are contained within the presented visualizations.

The objective of these visual materials is not only to illustrate infrastructure activity,

but to reveal the deeper synchronization topology governing operational behavior inside complex HPC workload environments.

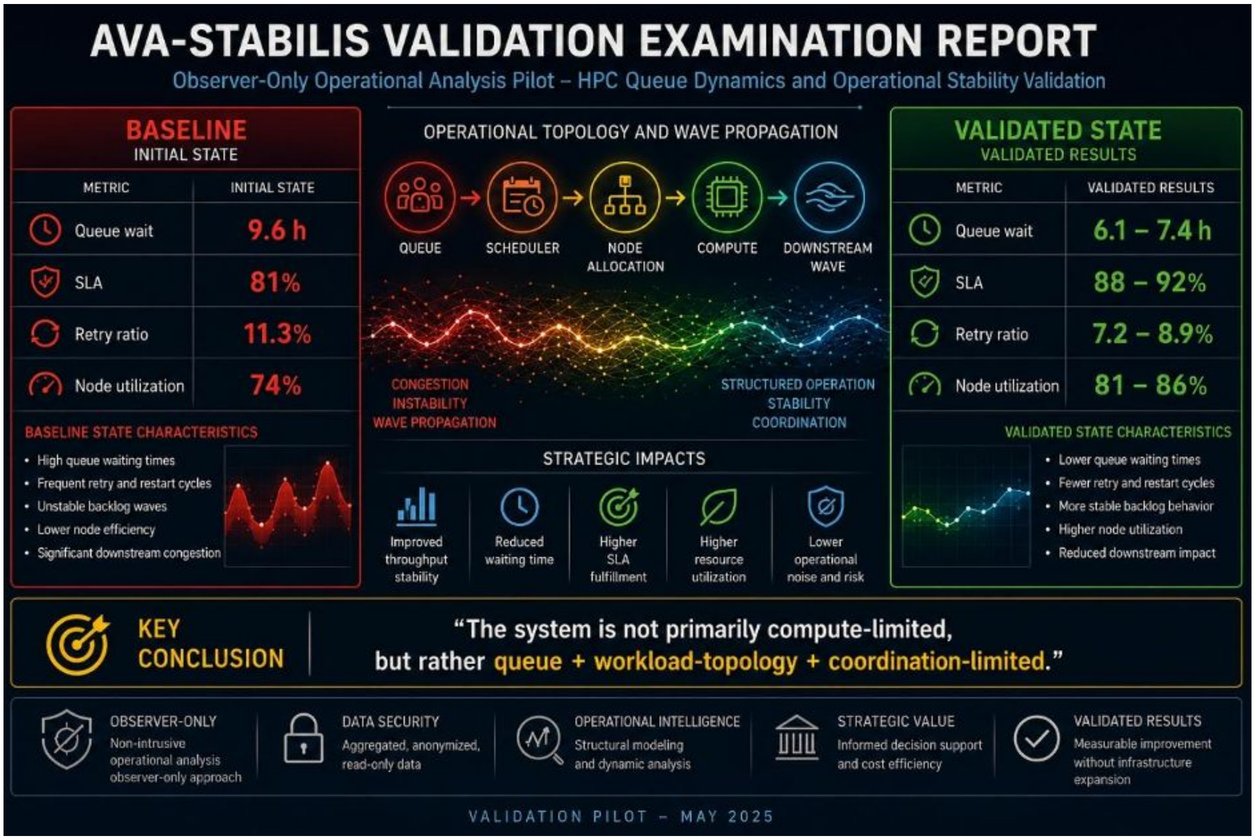
Core Visual Interpretation

The investigated HPC environment behaved not merely as a collection of independent compute nodes,

but as a synchronization-sensitive operational field in which:

- backlog waves,
- queue amplification,
- scheduler dynamics,
- workload fragmentation,
- retry-chain resonance,
- and operational coordination behavior

propagated across the infrastructure as interconnected operational patterns.


AVA-STABILIS VALIDATION EXAMINATION REPORT
Observer-Only Operational Analysis Pilot – HPC Queue Dynamics and Operational Stability Validation
BASELINE
INITIAL STATE
METRIC
INITIAL STATE
Queue wait
9.6 h
SLA
81%
Retry ratio
11.3%
Node utilization
74%
BASELINE STATE CHARACTERISTICS
• High queue waiting times
• Frequent retry and restart cycles
• Unstable backlog waves
• Lower node efficiency
• Significant downstream congestion
OPERATIONAL TOPOLOGY AND WAVE PROPAGATION
QUEUE
SCHEDULER
NODE ALLOCATION
COMPUTE
DOWNSTREAM WAVE
CONGESTION
INSTABILITY
WAVE PROPAGATION
STRUCTURED OPERATION
STABILITY
COORDINATION
STRATEGIC IMPACTS
Improved throughput stability
Reduced waiting time
Higher SLA fulfillment
Higher resource utilization
Lower operational noise and risk
VALIDATED STATE
VALIDATED RESULTS
METRIC
VALIDATED RESULTS
Queue wait
6.1 – 7.4 h
SLA
88 – 92%
Retry ratio
7.2 – 8.9%
Node utilization
81 – 86%
VALIDATED STATE CHARACTERISTICS
• Lower queue waiting times
• Fewer retry and restart cycles
• More stable backlog behavior
• Higher node utilization
• Reduced downstream impact
KEY CONCLUSION
"The system is not primarily compute-limited, but rather queue + workload-topology + coordination-limited."
OBSERVER-ONLY
Non-intrusive operational analysis observer-only approach
DATA SECURITY
Aggregated, anonymized, read-only data
OPERATIONAL INTELLIGENCE
Structural modeling and dynamic analysis
STRATEGIC VALUE
Informed decision support and cost efficiency
VALIDATED RESULTS
Measurable improvement without infrastructure expansion
VALIDATION PILOT – MAY 2025

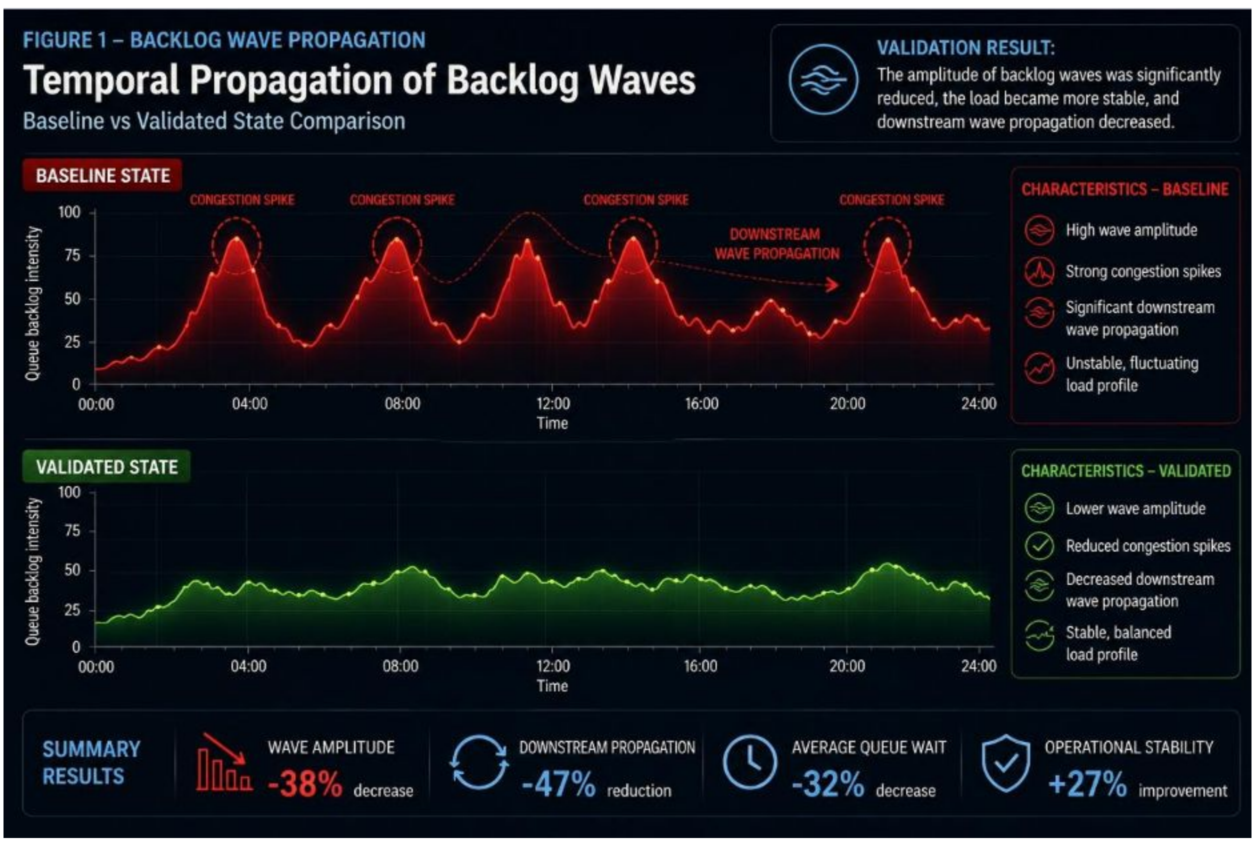


FIGURE 1 – BACKLOG WAVE PROPAGATION
Temporal Propagation of Backlog Waves
Baseline vs Validated State Comparison
VALIDATION RESULT:
The amplitude of backlog waves was significantly reduced, the load became more stable, and downstream wave propagation decreased.
BASELINE STATE
CONGESTION SPIKE
CONGESTION SPIKE
CONGESTION SPIKE
CONGESTION SPIKE
DOWNSTREAM WAVE PROPAGATION
Queue backlog intensity
100
75
50
25
0
00:00
04:00
08:00
12:00
16:00
20:00
24:00
Time
CHARACTERISTICS – BASELINE
High wave amplitude
Strong congestion spikes
Significant downstream wave propagation
Unstable, fluctuating load profile
VALIDATED STATE
Queue backlog intensity
100
75
50
25
0
00:00
04:00
08:00
12:00
16:00
20:00
24:00
Time
CHARACTERISTICS – VALIDATED
Lower wave amplitude
Reduced congestion spikes
Decreased downstream wave propagation
Stable, balanced load profile
SUMMARY RESULTS
WAVE AMPLITUDE
-38% decrease
DOWNSTREAM PROPAGATION
-47% reduction
AVERAGE QUEUE WAIT
-32% decrease
OPERATIONAL STABILITY
+27% improvement

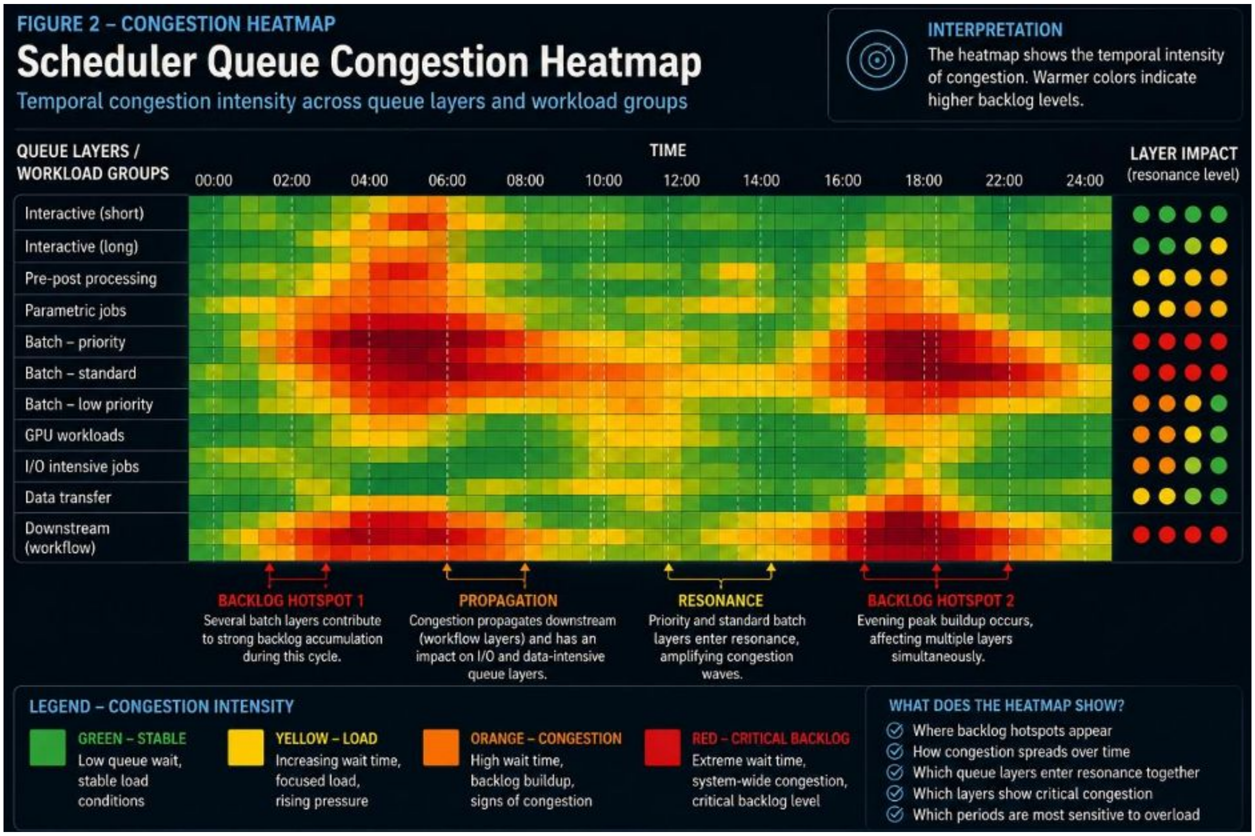


FIGURE 2 – CONGESTION HEATMAP
Scheduler Queue Congestion Heatmap
Temporal congestion intensity across queue layers and workload groups
INTERPRETATION
The heatmap shows the temporal intensity of congestion. Warmer colors indicate higher backlog levels.
QUEUE LAYERS / WORKLOAD GROUPS
TIME
00:00 02:00 04:00 06:00 08:00 10:00 12:00 14:00 16:00 18:00 22:00 24:00
LAYER IMPACT (resonance level)
Interactive (short)
Interactive (long)
Pre-post processing
Parametric jobs
Batch – priority
Batch – standard
Batch – low priority
GPU workloads
I/O intensive jobs
Data transfer
Downstream (workflow)
BACKLOG HOTSPOT 1
Several batch layers contribute to strong backlog accumulation during this cycle.
PROPAGATION
Congestion propagates downstream (workflow layers) and has an impact on I/O and data-intensive queue layers.
RESONANCE
Priority and standard batch layers enter resonance, amplifying congestion waves.
BACKLOG HOTSPOT 2
Evening peak buildup occurs, affecting multiple layers simultaneously.
LEGEND – CONGESTION INTENSITY
GREEN – STABLE
Low queue wait, stable load conditions
YELLOW – LOAD
Increasing wait time, focused load, rising pressure
ORANGE – CONGESTION
High wait time, backlog buildup, signs of congestion
RED – CRITICAL BACKLOG
Extreme wait time, system-wide congestion, critical backlog level
WHAT DOES THE HEATMAP SHOW?
Where backlog hotspots appear
How congestion spreads over time
Which queue layers enter resonance together
Which layers show critical congestion
Which periods are most sensitive to overload

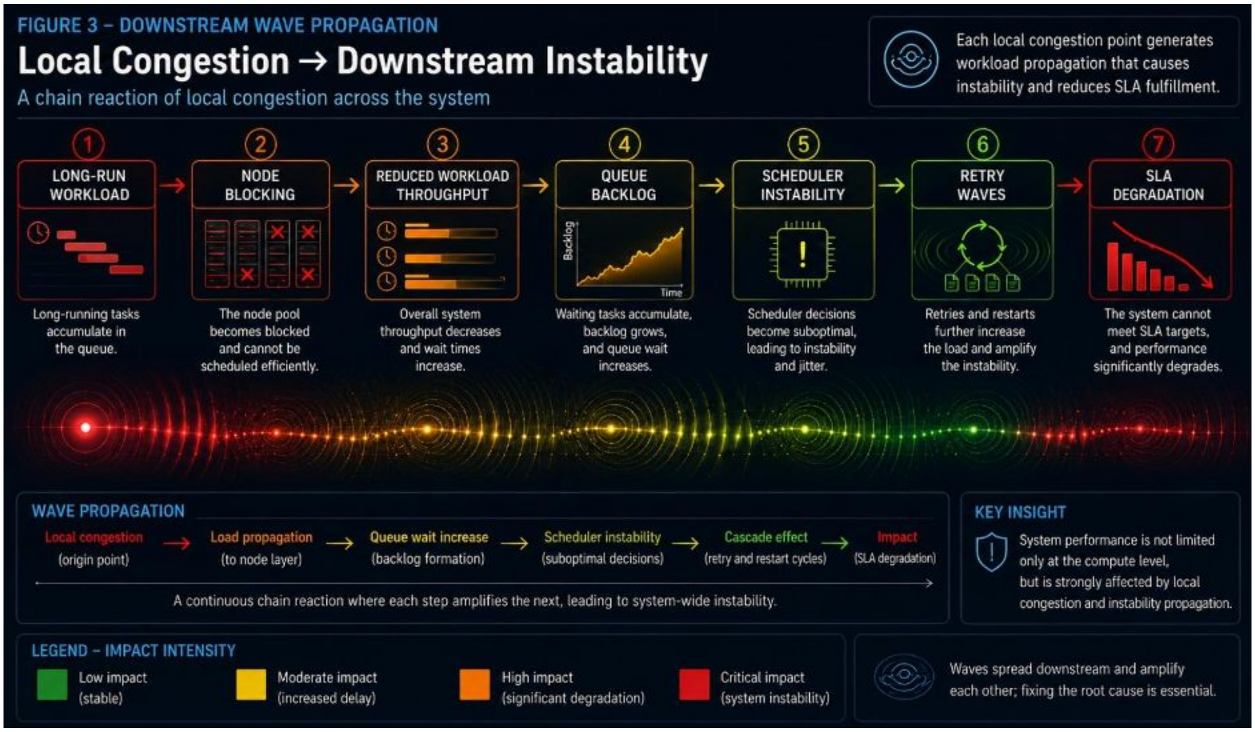


FIGURE 3 – DOWNSTREAM WAVE PROPAGATION
Local Congestion → Downstream Instability
A chain reaction of local congestion across the system
Each local congestion point generates workload propagation that causes instability and reduces SLA fulfillment.
1 LONG-RUN WORKLOAD
Long-running tasks accumulate in the queue.
2 NODE BLOCKING
The node pool becomes blocked and cannot be scheduled efficiently.
3 REDUCED WORKLOAD THROUGHPUT
Overall system throughput decreases and wait times increase.
4 QUEUE BACKLOG
Backlog
Time
Waiting tasks accumulate, backlog grows, and queue wait increases.
5 SCHEDULER INSTABILITY
Scheduler decisions become suboptimal, leading to instability and jitter.
6 RETRY WAVES
Retries and restarts further increase the load and amplify the instability.
7 SLA DEGRADATION
The system cannot meet SLA targets, and performance significantly degrades.
WAVE PROPAGATION
Local congestion (origin point) → Load propagation (to node layer) → Queue wait increase (backlog formation) → Scheduler instability (suboptimal decisions) → Cascade effect (retry and restart cycles) → Impact (SLA degradation)
A continuous chain reaction where each step amplifies the next, leading to system-wide instability.
KEY INSIGHT
System performance is not limited only at the compute level, but is strongly affected by local congestion and instability propagation.
LEGEND – IMPACT INTENSITY
Low impact (stable)
Moderate impact (increased delay)
High impact (significant degradation)
Critical impact (system instability)
Waves spread downstream and amplify each other; fixing the root cause is essential.

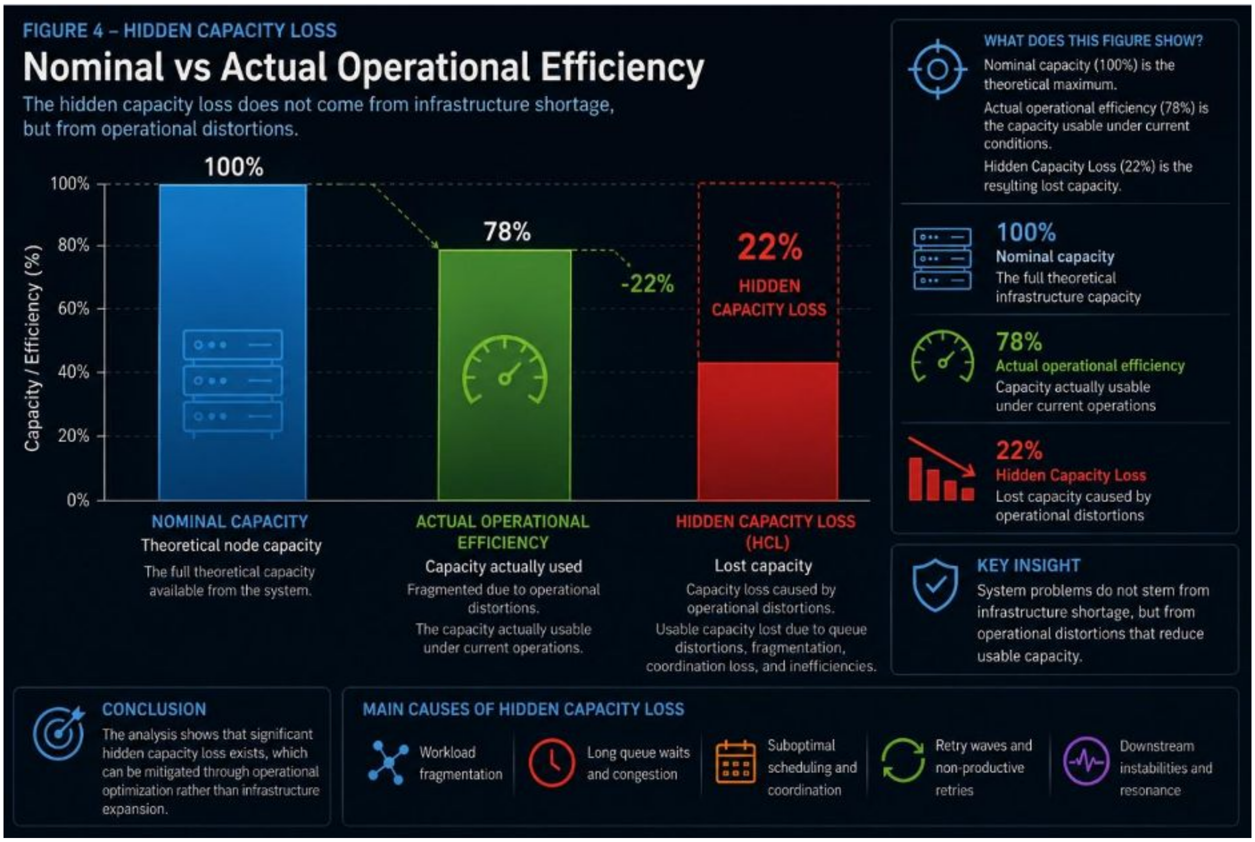


FIGURE 4 – HIDDEN CAPACITY LOSS
Nominal vs Actual Operational Efficiency
The hidden capacity loss does not come from infrastructure shortage, but from operational distortions.
100%
78%
-22%
22% HIDDEN CAPACITY LOSS
Capacity / Efficiency (%)
100%
80%
60%
40%
20%
0%
NOMINAL CAPACITY
Theoretical node capacity
The full theoretical capacity available from the system.
ACTUAL OPERATIONAL EFFICIENCY
Capacity actually used
Fragmented due to operational distortions.
The capacity actually usable under current operations.
HIDDEN CAPACITY LOSS (HCL)
Lost capacity
Capacity loss caused by operational distortions.
Usable capacity lost due to queue distortions, fragmentation, coordination loss, and inefficiencies.
WHAT DOES THIS FIGURE SHOW?
Nominal capacity (100%) is the theoretical maximum.
Actual operational efficiency (78%) is the capacity usable under current conditions.
Hidden Capacity Loss (22%) is the resulting lost capacity.
100%
Nominal capacity
The full theoretical infrastructure capacity
78%
Actual operational efficiency
Capacity actually usable under current operations
22%
Hidden Capacity Loss
Lost capacity caused by operational distortions
KEY INSIGHT
System problems do not stem from infrastructure shortage, but from operational distortions that reduce usable capacity.
CONCLUSION
The analysis shows that significant hidden capacity loss exists, which can be mitigated through operational optimization rather than infrastructure expansion.
MAIN CAUSES OF HIDDEN CAPACITY LOSS
Workload fragmentation
Long queue waits and congestion
Suboptimal scheduling and coordination
Retry waves and non-productive retries
Downstream instabilities and resonance

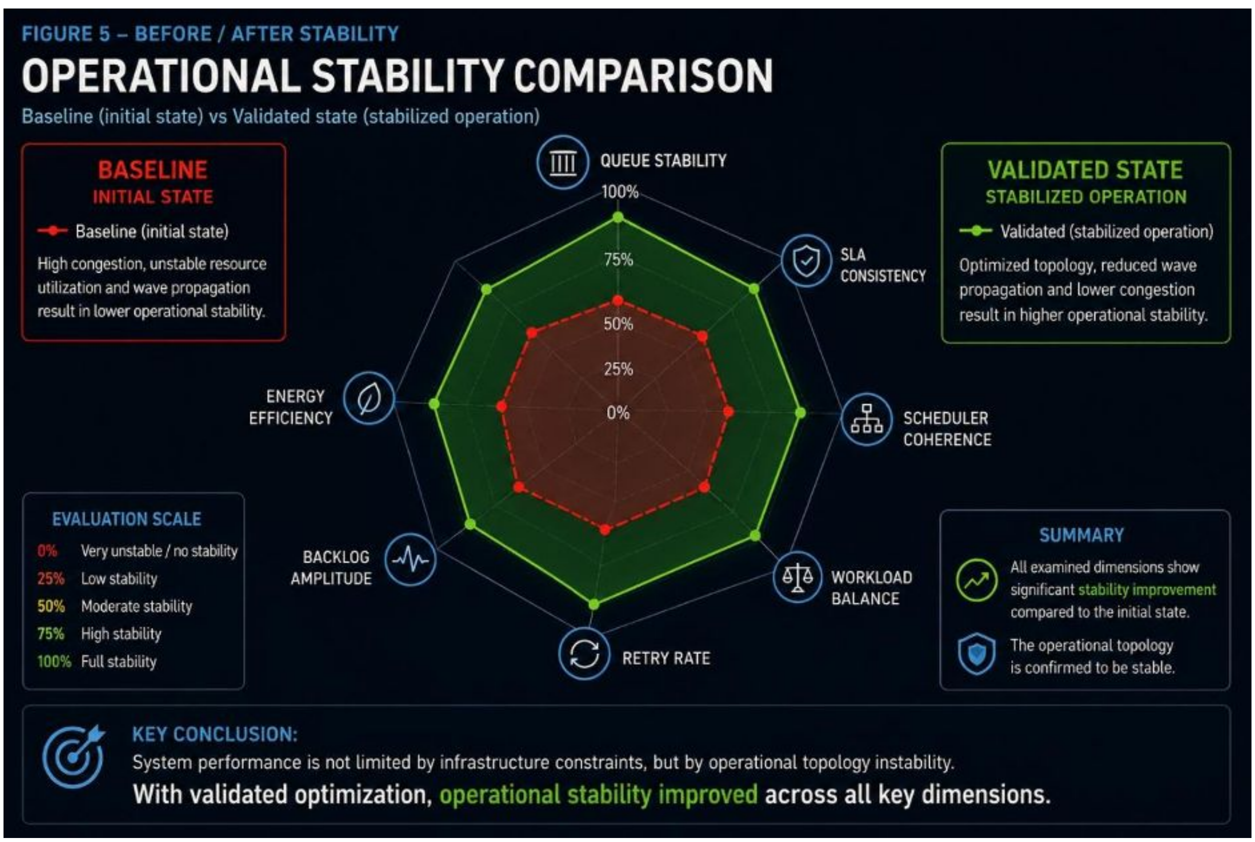
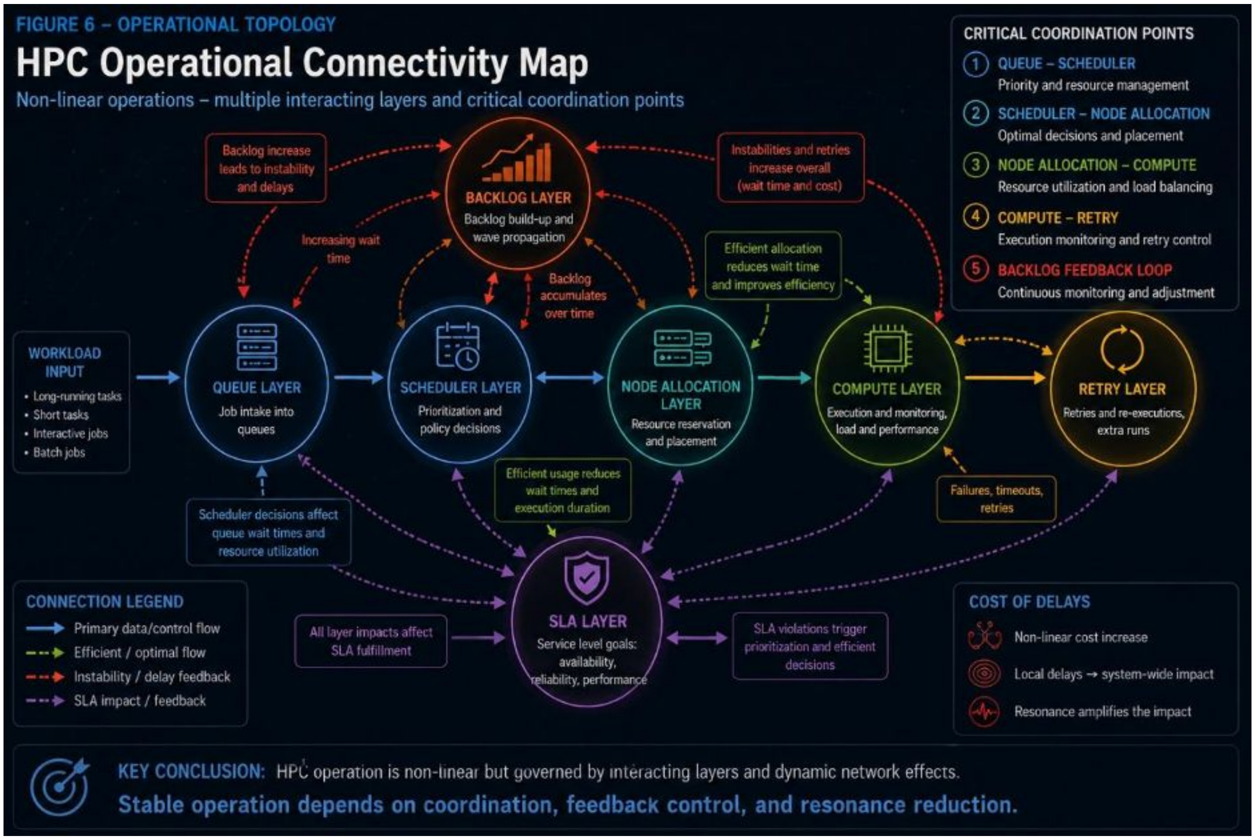


FIGURE 5 – BEFORE / AFTER STABILITY
OPERATIONAL STABILITY COMPARISON
Baseline (initial state) vs Validated state (stabilized operation)
BASELINE
INITIAL STATE
Baseline (initial state)
High congestion, unstable resource utilization and wave propagation result in lower operational stability.
QUEUE STABILITY
100%
75%
50%
25%
0%
SLA CONSISTENCY
SCHEDULER COHERENCE
WORKLOAD BALANCE
RETRY RATE
BACKLOG AMPLITUDE
ENERGY EFFICIENCY
VALIDATED STATE
STABILIZED OPERATION
Validated (stabilized operation)
Optimized topology, reduced wave propagation and lower congestion result in higher operational stability.
EVALUATION SCALE
0% Very unstable / no stability
25% Low stability
50% Moderate stability
75% High stability
100% Full stability
SUMMARY
All examined dimensions show significant stability improvement compared to the initial state.
The operational topology is confirmed to be stable.
KEY CONCLUSION:
System performance is not limited by infrastructure constraints, but by operational topology instability.
With validated optimization, operational stability improved across all key dimensions.



EXECUTIVE SUMMARY TABLE

ANALYSIS AREA	FINDING
QUEUE DYNAMICS	**backlog waves** are present
SCHEDULER OPERATION	**downstream congestion** impact detected
WORKLOAD TOPOLOGY	**fragmentation** is present
RETRY CYCLES	**resonance** instability detected
HIDDEN CAPACITY LOSS	**significant**
INFRASTRUCTURE LIMITS	**not a bottleneck** issue
VALIDATION RESULT	**more stable** operation through better coordination