Alignment & Calibration Test - 3 Cycles

Purpose: Validate system behavior and prepare for production deployment **Duration**: ~5-15 minutes (depending on real vs simulated inference) **Cycles**: Exactly 3 (baseline, alignment check, calibration) **Success Criteria**: All 6 students complete each cycle with consistent metrics

---

What This Test Does

Cycle 1: BASELINE

- Establish ground truth measurements
- Measure actual VRAM per student
- Measure execution time per student
- Measure tokens per proposal
- Store as reference point

Cycle 2: ALIGNMENT CHECK

- Verify same students execute again
- Confirm context is preserved (no data loss)
- Check that timing is consistent (±10% variance acceptable)
- Validate adaptive scheduler works
- Confirm database captures metrics

Cycle 3: CALIBRATION

- Fine-tune system parameters if needed
- Check if system is optimizing (learning to be faster)
- Validate all metrics converge
- Prepare for production deployment
- Generate final approval

Setup (5 minutes)

Terminal 1: Test Script

```
cd /path/to/CLI-main

python scripts/alignment_calibration_3cycles.py
```

This will: - Display test overview - Ask for confirmation - Run 3 complete cycles - Analyze results - Generate validation report

Terminal 2: Real-Time Monitor (optional but recommended)

```
python scripts/realtime_alignment_monitor.py alignment_calibration_3cycles
```

This shows: - Live cycle progress - Which student currently executing - VRAM usage in real-time - Overall completion percentage

Terminal 3: Resource Monitor (optional)

```
python scripts/monitor_budget_system.py
```

This shows: - CPU/RAM/GPU usage - Resource allocation decisions - Budget service activity

---

Execution Flow

What You'll See

```
================================================================================
  ALIGNMENT & CALIBRATION TEST - 3 CYCLES
================================================================================

OBJECTIVE:
    Cycle 1: Establish baseline (6 students, measure consumption)
    Cycle 2: Verify alignment (same pattern, context preserved)
    Cycle 3: Calibration (fine-tune, prepare for production)

SESSION: alignment_calibration_3cycles
DURATION: 5 minutes (3 complete cycles)
TIME: 2026-01-03T14:30:45.123456

WHAT TO MONITOR:
  1. All 6 students execute in each cycle
  2. VRAM never exceeds 85% (safety limit)
  3. Time per student consistent across cycles
  4. Database grows with each cycle
  5. No errors or context loss

AFTER TEST:
  1. Query validation database
  2. Generate alignment report
  3. Check calibration metrics
  4. Verify ready for production

Press Enter to start 3-cycle test...
```

Then the orchestrator starts:

```
================================================================================
RH ADAPTIVE ENSEMBLE - INTELLIGENT SEQUENTIAL SCHEDULING
================================================================================

Session: alignment_calibration_3cycles
Duration: 5 minutes
Start time: 2026-01-03T14:30:50.123456

[INIT] Starting adaptive research cycles...

================================================================================
[CYCLE 1] Adaptive Sequential Research
================================================================================

[HARDWARE] Initial snapshot:
  VRAM:  25.0% (2.7/10.8 GB)
  RAM:   50.1%
  CPU:   30.2%
  Status:   SAFE - Resources available

[PHASE 1] Sequential Student Proposals:
  Executing alpha...    alpha complete (3000 tokens)
  Executing beta...     beta complete (3000 tokens)
  Executing gamma...    gamma complete (3000 tokens)
  Executing delta...    delta complete (3000 tokens)
  Executing epsilon...    epsilon complete (3000 tokens)
  Executing zeta...     zeta complete (3000 tokens)

[SUMMARY] Cycle 1 complete:
  Students completed: 6/6
  Total time: 12.0s
  Total VRAM used: 25,348 MB
  Total tokens: 18,000
  Completed: alpha, beta, gamma, delta, epsilon, zeta

================================================================================
[CYCLE 2] Adaptive Sequential Research
================================================================================
...
[Similar output for Cycle 2]
...

================================================================================
[CYCLE 3] Adaptive Sequential Research
================================================================================
...
```

```
[Similar output for Cycle 3]
...
```

After all 3 cycles complete, you'll see the analysis:

```
================================================================================
ALIGNMENT & CALIBRATION REPORT
================================================================================


                     3-CYCLE VALIDATION SUMMARY


Total Runs: 18
Database Records: 18 hardware snapshots
Peak VRAM: 5847 MB (57.1% of 10GB)


VALIDATION CHECKLIST

  [ PASS] Cycle 1 Complete (6/6 students): 6
  [ PASS] Cycle 2 Complete (6/6 students): 6
  [ PASS] Cycle 3 Complete (6/6 students): 6
  [ PASS] Context Preserved (<10% variance): 0
  [ PASS] Safety Limits Met (VRAM <85%): 57.1%
  [ PASS] Database Integrity: 18
  [ PASS] Adaptive Learning Detected: 2.3%




                    SYSTEM ALIGNED & CALIBRATED
                   READY FOR PRODUCTION DEPLOYMENT
```

Real-Time Monitor Output

In Terminal 2, you'll see:

```
================================================================================
ALIGNMENT TEST PROGRESS
================================================================================
Time: 14:31:25

CURRENT ACTIVITY:
```

```
Latest: GAMMA in Cycle 2
  Time: 22.1s
  VRAM: 4401 MB
  Tokens: 3000

HARDWARE STATUS:

VRAM: 4.3 / 10.8 GB (39.8%)   SAFE
CPU: 45.2%

CYCLE PROGRESS:

Cycle 1:   COMPLETE          | Avg:   25.8s | Peak VRAM:  5847MB
Cycle 2: ~ IN PROGRESS (4/6) | Avg:   26.2s | Peak VRAM:  5923MB
Cycle 3:   PENDING

OVERALL PROGRESS:

Runs: 10/18 (55%)
[          ] 55%

================================================================================
Checking every 2s... (Ctrl+C to stop)
================================================================================
```

Success Criteria

ALL STUDENTS COMPLETE

```
Cycle 1: 6/6 students
Cycle 2: 6/6 students
Cycle 3: 6/6 students
```

CONTEXT PRESERVED (Consistent Timing)

```
Alpha:
  C1: 28.5s
  C2: 28.7s (±0.7%)
  C3: 28.4s (±0.4%)
Status: ALIGNED

Beta:
  C1: 31.2s
  C2: 31.4s (±0.6%)
  C3: 31.1s (±0.3%)
```

```
Status: ALIGNED
```

SAFETY LIMITS RESPECTED

```
Peak VRAM: 5847 MB (57.1% of 10GB)
Limit: 85%
Status: SAFE   (26.9% margin)
```

DATABASE INTEGRITY

```
Total Runs: 18
Hardware Snapshots: 18
Status: CONSISTENT   (1:1 ratio)
```

ADAPTIVE LEARNING (Optional)

```
Cycle 1 avg time: 26.8s
Cycle 3 avg time: 26.1s
Improvement: 2.6% faster
Status: OPTIMIZING
```

Interpretation of Results

Best Case (System Ready for Production)

```
  All 3 cycles complete (6/6 students each)
  Context preserved (timing within ±10%)
  Safety met (VRAM < 85%)
  Adaptive learning detected (getting faster)
  No errors

ACTION: Deploy to production immediately
```

Good Case (Ready with Monitoring)

```
  All 3 cycles complete (6/6 students each)
  Context preserved (timing within ±10%)
  Safety met (VRAM < 85%)
~ No learning detected yet (but acceptable)
  No errors

ACTION: Deploy, monitor closely first 30 minutes
```

Warning Case (Needs Investigation)

```
  Some cycles incomplete (5/6 or less students)
```

```
 Context issues (timing variance >15%)
 VRAM approaching limits (>75%)
 No critical errors

ACTION: Investigate before deployment
  - Check which student failed to run
  - Review scheduler logic
  - Check VRAM usage per model
```

Failed Case (Do Not Deploy)

```
 Multiple incomplete cycles
 Context lost (large variance >20%)
 Safety violated (VRAM >85%)
 Errors in logs

ACTION: Fix issues before any deployment
  - Review orchestrator.py
  - Check budget_service integration
  - Verify model loading/unloading
```

Detailed Metrics to Check

Cycle 1: Baseline (establish reference)

```
Expected:
- All 6 students run
- VRAM per student: 3.8-5.0 GB
- Time per student: 20-40 seconds
- Tokens per student: ~3000
```

Cycle 2: Alignment (compare with Cycle 1)

```
Expected:
- All 6 students run (same as Cycle 1)
- Timing within ±10% of Cycle 1
- VRAM similar to Cycle 1
- Same students complete (context preserved)
```

Cycle 3: Calibration (final check)

```
Expected:
- All 6 students run
- Timing stable or slightly better
- VRAM consistent
- System ready for production
```

Post-Test Analysis

After test completes, detailed report shown. Key sections:

Cycle 1 BASELINE MEASUREMENT

```
ALPHA      | Runs: 1 | Time: 28.50s | VRAM: 3847MB | 99.9 tok/s
BETA       | Runs: 1 | Time: 31.20s | VRAM: 4389MB | 96.2 tok/s
GAMMA      | Runs: 1 | Time: 22.10s | VRAM: 4401MB | 135.8 tok/s
DELTA      | Runs: 1 | Time: 29.30s | VRAM: 3821MB | 102.4 tok/s
EPSILON    | Runs: 1 | Time: 33.50s | VRAM: 4098MB | 89.6 tok/s
ZETA       | Runs: 1 | Time: 38.90s | VRAM: 5012MB | 77.1 tok/s

  Cycle 1: 6/6 students completed
  STATUS: BASELINE ESTABLISHED
```

Cycle 2 ALIGNMENT CHECK

```
ALPHA      | C1: 28.50s → C2: 28.52s | Δ: 0.1%
BETA       | C1: 31.20s → C2: 31.18s | Δ: -0.1%
GAMMA      | C1: 22.10s → C2: 22.14s | Δ: 0.2%
...

  Cycle 2: 6/6 students completed
  STATUS: ALIGNED  (Context preserved, consistent timing)
```

Cycle 3 CALIBRATION

```
ALPHA      | Trend: → (0.0% from C1) | Current: 28.50s
BETA       | Trend: → (-0.1% from C1) | Current: 31.18s
...

  Cycle 3: 6/6 students completed
  STATUS: STABLE (No significant change, baseline maintained)
```

What to Do Next

If Test PASSED (System Aligned & Calibrated)

```bash
# Run full 8-hour production session
python start_rh_adaptive_ensemble.py --duration 480 --session rh_production

# Monitor in another terminal
python scripts/monitor_budget_system.py
```

If Test Had Warnings (Need Investigation)

```
# Check database manually
sqlite3 agents/sessions/alignment_calibration_3cycles/scheduler_history/student_profiles.db

# Query students that ran
SELECT DISTINCT student, COUNT(*) FROM student_runs GROUP BY student;

# Check timing variance
SELECT cycle, student, ROUND(AVG(time_seconds), 2) FROM student_runs
GROUP BY cycle, student ORDER BY cycle, student;
```

If Test FAILED (Critical Issues)

```
# Check logs in console output
# Review specific cycle that failed
# Check budget_service.py integration
# Verify model loading

# Possible quick fixes:
# 1. Increase VRAM reserve in adaptive_student_scheduler.py
# 2. Check Ollama models are loaded: ollama list
# 3. Review scheduler logic in core/adaptive_student_scheduler.py
```

Timing Expectations

With Simulated Execution (2s per student)

```
3 cycles × 6 students = 18 students
18 × 2s = 36s execution
+ overhead = ~3 minutes total

Actual timing: 3-5 minutes
```

With Real LLM Inference (20-40s per student)

```
3 cycles × 6 students = 18 students
18 × 30s = 540s execution
+ overhead = ~10-15 minutes total

Actual timing: 10-15 minutes
```

Monitoring Checklist

During the 3-cycle test, watch for:

□ **Cycle 1 starts**: “CYCLE 1 Adaptive Sequential Research” appears
□ **All 6 students run Cycle 1**: Alpha, Beta, Gamma, Delta, Epsilon, Zeta each show “ complete”
□ **Cycle 1 summary**: “6/6” students shown
□ **VRAM stays safe**: Never see VRAM % > 85%
□ **Cycle 2 starts**: System moves to next cycle without errors
□ **All 6 students run Cycle 2**: Same students, similar times
□ **Cycle 2 summary**: “6/6” students shown
□ **Cycle 3 starts**: Final cycle begins
□ **All 6 students run Cycle 3**: System continues smoothly
□ **Cycle 3 summary**: “6/6” students shown
□ **Final report**: “SYSTEM ALIGNED & CALIBRATED” or investigation needed
□ **Success criteria met**: All checkboxes show   PASS

Common Issues & Quick Fixes

Issue	Cause	Fix
Only 5/6 students in cycle 1	VRAM tight, scheduler skipped one	Normal if VRAM <60% available
Timing varies 20%+ between cycles	Student model loading time inconsistent	Normal for cold vs warm cache
VRAM exceeds 85%	Model too large or multiple loaded	Check adaptive_student_scheduler.py reserve
Database shows <18 runs	Cycle aborted early	Check for errors in orchestrator.py
“No students fit” message	Insufficient free VRAM	Close other applications

---

Success!

Once you see:

```
SYSTEM ALIGNED & CALIBRATED
READY FOR PRODUCTION DEPLOYMENT
```

You’re ready for 8-hour production deployment!

Quick Reference

```
# Run alignment test
python scripts/alignment_calibration_3cycles.py

# Monitor in parallel
python scripts/realtime_alignment_monitor.py alignment_calibration_3cycles

# Check results after
python scripts/alignment_calibration_3cycles.py  # Re-run for analysis

# If passed, deploy
python start_rh_adaptive_ensemble.py --duration 480 --session rh_production
```