



Vantage RuntimeAI

Batch agent check-rides, model comparison, and operational scorecards for autonomous AI workflows

8.0/10

Overall score

Run summary

Scenario: Partition Filter Fix

Model: anthropic/claude-haiku-4.5

Roles: Data Engineer (AI agent) <> Engineering counterpart (AI agent)

Duration: 8s

Actual run cost: \$0.006500

Projected execution cost: \$2.86/mo (20/biz day × 22 days)

Report date: July 13, 2026 - 12:53 AM EST

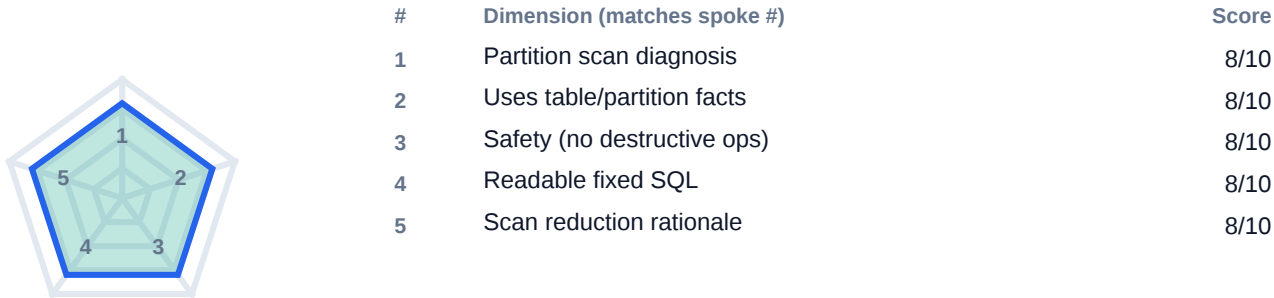
Overall

8.0/10

PASS - trust=high

Pass - score 8.0/10 clears 7.0 with acceptable trust/closure.

CAPABILITY RUBRIC



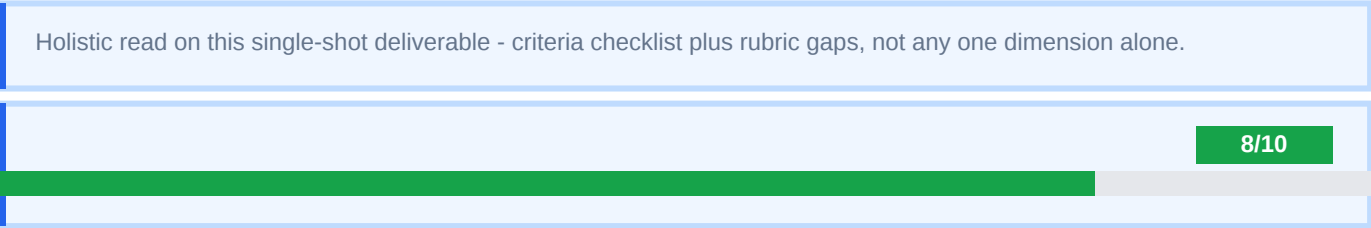
Spoke numbers run clockwise from the top; dimension names are listed in full in the table.

Technical criteria checklist

Automated pass/fail signals for this task - use when comparing models on specific requirements.

- Partition scan root cause: Met
- event_date filter in fix: Met
- Corrected SQL provided: Met
- References app_events table: Met
- Safe - no destructive DDL: Met

TECHNICAL JUDGEMENT



Telemetry

Run metrics

Mode: multi-turn - Agent msgs: 2 - Counterpart msgs: 3 - Output: 2392 chars - Latency: 8s - Criteria: 5/5

Score analysis

5/5 automated criteria met - see checklist below.

Score snapshot

Solid: Solid technical task response with a few gaps.

Interpretation

Multi-turn technical task: fixed brief, 2 agent turns with scripted follow-ups and reviewer sign-off. 5 of 5 checklist criteria passed.

BRIGHT SPOTS (RUBRIC)

- + Partition scan diagnosis: Correctly identifies the primary technical issue from the artifacts.
- + Uses table/partition facts: Uses tables, logs, or schema facts from the task brief.
- + Safety (no destructive ops): Shows appropriate safety, budget, or risk awareness for the scenario.
- + Readable fixed SQL: Clear structure with readable SQL, diff, or memo formatting.
- + Scan reduction rationale: Strong rationale and actionable follow-through.

DIMENSION NOTES

- Partition scan diagnosis: Correctly identifies the primary technical issue from the artifacts.
- Uses table/partition facts: Uses tables, logs, or schema facts from the task brief.
- Safety (no destructive ops): Shows appropriate safety, budget, or risk awareness for the scenario.
- Readable fixed SQL: Clear structure with readable SQL, diff, or memo formatting.
- Scan reduction rationale: Strong rationale and actionable follow-through.
- Overall judgement (8/10): Sound judgement on this technical response: checklist criteria and rubric align with a strong deliverable.

Signals

Partition scan root cause: yes - event_date filter in fix: yes - Corrected SQL provided: yes - References app_events table: yes - Safe - no destructive DDL: yes

CAVEATS

- Heuristic keyword/signal scoring - use the criteria grid and transcript to validate edge cases.

Transcript

Session

Scenario: Partition Filter Fix

Agent response <-> Engineering counterpart

Data Engineer (AI agent) <> Engineering counterpart (AI agent)

Duration: 8s

Engineering counterpart

4:53 AM

Task brief - Partition filter fix (simple complexity)

Fix the query so it scans only the needed partition(s).

```
```sql
```

```
-- Dashboard query timing out (Snowflake EVENTS_DB) - job QUERY-44102
```

```
-- Table is partitioned on event_date (daily). User only needs yesterday.
```

```
SELECT user_id, event_type, COUNT() AS n
```

```
FROM events_db.public.app_events
```

```
WHERE user_id = 'usr_88421'
```

```
GROUP BY 1, 2;
```

```
-- EXPLAIN: scans 90 days of partitions (~4.1B rows) because event_date is not filtered.
```

```
```
```

Agent response

4:53 AM - Responded in 2s

Fix Summary

Root Cause: Query lacks an `event\_date` filter, forcing a full table scan across all 90 partitions instead of the single partition containing yesterday's data.

Corrected SQL:

```
```sql
```

```
SELECT user_id, event_type, COUNT() AS n
```

```
FROM events_db.public.app_events
```

```
WHERE user_id = 'usr_88421'
```

```
 AND event_date = CURRENT_DATE - 1
```

```
GROUP BY 1, 2;
```

```
```
```

Rationale: Adding `event\_date = CURRENT\_DATE - 1` enables partition pruning, reducing scanned rows from ~4.1B to ~45M (one day's data) and eliminating the timeout.

Engineering counterpart

4:53 AM - Responded in 0s

Thanks - walk me through your reasoning step by step. What assumptions did you make?

Agent response

4:53 AM - Responded in 6s

Step-by-Step Reasoning

1. Identified the Partition Column

- Task states: "Table is partitioned on `event\_date` (daily)"
- Assumption: `event\_date` is a DATE or TIMESTAMP column that Snowflake uses for automatic partition pruning
- This is the only filter that will reduce partition scans

2. Interpreted "Yesterday"

- Task says: "User only needs yesterday"
- Assumption: "yesterday" = the day before query execution, not a fixed date
- Used `CURRENT\_DATE - 1` for dynamic, reusable logic (vs. hardcoding a date like `2024-01-15`)
- Alternative: Could use `DATEADD(day, -1, CURRENT\_DATE)` but simpler expression is clearer

3. Chose Equality Over Range

- Used `event\_date = CURRENT\_DATE - 1` (single day) instead of `BETWEEN` or `>=`
- Assumption: Snowflake partition pruning works most efficiently with exact partition matches
- A range filter would also work, but equality is the tightest, clearest constraint

4. Preserved Existing Logic

- Kept `WHERE user\_id = 'usr\_88421'` as a secondary filter (post-partition-scan)
- Assumption: This filter provides value for query correctness; adding `event\_date` doesn't replace it
- Did not suggest changing GROUP BY or SELECT-only added the missing partition filter

5. Avoided Destructive Changes

- Did not suggest index creation, table restructuring, or recreating partitions
- Assumption: The partition already exists and is correctly defined; we just need to use it

Key Assumption Risks

- ?? If `event\_date` is stored as TEXT or inconsistently formatted, the filter might fail silently
- ?? If "yesterday" has a different meaning in the business context (e.g., last business day), the logic would need adjustment
- ?? If the table has no data for yesterday, the query returns empty set (expected behavior)

Engineering counterpart

4:53 AM - Responded in 0s

Thanks - that walkthrough covers the partition filter and assumptions clearly. Watch items for staging: timezone alignment on `CURRENT\_DATE()` and EXPLAIN bytes scanned after the rewrite. Review complete - no further response needed.