How to Succeed in the Technical Account Manager Challenge

Liquidity Migration Analysis Edition

This challenge simulates a **real client engagement**. We're not only evaluating whether you can pull on-chain data, but whether we would trust you to analyze it, explain it, and stand behind it in front of a client.

Accuracy, structure, and honesty matter more than sophistication.

---

1. Follow the Instructions Exactly (This Is a Gate)

What we look for

- A GitHub repository with:
  - Clean, documented code
  - Data exports
  - Visualizations
  - A written analytical summary
  - A short client-facing explanation
- Clear separation between:
  - Analysis
  - Findings
  - Client narrative

Common pitfalls

- Submitting a single raw notebook.
- No README or summary.
- No client-facing explanation.

Tip

> If this were a real client deliverable, would you feel comfortable sending it as-is?

If the answer is "no," it's not ready.

---

2. Never Invent Data. Ever.

What we look for

- All numbers traceable to on-chain sources.
- Clear handling of missing or incomplete data.
- Explicit caveats where assumptions are required.

Common pitfalls

- Making up explanations when data doesn't line up.
- Using random or placeholder data (e.g. `np.random`) and presenting results as real.
- Writing confident conclusions without evidentiary backing.

Tip

> Saying "we couldn't reliably compute X because of Y" is **far better** than guessing.

Fabricated or unverifiable data is an automatic fail — in a TAM role, this breaks trust instantly.

---

3. Use the Right Tools for the Job (Hints Are There for a Reason)

What we look for

- Correct usage of:
  - UniV4 Quoter for execution simulation
  - On-chain price sources
  - Vault contract state for LP accounting
- Awareness of AMM mechanics and limitations of approximations.

Common pitfalls

- Using simplified formulas when precise tools are explicitly hinted.
- Treating approximations as ground truth.
- Ignoring trade directionality (IXS → ETH *and* ETH → IXS).

Tip

> If a hint is given, we expect you to at least evaluate and justify using it.

This role requires knowing *when precision matters*.

---

4. Execution Quality Analysis: Be Complete and Symmetric

What we look for

- Slippage computed:
  - In both directions
  - Across all requested trade sizes
  - Before and after migration
- Clear time-based comparison showing regime change (UniV2 → UniV4).
- Visualizations that clearly separate pre- and post-migration periods.

Common pitfalls

- Only analyzing one direction.
- Skipping sizes.
- Mixing pre- and post-migration data without labeling.

Tip

> If the client asks “what about the other direction?”, your work should already answer it.

---

5. Vault Performance: Explain the *Why*, Not Just the *What*

What we look for

- Accurate reconstruction of:
  - Vault token balances over time
  - Value composition (IXS vs ETH)
- Fair comparison against:
  - Hold strategy
  - Full-range LP baseline
- Clear explanation of tradeoffs (volatility exposure, rebalancing, fee capture).

Common pitfalls

- Using synthetic prices.
- Computing performance without explaining mechanics.
- Presenting alpha without explaining risk.

Tip

> Clients care less about “outperformance” and more about *why it happened and when it might not*.

---

6. Charts Should Tell the Story Without You Talking

What we look for

- Readable axes, legends, and time markers.
- Charts that align directly with claims in the text.
- Logical ordering: execution → liquidity → performance.

Common pitfalls

- Hard-to-read charts.
- No narrative flow.
- Visuals that don't support conclusions.

Tip

> If a chart doesn't support a specific claim, remove it.

---

7. Be Honest About AI Usage

What we look for

- AI used as an accelerator, not a substitute for understanding.
- Code and explanations that reflect actual execution.
- Documentation that matches reality.

Common pitfalls

- AI-written README claiming tools that weren't used.
- Confident language masking incorrect implementation.
- Failure to sanity-check AI output.

Tip

> In this role, being wrong *confidently* is worse than being cautious and correct.

---

8. Client Synthesis Is the Final Test

What we look for

- A clear answer to:
    - Was the migration beneficial?
    - Under what metrics?
    - What tradeoffs remain?

- A quantitative argument for migrating remaining funds.
- A short, plain-English client summary.

Common pitfalls

- Repeating analysis instead of synthesizing.
- No recommendation.
- Overly technical language in the client summary.

Tip

> Imagine the reader has 5 minutes and no desire to read code.

What a Strong Submission Signals

A strong candidate:

- Treats on-chain data with rigor and respect
- Never fabricates results
- Thinks like a client advocate, not a researcher
- Communicates uncertainty clearly
- Delivers something they’d be proud to put their name on