

74 questions that **win** the room.

Take these into your next Snowflake engagement and ask what the other bidders can't answer.

Every question here is answerable from a client's own operational metadata once APEX has built its Private Knowledge Graph. Where other bidders bring a top-queries list, you can say which dbt model wrote the spill and which of last year's recommendations never shipped.

74

ENGAGEMENT
QUESTIONS

17

COST DOMAINS
COVERED

29

NEED OWNERSHIP
HISTORY

3

ENGAGEMENT PHASES

WHY THE QUESTION WINS

The first meeting is a test. Most partners take the same one.

Walk into a Snowflake bake-off and you will see three decks built from the same views:

WAREHOUSE_METERING_HISTORY rolled up by month, the 25 most expensive queries, and a slide recommending smaller warehouses. Snowflake's own Workload Performance Analysis skill in Cortex Code now produces most of that natively, which is good for clients and raises the floor for every partner.

So the client has already seen detection. What they have rarely seen is a partner who can say who owns a cost, whether the last fix ever shipped, and what a change will do before anyone approves it. Those answers need ownership mapping, lineage back to the dbt model or notebook, and months of history. A consultant can assemble that by hand over six weeks. APEX keeps it in a Private Knowledge Graph (PKG) per client, so you can ask the question in week one.

WHAT A STANDARD DISCOVERY ANSWERS	WHAT YOU CAN ASK WITH APEX	
Which warehouses spent the most last month	Which team, engineer and dbt model drove the month-over-month delta, and whether the output justified it	Q53
Top 100 queries by credits	Whether each recurring spill is undersized memory or a modeling problem, and who authored it	Q1
Total Cortex AI spend	Spend by agent, human owner and ISV tenant, with model right-sizing per workflow	Q13, Q15
A list of recommendations	Which past recommendations never shipped, what blocked them, and realized vs. forecast ROI on the ones that did	Q73, Q74
"Consider resizing warehouse X"	The net credit impact of the resize, with assumptions and a confidence interval, before anyone approves it	Q58, Q62

WHAT THIS DOES TO THE ENGAGEMENT

Findings with a named owner get fixed, and the rest sit in a backlog until the next assessment finds them again. The 29 questions marked **PKG** in this playbook depend on ownership, lineage or history, and they are the ones that turn a two-week assessment into a managed service, because the client needs someone to keep asking them every quarter.

HOW APEX ANSWERS WITHOUT TOUCHING CLIENT DATA

METADATA ONLY

Zero data egress, operational metadata only: query fingerprints, credit deltas, warehouse and object IDs, anonymized ownership tokens. Never query bodies.

INSIDE THE BOUNDARY

Any model call that touches actual query content runs inside the client's account through CORTEX.COMPLETE(). Anavsan holds no credential into the client account.

HUMAN SIGN-OFF

Fixes arrive as draft pull requests routed to the owner. Nothing reaches production without a person approving it.

USING THE BANK

Pick questions by phase, then by who's in the room.

You won't ask all 74 in one engagement. Each section in the bank is tagged with the phase it fits and the person it lands with. The arc below is how we've seen it map to a typical partner SOW.

PHASE 1 / WEEKS 1-2

Win and scope

Open with the cost decomposition, read the anomaly history, and size the SOW around what you find.

Start: Q68, Q70, Q69, Q54

PHASE 2 / DAYS 15-90

Fix with owners

Work the quick-win inventory, going domain by domain with the author of each workload named.

Start: Q1, Q16, Q23, Q40

PHASE 3 / QUARTERLY

Prove and renew

Bring realized vs. forecast ROI to the QBR and forecast the next two quarters.

Start: Q73, Q74, Q59, Q71

FOUR PLAYS TO RUN THIS QUARTER

The opening slide

Q68 / Q70 / Q69

FIRST MEETING WITH A NEW ACCOUNT

Lead with Q68 as the first slide. Q70 decides the shape of the SOW: a client whose spikes were never resolved needs an enforcement process, while one that resolves spikes and keeps having them needs engineering discipline. Close with Q69 so the sponsor leaves with a 90-day list.

The Cortex owner

Q13 / Q15 / Q45 / Q50

ACCOUNTS SHIPPING AI WORKLOADS

Resource monitors don't govern AI services billing, so Cortex spend usually has no named owner. Q13 names one per agent. For ISV clients, Q15 ties that spend to contract value per tenant, and that conversation tends to reach the CFO.

The engineer champion

Q63 / Q64 / Q67

WINNING THE DATAOPS TEAM

Hand these to the engineers directly. Q63 gives each person a list they can fix in an afternoon, and Q67 checks a query against patterns already flagged in the account before it ships. Engineers who use it between status calls become the reason the client renews.

The expansion QBR

Q73 / Q74 / Q71

MONTH THREE ONWARD

Q73 prices the value left unrealized from earlier recommendations, whoever made them. Q74 shows which of your interventions beat their forecast. Q71 picks the three workloads that become your case study for the expanded scope.

THE QUESTION BANK

17 domains, 74 questions.

Every question is written the way a client would ask it, so you can read it aloud in a workshop. The wording is yours to adapt.

01	Modeling Debt and Query Design	Q1-3	10	Storage Health and Decay	Q32-38
02	Concurrency vs. Capacity	Q4-5	11	Pipeline and Orchestration Economics	Q39-44
03	Clustering Economics	Q6-7	12	Cortex AI Depth	Q45-52
04	Materialized View ROI	Q8	13	Organizational Attribution and Governance	Q53-57
05	Storage Design	Q9-10	14	Counterfactual and Scenario Reasoning	Q58-62
06	Serverless Arbitrage	Q11-12	15	Engineer-Facing Voice	Q63-67
07	Cortex AI	Q13-15	16	PS Onboarding Discovery Sweep	Q68-72
08	Warehouse Forensics and Idle Credit	Q16-22	17	Recommendation Effectiveness and Execution Intelligence	Q73-74
09	Query Design Deep Cuts	Q23-31			

PKG

Needs ownership, lineage or history from the Private Knowledge Graph. A one-off script against ACCOUNT_USAGE won't answer it.

01 Q1-3 / 3 QUESTIONS Modeling Debt and Query Design

A bigger warehouse can mask a bad model for months. These three trace the spill back to the dbt model or notebook that wrote it, so the fix goes to its author.

USE IN Discovery to Optimize

LANDS WITH DataOps, analytics engineering

- 01 Show me every query in the top 100 by credit consumption where disk spillage is recurring. For each one, tell me whether the root cause is undersized memory or a modeling problem, and which dbt model or notebook authored it. PKG
- 02 Which expensive query patterns are being recomputed across three or more downstream models when a single intermediate materialization would serve all of them?
- 03 For the queries with the same query_hash showing degrading performance over the last 60 days, decompose the cause: data volume growth, clustering decay, upstream model change, or warehouse contention? PKG

02 Q4-5 / 2 QUESTIONS Concurrency vs. Capacity

Scale-up and scale-out mistakes look identical on a monthly invoice. Asking which is which, with 30 days of credit waste attached, shows the client you read their workload shape.

USE IN Discovery

LANDS WITH Platform owners, FinOps

- 04 Which warehouses are sized large when they should be scaled out, and which are scaled out when they should be sized larger? Quantify the credit waste from each misclassification over the last 30 days.
 - 05 For each multi-cluster warehouse, show me whether the scaling policy matches the actual workload pattern. Where is an aggressive policy paying for empty clusters, and where is a conservative policy creating queue penalties that exceed the cost of adding capacity?
-

03 Q6-7 / 2 QUESTIONS Clustering Economics

Clustering is the line item clients rarely question because it was switched on by someone who left. These put a payback period on every table.

USE IN Optimize

LANDS WITH Platform owners

- 06 Which tables have automatic clustering enabled where the clustering credits exceeded scan savings over the last 30 days? And which tables should have clustering enabled but don't, ranked by projected payback period?
 - 07 Which manually clustered tables have degraded pruning effectiveness over time, and which are ready for a re-clustering pass? What's the right warehouse to run it on without affecting user-facing workloads?
-

04 Q8 / 1 QUESTION Materialized View ROI

One question, three outcomes: keep, drop, or build. It turns an MV inventory into a short decision list.

USE IN Optimize

LANDS WITH Platform owners, analytics engineering

- 08 Audit every materialized view: which are paying back maintenance through query rewrites, which are pure cost with no rewriter benefit, and which expensive recurring query patterns are missing an MV that would pay back maintenance in under 90 days?
-

05 Q9-10 / 2 QUESTIONS

Storage Design

Durability settings get inherited, never chosen. Pricing Time Travel and Fail-Safe on data nobody would restore is an easy first finding.

USE IN Discovery

LANDS WITH FinOps, platform owners

- 09 Which permanent tables hold data that, based on regeneration patterns and downstream usage, should be transient or temporary? How much Time Travel and Fail-Safe storage cost is going to data that doesn't need that durability?
-
- 10 Which datasets currently in native Snowflake storage are accessed infrequently enough that external tables would be the better architecture? And which existing external tables are queried often enough that ingestion would actually be cheaper?
-

06 Q11-12 / 2 QUESTIONS

Serverless Arbitrage

Both directions matter here. Partners who only ever recommend moving to serverless lose credibility the first time the client's bill goes up.

USE IN Optimize

LANDS WITH DataOps

- 11 Which scheduled tasks running on user-managed warehouses are short or infrequent enough that serverless tasks would be cheaper, and which serverless tasks run often enough they should be consolidated onto a dedicated warehouse?
-
- 12 Which Snowpipe or auto-ingest serverless workloads have spend disproportionate to data volume? Is the right fix batch size, file size, or moving to a triggered warehouse?
-

07 Q13-15 / 3 QUESTIONS

Cortex AI

Snowflake resource monitors don't cover AI services billing, so most clients cannot say who owns their Cortex spend. Asking it by agent, owner and tenant is often the first time anyone has.

USE IN Discovery

LANDS WITH FinOps, AI platform owners

- 13 For every Cortex AI workload in the last 30 days: which agent or workflow consumed the tokens, who is the human owner, and is the model selection right-sized? Specifically, where is Claude Opus being used when Haiku would deliver the same outcome at a fraction of the cost?
-

PKG

- 14 Which Cortex Search services have serving costs disproportionate to query volume, and are there index refresh patterns we could batch or stagger to cut serverless credit consumption without affecting freshness SLAs?
- 15 For our multi-tenant ISV clients, attribute Cortex AI and warehouse cost per end-customer or per workload. Which tenants are driving disproportionate cost relative to their contract value, and what would unit economics look like if we attributed the cost at the workflow level?

PKG

08 Q16-22 / 7 QUESTIONS Warehouse Forensics and Idle Credit

Idle credit is where most first-week findings come from. The difference is splitting tuning problems from workload problems, because they need different owners.

USE IN Discovery

LANDS WITH FinOps, platform owners

- 16 Which warehouses have auto-suspend settings that don't match their actual usage cadence, and what's the monthly idle-credit leak from each one?
- 17 Show me every warehouse where "warm" credits (the time between query end and auto-suspend) exceed 20% of total credit consumption. Which of those are tuning problems vs. workload-pattern problems?
- 18 Which warehouses are serving more than one workload class (BI dashboards, scheduled ETL, ad-hoc analyst, ML training) on a single resource? Quantify the contention cost.
- 19 For each resource monitor in the account, is the threshold actually doing anything, or is it set so high it's decorative? Which are missing entirely from warehouses that need them?
- 20 Which warehouses have statement timeout policies that allow runaway queries to burn credits for more than 10 minutes, and how many credits did those long-runners actually consume last month?
- 21 Identify the top five warehouses by after-hours credit consumption (defined as outside the team's working hours per the PKG's owner mapping). Is the consumption legitimate (overnight ETL) or unintended (forgotten ad-hoc sessions)?
- 22 Which warehouses show a stair-step credit pattern suggesting cluster spin-up overhead is being paid repeatedly for the same workload? Would a min_cluster_count change actually save credits or just shift cost?

PKG

09 Q23-31 / 9 QUESTIONS Query Design Deep Cuts

This is where senior consultants earn their rate. Each question ends in a rewrite someone can ship, with an author attached where it matters.

USE IN Optimize

LANDS WITH Analytics engineering, BI owners

- 23 Which queries are doing SELECT * on wide tables when downstream consumers only use a small column subset, and what's the scan cost difference?

- 24 Identify queries with Cartesian or unintended cross joins, distinguishing legitimate uses from accidents. Who authored each, and which are recurring? PKG

- 25 Which expensive queries would benefit from predicate pushdown that isn't happening? Show me the queries where the optimizer is scanning more than it needs because of a function wrapper or implicit cast.

- 26 What's the result cache hit rate per warehouse, per team, and per BI tool? Where are we recomputing results that should be cache hits but aren't, and why? PKG

- 27 Which queries are functionally identical (same logical output) but written differently enough that the cache misses every time? Show me the deduplication opportunity.

- 28 Identify queries doing heavy semi-structured (VARIANT / OBJECT) parsing at runtime that should be flattened at ingest. What's the compute cost of late parsing vs. shifting it upstream?

- 29 Show me window functions running over unnecessarily large partitions, where a tighter PARTITION BY would cut compute. Rank by credit impact.

- 30 Which queries are using JOIN orders that the optimizer can't fix on its own (typically broadcast joins that should be hash joins, or vice versa), and what's the per-query credit penalty?

- 31 Which BI tools (Sigma, Tableau, Looker) are generating SQL patterns systematically more expensive than equivalent hand-written queries? Show me the worst offending dashboards by credit consumption.

10 Q32-38 / 7 QUESTIONS Storage Health and Decay

Zombie tables and orphaned stage files rarely have a living owner. Finding the project that created them is the hard part, and it's the part the client can't do from a query.

USE IN Discovery

LANDS WITH FinOps, data owners

- 32 Identify zombie tables: created during projects, never deprecated, still accumulating storage. Who owned the original project, and is it still active? PKG
- 33 Show me Time Travel retention windows that exceed what the team's actual recovery needs require. Where is 90-day Time Travel set on tables that have never been time-traveled into?
- 34 Which cloned tables are diverging from their source enough that they're now incurring meaningful independent storage cost, and which clones are stale enough they should just be dropped?
- 35 Audit micro-partition health across the top 50 tables by query volume. Where is partition skew or oversized micro-partitions creating scan inefficiency that no clustering key can fix?
- 36 Which staged files in internal or external stages are accumulating without being purged? Show me the orphaned file inventory and the storage cost.
- 37 Show me cross-region replication costs by database. Which replicated databases have RPO/RTO requirements that justify the cost, and which are inherited configuration that nobody has revisited?
- 38 Identify tables where the column-level storage footprint suggests poor compression (typically wide string columns or unoptimized VARIANT) and quantify the savings from a recompression pass or schema change.

11 Q39-44 / 6 QUESTIONS Pipeline and Orchestration Economics

Over-scheduling compounds quietly. Tying cadence to the freshness SLA someone actually signed gives the data team a reason to say yes.

USE IN Optimize

LANDS WITH DataOps, analytics engineering

- 39 Map every dbt model to its credit consumption per run, then identify the top 20 models by credit-cost-per-business-value (defined by downstream consumer count and freshness SLA). Which are over-engineered? PKG
- 40 Which dbt or Airflow jobs are scheduled more frequently than the downstream data freshness SLA actually requires? Show me the compounded credit waste from over-scheduling.

- 41 Identify streams + tasks chains where the task cadence exceeds the rate at which the stream actually produces change data. Where are tasks firing on empty streams and burning credits?
- 42 For every dynamic table in the account, compare maintenance cost vs. the cost of the equivalent query if it were run on demand. Which dynamic tables aren't paying for themselves?
- 43 Snowpipe latency vs. credit cost per pipe: which pipes are configured for low latency they don't actually need, and which are over-batching and missing freshness SLAs?
- 44 Show me the model dependency depth, meaning the longest dbt DAG paths in production. Where is depth itself driving cost, and where would model consolidation cut the chain? PKG

12 Q 45 - 52 / 8 QUESTIONS Cortex AI Depth

Once the client knows who owns their AI spend, the next meeting is about whether each call earns its cost. Few partners have a point of view here yet.

USE IN Optimize

LANDS WITH AI/ML leads, FinOps

- 45 For every Cortex AI function call (COMPLETE, EXTRACT_ANSWER, SUMMARIZE, TRANSLATE, etc.), show me cost-per-call by model and by calling workflow. Where is the same task being done by an expensive model when a cheaper one would suffice? PKG
- 46 Audit embedding generation costs across the account. Which embeddings are being regenerated repeatedly for unchanged source data, and which Cortex Search indexes are rebuilding more aggressively than the data churn justifies?
- 47 Which Cortex Analyst usage patterns are generating high token spend per question? Are the semantic models too broad, the question patterns inefficient, or both?
- 48 For RAG-style workflows, show me retrieval cost vs. generation cost per query. Where is the retrieval step over-fetching context, inflating the prompt, and driving generation cost up?
- 49 Document AI workloads: which documents are being reprocessed unnecessarily? Show me the per-page cost and the duplicate-processing rate.
- 50 Identify Cortex AI agents whose run frequency exceeds the rate at which they produce a business outcome (a logged action, a record updated, a notification sent). Which agents are firing on noise? PKG
- 51 Compare the cost of current prompt-engineering approaches against the cost of a fine-tuned smaller model for the same task. Where would fine-tuning pay back?
- 52 Which Cortex AI workloads ran during the first 30 days of history, before native anomaly baselines exist? What did independent ingestion catch in that window? PKG

13 Q53-57 / 5 QUESTIONS Organizational Attribution and Governance

This is the section that gets the FinOps leader into the second meeting. RBAC says one thing about who spent the credits; the workload history often says another.

USE IN Discovery and Run

LANDS WITH FinOps, CIO, exec sponsor

- 53 Decompose this month's bill by cost center, by team, and by individual engineer. Which engineers are responsible for the largest credit deltas vs. last month, and is the delta justified by output? PKG
- 54 Identify shadow IT: service accounts and warehouses created outside the official provisioning process. What are they doing, and what's the monthly spend nobody is owning? PKG
- 55 Show me the gap between role-based cost allocation (what RBAC says) and actual workload attribution (what the PKG sees). Where is allocation lying about who actually consumed the credits? PKG
- 56 Which roles have privileges to create resources that incur cost (warehouses, automatic clustering, MVs) but no cost monitor attached? Map the governance blind spots.
- 57 For multi-region or multi-cloud deployments, show me cost per cloud and per region. Where are we paying a regional premium that isn't justified by data gravity or latency requirements?

14 Q58-62 / 5 QUESTIONS Counterfactual and Scenario Reasoning

Clients sign changes they can see the downside of. A forecast with its assumption set written out is easier to approve than a recommendation.

USE IN Run and planning

LANDS WITH FinOps, platform owners

- 58 If we moved every short, infrequent task from warehouse XS to serverless tomorrow, what's the projected monthly credit delta? Show me the assumption set and the confidence interval.
- 59 What's the projected cost if our current workload growth rate continues for the next two quarters, broken down by warehouse, by team, and by Cortex vs. compute? PKG
- 60 Simulate: if we dropped MV X, what would happen to query latency and credit cost across the queries that currently use it? Is the MV load-bearing or load-decorative?
- 61 If we tightened Time Travel from 30 days to 7 days on the top 20 storage-cost tables, what's the recoverable storage cost, and what's the operational risk based on past Time Travel usage in this account? PKG
- 62 If we doubled warehouse X's size to eliminate spillage, what's the net credit impact factoring in faster query completion vs. higher per-second cost? Is upsize cheaper than the spillage it would prevent?

15 Q 63-67 / 5 QUESTIONS Engineer-Facing Voice

Engineers decide whether a recommendation ever ships. Give them questions phrased in the first person and they start using the engagement between your status calls.

USE IN Run **LANDS WITH** Individual engineers, DataOps champion

- 63 Which of my queries from the last 30 days are the biggest credit consumers, and what's the simplest change for each one? I want a list I can fix this afternoon. PKG
- 64 Why is the query I just ran slower than it was last week? Decompose it for me: data volume, plan change, warehouse contention, clustering decay, or something else? PKG
- 65 Compare my query patterns to the team average. Where am I writing SQL that's measurably less efficient than my peers, and what should I learn from them? PKG
- 66 Show me which of my dbt models are likely to be flagged in the next FinOps review. Give me a heads-up so I can fix them before they end up in someone's slide deck. PKG
- 67 I'm about to ship this query to production. Run it through the PKG: does it match patterns we've already flagged as expensive in this account, and if so, what's the recommended rewrite? PKG

16 Q 68-72 / 5 QUESTIONS PS Onboarding Discovery Sweep

Five questions for the first two weeks of a new account. Question 70 alone tells you whether to scope the SOW around tooling or around process.

USE IN Discovery (week one) **LANDS WITH** Engagement lead, exec sponsor

- 68 Generate a 30-day cost decomposition for this new client: compute vs. storage vs. Cortex vs. serverless, by warehouse, by team, with month-over-month deltas. This is the slide I open the engagement with. PKG
- 69 Rank every optimization opportunity in this account by projected annual savings divided by estimated engineering effort. I want the quick-win inventory for the first 90 days.
- 70 Show me anomaly history for this account over the last 12 months: what spiked, how often, and was anything ever resolved? This tells me whether the client has an enforcement problem or a discipline problem. PKG
- 71 Identify the three workloads in this account that, if fixed first, would generate the case study that my firm needs to justify the expanded engagement at the QBR. PKG

- 72 Map the client's current Snowflake topology against best-practice patterns the PKG has seen across other accounts. Where is this client materially off-pattern, and where are they actually doing it right?

PKG

Q73-74 / 2 QUESTIONS

17 Recommendation Effectiveness and Execution Intelligence

The two questions that renew engagements. They measure what shipped against what was promised, which is the number a sponsor takes to their CFO.

USE IN Run (QBR)

LANDS WITH Exec sponsor, FinOps

- 73 Show me every optimization recommendation generated in the last 12 months that was never implemented. What was the projected savings, what blocked adoption, and how much unrealized value remains on the table today?

PKG

- 74 Across all cost optimization actions completed in this account, which interventions delivered the highest realized ROI versus forecasted ROI? Use that history to rank future recommendations by expected execution success, not just projected savings.

PKG

GETTING STARTED

Take the questions into your next pursuit. APEX answers them.

1

Book a partner briefing

30 minutes with the Anavsan team. We walk through the bank and match questions to the accounts you're pursuing now.

2

Connect one client account

APEX ingests operational metadata only and starts building that client's PKG. No query bodies leave the account, and Anavsan holds no credential into it.

3

Run the discovery sweep

Take Q68 to Q72 into your first workshop. Use the answers to scope the SOW around the client's actual problem instead of a generic optimization package.

4

Route every fix to an owner

Findings become draft pull requests assigned to the engineer who owns the workload, with a human sign-off before anything changes. You report what shipped at each QBR.

Pick one live pursuit. We'll help you choose the five questions to open with.

partner@anavsan.com
anavsan.com/partners