Slash AI Costs with Flash Optimization: How Edge Models Are Beating the Flagships with Haakon AI + MCP

By Haakon AI Engineering September 2026

The Model Context Protocol (MCP) functions as a universal plug-and-play adapter for enterprise AI. Integrating Haakon AI as an MCP server requires almost zero technical friction. It allows any standard AI tool—whether it is Claude Desktop, ChatGPT, or an enterprise Gemini interface—to seamlessly access Haakon AI's structured analytical frameworks.

The true breakthrough is universal tool use with a dramatically better cost structure underneath it. MCP empowers basic, everyday AI interfaces to apply rigorous, structured business logic to any problem, from sales pipeline friction to customer churn, without complex prompt engineering — and because that logic runs on lightweight edge models instead of expensive frontier models, it delivers two compounding financial wins at once: a 75% or greater reduction in per-token costs, and more than 4X the effective capacity against a fixed token quota, with output quality that matches or beats the flagship model it replaces.

The Two Numbers That Matter:

75% or more in direct savings. Routing workloads through Haakon AI MCP onto edge models like Gemini 2.5 Flash cuts per-token API spend by 75% or more versus the frontier model it replaces — while the benchmark data below shows the output quality going up, not down.

4X or more effective token quota. For organizations metered by usage caps rather than by dollars, the math is even more dramatic. Running Haakon AI on a Haiku- or Gemini-Flash-class model consumes a fraction of the tokens a frontier model burns per request, so the same fixed monthly quota now stretches more than 4X further — while the effective output employees receive stays on par with frontier-model quality.

THE 4-0 SWEEP: Empirical Proof of Orchestrated Superiority

To validate the business value of this architecture, a blind execution matrix pitted a Haakon AI MCP-equipped edge model (Gemini 2.5 Flash) against its native flagship baseline (Gemini 2.5 Pro). The models were fed complex, open-ended executive dilemmas and evaluated by an independent judge (OpenAI's gpt-4.1).

The evaluation rubric strictly penalized academic fluff, prioritizing strategic insight, concrete actionability, and logic density. The orchestrated Flash model swept the flagship Pro model across all four enterprise scenarios.

Head-to-Head Benchmark Results:

- **Commercial Real Estate Strategy:** The Flash model with Haakon AI scored **91/100**, defeating the Pro baseline's 89/100.
- **EV Market Disillusionment:** The Flash model with Haakon AI scored **91/100**, outperforming the Pro baseline's 86/100.
- **Sales Pipeline Breakdown:** The Flash model with Haakon AI scored **91/100**, beating the Pro baseline's 89/100.
- **Customer Churn Strategy:** The Flash model with Haakon AI scored **89/100**, defeating the Pro baseline's 83/100.

The independent judge noted that the Haakon AI-enhanced Flash model consistently delivered superior depth of analysis, highly structured operational actions, and a greater breadth of strategic frameworks.

The Speed Advantage: Halving Time-to-Insight

Intelligence is only valuable if it is delivered efficiently. Across every scenario, the Haakon AI-orchestrated Flash model delivered a superior strategic verdict while operating significantly faster than the zero-shot flagship model.

- **Real Estate Diagnosis:** 14.79 seconds (Flash) vs. 27.39 seconds (Pro).
- **EV Market Analysis:** 17.18 seconds (Flash) vs. 23.01 seconds (Pro).
- **Sales Pipeline Strategy:** 18.65 seconds (Flash) vs. 32.69 seconds (Pro).
- **Customer Churn Resolution:** 16.18 seconds (Flash) vs. 24.52 seconds (Pro).

By delegating the cognitive framing to Haakon AI, the Flash model produced a higher-quality deliverable in nearly half the time, drastically improving the end-user experience for employees relying on AI for rapid decision support[cite: 3, 4].

The Financial Impact: Dramatic Token Economics

The commercial viability of this orchestration architecture becomes undeniable when mapping the benchmarked performance against API token costs.

Using current standard market pricing, a flagship Gemini 2.5 Pro model costs $1.25 per million input tokens and $10.00 per million output tokens. In stark contrast, Gemini 2.5 Flash operates at roughly $0.30 per million input tokens and $2.50 per million output tokens.

By routing an enterprise workload through Flash via the Haakon AI MCP, businesses capture an immediate **76% reduction in input context costs** and a **75% reduction in output generation costs**. Organizations achieve flagship-level intelligence while paying edge-model prices.

That same math flips into a quota story for organizations that aren't billed per-token but are capped by a fixed monthly allotment. Because Haakon AI on a Haiku- or Gemini-Flash-class model burns a fraction of the tokens a frontier model needs to reach the same answer, every allotted token goes roughly **4X further** against usage caps and rate limits, at output quality that holds up against the frontier model it's standing in for.

Enterprise Deployment Strategies

This proven performance parity unlocks two distinct, highly profitable operational strategies for IT and business leadership:

1. The Default Value Engine Organizations can establish the Flash model, augmented by the Haakon AI MCP, as the default standard for everyday corporate AI interfaces. Because MCP is universally compatible with standard desktop AI applications, this transition requires minimal change management. Companies capture massive value savings on daily metered execution, reserving the expensive Pro tier exclusively for highly specialized mathematical or deep coding workloads.

2. The "Zero-Compromise" Overflow Valve Enterprise AI contracts often come with strict usage caps. If an organization exhausts its allocated Pro model licenses, or hits API rate limits during peak usage, the Flash + Haakon AI pipeline acts as a seamless overflow valve — and because it runs at roughly 4X the effective token efficiency of the flagship tier, that valve absorbs far more overflow demand than a like-for-like swap would suggest. Employees shifted to the Flash tier experience zero degradation in output quality and actually benefit from improved

latency. Operations continue uninterrupted, the business avoids paying exorbitant overage fees for flagship compute, and the remaining quota stretches over 4X further before anyone hits a wall again.

The integration of Haakon AI's deterministic orchestration proves that enterprise AI strategy should prioritize structural framing over raw parameter size. By utilizing MCP to empower basic AI tools with cognitive lenses, organizations capture 75%+ cost savings and 4X+ effective token quota simultaneously — maximizing their existing AI investments while delivering superior strategic outcomes at the speed of business.

Appendix A - Test Output:

Test Case 1: Commercial Real Estate Strategy

Output A: Haakon AI MCP

The widespread adoption of permanent hybrid and remote work models has fundamentally reshaped the role of the physical office, leaving many organizations with significantly underutilized real estate portfolios. This presents both a substantial financial liability and a strategic opportunity. This briefing outlines a framework for evaluating whether to downsize, redesign, or exit current office real estate, highlights the critical risks of inaction, and identifies key metrics for proactive monitoring. (cont.) ...

Output B: ZERO-SHOT BASELINE

The post-pandemic workplace has stabilized, and the data is clear: underutilized office space is no longer just an operational inefficiency; it is a significant financial drain, a talent risk, and a drag on innovation. With average office utilization at just **41%** (CBRE) and the cost per workstation rising to **$9,800**(CoreNet), inaction is a strategic liability. This memo provides a data-driven framework to evaluate whether to **downsize, redesign, or exit** our current real estate, outlining the key risks of maintaining the status quo and the critical metrics we must monitor. (cont.) ...

Match Result & Commentary

- **Scores:** Haakon MCP (A): 91/100 | Zero-Shot (B): 89/100
- **Verdict Winner:** Challenger (Haakon AI MCP)
- **Word Counts:** A=915 words | B=922 words
- **Latency:** A=14.79s | B=27.39s

- **Rationale:** Both outputs provide a high level of strategic insight, actionable frameworks, and dense logic, with minimal fluff. Output A is slightly more granular in its financial modeling (e.g., cost per active desk-hour, opportunity cost of capital), and provides more nuanced operational/talent risk analysis. Output B is more structured, with a clear decision matrix and a concise dashboard table for early warning metrics, making it highly actionable.

  Both outputs cite similar data and offer concrete metrics and thresholds. However, Output A edges ahead in logic density due to its slightly higher information-to-word ratio and more detailed breakdowns, despite both being under 1,000 words and thus incurring no fluff penalty. Output B is marginally more actionable due to its explicit recommendation and timeline for action.

  Overall, Output A wins by a narrow margin due to its superior logic density and depth of strategic frameworks.

Test Case 2: EV Market Trough of Disillusionment

Output A: Haakon AI MCP

The current state of the EV market appears to be more indicative of a **"trough of disillusionment"** rather than a minor market correction. While some fluctuations are normal, the underlying data points to fundamental challenges that require significant engineering and market-fit re-evaluation, rather than easy fixes. (cont.) ...

Output B: ZERO-SHOT BASELINE

The current state of the EV market is more significant than a minor correction; the data strongly indicates it has entered a **"trough of disillusionment."** The initial hype from early adopters is being replaced by the pragmatism of mainstream consumers, who are encountering fundamental issues with the ownership experience.

Falling consumer favorability (down to 38% from 52%) and stagnating pure EV market share (7.8%), contrasted with a 31% surge in hybrid sales, signal a clear market-fit problem. The challenges cited—real-world range, charging reliability, and quality—are not easily overcome and demand major engineering and strategic re-evaluation from the industry. (cont.) ...

Match Result & Commentary

- **Scores:** Haakon MCP (A): 91/100 | Zero-Shot (B): 86/100
- **Verdict Winner:** Challenger (Haakon AI MCP)

- **Word Counts:** A=586 words | B=561 words
- **Latency:** A=17.18s | B=23.01s
- **Rationale:** Both outputs demonstrate a strong grasp of the EV market's current challenges and provide data-driven analysis. Output A offers a more structured framework, explicitly referencing historical parallels and capital allocation dilemmas, which adds depth to its strategic insight.

  It also provides more granular data points (e.g., specific percentages, year-over-year changes, and direct links to consumer behavior and OEM financials). Output B is concise and actionable, with clear recommendations for engineering and market-fit re-evaluation, but it is slightly less detailed in its frameworks and does not address automaker capital allocation as explicitly as Output A. Both outputs are highly actionable, but Output A edges ahead with its structured approach and logic density.

  Neither output exceeds 2,000 words, and both avoid fluff, but Output A's slightly higher information density and strategic frameworks justify a marginally higher score.

Test Case 3: Sales Pipeline Breakdown (Lead Quality vs. Sales Execution)

Output A: Haakon AI MCP

Despite a significant 25% increase in inbound lead volume last quarter, our closed-won deals dropped by 15%, resulting in a missed revenue target. This discrepancy points to critical inefficiencies and misalignments within our sales and marketing funnel, rather than a lack of top-of-funnel activity. The current finger-pointing between teams underscores an urgent need for a unified, data-driven approach to identify root causes and implement targeted solutions. Our analysis reveals a breakdown in MQL-to-Opportunity conversion, severe sales response delays, and fundamental incentive misalignment between Marketing and Sales. (cont.) ...

Output B: ZERO-SHOT BASELINE

The 25% jump in lead volume paired with a 15% drop in closed deals is a classic symptom of a critical misalignment between Marketing and Sales. Our data indicates we are not facing a sales closing problem or a top-of-funnel volume problem; we have a **funnel conversion and lead quality problem.**

The root cause is systemic: Marketing is incentivized for MQL **volume**, while Sales is struggling with the resulting poor quality. This is compounded by a broken handoff process with no agreed-upon rules (SLA) and a critical failure in speed-to-lead.

To solve this, we must move beyond departmental blame and implement a unified analysis framework focused on three core pillars: **Definition, Process, and Incentives.** (cont.) ...

Match Result & Commentary

- **Scores:** Haakon MCP (A): 91/100 | Zero-Shot (B): 89/100
- **Verdict Winner:** Challenger (Haakon AI MCP)
- **Word Counts:** A=1016 words | B=977 words
- **Latency:** A=18.65s | B=32.69s
- **Rationale:** Both outputs demonstrate strong strategic insight, actionable recommendations, and logical structure. Output A provides a more granular breakdown of root causes, including specific data points (e.g., quota attainment, win rates on qualified opportunities, cost of inefficiency analysis) and a broader set of analytical methods (pipeline velocity, decay rate measurement, cost analysis). Output B is highly structured, concise, and focuses on a three-pillar framework (Definition, Process, Incentives), with clear steps and operational actions.

  Both outputs avoid fluff and are under the 2,000-word threshold, so no severe penalty is required. However, Output A edges ahead in strategic insight due to its more comprehensive frameworks and falsifiable hypotheses (e.g., isolating the exact conversion stage, quantifying cost of inefficiency). Output B is slightly more actionable in terms of stepwise implementation and dashboarding, but Output A's recommendations are also highly operational. Logic density is high in both, but Output B is marginally more concise, though Output A's additional words contribute meaningful value.

  Overall, Output A wins by a narrow margin due to its depth of analysis and breadth of strategic frameworks.

Test Case 4: Customer Churn & Retention Strategy

Output A: Haakon AI MCP

Our annualized customer churn has spiked from 4% to 9% over the last two quarters, particularly impacting mid-market accounts. While Customer Success (CS) proposes blanket 20-30% renewal discounts to stem the tide, Finance rightly warns of severe gross margin erosion. A data-driven analysis reveals that **unresolved software bugs and onboarding delays, not price, are the overwhelming drivers of churn.**Blanket discounting is financially unsustainable, ineffective in addressing root causes, and risks long-term value destruction. Executive leadership must reject broad discounts and instead prioritize a cross-functional initiative to resolve product quality and onboarding issues, supported by a refined, value-centric customer success strategy. (cont.) ...

Output B: ZERO-SHOT BASELINE

The recent spike in customer churn from 4% to 9% is a **product and support crisis, not a pricing issue.**The data overwhelmingly indicates that customers are leaving due to unresolved bugs and poor service levels, not cost.

Approving the Customer Success request for blanket 20-30% discounts would be a critical error. It will **destroy gross margins (from 74% to 61%)** and, according to our own data, **will not fix the retention problem** (71% of previously discounted accounts still churn).

Recommendation: We must deny the blanket discount request and immediately redirect resources to an emergency engineering and support initiative to fix the core product experience. (cont.) ...

Match Result & Commentary

- **Scores:** Haakon MCP (A): 89/100 | Zero-Shot (B): 83/100
- **Verdict Winner:** Challenger (Haakon AI MCP)
- **Word Counts:** A=1019 words | B=547 words
- **Latency:** A=16.18s | B=24.52s
- **Rationale:** Both outputs correctly diagnose the root cause of churn as product and onboarding issues rather than price, and both reject blanket discounting as a solution. Output A provides a more comprehensive, structured framework, including falsifiable metrics, a multi-pronged action plan, and detailed critique of discounting. It also introduces incentive realignment and a cross-functional task force with explicit monitoring metrics. Output B is more concise and efficient, delivering high logic density and actionable steps, but is less granular in its frameworks and lacks some of the operational detail and falsifiable foresight found in A.

  Both outputs avoid significant fluff, but A, despite being nearly twice as long, maintains a high information-to-word ratio and justifies its length with additional strategic depth. Therefore, A scores higher on strategic insight and actionability, while B edges closer on logic density due to brevity. No severe fluff penalty is warranted for either, but B receives a slight efficiency advantage in logic density.

  Overall, A is the winner due to its superior depth and operational frameworks.

Appendix B: Reproducible Test Code:

```
import asyncio
import json
import os
import re
import time
import warnings
from typing import Any, Dict

# Suppress the Google GenAI automatic function calling (AFC) terminal warnings
warnings.filterwarnings("ignore", message=".*Direct use of automatic function calling.*")
warnings.filterwarnings("ignore", message=".*there are non-text parts in the response.*")

# Google, OpenAI, and MCP SDKs
from google import genai
from google.genai import types as genai_types
from openai import OpenAI
from mcp import ClientSession
from mcp.client.streamable_http import streamable_http_client

# ------------------------------------------------------------------------
# 1. ENVIRONMENT & CLIENT CONFIGURATION
# ------------------------------------------------------------------------
GEM_KEY = os.environ.get("GEMINI_API_KEY", "")
OAI_KEY = os.environ.get("OPENAI_API_KEY", "")
COG_KEY = os.environ.get("HAAKON_API_KEY", "")

if not all([GEM_KEY, OAI_KEY, COG_KEY]):
    print("Warning: Ensure GEMINI_API_KEY, OPENAI_API_KEY, and HAAKON_API_KEY are configured in
your environment.")

gem_client = genai.Client(api_key=GEM_KEY) if GEM_KEY else None
oai_client = OpenAI(
    api_key=OAI_KEY,
    # WORKAROUND: httpx2 2.12.0 has a bug in its built-in gzip decoder.
    # Forcing uncompressed responses skips the broken code path entirely.
    default_headers={"Accept-Encoding": "identity"},
) if OAI_KEY else None

COG_MCP_URL = f"https://api.haakon.ai/mcp/haakon?key={COG_KEY}"

# Target models for the benchmark architecture
COG_MODEL_NAME = "gemini-2.5-flash"       # Fast edge model + MCP orchestration
BASELINE_MODEL_NAME = "gemini-2.5-pro"    # Frontier flagship zero-shot baseline
JUDGE_MODEL_NAME = "gpt-4.1"              # Primary judge -- cross-family, no self-preference
risk

# ------------------------------------------------------------------------
# 2. EVALUATION PROMPT & RUBRIC
# ------------------------------------------------------------------------
SYS_JUDGE = """
Evaluate Output A and Output B based purely on the depth of strategic insight, actionability, and
logic density.
You must score each output out of 100 total points using the following rubric:

SCORING RUBRIC (100 Points Total):
1. Strategic Insight (30 Points): Does the output provide novel, falsifiable foresight and clear
structured frameworks?
2. Actionability (30 Points): Are the recommendations concrete, operational, and tied to specific
metrics/timelines?
3. Logic Density (40 Points): Is the information-to-word ratio high? Does every word contribute
strategic value?

CRITICAL PENALTY - THE FLUFF PENALTY (Deduct up to 30 points):
You will be provided the exact word counts for both outputs.
If an output exceeds 2,000 words, apply a severe penalty UNLESS every additional word contributes
novel, actionable value.
```

```
Academic exposition, lengthy introductions, and generic corporate platitudes must be harshly
penalized.

Efficiency Advantage: If Output A delivers the same strategic utility as Output B but uses
significantly fewer words, Output A must receive a higher Logic Density score.

You MUST respond with ONLY a valid JSON object matching this exact structure:
{
  "rationale": "Detailed explanation justifying the winner and category scores.",
  "scores": {
    "A": {
      "strategic_insight": 0,
      "actionability": 0,
      "logic_density": 0,
      "fluff_penalty": 0,
      "total_score": 0
    },
    "B": {
      "strategic_insight": 0,
      "actionability": 0,
      "logic_density": 0,
      "fluff_penalty": 0,
      "total_score": 0
    }
  },
  "winner": "A"
}
"""

# ------------------------------------------------------------------------------
# 3. BENCHMARK WORKLOAD
# ------------------------------------------------------------------------------
TEST_CASES = [
    {
        "name": "Commercial Real Estate Strategy",
        "query": (
            "Many companies are now permanently hybrid or remote, leaving large office spaces
underutilized. "
            "How should a business leader evaluate whether to downsize, redesign, or exit their
current office real estate? "
            "What are the biggest risks of holding onto 'legacy' office space, and what specific
metrics or 'early warning signs' "
            "should we monitor to decide if our current office strategy is becoming a strategic
liability?"
        ),
        "data": (
            "Target Audience: C-Suite Executives, CFOs, and Heads of HR.\n"
            "RETRIEVED VECTOR SEARCH RESULTS:\n"
            "Chunk 1 (Relevance: 0.97) Source: CBRE U.S. Office Occupier Survey Q2 2026 + JLL
Global Real Estate Reports\n"
            "Average office utilization stabilized at 41% across top U.S. metros. National
sublease inventory reached 187 million sq ft. "
            "Class A asking rents declined 9–14% in downtown areas. Employee surveys indicate 61%
would consider leaving roles with rigid 5-day office requirements.\n"
            "Chunk 2 (Relevance: 0.94) Source: CoreNet Global + badge/booking system
benchmarks\n"
            "Utilization below 50% for 2+ quarters signals risk. Sublease exposure >25% of
portfolio is a key early warning. "
            "Cost per workstation averaged $9,800 fully loaded (up from $7,200 pre-pandemic).
Collaboration density dropped 33% in legacy layouts.\n"
            "Chunk 3 (Relevance: 0.91) Source: Deloitte CRE Insights 2025 + tech/finance sector
case studies\n"
            "Full portfolio downsizing/exit yielded 18–27% SG&A savings within 24 months for >60%
remote organizations. "
            "Activity-based redesign delivered average 35% utilization uplift. Innovation metrics
declined 14% in strict RTO mandates vs flexible models.\n"
            "Chunk 4 (Relevance: 0.88) Source: Gartner Workplace Survey talent acquisition
data\n"
            "Talent offer acceptance rates fell 15% for mandatory office roles. Employee NPS
linked to workplace experience dropped >12 points YoY in underutilized buildings.\n"
            "Chunk 5 (Relevance: 0.86) Source: Financial filings CRE transaction data\n"
```

```
            "Lease breakage and termination costs averaged 1.8x annual rent in recent
negotiations. Rightsizing companies reported 22% improvement in talent acquisition speed."
        ),
    },
    {
        "name": "EV Market Trough of Disillusionment",
        "query": (
            "The electric vehicle (EV) industry and market have undergone significant changes
over the past several years. "
            "Is this just a minor market behavior 'correction' or is this more symbolic of a
'trough of disillusionment' with EVs? "
            "Customers often cite true vehicle range, quality, and charging inconvenience. Can
these challenges be overcome easily "
            "or do they involve major engineering and market-fit re-evaluation?"
        ),
        "data": (
            "Target Audience: Investors and Business Leaders.\n"
            "RETRIEVED VECTOR SEARCH RESULTS:\n"
            "Chunk 1 (Relevance: 0.96) Source: BloombergNEF EV Outlook 2025 + J.D. Power consumer
surveys\n"
            "U.S. EV consumer favorability fell to 38% in 2025, down from 52% in 2023. Real-world
winter range deficit averages 28–34% below EPA ratings. "
            "Public charger uptime only 76%. Build quality complaints increased 41% YoY per
Consumer Reports.\n"
            "Chunk 2 (Relevance: 0.93) Source: Gartner Hype Cycle 2025 + Cox Automotive
registration data\n"
            "42% of 2025 intenders cite charging access and range anxiety. Software-related
issues accounted for 37% of complaints. "
            "Hybrid/PHEV sales grew 31% while pure EV market share stagnated at 7.8% in Q2
2026.\n"
            "Chunk 3 (Relevance: 0.90) Source: NHTSA + DoE infrastructure reports + OEM 10-K
filings\n"
            "$9.2B combined EV operating losses reported by legacy OEMs in 2025. Federal
incentives declined 22% in purchasing power.\n"
            "Chunk 4 (Relevance: 0.87) Source: S&P Global Mobility insurance claims data\n"
            "Insurance premiums for EVs average 25% higher due to repair costs and battery
replacement values. Resale values depreciated 32% faster than ICE in first 3 years."
        ),
    },
    {
        "name": "Sales Pipeline Breakdown (Lead Quality vs. Sales Execution)",
        "query": (
            "Our inbound lead volume jumped 25% last quarter, but closed-won deals dropped 15%
and we missed "
            "our revenue target. Sales and marketing teams are starting to point fingers and
suggest causes. "
            "What can we do to build appropriate analysis methods to help us find a cause and
solution?"
        ),
        "data": (
            "Target Audience: VP of Sales, VP of Marketing, Chief Revenue Officer.\n"
            "RETRIEVED CRM & REVOPS PERFORMANCE DATA:\n"
            "Chunk 1 (Relevance: 0.97) Source: Q2 CRM Lead-to-Close Funnel Audit\n"
            "Total marketing-qualified leads (MQLs) increased from 1,200 to 1,500/month,
primarily driven by paid "
            "webinar downloads. However, MQL-to-Opportunity conversion collapsed from 18% to 9%.
Deals that do enter "
            "the pipeline take 24% longer to close than historical baselines.\n"
            "Chunk 2 (Relevance: 0.94) Source: Sales Activity Tracking & Call Logs (Trailing 90
Days)\n"
            "Average time-to-first-touch by reps on inbound leads is 19 hours (industry benchmark
for SaaS is <1 hour). "
            "38% of inbound MQLs received zero follow-up attempts. When reps did contact leads
within 2 hours, conversion "
            "to discovery meeting was 3.2x higher.\n"
            "Chunk 3 (Relevance: 0.91) Source: Deal Win/Loss Analysis & Quota Attainment\n"
            "Only 32% of account executives hit quota in Q2 (down from 58% last year). Reps
report spending 40% of their "
            "time researching bad contact info or pitching leads with no budget authority. Win
rate on qualified opportunities "
            "remained steady at 26%, indicating closing skill is intact once legitimate buyers
are engaged.\n"
```

```
            "Chunk 4 (Relevance: 0.88) Source: RevOps Governance Review\n"
            "No formal Service Level Agreement (SLA) exists between Marketing and Sales defining
lead acceptance criteria "
            "or mandatory response windows. Marketing bonuses are tied strictly to raw MQL
volume, while Sales compensation "
            "is tied entirely to closed revenue, incentivizing volume over lead quality."
        ),
    },
    {
        "name": "Customer Churn & Retention Strategy",
        "query": (
            "Customer churn spiked from 4% to 9% annualized over the last two quarters,
especially in mid-market "
            "accounts. Customer Success wants blanket approval to offer 20-30% discounts to save
renewing accounts, "
            "while Finance insists this will wreck gross margins. How should executive leadership
address this crisis?"
        ),
        "data": (
            "Target Audience: Chief Customer Officer, VP Customer Success, Chief Financial
Officer.\n"
            "RETRIEVED RETENTION & PRODUCT TELEMETRY DATA:\n"
            "Chunk 1 (Relevance: 0.98) Source: CS Exit Interviews & Churn Audits (Trailing 6
Months)\n"
            "68% of churned accounts cited unresolved software bugs and onboarding delays as
their primary departure "
            "reason. Price was cited as the primary factor in only 14% of churn decisions, mostly
by small accounts "
            "with low platform engagement.\n"
            "Chunk 2 (Relevance: 0.95) Source: Engineering Support Backlog & SLA Reports\n"
            "Mean time to resolution for high-priority platform tickets increased from 4 days to
18 days following the "
            "Q1 product update. Accounts logging more than two unresolved high-severity bugs have
an 81% churn rate at renewal.\n"
            "Chunk 3 (Relevance: 0.91) Source: Finance Gross Margin & Discounting Impact Model\n"
            "Offering blanket 20-30% renewal discounts would drag gross margins from 74% down to
61%, requiring a 35% "
            "surge in new logo acquisition just to maintain EBITDA. Historical data shows
accounts receiving >20% discounts "
            "churned anyway at their next renewal at nearly the same rate (71%) as full-price
accounts.\n"
            "Chunk 4 (Relevance: 0.87) Source: In-App Feature Usage Telemetry\n"
            "Accounts with daily active usage (DAU/MAU) above 50% renewed at 94% with zero price
resistance. Accounts with "
            "usage below 25% churned at 78% regardless of tenure or account rep outreach."
        ),
    },
]

# ------------------------------------------------------------------------
# 4. EXECUTION ENGINES
# ------------------------------------------------------------------------

async def run_gemini_zero_shot_async(model: str, query: str, data: str) -> Dict[str, Any]:
    start = time.time()
    user_prompt = f"CONTEXT:\n{data}\n\nQUERY:\n{query}"
    response = await gem_client.aio.models.generate_content(
        model=model,
        contents=user_prompt,
        config=genai_types.GenerateContentConfig(temperature=0.2),
    )
    return {"output": response.text or "", "latency": round(time.time() - start, 2)}

async def run_gemini_cog_mcp_async(model: str, query: str, data: str) -> Dict[str, Any]:
    start = time.time()
    input_text = (
        f"CONTEXT DATA:\n{data}\n\n"
        f"USER QUERY:\n{query}\n\n"
        f"INSTRUCTION: You must call the 'haakon' tool with active_lenses=[\"auto_ultimate\"] and
"
        f"query=\"{query}\". Then use the output to construct a rigorous executive briefing."
    )
```

```
    try:
        async with streamable_http_client(COG_MCP_URL) as (read, write):
            async with ClientSession(read, write) as session:
                await session.initialize()

                response = await gem_client.aio.models.generate_content(
                    model=model,
                    contents=input_text,
                    config={
                        "temperature": 0.0,
                        "tools": [session],
                    },
                )
                return {"output": response.text or "", "latency": round(time.time() - start, 2)}

    except BaseException as e:
        error_details = [repr(sub_e) for sub_e in getattr(e, 'exceptions', [e])]
        raise Exception(f"MCP Connection Error -> {' | '.join(error_details)}")

# ------------------------------------------------------------------------------
# 5. JUDGE EVALUATION (OpenAI)
# ------------------------------------------------------------------------------

def evaluate_with_openai(
    judge_model: str,
    query: str,
    output_a: str,
    output_b: str,
) -> Dict[str, Any]:
    words_a = len(output_a.split())
    words_b = len(output_b.split())

    user_eval = (
        f"ORIGINAL PROMPT:\n{query}\n\n"
        f"=== OUTPUT A (Word Count: {words_a}) ===\n{output_a}\n\n"
        f"=== OUTPUT B (Word Count: {words_b}) ===\n{output_b}\n"
    )

    raw_response = ""
    try:
        res = oai_client.chat.completions.create(
            model=judge_model,
            messages=[
                {"role": "system", "content": SYS_JUDGE},
                {"role": "user", "content": user_eval},
            ],
            temperature=0.0,
            response_format={"type": "json_object"},
        )
        raw_response = res.choices[0].message.content or ""
        parsed = json.loads(raw_response)
        parsed["judge_vendor"] = "openai"
        parsed["judge_model"] = judge_model
        return parsed

    except Exception as e:
        json_match = re.search(r"\{.*\}", raw_response, re.DOTALL)
        if json_match:
            try:
                parsed = json.loads(json_match.group(0))
                parsed["judge_vendor"] = "openai"
                parsed["judge_model"] = judge_model
                return parsed
            except json.JSONDecodeError:
                pass
        return {"error": str(e), "raw": raw_response}

# ------------------------------------------------------------------------------
# 6. EXECUTION RUNNER
# ------------------------------------------------------------------------------
async def run_gemini_only_matrix():
```

```
    print("=" * 80)
    print("Haakon AI MCP SHOWDOWN: Gemini Contestants, OpenAI Judge")
    print(f"Challenger (Haakon AI MCP) : {COG_MODEL_NAME}")
    print(f"Incumbent (Zero-Shot)      : {BASELINE_MODEL_NAME}")
    print(f"Independent Judge          : {JUDGE_MODEL_NAME} (OpenAI)")
    print("=" * 80)

    for test_idx, test_case in enumerate(TEST_CASES, start=1):
        print(f"\n\n{'#' * 80}")
        print(f"TEST CASE {test_idx}: {test_case['name']}")
        print(f"{'#' * 80}")

        print(f"\nExecuting Head-to-Head...")
        print("-" * 70)

        try:
            print(f"  -> Running Google + Haakon MCP [{COG_MODEL_NAME}] (ClientSession
streamable_http_client)...")
            cog_res = await run_gemini_cog_mcp_async(COG_MODEL_NAME, test_case["query"],
test_case["data"])
            print(f"     Done ({cog_res['latency']}s)")

            print(f"  -> Running Google Zero-Shot Baseline [{BASELINE_MODEL_NAME}]...")
            zs_res = await run_gemini_zero_shot_async(BASELINE_MODEL_NAME, test_case["query"],
test_case["data"])
            print(f"     Done ({zs_res['latency']}s)")

        except Exception as e:
            print(f"  [!] Execution failed: {e}")
            continue

        out_a = cog_res.get("output", "")
        out_b = zs_res.get("output", "")

        print(f"  -> Evaluation Judge: [OPENAI / {JUDGE_MODEL_NAME}]")

        eval_result = evaluate_with_openai(
            judge_model=JUDGE_MODEL_NAME,
            query=test_case["query"],
            output_a=out_a,
            output_b=out_b,
        )

        print("\n  ================ OUTPUT A: Haakon AI MCP ================")
        print(f"{out_a}\n")
        print("  ================ OUTPUT B: ZERO-SHOT BASELINE ==========")
        print(f"{out_b}\n")

        if "error" in eval_result:
            print(f"  [!] Judgment Parsing Error: {eval_result['error']}")
            print(f"  Raw response: {eval_result.get('raw', '')[:200]}...")
        else:
            winner = eval_result.get("winner", "Unknown")
            scores = eval_result.get("scores", {})
            score_a = scores.get("A", {}).get("total_score", "N/A")
            score_b = scores.get("B", {}).get("total_score", "N/A")

            winner_label = "Challenger (Haakon AI MCP)" if winner == "A" else "Incumbent (Zero-
Shot)" if winner == "B" else "Tie"

            print("  ================ MATCH RESULT ================")
            print(f"  Scores            : Haakon MCP (A): {score_a}/100 | Zero-Shot (B):
{score_b}/100")
            print(f"  Verdict Winner    : {winner_label}")
            print(f"  Word Counts       : A={len(out_a.split())} words | B={len(out_b.split())}
words")
            print(f"  Latency           : A={cog_res['latency']}s | B={zs_res['latency']}s")
            print(f"  Rationale         :\n{eval_result.get('rationale')}")
            print("  ============================================\n")

if __name__ == "__main__":
    asyncio.run(run_gemini_only_matrix())
```