QUANTUM OPTIMIZATION → APPLIED EFFICIENCY

Quantum optimization methods, and where they already run

Eight quantum-computing mechanisms, what each buys you over the classical approach, and a straight comparison of where Token Architect (the shipped router) and Subtraction Architect (the studio's wider Quantum Subtraction Theory research) apply the same idea — and where Subtraction Architect's own implementation goes further.

Superposition

A qubit holds $|0\rangle$ and $|1\rangle$ at once — many candidate states explored in parallel before collapse.

Entanglement

Correlated qubits: resolving one instantly constrains the other — shared state instead of duplicated state.

Tunneling

Annealing bypasses high-energy barriers to settle directly into the lowest-energy (optimal) state.

```
> EFFICIENCY_PROTOCOL: ACTIVE | MAXIMIZE_LOGICAL_DEPTH | MINIMIZE_TOKEN_BLOAT
```

QCP-1 BENCHMARK

mean token reduction, 30 runs

79.6% SA

CONCEPT SURVIVAL

critical concepts retained

100%

LANE 1 RESOLUTION

of 200-request sample, free

78%

COMMIT FLOOR

confidence gate, SA only

0.45

1 · Encoding — how classical data enters the system

Basis encoding spends one qubit per bit — accurate but wasteful. Amplitude encoding spends $O(\log_2 N)$ qubits for N features by folding information into shared coefficients instead of one slot per item. Phase/angle encoding maps continuous values onto rotation angles.

Encoding	Token Architect (shipped)	Subtraction Architect (studio)
Basis	Full-prompt passthrough on cache misses — one token per fact, no compression	QCP-1's baseline pass before subtraction — same cost, used only to establish the floor
Amplitude	Prompt caching — repeat requests served from a compressed hash, not re-sent	QCP-1 core protocol — 79.6% mean token reduction, 100% critical-concept survival
Phase / angle	Not implemented — TA	Delta-only re-prompts in

	reports savings, doesn't delta-encode requests	the pitch/playbook workflow — resend only what changed

2 · Grover's search — quadratic retrieval

An oracle plus a diffusion step turns an O(N) unstructured scan into O(√N) by amplifying the target's amplitude each pass. Applied to text: don't scan full context linearly — narrow candidates in rounds so only load-bearing tokens reach the output.

Token Architect

Cache-key lookup only — exact or near-exact match, no ranked re-scan of context.

Subtraction Architect

Signal-score ranking in the live feed — every pasted listing is scored and re-ranked against the running set, not matched once and forgotten. Closer to oracle+diffusion than a flat cache hit.

3 · Deutsch-Jozsa — one query resolves a whole class

Reversible gates ($U^{\dagger}=U^{-1}$) never destroy intermediate state, and Deutsch-Jozsa answers constant-vs-balanced in one query instead of

iterating 2^(N-1)+1 classical cases. Applied to reasoning: keep intermediate drafts reversible, and design single queries that resolve a whole category at once.

Token Architect

Model-tier routing decision is a single classification call per request — resolves which model in one shot.

Subtraction Architect

Same one-query pattern, plus the reversible half: nothing commits below the 0.45 confidence floor, so a wrong single query never becomes a wrong committed state.

4 · HHL & quantum annealing — solving, not just searching

HHL (gate-model)

Solves linear systems Ax=b via phase estimation and inverted-eigenvalue extraction. Classical O(N) → Quantum O(log N).

Quantum annealing

Formulates a problem as QUBO/Ising and lets couplers settle qubits into the ground state — the global minimum, found by tunneling past local ones.

Token Architect

Subtraction Architect

Token Architect

Cheapest-competent-model selection is a static cost/quality lookup — not a search over a solution landscape.

Subtraction Architect

$/hr-signal ranking across many candidate listings is a small optimization landscape — closer in shape to annealing toward a ground-state top pick than TA's fixed lookup table.

5 · VQE — the hybrid iterate-and-measure loop

Define a Hamiltonian, run a parameterized trial on the QPU, feed the energy to a classical optimizer, adjust, repeat until convergence. The efficiency-system analogue: draft, measure quality, adjust, and only pay the irreversible "write" cost once you converge.

Token Architect

No iterate-and-refine loop — a request is routed once, answered once. Savings come from routing choice, not iterative compression.

Subtraction Architect

QCP-1 v2 is exactly this loop — compress, score concept-survival, adjust, re-run — validated over 30 runs to converge at 79.6% mean reduction with no critical-concept loss. TA has no equivalent.

6 · Quantum amplitude estimation — tail-risk without full resampling

Classical Monte Carlo needs exponentially many draws to characterize a distribution's tail. QAE encodes the probability directly as an amplitude for a quadratic speedup — fewer samples for the same tail-risk precision.

Token Architect

Savings dashboard reports realized cost after the fact — a measurement, not an estimator.

Subtraction Architect

Confidence scores (conf 88%, 94%…) are estimated per-listing before commit — an amplitude-style estimate of quality, not a full resample of every source.

7 · QSVM / QNN — feature maps for classification

Quantum feature maps project classical data into a high-dimensional Hilbert space so a hyperplane separates otherwise-tangled classes; QNNs use parameterized circuits as trainable layers.

Token Architect

Model routing is rule-based tiers, not a learned separator

Subtraction Architect

Not yet implemented on either side — category

over request features.

keyword-matching is linear, not a learned feature map. Open gap, not a crossover.

8 · The routing node — three lanes, cheapest wins

The NISQ-era answer isn't "quantum-only" — a router sends each task to the cheapest lane that can resolve it, escalating only on ambiguity. Both products use this shape; only one has shipped the reversible-draft gate alongside it.

Lane 1 · Classical

Keyword match / cheapest model tier — always runs, free, resolves the majority.

Lane 2 · Local check

No network call — flags escalation only when confidence is low or signals conflict.

Lane 3 · Full reasoning

Real model call — reserved for genuinely ambiguous cases, skipped entirely otherwise.

	Token Architect	Subtraction Architect
Lanes 1–2	Static rules: model tier + cache lookup	`lane1_classify()` / `lane2_needs_escalation()` — 78% of a 200-request

		sample resolves here, free, ~4ms median
Lane 3	Cheapest capable third-party model	Direct Claude call, 500-char cap, only on the 22% that escalate
Commit gate	None documented — routed requests are answered directly	0.45 confidence floor — low-confidence extractions held in draft, never written or broadcast

Crossover summary

Five of eight mechanisms are stronger in Subtraction Architect's own implementation than in the shipped Token Architect product; one is an open gap on both sides.

Method	Token Architect	Subtraction Architect	Verdict
Amplitude encoding	Prompt caching	QCP-1, 79.6% reduction	SA better
Grover's search	Exact cache match	Signal-score re-ranking	SA better
Deutsch-Jozsa	Single-shot routing call	Same, plus reversible-draft gate	SA better
Annealing / HHL	Static cost lookup	Live $/hr-signal ranking	SA better
VQE loop	None	QCP-1 v2 iterate-and-measure	SA only
QAE	Post-hoc savings report	Per-item confidence estimate	SA better
QSVM / QNN	Rule-based tiers	Keyword match (linear)	Open gap

QSVM / QNN	Rule-based tiers	Keyword match (linear)	Open gap
Hybrid routing	2-lane, no commit gate	3-lane + confidence floor	SA better

Read honestly: Subtraction Architect's edge comes from being the research vehicle these mechanisms were built for first — Token Architect is the productized, customer-facing subset. The QSVM/QNN gap is real on both sides and worth flagging before it's promised to anyone.