AI Deliberation & Excel

=SECONDOPINION()

Called from the assistant you already use in Excel. Three frontier models argue — you see where they split.

A WORKFLOW GUIDE

QUORUM.DOG

What is in here

Sections 03 to 05 are the argument. 06 and 07 are what you install and say. 08 to 11 are the work itself, with real exhibits. 12 and 13 are wiring. If you are short of time, read 05 and 06.

03

What Quorum is, in one page.

Quorum is a deliberation platform. One question goes to several frontier models from different labs at once. They answer independently, critique each other, and are judged — and what comes back includes **where they disagreed**.

If you have only ever used one assistant, the distinction that matters is this: asking the same model a second time gives you the same reasoning again, more politely. Independent models fail in different places. That is the whole mechanism.

A SEAT

One frontier model on the panel. Three seats, three independent lines of reasoning. Quorum represents them as shapes rather than naming vendors.

A MODE

The configuration of a deliberation — which engines sit, how many rounds, how deep. Not a knob on a single model: a repeatable decision setup you select.

A RECEIPT

The record of a completed deliberation: which model sat in each seat, judge scores, difficulty, cost, latency. §11.

WHERE IT RUNS

Quorum's own surfaces, an OpenAI-compatible REST API, and a hosted MCP server — which is how it reaches Excel.

THE CLAIM, STATED NARROWLY

Quorum does not claim a single model is bad. It claims that a single model, however capable, **cannot be its own second opinion** — it has one way of being wrong, and it applies that way consistently. Panels exist for the questions where that matters.

AND WHERE IT DOES NOT APPLY

Most work is execution, not judgement. Quorum is deliberately wrong for lookups, syntax, formatting and arithmetic — and its own `estimate` call will tell you so before you spend anything (§13).

04

=SECONDOPINION()

Your assistant is good. Give it somewhere to go when it isn't sure.

Claude for Excel writes the formula, reconciles the tab, drafts the summary and explains what it did. On most of what you ask it, that is the whole job.

Then there are the questions where it is **producing** an answer rather than knowing one — whether the growth assumption still holds, which of four scenarios you should defend. It answers those too, fluently and in a single voice, because a single voice is what one model has.

One point worth being precise about: **your assistant is not one of the panel.** It stays where it is — holding your workbook, your context and your standing instruction — and calls in a panel of independent models, none of which is the one you were just talking to. That independence is the mechanism. A model asked twice is the same opinion in a second coat.

Nothing else about how you work changes. You do not open a second tool. Your assistant simply stops guessing where guessing is expensive, because you told it once which questions those are.

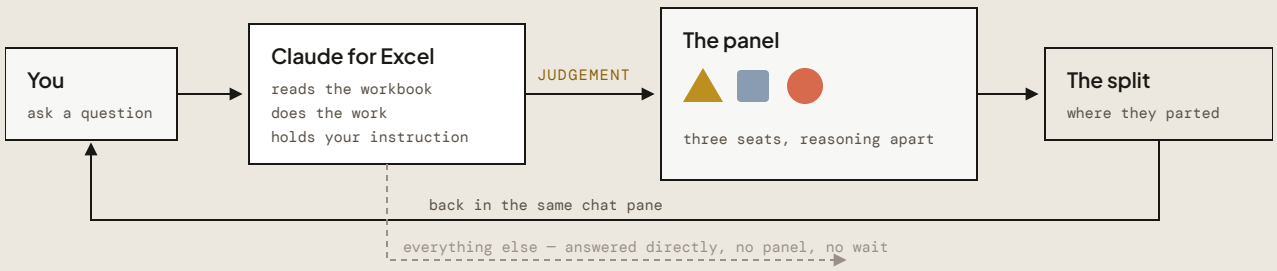

Exhibit 01 — Where the panel enters a turn your assistant is already handling. Only the judgement branch convenes it.

BEFORE, NOT AFTER

The strongest objection to your work arrives **while you can still act on it**.

THE REAL CHOICE

When options are close, you learn **which risk you are accepting**.

BEFORE THE REBUILD

A confident diagnosis gets a second read **before you spend the afternoon**.

AGREEMENT COUNTS

Three models landing together is **information too**.

05

The five moments it escalates on.

These are not one job or one industry. They recur in anything built in a spreadsheet — a model, a budget, a forecast, a comparison someone will be asked to defend.

01	**An assumption nobody has revisited**	A growth rate, a churn number, a ramp — typed in during a meeting, compounded across eighteen months, never argued with since. The panel attacks it. You get the objection nobody in the room raised. **§08 works this one end to end.**
02	**A tie the numbers cannot break**	Three suppliers, two hiring plans, four scenarios — all within a few percent. A single model picks one and sounds certain. A panel tells you the choice is closer than you want it to be, and which risk you accept by picking. **§10.**
03	**A model that looks wrong**	An interconnected model produces an output that smells off, and your assistant proposes a confident fix. It may be right. It may also cost you an hour rebuilding around the wrong lead. The panel argues the diagnosis before you act on it.
04	**What you did not think to consider**	Before a forecast goes out, ask several models what is missing rather than asking one to confirm what is there. Disagreement between them is the useful signal.
05	**Someone else's workbook**	Inheriting a model, or checking a vendor's, you have no intuition for what is load-bearing. A panel argues about it from several directions at once. **§09.**

THE COMMON SHAPE

Every one is a moment where the work is done and the **judgement** is what remains. That is the line your standing instruction draws in the next section.

AND WHAT IT SHOULD NEVER ESCALATE

Lookups, formula syntax, formatting, unit conversion, arithmetic your assistant can simply do. Quorum's own classifier agrees: it returns `likely_hurts` when the answer is a verbatim artifact a synthesis pass would only rewrite.

06

Teach it once.

Give your assistant a standing rule for when to convene the panel, and the decision stops being yours to remember. This is the whole setup.

Put it wherever your assistant keeps persistent guidance — a skill, project or workspace instruction if you have one, otherwise at the top of the session you are about to work in. The wording is the same either way.

STANDING INSTRUCTION — PASTE AS WRITTEN, THEN EDIT THE LAST LINE

```
You have a Quorum connector. It convenes a panel of frontier models
and reports where they disagree.

Use it on judgement, not on work you can do yourself.

Convene the panel when:
  - I am choosing between options that land within a few percent
    of each other;
  - I ask you to check an assumption, a driver or a scenario that
    I will have to defend to someone;
  - you have proposed a fix for something that looks wrong and you
    are not certain the diagnosis is right;
  - I ask what I have missed.

When you do:
  - put the actual figures in the question, not a description of
    them;
  - report where the seats disagreed and do not reconcile it for
    me;
  - tell me the strongest argument against the option I am
    leaning toward.

Do not convene it for lookups, formula syntax, formatting,
conversions or arithmetic. Answer those yourself.

Escalate without asking me first.   ← or "ask me first"
```

BEST PRACTICE

Install the standing instruction, and let it escalate on its own. The value of this setup is that it catches the moment you did not notice was a judgement call — and if you have to remember to invoke it, you will remember exactly when you least need it and never when you do. Approving each escalation is the right setting for your first week, and the wrong one after that.

THE ONE LINE THAT MATTERS MOST

"Put the actual figures in the question." The panel argues about what it is given. An assistant left to summarise will write *a growth assumption that may be aggressive*; you want *18% compounding monthly for eighteen months, set in March, never revisited*. Same escalation, an entirely different argument comes back.

07

Steering it in the moment.

The standing rule handles the moments you did not spot. These are for the moments you did — when you already know the question is worth an argument and you want one now.

"Before this goes out — put the growth assumption to the panel and tell me the strongest argument against it."

Forces an attack, not a review

"These two are within noise of each other. Convene the panel and tell me which risk I am accepting with each."

Reframes a tie as a risk choice

"You have given me a fix. Get a second opinion on the diagnosis before I rebuild anything."

Buys back the hour you were about to spend

"Ask the panel what I have not considered in this forecast."

Open-ended on purpose

"Where did the seats disagree? Do not reconcile it — show me the split."

The follow-up worth making a habit

"Send it the actual numbers from rows 12 to 40, not a summary."

When precision is the whole question

BEST PRACTICE

Use both. The standing instruction is the safety net; these are the times you reach for it deliberately. If you only ever do one, do the standing instruction — the escalations you would have typed by hand are the ones you already knew were hard.

WHAT MAKES AN ESCALATION LAND

Name the figures. Say what decision hangs on it. Say what you are leaning toward — a panel given a position to attack argues far harder than a panel asked an open question. And ask for the split explicitly; a tidy synthesis is the least useful thing a panel can hand you.

08

ASSUMPTION → MODEL → SCENARIO

One escalation, end to end.

An eighteen-month revenue build. The growth rate in row 2 was typed during a meeting in March and has compounded quietly ever since.

IN THE WORKBOOK

	FY1	FY2	FY3	FY4
Revenue	100.0	118.0	139.2	164.3
Growth	—	18.0%	18.0%	18.0%
EBITDA	21.0	26.5	33.4	42.1
Cash	31.0	38.2	47.9	60.4

ILLUSTRATIVE FIGURES

WHAT YOUR ASSISTANT SAYS

"The 18% growth assumption is applied consistently across all four years and the model calculates correctly."

Both true. Neither is the question. Consistency is not defensibility, and the assistant has no way to know the number has never been argued with.

WHAT THE PANEL RETURNS

- ▲ **Sustainable, with a condition.** 18% holds if pricing power holds; the build assumes volume growth and price growth are independent, and in this market they are not.
- ■ **Aggressive after FY2.** A flat rate across four years is the tell. Real cohorts decay; a constant compounding rate usually means nobody re-derived it.
- ● **Cannot be assessed as given.** Without churn and pricing split out, 18% is an output presented as an input.

The seats did not vote. Two questioned the rate, one questioned whether it is an assumption at all — and that third objection is the one that changes the model.

ELAPSED

A panel takes a median of **28.5 seconds**, 88.2 at the ninetieth percentile, on the branch you told it to escalate.

P50 28.5 S · P90 88.2 S · N=3,845 · 2026-08-20

Exhibit 02 — An assumption, and what three seats said about it. Seat positions are illustrative of the pattern, not a transcript of a specific call.

09

ASSUMPTION → **MODEL** → SCENARIO

Interrogating the model itself.

Everything so far starts with a question you already knew to ask. The harder case is a workbook you have **not** read — inherited from someone who left, received from a vendor, or built by you eleven months ago.

Your assistant can read the whole thing. What it cannot do is disagree with itself about which parts matter. So run the reading and the argument as separate steps.

The sequence below is four turns in one chat pane. The first three are your assistant working alone and cost nothing. Only the fourth convenes a panel — and it convenes it on the *shortlist*, not the workbook.

01	**"Which assumptions actually drive this model?"**	The assistant reads the dependency chain. Most workbooks turn out to rest on four or five numbers.
02	**"Rank them by how much the output moves."**	Sensitivity, which a spreadsheet is genuinely good at. Still no panel — this is arithmetic.
03	**"Which of those are least defensible?"**	Now the assistant is making a judgement call in one voice. Useful, and exactly the point at which a second opinion is worth having.
04	**"Put that shortlist to the panel."**	Three independent reads on the four numbers that carry the model. This is the call worth paying for — and it is one call, not forty.

WHY THE ORDER MATTERS

Escalating the *whole workbook* is the expensive mistake: you pay for a panel to do reading your assistant does for free, and the question arrives so broad that three models produce three summaries instead of three positions. **Narrow first, then deliberate.**

WHAT YOU ARE LEFT HOLDING

Not "the model is fine." A named shortlist of the assumptions doing the work, and for each one whether independent reasoners agreed it was defensible. That is a governance artefact, and it did not exist before you asked.

10

ASSUMPTION → MODEL → SCENARIO

Which scenario survives being attacked.

Three cases, built and agreed. The usual question is *which one do we present*. The better question is which one holds up when somebody argues with the assumptions underneath it.

THE SCENARIO GRID

	BULL	BASE	BEAR
Revenue	128	112	94
EBITDA	31	21	9
Cash	47	31	14

ILLUSTRATIVE FIGURES

WHAT TO ASK, IN ORDER

1. "What has to be true for Bear?" — surfaces the assumptions, not the outputs.
2. "Attack each case. Where is each one weakest?"
3. "Which survives reasonable movement in the assumptions that matter?"

WHERE THE PANEL ACTUALLY SPLIT

- ▲ Base is well constructed and the arithmetic ties. **Bear is not a scenario** — it is Base with the growth rate reduced, so it inherits every assumption Base makes.
- ■ Agrees Base is sound. **Disagrees that Bear is weak**: a downside case that shares Base's structure is the point, because it isolates one variable.
- ● Both are arguing about construction. **The real exposure is that all three cases assume the same cost base**, so none of them is a downside case at all.

The panel agreed on Base and split on Bear — and the split is the finding. A single model would have picked one of these three readings and sounded certain about it.

THE HABIT WORTH FORMING

Ask for the split explicitly, and do not let it be reconciled. A tidy consensus is the least useful thing a panel can hand you.

Exhibit 03 — A scenario grid, and where the panel actually split. Positions are illustrative of the pattern, not a transcript of a specific call.

11

The receipt, and what you can do with it.

Every deliberation leaves a record. It is the difference between “the AI said” and a decision you can hand to somebody else six months later.

WHAT GET_RECEIPT RETURNS

request_id	rq_8f2c41a9
mode	quorum-standard
difficulty	0.78
seat 1	judge score, model
seat 2	judge score, model
seat 3	judge score, model
cost	billed, per call
latency	31.2 s

SHAPE OF THE RESPONSE · ILLUSTRATIVE VALUES

The judge scores are the part people underuse. They tell you whether the seats **actually disagreed** or converged immediately — which is how you learn that a question did not need a panel at all.

Quorum names seats by shape rather than by vendor on anything shareable, so a receipt can leave your organisation without carrying somebody else’s trademark.

IT EXPORTS

From Quorum, a completed deliberation renders as a full transcript PDF, a carousel, a card or a short video — the seats, their positions and the split, laid out.

IT TRAVELS

That artefact is what goes into the memo, the IC pack or the audit file. The argument travels with the number instead of evaporating in a chat pane.

IT TUNES

Receipts that keep showing immediate convergence mean your threshold is too low — you are paying for agreement. §13.

WHERE THIS IS GOING, STATED AS DIRECTION AND NOT AS A FEATURE

A receipt records one deliberation. What it does not yet do is accumulate: assumption, disagreement, decision taken, and eventually what actually happened. An organisation that kept that would be able to ask what its own past judgements were worth. Quorum does not do this today — there is no store, no outcome linkage and no retrieval — and it is named here as the direction the receipt makes possible, not as something you can switch on.

Exhibit 04 — A receipt.

12 Connect it.

Quorum runs as a hosted MCP server. Add it in Claude for Excel as a custom connector: the URL below, and your key as a static bearer token. No OAuth, no app registration, no redirect URL.

No click-by-click screens here on purpose — connector menus move, and a stale set of menu names is worse than none. The outcome you want is a custom connector on that URL carrying your key in the Authorization header.

SERVER URL · AND A CHECK YOU CAN RUN BEFORE INVOLVING EXCEL AT ALL

```
https://www.quorum.dog/mcp

curl -sS -X POST https://www.quorum.dog/mcp \
  -H "Content-Type: application/json" \
  -H "Authorization: Bearer qk_..." \
  -d '{"jsonrpc":"2.0","id":1,"method":"tools/list"}'
```

WHAT APPEARS IN THE PANE

TOOL	COST	WHAT IT DOES
`estimate`	Free	Classifies and prices a question without running it.
`list_modes`	Free	Which panels your key is allowed to convene.
`deliberate`	Paid	Convenes the panel on the question. The one that costs, and the one you wait for.
`get_receipt`	Free	For a finished call: which model sat in each seat, judge scores, cost and latency.

TWO THINGS THAT LOOK LIKE FAULTS AND ARE NOT

A **GET** to that URL returns 405 — the server is Streamable HTTP, POST only, and that is the correct response. And the connection is stateless, so there is no session to keep warm.

THE SAME KEY, ELSEWHERE

It also works against the OpenAI-compatible REST API at `quorum.dog/v1`. Two further tools, `certify_mode` and `run_test` are also exposed on the hosted server, so six tools appear in all. They drive Quorum's mode testing and certification rather than day-to-day work, and an analyst can ignore them.

13

When the workbook runs itself.

Everything so far assumes a person is at the keyboard, and for one analyst working through one model that is the right assumption — when the question genuinely matters, the cost of the call is not what you are weighing.

Automation changes the arithmetic. A workbook that runs a hundred reconciliations overnight, or a process that fires the same check across every line of a portfolio, is making that decision a hundred times without anyone present to judge which ones deserved a panel.

That is what **estimate** is for. It classifies and prices a question without running it, free on every call, so an automated process can decide for itself which items are worth escalating and which are routine.

The threshold is yours and belongs in your own logic. Quorum's job is to hand you the facts that branch needs, at no cost.

WHAT ESTIMATE HANDS BACK

FIELD	WHAT YOU BRANCH ON
`difficulty_score`	0 to 1 — the main dial. Set your own threshold from your own measured distribution, then move it with evidence.
`estimated_price_usd`	What the real call would cost, so a budget ceiling can sit in front of it.
`billed_usd`	Returns 0 on every estimate call, by design — not as a trial allowance.

FAIL OPEN, ALWAYS

If the classification call fails, answer with your single model rather than erroring or defaulting to the expensive path. A router that breaks your process when one endpoint has a bad minute is worse than no router.

TUNE IT WITH RECEIPTS

If receipts on escalated items keep showing the panel converging with no dissent, raise the threshold — you are paying for agreement. If people keep pushing back on single-model answers, lower it.

Start with the standing instruction and one workbook you already care about. You will know inside a week which of the five moments is yours.

WHERE TO GO NEXT

The escalation router guide at `quorum.dog/docs` covers the code version of this decision in full, including the shape of the branch and how to pick a starting threshold.