First Principles

Facilitator guide

About this activity: the team breaks the problem it is stuck on into single claims, tags each one, and strips away everything it cannot show is true. What is left is rebuilt into a new problem — usually different from the one the team walked in with, and open to options the old wording was hiding. You are not here to solve the problem. You are here to hold one test: if the team cannot show a claim is true, it comes off the board.

What is a claim? A claim is any single statement buried inside the problem — a fact, an assumption, an old rule, or a solution the team has already half-chosen. Most of what a team calls "the problem" is really a pile of claims it has never separated. This activity separates them.

· WHEN DO I USE THIS?

The right moment

Picture this: the team has been stuck on the same problem for weeks. Every new idea sounds like a version of the last one. The team has "tried everything" and nothing has moved. Often the problem statement already names a solution — "we need a better dashboard", "we need more people". That is the moment for this activity.

USE IT WHEN

You have one real problem the team has been stuck on for a while, without a breakthrough. Every new idea sounds like the last one. Or the problem statement already names a solution — "we need a [tool / team / process]".

DO NOT USE IT WHEN

The problem is brand new and nobody has worked it yet — let the team try first. Or the problem is simple and well understood, with one obvious fix — there is nothing buried to strip.

The one test before you start: has the team been circling this problem without a breakthrough? If yes, this is the tool. If it is fresh, or already clear, pick a different one.

· WHAT WILL THE TEAM LEAVE WITH?

What you'll leave with

WHAT YOU GET

The problem stripped to what is actually true — assumptions, old rules and hidden solutions removed. A rebuilt problem statement, often visibly different from the one you started with. A fresh set of options the old wording was hiding, with one owned next step.

WHY IT MATTERS

The team stops pouring effort into the wrong problem. A problem that felt stuck for months dissolves on the board, and a realer one appears in its place — one the team can actually move.

© Witmer

PART 1 BEFORE THE ACTIVITY

get ready

1 · WHAT IS MY JOB HERE?

Your job

Run the activity so the team separates what is true from what it has only assumed or pre-decided. Do not solve the problem yourself. Do not defend the problem. Your job is to:

- write the team's real problem at the top of the board
- keep the team breaking it into single claims — one per card
- hold the truth test as each claim is tagged
- read out the surviving truths, and write the rebuilt problem
- keep time, and protect the last step — the rebuild

> **The one thing the team should learn:** most of what it called "the problem" was an assumption, an old rule, or a solution it had already chosen. Strip those away, and a more real problem appears.

2 · WHO DOES WHAT?

Roles

Who	What they do
You (facilitator)	Run the flow. Write the problem on the board. Hold the truth test. Draw Test-prompt cards when the team cannot agree. Break ties and make the final call. Read out the truths and write the rebuilt problem. Stay neutral on the answer.
The team (3–6)	Break the problem into single claims. Tag each claim honestly. Test the contested ones instead of debating them. Help rebuild the problem from the truths.

> **Say this to the team before you start (in your own words):** "We'll take our problem apart, sort the pieces into what's true and what we've only assumed or pre-decided, throw out everything that isn't true, and rebuild. Don't defend the problem — test it."

3 · HOW DO I START?

Opening script — 2 minutes

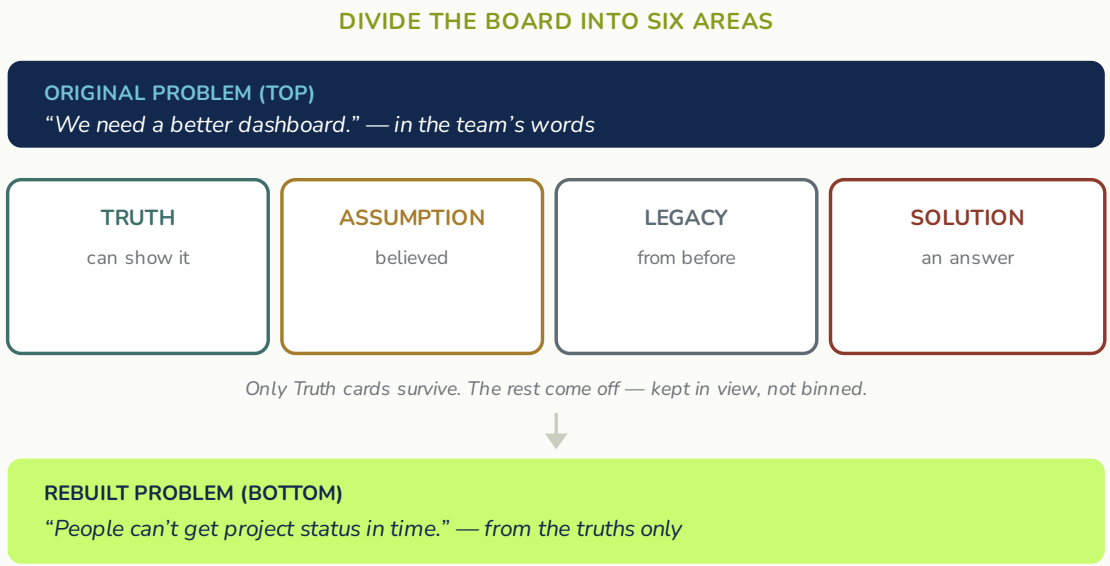
Say this in your own words before you start. Do not read the app page out loud.

> SAY
>
> "We have been stuck on this problem for a while. Today we will not try to solve it harder. We will take it apart. Each of us writes the separate statements buried inside it — one per card. Then we sort every card into one of four things: something we can show is true, something we only assume, an old rule carried over from before, or a solution we have already chosen. We throw out everything that is not true, and rebuild the problem from what is left. Don't defend the problem — test it."

4 · HOW DO I PREPARE?

© Witmer

Setup and materials



Set the board out before the team arrives: the original problem at the top, the four tag areas in the middle, the rebuilt problem at the foot. What survived and what was stripped stay visible side by side.

© Witmer

Material	What it is, and when you use it
Claim cards (blank, write-on)	The team writes one statement from the problem per card — a claim, assumption, old rule or solution. These become the cards you tag and strip. Step 2.
Four tag markers	Truth, Assumption, Legacy, Solution. Each claim gets exactly one. Only Truths survive. Step 3.
Test-prompt cards	Kept face down beside you. Used when the team cannot agree how to tag a claim: draw one, read it, then you make the call. Step 3.
Starter-problem cards (optional)	A few stuck-problem examples. Use only if the team has no real problem of its own to bring.
The board	Original problem at the top, four tag areas in the middle, rebuilt problem at the bottom.
Capture sheet	One sheet. You write the surviving truths, the rebuilt problem, and the owned next step as the activity runs.

> **Legacy is the tag teams find hardest.** It means a rule, decision or convention carried over from before — not a fact. “It must be one screen”, “only managers can approve”, “we always use this template”. If it could have been decided differently, it is Legacy, not a Truth.

Words to know: a **claim** is one statement pulled out of the problem. A **Truth** is a claim the team can show is true. Everything else is an **Assumption** (believed, untested), **Legacy** (a rule from before), or a **Solution** (an answer hiding in the problem).

5 · HOW DO I FRAME THE PROBLEM?

Get the problem right first

Write the team’s real, stuck problem at the top of the board, in its own words. If the problem already names a solution — “we need a better dashboard” — that is fine. The activity is built to break exactly that kind of statement apart.

> **Rule:** write it as the team actually says it. Do not clean it up or make it neutral. The whole point is to see what breaks apart when you test it.

© Witmer

PART 2 DURING THE ACTIVITY

run it

Before you begin — 30-second check

- ✓ The team's real, stuck problem is written at the top of the board, in its own words
- ✓ The board is divided into six areas — original, the four tags, rebuilt
- ✓ The Claim cards, tag markers, Test-prompt cards, capture sheet and timer are ready
- ✓ The team understands the one rule: if you can't show it's true, it comes off the board
- ✓ Everyone is ready to write claims on their own first, before any discussion

6 · WHAT DO I DO NOW?

Run the activity

Set up › Break into claims › Tag every claim › Keep the truths › Rebuild › Commit

1. Set up

2 min

Appoint yourself facilitator. Write the team's real, stuck problem at the top of the board, in its own words. Solo: write your problem at the top and begin.

2. Break the problem into claims

8–10 min

Each person, working alone, breaks the problem into smaller statements — one per Claim card. Look for facts, assumptions, rules everyone accepts, and proposed solutions already hiding inside the problem.

> **Don't let them debate yet** — just get every piece onto its own card. When everyone is finished, place all Claim cards on the board. **Keep moving; the first version is usually enough.**

3. Tag every claim

12–15 min

Work through one card at a time. Read it aloud and ask the team to tag it: **Truth** (can show it), **Assumption** (believed, untested), **Legacy** (a rule from before), or **Solution** (an answer, not the problem). Quick show of hands. If the team cannot agree, draw a Test-prompt card, read it, then you make the final call.

> **Hold the line:** if the team can't show a claim is true, it isn't a Truth.

4. Keep only the truths

4–6 min

Move the Truth cards into the Truth area. Set the rest aside on the board — kept in view, not binned. Read the survivors out. Read the count out and **pause** — "three of six were true." That difference is the reveal.

> **Say something like:** "Notice how much disappeared. We're no longer solving the problem we walked in with."

© Witmer

5. Rebuild the problem
8–10 min

Using only the truths, write the problem again at the bottom — the simplest single statement that connects the surviving truths, not a list of them. Then compare with the original at the top.

> **Ask:** Which words disappeared? Are we solving the same problem we started with? Does this open up new options? Most teams notice the solution they started with has disappeared.

6. Commit
4–6 min

As a team, decide which new problem statement — or which new option it revealed — you will act on first. Assign an owner and a date. Write down the question to reuse: **"What's actually true here?"**

These times are indicative. A full team run is about 45–60 minutes; solo, about 15–25 minutes. The time you need will vary with the problem, the size of the team, and the time available. The step that overruns is Tag (step 3). The step that gets squeezed is Rebuild (step 5) — protect it.

GOLDEN RULE

If we can't show it's true, it comes off the board. A claim counts as a Truth only if the team can point to how it is known. Everything else is an assumption, an old rule, or a solution — and gets stripped. You don't need absolute proof: "true" means the team can show how it knows.

One more thing: the reveal is not the pile of stripped cards — it is the new problem that replaces the old one. If time runs short, cut tagging, never the rebuild.

While you run — when to step in, when to wait · a principle, not a step

STEP IN

A claim is tagged Truth but no one can show how it's known. → Re-tag it Assumption.

The tagging stalls into a debate. → Draw a Test-prompt card and make the call.

The "problem" is really one big solution ("we need X"). → Break it into the claims underneath it.

WAIT

The team is uncomfortable that its problem fell apart. → That's the work; let it sit.

The rebuilt problem is bigger or harder than the original. → That's often the real one; don't shrink it.

People go quiet when the truths are read out. → Let it land.

© Witmer

PART 3 AFTER THE ACTIVITY

use it again

7 · HOW DO I FINISH?

Closing script — 2 minutes

SAY

"Today we saw something. We were not short of effort — we were solving the wrong problem. Most of what we called 'the problem' was an assumption, an old rule, or a solution we had already chosen. When we stripped those away, a more real problem appeared. Next time you hear 'we need [some solution]', stop and ask: what's actually true here? You are leaving with a rebuilt problem and one new option to act on. Try it this week."

Before people leave — the finish check

- ✓ The team's original problem is written at the top of the board
- ✓ Every claim was tagged Truth, Assumption, Legacy or Solution
- ✓ Only the Truth cards fed the rebuild
- ✓ A rebuilt problem is written at the bottom, and compared with the original
- ✓ One new problem or option is chosen, with an owner named
- ✓ A photo of the board has been taken and shared

© Witmer

8 · DID WE GET A GOOD RESULT?

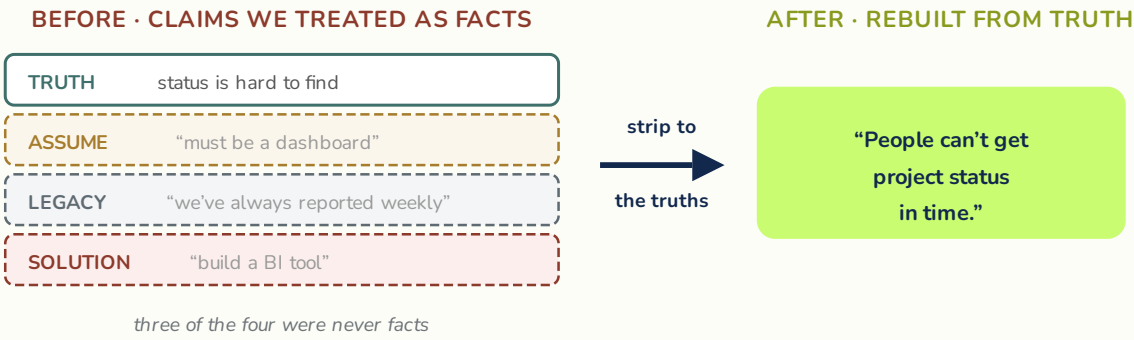
Check your run

GOOD RUN

Most claims were stripped. The rebuilt problem is visibly different from the original — often bigger, and open to options the old wording hid. The team commits one new option, with an owner.

HOLLOW RUN

Everything got tagged “true”. Nothing was stripped, and the rebuilt problem matches the original word for word. The truth test was not held.



three of the four were never facts

The dashboard was never the problem — it was an assumption wearing a fact’s clothes.

Solve the rebuilt problem, not the one you inherited.

9 · HOW DO I MAKE THIS A HABIT?

Build the habit — 30 seconds

THE HABIT

“What’s actually true here?”

A team that has run this once tends to catch a solution dressed up as a problem: someone says “we need X”, and another asks “what’s the problem X is for?” That catch, in normal work, is the lasting change — not the session itself.

© Witmer

Appendix · Reference

You should not need this while you run the activity. Read it before, or use it if something goes wrong.

10 · WHAT IF SOMETHING GOES WRONG?

Recovery guide

What happens	What to do
Everything gets tagged Truth	The single biggest failure — nothing strips, nothing rebuilds. Take the most confident claim and ask “how would we show this to someone who doubted it?” If they can’t, it’s an Assumption.
The team debates who’s right	The job is to tag, not to win. If a claim is contested and can’t be shown, tag it Assumption and move on — don’t argue about it.
The rebuilt problem equals the original	Either the problem was already well-framed (rare), or claims were over-tagged as truths. Re-check the Truth pile against the test.
The team can’t agree on a tag	Draw a Test-prompt card, give it 60 seconds, then you make the call. If a claim is contested and can’t be shown, tag it Assumption.
A claim is part true, part assumption	It’s a compound claim — split it onto two cards and tag each. This is why one claim per card matters.
The whole problem is one big solution	Good catch — that’s the tool’s home ground. Break “we need X” into the claims underneath it, and tag those.
The rebuilt problem feels bigger or harder	Good — bigger often means you’ve reached the real problem. Don’t shrink it back; commit to one option that moves it.
Time runs out before the rebuild	Cut tagging short, never the rebuild. A rebuilt problem beats a fully-sorted board with no new problem.

© Witmer

11 · WHAT DO I WRITE DOWN?

The capture sheet — what you fill in

One sheet runs the whole activity. You write on it throughout (or a nominated scribe does). It carries the surviving truths, the rebuilt problem, and the owned next step — the reveal — on a single page the team keeps.

1 · THE PROBLEM THE TEAM BROUGHT — as first stated, before the activity

"We need a better dashboard."

2 · THE CLAIMS, SORTED

Claim	Tag
Managers can't see status quickly	Truth
Information arrives late	Truth
It must be one screen	Legacy
Build a dashboard	Solution

3 · THE REBUILT PROBLEM (from the truths only) — plus the owned next step

"People can't get project status in time." · **Next step:** trial a shared status feed — owner: Priya, by Friday.

Box 1 holds the original problem. Box 2 holds every claim and its tag — most turn out not to be Truths. Box 3 is the reveal: the rebuilt problem, from the truths only, and the one next step the team owns.

12 · WHAT DOES A GOOD RUN LOOK LIKE?

Worked example

The problem: a team writes "we need a better dashboard" at the top of the board. They break it into six claims and tag each one. Three hold up as Truths — managers can't see status quickly, information arrives late, teams report differently. Three are stripped: "people want more detail" (Assumption), "it must be one screen" (Legacy), and "build a dashboard" (Solution). Rebuilt from the three truths, the problem becomes "people can't get project status in time" — a different, bigger problem, and a dashboard is only one answer to it. New options the old wording hid: a status feed, a shared format, an earlier check-in.

13 · WHAT GOES WRONG MOST OFTEN?

Common mistakes

- **Tagging everything "true".** The single biggest failure — nothing strips, nothing rebuilds. Hold the test: can the team show how it knows? If not, it's an Assumption.
- **Debating who's right.** The job is to tag, not to win. If a claim is contested and can't be shown, tag it Assumption and move on.
- **Stopping at the strip.** "Look how much wasn't true" with no rebuild is half the tool. The rebuilt problem is the point — always reach step 5.
- **Rebuilding from stripped cards.** Only the Truth cards feed the rebuild. Nothing set aside sneaks back in.
- **Leaving with an insight, not a step.** End with one owned next step, not just a realisation.

© Witmer

14 · WHERE DOES THIS FIT?

Other tools

If instead…	Use this
the team treats one belief as a fixed fact that blocks it	**Assumption Breaker** — test whether that one “because” is really fixed.
the team needs fresh options, not a redefined problem	**Break the Frame** — see the problem through other people’s eyes.
the process around the problem has just grown too heavy	**Subtraction Game** — strip the process back by removing.

© Witmer