

The six stages of agent **HANDOVER**

- **Brief** you
- **Plan** together
- **Run** the agent
- **Check** together
- **Verify** you
- **Learn** together

Where the research came from

We went through what Google, Anthropic, Microsoft, LangChain, Asana and Notion have shipped in their agent products, alongside the studies published in 2025 and 2026.

One gap showed up again and again.

Products put their effort into the agent doing the job. Very little goes into how you hand the job over, and how you check what came back.

Agents are moving from demos into real work this year. What decides whether people keep using one is usually **trust, not capability.**

The cycle

Who is in charge changes at every stage.

You decide
At the start and at the end.

Together
Planning, checking, learning.

Who is in charge		
●	Brief	you
●	Plan	together
●	Run	the agent
●	Check	together
●	Verify	you
●	Learn	together

On its own
One stage out of six.

It comes round
What it learns shapes the next job.

Products invest almost everything in **Run**. That is the stage that gets demoed.

1: Brief

What you want, what it may spend,
how far it may go alone.

Say what good
looks like

Not how to do it.

Set a ceiling

Time and money.

Limits for this job

Time	20 minutes
Budget	\$4.00
At the limit	stop and show me
How far it may go alone: check with me first	
☐ ☐ ☒ ☐	

Say what happens
at the limit

Stop, or carry on
and tell you.

One setting per
job

Booking a room is
not signing a
contract.

This is the cheapest stage to add and the one most products skip entirely.

2: Plan

The agent drafts it. You change it before anything runs.

Editable, not just approvable

Fix one step without rejecting six.

Reorder and remove

Move a step, delete one it added.

The plan, not running yet

- 1 Pull the three supplier quotes last 12 months only · edit
- 2 Email each one for updated pricing
added by the agent · remove
- 3 Build the comparison, do not send it
changed by you

+ Add a step

Start

Mark who changed what

So the plan carries its own history.

Microsoft Research ships this

They call it co-planning.

With approval alone, the only way to fix one wrong step is to **reject the whole plan** and start again.

3: Run

Enough detail to follow, not so much that nobody looks.

A summary by default

One line on what it is doing now.

Detail when asked

Let people open one step, not read all of them.

Working, 4 minutes in

- ✓ Found 3 suppliers worth comparing
read 40 pages, kept 3 open
- › Comparing delivery times
two sources disagree, checking a third
- 3 Draft the comparison

Overload is a failure too

A log nobody reads is not transparency.

Name the snag

Say out loud when something does not add up.

Microsoft Research names the balance between transparency and overload as an **open challenge**, not a solved one.

4: Check

A few check-ins, agreed in advance.

Not every step

That is doing the job yourself.

Not only at the end

One early mistake ruins the whole run.

Updating 1,400 records

Step 5 of 12

☒ Checked with you at step 3

7 Next check at step 7

11 Then step 11

You can close this. I will wait at step 7.

81% preferred this

In a 2026 study of 48 people.

Say when the next check is

That is what lets someone walk away.

In the same study, the job also finished **13.5% faster** than confirming only at the end.

5: Verify

Come back with proof, not a log.

Checkable in a minute

Or nobody checks it.

Flag what you could not confirm

That is the most useful part.

Supplier review, finished

- ☒ The plan it followed
- ☒ 14 sources read, 9 quoted
- ☒ The comparison sheet
- ☐ 2 delivery times it could not confirm

Accept

See the detail

Google builds this in

It calls them artifacts.

Undo with a deadline

Say when it stops being reversible.

The strongest pattern we found, and it has barely left developer tools.

6: Learn

The stage nobody builds. It makes the next job quicker.

Save what worked

A good run becomes a repeatable one.

Keep it outside the chat

Or it goes when the window closes.

After a good run

Save this as a reusable plan?

6 steps, one changed by you, ran clean.

Save it

Change it first

No



Also remembered: direct flights only, under \$400

Microsoft Research builds this

They call it plan learning.

Delegation should build up

Not start from zero each time.

Without this stage, every job starts from scratch and the briefing never gets shorter.

Score the tools you use

- 1 Brief.** Can you set the goal, the budget and how far it goes alone?
- 2 Plan.** Can you change the plan, or only say yes to it?
- 3 Run.** Can you choose how much you see?
- 4 Check.** Are there check-ins, and do you know when?
- 5 Verify.** Does it come back with proof?
- 6 Learn.** Does a good run make the next one quicker?

Answering no four times out of six is normal today. Start with Brief and Verify.

Which stage is weakest where you work?

Comment with one word.

If you change one thing

Put a brief in front of the run

Why this one

One screen before the agent starts: the goal, the budget, and how far it may go alone. It is the cheapest of the six stages to build.

Brief and Verify are the two stages people notice missing. **Brief is the one you can ship this month.**

Go deeper

The whole pattern library

On the site

Seven families, forty-one patterns.
What each one prevents, and the
product or the paper it came from.
Free, and no sign-up.

thinkbigleaders.com/resources/agent-patterns

Or talk it through

Curious, and nowhere near a decision?
Good. Have a use case in mind? Also
good. Already building one, or want
agents inside your own product? Bring
what you have.



Thirty minutes with Think Big Leaders.

Scan the code to pick a time.