

AN ESTIMATOR'S GUIDE TO

Thinking With AI

How to use AI without losing the judgment that makes you good at your job.

BY DANIEL BERRY

Built by Berry

A Note Before You Start

This book is not about Copilot. It is not about ChatGPT, or any other AI tool by name. Those tools will change. Some will get better. Some will be replaced. A book written about any specific one of them would be obsolete before your next bid cycle.

This book is about you.

More precisely, it is about how an estimator should think when AI enters the workflow. Because the part that makes you valuable — your ability to read a spec, count what matters, catch what others miss, and price work that hasn't happened yet — is the part AI cannot replace. It is also, somewhat paradoxically, the part you need most when using AI.

If you take one thing from these pages, take this: AI is not a magic wand. It is not a junior estimator you can hand a stack of plans to and trust to come back with a bid. It is a tool that amplifies whatever you bring to it. If you bring clear thinking, sharp instructions, and a habit of verification, AI will make you measurably faster at work that used to take hours. If you bring vague questions and blind trust, AI will hand you confident garbage and let you submit it.

The chapters that follow are short on purpose. Read them in order the first time. After that, treat the book as a reference — the framework on the back page is the part you'll come back to most.

Welcome in. Let's get to work.

— *Daniel Berry*

Contents

01	What AI Is, and What It Isn't.....	3
02	The Estimator's Edge.....	5
03	The Five-Step Framework.....	7
04	Prompting Like an Estimator.....	9
05	Three Ways AI Earns Its Keep.....	12
06	Trust, but Verify.....	14
07	Building What You Use Twice.....	17

A P P E N D I C E S

A	The Five-Step Framework — One-Page Reference.....	19
B	Glossary.....	20

01 / What AI Is, and What It Isn't

Before you write a single prompt, get the mental model right. Most of the problems people have with AI come from misunderstanding what it actually does.

What it is

AI — specifically the large language models behind tools like Copilot — is a pattern engine. It has read more text than any human ever could, and it has learned, in a deep statistical way, what kinds of words and ideas tend to follow other kinds of words and ideas. When you ask it something, it produces a response that fits the pattern of a good answer to a question like yours.

That is genuinely useful. It means AI is excellent at:

- Writing first drafts of things that follow a familiar structure
- Reorganizing information you already have into a different format
- Summarizing long documents into shorter ones
- Comparing two things side by side
- Pulling specific data points out of unstructured text

For an estimator, that is a lot. Bid documents are long. Specs are dense. Vendor proposals come in inconsistent formats. The work of reshaping all of that into something you can use is exactly the work AI is good at.

What it isn't

AI is not a replacement for judgment. It does not know which of your vendors is reliable, which manufacturer's lead times you can actually trust, or which line item in a spec is the one that quietly blows up a budget. It does not know that the South Dakota job has different inspection rhythms than the one in Texas. It does not know what your company's margin tolerance is on a tight schedule.

It also does not know when it is wrong. This is the part that catches people. AI will produce confident, well-formatted, professional-sounding answers that are simply incorrect. It will quote prices that don't exist. It will summarize a spec in a way that subtly inverts the meaning. It will tell you a manufacturer offers a finish they don't offer. None of this is malicious — the pattern engine is just generating what looks right, not what is right.

This is why expertise still matters. Maybe more than before.

THE SHIFT TO MAKE

Stop thinking: "AI will do this estimate for me."

Start thinking: “AI will make me faster at the parts of estimating that don't require my judgment, so I can spend more time on the parts that do.”

Why this matters now

There is a temptation, especially when the tool is new and impressive, to either over-rely on it or dismiss it. Both reactions miss the point. The estimators who get the most out of AI over the next several years will be the ones who treat it the way a senior tradesperson treats a new piece of equipment: respect it, learn its quirks, use it where it earns its keep, and never let it do work that should have your judgment behind it.

That posture — measured, confident, expertise-first — is what this book is going to help you build.

PAUSE AND CONSIDER

Think about the last estimate you put together. Which parts of it were the actual hard thinking — the parts where your experience mattered most? And which parts were repetitive work that, if a smart assistant had helped you with them, would have given you more time for the hard thinking? Hold that distinction in mind as you keep reading. It's the frame the rest of the book sits on.

02 / The Estimator's Edge

You already know how to estimate. That sounds obvious, but it's the most important thing in this book. Everything that follows assumes it.

Here's why it matters. AI does not have your edge. It hasn't walked a job site. It hasn't been burned by a vendor whose lead times slipped twice in a row. It hasn't learned, the hard way, that one manufacturer's "complete door assembly" quietly excludes hardware that another manufacturer includes. The things you know — the things that took years to learn — are the things AI most needs you to bring to the table.

What your expertise actually does

In any AI-assisted task, your expertise does three jobs:

1. It defines what "good" looks like

Before AI can help you produce a quantity takeoff, you have to know what a complete, correct takeoff looks like. What units. What level of detail. Which items are typically called out separately and which get rolled up. AI can produce the format, but only if you know what format is right.

2. It catches what AI misses

When AI hands you a summary of a spec section, you read it knowing what the spec is supposed to contain. If something important is missing, you notice. If something is described in a way that doesn't match how the work actually goes, you notice. Someone without your background would read the same summary, find it plausible, and move on.

3. It tells you when to stop trusting the output

This is the most important one. AI sometimes goes off the rails in ways that are obvious to an expert and invisible to a novice. Prices that are an order of magnitude off. Scope inclusions that contradict the spec. Schedules that ignore site constraints. You catch these because you've seen the real version a thousand times. AI hasn't.

THE PRINCIPLE

Your expertise is the input, not the output. AI works best when you bring strong domain knowledge to it. The clearer your sense of what good looks like, the better AI can help you produce it.

The estimating fundamentals AI assumes you have

This list is not new to you. It's here as a reminder of what stays yours, no matter how good the tools get:

- Reading and interpreting bid documents, specifications, and drawings
- Performing accurate takeoffs and quantity counts
- Understanding scope — what's included, what's excluded, and where the gaps between trades live
- Recognizing the differences between manufacturers and what those differences cost
- Knowing which line items carry the most risk and need the most attention
- Comparing vendor proposals on an apples-to-apples basis
- Sanity-checking pricing against trusted references like RS Means

If any of these are weak, AI will not fix them. It will amplify them. A vague sense of scope plus AI assistance produces a vague estimate, faster. Strong scope discipline plus AI assistance produces a strong estimate, much faster.

PAUSE AND CONSIDER

Pick one of the fundamentals above that you'd call your strongest. Now pick the one you'd call your weakest. As you read the rest of this book, watch how each of them shows up. The strongest will tell you where AI can pay off fastest. The weakest will tell you where you'll need to be most careful.

03 / The Five-Step Framework

This is the spine of the book. Every chapter from here on hangs off it. Learn it well enough that it becomes the way you approach any AI-assisted task without thinking about it.

THE FIVE-STEP FRAMEWORK

- 1. Know the task.** What are you actually trying to produce?
- 2. Define the output.** What does “good” look like — format, fields, level of detail?
- 3. Give clear constraints.** Rules, sources, exclusions, scope.
- 4. Verify the result.** Against trusted sources, against your experience.
- 5. Reuse what works.** Turn good prompts into templates and workflows.

The framework is sequential, but not rigid. With practice, the first three steps blur together — you'll be thinking about task, output, and constraints almost at once. The last two never go away. Verification is not optional. Reuse is what turns good work into compounding leverage.

Step 1 / Know the task

Most weak AI output comes from a weak understanding of what you actually need. “Help me with this estimate” is not a task. “Extract the door schedule from this spec section and put it in a table with door number, type, hardware group, and fire rating” is a task. The clearer you are with yourself about what you want, the less work AI has to guess about.

Step 2 / Define the output

Once you know the task, define what good looks like. Is it a table? A bulleted summary? A spreadsheet? What columns? What units? How much detail per row? If you can describe the finished product in one sentence, AI can produce it. If you can't describe it, no AI in the world will guess right.

Step 3 / Give clear constraints

Constraints are the rails that keep AI on track. The most useful ones for estimators:

- Source constraints: “Use only the document I uploaded.” “Do not pull pricing from outside sources.”

- Scope constraints: “Exclude electrical work.” “Only include items in Division 8.”
- Format constraints: “Return this as a markdown table with these exact columns.”
- Behavior constraints: “If information is missing from the document, mark the cell as TBD rather than guessing.”

That last one is underrated. AI defaults to filling in plausible-looking blanks. Telling it to say “TBD” or “not found in source” when it doesn’t know is one of the single biggest quality improvements you can make.

Step 4 / Verify the result

Chapter 6 is dedicated to this. For now, hold the principle: nothing AI produces goes anywhere consequential without a human check. Not your eyes scanning it for typos — a real check, against a real reference, by a person who knows what right looks like.

Step 5 / Reuse what works

When a prompt produces something genuinely useful, that prompt is now an asset. Save it. Refine it. Turn the spreadsheet template you built with AI into a reusable template the whole estimating team can run. The first time, the prompt takes you an hour to get right. The hundredth time, the work is done in seconds. Chapter 7 covers this.

PAUSE AND CONSIDER

Pick a task from your last week of work — something you’d be willing to try with AI. Walk it through the framework in your head. Can you state the task in one sentence? Can you describe the output? What constraints would you set?

If any of those steps feel fuzzy, that’s exactly where your prompt would have gone weak. The framework’s job is to make the fuzziness visible before you write the prompt, not after.

04 / Prompting Like an Estimator

A prompt is just an instruction. The skill is not in the words — it's in the thinking behind them. Good prompts come from the framework you just learned. Bad prompts skip the framework and go straight to typing.

The fastest way to build prompting skill is to compare bad and good versions of the same request. Watch what changes from one to the other. The pattern — not any specific prompt — is what you're building.

Example 1 / Pricing research

You're working on a correctional facility bid and need a starting point on detention-grade hollow metal door pricing.

WEAK PROMPT	STRONGER PROMPT
<p>What do detention doors cost?</p>	<p>I'm estimating detention-grade hollow metal doors for a correctional facility expansion. For each of the following types, give me a range of typical 2025 list pricing per unit, list the assumptions behind the range, and note any factors that commonly drive cost up or down. Types: (1) single swing 3070 detention door with vision lite; (2) double swing 6070 detention door, no glazing. Mark any number you cannot confirm as "estimate, verify against vendor quote." Do not invent specific manufacturer prices.</p>

What changed:

- The task is named (pricing research for a specific work type)
- The output is defined (range, assumptions, cost drivers, per type)
- Constraints are set (year, document types, behavior when uncertain)
- Risk is bounded ("do not invent specific manufacturer prices" closes a common failure mode)

Example 2 / Spec review

You've uploaded a Division 8 spec section and want a quick read on what's in it before you do a full review.

<p>WEAK PROMPT</p> <p>Summarize this spec.</p>	<p>STRONGER PROMPT</p> <p>Using only the document I uploaded, give me a structured summary of this Division 8 spec section. Include: (1) a one-paragraph overview; (2) a list of all door types specified, with rating and material; (3) all hardware groups referenced; (4) any explicit inclusions or exclusions; (5) any unusual or non-standard requirements I should flag. If a section is incomplete or unclear in the source, say so rather than guessing.</p>
---	--

Notice that the strong version doesn't ask for an interpretation — it asks for extraction in a specific shape. That's deliberate. AI is much more reliable at pulling structured data out of a document than at telling you what the document means. You are the interpreter. AI is the extractor.

Example 3 / Data extraction

You need a clean schedule of door openings from a long, dense PDF.

<p>WEAK PROMPT</p> <p>Pull the door schedule out of this.</p>	<p>STRONGER PROMPT</p> <p>From the attached document, extract every door opening into a table with these columns, in this order: Door Number, Door Type, Width, Height, Fire Rating, Hardware Group, Frame Type, Notes. Include every door opening that appears in the document. If a field is not specified for a given door, enter "Not specified" — do not guess. Preserve the door numbering exactly</p>
--	---

as it appears in the source.

The columns, the order, the missing-data rule, the preservation instruction — every one of those is the difference between a usable schedule and one you have to redo.

THE PATTERN UNDER ALL OF IT

Bad prompts ask AI to figure out what you want. Good prompts tell AI what you want. The framework is what makes the second one possible — because you can't tell AI what you want if you haven't decided yourself first.

PAUSE AND CONSIDER

Think about the last time you read a spec section that gave you trouble. If you could have handed off the structured extraction work to a smart assistant — just the pulling-out-of-data part — what fields would you have asked for? What would you have had to keep doing yourself because no assistant could replace your judgment on it?

That mental split — extract vs. interpret — is the most useful one you can carry into AI-assisted work.

05 / Three Ways AI Earns Its Keep

With the framework and the prompting pattern in hand, you can apply both to the work that actually fills an estimator's day. Three use cases cover most of the value. Each gets the same treatment: what you bring, what AI does, what still needs verification.

Use case 1 / Research and pricing

What you bring

The scope. The work type. The job context that makes this question different from a generic version of itself. A correctional facility door is not a commercial office door. A budget reference for the South Dakota expansion is not the same as a budget reference for a Texas retrofit.

What AI does

Generates a starting point. Pulls together ranges, drivers, and considerations faster than you could research them from scratch. Suggests what to look up next. Frames the question more completely than you might have phrased it.

What still needs verification

Everything that becomes a number. Pricing from AI is a starting point, never a final figure. Cross-check against RS Means, against current vendor quotes, against pricing from your past jobs in similar scope. The faster AI gets you to a number, the more important it is that you verify the number before it lands in a proposal.

WATCH OUT FOR

AI is especially confident about pricing. It will give you a clean dollar figure and not flag that the figure may be two years out of date, region-mismatched, or fabricated outright. The cleaner the number looks, the more skepticism it deserves until you've verified it.

Use case 2 / Specs and bid doc review

What you bring

Knowledge of what a complete spec section is supposed to contain. Awareness of the gotchas in your trade. Sense for which divisions interact and where the seams between them tend to leak scope.

What AI does

Reads the dense parts fast. Pulls structured data out of unstructured text. Compares one document to another. Flags inconsistencies or missing items if you ask it to. Gives you a first-pass summary so your careful read can be more focused.

What still needs verification

Anything AI claims is in or not in the document. Trust-but-verify means actually opening the spec and checking. AI summaries can drop important qualifications, invert inclusions and exclusions, or quietly merge two requirements into one. The summary saves you reading time — it doesn't replace reading.

Use case 3 / Templates and data sheets**What you bring**

Knowledge of the format your team uses. What fields belong on a project data sheet. What columns belong on a takeoff. What a clean handoff to construction needs to contain.

What AI does

Auto-populates structured templates from raw source material. Reformats data from one shape into another. Generates first drafts of spreadsheets, summaries, and submittal packets that follow a known pattern.

What still needs verification

Every populated field, especially anything pulled from a source document. The most common failure here is AI confidently filling a field with something that looks right but isn't in the source. Spot-check enough cells to know whether you're dealing with a clean fill or one that needs a second pass.

Notice what these three use cases have in common. In all of them, AI is doing the work that scales poorly with attention — reading, formatting, extracting, drafting. You are doing the work that does not scale: knowing what matters, knowing what's wrong, knowing what to fix. That split is not going to change, no matter how good the tools get. Plan your habits around it.

PAUSE AND CONSIDER

Look at the three use cases above. Rank them, in your head, by which one would save you the most time on a typical week. That's the one to start practicing with first. Skill builds fastest where the payoff is highest — you'll naturally do more reps.

06 / Trust, but Verify

This is the most important chapter in the book. If everything else fades and only this lesson stays, you'll still come out ahead of most people using AI.

Here is the rule. State it out loud if you need to. Pin it above your desk. Nothing AI produces goes anywhere consequential without verification by someone who knows what right looks like.

Not a glance. Not a quick scan. Verification. Against a reference. By an expert. Every time.

WHY THIS MATTERS MORE THAN ANYTHING ELSE

AI does not know when it is wrong. It produces confident, professional-sounding output regardless of whether that output is accurate. Your job, as the human in the loop, is to be the part of the system that knows the difference. If you skip that job, the speed AI gives you turns into the speed at which mistakes reach the proposal.

Where AI commonly goes off the rails

Hallucinated pricing

AI will quote prices that look right and aren't. Sometimes the number is from a different year. Sometimes it's from a different region. Sometimes it's simply made up — a plausible-looking figure with no source behind it. The fix: never let an AI-generated price into a proposal without a current quote or a trusted reference like RS Means behind it.

Inverted scope

Summaries occasionally flip inclusions and exclusions. A spec says "Contractor shall provide X; Owner shall provide Y," and the AI summary cleans it up into something that subtly reassigns the responsibility. The fix: any scope statement AI hands you, find the corresponding line in the original document and read it yourself.

Fabricated citations

AI will invent section numbers, code references, manufacturer model names, and detail callouts. Confidently. With formatting that looks authoritative. The fix: any specific reference — "per section 087100," "Airteq model 7115," "per IBC 2021 1010.1.9" — gets confirmed against the actual document before it travels.

Plausible-but-wrong quantities

AI can produce a takeoff that totals to a number that's near-right but built from wrong counts. The total looks reasonable, so the underlying error hides. The fix: for any AI-assisted count, spot-check enough line items against the drawings to know whether you're looking at a real count or a confident guess.

Outdated information

AI's knowledge has a cutoff date. Manufacturer product lines change. Code editions change. Lead times change. A confident answer about a 2023 product line could be missing a 2025 update that changes everything. The fix: anything time-sensitive gets confirmed against current sources.

Building the verification habit

Verification is a habit, not a checklist. The estimators who do it well have built three practices into their workflow:

1. Always know what you would check the answer against before you ask the question. If you can't name a reference, you're not ready to use the output.
2. Treat AI output as a draft, never as a result. The mental model is "a capable assistant gave me a first pass," not "the answer arrived."
3. Verify proportionally to risk. A summary you're using to orient yourself needs less verification than a number going into a bid. Calibrate effort to consequence.

THE TRUSTED SOURCES EVERY ESTIMATOR SHOULD KEEP CLOSE

RS Means or equivalent pricing references, kept current

Manufacturer specification sheets and confirmed lead times — Airteq, C3, Trentech, Claborn, and others you rely on

Current vendor quotes for the specific job

Your company's own historical job cost data

The original source documents — specs, drawings, RFIs — not AI's version of them

AI does not replace any of these. It points you to them faster. The work of cross-checking is yours, and it stays yours.

PAUSE AND CONSIDER

Think about the last estimate you handed off. If an error had been in it — a wrong price, a missed exclusion, a fabricated reference — how would it have been caught? By whom? At what stage? Now imagine that error came in via AI-assisted work that nobody verified.

The system you already have for catching errors is the system AI plugs into. If that system is weak, AI will expose the weakness fast. If it's strong, AI will make it stronger.

07 / Building What You Use Twice

The last step of the framework was “reuse what works.” This chapter is short, but the principle behind it is one of the biggest sources of compounding value in AI-assisted work.

The rule is simple. If you've done a task once with AI and the result was good, the prompt you used to get there is now an asset. Don't throw it away.

From one-off to template

The first time you ask AI to extract a door schedule from a spec, you spend twenty or thirty minutes refining the prompt. By the end, you have something that works. The schedule comes out clean.

The second time you need to do that task, you start over. The prompt you took twenty minutes to refine is gone. You're refining it again, hoping to remember what made it work last time.

That's the wasted version. The better version: when you finish that first task, save the prompt. Save the spreadsheet template you built around it. Save the rules you put in place to handle missing data. The next time the task comes around, you don't refine — you reuse. The thirty-minute job becomes a thirty-second job.

Saved expertise: agents and reusable workflows

Agents are a more advanced version of the same idea. You may have heard about them in your General session, or seen them in Copilot. The full mechanics aren't the point of this chapter — you don't need to know how to configure one to benefit from the concept.

The concept is this: an agent is a saved set of instructions, configured once, that turns a repeated task into a button. Instead of writing the door schedule prompt every time, you build an agent called “Door Schedule Extractor.” It always returns the same columns. It always handles missing data the same way. It always behaves the way you decided it should behave on the day you built it.

The estimating team that builds three or four well-tuned agents for the work they do most often — takeoff structuring, spec extraction, vendor comparison — doesn't just save time. They standardize quality. The work comes out consistent across estimators and across jobs, because the saved expertise behaves the same way every time.

This is where AI stops being a personal productivity tool and starts being team leverage. It's also where this book stops — building agents in depth is a topic for a future session. For now, hold the principle: anything you do more than twice with AI is a candidate for being saved, refined, and reused.

The template instructions you can't skip

One last practical point. When you save a template or a prompt for someone else to use, include instructions on how to use it. This is not optional. It is, in fact, where most reusable AI work fails.

Someone receives a beautifully structured spreadsheet template with an AI workflow attached, doesn't know the assumptions baked into the prompt, fills in the wrong fields, gets nonsense out, and concludes that AI doesn't work. The tool didn't fail. The handoff did.

A good reusable template includes:

- A first tab or top section that explains what the template is for
- Notes on what the user needs to provide and in what format
- The prompt or agent name that the template is designed to work with
- A short list of the verification steps the user is expected to do
- A note on what to do if the AI output looks wrong

That last bullet is the most important one. Tell the next user what to do when something goes sideways. Because something, eventually, will.

PAUSE AND CONSIDER

Look at the work you do in a typical month. What's the task you repeat most often — the one you do five, ten, twenty times? That's the task most worth building a reusable workflow around. Even a five-minute saving, multiplied across the repetitions, becomes hours back in your month.

Start with the most-repeated task, not the most impressive one. Compounding leverage beats one-shot brilliance.

Appendix A / The Five-Step Framework

A one-page reference. Tear it out. Pin it up.

FIVE STEPS, EVERY TIME

1 / KNOW THE TASK

State in one sentence what you're trying to produce.

2 / DEFINE THE OUTPUT

Describe what good looks like — format, fields, level of detail.

3 / GIVE CLEAR CONSTRAINTS

Rules, sources, exclusions. Tell AI to say "TBD" rather than guess.

4 / VERIFY THE RESULT

Cross-check against trusted sources. RS Means. Vendor quotes. The original document. Your experience.

5 / REUSE WHAT WORKS

Save the prompts and templates that work. Build leverage over time.

Your expertise is the input, not the output.

Appendix B / Glossary

Plain-language definitions of the terms used in this book.

Agent

A saved configuration of AI instructions that can be reused for the same task across multiple jobs. Think of it as a prompt you've turned into a tool.

Constraint

A rule given to AI that bounds what it can do or how it should respond. Sources to use, scope to include or exclude, what to do when information is missing.

Grounding

Restricting AI to a specific source so its answer comes only from that source rather than from its general knowledge. Useful when accuracy matters more than breadth.

Hallucination

An AI output that sounds confident but is factually wrong — a price that doesn't exist, a section number that isn't in the document, a model name a manufacturer doesn't make. The most important failure mode for an estimator to recognize.

LLM (Large Language Model)

The kind of AI behind tools like Copilot. A pattern engine trained on enormous amounts of text. Good at language tasks, not a source of factual truth.

Prompt

The instruction you give AI. The quality of the prompt is the single biggest factor in the quality of the output.

Template

A reusable document or workflow built around a known-good prompt or process. The point at which AI work stops being one-off and starts being team leverage.

Verification

The non-negotiable step of checking AI output against a trusted reference before using it for anything consequential. The thing that makes AI safe to rely on.

Thinking With AI

An Estimator's Guide

Daniel Berry · Built by Berry