

Cross-Domain Reframing

- [The Cross-Domain Prompt Library](#)
- [The Stolen Technique](#)
- [The Framework Transplant](#)

The Cross-Domain Prompt Library

“ **One-liner:** Build a documented library of prompt patterns borrowed from 3+ different fields, with tested adaptations and transfer notes for your own domain.

? Jump in (Tinkerers start here)

You're going to build a personal prompt library of techniques stolen from other fields — documented well enough to teach someone else.

Phase 1 — Survey 3 domains. Pick 3 fields that are different from your own *and* from each other. Send this to three separate sessions:

“ How do professionals in **[field]** use AI in sophisticated ways? I don't want generic "they use ChatGPT" answers. Give me 5 advanced AI techniques or prompting patterns that are specific to this field. For each:

1. Name the technique
2. Describe the prompt pattern (what input, what instructions, what output format)
3. Why this technique works in this domain (what problem it solves)
4. Example prompt (ready to use)

Phase 2 — Select your top 5. From the 15 techniques across 3 domains, pick the 5 that are most interesting or most likely to transfer. For each one, send:

“ Analyze this technique from **[source domain]: [technique description]**

Map the transfer potential:

1. **Core principle:** What's the underlying mechanism that makes this work, independent of domain?

2. **Direct transfer:** What would this look like applied to **[your field]** with minimal modification?
3. **Modified transfer:** What would need to change to make it work well in my context?
4. **What doesn't transfer:** What aspect is domain-specific and should be replaced?
5. **Adapted prompt:** Write a ready-to-use version for my field

Phase 3 — Test each adapted prompt. Run all 5 adapted prompts on real tasks in your work. For each, document:

Technique	Source Domain	My Task	Result Quality (1-5)	What Worked	What Needed Adjustment
-----------	---------------	---------	----------------------	-------------	------------------------

Phase 4 — Build the library entry. For the 3 best-performing techniques, create a library card:

“ Create a "Prompt Library Card" for this technique:

Name: [give it a memorable name] **Borrowed from:** [source domain] **Core principle:** [1 sentence — why this works] **Original use:** [what it does in the source domain] **My adaptation:** [what it does in my domain] **Ready-to-use prompt:**

[the tested, refined prompt with placeholders]

When to use: [scenarios where this technique is the right choice] **When NOT to use:** [scenarios where it fails or is overkill] **Transfer notes:** [what I learned about adapting this — tips for others]

? Plan first (Planners start here)

Here's what you're about to do:

1. **Survey 3 unfamiliar domains** — Discover advanced AI techniques in three different fields. Cast a wide net — diversity of domains matters more than depth.
2. **Select the top 5 candidates** — From 15 techniques, choose 5 based on transfer potential and novelty. Look for techniques that solve a problem *structurally* similar to one in your work.
3. **Analyze transfer mechanics** — For each technique, separate the domain-specific elements from the core principle. Identify what transfers directly, what needs

modification, and what should be replaced.

4. **Test all 5 adaptations** — Run each adapted prompt on a real task. Document quality, surprises, and adjustments needed.
5. **Document the top 3** — Create library cards with enough detail for someone else to use the technique without your guidance.

"Done" looks like: A 3-entry prompt library with tested techniques from other domains, complete with ready-to-use prompts, usage guidance, and transfer notes.

? Why this matters (Strategists start here)

In [CDR-Basic-01](#), you borrowed a single technique. In [CDR-Intermediate-01](#), you transplanted an entire framework. Here, you're building a **systematic practice** — a personal library that compounds over time. The library card format forces you to articulate *why* a technique transfers, which is the meta-skill: once you can spot the structural similarity between domains, you can generate new cross-domain adaptations on your own. This library also becomes a shareable team asset — a collection of non-obvious AI techniques that others in your field won't have discovered.

Reflection

- Which of the 3 domains produced the most transferable techniques? Why?
- Did any technique work *better* in your domain than in its original domain? What does that tell you?
- What pattern do you notice in what transfers well vs. what doesn't? Can you articulate a rule of thumb?
- ☐ *Share your prompt library with colleagues and have them test the techniques on their own tasks. The techniques that work across multiple people's contexts are genuinely domain-agnostic — those are your keepers.* (Social Learners)

?? Level up

You've reached the advanced level for Cross-Domain Reframing. From here, consider:

- Expanding the library monthly — add one new cross-domain technique per month from a new field
- Sharing the library with colleagues and collecting their transfer notes

- Combining this with [WA-Advanced-01](#) to build cross-domain techniques into automated workflows

Back to [Cross-Domain Reframing](#)

The Stolen Technique

“ **One-liner:** Take an AI technique from a completely different field and apply it to your own work — discovering that the best prompting ideas are often borrowed.

? Jump in (Tinkerers start here)

Pick a field that is **not** your own. If you work in marketing, pick engineering. If you're a designer, pick finance. If you're a developer, pick journalism. The more unfamiliar, the better.

Step 1 — Discover a technique. Send this prompt:

“ How do professionals in **[unfamiliar field]** use AI in their daily work? Give me 5 specific, concrete techniques — not general concepts. For each technique, describe: what they prompt the AI to do, what input they provide, and what output they get. Focus on techniques that are unique to this field.

Step 2 — Steal the best one. Pick the technique that seems most interesting or most different from how you currently use AI. Then send:

“ I work in **[your field]**. Take the technique you described as **#[number]** — **[briefly describe it]** — and help me adapt it for my work. Specifically:

1. What would the equivalent input look like in my field?
2. How would I modify the prompt to fit my context?
3. What output would I expect?
4. Write me a ready-to-use prompt that applies this borrowed technique to **[a specific task you do]**.

Step 3 — Test it. Copy the adapted prompt. Use it on a real task. Compare the result to how you'd normally approach it.

Example — a marketer borrowing from investigative journalism:

The technique: Journalists use AI to cross-reference claims across multiple sources and flag inconsistencies.

The adaptation: A marketer uses the same technique to cross-reference their product claims against competitor claims and customer reviews, flagging gaps between promise and reality.

? Plan first (Planners start here)

Here's what you're about to do:

1. **Pick an unfamiliar field** — Choose something genuinely outside your expertise. The discomfort is the point — that's where non-obvious ideas live.
2. **Research AI techniques in that field** — Use AI to discover how professionals in that domain use AI tools. Look for specific techniques, not generalities.
3. **Identify a transferable technique** — Pick one that solves a problem similar to something in your work, even though it looks completely different on the surface.
4. **Adapt with AI's help** — Ask the AI to bridge the gap between the source domain and your domain. Get a ready-to-use prompt.
5. **Test the borrowed technique** — Apply it to a real task and evaluate whether it gives you a different (and possibly better) result than your usual approach.

"Done" looks like: You have a working prompt borrowed from another field that gives you a new angle on a familiar task.

? Why this matters (Strategists start here)

Most people prompt AI using patterns from their own field — but the most powerful AI techniques are often domain-agnostic. Researchers structure AI analysis differently than marketers, engineers test AI outputs differently than writers, and each field has developed prompting patterns the others rarely see. Cross-domain reframing is how you break out of local optima in your AI usage. At the intermediate level, you'll systematically adapt entire prompt strategies across domains; this exercise builds the muscle of looking outside your field for AI inspiration.

Reflection

- Did the borrowed technique produce a noticeably different result than your usual approach? Better, worse, or just different?
- What made the technique transferable? Was it the structure, the question type, or the underlying problem it solves?
- Which other field would you explore next for AI techniques? What made you choose it?
- ☐ *Ask a colleague from a different department how they use AI. You'll likely discover a technique you've never considered — that's cross-domain reframing in action.* (Social Learners)

?? Level up

Ready for more? Try [CDR-Intermediate-01](#) — where you'll systematically adapt an entire prompting strategy from an unfamiliar domain.

Back to [Cross-Domain Reframing](#)

The Framework Transplant

“ **One-liner:** Take a complete problem-solving framework from another domain and systematically adapt it to solve a challenge in your own work.

? Jump in (Tinkerers start here)

Pick a challenge you're currently facing in your work — something you've been approaching the same way without breakthrough results.

Step 1 — Find a foreign framework. Send this prompt:

“ I'm struggling with **[your challenge]** in my field of **[your field]**. I want a completely fresh approach. Give me 3 well-known problem-solving frameworks from **different** fields (engineering, medicine, military strategy, game design, ecology — anything outside my domain). For each framework:

1. Name and origin field
2. How it works (3-4 step process)
3. Why it might apply to my problem

Choose frameworks that are genuinely different from each other, not variations on the same idea.

Step 2 — Deep-dive one framework. Pick the most promising or most surprising framework. Send:

“ Let's go deeper on **[chosen framework]**. Walk me through how a professional in **[its origin field]** would apply this framework to a real problem in their domain. Be specific — give me a concrete example with actual steps, not abstractions.

Step 3 — Systematic transplant. Now adapt it:

Now help me transplant this framework to my challenge: **[restate your challenge]**.

Map each step of the framework to my context:

- **Step 1 of framework** → What does this look like in my situation?
- **Step 2 of framework** → What's the equivalent action?
- (continue for all steps)

For each mapping:

- What translates directly?
- What needs to be modified and how?
- What doesn't transfer at all, and what should replace it?

End with a concrete action plan I can execute this week.

Step 4 — Stress test. Send:

“ Play devil's advocate. Where does this transplanted framework break down when applied to my field? What assumptions from the original domain don't hold in mine? How should I adjust?

? Plan first (Planners start here)

Here's what you're about to do:

1. **Identify your challenge** — Pick something real where your current approaches have stalled. The exercise only works if you're genuinely stuck.
2. **Discover foreign frameworks** — Use AI to surface structured problem-solving approaches from unfamiliar fields. Look for frameworks with clear steps, not just theories.
3. **Study the framework in its native context** — Understand how it actually works in practice before trying to adapt it. This prevents shallow borrowing.
4. **Map step-by-step to your context** — Systematically translate each step, noting where the mapping is direct, where it needs modification, and where it fails entirely.
5. **Stress test the adaptation** — Identify where the transplant breaks down and adjust before committing to action.

"Done" looks like: A concrete action plan for your challenge, based on a framework from another field, with clear documentation of what translated, what was modified, and what was replaced.

? Why this matters (Strategists start here)

In [CDR-Basic-01](#), you borrowed a single technique from another field. Here, you're transplanting an **entire framework** — a much harder and more valuable skill. This is how breakthrough innovations happen: the structure of a solution transfers across domains even when the details don't. Toyota's production system was adapted from supermarket inventory management. Agile software development borrowed from lean manufacturing. The ability to systematically adapt frameworks across domains is what separates insight from coincidence. At the advanced level, you'll build an entire cross-domain prompt library; this exercise builds the adaptation methodology.

Reflection

- Which parts of the framework transferred most easily? What does that tell you about the underlying structure of your problem?
- Where did the transplant break down? Was the breakdown due to domain differences, or did it reveal an assumption you hadn't questioned?
- Did the stress test change your action plan significantly, or just refine the edges?
- ☐ *Explain the transplanted framework to someone in the original field. Their reaction ("that's not how we use it" or "interesting adaptation") tells you whether you captured the core principle or just the surface.* (Social Learners)

?? Level up

Ready for more? Try [CDR-Advanced-01](#) — where you'll build a cross-domain prompt library with documented transfer patterns.

Back to [Cross-Domain Reframing](#)