

SESSION ONE

AI for Business Knowledge Notes

The landscape, the tools, what an agent actually is, and how to automate real work without building a machine you can't trust.

PREPARED FOR

Business owners

No engineering background needed

PREPARED BY

Ilia Reingold

iliareingold.com

CHAPTER

1 of an ongoing series

13 August 2026

ABOUT THIS DOCUMENT

A living document, not a meeting recap

This is Chapter 1 of a knowledge base we build together, one session at a time. Each session adds chapters; along the way we clean out repetition so the material stays a book you can actually read rather than a pile of meeting summaries.

Two things are worth knowing before you start reading.

First, the structure is deliberate. The material is organised into six parts that move from orientation, to concepts, to plumbing, to how you actually work with these systems, to your specific business problem, and finally to the discipline that keeps all of it from getting out of hand. You can read it front to back, or jump to the part you need.

Second, some of it will age faster than the rest. Ideas like *context beats clever prompting* or *match supervision to risk* will hold for years. Product names, pricing tiers, and which company temporarily has the best feature will not. Anywhere the material is tied to a specific product rather than a durable idea, you'll see this marker:

PRODUCT-SPECIFIC · EXPECT CHANGE

Treat those passages as a snapshot of August 2026, not a permanent truth. The concept around them is what you're meant to keep.

IF YOU READ ONLY ONE PAGE

Pick one AI ecosystem and go deep. Feed it context instead of engineering clever prompts. Speak to it rather than typing at it. Automate in layers — read, summarise, classify, update, recommend, execute — and prove each layer before adding the next. Give it aggressive freedom on low-risk work and keep your hands on the wheel for anything expensive or irreversible.

CONTENTS

PART I

Choosing Your Ground

Why the winning move is depth in one ecosystem rather than breadth across all of them.

1. Don't try to learn every AI tool · 2. Frontier AI labs · 3. The model vs. the application · 4. Model selection and reasoning effort

PART II

From Chatbot to Agent

What actually changes when you let an AI use tools and work in a loop.

5. Chat vs. work vs. agents · 6. What is an AI agent? · 7. The agent loop · 8. The most important question for automation · 9. Computer-use agents · 10. Local agents vs. cloud agents

PART III**How Software Actually Connects**

Code, APIs, MCP, connectors — the plumbing, explained without jargon.

11. Codex · 12. Don't be afraid of code · 13. APIs · 14. Front end vs. back end · 15. MCP · 16. Connectors and plugins · 17. Why combining tools matters · 18. Zapier, Make, and n8n

PART IV**How to Actually Work With AI**

The habits that separate good output from mediocre output.

19. Prompt engineering is fading · 20. Context beats clever prompts · 21. Voice is a powerful interface · 22. AI as a thinking partner

PART V**Your Problem: Too Much Incoming**

Turning the concepts into a system that protects your attention.

23. The practical business problem · 24. The desired workflow · 25. AI as an attention firewall · 26. Start with the simplest architecture · 27. Build automation in layers · 28. Human supervision still matters

PART VI**Practical Discipline**

Spending, learning, and finding the right division of labour.

29. Don't upgrade before you need to · 30. A better way to learn AI · 31. The sweet spot

CLOSING**Homework, Takeaways & Glossary**

What to try before Session 2, the ten things to remember, and a plain-language reference for every term used here.

PART ONE

Choosing Your Ground

The AI ecosystem changes faster than anyone can follow. The winning move is not to follow it — it is to pick one strong environment and learn it deeply.

1. Don't try to learn every AI tool 2. Frontier AI labs 3. The model vs. the application 4. Model selection and reasoning effort

1 Don't try to learn every AI tool PRODUCT-SPECIFIC · EXPECT CHANGE

New models, interfaces, and products appear constantly. Most business users do not need to use any of them. Chasing the ecosystem is a full-time job that produces no revenue.

| Choose one strong AI ecosystem as your primary environment, and learn to use it deeply.

For you specifically, the recommendation from this session is to make **ChatGPT** your primary tool. The reasoning:

- You already prefer the interface, and comfort compounds.
- You already have substantial context and history built up there.
- Most of the tasks you want to perform can plausibly be handled inside that one ecosystem.
- Constantly moving between ChatGPT, Claude, Gemini and others creates friction with no matching payoff.

Claude remains useful as a secondary tool. But there is no reason to divide your work across multiple AI companies unless a concrete advantage forces the split.

2 Frontier AI labs PRODUCT-SPECIFIC · EXPECT CHANGE

A **frontier AI lab** is a company developing some of the most capable general-purpose AI models available. The ones that came up in this session:

OpenAI	ChatGPT, Codex, and related developer tooling.
Anthropic	Claude.
Google	Gemini.
xAI	Grok.

The important principle is that these companies converge. When one introduces computer use, agents, research tools, connectors, coding environments, or workplace features, the others generally ship something comparable within months.

Don't switch platforms every time a competitor temporarily has a better feature.

For ordinary business use, the differences between the leading systems matter far less than how effectively you use the one you already have.

3 The model vs. the application

A distinction that becomes more useful the deeper you go.

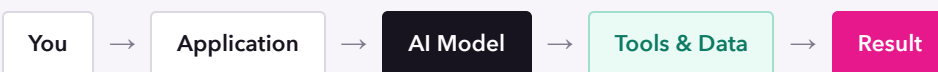
THE APPLICATION

The interface you interact with — the app, the website, the chat window. ChatGPT, Claude and Gemini are applications.

THE MODEL

The underlying AI system that generates the intelligence. The interface and the intelligence beneath it are not the same thing.

THE FULL PATH OF A REQUEST



This distinction becomes important as soon as you start working with agents, APIs, coding environments, automation, model selection, or reasoning settings — because at that point you are choosing the model, not just opening the app.

4 Model selection and reasoning effort

Different models are optimised for different levels of complexity, speed, cost, and reasoning depth. A useful strategy:

DIFFICULT PROBLEMS

Use the strongest available model, with more reasoning effort.

- Architecture and system design
- Complicated business strategy
- Difficult research
- Planning a large software system
- Analysing a complicated workflow

EASIER EXECUTION

Once the hard thinking is done, smaller and faster models can execute the individual pieces.

- Researching individual items
- Populating spreadsheets
- Writing sections of documents
- Repetitive, well-defined work

ORCHESTRATION

Strong model



Creates the complete plan



Fast models



Execute each piece

This is **AI orchestration**, and the principle behind it is simple: spend expensive intelligence on difficult decisions and cheap, fast intelligence on execution.

DON'T OVER-THINK THIS YET

For a normal business user there is no need to obsess over model selection at the start. Use the default model, and reach for more reasoning effort when a task is genuinely difficult. The orchestration idea matters most once you are running multi-step work.

PART TWO

From Chatbot to Agent

Chat, work, and agents sound like three different products. Underneath, they are the same thing with different permissions — and one crucial addition: the loop.

5. Chat vs. work vs. agents 6. What is an AI agent? 7. The agent loop 8. The most important question for automation 9. Computer-use agents 10. Local agents vs. cloud agents

5 Chat vs. work vs. agents

These categories sound more complicated than they are. At a fundamental level, they are all the same AI model. The only real differences are:

| What tools is the AI allowed to use, and how independently can it operate?

Traditional chat

A basic chatbot works like this: you ask something, the AI thinks, the AI responds. Its only tool is conversation.

“Explain the difference between gross profit and net profit.” — The AI answers. Nothing else happens.

Chat with tools

Now give the same model additional capabilities: web search, Gmail access, calendar access, spreadsheets, documents, file access. It can now *act* in addition to generating text.

EXAMPLE

“Look through today’s emails and tell me which ones require action.”

To answer, the AI must access email, read the messages, judge relevance, summarise, and respond. That is five steps and four of them are actions, not text generation. This is already starting to resemble an agent.

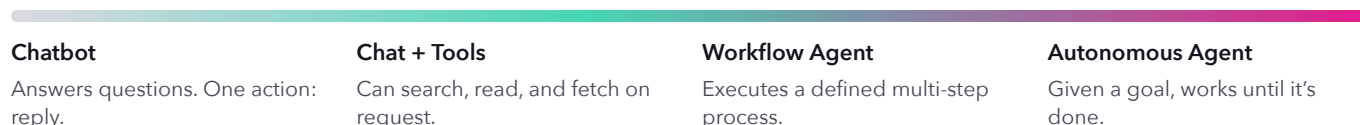
6 What is an AI agent?

| An AI agent is an AI model that can take actions using tools on your behalf.

A basic chatbot has exactly one available action: write text back to the user. An agent might also be able to:

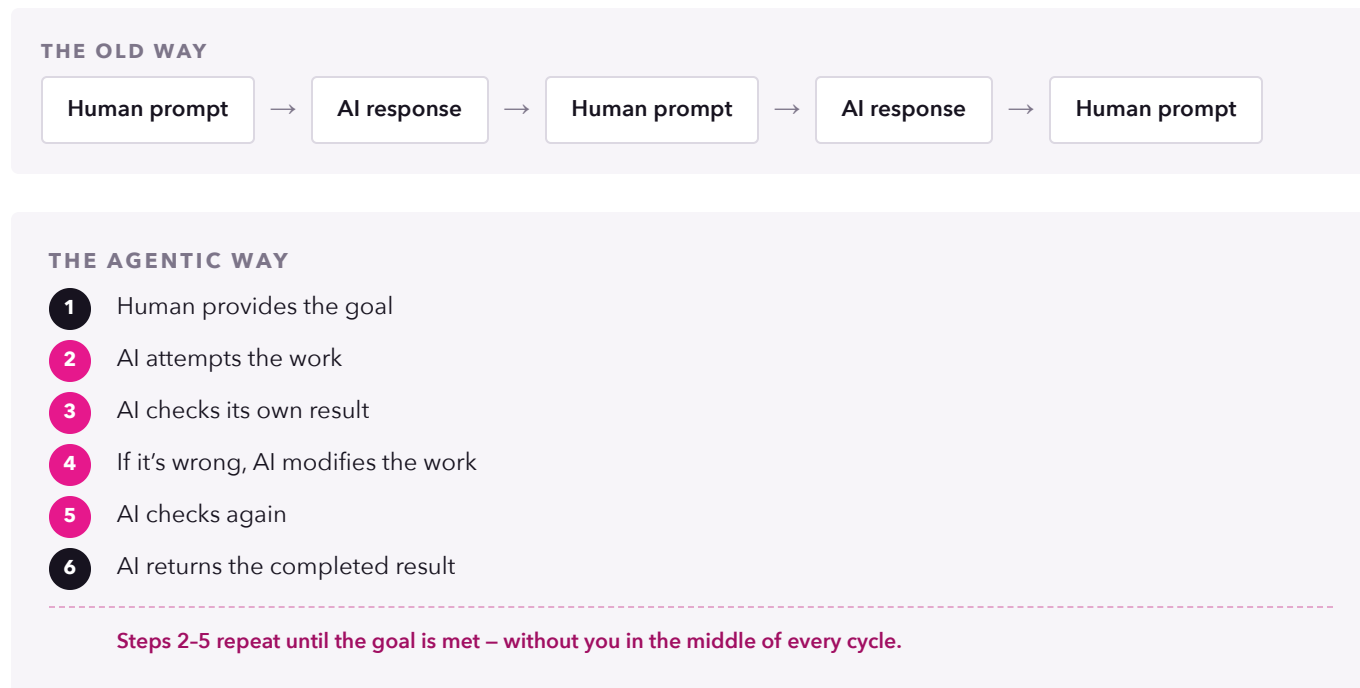
- Search the web
- Read files
- Write spreadsheets
- Send or draft emails
- Manipulate documents
- Interact with software
- Operate a browser
- Run code
- Check its own work
- Repeat a process until a goal is reached

There is no clean boundary between a chatbot and an agent. It is better understood as a spectrum:



7 The agent loop

One of the biggest developments in modern AI systems is letting the AI work through a problem repeatedly, instead of requiring a human prompt after every single step.



This is far more powerful, and it changes your role entirely.

BEFORE

You are the person performing every step, and the AI assists.

AFTER

You are the person defining the goal and supervising the system.

8 The most important question for automation

The question people usually ask is “*Can AI do my job?*” It is the wrong question, and it produces either panic or dismissal. Ask this instead:

Which parts of this process can AI perform reliably?

THE QUESTION THAT ACTUALLY LEADS SOMEWHERE

Any business process is a mix of very different kinds of work:

- Creative decisions
- Relationship management
- Judgment
- Physical work
- Data entry
- Research
- Communication
- Scheduling
- Document creation
- Repetitive software clicks

AI may automate only some of those. That is still extremely valuable. The goal was never complete automation.

THE ACTUAL GOAL

Remove as much repetitive, low-value work from the human as possible — and leave the human doing the work only a human can do.

9 Computer-use agents

Computer-use systems let an AI interact with software interfaces roughly the way a person does. They can open a browser, navigate websites, click buttons, fill fields, move between pages, extract information, and complete repetitive web workflows.

This is particularly useful for the software that never got modernised:

- Old government websites
- Administrative portals
- Repetitive data entry
- Sites with no usable API
- Anything requiring many manual clicks
- Permit and compliance systems

Instead of spending ten minutes fighting an unpleasant website, you give the agent a goal and let it perform the interaction.

IMPORTANT LIMIT

Computer use is genuinely useful, but it is not a licence for unlimited autonomy. High-impact or sensitive tasks still deserve human confirmation before the agent acts, and supervision while it does.

10 Local agents vs. cloud agents

An agent has to run *somewhere*. There are two options and the difference is practical, not philosophical.

LOCAL AGENT

Runs on your own computer.

ADVANTAGES

- Easy to start
- Direct access to local files and software
- Ideal for experimentation

DISADVANTAGE

Your machine has to stay on and available.

CLOUD AGENT

Runs on a remote computer or server.

ADVANTAGES

- Keeps working while your computer is off
- Reachable from anywhere
- Can run continuously
- Supports more advanced automation

DISADVANTAGE

More setup, more moving parts.

WHAT A CLOUD AGENT MAKES POSSIBLE

Phone or laptop



Cloud agent



Performs the work



Returns the result

Eventually this means work happening while you are on site, travelling, at the gym, or asleep. That is a real and achievable end state — but it is an advanced stage, not a starting point.

Start locally. Move to cloud infrastructure only when a real business need forces it.

How Software Actually Connects

Code, APIs, MCP connectors. You don't need to build any of it, but understanding roughly how the pieces talk to each other is what lets you specify a system worth building.

11. Codex 12. Don't be afraid of code 13. APIs 14. Front end vs. back end 15. MCP 16. Connectors and plugins
17. Why combining tools matters 18. Zapier, Make, and n8n

11 Codex PRODUCT-SPECIFIC · EXPECT CHANGE

Codex came up as an environment suited to software creation, coding, automation, agents, and computer-based workflows. The specific product will evolve; the lesson underneath it will not.

You do not need to know how to write code in order to use AI to build software or automation.

What you need to supply instead is the thing you already have:

- The business problem
- The desired workflow
- The inputs
- The output
- What should happen under different conditions

AI is increasingly capable of translating those requirements into working code. That opens software development to people who understand a business problem deeply but have never been programmers.

12 Don't be afraid of code

Code is becoming less of a barrier every year. The path from idea to working software has genuinely changed shape.

PREVIOUSLY

Business idea → Explain it to an engineer → Engineer writes the software

INCREASINGLY

Business idea → Explain it to AI → AI writes the software → You test and direct it

The consequence is that **domain knowledge becomes extremely valuable**. Someone who deeply understands construction, mortgages, real estate, law, accounting, or logistics may now be able to build specialised software without becoming a traditional programmer.

THE SKILL THAT REPLACES CODING

Clearly describing how the system should work: the inputs, the rules, the exceptions, and what “done correctly” looks like. That has always been the hard part of software. It is now the main part.

13 APIs

API stands for *Application Programming Interface*. It is the way one piece of software talks to another piece of software — without a human clicking anything.

A REQUEST AND ITS ANSWER

Software A → “Give me the details for customer #184” → Software B

Software A ← name · email · transactions · account status ← Software B

APIs matter because agents frequently need to interact directly with external software. Common examples you will encounter:

- Gmail API
- Google Calendar API
- Payment APIs
- Real estate APIs
- CRM APIs
- Mapping APIs

When a good API exists, software can do the task far more reliably than by clicking around a website.

This is the practical reason the question “*does this system have an API?*” is worth asking before you design an automation around it.

14 Front end vs. back end

FRONT END

The part humans see and touch.

- Buttons and menus
- Forms
- Dashboards
- Web pages

BACK END

The underlying system doing the actual work.

- Data and databases
- Calculations and business rules
- Handling requests
- Authentication
- Integrations with other systems

An API usually acts as the communication layer between an outside system and the back end. Which is why an agent working through an API is going straight to where the work happens, rather than pretending to be a person poking at the front end.

15 MCP – Model Context Protocol

MCP stands for *Model Context Protocol*. The idea is to make it easier for AI systems to connect to external tools, applications, databases, and information sources.

Without a shared standard, every AI system needs a bespoke integration for every tool — an enormous and endlessly duplicated amount of work. A common protocol instead gives AI systems a standardised way to discover what a tool can do and then use it.

WHAT YOU ACTUALLY NEED TO TAKE AWAY

MCP is part of the infrastructure that lets AI connect to other software and data. You do not need to master it — or even configure it — to benefit from the tools built on top of it. Recognise the acronym, understand the purpose, move on.

16 Connectors and plugins PRODUCT-SPECIFIC · EXPECT CHANGE

Connectors are what let an AI system reach into your other services. The ones discussed included Gmail, Google Docs, Google Sheets, GitHub, and calendars.

The value is best seen by comparing the two workflows:

WITHOUT CONNECTORS

AI generates text

→

You copy it

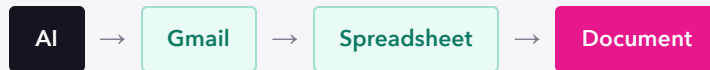
→

You open the spreadsheet

→

You paste it

WITH CONNECTORS



Every arrow you remove from the first diagram is a piece of your day handed back to you. Connectors eliminate the human transfer work — the copying, pasting, and re-typing that produces nothing but exists because two systems don't talk.

17 Why combining tools matters

The real power comes from connecting multiple capabilities into one chain, rather than asking an AI isolated questions.

AN EXAMPLE BUSINESS WORKFLOW

- 1 Emails arrive
- 2 AI reviews the messages
- 3 AI identifies the actionable items
- 4 AI updates the task database
- 5 AI assigns priorities
- 6 AI compares tasks against your calendar
- 7 AI produces a daily briefing
- 8 You review only what matters

Every one of those steps is individually unremarkable. Chained together, they replace a category of work you currently do by hand every single day — and that is a fundamentally different proposition from asking ChatGPT a good question.

18 Zapier, Make, and n8n PRODUCT-SPECIFIC • EXPECT CHANGE

These are workflow automation platforms. Their purpose is to connect software and trigger actions, typically as a chain of steps:



Historically these tools were extremely valuable, because every application had to be manually wired to every other one. AI agents increasingly offer another route to the same outcome.

THE RECOMMENDATION FOR NOW

You do not need to learn these tools yet. They remain genuinely useful, but learning them *before* you have determined your actual business workflow adds complexity you can't yet evaluate.

Start with the AI tools and integrations already available to you. Move to a specialised automation platform only when you hit something they can't do.

PART FOUR

How to Actually Work With AI

The elaborate prompt formulas are over. What replaces them is simpler and, oddly, more human: give the system the material, and talk to it.

19. Prompt engineering is fading 20. Context beats clever prompts 21. Voice is a powerful interface 22. AI as a thinking partner

19 Prompt engineering is becoming less important

Earlier generations of AI needed carefully structured prompts, and an entire cottage industry grew up around them: assigned roles, rigid templates, magic wording, elaborate instruction formats.

Modern models are substantially better at interpreting normal human communication. The formulas are mostly obsolete.

THE OLD PROBLEM

"Did I use the perfect prompt?"

THE REAL PROBLEM NOW

"Did I give the AI enough useful context?"

That is a major distinction, and it changes where you should spend your effort.

20 Context beats clever prompts

Better output comes from better information, not better phrasing. Useful context includes:

- Emails
- Documents
- Spreadsheets
- Examples
- Screenshots
- Previous work
- Business rules
- Customer history
- Goals and constraints

Rather than constructing an elaborate artificial prompt, hand the AI the actual source material.

EXAMPLE

Instead of writing three paragraphs summarising a twenty-message email thread from memory:

Give the AI the thread. Then say: "Handle this."

The source already contains the context. Your summary would only lose detail.

21 Voice is an extremely powerful interface

One of the strongest recommendations from this session: use voice input far more than you currently do. The reason is not convenience.

WHEN YOU TYPE

You naturally compress. You strip out the background, the exceptions, and the reasoning — precisely the material the AI needs most.

WHEN YOU SPEAK

You naturally expand. Context, nuance, examples, exceptions, uncertainties, reasoning and background all come out without effort.

A WORKFLOW WORTH TRYING

- 1 Start talking
- 2 Explain the situation naturally
- 3 Think through your uncertainties out loud
- 4 Explain what you want
- 5 Let the AI organise the information

You don't need to know exactly what you want before you start speaking. Your thinking process is itself useful context.

22 AI can help you think, not just execute

You are allowed to start from uncertainty. Saying *"I don't completely know what I want yet — here's the situation..."* and then describing the problem is a legitimate and often productive opening.

From there the AI can help you:

- Clarify the objective
- Identify missing information
- Organise the problem
- Suggest alternatives
- Build an execution plan
- Pressure-test your assumptions

This turns AI from a command executor into a thinking partner – and the thinking partner is worth considerably more.

PART FIVE

Your Problem: Too Much Incoming

Everything in the previous four parts exists to serve this: you don't have an information shortage, you have an interruption problem. Here is how the concepts turn into a system.

23. The practical business problem 24. The desired workflow 25. AI as an attention firewall 26. Start with the simplest architecture 27. Build automation in layers 28. Human supervision still matters

23 The practical business problem

You manage design and construction projects across a large number of communication channels:

- Multiple Gmail accounts
- Phone calls
- Google Sheets
- WhatsApp
- Project information
- iMessage
- Task lists

| The problem is not a lack of information. It is too much incoming information.

The goal is to stop the constant interruption. Rather than repeatedly checking phone, messages, email and WhatsApp on an unpredictable loop all day, you want an AI system acting as an information filter between all of those channels and your attention.

24 The desired workflow

The system, described end to end.

INPUTS

- Gmail account #1
- Gmail account #2
- Other email accounts
- WhatsApp
- Possibly text messages
- Existing task list
- Project information

AI PROCESSING – WHAT IT DECIDES ABOUT EACH MESSAGE

Project	Which job does this belong to?
Action required	Does anything actually need to happen, or is this FYI?
Importance	How much does this matter to the project?
Urgency	Does it matter <i>today</i> ?
Owner	Who is responsible for the next step?
Deadline	When does it need to be done?
Existing task	Is this new work, or an update to something already tracked?

TASK SYSTEM

The AI writes its conclusions into a centralised task database or Google Sheet — a single place that reflects the true state of every project.

BRIEFING

At designated times, the AI delivers: important messages, urgent tasks, overdue items, priorities, and the decisions that specifically require you.

THE OUTCOME

You work from an AI-generated prioritised briefing instead of constantly monitoring communication channels. The channels keep filling up. You just stop living inside them.

25 AI as an attention firewall

This is the single most valuable business use identified during the session. Most people think of AI as something that generates content. Its higher-value role here is to *protect human attention*.

TODAY

100 → 100

Every message reaches you. You read all of them, then decide what matters.

100 messages → you read 100 → you decide

WITH A FILTER

100 → 7

The AI reads everything. You see only what genuinely requires you.

100 messages → AI analyses 100 → you see the 7 that matter

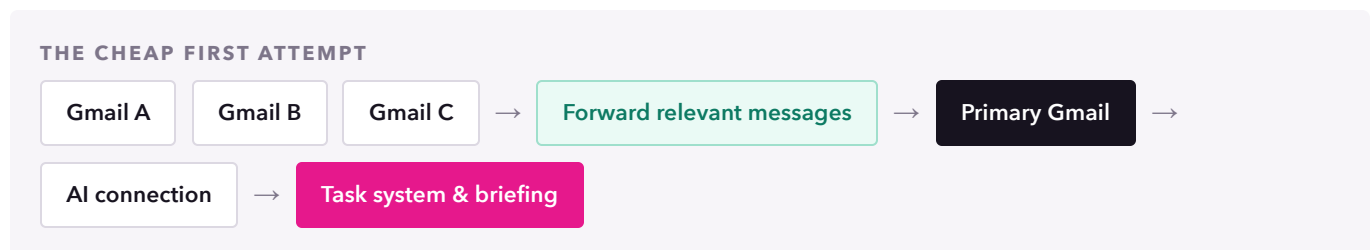
The productivity gain here is not incremental. It removes an entire category of work — the reading-to-decide-whether-to-read — that consumes a large share of a project manager’s day and produces nothing.

| The scarce resource is not information. The scarce resource is human attention.

26 Start with the simplest possible architecture

| Do not build an advanced system before proving that a simple one works.

For the multiple-Gmail problem specifically, one first experiment discussed was deliberately unglamorous:



This may let you skip building custom infrastructure entirely. If it fails, *then* you consider a more sophisticated solution — and you’ll have learned exactly why you need one, which makes the next version far better designed.

WHY THIS MATTERS MORE THAN IT SOUNDS

Complex systems fail in complex ways. A simple system that works badly teaches you the right lesson quickly and cheaply. A complex system that works badly can take weeks to diagnose.

27 Build automation in layers

Do not attempt to build the complete AI business operating system on day one. There is a much better progression, and each layer is independently useful.

LAYER 1	Read — AI reads the relevant information.
LAYER 2	Summarise — AI tells you what happened.
LAYER 3	Classify — AI determines project, importance, urgency, and whether action is required.
LAYER 4	Update — AI updates your task database.
LAYER 5	Recommend — AI tells you what to do next.
LAYER 6	Execute — AI performs the safe tasks itself.
LAYER 7	Autonomous — AI handles multi-step work independently, escalating to you only when necessary.

This progression is dramatically easier to debug and to trust. When something goes wrong at Layer 4, you already know Layers 1–3 are sound. Build all seven at once and a single bad output could come from anywhere.

28 Human supervision still matters

AI agents can perform substantial work. But there is an important gap between two things that sound similar:

CAPABILITY

Whether the AI is technically *able* to send the email, edit the record, or make the decision.

RELIABILITY

Whether it does so correctly, consistently, and in the cases that matter most.

Capability does not automatically earn permission. The useful pattern is to match the level of supervision to the cost of being wrong.

LOW RISK	MEDIUM RISK	HIGH RISK
Allow automatic execution. - Organising notes - Summarising email - Updating an internal spreadsheet	AI prepares → you approve. - Drafting a client email - Rescheduling a meeting - Modifying an important project record	Keep direct human control. - Major financial decisions - Contracts - Irreversible actions - Sensitive client commitments

Automate the left column aggressively. Be deliberate in the middle. Keep your hands on the right.

PART SIX

Practical Discipline

Three habits that determine whether this becomes a working system or an expensive hobby: spend late, learn by doing, and be honest about what humans are still better at.

29. Don't upgrade before you need to 30. A better way to learn AI 31. The sweet spot

29 Don't upgrade software before you need to PRODUCT-SPECIFIC · EXPECT CHANGE

Do not move to a more expensive subscription simply because it exists. Upgrade when your current plan becomes the thing standing in your way.

SIGNALS THAT IT'S TIME

- You regularly hit usage limits
- You need a feature genuinely unavailable on your current plan
- You're running many simultaneous agents
- You're performing consistently heavy workloads

| Until then, extract more value from the tools you already pay for.

30 A better way to learn AI

You identified two opposite mistakes during the session, and they are the two most common failure modes.

MISTAKE ONE

"There is so much I don't know that I can't begin."

Paralysis. The field is too large to master, so nothing starts.

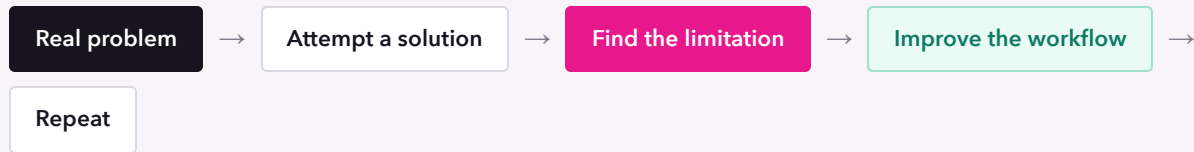
MISTAKE TWO

"AI should be able to magically do everything."

Disappointment. One failure gets read as proof the whole thing is hype.

The productive middle ground is experimentation.

THE ACTUAL LEARNING LOOP



This builds practical knowledge far faster than consuming endless courses — and it builds knowledge about *your* business, which no course can give you.

31 The sweet spot

The goal is not that AI does everything, and it is not that you keep doing everything manually.

Find the division of labour where humans do the work humans are best at, and AI handles everything else it can perform reliably.

For a design and construction business, the split looks roughly like this:

HUMAN STRENGTHS

- Relationships
- Judgment
- Negotiation
- Creative design
- On-site decisions
- Leadership
- Nuanced prioritisation

AI STRENGTHS

- Monitoring
- Summarising
- Organising
- Researching
- Extracting information
- Updating systems
- Creating drafts
- Repetitive digital tasks
- Cross-referencing large volumes of information

Notice that almost nothing appears on both lists. That is the point — and it is why this is a division of labour rather than a competition.

BEFORE SESSION TWO

Homework

The point of these is not to succeed at all of them. Where the system fails is the most valuable thing you can bring to the next session.

PRACTICAL EXPERIMENTS

PRODUCT-SPECIFIC · EXPECT CHANGE

- ☐ **Work inside ChatGPT and its connected tools.** Spend real working hours there rather than test questions — the goal is to find the edges of what it can reach.
- ☐ **Install Codex and experiment with it.** Treat it as a place to describe problems, not write code.
- ☐ **Ask Codex to create code or automation.** Don't hedge. Describe a real workflow and see how far it gets.
- ☐ **Investigate consolidating your business Gmail accounts** into a single workflow the AI can access.
- ☐ **Test email forwarding into a central inbox** if that turns out to be the simplest route.
- ☐ **Have AI interact with your existing Google Sheets task system.** Reading first, then writing.
- ☐ **Use voice input instead of typing** for at least a few complex requests, and notice the difference in output quality.
- ☐ **Identify one repetitive digital task** that could plausibly be delegated to AI entirely.
- ☐ **Record where the system fails** or produces poor results. These failures are the starting point for Session 2.

THE ONE THAT MATTERS MOST

The last item. A list of specific failures — *“it couldn't tell which project this belonged to”, “it kept marking supplier updates as urgent”* — is worth more than a list of successes. Successes tell us what to keep. Failures tell us what to build.

SUMMARY

Ten Things to Remember

If the rest of this document disappeared, these are the ideas worth keeping.

01 Pick a primary AI ecosystem.

Go deep in one environment. Don't switch tools every time a competitor announces something.

02 Context beats clever prompting.

Give the AI the information it needs to understand the problem. Stop engineering the wording.

03 Use voice.

Speaking naturally produces far richer context than typing, because typing makes you compress.

04 Think in terms of tools and agents.

AI becomes dramatically more valuable when it can take actions, not merely answer questions.

05 Automate incrementally.

Read → summarise → classify → update → recommend → execute. Prove each layer before adding the next.

06 Start simple.

Use existing integrations before building custom infrastructure. Forwarding rules before servers.

07 Domain expertise now beats coding expertise.

AI increasingly lets people who deeply understand a business build custom software for it.

08 Protect your attention.

One of the highest-value uses of AI is filtering information before it ever reaches you.

09 Match supervision to risk.

Automate low-risk work aggressively. Keep human control over anything expensive or irreversible.

10 Learn through your real business.

Build actual workflows, find the limitations, improve them. That beats any course.

APPENDIX

Glossary

Every term used in this chapter, in plain language. Product names will change; these concepts won't.

Agent

An AI model that can take actions using tools on your behalf, rather than only writing text back to you.

Agent loop

The pattern where an AI attempts work, checks its own result, corrects it, and repeats until the goal is met — without a human prompt at each step.

API

Application Programming Interface. The channel one piece of software uses to request something from another piece of software, with no human clicking involved.

Application

The interface you interact with — the app or website. Distinct from the model underneath it.

Back end

The part of a software system that handles data, calculations, business rules, authentication, and integrations. Users never see it.

Cloud agent

An agent running on a remote server, so it can keep working while your own computer is off.

Computer use

An AI capability for operating software interfaces the way a person does — opening browsers, clicking, filling forms, navigating pages.

Connector / plugin

A pre-built link that lets an AI system reach into another service such as Gmail, Google Sheets, or a calendar.

Context

The information you give an AI so it understands your situation: documents, emails, examples, rules, history, goals, constraints.

Frontier AI lab

A company building some of the most capable general-purpose AI models available — OpenAI, Anthropic, Google, xAI.

Front end

The part of a software system humans see and touch: buttons, forms, dashboards, pages.

Local agent

An agent running on your own computer, with access to your local files and software.

MCP

Model Context Protocol. A shared standard that lets AI systems discover and connect to external tools and data without a custom integration for every combination.

Model

The underlying AI system that generates the intelligence, as opposed to the interface you type into.

Orchestration

Using a strong, expensive model for difficult decisions and planning, then cheaper, faster models to execute the individual pieces.

Prompt engineering

The older practice of carefully structuring instructions with roles and templates. Largely superseded by simply supplying good context.

Reasoning effort

A setting controlling how much thinking a model does before answering. More effort for genuinely hard problems; default for everything else.

Workflow automation platform

Tools such as Zapier, Make, and n8n that connect applications and trigger chains of actions when something happens.



End of Chapter One

Session 2 picks up where your experiments break down. Bring the failures.

As further sessions are added, this document grows into a single reference book rather than a stack of meeting summaries — with the timeless concepts kept separate from the product-specific details, because product names and features will change far faster than the underlying knowledge.

Ilia Reingold

AI consulting for small business – Toronto & the GTA

iliareingold.com