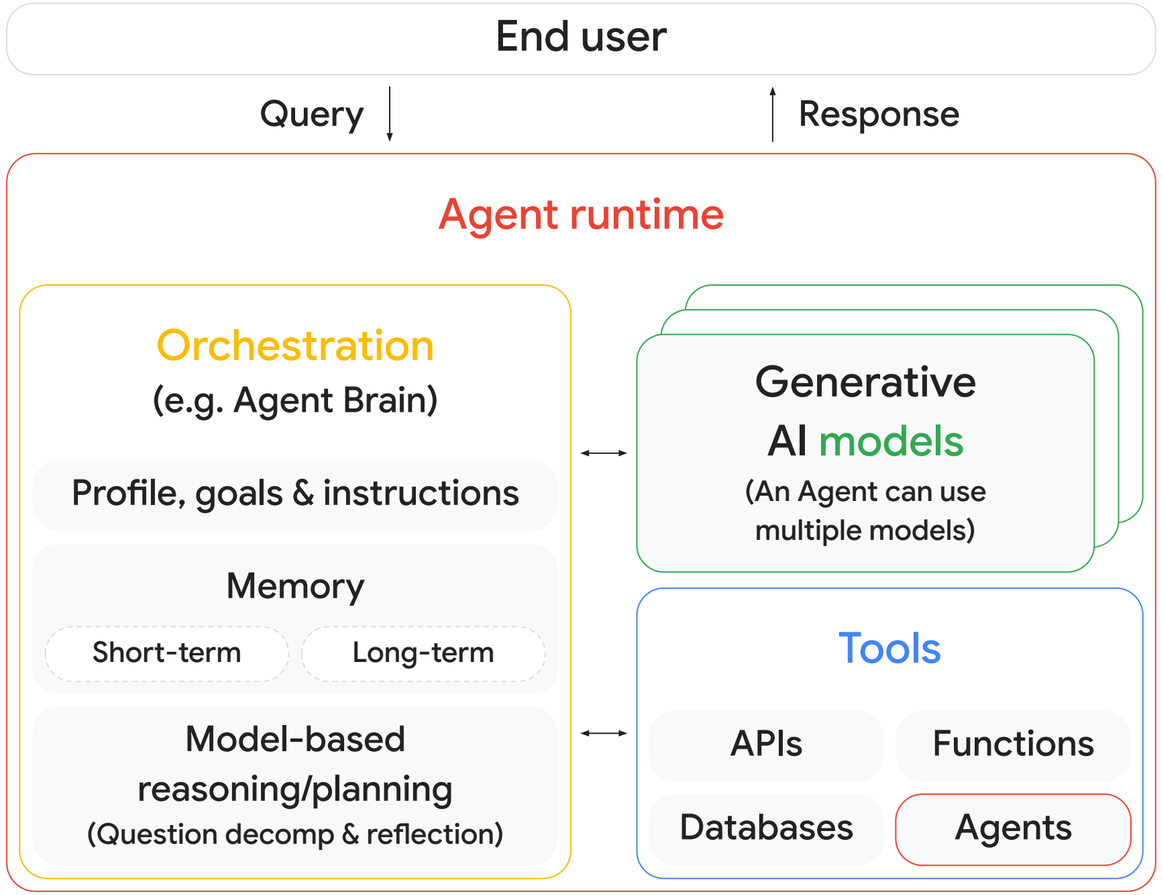
Building AI Agents That Work

Lessons Learnt

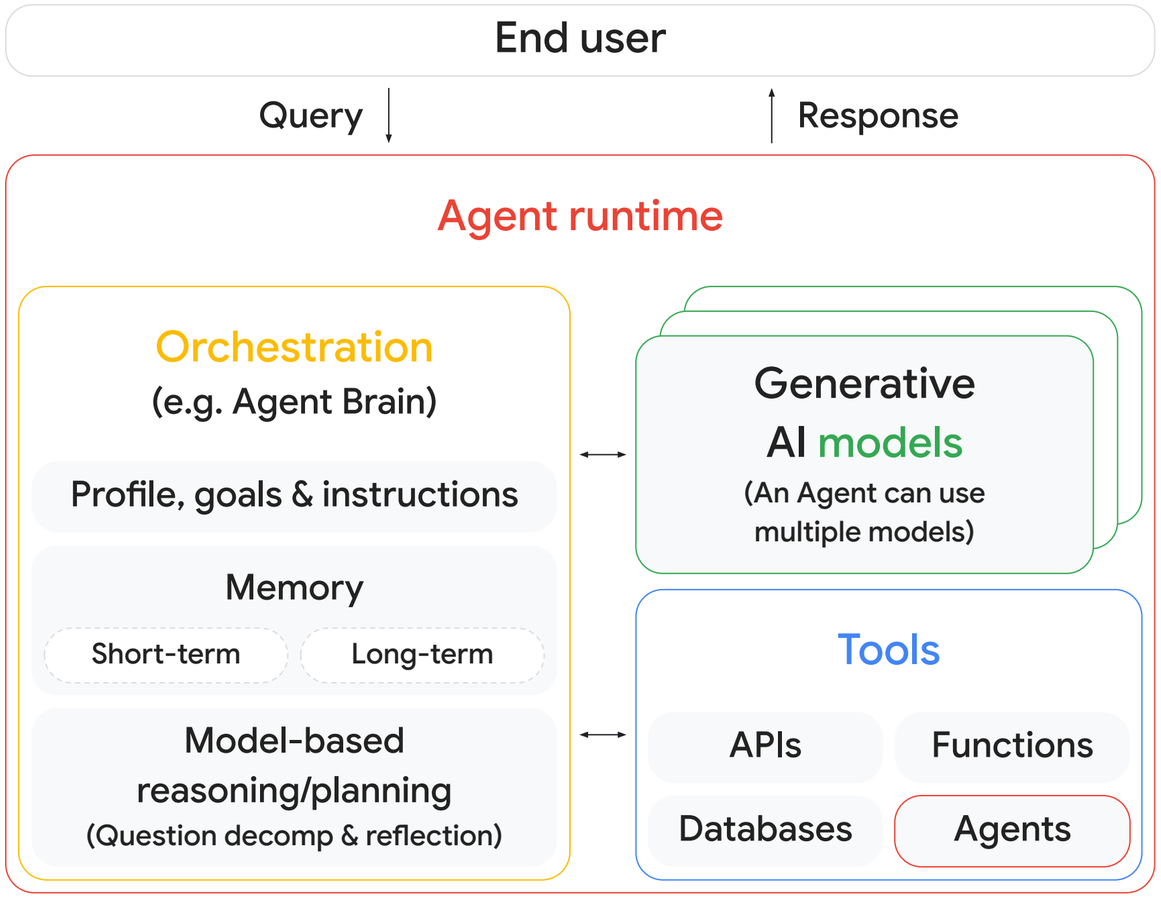
Google Cloud Next 25

Contents

What is an Agent?



What is an Agent?



An agent is an AI-driven application that uses a model or LLM, guided by instructions that shape its behavior. It also has access to additional tools that enhance its capabilities, all running within a dynamic runtime environment.

Lessons

01 Don’t use Agents!

02

03

04

Ask yourself...

Predictability	How predictable is your task structure?

Ask yourself...

Predictability	How predictable is your task structure?
Control	How critical is consistency and control in your application?

Ask yourself...

Predictability	How predictable is your task structure?
Control	How critical is consistency and control in your application?
Boundaries & Complexity	How clearly defined are your boundaries?

Ask yourself...

Predictability	How predictable is your task structure?
Control	How critical is consistency and control in your application?
Boundaries & Complexity	How clearly defined are your boundaries?
Latency & cost	What's your tolerance for latency and cost?

Agents vs. Workflows

End user

Query

Response

Agent runtime

Orchestration
(e.g. Agent Brain)

Profile, goals & instructions

Memory

Short-term

Long-term

Model-based reasoning/planning
(Question decomp & reflection)

Generative AI models
(An Agent can use multiple models)

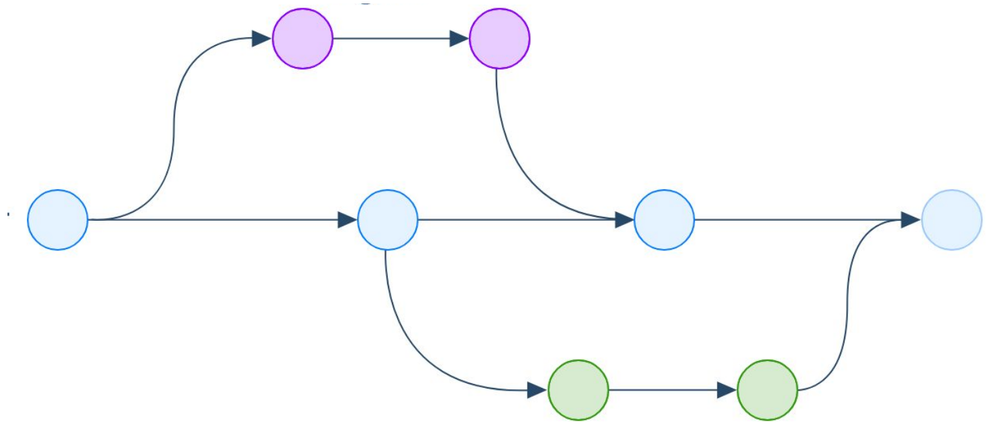
Tools

APIs

Functions

Databases

Agents



LLM + Instructions + Tools

Lessons

01 Don’t use Agents!

02 Frameworks

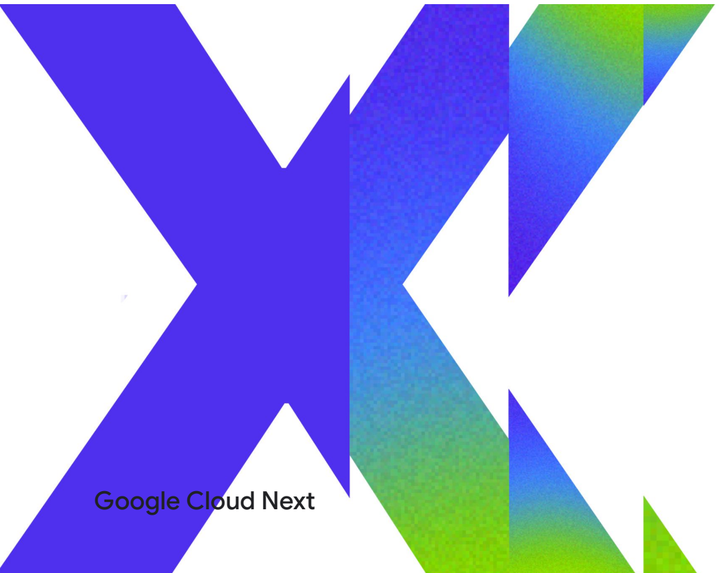
03

04

Frameworks & Abstractions

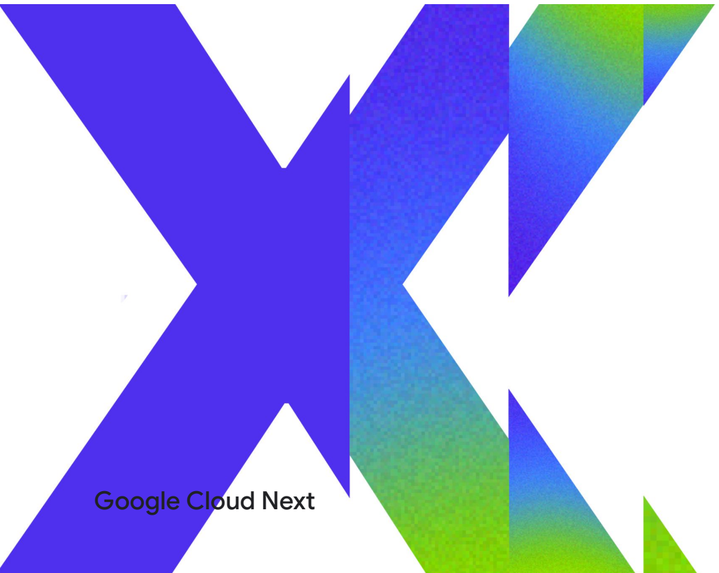
Quick PoC

They enable rapid prototyping, a great start but only for the proof of concept



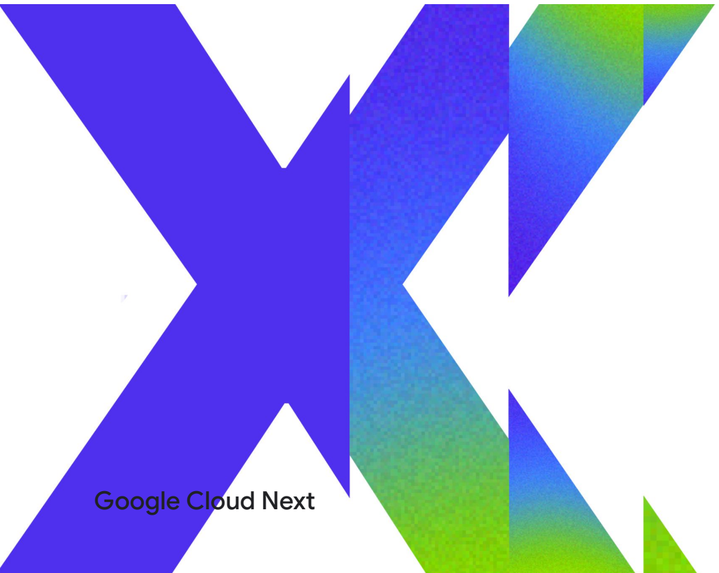
Frameworks & Abstractions

Quick PoC	They enable rapid prototyping, a great start but only for the proof of concept
Hidden Details	Usually hides the inner working and primitives being used by the AI agents, leading to lack of control



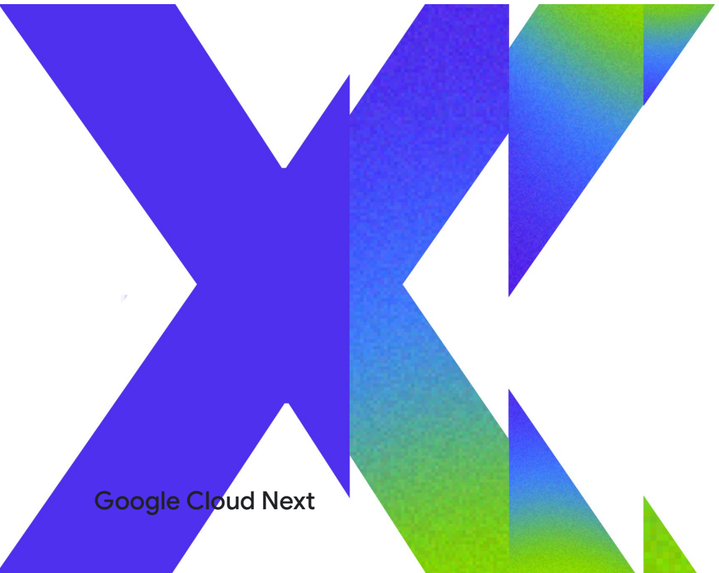
Frameworks & Abstractions

Quick PoC	They enable rapid prototyping, a great start but only for the proof of concept
Hidden Details	Usually hides the inner working and primitives being used by the AI agents, leading to lack of control
Defer Decisions	Frameworks force design choices before you have identified your specific limitations and goals of system.



Frameworks & Abstractions

Quick PoC	They enable rapid prototyping, a great start but only for the proof of concept
Hidden Details	Usually hides the inner working and primitives being used by the AI agents, leading to lack of control
Defer Decisions	Frameworks force design choices before you have identified your specific limitations and goals of system.
Suboptimal Systems	May lead to sub-optimal performance



Lessons

01 Don’t use Agents!

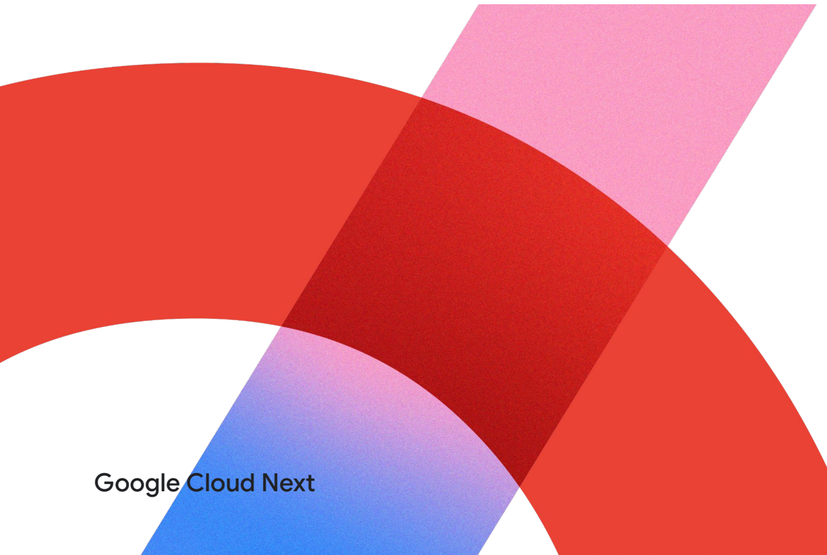
02 Frameworks

03 Start Simple!

04

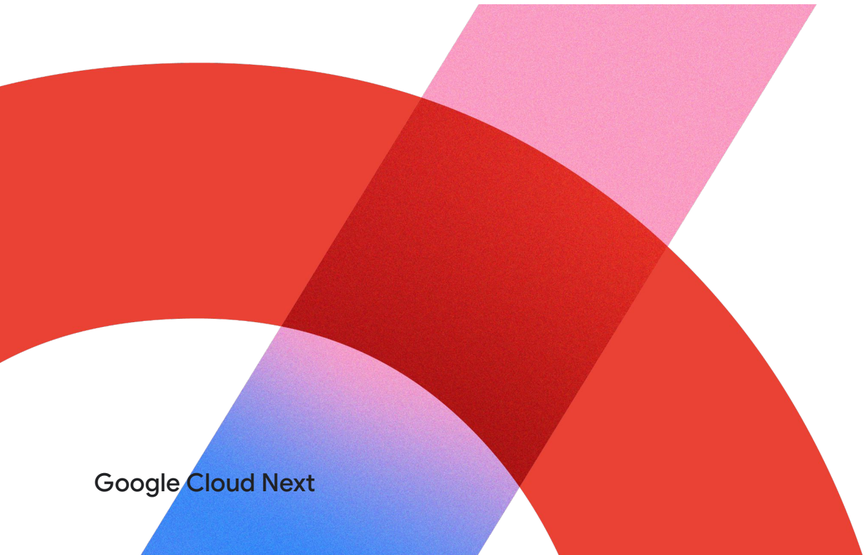
Start Simple!

Single Agent first	Build a focused agent for one specific task



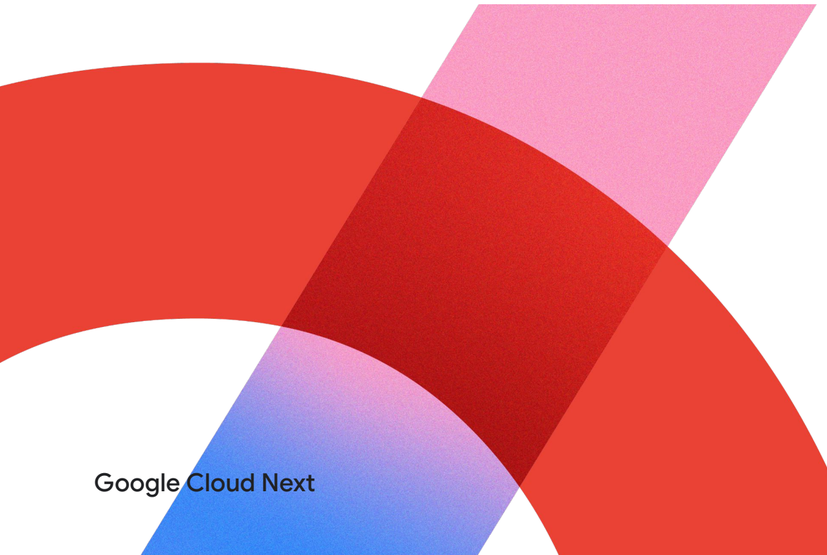
Start Simple!

Single Agent first	Build a focused agent for one specific task
Observe	Learn from real usage to see which functionalities work and which don’t!



Start Simple!

Single Agent first	Build a focused agent for one specific task
Observe	Learn from real usage to see which functionalities work and which don’t!
Prefer functionality	Functionality over complexity. Develop a working system, prioritize usefulness, not over-engineering



Lessons

01 Don’t use Agents!

02 Frameworks

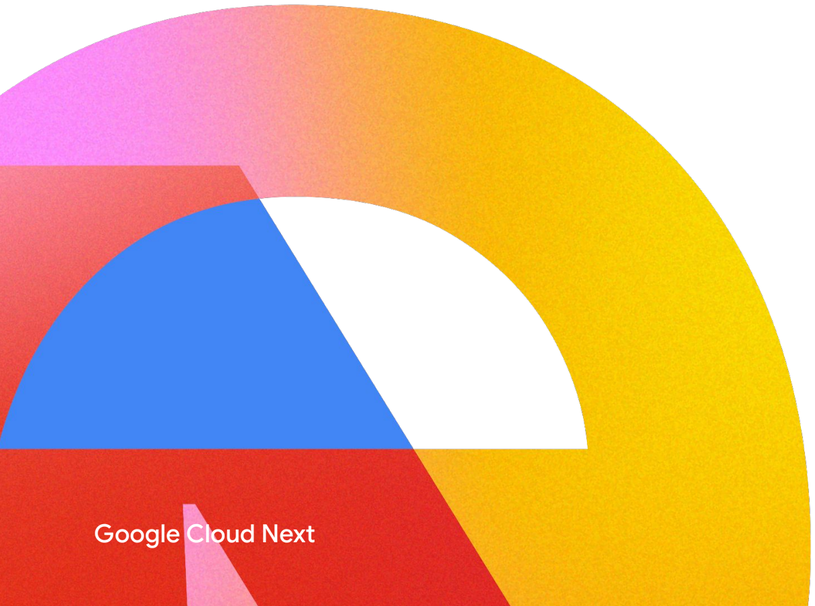
03 Start Simple!

04 Prompting

Prompts...

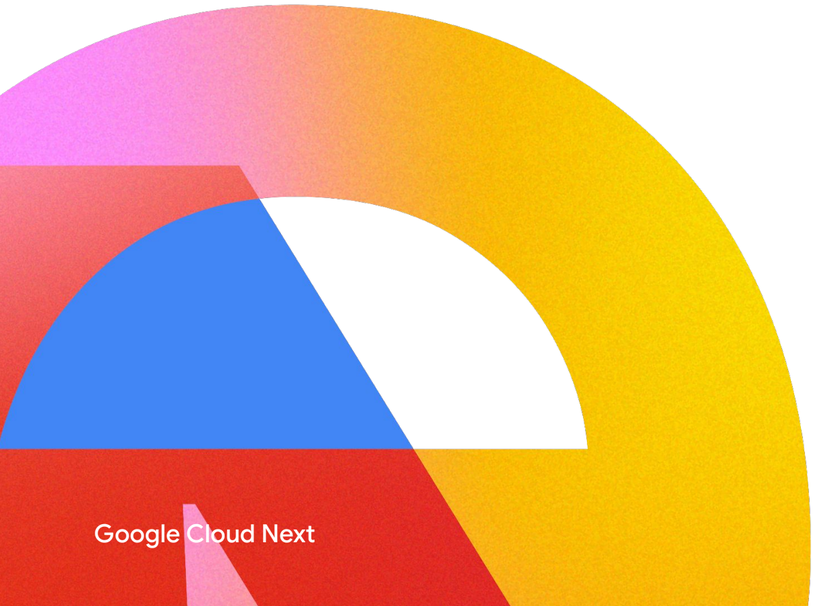
Clarity in Prompts

Prioritize clarity in prompts that the model can understand.



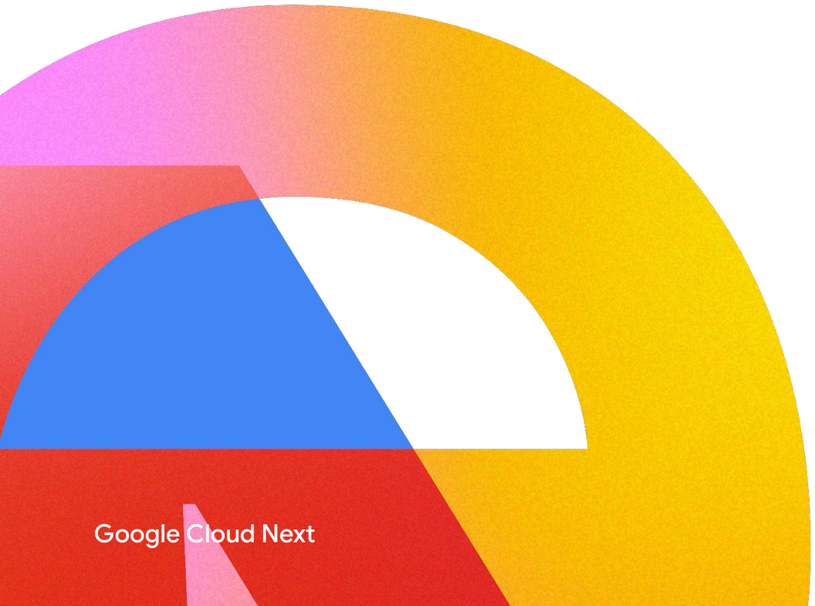
Prompts...

Clarity in Prompts	Prioritize clarity in prompts that the model can understand.
Don’t forget tools	Writing a good tool description impacts the overall performance of the agent.



Prompts...

Clarity in Prompts	Prioritize clarity in prompts that the model can understand.
Don’t forget tools	Writing a good tool description impacts the overall performance of the agent.
Iterative Refinement	Just as in software development, testing and feedback are crucial.



Lessons

01 Don’t use Agents!

02 Frameworks

03 Start Simple!

04 Prompting

Thank you!

@engineerprompt @engineerrprompt

engineerprompt.ai