

Agent Primitives — Field Guide

Fora Soft Learn · AI for Video Engineering

1 — The three primitives (give a model all three and it becomes an agent)

Tool use	Reach the world outside the model's text: run a detector, transcribe, query the archive.
Memory	Keep and recall facts across steps and sessions. The context window is only working memory.
Planning	Turn a goal into ordered steps; decompose a big task into small actions to run.

2 — Anatomy of a tool call (function calling)

Step	What happens
1. DESCRIBE	each tool = name + description + JSON Schema of its inputs
2. FILL	model picks a tool and fills in the arguments (does NOT run it)
3. EXECUTE	your runtime validates the form and runs the real function
4. OBSERVE	the result returns to the model as the next observation
MCP	one open standard wires tools in — discovery, schema, invocation (Nov 2024)

3 — Four memory types (only working memory fits in the context window)

Type	Holds	Lives in	Scope
Working	the current task + last steps	→ context window	(this task only)
Episodic	specific past events	→ external store / DB	(across sessions)
Semantic	general facts & knowledge	→ searchable database	(long-lived)
Procedural	how to do the task	→ system prompt/weights	(persistent)

4 — Three planning patterns (pick by interactive / repeatable / high-stakes)

Pattern	How it runs	Use for
ReAct	think → act → observe, one step at a time	interactive (live investigation)
Plan-and-execute	plan all steps up front, then run them	repeatable (overnight batch)
Reflection	produce → self-critique → retry	high-stakes (compliance report)

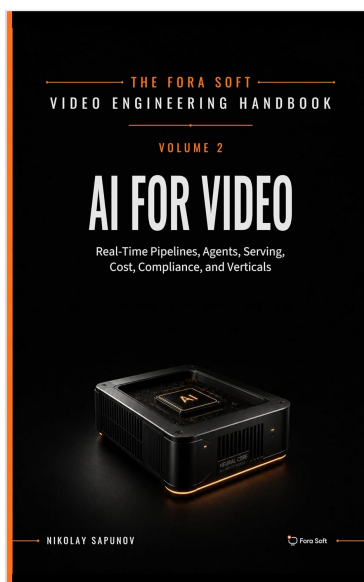
5 — Keep it simple: add a primitive only when the task needs it

- ☐ Fits in one context window? Use no external memory — do not stand up a vector DB.
- ☐ Steps never change? Write them as code (a workflow), not an open-ended planner.
- ☐ Calls one service once? Give the agent one tool, not a tool framework.
- ☐ Too many tools hurts: more tools = worse tool selection. Start with the smallest set.
- ☐ Do not stuff the window: it costs money per token and buries the facts that matter.

Companion to: 'Tool Use, Memory, Planning — The Agent Primitives'

www.forasoft.com/learn/ai-for-video-engineering/articles-ai/tool-use-memory-planning-agent-primitives

Fora Soft · www.forasoft.com



GO DEEPER · VOLUME 7 OF 7

AI for Video: A Complete Guide to Intelligence in Video Products

From Real-Time Pipelines and Agents to Serving, Cost, Compliance, and Verticals

By Nikolay Sapunov, CEO at Fora Soft · 1,221 pages · Kindle & paperback on Amazon

[Read it on Amazon →](#)
amazon.com/dp/B0GX3DGBH4

This guide lays out tool use, memory and planning and when each one earns its place. Volume 7 treats them as engineering: loop primitives, frameworks, memory and tools, plus where an agent plugs into a WebRTC pipeline and what it costs to serve.

WHAT YOU WILL FIND INSIDE

- Tool calling, schemas and MCP in practice
- Working, episodic and semantic memory design
- Plugging agents into a WebRTC pipeline
- Serving and cost per stream-hour

THE FULL HANDBOOK · 7 VOLUMES · 9,000+ PAGES · ALL ON AMAZON

Vol. 1 · Video Encoding — pixels and color to H.264, HEVC, AV1, quality metrics and AI

Vol. 2 · Video Streaming I — network transport, contribution and ingest, HLS, MPEG-DASH, CMAF

Vol. 3 · Video Streaming II — ABR, CDNs, players, WebRTC, DRM and the economics of delivery

Vol. 4 · Audio for Video I — PCM, codecs, loudness, accessibility, real-time voice processing

Vol. 5 · Audio for Video II — jitter buffers, lip sync, Atmos, quality metrics, neural codecs

Vol. 6 · AI for Video I — models and inference, computer vision, speech, search, generative video

Vol. 7 · AI for Video II — real-time pipelines, agents, serving, cost, compliance, verticals

Designing an agent's tools, memory and planning for a video product?

We will review your agent design, cut what the task does not need and plan the tool and memory layers. 250+ video products shipped since 2005.

[Book a 30-min call →](#)
[WhatsApp →](#)
[Email us →](#)