Hallucination Considerations

Avoiding architectures that tend to compound hallucinations

- In multilayered systems, hallucinations compound. One layer's mistake becomes the next layer's input. If your reasoning layer hallucinates a fact and your content layer writes it beautifully, you've just produced confidently wrong output. The more layers, the more opportunities for errors to slip through and get amplified
- Correction layers should come before memory consolidation. If you don't catch errors before they enter your system's permanent memory, those minor mistakes slip into history and slowly expand. They reintroduce themselves ad infinitum, compounding with each cycle until your system's "source of truth" is corrupted

Major Takeaways

What to remember

- Multilayered architectures compound transformations. Each layer takes input, transforms it, and passes it forward. The power comes from coordinating these transformations to achieve results no single prompt could accomplish

- Layer types provide a vocabulary for building. Correction, reasoning, memory consolidation, content, and catch-all layers each serve distinct purposes. Understanding these patterns helps you architect intentionally rather than intuitively

- Cyclical systems run the same flow every time. Circumstantial systems adapt their pathway based on outcomes. Most production systems end up somewhere in-between

- Backend integration handles what LLMs can't. True randomness, deterministic calculations, unbiased judgment, and data persistence belong outside the AI layers