What is a Multilayered AI Architecture?

And how can it supercharge the power of your app?

- If one prompt takes an input, runs it through a series of filters and rules and fundamentally transforms the input data then outputs it to you, multiple coordinated prompts compound those transformations substantially

- The separation of concerns between AI layers not only makes these systems more manageable, but it helps avoid confusing the AI with a million tasks, bolstering performance for the most important functionalities

- For systems that are aiming for near-perfect performance or just more consistent results

An Example from Emstrata

The Emstrata Cycle

- The Emstrata Cycle is a standardized series of prompts that run of every turn in an Emstrata simulation
- This cycle retains a comprehensive memory of all entities in the simulation, plans/positions entities on an interactive coordinate plane, writes prose according to exacting instruction, captures secrets and memories, and corrects all continuity errors after the narrative is written.
- No single prompt or backend wizardry would be able to accomplish this by itself
- These are the layers (simplified for the example):
    - Groundskeeper (system memory)
    - Discovery (planning/consequence handling)
    - Narration (writing the narrative)
    - Chron-Con (correcting any minor errors)