

Agnostic Backend Interaction

What happens between AI layers

- Between AI layers, it's important to save important, transformed data to the backend for future retrieval, debugging, rerunning if there's an error, etc.
- Saving data also allows you to present that data in interesting ways later or feed that data into other layers in the future
- Also, when you need an unbiased judge, the backend is the place to go. The backend is 'agnostic' to outcome, whereas the AI may or may not have a strong preference and display it
- In Emstrata, consequences are rolled and use weighted randomness. The Discovery layer determines the likelihood something happens, and then the backend returns a random number out of 1000. If that number is within the set likelihood range, the backend serves the confirmed consequence to the Narration layer, if it's outside of the range, it sends the failure outcome

Randomness Injection

A jolt of creativity

- If you grow tired of tropes and clichés in your responses, I have an answer: Random Concept Injection
- This is something I do for parts of the system that do creative heavy-lifting. Oftentimes, AIs will hop to really tried and true answers to creative questions, which is great for reasoning well, but not so much for surprising an audience
- I use this to get names for characters that aren't baked into the training data, inject interesting concepts into simulations, and build out characters based on character archetypes
- It can be used for any list of random strings you'd like to be potentially incorporated into a particular decision-making process