[RF-2a]_DD-AI_Perspective.md

🧠 The Producer.ai Signal Flow: The Three Layers

1. The Translator Layer (The LLM Interface)

The AI interface acts as a **Prompt Engineer** before calling the API. It processes user input and translates adjectives into numerical and structural parameters.

- **Intent Parsing:** Converts "Make it heavier" into `weirdness: 0.8 + text: "distorted, industrial, aggressive"` .
- **Prompt Expansion:** Takes simple user tags (e.g., "Grime") and expands them into rich keyword strings (e.g., "UK Grime instrumental, 140 bpm, eski-beat style...").
- **Implicit Key/BPM:** If not specified, the AI guesses the key and tempo based on the prompt or reference audio, which can lead to unwanted results if the guess is wrong.

2. The Diffusion Backend (The Black Box)

The audio engine operates on a **Latent Diffusion Model** (FUZZ-2.0), not a conventional DAW structure.

- **Spectrogram "Painting" vs. Sampling:** The model does not think in "stems" or "tracks." It generates the entire frequency spectrum (spectrogram) at once by adding and then "denoising" noise based on the text prompt.
- **The Failure Mode (Cover Strength > 0.3):** When remixing, if `cover_strength` is too high, the model sees a shape in the spectrogram (e.g., a guitar waveform) and attempts to **re-synthesize** it according to the *new* genre prompt (e.g., Grime). This results in the original organic texture being replaced by a synthesized texture (e.g., "synth-smearing").
- **The Fix:** Use `riff_split_stems_v2` first to isolate audio components *before* running the remix tool, allowing you to selectively process only the desired stems (e.g., only Drums and Bass).

3. Temporal Masking & Vibe Injection (The Complex Hack)

The `sound_prompts` array provides the most granular control over generation.

- **Temporal Masking (`time_start` / `time_end`):** Defines a timeline where the AI must switch between stylistic directives. The diffusion model is forced to mathematically *blend* the end of one style's spectrogram into the start of the next at the precise time marker.
- **Vibe Injection (`vibe: {audio_input: {id}}`):** Injecting the "sonic signature" of a source audio file (UUID) into a new generation. When combined with `text: null` and a low `strength` (e.g., 0.3), the model focuses on borrowing the *texture*, *air*, *mix quality*, or *spectral balance* of the reference audio, not the core melody.