Loompa — Five Threads for the Drive

DUNIN7 — 2026-04-28 — Thinking material, not conclusions

1. What does FORAY actually do, and does a Shape need the same thing?

FORAY gives all business transactions a universal grammar — parties, obligations, conditions, timelines, assets — regardless of domain. Do all specifications share structural elements the same way? A floor-plan spec, a clinical protocol, an application spec, a musical score — is there a common structure underneath? Candidates: **what is being produced**, **what it must contain**, **what constraints apply**, **what knowledge informs it**, **who it serves**, **what form it takes**, **what counts as faithful production**. Are those universal? What's missing?

2. Where does Loompa sit relative to Loom Protocol?

Loom records assertions (five transitions, provenance, no erasure). FORAY describes transactions. Loompa would describe specifications. Three positions:

A — Extension: Loompa adds specification-aware transitions to Loom's five. **B — Sibling:** peer protocol alongside Loom and FORAY, referencing but not nesting. **C — Layer on top:** consumes Loom assertions, adds its own grammar for how they compose into specifications.

The test: if someone implements Loom without Loompa, is their system incomplete? If yes → extension. If no → sibling or layer.

3. The Oompa Loompa insight

In Wonka's factory, the Oompa Loompas read recipes and produce candy. They're skilled in production, not in recipe creation. The recipe is complete — the Oompa Loompa follows it faithfully. Different Oompa Loompas work different machines but read the same recipe format. The deeper insight: **the factory is the protocol, not the candy.** Wonka didn't build a chocolate company — he built a production system. DUNIN7 didn't build a document company — it built a specification-to-production protocol. The Loompa protocol is the recipe language. The specialists are the Oompa Loompas. Loomworks is the factory.

Does this metaphor hold? Where does it break? What does the factory know that the individual Oompa Loompa doesn't?

4. What makes a specification grammar universal?

FORAY's universality: all transactions share structural elements combined differently. For Loompa, specifications across all domains would need the same. Candidate elements:

Subject — what is being specified. **Components** — parts and their relationships. **Properties** — measurable attributes of each component. **Constraints** — rules the spec must satisfy. **Provenance** — which assertions inform each element. **Completeness criteria** — what must be present. **Consumer** — who receives the artifact. **Medium hints** — what the spec knows about how it will be produced.

Is this list universal? When an architect writes a spec and a composer writes a score — are they doing the same structural thing?

5. Does the Shape grammar pre-exist or emerge?

FORAY's grammar was conceived from observation — the structural commonality in transactions pre-existed; FORAY named it. Does a specification grammar pre-exist the same way? If yes, Loompa discovers and names it. If no — if specifications are genuinely domain-specific in structure — then Loompa is a framework for describing domain-specific grammars. A grammar for grammars.

FORAY's evidence says look harder. Forty-five years between the insight and the description. The specification grammar might be equally non-obvious and equally real.