

RFC: PSL-C Language Specification — Part I (Sections 1–10)

Author: Zach Bogart — PromptBuilder.us / PB3K Project Lead

1. Overview

PSL-C (Prompt Shorthand Language — Cinematic) is a domain specific language designed to compactly encode cinematic instructions, audiovisual attributes, structural intent, and environmental behavior for AI-driven video generation systems. PSL-C allows creators to express scene-level constraints, motion primitives, lighting grammars, and semantic cues using deterministic shorthand tokens that map reliably to model behavior regardless of underlying architecture.

2. Motivations

Video generative systems require precise, structured guidance beyond natural-language prompts. Traditional prose prompts introduce ambiguity, formatting inconsistencies, and model-dependent interpretation. PSL-C provides a stable, minimal, and extensible shorthand that reduces ambiguity by establishing token families, standardized modifiers, and predictable compositional behavior. Its purpose is to enable reproducible video sequences, improved model alignment, script-to-video workflows, and cross model compatibility.

3. Design Goals

PSL-C is built around six primary design goals:

1. Compactness — expressing complex cinematic intent in minimal characters.
2. Determinism — ensuring token meaning is consistent across models.
3. Domain-Oriented Semantics — mapping directly to cinematography, physics, lighting, audio, and motion.
4. Extensibility — enabling future categories without breaking backward compatibility.
5. Interoperability — functioning across multiple model APIs and interpreters.
6. Human Readability — retaining clarity for animators, filmmakers, and developers.

4. Scope

This specification defines:

- PSL-C syntax, grammar, morphology, and token families
- Category structures, root-modifier logic, and canonical token lists
- Interpretive semantics and model-agnostic mapping behaviors
- Compliance levels for interpreters
- Reserved namespaces and versioning systems

This RFC does NOT define:

- Specific model embedding formats
- Proprietary weight mappings
- Renderer-specific implementations

The goal is a universal, open standard.

5. Definitions

Key terms used in the PSL-C specification:

Token: The smallest meaningful unit in PSL-C (e.g., rim-hard, pan-slow).

Root: A category-specific semantic base term (e.g., rim, pan, pace).

Modifier: A refining attribute applied to a root (e.g., slow, soft, wide).

Category: A semantic grouping of tokens (e.g., LIGHT, CAM, MOT-H).

Block: A collection of tokens representing a major cinematic subsystem.

Interpreter: Software that parses PSL-C and maps tokens to generator parameters.

Compliance Level: A required minimum feature set for interpreting PSL-C.

Scene: A top-level compositional structure composed of one or more blocks.

6. PSL-C Syntax

PSL-C uses the following high-level syntactic rules:

- Tokens use the form ROOT-MOD.
- Tokens are case-insensitive (rim-hard == RIM-HARD).
- Blocks are unordered and free-form, but category grouping is recommended.
- Whitespace and line breaks have no semantic meaning.
- Punctuation is optional; commas may separate tokens.

Token Example:

rim-hard dolly-slow pace-nerv warm-amb

Block Example:

LIGHT: rim-hard bounce-soft

CAM: dolly-slow snap-zoom

MOT-A: pace-nerv freeze-short

COLOR: warm-amb desat-mild

7. Grammar

The PSL-C grammar follows this EBNF-like structure:

```
::= *
 ::= ":" 
 ::= IDENTIFIER
 ::= +
 ::= "-"
 ::= IDENTIFIER
 ::= IDENTIFIER
```

Categories may appear in any order. Scenes may contain any number of blocks.

8. Morphology

Token morphology is constructed from two components:

ROOT:

The semantic base describing the behavior class.

Examples: rim, dolly, pace, scatter, pulse, dusk

MODIFIER:

The refinement describing intensity, direction, timing, or style.

Examples: slow, hard, soft, wide, deep, micro, warm

Together they form deterministic shorthand units:

rim-hard pan-wide pace-nerv dusk-soft

Modifiers may be overloaded depending on category context; e.g., 'soft' means low contrast in LIGHT, but low intensity in MOT-H.

9. Category Definitions

PSL-C defines 17 primary categories:

STYLE — aesthetic mode or rendering profile
COLOR — color palette, grade, tonal behavior
LIGHT — lighting direction, quality, and interaction
CAM — camera movement and lens behavior
MOT-H — human motion primitives
MOT-A — animal motion primitives
PHYS — physics and material interactions
WEATH — weather and atmospheric conditions
AUDIO — sound design tokens
DIA — dialogue tone modifiers
TIME — timing structures and pacing behaviors
MAT — materials, textures, surfaces
VIBE — emotional, tonal, or narrative undertones
SHOT — compositional shot structure
ENV — environment interaction and micro-reactions
FX — special and visual effects
ANIM — animation timing and interpolation

10. Category Cardinality Rules

Each category has different expectations:

Required Categories: None

Recommended Categories for full scenes: STYLE, LIGHT, CAM, MOT-H, SHOT, VIBE

Optional Categories: DIA, MAT, FX, ENV

Interpreters MUST:

- Accept any subset of categories
- Process unknown categories safely
- Preserve order-insensitivity
- Support multi-token blocks

This ensures backward compatibility as PSL-C evolves.