Praxeological Meta-Structure (PMS) – Model Specification

A structural operator grammar for praxis, asymmetry, development and self-binding

Version: 1.1 · Spec basis: `PMS.yaml`
Author: T. Zöller · Formalisation assistance: ChatGPT (GPT-5.1 Thinking)
Language: EN · Status: Model spec (aligned with `schema_meta.status = "draft"`)

---

1. Purpose and scope of this specification

This document specifies the *Praxeological Meta-Structure* (PMS) model in a concise, technical form. It is based on the YAML file `PMS.yaml` (with `schema_version = "PMS_1.1"`) and makes its structure, concepts and guardrails transparent for human readers and software systems.

PMS is a **meta-model**: it defines a small, irreducible set of structural operators (Δ–Ψ) and describes how complex forms of praxis (awareness, coherence, responsibility, action, dignity-in-practice, structural IA-patterns) arise from **operator compositions**. It is **non-physical, non-metaphysical and non-psychological** and does not contain a case schema or scoring system.

The specification covers, in particular:

- the **schema_meta** block (model name, status, intended use, normative position, tragedy clause, dignity clause);
- the **meta-axioms Δ–Ψ** with order, definitions, dependency relations and operator layers (L1–L4);
- the **derived structures** (Awareness, Coherence, Responsibility, Action, Dignity-in-Practice, IA-patterns, self-model fixpoint);
- **example operator chains** for minimal praxis, pattern formation, asymmetry and self-binding;
- the **AI interface** (welcome message, modes, guardrails and suggested questions) for safe integration into agent architectures.

The intended use of this specification is threefold:

- as a **reference for theorists** in philosophy, anthropology, social theory, systems theory and AI;
- as a **technical artefact** that can be cited, critiqued and extended in research and teaching;
- as a **governance layer for software and AI systems**, where the YAML file acts as a single source of truth for the structural grammar of praxis.

> **Core idea**
> PMS does not analyse persons, traits or mental states. It defines a structural operator grammar for praxis. Other models – such as praxeological action models (ACRPD / IA) – MAY be derived from PMS but are **not part** of this specification.

2. High-level structure of the YAML model

2.1 Top-level keys

Key	Description	Role in the model
`schema_version`	Version string identifying the release of the YAML schema (`"PMS_1.1"`). Ensures compatibility across updates and allows unambiguous citation.	Versioning / compatibility
`schema_meta`	Contains name, status, authors, description, intended uses and explicit governance constraints (`intended_use` with its `allowed_profiles` , `not_intended_for` , `normative_position` with dignity clause and tragedy clause, and the `terminology` block).	Meta-information / governance / normative stance
`core_principles`	Enumerates the core principles of the PMS model (praxeological focus, operator minimality, generativity, temporal and asymmetry awareness, non-diagnostic use, scenic/context-bound interpretation).	Operational guardrails for all applications
`pms_model_reference`	Contains the core structural definition of PMS: meta-axioms Δ–Ψ with order and dependencies, operator layers (L1–L4) and the dependency table.	Normative model: operators, layers, dependency graph
`derived_structures`	Defines higher-level constructs that are derived from Δ–Ψ: Awareness (A), Coherence (C), Responsibility (R), Action (E), Dignity-in-Practice (D), the self-model fixpoint and IA-patterns.	Derived grammar: axes, patterns, self-model
`example_operator_chains`	Provides canonical examples of operator compositions (minimal praxis, pattern formation, asymmetry emergence, developmental trajectories, reflexive praxis, self-binding chain).	Illustrative examples and templates
`ai_interface_pms`	Specifies how interactive agents should use PMS: onboarding text, usage modes, general and AI-specific guardrails, and suggested structural questions.	Agent interface / safety and interaction layer

2.2 Conceptual separation

`pms_model_reference`

This block contains the complete structural definition of the PMS operator grammar:

- the eleven meta-axioms Δ–Ψ with order and dependencies;
- the four operator layers L1–L4 (ontological patterning, relational asymmetry & temporality, meta-structural reflexivity, self-binding fixpoint);
- the dependency table recording which axiom presupposes which others.

In short: **this block defines the operator system itself**.

`derived_structures`

This block contains derived constructs that are **computed** from Δ–Ψ:

- the derived axes A, C, R, E, D with explicit formulas and generative reasoning;
- the self-model fixpoint sequence;
- IA-patterns such as `IA_A>>E` and under-integration patterns.

In short: this block defines **what PMS can generate structurally**, without introducing case schemas or scores.

2.3 Terminology (from `schema_meta.terminology`)

The YAML defines three core terms that are used throughout this specification:

- **Praxis** – situated, meaningful action under asymmetry, constraints, expectations, temporal extension and self-interpretation; modelled structurally via operator compositions, not via subjective experience.
- **Meta-axiom** – an irreducible structural operator (Δ–Ψ) that cannot be derived from any other operator and participates in the generative grammar of praxis.
- **Operator composition** – the ordered application of meta-axioms to generate complex praxeological structures such as patterns, roles, trajectories, integration and self-binding.

3. Core rules and guardrails

3.1 Core principles

The `core_principles` block defines non-negotiable rules for all applications of PMS:

- **P1 – Praxeological focus:** PMS analyses enactments, roles and structures of praxis, not persons as ontological or psychological entities.
- **P2 – Operator minimality:** Δ–Ψ are treated as a minimal and complete set of structural operators; each has explicit dependency relations.
- **P3 – Generativity:** complex forms of praxis are derived from operator compositions, not introduced as independent constructs.
- **P4 – Temporal and asymmetry awareness:** asymmetry (Ω) and temporality (Θ) are foundational; responsibility, maturity and dignity are always analysed under asymmetric and temporal conditions.
- **P5 – Non-diagnostic use:** PMS is not a diagnostic tool and must not be used for clinical, therapeutic or forensic decisions.
- **P6 – Scenic, context-bound interpretation:** all readings remain tied to concrete scenes, roles and structures. Global person labels are out of scope.

3.2 Normative position (ontology, dignity, tragedy)

The `schema_meta.normative_position` block specifies the ontological stance of PMS:

- PMS is **non-physical, non-metaphysical and non-psychological**. It makes no claims about consciousness, qualia or inner experience.
- It formalises **structural conditions of praxis**, not mental states.
- The **dignity clause** states that ontological dignity is never evaluated; Dignity-in-Practice (D) is a structural parameter about enacted restraint and respect in asymmetry, not about human ranking.
- The **tragedy clause** states that structural maturity and responsibility do not eliminate tragedy; PMS makes explicit how asymmetry, non-event and temporality create unavoidable tragic tensions in praxis.

3.3 Intended and forbidden uses

Under `schema_meta.intended_use`, PMS is declared suitable for:

- theoretical reflection and structural action theory;
- anthropology and praxeology;
- systems theory and AI architecture / safety design;
- model documentation and specification.

In the YAML, these uses are encoded as `allowed_profiles`: `theoretical_reflection`, `structural_action_theory`, `anthropology_and_praxeology`, `systems_theory`, `ai_architecture_and_safety` and `model_documentation_and_specification`.

Under `not_intended_for`, the YAML explicitly forbids:

- clinical diagnosis or personality typing;
- mental health risk assessment;
- automated moral judgement or individual person evaluation.

Misuse hint
Any attempt to use PMS as a diagnostic or moral machine constitutes a structural misuse of the model itself. Such misuse can be analysed as an inadult asymmetry at the level of model governance.

4. Meta-axioms Δ–Ψ and operator layers

4.1 Meta-axioms

The block `pms_model_reference.meta_axioms` lists the eleven meta-axioms Δ–Ψ, each with:

- **id** (symbol), **name**, **order**,
- **definition**, **depends_on**, **provides**,
- short **examples** from praxis.

Axioms 1–11 form a **non-interchangeable sequence**; each operator presupposes those that come before it and enables those that follow.

4.2 Operator layers (L1–L4)

The `operator_layers` block groups the axioms into four layers:

- **L1 – Ontological patterning**
  (`["Δ","∇","□","Λ","A"]`):
  difference, impulse, frame, non-event and attractor formation.
- **L2 – Relational asymmetry and temporality**
  (`["Ω","Θ"]`):
  asymmetry and temporal trajectories.
- **L3 – Meta-structural reflexivity**
  (`["Φ","X","Σ"]`):
  recontextualization, distance and integration.
- **L4 – Self-binding fixpoint**
  (`["Ψ"]`):
  self-binding and identity as structural fixpoint.

4.3 Dependency table

The `dependency_table` redundantly encodes the logical order: each entry states `axiom`, its `order`, `depends_on` and what it `provides`. This allows software to validate operator chains, check completeness and reason about possible compositions.

5. Derived structures (A, C, R, E, D, IA, self-model)

5.1 Derived axes

The `derived_structures.derived_axes` block defines five structural axes as operator compositions:

Axis	Symbol	Formula	Short definition
Awareness	A	`["Θ","□","Δ"]`	Sustained, framed differentiation across time.
Coherence	C	`["Θ","Λ","□","∇"]`	Temporally stabilised structuring of impulse and expectation within a frame.
Responsibility	R	`["Ψ","Φ","Θ","Ω"]`	Self-binding orientation toward asymmetry across time and recontextualization.
Action / Enactment	E	`["Σ","Θ","∇"]`	Integrated realisation of directedness across time.
Dignity in practice	D	`["Ψ","X","Ω"]`	Self-bound reflective restraint and protection in asymmetrical relations.

For each axis, the YAML stores a short generative reasoning and notes that these constructs are **structural, not psychological**.

5.2 Self-model fixpoint

The `self_model` section defines the PMS self-model as a structural fixpoint:

```
Self = Ψ ∘ Σ ∘ X ∘ Φ ∘ Θ ∘ Ω ∘ A ∘ Λ ∘ □ ∘ ∇ ∘ Δ
```

The YAML records this as an ordered `formula_sequence` and lists implications (selfhood as result of praxis, identity as stable self-binding, no claims about subjectivity).

5.3 IA-patterns

The `ia_patterns` block defines structural distortion patterns (IA) as specific operator constellations, e.g.:

- `IA_A>>E` – excessive distance between Awareness and Enactment, with basis `["Ω","A","Φ"]`;
- `IA_Sigma_low` – fragmented integration, modelled as high Φ and low Σ.

Each pattern stores its structural basis, generative mechanism and effects on the derived axes. All IA-entries are explicitly labelled as **praxeological patterns, not person types**.

6. Example operator chains and AI interface

6.1 Example chains

The `example_operator_chains` section contains canonical sequences such as:

- **minimal_praxis**: `["□","∇","Δ"]`;
- **pattern_formation**: `["A","Λ","□","∇","Δ"]`;
- **reflexive_praxis**: `["Σ","X","Φ","Θ","Ω","A","Λ","□","∇","Δ"]`;
- **self_binding_chain**: full Ψ–Δ sequence.

These chains serve as templates for human reasoning and as test cases for software that loads and operates on the PMS grammar.

6.2 AI interface and guardrails

The `ai_interface_pms` block defines:

- a **welcome_message** that explains PMS to users;
- explicit **modes** for different uses:
  - `axiomatic` – axiomatic / theory mode (Δ–Ψ and their compositions);
  - `derived_axes_inspection` – mapping PMS structures to the five derived axes (A, C, R, E, D);
  - `ai_architecture` – using PMS operators as conceptual building blocks for agent design and safety, without anthropomorphising.
- **general guardrails** (no mental state or trait inference, no person ranking, structural focus on enactments and roles);
- **AI-specific guardrails** (no anthropomorphising; Ψ in machines refers to policy and constraint stability, not consciousness; clear separation between structural agency models and real human vulnerability);
- a list of **suggested questions** that guide structural reflection, such as:
  - "What structural operators are involved in this pattern of praxis?"
  - "Which asymmetries (Ω) and temporal trajectories (Θ) are visible?"
  - "Where do recontextualization (Φ), distance (X) and integration (Σ) appear or fail?"
  - "How would this configuration project onto the derived axes (Awareness, Coherence, Responsibility, Action, Dignity-in-Practice)?"

Together, these elements allow LLMs and other agents to load PMS as a **governance layer** and to keep applications within the intended structural scope.

7. Implementation notes and licensing

7.1 YAML file and integration

The official PMS YAML specification is provided as:

`PMS.yaml`

This file constitutes the **single source of truth** for:

- the PMS operator system (Δ–Ψ);
- the derived structures (A, C, R, E, D, IA, self-model);
- the AI interface and guardrails for structural use.

Implementations SHOULD consume this YAML directly rather than re-implementing parts manually. It is suitable for:

- formal reasoning about structural operator chains;
- simulation and analysis of praxeological patterns;
- AI systems that require transparent, non-psychological reasoning templates;
- research tools and teaching materials in structural action theory.

7.1.1 Recommended bootstrap for LLM-based agents (non-normative)

After loading the YAML, an LLM-based agent SHOULD run a short system prompt that:

- parses the YAML and activates the `ai_interface_pms` block;
- outputs the `welcome_message` to the user;
- applies all guardrails and suggested structural questions when using the model.

7.2 Citation and license

When referencing the Praxeological Meta-Structure (PMS), please cite both the theoretical paper and this model specification:

Primary reference:
T. Zöller (2025): *Towards a Praxeological Meta-Structure Theory.*

Technical reference:
PMS.yaml – Axiomatic Operator Schema
YAML Specification and Model Definition.