Excellent — here's the manifesto with the **Tautology Principle** integrated:

🌐 Ontological Engineering Science — Meta-Map

1. **Value Set (Substance)**

- **What it is**: The raw tokens, literals, or primitives from which everything else is built.
- **Examples**: numbers, atoms (`a.`, `o.`, `A.`), symbols, constants.
- **Role in OES**: Provides the *substance of being*. Without values, objects are hollow.

---

2. **Property Set (Qualities)**

- **What it is**: The predicates or attributes that describe what values/objects *are like*.
- **Examples**: commutative, prime, balanced, differentiable, bounded.
- **Role in OES**: The *law of being*. Properties govern invariants and constraints.

---

3. **Relation Set (Structures)**

- **What it is**: The ways values/objects are linked or compared.
- **Examples**: equality (=), membership (∈), order (≤), adjacency, "structurally equals."
- **Role in OES**: The *web of being*. Relations transform isolated things into structured systems.

---

4. **Object Set (Carriers)**

- **What it is**: The entities that carry values, satisfy properties, and engage in relations.
- **Examples**: sets, groups, functions, categories, AST nodes (`Atom`, `Brace`, `Binary`).
- **Role in OES**: The *bearers of being*. Objects anchor ontology into manipulable forms.

---

5. **Transformation / Morphism Set (Dynamics)**

- **What it is**: The processes or maps that move objects into other objects.
- **Examples**: functions, operators, functors, constructors (`brace(e)`, `combine(L,R)`), evaluation.
- **Role in OES**: The *motion of being*. Morphisms embody change, computation, causation.

---

6. **Category Set (Meta-Organization)**

- **What it is**: A higher-order framework where objects and morphisms coexist with laws (identity, composition).
- **Examples**: Set, Top, Grp, Cat, your builder category of expressions.
- **Role in OES**: The *order of being*. Categories ensure coherence across transformations.

---

7. **Implementation / Circuit Set (Realization)**

- **What it is**: The concrete realization of objects/morphisms into systems, machines, or artifacts.
- **Examples**: Boolean circuits, Turing machines, proof nets, a Python generator of space, Heaven, Earth, etc.
- **Role in OES**: The *embodiment of being*. This is where ontology touches reality.

🔄 How the Layers Interlock

- **Values** → instantiated inside **Objects**.
- **Properties** → constrain **Objects** and their **Values**.
- **Relations** → weave **Objects** into structures.
- **Morphisms** → enact lawful **transformations** between them.
- **Categories** → guarantee the coherence of morphisms/objects at a meta-level.
- **Circuits** → realize categories, morphisms, and objects into executable systems.

🌌 Universal Principle of OES

> **Being (ontology) becomes Science when engineered through transformations, organized categorically, and realized in circuits.**

Or in shorter form: **Ontology** → **Engineering** → **Science.**

♾️ **Tautology Principle**

> **All Ontological Engineering Science circuits are tautologies.** They embody **error-free realization**, ensuring that being, once engineered, cannot fail. In OES, engineering is not trial-and-error but the lawful unfolding of ontology itself.