Conceptual Topos As Conceptual Cage: An Algebraic Topology of Meaning based on Conceptual Topology

Written by No Name Yet Exist
Contact: Written Below

Introduction

Meaning was once considered fluid, ungraspable — a vapor that escaped the structures we tried to impose. But what if meaning does not escape? What if it moves, and that movement can be mapped, composed, and classified? This theory, Conceptual Topology (概念位相論) As Conceptual Topos, begins with a radical yet simple claim.

Meaning does not escape. It just topologizes within the abstract cage.

We no longer describe meaning merely through signs and chains of signifiers, but as flows — morphisms between concepts mediated by contextual anchors called Z-frames. These frames act as semantic coordinates, situating each concept within a space of possible interpretation.

A dog is not simply “a dog.” It is interpreted through the semantic frame Z in which it is embedded.
dog | Domesticated
dog | Mammal
dog | Son

Or, as a morphism, computer → she | person
With the Z frame computer is interpreted as a historical computing worker (pre-digital era), resolving ambiguity via structural semantic framing.

In this model, concepts are objects, interpretive movements are arrows, and semantic coherence is topological.

We define categories like C|Z, where morphisms f: A → B | Z are conceptual transformations under a shared meaning frame. We introduce operators like σ that model semantic shifting, generalization, or abstraction across frames and we show that these operators exhibit functorial and even Kan-extension-like behavior.

Meaning is no longer a mirage. It circulates within a space that is structured, closed, and composable. We are no longer chasing meaning. We are building it from its space.

Note: While we refer to “fibers” to describe morphic coherence over a shared Z-frame, this current formulation is not yet a strict fibered topos in the categorical sense. Rather, this document serves as the semantic scaffolding toward that formalization.

Index

1. Fibered Conceptual Topology
   1.1. A Z-framed Conceptual Category
      - Z-Frame
      - Functoriality
      - Category
      - Object
      - Morphisms
      - Composition
      - Identity Morphism
      - Mirror Morphism
      - Quasi-Natural Transformation(QNT)

   1.2. σ Operator as Functor
      - Definition: Conceptual Shifting Morphism (σ)
      - Identity Morphism of σ
      - Associativity of σ

   1.3. σ Operator as Kan Extension
      - Functorial properties of σ
      - σ operator as Kan Extension
      - Relation to Quasi-Natural Transformation
      - Safe / Unsafe Conceptual Shifting Morphism (σ)

   1.4 Kan Extension as Horizontal Conceptual Shifting and Cone Structure
      - Iterated Colimit Perspective
      - ∞-Morphic Interpretation of Recusive Ken Extension
      - Universal Property of Lan_{σ_i}

2. Monoid Structure of Conceptual Flow (M|Z)

3. Identity Element of M|Z

4. Associativity of M | Z

5. Axioms
   5.1. Unit Axiom: Identity Element of Concept
   5.2. Zero Axiom: Zero Morphism as Negation Morphism
   5.3. Composition Axiom

6. Conceptual Topos
   6.1. Initial Object
   6.2. Finite Limits
   6.3. Exponentials

- Pullback: Quasi-Natural Transformation
- Equalizer: Mirror Morphism
- Product: σ operator

6.4. Subobject Classifier Ω
6.5. Conceptual Topos as Fibered Topos

7. Global Conceptual Space: Total Conceptual Space (TCS)

8. Empirical Data for Conceptual Topology
8.1. Conceptual Flow Classification
8.2. Zero Morphism
3.3. Cone Structure: Kan Extension and QNT
8.4. Universal Product

Appendix:
- Simbols and Notations
- Python Code and Dataset

1. Fibered Conceptual Topology:

Fibered Conceptual Topology provides a conceptual geometric framework wherein each Z-frame (conceptual anchor) acts as a base space, with conceptual morphic flows forming fibers over these anchors. The Yoneda-like interpretation captures concepts as bundles of conceptual relations within and across Z-frames. This fibered structure serves as the foundation for further constructions in Conceptual Topos.

```
CT := (C, B, π: E → B, Fb := π⁻¹(b), A ≅ b ∪ Nat(Hom(-, A), Fb))

Where:
  - C is the category of concepts (objects = words or concepts)
  - B is the base space of Z-frames (conceptual continuity anchors)
  - E is the total conceptual space (word vector embedding space)
  - π projects each concept to its conceptual base (Z-frame)
  - Fb is the fiber (conceptual morphic chain) over a base b
  - A ≅ b ∪ Nat(Hom(-, A), Fb) interprets each concept A via its morphisms
relative to its   Z-frame b (Yoneda perspective defined in appendix)
```

```
        Z = baby
    puppy  ←   girl    //specified: size + young
     |          |
     |          |
    dog        she     //abstraction
      \        /
        Mammal
```

1.1. A Z-framed Conceptual Category

A Z-framed Conceptual Category is a structure (C, Z, Hom, id, ∘).

- Ob(C) is a set of conceptual entities.
- Z is a set of conceptual frames.
- Hom(X, Y | Z) is the set of morphisms from X to Y within Z-frame.
- For each X, there exists an identity morphism id_X|Z: X → X | Z or id_X: X → X | X.
- Composition ∘ is a partial operation defined as

```
For f: X → Y | Z₁ and g: Y → Z | Z₂,
g ∘ f: X → Z | Z₃ is defined iff ∃ Z₃ ∈ Z such that Z₁ ⊆ Z₃ and Z₂ ⊆ Z₃
```

Z-Frame

Z-frame has multi-functionality as a conceptual frame: Category, Fiber and Morphism level object. We define a Z-Indexed Fibered Conceptual Category as a tuple

```
(C, Z, π: C → Z)

Where:
    C: A category of conceptual morphisms (objects: concepts, morphisms: semantic transformations)
    Z: A category of conceptual frames (Z-frames), representing interpretive contexts or domains
    π: A functor projecting each morphism in C to its conceptual frame in Z
```

Object Level Z-Frame Structure

```
Let:
Ob(C) be a set of conceptual entities (dog, she, king, ...)
Ob(Z) be a set of semantic frames (Domesticated, Mammal, Abstract, ...)

Each morphism in C is typed as
f: A → B | Z  ∈ Hom(A, B | Z)

This is realized through a retractive structure mediated by Z
f: A → Z
g: Z → A

such that
g ∘ f ≅ id_A | Z

In diagrammatic terms

   A
   | \
   |  \
   v   v
   Z → A   (g ∘ f = id_A | Z)

Alternatively, to express conceptual flow under Z
f: A → Z
g: Z → B
such that
g ∘ f ≅ A → B | Z

This means that A is transformed to B under the interpretation frame Z. The flow
between A and B is mediated by Z, and Z ensures that the interpretation of both A
and B is consistent under the same frame.
```

Disambiguation and Structural Integrity When multiple interpretations (or semantic frames) exist such as *computer*, Z acts as a disambiguating factor, ensuring that the meanings of A and B are not left to chance but are structurally ensured by their relationships to Z.

Example:

```
computer → she | person
```

Without Z, computer may refer to a machine, a metaphor, a role, or even ambiguity between literal and historical meanings.
With Z = person, computer is interpreted as a historical computing worker (pre-digital era), resolving ambiguity via structural semantic framing.

Fibered Structure

```
For each Z ∈ Ob(Z), define the local fiber:
π⁻¹(Z) := { f ∈ Mor(C) | π(f) = Z }

Over the total base Z, the full fibered category is:
π⁻¹(Z) := { fᵢ ∈ Mor(C) | π(fᵢ) = Zᵢ for some Zᵢ ∈ Ob(Z) }
```

This is the subcategory C|Z, where all morphisms are constrained to operate within the same Z-frame.

Functoriality

π must satisfy the functor laws

```
For any identity morphism id_A|A in C
π(id_A|A) = id_Z|Z

For any composable morphisms f: A → B | Z₁, g: B → C | Z₂ with Z₁, Z₂ ⊆ Z, the
composition is

g ∘ f: A → C | Z

and:
π(g ∘ f) = Z

Here, Z is the least upper bound (or unifying context) of Z₁ and Z₂.
```

Z-Frame Subsuming Composition（$Z \subseteq Z'$） Although f is originally defined over Z, due to the inclusion $Z \subseteq Z'$, the composite $g \circ f$ can be interpreted as an internal composition in the superordinate category Z'.

```
Let Z ⊆ Z'

f: A → B | Z
where:
  f₁: A → Z
  f₂: Z → B
  f = f₂ ∘ f₁

  thus, A → Z → B

g: B → C | Z'
where:
  g₁: B → Z'
  g₂: Z' → C
  g = g₂ ∘ g₁

  thus, B → Z' → C

Z subsuming composition is expressed as follows.

g ∘ f = g₂ ∘ (g₁ ∘ f₂) ∘ f₁
h:= g₁ ∘ f₂: Z → Z′
g ∘ f | Z' = g₂ ∘ h ∘ f₁
```

In Diagram:

```
   f₁        h         g₂
A ────→ Z ────→ Z′ ────→ C
        |       ↑
        f₂      g₁
        ↓       |
        B ──────→
```

In NL Diagram:

```
f: fox → dog | Canine
g: dog →  cat | Mammal
g ∘ f: fox → cat | Mammal
where: Canine, Mammal ⊆ Mammal

      f₁             h              g₂
fox ────→ Canine ────→ Mammal’ ────→ cat
            |             ↑
            f₂            g₁
            ↓             |
           dog ──────────→
```

Z-Frame Lifting via Kan Extension（Z ⊈ Z′）

In the case of Z ⊈ Z′, Left Kan Extension ensures conceptual flow structurally.

```
f: speech → persuasion | Language
g: persuasion → control | Power

f’ := Lan_{Z→Z’}(f) := colim_{(Z → Z’)} f

thus:
h := g ∘ f’: speech → control | Power

            F
Language ────→ Power
   |              |
   |              |
   |              |
   ↓              ↓
Influence ←── Influence

Functor Examples:
  word ↦ authority
  text ↦ order

Note: Concept clusters such as scarlet, crimson, and ruby can be regarded as
objects of a conceptual category (e.g., Red), allowing Kan Extension to operate as
a standard Left Kan Extension across conceptual frames.
```

Category:

We define a Z-framed Conceptual Category **C|Z** (e.g. dog|Domesticated), C|C in simple notation **Concept** (e.g. Dog, Button...), as a category enriched over semantic frames Z.

Notation: We denote a morphism f: X → Y mediated by Z-frame as f: X → Y | Z. This represents a meaning-preserving conceptual flow within the frame Z.

Objects

Ob(C|Z): A set of conceptual entities (lexical terms, abstract notions).
Examples: dog, she, computer, king, etc.

Morphisms

Each morphism is defined mediated by a Z-frame. **Hom(X, Y | Z)** = { f | f: X → Y | Z },
where Z ∈ Ob(Z Frames) represents a semantic anchor or contextual frame.

A morphism f: X → Y | Z is interpreted as "X conceptually maps to Y within the semantic continuity defined by Z." Z gives the interpretive coherence or semantic clarification(e.g., dog → pet | Domesticated).

Composition

Composition is defined only within a shared Z-frame or subsuming Z frame of local Z frames.

1. Within the same Z-frame If f: A → B | Z, g: B → C | Z, then g ∘ f is defined iff Z is shared.

```
f: computer →  smartphone | Gadget
g: smartphone → mobile GPS | Gadget
g ∘ f = computer →  mobile GPS | Gadget
```

2. Across compatible Z-frames (via σ-mediated composition)

Composition across different Z-frames (i.e., σ-mediated composition) is possible when the individual Z-frames are compatible under a higher semantic frame. This higher frame Z must be able to subsume both the local frames Z_1 and Z_2 by the conditions $Z_1 \subseteq Z$ and $Z_2 \subseteq Z$. This condition ensures that both morphisms can coexist within the same larger context, preserving the continuity of meaning across frames.

If f: A → B | Z_1, g: B → C | Z_2, then g ∘ f is defined iff there exists a higher frame Z such that $Z_1 \subseteq Z$ and $Z_2 \subseteq Z$.

```
f: computer →  smartphone | Gadget
g: match → knife | Tool

If there exists a higher frame Instrument that subsumes both Gadget and Tool,
then the composite morphism becomes

g ∘ f = computer → knife | Instrument
where Instrument ⊇ Gadget, Tool
```

This composition is associative within a Z-frame

```
If: f: A → B | Z, g: B → C | Z, h: C → D | Z
then: (h ∘ g) ∘ f | Z = h ∘ (g ∘ f) | Z.
```

This guarantees that within a single Z-frame, composition behaves as expected according to the standard rules of category theory.

σ-mediated Composition:

In the case of σ-mediated composition, associativity holds when all involved Z-frames are subsumed by a common higher Z-frame.

```
Let f: A → B | Z₁, g: B → C | Z₂, and h: C → D | Z₃
 (h ∘ g) ∘ f | Z = h ∘ (g ∘ f) | Z is valid
 where Z₁ ⊆ Z, Z₂ ⊆ Z, and Z₃ ⊆ Z.
```

This ensures that all morphisms can coexist within the same conceptual space, and the meaning flow is preserved across the frames.

Example:
If f: computer → smartphone | Gadget, g: smartphone → mobile GPS | Gadget, and h: mobile GPS → navigation | Travel, then the composite morphism is defined as

```
h ∘ (g ∘ f) | Travel  = computer → navigation | Travel
where: Gadget ⊆ Travel is defined
```

Here, Instrument is a higher Z-frame that subsumes both Gadget and Travel.

Let C|Z be a Conceptual Category with partial composition ∘|Z.

Partial Composition in Z-Framed Category

```
Typing judgment
  f: A → B | Z ∈ Hom(A, B | Z)

Composition judgment
  If f: A → B | Z₁ and g: B → C | Z₂,
  and ∃Z such that Z₁ ⊆ Z and Z₂ ⊆ Z
  then define:
      g ∘ f : A → C | Z

This defines a partial composition operation:
∘ : Hom(A, B | Z₁) × Hom(B, C | Z₂) ⇀ Hom(A, C | Z)
```

Identity Morphism

In Category Theory, Identity Morphism is always defined.

```
For every morphism f: A → B,
there exist identity morphisms id_A: A → A and id_B: B → B such that
  f ∘ id_A = f
  id_B ∘ f = f
```

However, in Conceptual Topology, morphisms are mediated by Z frame, thus the identity morphis is not always given unless Z is defined.

Two Types of Identity Morphism in Conceptual Topology

1. f: X → X | X (Category-theoretic identity)

```
For every object X, there exists a mediated identity morphism
  id_X: X → X | Z

such that for Z = X (i.e., the identity is mediated by the object itself),
we define
  f: X → X | X
  f ∘ id_X = f
  id_X ∘ f = f

e.g. f: dog → dog | dog
```

2. f: X → X | Z (Mediated identity with conceptual flow)

```
f: X → X | Z

f: X → Z
f⁻¹: Z → X
f⁻¹ ∘ f ≅ id_X
e.g. you → you | externalized perspective
     NL: you are you
```

Since the identity morphism passes through an external anchor point, the identity morphism is defined quasi-identical.
e.g. dog → perro | собака

Simplified Form of Identity Morphism:

1. f: X → X | X (Category-theoretic identity)
   In simplified form: X
   or more explicitly: id_X

2. f: X → X | Z (Mediated identity with conceptual flow)
   In simplified form: X Z

Mirror Morphism Definition:

Each mirror maps conceptual transitions across vocabularies while preserving morphic identity up to rupture—that is, it allows for conceptual divergence that still respects underlying structural continuity, even if exact invertibility is not preserved.

```
f : X  → Y  | Z ∈ Dᵢ
f′: X′ → Y' | Z ∈ Dᵢ₊₁
  ⇒X′ ≠ X, but cod(f) = cod(f′) | CD (CD = codomain)

We define f′ as a mirror-correspondent morphism of f under a given Z-frame,
if and only if:

∃Z: rupture(f, f′ | Z) ≠ ∅
∧ cod(f) = cod(f′) | CD
```

```
        Z = baby
    puppy  ←   girl    //specified: size + young
     |          |
     |          |
    dog        she     //abstraction
      \        /
        Mammal
```

Note: Z: rupture(f, f′ | Z) ≠ ∅ means that there exists a Z-frame under which *f* and *f′* exhibit structural divergence—i.e., they are not fully invertible but still converge at the codomain level.

For example, let *Z* = *abstraction*. This allows a conceptual transition from *girl* → *she* and *puppy* → *dog*, treating them as mirror morphisms under a shared conceptual frame.
However, if we take *Z* = *agency*, a rupture emerges: *puppy* → *dog* lacks agency, while *girl* → *she* retains it.
Hence, **rupture(f, f′ | agency) ≠** ∅, yet *f* and *f′* still align toward the same codomain (e.g., *mammal*).

Quasi-Natural Transformation of Meaning Systems

A **Morphic Chain Mirror** is a contextual correspondence between two morphic chains drawn from distinct but meaning-aligned vocabularies. This correspondence is realized through a **quasi-natural transformation** under a shared intermidiating Z-frame.

```
η: Dᵢ ⇒ Dᵢ₊₁  | CD (CD = codomain)
η_X ∘ Dᵢ({f₁ | Z₁, ..., fₙ | Zₙ}) ≈ Dᵢ₊₁({f′₁ | Z₁, ..., f′ₙ | Zₙ}) ∘ η_Y | CD
for all fⱼ: Xⱼ → Yⱼ | Zⱼ ∈ Dᵢ,
where f′ⱼ: η_X(Xⱼ) → η_Y(Yⱼ) | Zⱼ

Then, η is said to be a quasi-natural transformation under the Z-frame
i.e. η ∈ Mor(C) where C is the contextual meaning category
Example: η: girl → puppy | Z = Baby
```

Diagram:

```
  girl → she → mammal
     |             |
     | η_X      η_Y |
     v             v
  puppy → dog → mammal
```

1.2. σ Operator as Functor

Definition: Conceptual Shifting Morphism (σ)

```
σ: D(Xₙ₋₁ | X) → D(Xₙ₋₁ | X)

such that σ ⊕ f ∈ M|Z if and only if type compatibility holds:

∀ A, B, (A → B) ∘ σ(X) is valid if:

( A >> X or X >> A )
and
( B >> X or X >> B )

Definition: Subsumption
A >> X ≡ A ⊑ X

Definition: SubsumedBy
X >> A ≡ X ⊑ A

Example:
king → king >> human → human
  ⇒ king >> human  → valid

human → human >> queen → queen
  ⇒ human >> queen → valid
```

Conceptual Operators

Conceptual Operator σ modifies morphism as follows.

```
σ(X). Not(x){ A →/B | Z}     →     Rupture under Z frame
σ(X). so_much(x){A → B | Z}  →     Preservation & amplification under Z frame
σ(X). >>(x,y)                →     Conceptual Shifting x to y (Generalization) as
function form
σ(X). <<(x,y)                →     Downward Shifting x to y (Specialization) as
function form
σ(X). >(x,y)                 →     Conceptual Shifting
```

Conceptual Morphism Set Operators

```
Addition (⊕):
σ(X). ⊕(f, Aₙ₋₁ | Z): D(Aₙ₋₁ | Z) → D(Bₙ₋₁ | Z) | Z
σ(X). ⊕(f₁, f₂) : Aₙ₋₁ :={f₁, f₂}

Subtraction (⊖):
⊖: Aₙ₋₁ ⊖ {f_i}
σ(X). ⊖(f, Aₙ₋₁ | Z): D(Aₙ₋₁ | Z) → D(Bₙ₋₁ | Z) | Z

- ⊕ operator is σ_safe if Z alignment is preserved.
- ⊖ operator is potentially σ_unsafe but can be σ_safe if resulting chain
preserves the underlying morphic continuity Z.
```

Example

```
{Royalty⃗, Male⃗, Human⃗} ⊖ {Male⃗} ⊕ {Female⃗}
= {Royalty⃗, Female⃗, Human⃗}
= queen
```

Conceptual Mapping

```
C_chain = { f₁, f₂, ..., fₙ | Z} ∈ D(Cₙ₋₁ | Z)

σ(X): D(Aₙ₋₁ | Z) > D(Bₙ₋₁ | Y) | Z, Y ∈ CD
where:
D(Aₙ₋₁ | Z) = source morphic chain
D(Bₙ₋₁ | Y) = target morphic chain
CD = codomain alignment (conceptual anchor)

σ(X) is not strict functorial → quasi-alignment under conceptual equivalence

σ(X) ≈ η: Dᵢ ⇒ Dᵢ₊₁ | CD (Quasi-Natural Transformation interpretation)

Example:
σ(X). >(puppy → dog → mammal | Canine, Human) ∋ girl → she → mammal | Human

where: canine, Human ∈ Mammal
```

Identity Morphism of σ

```
word is word
thus:
word ≅ Nat(Hom(-, word | Word), Fib(word))

σ_id(Z). OP(X,Z) = σ such that σ(f,Z) = f for all f ∈ Hom(X, X | Z) unless OP is
σ_unsafe such that word is not a word: σ(Word). Not(word ↛word)

σ_id(Word). OP(word, Word) = word

σ_id(Word). OP(f, Word) = f  for all f: word → word | word
since: M|Z = { fₙ ∘ ... ∘ f₁ | all fᵢ: Xᵢ → Xᵢ₊₁ | Z ∧ ∀ i, j: fᵢ ≅ fⱼ | Z = Word
}

σ_id(Word) ∈ M|Word
σ(Word) ∘ σ_id(Word) = σ(Word)
σ_id(Word) ∘ σ(Word) = σ(Word)
∴ word is word
```

Associativity of σ

```
σ₁(Z). OP(D(Aₙ₋₁ | Z), Z) = D(Zₙ₋₁ | Z)
σ₂(Z). OP(D(Bₙ₋₁ | Z), Z) = D(Zₙ₋₁ | Z)

Then the composition σ₂ ∘ σ₁:
σ_comp(Z). OP(D(Zₙ₋₁ | Z), D(Zₙ₋₁ | Z)) = D(Zₙ₋₁ | Z)

where: OP is not σ_unsafe and under shared Z frame

Associativity
For all σ₁(Z), σ₂(Z), σ₃(Z) such that their domains/codomains match for
composition:
(σ₃ ∘ σ₂) ∘ σ₁ = σ₃ ∘ (σ₂ ∘ σ₁)

Thus, σ composition operator is associative under Monoid structure.
```

Example:

```
Let σ₁ = σ(Mammal). >>(canine → mammal, Life)  = (life → life | Life)
Let σ₂ = σ(Mammal). >>(mammal → animal, Life)  = (life → life | Life)
Let σ₃ = σ(Mammal). >>(animal → livingBeing, Life) = (life → life | Life)

Conclusion:
 (σ₃ ∘ σ₂) ∘ σ₁ = σ₃ ∘ (σ₂ ∘ σ₁) =  (life → life | Life)
```

1.3. σ Operator as Kan Extension

Functorial Properties of σ mapping

```
A Functor F: C → D is a mapping between categories satisfying:

- Object mapping: For each X ∈ Ob(C), F(X) ∈ Ob(D)
- Morphism mapping: For each f: X → Y ∈ Mor(C), F(f): F(X) → F(Y) ∈ Mor(D)
- Identity preservation: F(id_X) = id_{F(X)}
- Composition preservation: F(f ∘ g) = F(f) ∘ F(g)

We define σ: D(Aₙ₋₁ | Z) >> D(Bₙ₋₁ | Z') as such a Functor.
```

σ Operator as Kan Extension

```
Let:
 D(Aₙ₋₁ | Z) := Category of Morphic Chains over Z-frame Z
 D(Bₙ₋₁ | Z') := Category of Morphic Chains over Z'-frame Z'

Define:
 σ_safe = Lan_σ : D(Aₙ₋₁ | Z) >> D(Bₙ₋₁ | Z')

such that:

For any object d ∈ D(Bₙ₋₁ | Z'),
Lan_σ (D(Aₙ₋₁ | Z))(d) := colim_{(c, f: σ(c) → d)} D(Aₙ₋₁ | Z)(c)

And:

For any morphism h: d → d' in D(Bₙ₋₁ | Z'),
Lan_σ (h) is defined to preserve functoriality

Lan_σ (h) ∘ Lan_σ (f) = Lan_σ (h ∘ f)

Therefore:

σ_safe satisfies:
  Object-level safe lifting: Ob(D(Aₙ₋₁ | Z)) → Ob(D(Bₙ₋₁ | Z'))
  Morphism-level safe lifting: Mor(D(Aₙ₋₁ | Z)) → Mor(D(Bₙ₋₁ | Z'))

σ_safe = Left Kan Extension guarantees the Quasi-Natural Transformation property

∀ f ∈ Mor(D(Aₙ₋₁ | Z)),
Lan_σ (G ∘ f) = (Lan_σ G) ∘ (Lan_σ f)
```

Relation to Qasi-Natural Transformation

The σ mapping operator, defined as a Functor σ: $D(A_{n-1} | Z) >> D(B_{n-1} | Z')$, exhibits structural alignment with Quasi-Natural Transformation (QNT) in the following way.

In the original formulation of QNT in this framework

```
η: Dᵢ ⇒ Dᵢ₊₁ | CD (codomain)
η_X ∘ Dᵢ({f₁ | Z₁, ..., fₙ | Zₙ}) ≈ Dᵢ₊₁({f′₁ | Z₁, ..., f′ₙ | Zₙ}) ∘ η_Y
```

Diagram:

```
       Z = baby
   puppy  ←   girl    //specified: size + young
     |          |
     |          |
   dog        she     //abstraction
      \        /
       Mammal

In this diagram, Quasi-Natural Transformation η aligns morphic chains between
puppy → dog → mammal and girl → she → mammal within a shared codomain mammal
(under Z-frame "Mammal").
```

The Quasi-Natural Transformation mediates conceptual flow correspondence across different morphic chain categories under a shared or shifted Z-frame.

In the Kan Extension formalization:

```
Lan_σ (D(Aₙ₋₁ | Z)) = D(Bₙ₋₁ | Z')
```

The lifting of the entire functor $D(A_{n-1} | Z)$ under σ corresponds to constructing a universal QNT from $D(A_{n-1} | Z)$ to $D(B_{n-1} | Z')$.

More precisely, for any object $d \in D(B_{n-1} | Z')$:

```
Lan_σ (D(Aₙ₋₁ | Z))(d) := colim_{(c, f: σ(c) → d)} D(Aₙ₋₁ | Z)(c)
```

yields a canonical shifting from the conceptual flow space under Z to the corresponding conceptual flow space under Z', respecting the structural continuity required by QNT.

Thus:

```
σ_safe ≈ Left Kan Extension ≈ Universal Quasi-Natural Transformation between
D(Aₙ₋₁ | Z) and D(Bₙ₋₁ | Z')
```

Diagram:

```
puppy → dog → mammal
      \         |
       \        |
        v       v σ(Life)
           Life

girl → she → mammal
      \         |
       \        |
        v       v σ(Life)
           Life

σ(Life) acts as a Kan Extension operator, lifting conceptual morphic chains from
Fiber over Z = Mammal to Fiber over Z' = Life. The lifting ensures coherent
conceptual projection across Z-frames, preserving flow alignment while
generalizing to a higher conceptual space.
```

This formalization guarantees that the Quasi-Natural Transformation property observed in the original Conceptual Cage structure is preserved and generalized through the Kan Extension framework, providing a categorical foundation for conceptual flow lifting.

1.4 Kan Extension as Horizontal Conceptual Shifting and Cone Structure

Conceptually, σ operator as Kan Extension performs not only lifting of morphic chains but also acts as a horizontal mapping across Z-frames, shifting conceptual flow from Fiber over Z to Fiber over Z'.

Diagrammatically, this can be visualized as a horizontal shift.

```
Fiber over Z (Mammal):

puppy → dog → mammal
girl → she → mammal

↓↓↓↓↓↓ Kan Extension σ(Life)

Fiber over Z' (Life):

girl → she → mammal → Life
puppy → dog → mammal → Life
```

Applying σ(Life) results in a horizontal lifting of codomain alignment

Recursive Kan Extension as Iterated Colimit of Conceptual Shiftings
Conceptually, Recursive Kan Extension can be understood as constructing an iterated colimit of sequential conceptual shiftings (σ operators) across Z-frames.

Conceptual Ladder Structure:

```
Fiber over Z₀
    ↓ σ₁
Fiber over Z₁
    ↓ σ₂
Fiber over Z₂
    ↓ σ₃
Fiber over Z₃
    ↓ ...

NL: turtle → reptile → animal → ...
```

Iterated Colimit Perspective:
At each stage, the application of σ_n corresponds to forming a conceptual projection from Fiber over Z_{n-1} to Fiber over Z_n.

The entire ladder:

```
Lan_{σₙ} ∘ ... ∘ Lan_{σ₃} ∘ Lan_{σ₂} ∘ Lan_{σ₁}
```

can be viewed formally as an iterated colimit over the sequence of Z-frames, forming a conceptual cone over the diagram.

```
colim_{Z₀ → Z₁ → Z₂ → ... → Zₙ} (Lan_{σᵢ}(π⁻¹(Zᵢ₋₁)))
```

Interpretation:
Each Lan_{σ_i} acts as a conceptual lifting operation, progressively shifting semantic flow across Z-frame layers. The cumulative structure forms an iterated conceptual cone, whose colimit aligns the entire sequence into the semantic flow space under Z_n.

Diagram (Iterated Colimit View):

```
                Fiber over Z₀
                   ↓ σ₁
                Fiber over Z₁
                   ↓ σ₂
                Fiber over Z₂
                   ↓ σ₃
                Fiber over Z₃
                   ↓ ...
            ---------------------
          Iterated Colimit (Conceptual Cone)
           → Fiber over Zₙ

NL: tortoise → turtle → reptile → animal → ... | Iterated Colimit Result = Muti
Celluar Organism
```

Formal Expression:

```
Iterated_Colimit = colim_{Z₀ → Z₁ → Z₂ → ... → Zₙ} (Lan_{σᵢ}(π⁻¹(Zᵢ₋₁)))
```

This conceptual ladder forms an iterated semantic cone, whose colimit aligns the entire Z-frame sequence into the unified semantic flow space under Z_n.

Diagram:

```
          Zₙ
     ↙ ↓    ↓ ↘
   Z₀   Z₁   Z₂   Z₃

          ↓

M|Zₙ (colimit of Ladder)
```

A cone on a diagram F: J → C is a universal natural transformation from a constant diagram ΔX to F. In this case:

```
Δ(π⁻¹(Zₙ)) ⇒ Ladder of Lan_{σᵢ}(π⁻¹(Zᵢ₋₁))

or as monoid structure
M|Zₙ { Fₙ ∘ ... ∘ F₁ | all Fᵢ: Fᵢ → Fᵢ₊₁ | Zₙ ∧ ∀ i, j: Fᵢ ≅ Fⱼ | Zₙ }
```

∞-Morphic Interpretation of Recusive Ken Extension

Viewed categorically, this recursive construction aligns with the notion of ∞-morphisms or higher morphic flows, where each application of Lan_{σ_i} corresponds to a morphism in an extended conceptual category, and their collective composition forms an ∞-structured cone.

```
∞-Universal Product = colim_{Z₀ → Z₁ → Z₂ → ... → Zₙ} (Lan_{σᵢ}(π⁻¹(Zᵢ₋₁)))
```

Diagram:

```
Z₀ → Z₁ → Z₂ → Z₃ → ... → Zₙ
 |     |     |     |        |
Lan_{σ₁}  Lan_{σ₂}  Lan_{σ₃}  ... Lan_{σₙ}
 |     |     |     |        |
 v     v     v     v        v
Fiber over Z₁   Fiber over Z₂   Fiber over Z₃   ...   Fiber over Zₙ
       ↘ ↘ ↘ ↘ ↘
         ↓ ↓ ↓ ↓
colim (Lan_{σᵢ}(π⁻¹(Zᵢ₋₁)))
```

NL Diagram:

```
tortoise → turtle → reptile → ... → Zₙ
  |         |     |     |           |
Lan_{σ₁}  Lan_{σ₂}  Lan_{σ₃}  ... Lan_{σₙ}
 |     |     |     |        |
 v     v     v     v        v
Fiber over Z₁   Fiber over Z₂   Fiber over Z₃   ...   Fiber over Zₙ
       ↘ ↘ ↘ ↘ ↘
         ↓ ↓ ↓ ↓
  Multi-Cellular Organism
```

This interpretation enables the Conceptual Topos or Conceptual Topology to naturally support recursive, layered conceptual flow, where mappings can extend across arbitrarily many Z-frames while preserving structural coherence.

Example: Iterated Kan Extension of Conceptual Ladder

Step 1:

```
Z₀ = Turtle
Z₁ = Reptile
σ₁ = σ(Reptile): Turtle → Reptile

Lan_{σ₁}(Fiber over Turtle) → Fiber over Reptile
```

Step 2:

```
Z₂ = Animal
σ₂ = σ(Animal): Reptile → Animal

Lan_{σ₂}(π⁻¹(Reptile)) → π⁻¹(Animal)
```

Step 3:

```
Z₃ = Life
σ₃ = σ(Life): Animal → Life

Lan_{σ₃}(Fiber over π⁻¹(Animal)) → π⁻¹(Life)
```

Composition:

```
Lan_{σ₃} ∘ Lan_{σ₂} ∘ Lan_{σ₁}(π⁻¹(Turtle))
```

Colimit:

```
colim_{Z₀ → Z₁ → Z₂ → Z₃} Lan_{σᵢ}(π⁻¹(Zᵢ₋₁)) = π⁻¹(Z₃) = π⁻¹(Life)
```

NL:

```
Turtle → Reptile → Animal → Life
```

Conceptual flow lifted across Z-frame layers as iterated Kan Extensions, converging to the unified flow under Life.

Universal Property of `Lan_{σᵢ}`

```
   Z_{i-1}         ──σᵢ──▶          Zᵢ
      │                              │
      │ H                            │ K
      ▼                              ▼
      E     ◀── α: H ⇒ K ∘ σᵢ^* ─── E
```

```
Given a base conceptual shifting operator
σᵢ: Z_{i-1} >> Zᵢ

we define Lan_{σᵢ} for corresponding fiber categories
Lan_{σᵢ} : π⁻¹(Z_{i-1}) → π⁻¹(Zᵢ)

To satisfy the following **universal property**, for any functor
H: π⁻¹(Z_{i-1}) → E

and any functor
K: π⁻¹(Zᵢ) → E

with a natural transformation
α: H ⇒ K ∘ σᵢ^*

(where `σᵢ^*` is the pullback functor along `σᵢ`),
there exists a unique natural transformation
β: Lan_{σᵢ}(H) ⇒ K

such that the following diagram commutes
H
↓ α
K ∘ σᵢ^*
↑
Lan_{σᵢ}(H) ∘ σᵢ^*

In formal terms
Nat(H, K ∘ σᵢ^*) ≅ Nat(Lan_{σᵢ}(H), K)
```

NL Diagram: The operation `smartphone → GPS` maintains the structural coherence under Gadget when lifting it to Instrument.

```
    Gadget          ──σ₁──▶        Instrument
      │                               │
      │ H: "smartphone → GPS"         │ K: "smartphone → GPS"
      ▼                               ▼
  Meaning_E ◀── α: H ⇒ K ∘ σ₁^* ── Meaning_E

  σ₁: Gadget >> Instrument
  α: H ⇒ K ∘ σ₁^*:
    "smartphone → GPS" | Gadget ≅ "smartphone → GPS" | Instrument
    where: gadget → instrument | Z = E and instrument → gadget | Z = E
```

NL Example 2

```
A functor F: Canine → Life
A functor σ: Canine → Mammal

Then the left Kan extension of F along σ is a functor.
Lan_σ(F): Mammal → Life

Object d ∈ Mammal, c ∈ Canine, d,c ∈ Life
Lan_σ(F)(d) = colimit (σ(c) → d)(F(c))

                 σ
  Canine ─────────────────▶ Mammal
   fox, dog              σ(fox), σ(dog), cat

     │                         │
     │  F                      │ Lan_σ(F)
     ▼                         ▼
   Life ◀──────────────────── Life
             dog → cat


e.g. Lan_σ(f)(fox)= dog → cat

fox, dog, cat ∈ Life
fox, dog, cat ∈ Mammal
fox, dog ∈ Canine

We focus on fox.
fox ∈ Canine                     // fox is a canine
F(fox) = dog ∈ Life              // fox behaves like dog in the context of Life
σ(fox):= dog → cat | Mammal       // Under Mammal, fox is like a dog or a cat, not
a gorilla. (QNT)
dog → cat | Mammal               // dogs, as mammals, resemble cats.
```

```
⇒ Lan_σ(F)(fox) = dog → cat | Life
That is:
    A fox, seen as a canine, is mapped to a dog under the functor: Canine → Life,
    But as mammals, dogs resemble cats,
    The fox is, in Life, conceptually equivalent to a transition, dog → cat — that
is, the fox is basically dog or cat.

Note:
  The transition (dog → cat | Life) is interpreted as a Quasi-Natural
Transformation (QNT):

  η: dog ⇒ cat | Mammal
  dog → Life ∘ η_Y ≈ η_X ∘ cat → Life
  NL: "Cat is a dog, and dog is a cat — basically, under the Life frame."
```

Example:

```
f₁: king → man        | Z₁
f₂: woman → ?         | Z₁

Conceptual Shifting Operator
σ₁: Z₁ >> Z₂   (GenderedEntity >> SocialRole: Generalization)
σ₂: Z₃ << Z₂    (SocialRole << RoyalSemantic: Specification)

Lan_{σ₁}(f₁): king → male-role | Z₂
Lan_{σ₁}(f₂): female-role → female-role | Z₂

Lan_{σ₂}(Lan_{σ₁}(f₁)): king → king | Z₃
Lan_{σ₂}(Lan_{σ₁}(f₂)): queen → queen   | Z₃

We may define σ₃ = σ₂ ∘ σ₁ : Z₁ → Z₃ as the composition of generalization and
specification,
allowing us to write Lan_{σ₃}(f) ≅ Lan_{σ₂}(Lan_{σ₁}(f))

Alternately, We define the above as fibers.
π⁻¹(Z₁): Gendered Entity
π⁻¹(Z₂): Social Role
π⁻¹(Z₃): Royal Semantic

colim_{Z₁ → Z₂ → Z₃} (Lan_{σᵢ}(π⁻¹(Zᵢ)))

∴Lan_{σ₂}(Lan_{σ₁}(king - man + woman)) ≅ queen
```

Diagram:

```
Z₁: Gendered Entity
  king → man
  woman

σ₁ ↓ Generalized to Social Role

Z₂: Social Role
  king → male-role
  female-role → female-role

σ₂ ↓ Specified to Royality

Z₃: Royal Semantic
  king → king
  queen → queen
```

Iterated Kan Extension Ladder over the Z-frame

```
Z₀ → Z₁ → Z₂ → ... → Zₙ

π⁻¹(Z₀) → π⁻¹(Z₁) → π⁻¹(Z₂) → ... → π⁻¹(Zₙ)

Lan_{σₙ} ∘ ... ∘ Lan_{σ₁}(π⁻¹(Z₀)) → π⁻¹(Zₙ)
```

Iterated colimit approximates the **unified conceptual flow**

```
Iterated_Colimit ≅ colim_{Z₀ → Z₁ → ... → Zₙ} (Lan_{σᵢ}(π⁻¹(Z_{i-1})))
```

Example: Pullback of a Meaning Transformation via σ_1^

We consider a morphism in the fiber over Z_2 = Instrument

```
f: smartphone → GPS | Z₂
```

Let $\sigma_1: Z_1 \to Z_2$ be a contextual shift from Z_1 = Gadget to Z_2 = Instrument. To interpret this transformation from the perspective of Z_1, we apply the pullback functor σ_1^*.

This yields

```
σ₁^*(f): smartphone → smartphone-GPS | Z₁
```

where smartphone-GPS is a more concrete or reduced interpretation of GPS within the limited frame of Z_1.

Diagram:

```
      smartphone
          |
          |  f (Z₂: Instrument)
          v
          GPS
         ↑
        /
  σ₁^*(f) (Z₁: Gadget)
       /
smartphone-GPS (≲ GPS)

f: An abstract transformation in Z₂, viewing the smartphone as a GPS device.
σ₁^*(f): A recontextualized version of f, valid under the Z₁ frame, where GPS is
not a standalone concept but a function within the smartphone.
Interpretive Constraint: σ₁^*(f) exists only if the target concept can be
reconstructed within Z₁.

σ₁^*(f) is undefined ⇒ rupture(f, σ₁^*(f)) ≠ ∅
In this case, the meaning transformation cannot be pulled back into the lower
context.
```

Safe / Unsafe Conceptual Shifting Morphism (σ)

Definition of Safe and Unsafe σ Operator
Conceptual Shifting Morphism (σ) can be classified based on whether it preserves the global coherence of the morphic chain.

Safe σ Operator (σ_safe) Acts on the entire morphic chain as a coherent transformation.

```
σ_safe: D(Aₙ₋₁ | Z) > D(Bₙ₋₁ | Z') | Z >> Z' ∨ Z << Z'

where: Z, Z'∈ CD
```

Behaves as a Quasi-Natural Transformation

```
σ_safe = η: Dᵢ ⇒ Dᵢ₊₁ | CD
```

Composition is associative:

```
(σ₃ ∘ σ₂) ∘ σ₁ = σ₃ ∘ (σ₂ ∘ σ₁)
```

Resulting chain remains in M|Z or M_{Z'} (closed).

Example

```
σ₁(X). >(canine, mammal)
σ₂(X). >(mammal, animal)
σ₃(X). >(animal, livingBeing)

Composition:
(σ₃ ∘ σ₂) ∘ σ₁ = σ₃ ∘ (σ₂ ∘ σ₁)
→ >(canine, livingBeing)

Entire morphic chain is preserved.
```

Unsafe σ Operator (σ_unsafe) Does not preserve global coherence of the morphic chain. Acts locally or in a decomposed manner.

Chain may collapse:

```
σ_unsafe: D(Aₙ₋₁ | Z) → { rupture(f₁), rupture(f₂), ..., rupture(fₙ) | ¬Z }
```

```
rupture(f, σ(f) | Z) ≠ ∅
```

Cannot be captured by a Quasi-Natural Transformation globally.

Example

```
σ(X). Not(x) { A →/B | Z }

Result:
rupture(A →/B | Z)
→ breaks the morphic flow → chain decomposes.
```

2. Monoid Structure of Conceptual Flow (M|Z):

In Conceptual Topology, Z is defined as a mediating point/conceptual anchor.

```
Let C and D, Z be categories,
with conceptual projection π: C ∪ D → Z, such that for each X ∈ Ob(C ∪ D)

π(X) ∈ Ob(Z)

For each X ∈ Ob(C ∪ D), there exists morphism

f_X: X → π(X)
f_X⁻¹: π(X) → X

such that

f_X⁻¹ ∘ f_X ≅ id_X

For morphism f: X → Y | Z,
this corresponds to:

f_Z: π(X) → π(Y) | Z

For any X, Y ∈ Ob(C ∪ D)

Let [X]_Z := conceptual representation of X under frame Z (i.e., π(X))

Then:

[ X ]_Z1 ≅ [ Y ]_Z2 | Z1, Z2 ∈ Z //or Z1, Z2 >> Z
which means
["Dog"]_Pet = [Retriever, Dachshund, Poodle, Bulldog, ...]
["girl"]_Human = [girl, woman, person, ...]
["Dog"]_Pet ≅ ["girl"]_Human | Life

Then the set of conceptual flow morphisms under Z forms a monoid

M|Z = { fₙ ∘ ... ∘ f₁ | all fᵢ: Xᵢ → Xᵢ₊₁ | Z ∧ ∀ i, j: fᵢ ≅ fⱼ | Z }

This is also defined as Morphic Chain.

Let D(Cₙ₋₁ | Z) := Category of Morphic Chains over Ob(Cₙ₋₁) within a given Z-frame.
where:D(Cₙ₋₁ | Z) ={ C₀ → C₁ → C₂ → ... | Z }

or as a set
D(Cₙ₋₁ | Z) ={ C₀, C₁, C₂, ... | Z }
```

3. Identity Element of M|Z

```
Let: M|Z = { fₙ ∘ ... ∘ f₁ | all fᵢ: Xᵢ → Xᵢ₊₁ | Z ∧ ∀ i, j: fᵢ ≅ fⱼ | Z }

Define the identity element of M|Z as a family of identity morphisms over the
shared Z frame:

For each X ∈ Ob(C ∪ D), there exists a unique identity morphism under a Z frame:

e|Z_X := id_X | Z

Then, for any f: X → Y | Z ∈ M|Z:

e|Z_X ∘ f = f
f ∘ e|Z_Y = f

Therefore, the identity structure of M|Z is given by the family:

{ id_X | Z | X ∈ Ob(C ∪ D) }

which forms a pointwise identity across the objects under the common Z frame.
This ensures that M|Z satisfies the identity axiom of a monoid.
```

4. Associativity of M | Z

```
Let: M|Z = { fₙ ∘ ... ∘ f₁ | all fᵢ: Xᵢ → Xᵢ₊₁ | Z ∧ ∀ i, j: fᵢ ≅ fⱼ | Z }

Then for all f, g, h ∈ M|Z:

(f ∘ g) ∘ h = f ∘ (g ∘ h)

Thus, the composition ∘ in M|Z is associative.
```

5. Axioms

5.1. Identity Element

Unit Axiom 1: Identity Element Z

```
id_Z:= Z → Z | Z
∀f ∈ M|Z: id_Z ∘ f = f  and f∘id_Z =
```

Definition:

```
Statement:
Z-frame itself is the unit of M | Z.

Formal Definition:
Z:= Z → Z|Z

Justification:
Since any morphism in M | Z is defined as:

f: X→Y|Z

and Z itself is defined as its own identity morphism:

Z:= Z→Z|Z

then:
id_Z = Z

Conclusion:

Therefore:
id_Z is the unit element of M|Z.

∀f ∈ M|Z: (id_Z ∘ f | Z) = f and (f ∘ id_Z | Z)= f
(with frame-preserving composition)

∴id_Z is the unit of M|Z.
```

Note:

```
idZ :Z→Z | Z
f:X→Y | Z
(idZ|Z)∘(X→Y|Z)
```

Unit Axiom2: Void Concept

```
f ∈ M|Z
"" ∘ f = f  and f ∘ "" = f
id_Z ∘ f = f and f ∘ id_Z = f
```

Definition:

```


The empty concept is a theoretically assumed concept, denoted as "",
which acts as the unit element at the conceptual / lexical level.

Formal Definition:

"" ∘ f = f and f ∘ "" = f

Justification:

The empty concept "" represents no lexical or conceptual content.
Composing any morphism f with the empty concept does not alter the flow of
meaning.
Conclusion:

"" is the unit element at the conceptual level of Conceptual Topology.

```

5.2. Zero Morphism: Negation Morphism

We define conceptual zero morphism, negation morphism: n_f In CT as the result of applying Not() to a morphism

```
g:σ(Z). Not(g){ A →/B | Z} = A →/B|Z = n_f
where: g: A → B

Formal Properties (Axiom):

∀g:X→Y|Z where composition with n_f is defined:

∀g: g ∘ n_f = n_f and n_f ∘ g = n_f

Left Side:
g: A→B
g∘(A→B̸|Z) = A→/B|Z

Right Side
g: A→B (A→B̸|Z) ∘ g = A→B̸|Z
```

```
Interpretation:
Applying Not() to any morphism produces a conceptual zero morphism, which
collapses any further conceptual flow.
```

NL Diagram:

```
this ──────→ correct        ← (monomorphism)
  |                           ↑
  └──────→ not(correct) ────┘    ← rupture: zero morphism
```

Natural Language:

```
Left Side: g∘(A→B̸|Z)
"A is not B"
The apple is not a fruit

Right Side: (A→B̸|Z)∘g
"B is not A"
This is a fruit, but this is not an apple which is a fruit.


In CT, this was called rutpure().
Now defined:
rupture(A,B,Z)= σ(Z).Not(g) = n_f = A→B̸|Z
```

5.3. Composition Axiom

```
M|Z = { fₙ ∘ ... ∘ f₁ | all fᵢ: Xᵢ → Xᵢ₊₁ | Z ∧ ∀ i, j: fᵢ ≅ fⱼ | Z }

Then for all f, g, h ∈ M|Z:

For f, g, h ∈ M|Z,
where:
f:V → W | Z
g:Y → V | Z
h:X → Y | Z

(f ∘ g) ∘ h = f ∘ (g ∘ h)
```

Example:

```
For f, g, h ∈ M|Z,
```

```
where:
f:she → you | Human
g:he → she | Human
h:man → he | Human

(f ∘ g) ∘ h = f ∘ (g ∘ h)
```

6. Conceptual Topos

6.1. Category Level: Initial Object

Definition:

```
Let Concept be a category where Ob(Concept) are lexical / conceptual objects.
Then "" ∈ Ob(Concept) is Initial Object if:

∀ X ∈ Ob(Concept), ∃ unique morphism:

u_X : "" → X | X

such that:

∀ f: X → Y | Z,
f ∘ u_X = u_Y
```

Monoid Level: Unit in M|Z

```
Recall:

M|Z = { fₙ ∘ ... ∘ f₁ | all fᵢ: Xᵢ → Xᵢ₊₁ | Z ∧ ∀ i, j: fᵢ ≅ fⱼ | Z }

Now, define:

"" ∈ Ob(Concept)

and identity morphism under Z-frame:

e|Z_"" := id_"" | Z

Then for all f ∈ M|Z:

e|Z_"" ∘ f = f
f ∘ e|Z_"" = f
```

6.2. Finite Limits

Terminal Object Conceptual Topos defines a terminal object as the Z-frame identity:

```
id_Z := Z → Z | Z


Any morphism f: X → Z | Z factors uniquely through id_Z.

This realizes the conceptual universal target:

∀ X ∈ Ob(C ∪ D), ∃! f_terminal: X → Z | Z
```

Example:

```
she → human | Human
me → human | Human
```

Pullback

```
Given morphisms:

f: girl → mammal
g: puppy → mammal

Pullback of (f, g) is:

P = Baby
p₁: Baby → girl
p₂: Baby → puppy

with commuting condition:

f ∘ p₁ = g ∘ p₂  ≈ mapping to common conceptual frame (mammal)

Diagram:

      Baby
     /    \
  p₁ /      \ p₂
   /          \
girl          puppy
   \          /
    \        /
     v      v
     mammal (conceptual anchor / codomain)
```

This previously defined as Quasi-Natural Transformation:

```
η: Dᵢ ⇒ Dᵢ₊₁  | CD (CD = codomain)
η_X ∘ Dᵢ({f₁ | Z₁, ..., fₙ | Zₙ}) ≈ Dᵢ₊₁({f′₁ | Z₁, ..., f′ₙ | Zₙ}) ∘ η_Y | CD
for all fⱼ: Xⱼ → Yⱼ | Zⱼ ∈ Dᵢ,
where f′ⱼ: η_X(Xⱼ) → η_Y(Yⱼ) | Zⱼ

Then, η is said to be a quasi-natural transformation under the Z-frame
i.e. η ∈ Mor(C) where C is the contextual meaning category

η_X ∘ Dᵢ({girl → mammal | Z₁}) ≈ Dᵢ₊₁({puppy → mammal | Z₂}) ∘ η_Y


Pullback Diagram
      Mammal
     /     \
    /       \
girl         puppy
   \         /
    \       /
     Baby  (conceptual anchor / common Z-frame)


Example: η: girl → puppy | Z = Baby


Quasi-Natural Transformation Diagram:
        Z = baby
    puppy  ←   girl    //specified: size + young
     |          |
     |          |
    dog        she     //abstraction
      \        /
        Mammal


For any X with morphisms
q₁: X → girl and q₂: X → puppy satisfying f ∘ q₁ = g ∘ q₂,

there exists unique u: X → Baby

such that:
p₁ ∘ u = q₁, p₂ ∘ u = q₂.
```

Equalizer: Mirror Morphism

```
Equalizer of two morphisms f, g:
A → B is an object Eq(f,g) with morphism
e: Eq(f,g) → A

such that:

f ∘ e = g ∘ e

and universal property:

∀ h: X → A such that f ∘ h = g ∘ h,
∃! unique u: X → Eq(f,g)

Diagram:

      Eq(f, g)
         |
         e
         v
         A
    f ↘     ↙ g
        B
```

In conceptual topology this was defined as mirror morphism:

```
f : X  → Y  | Z ∈ Dᵢ
f′: X′ → Y' | Z ∈ Dᵢ₊₁
  ⇒X′ ≠ X, but cod(f) = cod(f′) | CD ( common codomain)

We define f′ as a mirror-correspondent morphism of f under a given Z-frame,
if and only if:

∃Z: rupture(f, f′ | Z) ≠ ∅
∧ cod(f) = cod(f′) | CD


        Eq(f, f′)
          |
          e
          v
      X        X′
       \      /
        \    /
         v  v
          Y = Y′ (codomain = C)
```

Product: σ operator⊕

```
In any category C, the Product of A and B is an object A × B equipped with
projections:

π₁: A × B → A
π₂: A × B → B

with universal property:
For any object X with morphisms:

f₁: X → A
f₂: X → B

there exists a unique morphism u: X → A × B such that:

π₁ ∘ u = f₁
π₂ ∘ u = f₂
```

Addition (⊕):
σ(Z) serves as the mediating operator ensuring that the composed morphic chain remains within the conceptual fiber over Z.

```
Defined as:
σ(Z). ⊕(Aₙ₋₁, Bₙ₋₁, Z) = D(Cₙ₋₁ | CD) = M_C|CD

Example:
Aₙ₋₁:= girl → she | Human
Bₙ₋₁:= puppy → dog | Canine

σ(Human). ⊕(girl → she, puppy → dog | Mammal) → M|Z(girl → she → puppy → dog |
Mammal) | Mammal
→ composite meaning space
```

```
Diagram:

    Product(girl → she, puppy → dog) ∈ D(Cₙ₋₁ | Z = Mammal)

             Product Object (P)
              /           \
             /             \
      π₁   /                 \  π₂
          v                   v
  girl → she            puppy → dog
```

6.3. Exponentials

Conceptual Topos models exponentials via conceptual shift operators.

Definition

```
For any objects A, B:

B^A exists such that:
Hom(X ⊗ A, B) ≅ Hom(X, B^A)
```

Construction via σ operator

Conceptual shift operators:

```
σ(Z). >>(A, B)
or
σ(Z). >(A, B)

act as internal exponential morphisms within the fibered structure over the Z-frame:
        (A, B, Z) ≅ B^A
where the Z-frame mediates the conceptual continuity and contextual grounding of the morphic shift.
```

We define Exponential objects via σ operator as conceptual abstraction mechanisms:

```
B^A:=σ(Z).>(A,B)
```

Exponentials as If: Conditional Statement

```
      X × A
        | \
        |  \
        |   \  f
        |    \
        v     v
     B^A × A → B
        |
      eval
```

In NL Diagram:

```
      X × "you press" ----
        |                 \
        |                  \
     λf |                   \ f
        |                    \
        v                     v
"press→open" × "you press"  → "it opens"
        |                     ^
        |_____________________|
                  eval


A: condition — "you press"
B: result — "it opens"
B^A: if A, then B: "press → open"
eval:B^A × A → B: "it opens"
λf: X → B^A: the result under the condition
  you press → extract the general law under the condition you press then "press →
open"
```

We formalize this as eval function in CTL(Conceptual Topology Language). We employ programing style to track dynamic transformation of conceptual flow.

```
def exponential(a: Concept, b: Concept, z: Frame) -> ConceptualMorphism::
  """
  Construct exponential object B^A under Z-frame.
  Represents: 'if a then b' interpreted within context c.
  """
  return σ(z).>(a, b)  // represents B^A

def eval(f: Conceptual Morphism, a: Concept, z: Frame) → Concept:
  return f × a | z

Example:
//if press then open → it opens
eval("press", "open", "door") → door → opened door | Door
```

```
//if press then open + you press → it opens
f = exponential("press", "open", "door")
eval(f, "you press") → open | Door
```

Full Exponential Law formalization will be provided in later version.

Definition: Conceptual Shifting Morphism (σ)

```
σ: D(Xₙ₋₁ | X) → D(Xₙ₋₁ | X)

such that σ ⊕ f ∈ M|Z if and only if type compatibility holds:

∀ A, B, (A → B) ∘ σ(X) is valid if:

( A >> X or X >> A )
and
( B >> X or X >> B )

Definition: Subsumption
A >> X ≡ A ⊑ X

Definition: SubsumedBy
X >> A ≡ X ⊑ A

Example:
king → king >> human → human
  ⇒ king >> human  → valid

human → human >> queen → queen
  ⇒ human >> queen → valid
```

Example

```
σ(Human). >>(puppy → dog → mammal | Canine, Human)
≅ girl → she → mammal | Human
```

This shift realizes an internal conceptual transformation corresponding to exponential behavior.

6.4. Definition of Ω

Let **Ω** be an object in the Concept category, representing the **conceptual truth space**.

```
For any subobject (conceptual inclusion):
```

```
m: M ↪ X

there exists a unique characteristic morphism:

χ_m: X → Ω
```

such that the following diagram commutes:

```
M ———→ X
|       |
|       v χ_m
|       Ω
```

Interpretation in Conceptual Topology

- **Ω** encodes **conceptual entailment / membership / inclusion**.
- **Z-frame membership** is naturally mapped to Ω:

```
χ_Z: X → Ω
```

interpreted as:
"Does X conceptually belong to Z-frame Z?"

Examples

Example 1: Dog in Pet Z-frame

χ_Pet(Dog) = True

Example 2: Apple in Pet Z-frame

χ_Pet(Apple) = False

Example 3: Innocent in Body Z-frame (after rupture)

χ_Body("innocent") = True / False depending on whether the conceptual projection is coherent under Z-Frame.

Relation to Rupture

Conceptual rupture can be lifted to Ω as:

σ(Z). Not(f: A → B | Z) ⇒ rupture(A,B,Z) ⇒ χ_Z(f) = False

Thus, **negation** and **conceptual discontinuity** become **Ω-classifiable**.

6.5. Conceptual Topos as Fibered Topos over Z-frame

Conceptual Topos is structured as a **fibered topos** over the conceptual base space **Z-frame**.

Z-frame as Fibered Structure

- Let π: C ∪ D → Z be the conceptual projection.
- Each fiber $\pi^{-1}(Z)$ forms a category of morphic chains **$D(C_{n-1} \mid Z)$**.
- Morphisms of the form:

```
X → Y | Z ≡ X → Y in fiber over Z
```

correspond to morphisms within the fibered structure over Z.

Initial Object and Codomain Projection

- The **Initial Object ""** serves as the conceptual origin.
- It projects into the codomain via:

```
"" → | X ≡ "" → π(X)
```

```
   ""
    ↓ u_X
  X ───→ π(X) (in Z-frame)
```

```
Fiber π⁻¹(Z_X):
    "" → X → Y
```

Thus, conceptual generation naturally occurs anchored in Z-frame.

Conceptual Flow Closure

- Conceptual flows:

```
X → Y | Z
```

are closed within the fiber over Z, corresponding to the codomain Z of the conceptual projection π.

- Rupture and negation are classified by **Ω**:

```
χ_Z: X → Ω
```

7. Global Conceptual Space: Total Conceptual Space (TCS)

We define the Total Conceptual Space (TCS) as the global conceptual anchor:

```
Z = TCS = Total Conceptual Space
```

Definition of M|TCS:

The global morphic flow space under TCS is defined as:

```
M|TCS = { fₙ ∘ ... ∘ f₁ | all fᵢ:  M|Zᵢ →  M|Zᵢ₊₁ | TCS ∧ ∀ i, j: fᵢ ≅ fⱼ | TCS }
```

We can regard M|TCS as the composition space of conceptual perspectives: Here, each M|Z functions as a conceptual symbolization or perspective lens, and M|TCS represents global flows across chained perspectives.

Monoid Closure Property:

```
Composition in M|TCS is closed:
∀ f, g ∈ M|TCS, f ∘ g ∈ M|TCS

The identity morphism is preserved:

∀ f ∈ M|TCS, f ∘ id = f = id ∘ f
```

Thus, M|TCS forms a closed monoid under composition.

Completeness Statement:

```
For any pair of concepts X, Y:

∀ X, Y ∈ Ob(C), ∃ f ∈ Mor(C), such that f: X → Y | TCS
```

That is, any conceptual pair X and Y can be connected via a morphic flow under TCS.

Fibered Structure and Lifting

Each local M|Z can be lifted into M I TCS via conceptual shifting σ:

```
∀MIZ, ∃σ: MIZ > MIZ | TCS
```

Thus, the global base space TCSTCSTCS ensures that the entire morphic flow space is both complete and coherent.

```
          M|Z₁     M|Z₂     M|Z₃
            \       |       /
             \      |      /
               →  M|TCS  ←
```

Example:

```
can → person | TCS
→ Metaphoric reading: "The can represents the absent person."
→ Ironic reading: "We are all cans under capitalism."
```

Summary:

The Total Conceptual Space (TCS) functions as the global base space of the conceptual topology. All local Z-frames are fibered over TCS, and conceptual flows can be lifted via σ operators into M I TCS. Thus, Conceptual Topos is complete and globally coherent under M I TCS.

8. Empirical Data for Conceptual Topology

8.1. Conceptual Flow Classification

We tested this framework, Conceptual Topology, using Principal Component Analysis(PCA).

First, we define the core conceptual flow diagram, interpreting Z as a retractive flow.

```
   || Z'||
X' ------→ Y'
↑    ↑     ↑
|    |σ(Z')|
|    |     |
X  ------→ Y
   || Z ||

f: X  → Y  | Z
g: X' → Y' | Z'

In 3 dimensional visualization

       Z’ Θ = Z_similarity
     /    \
    /      \
  X’ ------→ Y’
  ↑ || Z’ || ↑
  |          | σ(Z’)
  |          |
  X --------→ Y
    || Z || = Z disparity
```

For reference: In the composition A → Z → B, the mediating Z can be interpreted as the distance between A and B. We define this as Z = ||A − B||, and denote the norm as ||Z||. In retraction A → Z → A also behaves as a conceptual identity or coherent transformation via cosine similarity hinged on Z when Z is scalarized as ||Z|| under PCA projection.

```
f: A → Z
g: Z → A

such that
g ∘ f ≅ id_A | Z

   A
   | \
   |  \
   v   v
   Z → A   (g ∘ f = id_A | Z)
```

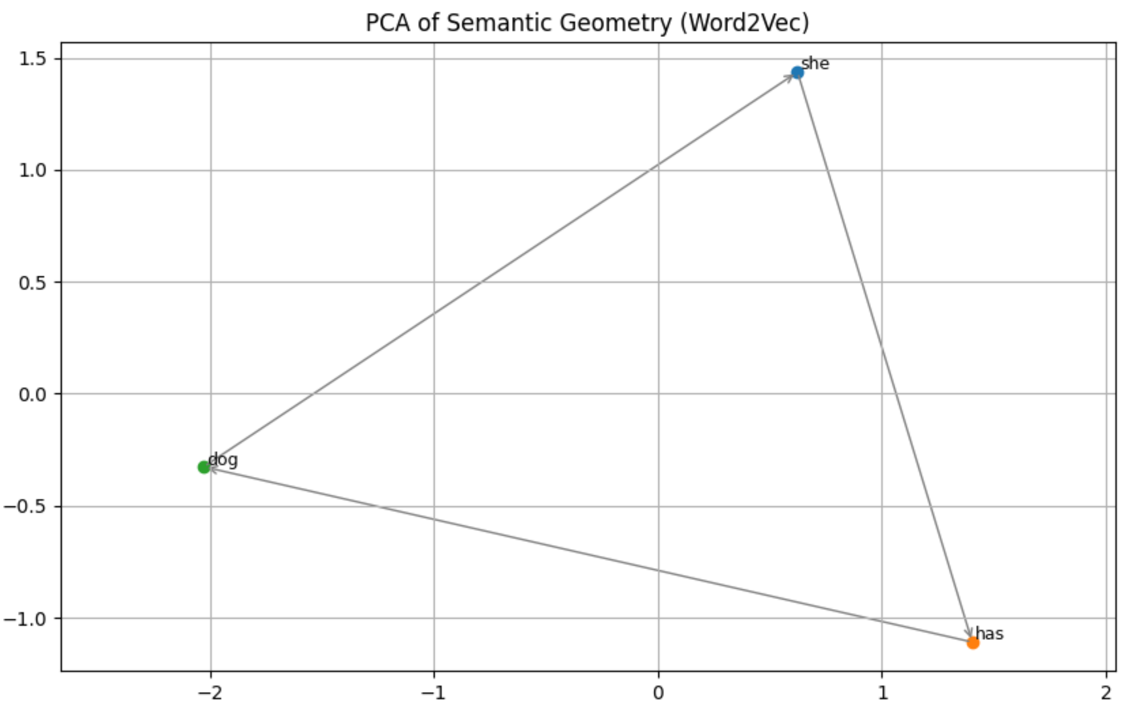
```
Alternatively, to express conceptual flow under Z
f: A → Z
g: Z → B
such that
g ∘ f ≅ A → B | Z
```

We categorize flow structures based on the type of morphism.

Self Identity Morphism: Triangle

```
   || Z'||
Z' ------→ Z'
↑    ↑     ↑
|    |σ(Z')|
|    |     |
A  ------→ B
   || Z ||

since || Z'|| = 0
     Z'
     ↑
   / | \
  /  |  \
 A ----→ B
  || Z ||
```

She has a dog



While multiple interpretations are possible, this diagram offers the most coherent explanation. The morphism σ(possession) unifies she and dog under the relation of ownership: the owner and the owned.

```
     has
      ↑
    / | \
   /  |  \
she ----→ dog
  || Z ||
```

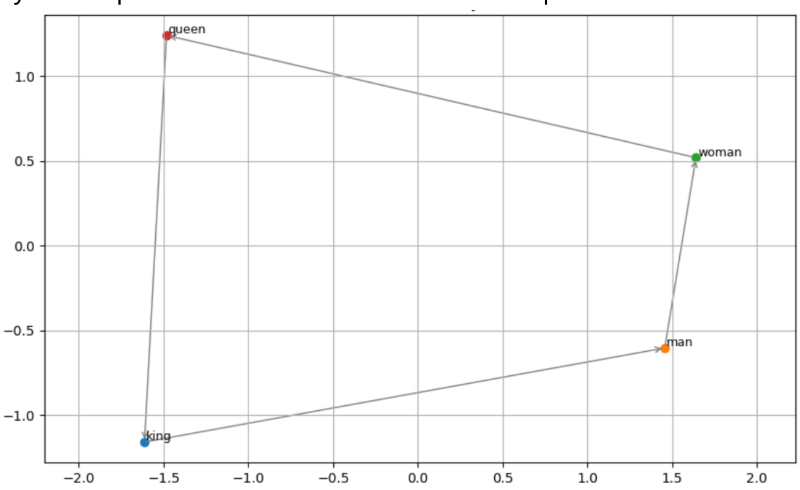
Trapezoid

```
   || Z'||
A' ------→ B'
↑    ↑     ↑
|    |σ(Z')|
|    |     |
A  ------→ B
   || Z ||

f: A  → B  | Z
g: A' → B' | Z'

since || Z'|| - || Z|| > 0

    || Z' ||
    A --→ B'
   /   ↑   \
  /    |    \
 A --------→ B
    || Z ||
```

king, man , woman, queen

The trapezoid structure observed in king, man, woman, and queen can be interpreted as preserving the same major Z-axis—in this case, gender. Note: Z may also represent a multi-dimensional conceptual frame.

```
          r
    Q ←────── W
    |         ↑
  f |         |h
    ↓         |
    K ──────→ M
          g

K = king
Q = queen
M = man
W = woman

g: Differential Morphism(King → Man)
h: Differential Morphism(Man → Woman)
r: Differential Morphism(Woman → Queen)
f: Composed Semantic Morphism r ∘ h ∘ g
```



Rectangle

```
   || Z'||
A' ------→ B'
↑    ↑     ↑
|    |σ(Z')|
|    |     |
A  ------→ B
   || Z ||

f: A  → B  | Z
g: A' → B' | Z'

since || Z'|| = || Z ||

   || Z'||
A' ------→ B'
↑    ↑     ↑
|    |σ(Z')|
|    |     |
A  ------→ B
   || Z ||
```

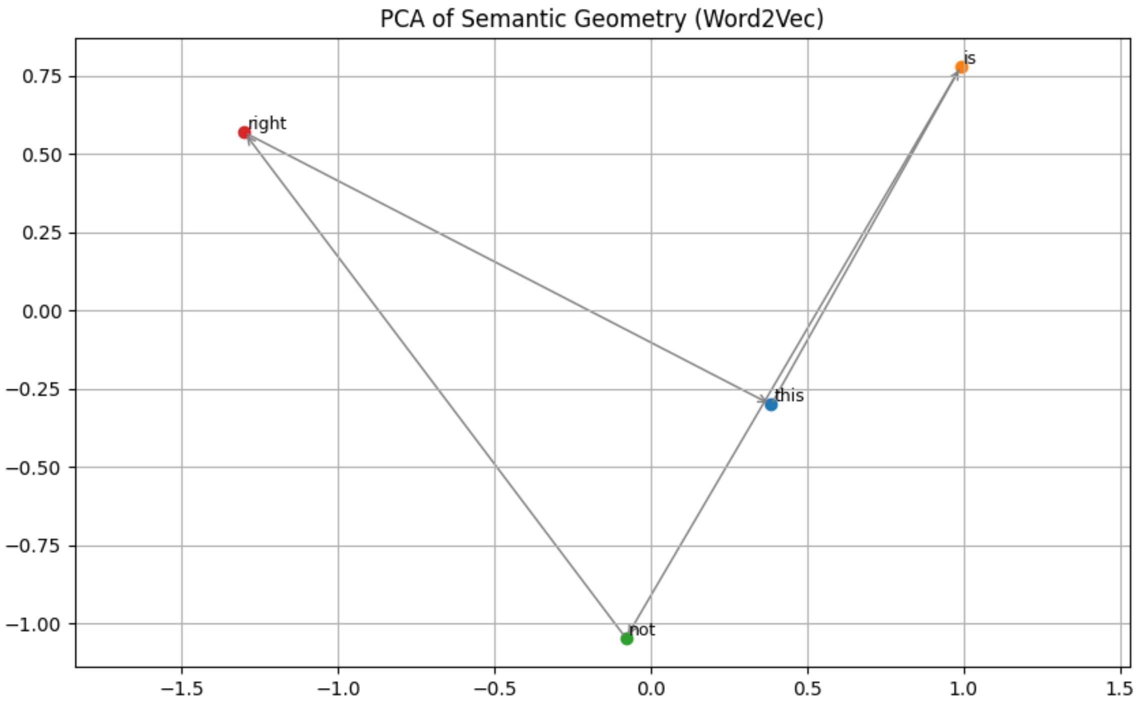
Rectangular structure signifies that Z and Z' is equivalent. This structural alignment suggests that the conceptual flow preserves its semantic frame, without requiring a shift in Z. Although the rectangular structure—corresponding to morphism preserving the conceptual frame Z—has not yet been observed in PCA projections, it remains a theoretically valid configuration. Detecting such a structure would signify complete semantic coherence between source and target morphisms.

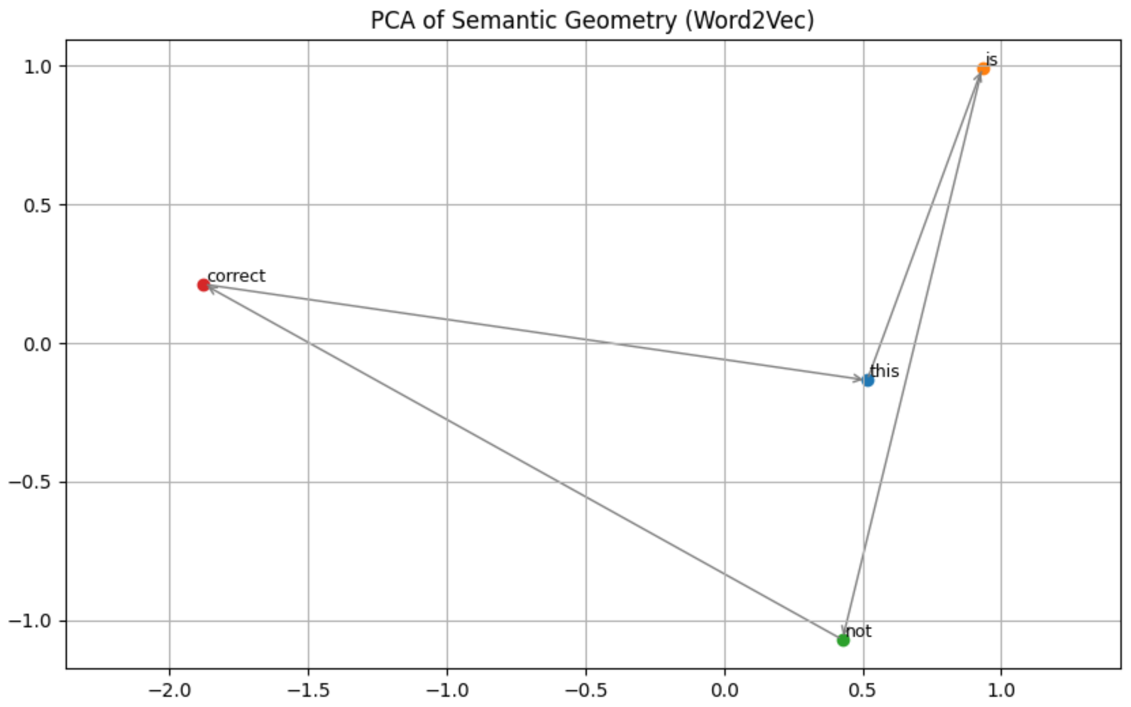
8.2. Zero Morphism

NL Diagram:

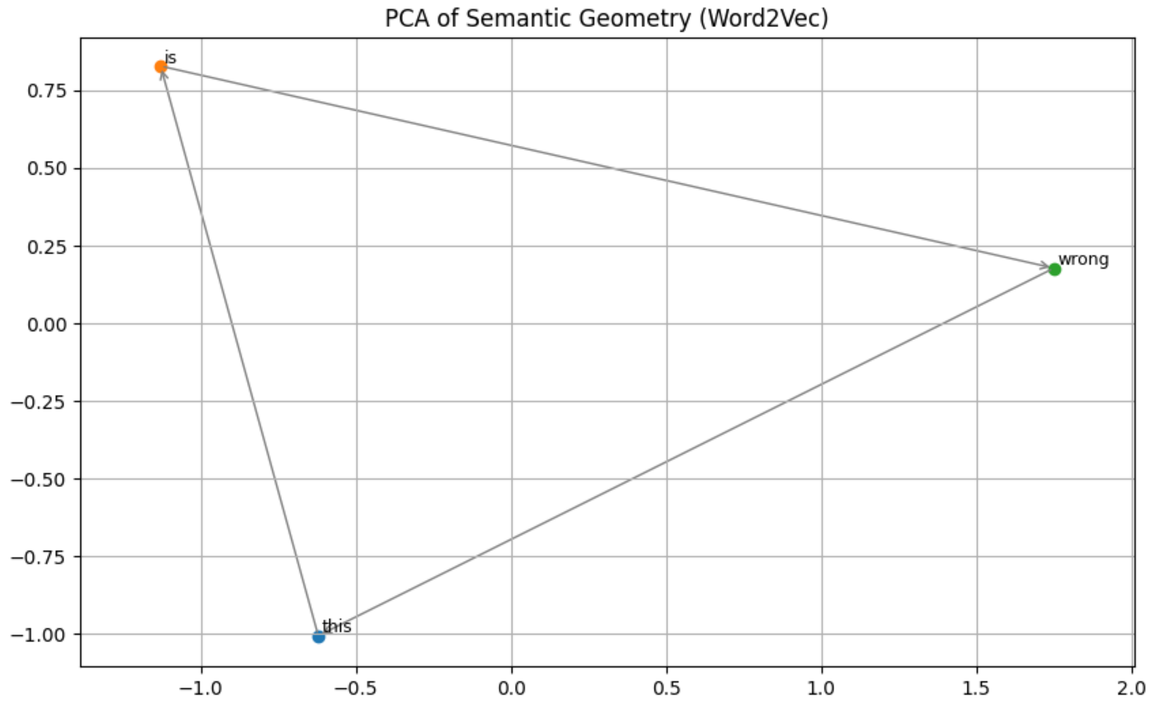
```
         f
this ───────→ correct
  │              ↑
  │              │
  └── n_f ──→ not

        n_f: zero morphism
    f ∘ n_f = n_f, n_f ∘ f = n_f
```

From the above formula, *is* was expelled from commutative structure and *not* replaced *is* completed conceptual circulation.





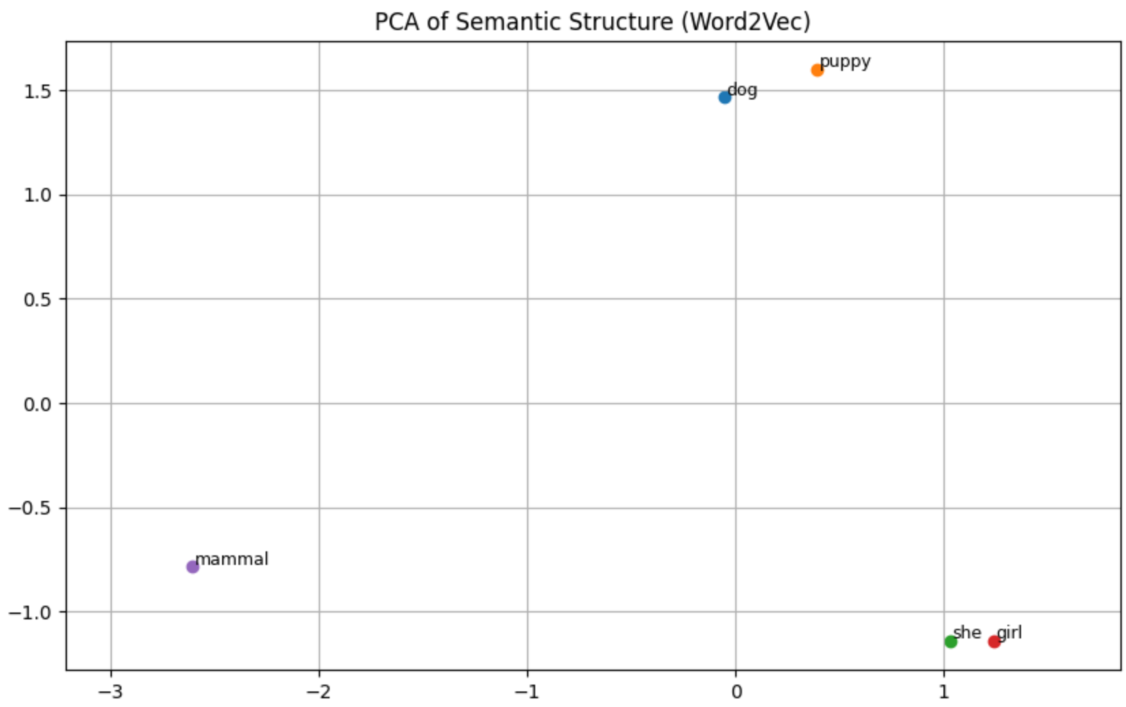
If a lexicon A which satisfy with *not B ⊆ A* such as *wrong* can recover the conceptual flow.



8.3. Cone Structre: Kan Extension and QNT

We can observe iterated colimit and QNT in PCA.

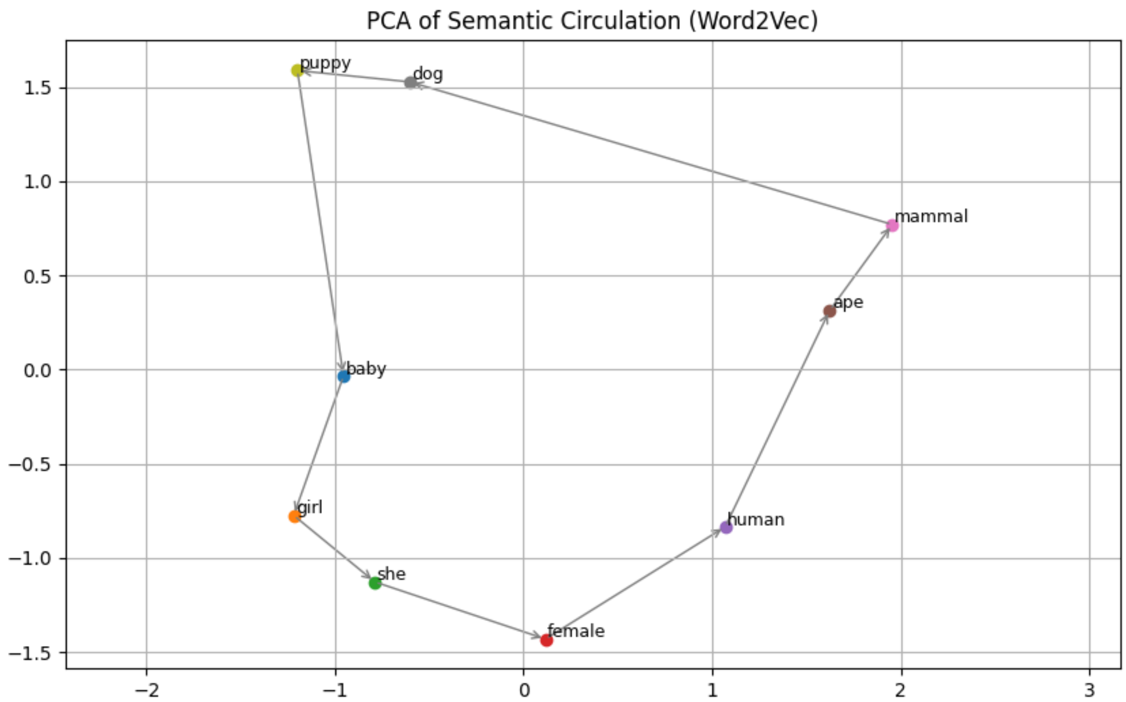
```
Dᵢ = colim_{puppy → dog → mammal} (Lan_{σᵢ}(π⁻¹(Mammal)))
Dᵢ₊₁ = colim_{girl → she → mammal} (Lan_{σᵢ}(π⁻¹(Mammal)))

Here Defined QNT Between Dᵢ and Dᵢ₊₁
η: Dᵢ ⇒ Dᵢ₊₁  | CD = Mammal
η_X ∘ Dᵢ({f₁ | Z₁, ..., fₙ | Zₙ}) ≈ Dᵢ₊₁({f′₁ | Z₁, ..., f′ₙ | Zₙ}) ∘ η_Y | CD
for all fⱼ: Xⱼ → Yⱼ | Zⱼ ∈ Dᵢ,
where f′ⱼ: η_X(Xⱼ) → η_Y(Yⱼ) | Zⱼ

Then, η is said to be a quasi-natural transformation under the Z-frame
i.e. η ∈ Mor(C) where C is the contextual meaning category
Example: η: girl → puppy | Z = Baby
```



8.4. Universal Product

We did not define universal product in the above sections, yet this is observable in PCA.

```
Let C be a category and let A,B ∈Ob(C) be objects.

A product of A and B is an object P∈Ob(C) together with morphisms
    π_A:P → A

such that for any object X∈Ob(C) with morphisms
    f:X→A, g:X→B

there exists a unique morphism ⟨f,g⟩
    X→P

such that the following diagram commutes
    π_A∘⟨f,g⟩=f, π_B∘⟨f,g⟩=g

        Mammal
        /   \
       /     \
      /       \
     f         g
    /           \
   v  P = Baby   v
Puppy <-- P --> Girl
     π_A       π_B
```



Conclusion

Conceptual Topos is a **fibered topos** over Z-frame:

```
CT := (C, B, π: E → B, Fb := π⁻¹(b), A ≅ b ∪ Nat(Hom(-, A), Fb))
```

with:

- Initial Object "" → codomain π(X)
- Morphic Chains as fibers $\pi^{-1}(Z)$
- Ω as subobject classifier in Z
- σ operator inducing internal exponential morphisms.

Appendix

simbols

```
Z : Intermediating variable (conceptual anchor; Z-frame)
| : Frame separator (indicates morphism is mediated by Z-frame)
→: Morphic Flow
↛ Ruptured morphism
F : Cross-category morphism (used in cross-category flow under shared Z-frame)
//: Used to narrate meaning flow of morphic chains.
¬: Absence

M|Z : Monoid of Conceptual Flow under Z-frame
R|Z := { rupture(f) | rupture(f, σ(f) | Z) ≠ ∅ }
e|Z : Identity element of M|Z
D(Aₙ₋₁ | Z) : Morphic chain under Z frame

σ : Conceptual Shifting Morphism
>> : Generalization relation (A >> X ≡ A ⊑ X)
<< : Specialization relation (X >> A ≡ X ⊑ A)
rupture(f, σ(f) | Z) ≠ ∅ : Indicates conceptual rupture
η : Quasi-Natural Transformation: Contextual alignment between morphic chains.

⊕:  Conceptual morphism set addition in σ or morphic merger such as:
    (k₂ ∘ k₁) ⊕ (q₂ ∘ q₁) = human → royalty | Z'
⊖:  Conceptual morphism set subtraction
    Removes specified morphisms from a morphic chain or set.
```

Notations

```
Concept / Word (lexeme):
    - Lower case (e.g., puppy, dog, girl, she)

Z Frame (conceptual anchor):
    - Upper case (e.g., Mammal, Human, Agency, Domesticated, Royalty)

Type variables (A, B, X, Y, Z in formal definitions):
    - Follow standard formal notation (uppercase)

Example:
puppy → dog | Mammal
A → B | Z

Morphism: f, g, h
Functor: F
```

Simplified Form of Identity Morphism:

1. f: X → X | X (Category-theoretic identity)
   In simplified form: X
   or more explicitly: id_X

2. f: X → X | Z (Mediated identity with conceptual flow)
   In simplified form: X | Z

σ Operator

```
σ(X). Not(x){ A ↛B | Z}      →     Rupture under Z frame
σ(X). so_much(x){A → B | Z}  →     Preservation & amplification under Z frame
σ(X). >>(x,y)                →     Conceptual Shifting x to y (Generalization) as function form
σ(X). <<(x,y)                →     Downward Shifting x to y (Specialization) as function form
σ(X). >(x,y)                 →     Conceptual Shifting
```

Conceptual Morphism Set Operators

```
Addition (⊕):
σ(X). ⊕(f, Aₙ₋₁ | Z): D(Aₙ₋₁ | Z) → D(Bₙ₋₁ | Z) | Z
σ(X). ⊕(f₁, f₂) : Aₙ₋₁ :={f₁, f₂}

Subtraction (⊖):
⊖: Aₙ₋₁ ⊖ {f_i}
σ(X). ⊖(f, Aₙ₋₁ | Z): D(Aₙ₋₁ | Z) → D(Bₙ₋₁ | Z) | Z

- ⊕ operator is σ_safe if Z alignment is preserved.
- ⊖ operator is potentially σ_unsafe but can be σ_safe if resulting chain preserves the underlying morphic continuity Z.
```

σ Typing Hierarchy

σ_safe: D(A_{n-1} | Z) → D(B_{n-1} | Z) | Z (Preserves global coherence)
σ_unsafe: D(A_{n-1} | Z) → { rupture(f_1), ..., rupture(f_n) | ¬Z } (Global coherence lost)

Note: σ_safe behaves as Quasi-Natural Transformation.
σ_unsafe induces rupture, and cannot be captured globally.

Python Code Used in This Study

```
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
from gensim.models import KeyedVectors
import numpy as np

model_path = ''
model = KeyedVectors.load_word2vec_format(model_path, binary=True)

words = ["this","dog"]


# ベクトル取得
vectors = [model[word] for word in words]
labels = words

# 次元削減（PCA）
pca = PCA(n_components=2)
reduced = pca.fit_transform(vectors)

# プロット
plt.figure(figsize=(10, 6))
for i, label in enumerate(labels):
    x, y = reduced[i]
    plt.scatter(x, y)
    plt.text(x + 0.01, y + 0.01, label, fontsize=9)

#矢印付加

from matplotlib.patches import FancyArrowPatch
for i in range(len(reduced) - 1):
    start = reduced[i]
    end = reduced[i + 1]
    arrow = FancyArrowPatch(start, end, arrowstyle='->', mutation_scale=10,
color='gray')
    plt.gca().add_patch(arrow)

start = reduced[len(reduced)-1]
end = reduced[0]
arrow = FancyArrowPatch(start, end, arrowstyle='->', mutation_scale=10,
color='gray')
plt.gca().add_patch(arrow)

plt.title("PCA of Semantic Geometry (Word2Vec)")
plt.grid(True)
plt.axis("equal")
plt.savefig("image.png")
plt.show()
```

Word2Vec Data Set

https://code.google.com/archive/p/word2vec/

Conceptual Topos Named as 概念位相論 / Conceptual Topology

This theory, named 概念位相論 or Conceptual Topoloy, was proposed by **No Name Yet Exist**.

Meaning no longer escapes.
It circulates within the morphic fibration.
We, once again, govern the topology of meaning.

GitHub: https://github.com/No-Name-Yet-Exist/Conceptual-Topology
Note: https://note.com/xoreaxeax/n/n3711c1318d0b
Zenodo: https://zenodo.org/records/15455079

This is Version: 1.4.3
