Reference Manual

Logos: A Logic for Verified and Interpreted AI Reasoning

January 11, 2026

Benjamin Brast-McKie

www.benbrastmckie.com

Primary References:

Brast-McKie, *Counterfactual Worlds*, *J. Phil. Logic*, 2025.

Brast-McKie, *Identity and Aboutness*, *J. Phil. Logic*, 2021.

Abstract

This reference manual provides the formal specification of the Logos logic system. Logos is a hyperintensional modal-temporal logic with exact truthmaker semantics, designed for reasoning about necessity, possibility, time, and counterfactual conditionals. The semantics extends from a mereological state space foundation through increasingly expressive extensions.

Contents

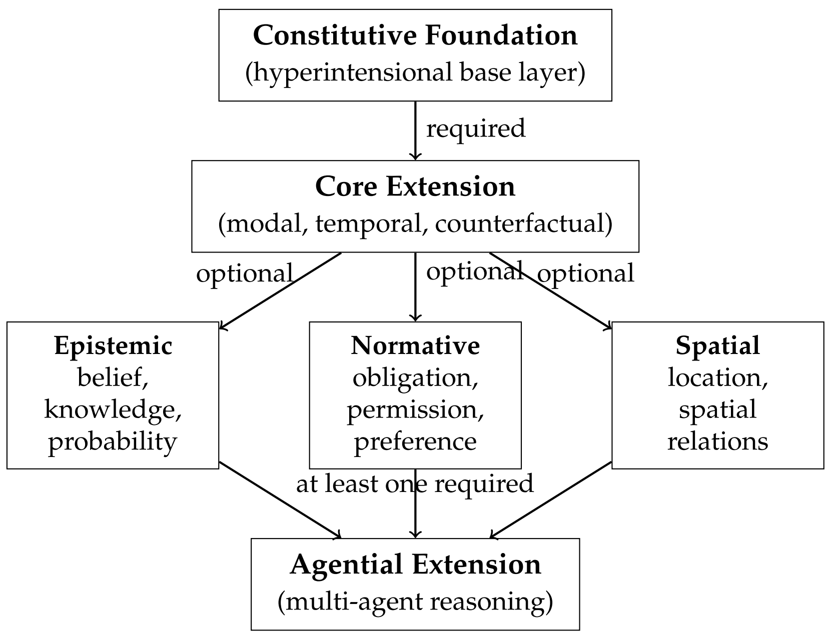
1 Introduction

A semantic frame provides the primitive structures used to interpret a formal language. Extending the expressive power of a language requires strategic extensions to the primitive semantic resources provided by the frame, including precisely the resources needed and nothing more. This ensures that language and frame remain in perfect step with each other.

This reference manual provides the formal specification of the Logos logic system. The semantics proceeds through increasingly expressive extensions, each extending the frame and evaluation mechanisms of the previous.

1.1 Extension Dependencies

The following diagram shows the dependency structure among extensions:



The Constitutive Foundation and Core Extension form the required base. The Epistemic, Normative, and Spatial Extensions are modular plugins that can be combined in any subset. The Agential Extension requires at least one middle extension to be loaded.

1.2 Layer Descriptions

1. **Constitutive Foundation**: Hyperintensional semantics over a mereological state space. Provides the foundational structure with bilateral propositions (verifier/falsifier pairs).

2. **Core Extension**: Hyperintensional and intensional semantics over a task space. Extends the foundation with temporal structure (a totally ordered abelian group) and a task relation constraining possible state transitions, enabling evaluation of truth relative to world-histories and times.

3. **Epistemic Extension**: Extensions for belief, knowledge, and probability. [DETAILS PENDING DEVELOPMENT]

4. **Normative Extension**: Extensions for obligation, permission, and value. [DETAILS PENDING DEVELOPMENT]

5. **Spatial Extension**: Extensions for spatial reasoning. [DETAILS PENDING DEVELOPMENT]

6. **Agential Extension**: Extensions for multi-agent reasoning. [DETAILS PENDING DEVELOPMENT]

1.3 Document Organization

This reference manual is organized as follows:

?? presents the Constitutive Foundation, including the mereological state space, verification and falsification clauses, and bilateral propositions.

?? introduces the syntactic primitives of the Core Extension, including modal, temporal, and counterfactual operators.

?? provides the semantic framework for the Core Extension, including core frames, world-histories, and truth conditions.

?? presents the axiom system for counterfactual logic.

1.4 Notation

This document uses the following notational conventions:

- σ denotes variable assignments (compatible across LaTeX, markdown, and Lean notation)

- τ denotes world-histories
- $\vec{\imath} = \langle i_1, i_2, \ldots \rangle$ denotes the temporal index (vector of stored times)
- $t \rightarrow_w$ denotes imposition ("imposing t on w yields w'")

See the `logos-notation.sty` package for the complete list of notation macros.

1.5 Lean Implementation

The Logos system is implemented in Lean 4 with Mathlib. Each definition in this reference has a corresponding Lean implementation, referenced using the `\leansrc{}{}` command.

For example, `Logos.Foundation.Frame.ConstitutiveFrame` refers to the `ConstitutiveFrame` definition in the `Logos.Foundation.Frame` module.

The implementation can be found in the `Logos/` directory of the ProofChecker repository.

2 Constitutive Foundation

The Constitutive Foundation provides the foundational semantic structure based on exact truthmaker semantics. Evaluation is hyperintensional, distinguishing propositions that agree on truth-value across all possible worlds but differ in their exact verification and falsification conditions.

2.1 Syntactic Primitives

The Constitutive Foundation interprets the following syntactic primitives:

- **Variables**: $x, y, z, \ldots$ (ranging over states)
- **Individual constants**: $a, b, c, \ldots$ (0-place function symbols)
- **n-place function symbols**: $f, g, h, \ldots$
- **n-place predicates**: $F, G, H, \ldots$
- **Sentence letters**: $p, q, r, \ldots$ (0-place predicates)
- **Lambda abstraction**: $\lambda x.A$ (binding variable x in formula A)
- **Logical connectives**: $\neg, \wedge, \vee, \top, \bot, \equiv$

2.2 Constitutive Frame

Definition 2.1 (Constitutive Frame)**.** A *constitutive frame* is a structure $\mathbf{F} = \langle S, \sqsubseteq \rangle$ where:

- S is a nonempty set of *states*
- $\sqsubseteq$ is a partial order on S making $\langle S, \sqsubseteq \rangle$ a complete lattice

Remark 2.2. The constitutive frame is non-modal: possibility and compatibility cannot be defined at this level since they require the task relation introduced in the Core Extension.

Remark 2.3. The lattice structure provides:

- **Null state** $\square$: The bottom element (fusion of the empty set)
- **Full state** $\blacksquare$: The top element (fusion of all states)
- **Fusion** $s \cdot t$: The least upper bound of s and t

See `Logos.Foundation.Frame.ConstitutiveFrame` for the Lean implementation.

2.3 Constitutive Model

Definition 2.4 (Constitutive Model)**.** A *constitutive model* is a structure $\mathbf{M} = \langle S, \sqsubseteq, I \rangle$ where:

- $\langle S, \sqsubseteq \rangle$ is a constitutive frame
- I is an interpretation function assigning meanings to non-logical vocabulary

Definition 2.5 (Interpretation Function)**.** The interpretation function I assigns:

- **n-place function symbols** $f \mapsto I(f) : S^n \to S$ (0-place symbols are individual constants mapping to states)
- **n-place predicates** $F \mapsto \langle v_F, f_F \rangle$ where:
  - v_F: set of functions $S^n \to S$ (verifier functions)
  - f_F: set of functions $S^n \to S$ (falsifier functions)
- **Sentence letters** (0-place predicates) $p \mapsto \langle v_p, f_p \rangle$ of verifier and falsifier state sets

Remark 2.6 (Containment Constraint)*.* For any function f in v_F or f_F and any n states $a_1, \ldots, a_n$, these states are all parts of $f(a_1, \ldots, a_n)$. However, $f(a_1, \ldots, a_n)$ may have additional parts beyond the input states.

Remark 2.7 (Predicate Intuition)*.* For 1-place predicates, the functions in v_F and f_F take an object (which is itself a state) as input and return that object instantiating a verifying or falsifying property instance for the property in question.

See `Logos.Foundation.Model.ConstitutiveModel` for the Lean implementation.

2.4 Variable Assignment

Definition 2.8 (Variable Assignment)**.** A *variable assignment* σ is a function from variables to states: $\sigma : \mathrm{Var} \to S$

Notation 2.9*.* Greek letters $(\tau, \alpha, \beta, \ldots)$ are reserved for world-histories. The letter σ (with subscripts $\sigma_1, \sigma_2, \ldots$) denotes variable assignments, chosen for compatibility across LaTeX, markdown, and Lean notation.

Definition 2.10 (Term Extension)**.** The *extension* of a term relative to model $\mathbf{M}$ and assignment σ:

- **Variable** x: $[\![x]\!]^\sigma_{\mathbf{M}} = \sigma(x)$
- **Function application** $f(t_1, \ldots, t_n)$: $[\![f(t_1, \ldots, t_n)]\!]^\sigma_{\mathbf{M}} = I(f)([\![t_1]\!]^\sigma_{\mathbf{M}}, \ldots, [\![t_n]\!]^\sigma_{\mathbf{M}})$

See `Logos.Foundation.Assignment.Assignment` for the Lean implementation.

2.5 Verification and Falsification Clauses

A state s *verifies* ($\Vdash^+$) or *falsifies* ($\Vdash^-$) a formula A relative to model $\mathbf{M}$ and assignment σ:

2.5.1 Atomic Formulas

Definition 2.11 (Atomic Verification and Falsification)**.**

$$\mathbf{M}, \sigma, s \Vdash^+ F(t_1, \ldots, t_n) \iff \exists f \in v_F : s = f([\![t_1]\!]^\sigma_{\mathbf{M}}, \ldots, [\![t_n]\!]^\sigma_{\mathbf{M}}) \quad (1)$$
$$\mathbf{M}, \sigma, s \Vdash^- F(t_1, \ldots, t_n) \iff \exists f \in f_F : s = f([\![t_1]\!]^\sigma_{\mathbf{M}}, \ldots, [\![t_n]\!]^\sigma_{\mathbf{M}}) \quad (2)$$

2.5.2 Lambda Abstraction

Definition 2.12 (Lambda Verification and Falsification)**.**

$$\mathbf{M}, \sigma, s \Vdash^{+} (\lambda x.A)(t) \iff \mathbf{M}, \sigma[[\![t]\!]^{\sigma}_{\mathbf{M}}/x], s \Vdash^{+} A \tag{3}$$
$$\mathbf{M}, \sigma, s \Vdash^{-} (\lambda x.A)(t) \iff \mathbf{M}, \sigma[[\![t]\!]^{\sigma}_{\mathbf{M}}/x], s \Vdash^{-} A \tag{4}$$

where $\sigma[v/x]$ is the assignment that maps x to v and agrees with σ on all other variables.

2.5.3 Negation

Definition 2.13 (Negation Verification and Falsification)**.**

$$\mathbf{M}, \sigma, s \Vdash^{+} \neg A \iff \mathbf{M}, \sigma, s \Vdash^{-} A \tag{5}$$
$$\mathbf{M}, \sigma, s \Vdash^{-} \neg A \iff \mathbf{M}, \sigma, s \Vdash^{+} A \tag{6}$$

2.5.4 Conjunction

Definition 2.14 (Conjunction Verification and Falsification)**.**

$$\mathbf{M}, \sigma, s \Vdash^{+} A \wedge B \iff s = t \cdot u \text{ for some } t, u \text{ where } \mathbf{M}, \sigma, t \Vdash^{+} A \text{ and } \mathbf{M}, \sigma, u \Vdash^{+} B \tag{7}$$

$$\begin{aligned}\mathbf{M}, \sigma, s \Vdash^{-} A \wedge B \iff\ & \mathbf{M}, \sigma, s \Vdash^{-} A, \text{ or } \mathbf{M}, \sigma, s \Vdash^{-} B, \text{ or} \\ & s = t \cdot u \text{ for some } t, u \text{ where } \mathbf{M}, \sigma, t \Vdash^{-} A \text{ and } \mathbf{M}, \sigma, u \Vdash^{-} B\end{aligned} \tag{8}$$

2.5.5 Disjunction

Definition 2.15 (Disjunction Verification and Falsification)**.**

$$\begin{aligned}\mathbf{M}, \sigma, s \Vdash^{+} A \vee B \iff\ & \mathbf{M}, \sigma, s \Vdash^{+} A, \text{ or } \mathbf{M}, \sigma, s \Vdash^{+} B, \text{ or} \\ & s = t \cdot u \text{ for some } t, u \text{ where } \mathbf{M}, \sigma, t \Vdash^{+} A \text{ and } \mathbf{M}, \sigma, u \Vdash^{+} B\end{aligned} \tag{9}$$

$$\mathbf{M}, \sigma, s \Vdash^{-} A \vee B \iff s = t \cdot u \text{ for some } t, u \text{ where } \mathbf{M}, \sigma, t \Vdash^{-} A \text{ and } \mathbf{M}, \sigma, u \Vdash^{-} B \tag{10}$$

2.5.6 Top and Bottom

Definition 2.16 (Top and Bottom Verification and Falsification)**.**

$$\mathbf{M}, \sigma, s \Vdash^{+} \top \text{ for all } s \in S \tag{11}$$
$$\mathbf{M}, \sigma, s \Vdash^{-} \top \iff s = \blacksquare \text{ (full state)} \tag{12}$$
$$\mathbf{M}, \sigma, s \nVdash^{+} \bot \text{ for all } s \tag{13}$$
$$\mathbf{M}, \sigma, s \Vdash^{-} \bot \iff s = \square \text{ (null state)} \tag{14}$$

2.5.7 Propositional Identity

Definition 2.17 (Propositional Identity Verification and Falsification)**.**

$$\begin{aligned}\mathbf{M}, \sigma, s \Vdash^{+} A \equiv B \iff{}& s = \square \text{ and } \{t : \mathbf{M}, \sigma, t \Vdash^{+} A\} = \{t : \mathbf{M}, \sigma, t \Vdash^{+} B\} \\ & \text{and } \{t : \mathbf{M}, \sigma, t \Vdash^{-} A\} = \{t : \mathbf{M}, \sigma, t \Vdash^{-} B\} \end{aligned} \tag{15}$$
$$\begin{aligned}\mathbf{M}, \sigma, s \Vdash^{-} A \equiv B \iff{}& s = \square \text{ and } (\{t : \mathbf{M}, \sigma, t \Vdash^{+} A\} \neq \{t : \mathbf{M}, \sigma, t \Vdash^{+} B\} \\ & \text{or } \{t : \mathbf{M}, \sigma, t \Vdash^{-} A\} \neq \{t : \mathbf{M}, \sigma, t \Vdash^{-} B\}) \end{aligned} \tag{16}$$

See `Logos.Foundation.Semantics.verifies` for the Lean implementation.

2.6 Essence and Ground

These constitutive relations can be defined via propositional identity:

Definition 2.18 (Essence and Ground)**.**

$$A \sqsubseteq B := A \wedge B \equiv B \qquad \text{“}A \text{ is essential to } B\text{” (conjunctive part)} \tag{17}$$
$$A \leq B := A \vee B \equiv B \qquad \text{“}A \text{ grounds } B\text{” (disjunctive part)} \tag{18}$$

Remark 2.19 (Negation Distribution). These relations are interrelated through negation:

- $A \sqsubseteq B$ iff $\neg A \leq \neg B$
- $A \leq B$ iff $\neg A \sqsubseteq \neg B$

Remark 2.20 (Bilattice Structure). The space of bilateral propositions forms a non-interlaced bilattice where:

- $\langle P, \sqsubseteq \rangle$ and $\langle P, \leq \rangle$ are complete lattices
- Negation exchanges the two orderings: $X \leq Y$ iff $\neg X \sqsubseteq \neg Y$

- Conjunction and disjunction are the least upper bounds with respect to $\sqsubseteq$ and $\leq$ respectively

See `Logos.Foundation.Relations.essence` for the Lean implementation.

2.7 Bilateral Propositions

Definition 2.21 (Bilateral Proposition)**.** A *bilateral proposition* is an ordered pair $\langle V, F \rangle$ where:

- V and F are subsets of S closed under fusion
- $\langle V, F \rangle$ is *exclusive*: states in V are incompatible with states in F
- $\langle V, F \rangle$ is *exhaustive*: every possible state is compatible with some state in V or F

Definition 2.22 (Propositional Operations)**.**

$$X \otimes Y := \{s \cdot t : s \in X, t \in Y\} \qquad \text{(Product)} \qquad (19)$$
$$X \oplus Y := X \cup Y \cup (X \otimes Y) \qquad \text{(Sum)} \qquad (20)$$
$$\langle V, F \rangle \wedge \langle V', F' \rangle := \langle V \otimes V', F \oplus F' \rangle \qquad \text{(Conjunction)} \qquad (21)$$
$$\langle V, F \rangle \vee \langle V', F' \rangle := \langle V \oplus V', F \otimes F' \rangle \qquad \text{(Disjunction)} \qquad (22)$$
$$\neg \langle V, F \rangle := \langle F, V \rangle \qquad \text{(Negation)} \qquad (23)$$

See `Logos.Foundation.Proposition.BilateralProp` for the Lean implementation.

2.8 Logical Consequence

Definition 2.23 (Logical Consequence (Constitutive Foundation))**.** Logical consequence at the Constitutive Foundation is restricted to propositional identity sentences:

$$\Gamma \models A \iff \text{for any model } \mathbf{M} \text{ and assignment } \sigma : \text{if } \mathbf{M}, \sigma, \square \Vdash^+ B \text{ for all } B \in \Gamma, \text{ then } \mathbf{M}, \sigma, \square \Vdash^+ A$$

That is, A is a consequence of Γ iff the null state verifies A in any model where it verifies all premises.

Remark 2.24 (Identity Sentences and Evaluation Overlap). Identity sentences are formed from extensional (non-identity) sentences: $A \equiv B$ where A and B are atomic sentences or built from $\neg, \wedge, \vee$. The logical consequences holding between identity sentences are preserved by further extensions. However, the Constitutive Foundation lacks the semantic resources to evaluate non-identity sentences, which depend on contingent states of affairs rather than purely structural relations in state space. The Constitutive Foundation is nevertheless important for defining a logic of propositional identity. The same theorems valid at this level remain valid in the Core Extension, though the Core Extension's definition of logical consequence differs, quantifying over world-histories and times in addition to models and variable assignments.

3 Core Extension: Syntactic Primitives

The Core Extension extends the Constitutive Foundation with temporal structure and a task relation, enabling evaluation of truth relative to world-histories and times. While the hyperintensional foundation remains (distinguishing propositions by their exact verifiers and falsifiers), this extension adds intensional evaluation relative to contextual parameters (world-history, time, variable assignment) to determine truth-values for all Core Extension sentences.

3.1 Additional Syntactic Primitives

The Core Extension interprets the following additional syntactic primitives beyond those of the Constitutive Foundation:

- **Modal operators**: $\Box$ (necessity), $\Diamond$ (possibility)
- **Temporal operators**: H (always past), G (always future), P (some past), F (some future)
- **Extended temporal operators**: $\triangleleft$ (since), $\triangleright$ (until)
- **Counterfactual conditional**: $\Box\!\!\rightarrow$ (would-counterfactual)
- **Store/recall operators**: $\uparrow^i$ (store), $\downarrow^i$ (recall)

3.2 Well-Formed Sentences

A well-formed sentence at this extension includes:

1. All Constitutive Foundation sentences (atomic formulas, $\neg$, $\wedge$, $\vee$, $\top$, $\bot$, $\equiv$)
2. Modal sentences: $\Box A$, $\Diamond A$

3. Temporal sentences: HA, GA, PA, FA
4. Extended temporal sentences: $A \triangleleft B$, $A \triangleright B$
5. Counterfactual sentences: $\varphi \Box\!\!\rightarrow C$
6. Indexed sentences: $\uparrow^i A$, $\downarrow^i A$
7. Universal quantification: $\forall x.A$
8. Actuality predicate: $\mathsf{Act}(t)$

3.3 Operator Summary

Category	**Symbol**	**Reading**
Modal	$\Box A$	"Necessarily A"
	$\Diamond A$	"Possibly A"
Past Temporal	HA	"It has always been that A"
	PA	"It was the case that A"
Future Temporal	GA	"It will always be that A"
	FA	"It will be the case that A"
Extended	$A \triangleleft B$	"A since B"
	$A \triangleright B$	"A until B"
Counterfactual	$\varphi \Box\!\!\rightarrow C$	"If φ were the case, then C"
Store/Recall	$\uparrow^i A$	"Store current time and evaluate A"
	$\downarrow^i A$	"Recall stored time and evaluate A"

Table 1: Core Extension operators

3.4 Derived Operators

Definition 3.1 (Derived Temporal Operators)**.**

$$\triangle A := HA \wedge A \wedge GA \qquad \text{“Always } A\text{” (at all times)} \tag{24}$$

$$\nabla A := PA \vee A \vee FA \qquad \text{“Sometimes } A\text{” (at some time)} \tag{25}$$

Definition 3.2 (Material Conditional)**.**

$$A \rightarrow B := \neg A \vee B$$

Definition 3.3 (Biconditional)**.**

$$A \leftrightarrow B := (A \to B) \land (B \to A)$$

See `Logos.Core.Syntax.CoreFormula` for the Lean implementation.

4 Core Extension: Semantic Framework

4.1 Core Frame

Definition 4.1 (Core Frame)**.** A *core frame* is a structure $\mathbf{F} = \langle S, \sqsubseteq, D, \Rightarrow \rangle$ where:

- $\langle S, \sqsubseteq \rangle$ is a constitutive frame
- $D = \langle D, +, \leq \rangle$ is a totally ordered abelian group
- $\Rightarrow$ is a ternary relation on $S \times D \times S$ satisfying the constraints below

Notation 4.2. We write $s \Rightarrow_d t$ to denote that $(s, d, t) \in \Rightarrow$, read "there is a task from s to t of duration d."

4.2 State Modality Definitions

Definition 4.3 (State Modality)**.** Let $\mathbf{F} = \langle S, \sqsubseteq, D, \Rightarrow \rangle$ be a core frame.

$$
\begin{aligned}
s \in P &\iff s \Rightarrow_0 s && \text{(Possible state)} && (26)\\
s \notin P &\iff \neg(s \Rightarrow_0 s) && \text{(Impossible state)} && (27)\\
s \sim t &\iff s \Rightarrow_d t \text{ or } t \Rightarrow_d s \text{ for some } d && \text{(Connected states)} && (28)\\
s \circ t &\iff s \cdot t \in P && \text{(Compatible states)} && (29)\\
w \in W &\iff w \text{ is a maximal possible state} && \text{(World-state)} && (30)\\
s \in N &\iff s \sim t \text{ implies } s = t && \text{(Necessary state)} && (31)
\end{aligned}
$$

Definition 4.4 (Maximal Compatible Part)**.** The set of *maximal t-compatible parts* of r is:

$$r_t := \{s : s \sqsubseteq r, s \circ t, \text{ and } s' \sqsubseteq s \text{ for all } s' \text{ where } s \sqsubseteq s' \sqsubseteq r \text{ and } s' \circ t\}$$

Remark 4.5. A world-state $w \in W$ is maximal in the sense that $t \sqsubseteq w$ for every compatible state $t \circ w$.

4.3 Task Relation Constraints

Definition 4.6 (Task Relation Constraints)**.** The task relation $\Rightarrow$ satisfies:

1. **Compositionality**: If $s \Rightarrow_x t$ and $t \Rightarrow_y u$, then $s \Rightarrow_{x+y} u$
2. **Parthood (Left)**: If $d \sqsubseteq s$ and $s \Rightarrow_x t$, then $d \Rightarrow_x r$ for some $r \sqsubseteq t$
3. **Parthood (Right)**: If $r \sqsubseteq t$ and $s \Rightarrow_x t$, then $d \Rightarrow_x r$ for some $d \sqsubseteq s$
4. **Containment (Left)**: If $s \in P$, $d \sqsubseteq s$, and $d \Rightarrow_x r$, then $s \Rightarrow_x t \cdot r$ for some $t \in S$
5. **Containment (Right)**: If $t \in P$, $r \sqsubseteq t$, and $d \Rightarrow_x r$, then $s \cdot d \Rightarrow_x t$ for some $s \in S$
6. **Maximality**: If $s \in S$ and $t \in P$, there is a maximal t-compatible part $r \in s_t$

Remark 4.7. The Containment constraints ensure that tasks between parts of possible states can be extended to tasks between the states themselves. The Maximality constraint ensures that for any state and possible state, there exists a maximal part compatible with that possible state.

See `Logos.Core.Frame.CoreFrame` for the Lean implementation.

4.4 World-History

Definition 4.8 (World-History)**.** A *world-history* over a core frame **F** is a function $\tau : X \to W$ where:

- $X \subseteq D$ is a convex subset of the temporal order
- $\tau(x) \Rightarrow_{y-x} \tau(y)$ for all times $x, y \in X$ where $x \leq y$

Notation 4.9. The set of all world-histories over **F** is denoted $H_{\mathbf{F}}$.

Remark 4.10. World-histories assign world-states to times in a way that respects the task relation. The constraint ensures that consecutive world-states are connected by appropriate tasks. The set of maximal possible evolutions equals the set of world-histories (proven in Brast-McKie, "Counterfactual Worlds").

4.5 Core Model

Definition 4.11 (Core Model)**.** A *core model* is a structure $\mathbf{M} = \langle S, \sqsubseteq, D, \Rightarrow, I \rangle$ where:

- $\langle S, \sqsubseteq, D, \Rightarrow \rangle$ is a core frame
- I is an interpretation as in the Constitutive Foundation

See `Logos.Core.Model.CoreModel` for the Lean implementation.

4.6 Truth Conditions

Truth is evaluated relative to a model $\mathbf{M}$, world-history τ, time $x \in D$, variable assignment σ, and temporal index $\vec{\imath} = \langle i_1, i_2, \ldots \rangle$:

4.6.1 Atomic Sentences

Definition 4.12 (Atomic Truth)**.**

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models F(t_1, \ldots, t_n) \iff \text{there is some } f \in v_F \text{ where } f([\![t_1]\!]^{\sigma}_{\mathbf{M}}, \ldots, [\![t_n]\!]^{\sigma}_{\mathbf{M}}) \sqsubseteq \tau(x) \tag{32}$$

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \not\models F(t_1, \ldots, t_n) \iff \text{there is some } f \in f_F \text{ where } f([\![t_1]\!]^{\sigma}_{\mathbf{M}}, \ldots, [\![t_n]\!]^{\sigma}_{\mathbf{M}}) \sqsubseteq \tau(x) \tag{33}$$

Remark 4.13. It is derivable that $\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models A$ iff it is not the case that $\mathbf{M}, \tau, x, \sigma, \vec{\imath} \not\models A$. This justifies using $\models$ alone for truth and $\not\models$ for falsehood.

4.6.2 Lambda Abstraction and Quantification

Definition 4.14 (Lambda and Quantification)**.**

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models (\lambda y.A)(t) \iff \mathbf{M}, \tau, x, \sigma[[\![t]\!]^{\sigma}_{\mathbf{M}}/y], \vec{\imath} \models A \tag{34}$$

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models \forall y.A \iff \mathbf{M}, \tau, x, \sigma[s/y], \vec{\imath} \models A \text{ for all } s \in S \tag{35}$$

4.6.3 Actuality Predicate

Definition 4.15 (Actuality)**.**

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models \mathsf{Act}(t) \iff [\![t]\!]^{\sigma}_{\mathbf{M}} \sqsubseteq \tau(x)$$

Remark 4.16 (Restricted Quantification). The actuality predicate enables restricting quantification to actually existing objects:

$\forall y(\mathsf{Act}(y) \rightarrow A)$ is true iff A holds for all states that are parts of $\tau(x)$

The unrestricted universal quantifier validates the Barcan formulas ($\forall x \Box A \rightarrow \Box \forall x A$ and its converse), while the actuality-restricted quantifier does not.

4.6.4 Extensional Connectives

Definition 4.17 (Extensional Connectives)**.**

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models \neg A \iff \mathbf{M}, \tau, x, \sigma, \vec{\imath} \not\models A \tag{36}$$

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models A \wedge B \iff \mathbf{M}, \tau, x, \sigma, \vec{\imath} \models A \text{ and } \mathbf{M}, \tau, x, \sigma, \vec{\imath} \models B \tag{37}$$

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models A \vee B \iff \mathbf{M}, \tau, x, \sigma, \vec{\imath} \models A \text{ or } \mathbf{M}, \tau, x, \sigma, \vec{\imath} \models B \tag{38}$$

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models A \rightarrow B \iff \mathbf{M}, \tau, x, \sigma, \vec{\imath} \not\models A \text{ or } \mathbf{M}, \tau, x, \sigma, \vec{\imath} \models B \tag{39}$$

4.6.5 Modal Operators

Definition 4.18 (Modal Operators)**.**

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models \Box A \iff \mathbf{M}, \alpha, x, \sigma, \vec{\imath} \models A \text{ for all } \alpha \in H_{\mathbf{F}} \tag{40}$$

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models \Diamond A \iff \mathbf{M}, \alpha, x, \sigma, \vec{\imath} \models A \text{ for some } \alpha \in H_{\mathbf{F}} \tag{41}$$

Remark 4.19. $\Diamond A \equiv \neg\Box\neg A$. Metaphysical necessity can also be defined via counterfactuals: $\Box A := \top \,\Box\!\!\rightarrow A$. This yields an S5 modal logic.

4.6.6 Core Tense Operators

Definition 4.20 (Core Tense Operators)**.**

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models HA \iff \mathbf{M}, \tau, y, \sigma, \vec{\imath} \models A \text{ for all } y \in D \text{ where } y < x \tag{42}$$

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models GA \iff \mathbf{M}, \tau, y, \sigma, \vec{\imath} \models A \text{ for all } y \in D \text{ where } y > x \tag{43}$$

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models PA \iff \mathbf{M}, \tau, y, \sigma, \vec{\imath} \models A \text{ for some } y \in D \text{ where } y < x \tag{44}$$

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models FA \iff \mathbf{M}, \tau, y, \sigma, \vec{\imath} \models A \text{ for some } y \in D \text{ where } y > x \tag{45}$$

Remark 4.21 (Equivalences). $PA \equiv \neg H \neg A$ and $FA \equiv \neg G \neg A$.

4.6.7 Extended Tense Operators

Definition 4.22 (Since and Until)**.**

$$\begin{aligned}
\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models A \lhd B \iff & \text{ there exists } z < x \text{ where } \mathbf{M}, \tau, z, \sigma, \vec{\imath} \models B \\
& \text{and } \mathbf{M}, \tau, y, \sigma, \vec{\imath} \models A \text{ for all } y \text{ where } z < y < x \quad (46) \\
\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models A \rhd B \iff & \text{ there exists } z > x \text{ where } \mathbf{M}, \tau, z, \sigma, \vec{\imath} \models B \\
& \text{and } \mathbf{M}, \tau, y, \sigma, \vec{\imath} \models A \text{ for all } y \text{ where } x < y < z \quad (47)
\end{aligned}$$

Remark 4.23. "A since B" means B was true at some past time, and A has been true ever since. "A until B" means B will be true at some future time, and A is true until then.

4.6.8 Counterfactual Conditional

Definition 4.24 (Counterfactual Conditional)**.**

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models \varphi \Box\!\!\rightarrow C \iff \text{for all } t \in v_\varphi \text{ and } \beta \in H_{\mathbf{F}} :$$

if $s \cdot t \sqsubseteq \beta(x)$ for some maximal t-compatible part $s \in \tau(x)_t$, then $\mathbf{M}, \beta, x, \sigma, \vec{\imath} \models C$

Definition 4.25 (Imposition)**.** We write $t \to_w w'$ ("imposing t on w yields w'") iff there exists a maximal t-compatible part $s \in w_t$ where $s \cdot t \sqsubseteq w'$.

Remark 4.26 (Intuitive Reading). A counterfactual "if φ were the case, then C" is true at world τ and time x iff the consequent C is true in any world β at x where $\beta(x)$ is the result of minimally changing $\tau(x)$ to make the antecedent φ true.

4.6.9 Store and Recall Operators

Definition 4.27 (Store and Recall)**.** For cross-temporal reference within counterfactual evaluation, the context includes a temporal index $\vec{\imath} = \langle i_1, i_2, \ldots \rangle$ of stored times:

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models \uparrow^k A \iff \mathbf{M}, \tau, x, \sigma, \vec{\imath}[x/i_k] \models A \quad (48)$$

$$\mathbf{M}, \tau, x, \sigma, \vec{\imath} \models \downarrow^k A \iff \mathbf{M}, \tau, i_k, \sigma, \vec{\imath} \models A \quad (49)$$

where $\vec{\imath}[x/i_k]$ replaces i_k with current time x.

Example 4.28 (Tensed Counterfactuals)**.**
- $\downarrow^1 (B \Box\!\!\rightarrow FH)$ — "If B had occurred at the stored time, then H would have occurred at some future time"

- $\uparrow^2\downarrow^1 (B\Box \rightarrow\downarrow^2 H)$ — "Store now, recall past time, if B then H at the stored present"

See `Logos.Core.Semantics.Satisfies` for the Lean implementation.

4.7 Bimodal Interaction Principles

The task semantics validates these perpetuity principles connecting modal and temporal operators:

Definition 4.29 (Perpetuity Principles)**.**

$$\mathbf{P1}: \quad \Box\varphi \rightarrow \triangle\varphi \qquad \text{Necessary implies always} \tag{50}$$

$$\mathbf{P2}: \quad \triangledown\varphi \rightarrow \Diamond\varphi \qquad \text{Sometimes implies possible} \tag{51}$$

$$\mathbf{P3}: \quad \Box\triangle\varphi \leftrightarrow \triangle\Box\varphi \qquad \text{Necessary-always commutativity} \tag{52}$$

$$\mathbf{P4}: \quad \Diamond\triangledown\varphi \leftrightarrow \triangledown\Diamond\varphi \qquad \text{Possible-sometimes commutativity} \tag{53}$$

$$\mathbf{P5}: \quad \Box\varphi \rightarrow \Box\triangle\varphi \qquad \text{Necessary implies necessary-always} \tag{54}$$

$$\mathbf{P6}: \quad \triangledown\Diamond\varphi \rightarrow \Diamond\varphi \qquad \text{Sometimes-possible implies possible} \tag{55}$$

4.8 Temporal Frame Constraints

Different temporal structures yield different valid principles. The framework does not assume discrete time:

Constraint	Description	Corresponding Axiom
Dense	Between any two times there is another	$GG\varphi \rightarrow G\varphi$
Complete	Every bounded set has a least upper bound	$\triangle(H\varphi \rightarrow FH\varphi) \rightarrow (H\varphi \rightarrow G\varphi)$
Unbounded Past	No earliest time	$P\top$
Unbounded Future	No latest time	$F\top$

Table 2: Temporal frame constraints and corresponding axioms

4.9 Logical Consequence

Definition 4.30 (Logical Consequence (Core Extension))**.**

$$\Gamma \models A \iff \text{for any } \mathbf{M}, \tau \in H_{\mathbf{F}}, x \in D, \sigma, \vec{\imath} : \text{if } \mathbf{M}, \tau, x, \sigma, \vec{\imath} \models B \text{ for all } B \in \Gamma, \text{ then } \mathbf{M}, \tau, x, \sigma, \vec{\imath} \models A$$

Remark 4.31. Unlike the Constitutive Foundation, the Core Extension evaluates truth relative to world-histories and times, not just verification by states.

The hyperintensional distinctions are preserved—propositions still differ if their exact verifiers differ—but the Core Extension adds the intensional layer needed to evaluate sentences with modal and temporal operators.

5 Counterfactual Logic Axiom System

This section presents the axiom system for the counterfactual logic of the Core Extension.

5.1 Core Rules

Definition 5.1 (Closure Under Deduction)**.**

$$\textbf{R1}: \quad \text{If } \Gamma \vdash C, \text{ then } \varphi \Box\!\!\rightarrow \Gamma \vdash \varphi \Box\!\!\rightarrow C$$

Remark 5.2. This rule states that if C is derivable from Γ, then for any antecedent φ, the counterfactual with consequent C is derivable from the counterfactuals with consequents from Γ.

5.2 Counterfactual Axiom Schemata

Definition 5.3 (Counterfactual Axioms)**.**

$$\textbf{C1}: \quad \varphi \Box\!\!\rightarrow \varphi \qquad \text{(Identity)} \tag{56}$$

$$\textbf{C2}: \quad \varphi, \varphi \Box\!\!\rightarrow A \vdash A \qquad \text{(Counterfactual Modus Ponens)} \tag{57}$$

$$\textbf{C3}: \quad \varphi \Box\!\!\rightarrow \psi, \varphi \wedge \psi \Box\!\!\rightarrow A \vdash \varphi \Box\!\!\rightarrow A \qquad \text{(Weakened Transitivity)} \tag{58}$$

$$\textbf{C4}: \quad \varphi \vee \psi \Box\!\!\rightarrow A \vdash \varphi \wedge \psi \Box\!\!\rightarrow A \qquad \text{(Disjunction-Conjunction)} \tag{59}$$

$$\textbf{C5}: \quad \varphi \vee \psi \Box\!\!\rightarrow A \vdash \varphi \Box\!\!\rightarrow A \qquad \text{(Simplification Left)} \tag{60}$$

$$\textbf{C6}: \quad \varphi \vee \psi \Box\!\!\rightarrow A \vdash \psi \Box\!\!\rightarrow A \qquad \text{(Simplification Right)} \tag{61}$$

$$\textbf{C7}: \quad \varphi \Box\!\!\rightarrow A, \psi \Box\!\!\rightarrow A, \varphi \wedge \psi \Box\!\!\rightarrow A \vdash \varphi \vee \psi \Box\!\!\rightarrow A \qquad \text{(Disjunction Introduction)} \tag{62}$$

Remark 5.4 (Explanation of Counterfactual Axioms).

- **C1** (Identity): Every proposition counterfactually implies itself.

- **C2** (Modus Ponens): If φ is actually true and φ would imply A, then A is true.
- **C3** (Weakened Transitivity): This weaker form of transitivity avoids the problematic full transitivity principle.
- **C4-C7** govern the interaction between counterfactuals and Boolean operations on antecedents.

5.3 Modal Axiom Schemata

Definition 5.5 (Modal Axioms)**.**

$$\begin{aligned} &\textbf{M1}: \quad \top && \text{(Truth)} && (63)\\ &\textbf{M2}: \quad \neg\bot && \text{(Non-Contradiction)} && (64)\\ &\textbf{M3}: \quad A \to \Box\Diamond A && \text{(Brouwer)} && (65)\\ &\textbf{M4}: \quad \Box A \to \Box\Box A && \text{(S4 Transitivity)} && (66)\\ &\textbf{M5}: \quad \Box(\varphi \to A) \vdash \varphi \Box\!\!\to A && \text{(Strict-to-Counterfactual)} && (67) \end{aligned}$$

Remark 5.6 (S5 Modal Logic). The combination of M3 and M4 yields an S5 modal logic, which is characterized by an equivalence relation on possible worlds. In the task semantics, this arises from the structure of world-histories: any world-history is accessible from any other at any given time.

Remark 5.7 (Strict-to-Counterfactual). Axiom M5 states that if A is a strict consequence of φ (i.e., necessarily, if φ then A), then φ counterfactually implies A. This connects the modal operator with the counterfactual conditional.

5.4 Derived Theorems

Theorem 5.8 (Counterfactual Strengthening)**.** *If $\varphi \Box\!\!\to A$ and $\varphi \Box\!\!\to \psi$, then $\varphi \Box\!\!\to (A \wedge \psi)$.*

Theorem 5.9 (Counterfactual Weakening)**.** *If $\varphi \Box\!\!\to A$ and $A \vdash B$, then $\varphi \Box\!\!\to B$.*

Theorem 5.10 (Necessity Introduction)**.** *$\top \Box\!\!\to A$ is equivalent to $\Box A$.*

Theorem 5.11 (Excluded Middle)**.** $(\varphi \Box\!\!\to A) \vee (\varphi \Box\!\!\to \neg A)$ *is* not *generally valid.*

Remark 5.12. The failure of excluded middle for counterfactuals (Theorem **??**) reflects the mereological structure: when imposing an antecedent φ on the actual world, there may be multiple ways to do so, leading to different worlds where A or $\neg A$ holds. The counterfactual is true only when the consequent holds in *all* such imposed worlds.

5.5 Soundness and Completeness

Theorem 5.13 (Soundness)**.** *If* $\Gamma \vdash A$ *using the axiom system above, then* $\Gamma \models A$*.*

> [OPEN QUESTION]: Completeness of the axiom system with respect to the mereological counterfactual semantics remains an open problem. The main challenge is showing that every consistent set of formulas has a model with the required mereological structure.

See `Logos.Core.Axioms.CounterfactualAxioms` for the Lean implementation.

6 Epistemic Extension

[DETAILS PENDING DEVELOPMENT]

The Epistemic Extension extends the Core Extension with structures for belief, knowledge, and probability.

6.1 Frame Extension

[DETAILS PENDING DEVELOPMENT]

The epistemic frame extends the core frame with a credence function assigning probabilities to state transitions.

> [OPEN QUESTION]: What is the exact structure of the credence function? Does it assign probabilities to individual state transitions or to sets of transitions?

6.2 Operators

[FULL SEMANTIC CLAUSES FOR EPISTEMIC OPERATORS PENDING SPECIFICATION]

Operator	Intended Reading
$B_a \varphi$	Agent a believes that φ
$K_a \varphi$	Agent a knows that φ
$\Pr(\varphi) \geq r$	The probability of φ is at least r

Table 3: Epistemic operators

6.3 Indicative Conditionals

[DETAILS PENDING DEVELOPMENT]

> [OPEN QUESTION]: How do indicative conditionals relate to counterfactual conditionals in the semantic framework?

7 Normative Extension

[DETAILS PENDING DEVELOPMENT]

The Normative Extension extends the Core Extension with structures for obligation, permission, and value.

7.1 Frame Extension

[DETAILS PENDING DEVELOPMENT]

The normative frame extends the core frame with value orderings over states.

> [OPEN QUESTION]: How are value orderings structured? Are they complete orderings or partial orderings? Are they agent-relative?

7.2 Operators

Operator	Intended Reading
$O\varphi$	It ought to be that φ
$P\varphi$	It is permitted that φ
$\varphi \succ \psi$	φ is preferred to ψ

Table 4: Normative operators

[FULL SEMANTIC CLAUSES FOR NORMATIVE OPERATORS PENDING SPECIFICATION]

7.3 Normative Explanation

[DETAILS PENDING DEVELOPMENT]

> [OPEN QUESTION]: How does normative explanation relate to causal explanation?

8 Spatial Extension

[DETAILS PENDING DEVELOPMENT]

The Spatial Extension extends the Core Extension with structures for spatial reasoning and location.

8.1 Frame Extension

[DETAILS PENDING DEVELOPMENT]

The spatial frame extends the core frame with:

- **Location space** L = set of spatial locations
- **Spatial relations**: adjacency, containment, distance

> [OPEN QUESTION]: What spatial primitives are required? Should locations be mereological (with parts) or set-theoretic?

8.2 Operators

Operator	Intended Reading
Here(A)	A holds at the current location
Somewhere(A)	A holds at some location
Everywhere(A)	A holds at all locations
Near(A)	A holds at an adjacent location

Table 5: Spatial operators

[FULL SEMANTIC CLAUSES FOR SPATIAL OPERATORS PENDING SPECIFICATION]

9 Agential Extension

[DETAILS PENDING DEVELOPMENT]

The Agential Extension requires at least one of the Epistemic, Normative, or Spatial Extensions to be loaded. It provides structures for multi-agent reasoning.

Remark 9.1 (Dependency). This extension depends on at least one middle extension (Epistemic, Normative, or Spatial) being loaded. See the extension dependency diagram in **??**.

9.1 Frame Extension

[DETAILS PENDING DEVELOPMENT]

> [OPEN QUESTION]: What frame extensions are required for multi-agent reasoning? Does this extension add agent indices, or agent-relative accessibility relations?

9.2 Multi-Agent Operators

[DETAILS PENDING DEVELOPMENT]

> [OPEN QUESTION]: How do individual and collective agency interact in the semantic framework?