

HyperLogic® For Teaching:

- **Context;**
- **Videos:**
 - 1 - Orienting Demo (Instructor);**
 - 2 - Orienting Demo (Student);**
 - 3 - Proof Construction in Cog. Calculi (Pro);**
- **On the Road to 3D HyperLogic**

TEAL 2026@FLoC 2026
ISCTE
Lisbon PT
July 25 2026

Alexander Bringsjord • Naveen Sundar Govindarajulu • Selmer Bringsjord



Context ...



HYPERLOGIC

Interactive AI-Infused Proof, Program & Argument Construction



HYPERLOGIC

Interactive AI-Infused Proof, Program & Argument Construction

Available-to-Student Workspace Types:

Propositional Calculus

D

S4

S5

L_0 = Pure Predicate Calculus

L_1 = First-order Logic

L_2 = Second-order Logic

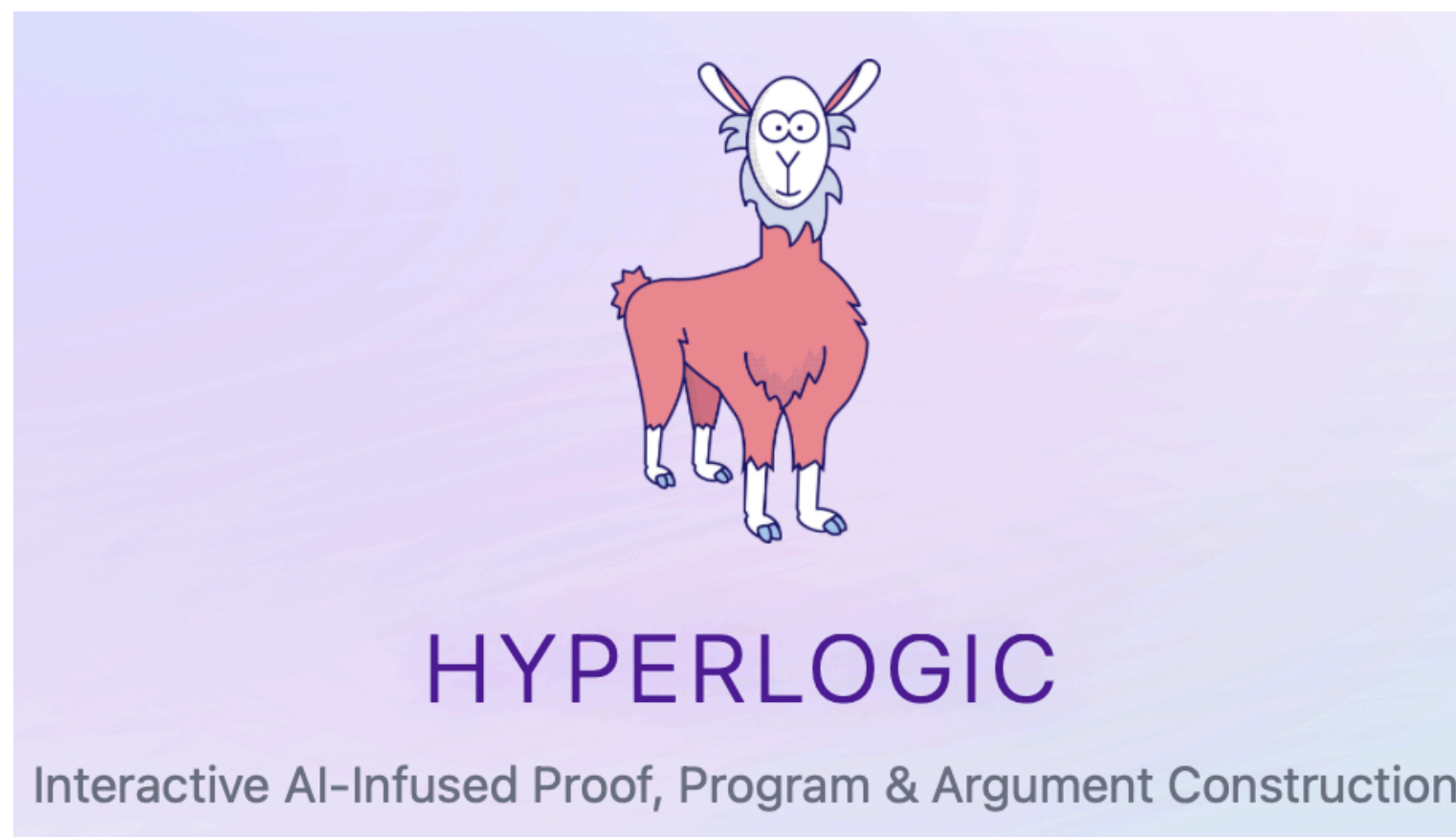
K

T

DCEC (fragment)

Hyperlog

+ New Folder



Available-to-Student Workspace Types:

Propositional Calculus

D

S4

S5

L_0 = Pure Predicate Calculus

L_1 = First-order Logic

L_2 = Second-order Logic

K

T

DCEC (fragment)

Hyperlog

+ New Folder

But HyperLogic can cover the entire Universe!



Professional

Student

Instructor

Developer

HyperSlate®

HyperSlate®

HyperGrader®

Vivid

CAC

DC $\mathcal{E}$ C

$\mathcal{L}_{PC}$

$\mathcal{L}_1$

$\mathcal{L}_2$

...

Hyperlog®

new logic of animations

Videos ...

Instructor POV (Curry Paradox)

<http://www.logicamodernapproach.com/demos/InstructorOverviewCurry2.mov>



HYPERLOGIC

Interactive AI-Infused Proof, Program & Argument Construction

Sign In

Register

Instructor POV (Curry Paradox)

<http://www.logicamodernapproach.com/demos/InstructorOverviewCurry2.mov>



HYPERLOGIC

Interactive AI-Infused Proof, Program & Argument Construction

Sign In

Register

Instructor Assignment Creation & Publishing

(no AI-generation of personalized problems covered)

<http://www.logicmodernapproach.com/demos/DemoOfFastTestCreation.mov>

HyperGrader®

Problem Categories ▾

HyperSlate®

My Progression

Leader Board

TEAL 2026 ▾

☾

Selmer.Bringsjord@gmail.com (longsnowflake876) ▾

 Problem Bank

 Edit Problems

all

 Metrics for all

Download:
IFLWAI_HLHGHS_042626.pdf

Problems

☐ TrueQuatroNoComp

☒ ACurryParadox

☒ SwitchingDisjunctsFineIFLA
...

☒ TertiumNonDaturIFLAI1S24

☐ CheatersNeverProsperS25

 New

 Leaderboard for all

☐  TrueQuatroNoComp

The theorem to be proved here is this:

Take any property Z you like; if whenever any object has every property Y it also has Z , every object has Z .

Displayed more fastidiously with help from LaTeX:

$$\forall Z [\forall Y \forall x (Y(x) \rightarrow Z(x)) \rightarrow \forall x Z(x)]$$

So far, those who have discovered a solving proof have had to make use of the inference schema/rule comprehension. If unfamiliar with it, it probably makes sense to review the explanation of it in your textbook from Bringsjord et al.

You are free to make use either or both of the (standard) provability oracles for the propositional calculus and first-order logic.

Deadline May 6, 2026 at 11:59 PM EDT

Solve

 Submissions

 Metrics

☒  ACurryParadox

Instructor Assignment Creation & Publishing

(no AI-generation of personalized problems covered)

<http://www.logicmodernapproach.com/demos/DemoOfFastTestCreation.mov>

HyperGrader®

Problem Categories ▾

HyperSlate®

My Progression

Leader Board

TEAL 2026 ▾

☾

Selmer.Bringsjord@gmail.com (longsnowflake876) ▾

 Problem Bank

 Edit Problems

all

 Metrics for all

Download:
IFLWAI_HLHGHS_042626.pdf

Problems

☐ TrueQuatroNoComp

☒ ACurryParadox

☒ SwitchingDisjunctsFineIFLA
...

☒ TertiumNonDaturIFLAI1S24

☐ CheatersNeverProsperS25

 New

 Leaderboard for all

☐  TrueQuatroNoComp

The theorem to be proved here is this:

Take any property Z you like; if whenever any object has every property Y it also has Z , every object has Z .

Displayed more fastidiously with help from LaTeX:

$$\forall Z [\forall Y \forall x (Y(x) \rightarrow Z(x)) \rightarrow \forall x Z(x)]$$

So far, those who have discovered a solving proof have had to make use of the inference schema/rule comprehension. If unfamiliar with it, it probably makes sense to review the explanation of it in your textbook from Bringsjord et al.

You are free to make use either or both of the (standard) provability oracles for the propositional calculus and first-order logic.

Deadline May 6, 2026 at 11:59 PM EDT

Solve

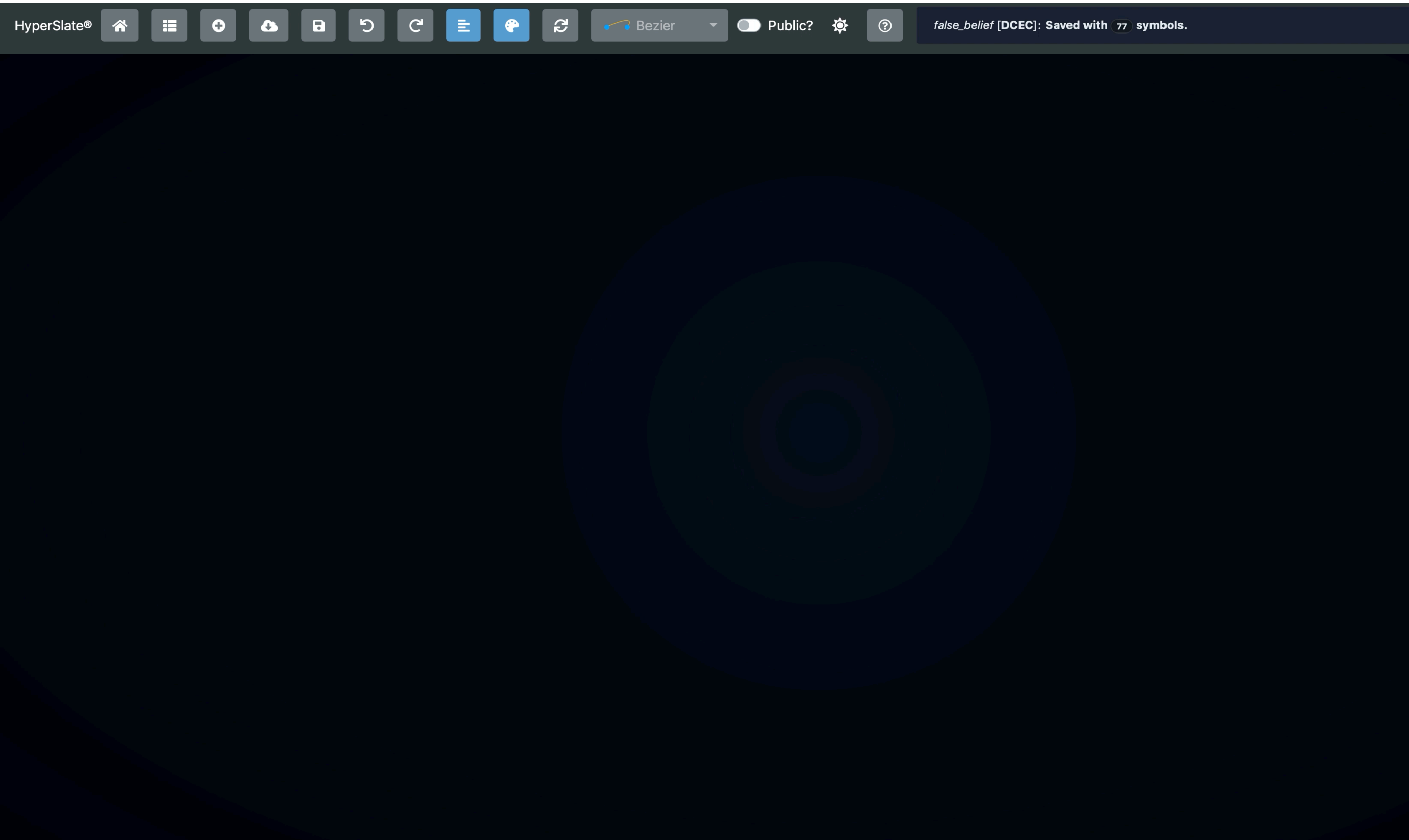
 Submissions

 Metrics

☒  ACurryParadox

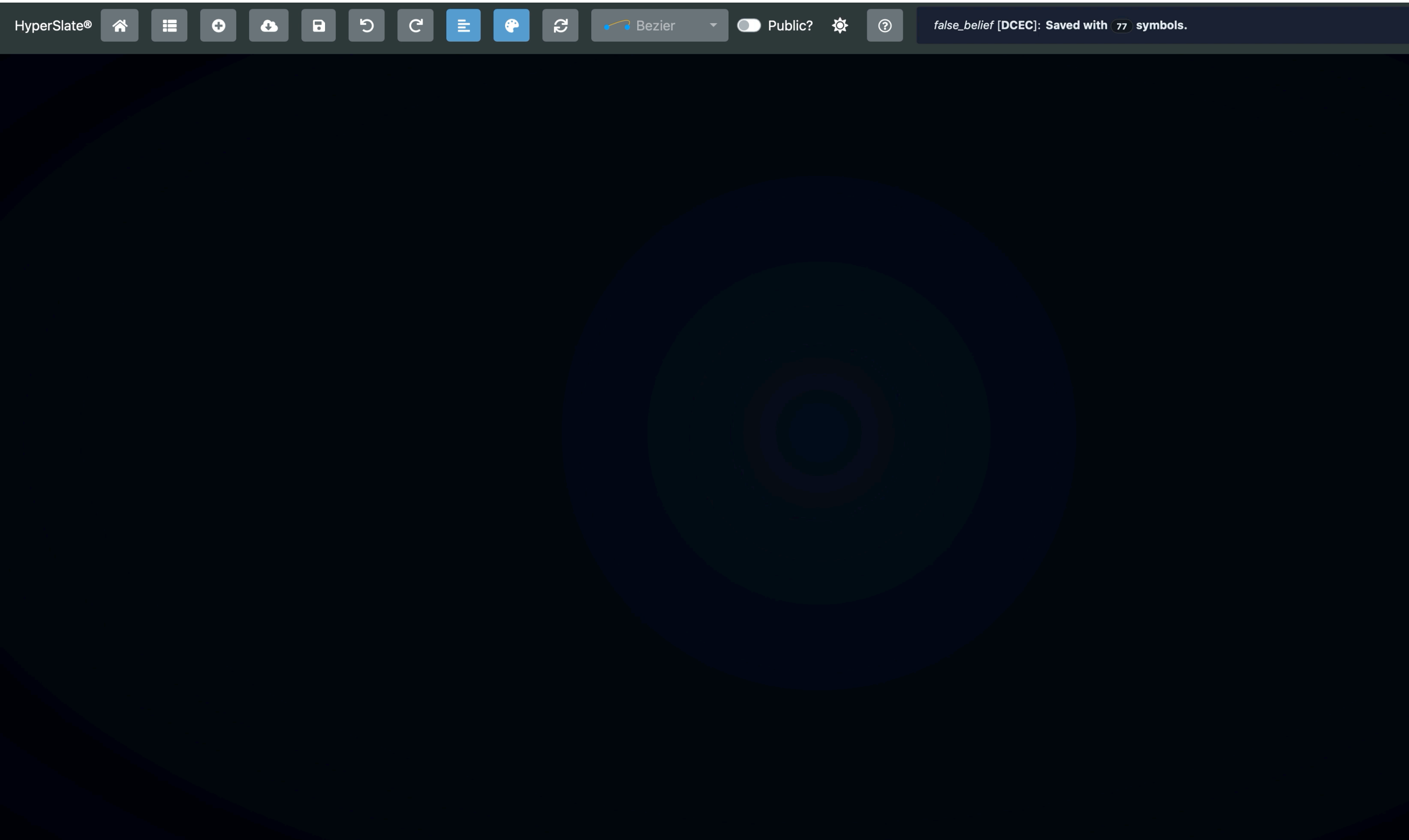
Professional POV (Proof in $\mathcal{DCEC}^*$)

http://www.logicamodernapproach.com/demos/dcec_hyperslate_demo.mov



Professional POV (Proof in $\mathcal{DCEC}^*$)

http://www.logicamodernapproach.com/demos/dcec_hyperslate_demo.mov



Public URL Feature Available for Instructors/Admins

<https://world.logicamodernapproach.com/hyperslate/public/Selmer.Bringsjord@gmail.com/HypotheticalSyllogism>

Background Technical Views

...

Figure 1: Anatomy, Symbolically Described, of Formula/Function Type of Node Used in The Hypergraphical Proofs, Arguments, and Pure General Logic Programs

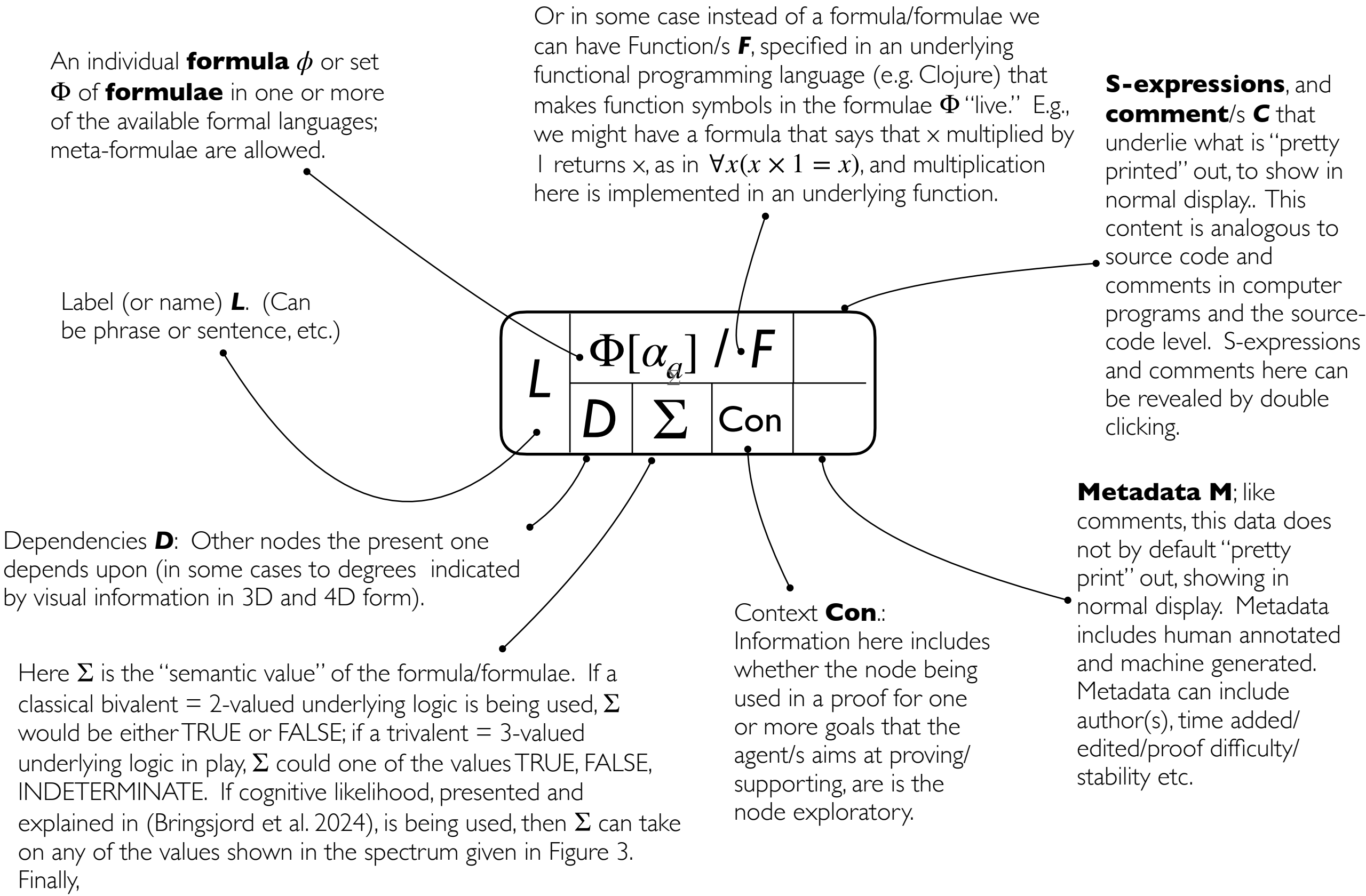


Figure 1: Anatomy, Symbolically Described, of Formula/Function Type of Node Used in The Hypergraphical Proofs, Arguments, and Pure General Logic Programs

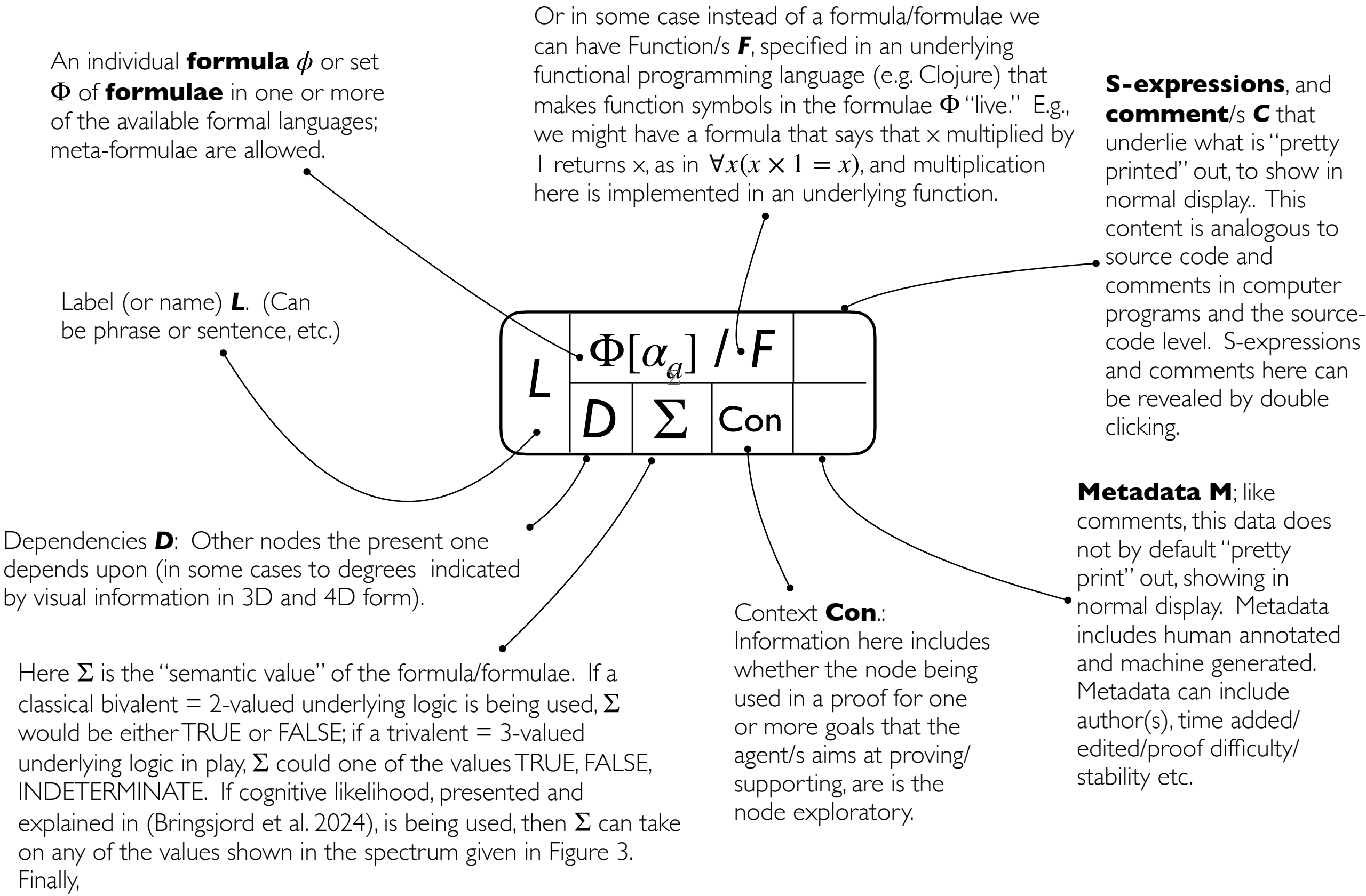


Figure 2: Simple Instantiation of Formula Node Anatomy in Figure 1, in HyperSlate System

Label **L**. Here is GOAL.

A simple example of an actual 2D function node:

The formula in the propositional calculus is shown here.

This particular formula is a theorem that has been proved by one of the AI oracles able to find proofs on its own. Finding such a proof took 11 milliseconds.

Here, the underlying S-content has been revealed by double-clicking:

GOAL $((p \Rightarrow q) \wedge (\neg r \Rightarrow \neg q)) \Rightarrow (p \Rightarrow r)$
from {}

Node GOAL. Computed in 11 (ms), size 107

PC ⊢ (Oracle)

GOAL $((p \Rightarrow q) \wedge (\neg r \Rightarrow \neg q)) \Rightarrow (p \Rightarrow r)$
from {}

Node GOAL. Computed in 11 (ms), size 107

Edit Node

```
1 (if (and (if p q)
2   (if (not r) (not q)))
3   (if p r))
4
```

$((p \Rightarrow q) \wedge (\neg r \Rightarrow \neg q)) \Rightarrow (p \Rightarrow r)$

Justification

PC ⊢ (Oracle)

Name

GOAL

Cancel

Save

Figure 3

Underlying Specification of Functions/Formulae

The screenshot displays the HyperSlate interface for formal logic. The background shows a proof diagram with nodes: **GIVEN1** (a universal quantifier formula), **GIVEN2** (a universal quantifier formula), **FOL ⊢ (Oracle)**, and **GOAL** (a complex formula). A green box labeled "assume" is connected to the GIVEN1 node. A blue box at the top right indicates "[FIRST-ORDER-LOGIC]: Saved with 82 symbols."

The foreground features an "Edit Node" dialog box. It contains a text editor with the following code:

```
1 (forall (x)
2   (implies (Zebra x)
3     (exists (y)
4       (and (Loves x y)
5         (forall (z)
6           (implies (Loves x z)
7             (= z y)))))))
```

Below the editor, the formula $\forall x: \text{Zebra}(x) \Rightarrow (\exists y: \text{Loves}(x, y) \wedge (\forall z: \text{Loves}(x, z) \Rightarrow (z = y)))$ is displayed. The "Justification" dropdown is set to "assume", and the "Name" field contains "GIVEN2". "Cancel" and "Save" buttons are at the bottom right.

At the bottom of the interface, a status message reads: "FOL-PROVABILITY not allowed in the final submission. Node GOAL. Computed in 2335 (ms), size 436".

Figure 4: Simple Instantiation of Function Node Anatomy in Figure 1, in HyperSlate System

The label..

A simple example of an actual function node:

The function here, a binary arithmetic function called **myfunc**, is a simple polynomial taking in a number *x* and a number *y*.

A SIMPLE POLYNOMIAL FUNCTION

myfunc(x,y) ↦ ((3 * x) - (2 * y))

from {A SIMPLE POLYNOMIAL FUNCTION}

Here, the underlying function, defined as a function in the Clojure programming language, has been revealed by double-clicking:

HyperSlate®

assume

A SIMPLE POLYNOMIAL FUNCTION

myfunc(x,y) ↦ ((3 * x) - (2 * y))

from {A SIMPLE POLYNOMIAL FUNCTION}

Edit Node

1 (defn myfunc [x y] (- (* 3 x) (* 2 y)))

myfunc(x,y) ↦ ((3 * x) - (2 * y))

Justification

assume

Name

A SIMPLE POLYNOMIAL FUNCTION

Cancel

Save

Figure 5

Cognitive-Likelihood Continuum

This is for the case where Σ is based on cognitive likelihood, presented and explained in (Bringsjord et al. 2024). Σ can take on any of the 13 values shown in the spectrum shown here.,

Positive Cognitive Likelihood

.....

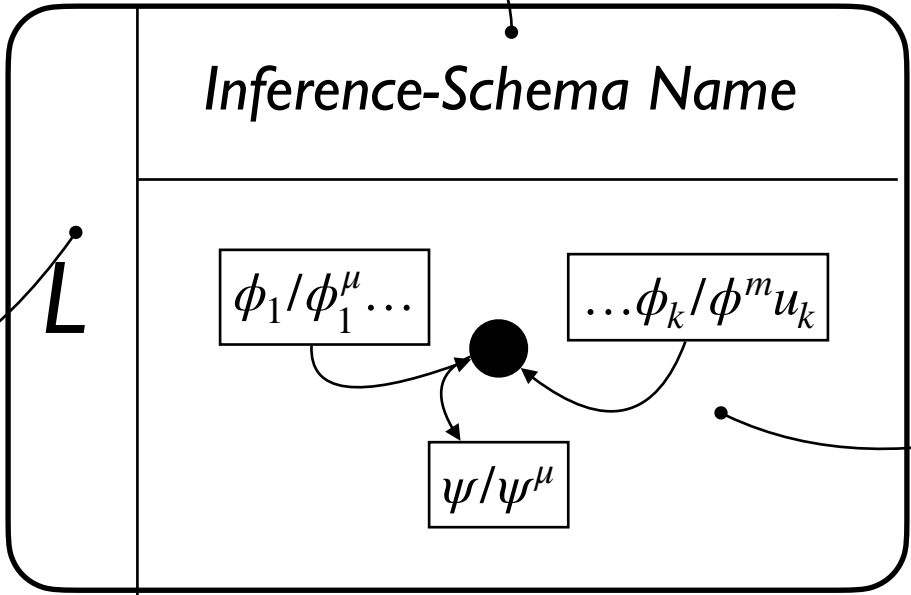
Negative Cognitive Likelihood

- (6) Certain
- (5) Evident
- (4) Overwhelmingly Likely/Beyond Reasonable Doubt
- (3) Highly Likely
- (2) Likely
- (1) More Likely Than Not
- (0) Counterbalanced
- (-1) More Unlikely Than Not
- (-2) Unlikely
- (-3) Highly Unlikely
- (-4) Overwhelmingly Unlikely/Beyond Reasonable Belief
- (-5) Evidently False
- (-6) Certainly False

Figure 6: Anatomy, Symbolically Described, of Inference-Schema Type of Node Used in The Hypergraphs (which encode PGLPs, proofs, arguments)

E.g.. $\wedge$ elimination, which says that from a formula that is a conjunction, one can infer either of the conjuncts.

Label. (Can be phrase or sentence, etc.)



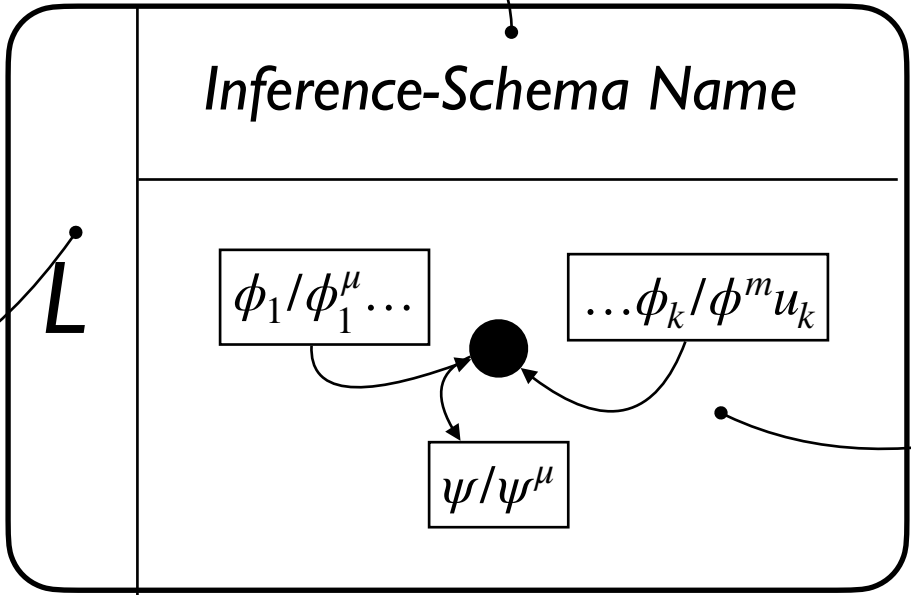
The specification of the schema is expressed as a meta-hypergraph itself. This information makes the entire edifice of a PGLP or an proof or an argument recursive, since the call of an inference schema to sanction an inference holds within it the specification of the schema itself. For users who have inference schemata memorized, which is standard, the specification is hidden, and can be revealed by double-clicking.

The situation here is generalized, and abstract.

Figure 6: Anatomy, Symbolically Described, of Inference-Schema Type of Node Used in The Hypergraphs (which encode PGLPs, proofs, arguments)

E.g.. $\wedge$ elimination, which says that from a formula that is a conjunction, one can infer either of the conjuncts.

Label. (Can be phrase or sentence, etc.)



The specification of the schema is expressed as a meta-hypergraph itself. This information makes the entire edifice of a PGLP or an proof or an argument recursive, since the call of an inference schema to sanction an inference holds within it the specification of the schema itself. For users who have inference schemata memorized, which is standard, the specification is hidden, and can be revealed by double-clicking.

The situation here is generalized, and abstract.

Figure 7: Example of Two Inference-Schemata Nodes Used in a Hypergraphical Proof Constructed in the HyperSlate System

Two inference schemata are used in this snippet of a proof: The simple one called Assumption (used to put assumptions (or suppositions) in to the workspace), and the inference schema $\wedge$ Elimination, which allows for the inferring of individual conjuncts from conjunctions.

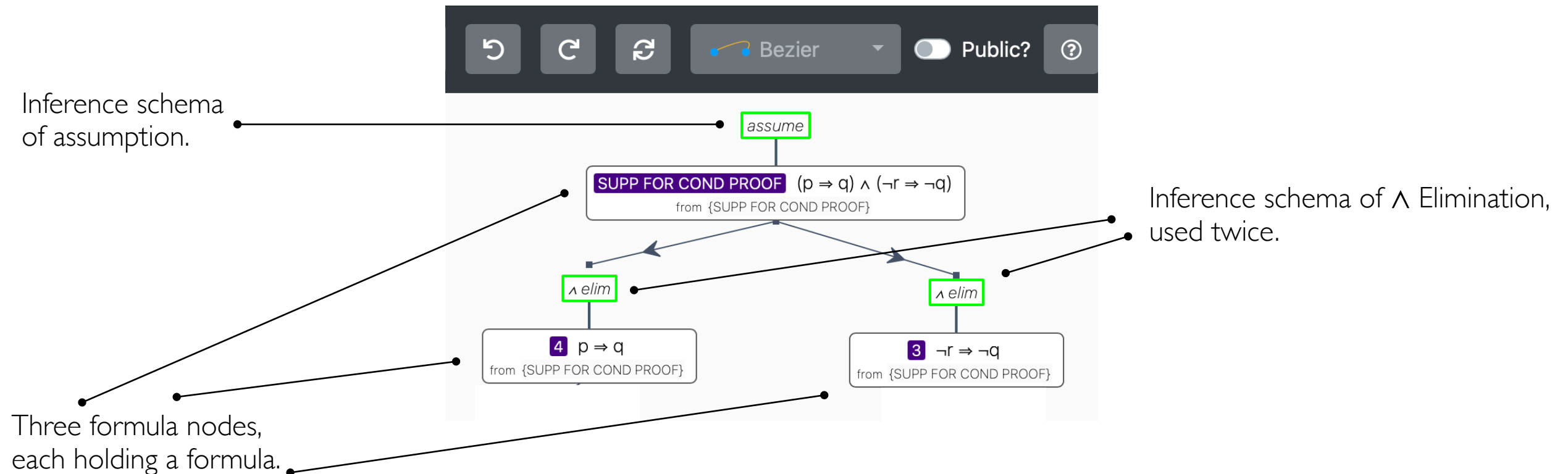


Figure 7: Example of Two Inference-Schemata Nodes Used in a Hypergraphical Proof Constructed in the HyperSlate System

Two inference schemata are used in this snippet of a proof: The simple one called Assumption (used to put assumptions (or suppositions) in to the workspace), and the inference schema $\wedge$ Elimination, which allows for the inferring of individual conjuncts from conjunctions.

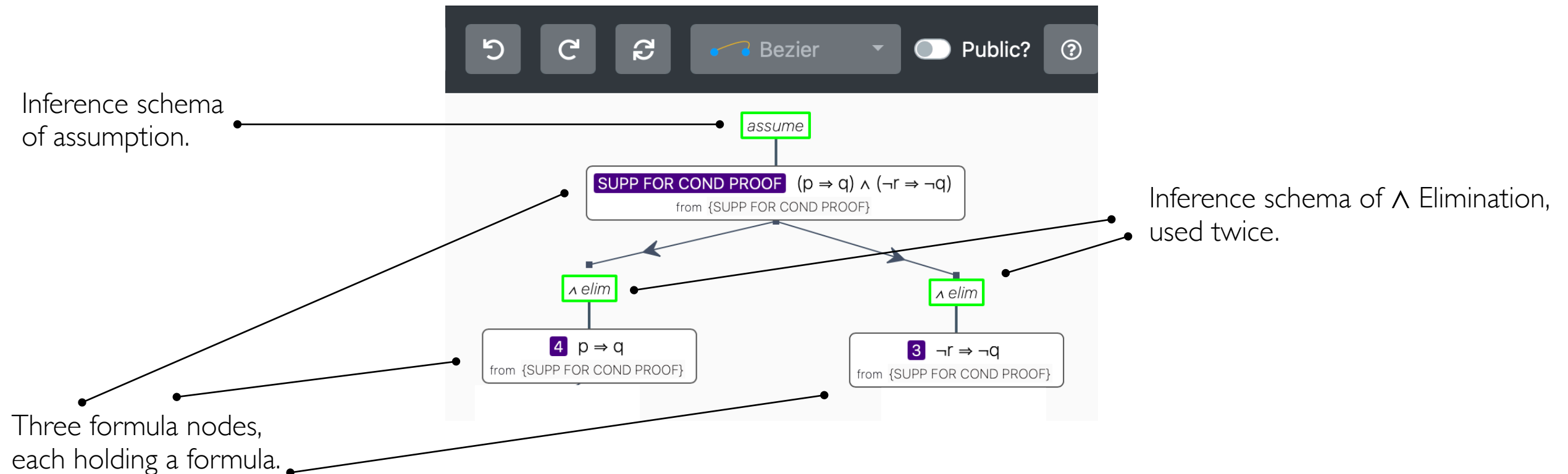


Figure 8: Underlying Anatomy, Expressed Symbolically, of Pure General Logic Programming and Programs

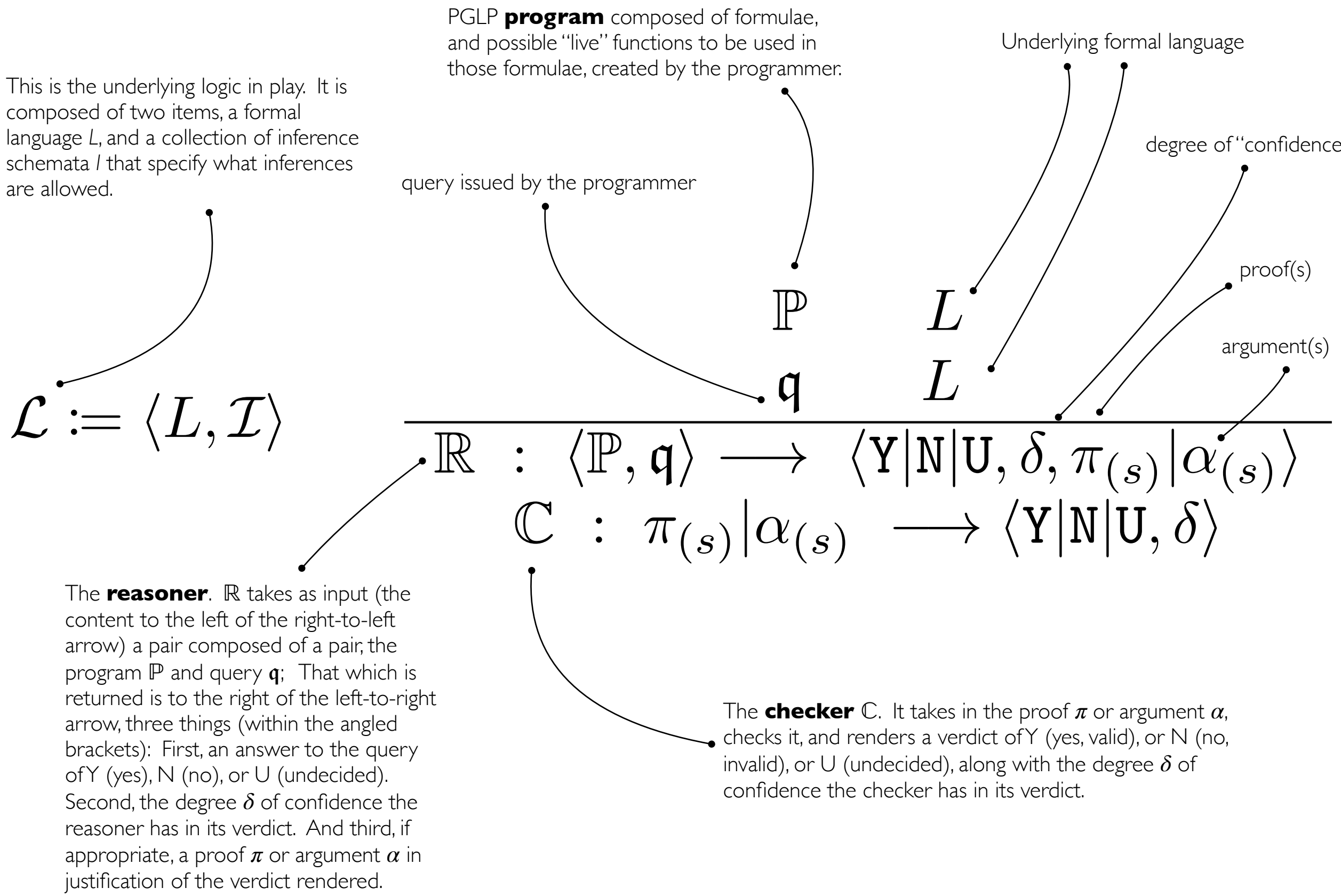
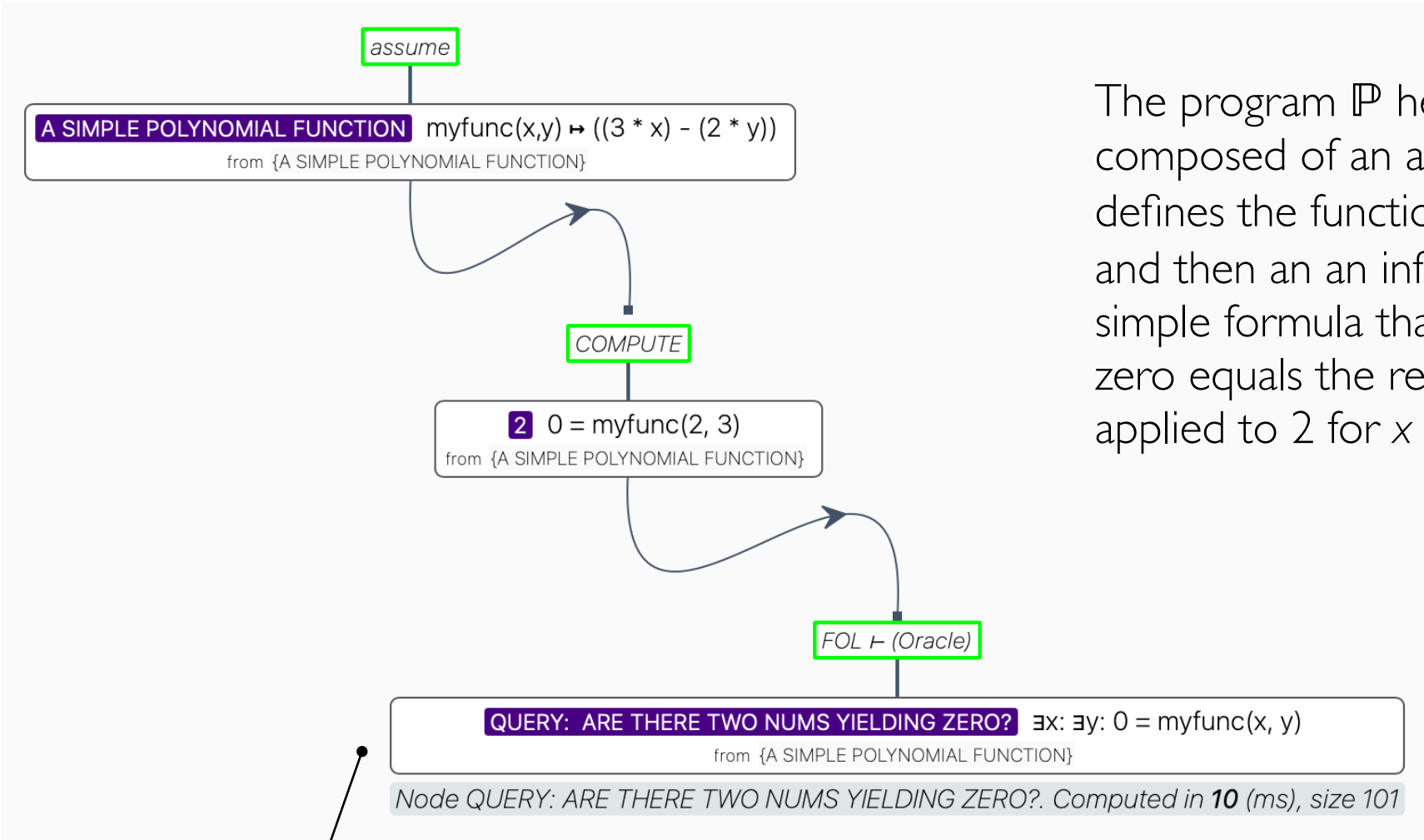


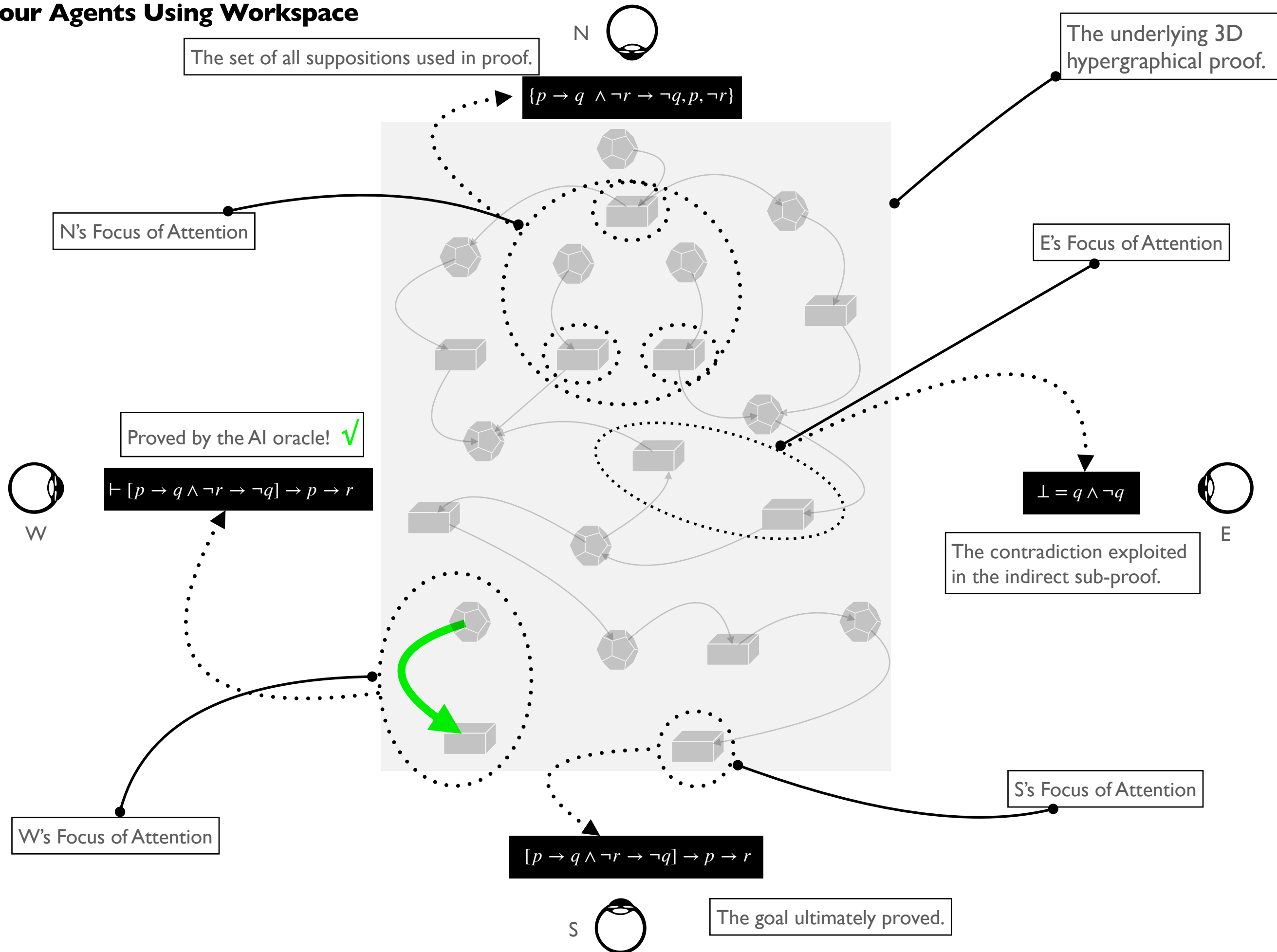
Figure 9: A Simple 2D Pure General Logic Program (PGLP) at the Level of First-Order Logic



The program $\mathbb{P}$ here is composed of an assumption that defines the function **myfunc**, and then an an inference to a simple formula that says that zero equals the result of myfunc applied to 2 for x and 3 for y.

The query **q** issued by the programmer: Does there exist an x and a y such that the function **myfunc** applied to x and y yields zero? The answer given is Y (yes), since the query succeeds, taking the AI only 10 milliseconds to find a proof that certifies the affirmative.

Four Agents Using Workspace





HYPERLOGIC

Interactive AI-Infused Proof, Program & Argument Construction

Evolution ...

Alexander Bringsjord • Naveen Sundar Govindarajulu • Selmer Bringsjord



Łukasiewicz's Challenge 1926

Logicize the “natural” concept of making “arbitrary assumptions” and exploring therefrom!

Łukasiewicz's Challenge 1926

Logicize the “natural” concept of making “arbitrary assumptions” and exploring therefrom!

(This has nothing *per se* to do with deduction, but that's where things started, & many, to this day, are fixated on deductive reasoning.)

Jáskowski 1934

1.	$((P \supset Q) \& (\sim R \supset \sim Q))$
2.	P
3.	$((P \supset Q) \& (\sim R \supset \sim Q))$
4.	$(P \supset Q)$
5.	Q
6.	$(\sim R \supset \sim Q)$
7.	$\sim R$
8.	$(\sim R \supset \sim Q)$
9.	$\sim Q$
10.	Q
11.	R
12.	$P \supset R$
13.	$((P \supset Q) \& (\sim R \supset \sim Q)) \supset (P \supset R)$

Gentzen 1934

Trees: “The following is a tree ...”

$$\begin{array}{c}
 \begin{array}{c}
 \dagger \\
 \hline
 3 \quad \dfrac{((P \supset Q) \& (\sim R \supset \sim Q))}{\sim R \quad (\sim R \supset \sim Q) \quad \&E} \\
 \hline
 \sim Q \quad \supset E
 \end{array}
 \qquad
 \begin{array}{c}
 \dagger \\
 \hline
 \dfrac{((P \supset Q) \& (\sim R \supset \sim Q))}{(P \supset Q) \quad \&E} \quad 2 \\
 \hline
 P
 \end{array}
 \end{array}$$

$$\begin{array}{c}
 \dfrac{\dfrac{\dfrac{\perp}{R} \quad \perp I}{\perp} \quad \perp E(3)}{(P \supset R) \quad \supset I(2)} \\
 \hline
 (((P \supset Q) \& (\sim R \supset \sim Q)) \supset (P \supset R)) \quad \supset I(1)
 \end{array}$$

$$\begin{array}{c}
 \begin{array}{c}
 \dagger \\
 \hline
 3 \quad \dfrac{((p \rightarrow q) \wedge (\neg r \rightarrow \neg q))}{(\neg r \rightarrow \neg q)} \quad \wedge E \\
 \hline
 \neg r \quad \rightarrow E \\
 \hline
 \neg q
 \end{array}
 \qquad
 \begin{array}{c}
 \dagger \\
 \hline
 \dfrac{((p \rightarrow q) \wedge (\neg r \rightarrow \neg q))}{(p \rightarrow q)} \quad \wedge E \\
 \hline
 p \quad \rightarrow E \\
 \hline
 q \quad \perp I
 \end{array}
 \end{array}$$

$$\begin{array}{c}
 \dfrac{\dfrac{\dfrac{\perp}{\neg \neg r} \quad \neg I(3)}{r} \quad \neg \neg E}{(p \rightarrow r) \quad \rightarrow I(2)} \\
 \hline
 (((p \rightarrow q) \wedge (\neg r \rightarrow \neg q)) \rightarrow (p \rightarrow r)) \quad \rightarrow I(1)
 \end{array}$$

Fitch 1952

1.		<u>$(\exists x)(\forall y) Fxy$</u>	hypothesis
2.		<u>$(\exists x)(\forall y) Fxy$</u>	1, Reiteration
3.		w <u>$(\forall y) Fwy$</u>	hypothesis
4.		Fwy	3, uq elim
5.		$(\exists x)Fxy$	4, eq int
6.		y $(\exists x)Fxy$	2, 3-5, eq elim
7.		$(\forall y)(\exists x)Fxy$	2-6, uq int
8.		$((\exists x)(\forall y) Fxy \supset (\forall y)(\exists x) Fxy)$	1-7, $\supset$ int

SB NSG AB et al. (*U*; new millennium)

$\mathcal{U}$

The Universe of Logics (including informal)

$DC\mathcal{E}\mathcal{C}^*$

$\leq \mathcal{L}_2 =$ second-order logic

$\leq \mathcal{L}_1 =$ first-order logic

$\mathcal{L}_0 =$ zero-order logic

$\mathcal{L}_{pc} \leq \mathcal{L} \leq \mathcal{L}_1 =$ description logics

$\mathcal{L}_{pc} =$ propositional logic

$IDC\mathcal{E}\mathcal{C}$

$\mathcal{L}_{\omega_1\omega}$

Infinitary

Finitary

Intensional

Extensional

Deductive

Inductive

Deductive

Inductive

Symbolic

Diagram

Symbolic

Diagram

Symbolic

Diagram

Symbolic

Diagram

And a key part of HyperLogic's coverage of $\mathcal{U}$ is ...

hypergraphical reasoning!

Hypergraphical Reasoning Key to Covering The Universe $\mathcal{U}$!

<http://www.logicamodernapproach.com/demos/UniverseViaHypergraphicalReasoning.mov>



Evolution ...

Alexander Bringsjord • Naveen Sundar Govindarajulu • Selmer Bringsjord



Hypergraphical Reasoning Key to Covering The Universe $\mathcal{U}$!

<http://www.logicamodernapproach.com/demos/UniverseViaHypergraphicalReasoning.mov>



Evolution ...

Alexander Bringsjord • Naveen Sundar Govindarajulu • Selmer Bringsjord



On the Road to 3D ...

Two-Dimensional (2D) Hypergraphical Proof Problem

(Leibnizian Theorem; Second-Order Logic, With Oracle Usage)

☒  Variant1LeibnizsLaw

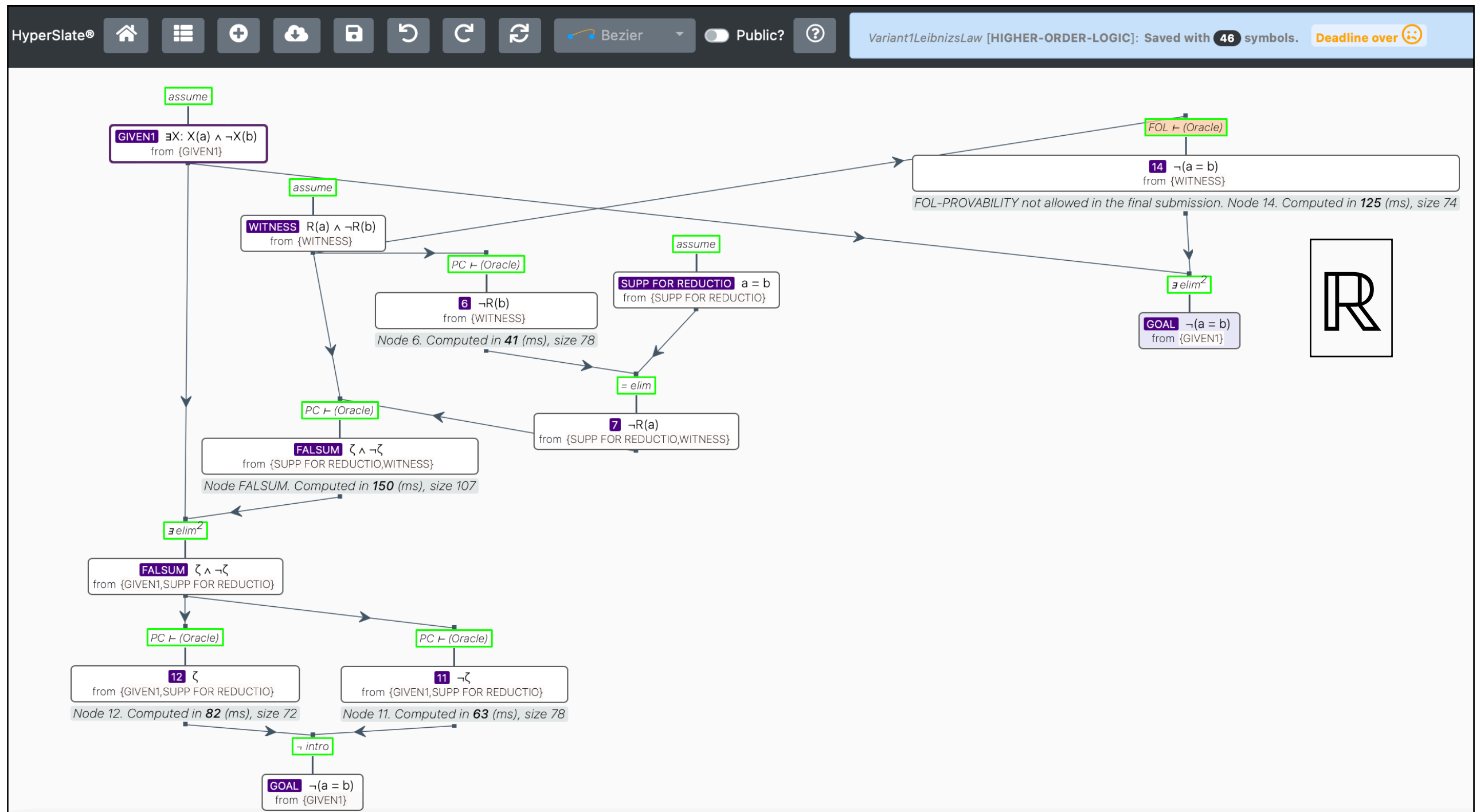
Here you must prove in second-order logic = $\mathcal{L}_2$ a consequence of Leibniz's Law (a.k.a. The Identity of Indiscernibles, nicely explained [here](#)), which says that any two things having exactly the same properties are identical, and *vice versa*. Here's the consequence in meta-logic (informally put, that if two things a and b differ with respect to a property, they can't be identical):

$$\exists X(Xa \wedge \neg Xb) \vdash_2 a \neq b$$

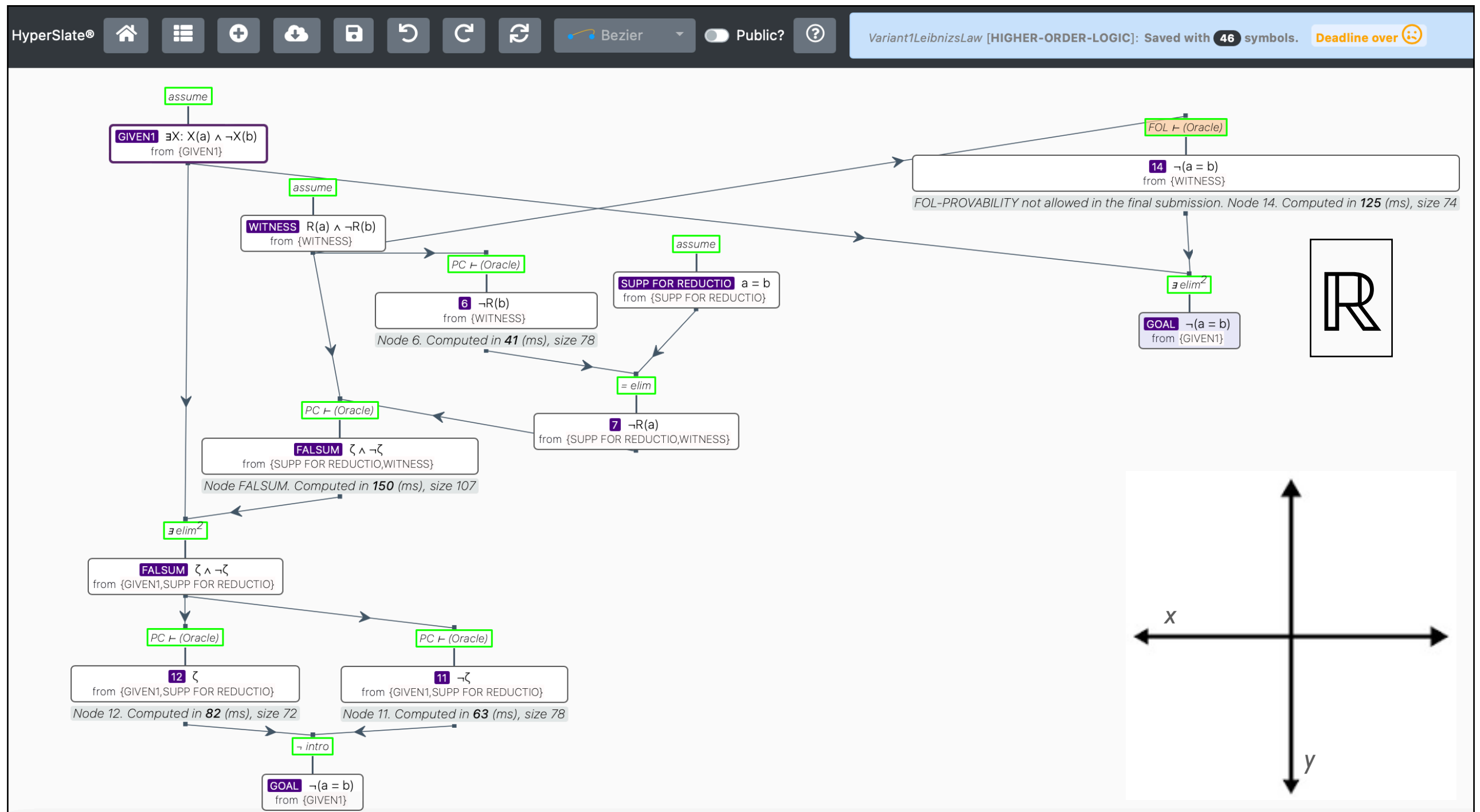
We use a subscript of '2' on the provability relation to indicate that the allowable inference schemata in your HyperSlate® are those from *second*-order logic.

You're permitted to leave remaining use of the PC provability oracle in your completed proof. (No deadline for now.)

Two-Dimensional (2D) Hypergraphical Proof (Leibnizian Theorem; Second-Order Logic, With Oracle Usage)



Two-Dimensional (2D) Hypergraphical Proof (Leibnizian Theorem; Second-Order Logic, With Oracle Usage)



This is all 2D.

Two.5-Dimensional (2.5D) Hypergraphical Proof of a Popular-for-Exposition Theorem in the Propositional Calculus

HyperSlate®

🏠

☰

⊕

☁

🔒

↶

↷

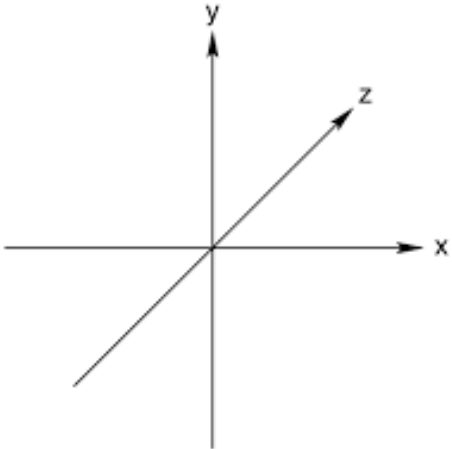
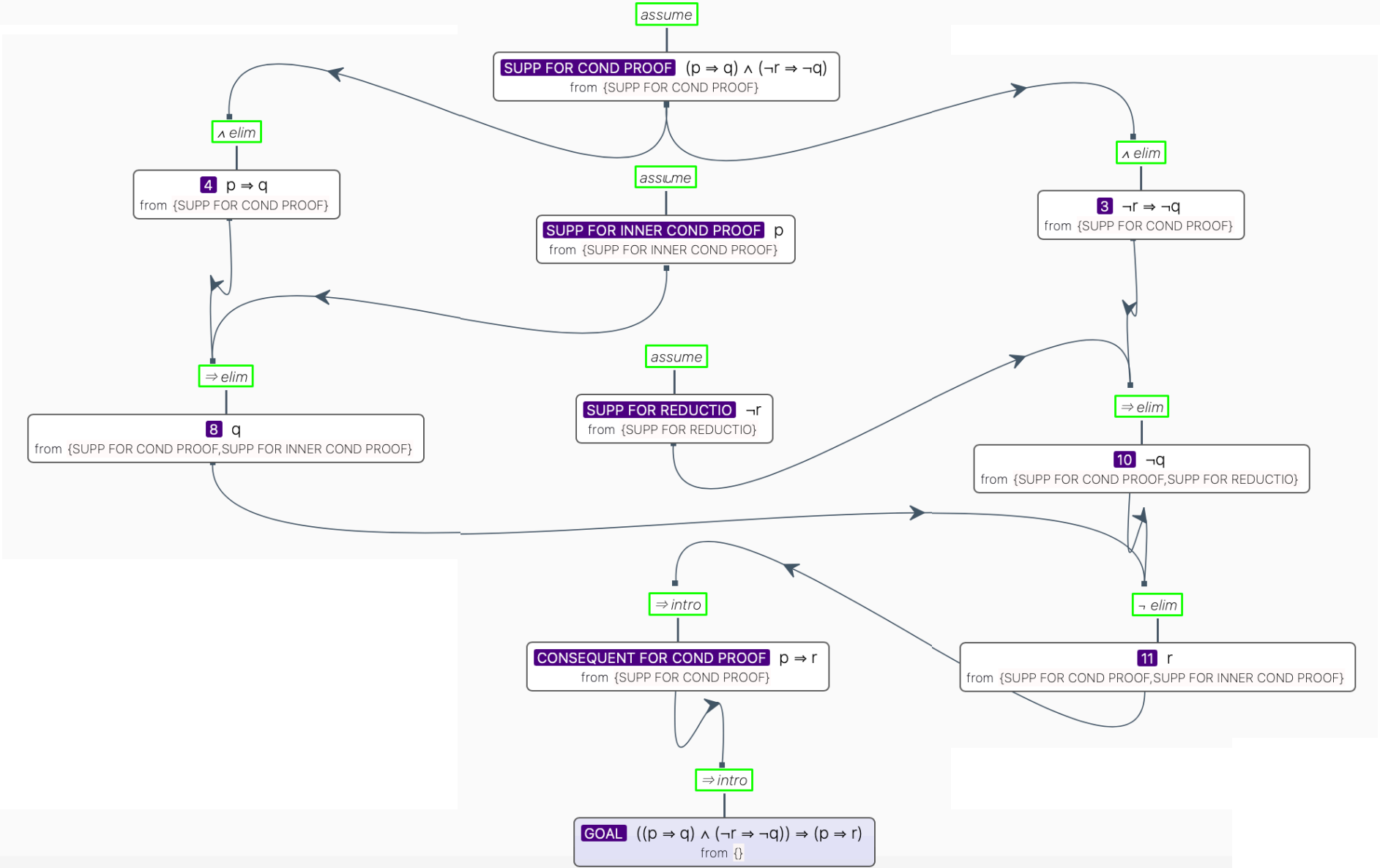
↺

Bezier

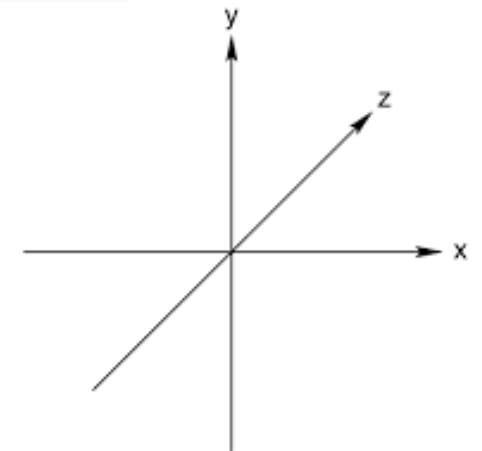
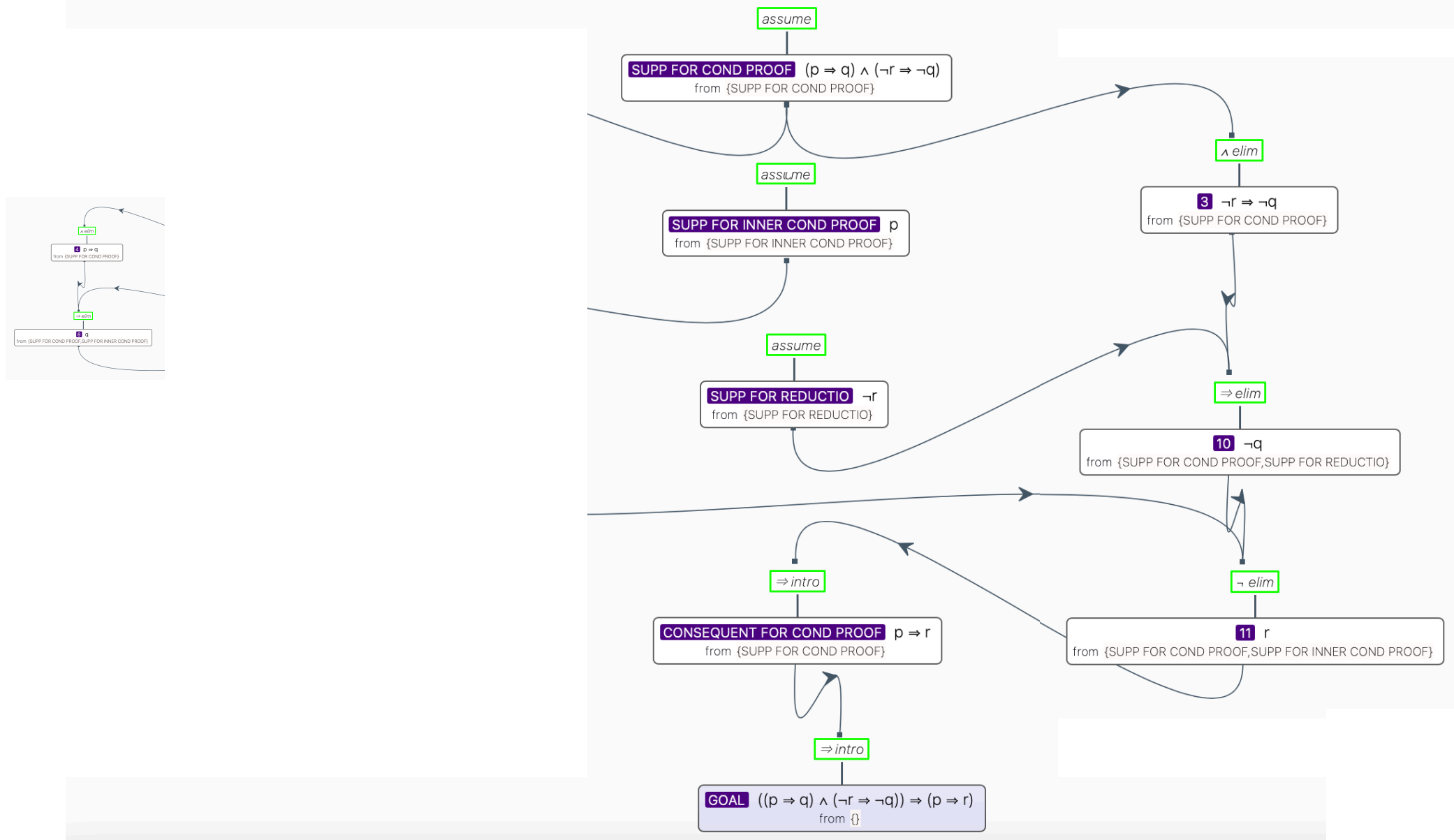
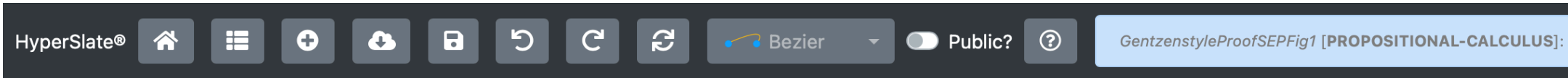
Public?

?

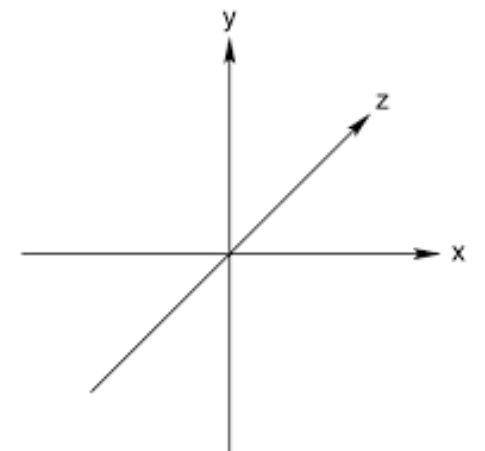
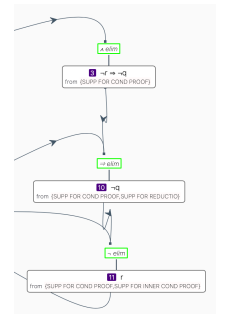
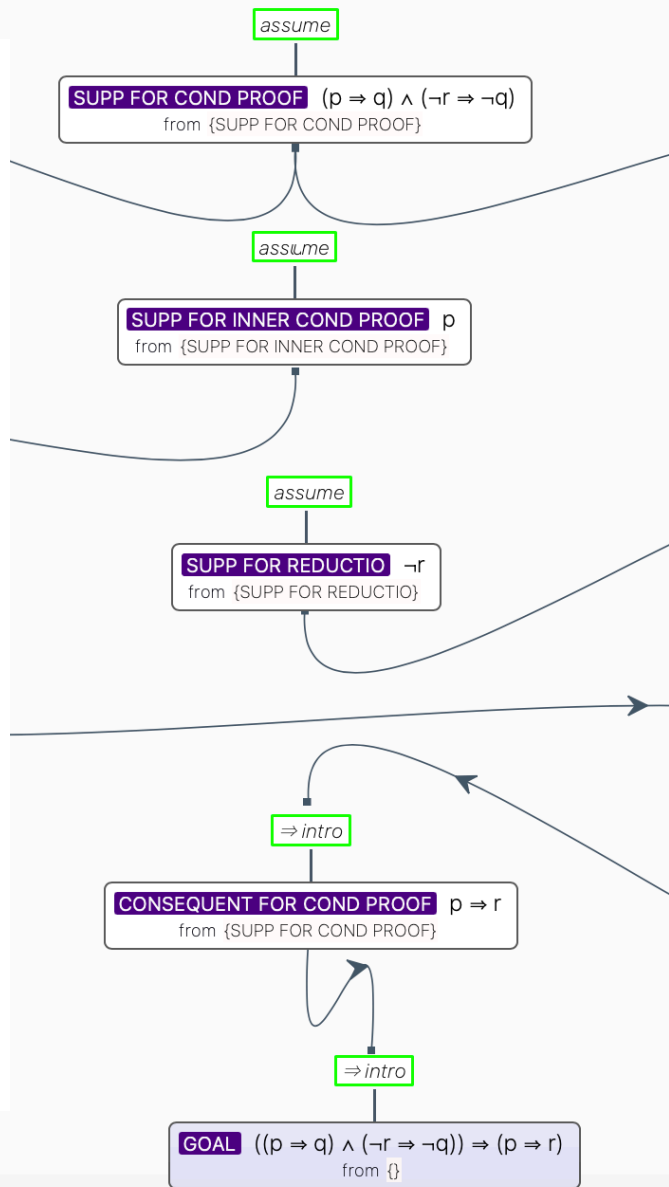
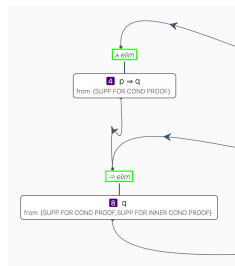
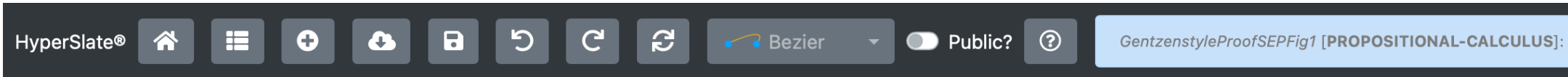
GentzenstyleProofSEPFig1 [PROPOSITIONAL-CALCULUS]:



Two.5-Dimensional (2.5D) Hypergraphical Proof of a Popular-for-Exposition Theorem in the Propositional Calculus

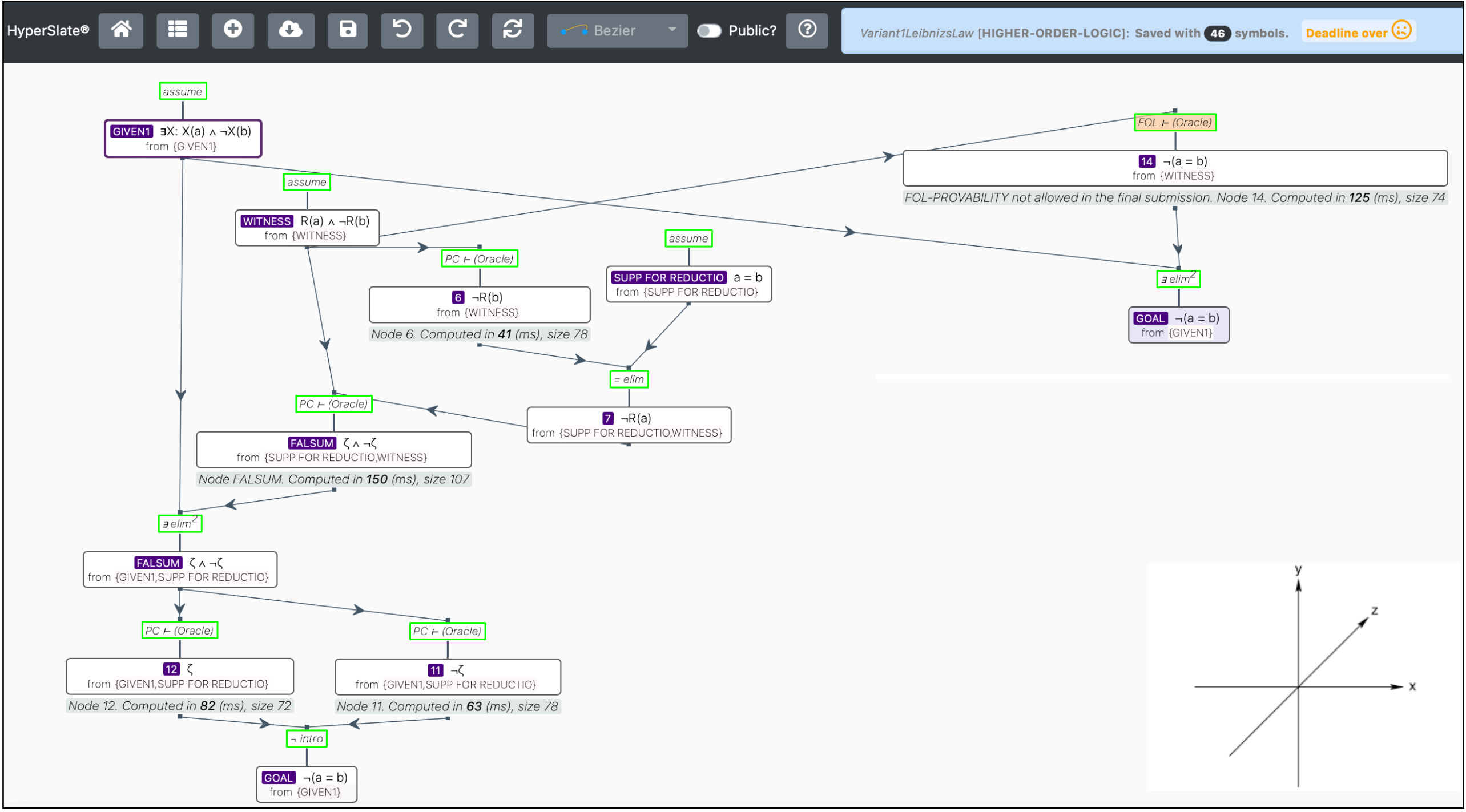


Two.5-Dimensional (2.5D) Hypergraphical Proof of a Popular-for-Exposition Theorem in the Propositional Calculus



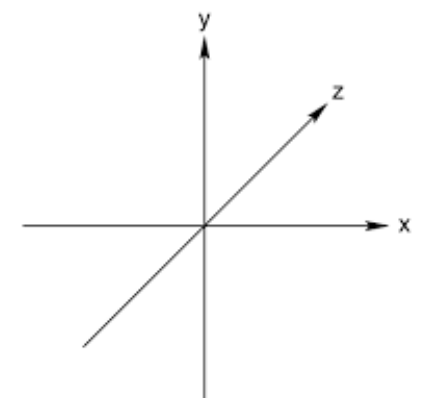
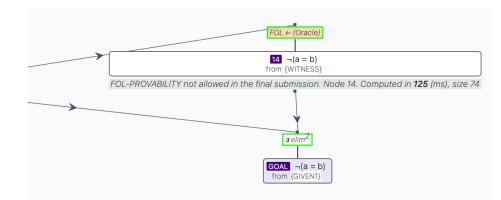
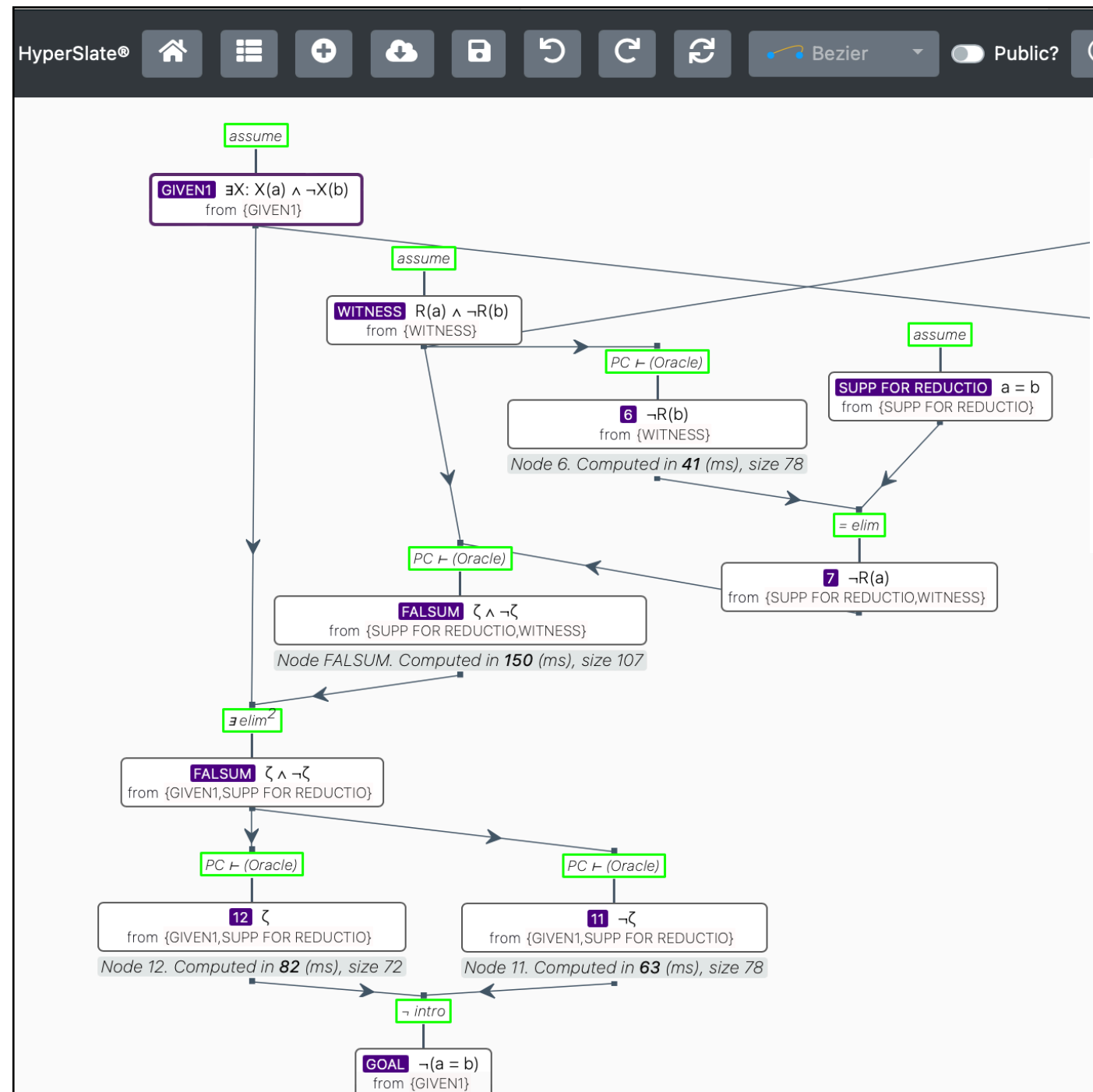
Three-Dimensional (2.5D) Hypergraphical Proof

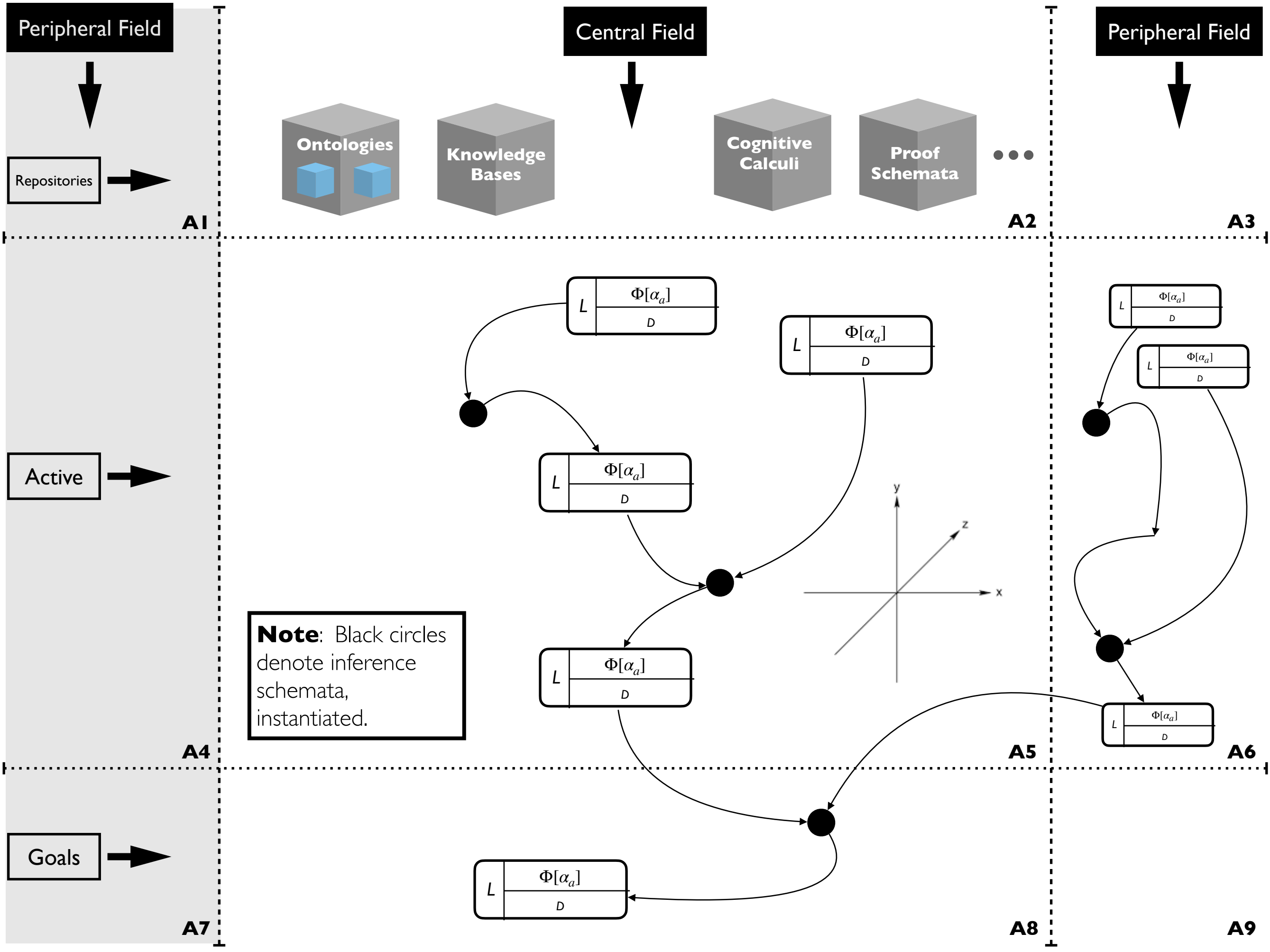
(Leibnizian Theorem; Second-Order Logic, With Oracle Usage)

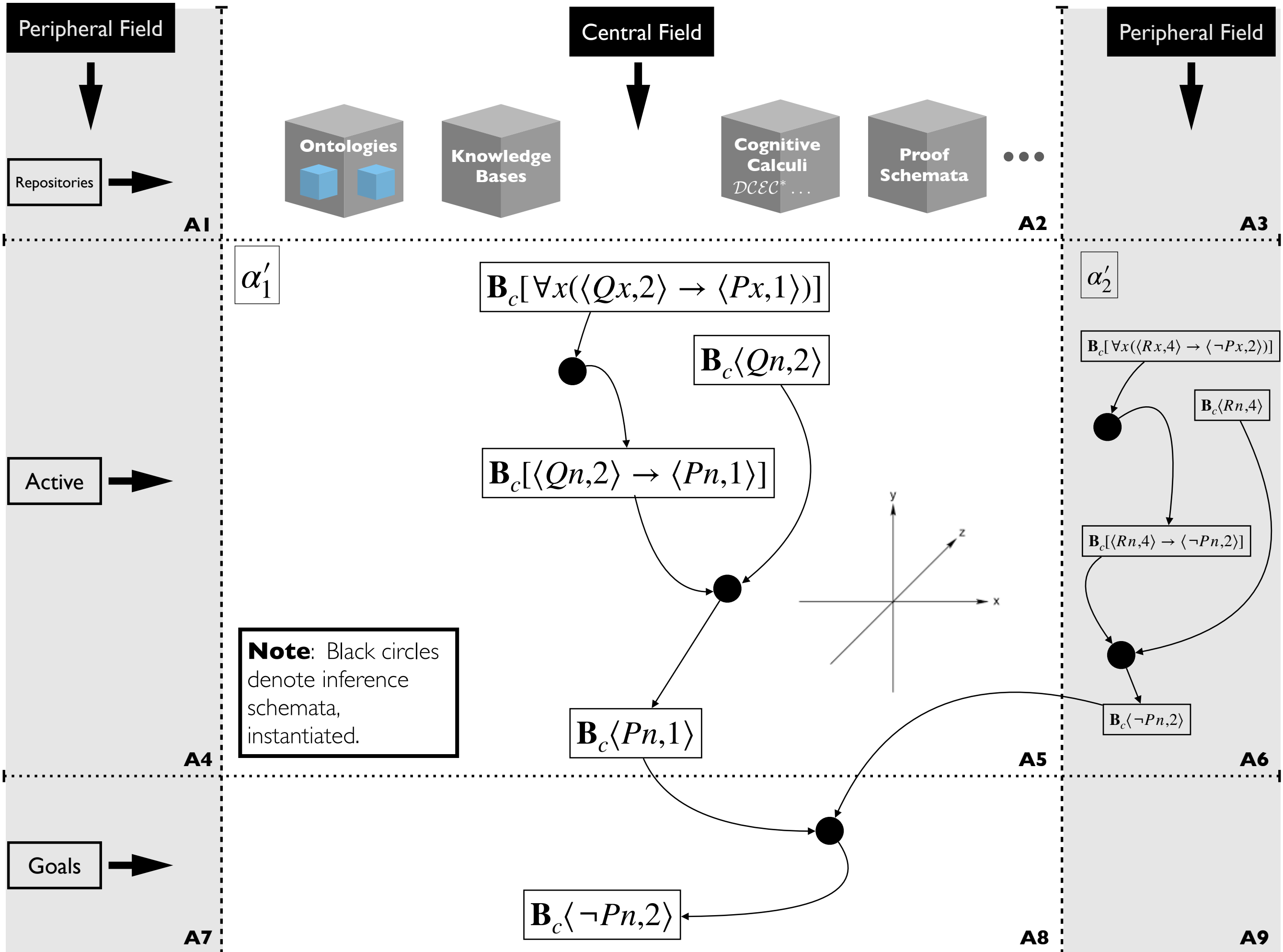


Three-Dimensional (2.5D) Hypergraphical Proof

(Leibnizian Theorem; Second-Order Logic, With Oracle Usage)







Peripheral Field



Repositories



A1

Central Field



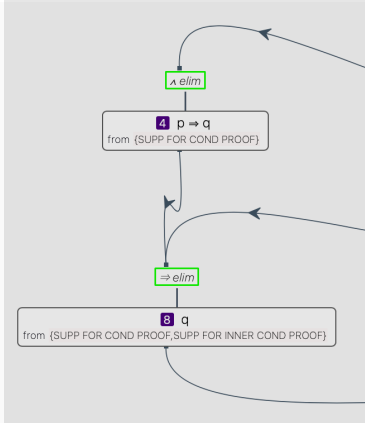
Cognitive Calculi; Axioms/Axioms Systems; Lemmas; Ontologies;
Knowledge/Belief Bases; Proof Schemata ...

A2

Peripheral Field



A3

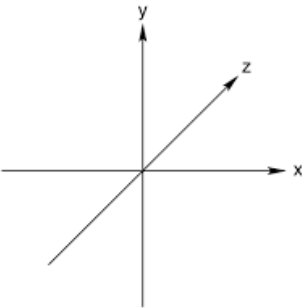
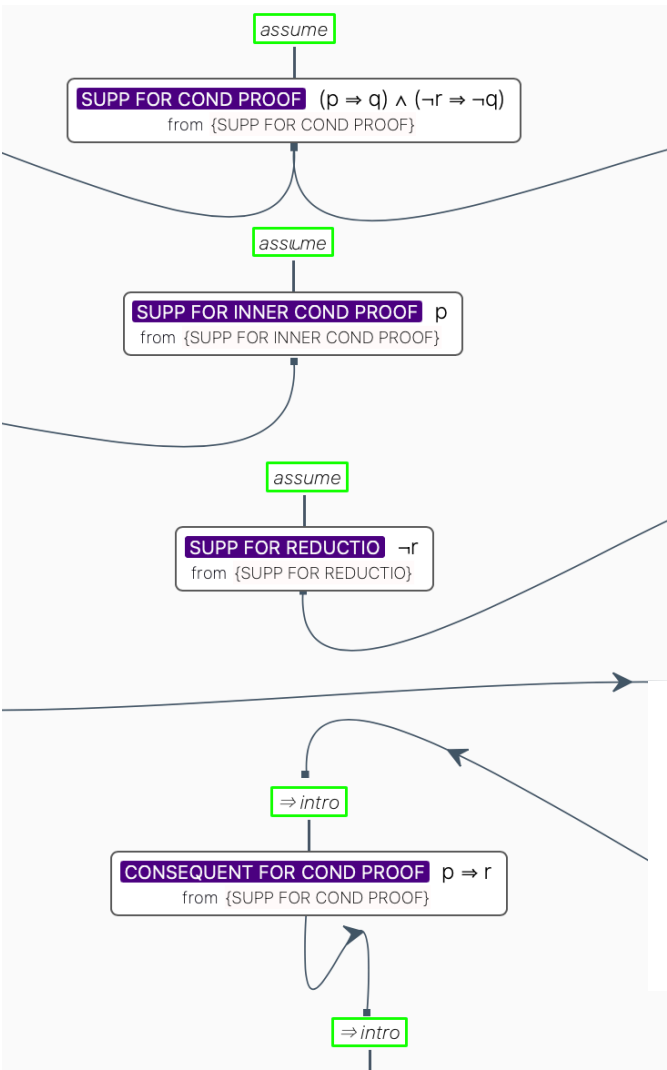


Active

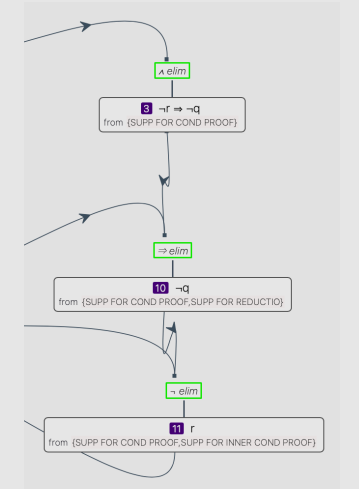


A4

Note: Smaller boxes preceding full nodes specify inference schema used to pass to full “downstream” nodes.



A5

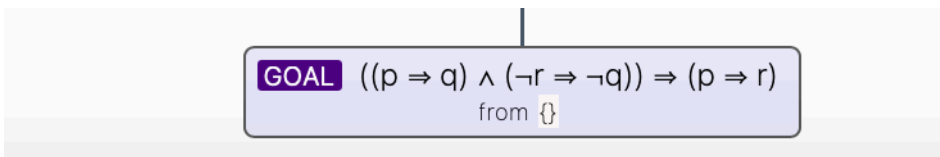


A6

Goals



A7



A8

A9

$DC\mathcal{E}\mathcal{C}^*$ and $IDC\mathcal{E}\mathcal{C}^*$

$DC\mathcal{E}\mathcal{C}$ Signature

$$\begin{aligned}
 S &::= \text{Agent} \mid \text{ActionType} \mid \text{Action} \sqsubseteq \text{Event} \mid \text{Moment} \mid \text{Fluent} \\
 f &::= \begin{cases} \text{action} : \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \\ \text{initially} : \text{Fluent} \rightarrow \text{Formula} \\ \text{holds} : \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{happens} : \text{Event} \times \text{Moment} \rightarrow \text{Formula} \\ \text{clipped} : \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{initiates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{terminates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{prior} : \text{Moment} \times \text{Moment} \rightarrow \text{Formula} \end{cases} \\
 t &::= x : S \mid c : S \mid f(t_1, \dots, t_n) \\
 \phi &::= \begin{cases} q : \text{Formula} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid \forall x : \phi(x) \mid \exists x : \phi(x) \\ \mathbf{P}(a, t, \phi) \mid \mathbf{K}(a, t, \phi) \mid \mathbf{S}(a, b, t, \phi) \mid \mathbf{S}(a, t, \phi) \\ \mathbf{C}(t, \phi) \mid \mathbf{B}(a, t, \phi) \mid \mathbf{D}(a, t, \phi) \mid \mathbf{I}(a, t, \phi) \\ \mathbf{O}(a, t, \phi, (\neg)\text{happens}(\text{action}(a^*, \alpha), t')) \end{cases}
 \end{aligned}$$

Modal Operator Descriptors:
Perceives, **K**nows, **S**ays, **C**ommon-knowledge
Believes, **D**esires, **I**ntends, **O**ught-to

$DC\mathcal{E}\mathcal{C}$ Inference Schemata

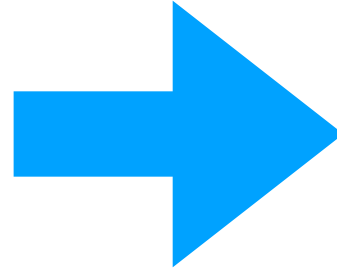
$$\begin{aligned}
 &\frac{\mathbf{K}(a, t_1, \Gamma), \Gamma \vdash \phi, t_1 \leq t_2}{\mathbf{K}(a, t_2, \phi)} [I_K] \quad \frac{\mathbf{B}(a, t_1, \Gamma), \Gamma \vdash \phi, t_1 \leq t_2}{\mathbf{B}(a, t_2, \phi)} [I_B] \\
 &\frac{}{\mathbf{C}(t, \mathbf{P}(a, t, \phi) \rightarrow \mathbf{K}(a, t, \phi))} [I_1] \quad \frac{}{\mathbf{C}(t, \mathbf{K}(a, t, \phi) \rightarrow \mathbf{B}(a, t, \phi))} [I_2] \\
 &\frac{\mathbf{C}(t, \phi) \ t \leq t_1 \dots t \leq t_n}{\mathbf{K}(a_1, t_1, \dots \mathbf{K}(a_n, t_n, \phi) \dots)} [I_3] \quad \frac{\mathbf{K}(a, t, \phi)}{\phi} [I_4] \\
 &\frac{}{\mathbf{C}(t, \mathbf{K}(a, t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{K}(a, t_2, \phi_1) \rightarrow \mathbf{K}(a, t_3, \phi_2)} [I_5] \\
 &\frac{}{\mathbf{C}(t, \mathbf{B}(a, t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{B}(a, t_2, \phi_1) \rightarrow \mathbf{B}(a, t_3, \phi_2)} [I_6] \\
 &\frac{}{\mathbf{C}(t, \mathbf{C}(t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{C}(t_2, \phi_1) \rightarrow \mathbf{C}(t_3, \phi_2)} [I_7] \\
 &\frac{}{\mathbf{C}(t, \forall x. \phi \rightarrow \phi[x \mapsto t])} [I_8] \quad \frac{}{\mathbf{C}(t, \phi_1 \leftrightarrow \phi_2 \rightarrow \neg\phi_2 \rightarrow \neg\phi_1)} [I_9] \\
 &\frac{}{\mathbf{C}(t, [\phi_1 \wedge \dots \wedge \phi_n \rightarrow \phi] \rightarrow [\phi_1 \rightarrow \dots \rightarrow \phi_n \rightarrow \phi])} [I_{10}] \\
 &\frac{\mathbf{B}(a, t, \phi) \ \mathbf{B}(a, t, \phi \rightarrow \psi)}{\mathbf{B}(a, t, \psi)} [I_{11a}] \quad \frac{\mathbf{B}(a, t, \phi) \ \mathbf{B}(a, t, \psi)}{\mathbf{B}(a, t, \phi \wedge \psi)} [I_{11b}] \\
 &\frac{\mathbf{S}(s, h, t, \phi)}{\mathbf{B}(h, t, \mathbf{B}(s, t, \phi))} [I_{12}] \quad \frac{\mathbf{I}(a, t, \text{happens}(\text{action}(a^*, \alpha), t'))}{\mathbf{P}(a, t, \text{happens}(\text{action}(a^*, \alpha), t'))} [I_{13}] \\
 &\frac{\mathbf{B}(a, t, \phi) \ \mathbf{B}(a, t, \mathbf{O}(a, t, \phi, \chi)) \ \mathbf{O}(a, t, \phi, \chi)}{\mathbf{K}(a, t, \mathbf{I}(a, t, \chi))} [I_{14}]
 \end{aligned}$$

$DC\mathcal{E}\mathcal{C}^*$ and $IDC\mathcal{E}\mathcal{C}^*$

$DC\mathcal{E}\mathcal{C}$ Signature

$S ::= \text{Agent} \mid \text{ActionType} \mid \text{Action} \sqsubseteq \text{Event} \mid \text{Moment} \mid \text{Fluent}$
 $f ::= \begin{cases} \text{action} : \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \\ \text{initially} : \text{Fluent} \rightarrow \text{Formula} \\ \text{holds} : \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{happens} : \text{Event} \times \text{Moment} \rightarrow \text{Formula} \\ \text{clipped} : \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{initiates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{terminates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{prior} : \text{Moment} \times \text{Moment} \rightarrow \text{Formula} \end{cases}$
 $t ::= x : S \mid c : S \mid f(t_1, \dots, t_n)$
 $\phi ::= \begin{cases} q : \text{Formula} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid \forall x : \phi(x) \mid \exists x : \phi(x) \\ \mathbf{P}(a, t, \phi) \mid \mathbf{K}(a, t, \phi) \mid \mathbf{S}(a, b, t, \phi) \mid \mathbf{S}(a, t, \phi) \\ \mathbf{C}(t, \phi) \mid \mathbf{B}(a, t, \phi) \mid \mathbf{D}(a, t, \phi) \mid \mathbf{I}(a, t, \phi) \\ \mathbf{O}(a, t, \phi, (\neg)\text{happens}(\text{action}(a^*, \alpha), t')) \end{cases}$

Modal Operator Descriptors:
 Perceives, **K**nows, Says, **C**ommon-knowledge
 Believes, **D**esires, **I**ntends, **O**ught-to



$DC\mathcal{E}\mathcal{C}$ Inference Schemata

$$\frac{\mathbf{K}(a, t_1, \Gamma), \Gamma \vdash \phi, t_1 \leq t_2}{\mathbf{K}(a, t_2, \phi)} [I_K] \quad \frac{\mathbf{B}(a, t_1, \Gamma), \Gamma \vdash \phi, t_1 \leq t_2}{\mathbf{B}(a, t_2, \phi)} [I_B]$$

$$\frac{}{\mathbf{C}(t, \mathbf{P}(a, t, \phi) \rightarrow \mathbf{K}(a, t, \phi))} [I_1] \quad \frac{}{\mathbf{C}(t, \mathbf{K}(a, t, \phi) \rightarrow \mathbf{B}(a, t, \phi))} [I_2]$$

$$\frac{\mathbf{C}(t, \phi) \ t \leq t_1 \dots t \leq t_n}{\mathbf{K}(a_1, t_1, \dots \mathbf{K}(a_n, t_n, \phi) \dots)} [I_3] \quad \frac{\mathbf{K}(a, t, \phi)}{\phi} [I_4]$$

$$\frac{}{\mathbf{C}(t, \mathbf{K}(a, t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{K}(a, t_2, \phi_1) \rightarrow \mathbf{K}(a, t_3, \phi_2)} [I_5]$$

$$\frac{}{\mathbf{C}(t, \mathbf{B}(a, t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{B}(a, t_2, \phi_1) \rightarrow \mathbf{B}(a, t_3, \phi_2)} [I_6]$$

$$\frac{}{\mathbf{C}(t, \mathbf{C}(t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{C}(t_2, \phi_1) \rightarrow \mathbf{C}(t_3, \phi_2)} [I_7]$$

$$\frac{}{\mathbf{C}(t, \forall x. \phi \rightarrow \phi[x \mapsto t])} [I_8] \quad \frac{}{\mathbf{C}(t, \phi_1 \leftrightarrow \phi_2 \rightarrow \neg\phi_2 \rightarrow \neg\phi_1)} [I_9]$$

$$\frac{}{\mathbf{C}(t, [\phi_1 \wedge \dots \wedge \phi_n \rightarrow \phi] \rightarrow [\phi_1 \rightarrow \dots \rightarrow \phi_n \rightarrow \phi])} [I_{10}]$$

$$\frac{\mathbf{B}(a, t, \phi) \ \mathbf{B}(a, t, \phi \rightarrow \psi)}{\mathbf{B}(a, t, \psi)} [I_{11a}] \quad \frac{\mathbf{B}(a, t, \phi) \ \mathbf{B}(a, t, \psi)}{\mathbf{B}(a, t, \phi \wedge \psi)} [I_{11b}]$$

$$\frac{\mathbf{S}(s, h, t, \phi)}{\mathbf{B}(h, t, \mathbf{B}(s, t, \phi))} [I_{12}] \quad \frac{\mathbf{I}(a, t, \text{happens}(\text{action}(a^*, \alpha), t'))}{\mathbf{P}(a, t, \text{happens}(\text{action}(a^*, \alpha), t'))} [I_{13}]$$

$$\frac{\mathbf{B}(a, t, \phi) \ \mathbf{B}(a, t, \mathbf{O}(a, t, \phi, \chi)) \ \mathbf{O}(a, t, \phi, \chi)}{\mathbf{K}(a, t, \mathbf{I}(a, t, \chi))} [I_{14}]$$

$DC\mathcal{E}C^*$ and $IDC\mathcal{E}C^*$

$DC\mathcal{E}C$ Signature

$S ::= \text{Agent} \mid \text{ActionType} \mid \text{Action} \sqsubseteq \text{Event} \mid \text{Moment} \mid \text{Fluent}$
 $f ::= \begin{cases} \text{action} : \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \\ \text{initially} : \text{Fluent} \rightarrow \text{Formula} \\ \text{holds} : \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{happens} : \text{Event} \times \text{Moment} \rightarrow \text{Formula} \\ \text{clipped} : \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{initiates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{terminates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \text{prior} : \text{Moment} \times \text{Moment} \rightarrow \text{Formula} \end{cases}$
 $t ::= x : S \mid c : S \mid f(t_1, \dots, t_n)$
 $\phi ::= \begin{cases} q : \text{Formula} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid \forall x : \phi(x) \mid \exists x : \phi(x) \\ \mathbf{P}(a, t, \phi) \mid \mathbf{K}(a, t, \phi) \mid \mathbf{S}(a, b, t, \phi) \mid \mathbf{S}(a, t, \phi) \\ \mathbf{C}(t, \phi) \mid \mathbf{B}(a, t, \phi) \mid \mathbf{D}(a, t, \phi) \mid \mathbf{I}(a, t, \phi) \\ \mathbf{O}(a, t, \phi, (\neg)\text{happens}(\text{action}(a^*, \alpha), t')) \end{cases}$

Modal Operator Descriptors:
 Perceives, Knows, Says, Common-knowledge
 Believes, Desires, Intends, Ought-to

$DC\mathcal{E}C$ Inference Schemata

$\frac{\mathbf{K}(a, t_1, \Gamma), \Gamma \vdash \phi, t_1 \leq t_2}{\mathbf{K}(a, t_2, \phi)} [I_K] \quad \frac{\mathbf{B}(a, t_1, \Gamma), \Gamma \vdash \phi, t_1 \leq t_2}{\mathbf{B}(a, t_2, \phi)} [I_B]$
 $\frac{\mathbf{C}(t, \mathbf{P}(a, t, \phi) \rightarrow \mathbf{K}(a, t, \phi))}{\mathbf{C}(t, \phi)} [I_1] \quad \frac{\mathbf{C}(t, \mathbf{K}(a, t, \phi) \rightarrow \mathbf{B}(a, t, \phi))}{\mathbf{C}(t, \phi)} [I_2]$
 $\frac{\mathbf{C}(t, \phi) \quad t \leq t_1 \dots t \leq t_n}{\mathbf{K}(a_1, t_1, \dots \mathbf{K}(a_n, t_n, \phi) \dots)} [I_3] \quad \frac{\mathbf{K}(a, t, \phi)}{\phi} [I_4]$
 $\frac{\mathbf{C}(t, \mathbf{K}(a, t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{K}(a, t_2, \phi_1) \rightarrow \mathbf{K}(a, t_3, \phi_2)}{\mathbf{C}(t, \mathbf{B}(a, t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{B}(a, t_2, \phi_1) \rightarrow \mathbf{B}(a, t_3, \phi_2)} [I_5]$
 $\frac{\mathbf{C}(t, \mathbf{B}(a, t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{B}(a, t_2, \phi_1) \rightarrow \mathbf{B}(a, t_3, \phi_2)}{\mathbf{C}(t, \mathbf{C}(t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{C}(t_2, \phi_1) \rightarrow \mathbf{C}(t_3, \phi_2)} [I_6]$
 $\frac{\mathbf{C}(t, \mathbf{C}(t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{C}(t_2, \phi_1) \rightarrow \mathbf{C}(t_3, \phi_2)}{\mathbf{C}(t, \forall x. \phi \rightarrow \phi[x \mapsto t])} [I_7] \quad \frac{\mathbf{C}(t, \phi_1 \leftrightarrow \phi_2 \rightarrow \neg\phi_2 \rightarrow \neg\phi_1)}{\mathbf{C}(t, \forall x. \phi \rightarrow \phi[x \mapsto t])} [I_8]$
 $\frac{\mathbf{C}(t, [\phi_1 \wedge \dots \wedge \phi_n \rightarrow \phi] \rightarrow [\phi_1 \rightarrow \dots \rightarrow \phi_n \rightarrow \phi])}{\mathbf{B}(a, t, \phi) \quad \mathbf{B}(a, t, \phi \rightarrow \psi)} [I_{11a}] \quad \frac{\mathbf{B}(a, t, \phi) \quad \mathbf{B}(a, t, \psi)}{\mathbf{B}(a, t, \phi \wedge \psi)} [I_{11b}]$
 $\frac{\mathbf{S}(s, h, t, \phi)}{\mathbf{B}(h, t, \mathbf{B}(s, t, \phi))} [I_{12}] \quad \frac{\mathbf{I}(a, t, \text{happens}(\text{action}(a^*, \alpha), t'))}{\mathbf{P}(a, t, \text{happens}(\text{action}(a^*, \alpha), t'))} [I_{13}]$
 $\frac{\mathbf{B}(a, t, \phi) \quad \mathbf{B}(a, t, \mathbf{O}(a, t, \phi, \chi)) \quad \mathbf{O}(a, t, \phi, \chi)}{\mathbf{K}(a, t, \mathbf{I}(a, t, \chi))} [I_{14}]$

2.2 Inductive Deontic Cognitive Event Calculus

$DC\mathcal{E}C$ employs no uncertainty system (e.g., probability measures, *strength factors*, or likelihood measures) and hence is purely deductive. Therefore, as we wish to enable our agents to reason about situations involving uncertainty, we must ultimately utilize the *Inductive $DC\mathcal{E}C$* : $IDC\mathcal{E}C$.

In general, to go from a deductive to an inductive cognitive calculus, we require two components: (1) an uncertainty system, and (2) inference schemata that delineate the methods by which inferences linking formulae and other information can be used to build formally valid arguments.

The particular uncertainty system we use herein is discussed in §2.3. The inference schemata of $IDC\mathcal{E}C$ consist of the union of the set presented in §2.1 with that in the box titled **Additional Inference Schemata for $IDC\mathcal{E}C$** . Likewise, the signature of $IDC\mathcal{E}C$ subsumes that of the deductive $DC\mathcal{E}C$; the syntax of $IDC\mathcal{E}C$ also includes the forms given in the box titled **Additional Syntax for $IDC\mathcal{E}C$** .

Additional Syntax for $IDC\mathcal{E}C$

$\phi ::= \{ \mathbf{B}^\sigma(a, t, \phi) \}$
 where $\sigma \in [-5, -4, \dots, 4, 5]$

Additional Inference Schemata for $IDC\mathcal{E}C$

$\frac{\mathbf{P}(a, t_1, \phi_1), \Gamma \vdash t_1 < t_2}{\mathbf{B}^4(a, t_2, \phi)} [I_P^s]$
 $\frac{\mathbf{B}^{\sigma_1}(a, t_1, \phi_1), \dots, \mathbf{B}^{\sigma_m}(a, t_m, \phi_m), \{\phi_1, \dots, \phi_m\} \vdash \phi, \{\phi_1, \dots, \phi_m\} \not\vdash \zeta, \Gamma \vdash t_i < t}{\mathbf{B}^{\min(\sigma_1, \dots, \sigma_m)}(a, t, \phi)} [I_B^s]$
 where $\sigma \in [0, 1, \dots, 5, 6]$
 $\frac{}{\mathbf{C}(t, \mathbf{B}^{-\sigma}(a, t, \phi) \leftrightarrow \mathbf{B}^\sigma(a, t, \neg\phi))} [I_\neg^s]$

Briefly, $\mathbf{B}^\sigma(a, t, \phi)$ denotes that agent a at time t believes ϕ with uncertainty σ . We justify in the next section the range of values for σ .

The first inference schema allows agents to infer evident beliefs ($\sigma = 4$, as defined in the next section) from what they perceive.⁵ The second schema allows agents to infer a belief that is provable from the beliefs they currently assert, so long as the belief set is not inconsistent. In practice, we usually check that the belief set is consistent by attempting to prove a reserved propositional atom ζ which does not