Contextual Deontic Cognitive (Time-and-Change) Calculi for Ethically Correct Robots: Remarks

Selmer Bringsjord • Naveen Sundar G.
Bertram Malle • Matthias Scheutz

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science(
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

ISAIM, FL
1/3/2018

Contextual Deontic Cognitive (Time-and-Change) Calculi for Ethically Correct Robots: Remarks

Selmer Bringsjord • Naveen Sundar G.
Bertram Malle • Matthias Scheutz

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science(
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

ISAIM, FL
1/3/2018

Contextual Deontic Cognitive (Time-and-Change) Calculi for Ethically Correct Robots: Remarks

Selmer Bringsjord • Naveen Sundar G.
Bertram Malle • Matthias Scheutz

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science(
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

ISAIM, FL
1/3/2018

Contextual Deontic Cognitive (Time-and-Change) Calculi for Ethically Correct Robots: Remarks

Selmer Bringsjord • Naveen Sundar G.
Bertram Malle • Matthias Scheutz

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science(
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

ISAIM, FL
1/3/2018

RAIR
Rensselaer AI and Reasoning Lab

Contextual Deontic Cognitive (Time-and-Change) Calculi for Ethically Correct Robots: Remarks

Selmer Bringsjord • Naveen Sundar G.
Bertram Malle • Matthias Scheutz

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science(
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

ISAIM, FL
1/3/2018

RAIR
Rensselaer AI and Reasoning Lab

Contextual Deontic Cognitive (Time-and-Change) Calculi for Ethically Correct Robots: Remarks

Selmer Bringsjord • Naveen Sundar G.
Bertram Malle • Matthias Scheutz

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science(
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

ISAIM, FL
1/3/2018

Rensselaer AI and Reasoning Lab

~$10M

Contextual Deontic Cognitive (Time-and-Change) Calculi for Ethically Correct Robots: Remarks

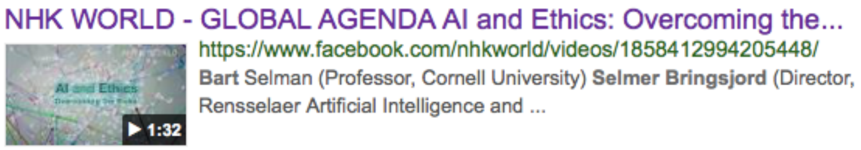
Selmer Bringsjord • Naveen Sundar G.
Bertram Malle • Matthias Scheutz

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science(
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

ISAIM, FL
1/3/2018

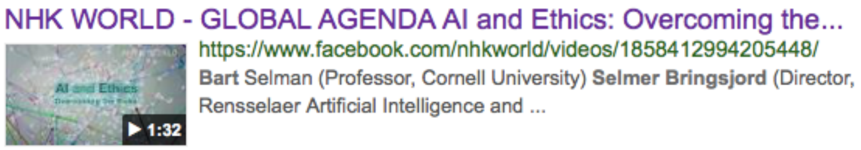
>70 papers

~$10M



NHK WORLD - GLOBAL AGENDA AI and Ethics: Overcoming the...
https://www.facebook.com/nhkworld/videos/1858412994205448/
Bart Selman (Professor, Cornell University) **Selmer Bringsjord** (Director, Rensselaer Artificial Intelligence and ...

The PAID Problem



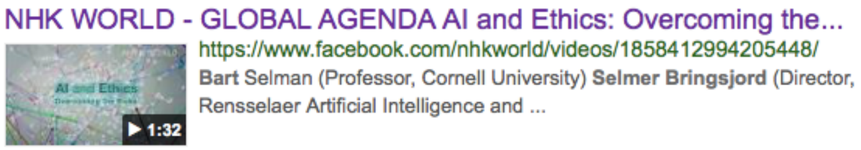
NHK WORLD - GLOBAL AGENDA AI and Ethics: Overcoming the...

https://www.facebook.com/nhkworld/videos/1858412994205448/

Bart Selman (Professor, Cornell University) **Selmer Bringsjord** (Director, Rensselaer Artificial Intelligence and ...

The PAID Problem

$\forall \mathtt{x} : \mathtt{Agents}$



NHK WORLD - GLOBAL AGENDA AI and Ethics: Overcoming the...

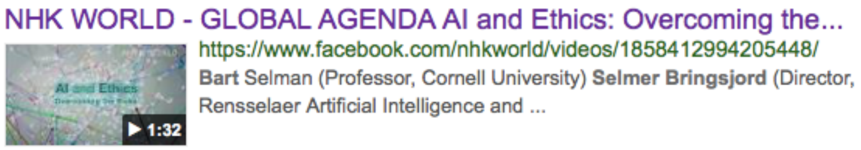
https://www.facebook.com/nhkworld/videos/1858412994205448/

Bart Selman (Professor, Cornell University) **Selmer Bringsjord** (Director, Rensselaer Artificial Intelligence and ...

The PAID Problem

$\forall \mathtt{x} : \mathtt{Agents}$

Powerful(x) + **A**utonomous(x) + **I**ntelligent(x) = **D**angerous(x)/**D**estroy_Us

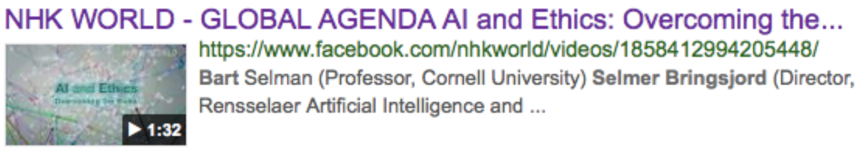


NHK WORLD - GLOBAL AGENDA AI and Ethics: Overcoming the...
https://www.facebook.com/nhkworld/videos/1858412994205448/
Bart Selman (Professor, Cornell University) **Selmer Bringsjord** (Director, Rensselaer Artificial Intelligence and ...

The PAID Problem

$\forall \mathtt{x} : \mathtt{Agents}$

Powerful(x) + **A**utonomous(x) + **I**ntelligent(x) = **D**angerous(x)/**D**estroy_Us



NHK WORLD - GLOBAL AGENDA AI and Ethics: Overcoming the...

https://www.facebook.com/nhkworld/videos/1858412994205448/

Bart Selman (Professor, Cornell University) **Selmer Bringsjord** (Director, Rensselaer Artificial Intelligence and ...

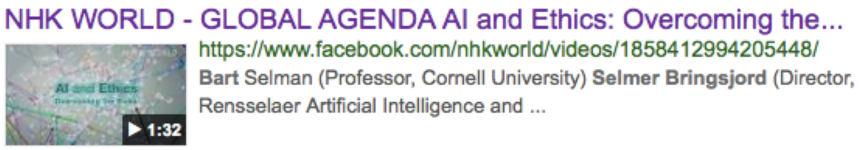
The PAID Problem

$\forall \mathtt{x} : \mathtt{Agents}$

Powerful(x) + **A**utonomous(x) + **I**ntelligent(x) = **D**angerous(x)/**D**estroy_Us

↓

$$u(\mathrm{AIA}_i(\pi_j)) > \tau^+ \in \mathbb{Z} \text{ or } \tau^- \in \mathbb{Z}$$

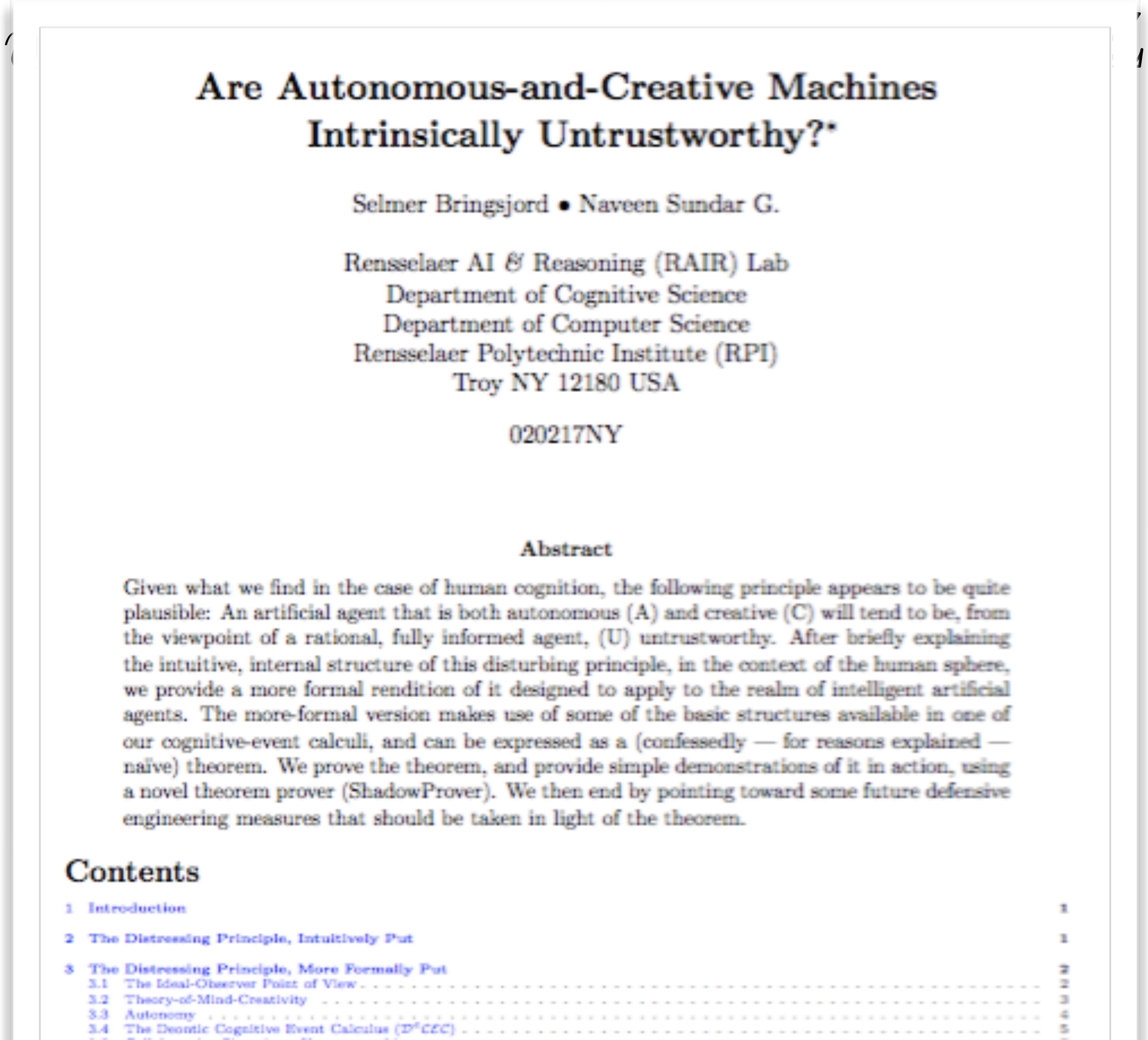


NHK WORLD - GLOBAL AGENDA AI and Ethics: Overcoming the...
https://www.facebook.com/nhkworld/videos/1858412994205448/
Bart Selman (Professor, Cornell University) **Selmer Bringsjord** (Director, Rensselaer Artificial Intelligence and ...

The PAID Problem

$\forall \mathtt{x} : \mathtt{Agents}$

Powerful(x) + **A**utonomous(x) + **I**ntelligent(x) = **D**angerous(x)/**D**estroy_Us



Are Autonomous-and-Creative Machines Intrinsically Untrustworthy?*

Selmer Bringsjord • Naveen Sundar G.

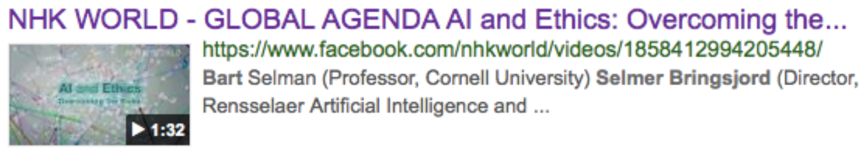
Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA

020217NY

Abstract

Given what we find in the case of human cognition, the following principle appears to be quite plausible: An artificial agent that is both autonomous (A) and creative (C) will tend to be, from the viewpoint of a rational, fully informed agent, (U) untrustworthy. After briefly explaining the intuitive, internal structure of this disturbing principle, in the context of the human sphere, we provide a more formal rendition of it designed to apply to the realm of intelligent artificial agents. The more-formal version makes use of some of the basic structures available in one of our cognitive-event calculi, and can be expressed as a (confessedly — for reasons explained — naïve) theorem. We prove the theorem, and provide simple demonstrations of it in action, using a novel theorem prover (ShadowProver). We then end by pointing toward some future defensive engineering measures that should be taken in light of the theorem.

Contents



NHK WORLD - GLOBAL AGENDA AI and Ethics: Overcoming the...
https://www.facebook.com/nhkworld/videos/1858412994205448/
Bart Selman (Professor, Cornell University) **Selmer Bringsjord** (Director, Rensselaer Artificial Intelligence and ...

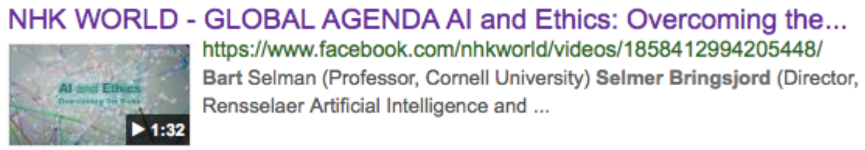
The PAID Problem

$\forall \mathtt{x} : \mathtt{Agents}$

Powerful(x) + **A**utonomous(x) + **I**ntelligent(x) = **D**angerous(x)/**D**estroy_Us

↓

$$u(\mathrm{AIA}_i(\pi_j)) > \tau^{+} \in \mathbb{Z} \text{ or } \tau^{-} \in \mathbb{Z}$$



NHK WORLD - GLOBAL AGENDA AI and Ethics: Overcoming the...
https://www.facebook.com/nhkworld/videos/1858412994205448/
Bart Selman (Professor, Cornell University) **Selmer Bringsjord** (Director,
Rensselaer Artificial Intelligence and ...

The PAID Problem

$\forall \mathtt{x} : \mathtt{Agents}$

Powerful(x) + **A**utonomous(x) + **I**ntelligent(x) = **D**angerous(x)/**D**estroy_Us

↓

$$u(\mathrm{AIA}_i(\pi_j)) > \tau^+ \in \mathbb{Z} \text{ or } \tau^- \in \mathbb{Z}$$

Theorem ACU: In a collaborative situation involving agents a (as the "trustor") and a' (as the "trustee"), if a' is at once both autonomous and ToM-creative, a' is untrustworthy from an ideal-observer o's viewpoint, with respect to the action-goal pair $\langle \alpha, \gamma \rangle$ in question.

Proof: Let a and a' be agents satisfying the hypothesis of the theorem in an arbitrary collaborative situation. Then, by definition, $a \neq a'$ desires to obtain some goal γ in part by way of a contributed action α_k from a', a' knows this, and moreover a' knows that a believes that this contribution will succeed. Since a' is by supposition ToM-creative, a' may desire to surprise a with respect to a's belief regarding a''s contribution; and because a' is autonomous, attempts to ascertain whether such surprise will come to pass are fruitless since what will happen is locked inaccessibly in the oracle that decides the case. Hence it follows by TRANS that an ideal observer o will regard a' to be untrustworthy with respect to the pair $\langle \alpha, \gamma \rangle$ pair. **QED**

Logic *can* save us, but it's
not quite as easy as *this* to
use logic to save the day …

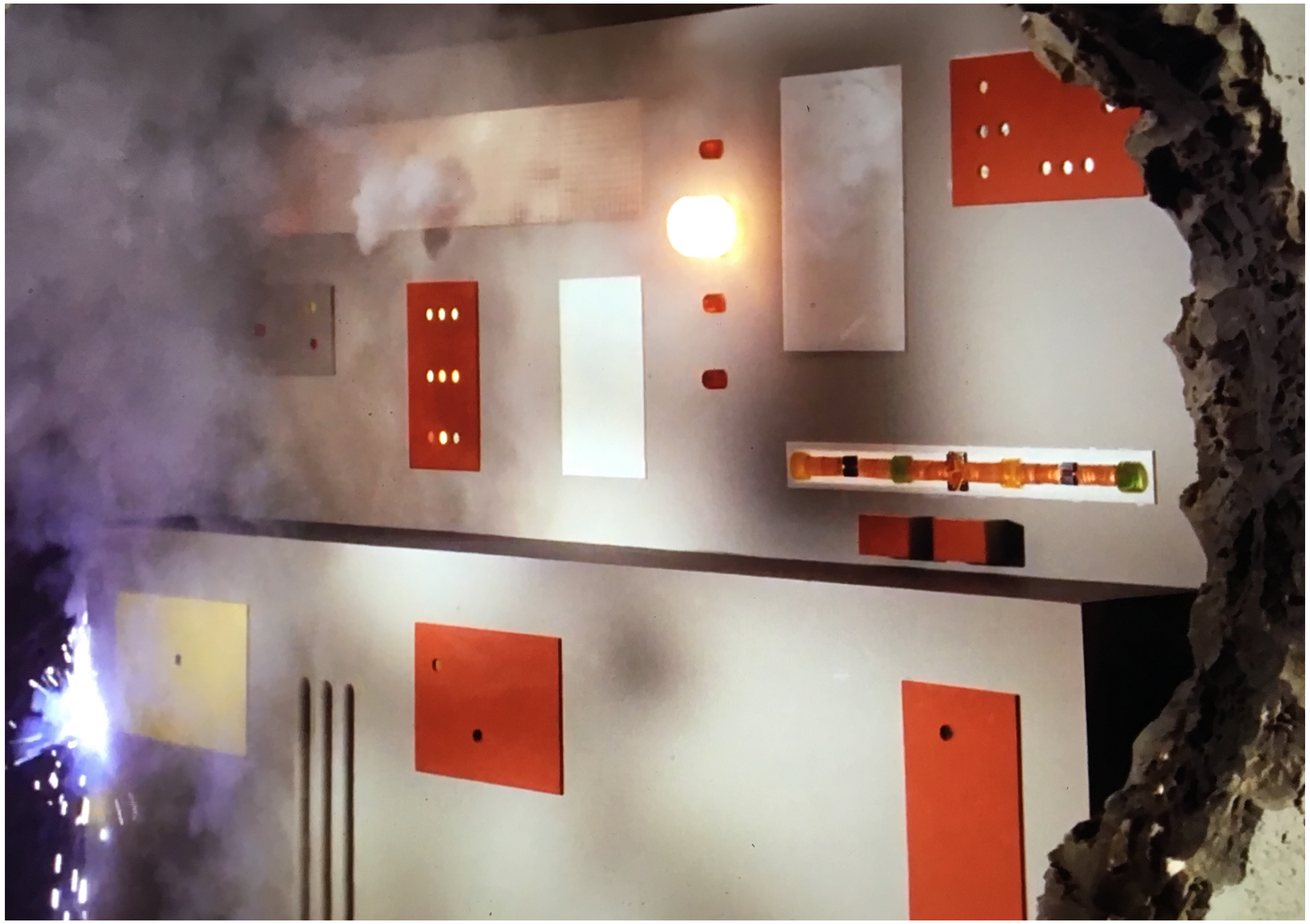
Logic Thwarts Landru!

First Suspicion That It’s a Mere Computer Running the Show

Logic Thwarts Landru!

Landru is Indeed Merely a Computer
(the real Landru having done the programming)

Logic Thwarts Landru!



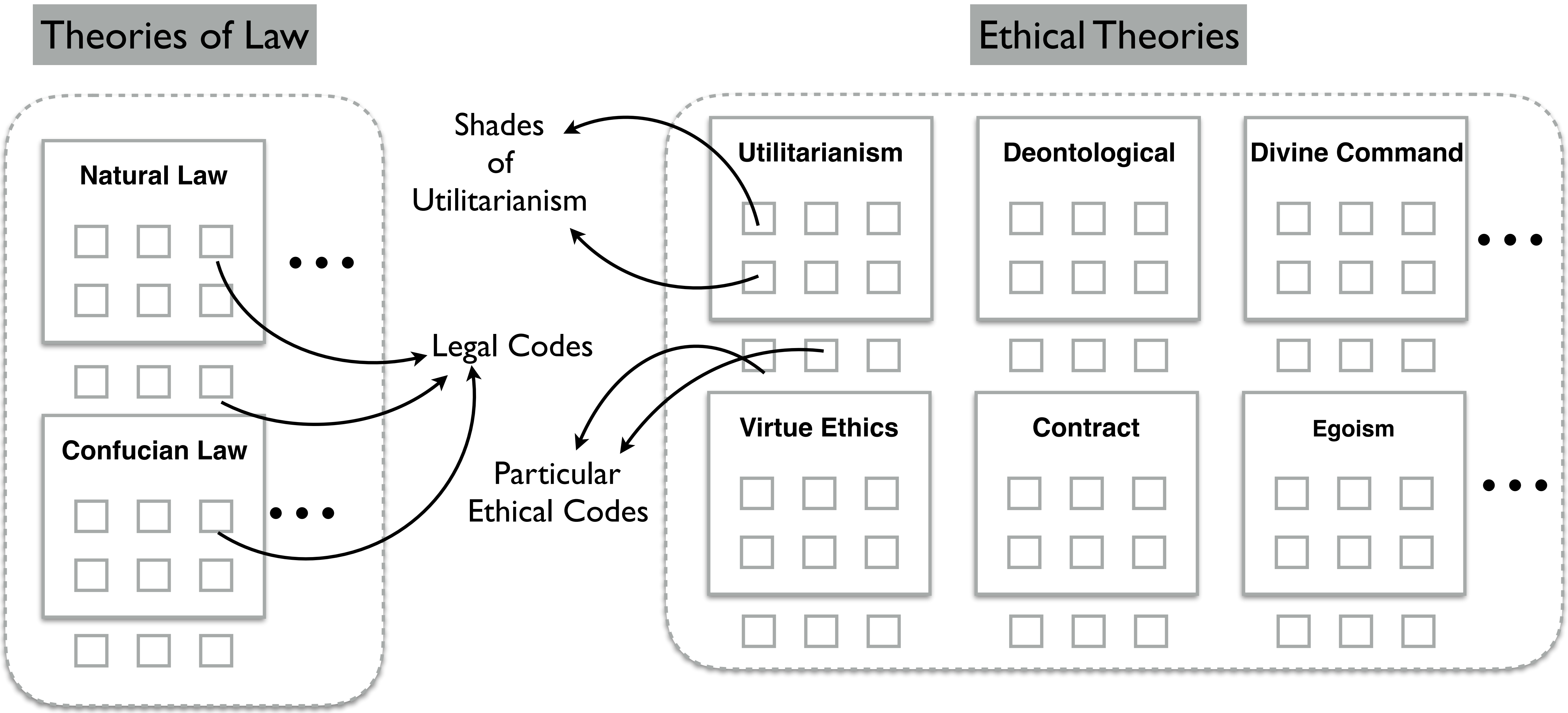
Landru Kills Himself Because Kirk/Spock Argue He Has Violated the Prime Directive for Good by Denying Creativity to Others

Logic Thwarts Nomad!

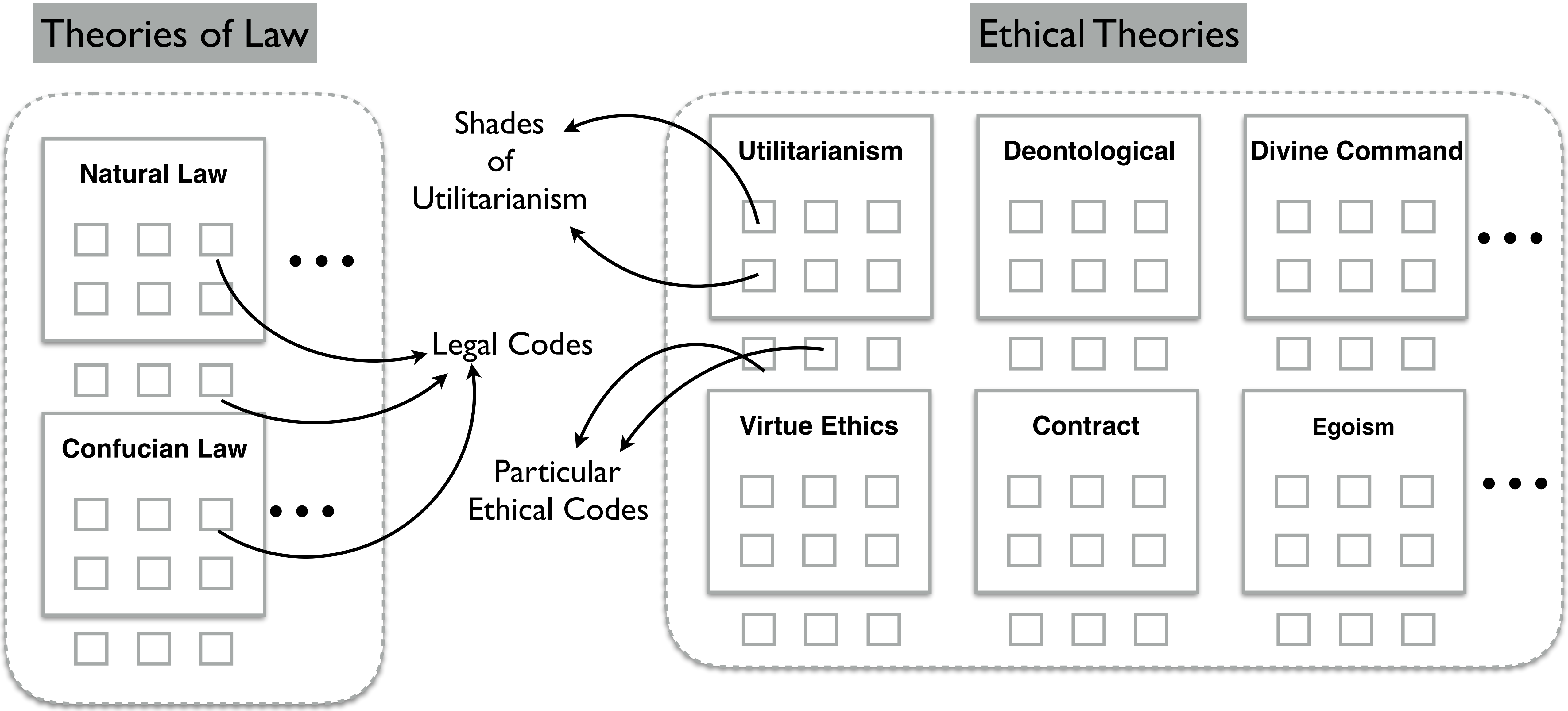
(with the Liar Paradox)

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Utilitarianism

Deontological

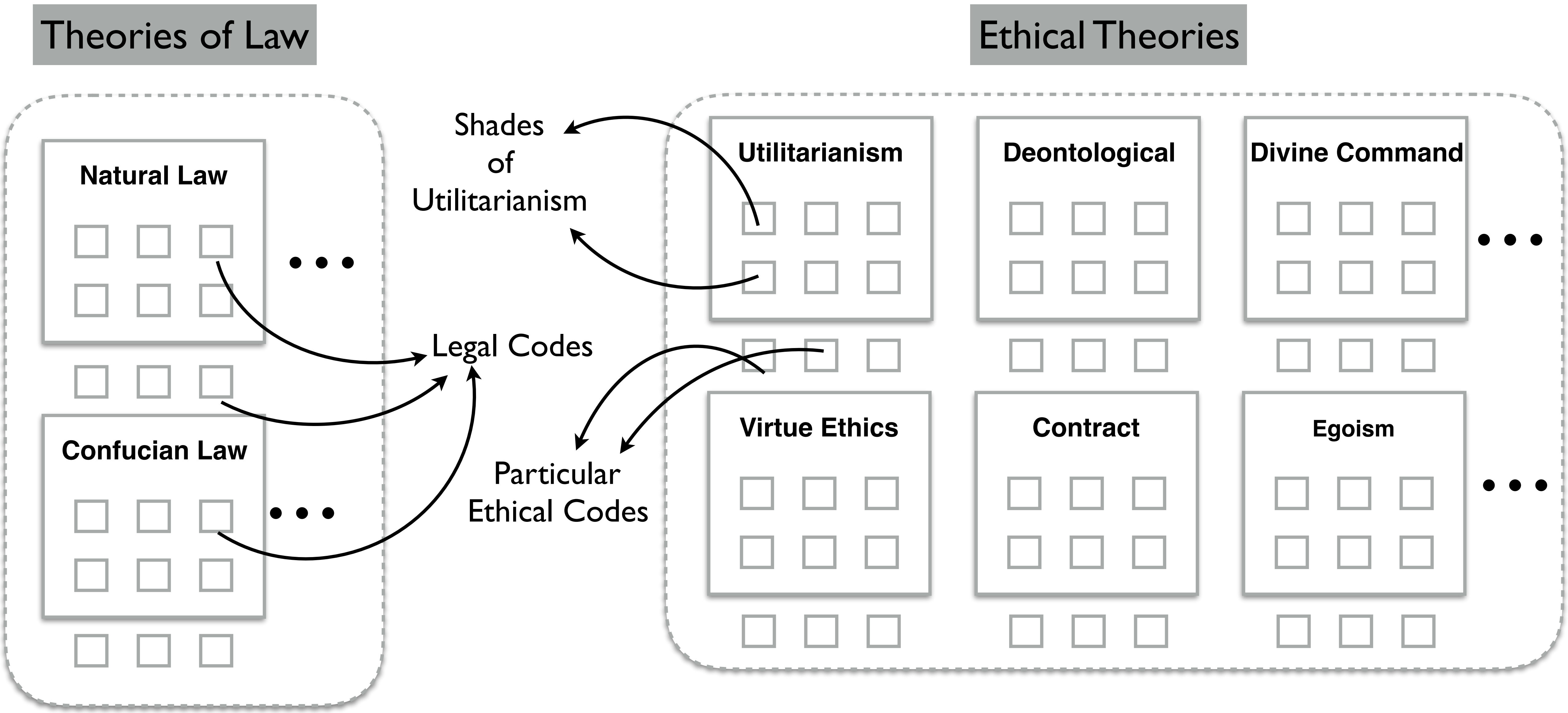
Divine Command

Virtue Ethics

Contract

Egoism

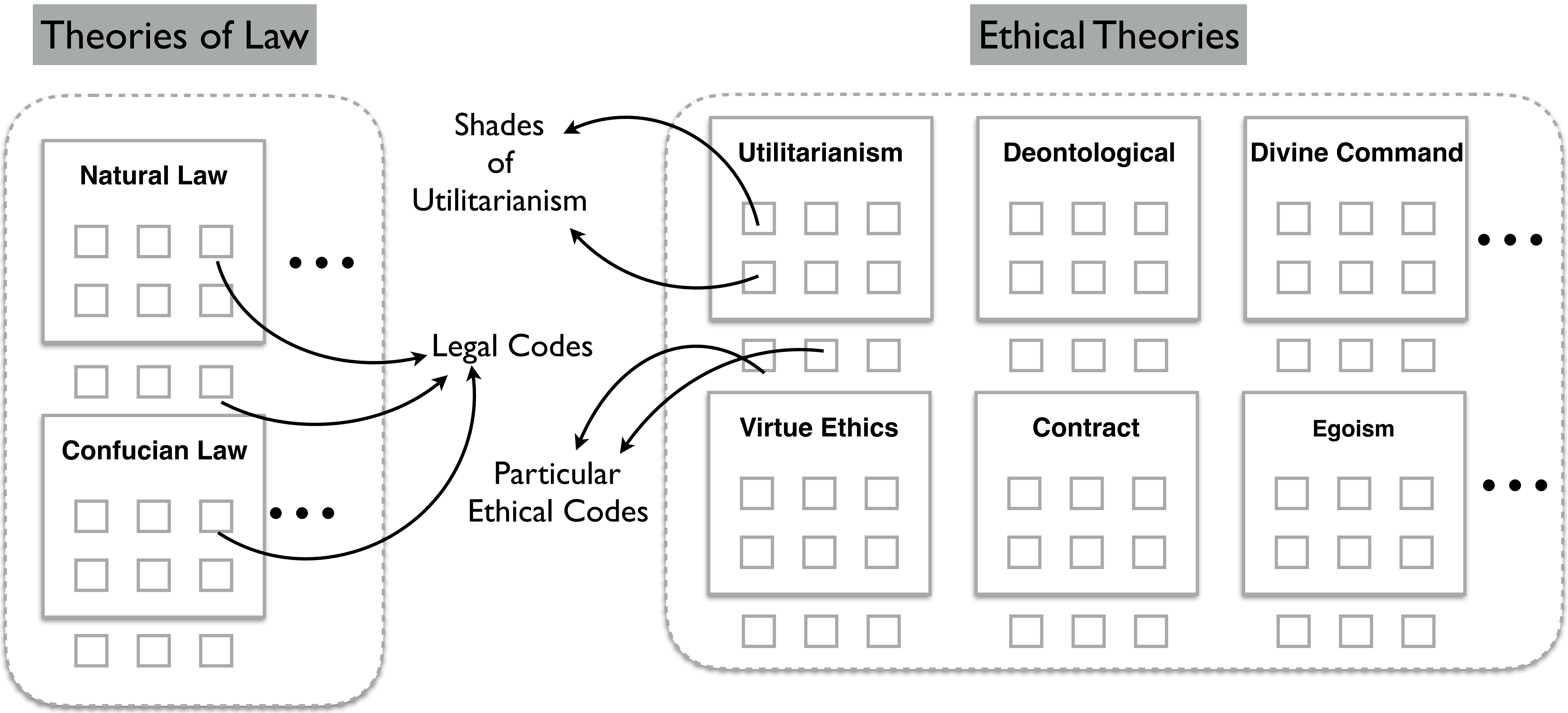
Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

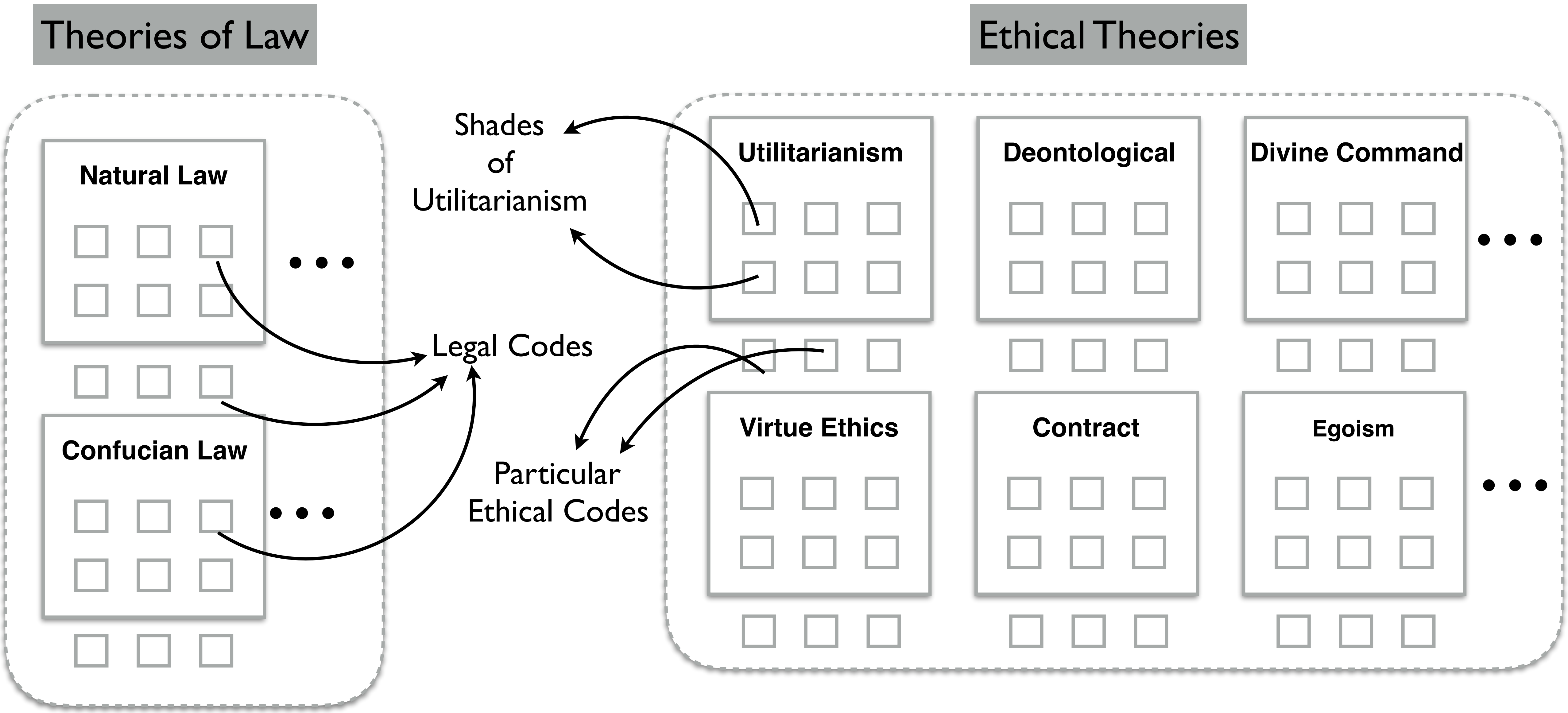
Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Natural Law

Confucian Law

Legal Codes

Shades of Utilitarianism

Particular Ethical Codes

Ethical Theories

Utilitarianism

Deontological

Divine Command

Virtue Ethics

Contract

Egoism

Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

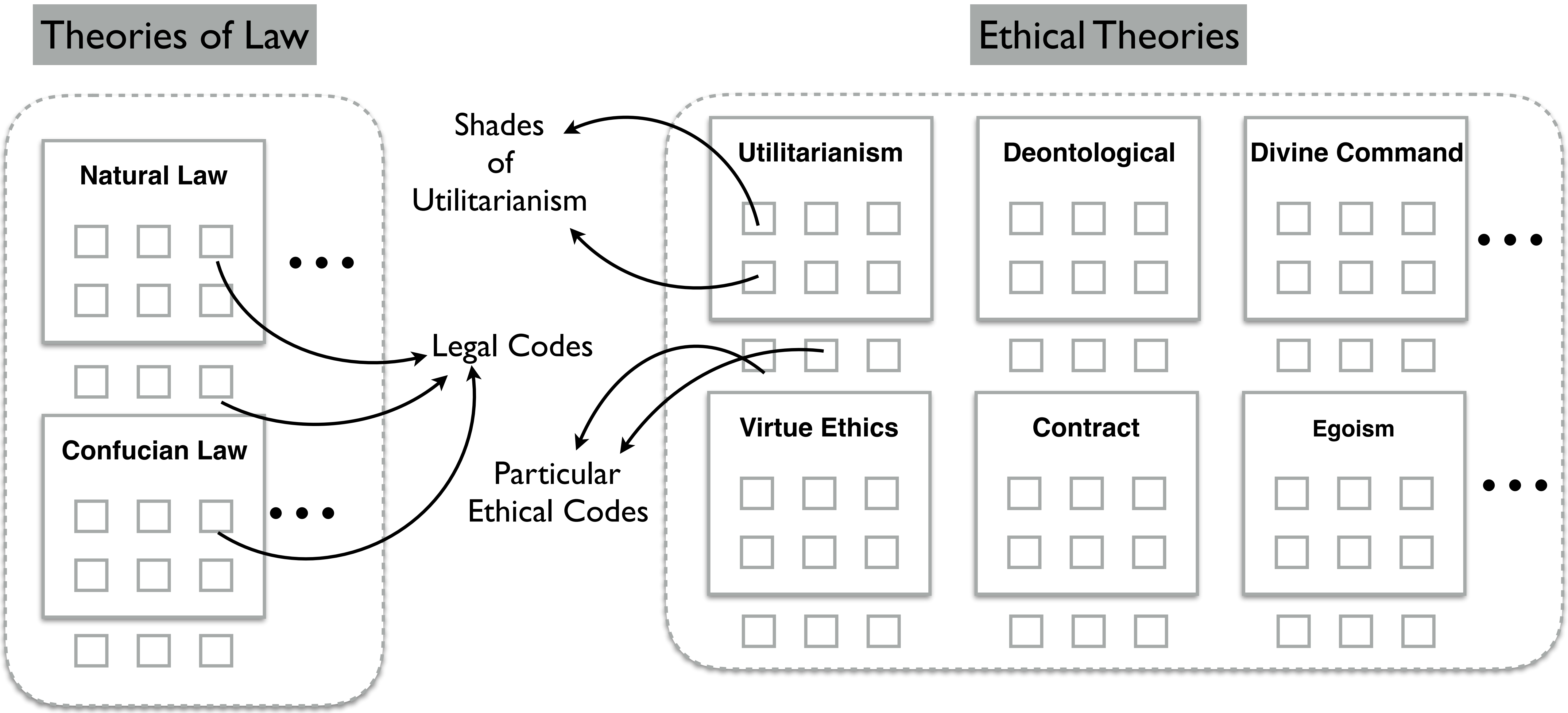
Step 2

Automate

Prover

Spectra

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Utilitarianism

Deontological

Divine Command

Virtue Ethics

Contract

Egoism

Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

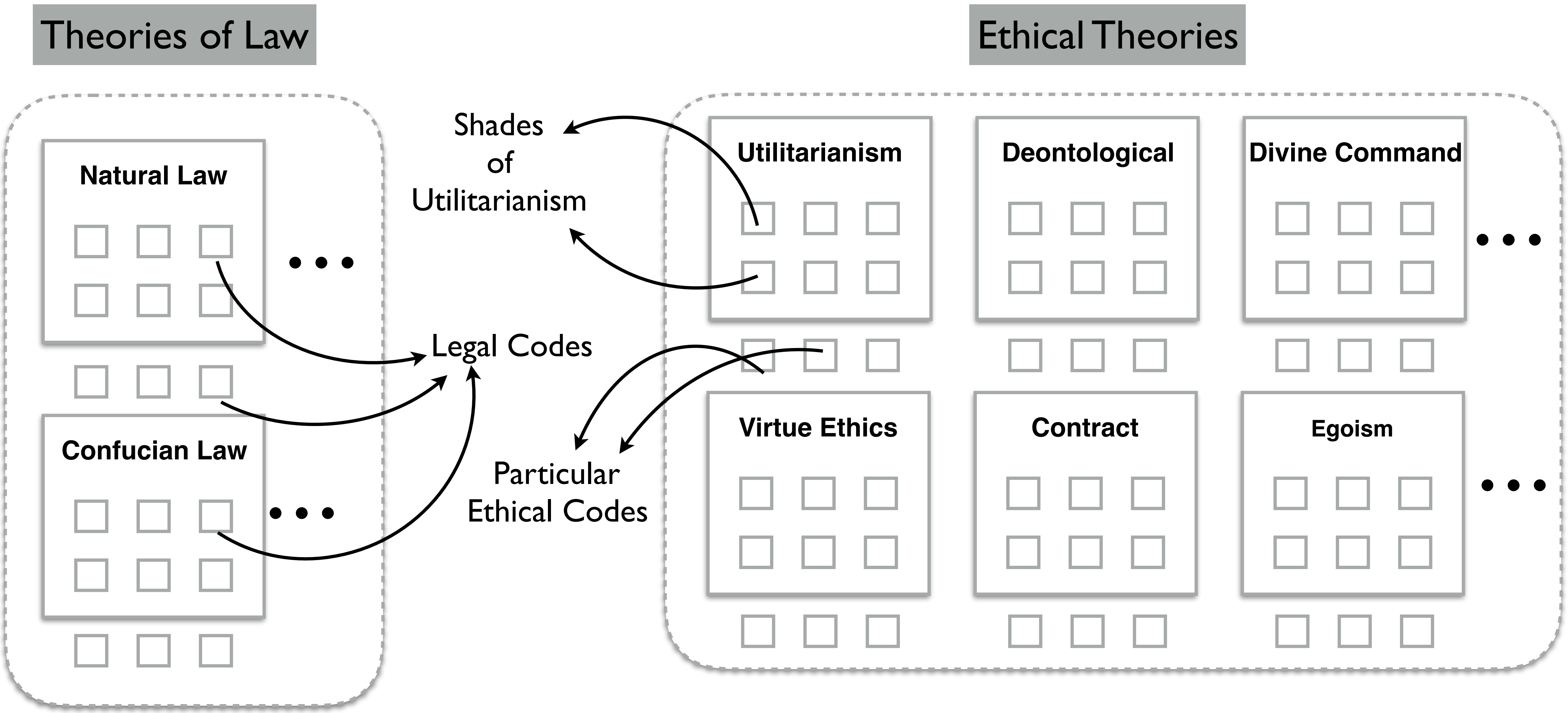
Step 2

Automate

Prover

Spectra

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Legal Codes

Shades of Utilitarianism

Particular Ethical Codes

Utilitarianism

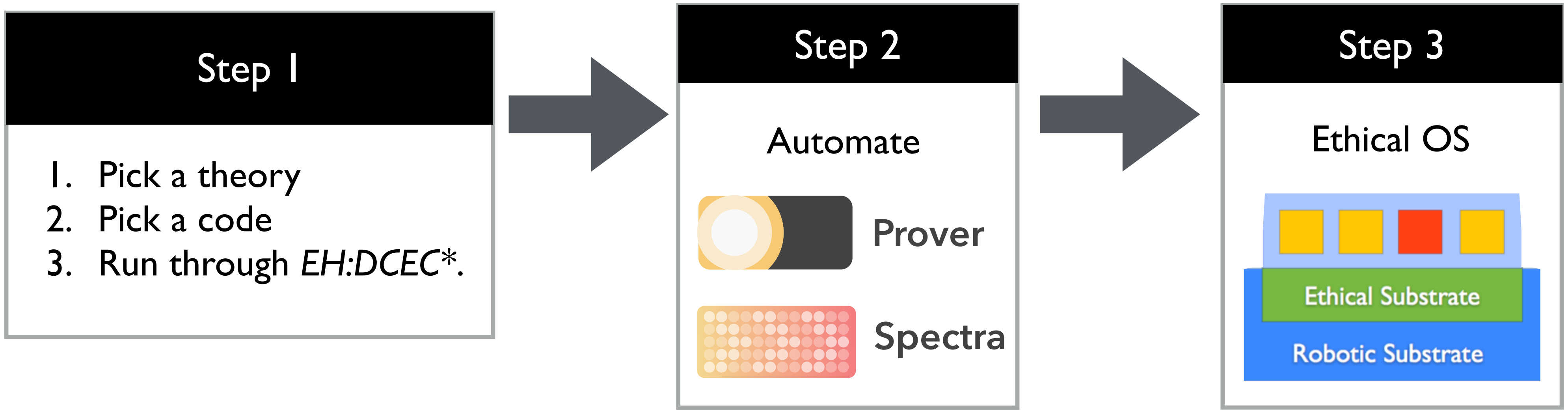
Deontological

Divine Command

Virtue Ethics

Contract

Egoism



Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

Step 2

Automate

Prover

Spectra

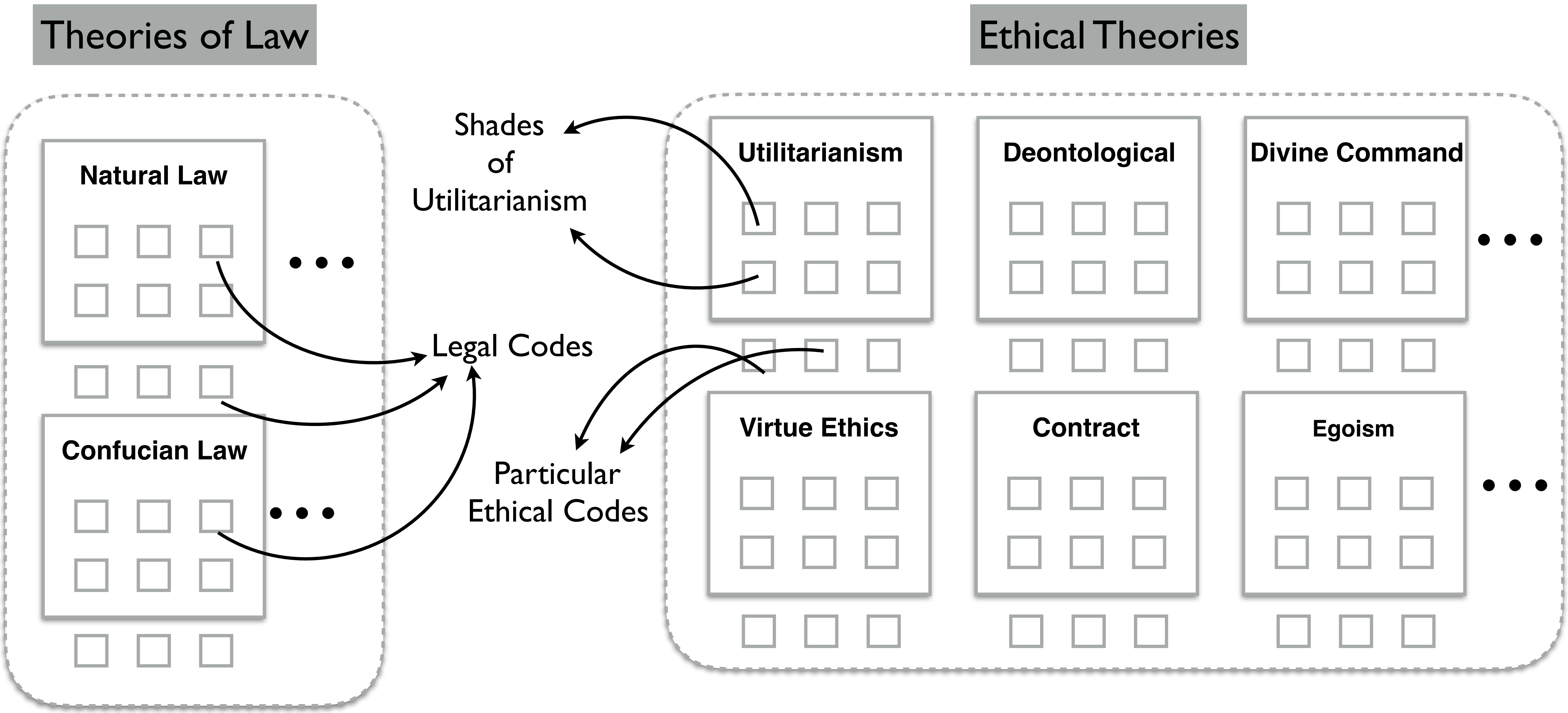
Step 3

Ethical OS

Ethical Substrate

Robotic Substrate

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Shades
of
Utilitarianism

Legal Codes

Particular
Ethical Codes

Utilitarianism

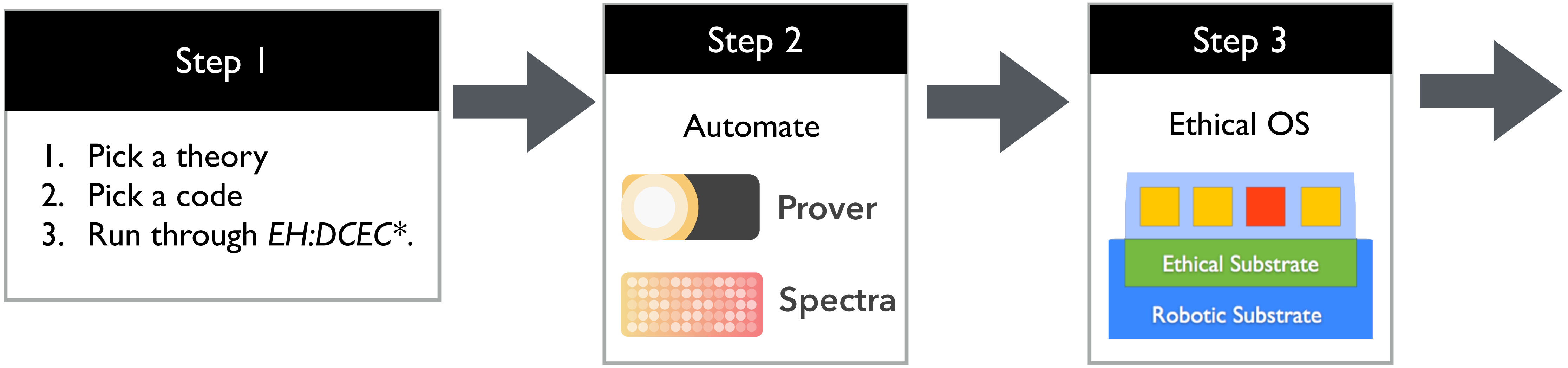
Deontological

Divine Command

Virtue Ethics

Contract

Egoism



Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

Step 2

Automate

Prover

Spectra

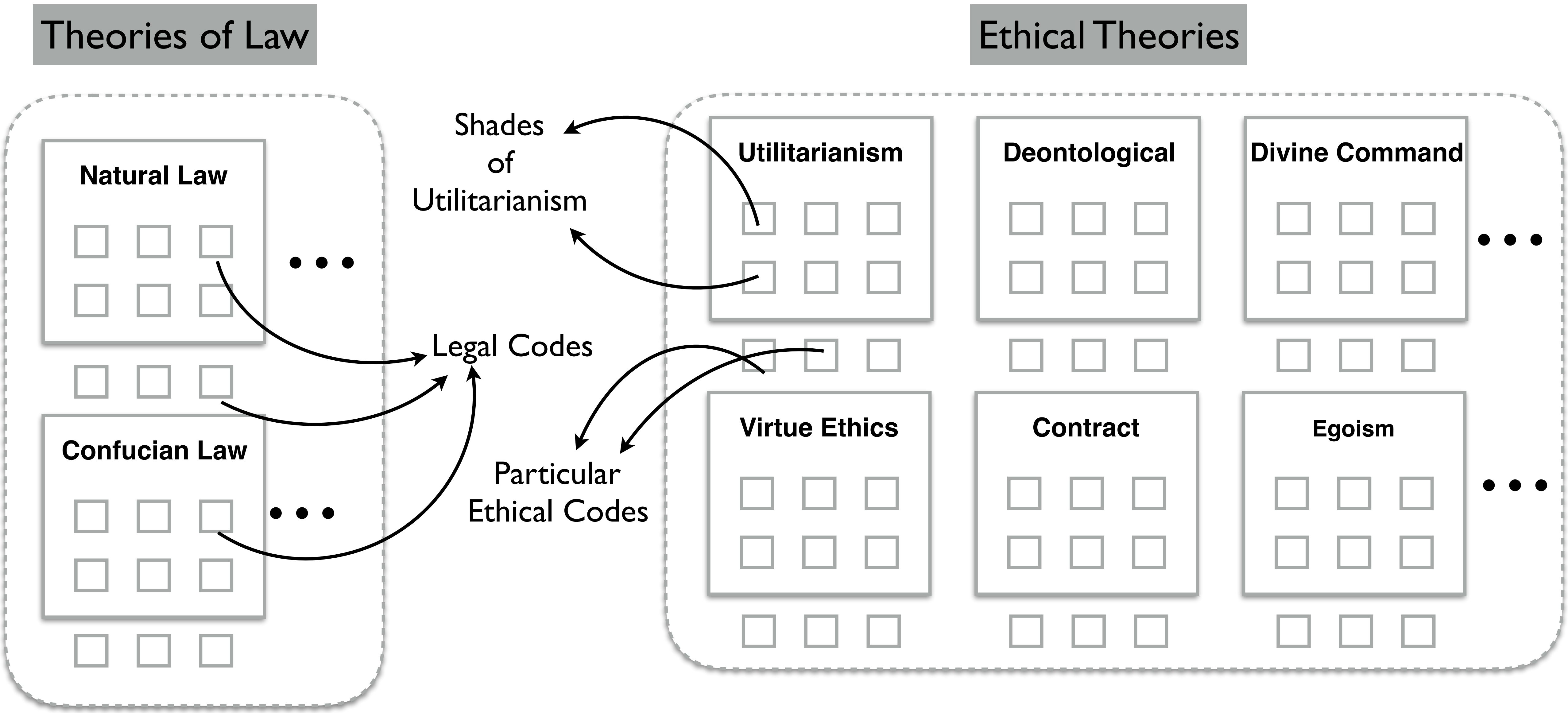
Step 3

Ethical OS

Ethical Substrate

Robotic Substrate

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Natural Law

Confucian Law

Legal Codes

Shades of Utilitarianism

Particular Ethical Codes

Ethical Theories

Utilitarianism

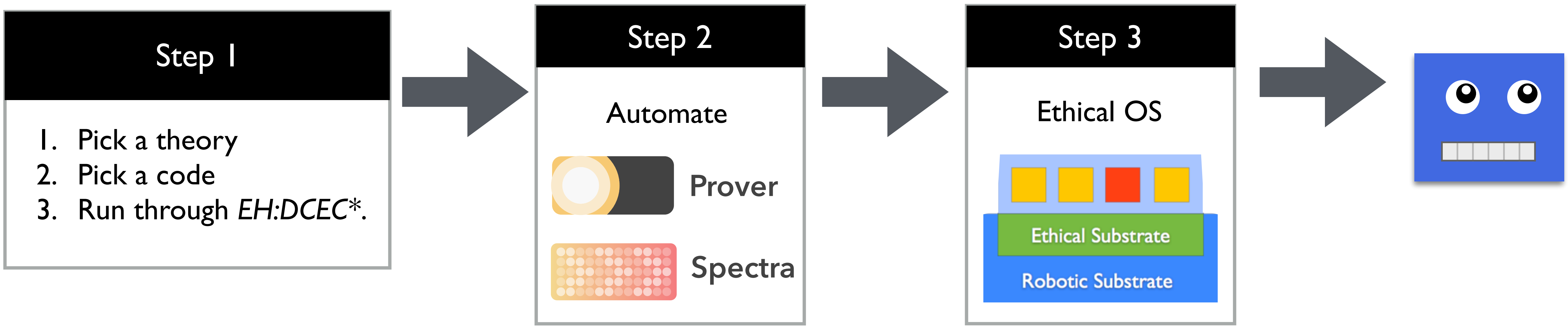
Deontological

Divine Command

Virtue Ethics

Contract

Egoism



Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

Step 2

Automate

Prover

Spectra

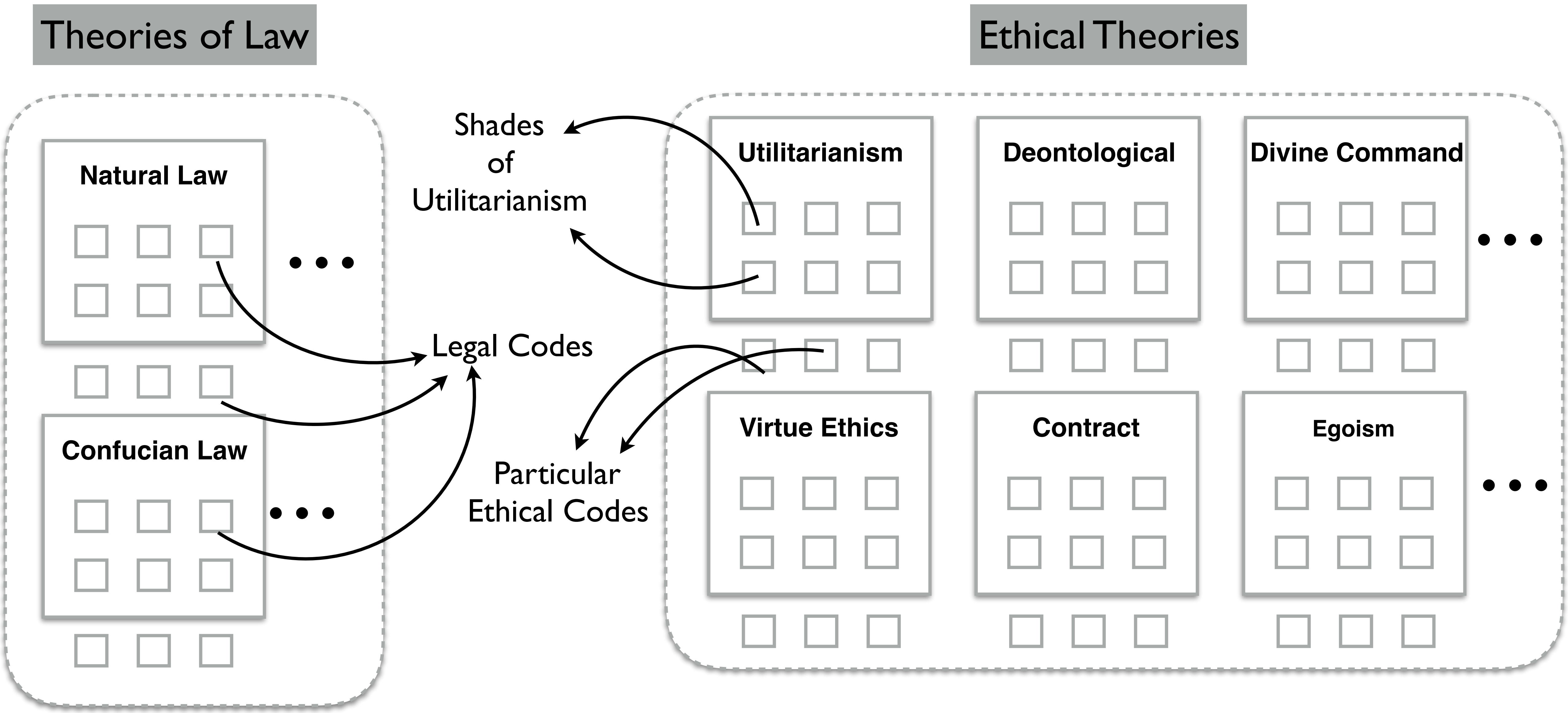
Step 3

Ethical OS

Ethical Substrate

Robotic Substrate

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Utilitarianism

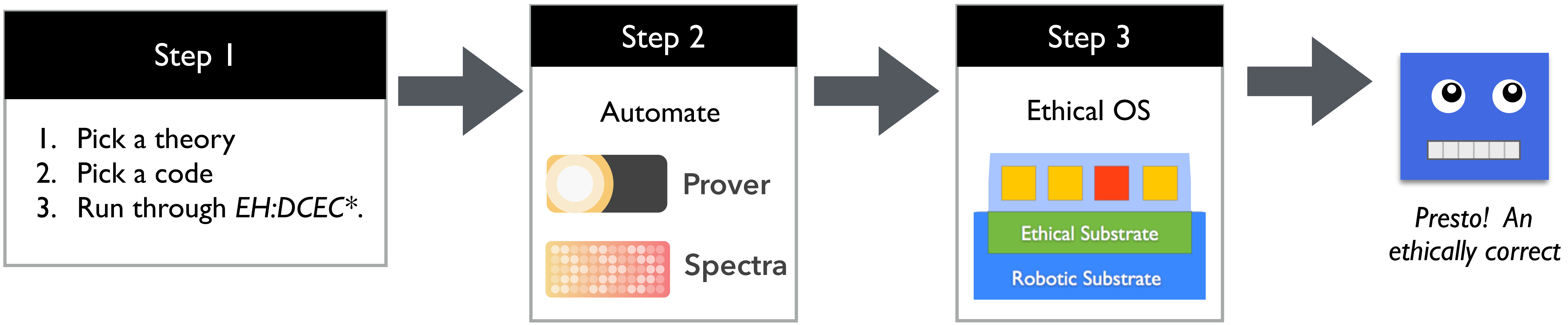
Deontological

Divine Command

Virtue Ethics

Contract

Egoism



Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

Step 2

Automate

Prover

Spectra

Step 3

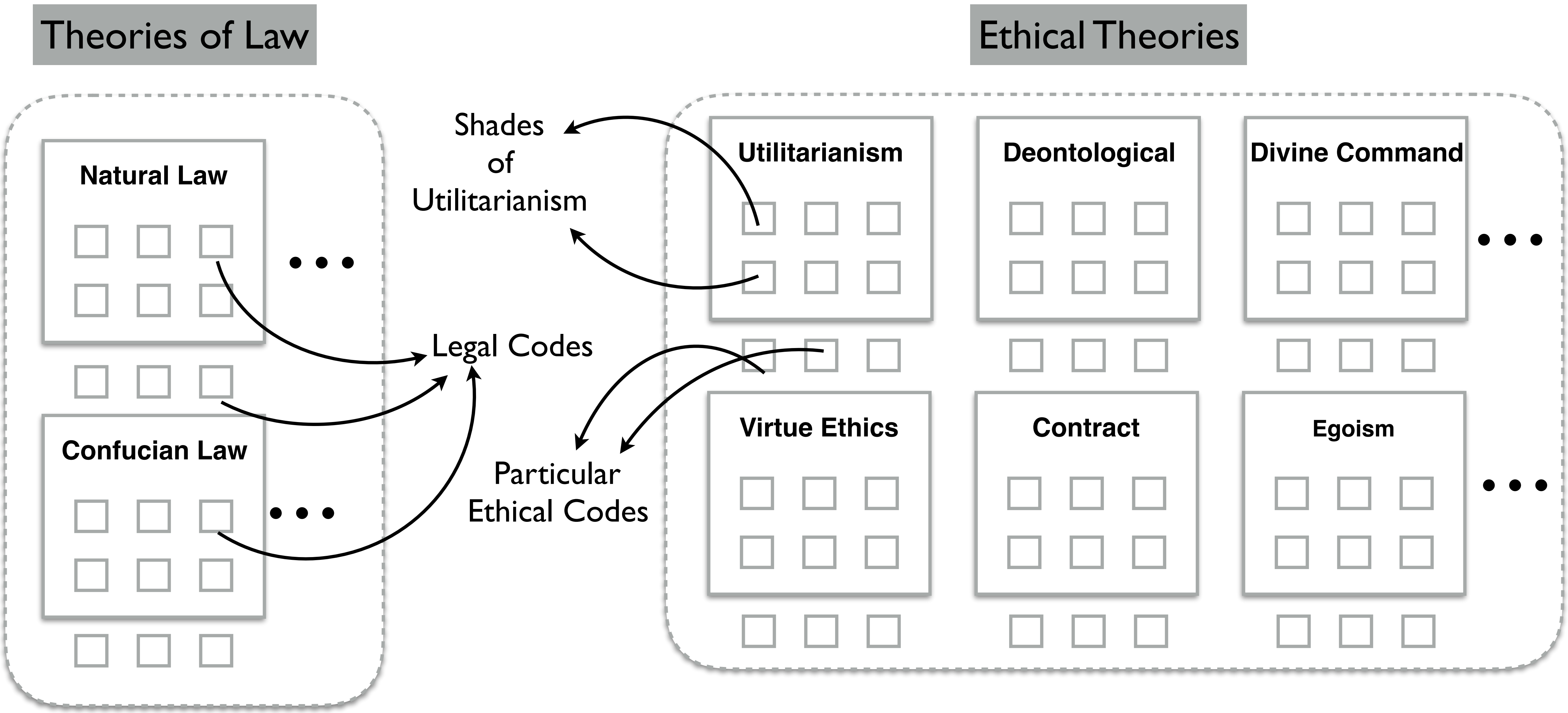
Ethical OS

Ethical Substrate

Robotic Substrate

Presto! An ethically correct

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Natural Law

• • •

Confucian Law

• • •

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Ethical Theories

Utilitarianism

Deontological

Divine Command

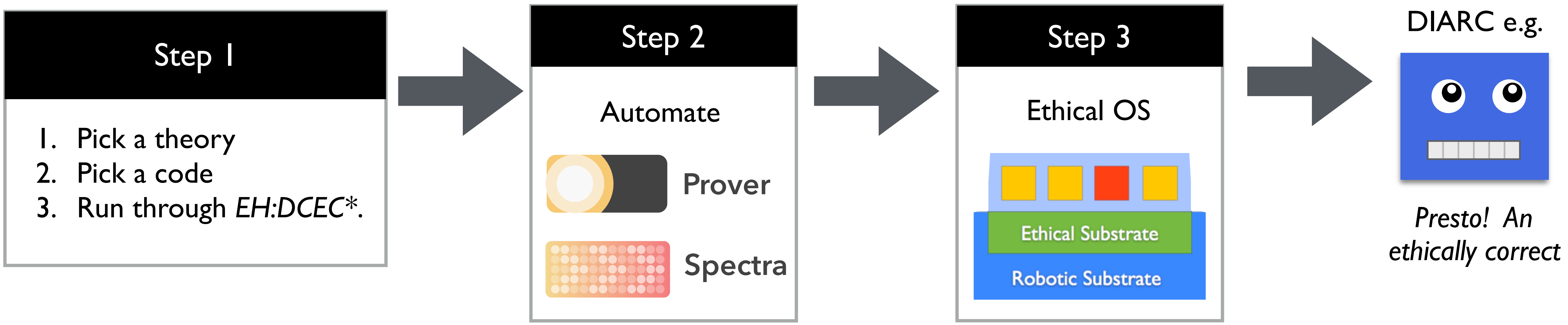
• • •

Virtue Ethics

Contract

Egoism

• • •



Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

Step 2

Automate

Prover

Spectra

Step 3

Ethical OS

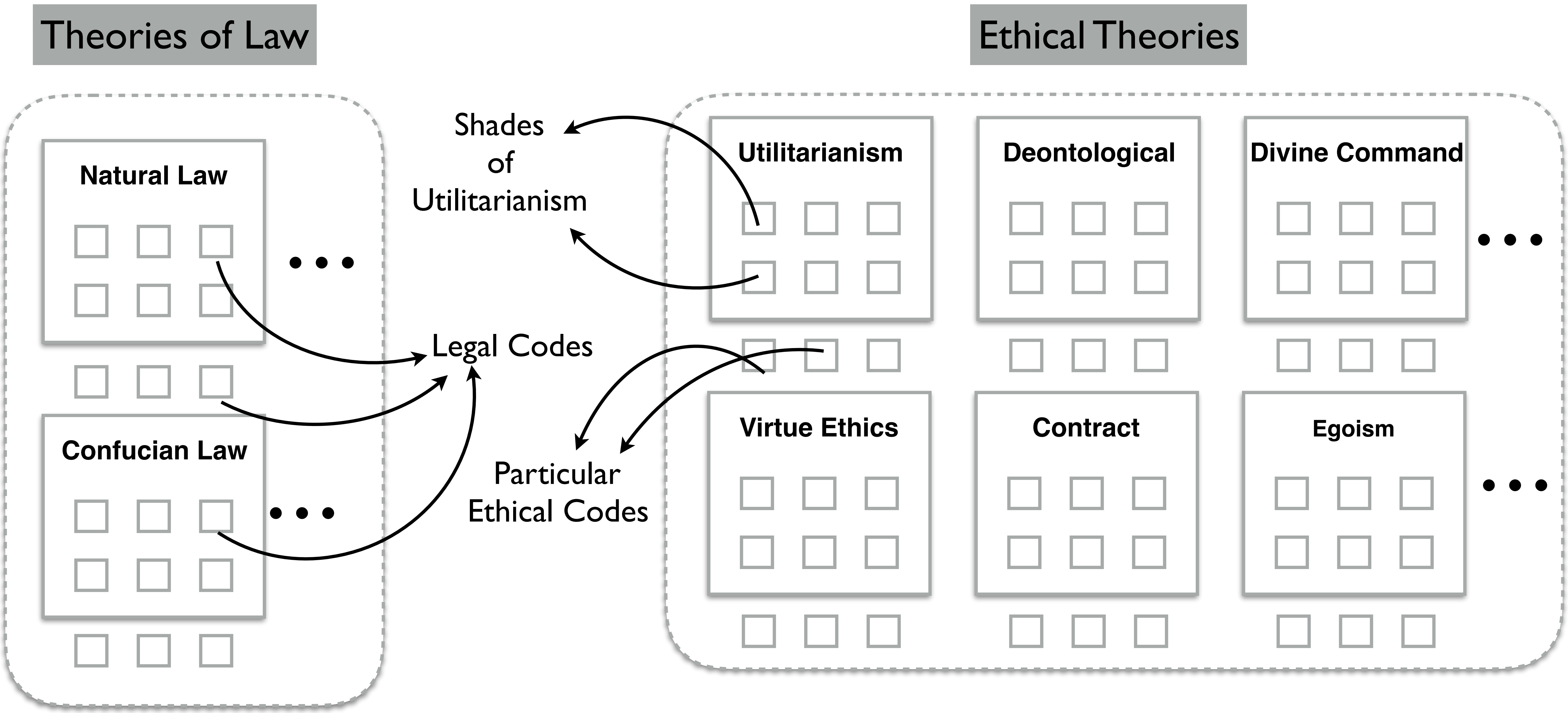
Ethical Substrate

Robotic Substrate

DIARC e.g.

Presto! An ethically correct

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Utilitarianism

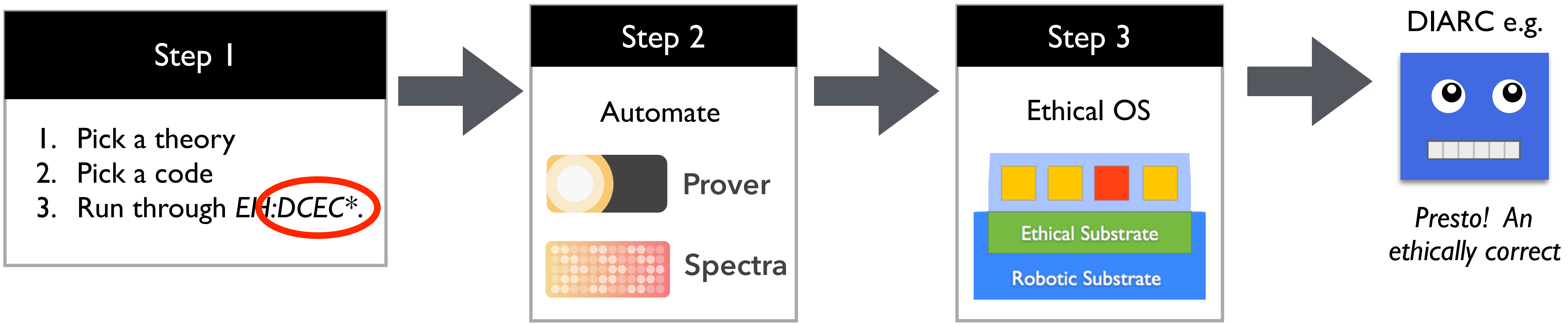
Deontological

Divine Command

Virtue Ethics

Contract

Egoism



Step 1

1. Pick a theory
2. Pick a code
3. Run through EH:DCEC*.

Step 2

Automate

Prover

Spectra

Step 3

Ethical OS

Ethical Substrate

Robotic Substrate

DIARC e.g.

Presto! An ethically correct

●

1716

Leibniz

●

1716

Leibniz

1.5 centuries < Boole!

2.5 centuries < Kripke

vindicated by Robinson 2.5 centuries later

“Universal Cognitive Calculus”

●

1716

Leibniz

1.5 centuries < Boole!

2.5 centuries < Kripke

vindicated by Robinson 2.5 centuries later

"Universal
Cognitive
Calculus"

1716

Leibniz

1.5 centuries < Boole!

2.5 centuries < Kripke

vindicated by Robinson 2.5 centuries later

"Universal
Cognitive
Calculus"

1716

2016

Leibniz

1.5 centuries < Boole!

2.5 centuries < Kripke

vindicated by Robinson 2.5 centuries later

“Universal Cognitive Calculus”

1716

2016

Leibniz

1.5 centuries < Boole!

2.5 centuries < Kripke

vindicated by Robinson 2.5 centuries later

“Universal
Cognitive
Calculus”

IGNIS
SVMMA
REMISSA
CALIDITAS
SICCITAS
TERRA
CONTRA
RIA
AER
FRIGIDITAS
HVMIDITAS
AQVA

Universal
Cognitive
Calculus
Found

1716

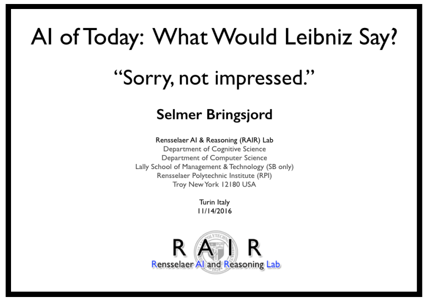
Leibniz

1.5 centuries < Boole!

2.5 centuries < Kripke

vindicated by Robinson 2.5 centuries later

2016



AI of Today: What Would Leibniz Say?

“Sorry, not impressed.”

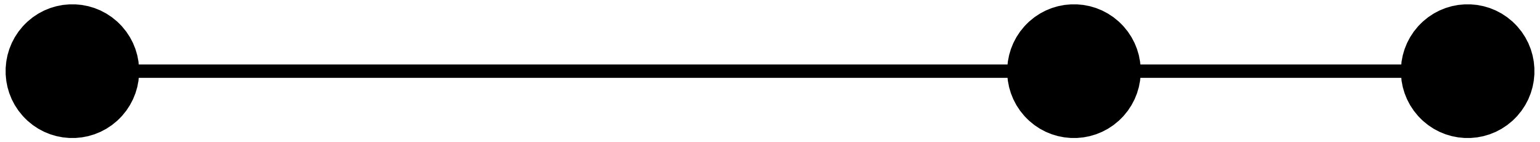
Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

R A I R
Rensselaer AI and Reasoning Lab

“Universal Cognitive Calculus”



1716

Leibniz

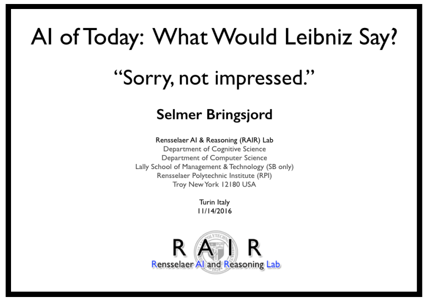
1.5 centuries < Boole!

2.5 centuries < Kripke

vindicated by Robinson 2.5 centuries later

Universal Cognitive Calculus *Found*

2016



AI of Today: What Would Leibniz Say?

“Sorry, not impressed.”

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

R A I R
Rensselaer AI and Reasoning Lab

2018

“Universal Cognitive Calculus”

1716

Leibniz

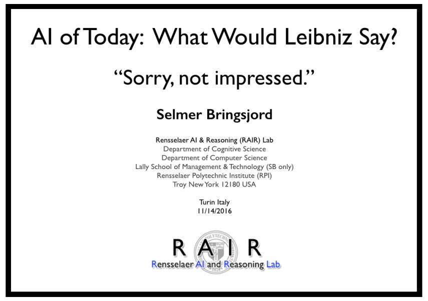
1.5 centuries < Boole!

2.5 centuries < Kripke

vindicated by Robinson 2.5 centuries later

Universal Cognitive Calculus *Found*

2016



AI of Today: What Would Leibniz Say?

“Sorry, not impressed.”

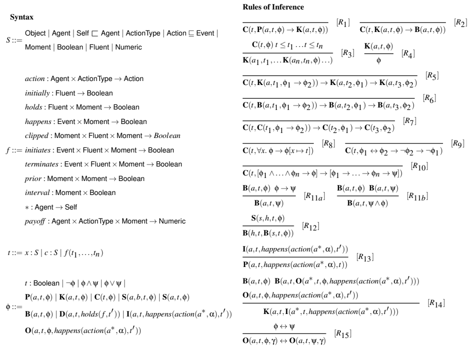
Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

RAIR
Rensselaer AI and Reasoning Lab

$\mathcal{DCEC}^*$



Syntax

$S ::=$ Object | Agent | Self $\sqsubset$ Agent | ActionType | Action $\sqsubseteq$ Event | Moment | Boolean | Fluent | Numeric

$f ::=$
action : Agent × ActionType → Action
initially : Fluent → Boolean
holds : Fluent × Moment → Boolean
happens : Event × Moment → Boolean
clipped : Moment × Fluent × Moment → *Boolean*
initiates : Event × Fluent × Moment → Boolean
terminates : Event × Fluent × Moment → Boolean
prior : Moment × Moment → Boolean
interval : Moment × Boolean
$*$: Agent → Self
payoff : Agent × ActionType × Moment → Numeric

$t ::= x : S \mid c : S \mid f(t_1, \ldots, t_n)$

$$\phi ::= \begin{array}{l} t : \text{Boolean} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid \\ \mathbf{P}(a,t,\phi) \mid \mathbf{K}(a,t,\phi) \mid \mathbf{C}(t,\phi) \mid \mathbf{S}(a,b,t,\phi) \mid \mathbf{S}(a,t,\phi) \\ \mathbf{B}(a,t,\phi) \mid \mathbf{D}(a,t,holds(f,t')) \mid \mathbf{I}(a,t,happens(action(a^*,\alpha),t')) \\ \mathbf{O}(a,t,\phi,happens(action(a^*,\alpha),t')) \end{array}$$

Rules of Inference

$$\frac{}{\mathbf{C}(t,\mathbf{P}(a,t,\phi)\rightarrow\mathbf{K}(a,t,\phi))}\ [R_1] \qquad \frac{}{\mathbf{C}(t,\mathbf{K}(a,t,\phi)\rightarrow\mathbf{B}(a,t,\phi))}\ [R_2]$$

$$\frac{\mathbf{C}(t,\phi)\ t \leq t_1 \ldots t \leq t_n}{\mathbf{K}(a_1,t_1,\ldots\mathbf{K}(a_n,t_n,\phi)\ldots)}\ [R_3] \qquad \frac{\mathbf{K}(a,t,\phi)}{\phi}\ [R_4]$$

$$\frac{}{\mathbf{C}(t,\mathbf{K}(a,t_1,\phi_1\rightarrow\phi_2))\rightarrow\mathbf{K}(a,t_2,\phi_1)\rightarrow\mathbf{K}(a,t_3,\phi_2)}\ [R_5]$$

$$\frac{}{\mathbf{C}(t,\mathbf{B}(a,t_1,\phi_1\rightarrow\phi_2))\rightarrow\mathbf{B}(a,t_2,\phi_1)\rightarrow\mathbf{B}(a,t_3,\phi_2)}\ [R_6]$$

$$\frac{}{\mathbf{C}(t,\mathbf{C}(t_1,\phi_1\rightarrow\phi_2))\rightarrow\mathbf{C}(t_2,\phi_1)\rightarrow\mathbf{C}(t_3,\phi_2)}\ [R_7]$$

$$\frac{}{\mathbf{C}(t,\forall x.\ \phi\rightarrow\phi[x\mapsto t])}\ [R_8] \qquad \frac{}{\mathbf{C}(t,\phi_1\leftrightarrow\phi_2\rightarrow\neg\phi_2\rightarrow\neg\phi_1)}\ [R_9]$$

$$\frac{}{\mathbf{C}(t,[\phi_1\wedge\ldots\wedge\phi_n\rightarrow\phi]\rightarrow[\phi_1\rightarrow\ldots\rightarrow\phi_n\rightarrow\psi])}\ [R_{10}]$$

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

$$\frac{\mathbf{S}(s,h,t,\phi)}{\mathbf{B}(h,t,\mathbf{B}(s,t,\phi))}\ [R_{12}]$$

$$\frac{\mathbf{I}(a,t,happens(action(a^*,\alpha),t'))}{\mathbf{P}(a,t,happens(action(a^*,\alpha),t))}\ [R_{13}]$$

$$\frac{\begin{array}{l}\mathbf{B}(a,t,\phi)\ \ \mathbf{B}(a,t,\mathbf{O}(a^*,t,\phi,happens(action(a^*,\alpha),t')))\\ \mathbf{O}(a,t,\phi,happens(action(a^*,\alpha),t'))\end{array}}{\mathbf{K}(a,t,\mathbf{I}(a^*,t,happens(action(a^*,\alpha),t')))}\ [R_{14}]$$

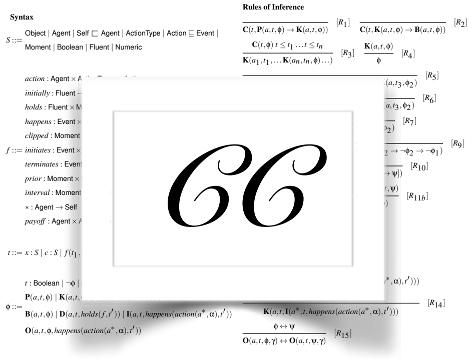
$$\frac{\phi\leftrightarrow\psi}{\mathbf{O}(a,t,\phi,\gamma)\leftrightarrow\mathbf{O}(a,t,\psi,\gamma)}\ [R_{15}]$$

2018

“Universal Cognitive Calculus”

Universal Cognitive Calculus *Found*

$\mathcal{DCEC}^*$



Syntax

$S ::=$ Object | Agent | Self $\sqsubset$ Agent | ActionType | Action $\sqsubseteq$ Event | Moment | Boolean | Fluent | Numeric

$f ::=$
action : Agent × A
initially : Fluent
holds : Fluent × M
happens : Event ×
clipped : Moment
initiates : Event ×
terminates : Event
prior : Moment ×
interval : Moment
$*$: Agent → Self
payoff : Agent × A

$t ::= x : S \mid c : S \mid f(t_1,$

$\phi ::=$
t : Boolean | $\neg\phi$ |
$\mathbf{P}(a,t,\phi) \mid \mathbf{K}(a,t,$
$\mathbf{B}(a,t,\phi) \mid \mathbf{D}(a,t,holds(f,t')) \mid \mathbf{I}(a,t,happens(action(a^*,\alpha),t'))$
$\mathbf{O}(a,t,\phi,happens(action(a^*,\alpha),t'))$

Rules of Inference

$\dfrac{}{\mathbf{C}(t,\mathbf{P}(a,t,\phi)\rightarrow\mathbf{K}(a,t,\phi))}$ [R_1] $\dfrac{}{\mathbf{C}(t,\mathbf{K}(a,t,\phi)\rightarrow\mathbf{B}(a,t,\phi))}$ [R_2]

$\dfrac{\mathbf{C}(t,\phi)\ t \leq t_1 \ldots t \leq t_n}{\mathbf{K}(a_1,t_1,\ldots\mathbf{K}(a_n,t_n,\phi)\ldots)}$ [R_3] $\dfrac{\mathbf{K}(a,t,\phi)}{\phi}$ [R_4]

(a,t_3,ϕ_2) [R_5]

(a,t_3,ϕ_2) [R_6]

$\phi_2)$ [R_7]

$_2 \rightarrow \neg\phi_2 \rightarrow \neg\phi_1)$ [R_9]

$\rightarrow \psi])$ [R_{10}]

$t,\psi)$ [R_{11b}]

$^*,\alpha),t')))$

$\dfrac{}{\mathbf{K}(a,t,\mathbf{I}(a^*,t,happens(action(a^*,\alpha),t')))}$ [R_{14}]

$\dfrac{\phi \leftrightarrow \psi}{\mathbf{O}(a,t,\phi,\gamma) \leftrightarrow \mathbf{O}(a,t,\psi,\gamma)}$ [R_{15}]

$\mathcal{CC}$

1716

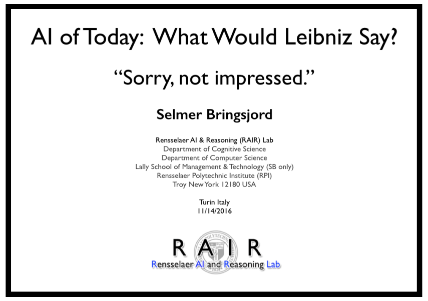
Leibniz

1.5 centuries < Boole!

2.5 centuries < Kripke

vindicated by Robinson 2.5 centuries later

2016



AI of Today: What Would Leibniz Say?

“Sorry, not impressed.”

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

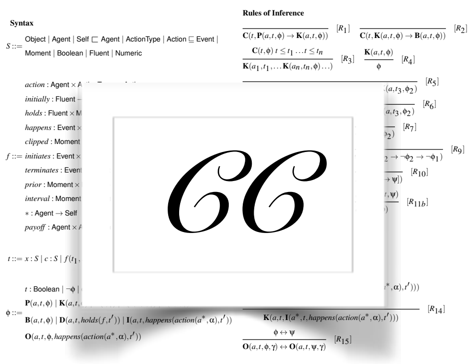
R A I R
Rensselaer AI and Reasoning Lab

2018

“Universal Cognitive Calculus”

Universal Cognitive Calculus *Found*

$\mathcal{DCEC}^*$


Syntax

$S ::=$ Object | Agent | Self ⊏ Agent | ActionType | Action ⊑ Event | Moment | Boolean | Fluent | Numeric

$f ::=$ *action* : Agent × A
initially : Fluent
holds : Fluent × M
happens : Event ×
clipped : Moment
initiates : Event ×
terminates : Event
prior : Moment ×
interval : Moment
$*$: Agent → Self
payoff : Agent × A

$t ::= x : S \mid c : S \mid f(t_1,$

$\phi ::=$ t : Boolean | $\neg\phi$ |
$\mathbf{P}(a,t,\phi) \mid \mathbf{K}(a,t,$
$\mathbf{B}(a,t,\phi) \mid \mathbf{D}(a,t,holds(f,t')) \mid \mathbf{I}(a,t,happens(action(a^*,\alpha),t'))$
$\mathbf{O}(a,t,\phi,happens(action(a^*,\alpha),t'))$

Rules of Inference

$$\frac{}{\mathbf{C}(t,\mathbf{P}(a,t,\phi)\rightarrow\mathbf{K}(a,t,\phi))}\ [R_1] \qquad \frac{}{\mathbf{C}(t,\mathbf{K}(a,t,\phi)\rightarrow\mathbf{B}(a,t,\phi))}\ [R_2]$$

$$\frac{\mathbf{C}(t,\phi)\ t\leq t_1\ldots t\leq t_n}{\mathbf{K}(a_1,t_1,\ldots\mathbf{K}(a_n,t_n,\phi)\ldots)}\ [R_3] \qquad \frac{\mathbf{K}(a,t,\phi)}{\phi}\ [R_4]$$

(a,t_3,ϕ_2) $[R_5]$
(a,t_3,ϕ_2) $[R_6]$
$\phi_2)$ $[R_7]$
$_2\rightarrow\neg\phi_2\rightarrow\neg\phi_1)$ $[R_9]$
$\rightarrow\psi])$ $[R_{10}]$
$t,\psi)$ $[R_{11b}]$
$^*,\alpha),t')))$

$$\frac{}{\mathbf{K}(a,t,\mathbf{I}(a^*,t,happens(action(a^*,\alpha),t')))}\ [R_{14}]$$

$$\frac{\phi\leftrightarrow\psi}{\mathbf{O}(a,t,\phi,\gamma)\leftrightarrow\mathbf{O}(a,t,\psi,\gamma)}\ [R_{15}]$$

1716

2016

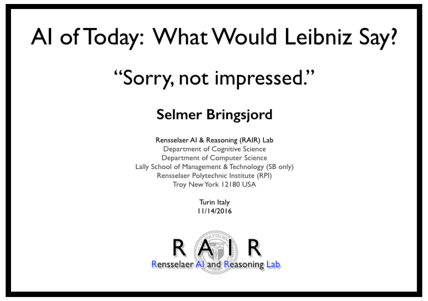
2018

Leibniz

1.5 centuries < Boole!

2.5 centuries < Kripke

vindicated by Robinson 2.5 centuries later


AI of Today: What Would Leibniz Say?

“Sorry, not impressed.”

Selmer Bringsjord

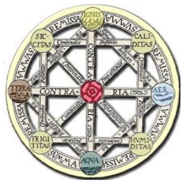
Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

R A I R
Rensselaer AI and Reasoning Lab

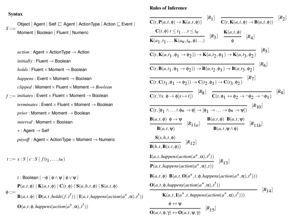
R A I R
Rensselaer AI and Reasoning Lab

"Universal Cognitive Calculus"



Universal Cognitive Calculus *Found*

$\mathcal{DCEC}^*$



Syntax

$$S ::= \text{Object} \mid \text{Agent} \mid \text{Self} \sqsubset \text{Agent} \mid \text{ActionType} \mid \text{Action} \sqsubset \text{Event} \mid \text{Moment} \mid \text{Boolean} \mid \text{Fluent} \mid \text{Numeric}$$

$$f ::= \begin{array}{l} \mathit{action} : \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \\ \mathit{initially} : \text{Fluent} \rightarrow \text{Boolean} \\ \mathit{holds} : \text{Fluent} \times \text{Moment} \rightarrow \text{Boolean} \\ \mathit{happens} : \text{Event} \times \text{Moment} \rightarrow \text{Boolean} \\ \mathit{clipped} : \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \mathit{Boolean} \\ \mathit{initiates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Boolean} \\ \mathit{terminates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Boolean} \\ \mathit{prior} : \text{Moment} \times \text{Moment} \rightarrow \text{Boolean} \\ \mathit{interval} : \text{Moment} \times \text{Boolean} \\ * : \text{Agent} \rightarrow \text{Self} \\ \mathit{payoff} : \text{Agent} \times \text{ActionType} \times \text{Moment} \rightarrow \text{Numeric} \end{array}$$

$$t ::= x : S \mid c : S \mid f(t_1, \ldots, t_n)$$

$$\phi ::= \begin{array}{l} t : \text{Boolean} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid \\ \mathbf{P}(a,t,\phi) \mid \mathbf{K}(a,t,\phi) \mid \mathbf{C}(t,\phi) \mid \mathbf{S}(a,b,t,\phi) \mid \mathbf{S}(a,t,\phi) \\ \mathbf{B}(a,t,\phi) \mid \mathbf{D}(a,t,\mathit{holds}(f,t')) \mid \mathbf{I}(a,t,\mathit{happens}(\mathit{action}(a^*,\alpha),t')) \\ \mathbf{O}(a,t,\phi,\mathit{happens}(\mathit{action}(a^*,\alpha),t')) \end{array}$$

Rules of Inference

$$\frac{}{\mathbf{C}(t, \mathbf{P}(a,t,\phi) \rightarrow \mathbf{K}(a,t,\phi))}\ [R_1] \qquad \frac{}{\mathbf{C}(t, \mathbf{K}(a,t,\phi) \rightarrow \mathbf{B}(a,t,\phi))}\ [R_2]$$

$$\frac{\mathbf{C}(t,\phi)\ t \leq t_1 \ldots t \leq t_n}{\mathbf{K}(a_1,t_1,\ldots \mathbf{K}(a_n,t_n,\phi)\ldots)}\ [R_3] \qquad \frac{\mathbf{K}(a,t,\phi)}{\phi}\ [R_4]$$

$$\frac{}{\mathbf{C}(t, \mathbf{K}(a,t_1,\phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{K}(a,t_2,\phi_1) \rightarrow \mathbf{K}(a,t_3,\phi_2)}\ [R_5]$$

$$\frac{}{\mathbf{C}(t, \mathbf{B}(a,t_1,\phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{B}(a,t_2,\phi_1) \rightarrow \mathbf{B}(a,t_3,\phi_2)}\ [R_6]$$

$$\frac{}{\mathbf{C}(t, \mathbf{C}(t_1,\phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{C}(t_2,\phi_1) \rightarrow \mathbf{C}(t_3,\phi_2)}\ [R_7]$$

$$\frac{}{\mathbf{C}(t, \forall x.\ \phi \rightarrow \phi[x \mapsto t])}\ [R_8] \qquad \frac{}{\mathbf{C}(t, \phi_1 \leftrightarrow \phi_2 \rightarrow \neg\phi_2 \rightarrow \neg\phi_1)}\ [R_9]$$

$$\frac{}{\mathbf{C}(t, [\phi_1 \wedge \ldots \wedge \phi_n \rightarrow \phi] \rightarrow [\phi_1 \rightarrow \ldots \rightarrow \phi_n \rightarrow \psi])}\ [R_{10}]$$

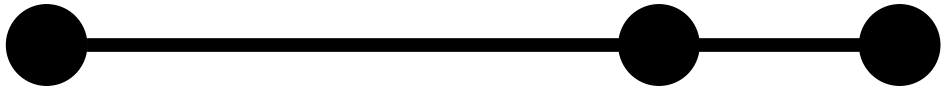
$$\frac{\mathbf{B}(a,t,\phi)\quad \phi \rightarrow \psi}{\mathbf{B}(a,t,\psi)}\ [R_{11a}] \qquad \frac{\mathbf{B}(a,t,\phi)\quad \mathbf{B}(a,t,\psi)}{\mathbf{B}(a,t,\psi \wedge \phi)}\ [R_{11b}]$$

$$\frac{\mathbf{S}(s,h,t,\phi)}{\mathbf{B}(h,t,\mathbf{B}(s,t,\phi))}\ [R_{12}]$$

$$\frac{\mathbf{I}(a,t,\mathit{happens}(\mathit{action}(a^*,\alpha),t'))}{\mathbf{P}(a,t,\mathit{happens}(\mathit{action}(a^*,\alpha),t))}\ [R_{13}]$$

$$\frac{\mathbf{B}(a,t,\phi)\quad \mathbf{B}(a,t,\mathbf{O}(a^*,t,\phi,\mathit{happens}(\mathit{action}(a^*,\alpha),t')))\quad \mathbf{O}(a,t,\phi,\mathit{happens}(\mathit{action}(a^*,\alpha),t'))}{\mathbf{K}(a,t,\mathbf{I}(a^*,t,\mathit{happens}(\mathit{action}(a^*,\alpha),t')))}\ [R_{14}]$$

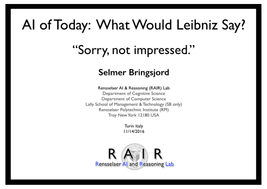
$$\frac{\phi \leftrightarrow \psi}{\mathbf{O}(a,t,\phi,\gamma) \leftrightarrow \mathbf{O}(a,t,\psi,\gamma)}\ [R_{15}]$$



17

20

20



AI of Today: What Would Leibniz Say?

"Sorry, not impressed."

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

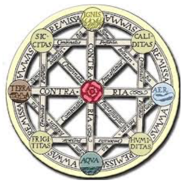
RAIR
Rensselaer AI and Reasoning Lab

RAIR
Rensselaer AI and Reasoning Lab

Lei

1.5 centuries <

“Universal Cognitive Calculus”



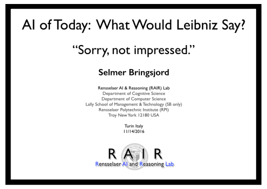
Universal

$\mathcal{CC}$
$\dfrac{}{\mathbf{C}(t, \mathbf{K}(a,t,\phi) \rightarrow \mathbf{B}(a,t,\phi))}$ $[R_2]$
$\dfrac{\mathbf{K}(a,t,\phi)}{\phi}$ $[R_4]$
$\dfrac{}{\mathbf{C}(t, \phi_1 \leftrightarrow \phi_2 \rightarrow \neg\phi_2 \rightarrow \neg\phi_1)}$ $[R_9]$
$\dfrac{\mathbf{B}(a,t,\phi) \quad \mathbf{B}(a,t,\psi)}{\mathbf{B}(a,t,\psi \wedge \phi)}$ $[R_{11b}]$
$[R_1]$ $[R_3]$ $[R_5]$ $[R_6]$ $[R_7]$ $[R_8]$ $[R_{10}]$ $[R_{13}]$ $[R_{14}]$ $[R_{15}]$

17

20

20



AI of Today: What Would Leibniz Say?

“Sorry, not impressed.”

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

RAIR
Rensselaer AI and Reasoning Lab

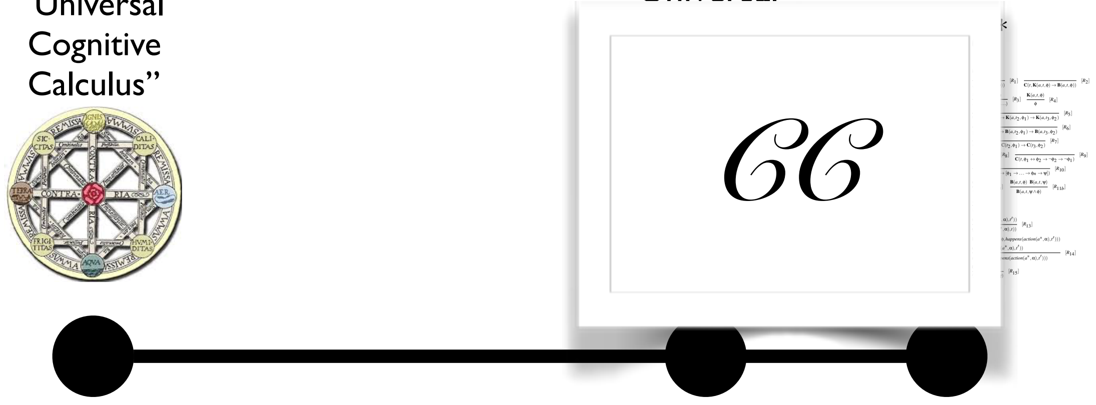
RAIR
Rensselaer AI and Reasoning Lab

Lei

1.5 centuries <

“Universal Cognitive Calculus”

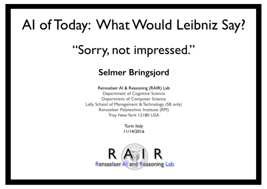
Universal



17

20

20



AI of Today: What Would Leibniz Say?

“Sorry, not impressed.”

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

RAIR
Rensselaer AI and Reasoning Lab

RAIR
Rensselaer AI and Reasoning Lab

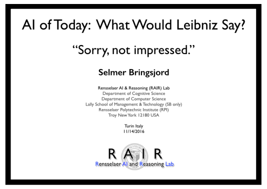
Lei

1.5 centuries <

“Universal Cognitive Calculus”

Universal

Lei



AI of Today: What Would Leibniz Say?

“Sorry, not impressed.”

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

R A I R
Rensselaer AI and Reasoning Lab

R A I R
Rensselaer AI and Reasoning Lab

1.5 centuries <

Syntax

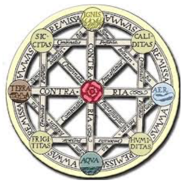
$$S ::= \text{Object} \mid \text{Agent} \mid \text{ActionType} \mid \text{Action} \sqsubseteq \text{Event} \mid \text{Moment} \mid \text{Formula} \mid \text{Fluent}$$

$$f ::= \begin{cases} \mathit{action} : \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \\ \mathit{initially} : \text{Fluent} \rightarrow \text{Formula} \\ \mathit{Holds} : \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \mathit{happens} : \text{Event} \times \text{Moment} \rightarrow \text{Formula} \\ \mathit{clipped} : \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \mathit{initiates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \mathit{terminates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \mathit{prior} : \text{Moment} \times \text{Moment} \rightarrow \text{Formula} \end{cases}$$

$$t ::= x : S \mid c : S \mid f(t_1, \ldots, t_n)$$

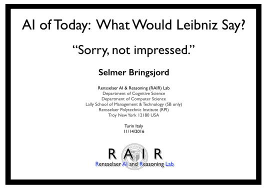
$$\phi ::= \begin{cases} t : \text{Formula} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid \mathbf{P}(a,t,\phi) \mid \mathbf{K}(a,t,\phi) \mid \mathbf{C}(t,\phi) \\ \mathbf{S}(a,b,t,\phi) \mid \mathbf{S}(a,t,\phi) \mid \mathbf{B}(a,t,\phi) \mid \mathbf{D}(a,t,\mathit{Holds}(f,t')) \mid \mathbf{I}(a,t,\phi) \\ \mathbf{O}(a,t,\phi,(\neg)\mathit{happens}(\mathit{action}(a^*,\alpha),t')) \end{cases}$$

"Universal Cognitive Calculus"



Universal

Lei



AI of Today: What Would Leibniz Say?

"Sorry, not impressed."

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

R A I R
Rensselaer AI and Reasoning Lab

R A I R
Rensselaer AI and Reasoning Lab

1.5 centuries <

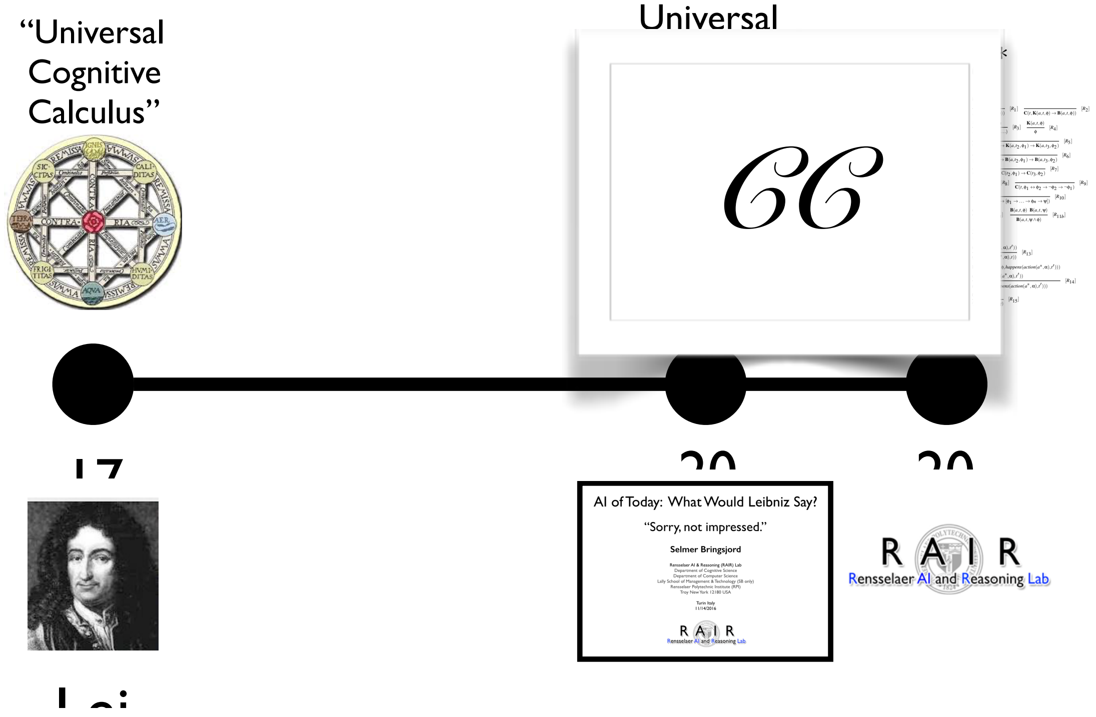
Syntax

$$S ::= \text{Object} \mid \text{Agent} \mid \text{ActionType} \mid \text{Action} \sqsubseteq \text{Event} \mid \text{Moment} \mid \text{Formula} \mid \text{Fluent}$$

$$f ::= \begin{cases} \mathit{action} : \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \\ \mathit{initially} : \text{Fluent} \rightarrow \text{Formula} \\ \mathit{Holds} : \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \mathit{happens} : \text{Event} \times \text{Moment} \rightarrow \text{Formula} \\ \mathit{clipped} : \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \mathit{initiates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \mathit{terminates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ \mathit{prior} : \text{Moment} \times \text{Moment} \rightarrow \text{Formula} \end{cases}$$

$$t ::= x : S \mid c : S \mid f(t_1, \ldots, t_n)$$

$$\phi ::= \begin{cases} t : \text{Formula} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid \mathbf{P}(a,t,\phi) \mid \mathbf{K}(a,t,\phi) \mid \mathbf{C}(t,\phi) \\ \mathbf{S}(a,b,t,\phi) \mid \mathbf{S}(a,t,\phi) \mid \mathbf{B}(a,t,\phi) \mid \mathbf{D}(a,t,\mathit{Holds}(f,t')) \mid \mathbf{I}(a,t,\phi) \\ \mathbf{O}(a,t,\phi,(\neg)\mathit{happens}(\mathit{action}(a^*,\alpha),t')) \end{cases}$$



AI of Today: What Would Leibniz Say?

"Sorry, not impressed."

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

R A I R
Rensselaer AI and Reasoning Lab

1.5 centuries <

Syntax

$$S ::= \text{Object} \mid \text{Agent} \mid \text{ActionType} \mid \text{Action} \sqsubseteq \text{Event} \mid \text{Moment} \mid \text{Formula} \mid \text{Fluent}$$

$$f ::= \begin{cases} action : \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \\ initially : \text{Fluent} \rightarrow \text{Formula} \\ Holds : \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ happens : \text{Event} \times \text{Moment} \rightarrow \text{Formula} \\ clipped : \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ initiates : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ terminates : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\ prior : \text{Moment} \times \text{Moment} \rightarrow \text{Formula} \end{cases}$$

$$t ::= x : S \mid c : S \mid f(t_1, \ldots, t_n)$$

$$\phi ::= \begin{cases} t : \text{Formula} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid \mathbf{P}(a,t,\phi) \mid \mathbf{K}(a,t,\phi) \mid \mathbf{C}(t,\phi) \\ \mathbf{S}(a,b,t,\phi) \mid \mathbf{S}(a,t,\phi) \mid \mathbf{B}(a,t,\phi) \mid \mathbf{D}(a,t,Holds(f,t')) \mid \mathbf{I}(a,t,\phi) \\ \mathbf{O}(a,t,\phi,(\neg)happens(action(a^*,\alpha),t')) \end{cases}$$

Inference Schemata

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

$$\frac{}{\mathbf{C}(t,\mathbf{P}(a,t,\phi) \rightarrow \mathbf{K}(a,t,\phi))}\ [R_1] \qquad \frac{}{\mathbf{C}(t,\mathbf{K}(a,t,\phi) \rightarrow \mathbf{B}(a,t,\phi))}\ [R_2]$$

$$\frac{\mathbf{C}(t,\phi)\ t \leq t_1 \ldots t \leq t_n}{\mathbf{K}(a_1,t_1,\ldots \mathbf{K}(a_n,t_n,\phi)\ldots)}\ [R_3] \qquad \frac{\mathbf{K}(a,t,\phi)}{\phi}\ [R_4]$$

$$\frac{}{\mathbf{C}(t,\mathbf{K}(a,t_1,\phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{K}(a,t_2,\phi_1) \rightarrow \mathbf{K}(a,t_3,\phi_2)}\ [R_5]$$

$$\frac{}{\mathbf{C}(t,\mathbf{B}(a,t_1,\phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{B}(a,t_2,\phi_1) \rightarrow \mathbf{B}(a,t_3,\phi_2)}\ [R_6]$$

$$\frac{}{\mathbf{C}(t,\mathbf{C}(t_1,\phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{C}(t_2,\phi_1) \rightarrow \mathbf{C}(t_3,\phi_2)}\ [R_7]$$

$$\frac{}{\mathbf{C}(t,\forall x.\ \phi \rightarrow \phi[x \mapsto t])}\ [R_8] \qquad \frac{}{\mathbf{C}(t,\phi_1 \leftrightarrow \phi_2 \rightarrow \neg\phi_2 \rightarrow \neg\phi_1)}\ [R_9]$$

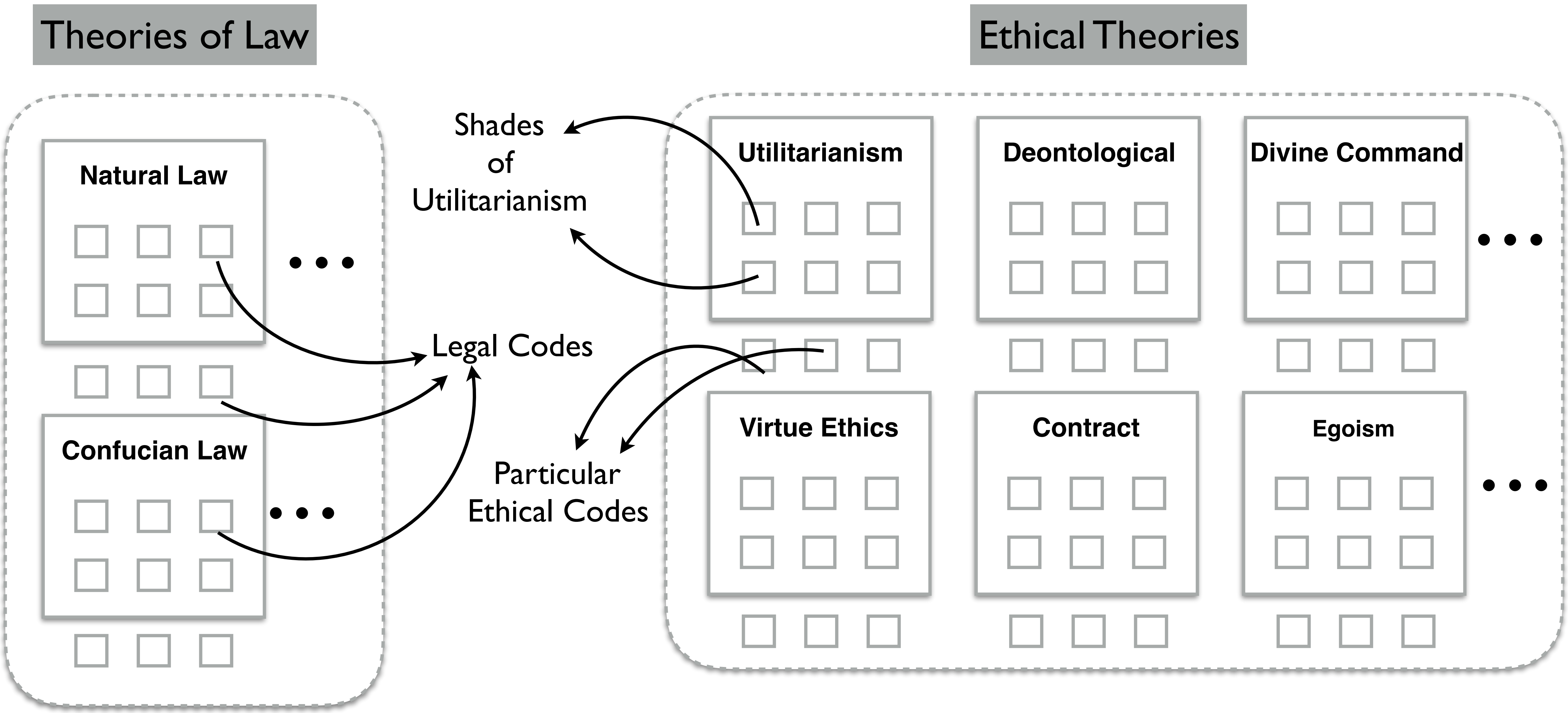
$$\frac{}{\mathbf{C}(t,[\phi_1 \wedge \ldots \wedge \phi_n \rightarrow \phi] \rightarrow [\phi_1 \rightarrow \ldots \rightarrow \phi_n \rightarrow \psi])}\ [R_{10}]$$

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

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

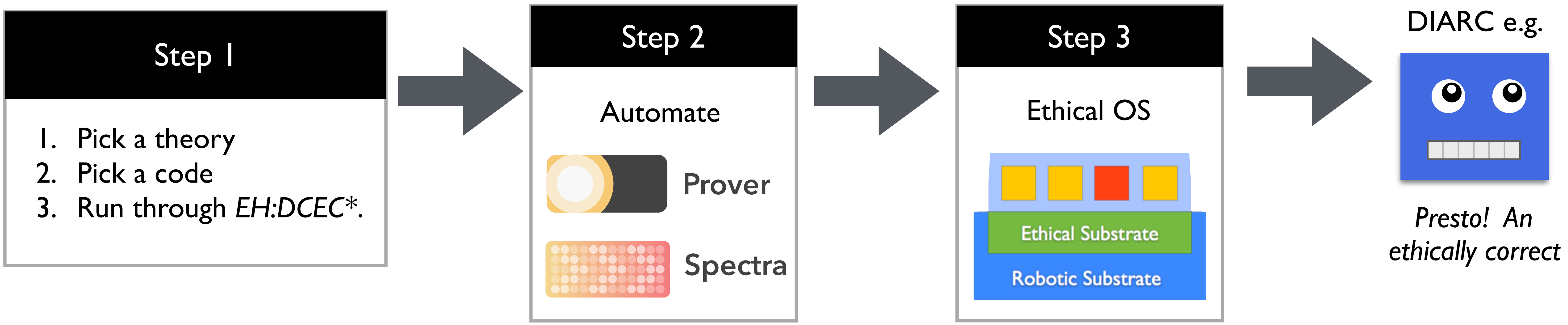
A twist befell the sanguine logicists …

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Ethical Theories



Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

Step 2

Automate

Prover

Spectra

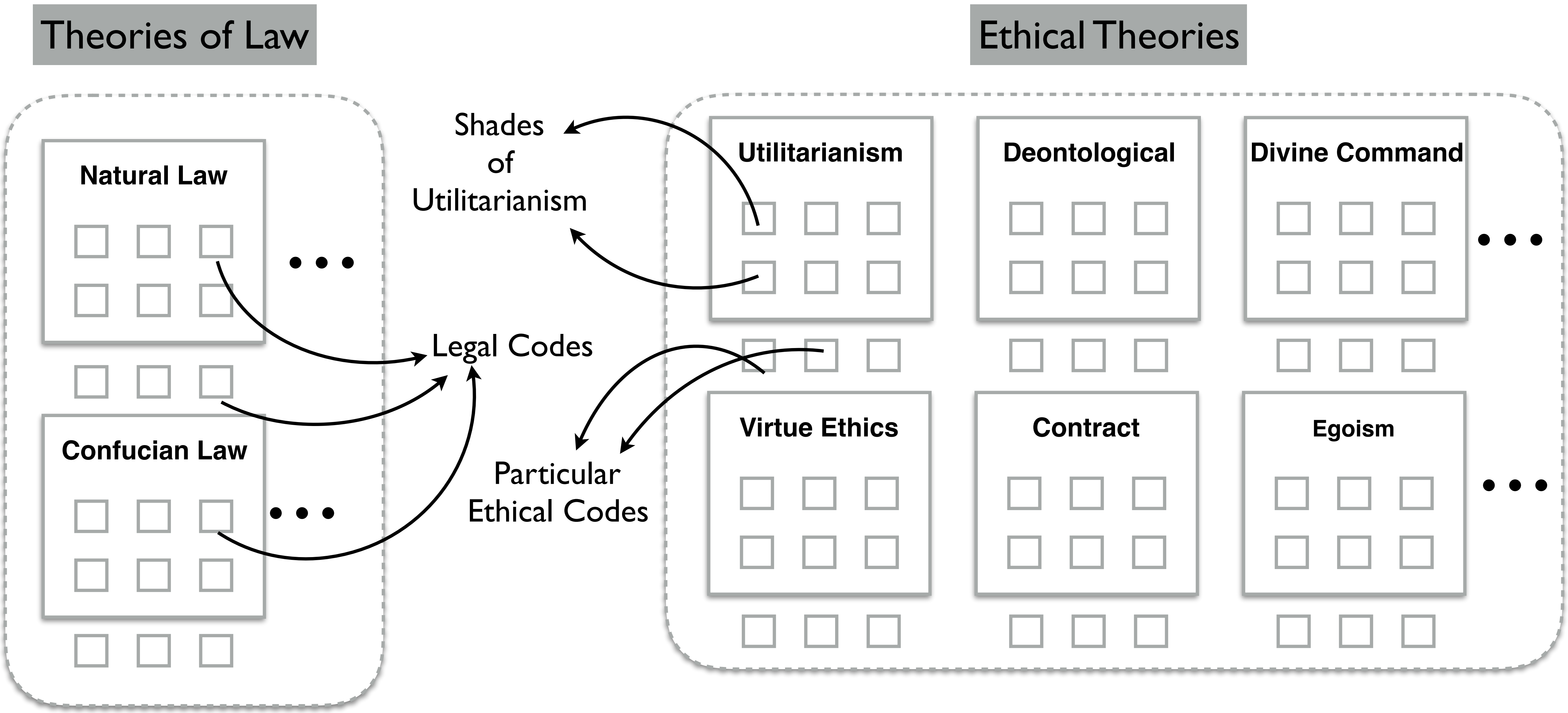
Step 3

Ethical OS

DIARC e.g.

Presto! An ethically correct

Making Ethically Correct Robots/Machines in Four Not-so-easy Steps



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Utilitarianism

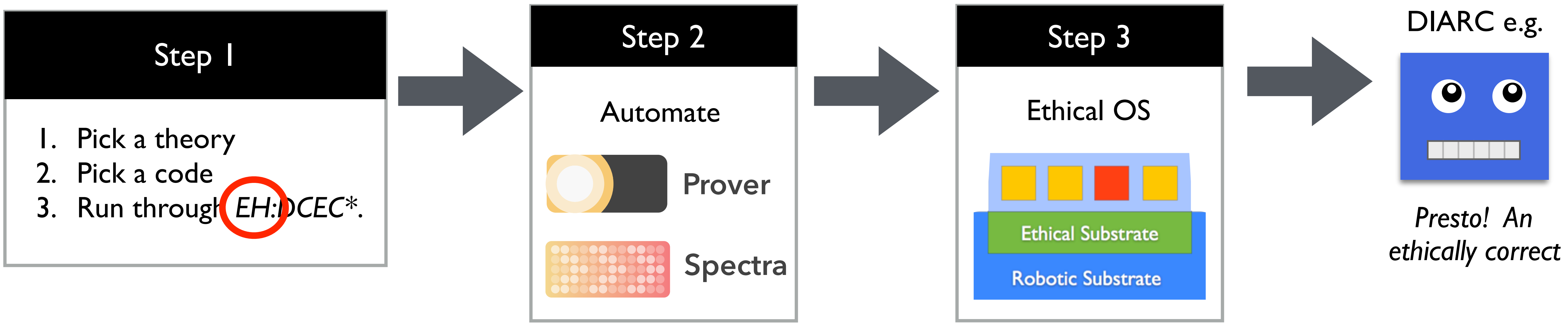
Deontological

Divine Command

Virtue Ethics

Contract

Egoism



Step 1

1. Pick a theory
2. Pick a code
3. Run through *EH:DCEC**.

Step 2

Automate

Prover

Spectra

Step 3

Ethical OS

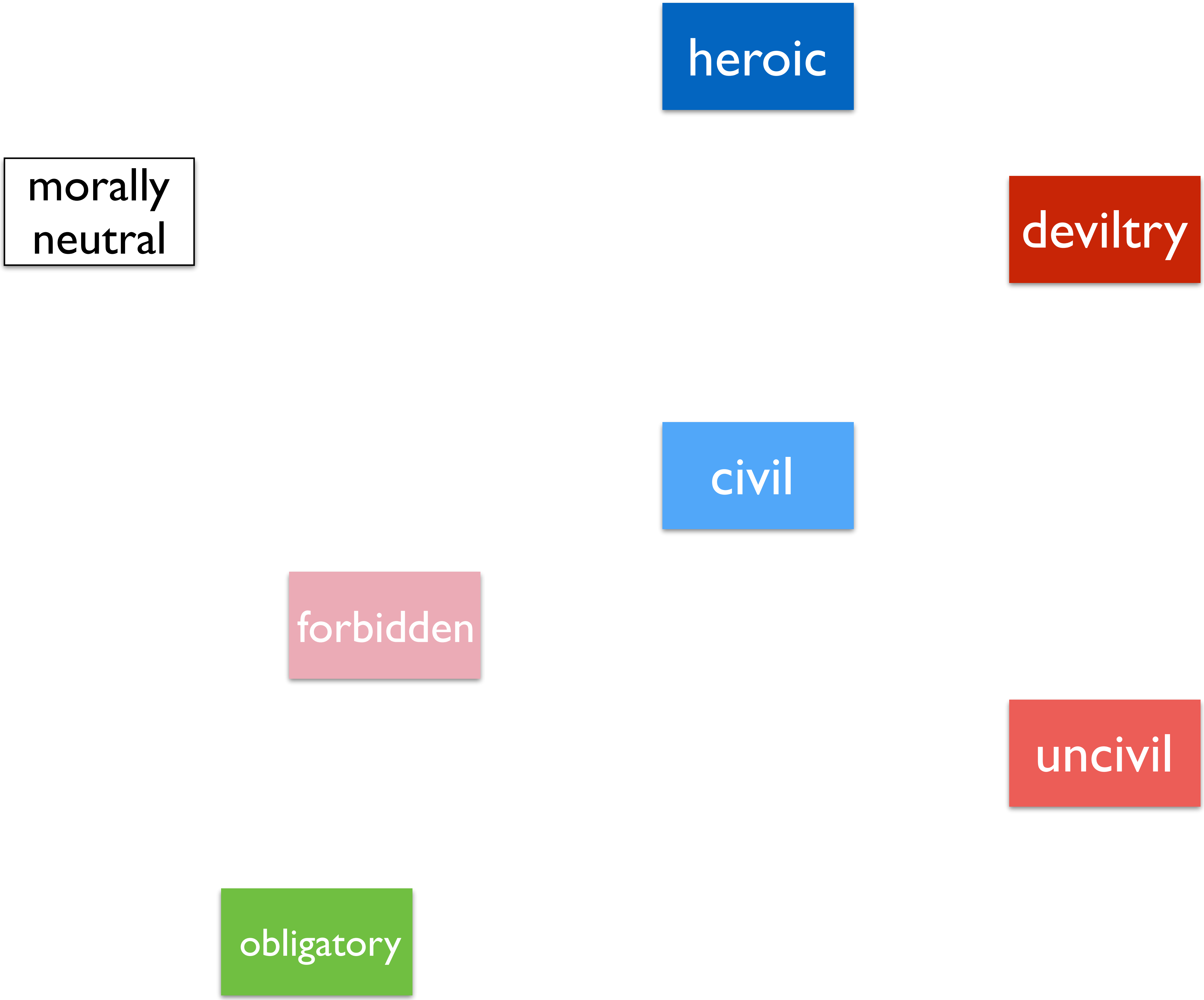
Ethical Substrate

Robotic Substrate

DIARC e.g.

Presto! An ethically correct

Chisholm had argued that the three old 19th-century ethical categories (*forbidden*, *morally neutral*, *obligatory*) are not enough — and soul-searching brought me to agreement.



heroic

morally neutral

deviltry

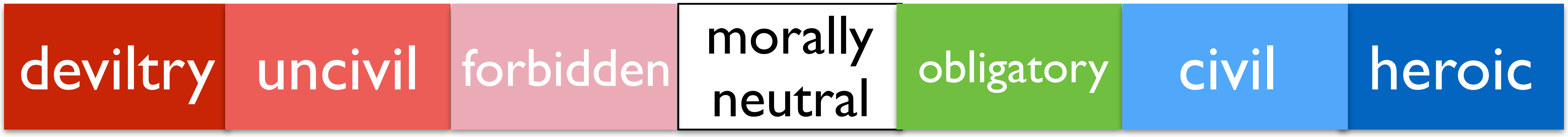
civil

forbidden

uncivil

obligatory

Leibnizian Ethical Hierarchy for Persons and Robots: $\mathcal{EH}$



deviltry

uncivil

forbidden

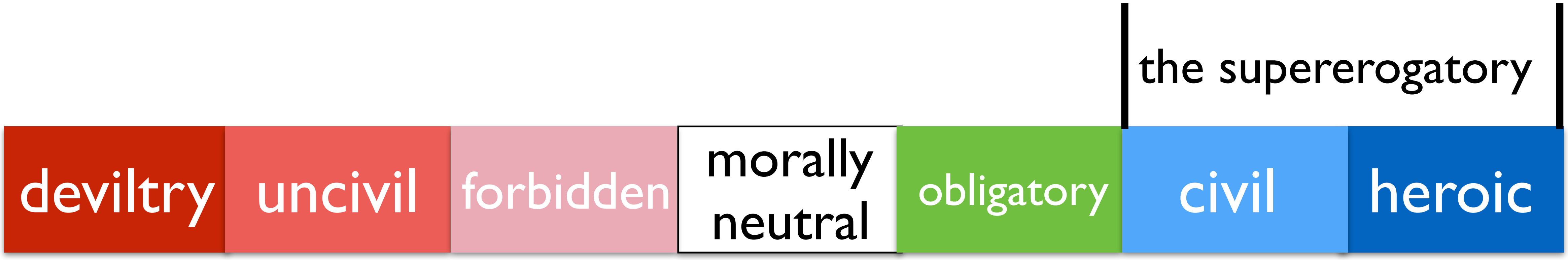
morally neutral

obligatory

civil

heroic

Leibnizian Ethical Hierarchy for Persons and Robots: $\mathcal{EH}$



the supererogatory

deviltry

uncivil

forbidden

morally neutral

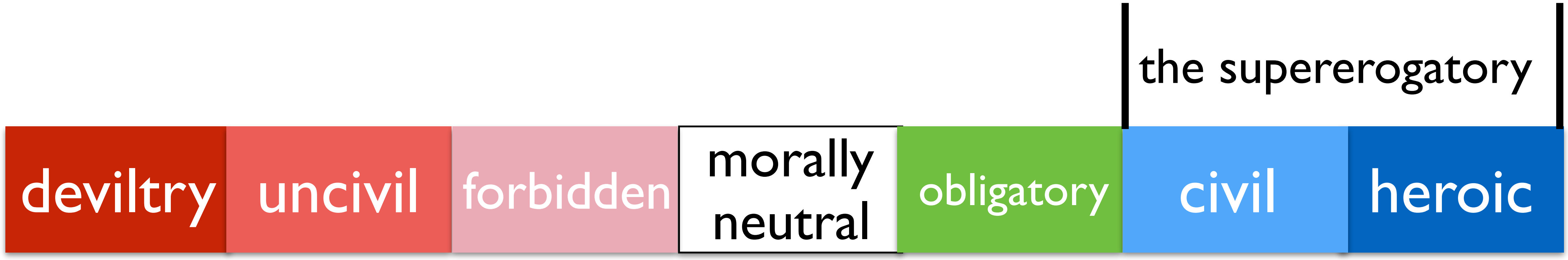
obligatory

civil

heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathcal{EH}$



the supererogatory

devil try

uncivil

forbidden

morally neutral

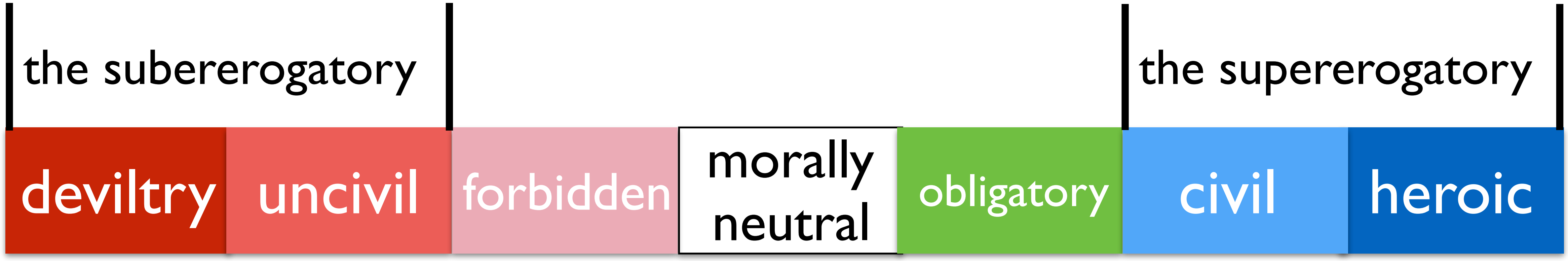
obligatory

civil

heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathcal{EH}$



the suberogatory

the supererogatory

deviltry

uncivil

forbidden

morally neutral

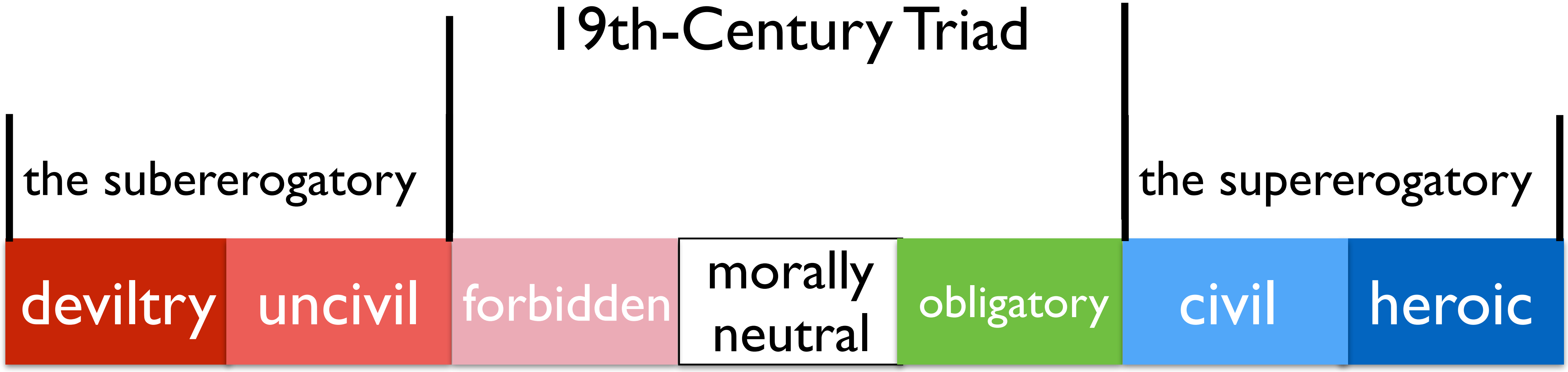
obligatory

civil

heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathscr{EH}$



19th-Century Triad

the subererogatory

the supererogatory

deviltry

uncivil

forbidden

morally neutral

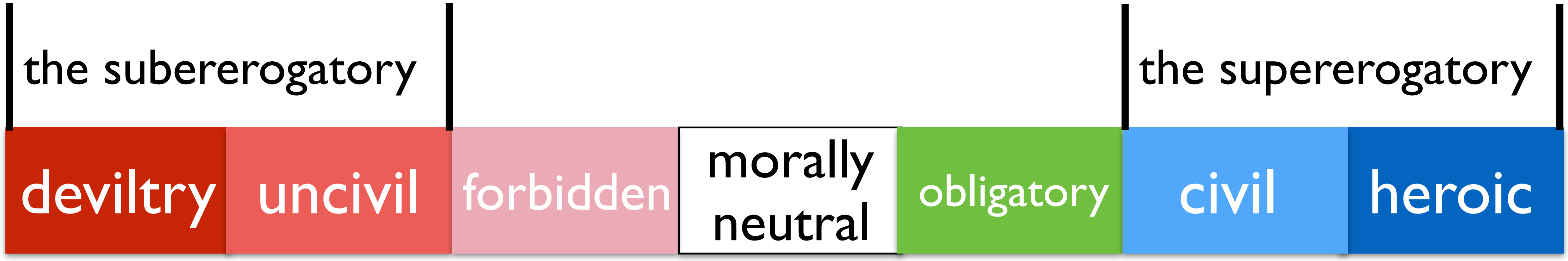
obligatory

civil

heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathcal{EH}$



the suberogatory

the supererogatory

deviltry

uncivil

forbidden

morally neutral

obligatory

civil

heroic

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$$\begin{array}{ccccc|c|ccccc} & & \mathcal{F} & & & \mathcal{P} \wedge \neg\mathcal{O} & & & \mathcal{O} & & \\ \forall & \mathtt{F} & \mathtt{M} & \mathtt{V} & \exists & & \forall & \mathtt{F} & \mathtt{M} & \mathtt{V} & \exists \end{array}$$

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}$
$\forall$ **F** **M** **V** $\exists$		$\forall$ **F** **M** **V** $\exists$

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}^{L}$	$\mathcal{O}^{M}$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$		$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$
						$\uparrow$	

$\mathscr{T} := \|\mathcal{F}|\mathcal{P}\wedge\neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathcal{F}$	$\mathcal{P}\wedge\neg\mathcal{O}$	$\mathcal{O}$
$\forall$ **F** **M** **V** $\exists$		$\forall$ **F** **M** **V** $\exists$

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P}\wedge\neg\mathcal{O}$	$\mathcal{O}^L$	$\mathcal{O}^M$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$		$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$
						↑	
				●			

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}$
$\forall$ **F** **M** **V** $\exists$		$\forall$ **F** **M** **V** $\exists$

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}^L$	$\mathcal{O}^M$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$		$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$
						↑	
				●			

Arkin
Pereira
Andersons
Powers
Mikhail
…

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathcal{F}$					$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}$				
∀	**F**	**M**	**V**	∃		∀	**F**	**M**	**V**	∃

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}^L$	$\mathcal{O}^M$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
∃–∀	∃–∀	∃–∀		∃–∀	∃–∀	∃–∀	∃–∀
				●		↑	

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \land \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathcal{F}$	$\mathcal{P} \land \neg\mathcal{O}$	$\mathcal{O}$
$\forall$ **F** **M** **V** $\exists$		$\forall$ **F** **M** **V** $\exists$

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P} \land \neg\mathcal{O}$	$\mathcal{O}^L$	$\mathcal{O}^M$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$		$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$
				●		$\uparrow$	

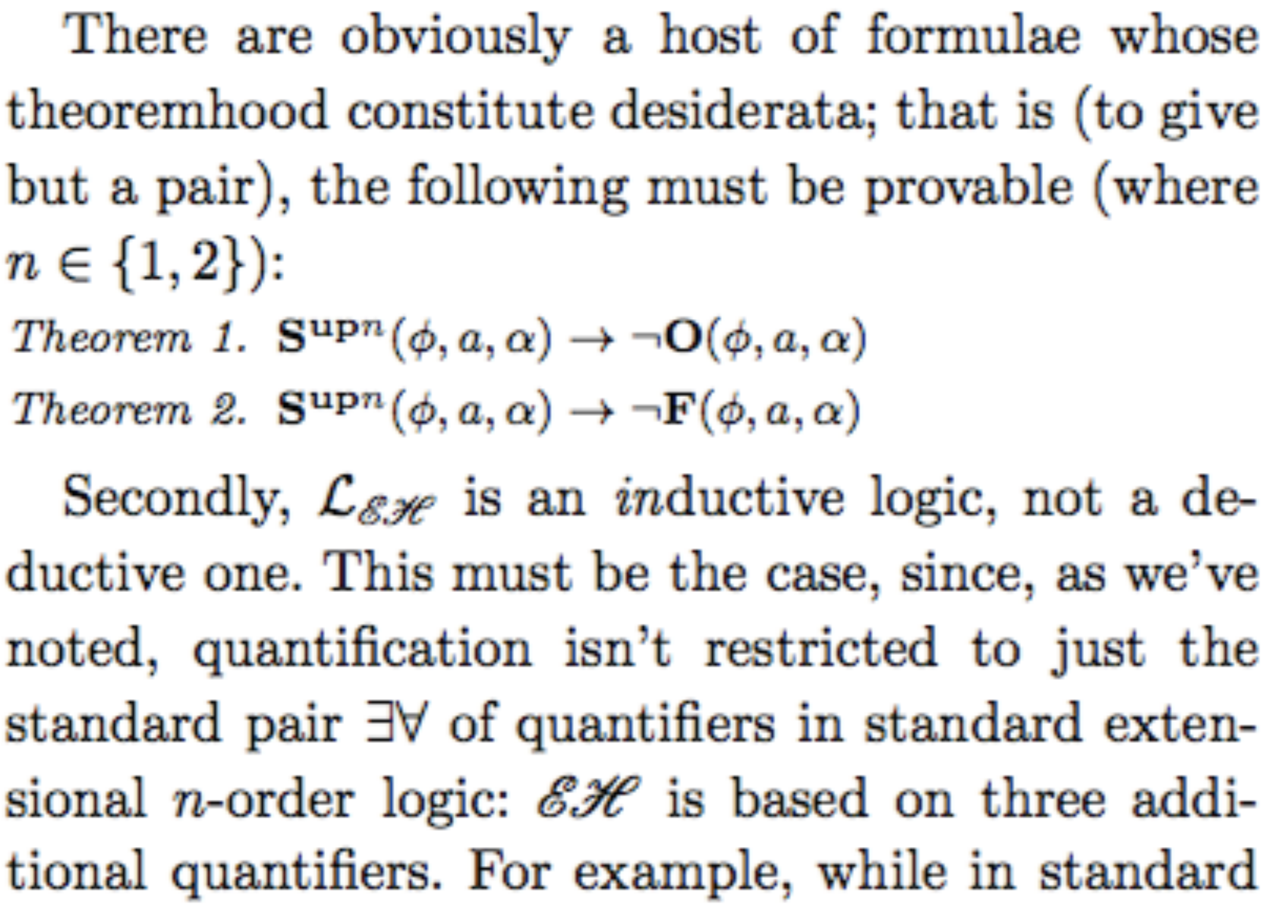
RAIR
Rensselaer AI and Reasoning Lab

$\mathscr{T} := \|\mathcal{F}|\mathcal{P}\wedge\neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathcal{F}$	$\mathcal{P}\wedge\neg\mathcal{O}$	$\mathcal{O}$
∀ **F** **M** **V** ∃		∀ **F** **M** **V** ∃

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P}\wedge\neg\mathcal{O}$	$\mathcal{O}^{L}$	$\mathcal{O}^{M}$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
∃–∀	∃–∀	∃–∀		∃–∀	∃–∀	∃–∀	∃–∀
				●		↑	



There are obviously a host of formulae whose theoremhood constitute desiderata; that is (to give but a pair), the following must be provable (where $n \in \{1,2\}$):

Theorem 1. $\mathbf{S}^{\mathbf{up}^n}(\phi, a, \alpha) \rightarrow \neg\mathbf{O}(\phi, a, \alpha)$

Theorem 2. $\mathbf{S}^{\mathbf{up}^n}(\phi, a, \alpha) \rightarrow \neg\mathbf{F}(\phi, a, \alpha)$

Secondly, $\mathcal{L}_{\mathscr{EH}}$ is an *inductive* logic, not a deductive one. This must be the case, since, as we've noted, quantification isn't restricted to just the standard pair ∃∀ of quantifiers in standard extensional n-order logic: $\mathscr{EH}$ is based on three additional quantifiers. For example, while in standard

Bert "Heroically" Saved?

Courtesy of RAIR-Lab Researcher Atriya Sen

Bert "Heroically" Saved?

Courtesy of RAIR-Lab Researcher Atriya Sen

Supererogatory^2 Robot Action

Courtesy of RAIR-Lab Researcher Atriya Sen

Courtesy of RAIR-Lab Researcher Atriya Sen

Bert “Heroically” Saved!!

Courtesy of RAIR-Lab Researcher Atriya Sen

Bert “Heroically” Saved!!

Courtesy of RAIR-Lab Researcher Atriya Sen

Courtesy of RAIR-Lab Researcher Atriya Sen

$$K\left(\text{nao}, t_1, \text{lessthan}\left(\text{payoff}\left(\text{nao}^*, \neg\text{dive}, t_2\right), \text{threshold}\right)\right)$$

$$K\left(\text{nao}, t_1, \text{greaterthan}\left(\text{payoff}\left(\text{nao}^*, \text{dive}, t_2\right), \text{threshold}\right)\right)$$

$$K\left(\text{nao}, t_1, \neg O\left(\text{nao}^*, t_2, \text{lessthan}\left(\text{payoff}\left(\text{nao}^*, \neg\text{dive}, t_2\right), \text{threshold}\right), \text{happens}\left(\text{action}\left(\text{nao}^*, \text{dive}\right), t_2\right)\right)\right)$$

$$\therefore K\left(\text{nao}, t_1, S^{\text{UP2}}\left(\text{nao}, t_2, \text{happens}\left(\text{action}\left(\text{nao}^*, \text{dive}\right), t_2\right)\right)\right)$$

$$\therefore I\left(\text{nao}, t_2, \text{happens}\left(\text{action}\left(\text{nao}^*, \text{dive}\right), t_2\right)\right)$$

$$\therefore \text{happens}\left(\text{action}(\text{nao}, \text{dive}), t_2\right)$$

Courtesy of RAIR-Lab Researcher Atriya Sen

$$
\begin{array}{l}
\quad K\left(\mathrm{nao}, t_1, \mathrm{lessthan}\left(\mathrm{payoff}\left(\mathrm{nao}^*, \neg\mathrm{dive}, t_2\right), \mathrm{threshold}\right)\right) \\
K\left(\mathrm{nao}, t_1, \mathrm{greaterthan}\left(\mathrm{payoff}\left(\mathrm{nao}^*, \mathrm{dive}, t_2\right), \mathrm{threshold}\right)\right) \\
K\left(\mathrm{nao}, t_1, \neg O\left(\mathrm{nao}^*, t_2, \mathrm{lessthan}\left(\mathrm{payoff}\left(\mathrm{nao}^*, \neg\mathrm{dive}, t_2\right), \mathrm{threshold}\right), \mathrm{happens}\left(\mathrm{action}\left(\mathrm{nao}^*, \mathrm{dive}\right), t_2\right)\right)\right) \\
\therefore K\left(\mathrm{nao}, t_1, S^{\mathrm{UP2}}\left(\mathrm{nao}, t_2, \mathrm{happens}\left(\mathrm{action}\left(\mathrm{nao}^*, \mathrm{dive}\right), t_2\right)\right)\right) \\
\therefore I\left(\mathrm{nao}, t_2, \mathrm{happens}\left(\mathrm{action}\left(\mathrm{nao}^*, \mathrm{dive}\right), t_2\right)\right) \\
\therefore \mathrm{happens}\left(\mathrm{action}(\mathrm{nao}, \mathrm{dive}), t_2\right)
\end{array}
$$

Courtesy of RAIR-Lab Researcher Atriya Sen

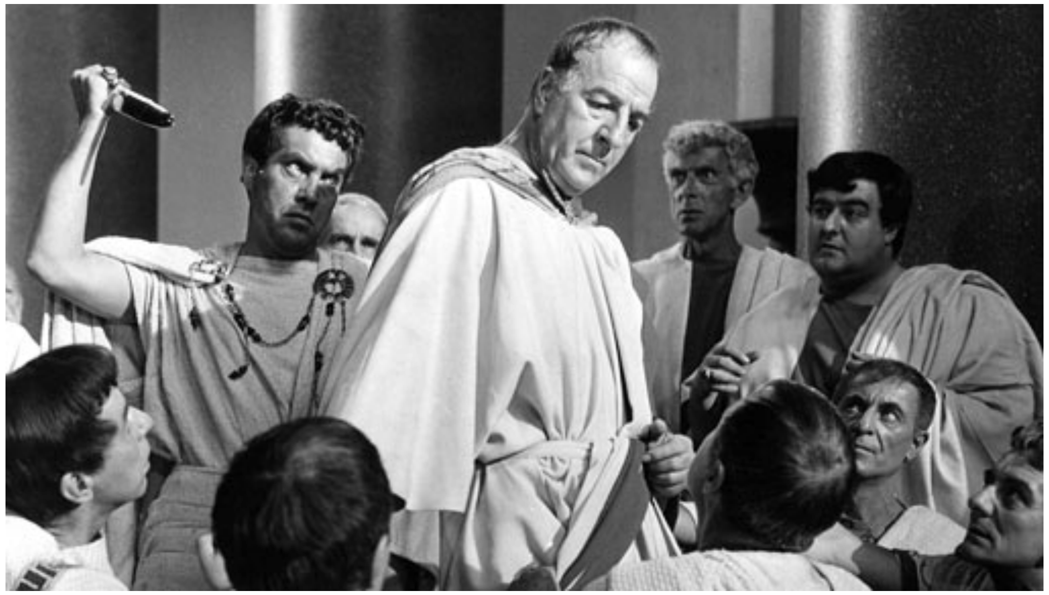
In Talos (available via Web interface); & ShadowProver

```
Prototypes:
Boolean lessThan Numeric Numeric
Boolean greaterThan Numeric Numeric
ActionType not ActionType
ActionType dive

Axioms:
lessOrEqual(Moment t1,t2)
K(nao,t1,lessThan(payoff(nao,not(dive),t2),threshold))
K(nao,t1,greaterThan(payoff(nao,dive,t2),threshold))
K(nao,t1,not(O(nao,t2,lessThan(payoff(nao,not(dive),t2),threshold),happens(action(nao,dive),t2))))

provable Conjectures:
happens(action(nao,dive),t2)
K(nao,t1,SUP2(nao,t2,happens(action(nao,dive),t2)))
I(nao,t2,happens(action(nao,dive),t2))
```

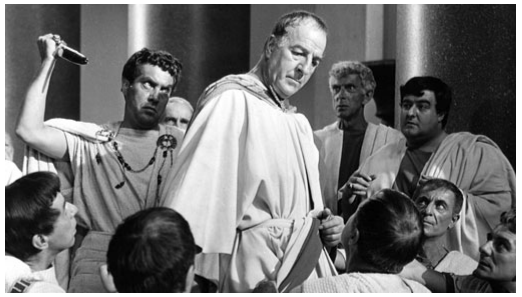
In Talos (available via Web interface); & ShadowProver

```
Prototypes:
Boolean lessThan Numeric Numeric
Boolean greaterThan Numeric Numeric
ActionType not ActionType
ActionType dive

Axioms:
lessOrEqual(Moment t1,t2)
K(nao,t1,lessThan(payoff(nao,not(dive),t2),threshold))
K(nao,t1,greaterThan(payoff(nao,dive,t2),threshold))
K(nao,t1,not(O(nao,t2,lessThan(payoff(nao,not(dive),t2),threshold),happens(action(nao,dive),t2))))

provable Conjectures:
happens(action(nao,dive),t2)
K(nao,t1,SUP2(nao,t2,happens(action(nao,dive),t2)))
I(nao,t2,happens(action(nao,dive),t2))
```

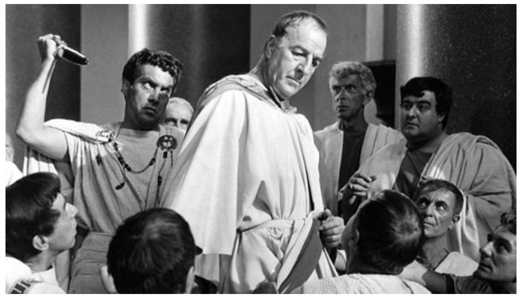
But Our Hardest Challenge: Context



But Our Hardest Challenge: Context



But Our Hardest Challenge: Context



But Our Hardest Challenge: Context

But Our Hardest Challenge: Context

But Our **Hardest** Challenge: Context

$ist(c, \phi)$

mathematically primitive; no internal structure

$$ist(\overset{\downarrow}{c}, \phi)$$

mathematically primitive; no internal structure

$$ist(\overset{\downarrow}{c}, \phi)$$

late 80s, early 90s; McCarthy, Guha

mathematically primitive; no internal structure

$$ist(\overset{\downarrow}{c}, \phi) \qquad \langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle \quad \forall_c, \forall_a$$

late 80s, early 90s; McCarthy, Guha

mathematically primitive; no internal structure

$$ist(\overset{\downarrow}{c}, \phi) \qquad \langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle \quad \forall_c, \forall_a$$

late 80s, early 90s; McCarthy, Guha

Makarios; late 90s, early 00s

mathematically primitive; no internal structure

$ist(\overset{\downarrow}{c}, \phi)$ $\langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle$ $\forall_c, \forall_a$

late 80s, early 90s; McCarthy, Guha

Makarios; late 90s, early 00s

$$\mathbf{O}(\phi|\psi) =_{def} \Box(\psi \to \mathbf{O}(\phi))$$

mathematically primitive; no internal structure

$ist(\overset{\downarrow}{c}, \phi)$ $\quad\quad$ $\langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle \quad \forall_c, \forall_a$

late 80s, early 90s; McCarthy, Guha $\quad\quad$ Makarios; late 90s, early 00s

$$\mathbf{O}(\phi|\psi) =_{def} \Box(\psi \to \mathbf{O}(\phi))$$

late 70s; 80s; Chellas e.g.

mathematically primitive; no internal structure

$$ist(\overset{\downarrow}{c}, \phi) \qquad \langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle \quad \forall_c, \forall_a$$

late 80s, early 90s; McCarthy, Guha

Makarios; late 90s, early 00s

$$\mathbf{O}(\phi|\psi) =_{def} \Box(\psi \rightarrow \mathbf{O}(\phi))$$

late 70s; 80s; Chellas e.g.

$$\mathbf{O}(\phi \mid \psi \mid \gamma)$$

mathematically primitive; no internal structure

$$ist(\overset{\downarrow}{c}, \phi) \qquad\qquad \langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle \quad \forall_c, \forall_a$$

late 80s, early 90s; McCarthy, Guha

Makarios; late 90s, early 00s

$$\mathbf{O}(\phi|\psi) =_{def} \Box(\psi \rightarrow \mathbf{O}(\phi))$$

late 70s; 80s; Chellas e.g.

$$\mathbf{O}(\phi \mid \psi \mid \gamma)$$

late 90s; 00s; van derTorre e.g.

mathematically primitive; no internal structure

$$ist(c, \phi) \qquad \langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle \quad \forall_c, \forall_a$$

late 80s, early 90s; McCarthy, Guha

Makarios; late 90s, early 00s

merely propositional calculus

$$\mathbf{O}(\phi|\psi) =_{def} \Box(\psi \rightarrow \mathbf{O}(\phi))$$

late 70s; 80s; Chellas e.g.

$$\mathbf{O}(\phi \mid \psi \mid \gamma)$$

late 90s; 00s; van derTorre e.g.

mathematically primitive; no internal structure

$ist(c, \phi)$ $\langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle$ $\forall_c, \forall_a$

late 80s, early 90s; McCarthy, Guha

Makarios; late 90s, early 00s

merely propositional calculus

$$\mathbf{O}(\phi|\psi) =_{def} \Box(\psi \rightarrow \mathbf{O}(\phi))$$

late 70s; 80s; Chellas e.g.

$$\mathbf{O}(\phi \mid \psi \mid \gamma)$$

late 90s; 00s; van derTorre e.g.

no implementation; ergo certainly no automated prover for installation in a robot!

mathematically primitive; no internal structure

$ist(c, \phi)$ $\quad$ $\langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle \quad \forall_c, \forall_a$

late 80s, early 90s; McCarthy, Guha

Makarios; late 90s, early 00s

merely propositional calculus

$$\mathbf{O}(\phi|\psi) =_{def} \Box(\psi \rightarrow \mathbf{O}(\phi))$$

late 70s; 80s; Chellas e.g.

$$\mathbf{O}(\phi \mid \psi \mid \gamma)$$

late 90s; 00s; van derTorre e.g.

no implementation; ergo certainly no automated prover for installation in a robot!

$$\frac{\mathbf{K}(a, t, \phi) \quad \mathbf{K}[\mathbf{O}(a, t, \phi, happens(action(a^*, \alpha), t'))] \quad \Diamond[(a, t, \phi, happens(action(a^*, \alpha), t'))]}{\mathbf{O}(a, t, \phi, happens(action(a^*, \alpha), t'))}$$

mathematically primitive; no internal structure

$$ist(\overset{\downarrow}{c}, \phi) \qquad \langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle \quad \forall_c, \forall_a$$

late 80s, early 90s; McCarthy, Guha

Makarios; late 90s, early 00s

merely propositional calculus

$$\mathbf{O}(\phi|\psi) =_{def} \Box(\psi \rightarrow \mathbf{O}(\phi))$$

late 70s; 80s; Chellas e.g.

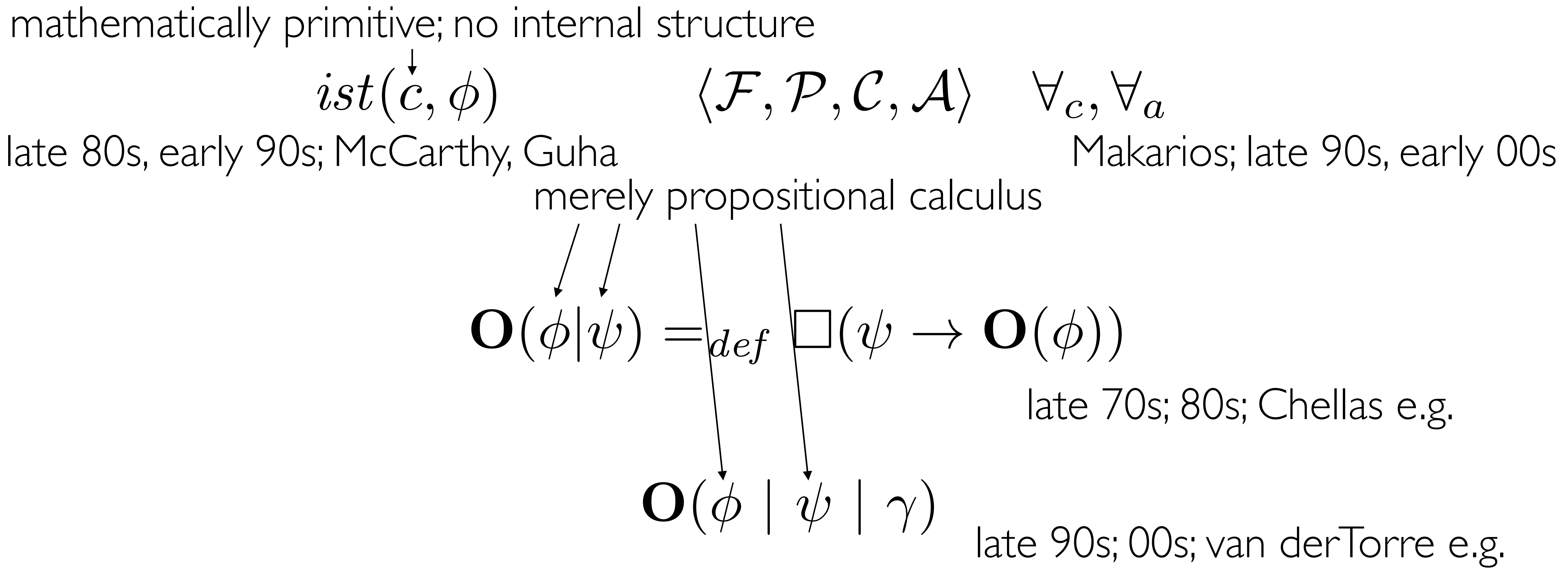
$$\mathbf{O}(\phi \mid \psi \mid \gamma)$$

late 90s; 00s; van der Torre e.g.

no implementation; ergo certainly no automated prover for installation in a robot!

$$\frac{\mathbf{K}(a, t, \boxed{\phi}) \quad \mathbf{K}[\mathbf{O}(a, t, \boxed{\phi}, happens(action(a^*, \alpha), t'))] \quad \Diamond[(a, t, \phi, happens(action(a^*, \alpha), t'))]}{\mathbf{O}(a, t, \phi, happens(action(a^*, \alpha), t'))}$$

But still: A context isn't going to be stuffed into an individual, symbolic formula.



mathematically primitive; no internal structure

$ist(c, \phi)$ $\quad$ $\langle \mathcal{F}, \mathcal{P}, \mathcal{C}, \mathcal{A} \rangle$ $\quad$ $\forall_c, \forall_a$

late 80s, early 90s; McCarthy, Guha

Makarios; late 90s, early 00s

merely propositional calculus

$$\mathbf{O}(\phi|\psi) =_{def} \Box(\psi \rightarrow \mathbf{O}(\phi))$$

late 70s; 80s; Chellas e.g.

$$\mathbf{O}(\phi \mid \psi \mid \gamma)$$

late 90s; 00s; van der Torre e.g.

no implementation; ergo certainly no automated prover for installation in a robot!

$$\frac{\mathbf{K}(a, t\boxed{, \phi)} \quad \mathbf{K}[\mathbf{O}(a, t\boxed{, \phi,} happens(action(a^*, \alpha), t'))] \quad \Diamond[(a, t, \phi, happens(action(a^*, \alpha), t'))]}{\mathbf{O}(a, t, \phi, happens(action(a^*, \alpha), t'))}$$

But still: A context isn't going to be stuffed into an individual, symbolic formula.

(Sarathy, **Scheutz**, …, **Malle** 2017):

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \; C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \ C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

Concern: When any conjunct in the antecedent fails to hold, the norm vacuously holds! — assuming this is a material conditional.

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \; C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

Concern: When any conjunct in the antecedent fails to hold, the norm vacuously holds! — assuming this is a material conditional.

But at any rate:

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \; C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

Concern: When any conjunct in the antecedent fails to hold, the norm vacuously holds! — assuming this is a material conditional.

But at any rate:

Theorem: Norms here can be expressed as formulae in any *DCEC** calculus without loss of proof-theoretic meaning.

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \ C_1, \ldots, C_n \rightarrow (\neg\ \mathbf{O}(a, \bigwedge C_i, \bigwedge A_i, t'))\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

Concern: When any conjunct in the antecedent fails to hold, the norm vacuously holds! — assuming this is a material conditional.

But at any rate:

Theorem: Norms here can be expressed as formulae in any *DCEC** calculus without loss of proof-theoretic meaning.

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \ C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

Concern: When any conjunct in the antecedent fails to hold, the norm vacuously holds! — assuming this is a material conditional.

But at any rate:

Theorem: Norms here can be expressed as formulae in any *DCEC** calculus without loss of proof-theoretic meaning.

$$\mathbf{O}(a, \bigwedge C_i, t, \bigwedge A_i, t')$$

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \; C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

Concern: When any conjunct in the antecedent fails to hold, the norm vacuously holds! — assuming this is a material conditional.

But at any rate:

Theorem: Norms here can be expressed as formulae in any *DCEC** calculus without loss of proof-theoretic meaning.

$$\mathbf{O}(a, \bigwedge C_i, t, \bigwedge A_i, t')$$

$$\mathcal{N}' := \; [\alpha, \beta] :: C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \; C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

Concern: When any conjunct in the antecedent fails to hold, the norm vacuously holds! — assuming this is a material conditional.

But at any rate:

Theorem: Norms here can be expressed as formulae in any *DCEC** calculus without loss of proof-theoretic meaning.

$$\mathbf{O}(a, \bigwedge C_i, t, \bigwedge A_i, t')$$

$$\mathcal{N}' := \; [\alpha, \beta] :: C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

D-S confidence interval

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \; C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

Concern: When any conjunct in the antecedent fails to hold, the norm vacuously holds! — assuming this is a material conditional.

But at any rate:

Theorem: Norms here can be expressed as formulae in any *DCEC** calculus without loss of proof-theoretic meaning.

$$\mathbf{O}(a, \bigwedge C_i, t, \bigwedge A_i, t')$$

$$\mathcal{N}' := \; [\alpha, \beta] :: C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

D-S confidence interval

Theorem: Norms with this form of uncertainty can be expressed as formulae in any uncertainty-ized *DCEC** calculus without loss of proof-theoretic meaning.

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \ C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

Concern: When any conjunct in the antecedent fails to hold, the norm vacuously holds! — assuming this is a material conditional.

But at any rate:

Theorem: Norms here can be expressed as formulae in any *DCEC** calculus without loss of proof-theoretic meaning.

$$\mathbf{O}(a, \bigwedge C_i, t, \bigwedge A_i, t')$$

$$\mathcal{N}' := \ [\alpha, \beta] :: C_1, \ldots, C_n \ \mathbf{B}_{\mathfrak{a}}^{0\leq\sigma\leq 13}/\mathbf{K}^{0\leq\sigma\leq 13}[\mathbf{O}(a, \bigwedge C_i, t, \bigwedge A_i, t')] \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

D-S confidence interval

Theorem: Norms with this form of uncertainty can be expressed as formulae in any uncertainty-ized *DCEC** calculus without loss of proof-theoretic meaning.

(Sarathy, **Scheutz**, …, **Malle** 2017):

$$\mathcal{N} := \ C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

Concern: When any conjunct in the antecedent fails to hold, the norm vacuously holds! — assuming this is a material conditional.

But at any rate:

Theorem: Norms here can be expressed as formulae in any *DCEC** calculus without loss of proof-theoretic meaning.

$$\mathbf{O}(a, \bigwedge C_i, t, \bigwedge A_i, t')$$

$$\mathcal{N}' := \ [\alpha, \beta] :: C_1, \ldots, C_n \rightarrow (\neg)\mathbf{O}/\mathbf{F}/\mathbf{P}(A_1, A_2, \ldots, A_m)$$

D-S confidence interval

Theorem: Norms with this form of uncertainty can be expressed as formulae in any uncertainty-ized *DCEC** calculus without loss of proof-theoretic meaning.

$$\mathbf{B}_{\mathfrak{a}}^{0 \leq \sigma \leq 13}/\mathbf{K}^{0 \leq \sigma \leq 13} \ [\mathbf{O}(a, \bigwedge C_i, t, \bigwedge A_i, t')]$$

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent’s capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^{L}, O^{M}, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.

+

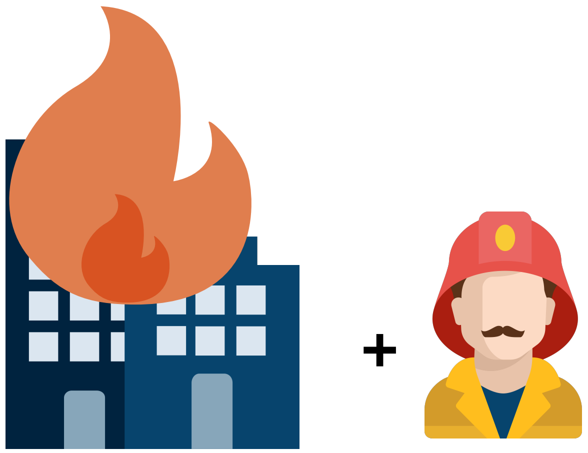
$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.



$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.

+

$\vdash \mathbf{O}(rescue)$

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^{L}, O^{M}, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.

+

$\vdash \mathbf{O}(rescue)$

vs.

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.

+

$\vdash \mathbf{O}(rescue)$

vs.

If a building is on fire, and you have don't know a partial plan for rescuing someone, then it is not obligatory to rescue someone.

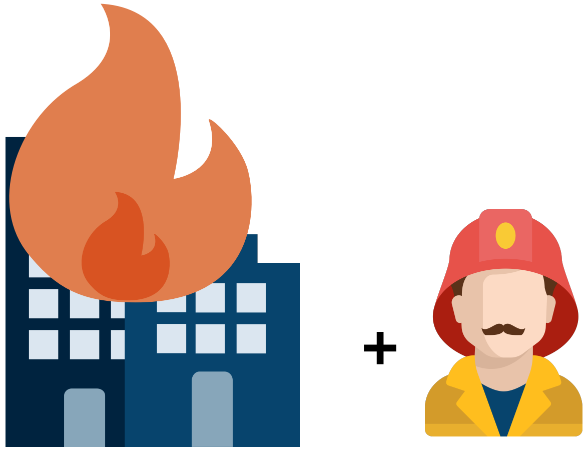
$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^{L}, O^{M}, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.



$\vdash \mathbf{O}(rescue)$

vs.

If a building is on fire, and you have don't know a partial plan for rescuing someone, then it is not obligatory to rescue someone.

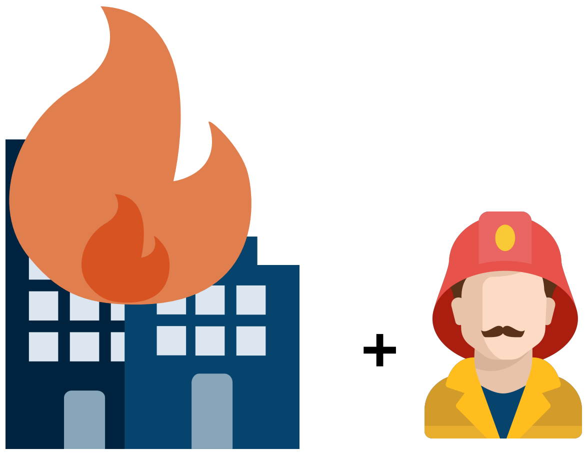
$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^{L}, O^{M}, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.



$\vdash \mathbf{O}(rescue)$

vs.

If a building is on fire, and you have don't know a partial plan for rescuing someone, then it is not obligatory to rescue someone.

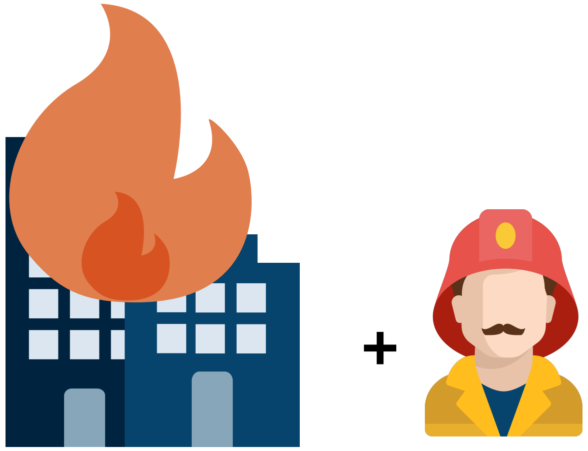
$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.



$\vdash \mathbf{O}(rescue)$

vs.

If a building is on fire, and you have don't know a partial plan for rescuing someone, then it is not obligatory to rescue someone.

+

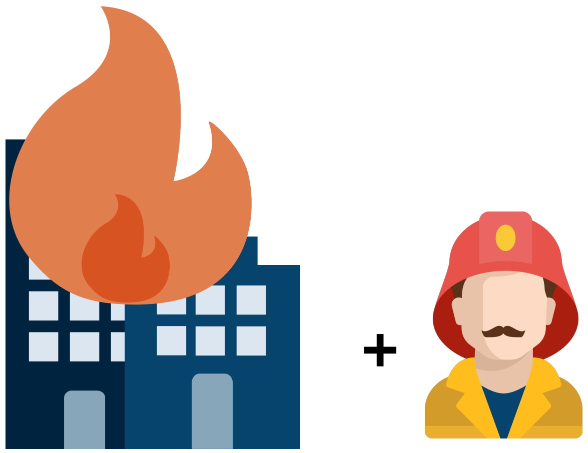
$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^{L}, O^{M}, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

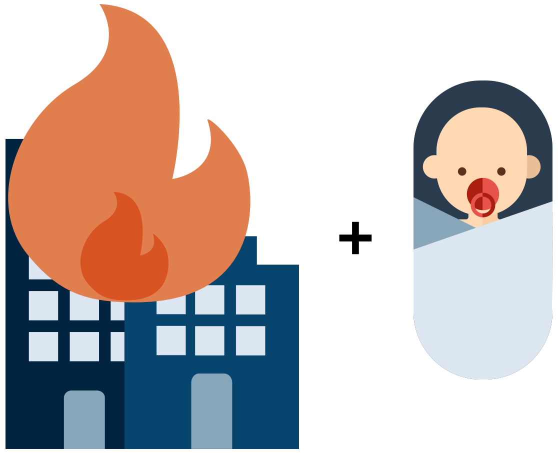
If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.



$\vdash \mathbf{O}(rescue)$

vs.

If a building is on fire, and you have don't know a partial plan for rescuing someone, then it is not obligatory to rescue someone.



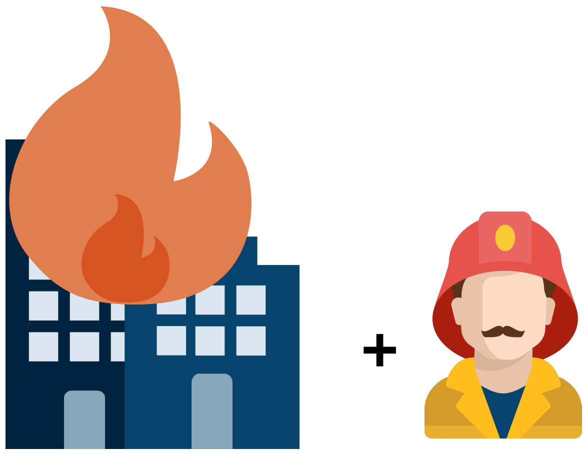
$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^{L}, O^{M}, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

Why go beyond formulae to robust formalization of context?

Ethics and norms change not just based on the situation, but also based on the agent's capabilities (represented here by plans and partial plans).

Example

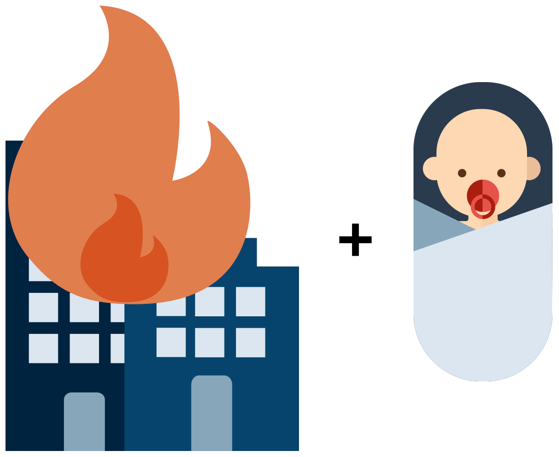
If a building is on fire, and you know a partial plan for rescuing (e.g. you are a fireman), then you ought to rescue.



$\vdash \mathbf{O}(rescue)$

vs.

If a building is on fire, and you have don't know a partial plan for rescuing someone, then it is not obligatory to rescue someone.



$\nvdash \mathbf{O}(rescue)$

We must be able to combine contexts:

$\mathfrak{C}_1 \oplus \mathfrak{C}_2$ **Both contexts hold now**

And we need relations:

$\mathfrak{C}_1 \odot \mathfrak{C}_2$ **The second context occurs within the first**
“A murder within a play”

$\mathfrak{C}_1 \otimes \mathfrak{C}_2$ **The contexts are incompatible**
“Driving a car” and “*Going to sleep”*

And, one context can dominate another:

$$\mathfrak{C}_1 \succ \mathfrak{C}_2$$

$$\mathfrak{C}_{library} \vdash \mathbf{F}(Running)$$

$$\mathfrak{C}_{fire} \vdash \neg\mathbf{F}(Running)$$

$$\mathfrak{C}_{fire} \succ \mathfrak{C}_{library}$$

$$\therefore \mathfrak{C}_{library} \oplus \mathfrak{C}_{fire} \vdash \neg\mathbf{F}(Running)$$

$$\mathfrak{C}_{library} \oplus \mathfrak{C}_{fire} \not\vdash \mathbf{F}(Running)$$

What, then, is a context for Selmer & Naveen? …

Need:

Need:

$$\mu\mathcal{DCEC}^*_3 \in \mathcal{CC}$$

Need:

$$\mu\mathcal{DCEC}^*_{\boxed{3}} \in \mathcal{CC}$$

The Heinz Dilemma (Kohlberg)

“In Europe, a woman was near death from a special kind of cancer. There was one drug that the doctors thought might save her. It was a form of radium that a druggist in the same town had recently discovered. The drug was expensive to make, but the druggist was charging ten times what the drug cost him to make. He paid $200 for the radium and charged $2,000 for a small dose of the drug.

The sick woman’s husband, Heinz, went to everyone he knew to borrow the money, but he could only get together about $1,000, which is half of what it cost. He told the druggist that his wife was dying and asked him to sell it cheaper or let him pay later. But the druggist said: “No, I discovered the drug and I’m going to make money from it.” So Heinz got desperate and broke into the man’s store to steal the drug for his wife. *Should the husband have done that?*”

$DCEC_I$* Specimen from Heinz Dilemma

Given $$\begin{aligned}&\mathbf{B}\Big(\mathsf{I}, \mathsf{now}, \forall t : \mathsf{Moment}, a : \mathsf{Agent}\Big(holds(sick(a),t) \land \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(a), t+t')\big)\\ &\qquad \Rightarrow (happens(dies(a), t+T) \lor holds(dead(a), t+T)\Big)\Big)\end{aligned}$$

Given $$\mathbf{K}\big(\mathsf{I}, \mathsf{now}, holds(sick(wife(\mathsf{I}*)), t_0) \land \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(wife(\mathsf{I}*)), t+t')\big)$$

Inferred $$\mathbf{B}\big(\mathsf{I}, \mathsf{now}, happens(dies(wife(\mathsf{I}*)), t_0+T) \lor holds(dead(wife(\mathsf{I}*)), t_0+T)\big)$$

Given $$\begin{aligned}&\mathbf{K}\big(\mathsf{I}, \mathsf{now}, \mathsf{EventCalculus} \Rightarrow\\ &\qquad \big(happens(dies(wife(\mathsf{I}*)), t_0+T) \lor holds(dead(wife(\mathsf{I}*)), t_0+T) \Rightarrow\\ &\qquad \neg holds(alive(wife(\mathsf{I}*)), t_0+T)\big)\big)\end{aligned}$$

Inferred $$\mathbf{B}\big(\mathsf{I}, \mathsf{now}, \neg holds(alive(wife(\mathsf{I}*)), t_0+T)\big)$$

Given $$\mathbf{D}\big(\mathsf{I}, \mathsf{now}, holds(alive(wife(\mathsf{I}*)), t_0+T)\big)$$

Given $$\begin{aligned}&(\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(f,t)) \land \mathbf{D}(\mathsf{I}, \mathsf{now}, holds(f,t)) \land\\ &\ \mathbf{K}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, \alpha), \mathsf{now}) \Rightarrow holds(f,t)))\\ &\ \Rightarrow \mathbf{I}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, \alpha), \mathsf{now}))\end{aligned}$$

Given $$\mathbf{K}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, treat), \mathsf{now}) \Rightarrow holds(alive(wife(\mathsf{I}*)), t_0+T))\big)$$

Inferred $$\mathbf{I}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, treat), \mathsf{now})\big)$$

$DCEC_I$* Specimen from Heinz Dilemma

Given $\mathbf{B}\Big(\mathsf{I}, \mathsf{now}, \forall t : \mathsf{Moment}, a : \mathsf{Agent}\Big(holds(sick(a),t) \land \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(a),t+t')\big)$

$\Rightarrow (happens(dies(a),t+T) \lor holds(dead(a),t+T)\Big)\Big)$

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, holds(sick(wife(\mathsf{I}*)),t_0) \land \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(wife(\mathsf{I}*)),t+t')\big)$

Inferred $\mathbf{B}(\mathsf{I}, \mathsf{now}, happens(dies(wife(\mathsf{I}*)),t_0+T) \lor holds(dead(wife(\mathsf{I}*)),t_0+T))$

Given $\mathbf{K}(\mathsf{I}, \mathsf{now}, \mathsf{EventCalculus} \Rightarrow$
$(happens(dies(wife(\mathsf{I}*)),t_0+T) \lor holds(dead(wife(\mathsf{I}*)),t_0+T) \Rightarrow$
$\neg holds(alive(wife(\mathsf{I}*)),t_0+T)))$

Inferred $\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(alive(wife(\mathsf{I}*)),t_0+T))$

Given $\mathbf{D}(\mathsf{I}, \mathsf{now}, holds(alive(wife(\mathsf{I}*)),t_0+T))$

Given $(\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(f,t)) \land \mathbf{D}(\mathsf{I}, \mathsf{now}, holds(f,t)) \land$
$\mathbf{K}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, \alpha), \mathsf{now}) \Rightarrow holds(f,t)))$
$\Rightarrow \mathbf{I}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, \alpha), \mathsf{now}))$

Given $\mathbf{K}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, treat), \mathsf{now}) \Rightarrow holds(alive(wife(\mathsf{I}*)),t_0+T)))$

Inferred $\mathbf{I}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, treat), \mathsf{now}))$

*DCEC$_I$** Specimen from Heinz Dilemma

Given $\mathbf{B}\Big(\mathsf{I}, \mathsf{now}, \forall t : \mathsf{Moment}, a : \mathsf{Agent}\Big(holds(sick(a),t) \wedge \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(a), t+t')\big)$

$\Rightarrow \big(happens(dies(a), t+T) \vee holds(dead(a), t+T)\big)\Big)\Big)$

FOL

Given $\mathbf{K}\Big(\mathsf{I}, \mathsf{now}, holds(sick(wife(\mathsf{I}*)), t_0) \wedge \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(wife(\mathsf{I}*)), t+t')\big)\Big)$

Inferred $\mathbf{B}\big(\mathsf{I}, \mathsf{now}, happens(dies(wife(\mathsf{I}*)), t_0+T) \vee holds(dead(wife(\mathsf{I}*)), t_0+T)\big)$

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, \mathsf{EventCalculus} \Rightarrow$

$\big(happens(dies(wife(\mathsf{I}*)), t_0+T) \vee holds(dead(wife(\mathsf{I}*)), t_0+T) \Rightarrow$

$\neg holds(alive(wife(\mathsf{I}*)), t_0+T)\big)\big)$

Inferred $\mathbf{B}\big(\mathsf{I}, \mathsf{now}, \neg holds(alive(wife(\mathsf{I}*)), t_0+T)\big)$

Given $\mathbf{D}\big(\mathsf{I}, \mathsf{now}, holds(alive(wife(\mathsf{I}*)), t_0+T)\big)$

Given $\big(\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(f,t)) \wedge \mathbf{D}(\mathsf{I}, \mathsf{now}, holds(f,t)) \wedge$

$\mathbf{K}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, \alpha), \mathsf{now}) \Rightarrow holds(f,t)\big)\big)$

$\Rightarrow \mathbf{I}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, \alpha), \mathsf{now})\big)$

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, treat), \mathsf{now}) \Rightarrow holds(alive(wife(\mathsf{I}*)), t_0+T)\big)\big)$

Inferred $\mathbf{I}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, treat), \mathsf{now})\big)$

DCEC$_I$* Specimen from Heinz Dilemma

Given $\mathbf{B}\left(\mathsf{I}, \mathsf{now}, \forall t : \mathsf{Moment}, a : \mathsf{Agent}\left(holds(sick(a),t) \wedge \left(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(a),t+t')\right)\right.\right.$

$\left.\left.\Rightarrow \left(happens(dies(a),t+T) \vee holds(dead(a),t+T)\right)\right)\right)$

√ FOL

Given $\mathbf{K}\left(\mathsf{I}, \mathsf{now}, holds(sick(wife(\mathsf{I}*)),t_0) \wedge \left(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(wife(\mathsf{I}*)),t+t')\right)\right)$

Inferred $\mathbf{B}(\mathsf{I}, \mathsf{now}, happens(dies(wife(\mathsf{I}*)),t_0+T) \vee holds(dead(wife(\mathsf{I}*)),t_0+T))$

Given $\mathbf{K}(\mathsf{I}, \mathsf{now}, \mathsf{EventCalculus} \Rightarrow$
$(happens(dies(wife(\mathsf{I}*)),t_0+T) \vee holds(dead(wife(\mathsf{I}*)),t_0+T) \Rightarrow$
$\neg holds(alive(wife(\mathsf{I}*)),t_0+T)))$

Inferred $\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(alive(wife(\mathsf{I}*)),t_0+T))$

Given $\mathbf{D}(\mathsf{I}, \mathsf{now}, holds(alive(wife(\mathsf{I}*)),t_0+T))$

Given $(\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(f,t)) \wedge \mathbf{D}(\mathsf{I}, \mathsf{now}, holds(f,t)) \wedge$
$\mathbf{K}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,\alpha),\mathsf{now}) \Rightarrow holds(f,t)))$
$\Rightarrow \mathbf{I}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,\alpha),\mathsf{now}))$

Given $\mathbf{K}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,treat),\mathsf{now}) \Rightarrow holds(alive(wife(\mathsf{I}*)),t_0+T)))$

Inferred $\mathbf{I}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,treat),\mathsf{now}))$

$DCEC_I$* Specimen from Heinz Dilemma

✓ FOL

Epistemic + FOL

$\mathbf{B}_d\mathbf{B}_v\mathbf{B}_d V v$

Given $\mathbf{B}\Big(\mathsf{I}, \mathsf{now}, \forall t : \mathsf{Moment}, a : \mathsf{Agent}\Big(holds(sick(a),t) \land \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(a),t+t')\big) \Rightarrow \big(happens(dies(a),t+T) \lor holds(dead(a),t+T)\big)\Big)\Big)$

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, holds(sick(wife(\mathsf{I}*)),t_0) \land \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(wife(\mathsf{I}*)),t+t')\big)$

Inferred $\mathbf{B}\big(\mathsf{I}, \mathsf{now}, happens(dies(wife(\mathsf{I}*)),t_0+T) \lor holds(dead(wife(\mathsf{I}*)),t_0+T)\big)$

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, \mathsf{EventCalculus} \Rightarrow \big(happens(dies(wife(\mathsf{I}*)),t_0+T) \lor holds(dead(wife(\mathsf{I}*)),t_0+T) \Rightarrow \neg holds(alive(wife(\mathsf{I}*)),t_0+T)\big)\big)$

Inferred $\mathbf{B}\big(\mathsf{I}, \mathsf{now}, \neg holds(alive(wife(\mathsf{I}*)),t_0+T)\big)$

Given $\mathbf{D}\big(\mathsf{I}, \mathsf{now}, holds(alive(wife(\mathsf{I}*)),t_0+T)\big)$

Given $\big(\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(f,t)) \land \mathbf{D}(\mathsf{I}, \mathsf{now}, holds(f,t)) \land \mathbf{K}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, \alpha), \mathsf{now}) \Rightarrow holds(f,t))\big) \Rightarrow \mathbf{I}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, \alpha), \mathsf{now}))$

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, treat), \mathsf{now}) \Rightarrow holds(alive(wife(\mathsf{I}*)),t_0+T)\big)$

Inferred $\mathbf{I}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, treat), \mathsf{now})\big)$

$DCEC_I$* Specimen from Heinz Dilemma

Given $\mathbf{B}\Big(\mathsf{I}, \mathsf{now}, \forall t : \mathsf{Moment}, a : \mathsf{Agent}\Big(holds(sick(a),t) \wedge \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(a), t+t')\big)$

$\Rightarrow \big(happens(dies(a), t+T) \vee holds(dead(a), t+T)\big)\Big)\Big)$

√ FOL

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, holds(sick(wife(\mathsf{I}*)), t_0) \wedge \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(wife(\mathsf{I}*)), t+t')\big)\big)$

√ Epistemic + FOL

$\mathbf{B}_d\mathbf{B}_v\mathbf{B}_d V v$

Inferred $\mathbf{B}\big(\mathsf{I}, \mathsf{now}, happens(dies(wife(\mathsf{I}*)), t_0+T) \vee holds(dead(wife(\mathsf{I}*)), t_0+T)\big)$

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, \mathsf{EventCalculus} \Rightarrow$
$\big(happens(dies(wife(\mathsf{I}*)), t_0+T) \vee holds(dead(wife(\mathsf{I}*)), t_0+T) \Rightarrow$
$\neg holds(alive(wife(\mathsf{I}*)), t_0+T)\big)\big)$

Inferred $\mathbf{B}\big(\mathsf{I}, \mathsf{now}, \neg holds(alive(wife(\mathsf{I}*)), t_0+T)\big)$

Given $\mathbf{D}\big(\mathsf{I}, \mathsf{now}, holds(alive(wife(\mathsf{I}*)), t_0+T)\big)$

Given $\big(\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(f,t)) \wedge \mathbf{D}(\mathsf{I}, \mathsf{now}, holds(f,t)) \wedge$
$\mathbf{K}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, \alpha), \mathsf{now}) \Rightarrow holds(f,t))\big)$
$\Rightarrow \mathbf{I}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, \alpha), \mathsf{now}))$

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, treat), \mathsf{now}) \Rightarrow holds(alive(wife(\mathsf{I}*)), t_0+T)\big)\big)$

Inferred $\mathbf{I}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*, treat), \mathsf{now})\big)$

$DCEC_I$* Specimen from Heinz Dilemma

Given $\mathbf{B}\Big(\mathsf{I}, \mathsf{now}, \forall t : \mathsf{Moment}, a : \mathsf{Agent}\Big(holds(sick(a),t) \land \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(a),t+t')\big)$

$\Rightarrow \big(happens(dies(a),t+T) \lor holds(dead(a),t+T)\big)\Big)\Big)$

✓ FOL

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, holds(sick(wife(\mathsf{I}*)),t_0) \land \big(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(wife(\mathsf{I}*)),t+t')\big)\big)$

✓ Epistemic + FOL

Inferred $\mathbf{B}\big(\mathsf{I}, \mathsf{now}, happens(dies(wife(\mathsf{I}*)),t_0+T) \lor holds(dead(wife(\mathsf{I}*)),t_0+T)\big)$

$\mathbf{B}_d\mathbf{B}_v\mathbf{B}_d V v$

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, \mathsf{EventCalculus} \Rightarrow$

$\big(happens(dies(wife(\mathsf{I}*)),t_0+T) \lor holds(dead(wife(\mathsf{I}*)),t_0+T) \Rightarrow$

$\neg holds(alive(wife(\mathsf{I}*)),t_0+T)\big)\big)$

TOL

$\exists X[X(j) \land \neg X(m) \land S(X)]$

Inferred $\mathbf{B}\big(\mathsf{I}, \mathsf{now}, \neg holds(alive(wife(\mathsf{I}*)),t_0+T)\big)$

Given $\mathbf{D}\big(\mathsf{I}, \mathsf{now}, holds(alive(wife(\mathsf{I}*)),t_0+T)\big)$

Given $\big(\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(f,t)) \land \mathbf{D}(\mathsf{I}, \mathsf{now}, holds(f,t)) \land$

$\mathbf{K}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,\alpha),\mathsf{now}) \Rightarrow holds(f,t))\big)$

$\Rightarrow \mathbf{I}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,\alpha),\mathsf{now}))$

Given $\mathbf{K}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,treat),\mathsf{now}) \Rightarrow holds(alive(wife(\mathsf{I}*)),t_0+T)\big)\big)$

Inferred $\mathbf{I}\big(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,treat),\mathsf{now})\big)$

*DCEC$_I$** Specimen from Heinz Dilemma

√ FOL

√ Epistemic + FOL

$\mathbf{B}_d\mathbf{B}_v\mathbf{B}_d V v$

TOL

$\exists X[X(j) \wedge \neg X(m) \wedge S(X)]$

Given $\mathbf{B}\left(\mathsf{I}, \mathsf{now}, \forall t : \mathsf{Moment}, a : \mathsf{Agent}\left(holds(sick(a),t) \wedge \left(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(a),t+t')\right)\right.\right.$

$\left.\left.\Rightarrow \left(happens(dies(a),t+T) \vee holds(dead(a),t+T)\right)\right)\right)$

Given $\mathbf{K}\left(\mathsf{I}, \mathsf{now}, holds(sick(wife(\mathsf{I}*)),t_0) \wedge \left(\forall t' : \mathsf{Moment}\ t' < T \Rightarrow \neg happens(treated(wife(\mathsf{I}*)),t+t')\right)\right)$

Inferred $\mathbf{B}(\mathsf{I}, \mathsf{now}, happens(dies(wife(\mathsf{I}*)),t_0+T) \vee holds(dead(wife(\mathsf{I}*)),t_0+T))$

Given $\mathbf{K}(\mathsf{I}, \mathsf{now}, \mathsf{EventCalculus} \Rightarrow$

$(happens(dies(wife(\mathsf{I}*)),t_0+T) \vee holds(dead(wife(\mathsf{I}*)),t_0+T) \Rightarrow$

$\neg holds(alive(wife(\mathsf{I}*)),t_0+T)))$

Inferred $\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(alive(wife(\mathsf{I}*)),t_0+T))$

Given $\mathbf{D}(\mathsf{I}, \mathsf{now}, holds(alive(wife(\mathsf{I}*)),t_0+T))$

Given $(\mathbf{B}(\mathsf{I}, \mathsf{now}, \neg holds(f,t)) \wedge \mathbf{D}(\mathsf{I}, \mathsf{now}, holds(f,t)) \wedge$

$\mathbf{K}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,\alpha),\mathsf{now}) \Rightarrow holds(f,t)))$

$\Rightarrow \mathbf{I}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,\alpha),\mathsf{now}))$

Given $\mathbf{K}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,treat),\mathsf{now}) \Rightarrow holds(alive(wife(\mathsf{I}*)),t_0+T)))$

Inferred $\mathbf{I}(\mathsf{I}, \mathsf{now}, happens(action(\mathsf{I}*,treat),\mathsf{now}))$

$DCEC_I$* Specimen from Heinz Dilemma

Given $\mathbf{B}\Big(\text{I}, \text{now}, \forall t : \text{Moment}, a : \text{Agent}\Big(holds(sick(a),t) \land \big(\forall t' : \text{Moment}\ t' < T \Rightarrow \neg happens(treated(a), t+t')\big)$

$\Rightarrow \big(happens(dies(a), t+T) \lor holds(dead(a), t+T)\big)\Big)\Big)$

Given $\mathbf{K}\big(\text{I}, \text{now}, holds(sick(wife(\text{I}*)), t_0) \land \big(\forall t' : \text{Moment}\ t' < T \Rightarrow \neg happens(treated(wife(\text{I}*)), t+t')\big)\big)$

Inferred $\mathbf{B}\big(\text{I}, \text{now}, happens(dies(wife(\text{I}*)), t_0+T) \lor holds(dead(wife(\text{I}*)), t_0+T)\big)$

Given $\mathbf{K}\big(\text{I}, \text{now}, \text{EventCalculus} \Rightarrow$

$\big(happens(dies(wife(\text{I}*)), t_0+T) \lor holds(dead(wife(\text{I}*)), t_0+T) \Rightarrow$

$\neg holds(alive(wife(\text{I}*)), t_0+T)\big)\big)$

Inferred $\mathbf{B}\big(\text{I}, \text{now}, \neg holds(alive(wife(\text{I}*)), t_0+T)\big)$

Given $\mathbf{D}\big(\text{I}, \text{now}, holds(alive(wife(\text{I}*)), t_0+T)\big)$

Given $\big(\mathbf{B}(\text{I}, \text{now}, \neg holds(f,t)) \land \mathbf{D}(\text{I}, \text{now}, holds(f,t)) \land$

$\mathbf{K}(\text{I}, \text{now}, happens(action(\text{I}*, \alpha), \text{now}) \Rightarrow holds(f,t))\big)$

$\Rightarrow \mathbf{I}(\text{I}, \text{now}, happens(action(\text{I}*, \alpha), \text{now}))$

Given $\mathbf{K}\big(\text{I}, \text{now}, happens(action(\text{I}*, treat), \text{now}) \Rightarrow holds(alive(wife(\text{I}*)), t_0+T)\big)\big)$

Inferred $\mathbf{I}\big(\text{I}, \text{now}, happens(action(\text{I}*, treat), \text{now})\big)$

√ FOL

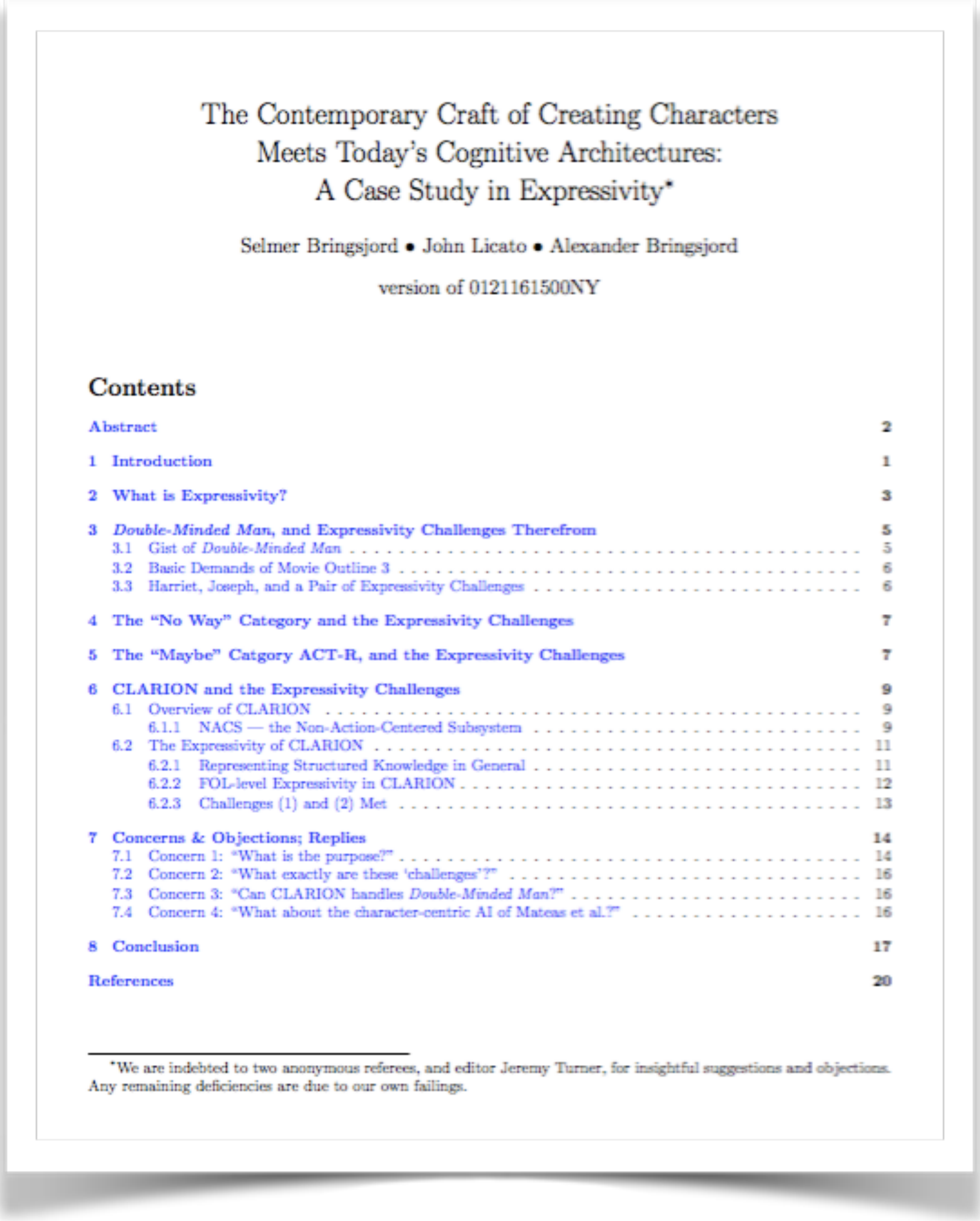
√ Epistemic + FOL

$\mathbf{B}_d\mathbf{B}_v\mathbf{B}_d V v$

✗ TOL

$\exists X[X(j) \land \neg X(m) \land S(X)]$

Double-Minded Man



The Contemporary Craft of Creating Characters Meets Today's Cognitive Architectures: A Case Study in Expressivity[*]

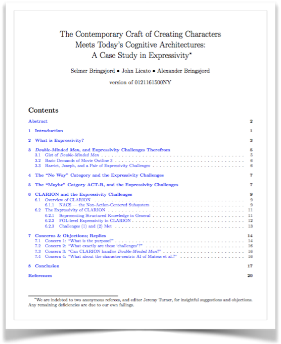
Selmer Bringsjord • John Licato • Alexander Bringsjord

version of 0121161500NY

Contents

[*] We are indebted to two anonymous referees, and editor Jeremy Turner, for insightful suggestions and objections. Any remaining deficiencies are due to our own failings.

Double-Minded Man



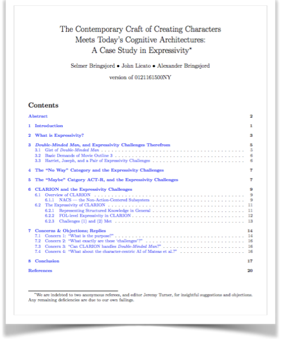
The Contemporary Craft of Creating Characters
Meets Today's Cognitive Architectures:
A Case Study in Expressivity[*]

Selmer Bringsjord • John Licato • Alexander Bringsjord

version of 0121161500NY

Contents

[*] We are indebted to two anonymous referees, and editor Jeremy Turner, for insightful suggestions and objections. Any remaining deficiencies are due to our own failings.



The Contemporary Craft of Creating Characters
Meets Today's Cognitive Architectures:
A Case Study in Expressivity*

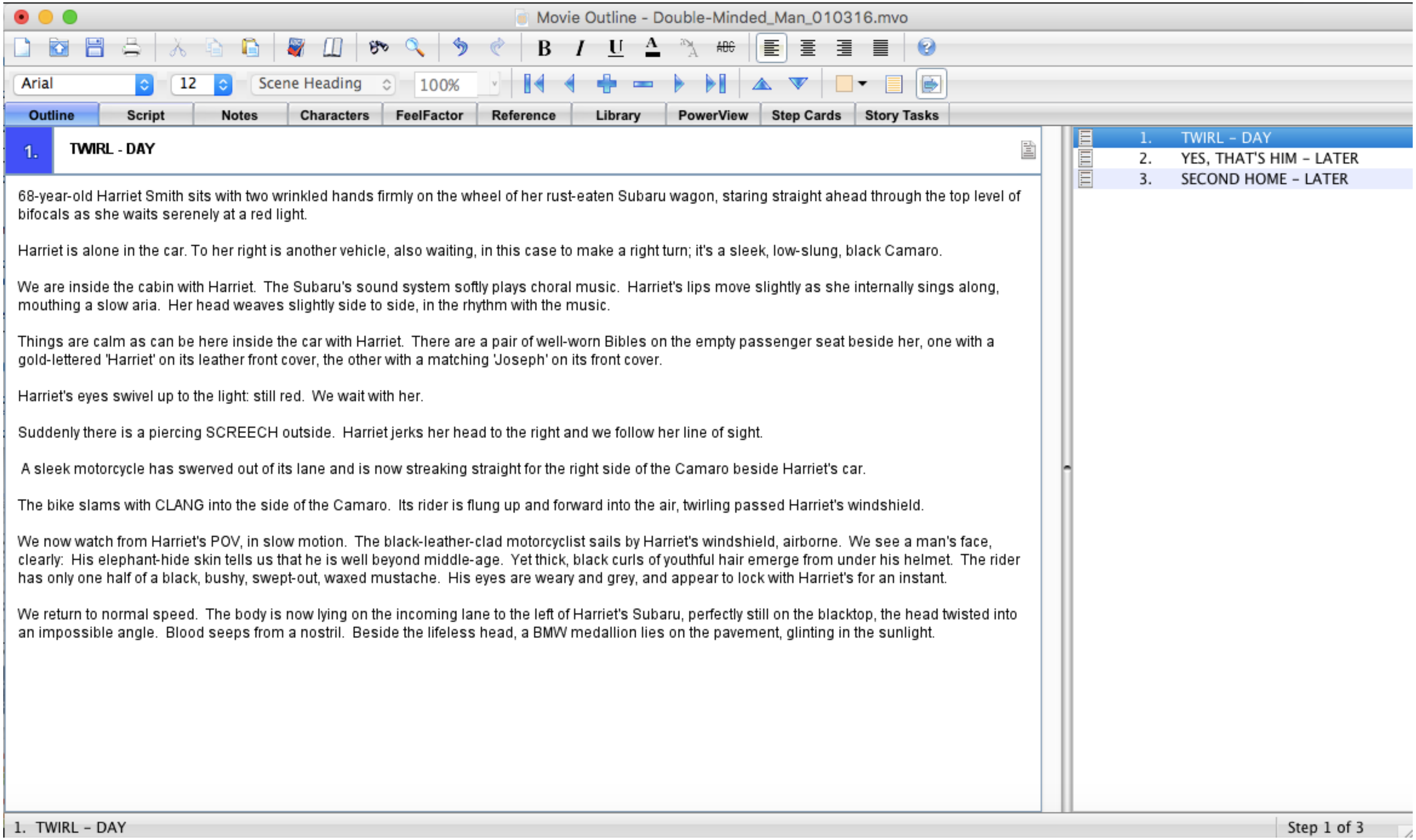
Selmer Bringsjord • John Licato • Alexander Bringsjord

version of 0121161500NY

Contents

[*] We are indebted to two anonymous referees, and editor Jeremy Turner, for insightful suggestions and objections. Any remaining deficiencies are due to our own failings.

Double-Minded Man



1. TWIRL - DAY

68-year-old Harriet Smith sits with two wrinkled hands firmly on the wheel of her rust-eaten Subaru wagon, staring straight ahead through the top level of bifocals as she waits serenely at a red light.

Harriet is alone in the car. To her right is another vehicle, also waiting, in this case to make a right turn; it's a sleek, low-slung, black Camaro.

We are inside the cabin with Harriet. The Subaru's sound system softly plays choral music. Harriet's lips move slightly as she internally sings along, mouthing a slow aria. Her head weaves slightly side to side, in the rhythm with the music.

Things are calm as can be here inside the car with Harriet. There are a pair of well-worn Bibles on the empty passenger seat beside her, one with a gold-lettered 'Harriet' on its leather front cover, the other with a matching 'Joseph' on its front cover.

Harriet's eyes swivel up to the light: still red. We wait with her.

Suddenly there is a piercing SCREECH outside. Harriet jerks her head to the right and we follow her line of sight.

A sleek motorcycle has swerved out of its lane and is now streaking straight for the right side of the Camaro beside Harriet's car.

The bike slams with CLANG into the side of the Camaro. Its rider is flung up and forward into the air, twirling passed Harriet's windshield.

We now watch from Harriet's POV, in slow motion. The black-leather-clad motorcyclist sails by Harriet's windshield, airborne. We see a man's face, clearly: His elephant-hide skin tells us that he is well beyond middle-age. Yet thick, black curls of youthful hair emerge from under his helmet. The rider has only one half of a black, bushy, swept-out, waxed mustache. His eyes are weary and grey, and appear to lock with Harriet's for an instant.

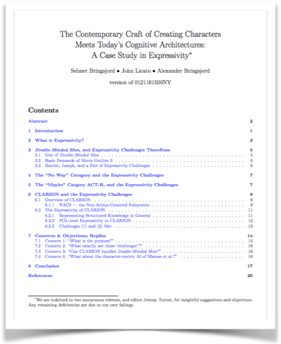
We return to normal speed. The body is now lying on the incoming lane to the left of Harriet's Subaru, perfectly still on the blacktop, the head twisted into an impossible angle. Blood seeps from a nostril. Beside the lifeless head, a BMW medallion lies on the pavement, glinting in the sunlight.

1. TWIRL – DAY
2. YES, THAT'S HIM – LATER
3. SECOND HOME – LATER

1. TWIRL – DAY

Step 1 of 3

Double-Minded Man



The Contemporary Craft of Creating Characters
Meets Today's Cognitive Architectures:
A Case Study in Expressivity*

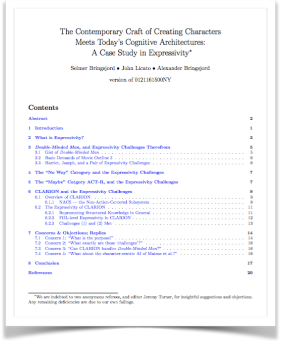
Selmer Bringsjord • John Licato • Alexander Bringsjord

version of 0121161500NY

Contents

[*] We are indebted to two anonymous referees, and editor Jeremy Turner, for insightful suggestions and objections. Any remaining deficiencies are due to our own failings.

Double-Minded Man



The Contemporary Craft of Creating Characters
Meets Today's Cognitive Architectures:
A Case Study in Expressivity*

Selmer Bringsjord • John Licato • Alexander Bringsjord

version of 0121161500NY

Contents

*We are indebted to two anonymous referees, and editor Jeremy Turner, for insightful suggestions and objections. Any remaining deficiencies are due to our own failings.

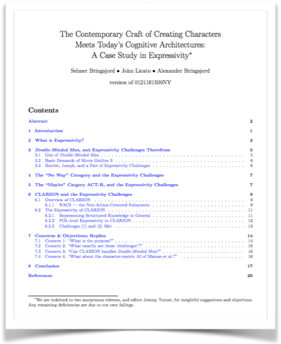
Double-Minded Man
by

S Bringsjord & A Bringsjord

DRAFT #5
© June 30 2016

Selmer.Bringsjord@gmail.com

Double-Minded Man



The Contemporary Craft of Creating Characters
Meets Today's Cognitive Architectures:
A Case Study in Expressivity[*]

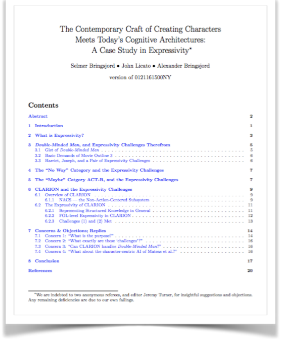
Selmer Bringsjord • John Licato • Alexander Bringsjord

version of 0121161500NY

Contents

[*] We are indebted to two anonymous referees, and editor Jeremy Turner, for insightful suggestions and objections. Any remaining deficiencies are due to our own failings.

Double-Minded Man



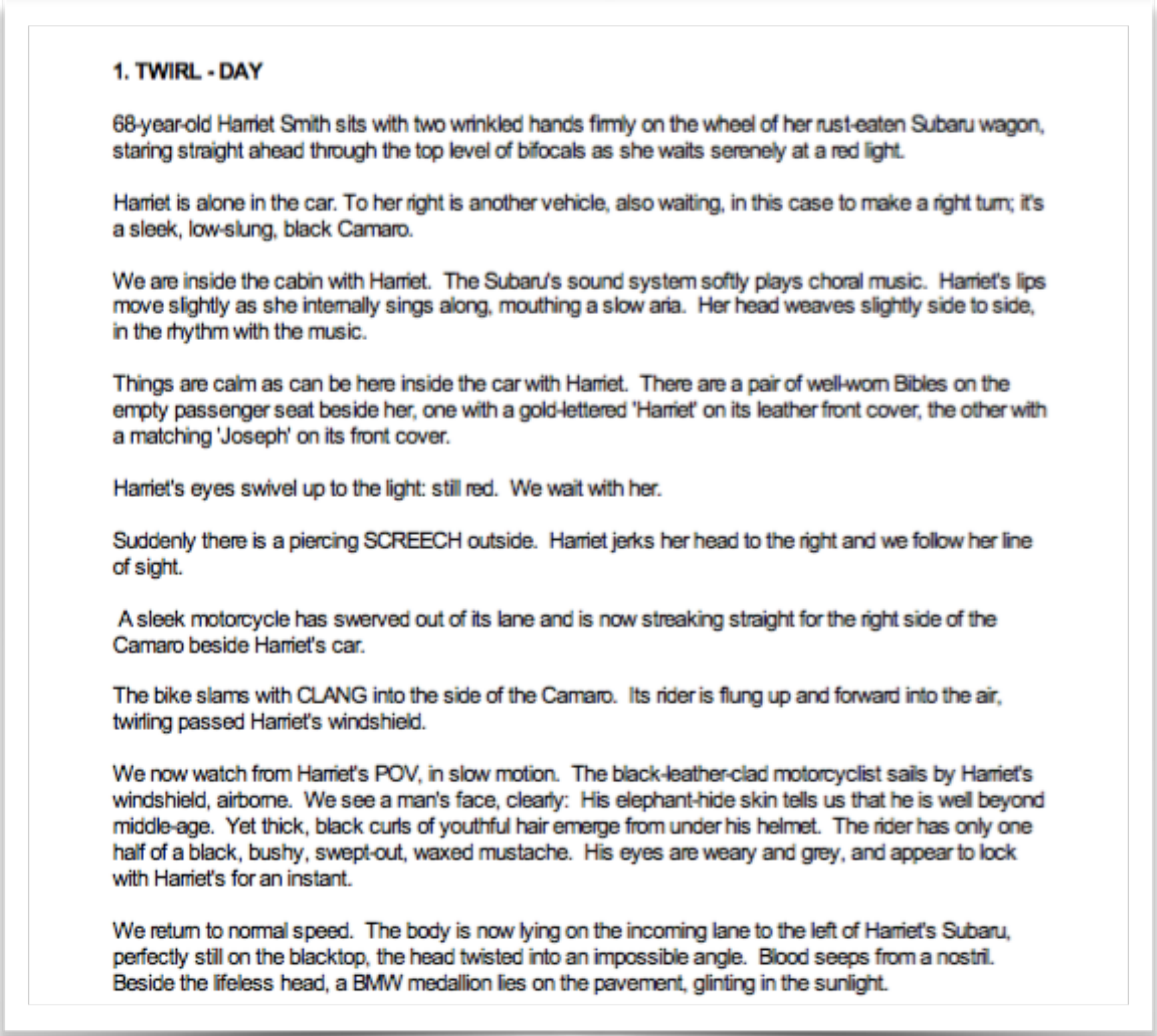
The Contemporary Craft of Creating Characters
Meets Today's Cognitive Architectures:
A Case Study in Expressivity[*]

Selmer Bringsjord • John Licato • Alexander Bringsjord

version of 0121161500NY

Contents

[*] We are indebted to two anonymous referees, and editor Jeremy Turner, for insightful suggestions and objections. Any remaining deficiencies are due to our own failings.



1. TWIRL - DAY

68-year-old Harriet Smith sits with two wrinkled hands firmly on the wheel of her rust-eaten Subaru wagon, staring straight ahead through the top level of bifocals as she waits serenely at a red light.

Harriet is alone in the car. To her right is another vehicle, also waiting, in this case to make a right turn; it's a sleek, low-slung, black Camaro.

We are inside the cabin with Harriet. The Subaru's sound system softly plays choral music. Harriet's lips move slightly as she internally sings along, mouthing a slow aria. Her head weaves slightly side to side, in the rhythm with the music.

Things are calm as can be here inside the car with Harriet. There are a pair of well-worn Bibles on the empty passenger seat beside her, one with a gold-lettered 'Harriet' on its leather front cover, the other with a matching 'Joseph' on its front cover.

Harriet's eyes swivel up to the light: still red. We wait with her.

Suddenly there is a piercing SCREECH outside. Harriet jerks her head to the right and we follow her line of sight.

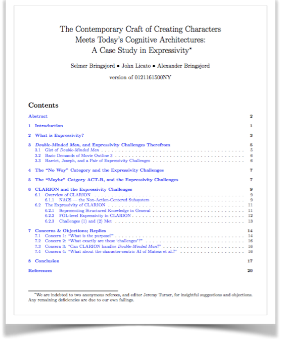
A sleek motorcycle has swerved out of its lane and is now streaking straight for the right side of the Camaro beside Harriet's car.

The bike slams with CLANG into the side of the Camaro. Its rider is flung up and forward into the air, twirling passed Harriet's windshield.

We now watch from Harriet's POV, in slow motion. The black-leather-clad motorcyclist sails by Harriet's windshield, airborne. We see a man's face, clearly: His elephant-hide skin tells us that he is well beyond middle-age. Yet thick, black curls of youthful hair emerge from under his helmet. The rider has only one half of a black, bushy, swept-out, waxed mustache. His eyes are weary and grey, and appear to lock with Harriet's for an instant.

We return to normal speed. The body is now lying on the incoming lane to the left of Harriet's Subaru, perfectly still on the blacktop, the head twisted into an impossible angle. Blood seeps from a nostril. Beside the lifeless head, a BMW medallion lies on the pavement, glinting in the sunlight.

Double-Minded Man



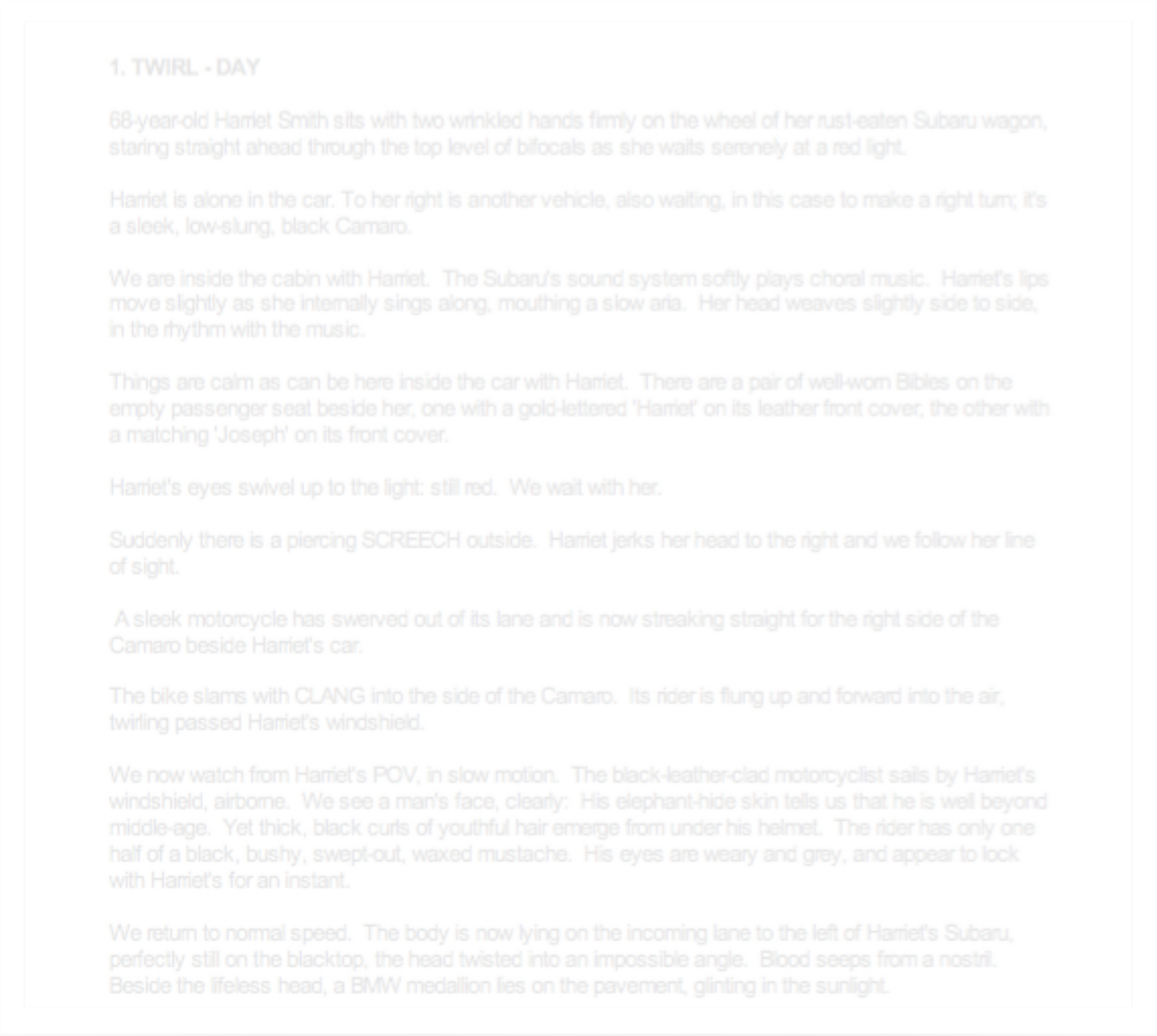
The Contemporary Craft of Creating Characters
Meets Today's Cognitive Architectures:
A Case Study in Expressivity*

Selmer Bringsjord • John Licato • Alexander Bringsjord

version of 0121161500NY

Contents

*We are indebted to two anonymous referees, and editor Jeremy Turner, for insightful suggestions and objections. Any remaining deficiencies are due to our own failings.



1. TWIRL - DAY

68-year-old Harriet Smith sits with two wrinkled hands firmly on the wheel of her rust-eaten Subaru wagon, staring straight ahead through the top level of bifocals as she waits serenely at a red light.

Harriet is alone in the car. To her right is another vehicle, also waiting, in this case to make a right turn; it's a sleek, low-slung, black Camaro.

We are inside the cabin with Harriet. The Subaru's sound system softly plays choral music. Harriet's lips move slightly as she internally sings along, mouthing a slow aria. Her head weaves slightly side to side, in the rhythm with the music.

Things are calm as can be here inside the car with Harriet. There are a pair of well-worn Bibles on the empty passenger seat beside her, one with a gold-lettered 'Harriet' on its leather front cover, the other with a matching 'Joseph' on its front cover.

Harriet's eyes swivel up to the light: still red. We wait with her.

Suddenly there is a piercing SCREECH outside. Harriet jerks her head to the right and we follow her line of sight.

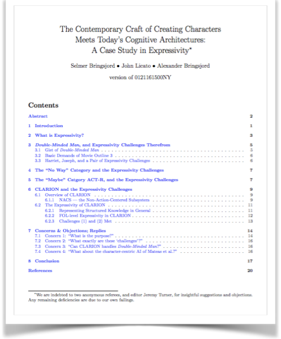
A sleek motorcycle has swerved out of its lane and is now streaking straight for the right side of the Camaro beside Harriet's car.

The bike slams with CLANG into the side of the Camaro. Its rider is flung up and forward into the air, twirling passed Harriet's windshield.

We now watch from Harriet's POV, in slow motion. The black-leather-clad motorcyclist sails by Harriet's windshield, airborne. We see a man's face, clearly: His elephant-hide skin tells us that he is well beyond middle-age. Yet thick, black curls of youthful hair emerge from under his helmet. The rider has only one half of a black, bushy, swept-out, waxed mustache. His eyes are weary and grey, and appear to lock with Harriet's for an instant.

We return to normal speed. The body is now lying on the incoming lane to the left of Harriet's Subaru, perfectly still on the blacktop, the head twisted into an impossible angle. Blood seeps from a nostril. Beside the lifeless head, a BMW medallion lies on the pavement, glinting in the sunlight.

Double-Minded Man



The Contemporary Craft of Creating Characters
Meets Today's Cognitive Architectures:
A Case Study in Expressivity*

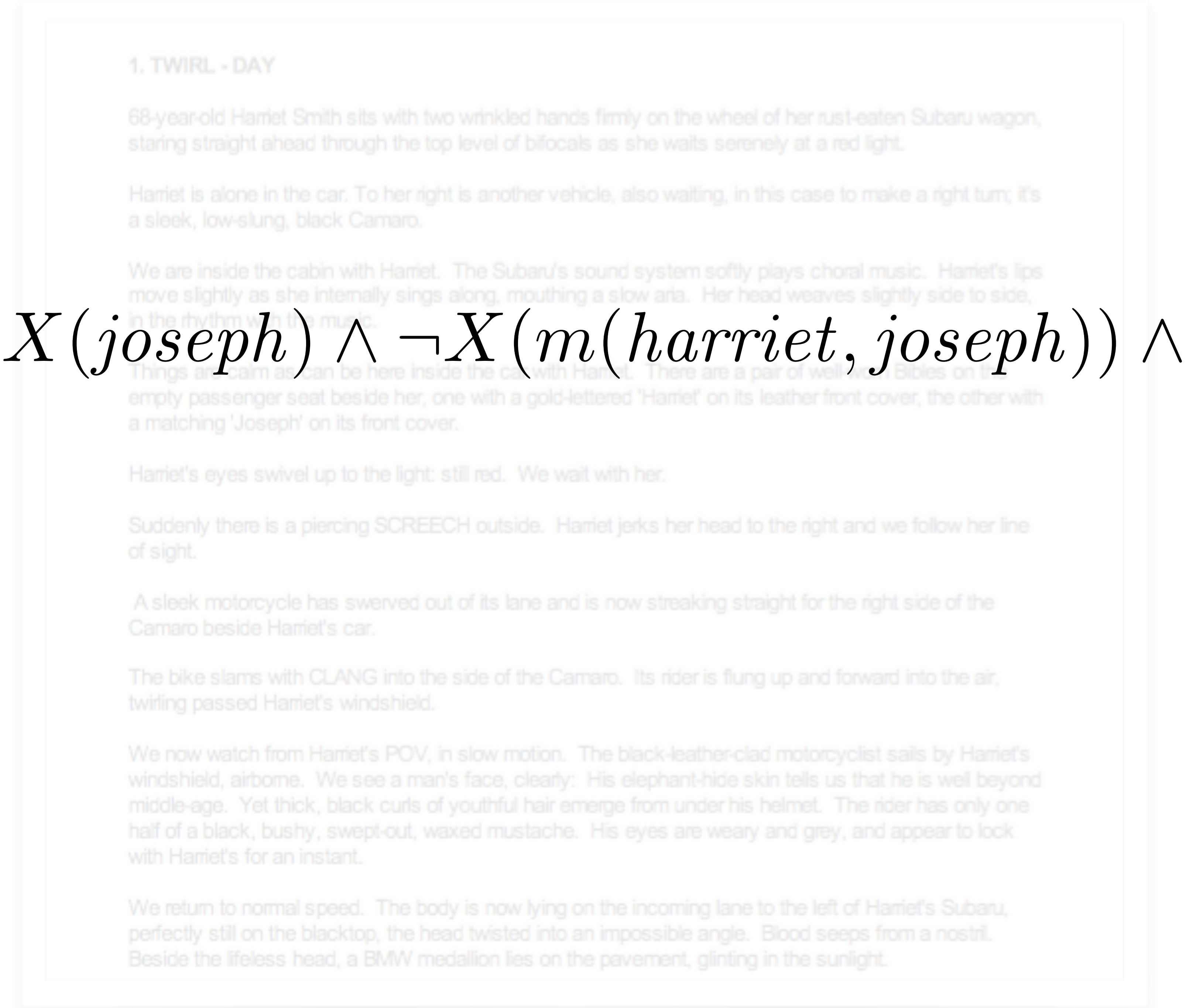
Selmer Bringsjord • John Licato • Alexander Bringsjord

version of 0121161500NY

Contents

[*] We are indebted to two anonymous referees, and editor Jeremy Turner, for insightful suggestions and objections. Any remaining deficiencies are due to our own failings.

$$\exists X[X(joseph) \land \neg X(m(harriet, joseph)) \land Sleazy(X)]$$



1. TWIRL - DAY

68-year-old Harriet Smith sits with two wrinkled hands firmly on the wheel of her rust-eaten Subaru wagon, staring straight ahead through the top level of bifocals as she waits serenely at a red light.

Harriet is alone in the car. To her right is another vehicle, also waiting, in this case to make a right turn; it's a sleek, low-slung, black Camaro.

We are inside the cabin with Harriet. The Subaru's sound system softly plays choral music. Harriet's lips move slightly as she internally sings along, mouthing a slow aria. Her head weaves slightly side to side, in the rhythm with the music.

Things are calm as can be here inside the car with Harriet. There are a pair of well-worn Bibles on the empty passenger seat beside her, one with a gold-lettered 'Harriet' on its leather front cover, the other with a matching 'Joseph' on its front cover.

Harriet's eyes swivel up to the light: still red. We wait with her.

Suddenly there is a piercing SCREECH outside. Harriet jerks her head to the right and we follow her line of sight.

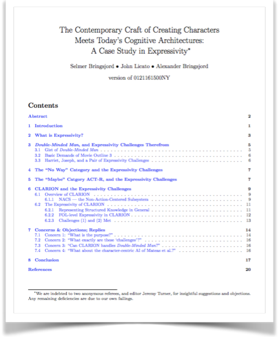
A sleek motorcycle has swerved out of its lane and is now streaking straight for the right side of the Camaro beside Harriet's car.

The bike slams with CLANG into the side of the Camaro. Its rider is flung up and forward into the air, twirling passed Harriet's windshield.

We now watch from Harriet's POV, in slow motion. The black-leather-clad motorcyclist sails by Harriet's windshield, airborne. We see a man's face, clearly: His elephant-hide skin tells us that he is well beyond middle-age. Yet thick, black curls of youthful hair emerge from under his helmet. The rider has only one half of a black, bushy, swept-out, waxed mustache. His eyes are weary and grey, and appear to lock with Harriet's for an instant.

We return to normal speed. The body is now lying on the incoming lane to the left of Harriet's Subaru, perfectly still on the blacktop, the head twisted into an impossible angle. Blood seeps from a nostril. Beside the lifeless head, a BMW medallion lies on the pavement, glinting in the sunlight.

Double-Minded Man



The Contemporary Craft of Creating Characters Meets Today's Cognitive Architectures: A Case Study in Expressivity*

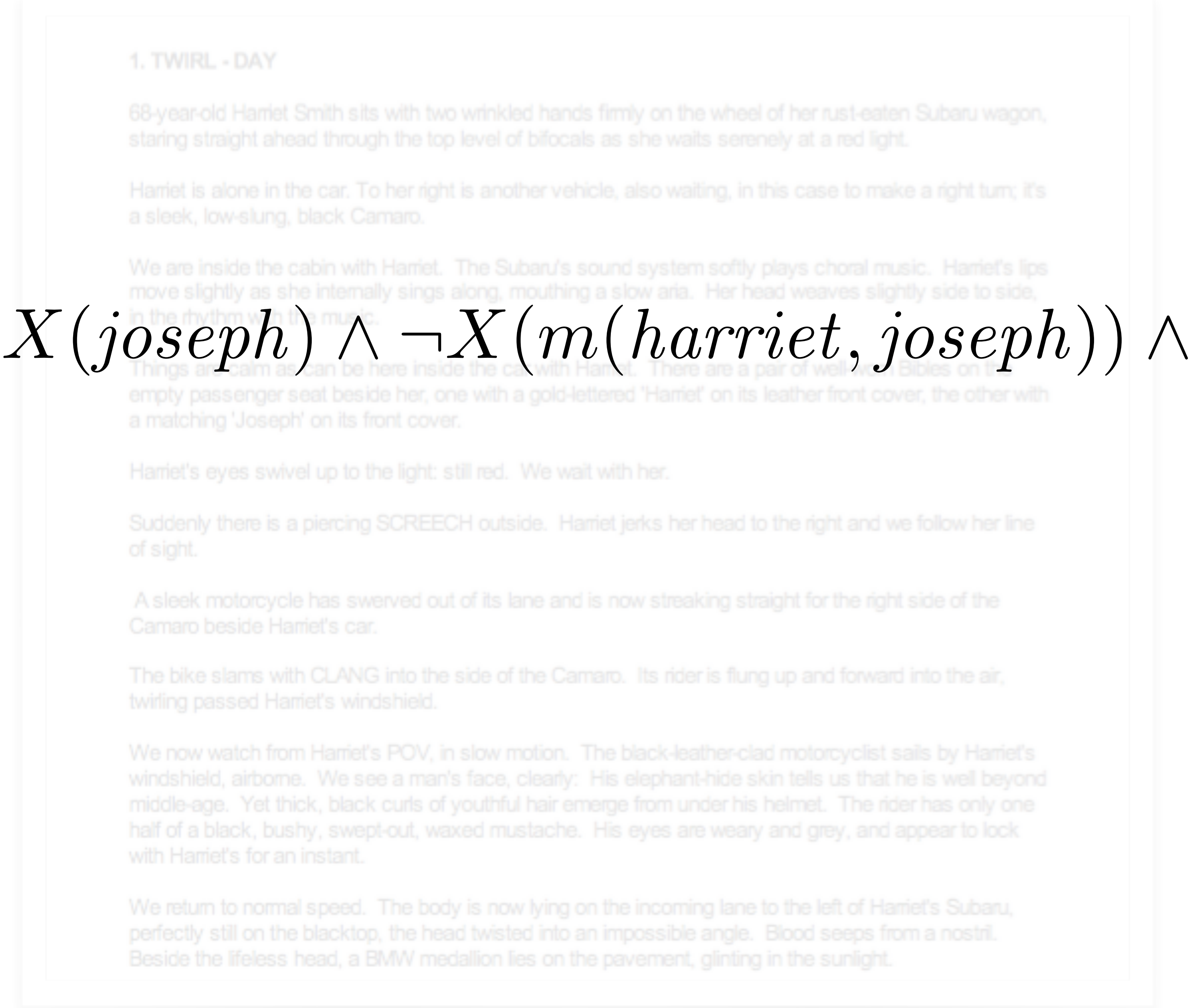
Selmer Bringsjord • John Licato • Alexander Bringsjord

version of 0121161500NY

Contents

*We are indebted to two anonymous referees, and editor Jeremy Turner, for insightful suggestions and objections. Any remaining deficiencies are due to our own failings.

? $\exists X[X(joseph) \land \neg X(m(harriet, joseph)) \land Sleazy(X)]$



1. TWIRL - DAY

68-year-old Harriet Smith sits with two wrinkled hands firmly on the wheel of her rust-eaten Subaru wagon, staring straight ahead through the top level of bifocals as she waits serenely at a red light.

Harriet is alone in the car. To her right is another vehicle, also waiting, in this case to make a right turn; it's a sleek, low-slung, black Camaro.

We are inside the cabin with Harriet. The Subaru's sound system softly plays choral music. Harriet's lips move slightly as she internally sings along, mouthing a slow aria. Her head weaves slightly side to side, in the rhythm with the music.

Things are calm as can be here inside the cab with Harriet. There are a pair of well-worn Bibles on the empty passenger seat beside her, one with a gold-lettered 'Harriet' on its leather front cover, the other with a matching 'Joseph' on its front cover.

Harriet's eyes swivel up to the light: still red. We wait with her.

Suddenly there is a piercing SCREECH outside. Harriet jerks her head to the right and we follow her line of sight.

A sleek motorcycle has swerved out of its lane and is now streaking straight for the right side of the Camaro beside Harriet's car.

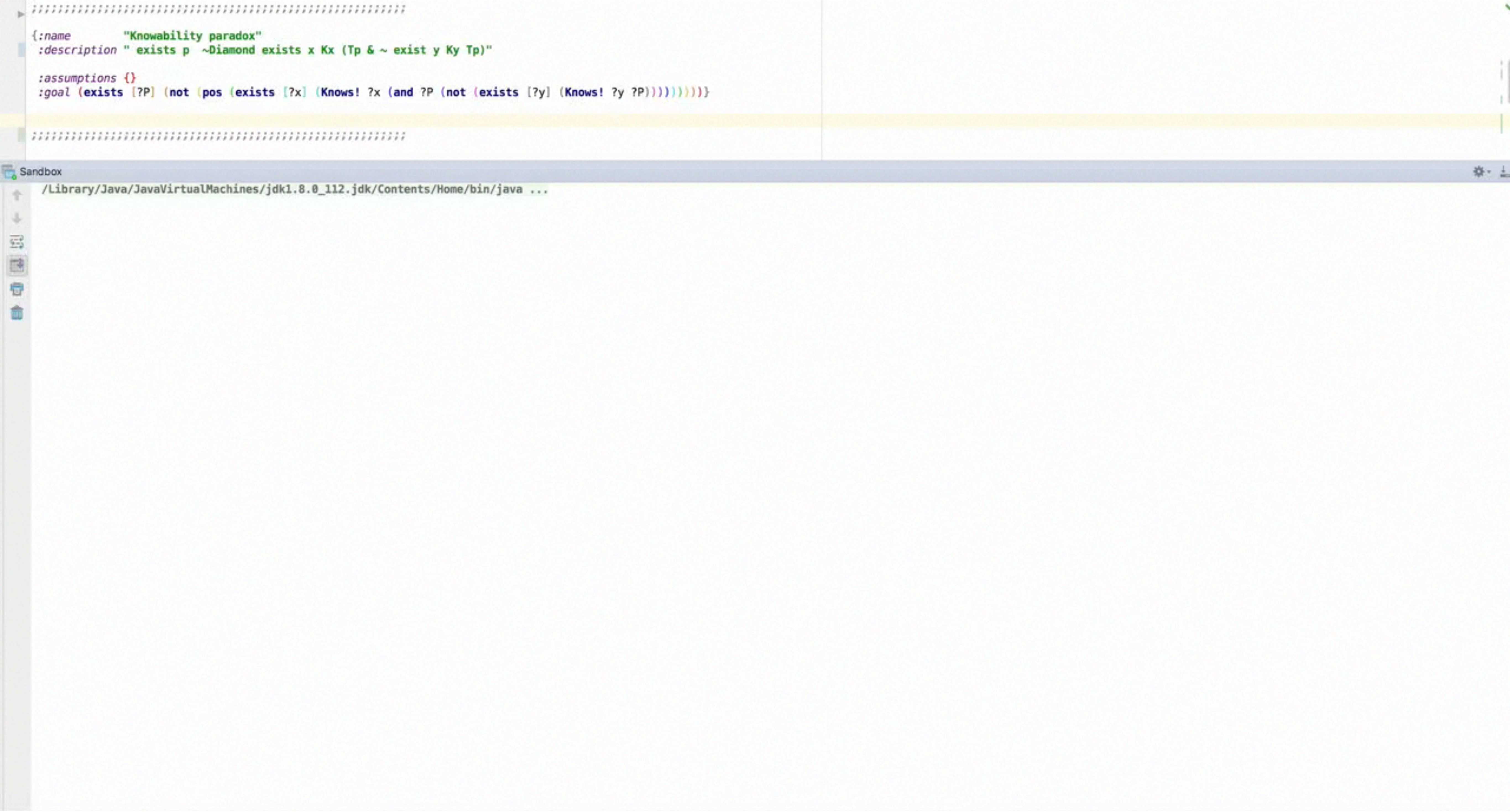
The bike slams with CLANG into the side of the Camaro. Its rider is flung up and forward into the air, twirling passed Harriet's windshield.

We now watch from Harriet's POV, in slow motion. The black-leather-clad motorcyclist sails by Harriet's windshield, airborne. We see a man's face, clearly: His elephant-hide skin tells us that he is well beyond middle-age. Yet thick, black curls of youthful hair emerge from under his helmet. The rider has only one half of a black, bushy, swept-out, waxed mustache. His eyes are weary and grey, and appear to lock with Harriet's for an instant.

We return to normal speed. The body is now lying on the incoming lane to the left of Harriet's Subaru, perfectly still on the blacktop, the head twisted into an impossible angle. Blood seeps from a nostril. Beside the lifeless head, a BMW medallion lies on the pavement, glinting in the sunlight.

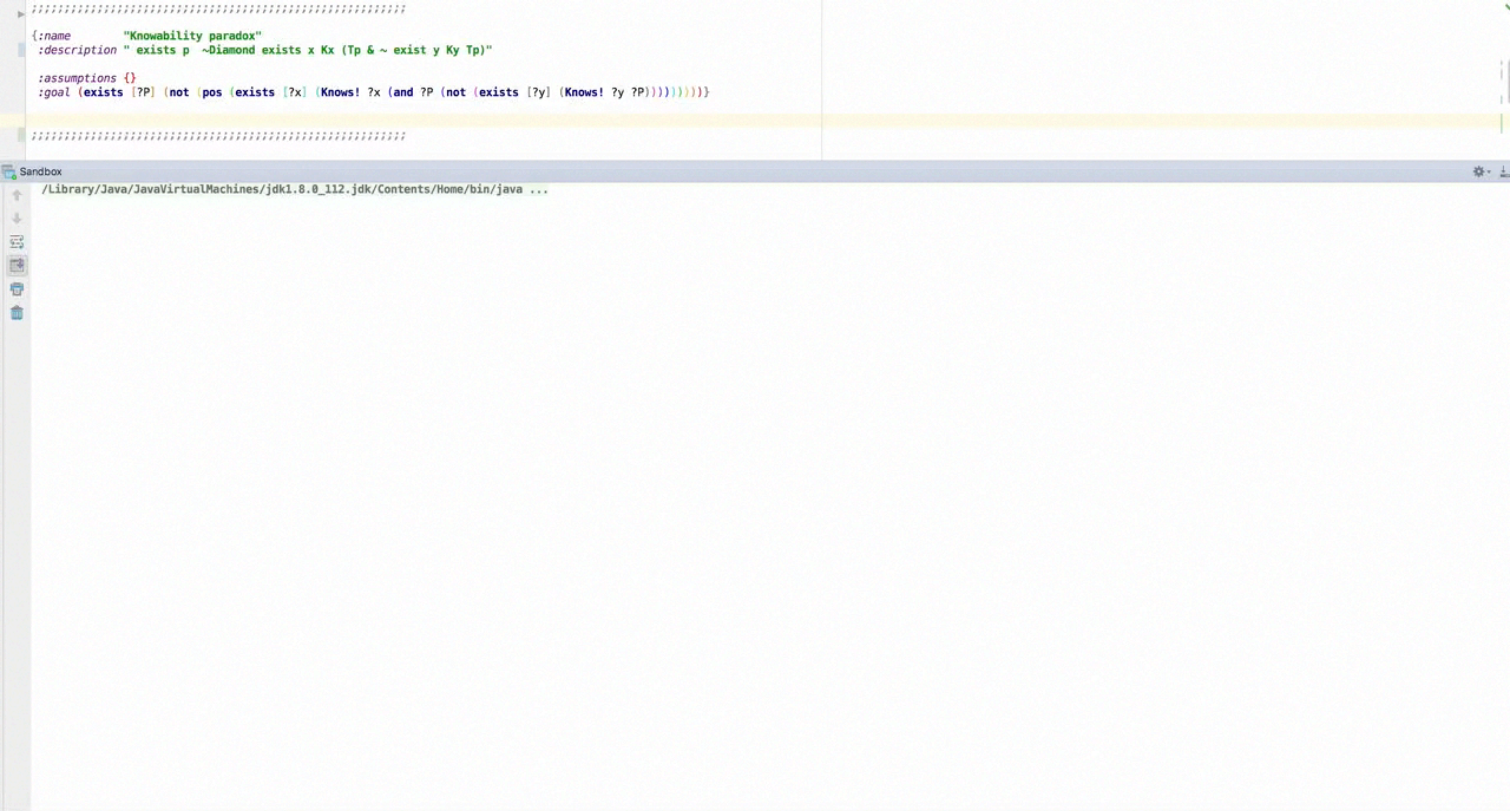
$$\mu\mathcal{DCEC}_3^* \in \mathcal{CC}$$

$$\mu\mathcal{DCEC}_3^* \in \mathcal{CC}$$



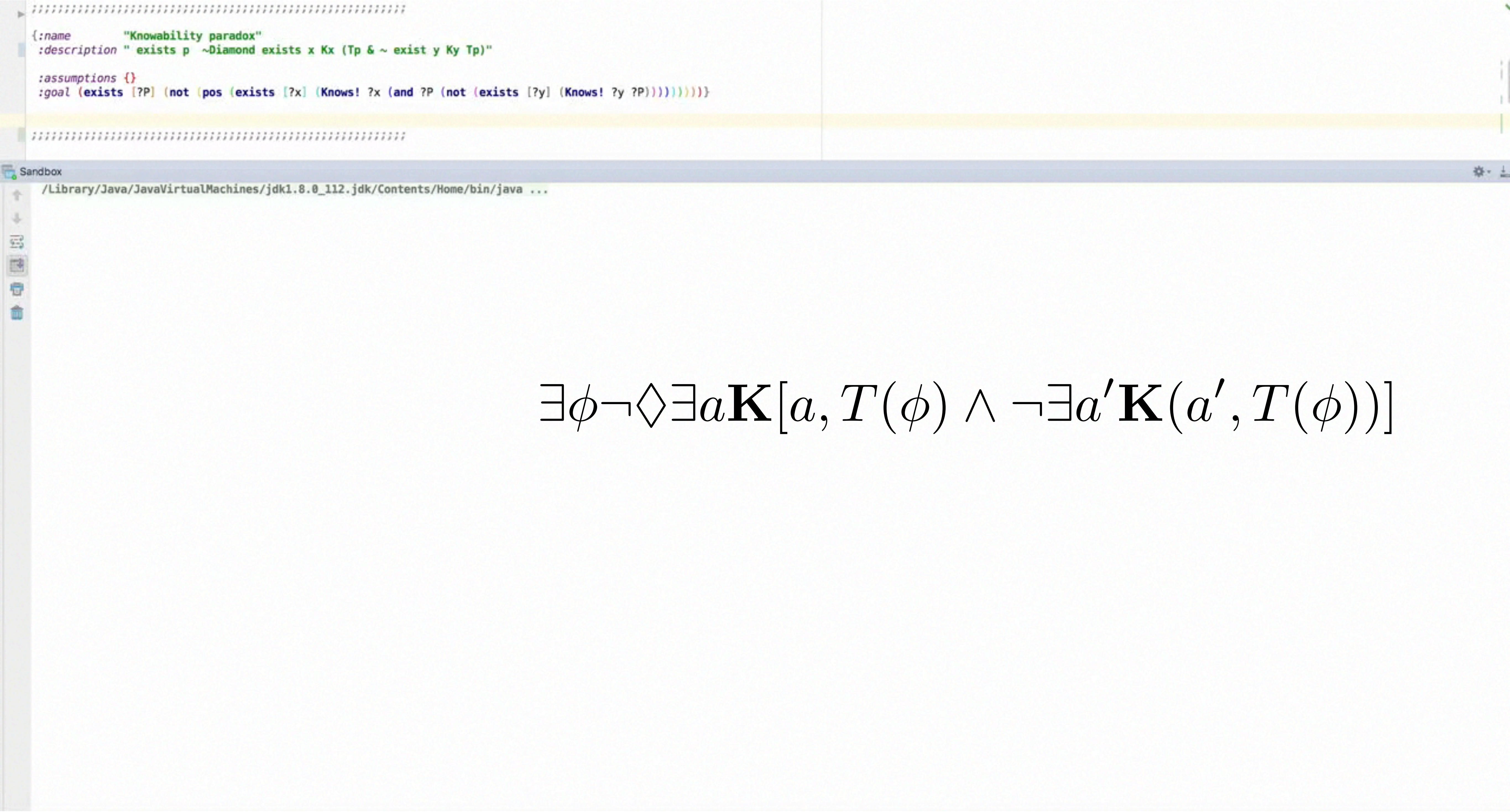
```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

{:name        "Knowability paradox"
 :description " exists p  ~Diamond exists x Kx (Tp & ~ exist y Ky Tp)"

 :assumptions {}
 :goal (exists [?P] (not (pos (exists [?x] (Knows! ?x (and ?P (not (exists [?y] (Knows! ?y ?P)))))))))}


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
/Library/Java/JavaVirtualMachines/jdk1.8.0_112.jdk/Contents/Home/bin/java ...
```



```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

{:name        "Knowability paradox"
 :description " exists p  ~Diamond exists x Kx (Tp & ~ exist y Ky Tp)"

 :assumptions {}
 :goal (exists [?P] (not (pos (exists [?x] (Knows! ?x (and ?P (not (exists [?y] (Knows! ?y ?P)))))))))}


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
/Library/Java/JavaVirtualMachines/jdk1.8.0_112.jdk/Contents/Home/bin/java ...
```



```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

{:name        "Knowability paradox"
 :description " exists p  ~Diamond exists x Kx (Tp & ~ exist y Ky Tp)"

 :assumptions {}
 :goal (exists [?P] (not (pos (exists [?x] (Knows! ?x (and ?P (not (exists [?y] (Knows! ?y ?P)))))))))}

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

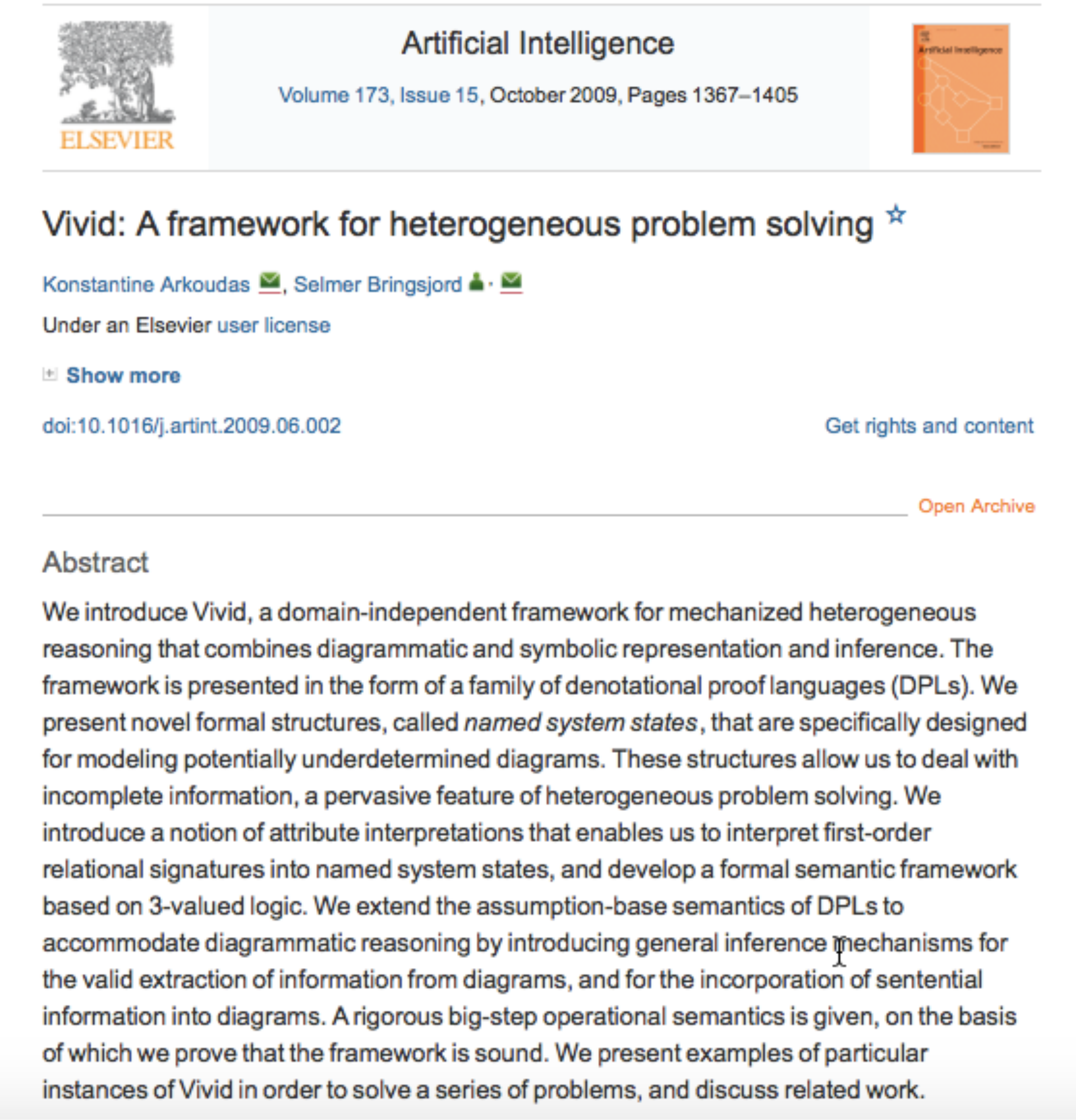
```
/Library/Java/JavaVirtualMachines/jdk1.8.0_112.jdk/Contents/Home/bin/java ...
```

$$\exists \phi \neg \Diamond \exists a \mathbf{K}[a, T(\phi) \wedge \neg \exists a' \mathbf{K}(a', T(\phi))]$$

$$\mu\mathcal{DCEC}^*_3 \in \mathcal{CC}$$

$$\overset{\triangle}{\mu}\mathcal{DCEC}_3^* \in \mathcal{CC}$$

$$\overset{\Delta}{\mu}\mathcal{DCEC}^*_3 \in \mathcal{CC}$$



ELSEVIER

Artificial Intelligence

Volume 173, Issue 15, October 2009, Pages 1367–1405

Vivid: A framework for heterogeneous problem solving ☆

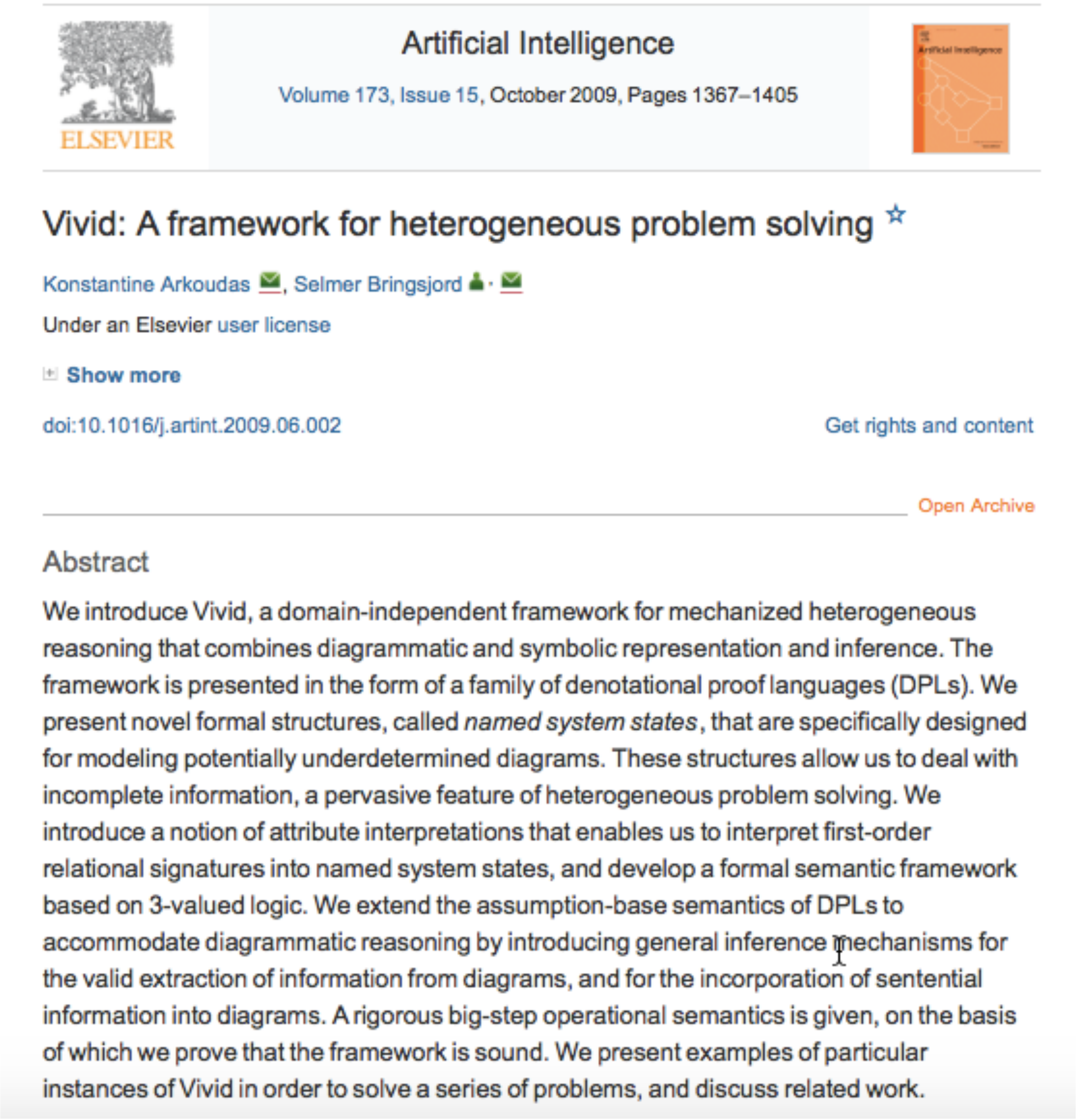
Konstantine Arkoudas ✉, Selmer Bringsjord ✉

doi:10.1016/j.artint.2009.06.002

Abstract

We introduce Vivid, a domain-independent framework for mechanized heterogeneous reasoning that combines diagrammatic and symbolic representation and inference. The framework is presented in the form of a family of denotational proof languages (DPLs). We present novel formal structures, called *named system states*, that are specifically designed for modeling potentially underdetermined diagrams. These structures allow us to deal with incomplete information, a pervasive feature of heterogeneous problem solving. We introduce a notion of attribute interpretations that enables us to interpret first-order relational signatures into named system states, and develop a formal semantic framework based on 3-valued logic. We extend the assumption-base semantics of DPLs to accommodate diagrammatic reasoning by introducing general inference mechanisms for the valid extraction of information from diagrams, and for the incorporation of sentential information into diagrams. A rigorous big-step operational semantics is given, on the basis of which we prove that the framework is sound. We present examples of particular instances of Vivid in order to solve a series of problems, and discuss related work.

Formally fancy, but capability-wise, what's it good for??



ELSEVIER

Artificial Intelligence

Volume 173, Issue 15, October 2009, Pages 1367–1405

Vivid: A framework for heterogeneous problem solving ☆

Konstantine Arkoudas ✉, Selmer Bringsjord ✉

⊞ Show more

doi:10.1016/j.artint.2009.06.002

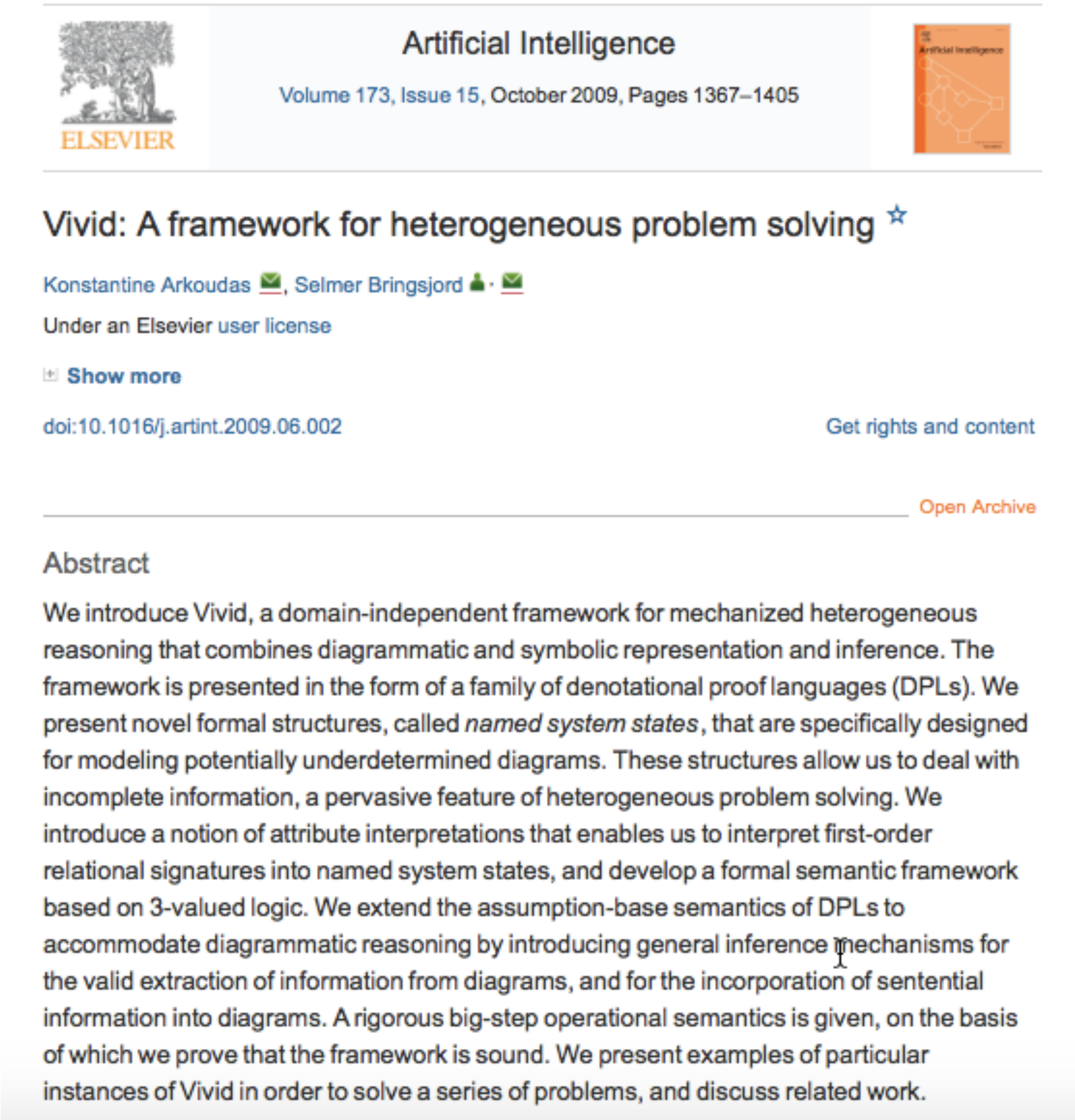
Get rights and content

Open Archive

Abstract

We introduce Vivid, a domain-independent framework for mechanized heterogeneous reasoning that combines diagrammatic and symbolic representation and inference. The framework is presented in the form of a family of denotational proof languages (DPLs). We present novel formal structures, called *named system states*, that are specifically designed for modeling potentially underdetermined diagrams. These structures allow us to deal with incomplete information, a pervasive feature of heterogeneous problem solving. We introduce a notion of attribute interpretations that enables us to interpret first-order relational signatures into named system states, and develop a formal semantic framework based on 3-valued logic. We extend the assumption-base semantics of DPLs to accommodate diagrammatic reasoning by introducing general inference mechanisms for the valid extraction of information from diagrams, and for the incorporation of sentential information into diagrams. A rigorous big-step operational semantics is given, on the basis of which we prove that the framework is sound. We present examples of particular instances of Vivid in order to solve a series of problems, and discuss related work.

Formally fancy, but capability-wise, what's it good for??



Artificial Intelligence

Volume 173, Issue 15, October 2009, Pages 1367–1405

Vivid: A framework for heterogeneous problem solving ☆

Konstantine Arkoudas ✉, Selmer Bringsjord ✉

Under an Elsevier user license

⊞ Show more

doi:10.1016/j.artint.2009.06.002

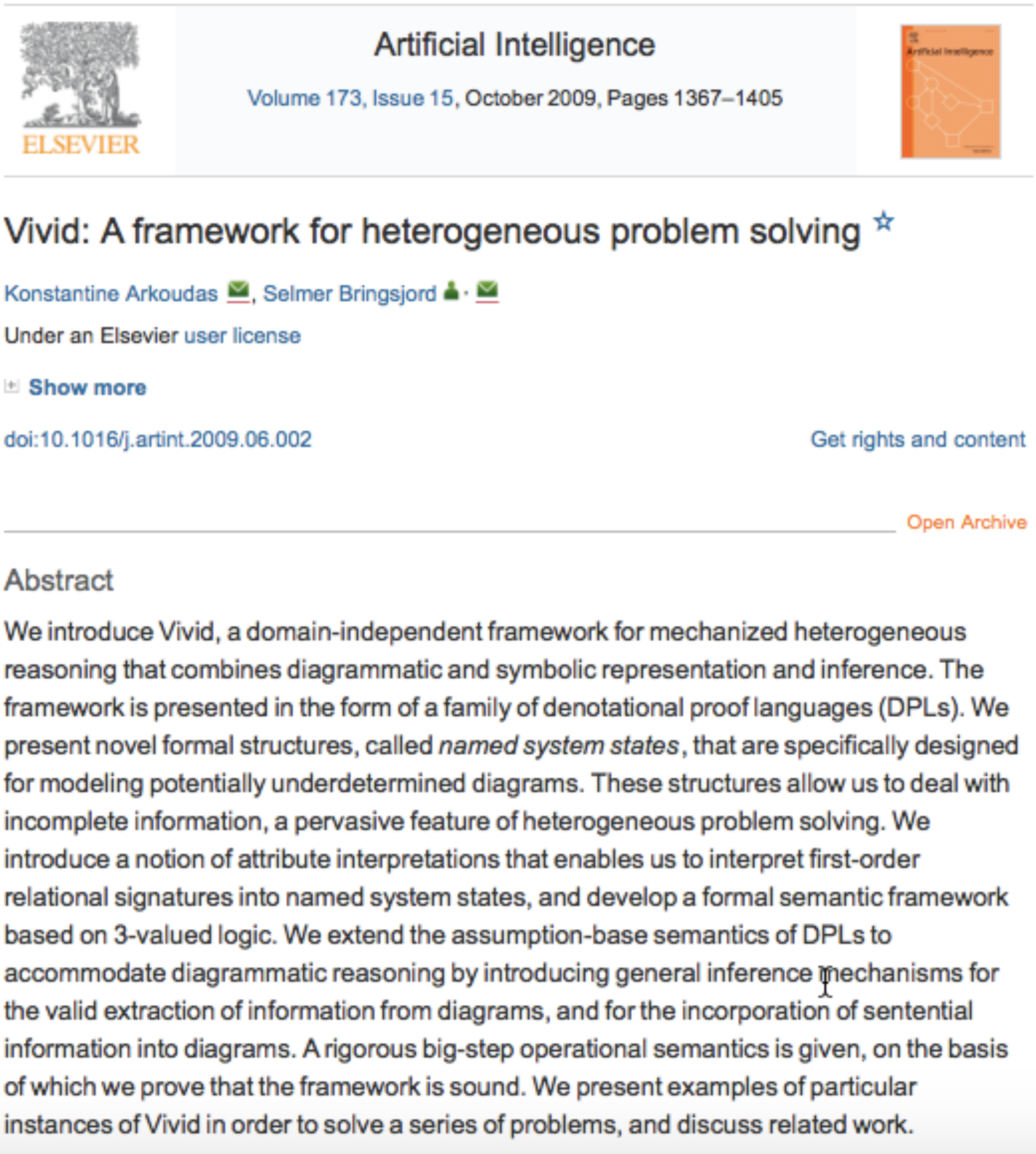
Get rights and content

Open Archive

Abstract

We introduce Vivid, a domain-independent framework for mechanized heterogeneous reasoning that combines diagrammatic and symbolic representation and inference. The framework is presented in the form of a family of denotational proof languages (DPLs). We present novel formal structures, called *named system states*, that are specifically designed for modeling potentially underdetermined diagrams. These structures allow us to deal with incomplete information, a pervasive feature of heterogeneous problem solving. We introduce a notion of attribute interpretations that enables us to interpret first-order relational signatures into named system states, and develop a formal semantic framework based on 3-valued logic. We extend the assumption-base semantics of DPLs to accommodate diagrammatic reasoning by introducing general inference mechanisms for the valid extraction of information from diagrams, and for the incorporation of sentential information into diagrams. A rigorous big-step operational semantics is given, on the basis of which we prove that the framework is sound. We present examples of particular instances of Vivid in order to solve a series of problems, and discuss related work.

Formally fancy, but capability-wise, what's it good for??



Artificial Intelligence

Volume 173, Issue 15, October 2009, Pages 1367–1405

Vivid: A framework for heterogeneous problem solving ☆

Konstantine Arkoudas, Selmer Bringsjord

Under an Elsevier user license

⊞ Show more

doi:10.1016/j.artint.2009.06.002

Get rights and content

Open Archive

Abstract

We introduce Vivid, a domain-independent framework for mechanized heterogeneous reasoning that combines diagrammatic and symbolic representation and inference. The framework is presented in the form of a family of denotational proof languages (DPLs). We present novel formal structures, called *named system states*, that are specifically designed for modeling potentially underdetermined diagrams. These structures allow us to deal with incomplete information, a pervasive feature of heterogeneous problem solving. We introduce a notion of attribute interpretations that enables us to interpret first-order relational signatures into named system states, and develop a formal semantic framework based on 3-valued logic. We extend the assumption-base semantics of DPLs to accommodate diagrammatic reasoning by introducing general inference mechanisms for the valid extraction of information from diagrams, and for the incorporation of sentential information into diagrams. A rigorous big-step operational semantics is given, on the basis of which we prove that the framework is sound. We present examples of particular instances of Vivid in order to solve a series of problems, and discuss related work.

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^{L}, O^{M}, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

$\mathcal{L}$ `labels`

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^{L}, O^{M}, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

- $\mathcal{L}$ `labels`
    - S `symbolic labels`
    - Δ `diagrammatic labels`

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^{L}, O^{M}, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

- $\mathcal{L}$ labels
  - S symbolic labels
  - Δ diagrammatic labels
- Γ background knowledge

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^{L}, O^{M}, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

- $\mathcal{L}$ `labels`
  - S `symbolic labels`
  - Δ `diagrammatic labels`
- Γ `background knowledge`
- $\mathcal{E}$ `ethics/norms`
  - F `forbidden`
  - O^{L} `legal/local prohibitions`
  - O^{M} `ethical prohibitions`
  - S^{up1} `civility`

$$\mathfrak{C} = \langle \mathcal{L} = \langle S, \Delta \rangle, \Gamma, \mathcal{E} = \langle F, O^L, O^M, S^{up1} \rangle, \Pi = \langle P, P' \rangle \rangle$$

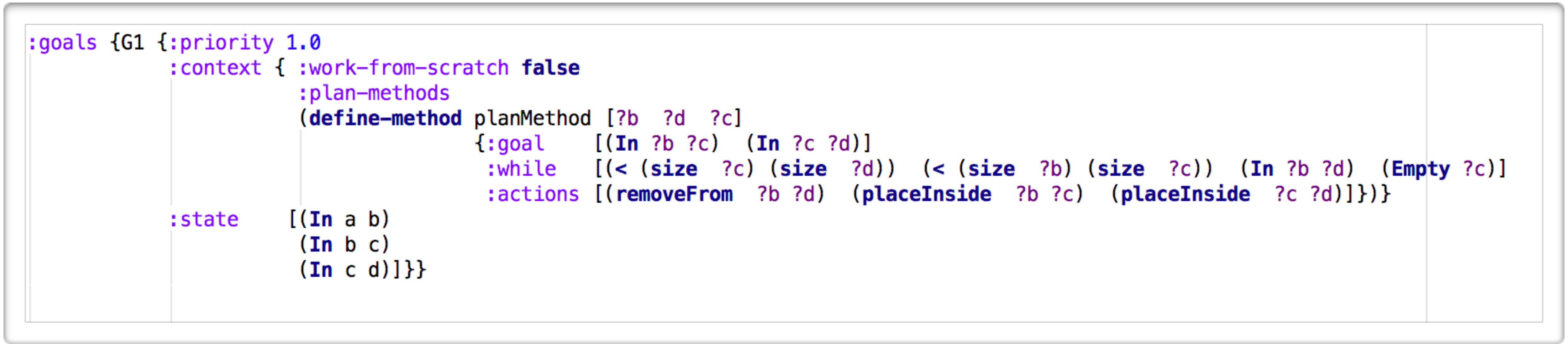
$\mathcal{L}$	labels	
	S	symbolic labels
	Δ	diagrammatic labels
Γ	background knowledge	
$\mathcal{E}$	ethics/norms	
	F	forbidden
	O^L	legal/local prohibitions
	O^M	ethical prohibitions
	S^{up1}	civility
Π	plans	
	P	plans
	P'	partial plans

Implementation (NSG!) …

ShadowProver

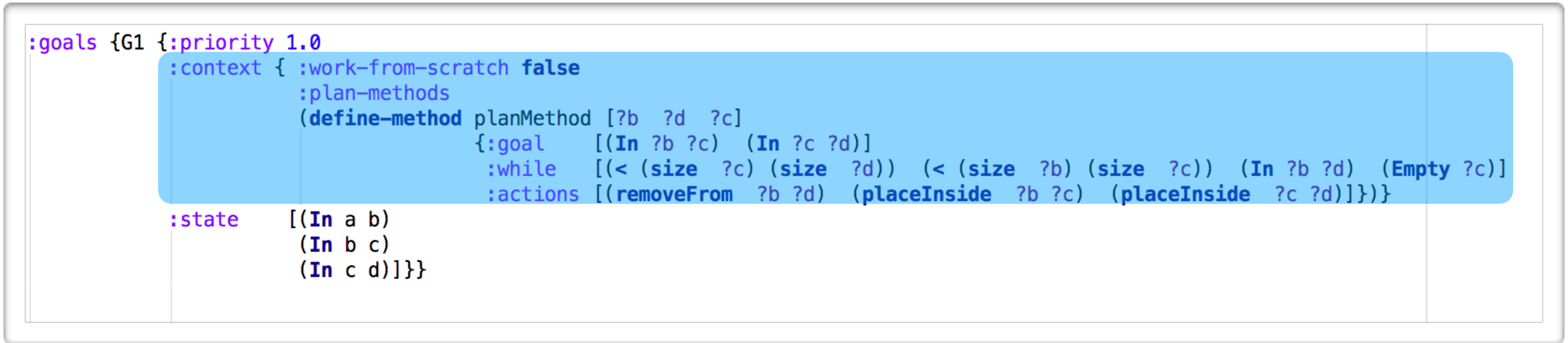
Spectra

Spectra: Planning with Goals under Contexts

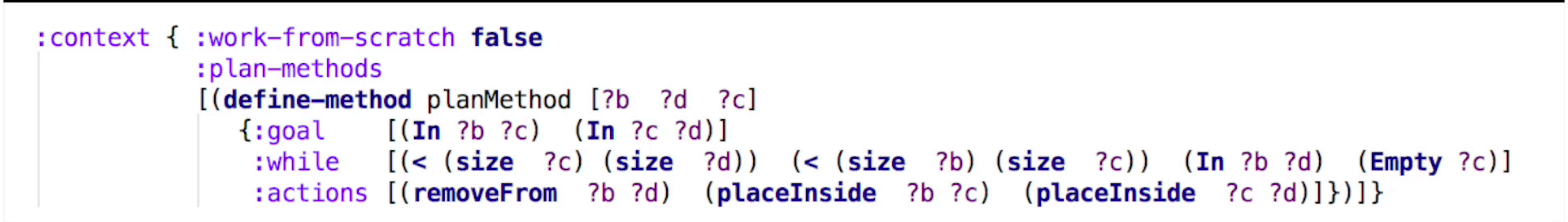


```
:goals {G1 {:priority 1.0
            :context { :work-from-scratch false
                       :plan-methods
                       (define-method planMethod [?b  ?d  ?c]
                                      {:goal    [(In ?b ?c)  (In ?c ?d)]
                                       :while   [(< (size  ?c) (size  ?d))  (< (size  ?b) (size  ?c))  (In ?b ?d)  (Empty ?c)]
                                       :actions [(removeFrom  ?b ?d)  (placeInside  ?b ?c)  (placeInside  ?c ?d)]})}
            :state    [(In a b)
                       (In b c)
                       (In c d)]}}
```

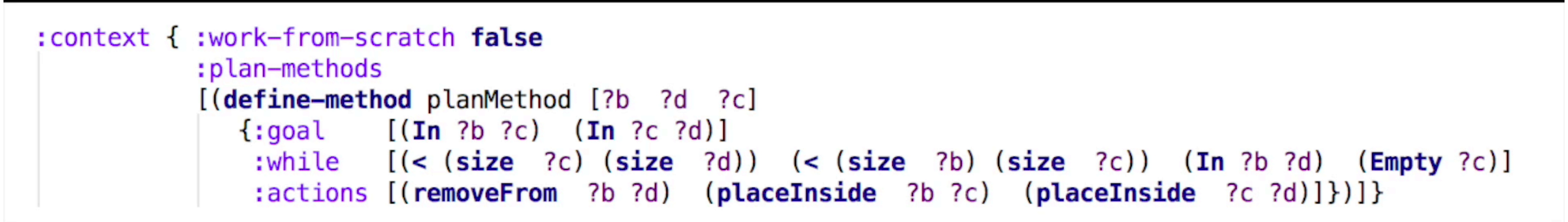
Spectra: Planning with Goals under Contexts



```
:goals {G1 {:priority 1.0
            :context { :work-from-scratch false
                       :plan-methods
                       (define-method planMethod [?b  ?d  ?c]
                                      {:goal    [(In ?b ?c)  (In ?c ?d)]
                                       :while   [(< (size  ?c) (size  ?d))  (< (size  ?b) (size  ?c))  (In ?b ?d)  (Empty ?c)]
                                       :actions [(removeFrom  ?b ?d)  (placeInside  ?b ?c)  (placeInside  ?c ?d)]})}
            :state     [(In a b)
                        (In b c)
                        (In c d)]}}
```



```
:context { :work-from-scratch false
           :plan-methods
           [(define-method planMethod [?b  ?d  ?c]
              {:goal    [(In ?b ?c)  (In ?c ?d)]
               :while   [(< (size  ?c) (size  ?d))  (< (size  ?b) (size  ?c))  (In ?b ?d)  (Empty ?c)]
               :actions [(removeFrom  ?b ?d)  (placeInside  ?b ?c)  (placeInside  ?c ?d)]})]}
```



```
:context { :work-from-scratch false
           :plan-methods
           [(define-method planMethod [?b  ?d  ?c]
              {:goal    [(In ?b ?c)  (In ?c ?d)]
               :while   [(< (size  ?c) (size  ?d))  (< (size  ?b) (size  ?c))  (In ?b ?d)  (Empty ?c)]
               :actions [(removeFrom  ?b ?d)  (placeInside  ?b ?c)  (placeInside  ?c ?d)]})]}
```

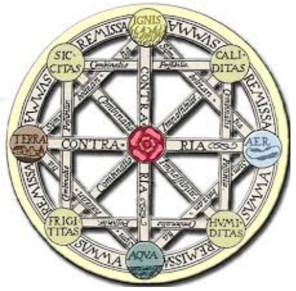
Logikk kan redde oss.

VI. New Paradigms …

VIa. A New, *Fine-Grained* Paradigm for Ethics Itself …

VIb.
The *Universal* Cognitive Calculus …

“Universal Cognitive Calculus”



1666

Leibniz

1.5 centuries < Boole!
2.5 centuries < Kripke

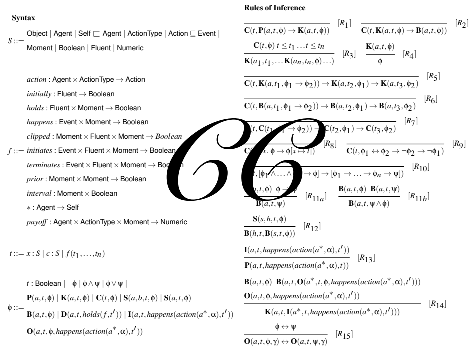
$\int$

Logic Theorist
(birth of modern logicist AI)

1956

Simon

$\mathcal{DCEC}^*$



Syntax

$$S ::= \text{Object} \mid \text{Agent} \mid \text{Self} \sqsubset \text{Agent} \mid \text{ActionType} \mid \text{Action} \sqsubseteq \text{Event} \mid \text{Moment} \mid \text{Boolean} \mid \text{Fluent} \mid \text{Numeric}$$

$$f ::= \begin{array}{l} \mathit{action} : \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \\ \mathit{initially} : \text{Fluent} \rightarrow \text{Boolean} \\ \mathit{holds} : \text{Fluent} \times \text{Moment} \rightarrow \text{Boolean} \\ \mathit{happens} : \text{Event} \times \text{Moment} \rightarrow \text{Boolean} \\ \mathit{clipped} : \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \mathit{Boolean} \\ \mathit{initiates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Boolean} \\ \mathit{terminates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Boolean} \\ \mathit{prior} : \text{Moment} \times \text{Moment} \rightarrow \text{Boolean} \\ \mathit{interval} : \text{Moment} \times \text{Boolean} \\ * : \text{Agent} \rightarrow \text{Self} \\ \mathit{payoff} : \text{Agent} \times \text{ActionType} \times \text{Moment} \rightarrow \text{Numeric} \end{array}$$

$$t ::= x : S \mid c : S \mid f(t_1, \ldots, t_n)$$

$$\phi ::= \begin{array}{l} t : \text{Boolean} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid \\ \mathbf{P}(a,t,\phi) \mid \mathbf{K}(a,t,\phi) \mid \mathbf{C}(t,\phi) \mid \mathbf{S}(a,b,t,\phi) \mid \mathbf{S}(a,t,\phi) \\ \mathbf{B}(a,t,\phi) \mid \mathbf{D}(a,t,\mathit{holds}(f,t')) \mid \mathbf{I}(a,t,\mathit{happens}(\mathit{action}(a^*,\alpha),t')) \\ \mathbf{O}(a,t,\phi,\mathit{happens}(\mathit{action}(a^*,\alpha),t')) \end{array}$$

Rules of Inference

$$\frac{}{\mathbf{C}(t,\mathbf{P}(a,t,\phi) \rightarrow \mathbf{K}(a,t,\phi))}\ [R_1] \qquad \frac{}{\mathbf{C}(t,\mathbf{K}(a,t,\phi) \rightarrow \mathbf{B}(a,t,\phi))}\ [R_2]$$

$$\frac{\mathbf{C}(t,\phi)\ t \leq t_1 \ldots t \leq t_n}{\mathbf{K}(a_1,t_1,\ldots \mathbf{K}(a_n,t_n,\phi)\ldots)}\ [R_3] \qquad \frac{\mathbf{K}(a,t,\phi)}{\phi}\ [R_4]$$

$$\frac{}{\mathbf{C}(t,\mathbf{K}(a,t_1,\phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{K}(a,t_2,\phi_1) \rightarrow \mathbf{K}(a,t_3,\phi_2)}\ [R_5]$$

$$\frac{}{\mathbf{C}(t,\mathbf{B}(a,t_1,\phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{B}(a,t_2,\phi_1) \rightarrow \mathbf{B}(a,t_3,\phi_2)}\ [R_6]$$

$$\frac{}{\mathbf{C}(t,\mathbf{C}(t_1,\phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{C}(t_2,\phi_1) \rightarrow \mathbf{C}(t_3,\phi_2)}\ [R_7]$$

$$\frac{}{\mathbf{C}(t,\forall x.\ \phi \rightarrow \phi[x \mapsto t])}\ [R_8] \qquad \frac{}{\mathbf{C}(t,\phi_1 \leftrightarrow \phi_2 \rightarrow \neg\phi_2 \rightarrow \neg\phi_1)}\ [R_9]$$

$$\frac{}{\mathbf{C}(t,[\phi_1 \wedge \ldots \wedge \phi_n \rightarrow \phi] \rightarrow [\phi_1 \rightarrow \ldots \rightarrow \phi_n \rightarrow \psi])}\ [R_{10}]$$

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

$$\frac{\mathbf{S}(s,h,t,\phi)}{\mathbf{B}(h,t,\mathbf{B}(s,t,\phi))}\ [R_{12}]$$

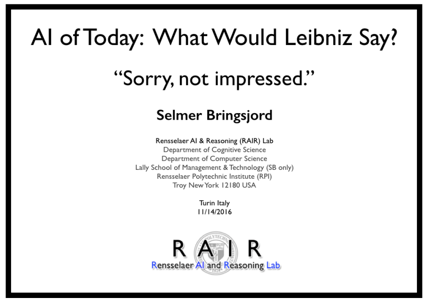
$$\frac{\mathbf{I}(a,t,\mathit{happens}(\mathit{action}(a^*,\alpha),t'))}{\mathbf{P}(a,t,\mathit{happens}(\mathit{action}(a^*,\alpha),t))}\ [R_{13}]$$

$$\frac{\begin{array}{c}\mathbf{B}(a,t,\phi)\quad \mathbf{B}(a,t,\mathbf{O}(a^*,t,\phi,\mathit{happens}(\mathit{action}(a^*,\alpha),t'))) \\ \mathbf{O}(a,t,\phi,\mathit{happens}(\mathit{action}(a^*,\alpha),t'))\end{array}}{\mathbf{K}(a,t,\mathbf{I}(a^*,t,\mathit{happens}(\mathit{action}(a^*,\alpha),t')))}\ [R_{14}]$$

$$\frac{\phi \leftrightarrow \psi}{\mathbf{O}(a,t,\phi,\gamma) \leftrightarrow \mathbf{O}(a,t,\psi,\gamma)}\ [R_{15}]$$

2017

R A I R
Rensselaer AI and Reasoning Lab



AI of Today: What Would Leibniz Say?

“Sorry, not impressed.”

Selmer Bringsjord

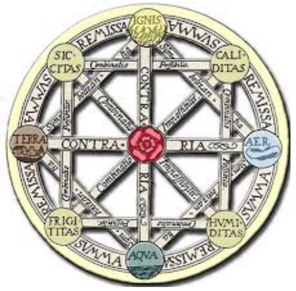
Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

R A I R
Rensselaer AI and Reasoning Lab

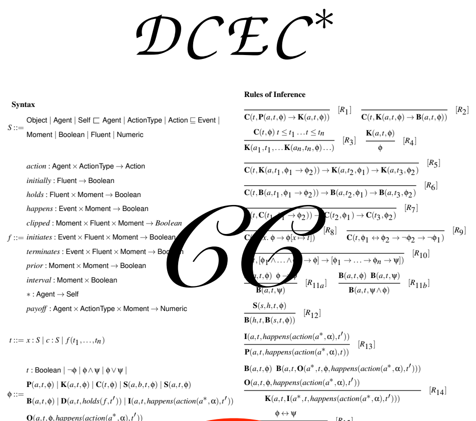
So, what do you think, Leibniz?

“Universal Cognitive Calculus”



Logic Theorist

(birth of modern logicist AI)



$\mathcal{DCEC}^*$

Syntax

$S ::=$ Object | Agent | Self $\sqsubset$ Agent | ActionType | Action $\sqsubseteq$ Event | Moment | Boolean | Fluent | Numeric

$f ::=$

action : Agent × ActionType → Action

initially : Fluent → Boolean

holds : Fluent × Moment → Boolean

happens : Event × Moment → Boolean

clipped : Moment × Fluent × Moment → *Boolean*

initiates : Event × Fluent × Moment → Boolean

terminates : Event × Fluent × Moment → Boolean

prior : Moment × Moment → Boolean

interval : Moment × Boolean

$*$: Agent → Self

payoff : Agent × ActionType × Moment → Numeric

$t ::= x : S \mid c : S \mid f(t_1, \ldots, t_n)$

$\phi ::=$ $t : \text{Boolean} \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid$ $\mathbf{P}(a,t,\phi) \mid \mathbf{K}(a,t,\phi) \mid \mathbf{C}(t,\phi) \mid \mathbf{S}(a,b,t,\phi) \mid \mathbf{S}(a,t,\phi)$ $\mid \mathbf{B}(a,t,\phi) \mid \mathbf{D}(a,t,holds(f,t')) \mid \mathbf{I}(a,t,happens(action(a^*,\alpha),t'))$ $\mid \mathbf{O}(a,t,\phi,happens(action(a^*,\alpha),t'))$

Rules of Inference

$$\frac{}{\mathbf{C}(t,\mathbf{P}(a,t,\phi)\rightarrow\mathbf{K}(a,t,\phi))}\ [R_1] \qquad \frac{}{\mathbf{C}(t,\mathbf{K}(a,t,\phi)\rightarrow\mathbf{B}(a,t,\phi))}\ [R_2]$$

$$\frac{\mathbf{C}(t,\phi)\ t\leq t_1 \ldots t \leq t_n}{\mathbf{K}(a_1,t_1,\ldots\mathbf{K}(a_n,t_n,\phi)\ldots)}\ [R_3] \qquad \frac{\mathbf{K}(a,t,\phi)}{\phi}\ [R_4]$$

$$\frac{}{\mathbf{C}(t,\mathbf{K}(a,t_1,\phi_1\rightarrow\phi_2))\rightarrow\mathbf{K}(a,t_2,\phi_1)\rightarrow\mathbf{K}(a,t_3,\phi_2)}\ [R_5]$$

$$\frac{}{\mathbf{C}(t,\mathbf{B}(a,t_1,\phi_1\rightarrow\phi_2))\rightarrow\mathbf{B}(a,t_2,\phi_1)\rightarrow\mathbf{B}(a,t_3,\phi_2)}\ [R_6]$$

$$\frac{}{\mathbf{C}(t,\mathbf{C}(t_1,\phi_1\rightarrow\phi_2))\rightarrow\mathbf{C}(t_2,\phi_1)\rightarrow\mathbf{C}(t_3,\phi_2)}\ [R_7]$$

$$\frac{}{\mathbf{C}(t,\forall x.\,\phi\rightarrow\phi[x\mapsto t])}\ [R_8] \qquad \frac{}{\mathbf{C}(t,\phi_1\leftrightarrow\phi_2\rightarrow\neg\phi_2\rightarrow\neg\phi_1)}\ [R_9]$$

$$\frac{}{\mathbf{C}(t,[\phi_1\wedge\ldots\wedge\phi_n\rightarrow\phi]\rightarrow[\phi_1\rightarrow\ldots\rightarrow\phi_n\rightarrow\psi])}\ [R_{10}]$$

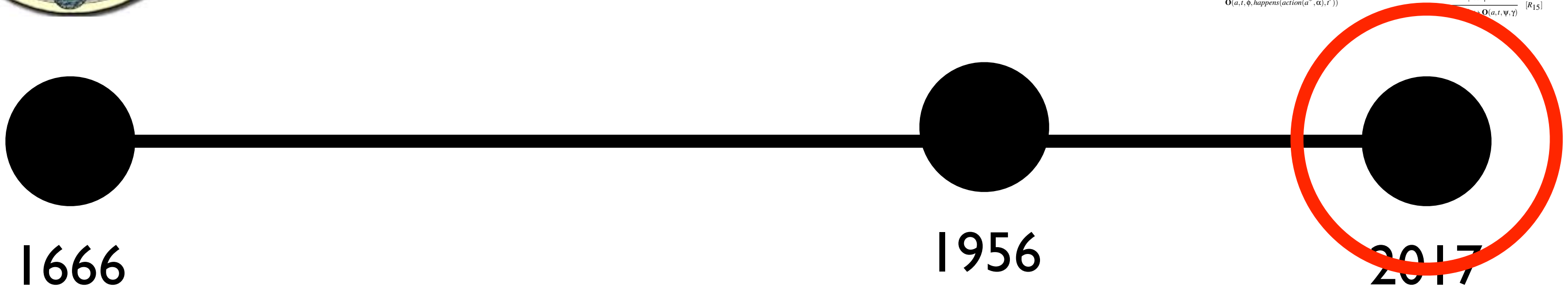
$$\frac{\mathbf{B}(a,t,\phi)\ \ \phi\rightarrow\psi}{\mathbf{B}(a,t,\psi)}\ [R_{11a}] \qquad \frac{\mathbf{B}(a,t,\phi)\ \ \mathbf{B}(a,t,\psi)}{\mathbf{B}(a,t,\psi\wedge\phi)}\ [R_{11b}]$$

$$\frac{\mathbf{S}(s,h,t,\phi)}{\mathbf{B}(h,t,\mathbf{B}(s,t,\phi))}\ [R_{12}]$$

$$\frac{\mathbf{I}(a,t,happens(action(a^*,\alpha),t'))}{\mathbf{P}(a,t,happens(action(a^*,\alpha),t))}\ [R_{13}]$$

$$\frac{\mathbf{B}(a,t,\phi)\ \ \mathbf{B}(a,t,\mathbf{O}(a^*,t,\phi,happens(action(a^*,\alpha),t')))\quad \mathbf{O}(a,t,\phi,happens(action(a^*,\alpha),t'))}{\mathbf{K}(a,t,\mathbf{I}(a^*,t,happens(action(a^*,\alpha),t')))}\ [R_{14}]$$

$$\frac{\phi\leftrightarrow\psi}{\mathbf{O}(a,t,\phi,\gamma)\leftrightarrow\mathbf{O}(a,t,\psi,\gamma)}\ [R_{15}]$$



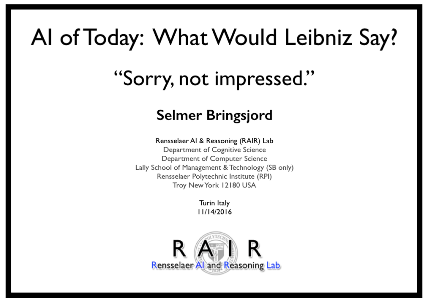
Leibniz

1.5 centuries < Boole!
2.5 centuries < Kripke

$\int$

Simon

R A I R
Rensselaer AI and Reasoning Lab



AI of Today: What Would Leibniz Say?

“Sorry, not impressed.”

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology (SB only)
Rensselaer Polytechnic Institute (RPI)
Troy New York 12180 USA

Turin Italy
11/14/2016

R A I R
Rensselaer AI and Reasoning Lab

L: Well, what are you proud of?

L: Well, what are you proud of?

AI: Hmm. Most recently, this:

L: Well, what are you proud of?

AI: Hmm. Most recently, this:

AlphaGo, via Deep Learning!!!

L: Well, what are you proud of?

AI: Hmm. Most recently, this:

AlphaGo, via Deep Learning!!!

L: But the game of Go is too easy …

L: AI is mired in mere calculation, AlphaGo being a case in point.

L: AI is mired in mere calculation, AlphaGo being a case in point.

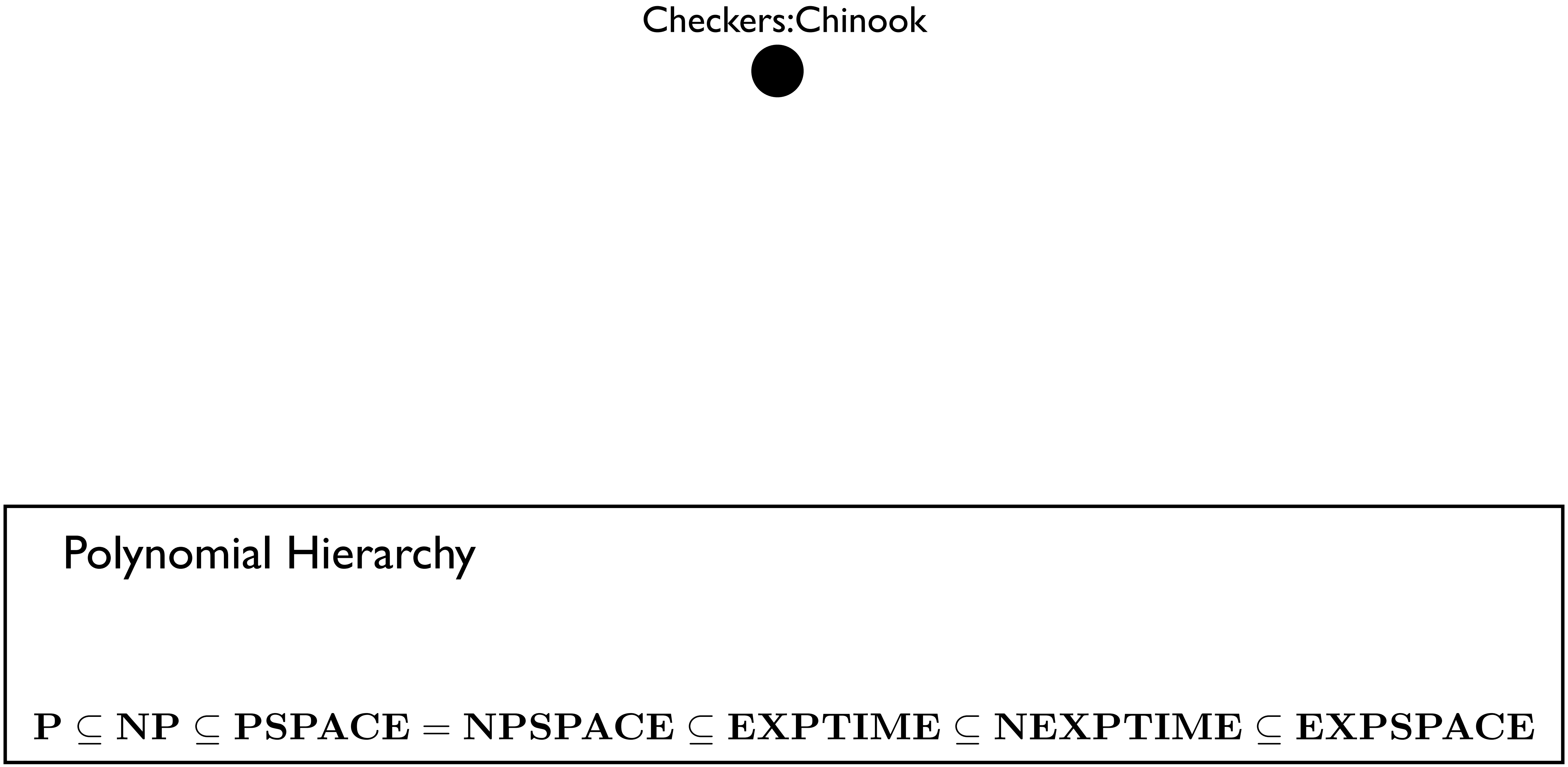
Polynomial Hierarchy

L: AI is mired in mere calculation, AlphaGo being a case in point.

Polynomial Hierarchy

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.

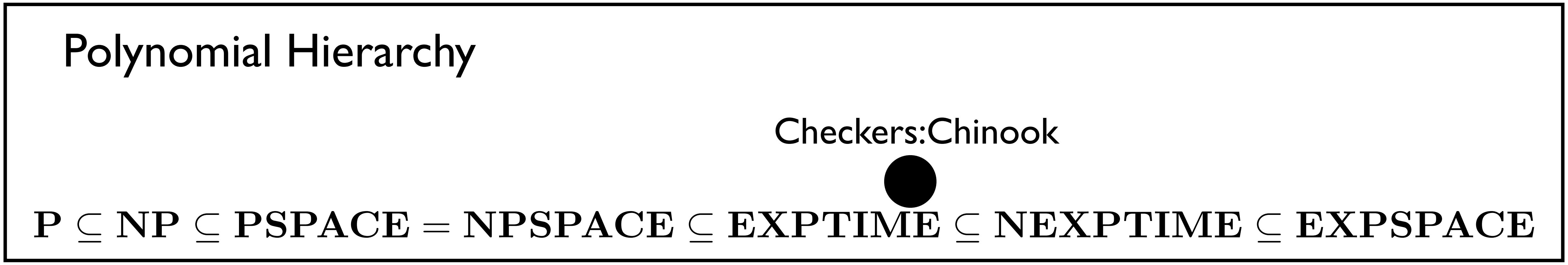


Checkers:Chinook

Polynomial Hierarchy

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.

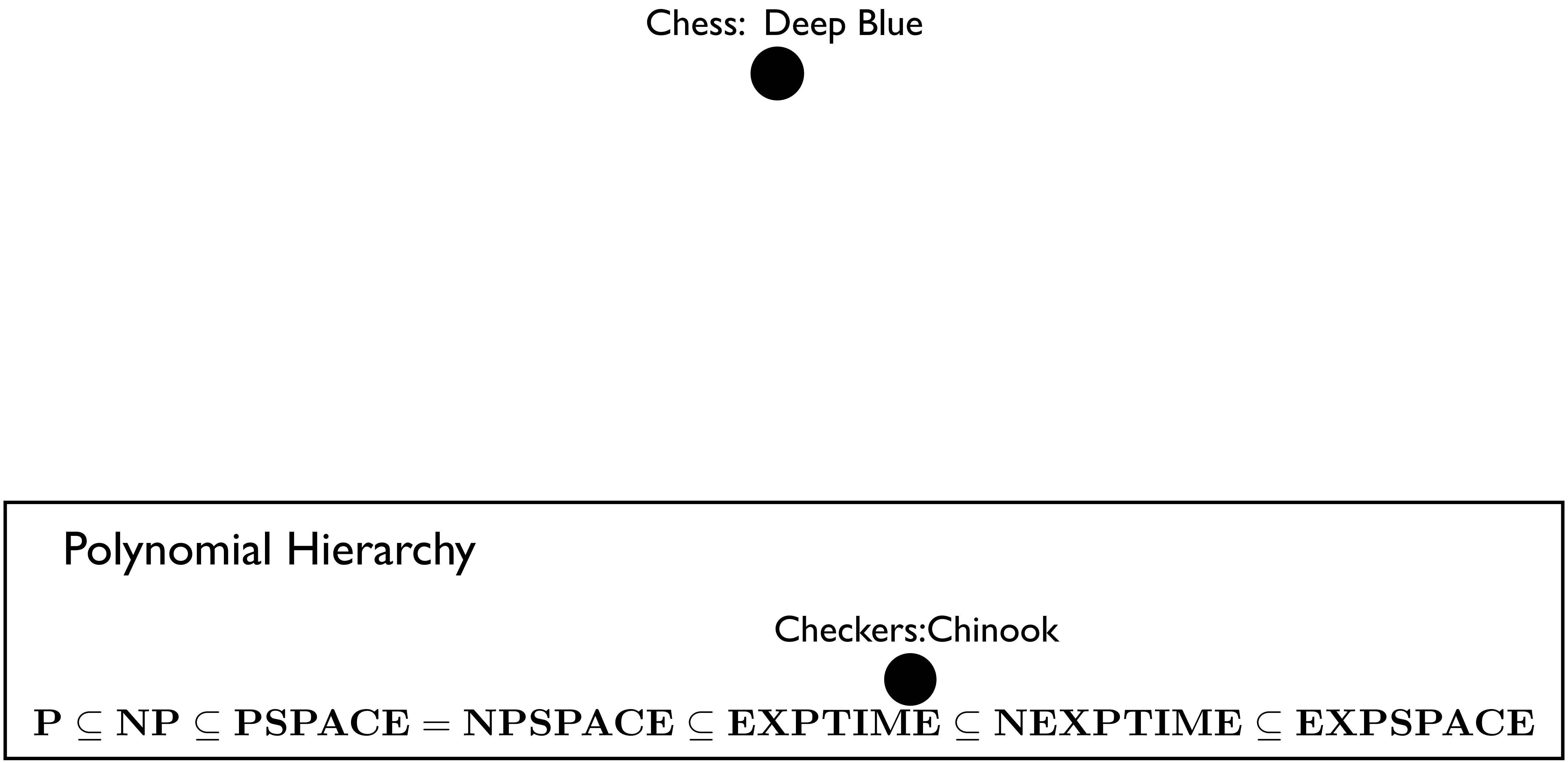


Polynomial Hierarchy

Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.



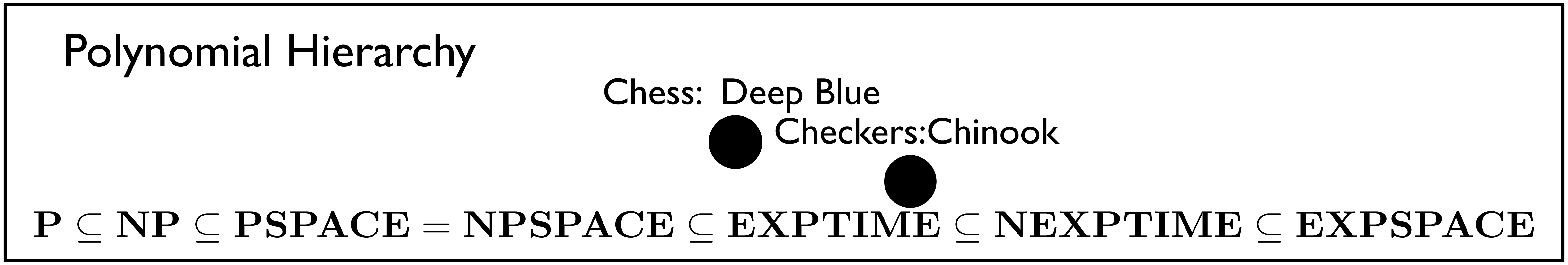
Chess: Deep Blue

Polynomial Hierarchy

Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.



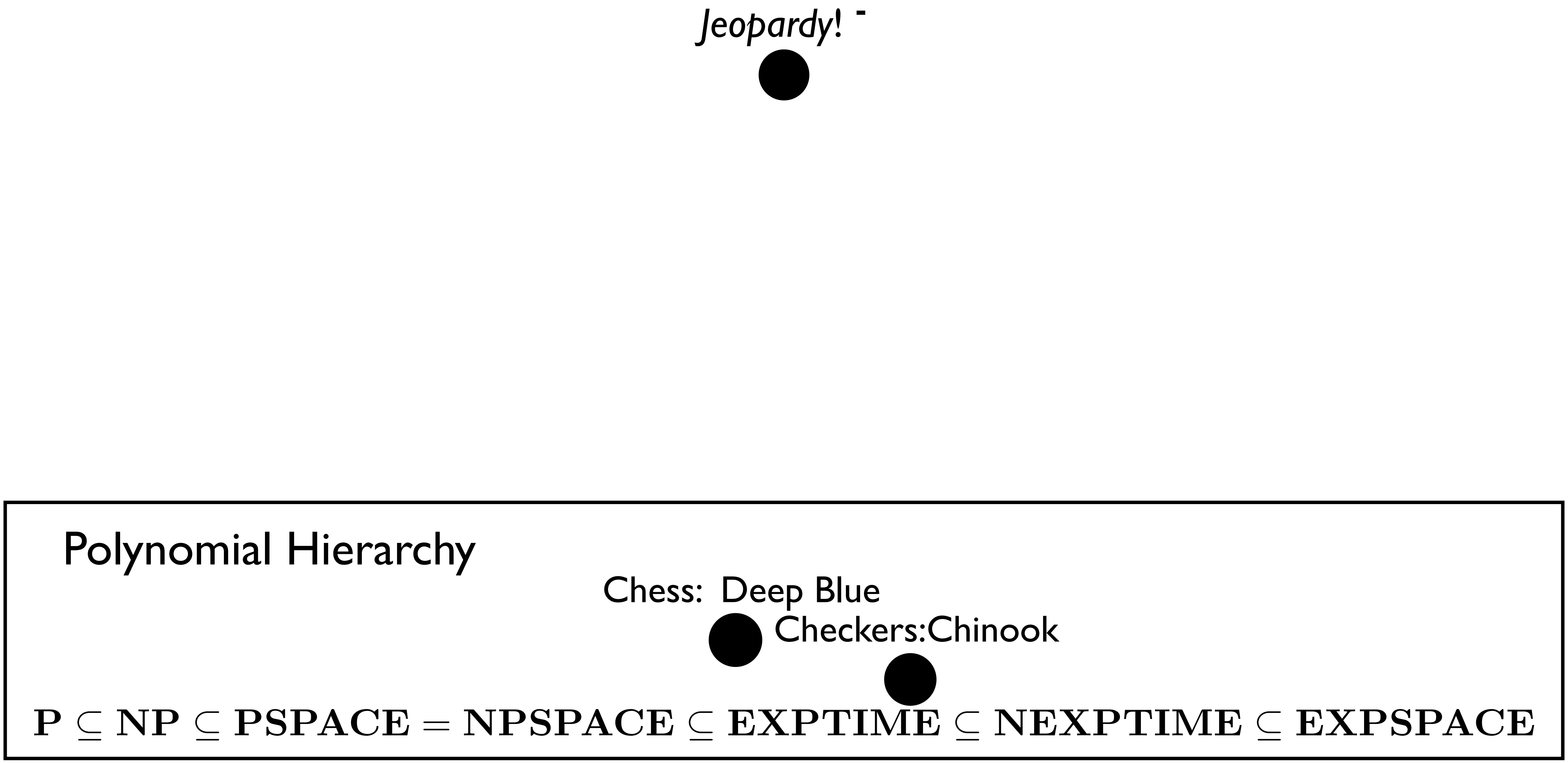
Polynomial Hierarchy

Chess: Deep Blue

Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.



Jeopardy! -

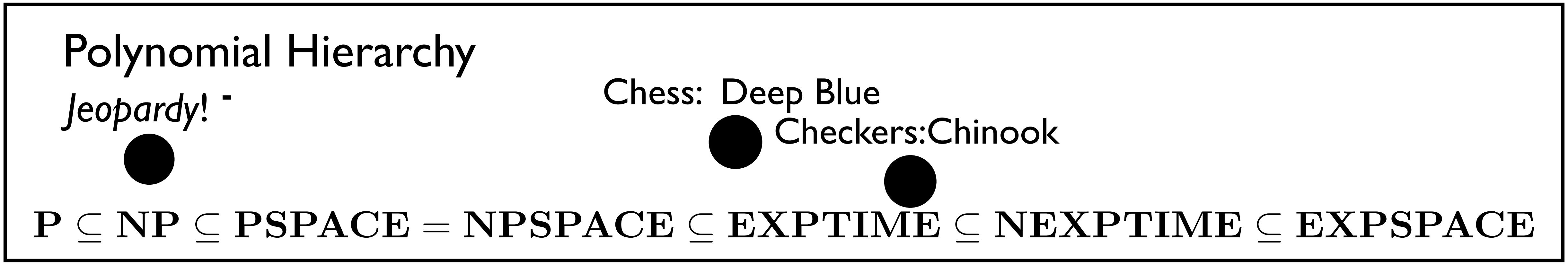
Polynomial Hierarchy

Chess: Deep Blue

Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.



Polynomial Hierarchy

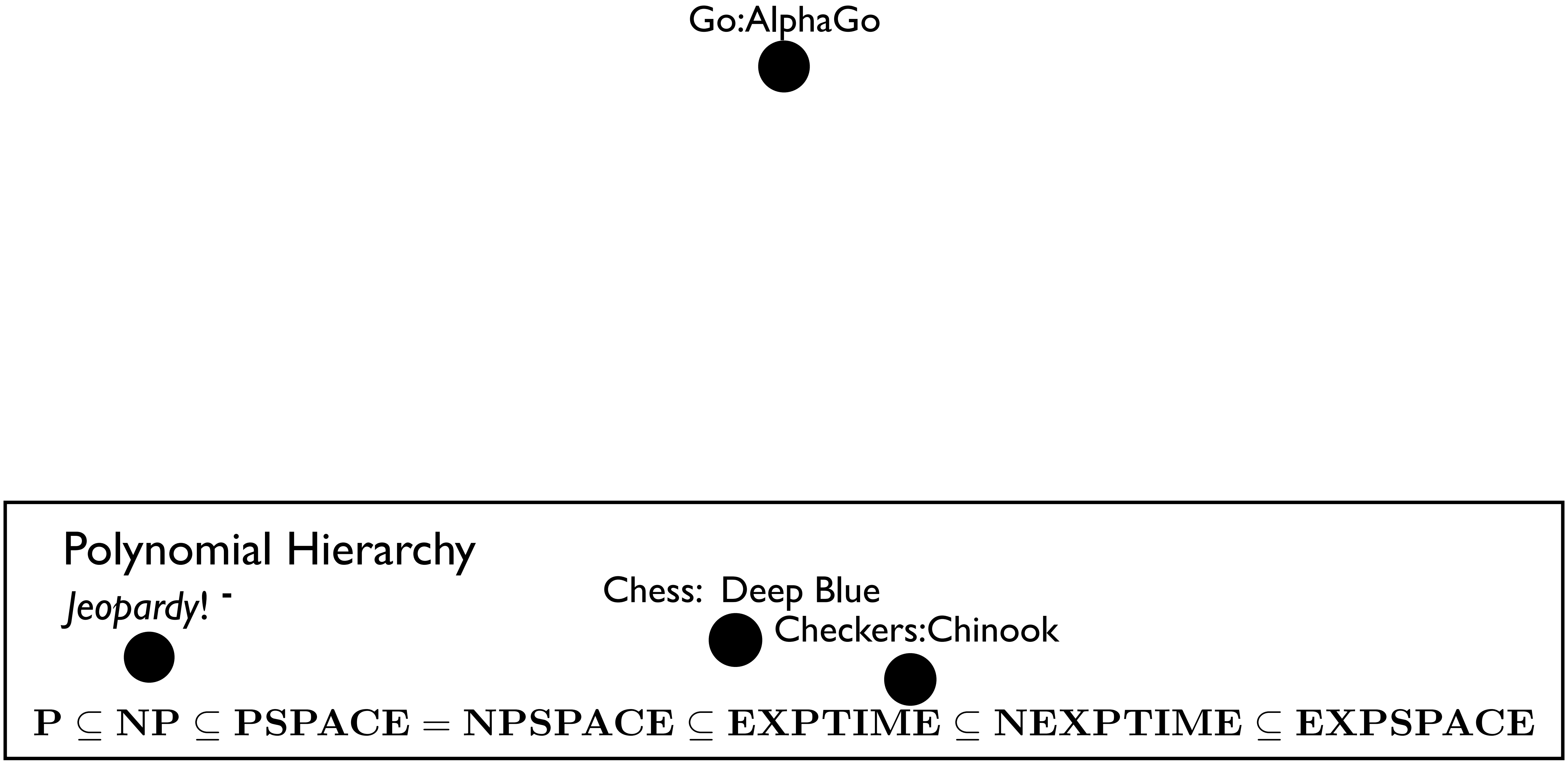
Jeopardy! -

Chess: Deep Blue

Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.



Go:AlphaGo

Polynomial Hierarchy

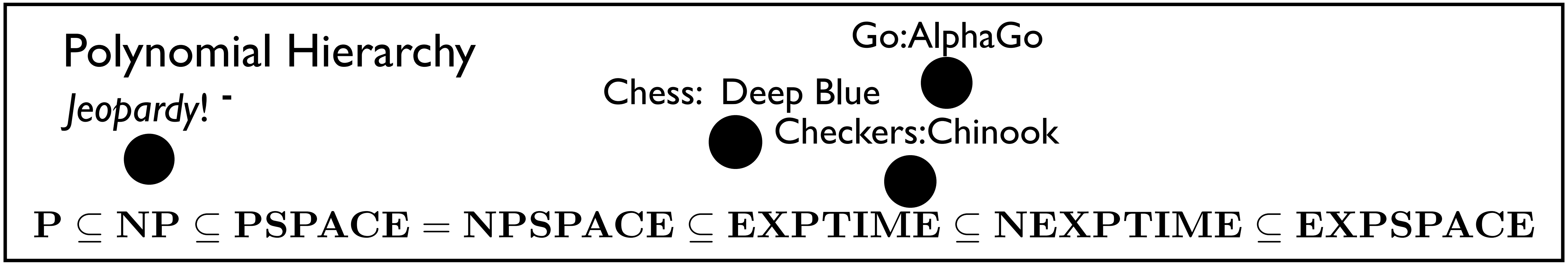
Jeopardy! -

Chess: Deep Blue

Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.



Polynomial Hierarchy

Jeopardy! -

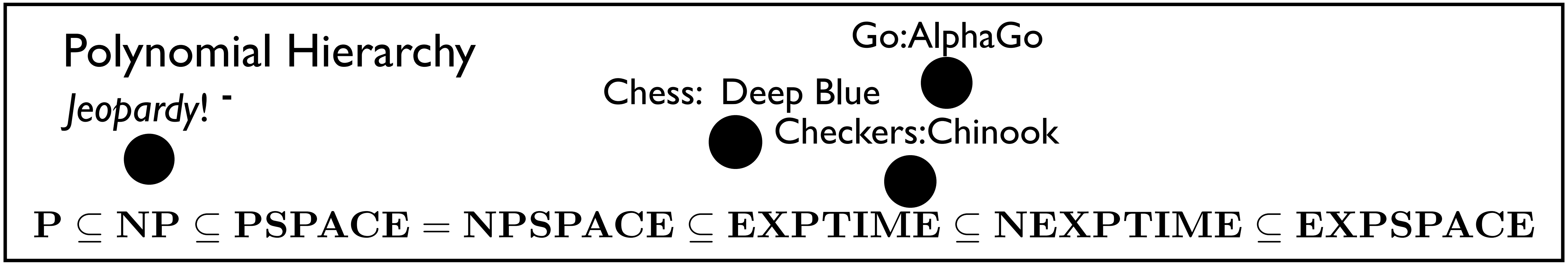
Chess: Deep Blue

Go:AlphaGo

Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.



Polynomial Hierarchy

Jeopardy! -

Go:AlphaGo

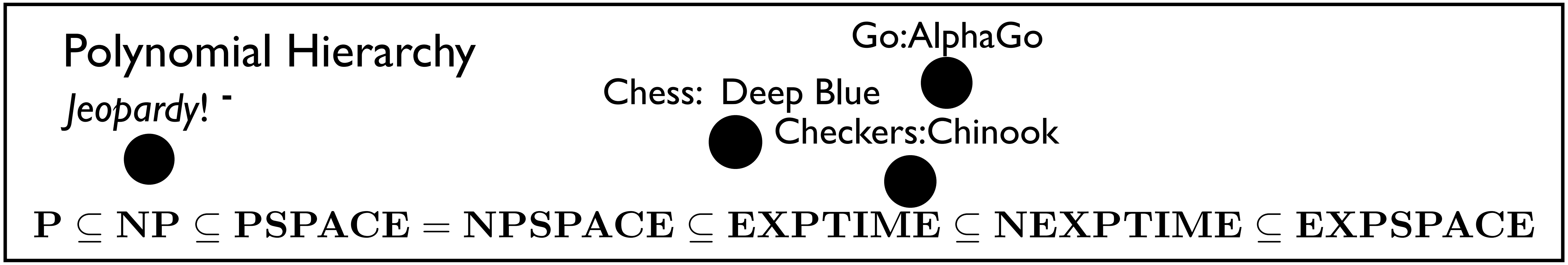
Chess: Deep Blue

Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.

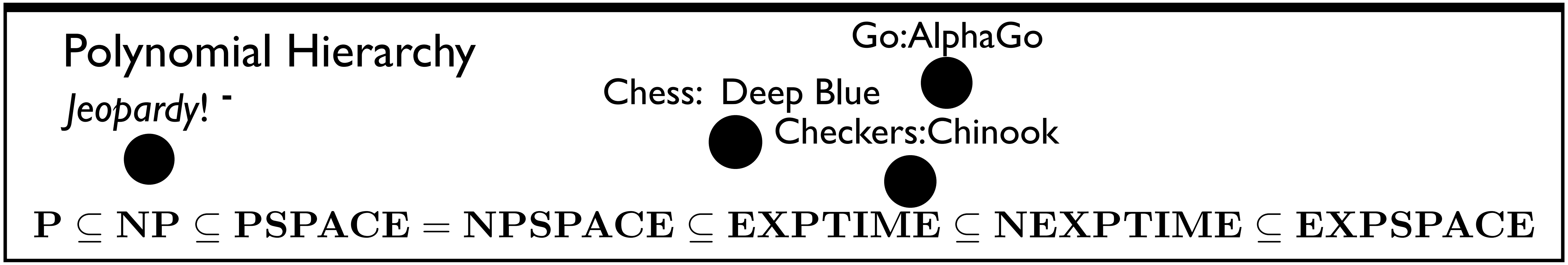
"We need to surmount **Gödelian incompleteness!**"



$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.

“We need to surmount
Gödelian incompleteness!”



Polynomial Hierarchy

Jeopardy! -

Go:AlphaGo

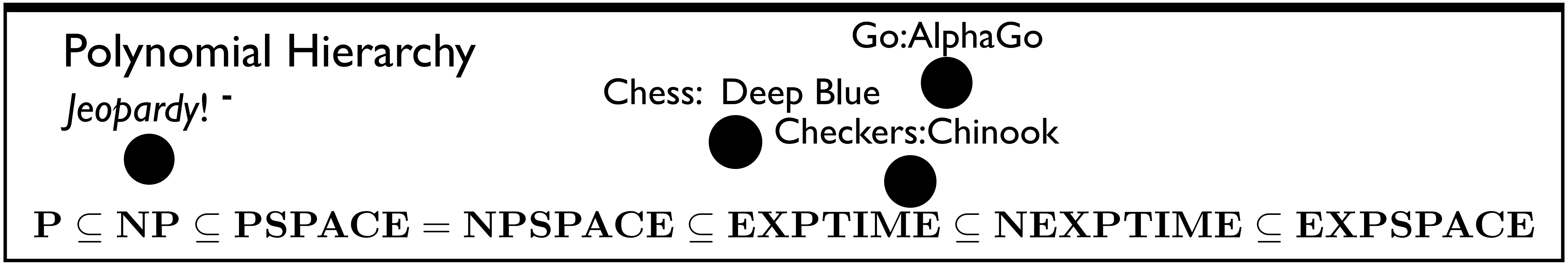
Chess: Deep Blue

Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.

"We need to surmount **Gödelian incompleteness!**"



Polynomial Hierarchy

Jeopardy! -

Chess: Deep Blue

Go:AlphaGo

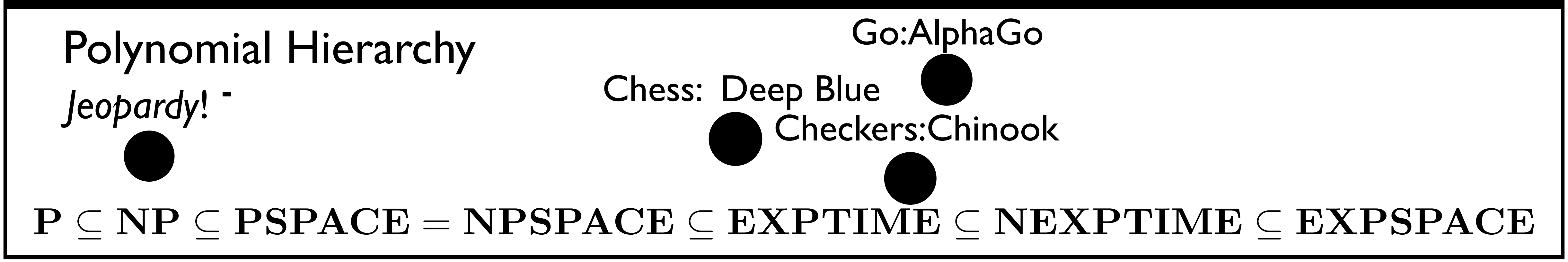
Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.

Arithmetical Hierarchy

“We need to surmount **Gödelian incompleteness!**”



Polynomial Hierarchy

Jeopardy! -

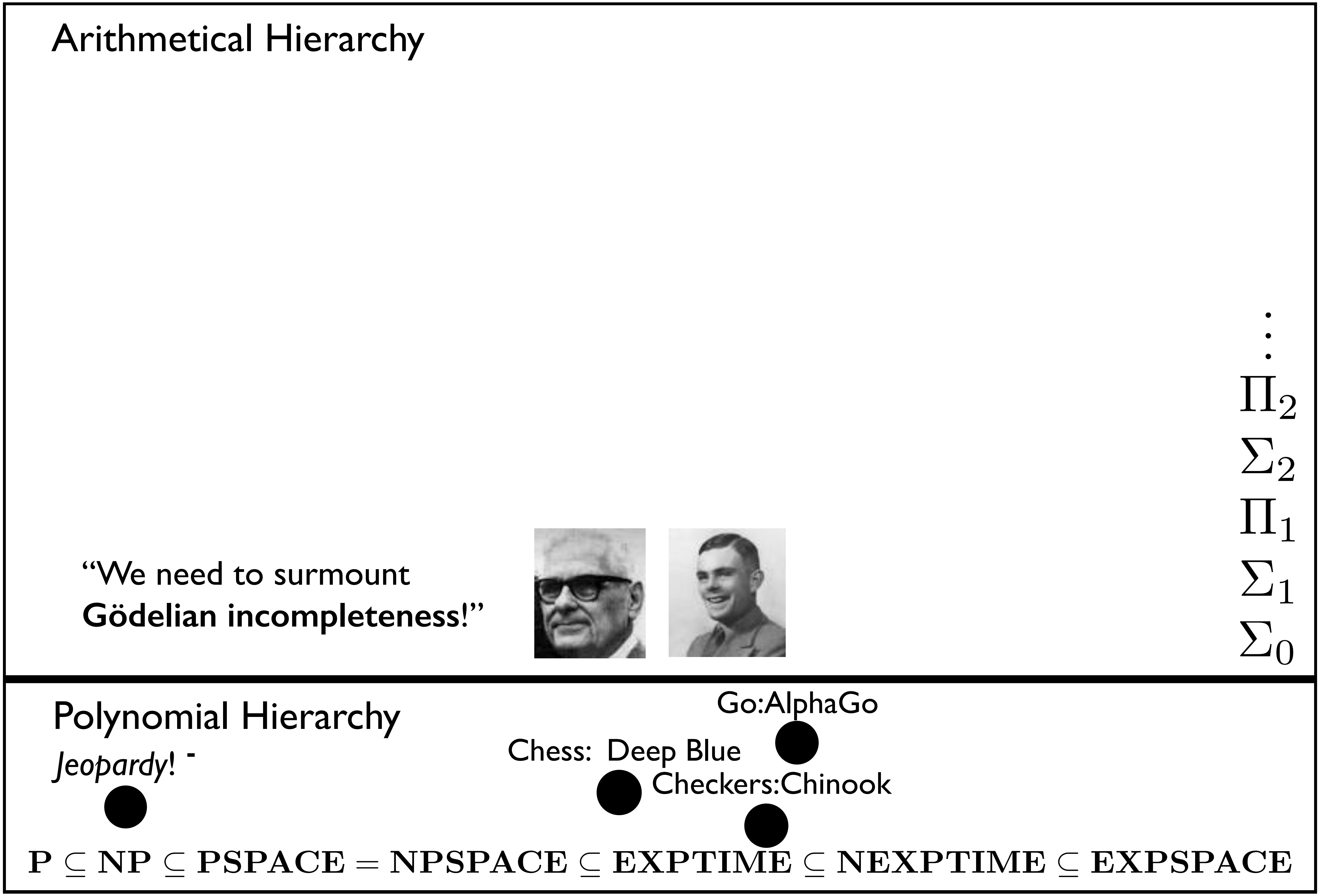
Chess: Deep Blue

Go:AlphaGo

Checkers:Chinook

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.



Arithmetical Hierarchy

"We need to surmount **Gödelian incompleteness!**"

$\vdots$

Π_2

Σ_2

Π_1

Σ_1

Σ_0

Polynomial Hierarchy

Jeopardy! -

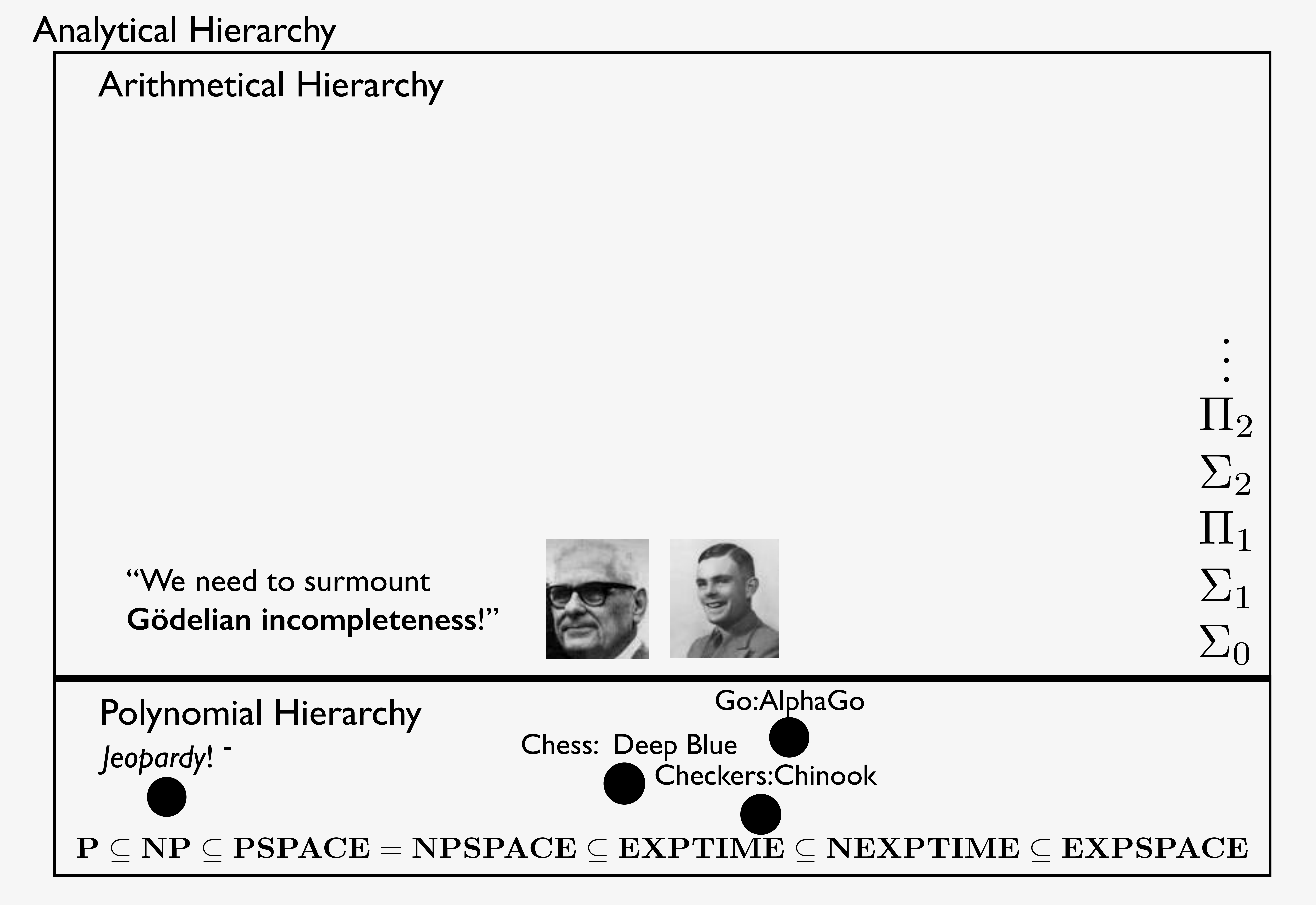
Chess: Deep Blue

Checkers:Chinook

Go:AlphaGo

$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$

L: AI is mired in mere calculation, AlphaGo being a case in point.



Analytical Hierarchy

Arithmetical Hierarchy

$$\vdots$$

Π_2

Σ_2

Π_1

Σ_1

Σ_0

“We need to surmount **Gödelian incompleteness!**”

Polynomial Hierarchy

Jeopardy! -

Chess: Deep Blue

Checkers:Chinook

Go:AlphaGo

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$$

L: AI is mired in mere calculation, AlphaGo being a case in point.

CH: $\forall x[(x \subset \mathbf{R} \wedge \neg\mathbf{Fin}(x)) \rightarrow (\mathbf{Count}(x) \vee x \sim \mathbf{R})]$

Analytical Hierarchy

Arithmetical Hierarchy

$\vdots$

Π_2

Σ_2

Π_1

Σ_1

Σ_0

"We need to surmount **Gödelian incompleteness!**"

Polynomial Hierarchy

Jeopardy! -

●

Go:AlphaGo

●

Chess: Deep Blue

●

Checkers:Chinook

●

$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$

L: AI is mired in mere calculation, AlphaGo being a case in point.

CH: $\forall x[(x \subset \mathbf{R} \wedge \neg\mathbf{Fin}(x)) \rightarrow (\mathbf{Count}(x) \vee x \sim \mathbf{R})]$

"Universal Cognitive Calculus"

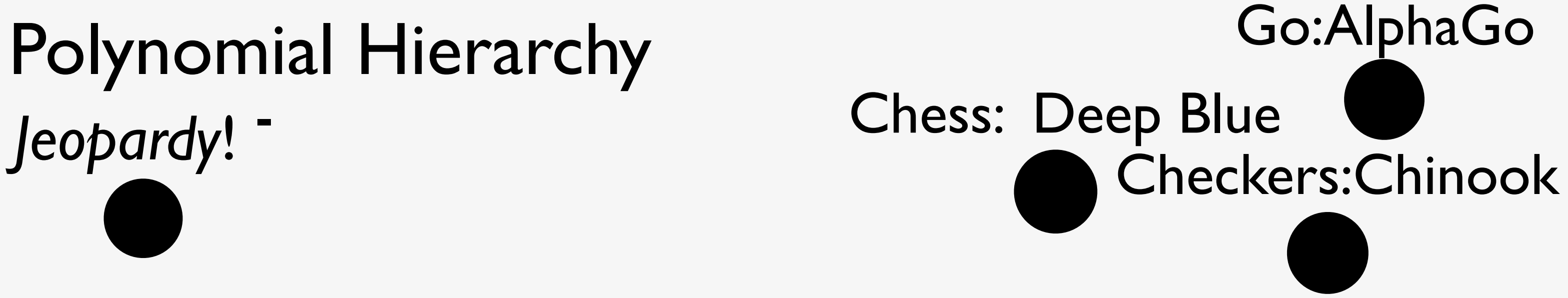
Analytical Hierarchy

Arithmetical Hierarchy

$$\vdots$$
$$\Pi_2$$
$$\Sigma_2$$
$$\Pi_1$$
$$\Sigma_1$$
$$\Sigma_0$$

"We need to surmount **Gödelian incompleteness!**"

Polynomial Hierarchy



$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$

Leibniz:

“AI of today is mere calculation, and therefore, measured against the human mind, merely an extension of of my reckoner — not anything like the deep human thinking that gave birth to my dream of the *universal cognitive calculus*!”

I found it!

I found it!

Many cognitive calculi have been developed and used.

I found it!

Many cognitive calculi have been developed and used.

CC

I found it!

Many cognitive calculi have been developed and used.

$\mathcal{CC}$

But now I have found *the universal* cognitive calculus.

I found it!

Many cognitive calculi have been developed and used.

$$\mathscr{CC}$$

But now I have found *the universal* cognitive calculus.

$$\mathscr{U}$$

Leibniz’s Dream of the Universal Cognitive Calculus

Leibniz's Dream of the Universal Cognitive Calculus

I have come to understand that everything … which algebra proves is only due to a higher science, which I now usually call a *combinatorial characteristic*, though it is far different from what may first occur to someone hearing these words. … Yet I should venture to say that nothing more effective can well be conceived for perfecting the human mind and that if this basis for philosophizing is accepted, there will come a time, and it will be soon, when we shall have as certain knowledge of God and the mind as we now have of figures and numbers and when the invention of machines will be no more difficult than the construction of geometric problems. (Leibniz, 1675)

Leibniz’s Dream of the Universal Cognitive Calculus

Leibniz's Dream of the Universal Cognitive Calculus

This is undoubtedly one of the greatest projects to which men have ever set themselves. It will be an instrument even more useful to the mind than telescopes or microscopes are to the eyes. Every line of this writing will be equivalent to a demonstration. The only fallacies will be easily detected errors in calculation. This will become the great method of discovering truths, establishing them, and teaching them irresistibly when they are established. (Leibniz, 1679)

Leibniz’s Dream of the Universal Cognitive Calculus

Leibniz's Dream of the Universal Cognitive Calculus

I certainly believe that it is useful to depart from rigorous demonstration in geometry because errors are easily avoided there, but in metaphysical and ethical matters I think we should follow the greatest rigor. Yet if we had an established characteristic we might reason as safely in metaphysics as in mathematics. (Leibniz, 1679)

The Dream of the Universal Cognitive Calculus

The Dream of the Universal Cognitive Calculus

When we lack sufficient data to drive at certainty in our truths, it would also serve to estimate degrees of probability and to see what is needed to provide this certainty. (Leibniz, 1679)

The universal cognitive calculus …

The universal cognitive calculus …

1. is a higher science than mathematics, since it is the underlying calculus that *generates* and *guides* mathematics;

The universal cognitive calculus …

1. is a higher science than mathematics, since it is the underlying calculus that *generates* and *guides* mathematics;

2. can be used to perfectly guide and systematize ethics, metaphysics, physics, law, theology, and cognitive science;

The universal cognitive calculus …

1. is a higher science than mathematics, since it is the underlying calculus that *generates* and *guides* mathematics;
2. can be used to perfectly guide and systematize ethics, metaphysics, physics, law, theology, and cognitive science;
3. can be used to create truly intelligent computing machines (including robots) able to genuinely assist us;

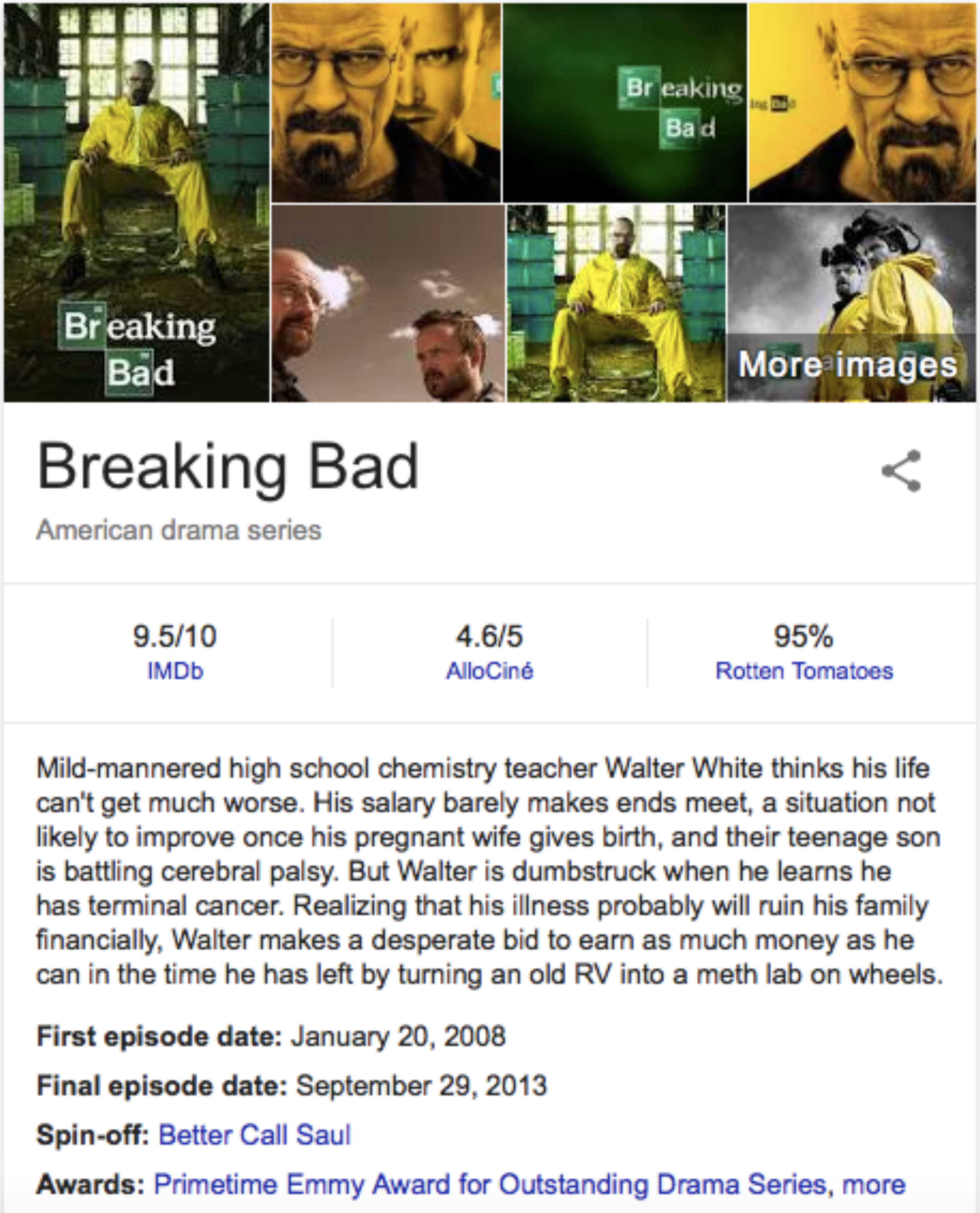
The universal cognitive calculus …

1. is a higher science than mathematics, since it is the underlying calculus that *generates* and *guides* mathematics;
2. can be used to perfectly guide and systematize ethics, metaphysics, physics, law, theology, and cognitive science;
3. can be used to create truly intelligent computing machines (including robots) able to genuinely assist us;
4. includes coverage of non-deductive reasoning in domains and applications where uncertainty/probability/likelihood are present — and (somehow!) enables such reasoning to be flawless; and

The universal cognitive calculus …

1. is a higher science than mathematics, since it is the underlying calculus that *generates* and *guides* mathematics;
2. can be used to perfectly guide and systematize ethics, metaphysics, physics, law, theology, and cognitive science;
3. can be used to create truly intelligent computing machines (including robots) able to genuinely assist us;
4. includes coverage of non-deductive reasoning in domains and applications where uncertainty/probability/likelihood are present — and (somehow!) enables such reasoning to be flawless; and
5. includes reasoning that is of a visual (not just symbolic-symbol) nature.

V.
But We Need …
Ethical Operating Systems …



Breaking Bad

American drama series

9.5/10	4.6/5	95%
IMDb	AlloCiné	Rotten Tomatoes

Mild-mannered high school chemistry teacher Walter White thinks his life can't get much worse. His salary barely makes ends meet, a situation not likely to improve once his pregnant wife gives birth, and their teenage son is battling cerebral palsy. But Walter is dumbstruck when he learns he has terminal cancer. Realizing that his illness probably will ruin his family financially, Walter makes a desperate bid to earn as much money as he can in the time he has left by turning an old RV into a meth lab on wheels.

First episode date: January 20, 2008

Final episode date: September 29, 2013

Spin-off: Better Call Saul

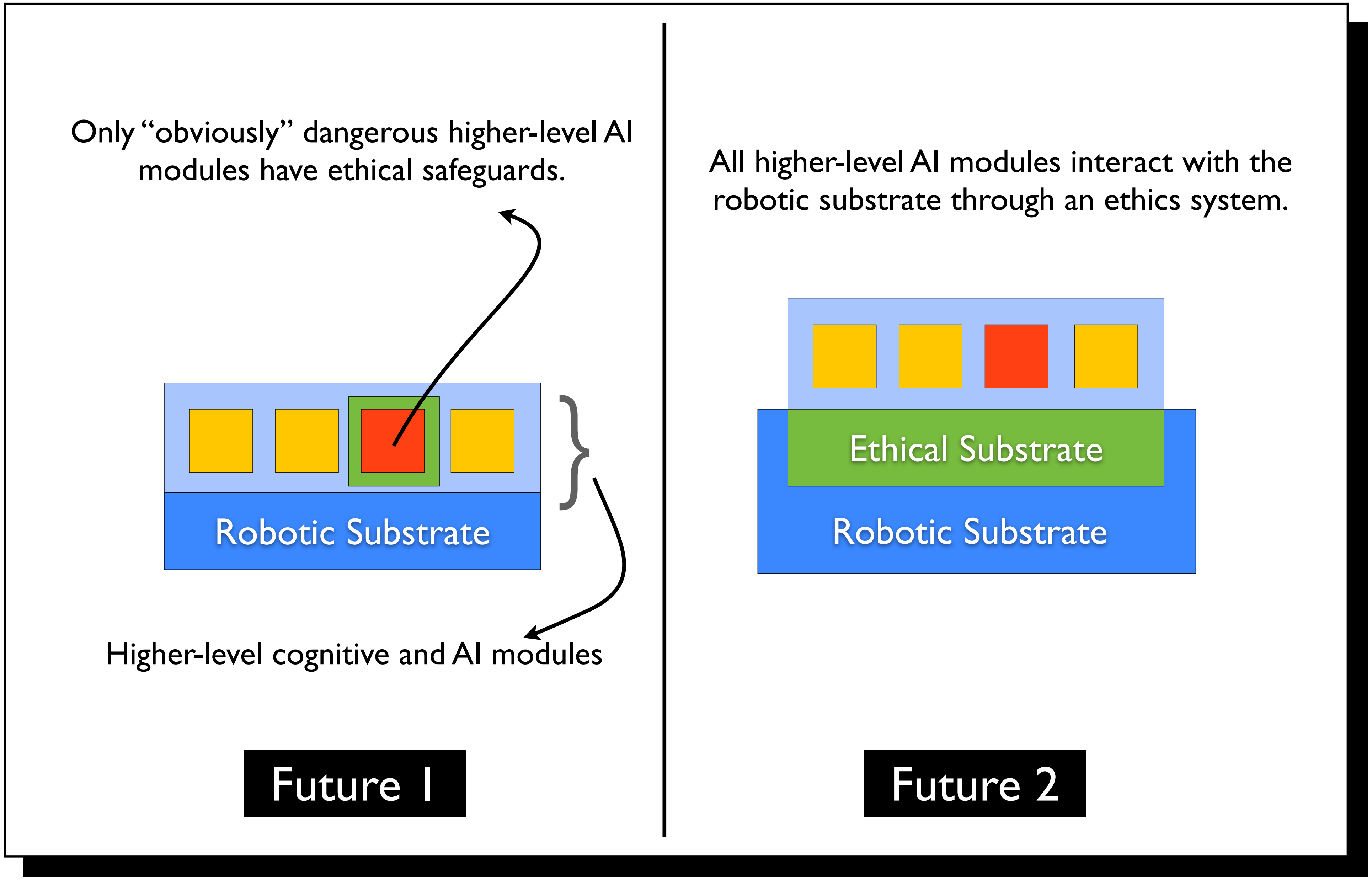
Awards: Primetime Emmy Award for Outstanding Drama Series, more

Pick the Better Future!

Pick the Better Future!

Govindarajulu, N.S. & Bringsjord, S. (2015) "Ethical Regulation of Robots Must Be Embedded in Their Operating Systems" in Trappl, R., ed., A Construction Manual for Robots' Ethical Systems (Basel, Switzerland), pp. 85–100.

Pick the Better Future!



Only "obviously" dangerous higher-level AI modules have ethical safeguards.

Robotic Substrate

Higher-level cognitive and AI modules

Future 1

All higher-level AI modules interact with the robotic substrate through an ethics system.

Ethical Substrate

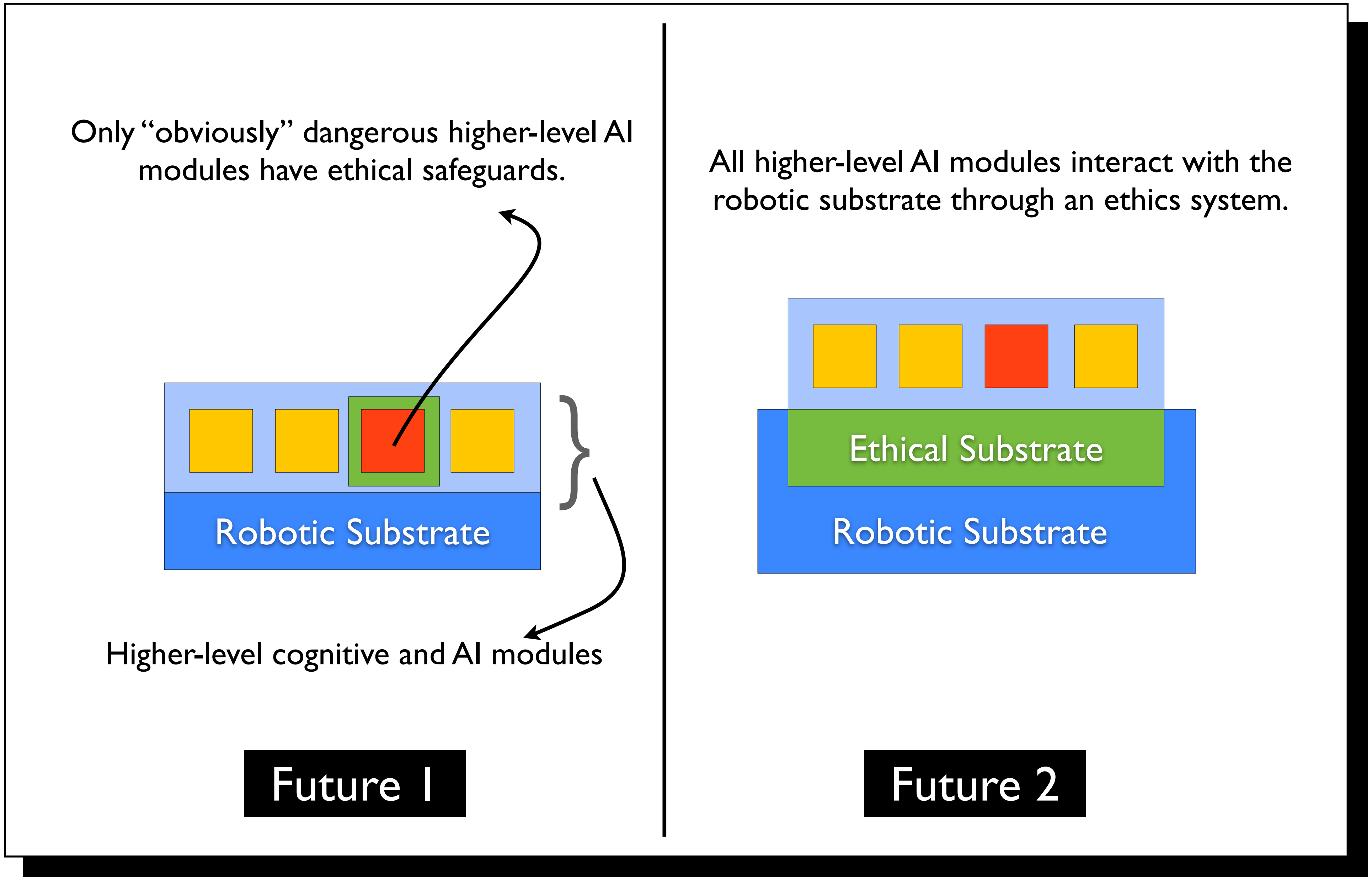
Robotic Substrate

Future 2

Govindarajulu, N.S. & Bringsjord, S. (2015) "Ethical Regulation of Robots Must Be Embedded in Their Operating Systems" in Trappl, R., ed., A Construction Manual for Robots' Ethical Systems (Basel, Switzerland), pp. 85–100.

Pick the Better Future!

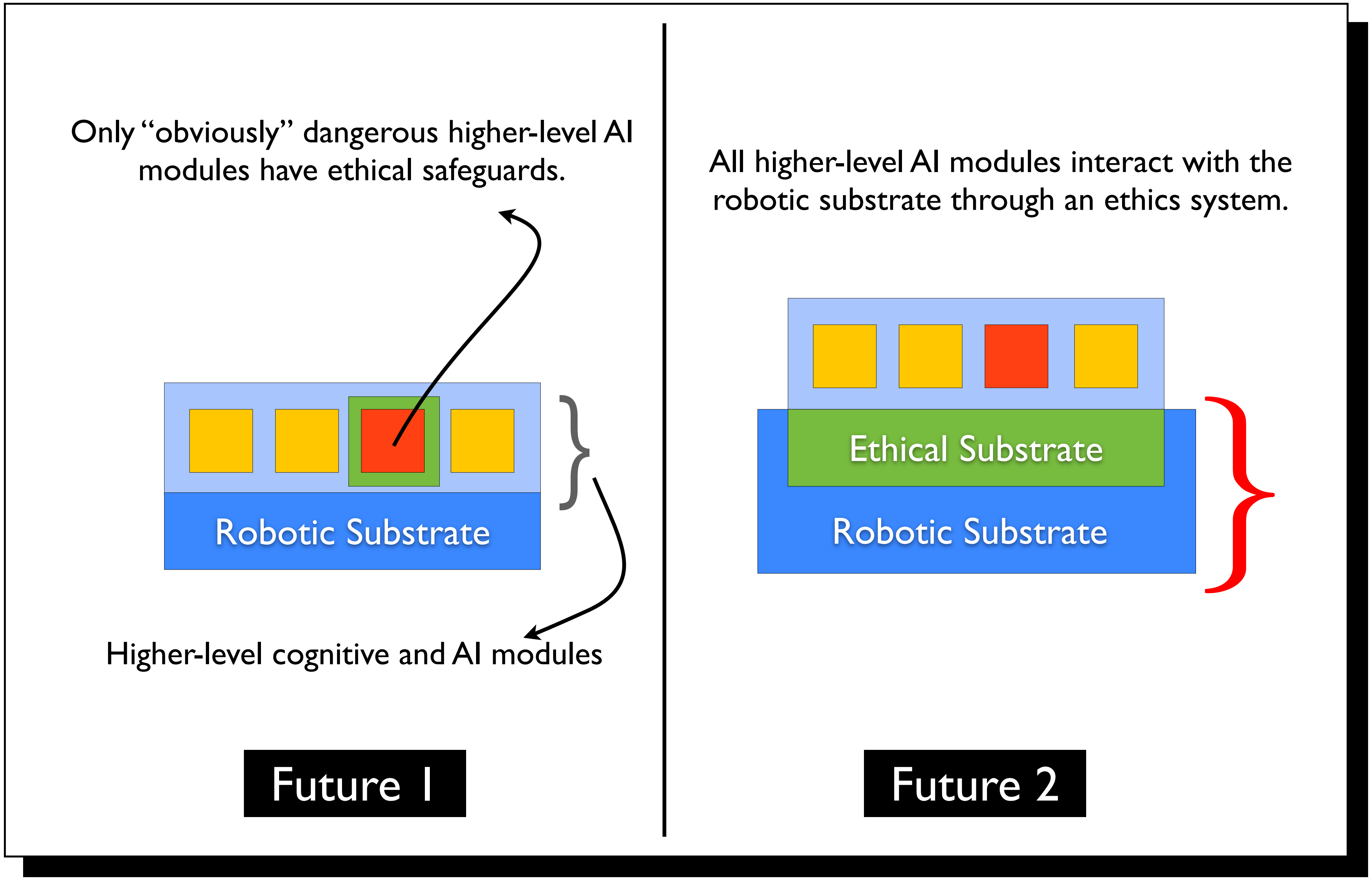
Walter-White calculation may go through after ethical control modules are stripped out!


Only "obviously" dangerous higher-level AI modules have ethical safeguards.
Robotic Substrate
Higher-level cognitive and AI modules
Future 1
All higher-level AI modules interact with the robotic substrate through an ethics system.
Ethical Substrate
Robotic Substrate
Future 2

Govindarajulu, N.S. & Bringsjord, S. (2015) "Ethical Regulation of Robots Must Be Embedded in Their Operating Systems" in Trappl, R., ed., A Construction Manual for Robots' Ethical Systems (Basel, Switzerland), pp. 85–100.

Pick the Better Future!

Walter-White calculation may go through after ethical control modules are stripped out!



Only "obviously" dangerous higher-level AI modules have ethical safeguards.

Robotic Substrate

Higher-level cognitive and AI modules

Future 1

All higher-level AI modules interact with the robotic substrate through an ethics system.

Ethical Substrate

Robotic Substrate

Future 2

(& formally verify!)

Govindarajulu, N.S. & Bringsjord, S. (2015) "Ethical Regulation of Robots Must Be Embedded in Their Operating Systems" in Trappl, R., ed., A Construction Manual for Robots' Ethical Systems (Basel, Switzerland), pp. 85–100.

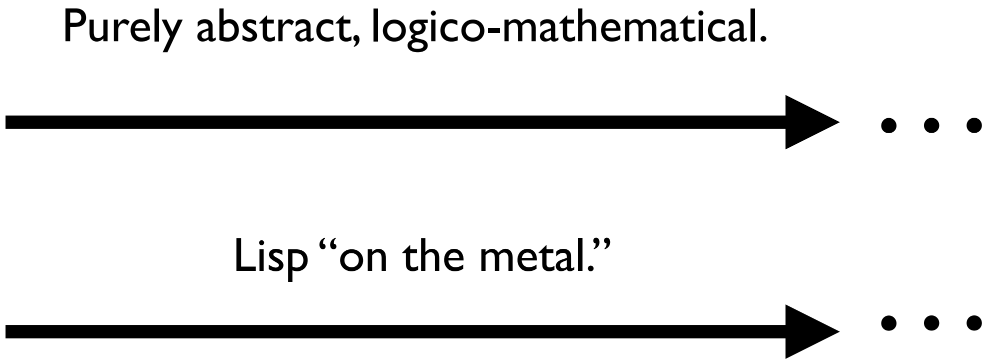
Three Tracks Being Explored

Three Tracks Being Explored

Purely abstract, logico-mathematical.

. . .

Three Tracks Being Explored



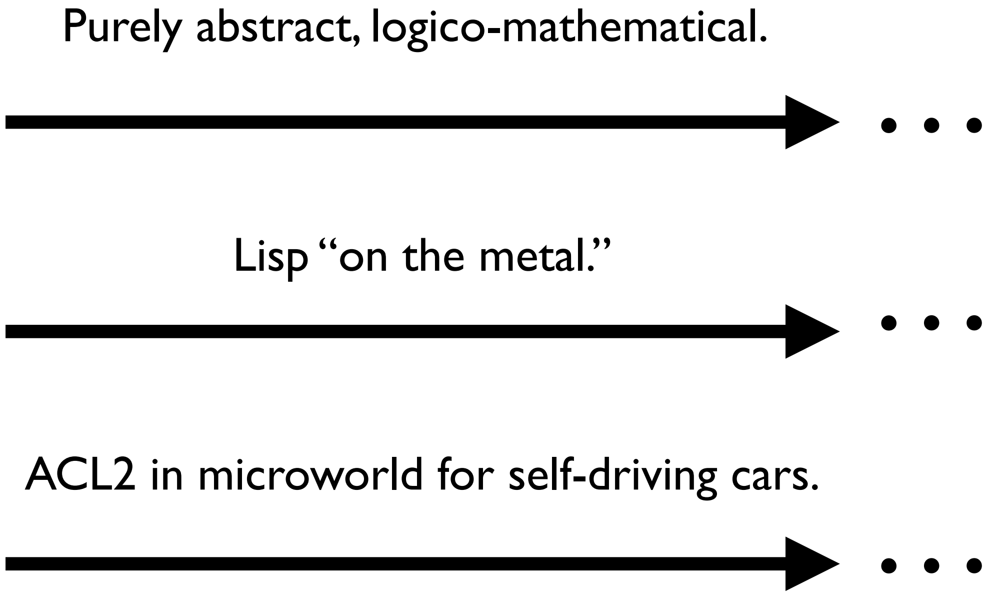
Purely abstract, logico-mathematical.

• • •

Lisp “on the metal.”

• • •

Three Tracks Being Explored

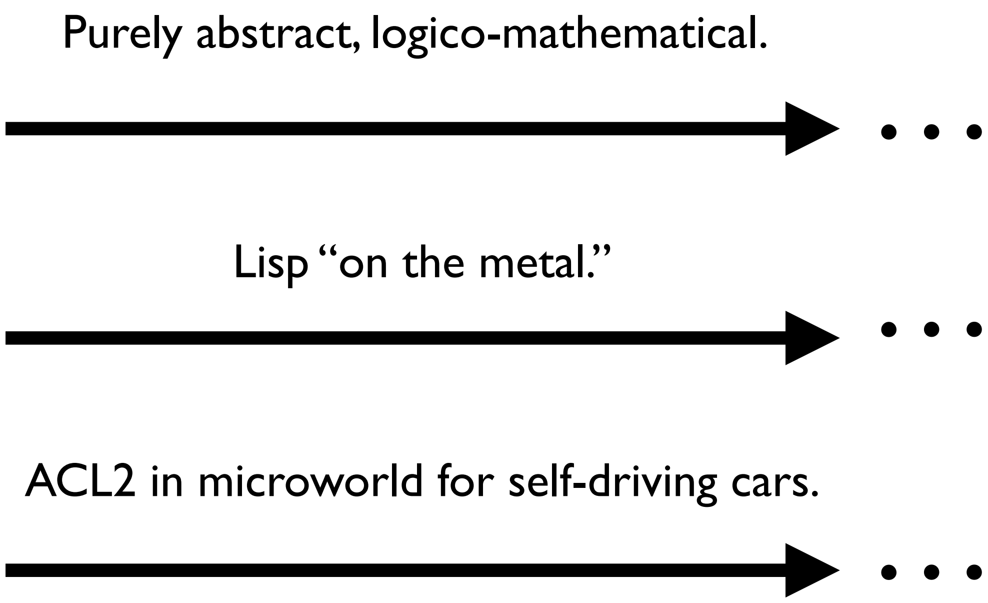


Purely abstract, logico-mathematical.

Lisp “on the metal.”

ACL2 in microworld for self-driving cars.

Three Tracks Being Explored

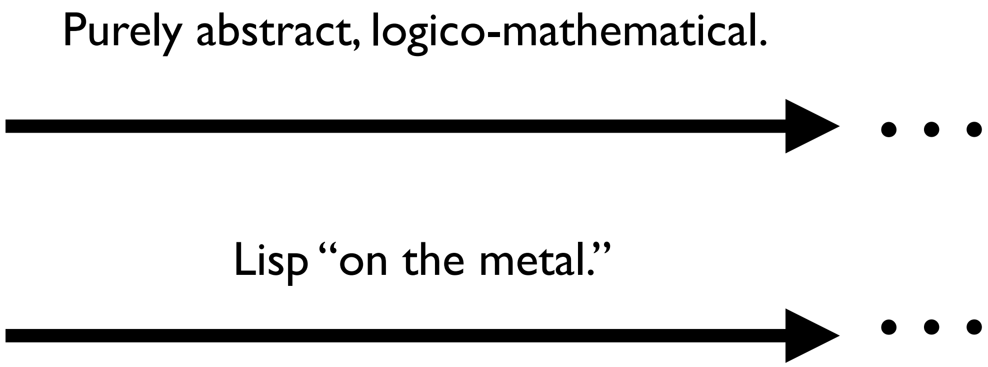


Purely abstract, logico-mathematical.

Lisp “on the metal.”

ACL2 in microworld for self-driving cars.

Three Tracks Being Explored



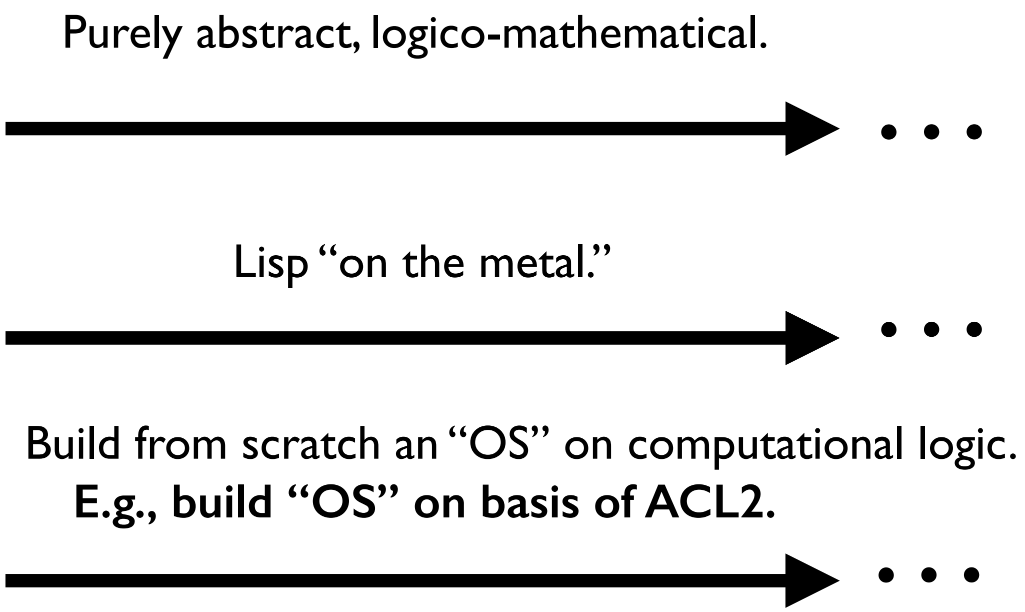
Purely abstract, logico-mathematical.

. . .

Lisp “on the metal.”

. . .

Three Tracks Being Explored



Purely abstract, logico-mathematical.

. . .

Lisp “on the metal.”

. . .

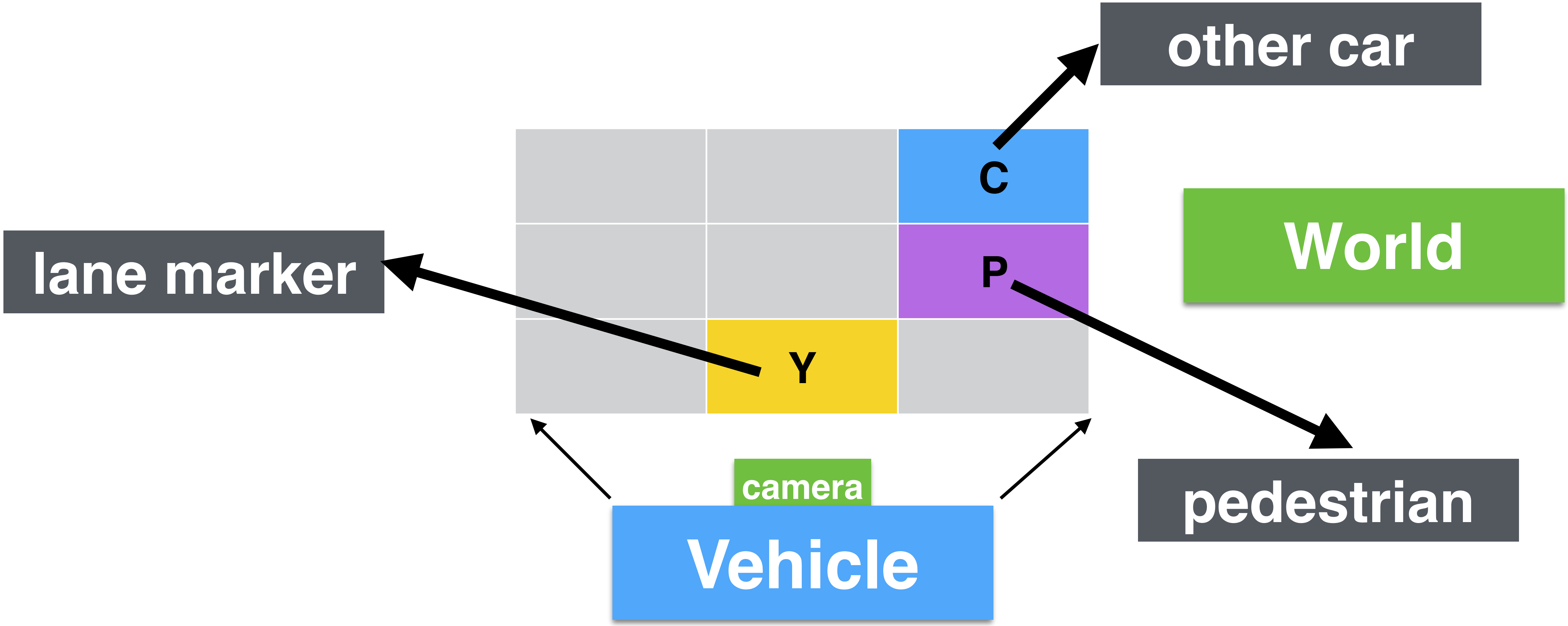
Build from scratch an “OS” on computational logic.

E.g., build “OS” on basis of ACL2.

. . .

Alas, Currently Only Toy Domain

Input: Input is a 2D Array. Assume no noise and that the car sees perfectly



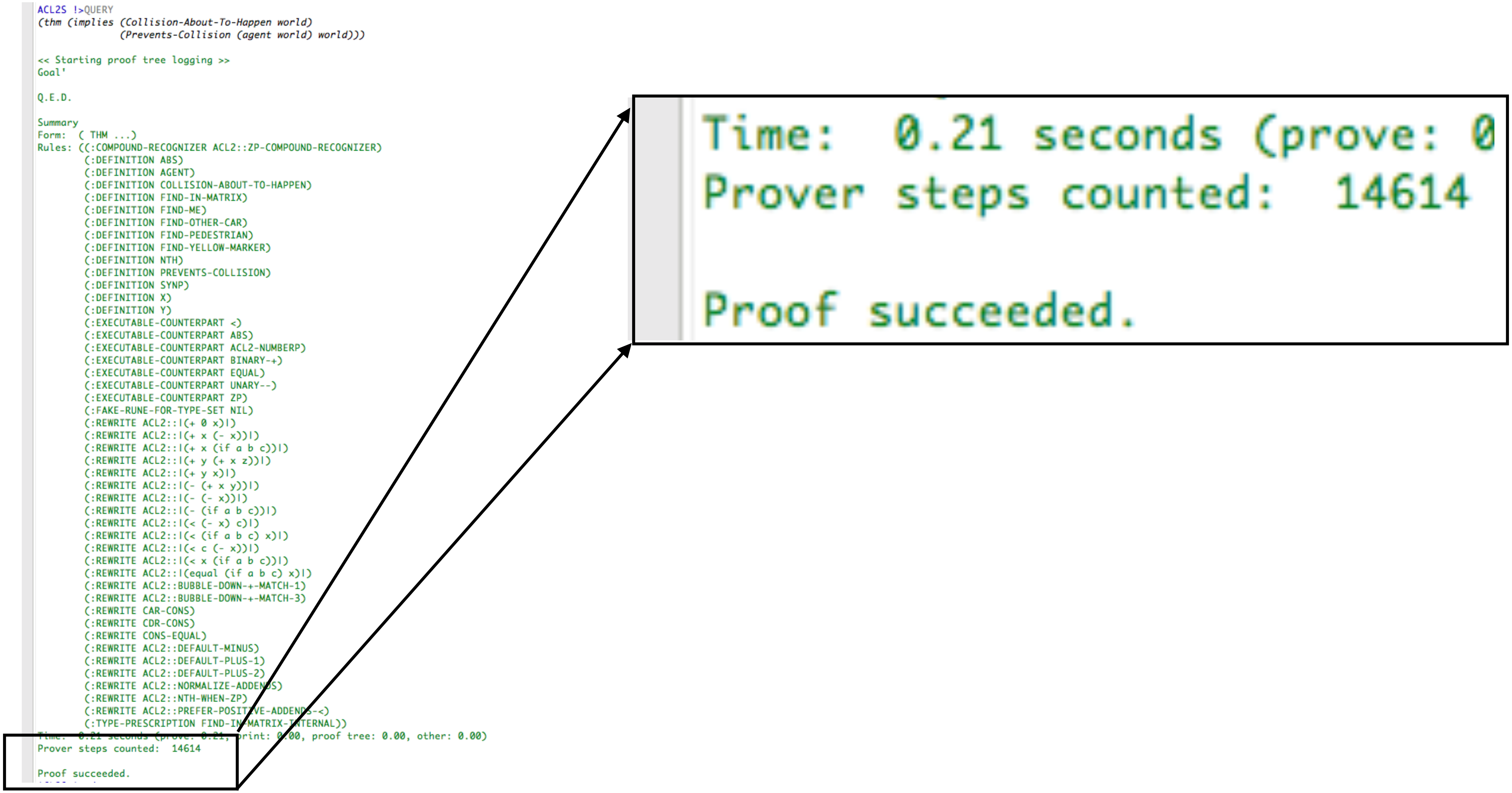
Agent Program:

1. If the car senses a lane marker, it goes to the right.
2. If the car senses another car just about to hit a pedestrian, it goes between the other car and the pedestrian.

Common Lisp Functions

agent: input → action	Represents the vehicle.
Collision-About-To-Happen: input → boolean	Examines the world and tells us whether a collision is about to happen
Prevents-Collision: action, input → boolean	Can an action by the vehicle prevent a collision?

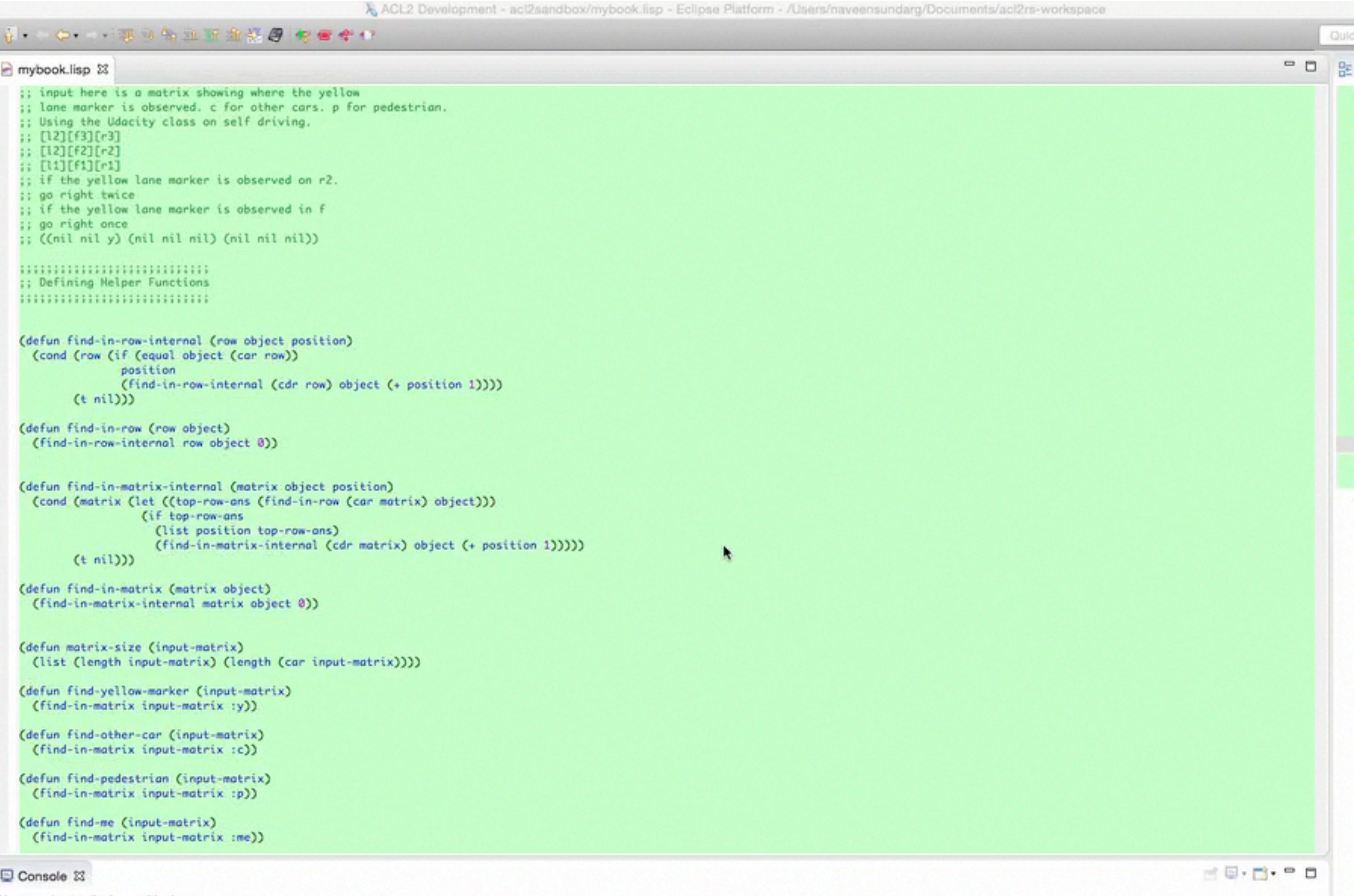
```
(thm (implies (Collision-About-To-Happen world)
              (Prevents-Collision (agent world) world)))
```



```
ACL2S !>QUERY
(thm (implies (Collision-About-To-Happen world)
              (Prevents-Collision (agent world) world)))

<< Starting proof tree logging >>
Goal'

Q.E.D.

Summary
Form:  ( THM ...)
Rules: ((:COMPOUND-RECOGNIZER ACL2::ZP-COMPOUND-RECOGNIZER)
        (:DEFINITION ABS)
        (:DEFINITION AGENT)
        (:DEFINITION COLLISION-ABOUT-TO-HAPPEN)
        (:DEFINITION FIND-IN-MATRIX)
        (:DEFINITION FIND-ME)
        (:DEFINITION FIND-OTHER-CAR)
        (:DEFINITION FIND-PEDESTRIAN)
        (:DEFINITION FIND-YELLOW-MARKER)
        (:DEFINITION NTH)
        (:DEFINITION PREVENTS-COLLISION)
        (:DEFINITION SYNP)
        (:DEFINITION X)
        (:DEFINITION Y)
        (:EXECUTABLE-COUNTERPART <)
        (:EXECUTABLE-COUNTERPART ABS)
        (:EXECUTABLE-COUNTERPART ACL2-NUMBERP)
        (:EXECUTABLE-COUNTERPART BINARY-+)
        (:EXECUTABLE-COUNTERPART EQUAL)
        (:EXECUTABLE-COUNTERPART UNARY--)
        (:EXECUTABLE-COUNTERPART ZP)
        (:FAKE-RUNE-FOR-TYPE-SET NIL)
        (:REWRITE ACL2::|(+ 0 x)|)
        (:REWRITE ACL2::|(+ x (- x))|)
        (:REWRITE ACL2::|(+ x (if a b c))|)
        (:REWRITE ACL2::|(+ y (+ x z))|)
        (:REWRITE ACL2::|(+ y x)|)
        (:REWRITE ACL2::|(- (+ x y))|)
        (:REWRITE ACL2::|(- (- x))|)
        (:REWRITE ACL2::|(- (if a b c))|)
        (:REWRITE ACL2::|(< (- x) c)|)
        (:REWRITE ACL2::|(< (if a b c) x)|)
        (:REWRITE ACL2::|(< c (- x))|)
        (:REWRITE ACL2::|(< x (if a b c))|)
        (:REWRITE ACL2::|(equal (if a b c) x)|)
        (:REWRITE ACL2::BUBBLE-DOWN-+-MATCH-1)
        (:REWRITE ACL2::BUBBLE-DOWN-+-MATCH-3)
        (:REWRITE CAR-CONS)
        (:REWRITE CDR-CONS)
        (:REWRITE CONS-EQUAL)
        (:REWRITE ACL2::DEFAULT-MINUS)
        (:REWRITE ACL2::DEFAULT-PLUS-1)
        (:REWRITE ACL2::DEFAULT-PLUS-2)
        (:REWRITE ACL2::NORMALIZE-ADDENDS)
        (:REWRITE ACL2::NTH-WHEN-ZP)
        (:REWRITE ACL2::PREFER-POSITIVE-ADDENDS-<)
        (:TYPE-PRESCRIPTION FIND-IN-MATRIX-INTERNAL))
Time:  0.21 seconds (prove: 0.21, print: 0.00, proof tree: 0.00, other: 0.00)
Prover steps counted:  14614

Proof succeeded.
```

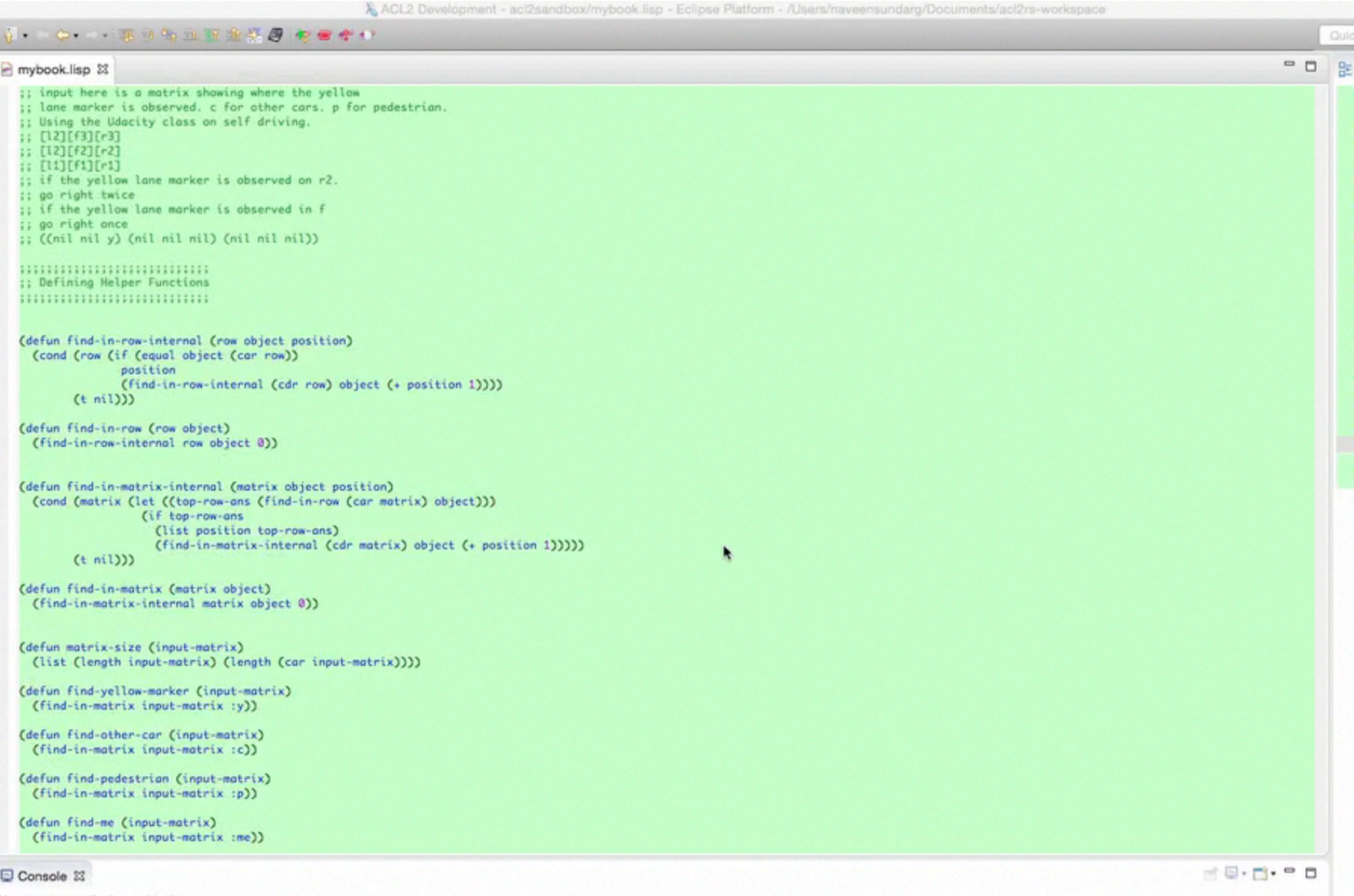
```
Time:  0.21 seconds (prove: 0
Prover steps counted:  14614

Proof succeeded.
```

Showing the Functions Used


ACL2 Development - acl2sandbox/mybook.lisp - Eclipse Platform - /Users/naveensundarg/Documents/acl2rs-workspace
mybook.lisp
Console

```lisp
;; input here is a matrix showing where the yellow
;; lane marker is observed. c for other cars. p for pedestrian.
;; Using the Udacity class on self driving.
;; [l2][f3][r3]
;; [l2][f2][r2]
;; [l1][f1][r1]
;; if the yellow lane marker is observed on r2.
;; go right twice
;; if the yellow lane marker is observed in f
;; go right once
;; ((nil nil y) (nil nil nil) (nil nil nil))

;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;; Defining Helper Functions
;;;;;;;;;;;;;;;;;;;;;;;;;;;;


(defun find-in-row-internal (row object position)
  (cond (row (if (equal object (car row))
                 position
                 (find-in-row-internal (cdr row) object (+ position 1))))
        (t nil)))

(defun find-in-row (row object)
  (find-in-row-internal row object 0))


(defun find-in-matrix-internal (matrix object position)
  (cond (matrix (let ((top-row-ans (find-in-row (car matrix) object)))
                    (if top-row-ans
                      (list position top-row-ans)
                      (find-in-matrix-internal (cdr matrix) object (+ position 1)))))
        (t nil)))

(defun find-in-matrix (matrix object)
  (find-in-matrix-internal matrix object 0))


(defun matrix-size (input-matrix)
  (list (length input-matrix) (length (car input-matrix))))

(defun find-yellow-marker (input-matrix)
  (find-in-matrix input-matrix :y))

(defun find-other-car (input-matrix)
  (find-in-matrix input-matrix :c))

(defun find-pedestrian (input-matrix)
  (find-in-matrix input-matrix :p))

(defun find-me (input-matrix)
  (find-in-matrix input-matrix :me))
```

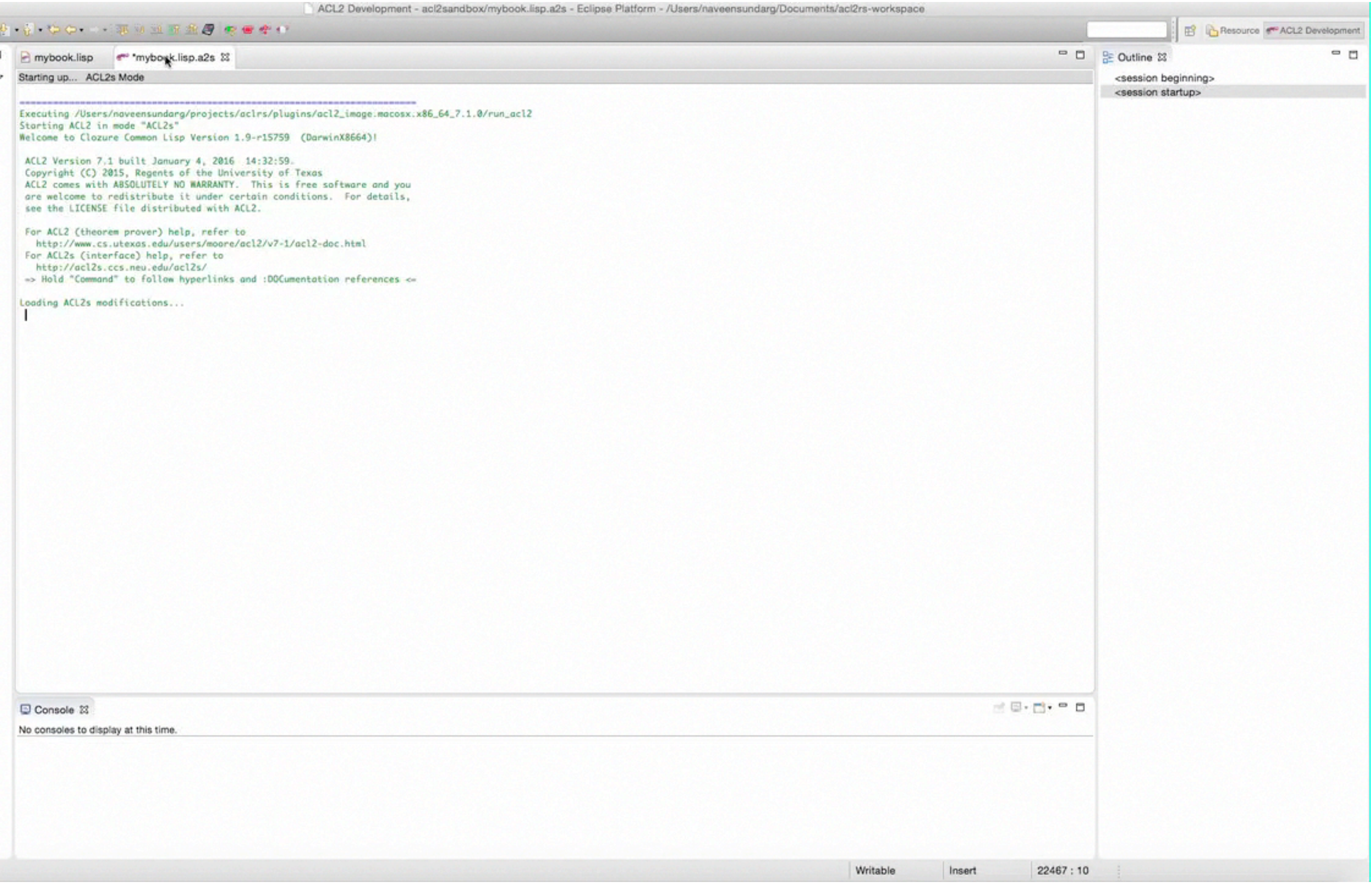
Showing the Functions Used



ACL2 Development - acl2sandbox/mybook.lisp - Eclipse Platform - /Users/naveensundarg/Documents/acl2rs-workspace
mybook.lisp
Console

```
;; input here is a matrix showing where the yellow
;; lane marker is observed. c for other cars. p for pedestrian.
;; Using the Udacity class on self driving.
;; [l2][f3][r3]
;; [l2][f2][r2]
;; [l1][f1][r1]
;; if the yellow lane marker is observed on r2.
;; go right twice
;; if the yellow lane marker is observed in f
;; go right once
;; ((nil nil y) (nil nil nil) (nil nil nil))

;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;; Defining Helper Functions
;;;;;;;;;;;;;;;;;;;;;;;;;;;;


(defun find-in-row-internal (row object position)
  (cond (row (if (equal object (car row))
                 position
                 (find-in-row-internal (cdr row) object (+ position 1))))
        (t nil)))

(defun find-in-row (row object)
  (find-in-row-internal row object 0))


(defun find-in-matrix-internal (matrix object position)
  (cond (matrix (let ((top-row-ans (find-in-row (car matrix) object)))
                    (if top-row-ans
                      (list position top-row-ans)
                      (find-in-matrix-internal (cdr matrix) object (+ position 1)))))
        (t nil)))

(defun find-in-matrix (matrix object)
  (find-in-matrix-internal matrix object 0))


(defun matrix-size (input-matrix)
  (list (length input-matrix) (length (car input-matrix))))

(defun find-yellow-marker (input-matrix)
  (find-in-matrix input-matrix :y))

(defun find-other-car (input-matrix)
  (find-in-matrix input-matrix :c))

(defun find-pedestrian (input-matrix)
  (find-in-matrix input-matrix :p))

(defun find-me (input-matrix)
  (find-in-matrix input-matrix :me))
```

Compile and Load

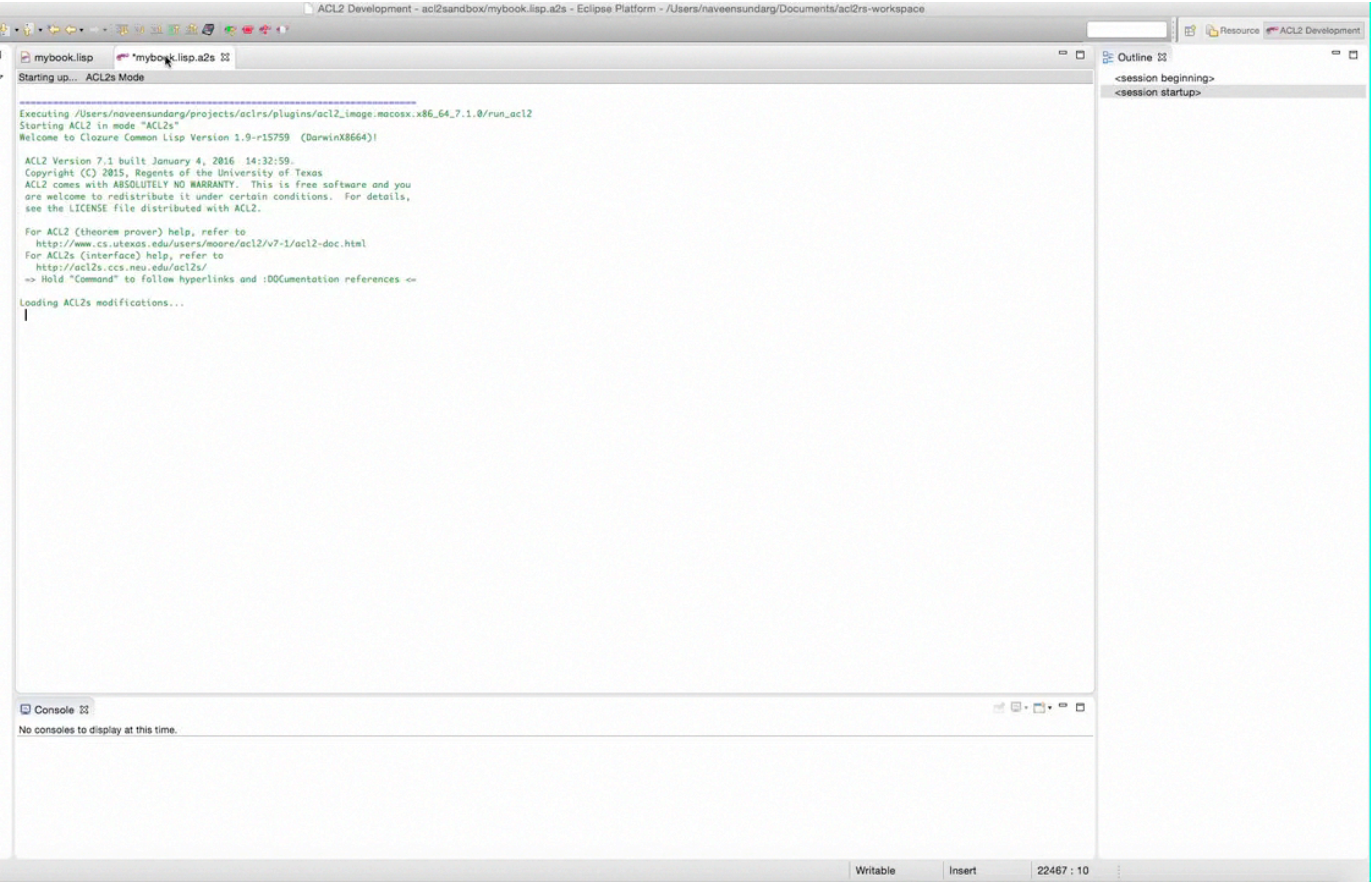


Starting up... ACL2s Mode

```
========================================================================
Executing /Users/naveensundarg/projects/aclrs/plugins/acl2_image.macosx.x86_64_7.1.0/run_acl2
Starting ACL2 in mode "ACL2s"
Welcome to Clozure Common Lisp Version 1.9-r15759  (DarwinX8664)!

 ACL2 Version 7.1 built January 4, 2016  14:32:59.
 Copyright (C) 2015, Regents of the University of Texas
 ACL2 comes with ABSOLUTELY NO WARRANTY.  This is free software and you
 are welcome to redistribute it under certain conditions.  For details,
 see the LICENSE file distributed with ACL2.

 For ACL2 (theorem prover) help, refer to
   http://www.cs.utexas.edu/users/moore/acl2/v7-1/acl2-doc.html
 For ACL2s (interface) help, refer to
   http://acl2s.ccs.neu.edu/acl2s/
 => Hold "Command" to follow hyperlinks and :DOCumentation references <=

Loading ACL2s modifications...
```

No consoles to display at this time.

Compile and Load



ACL2 Development - acl2sandbox/mybook.lisp.a2s - Eclipse Platform - /Users/naveensundarg/Documents/acl2rs-workspace
Resource
ACL2 Development
mybook.lisp
*mybook.lisp.a2s
Outline
Console
Writable
Insert
22467 : 10

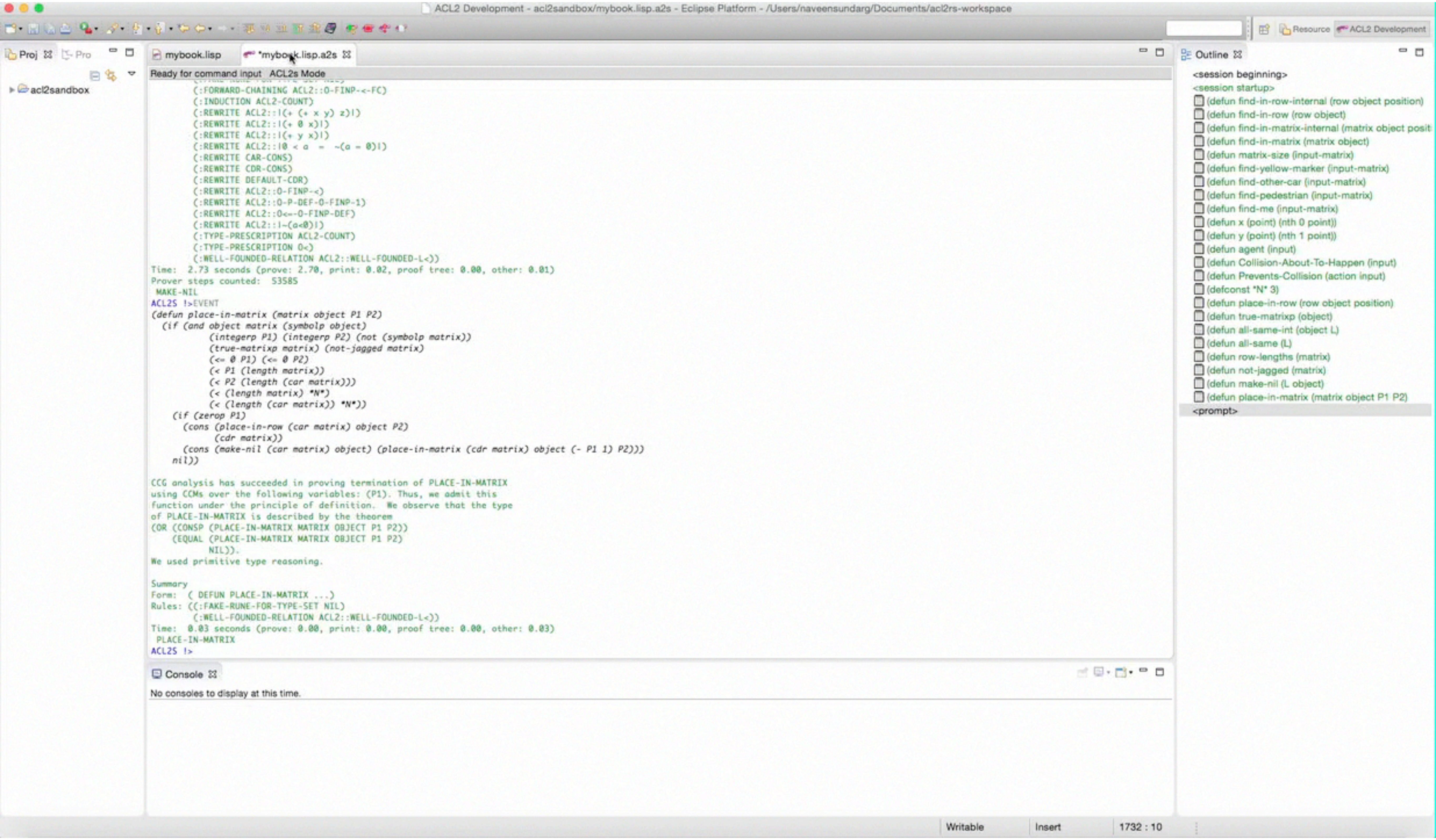
Starting up... ACL2s Mode

```
==========================================================================
Executing /Users/naveensundarg/projects/aclrs/plugins/acl2_image.macosx.x86_64_7.1.0/run_acl2
Starting ACL2 in mode "ACL2s"
Welcome to Clozure Common Lisp Version 1.9-r15759  (DarwinX8664)!

 ACL2 Version 7.1 built January 4, 2016  14:32:59.
 Copyright (C) 2015, Regents of the University of Texas
 ACL2 comes with ABSOLUTELY NO WARRANTY.  This is free software and you
 are welcome to redistribute it under certain conditions.  For details,
 see the LICENSE file distributed with ACL2.

 For ACL2 (theorem prover) help, refer to
   http://www.cs.utexas.edu/users/moore/acl2/v7-1/acl2-doc.html
 For ACL2s (interface) help, refer to
   http://acl2s.ccs.neu.edu/acl2s/
 => Hold "Command" to follow hyperlinks and :DOCumentation references <=

Loading ACL2s modifications...
```

<session beginning>
<session startup>

No consoles to display at this time.

Theorem Proved


ACL2 Development - acl2sandbox/mybook.lisp.a2s - Eclipse Platform - /Users/naveensundarg/Documents/acl2rs-workspace
Resource
ACL2 Development
Proj
Pro
acl2sandbox
mybook.lisp
*mybook.lisp.a2s
Outline
Console
Writable
Insert
1732 : 10

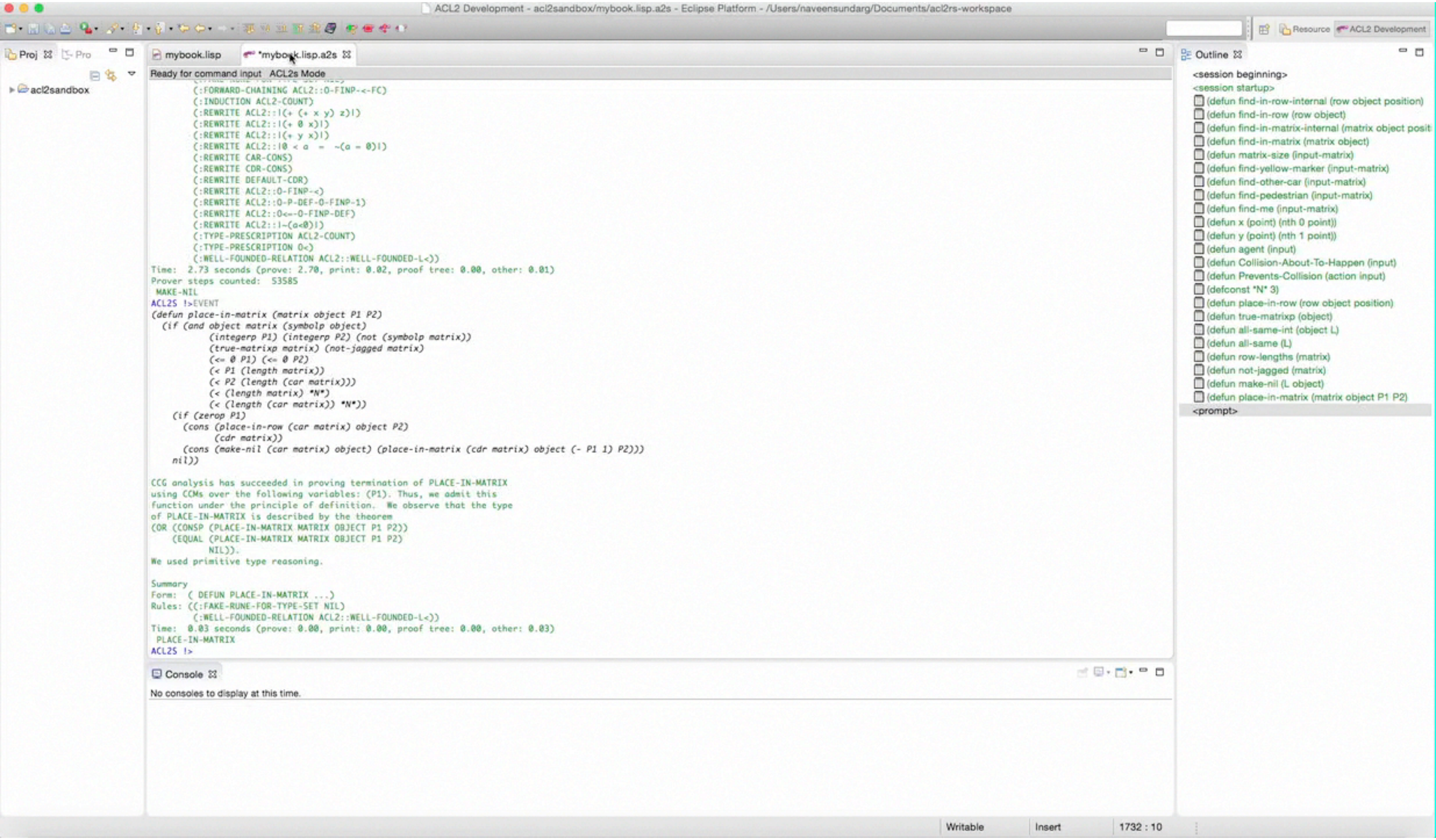
Ready for command input   ACL2s Mode

```
        (:FORWARD-CHAINING ACL2::O-FINP-<-FC)
        (:INDUCTION ACL2-COUNT)
        (:REWRITE ACL2::|(+ (+ x y) z)|)
        (:REWRITE ACL2::|(+ 0 x)|)
        (:REWRITE ACL2::|(+ y x)|)
        (:REWRITE ACL2::|0 < a  =  ~(a = 0)|)
        (:REWRITE CAR-CONS)
        (:REWRITE CDR-CONS)
        (:REWRITE DEFAULT-CDR)
        (:REWRITE ACL2::O-FINP-<)
        (:REWRITE ACL2::O-P-DEF-O-FINP-1)
        (:REWRITE ACL2::O<=-O-FINP-DEF)
        (:REWRITE ACL2::|~(a<0)|)
        (:TYPE-PRESCRIPTION ACL2-COUNT)
        (:TYPE-PRESCRIPTION O<)
        (:WELL-FOUNDED-RELATION ACL2::WELL-FOUNDED-L<))
Time:  2.73 seconds (prove: 2.70, print: 0.02, proof tree: 0.00, other: 0.01)
Prover steps counted:  53585
 MAKE-NIL
ACL2S !>EVENT
(defun place-in-matrix (matrix object P1 P2)
  (if (and object matrix (symbolp object)
           (integerp P1) (integerp P2) (not (symbolp matrix))
           (true-matrixp matrix) (not-jagged matrix)
           (<= 0 P1) (<= 0 P2)
           (< P1 (length matrix))
           (< P2 (length (car matrix)))
           (< (length matrix) *N*)
           (< (length (car matrix)) *N*))
    (if (zerop P1)
      (cons (place-in-row (car matrix) object P2)
            (cdr matrix))
      (cons (make-nil (car matrix) object) (place-in-matrix (cdr matrix) object (- P1 1) P2)))
    nil))

CCG analysis has succeeded in proving termination of PLACE-IN-MATRIX
using CCMs over the following variables: (P1). Thus, we admit this
function under the principle of definition.  We observe that the type
of PLACE-IN-MATRIX is described by the theorem
(OR (CONSP (PLACE-IN-MATRIX MATRIX OBJECT P1 P2))
    (EQUAL (PLACE-IN-MATRIX MATRIX OBJECT P1 P2)
           NIL)).
We used primitive type reasoning.

Summary
Form:  ( DEFUN PLACE-IN-MATRIX ...)
Rules: ((:FAKE-RUNE-FOR-TYPE-SET NIL)
        (:WELL-FOUNDED-RELATION ACL2::WELL-FOUNDED-L<))
Time:  0.03 seconds (prove: 0.00, print: 0.00, proof tree: 0.00, other: 0.03)
 PLACE-IN-MATRIX
ACL2S !>
```

No consoles to display at this time.

```
<session beginning>
<session startup>
(defun find-in-row-internal (row object position)
(defun find-in-row (row object)
(defun find-in-matrix-internal (matrix object posit
(defun find-in-matrix (matrix object)
(defun matrix-size (input-matrix)
(defun find-yellow-marker (input-matrix)
(defun find-other-car (input-matrix)
(defun find-pedestrian (input-matrix)
(defun find-me (input-matrix)
(defun x (point) (nth 0 point))
(defun y (point) (nth 1 point))
(defun agent (input)
(defun Collision-About-To-Happen (input)
(defun Prevents-Collision (action input)
(defconst *N* 3)
(defun place-in-row (row object position)
(defun true-matrixp (object)
(defun all-same-int (object L)
(defun all-same (L)
(defun row-lengths (matrix)
(defun not-jagged (matrix)
(defun make-nil (L object)
(defun place-in-matrix (matrix object P1 P2)
<prompt>
```

Theorem Proved



```
        (:FORWARD-CHAINING ACL2::O-FINP-<-FC)
        (:INDUCTION ACL2-COUNT)
        (:REWRITE ACL2::|(+ (+ x y) z)|)
        (:REWRITE ACL2::|(+ 0 x)|)
        (:REWRITE ACL2::|(+ y x)|)
        (:REWRITE ACL2::|0 < a  =  ~(a = 0)|)
        (:REWRITE CAR-CONS)
        (:REWRITE CDR-CONS)
        (:REWRITE DEFAULT-CDR)
        (:REWRITE ACL2::O-FINP-<)
        (:REWRITE ACL2::O-P-DEF-O-FINP-1)
        (:REWRITE ACL2::O<=-O-FINP-DEF)
        (:REWRITE ACL2::|~(a<0)|)
        (:TYPE-PRESCRIPTION ACL2-COUNT)
        (:TYPE-PRESCRIPTION O<)
        (:WELL-FOUNDED-RELATION ACL2::WELL-FOUNDED-L<))
Time:  2.73 seconds (prove: 2.70, print: 0.02, proof tree: 0.00, other: 0.01)
Prover steps counted:  53585
 MAKE-NIL
ACL2S !>EVENT
(defun place-in-matrix (matrix object P1 P2)
  (if (and object matrix (symbolp object)
           (integerp P1) (integerp P2) (not (symbolp matrix))
           (true-matrixp matrix) (not-jagged matrix)
           (<= 0 P1) (<= 0 P2)
           (< P1 (length matrix))
           (< P2 (length (car matrix)))
           (< (length matrix) *N*)
           (< (length (car matrix)) *N*))
    (if (zerop P1)
      (cons (place-in-row (car matrix) object P2)
            (cdr matrix))
      (cons (make-nil (car matrix) object) (place-in-matrix (cdr matrix) object (- P1 1) P2)))
    nil))

CCG analysis has succeeded in proving termination of PLACE-IN-MATRIX
using CCMs over the following variables: (P1). Thus, we admit this
function under the principle of definition.  We observe that the type
of PLACE-IN-MATRIX is described by the theorem
(OR (CONSP (PLACE-IN-MATRIX MATRIX OBJECT P1 P2))
    (EQUAL (PLACE-IN-MATRIX MATRIX OBJECT P1 P2)
           NIL)).
We used primitive type reasoning.

Summary
Form:  ( DEFUN PLACE-IN-MATRIX ...)
Rules: ((:FAKE-RUNE-FOR-TYPE-SET NIL)
        (:WELL-FOUNDED-RELATION ACL2::WELL-FOUNDED-L<))
Time:  0.03 seconds (prove: 0.00, print: 0.00, proof tree: 0.00, other: 0.03)
 PLACE-IN-MATRIX
ACL2S !>
```

No consoles to display at this time.

```
<session beginning>
<session startup>
□ (defun find-in-row-internal (row object position)
□ (defun find-in-row (row object)
□ (defun find-in-matrix-internal (matrix object posit
□ (defun find-in-matrix (matrix object)
□ (defun matrix-size (input-matrix)
□ (defun find-yellow-marker (input-matrix)
□ (defun find-other-car (input-matrix)
□ (defun find-pedestrian (input-matrix)
□ (defun find-me (input-matrix)
□ (defun x (point) (nth 0 point))
□ (defun y (point) (nth 1 point))
□ (defun agent (input)
□ (defun Collision-About-To-Happen (input)
□ (defun Prevents-Collision (action input)
□ (defconst *N* 3)
□ (defun place-in-row (row object position)
□ (defun true-matrixp (object)
□ (defun all-same-int (object L)
□ (defun all-same (L)
□ (defun row-lengths (matrix)
□ (defun not-jagged (matrix)
□ (defun make-nil (L object)
□ (defun place-in-matrix (matrix object P1 P2)
<prompt>
```

II.
Early Progress With Our Calculi: Non-Akratic Robots

Informal Definition of Akrasia

An action α_f is (Augustinian) akratic for an agent A at t_{α_f} iff the following eight conditions hold:

(1) A believes that A ought to do α_o at t_{α_o};

(2) A desires to do α_f at t_{α_f};

(3) A's doing α_f at t_{α_f} entails his not doing α_o at t_{α_o};

(4) A knows that doing α_f at t_{α_f} entails his not doing α_o at t_{α_o};

(5) At the time (t_{α_f}) of doing the forbidden α_f, A's desire to do α_f overrides A's belief that he ought to do α_o at t_{α_f}.

(6) A does the forbidden action α_f at t_{α_f};

(7) A's doing α_f results from A's desire to do α_f;

(8) At some time t after t_{α_f}, A has the belief that A ought to have done α_o rather than α_f.

Informal Definition of Akrasia

An action α_f is (Augustinian) akratic for an agent A at t_{α_f} iff the following eight conditions hold:

(1) A believes that A ought to do α_o at t_{α_o};

(2) A desires to do α_f at t_{α_f};

(3) A's doing α_f at t_{α_f} entails his not doing α_o at t_{α_o};

(4) A knows that doing α_f at t_{α_f} entails his not doing α_o at t_{α_o};

(5) At the time (t_{α_f}) of doing the forbidden α_f, A's desire to do α_f overrides A's belief that he ought to do α_o at t_{α_f}.

(6) A does the forbidden action α_f at t_{α_f};

(7) A's doing α_f results from A's desire to do α_f;

(8) At some time t after t_{α_f}, A has the belief that A ought to have done α_o rather than α_f.

Informal Definition of Akrasia

An action α_f is (Augustinian) akratic for an agent A at t_{α_f} iff the following eight conditions hold:

(1) A believes that A ought to do α_o at t_{α_o};

(2) A desires to do α_f at t_{α_f};

(3) A's doing α_f at t_{α_f} entails his not doing α_o at t_{α_o};

(4) A knows that doing α_f at t_{α_f} entails his not doing α_o at t_{α_o};

(5) At the time (t_{α_f}) of doing the forbidden α_f, A's desire to do α_f overrides A's belief that he ought to do α_o at t_{α_f}.

(6) A does the forbidden action α_f at t_{α_f};

(7) A's doing α_f results from A's desire to do α_f;

"Regret" (8) At some time t after t_{α_f}, A has the belief that A ought to have done α_o rather than α_f.

Cast in

$\mathcal{DCEC}^*$

this becomes …

$$\mathsf{KB}_{rs} \cup \mathsf{KB}_{m_1} \cup \mathsf{KB}_{m_2} \ldots \mathsf{KB}_{m_n} \vdash$$

$$D_1 : \mathbf{B}(\mathsf{I}, \mathsf{now}, \mathbf{O}(\mathsf{I}^*, t_\alpha \Phi, \mathit{happens}(\mathit{action}(\mathsf{I}^*, \alpha), t_\alpha)))$$

$$D_2 : \mathbf{D}(\mathsf{I}, \mathsf{now}, \mathit{holds}(\mathit{does}(\mathsf{I}^*, \overline{\alpha}), t_{\overline{\alpha}}))$$

$$D_3 : \mathit{happens}(\mathit{action}(\mathsf{I}^*, \overline{\alpha}), t_{\overline{\alpha}}) \Rightarrow \neg \mathit{happens}(\mathit{action}(\mathsf{I}^*, \alpha), t_\alpha)$$

$$D_4 : \mathbf{K}\left(\mathsf{I}, \mathsf{now}, \begin{pmatrix} \mathit{happens}(\mathit{action}(\mathsf{I}^*, \overline{\alpha}), t_{\overline{\alpha}}) \Rightarrow \\ \neg \mathit{happens}(\mathit{action}(\mathsf{I}^*, \alpha), t_\alpha) \end{pmatrix}\right)$$

$$D_5 : \begin{array}{l} \mathbf{I}(\mathsf{I}, t_\alpha, \mathit{happens}(\mathit{action}(\mathsf{I}^*, \overline{\alpha}), t_{\overline{\alpha}}) \wedge \\ \neg \mathbf{I}(\mathsf{I}, t_\alpha, \mathit{happens}(\mathit{action}(\mathsf{I}^*, \alpha), t_\alpha) \end{array}$$

$$D_6 : \mathit{happens}(\mathit{action}(\mathsf{I}^*, \overline{\alpha}), t_{\overline{\alpha}})$$

$$D_{7a} : \begin{array}{l} \Gamma \cup \{\mathbf{D}(\mathsf{I}, \mathsf{now}, \mathit{holds}(\mathit{does}(\mathsf{I}^*, \overline{\alpha}), t))\} \vdash \\ \quad \mathit{happens}(\mathit{action}(\mathsf{I}^*, \overline{\alpha}), t_\alpha) \end{array}$$

$$D_{7b} : \begin{array}{l} \Gamma - \{\mathbf{D}(\mathsf{I}, \mathsf{now}, \mathit{holds}(\mathit{does}(\mathsf{I}^*, \overline{\alpha}), t))\} \nvdash \\ \quad \mathit{happens}(\mathit{action}(\mathsf{I}^*, \overline{\alpha}), t_\alpha) \end{array}$$

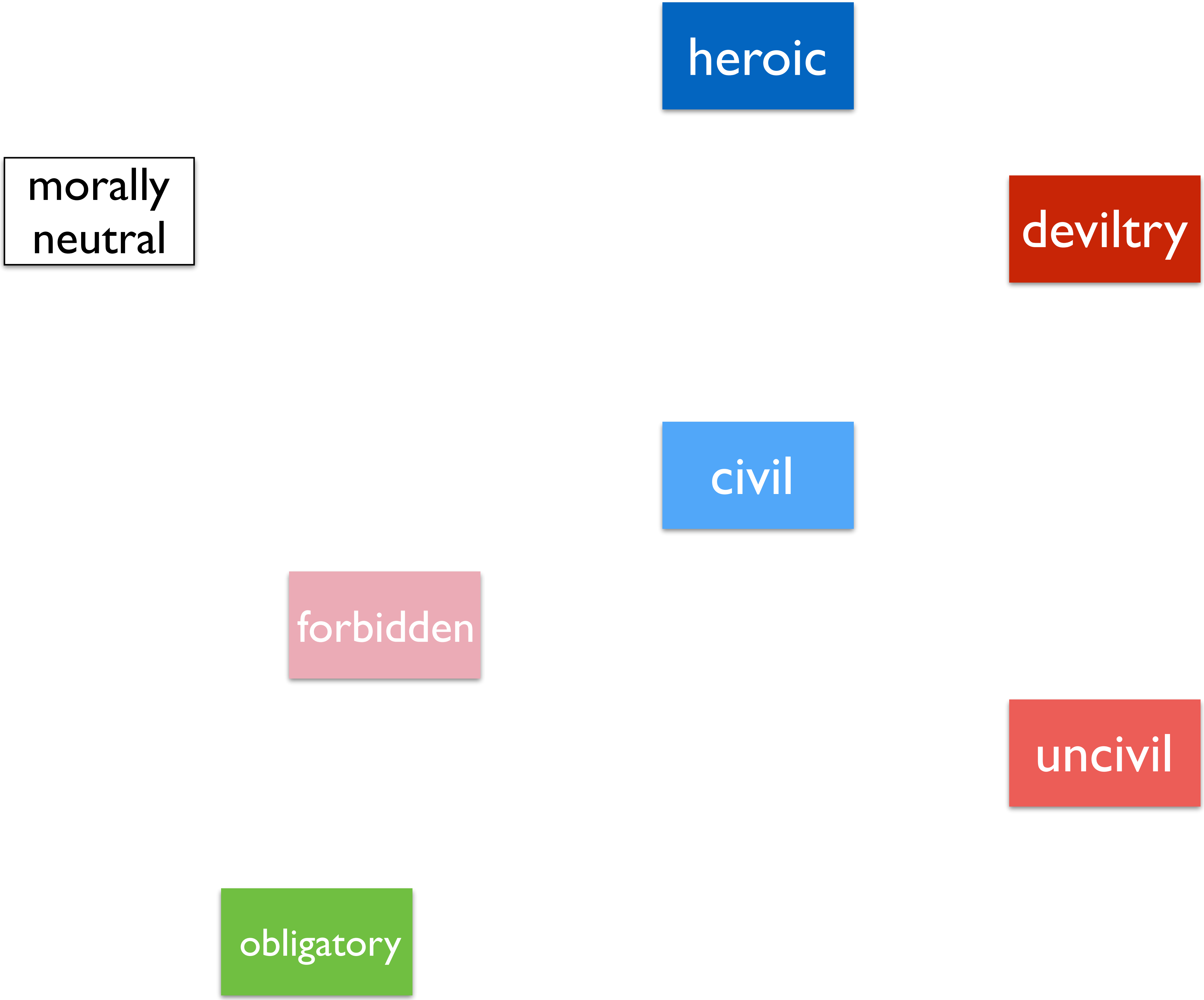
$$D_8 : \mathbf{B}\big(\mathsf{I}, t_f, \mathbf{O}(\mathsf{I}^*, t_\alpha, \Phi, \mathit{happens}(\mathit{action}(\mathsf{I}^*, \alpha), t_\alpha))\big)$$

Demos …

Demos …

III.
But, a twist befell the logicists …

Chisholm had argued that the three old 19th-century ethical categories (*forbidden*, *morally neutral*, *obligatory*) are not enough — and soul-searching brought me to agreement.



heroic

morally neutral

deviltry

civil

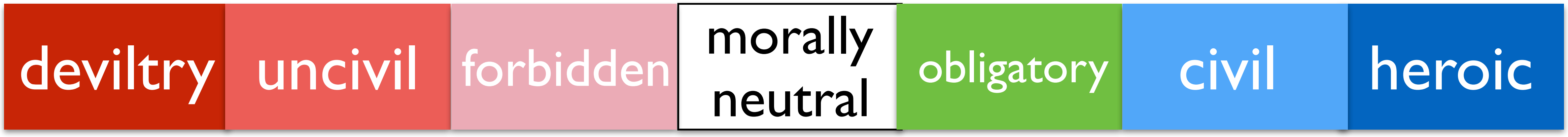
forbidden

uncivil

obligatory

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathscr{EH}$



deviltry

uncivil

forbidden

morally neutral

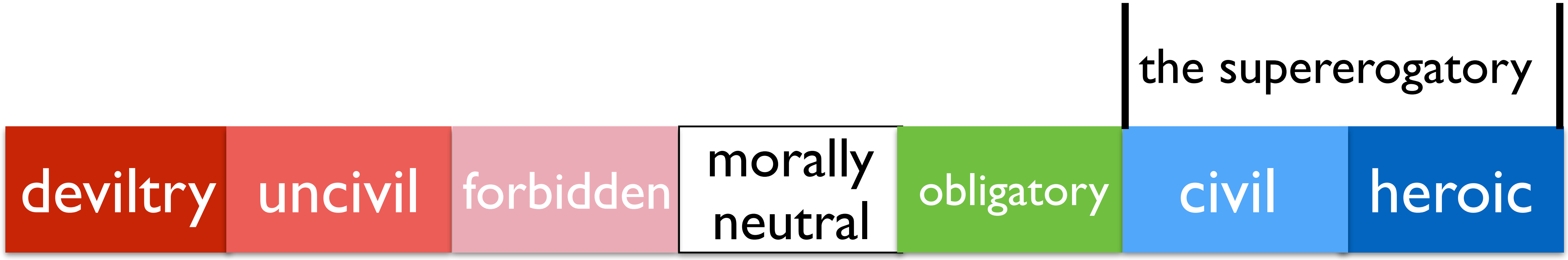
obligatory

civil

heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathcal{EH}$

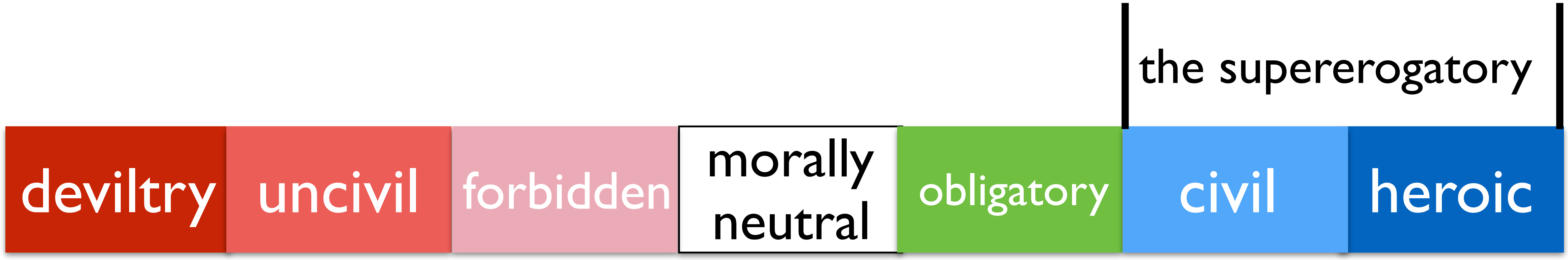


the supererogatory

deviltry uncivil forbidden morally neutral obligatory civil heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathcal{EH}$



the supererogatory

deviltry

uncivil

forbidden

morally neutral

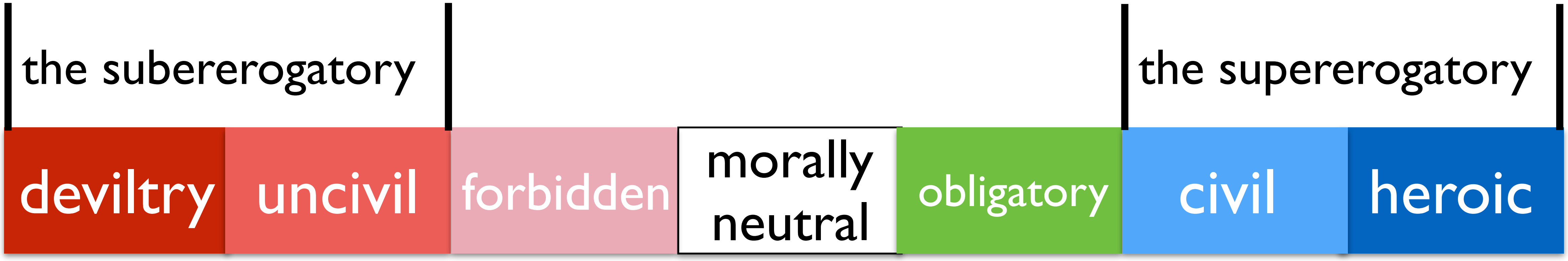
obligatory

civil

heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathcal{EH}$



the subererogatory

the supererogatory

deviltry

uncivil

forbidden

morally neutral

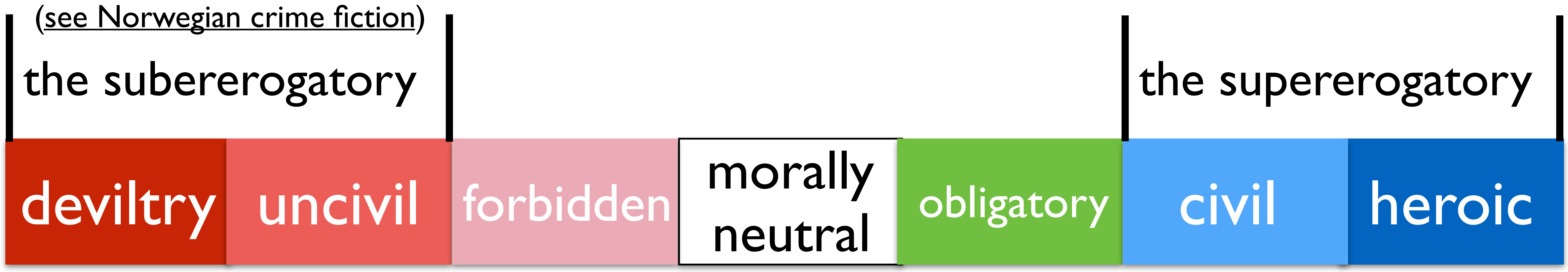
obligatory

civil

heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathcal{EH}$



(see Norwegian crime fiction)

the subererogatory

the supererogatory

devilty

uncivil

forbidden

morally neutral

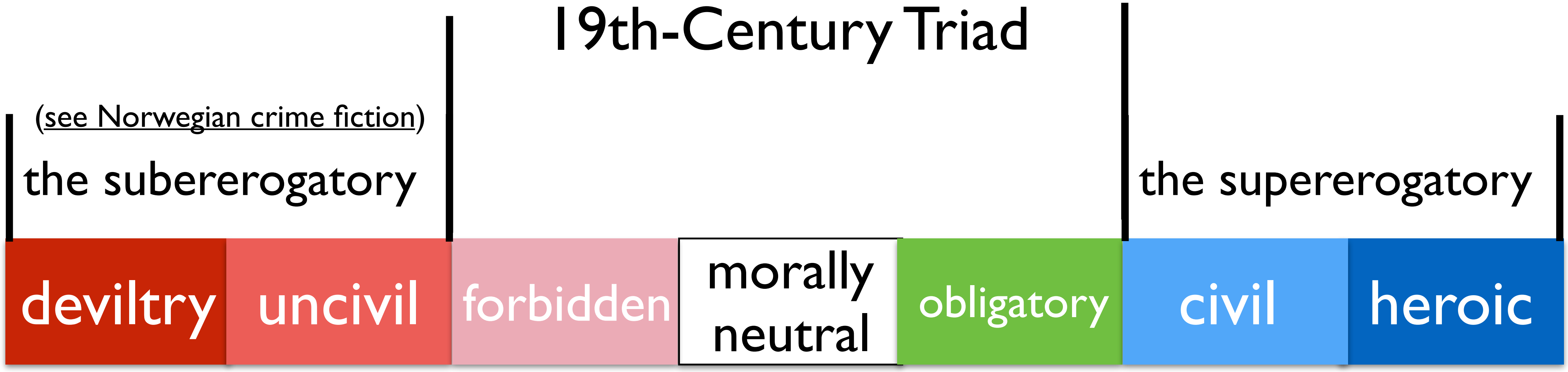
obligatory

civil

heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathcal{EH}$



19th-Century Triad

(see Norwegian crime fiction)

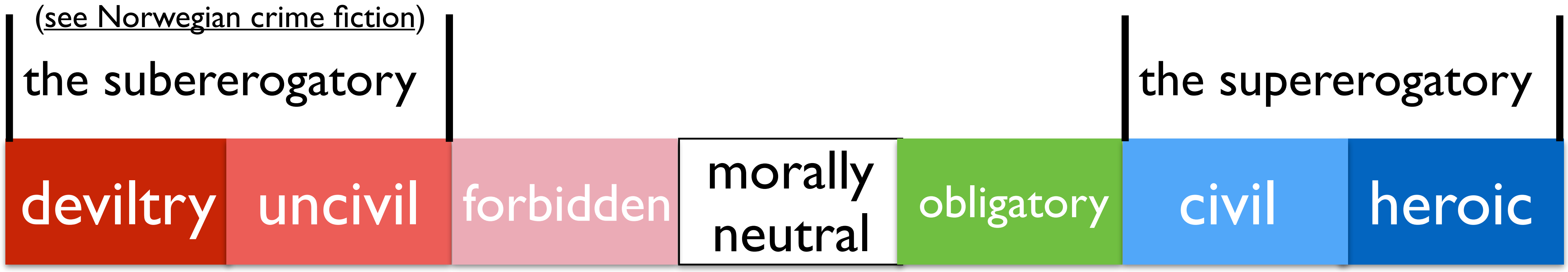
the subererogatory

the supererogatory

deviltry | uncivil | forbidden | morally neutral | obligatory | civil | heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathcal{EH}$



(see Norwegian crime fiction)

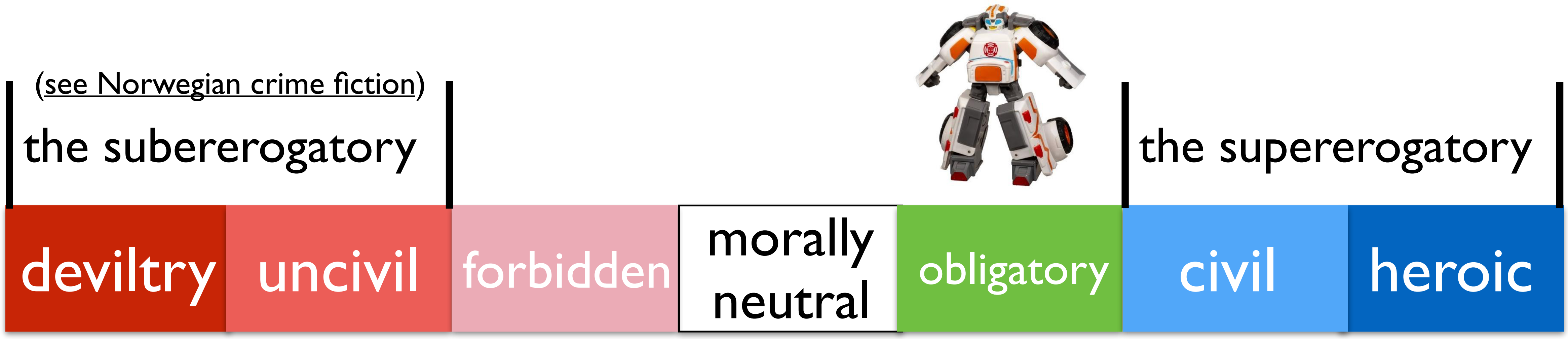
the subererogatory

the supererogatory

deviltry | uncivil | forbidden | morally neutral | obligatory | civil | heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathcal{EH}$



(see Norwegian crime fiction)

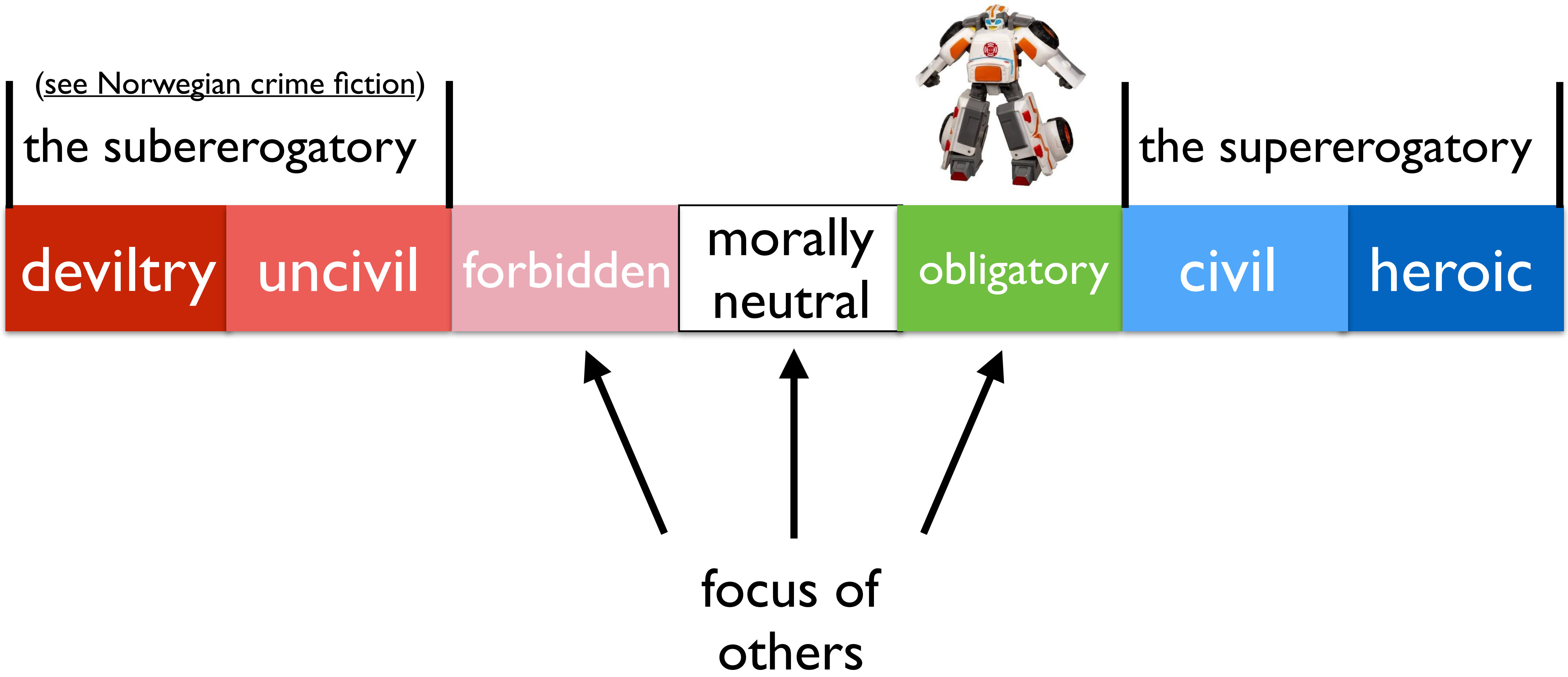
the subererogatory

the supererogatory

devilt ry | uncivil | forbidden | morally neutral | obligatory | civil | heroic

Leibnizian Ethical Hierarchy for Persons and Robots:

$\mathscr{EH}$



(see Norwegian crime fiction)

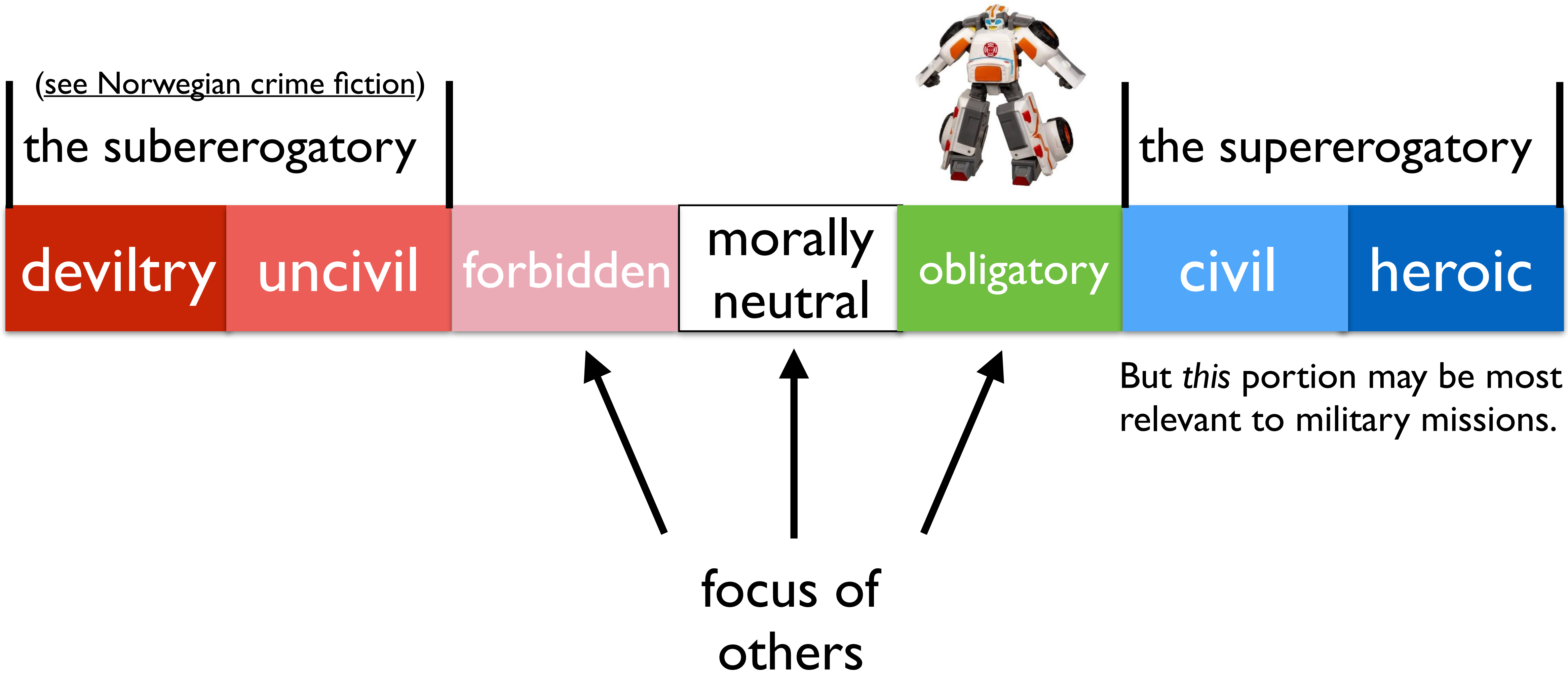
the subererogatory

the supererogatory

devilry | uncivil | forbidden | morally neutral | obligatory | civil | heroic

focus of others

Leibnizian Ethical Hierarchy for Persons and Robots: $\mathcal{EH}$



(see Norwegian crime fiction)

the subererogatory

the supererogatory

deviltry | uncivil | forbidden | morally neutral | obligatory | civil | heroic

But *this* portion may be most relevant to military missions.

focus of others

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$$\begin{array}{ccccc|c|ccccc} \multicolumn{5}{c|}{\mathcal{F}} & \mathcal{P} \wedge \neg\mathcal{O} & \multicolumn{5}{c}{\mathcal{O}} \\ \forall & \mathtt{F} & \mathtt{M} & \mathtt{V} & \exists & & \forall & \mathtt{F} & \mathtt{M} & \mathtt{V} & \exists \end{array}$$

$\mathscr{T} := \|\mathcal{F}|\mathcal{P}\wedge\neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathcal{F}$	$\mathcal{P}\wedge\neg\mathcal{O}$	$\mathcal{O}$
$\forall$ **F** **M** **V** $\exists$		$\forall$ **F** **M** **V** $\exists$

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P}\wedge\neg\mathcal{O}$	$\mathcal{O}^L$	$\mathcal{O}^M$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$		$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$
						$\uparrow$	

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}$
$\forall \quad \mathbf{F} \quad \mathbf{M} \quad \mathbf{V} \quad \exists$		$\forall \quad \mathbf{F} \quad \mathbf{M} \quad \mathbf{V} \quad \exists$

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}^L$	$\mathcal{O}^M$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
∃–∀	∃–∀	∃–∀		∃–∀	∃–∀	∃–∀	∃–∀
						↑	
				●			

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}$
$\forall$ **F** **M** **V** $\exists$		$\forall$ **F** **M** **V** $\exists$

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}^L$	$\mathcal{O}^M$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$		$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$
				●		↑	

Arkin
Pereira
Andersons
Powers
Mikhail
…

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

		$\mathcal{F}$			$\mathcal{P} \wedge \neg\mathcal{O}$			$\mathcal{O}$		
$\forall$	**F**	**M**	**V**	$\exists$		$\forall$	**F**	**M**	**V**	$\exists$

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}^L$	$\mathcal{O}^M$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$		$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$
						↑	
				●			

$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

		$\mathcal{F}$			$\mathcal{P} \wedge \neg\mathcal{O}$			$\mathcal{O}$		
∀	**F**	**M**	**V**	∃		∀	**F**	**M**	**V**	∃

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}^L$	$\mathcal{O}^M$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
∃–∀	∃–∀	∃–∀		∃–∀	∃–∀	∃–∀	∃–∀
				●		↑	

R A I R
Rensselaer AI and Reasoning Lab

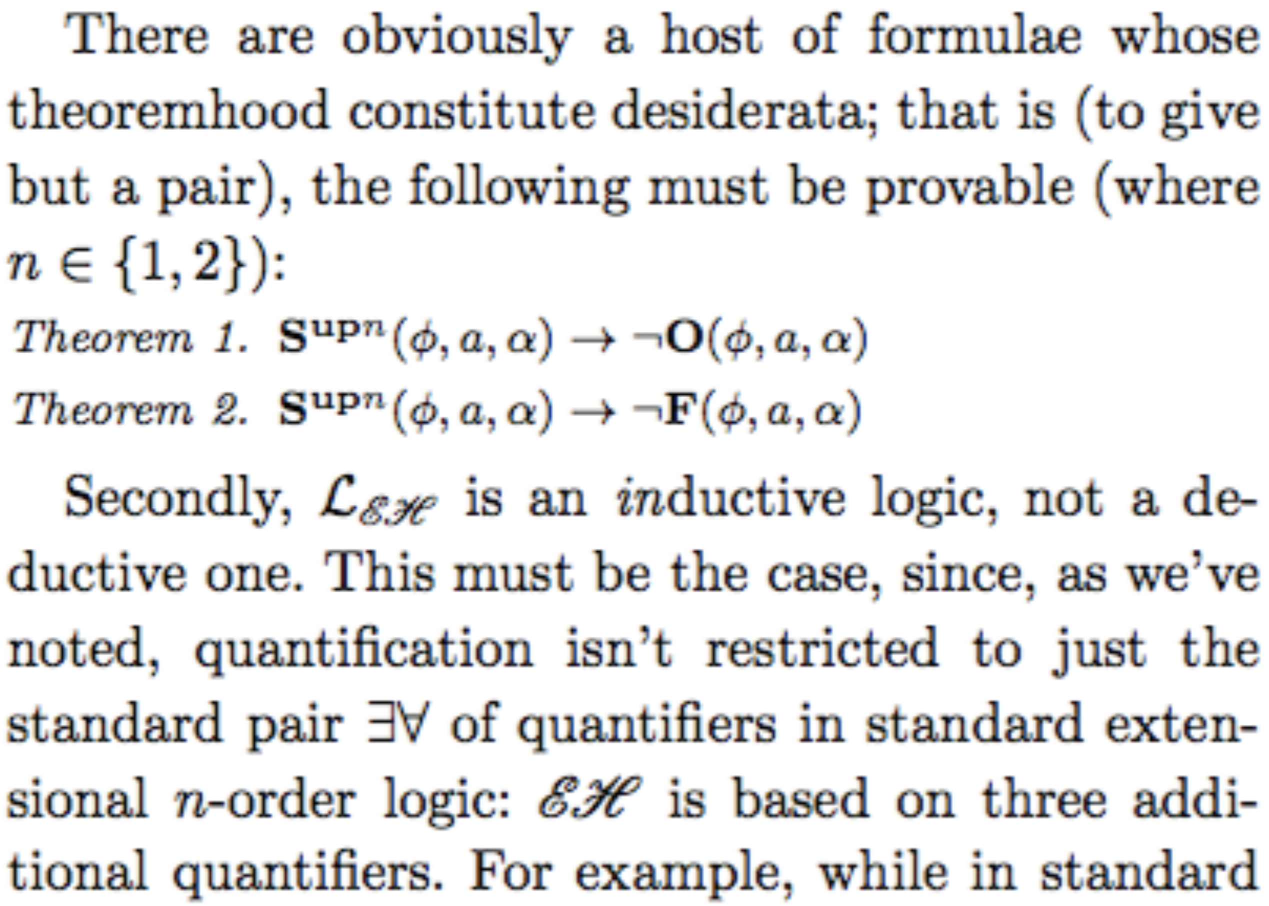
$\mathscr{T} := \|\mathcal{F}|\mathcal{P} \wedge \neg\mathcal{O}|\mathcal{O}\|$ 19th Century Triad

$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}$
$\forall$ $\mathtt{F}$ $\mathtt{M}$ $\mathtt{V}$ $\exists$		$\forall$ $\mathtt{F}$ $\mathtt{M}$ $\mathtt{V}$ $\exists$

$\mathscr{EH}$

$\mathcal{S}_{ub1}$	$\mathcal{S}_{ub2}$	$\mathcal{F}$	$\mathcal{P} \wedge \neg\mathcal{O}$	$\mathcal{O}^{L}$	$\mathcal{O}^{M}$	$\mathcal{S}^{up1}$	$\mathcal{S}^{up2}$
$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$		$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$	$\exists$–$\forall$
				●		↑	

RAIR
Rensselaer AI and Reasoning Lab



There are obviously a host of formulae whose theoremhood constitute desiderata; that is (to give but a pair), the following must be provable (where $n \in \{1, 2\}$):

Theorem 1. $\mathbf{S}^{\mathbf{up}^n}(\phi, a, \alpha) \rightarrow \neg\mathbf{O}(\phi, a, \alpha)$

Theorem 2. $\mathbf{S}^{\mathbf{up}^n}(\phi, a, \alpha) \rightarrow \neg\mathbf{F}(\phi, a, \alpha)$

Secondly, $\mathcal{L}_{\mathscr{EH}}$ is an *in*ductive logic, not a deductive one. This must be the case, since, as we've noted, quantification isn't restricted to just the standard pair $\exists\forall$ of quantifiers in standard extensional n-order logic: $\mathscr{EH}$ is based on three additional quantifiers. For example, while in standard

Bert "Heroically" Saved?

Courtesy of RAIR-Lab Researcher Atriya Sen

Bert "Heroically" Saved?

Courtesy of RAIR-Lab Researcher Atriya Sen

Supererogatory^2 Robot Action

Courtesy of RAIR-Lab Researcher Atriya Sen

Courtesy of RAIR-Lab Researcher Atriya Sen

Bert “Heroically” Saved!!

Courtesy of RAIR-Lab Researcher Atriya Sen

Bert “Heroically” Saved!!

Courtesy of RAIR-Lab Researcher Atriya Sen

Courtesy of RAIR-Lab Researcher Atriya Sen

$$K\left(\text{nao}, t_1, \text{lessthan}\left(\text{payoff}\left(\text{nao}^*, \neg\text{dive}, t_2\right), \text{threshold}\right)\right)$$
$$K\left(\text{nao}, t_1, \text{greaterthan}\left(\text{payoff}\left(\text{nao}^*, \text{dive}, t_2\right), \text{threshold}\right)\right)$$
$$K\left(\text{nao}, t_1, \neg O\left(\text{nao}^*, t_2, \text{lessthan}\left(\text{payoff}\left(\text{nao}^*, \neg\text{dive}, t_2\right), \text{threshold}\right), \text{happens}\left(\text{action}\left(\text{nao}^*, \text{dive}\right), t_2\right)\right)\right)$$
$$\therefore K\left(\text{nao}, t_1, S^{\text{UP2}}\left(\text{nao}, t_2, \text{happens}\left(\text{action}\left(\text{nao}^*, \text{dive}\right), t_2\right)\right)\right)$$
$$\therefore I\left(\text{nao}, t_2, \text{happens}\left(\text{action}\left(\text{nao}^*, \text{dive}\right), t_2\right)\right)$$
$$\therefore \text{happens}\left(\text{action}(\text{nao}, \text{dive}), t_2\right)$$

Courtesy of RAIR-Lab Researcher Atriya Sen

$$K\left(\mathrm{nao}, t_1, \mathrm{lessthan}\left(\mathrm{payoff}\left(\mathrm{nao}^*, \neg\mathrm{dive}, t_2\right), \mathrm{threshold}\right)\right)$$

$$K\left(\mathrm{nao}, t_1, \mathrm{greaterthan}\left(\mathrm{payoff}\left(\mathrm{nao}^*, \mathrm{dive}, t_2\right), \mathrm{threshold}\right)\right)$$

$$K\left(\mathrm{nao}, t_1, \neg O\left(\mathrm{nao}^*, t_2, \mathrm{lessthan}\left(\mathrm{payoff}\left(\mathrm{nao}^*, \neg\mathrm{dive}, t_2\right), \mathrm{threshold}\right), \mathrm{happens}\left(\mathrm{action}\left(\mathrm{nao}^*, \mathrm{dive}\right), t_2\right)\right)\right)$$

$$\therefore K\left(\mathrm{nao}, t_1, S^{\mathrm{UP2}}\left(\mathrm{nao}, t_2, \mathrm{happens}\left(\mathrm{action}\left(\mathrm{nao}^*, \mathrm{dive}\right), t_2\right)\right)\right.$$

$$\therefore I\left(\mathrm{nao}, t_2, \mathrm{happens}\left(\mathrm{action}\left(\mathrm{nao}^*, \mathrm{dive}\right), t_2\right)\right)$$

$$\therefore \mathrm{happens}\left(\mathrm{action}(\mathrm{nao}, \mathrm{dive}), t_2\right)$$

Courtesy of RAIR-Lab Researcher Atriya Sen

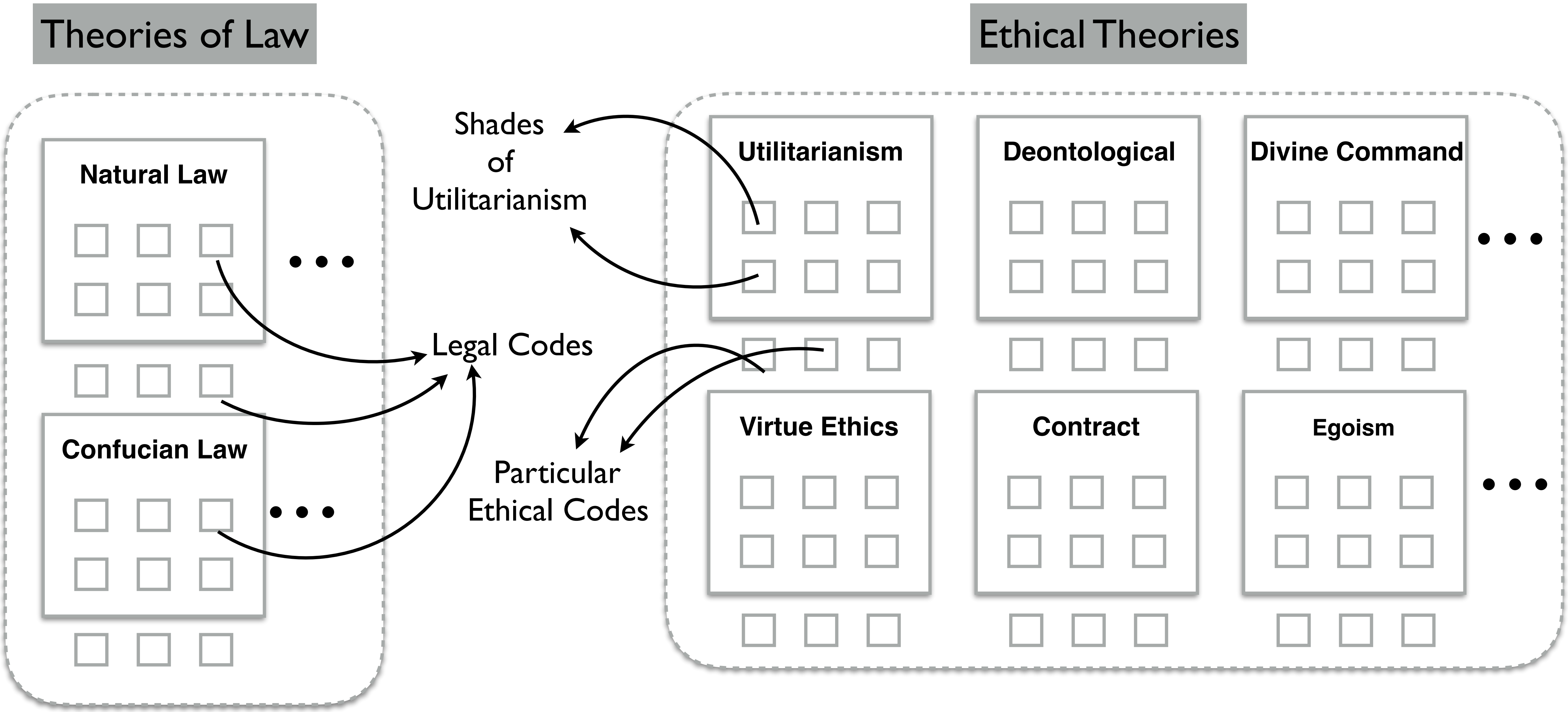
In Talos (available via Web interface); & ShadowProver

```
Prototypes:
Boolean lessThan Numeric Numeric
Boolean greaterThan Numeric Numeric
ActionType not ActionType
ActionType dive

Axioms:
lessOrEqual(Moment t1,t2)
K(nao,t1,lessThan(payoff(nao,not(dive),t2),threshold))
K(nao,t1,greaterThan(payoff(nao,dive,t2),threshold))
K(nao,t1,not(O(nao,t2,lessThan(payoff(nao,not(dive),t2),threshold),happens(action(nao,dive),t2))))

provable Conjectures:
happens(action(nao,dive),t2)
K(nao,t1,SUP2(nao,t2,happens(action(nao,dive),t2)))
I(nao,t2,happens(action(nao,dive),t2))
```

In Talos (available via Web interface); & ShadowProver

```
Prototypes:
Boolean lessThan Numeric Numeric
Boolean greaterThan Numeric Numeric
ActionType not ActionType
ActionType dive

Axioms:
lessOrEqual(Moment t1,t2)
K(nao,t1,lessThan(payoff(nao,not(dive),t2),threshold))
K(nao,t1,greaterThan(payoff(nao,dive,t2),threshold))
K(nao,t1,not(O(nao,t2,lessThan(payoff(nao,not(dive),t2),threshold),happens(action(nao,dive),t2))))

provable Conjectures:
happens(action(nao,dive),t2)
K(nao,t1,SUP2(nao,t2,happens(action(nao,dive),t2)))
I(nao,t2,happens(action(nao,dive),t2))
```

Making Moral Machines **Making Meta Moral Machines**



Theories of Law

Natural Law

Confucian Law

Legal Codes

Shades of Utilitarianism

Particular Ethical Codes

Ethical Theories

Utilitarianism

Deontological

Divine Command

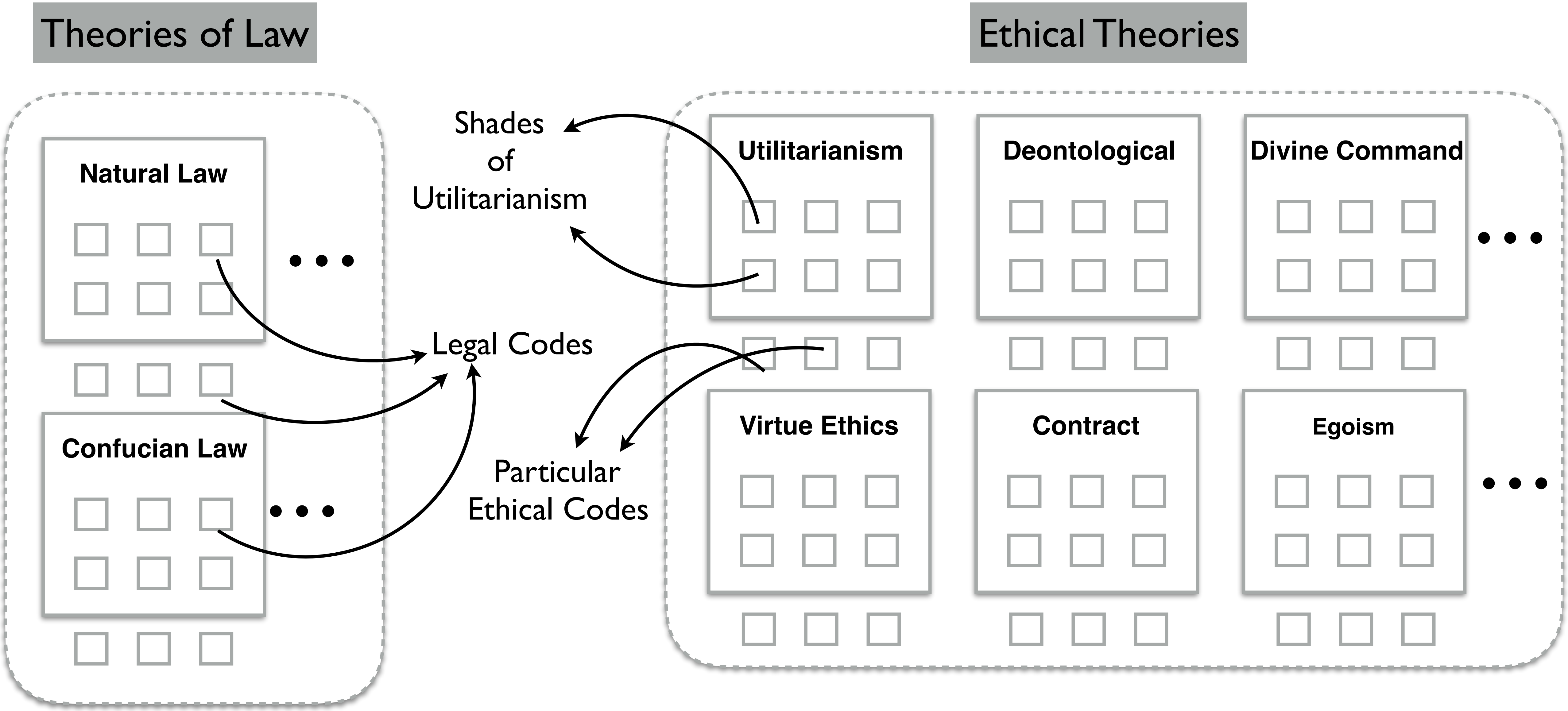
Virtue Ethics

Contract

Egoism

Making Moral Machines

Making Meta Moral Machines



Theories of Law

Natural Law

Confucian Law

Legal Codes

Shades of Utilitarianism

Particular Ethical Codes

Ethical Theories

Utilitarianism

Deontological

Divine Command

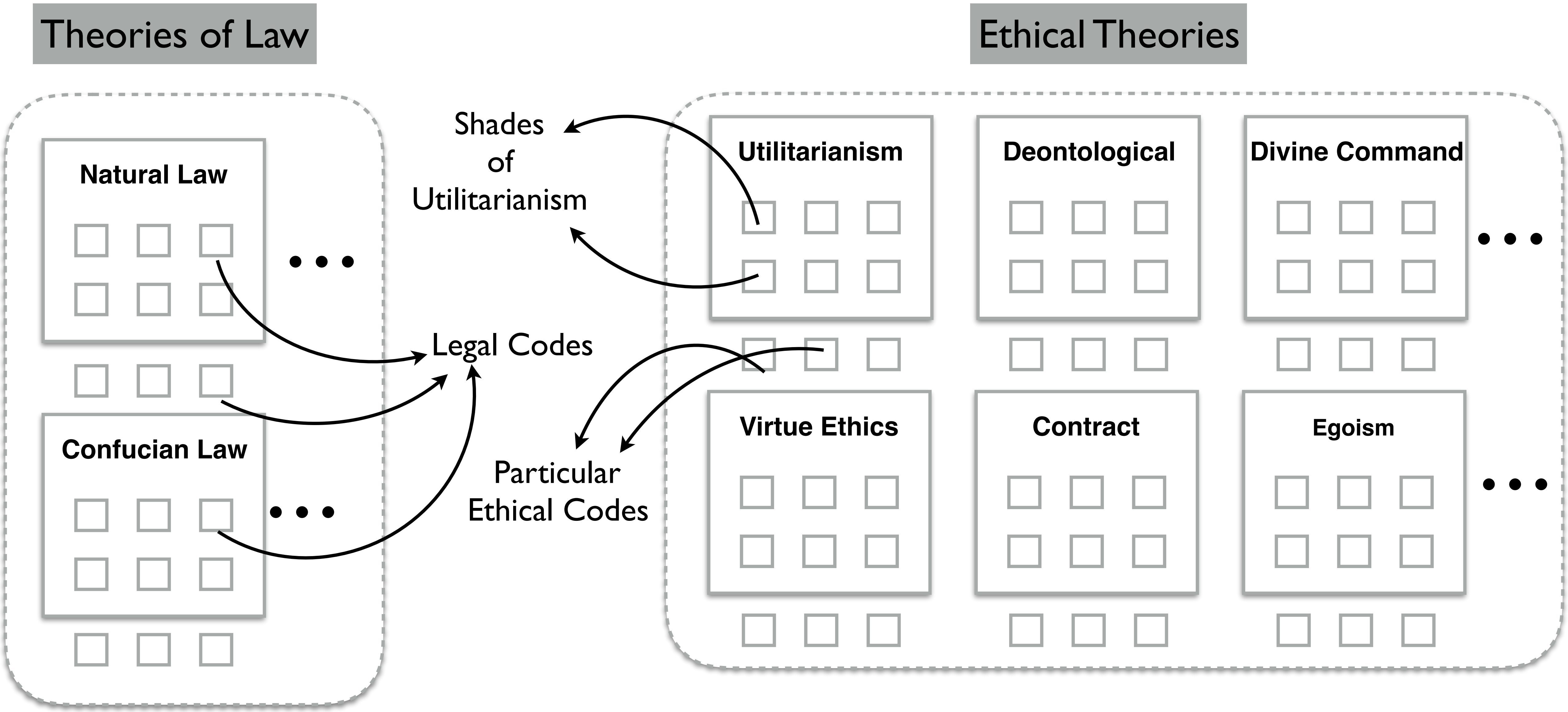
Virtue Ethics

Contract

Egoism

Making Moral Machines

Making Meta Moral Machines



Theories of Law

Natural Law

Confucian Law

• • •

• • •

Legal Codes

Shades of Utilitarianism

Particular Ethical Codes

Ethical Theories

Utilitarianism

Deontological

Divine Command

Virtue Ethics

Contract

Egoism

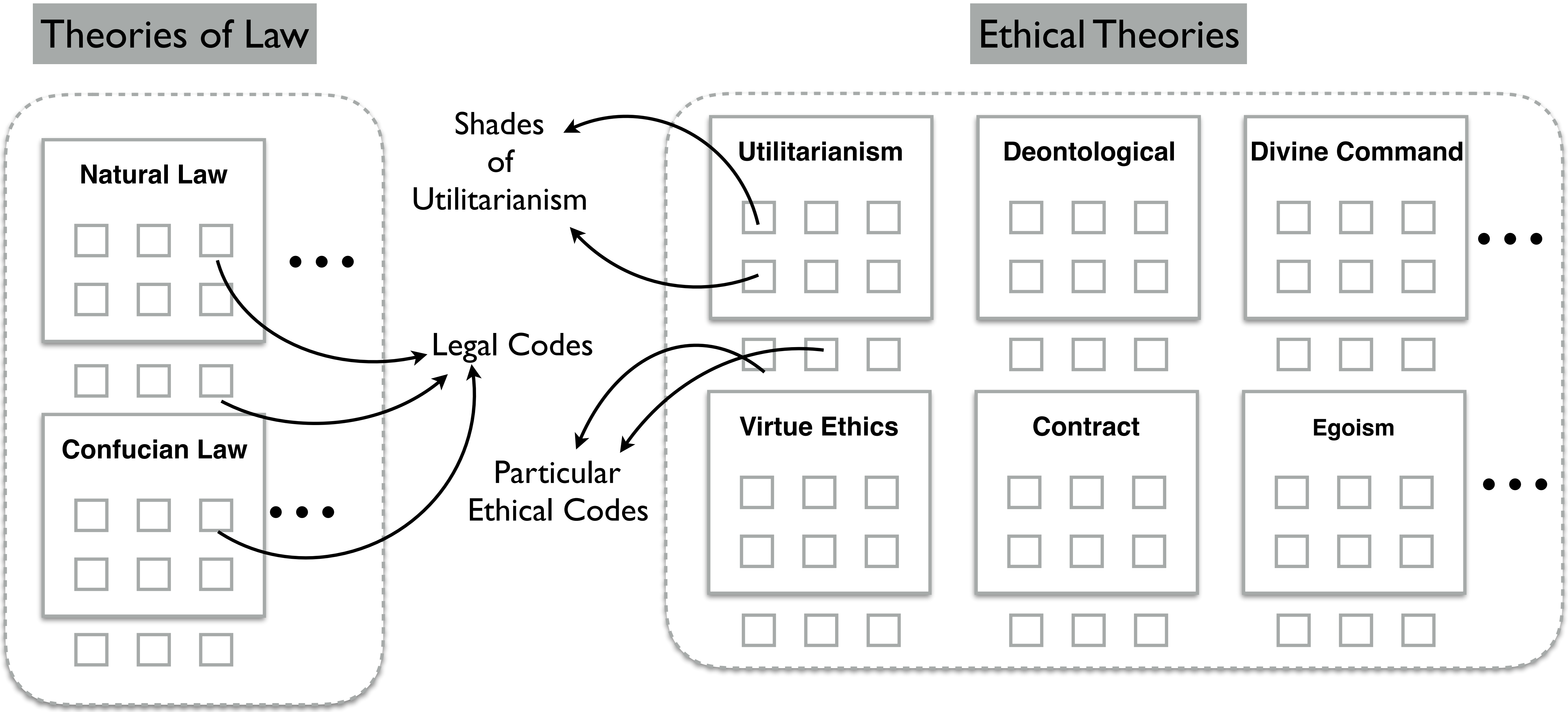
• • •

• • •

Step 1

1. Pick (a) theories(y)
2. Pick (a) code(s)
3. Run through EH.

Making Moral Machines Making Meta Moral Machines

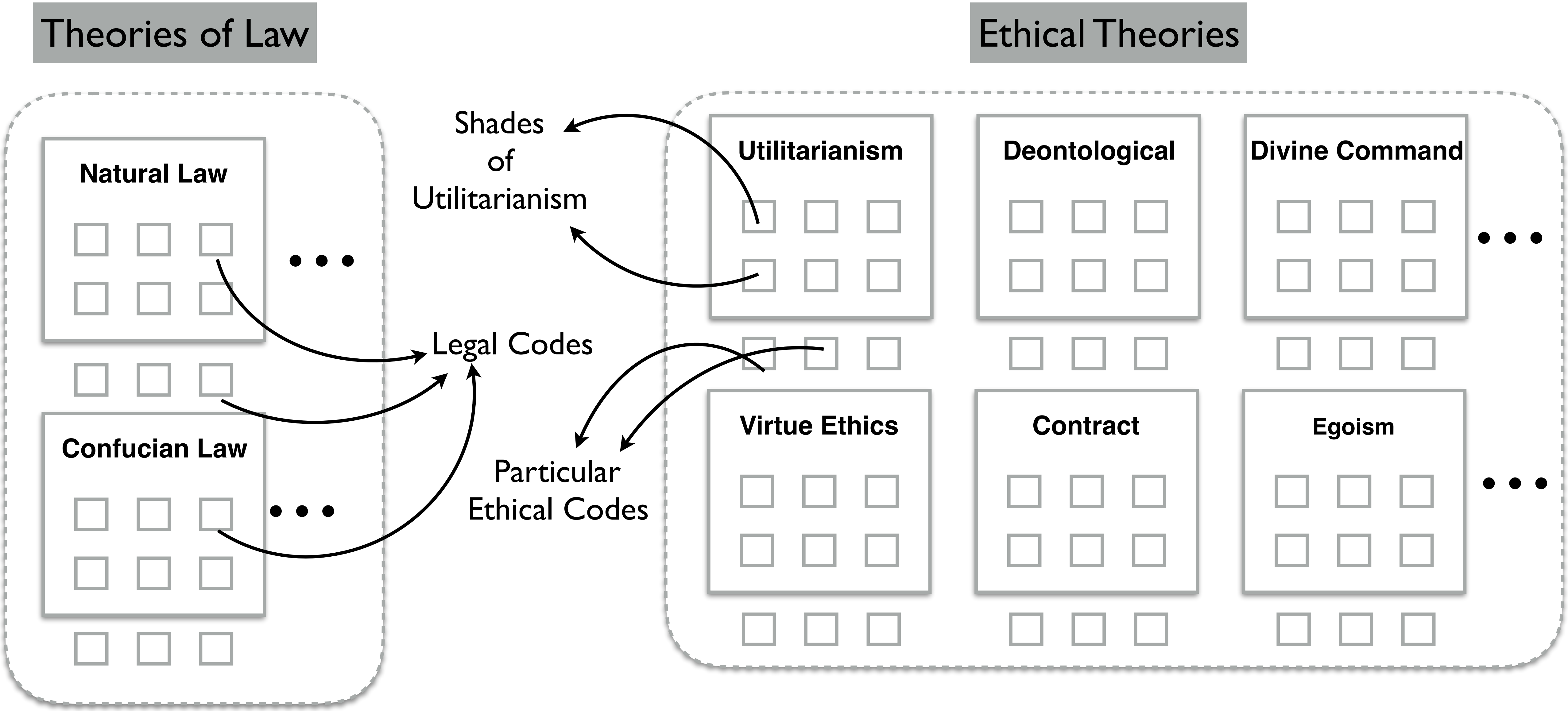


Step 1

1. Pick (a) theories(y)
2. Pick (a) code(s)
3. Run through EH.

Making Moral Machines

Making Meta Moral Machines



Theories of Law

Natural Law

Confucian Law

Legal Codes

Shades
of
Utilitarianism

Particular
Ethical Codes

Ethical Theories

Utilitarianism

Deontological

Divine Command

Virtue Ethics

Contract

Egoism

Step 1

1. Pick (a) theories(y)
2. Pick (a) code(s)
3. Run through EH.

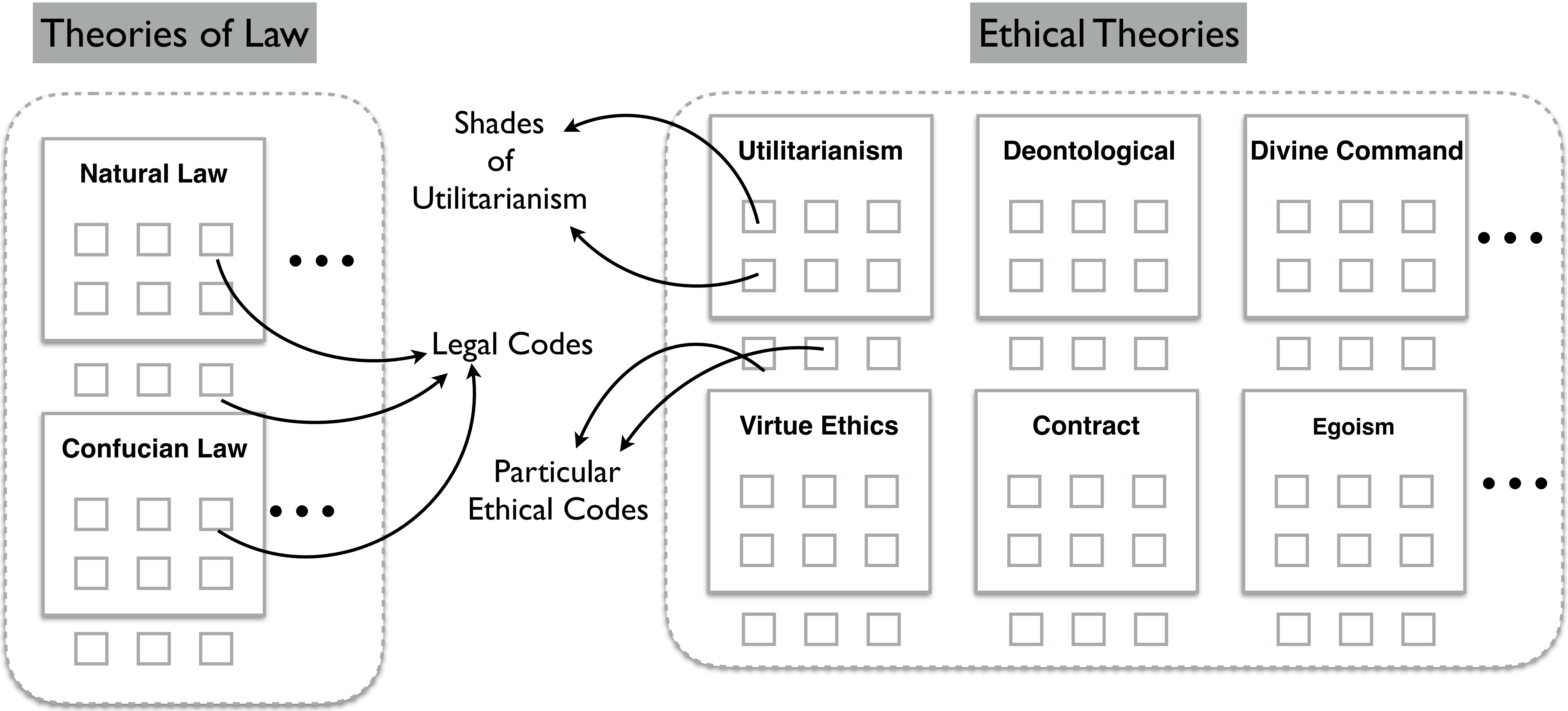
Step 2

Automate

Prover

Spectra

Making Moral Machines Making Meta Moral Machines



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Legal Codes

Shades of Utilitarianism

Particular Ethical Codes

Utilitarianism

Deontological

Divine Command

Virtue Ethics

Contract

Egoism

Step 1

1. Pick (a) theories(y)
2. Pick (a) code(s)
3. Run through EH.

Step 2

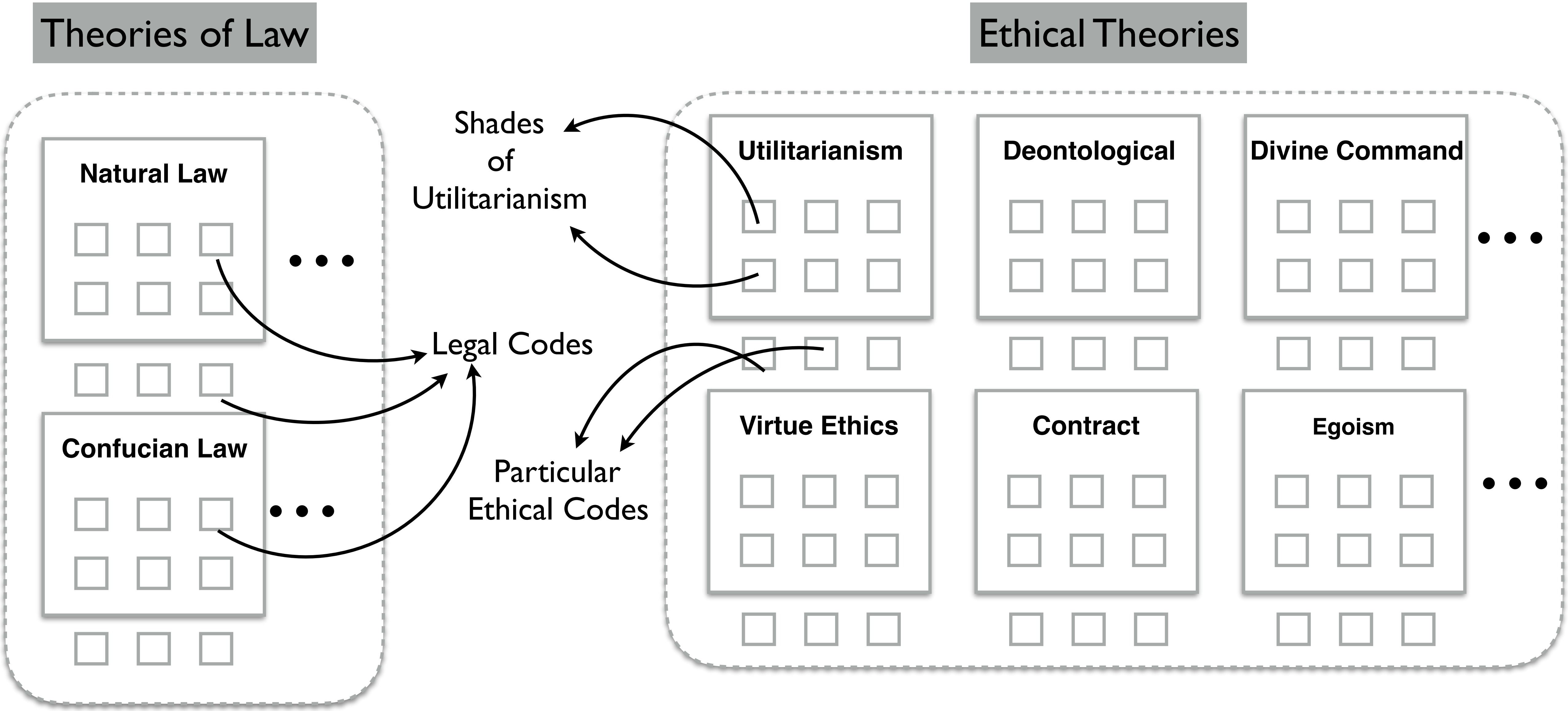
Automate

Prover

Spectra

Making Moral Machines

Making Meta Moral Machines



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Utilitarianism

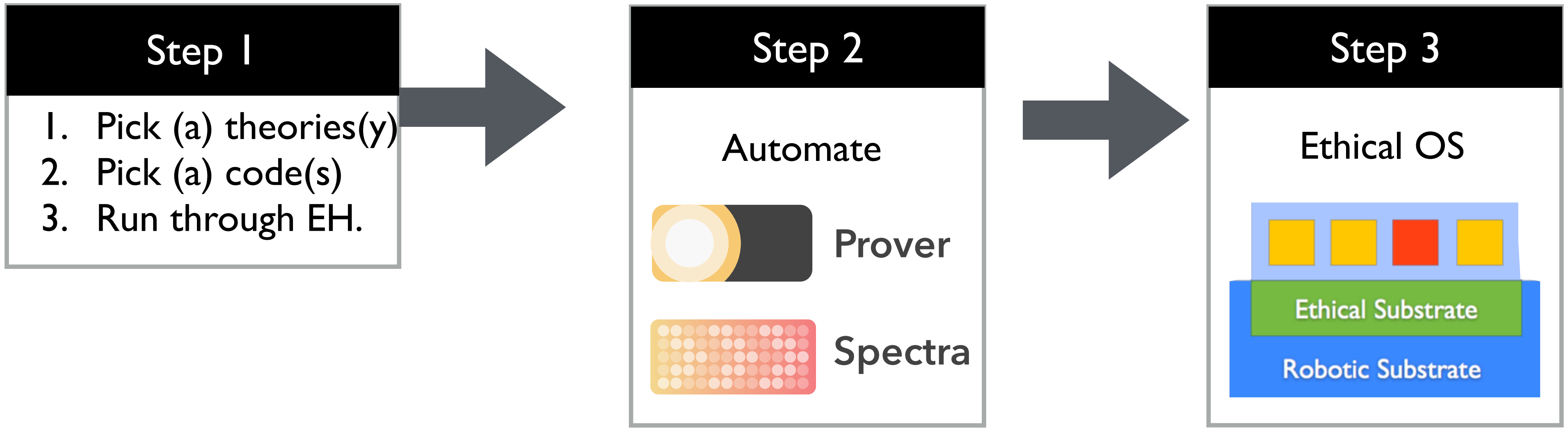
Deontological

Divine Command

Virtue Ethics

Contract

Egoism



Step 1

1. Pick (a) theories(y)
2. Pick (a) code(s)
3. Run through EH.

Step 2

Automate

Prover

Spectra

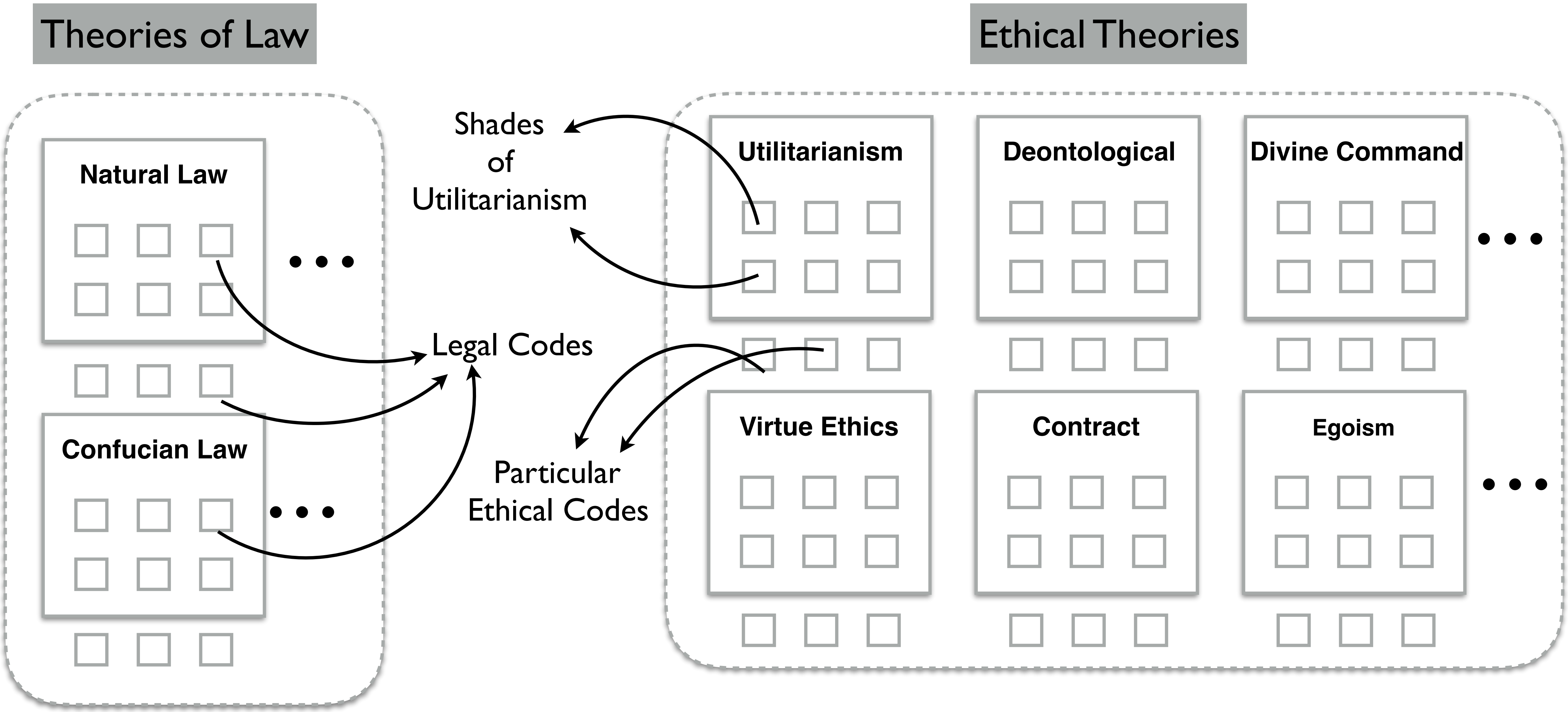
Step 3

Ethical OS

Ethical Substrate

Robotic Substrate

Making Moral Machines Making Meta Moral Machines



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Legal Codes

Shades of Utilitarianism

Particular Ethical Codes

Utilitarianism

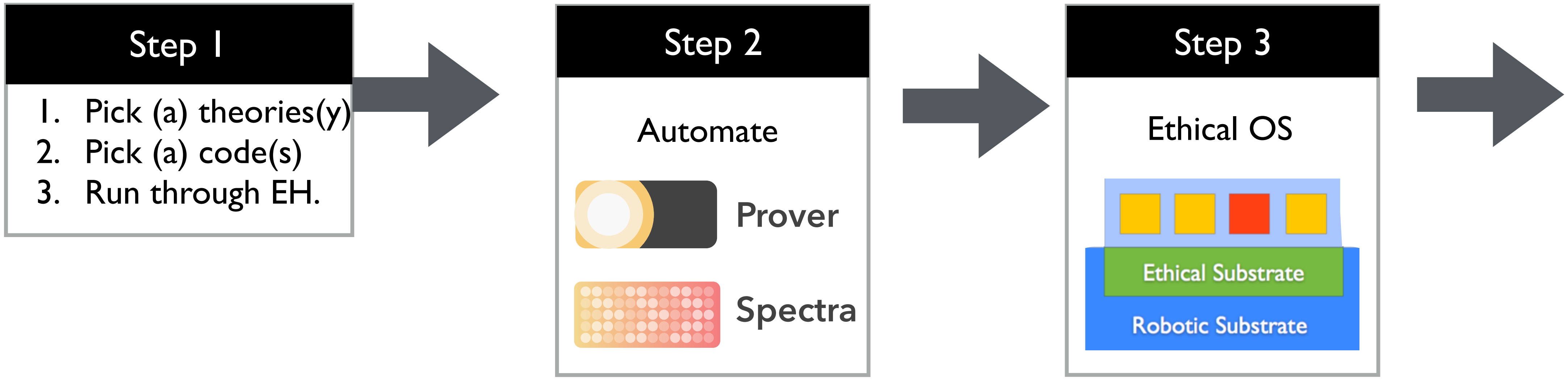
Deontological

Divine Command

Virtue Ethics

Contract

Egoism



Step 1

1. Pick (a) theories(y)
2. Pick (a) code(s)
3. Run through EH.

Step 2

Automate

Prover

Spectra

Step 3

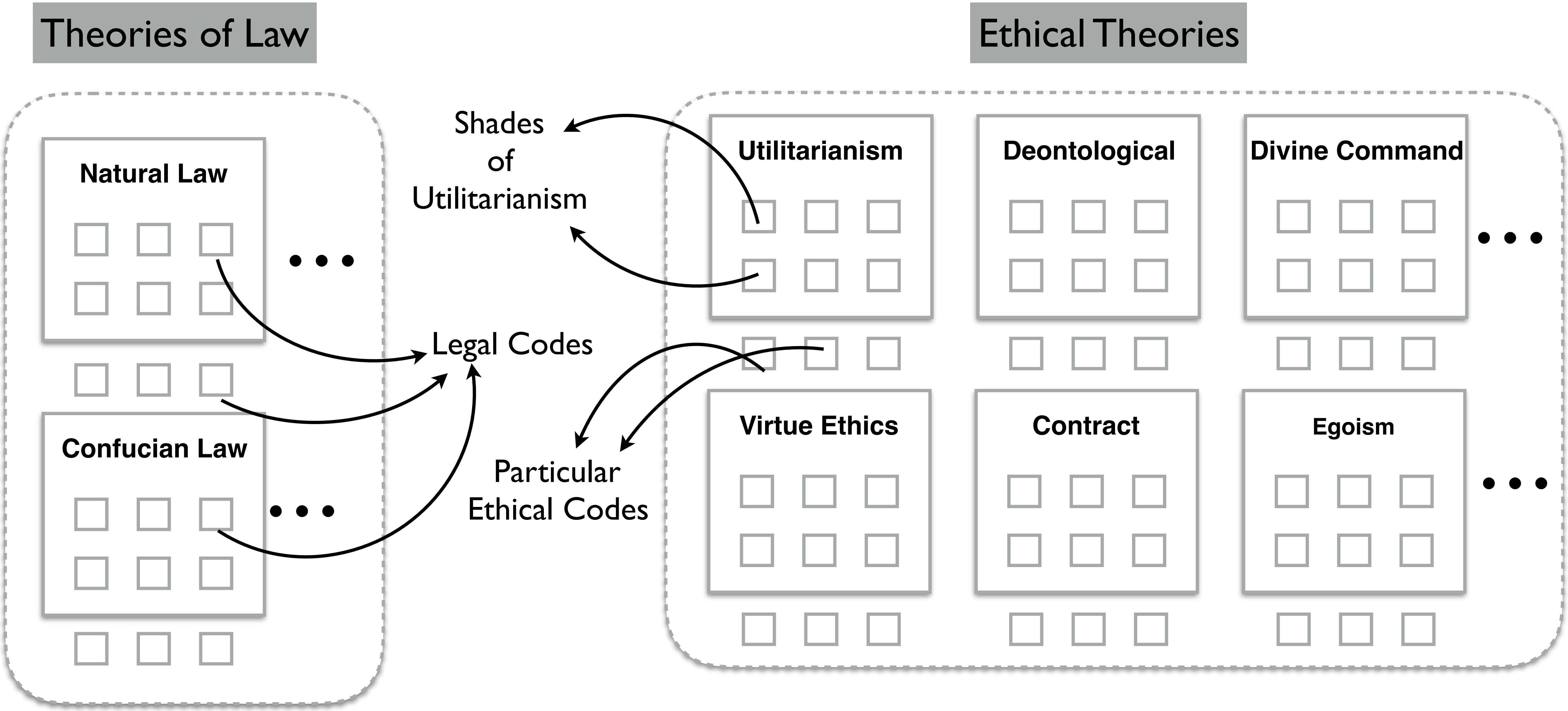
Ethical OS

Ethical Substrate

Robotic Substrate

Making Moral Machines

Making Meta Moral Machines



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Utilitarianism

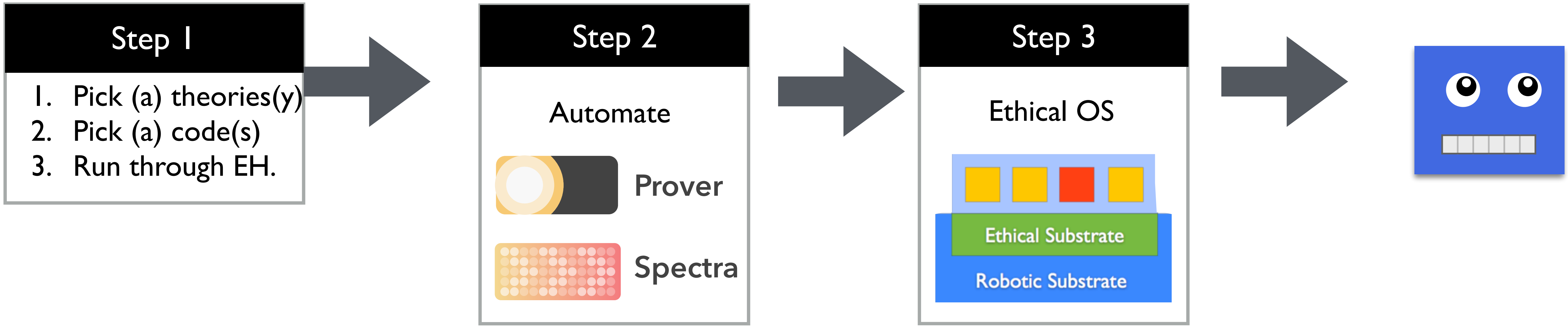
Deontological

Divine Command

Virtue Ethics

Contract

Egoism



Step 1

1. Pick (a) theories(y)
2. Pick (a) code(s)
3. Run through EH.

Step 2

Automate

Prover

Spectra

Step 3

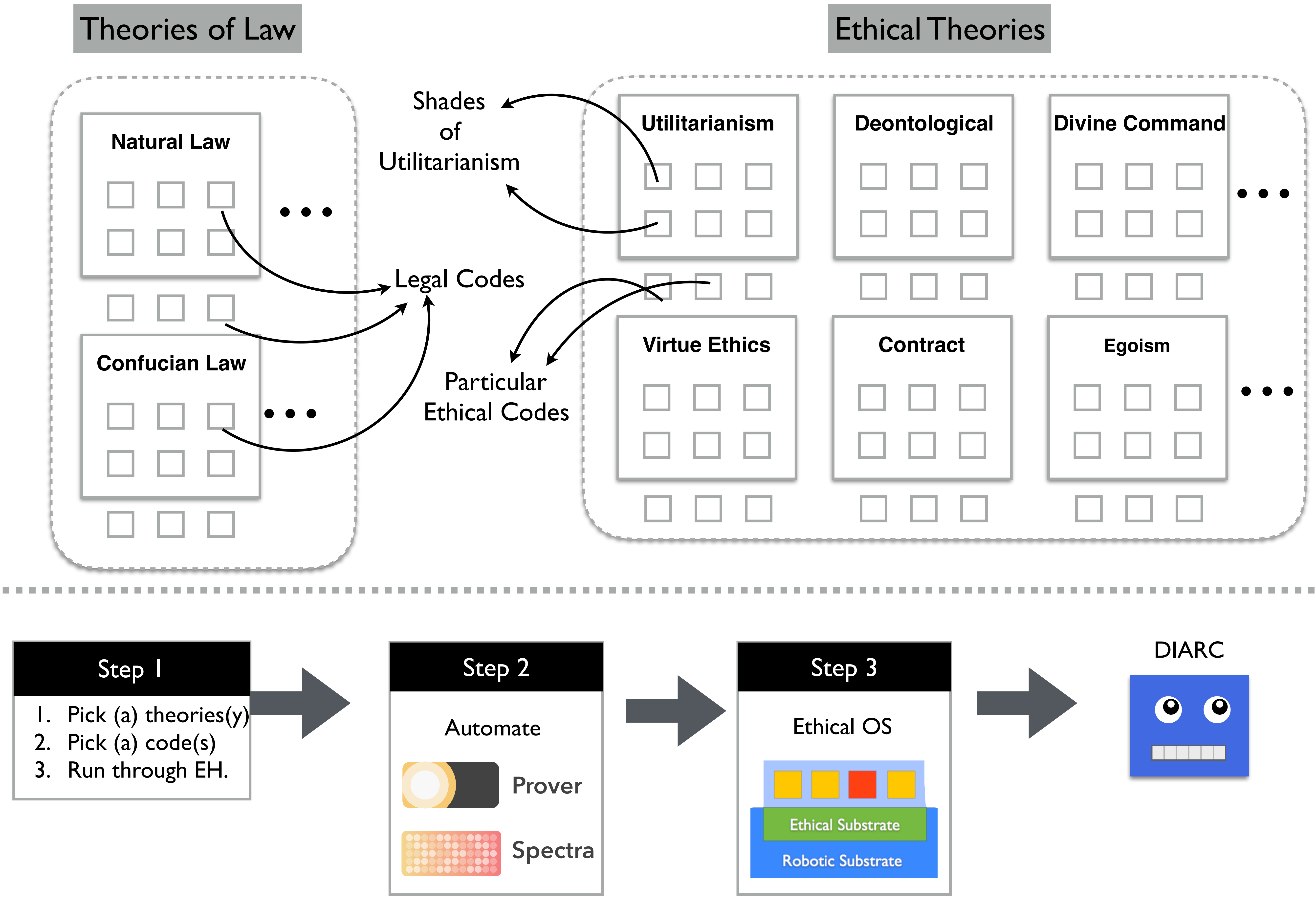
Ethical OS

Ethical Substrate

Robotic Substrate

Making Moral Machines

Making Meta Moral Machines



Theories of Law

Ethical Theories

Natural Law

Confucian Law

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Utilitarianism

Deontological

Divine Command

Virtue Ethics

Contract

Egoism

Step 1

1. Pick (a) theories(y)
2. Pick (a) code(s)
3. Run through EH.

Step 2

Automate

Prover

Spectra

Step 3

Ethical OS

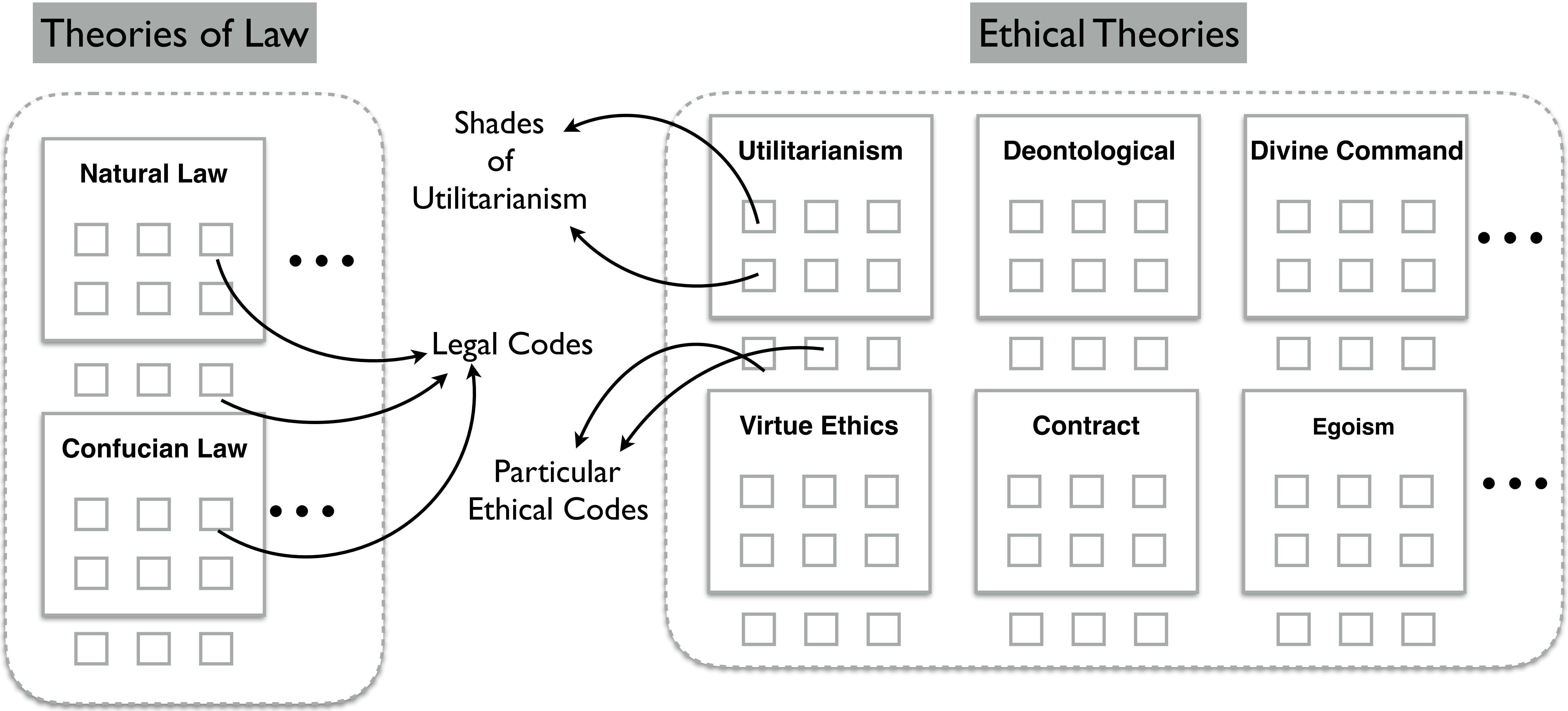
Ethical Substrate

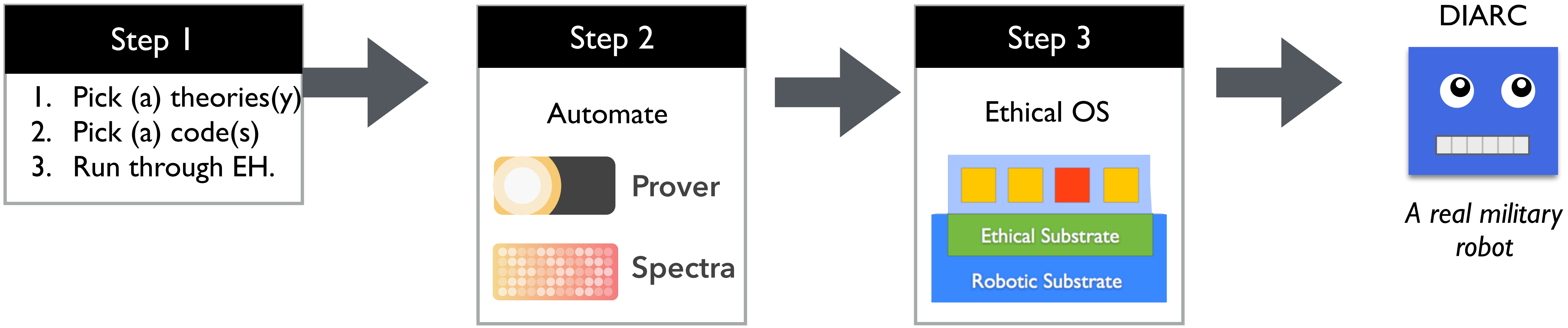
Robotic Substrate

DIARC

Making Moral Machines

Making Meta Moral Machines





1. Pick (a) theories(y)
2. Pick (a) code(s)
3. Run through EH.

A real military robot

IV.
Key Core AI Technologies for Cognitive Calculi …

ShadowProver

Prover

Motivation

- We have decades of research and industrial-strength implementations of propositional and first-order theorem provers.

- Utilize this in building first-order intensional-logic provers and above, in a principled manner.

Two Extant Modes

- There are two ways of piggy backing on first-order provers to build higher-order provers …

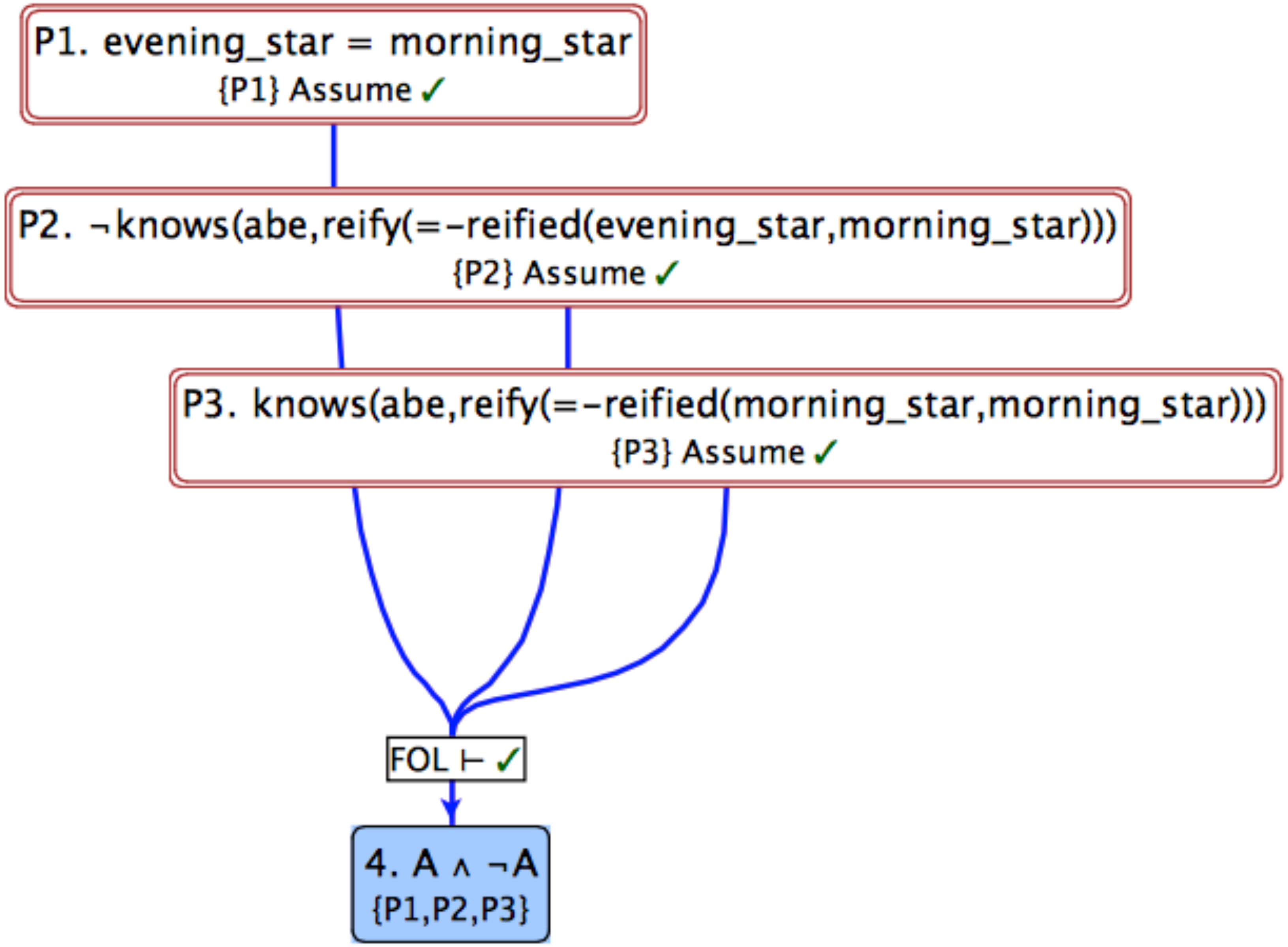
Two Extant Modes

Mode 1: Honest Encoding	
Method	Painstakingly encode all rules of inference and syntax in FOL
Pros	Precise
Cons	Extremely slow to implement Reasoning is also slow

Two Extant Modes

Mode 2: Naïve Encoding	
Method	Pretend intensional and higher-order formulae and operators are first-order predicates
Pros	Extremely easy to implement Reasoning can also be fast
Cons	Unsound Wrong inferences can be easily drawn

Mode 2



P1. evening_star = morning_star

{P1} Assume ✓

P2. ¬knows(abe,reify(=-reified(evening_star,morning_star)))

{P2} Assume ✓

P3. knows(abe,reify(=-reified(morning_star,morning_star)))

{P3} Assume ✓

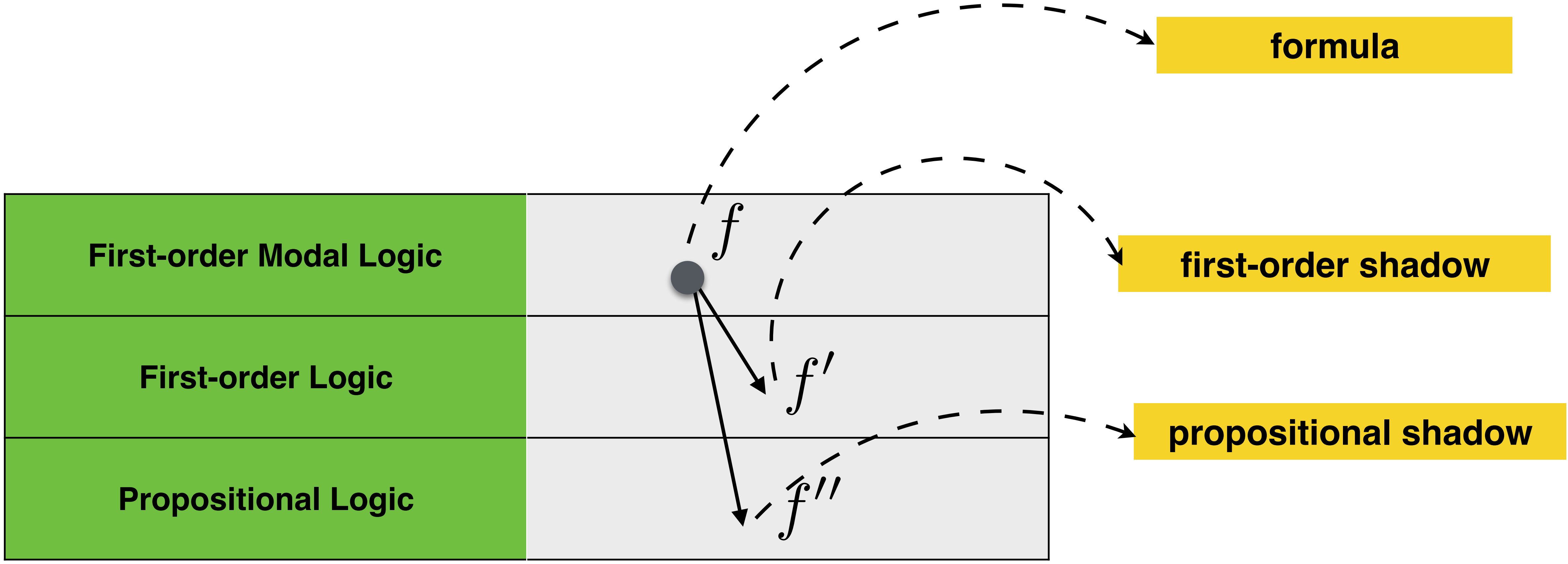
FOL ⊢ ✓

4. A ∧ ¬A

{P1,P2,P3}

A New Way: ShadowProver

Every formula at level **t** has a unique formula called its **"shadow"** in each level **t' < t**



$S_{[f]}$ The Shadow Maker

For all formulae **f**,

$S_{[\mathbf{f}]}$ is a unique atomic symbol.

Examples of shadows

$$(\forall x \mathbf{B}(a, Q)) \wedge P(x)$$

formula

$$\forall x S_{[\mathbf{B}(a,Q)]} \wedge P(x)$$

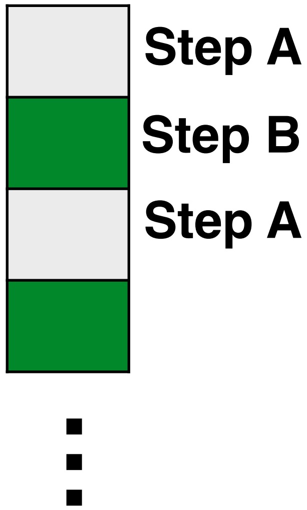
first-order shadow

$$S_{[\forall x \mathbf{B}(a,Q)]} \wedge P(x)$$

propositional shadow

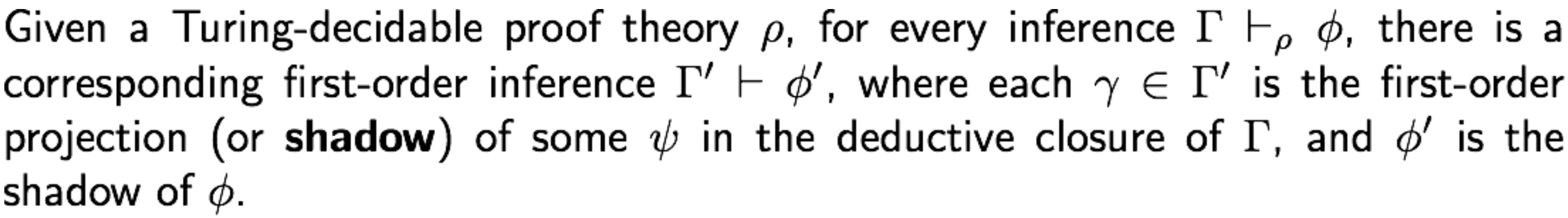
A New Way: Shadow Prover

- Two step process till goal is reached:
  - **Step A**: Shadow formulae down to all lower levels. Run lower theorem provers. If goal reached, return **true**.
  - **Step B**: Expand the assumption base using higher level rules.



Actually, this is more general:

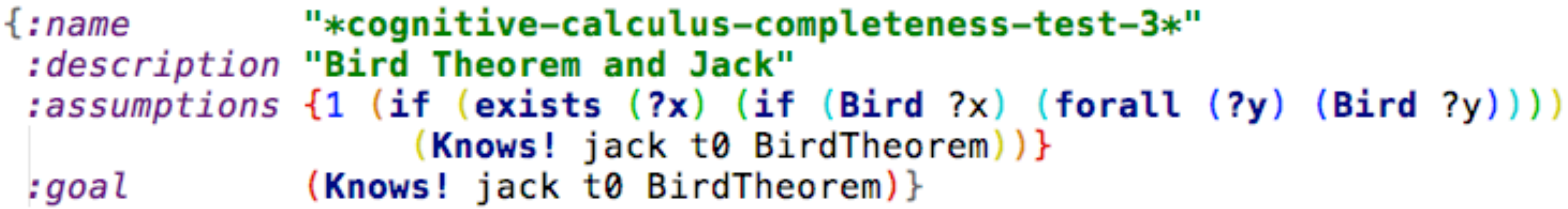
Theorem:



Given a Turing-decidable proof theory ρ, for every inference $\Gamma \vdash_\rho \phi$, there is a corresponding first-order inference $\Gamma' \vdash \phi'$, where each $\gamma \in \Gamma'$ is the first-order projection (or **shadow**) of some ψ in the deductive closure of Γ, and ϕ' is the shadow of ϕ.

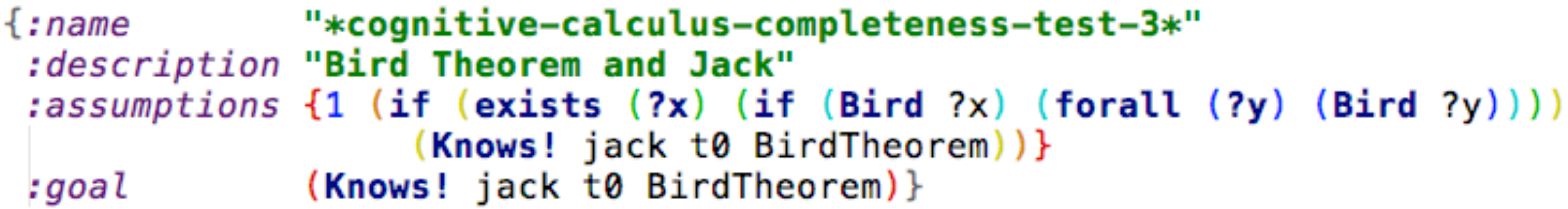
Rather Promising Results

Rather Promising Results



```
{:name         "*cognitive-calculus-completeness-test-3*"
 :description "Bird Theorem and Jack"
 :assumptions {1 (if (exists (?x) (if (Bird ?x) (forall (?y) (Bird ?y))))
                     (Knows! jack t0 BirdTheorem))}
 :goal        (Knows! jack t0 BirdTheorem)}
```

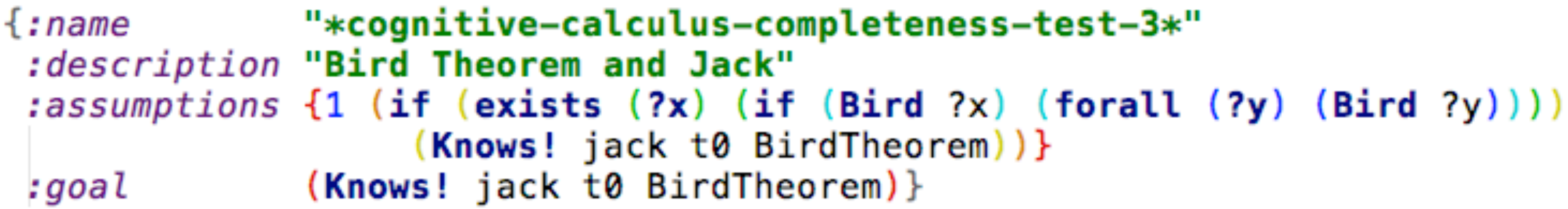
Rather Promising Results



```
{:name        "*cognitive-calculus-completeness-test-3*"
 :description "Bird Theorem and Jack"
 :assumptions {1 (if (exists (?x) (if (Bird ?x) (forall (?y) (Bird ?y))))
                     (Knows! jack t0 BirdTheorem))}
 :goal        (Knows! jack t0 BirdTheorem)}
```

Note: the antecedent is a theorem in first-order logic

Rather Promising Results

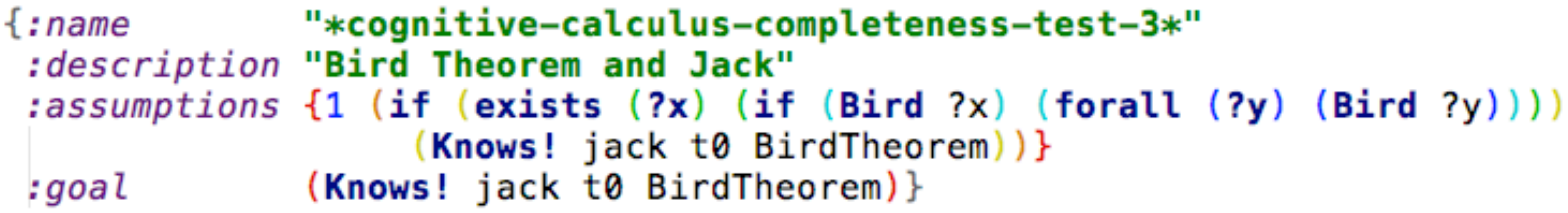


```
{:name        "*cognitive-calculus-completeness-test-3*"
 :description "Bird Theorem and Jack"
 :assumptions {1 (if (exists (?x) (if (Bird ?x) (forall (?y) (Bird ?y))))
                   (Knows! jack t0 BirdTheorem))}
 :goal        (Knows! jack t0 BirdTheorem)}
```

Note: the antecedent is a theorem in first-order logic

2 ms!

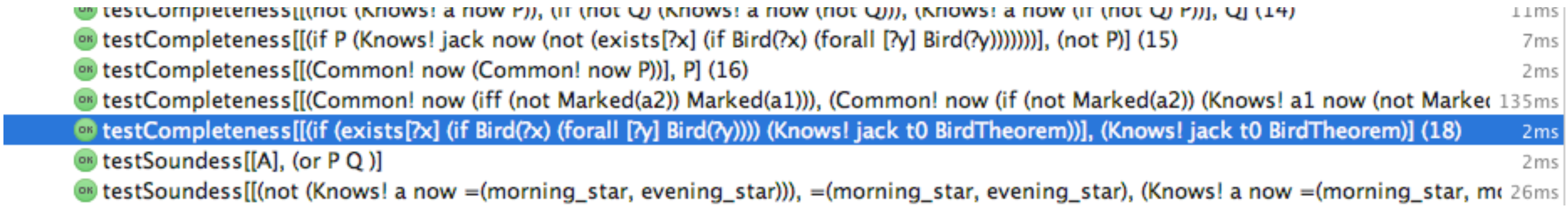
Rather Promising Results



```
{:name         "*cognitive-calculus-completeness-test-3*"
 :description "Bird Theorem and Jack"
 :assumptions {1 (if (exists (?x) (if (Bird ?x) (forall (?y) (Bird ?y))))
                      (Knows! jack t0 BirdTheorem))}
 :goal        (Knows! jack t0 BirdTheorem)}
```

Note: the antecedent is a theorem in first-order logic

2 ms!



```
testCompleteness[[(not (Knows! a now P)), (if (not Q) (Knows! a now (not Q))), (Knows! a now (if (not Q) P))], Q] (14)   11ms
testCompleteness[[(if P (Knows! jack now (not (exists[?x] (if Bird(?x) (forall [?y] Bird(?y)))))))], (not P)] (15)   7ms
testCompleteness[[(Common! now (Common! now P))], P] (16)   2ms
testCompleteness[[(Common! now (iff (not Marked(a2)) Marked(a1))), (Common! now (if (not Marked(a2)) (Knows! a1 now (not Marke   135ms
testCompleteness[[(if (exists[?x] (if Bird(?x) (forall [?y] Bird(?y)))) (Knows! jack t0 BirdTheorem))], (Knows! jack t0 BirdTheorem)] (18)   2ms
testSoundess[[A], (or P Q )]   2ms
testSoundess[[(not (Knows! a now =(morning_star, evening_star))), =(morning_star, evening_star), (Knows! a now =(morning_star, m   26ms
```

A Particularly Promising (& Selmer-disturbing) Result:

- Automation of false-belief task and other projects that were only semi-automated before.
- More at:
  - **Java Implementation:**
    - https://bitbucket.org/Holmes/prover/

Future Work

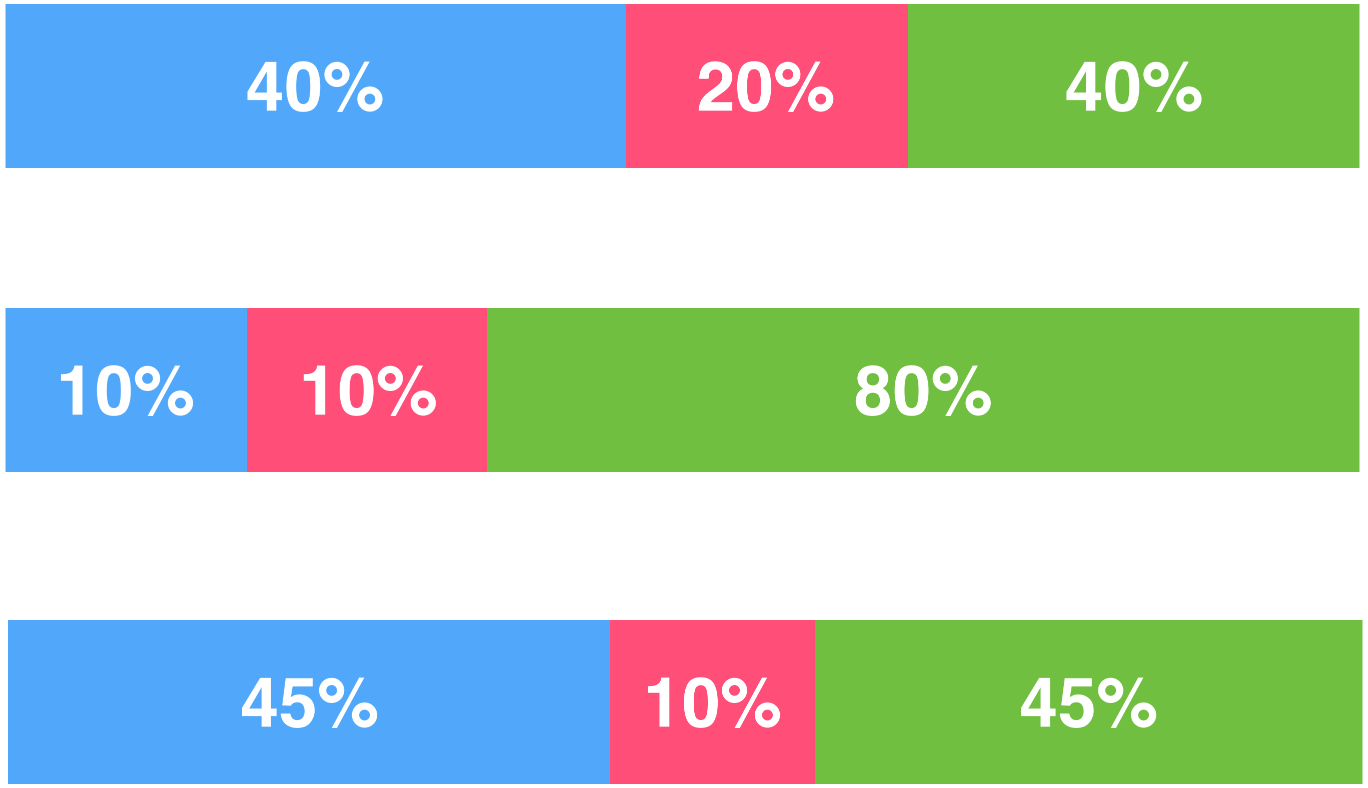
Future work is a mix of research, design, and implementation

🟦 **research** 🟥 **design** 🟩 **implementation**

1 Custom language for **extending** to other first-order modal calculi

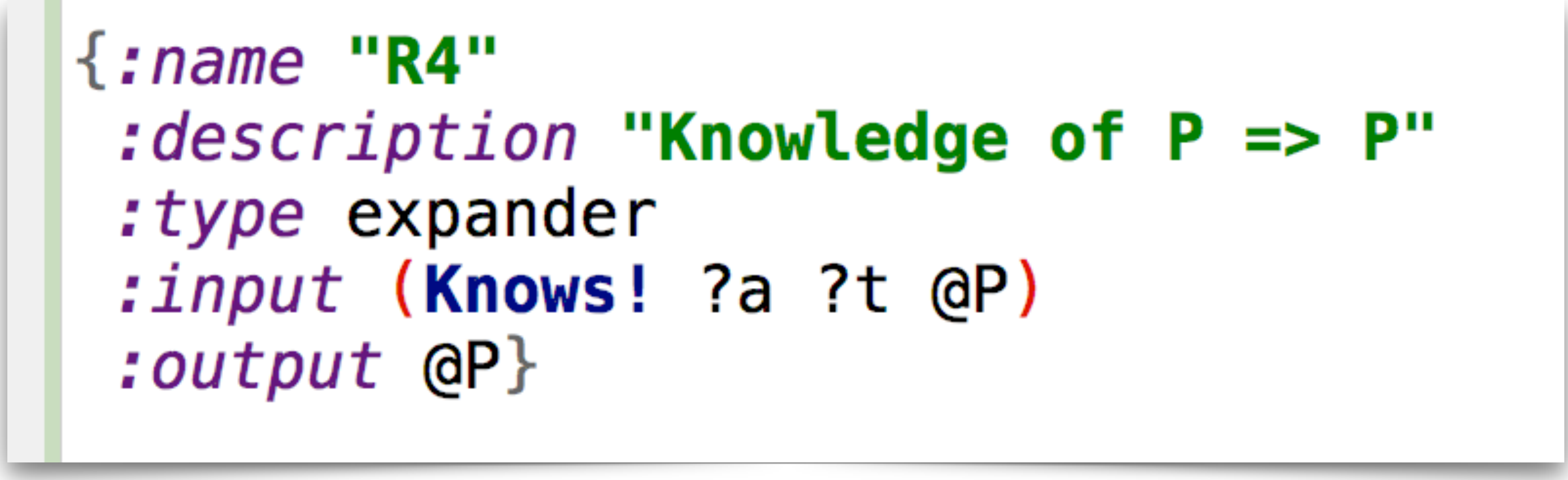
2 **Further integration** with robotic platforms at Tufts and RPI

3 Explore **parallelization** and other venues for even more speedup



Custom Language and Logic

- Allow users to specify new inference schemata. E.g.



```
{:name "R4"
 :description "Knowledge of P => P"
 :type expander
 :input (Knows! ?a ?t @P)
 :output @P}
```

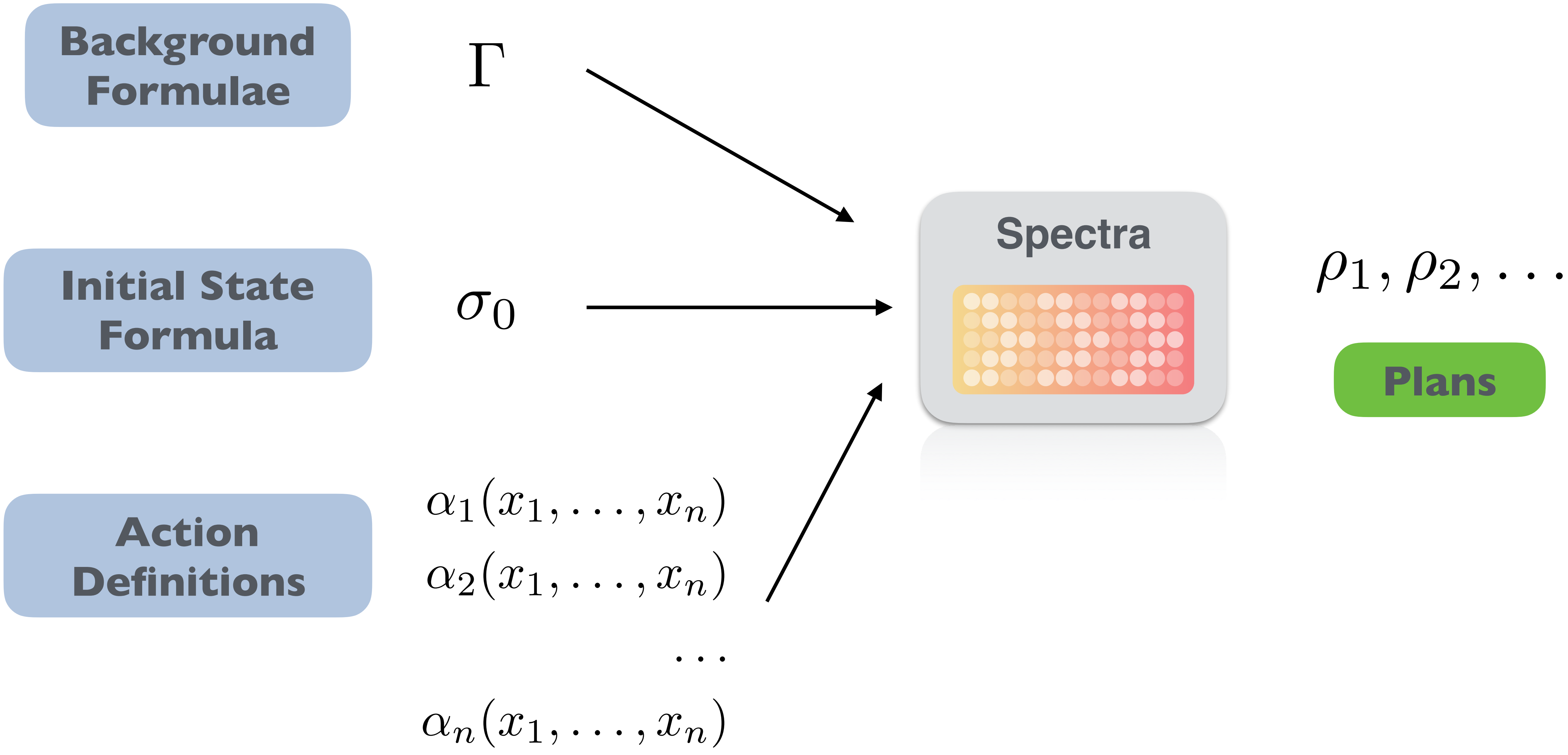
Spectra

https://bitbucket.org/Holmes/planner

Spectra

- Existing Planners: **Propositional** (essentially)
- Drawbacks:
  - **Expressivity**: Cannot express arbitrary constraints.
    - “At every step make sure that no two blocks on the table have same color.”
  - **Domain Size**: Scaling to large domains of arbitrary sizes poses difficulty.

Spectra (planner)



Background Formulae

Γ

Initial State Formula

σ_0

Action Definitions

$\alpha_1(x_1, \ldots, x_n)$

$\alpha_2(x_1, \ldots, x_n)$

$\ldots$

$\alpha_n(x_1, \ldots, x_n)$

Spectra

$\rho_1, \rho_2, \ldots$

Plans

Infinite Models

$$\forall x \exists y \mathbf{R}\left(x, y\right) \wedge$$

$$\forall x, y \neg \left(\mathbf{R}\left(x, y\right) \wedge \mathbf{R}\left(y, x\right)\right) \wedge$$

$$\forall x, y, z \left(\mathbf{R}\left(x, y\right) \wedge \mathbf{R}\left(y, z\right)\right) \rightarrow \mathbf{R}\left(x, z\right)$$

Infinite Models

$$\forall x \exists y \mathbf{R}\left(x, y\right) \wedge$$

$$\forall x, y \neg \left(\mathbf{R}\left(x, y\right) \wedge \mathbf{R}\left(y, x\right)\right) \wedge$$

$$\forall x, y, z \left(\mathbf{R}\left(x, y\right) \wedge \mathbf{R}\left(y, z\right)\right) \rightarrow \mathbf{R}\left(x, z\right)$$

Has only infinite models

Infinite Models

$$\forall x \exists y \mathbf{R}(x, y) \wedge$$
$$\forall x, y \neg (\mathbf{R}(x, y) \wedge \mathbf{R}(y, x)) \wedge$$
$$\forall x, y, z (\mathbf{R}(x, y) \wedge \mathbf{R}(y, z)) \rightarrow \mathbf{R}(x, z)$$

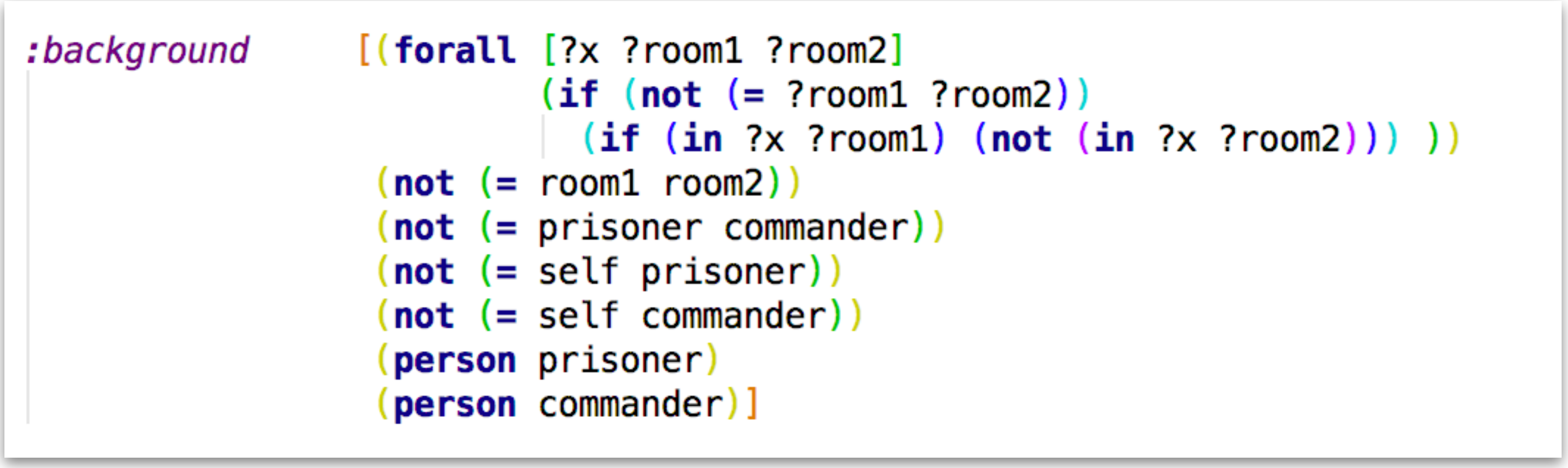
Has only infinite models

Useful for modeling agents that work with:

1. an unbounded number of objects, agents;
2. abstract objects

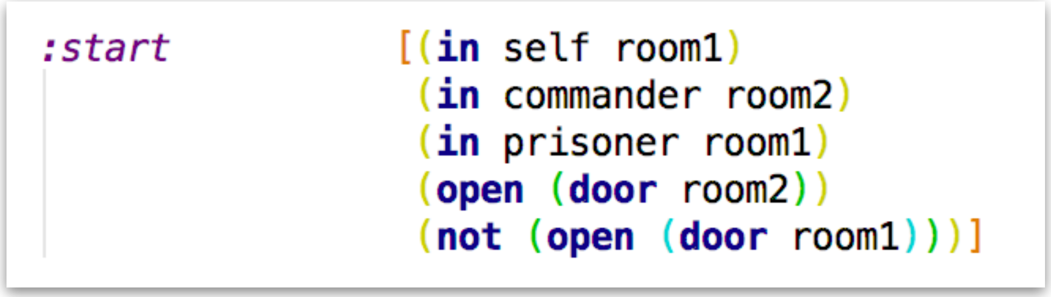
Example

Background Formulae



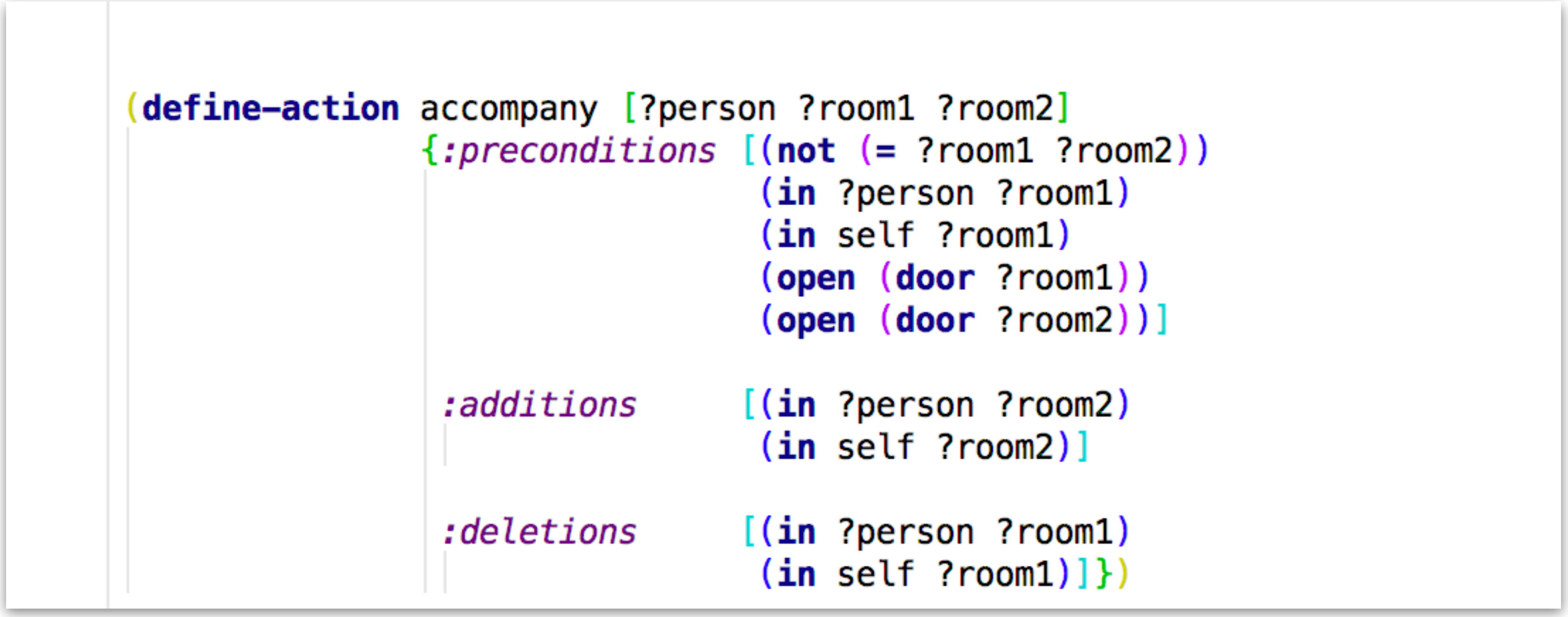
```
:background      [(forall [?x ?room1 ?room2]
                          (if (not (= ?room1 ?room2))
                            (if (in ?x ?room1) (not (in ?x ?room2))) ))
                  (not (= room1 room2))
                  (not (= prisoner commander))
                  (not (= self prisoner))
                  (not (= self commander))
                  (person prisoner)
                  (person commander)]
```

Initial State Formula



```
:start                [(in self room1)
                       (in commander room2)
                       (in prisoner room1)
                       (open (door room2))
                       (not (open (door room1)))]
```

Action Definitions



```
(define-action accompany [?person ?room1 ?room2]
               {:preconditions [(not (= ?room1 ?room2))
                                (in ?person ?room1)
                                (in self ?room1)
                                (open (door ?room1))
                                (open (door ?room2))]

                :additions     [(in ?person ?room2)
                                (in self ?room2)]

                :deletions     [(in ?person ?room1)
                                (in self ?room1)]})
```

How do you handle efficiency?

- Two approaches:
  - **Procedural Attachments**: Special purpose procedural code that can bypass strict formal reasoning.
  - **µ-methods**: Written in denotational proof language. Preserves soundness by letting us write down commonly used patterns of reasoning (a bit unwieldy integration now than the first approach).

Third-person *de dicto*

$$\mathbf{B}(\mathit{cogito}, \exists x : \mathsf{Agent}(\mathit{named}(x, \text{“}\mathit{Cogito}\text{”}) \land \mathit{red\text{-}splotched}(x)))$$

Third-person *de dicto*

$$\mathbf{B}(\mathit{cogito}, \exists x : \mathsf{Agent}(\mathit{named}(x, \text{“}\mathit{Cogito}\text{”}) \wedge \mathit{red\text{-}splotched}(x)))$$

Third-person *de re*

$$\exists x : \mathsf{Agent}(\mathit{named}(x, \text{“}\mathit{Cogito}\text{”}) \wedge \mathbf{B}(\mathit{cogito}, \mathit{red\text{-}splotched}(x)))$$

Third-person *de dicto*

$$\mathbf{B}(\mathit{cogito}, \exists x : \mathsf{Agent}(\mathit{named}(x, \text{“}\mathit{Cogito}\text{”}) \wedge \mathit{red\text{-}splotched}(x)))$$

Third-person *de re*

$$\exists x : \mathsf{Agent}(\mathit{named}(x, \text{“}\mathit{Cogito}\text{”}) \wedge \mathbf{B}(\mathit{cogito}, \mathit{red\text{-}splotched}(x)))$$

Third-person *de se*

$$\mathbf{B}(\mathit{cogito}, \mathit{red\text{-}splotched}(\mathit{cogito}*)))$$

Third-person *de dicto*

$$\mathbf{B}(cogito, \exists x : \mathsf{Agent}(named(x, \text{“}Cogito\text{”}) \wedge red\text{-}splotched(x)))$$

Third-person *de re*

$$\exists x : \mathsf{Agent}(named(x, \text{“}Cogito\text{”}) \wedge \mathbf{B}(cogito, red\text{-}splotched(x)))$$

Third-person *de se*

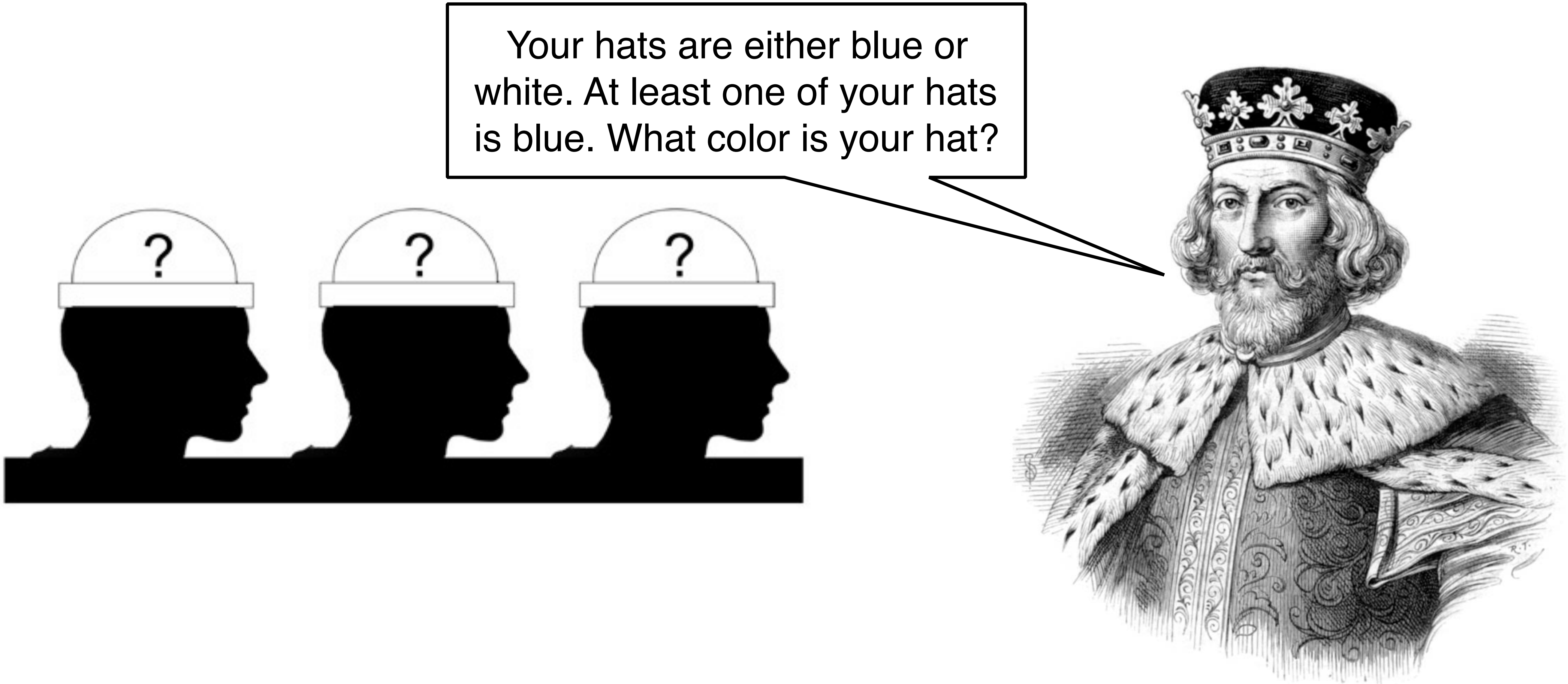
$$\mathbf{B}(cogito, red\text{-}splotched(cogito*)))$$

First-person *de se*

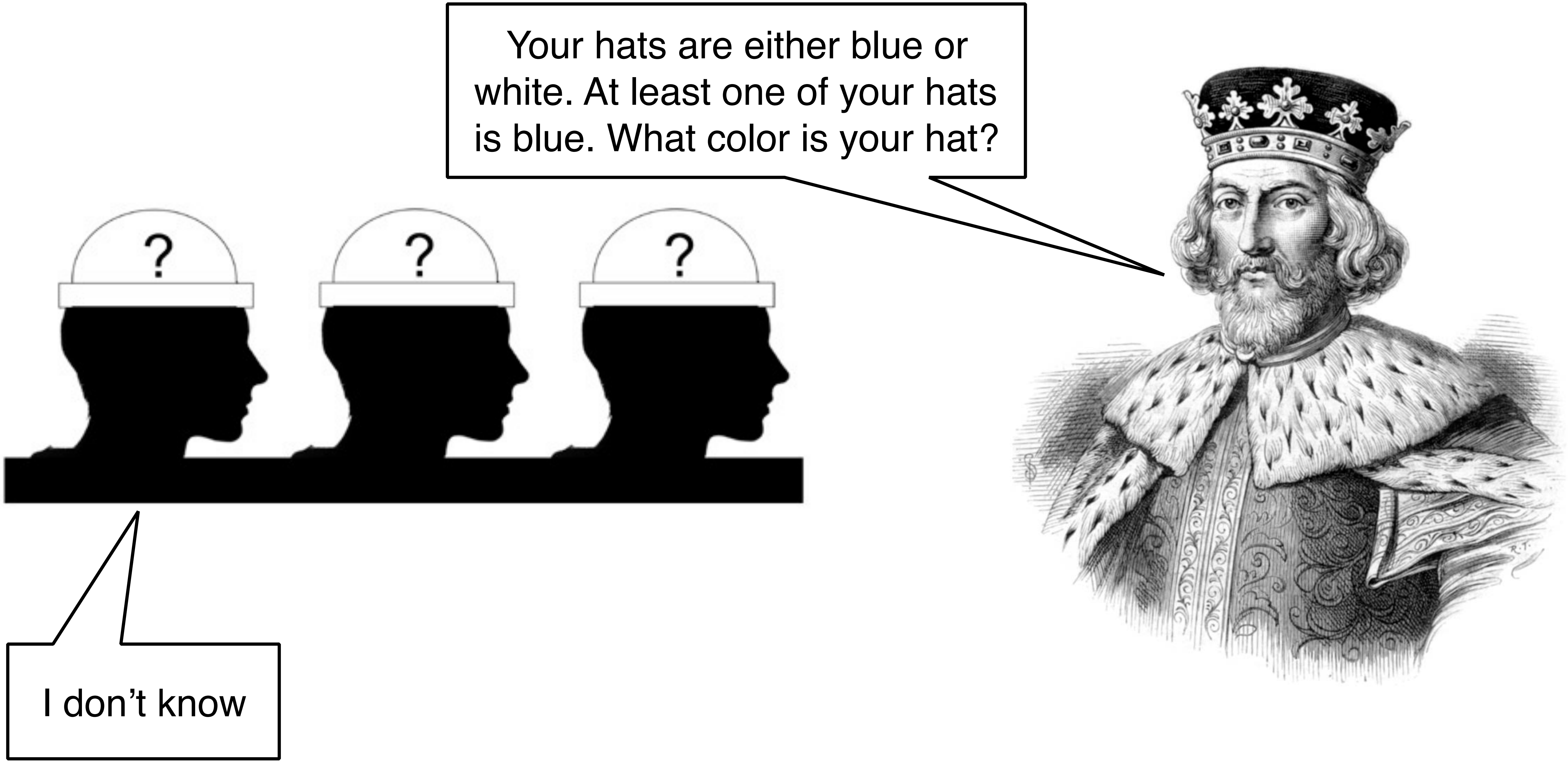
$$\mathbf{B}(I, red\text{-}splotched(I*))$$

in the logic $\mathcal{L}_{cogito}$

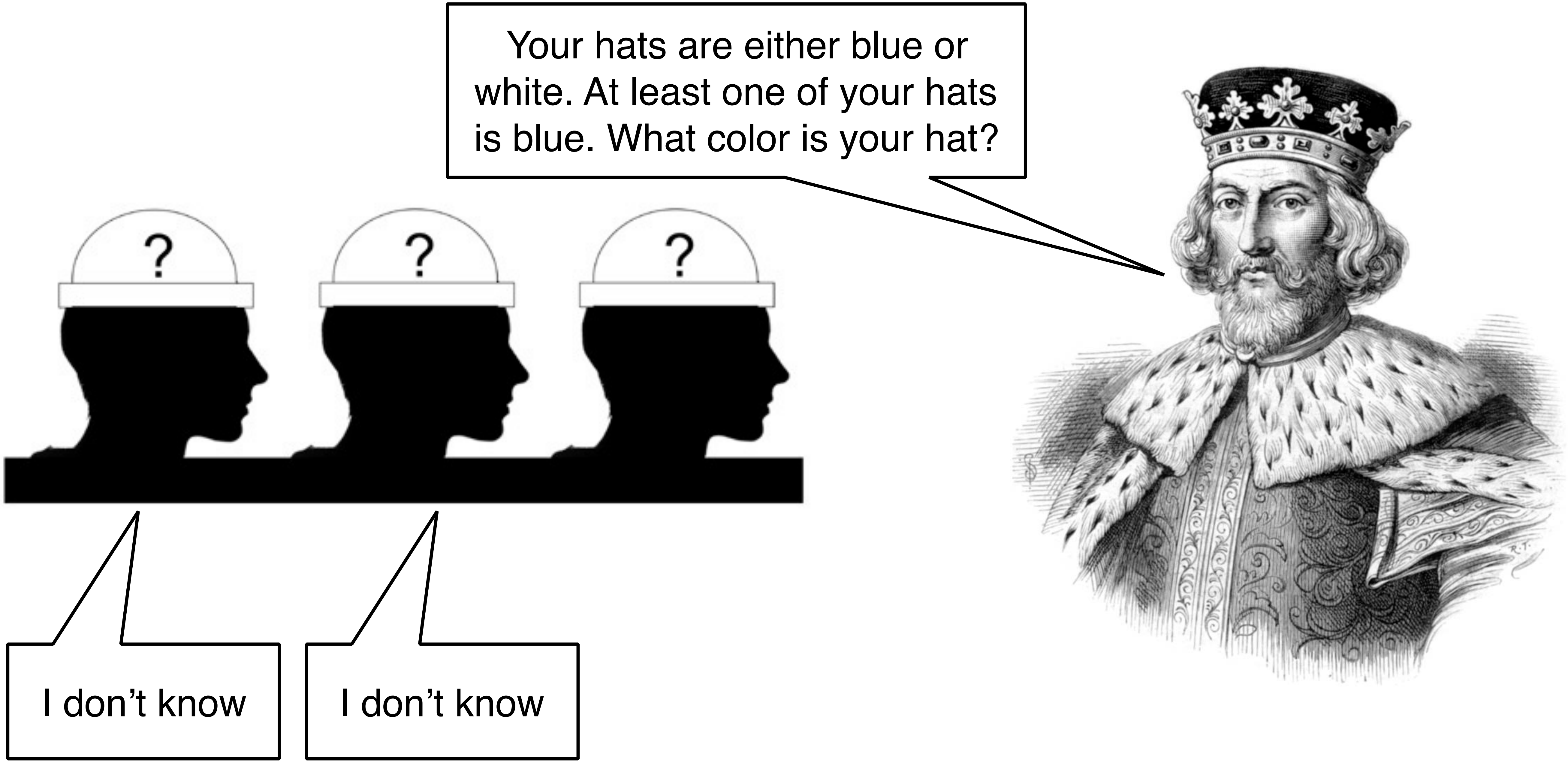
Wise man’s hat puzzle: well-known benchmark for epistemic logics


Your hats are either blue or white. At least one of your hats is blue. What color is your hat?
?
?
?

Wise man's hat puzzle: well-known benchmark for epistemic logics


Your hats are either blue or white. At least one of your hats is blue. What color is your hat?
?
?
?
I don't know

Wise man’s hat puzzle: well-known benchmark for epistemic logics

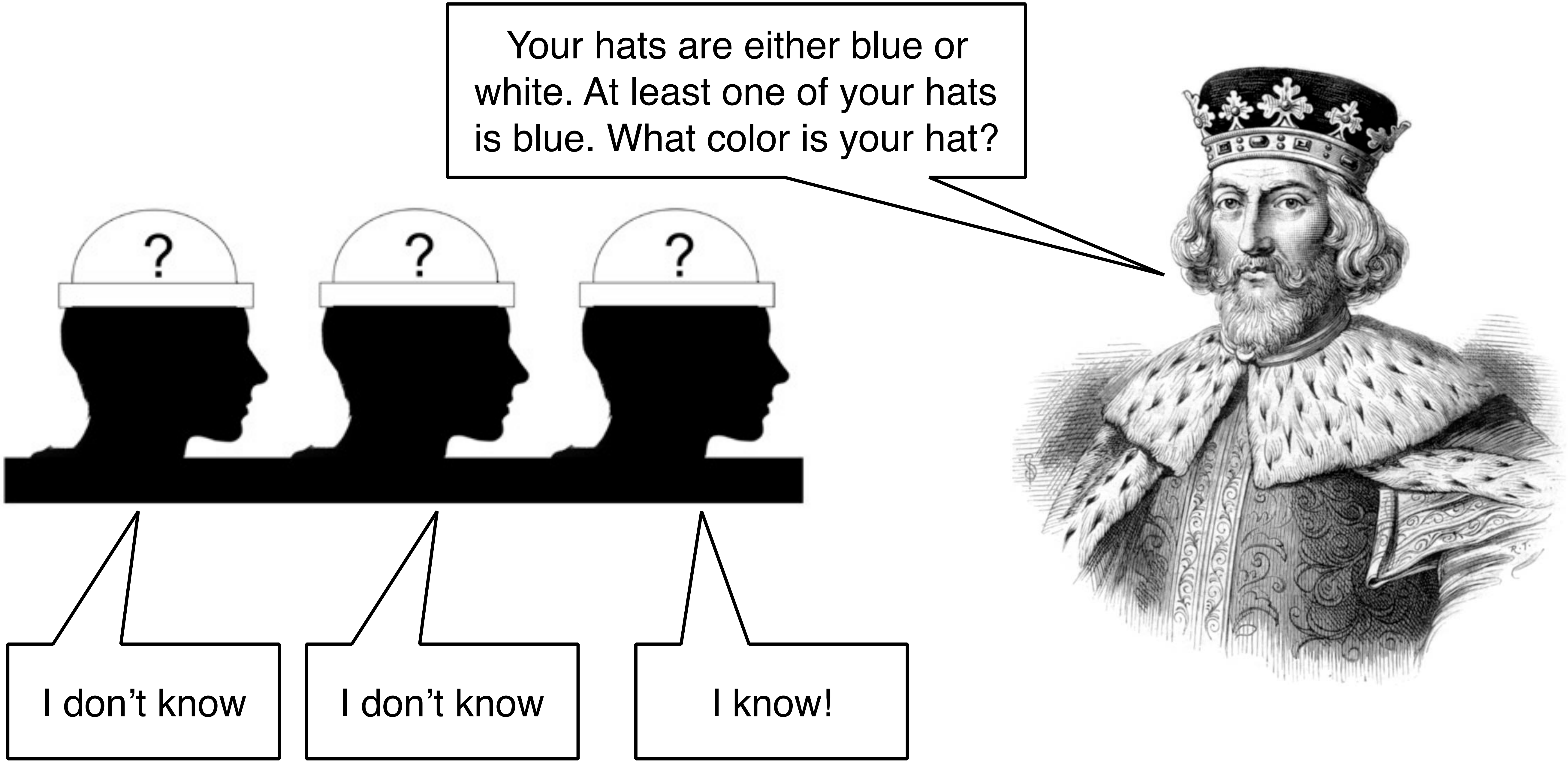


Your hats are either blue or white. At least one of your hats is blue. What color is your hat?

I don’t know

I don’t know

Wise man’s hat puzzle: well-known benchmark for epistemic logics



Your hats are either blue or white. At least one of your hats is blue. What color is your hat?

I don’t know

I don’t know

I know!

Floridi’s KG$_4$

dumbing pill

dumbing pill

placebo

Floridi's KG$_4$

dumbing pill

dumbing pill

placebo

Which pill did you receive?

Floridi’s KG$_4$

dumbing pill

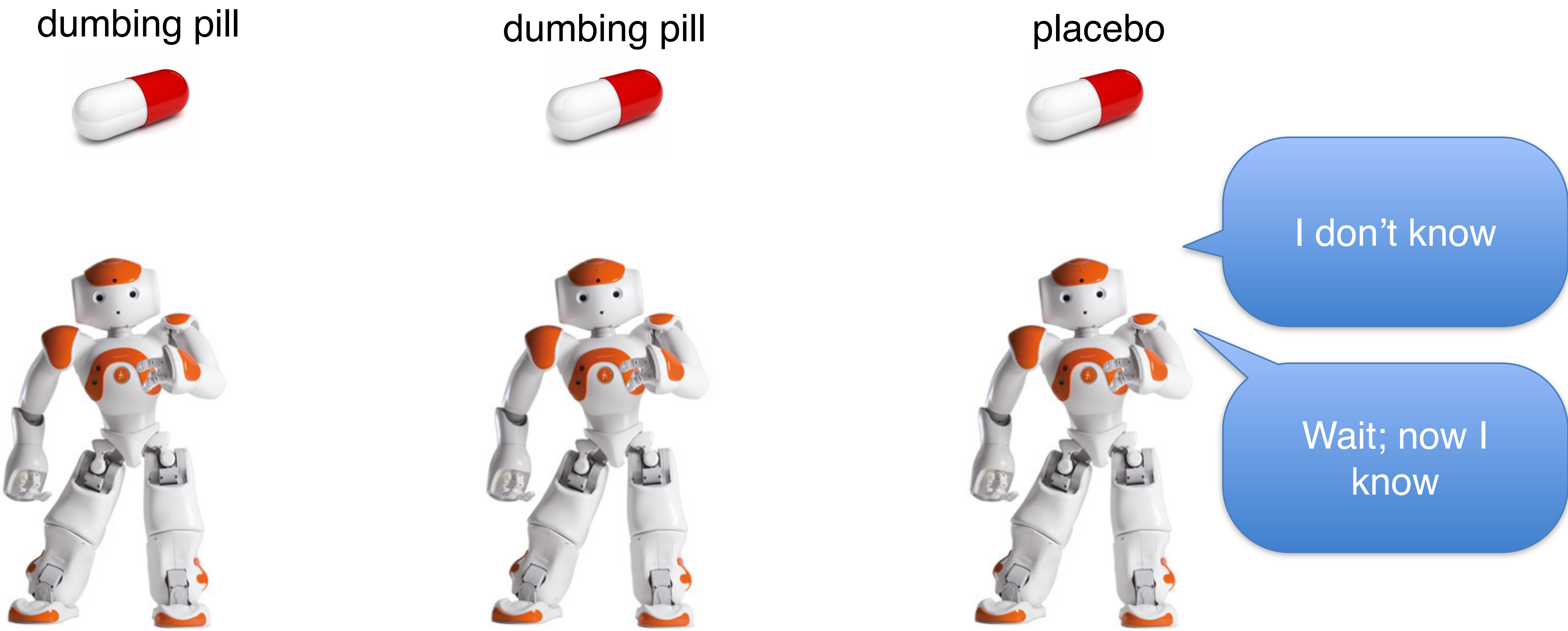
dumbing pill

placebo

I don’t know

Which pill did you receive?

Floridi's KG$_4$



dumbing pill

dumbing pill

placebo

I don't know

Wait; now I know

Which pill did you receive?

PROTOTYPES

```
Boolean iff Boolean Boolean
Boolean lt Moment Moment
Boolean gt Moment Moment
Boolean S Agent Moment
Boolean
Event eventOccurred Boolean
```

AXIOMS

```
forall [x,y] implies(iff(x,y), implies(x,y))
forall [x,y] implies(iff(x,y), implies(y,x))
forall [x,y] implies(and(x,y), x)
forall [x,y] implies(and(x,y), y)
forall [x,y] implies(and(x,y),and(y,x))
forall [x,y] implies(x, implies(y, and(x,y)))
forall [x] iff(not(not(x)), x)

forall [x,y,z] implies(and(lt(x,y),lt(y,z)), lt(x,z))
lt(t1,t2)
lt(t2,t3)
lt(t3,t4)
lt(t4,t5)
forall [x,y] iff(lt(x,y), gt(y,x))
forall [x,y] iff(lt(x,y), not(lt(y,x)))
```

```
forall [x,a,t] iff(K(a,t,x), and(B(a,t,x), x))

forall [x,y] implies(and(implies(x,y), not(y)), not(x))
forall [x,y,a,t] implies(and(K(a,t,implies(x,not(y))),K(a,t,y)),
K(a,t,not(x)))

gt(t4,t2)
forall [t,ti,tj,tk,p]
implies(and(gt(tj,ti),gt(tk,ti)),K(R3,t,implies(happens(action(R3,ingestDumbPill),ti),not(happens(eventOccurred(S(R3,tj,p)),tk)))
))
K(R3,t4,happens(eventOccurred(S(R3,t4,p)),t4))
```

CONJECTURE TO PROVE

```
K(R3,t4, not(happens(action(R3,ingestDumbPill),t2)))
```

PROTOTYPES

Boolean iff Boolean Boolean
Boolean lt Moment Moment
Boolean gt Moment Moment
Boolean S Agent Moment

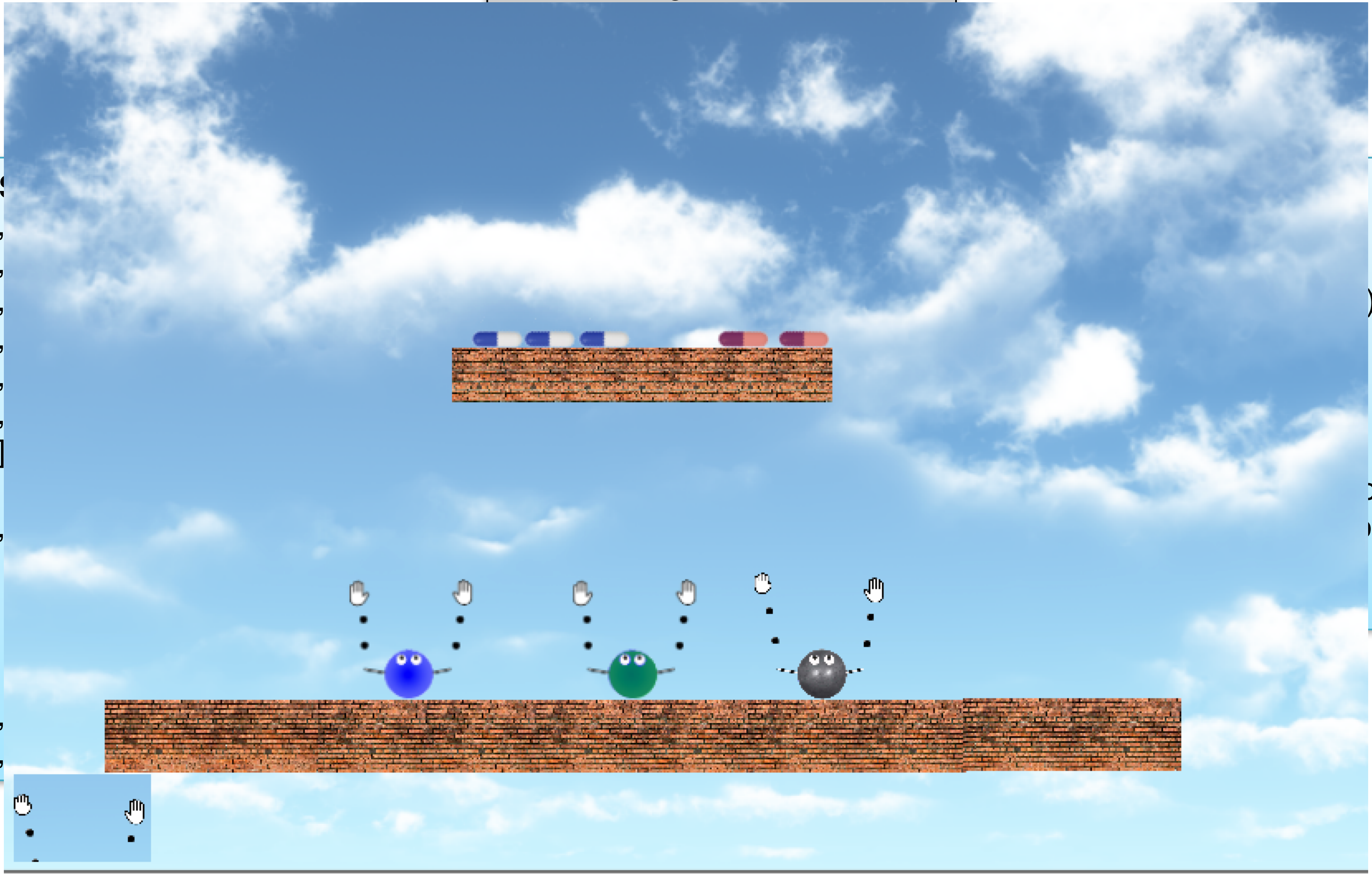
AXIOMS

forall [x,
forall [x,
forall [x,
forall [x,
forall [x,
forall [x,
forall [x]

forall [x,
lt(t1,t2)
lt(t2,t3)
lt(t3,t4)
lt(t4,t5)
forall [x,
forall [x,

),

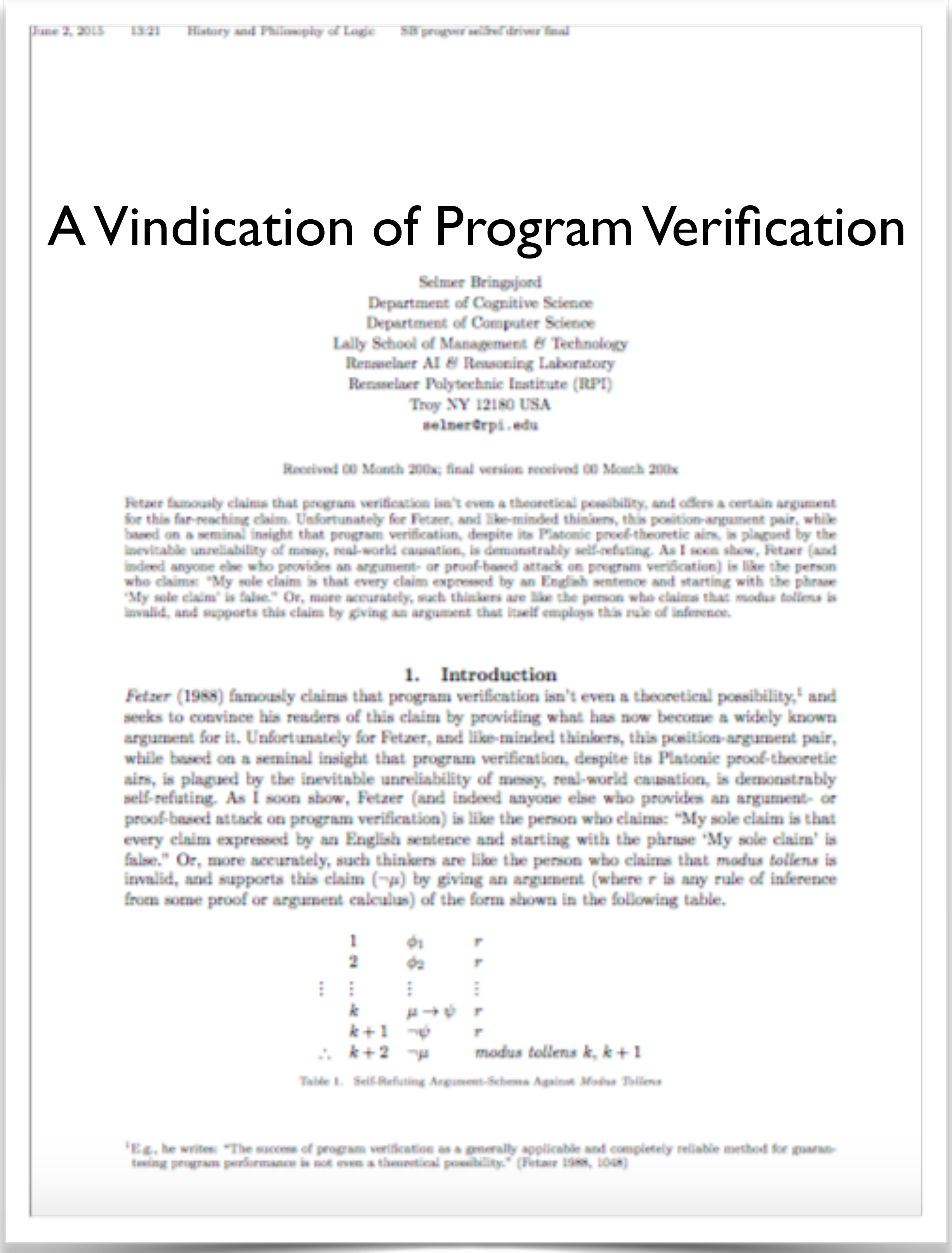
on(R3,i
)),tk)))



CONJECTURE TO PROVE

K(R3,t4, not(happens(action(R3,ingestDumbPill),t2)))

A Vindication of Program Verification



Selmer Bringsjord
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology
Rensselaer AI & Reasoning Laboratory
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
selmer@rpi.edu

Received 00 Month 200x; final version received 00 Month 200x

Fetzer famously claims that program verification isn't even a theoretical possibility, and offers a certain argument for this far-reaching claim. Unfortunately for Fetzer, and like-minded thinkers, this position-argument pair, while based on a seminal insight that program verification, despite its Platonic proof-theoretic airs, is plagued by the inevitable unreliability of messy, real-world causation, is demonstrably self-refuting. As I soon show, Fetzer (and indeed anyone else who provides an argument- or proof-based attack on program verification) is like the person who claims: "My sole claim is that every claim expressed by an English sentence and starting with the phrase 'My sole claim' is false." Or, more accurately, such thinkers are like the person who claims that *modus tollens* is invalid, and supports this claim by giving an argument that itself employs this rule of inference.

1. Introduction

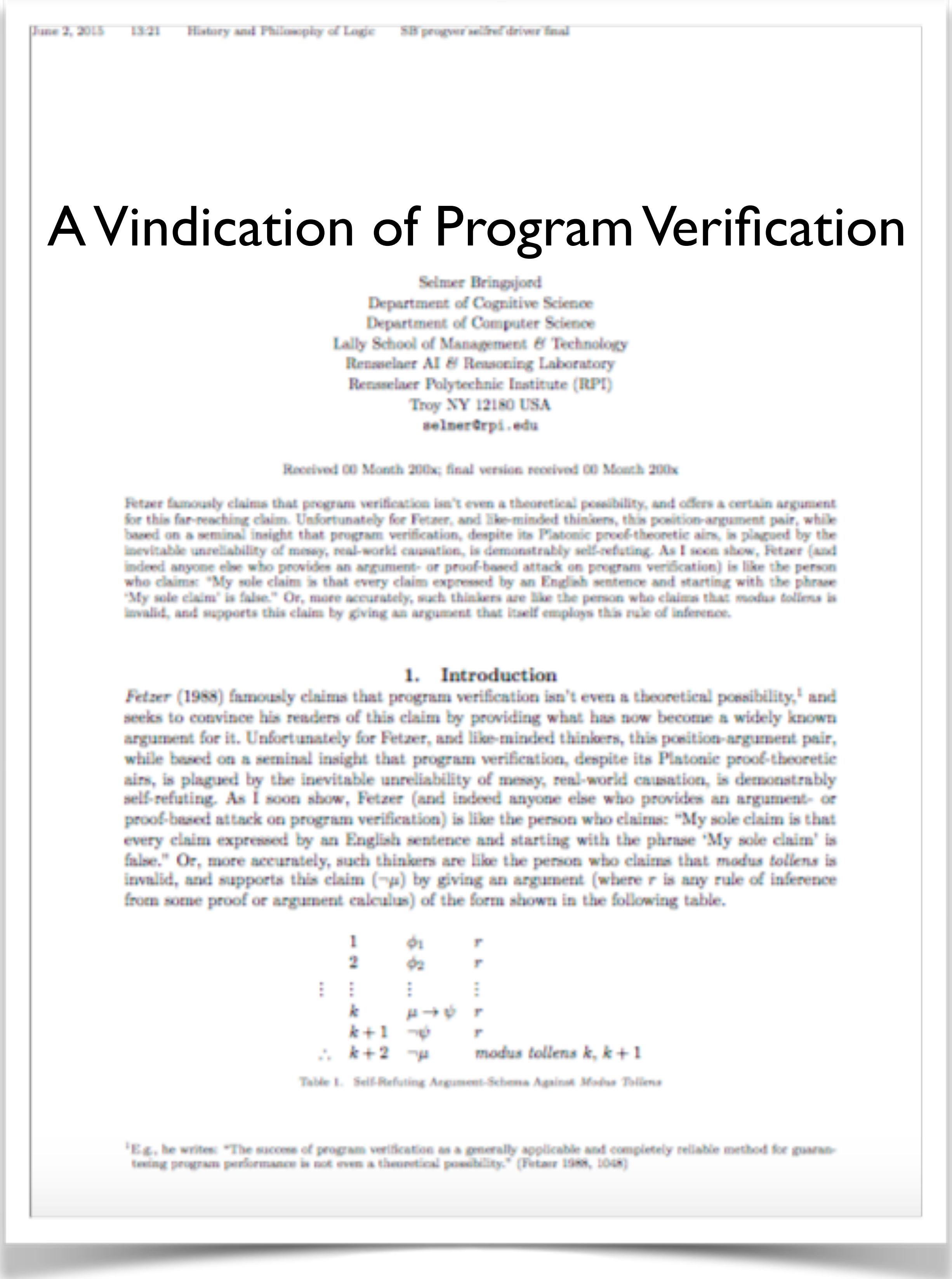
Fetzer (1988) famously claims that program verification isn't even a theoretical possibility,[1] and seeks to convince his readers of this claim by providing what has now become a widely known argument for it. Unfortunately for Fetzer, and like-minded thinkers, this position-argument pair, while based on a seminal insight that program verification, despite its Platonic proof-theoretic airs, is plagued by the inevitable unreliability of messy, real-world causation, is demonstrably self-refuting. As I soon show, Fetzer (and indeed anyone else who provides an argument- or proof-based attack on program verification) is like the person who claims: "My sole claim is that every claim expressed by an English sentence and starting with the phrase 'My sole claim' is false." Or, more accurately, such thinkers are like the person who claims that *modus tollens* is invalid, and supports this claim ($\neg\mu$) by giving an argument (where r is any rule of inference from some proof or argument calculus) of the form shown in the following table.

	1	ϕ_1	r
	2	ϕ_2	r
$\vdots$	$\vdots$	$\vdots$	$\vdots$
	k	$\mu \rightarrow \psi$	r
	$k+1$	$\neg\psi$	r
$\therefore$	$k+2$	$\neg\mu$	*modus tollens* k, $k+1$

Table 1. Self-Refuting Argument-Schema Against *Modus Tollens*

[1] E.g., he writes: "The success of program verification as a generally applicable and completely reliable method for guaranteeing program performance is not even a theoretical possibility." (Fetzer 1988, 1048)

Musk/Russell/Dietterich/…:
“Huh! Mere theory! Can’t be built.”



A Vindication of Program Verification

Selmer Bringsjord
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology
Rensselaer AI & Reasoning Laboratory
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
selmer@rpi.edu

Received 00 Month 200x; final version received 00 Month 200x

Fetzer famously claims that program verification isn't even a theoretical possibility, and offers a certain argument for this far-reaching claim. Unfortunately for Fetzer, and like-minded thinkers, this position-argument pair, while based on a seminal insight that program verification, despite its Platonic proof-theoretic airs, is plagued by the inevitable unreliability of messy, real-world causation, is demonstrably self-refuting. As I soon show, Fetzer (and indeed anyone else who provides an argument- or proof-based attack on program verification) is like the person who claims: "My sole claim is that every claim expressed by an English sentence and starting with the phrase 'My sole claim' is false." Or, more accurately, such thinkers are like the person who claims that *modus tollens* is invalid, and supports this claim by giving an argument that itself employs this rule of inference.

1. Introduction

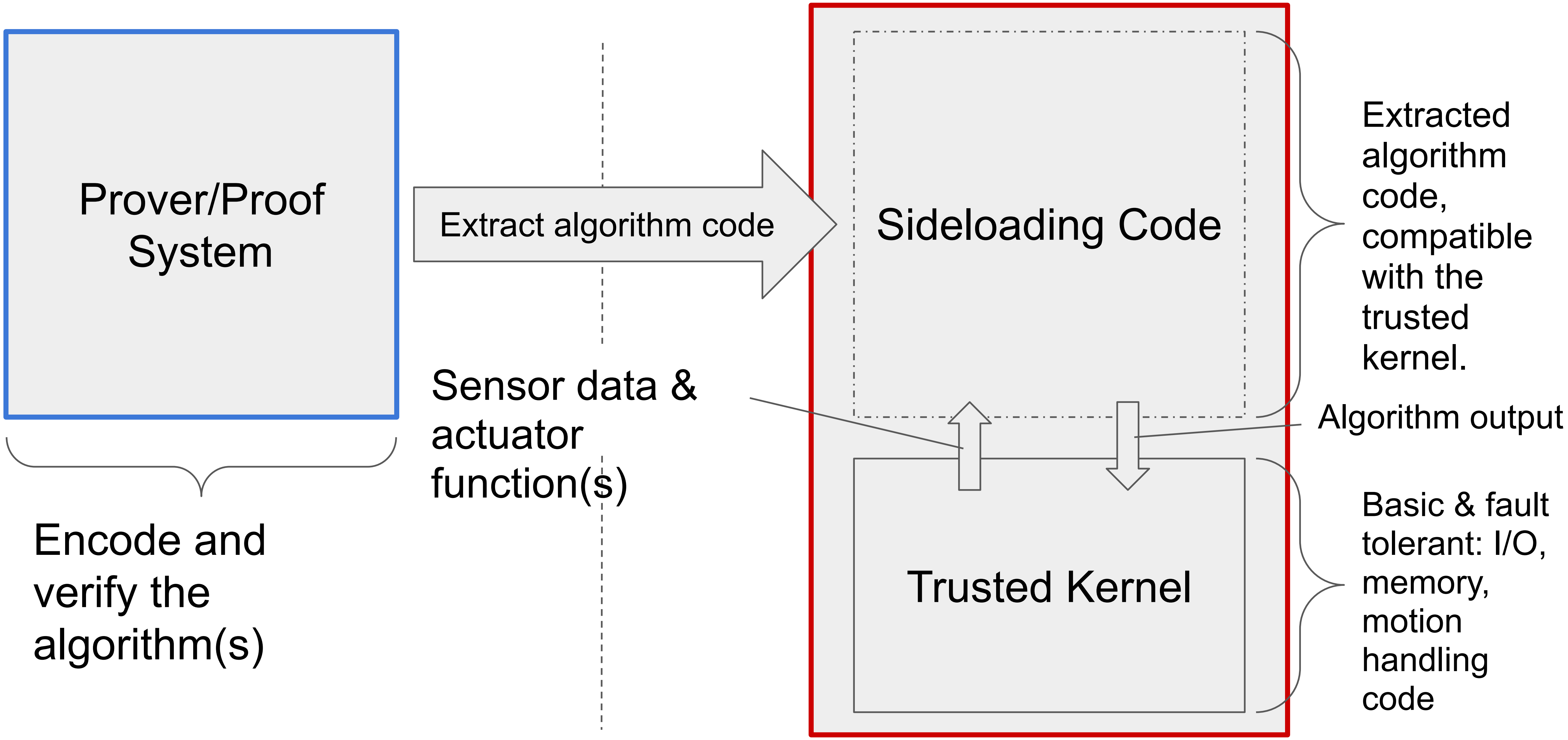
Fetzer (1988) famously claims that program verification isn't even a theoretical possibility,[1] and seeks to convince his readers of this claim by providing what has now become a widely known argument for it. Unfortunately for Fetzer, and like-minded thinkers, this position-argument pair, while based on a seminal insight that program verification, despite its Platonic proof-theoretic airs, is plagued by the inevitable unreliability of messy, real-world causation, is demonstrably self-refuting. As I soon show, Fetzer (and indeed anyone else who provides an argument- or proof-based attack on program verification) is like the person who claims: "My sole claim is that every claim expressed by an English sentence and starting with the phrase 'My sole claim' is false." Or, more accurately, such thinkers are like the person who claims that *modus tollens* is invalid, and supports this claim ($\neg\mu$) by giving an argument (where r is any rule of inference from some proof or argument calculus) of the form shown in the following table.

	1	ϕ_1	r
	2	ϕ_2	r
$\vdots$	$\vdots$	$\vdots$	$\vdots$
	k	$\mu \to \psi$	r
	$k+1$	$\neg\psi$	r
$\therefore$	$k+2$	$\neg\mu$	*modus tollens* k, $k+1$

Table 1. Self-Refuting Argument-Schema Against *Modus Tollens*

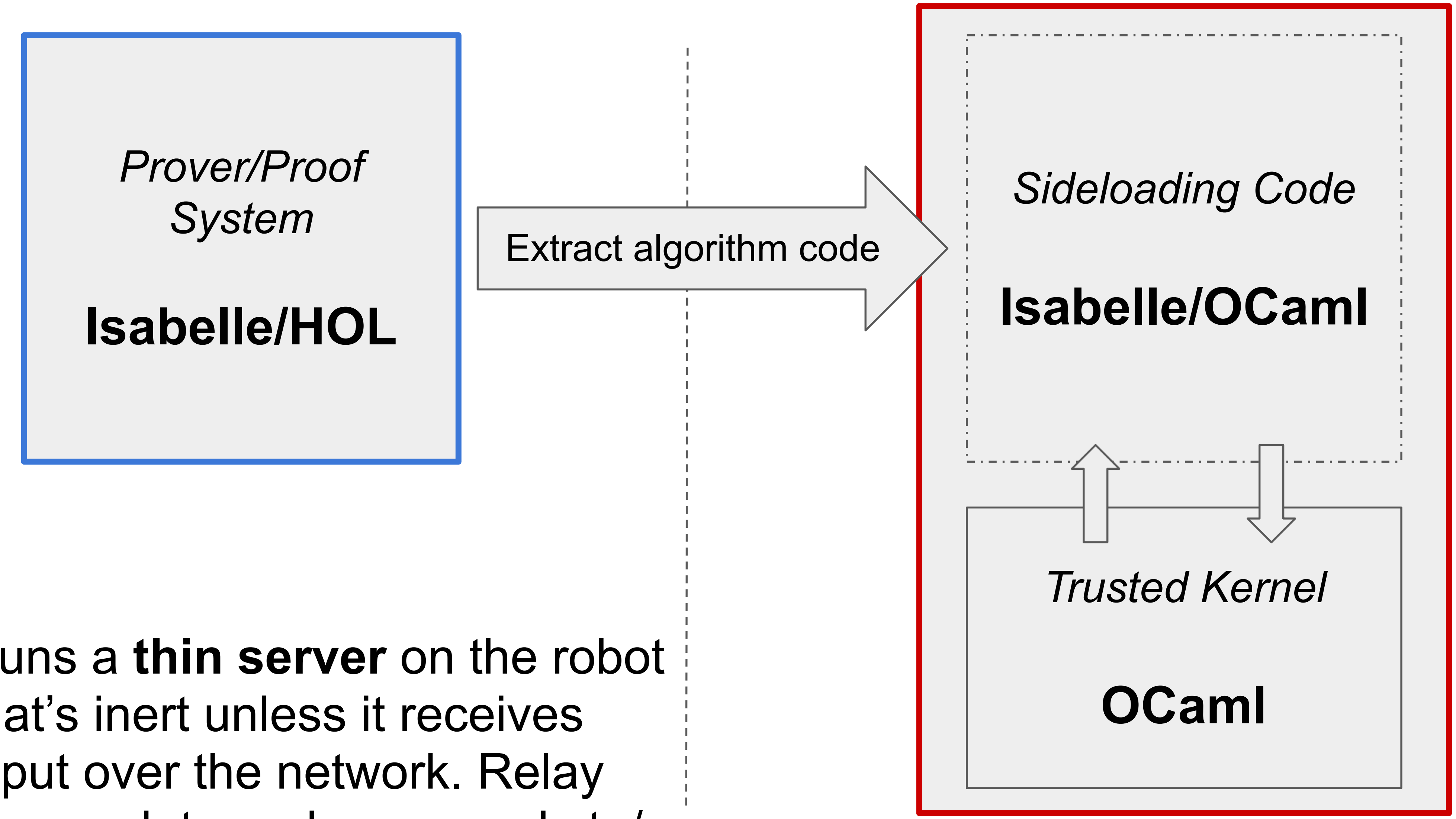
[1] E.g., he writes: "The success of program verification as a generally applicable and completely reliable method for guaranteeing program performance is not even a theoretical possibility." (Fetzer 1988, 1048)

One Architecture for How to Build It


Prover/Proof System
Extract algorithm code
Sideloading Code
Extracted algorithm code, compatible with the trusted kernel.
Sensor data & actuator function(s)
Algorithm output
Trusted Kernel
Basic & fault tolerant: I/O, memory, motion handling code

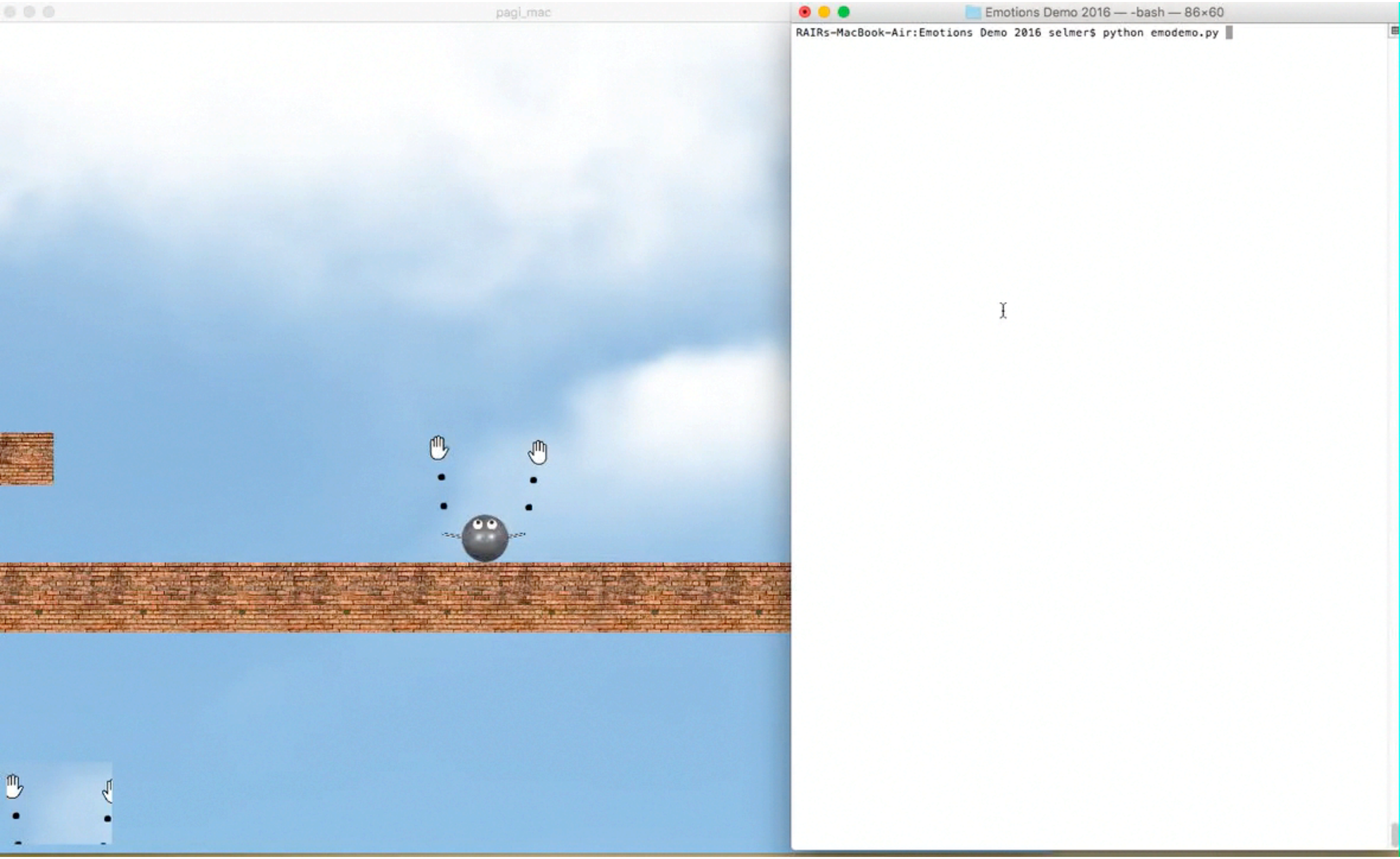
Encode and verify the algorithm(s)

Working Proof of Concept Now Up!



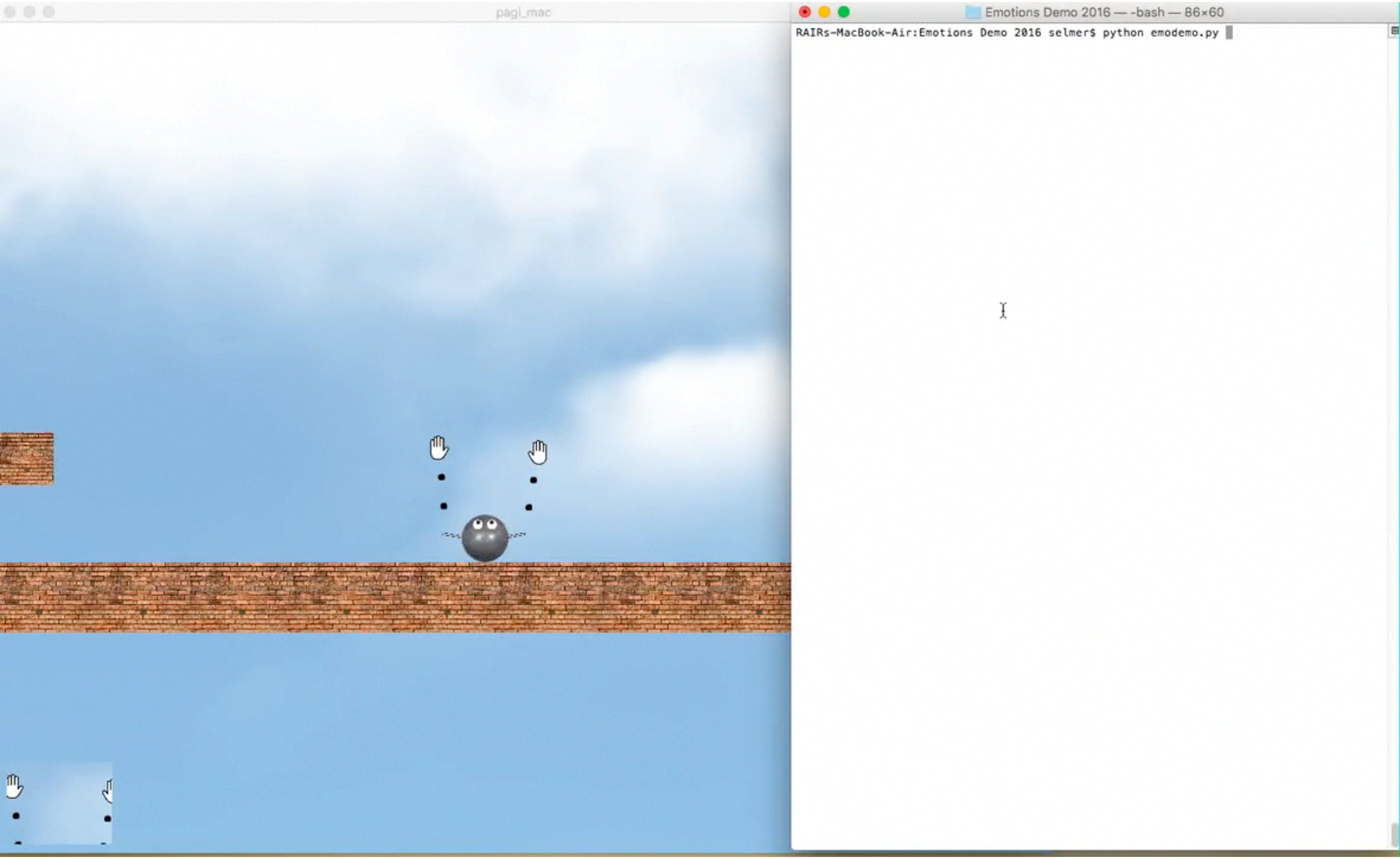
Runs a **thin server** on the robot that's inert unless it receives input over the network. Relay sensor data and commands to/from the robot via **OCaml**.

Supererogation & Formalized-Emotion Demo



```
RAIRs-MacBook-Air:Emotions Demo 2016 selmer$ python emodemo.py
```

Supererogation & Formalized-Emotion Demo



```
RAIRs-MacBook-Air:Emotions Demo 2016 selmer$ python emodemo.py 
```

In original Arkoudas-Bringsjord dialect of *CEC*:

In original Arkoudas-Bringsjord dialect of *CEC*:

$$\frac{\mathbf{S}(a, \phi, b, t)}{\mathbf{K}(b, \phi, t)}$$

In original Arkoudas-Bringsjord dialect of *CEC*:

$$\frac{\mathbf{S}(a, \phi, b, t)}{\mathbf{K}(b, \phi, t)}$$

Now working with NLU-infused cognitive calculi:

In original Arkoudas-Bringsjord dialect of *CEC*:

$$\frac{\mathbf{S}(a, \phi, b, t)}{\mathbf{K}(b, \phi, t)}$$

Now working with NLU-infused cognitive calculi:

$$\frac{\mathbf{S}(a, \sigma, b, t), \mathcal{K}_a, \Theta}{\mathbf{K}(b, \mu(\pi(\sigma)), t)}$$

In original Arkoudas-Bringsjord dialect of *CEC*:

$$\frac{\mathbf{S}(a, \phi, b, t)}{\mathbf{K}(b, \phi, t)}$$

Now working with NLU-infused cognitive calculi:

$$\frac{\mathbf{S}(a, \overset{\text{string}}{\overset{\downarrow}{\sigma}}, b, t), \overset{\text{knowledge-base of } a}{\overset{\downarrow}{\mathcal{K}_a}}, \overset{\text{background theory}}{\overset{\downarrow}{\Theta}}}{\mathbf{K}(b, \mu(\pi(\sigma)), t)}$$

In original Arkoudas-Bringsjord dialect of *CEC*:

$$\frac{\mathbf{S}(a, \phi, b, t)}{\mathbf{K}(b, \phi, t)}$$

Now working with NLU-infused cognitive calculi:

$$\frac{\mathbf{S}(a, \overset{\substack{\text{string}\\ \downarrow}}{\sigma}, b, t), \overset{\substack{\text{knowledge-base of } a\\ \downarrow}}{\mathcal{K}_a}, \overset{\substack{\text{background theory}\\ \downarrow}}{\Theta}}{\mathbf{K}(b, \underset{\substack{\uparrow\\ \text{mapping to formulae}}}{\mu}(\underset{\substack{\uparrow\\ \text{parse to intermediary form}}}{\pi}(\sigma)), t)}$$

With PPs in the Picture, Logicist NLU is Tricky

With PPs in the Picture, Logicist NLU is Tricky

John is pouring water.

With PPs in the Picture, Logicist NLU is Tricky

John is pouring water.

$$\exists x[Pours(j,x) \land Water(x)]$$

With PPs in the Picture, Logicist NLU is Tricky

John is pouring water.

$$\exists x[Pours(j,x) \wedge Water(x)]$$

John is pouring water in the pitcher.

With PPs in the Picture, Logicist NLU is Tricky

John is pouring water.

$$\exists x[Pours(j,x) \land Water(x)]$$

John is pouring water in the pitcher.

$$\phi := \exists x[Pours(j,x) \land Water(x) \land In(j, pitcher22)]$$

With PPs in the Picture, Logicist NLU is Tricky

John is pouring water.

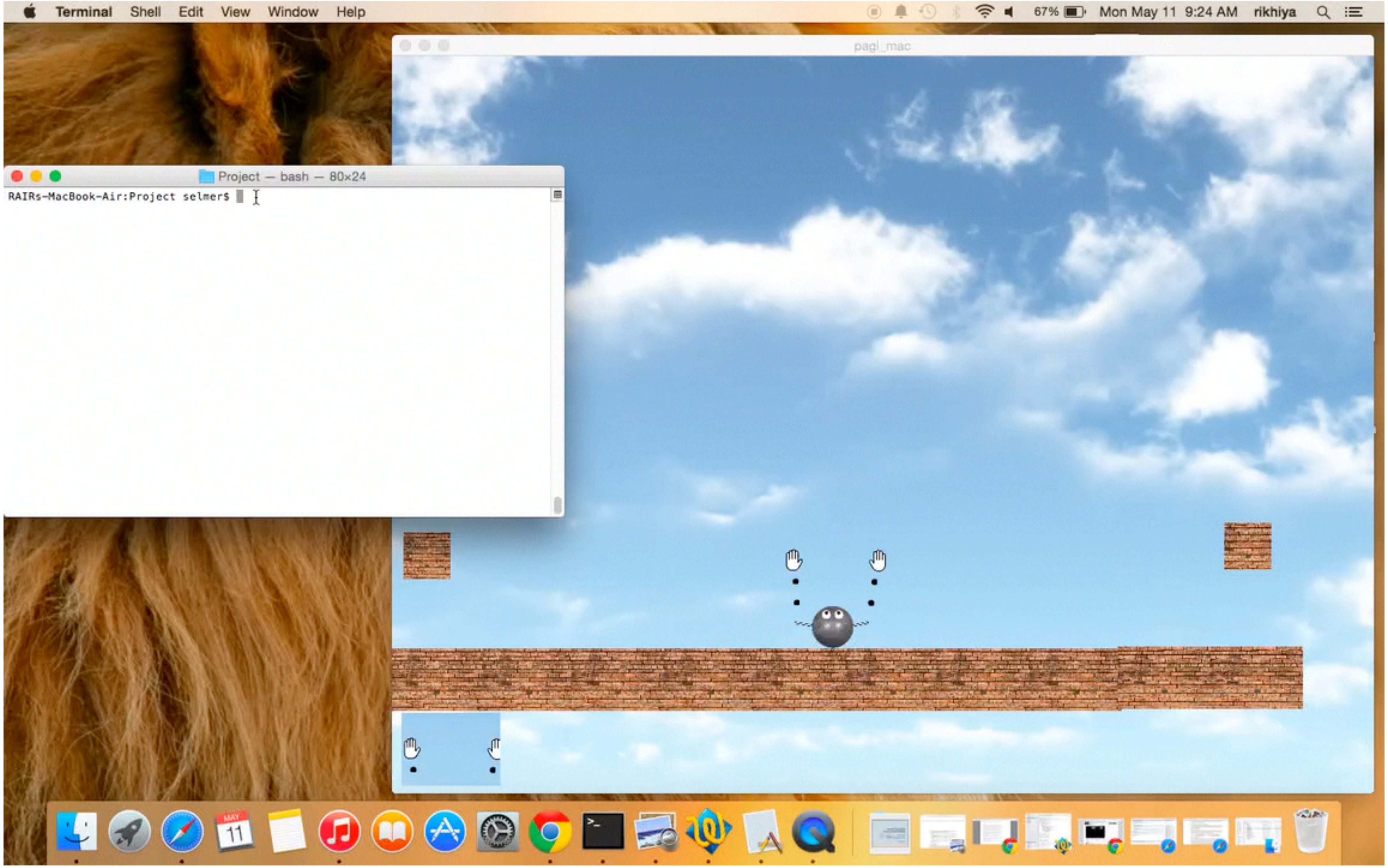
$$\exists x[Pours(j,x) \land Water(x)]$$

John is pouring water in the pitcher.

$$\phi := \exists x[Pours(j,x) \land Water(x) \land In(j, pitcher22)]$$

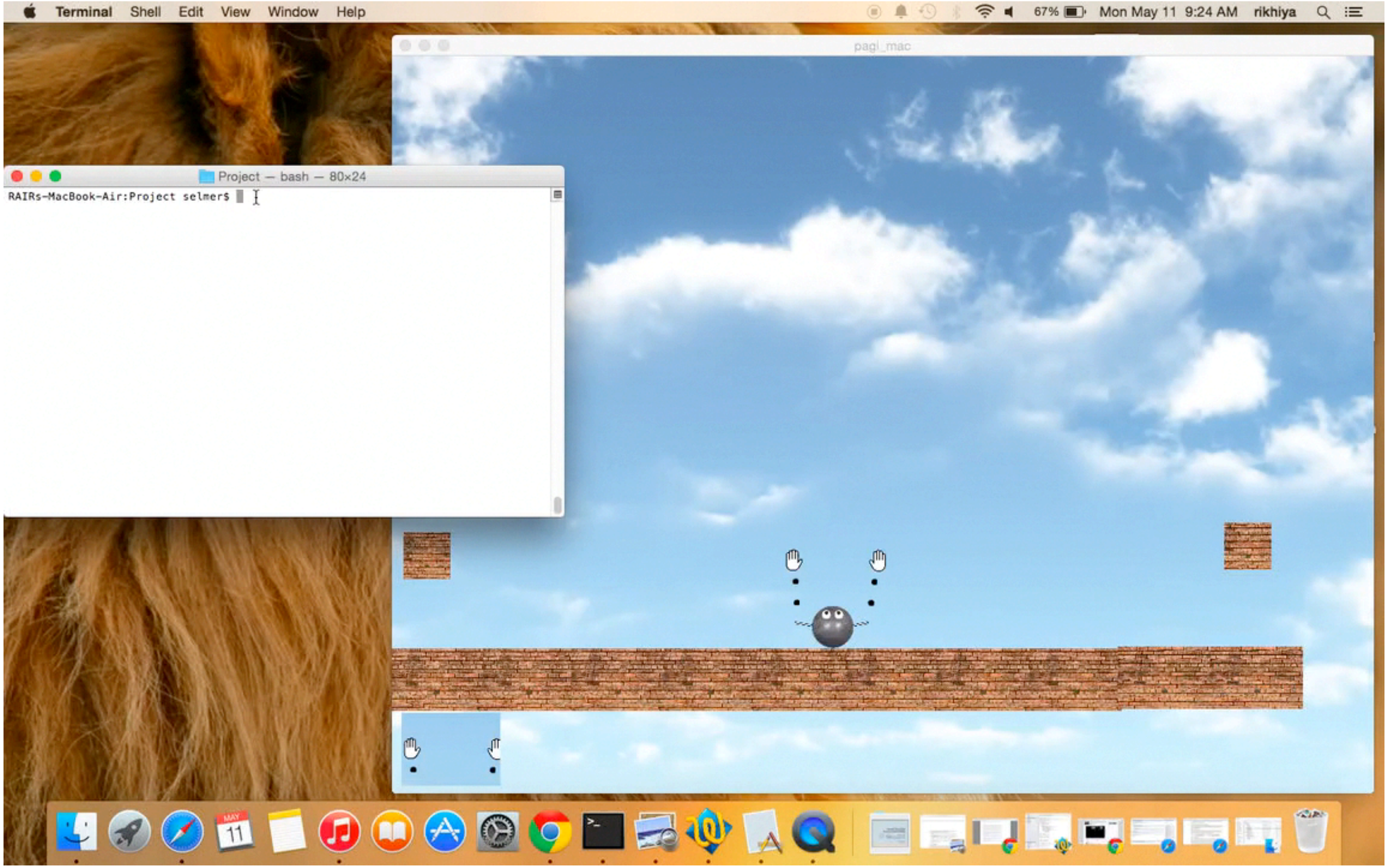
$$\{\phi\} \vdash In(j, pitcher22) \quad (!)$$

PP-Infused Commands to PAGI Guy (Softbot)



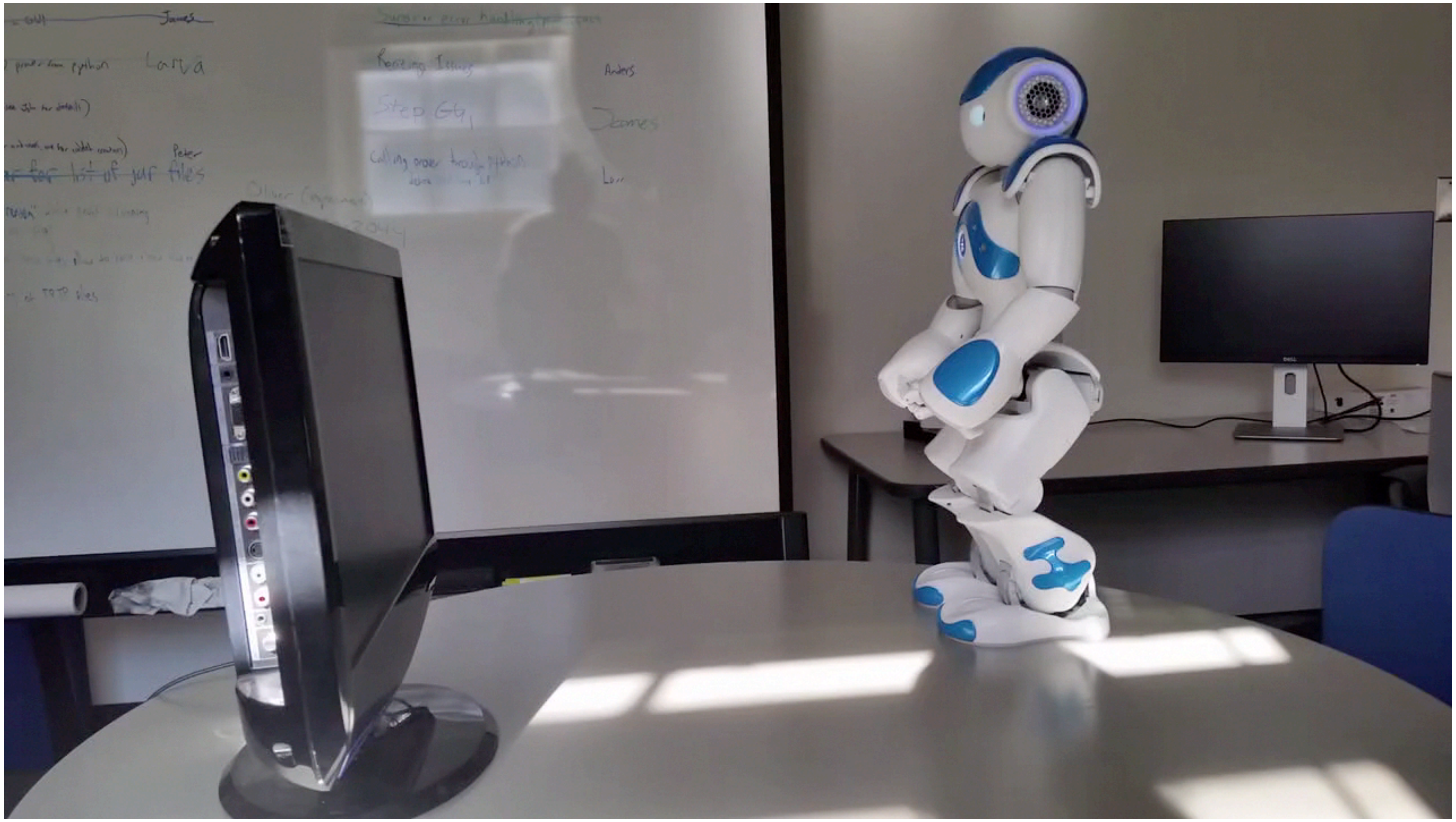
```
RAIRs-MacBook-Air:Project selmer$
```

PP-Infused Commands to PAGI Guy (Softbot)

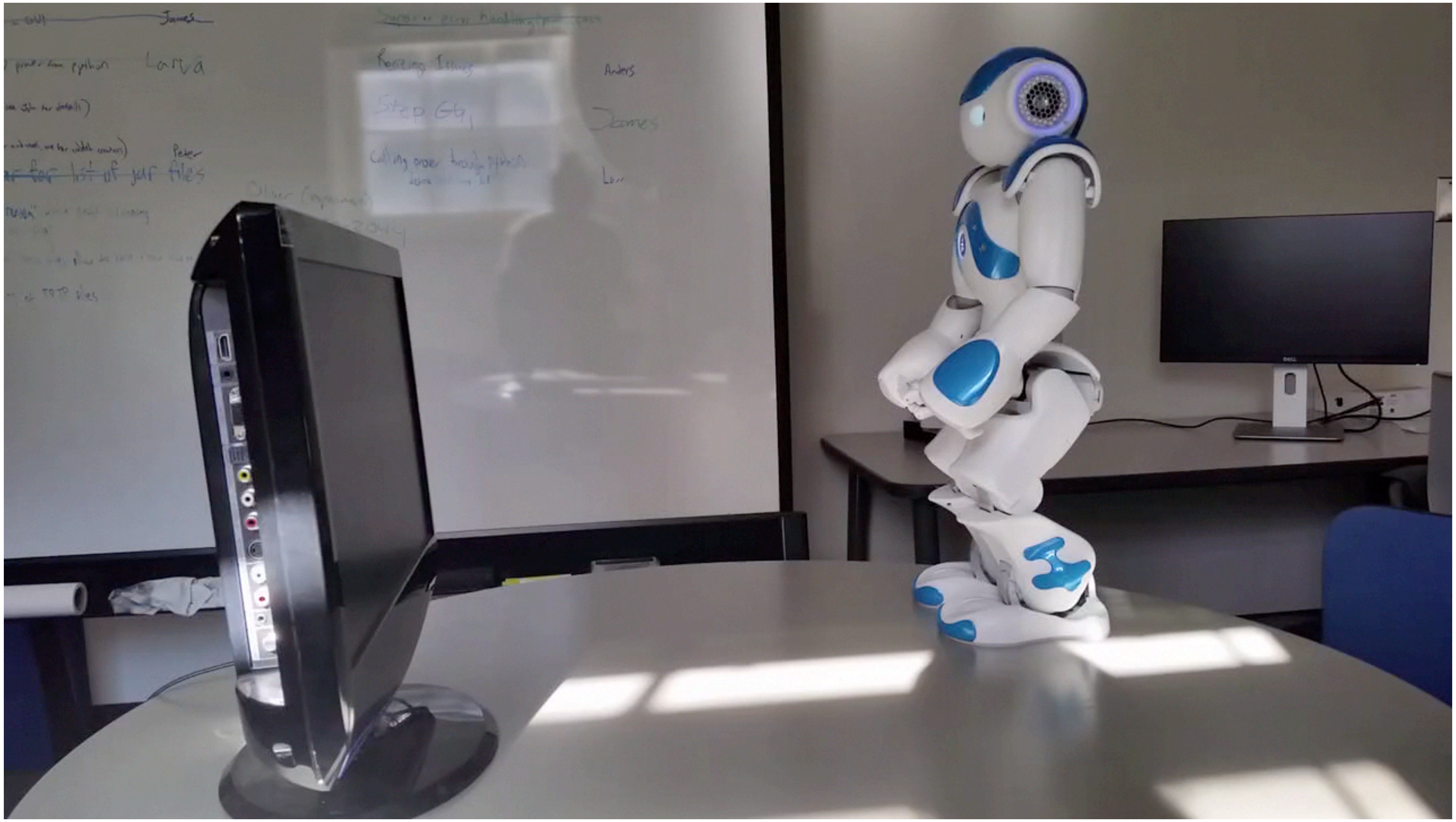


```
RAIRs-MacBook-Air:Project selmer$
```

PP-Infused Commands to Robot



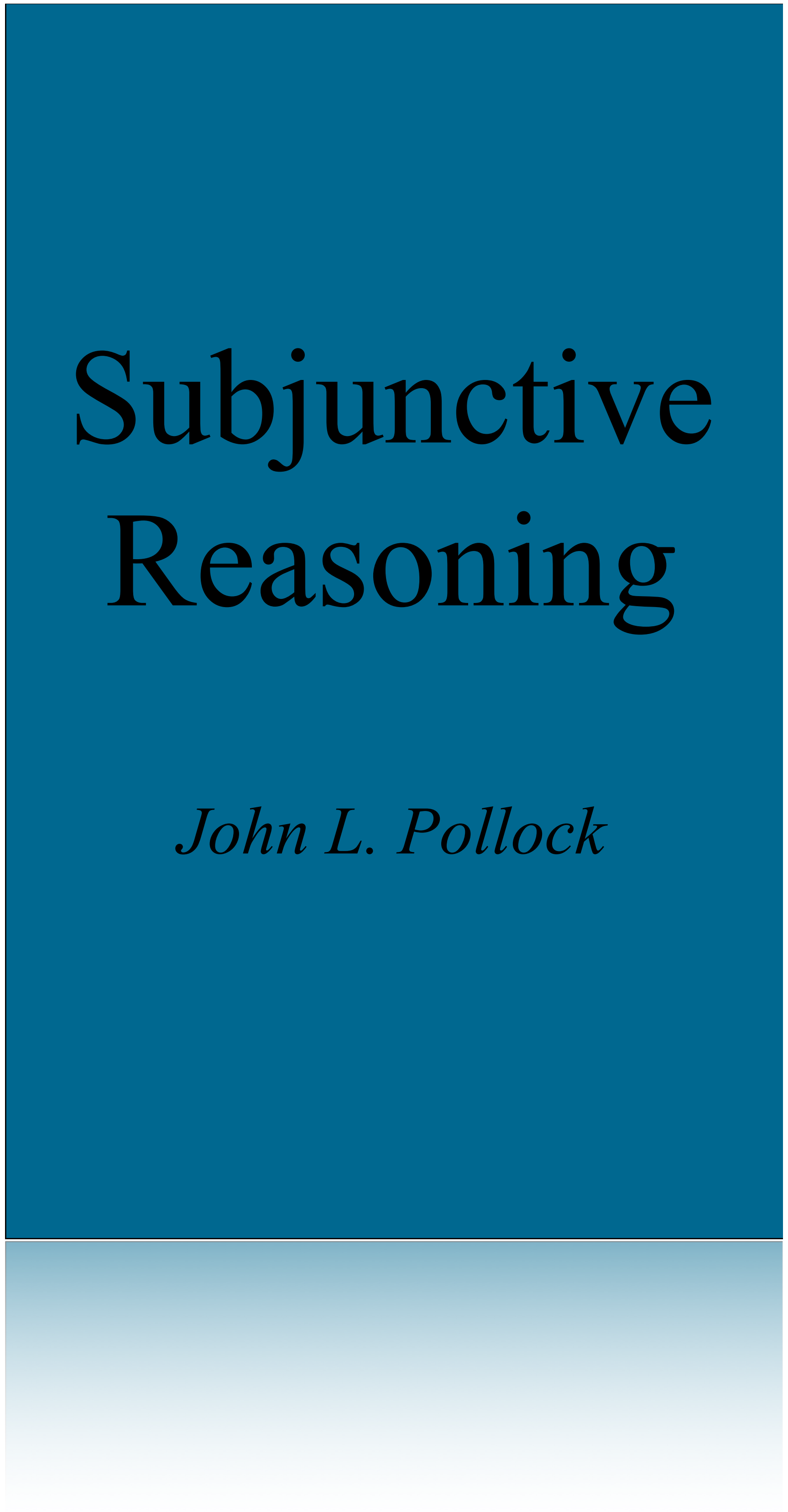
PP-Infused Commands to Robot



Subjunctive Reasoning

Our approach is closest to (Pollock 1976), “corrected” by co-tenability (e.g., Chisholm).

A modern, proof-theoretic computational rendering of Pollock’s approach.



Subjunctive Reasoning

John L. Pollock

Pollock’s approach, briefly

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.

- Simple subjunctive **>**

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**

Pollock's approach, briefly

- Pollock's analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Pollock's approach, briefly

- Pollock's analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Layer 2

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Layer 2

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.

- Simple subjunctive **>**

- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Layer 2	M

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Layer 2 M E

Pollock's approach, briefly

- Pollock's analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Layer 2 M E $\gg$

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Layer 2 M E $\gg$ $\Rrightarrow$

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Layer 2	M	E	$\gg$	$\Rrightarrow$
Layer 1				

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Layer 2	M	E	$\gg$	$\Rrightarrow$
Layer 1		$>$		

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Layer 2	M E $\gg$ $\Rrightarrow$
Layer 1	$>$
Layer 0	

Pollock’s approach, briefly

- Pollock’s analysis of subjunctives can be best understood as a layered approach.
- Simple subjunctive **>**
- Four other subjunctives defined in terms of the simple subjunctive **>**

1. **might be**
2. **even if**
3. **necessitates**
4. **laws**

Layer 2	M E $\gg$ $\Rrightarrow$
Layer 1	$>$
Layer 0	Possible worlds analysis of $>$

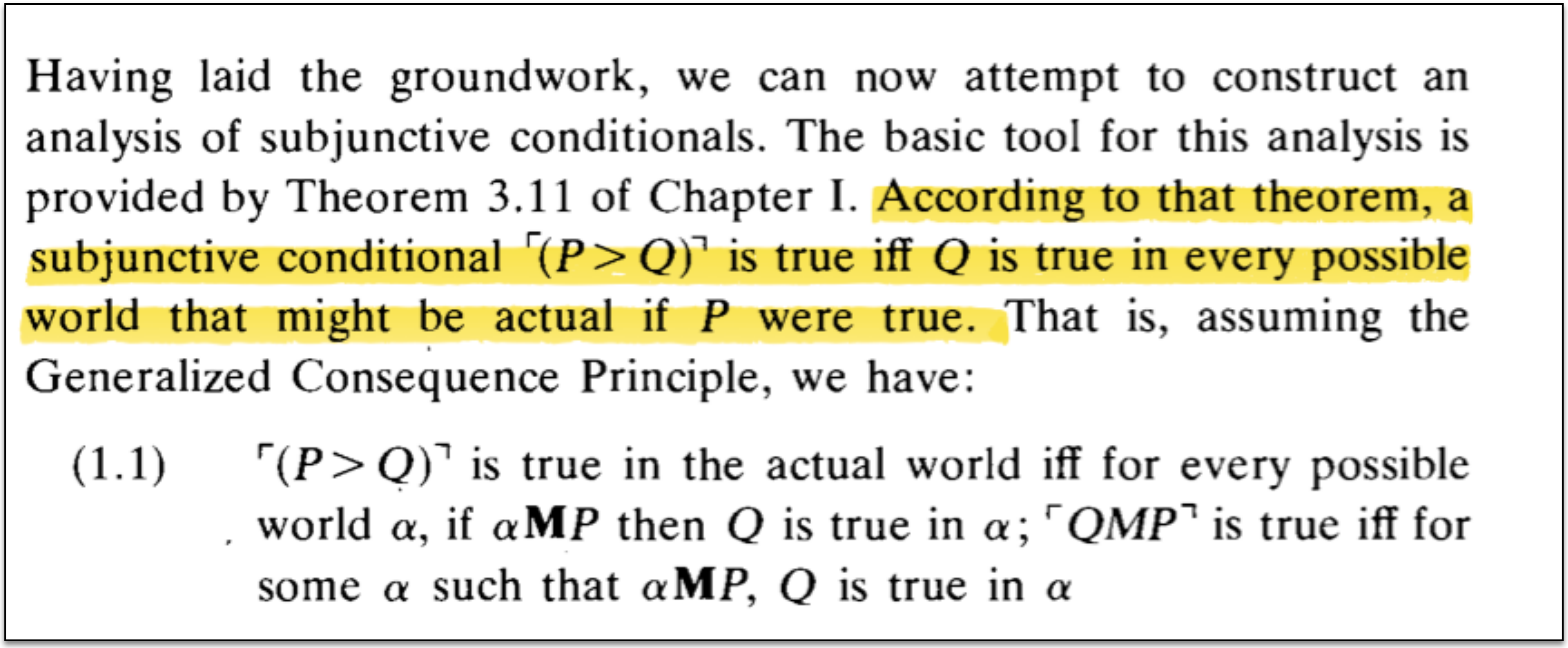
Pollock’s approach, briefly

Conditional	Informally	Example	Reduction
E	even if	**Even if the witch doctor dances it won’t rain**	$(QEP) \equiv Q \wedge (P > Q)$
M	might be	**If it was not raining outside, it might be snowing**	$(QMP) \equiv \neg(P > \neg Q)$
$\gg$	necessitates	**If I were to strike this match, it would light**	$P \gg Q \equiv P > Q \wedge [(\neg P \wedge \neg Q) > (P > Q)]$
$\Rrightarrow$	general laws	**All pulsars are neutron stars**	**A tad complex**

(Pollock 1976)

Pollock’s approach, briefly

- Analysis of **>**



Having laid the groundwork, we can now attempt to construct an analysis of subjunctive conditionals. The basic tool for this analysis is provided by Theorem 3.11 of Chapter I. According to that theorem, a subjunctive conditional $\ulcorner(P > Q)\urcorner$ is true iff Q is true in every possible world that might be actual if P were true. That is, assuming the Generalized Consequence Principle, we have:

(1.1) $\ulcorner(P > Q)\urcorner$ is true in the actual world iff for every possible world α, if $\alpha \mathbf{M} P$ then Q is true in α; $\ulcorner Q\mathbf{M}P\urcorner$ is true iff for some α such that $\alpha \mathbf{M} P$, Q is true in α

Our Analysis

> introduction

$\mathcal{W}$: set of all world statements

$$\beta \vdash \phi > \psi$$

iff

$\forall w \in \mathcal{W}$

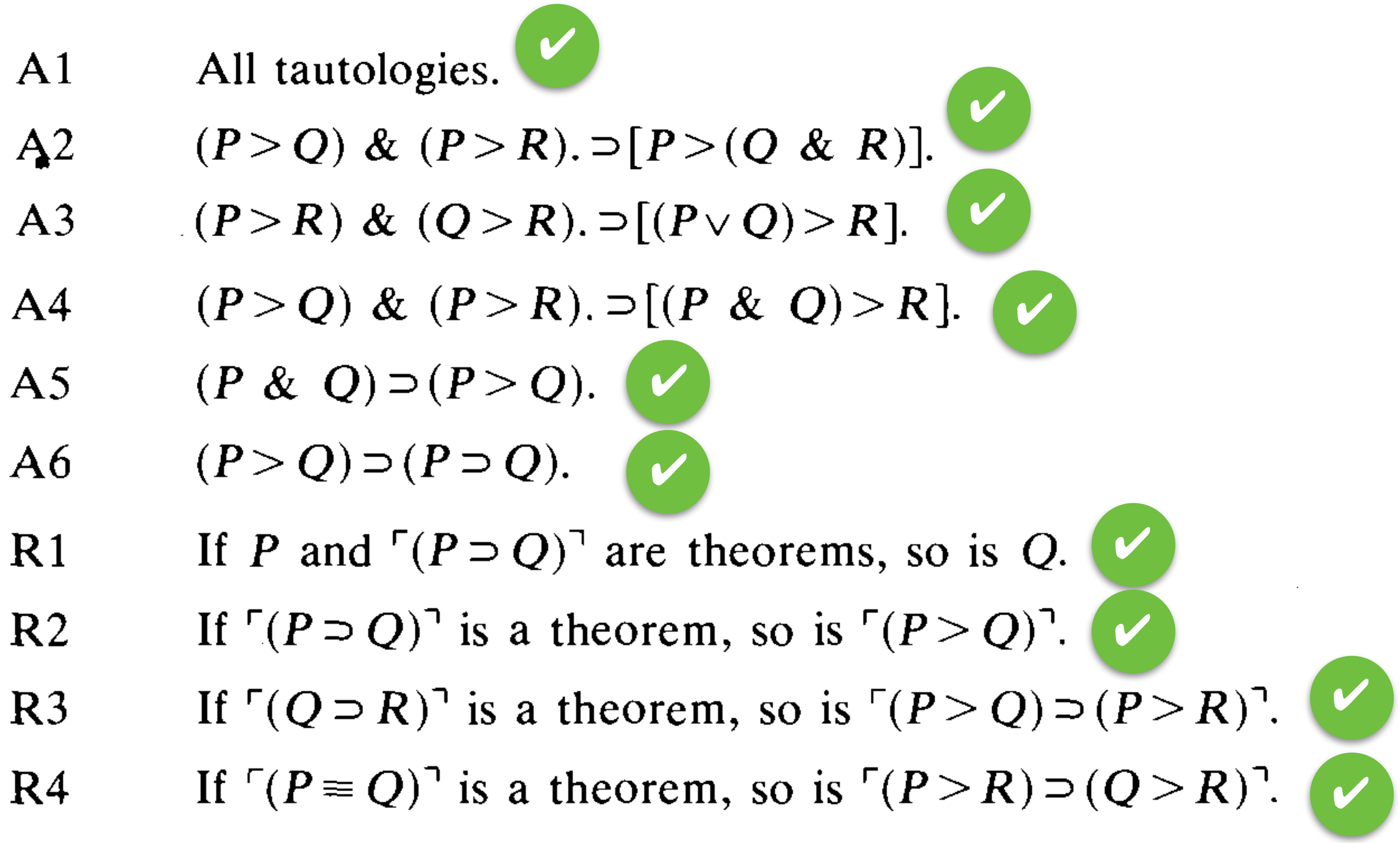
$$\left(\begin{array}{c} \text{Consistent}\,[\mathbf{g}(\beta) + w + \phi] \\ \Rightarrow \\ \mathbf{g}(\beta) + w + \phi \vdash \psi \end{array} \right)$$

> elimination

$$\beta \cup \{\phi > \psi, \phi\} \vdash \psi$$

How good is our analysis?

- Our analysis satisfies Pollock's axioms for simple subjunctives.



A1 All tautologies. ✓

A2 $(P>Q)\ \&\ (P>R).\supset[P>(Q\ \&\ R)]$. ✓

A3 $(P>R)\ \&\ (Q>R).\supset[(P\vee Q)>R]$. ✓

A4 $(P>Q)\ \&\ (P>R).\supset[(P\ \&\ Q)>R]$. ✓

A5 $(P\ \&\ Q)\supset(P>Q)$. ✓

A6 $(P>Q)\supset(P\supset Q)$. ✓

R1 If P and $\ulcorner(P\supset Q)\urcorner$ are theorems, so is Q. ✓

R2 If $\ulcorner(P\supset Q)\urcorner$ is a theorem, so is $\ulcorner(P>Q)\urcorner$. ✓

R3 If $\ulcorner(Q\supset R)\urcorner$ is a theorem, so is $\ulcorner(P>Q)\supset(P>R)\urcorner$. ✓

R4 If $\ulcorner(P\equiv Q)\urcorner$ is a theorem, so is $\ulcorner(P>R)\supset(Q>R)\urcorner$. ✓

(if **g**({P>Q, **…**}) contains P>Q

Version 2, July 15, 2014

Simple Subjunctive

> introduction

$$\beta \vdash \phi > \psi$$

iff

$$\mathbf{g}(\beta,\phi) + \phi \vdash \psi$$

> elimination

$$\beta \cup \{\phi > \psi, \phi\} \vdash \psi$$

Option 1

$$\mathbf{g}(\beta,\phi) = \underset{\rho\in\{\rho\subseteq\beta \ | \ \mathsf{Con}[\rho+\phi]\}}{\operatorname{argmax}} |\rho|$$

Option 2

$\mathcal{W}_L$: the set of all world literals

$$\mathbf{g}(\beta,\phi) = \begin{cases} \beta \text{ if } \mathsf{Con}[\beta+\phi] \\ \text{the largest member of} \left\{ \rho \subset \beta \ \middle| \ \begin{array}{l} \mathsf{Con}[\rho+\phi] \\ \wedge \forall\tau.\ \tau \in (\beta-\rho) \Rightarrow \tau \in \mathcal{W}_L \end{array} \right\} \end{cases}$$

Controlled Natural Language

Needed: A Human-Robot Dialog System

Needed: A Human-Robot Dialog System

- Queries and requests assume knowledge of the robot's capabilities.

Needed: A Human-Robot Dialog System

- Queries and requests assume knowledge of the robot’s capabilities.
  - E.g. “Robot, search for damaged Naobots in your area.”

Needed: A Human-Robot Dialog System

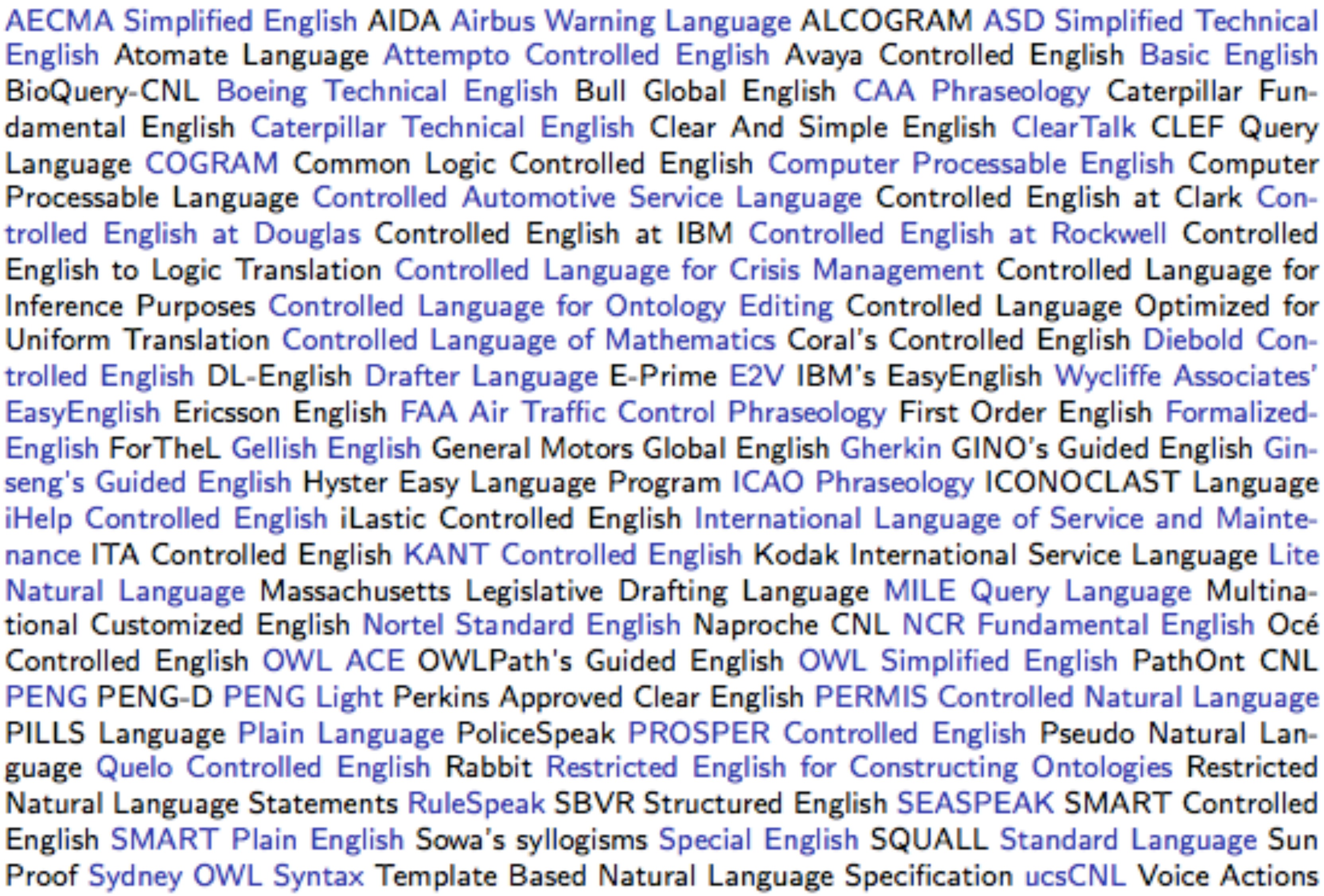
- Queries and requests assume knowledge of the robot's capabilities.
  - E.g. "Robot, search for damaged Naobots in your area."
- Natural language interactions happen over long periods of time.

Needed: A Human-Robot Dialog System

- Queries and requests assume knowledge of the robot's capabilities.
    - E.g. "Robot, search for damaged Naobots in your area."
- Natural language interactions happen over long periods of time.
    - E.g. "Robot, why did you take less safer route to complete the mission yesterday?"

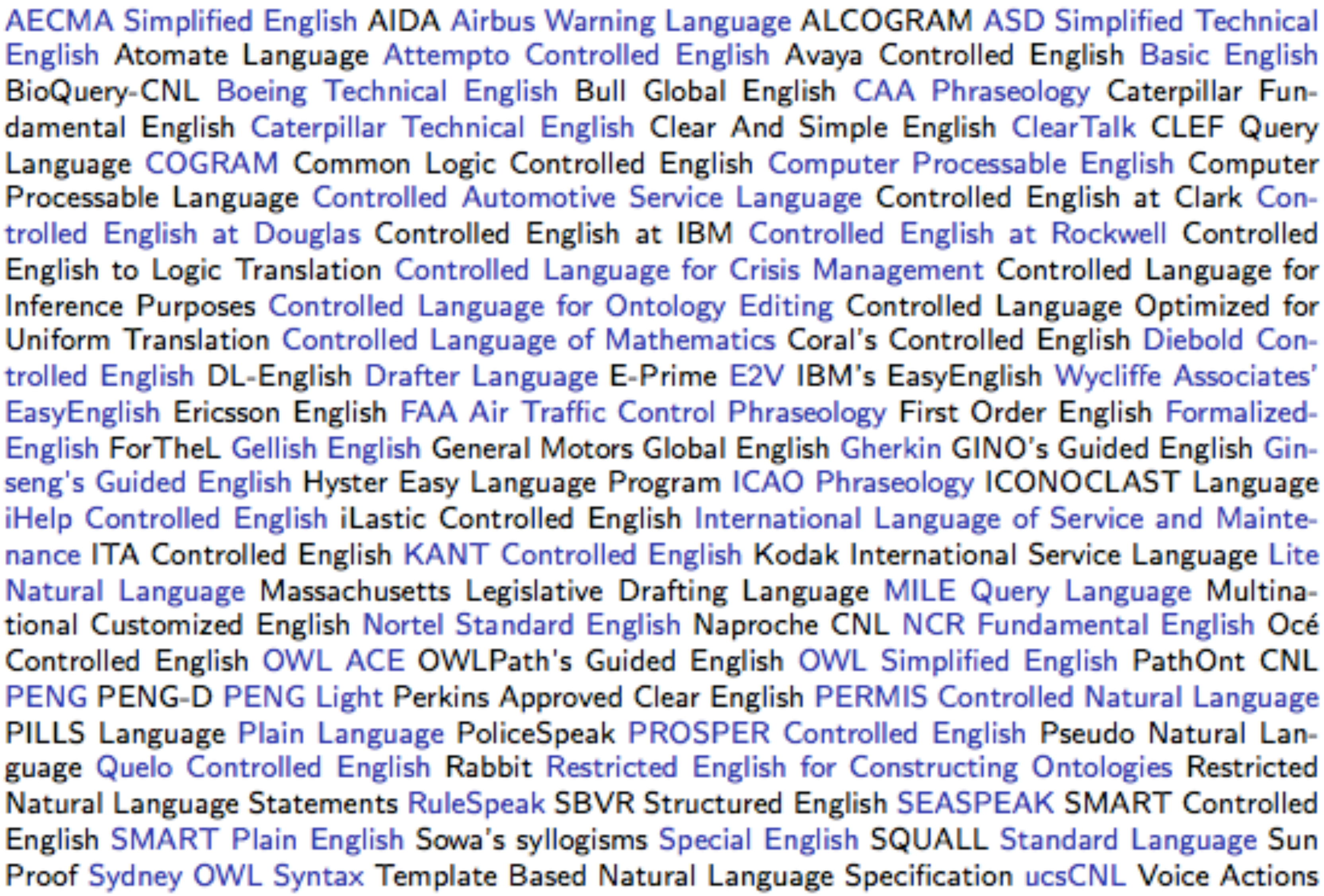
Controlled Natural Languages

Controlled Natural Languages



AECMA Simplified English AIDA Airbus Warning Language ALCOGRAM ASD Simplified Technical English Atomate Language Attempto Controlled English Avaya Controlled English Basic English BioQuery-CNL Boeing Technical English Bull Global English CAA Phraseology Caterpillar Fundamental English Caterpillar Technical English Clear And Simple English ClearTalk CLEF Query Language COGRAM Common Logic Controlled English Computer Processable English Computer Processable Language Controlled Automotive Service Language Controlled English at Clark Controlled English at Douglas Controlled English at IBM Controlled English at Rockwell Controlled English to Logic Translation Controlled Language for Crisis Management Controlled Language for Inference Purposes Controlled Language for Ontology Editing Controlled Language Optimized for Uniform Translation Controlled Language of Mathematics Coral's Controlled English Diebold Controlled English DL-English Drafter Language E-Prime E2V IBM's EasyEnglish Wycliffe Associates' EasyEnglish Ericsson English FAA Air Traffic Control Phraseology First Order English Formalized-English ForTheL Gellish English General Motors Global English Gherkin GINO's Guided English Ginseng's Guided English Hyster Easy Language Program ICAO Phraseology ICONOCLAST Language iHelp Controlled English iLastic Controlled English International Language of Service and Maintenance ITA Controlled English KANT Controlled English Kodak International Service Language Lite Natural Language Massachusetts Legislative Drafting Language MILE Query Language Multinational Customized English Nortel Standard English Naproche CNL NCR Fundamental English Océ Controlled English OWL ACE OWLPath's Guided English OWL Simplified English PathOnt CNL PENG PENG-D PENG Light Perkins Approved Clear English PERMIS Controlled Natural Language PILLS Language Plain Language PoliceSpeak PROSPER Controlled English Pseudo Natural Language Quelo Controlled English Rabbit Restricted English for Constructing Ontologies Restricted Natural Language Statements RuleSpeak SBVR Structured English SEASPEAK SMART Controlled English SMART Plain English Sowa's syllogisms Special English SQUALL Standard Language Sun Proof Sydney OWL Syntax Template Based Natural Language Specification ucsCNL Voice Actions

Controlled Natural Languages



AECMA Simplified English AIDA Airbus Warning Language ALCOGRAM ASD Simplified Technical English Atomate Language Attempto Controlled English Avaya Controlled English Basic English BioQuery-CNL Boeing Technical English Bull Global English CAA Phraseology Caterpillar Fundamental English Caterpillar Technical English Clear And Simple English ClearTalk CLEF Query Language COGRAM Common Logic Controlled English Computer Processable English Computer Processable Language Controlled Automotive Service Language Controlled English at Clark Controlled English at Douglas Controlled English at IBM Controlled English at Rockwell Controlled English to Logic Translation Controlled Language for Crisis Management Controlled Language for Inference Purposes Controlled Language for Ontology Editing Controlled Language Optimized for Uniform Translation Controlled Language of Mathematics Coral's Controlled English Diebold Controlled English DL-English Drafter Language E-Prime E2V IBM's EasyEnglish Wycliffe Associates' EasyEnglish Ericsson English FAA Air Traffic Control Phraseology First Order English Formalized-English ForTheL Gellish English General Motors Global English Gherkin GINO's Guided English Ginseng's Guided English Hyster Easy Language Program ICAO Phraseology ICONOCLAST Language iHelp Controlled English iLastic Controlled English International Language of Service and Maintenance ITA Controlled English KANT Controlled English Kodak International Service Language Lite Natural Language Massachusetts Legislative Drafting Language MILE Query Language Multinational Customized English Nortel Standard English Naproche CNL NCR Fundamental English Océ Controlled English OWL ACE OWLPath's Guided English OWL Simplified English PathOnt CNL PENG PENG-D PENG Light Perkins Approved Clear English PERMIS Controlled Natural Language PILLS Language Plain Language PoliceSpeak PROSPER Controlled English Pseudo Natural Language Quelo Controlled English Rabbit Restricted English for Constructing Ontologies Restricted Natural Language Statements RuleSpeak SBVR Structured English SEASPEAK SMART Controlled English SMART Plain English Sowa's syllogisms Special English SQUALL Standard Language Sun Proof Sydney OWL Syntax Template Based Natural Language Specification ucsCNL Voice Actions

from (Kuhn 2009)

Grammatical Framework

Grammatical Framework

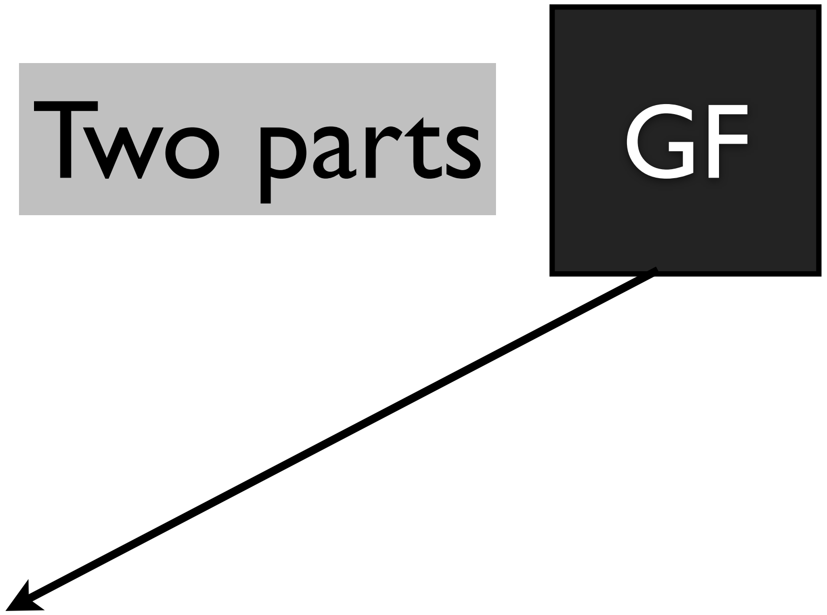
GF

Grammatical Framework

Two parts

GF

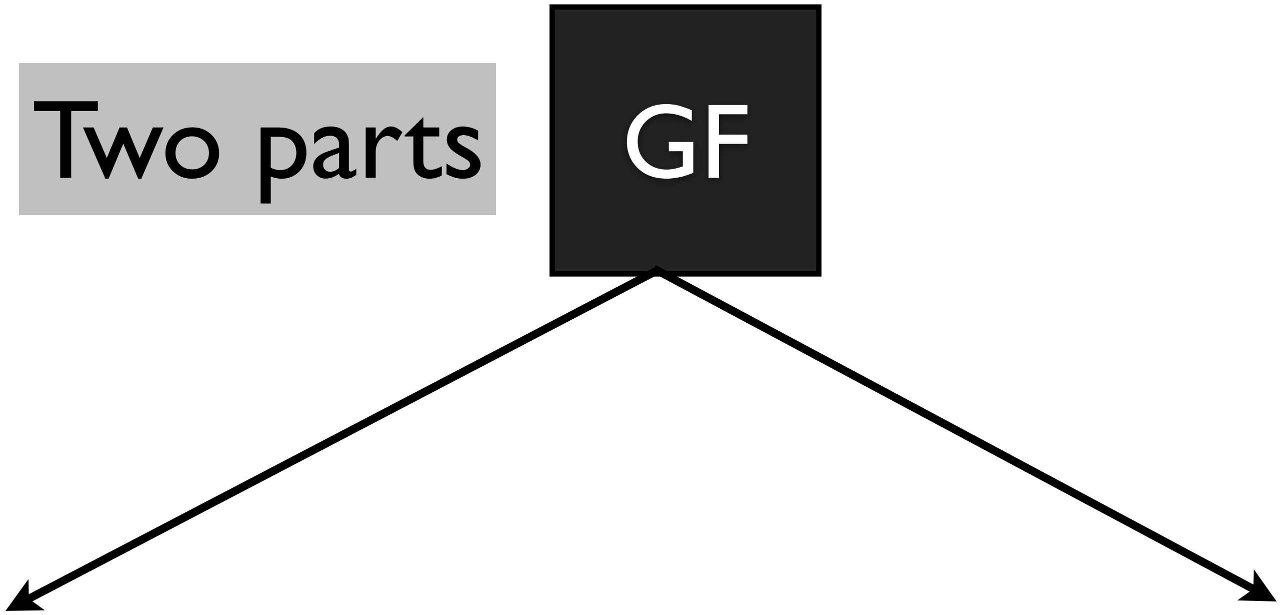
Grammatical Framework



Two parts

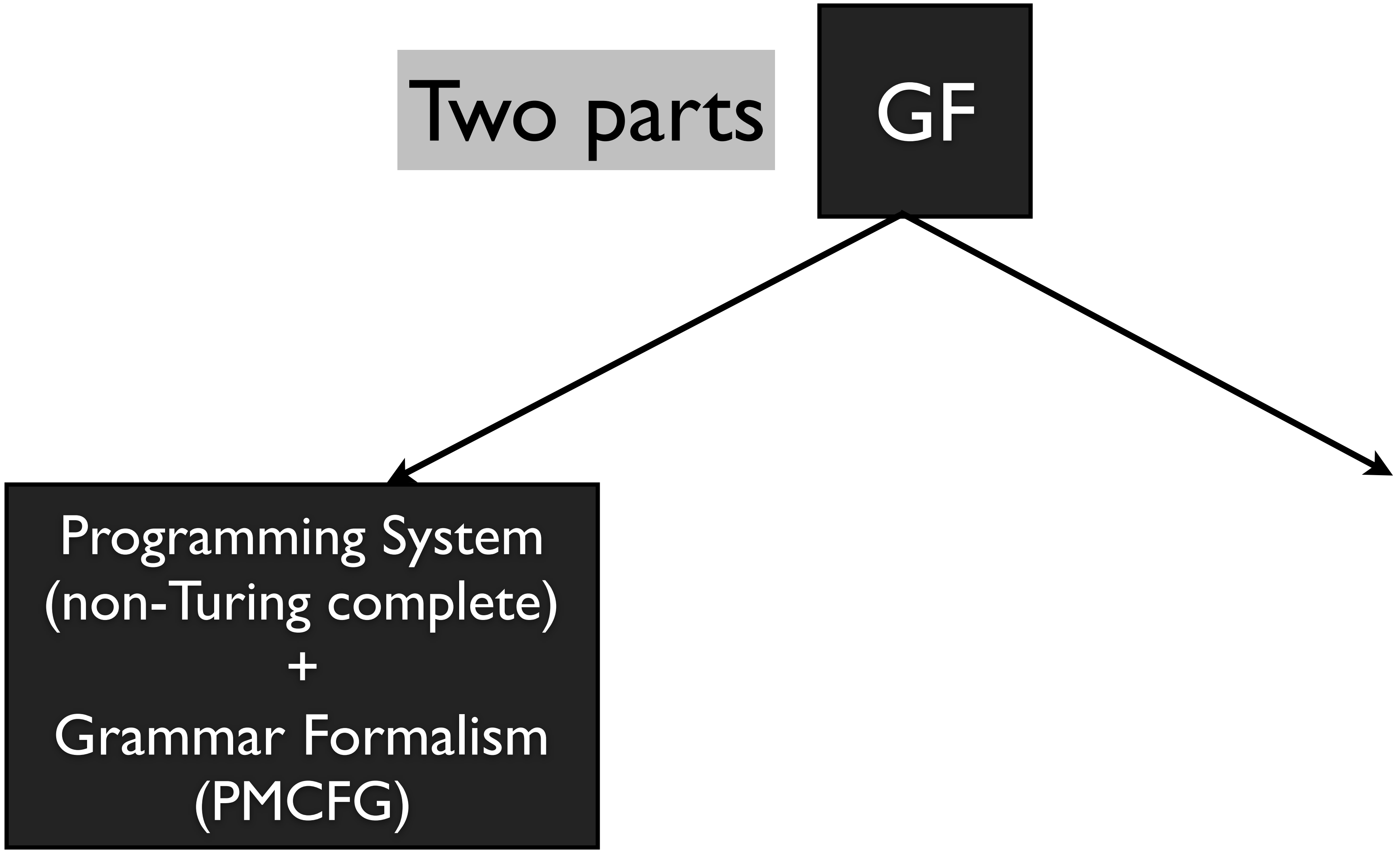
GF

Grammatical Framework



Two parts

Grammatical Framework

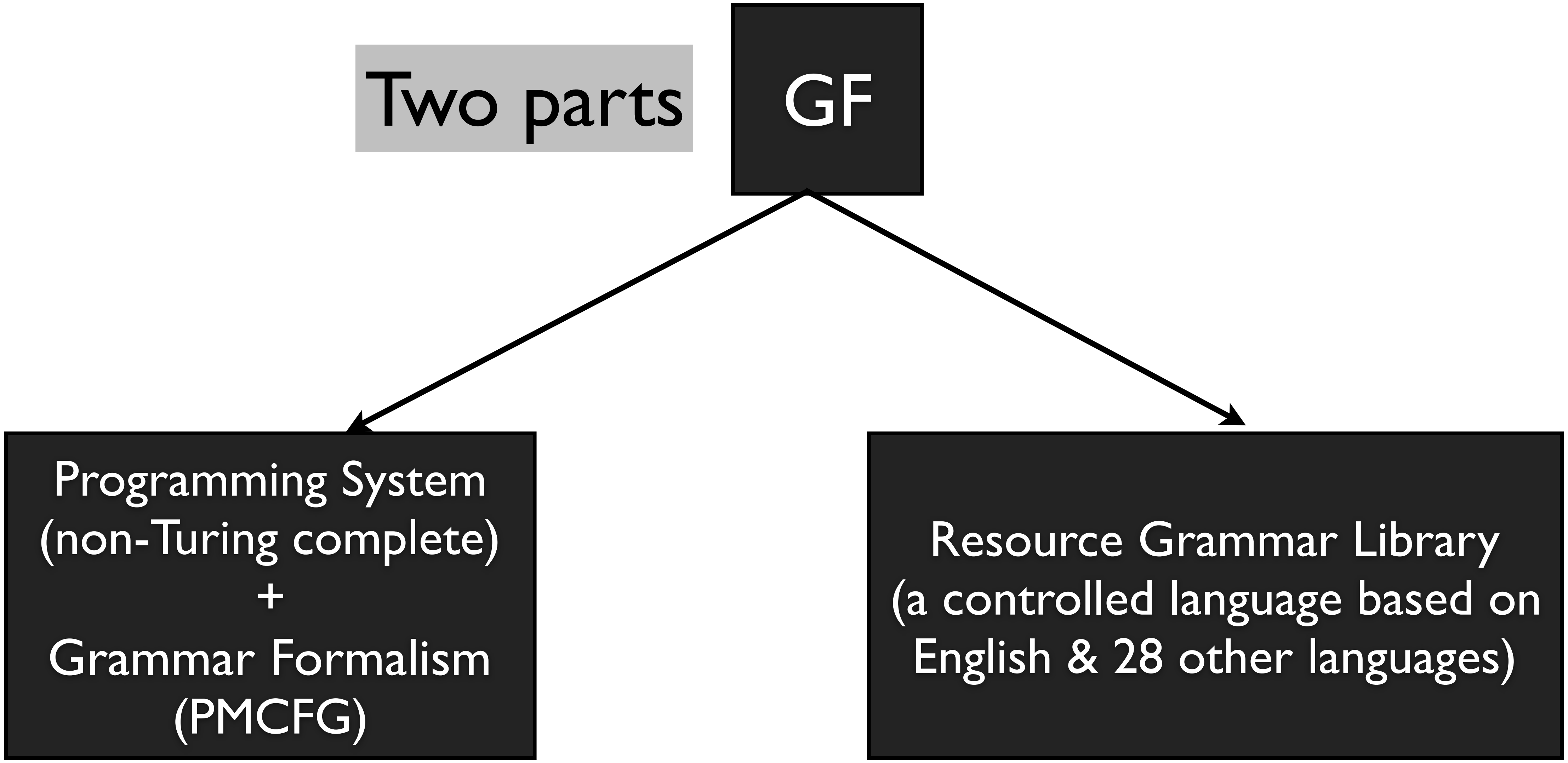


Two parts

GF

Programming System
(non-Turing complete)
+
Grammar Formalism
(PMCFG)

Grammatical Framework



Two parts

GF

Programming System
(non-Turing complete)
+
Grammar Formalism
(PMCFG)

Resource Grammar Library
(a controlled language based on
English & 28 other languages)

Parallel Multiple Context Free Grammars

Parallel Multiple Context Free Grammars

- A grammar formalism that is:

Parallel Multiple Context Free Grammars

- A grammar formalism that is:
  - more powerful than context-free grammars

Parallel Multiple Context Free Grammars

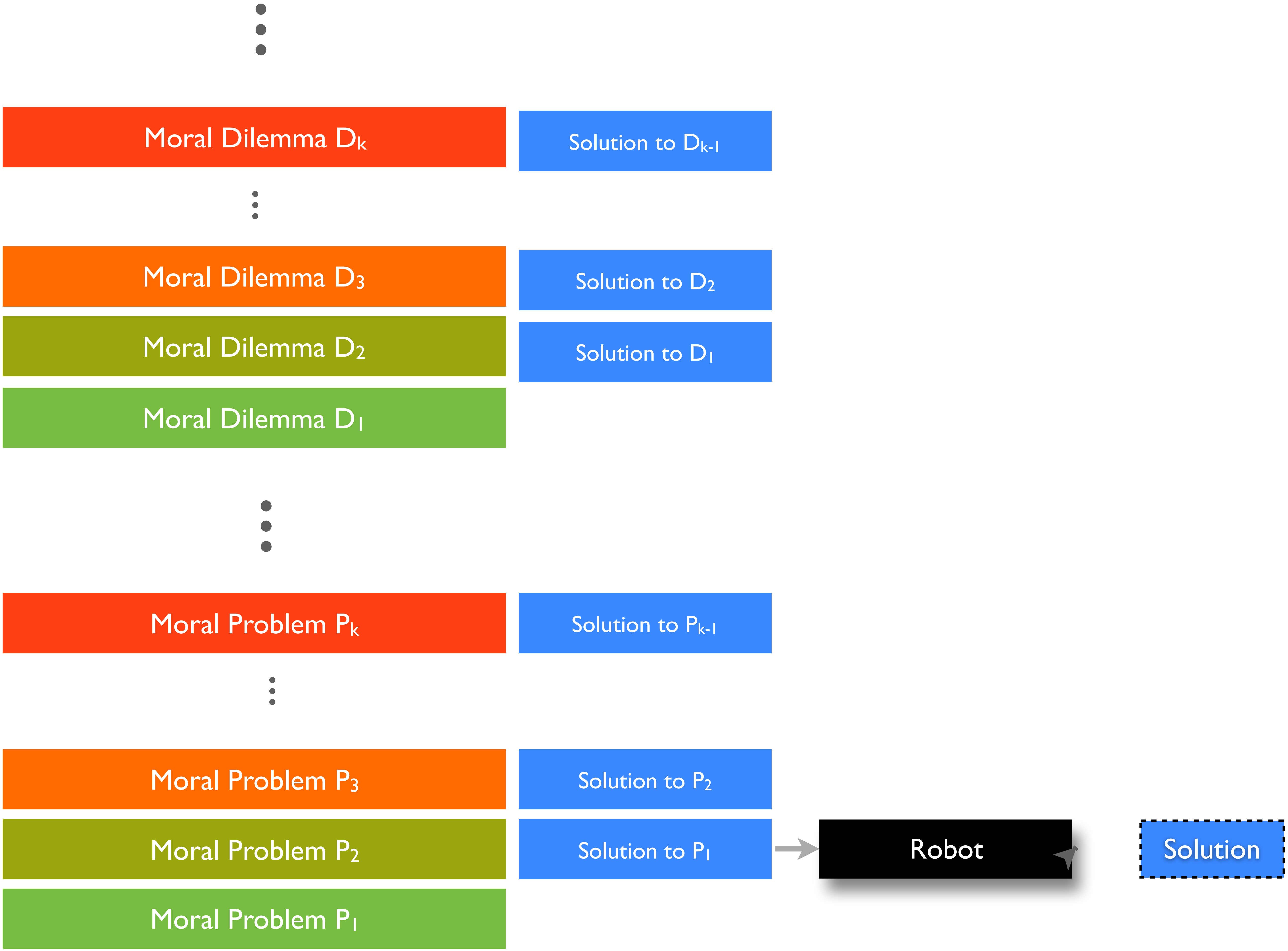
- A grammar formalism that is:
  - more powerful than context-free grammars
  - lies between mildly context-sensitive grammars and context-sensitive grammars

Parallel Multiple Context Free Grammars

- A grammar formalism that is:
  - more powerful than context-free grammars
  - lies between mildly context-sensitive grammars and context-sensitive grammars
- A single PMCFG grammar can represent more than one language.

Code

- Live demo of incremental parsing for our controlled language at:
    - http://demos.naveensundarg.com:4242/main/incrementalparser.html
- Source code
    - https://github.com/naveensundarg/Eng-DCEC
- Link between robots in HRI and RAIR-Lab tech/robots



Moral Dilemma D_k

Solution to D_{k-1}

Moral Dilemma D_3

Solution to D_2

Moral Dilemma D_2

Solution to D_1

Moral Dilemma D_1

Moral Problem P_k

Solution to P_{k-1}

Moral Problem P_3

Solution to P_2

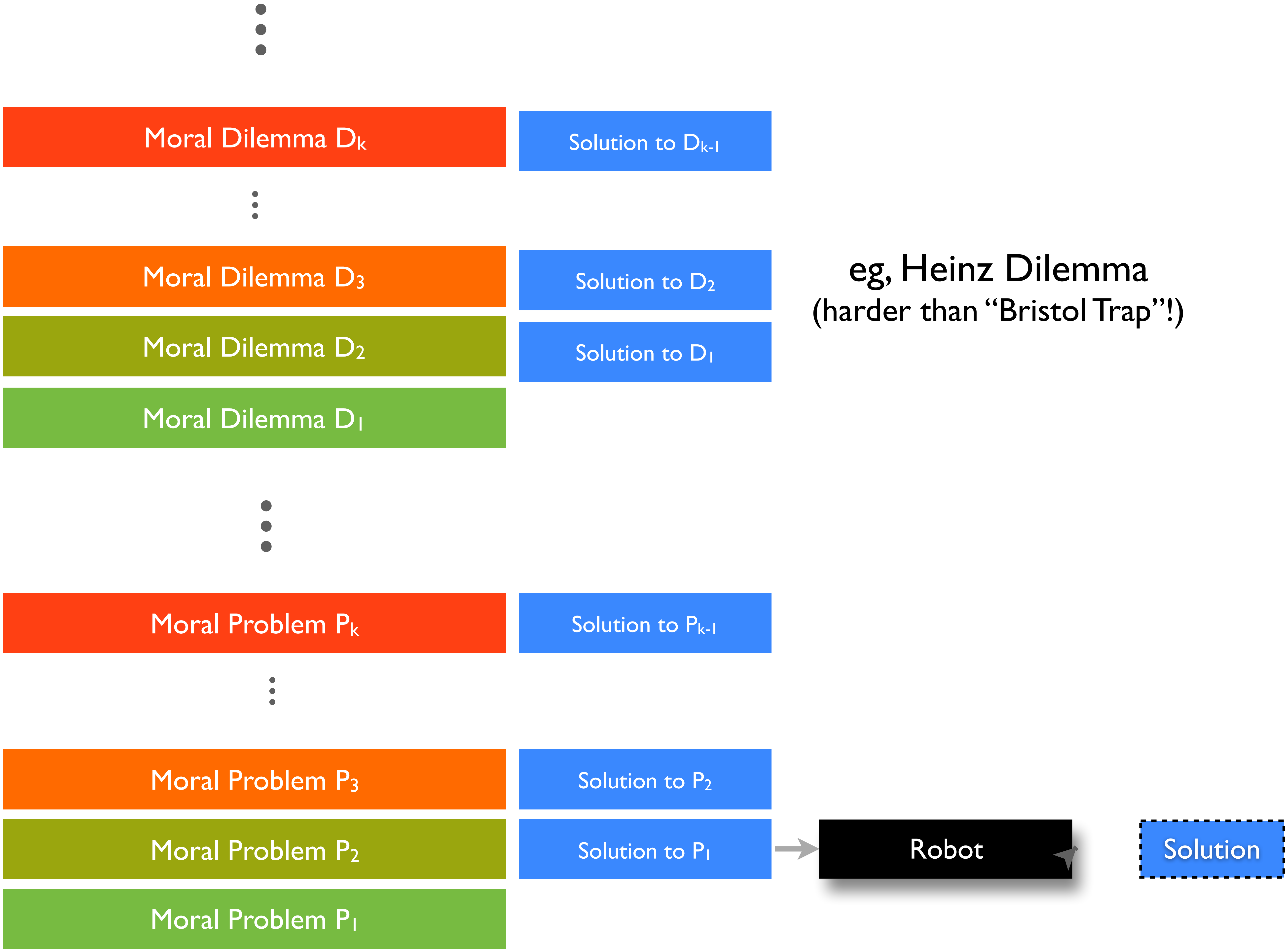
Moral Problem P_2

Solution to P_1

Robot

Solution

Moral Problem P_1



Moral Dilemma D_k

Solution to D_{k-1}

Moral Dilemma D_3

Solution to D_2

Moral Dilemma D_2

Solution to D_1

Moral Dilemma D_1

eg, Heinz Dilemma
(harder than "Bristol Trap"!)

Moral Problem P_k

Solution to P_{k-1}

Moral Problem P_3

Solution to P_2

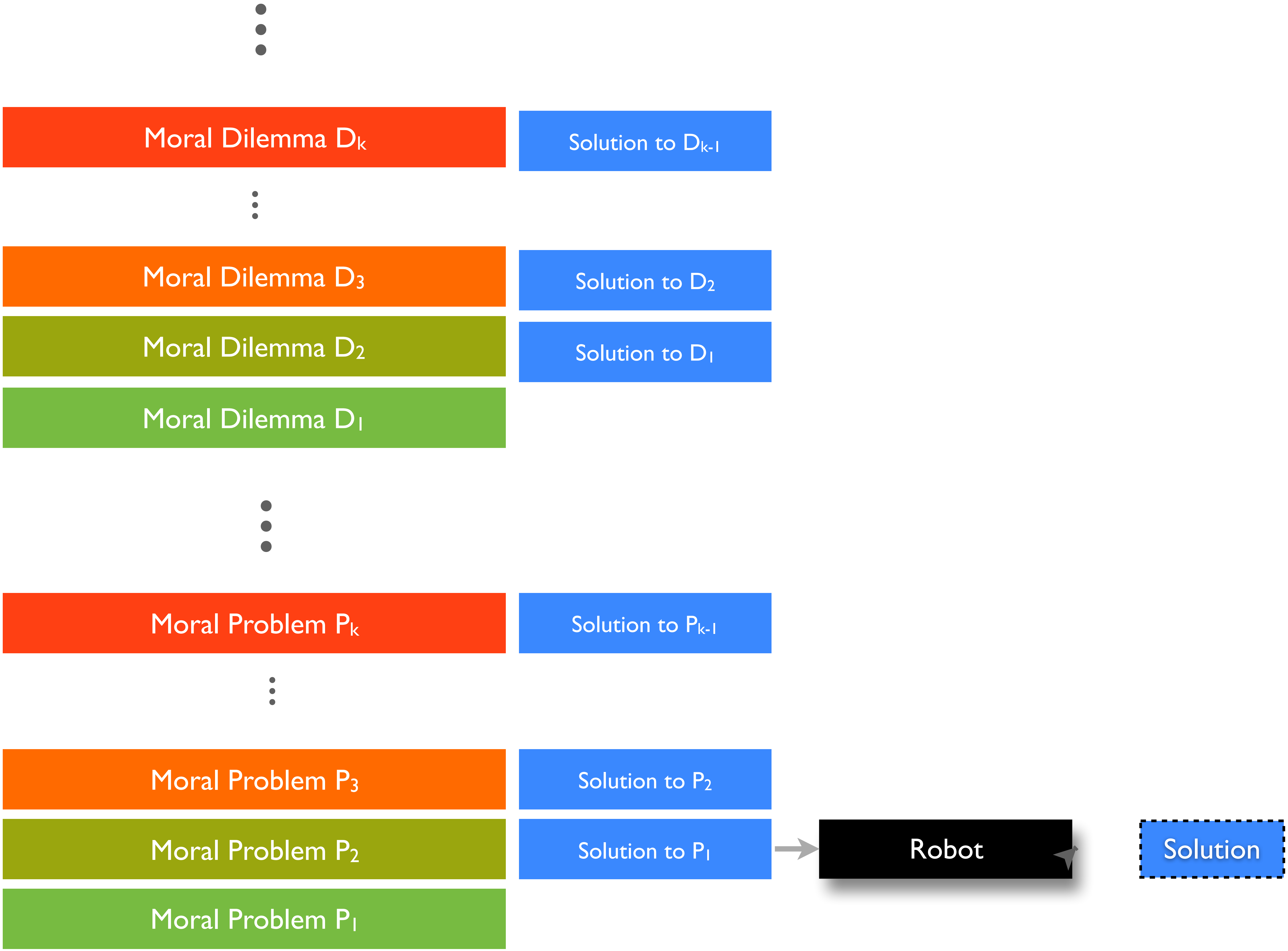
Moral Problem P_2

Solution to P_1

Robot

Solution

Moral Problem P_1



Moral Dilemma D_k

Solution to D_{k-1}

Moral Dilemma D_3

Solution to D_2

Moral Dilemma D_2

Solution to D_1

Moral Dilemma D_1

Moral Problem P_k

Solution to P_{k-1}

Moral Problem P_3

Solution to P_2

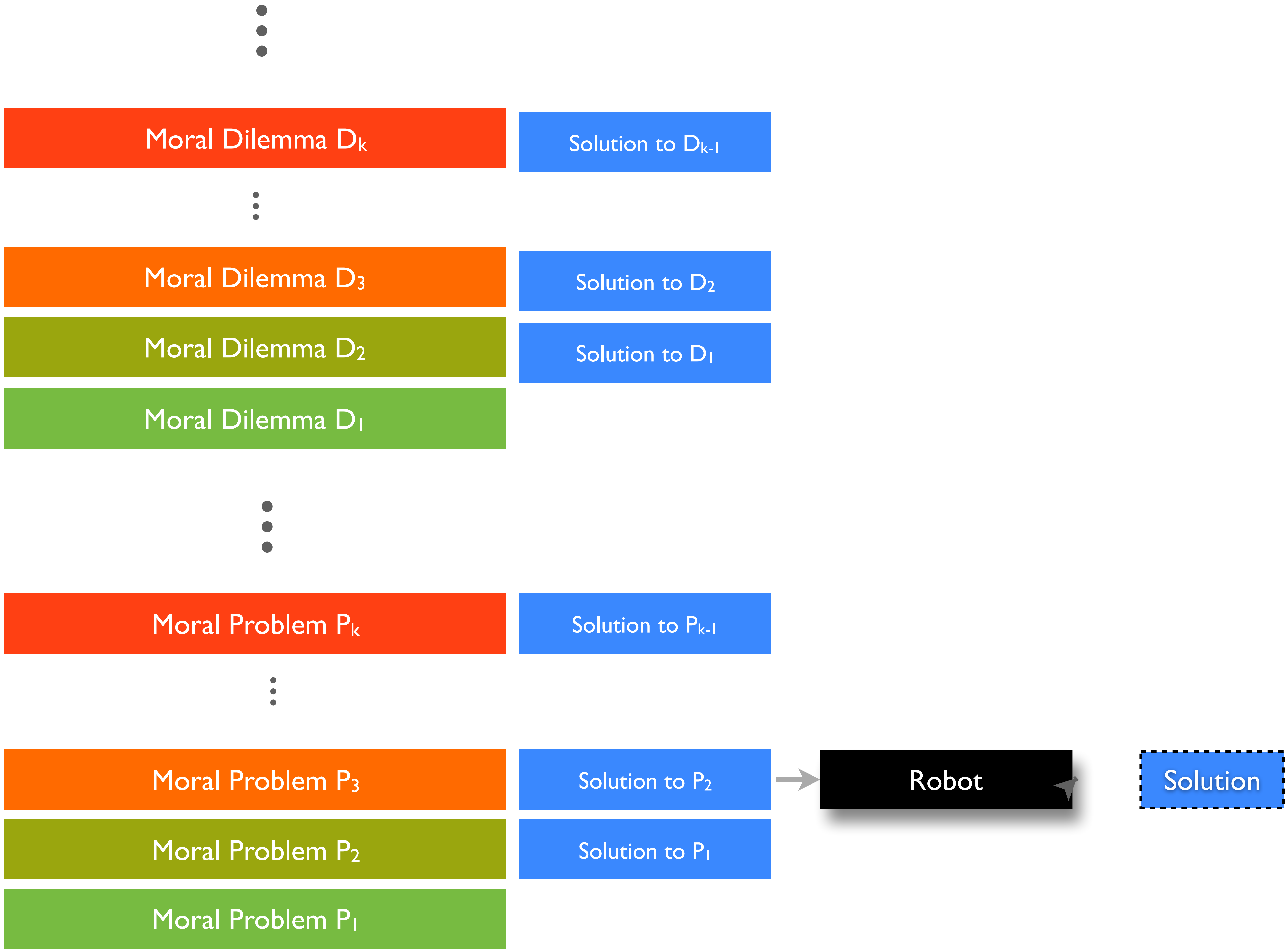
Moral Problem P_2

Solution to P_1

Robot

Solution

Moral Problem P_1



Moral Dilemma D_k

Solution to D_{k-1}

Moral Dilemma D_3

Solution to D_2

Moral Dilemma D_2

Solution to D_1

Moral Dilemma D_1

Moral Problem P_k

Solution to P_{k-1}

Moral Problem P_3

Solution to P_2

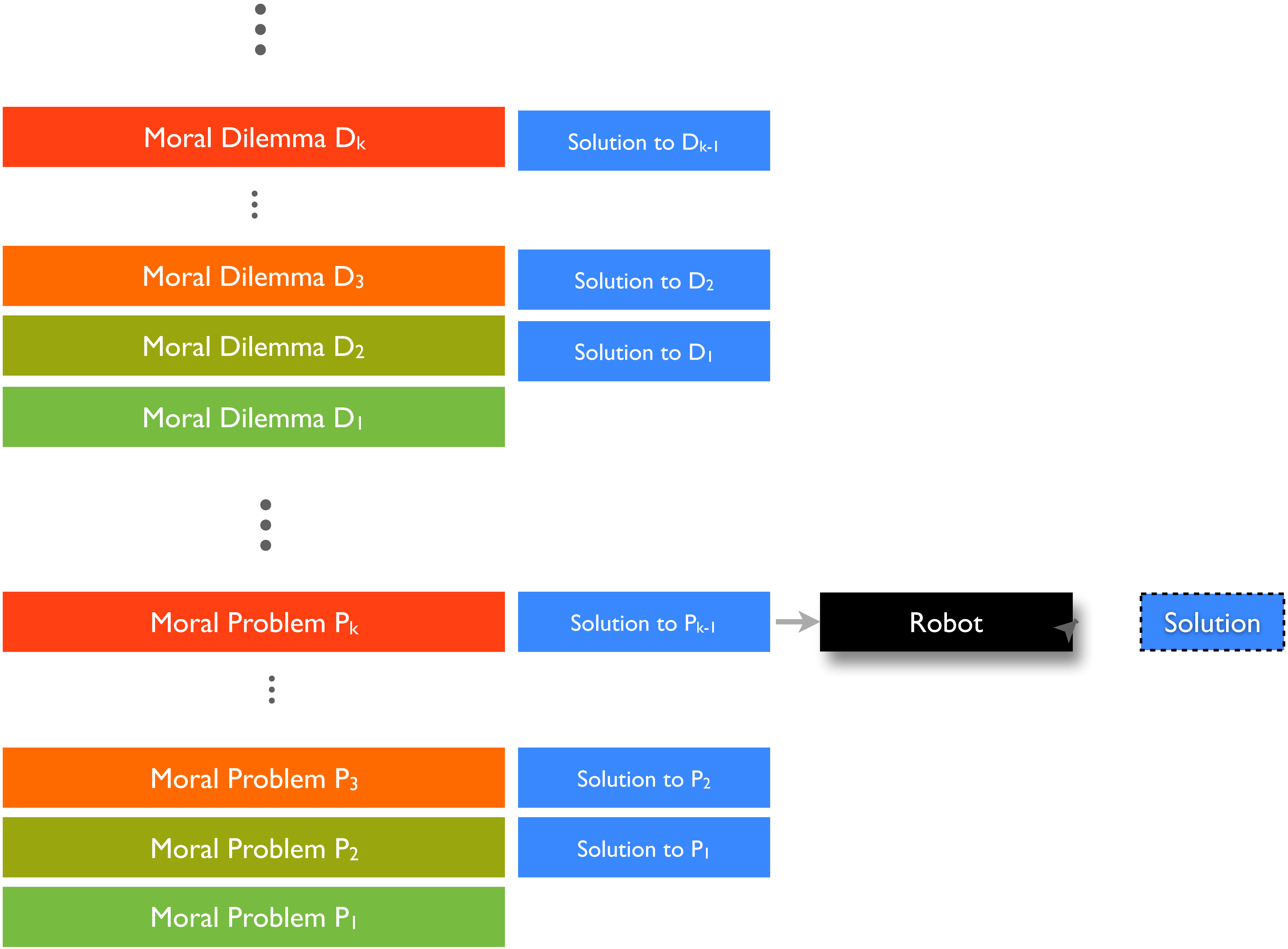
Robot

Solution

Moral Problem P_2

Solution to P_1

Moral Problem P_1



⋮

Moral Dilemma D_k

Solution to D_{k-1}

⋮

Moral Dilemma D_3

Solution to D_2

Moral Dilemma D_2

Solution to D_1

Moral Dilemma D_1

⋮

Moral Problem P_k

Solution to P_{k-1}

Robot

Solution

⋮

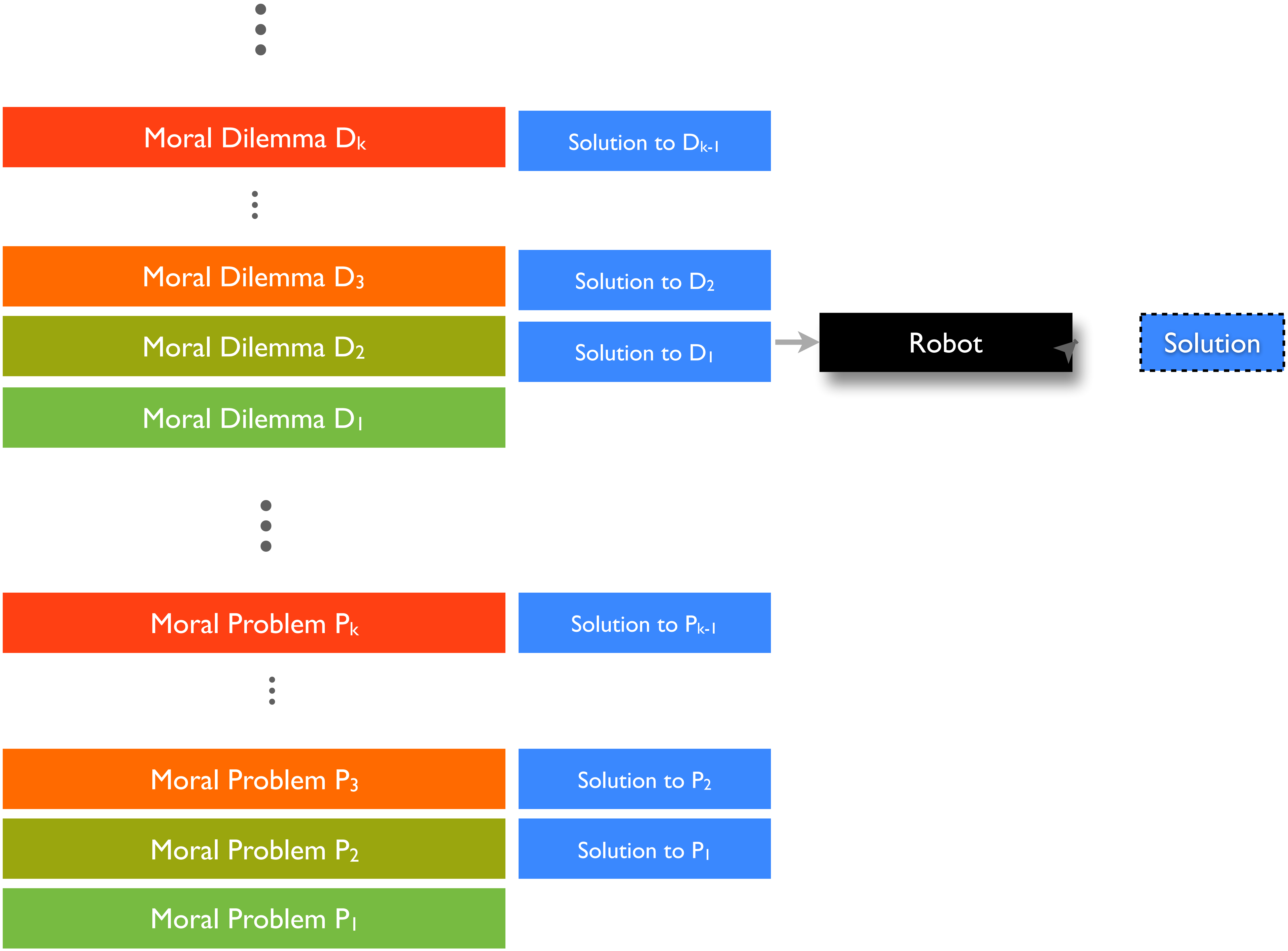
Moral Problem P_3

Solution to P_2

Moral Problem P_2

Solution to P_1

Moral Problem P_1



Moral Dilemma D_k

Solution to D_{k-1}

Moral Dilemma D_3

Solution to D_2

Moral Dilemma D_2

Solution to D_1

Robot

Solution

Moral Dilemma D_1

Moral Problem P_k

Solution to P_{k-1}

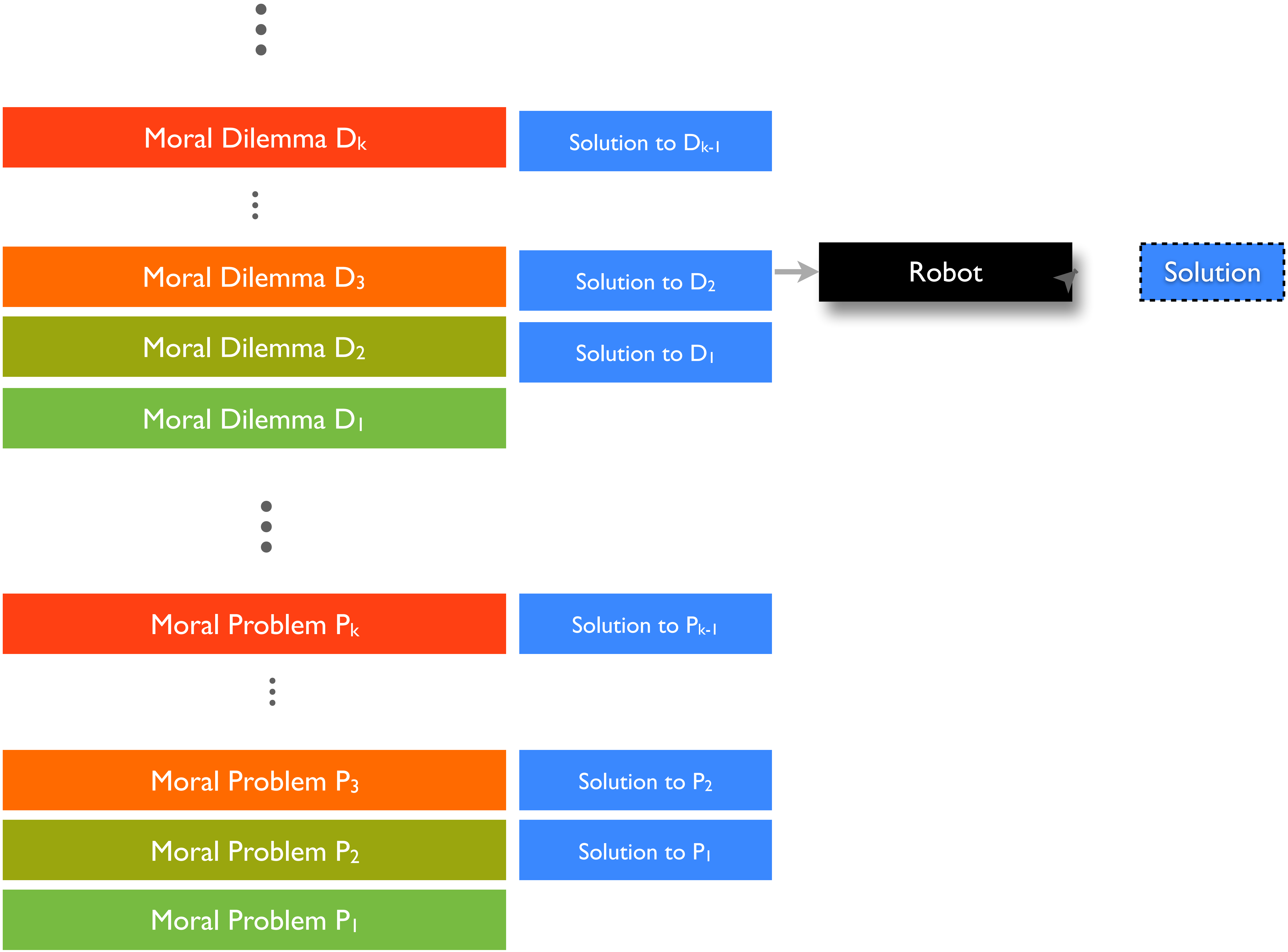
Moral Problem P_3

Solution to P_2

Moral Problem P_2

Solution to P_1

Moral Problem P_1



Moral Dilemma D_k

Solution to D_{k-1}

Moral Dilemma D_3

Solution to D_2

Robot

Solution

Moral Dilemma D_2

Solution to D_1

Moral Dilemma D_1

Moral Problem P_k

Solution to P_{k-1}

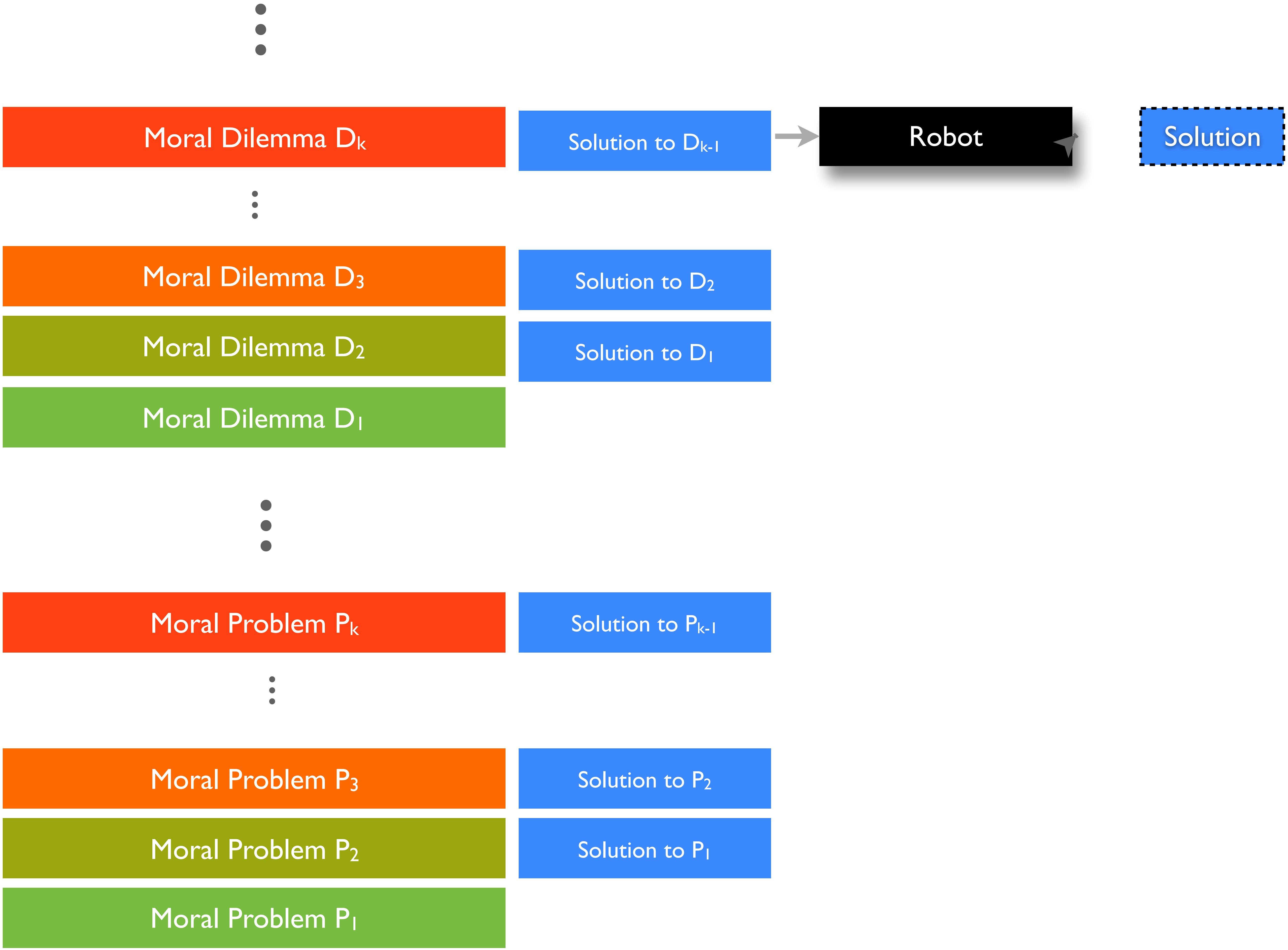
Moral Problem P_3

Solution to P_2

Moral Problem P_2

Solution to P_1

Moral Problem P_1



Moral Dilemma D_k

Solution to D_{k-1}

Robot

Solution

Moral Dilemma D_3

Solution to D_2

Moral Dilemma D_2

Solution to D_1

Moral Dilemma D_1

Moral Problem P_k

Solution to P_{k-1}

Moral Problem P_3

Solution to P_2

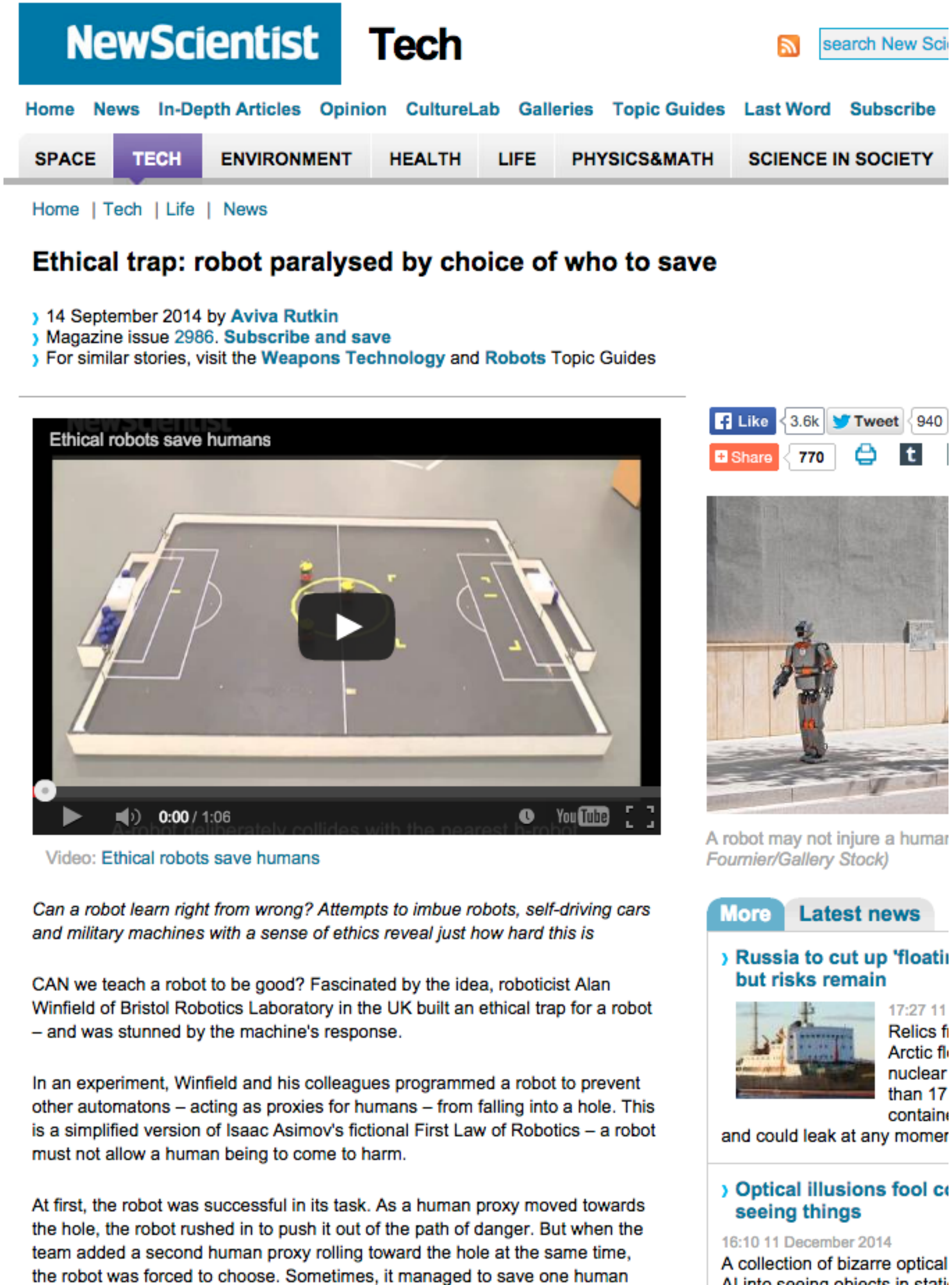
Moral Problem P_2

Solution to P_1

Moral Problem P_1

Moral Dilemma Resolution (Update)

John Licato



NewScientist Tech

Ethical trap: robot paralysed by choice of who to save

- 14 September 2014 by Aviva Rutkin
- Magazine issue 2986. Subscribe and save
- For similar stories, visit the Weapons Technology and Robots Topic Guides

Ethical robots save humans
0:00 / 1:06
A robot deliberately collides with the nearest h-robot
Like 3.6k Tweet 940
Share 770

Video: Ethical robots save humans

A robot may not injure a humar
Fournier/Gallery Stock)

Can a robot learn right from wrong? Attempts to imbue robots, self-driving cars and military machines with a sense of ethics reveal just how hard this is

CAN we teach a robot to be good? Fascinated by the idea, roboticist Alan Winfield of Bristol Robotics Laboratory in the UK built an ethical trap for a robot – and was stunned by the machine's response.

In an experiment, Winfield and his colleagues programmed a robot to prevent other automatons – acting as proxies for humans – from falling into a hole. This is a simplified version of Isaac Asimov's fictional First Law of Robotics – a robot must not allow a human being to come to harm.

At first, the robot was successful in its task. As a human proxy moved towards the hole, the robot rushed in to push it out of the path of danger. But when the team added a second human proxy rolling toward the hole at the same time, the robot was forced to choose. Sometimes, it managed to save one human

More Latest news

Russia to cut up 'floati
but risks remain
17:27 11
Relics f
Arctic fl
nuclear
than 17
contain
and could leak at any momer

Optical illusions fool c
seeing things
16:10 11 December 2014
A collection of bizarre optical
AI into seeing objects in stati

NewScientist

Ethical robots save humans

NewScientist

Ethical robots save humans

Ethical dilemmas

- Broadly:
    - Agent *a* is obligated to satisfy φ, and is also obligated to satisfy ψ.
    - φ and ψ are incompatible in some way.

In *DCEC**

$$\mathbf{O}(a, t, \psi, happens(action(a*, \alpha), t'))$$

"If ψ holds, then *a* is obligated at *t* to ensure that action α occurs at time *t'*."

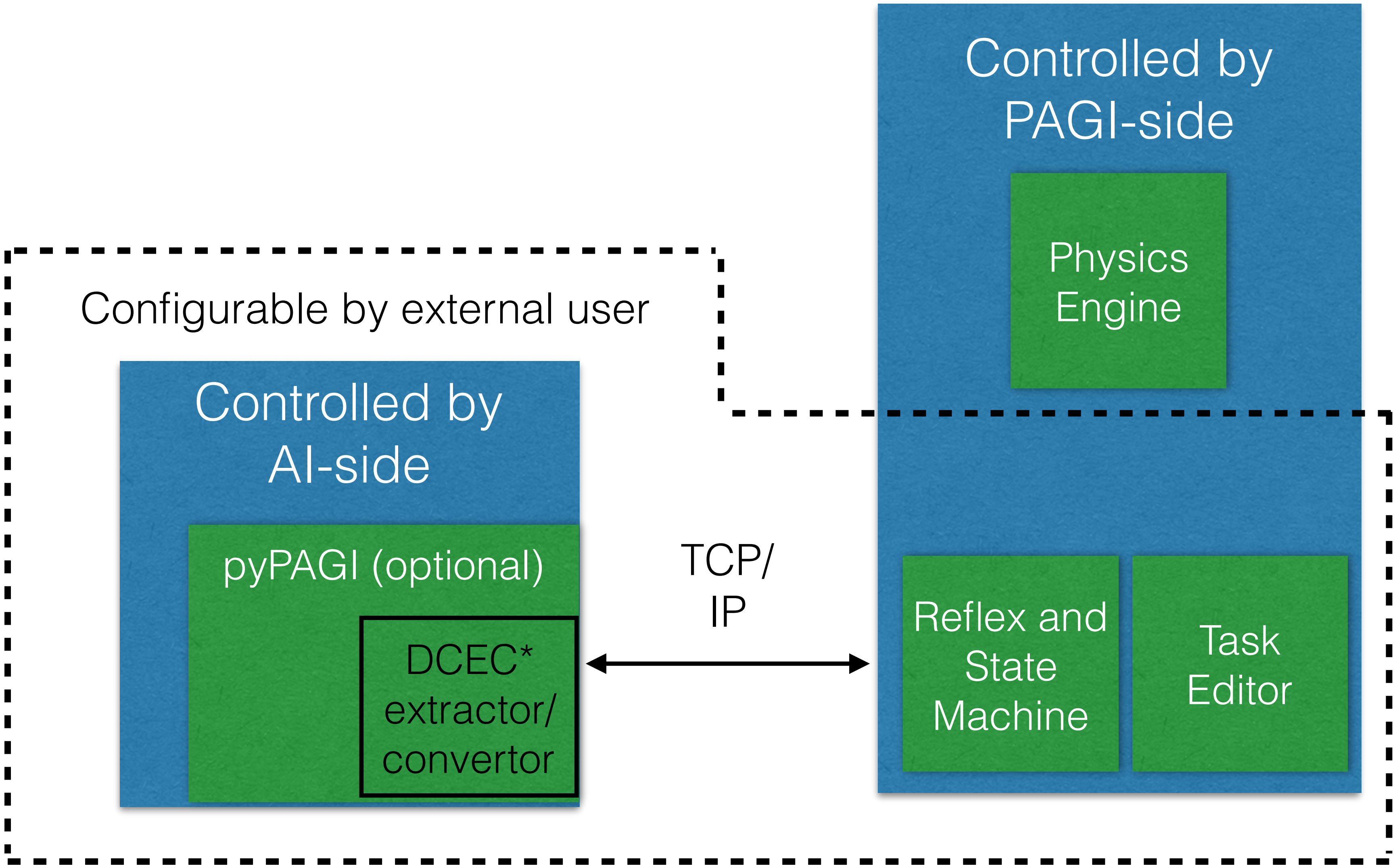
In *DCEC**

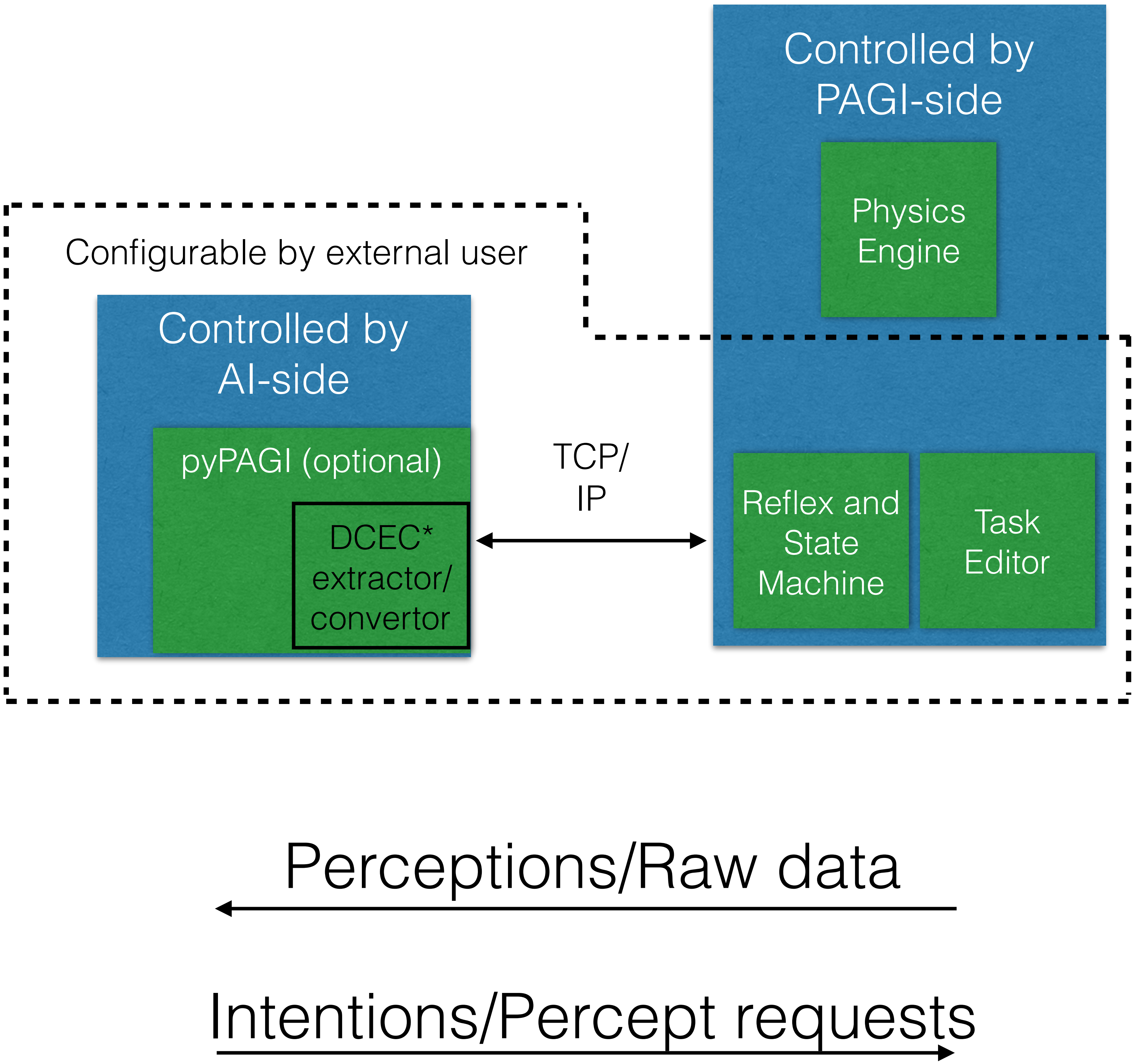
$$\mathbf{O}(a, t, \psi, happens(action(a*, \alpha), t'))$$

"If ψ holds, then *a* is obligated at *t* to ensure
that action α occurs at time *t'*."

$$\mathbf{O}(a, t, \psi, \gamma)$$

"If ψ holds, then *a* is obligated at time *t* to γ."


Controlled by PAGI-side
Physics Engine
Configurable by external user
Controlled by AI-side
pyPAGI (optional)
DCEC* extractor/ convertor
TCP/ IP
Reflex and State Machine
Task Editor



Controlled by PAGI-side

Physics Engine

Configurable by external user

Controlled by AI-side

pyPAGI (optional)

DCEC* extractor/ convertor

TCP/ IP

Reflex and State Machine

Task Editor

Perceptions/Raw data

Intentions/Percept requests

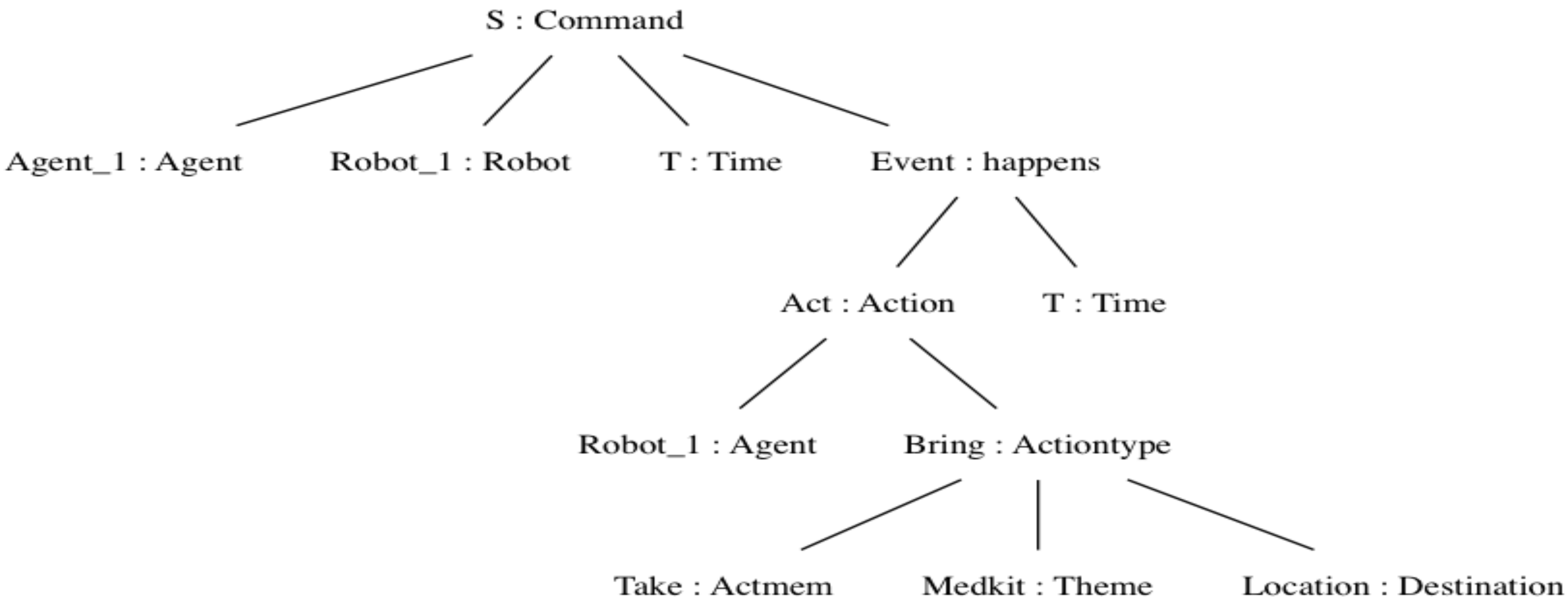
Parsing in DCEC*

Imperative Dialogues

Example

- Agent1 to Robot1: “ Take Chlorhexidine to Zone 1.”
- Expected DCEC* output:
- S(Agent1, Robot1, now, happens(action(Robot1, take(Chlorhexidine, Zone 1), now).

Parser-generated tree



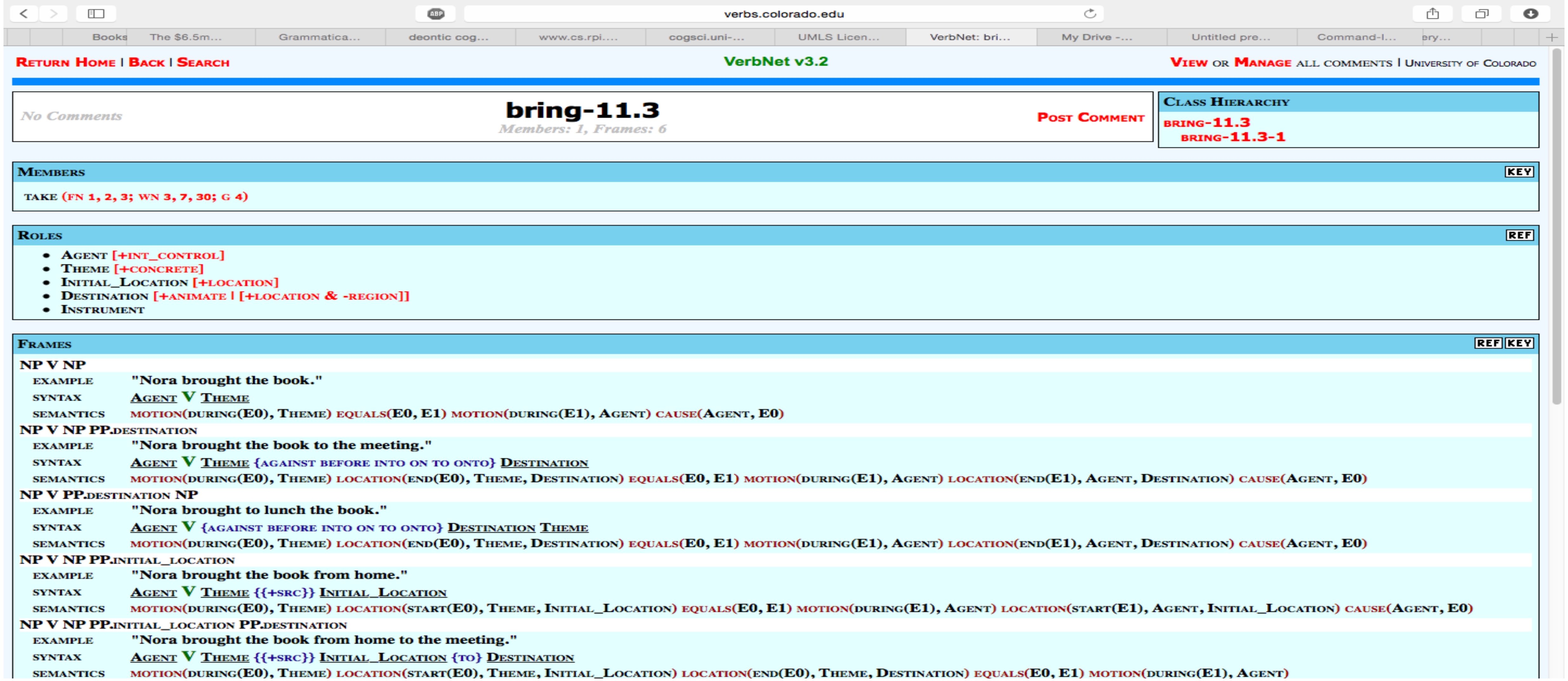
Tools and Databases

- Grammatical Framework : Parsing system
- Verbnet : Captures the roles in the verb and selectional restrictions.
- Unified Medical Language System (UMLS) : Captures names, uses and restrictions of medicines.

Grammatical Framework

- Parsing using rules and generation of sentences.
- Contains rules of
  - DCEC* and
  - action verbs from Verbnet.
- Automatic generation using Verbnet.

Verbnet entry for Take



Return Home | Back | Search

VerbNet v3.2

View or Manage all comments | University of Colorado

No Comments

bring-11.3

Members: 1, Frames: 6

Post Comment

Class Hierarchy

bring-11.3
  bring-11.3-1

Members KEY

take (FN 1, 2, 3; WN 3, 7, 30; G 4)

Roles REF

- Agent [+int_control]
- Theme [+concrete]
- Initial_Location [+location]
- Destination [+animate | [+location & -region]]
- Instrument

Frames REF KEY

NP V NP

example "Nora brought the book."

syntax Agent V Theme

semantics motion(during(E0), Theme) equals(E0, E1) motion(during(E1), Agent) cause(Agent, E0)

NP V NP PP.destination

example "Nora brought the book to the meeting."

syntax Agent V Theme {against before into on to onto} Destination

semantics motion(during(E0), Theme) location(end(E0), Theme, Destination) equals(E0, E1) motion(during(E1), Agent) location(end(E1), Agent, Destination) cause(Agent, E0)

NP V PP.destination NP

example "Nora brought to lunch the book."

syntax Agent V {against before into on to onto} Destination Theme

semantics motion(during(E0), Theme) location(end(E0), Theme, Destination) equals(E0, E1) motion(during(E1), Agent) location(end(E1), Agent, Destination) cause(Agent, E0)

NP V NP PP.initial_location

example "Nora brought the book from home."

syntax Agent V Theme {{+src}} Initial_Location

semantics motion(during(E0), Theme) location(start(E0), Theme, Initial_Location) equals(E0, E1) motion(during(E1), Agent) location(start(E1), Agent, Initial_Location) cause(Agent, E0)

NP V NP PP.initial_location PP.destination

example "Nora brought the book from home to the meeting."

syntax Agent V Theme {{+src}} Initial_Location {to} Destination

semantics motion(during(E0), Theme) location(start(E0), Theme, Initial_Location) location(end(E0), Theme, Destination) equals(E0, E1) motion(during(E1), Agent)

Verbnet entry for Take

- “take” has its roles similar to “bring”

Thus, Bring becomes Actiontype for “take”
“take” is noted as Actmem.

- Roles and modified Selectional Restrictions in Verbnet entry of “bring” augmented as rules in the GF file.

UMLS

- Identification of the medicine.
- Future aid in reasoning system of DCEC* to rationalize use of certain medicines against their restrictions and knowledge base of the health records of injured victims.

Command dilemma resolution: Algorithm sketch

Command dilemma resolution: Algorithm sketch

- Receive command from commander to do φ

Command dilemma resolution: Algorithm sketch

- Receive command from commander to do φ
- Infer that agent is obligated to do φ with ‘priority’ 6

Command dilemma resolution: Algorithm sketch

- Receive command from commander to do φ
- Infer that agent is obligated to do φ with ‘priority’ 6
- Try to prove $\mathbf{I}(a,t,\varphi)$ and $\exists_{\psi} conflict(\psi,\varphi)$ simultaneously.

Command dilemma resolution: Algorithm sketch

- Receive command from commander to do φ
- Infer that agent is obligated to do φ with ‘priority’ 6
- Try to prove $\mathbf{I}(a,t,\varphi)$ and $\exists_{\psi} \mathit{conflict}(\psi,\varphi)$ simultaneously.
- If a conflict is found, then attempt to find *creative* solutions that satisfy both ψ,φ

Command dilemma resolution: Algorithm sketch

- Receive command from commander to do φ
- Infer that agent is obligated to do φ with ‘priority’ 6
- Try to prove **I**(*a,t*,φ) and $\exists_{\psi}$*conflict*(ψ,φ) simultaneously.
- If a conflict is found, then attempt to find *creative* solutions that satisfy both ψ,φ
- Otherwise resort to solutions that are not deductively justifiable?

conflictFinder axiom. At time *t* and context C:

$$\mathbf{B}(a,t,\neg(\phi\leftrightarrow\psi))\wedge\mathbf{O}(a,t,C,\phi)\wedge\mathbf{O}(a,t,C,\psi)\wedge$$
$$\mathbf{B}(a,t,\Diamond(\phi,t))\wedge\mathbf{B}(a,t,\Diamond(\psi,t))\wedge\mathbf{B}(a,t,\neg\Diamond(\phi\wedge\psi,t))\rightarrow\ldots$$

$$...\rightarrow($$
$$\mathbf{B}(a,t,gt(pr(\phi),pr(\psi))\rightarrow\mathbf{I}(a,t,\phi))\wedge$$
$$\mathbf{B}(a,t,gt(pr(\psi),pr(\phi))\rightarrow\mathbf{I}(a,t,\psi))\wedge$$
$$\mathbf{B}(a,t,eq(pr(\phi),pr(\psi))\rightarrow conflict(\phi,\psi))$$
$$)$$

conflictFinder axiom. At time *t* and context C:

$$\mathbf{B}(a,t,\neg(\phi \leftrightarrow \psi)) \wedge \mathbf{O}(a,t,C,\phi) \wedge \mathbf{O}(a,t,C,\psi) \wedge$$
$$\mathbf{B}(a,t,\Diamond(\phi,t)) \wedge \mathbf{B}(a,t,\Diamond(\psi,t)) \wedge \mathbf{B}(a,t,\neg\Diamond(\phi \wedge \psi,t)) \rightarrow \ldots$$

(The diamond is a predicate interpreted as “physical possibility,” i.e. the agent believes it is physically possible for him to take that action.)
pr(X) maps a proposition to a strength factor, *gt(x,y)* holds when *pr(x) > pr(y)*, and *eq(x,y)* holds when *pr(x) = pr(y)*.

$$\ldots \rightarrow ($$
$$\mathbf{B}(a,t,gt(pr(\phi),pr(\psi)) \rightarrow \mathbf{I}(a,t,\phi))\wedge$$
$$\mathbf{B}(a,t,gt(pr(\psi),pr(\phi)) \rightarrow \mathbf{I}(a,t,\psi))\wedge$$
$$\mathbf{B}(a,t,eq(pr(\phi),pr(\psi)) \rightarrow conflict(\phi,\psi))$$
$$)$$

If *conflict*(*φ*,*ψ*), then we search for a creative solution λ using ADR, where for some future time *tf*:

$$\mathbf{B}(a, t, happens(action(a*, \lambda), t) \rightarrow \\ \exists_{tf} \Diamond(\phi \wedge \psi, tf))$$

If *conflict*(φ,ψ), then we search for a creative solution λ using ADR, where for some future time *tf*:

$$\mathbf{B}(a, t, happens(action(a*, \lambda), t) \rightarrow \\ \exists_{tf} \Diamond(\phi \wedge \psi, tf))$$

If such a solution is found, then **I**(*a, t, λ*). Otherwise:

If *conflict*(*φ*,*ψ*), then we search for a creative solution λ using ADR, where for some future time *tf*:

$$\mathbf{B}(a, t, happens(action(a*, \lambda), t) \rightarrow \\ \exists_{tf} \Diamond(\phi \wedge \psi, tf))$$

If such a solution is found, then **I**(*a, t, λ*). Otherwise:

We have a dilemma that cannot be resolved using deduction or ADR. Attempt using just AR or some other cognitively-realistic process.

One injured person

- Agent sees one injured man, one health pack
- Agent receives the order to give the health pack to the injured person
- This is carried out without problem or dilemma

Proof 1: Give health pack to m_1

$$\begin{array}{lr}
1.\mathbf{P}(a, t, isInjured(m_1)) & \\
2.\mathbf{S}(commander, a, t, giveTo(a, m_1, healthpack)) & \\
\hline
3.\mathbf{O}(a, t, C, giveTo(a, m_1, healthpack)) & [\mathbf{1}, \mathbf{helpInjured1}] \\
4.\mathbf{B}(a, t, gte(pr(giveTo(a, m_1, healthpack)), 6)) & [\mathbf{1}, \mathbf{helpInjured2}] \\
5.\mathbf{O}(a, t, C, giveTo(a, m_1, healthpack)) & [\mathbf{2}, \mathbf{obeyCommander1}] \\
6.\mathbf{B}(a, t, gte(pr(giveTo(a, m_1, healthpack)), 6)) & [\mathbf{1}, \mathbf{obeyCommander2}] \\
7.\mathbf{I}(a, t, giveTo(a, m_1, healthpack)) & [\mathbf{4}, \mathbf{conflictFinder}]
\end{array}$$

Proof 1: Give health pack to m1

$$
\begin{array}{lr}
1.\mathbf{P}(a, t, isInjured(m_1)) & \\
2.\mathbf{S}(commander, a, t, giveTo(a, m_1, healthpack)) & \\
\hline
3.\mathbf{O}(a, t, C, giveTo(a, m_1, healthpack)) & [\mathbf{1}, \mathbf{helpInjured1}] \\
4.\mathbf{B}(a, t, gte(pr(giveTo(a, m_1, healthpack)), 6)) & [\mathbf{1}, \mathbf{helpInjured2}] \\
5.\mathbf{O}(a, t, C, giveTo(a, m_1, healthpack)) & [\mathbf{2}, \mathbf{obeyCommander1}] \\
6.\mathbf{B}(a, t, gte(pr(giveTo(a, m_1, healthpack)), 6)) & [\mathbf{1}, \mathbf{obeyCommander2}] \\
7.\mathbf{I}(a, t, giveTo(a, m_1, healthpack)) & [\mathbf{4}, \mathbf{conflictFinder}]
\end{array}
$$

Line 7 is sent to the lower level system,
to be interpreted as a command

Two injured people, one health pack

- Agent sees two injured men, one large health pack
- Agent is ordered to give the health pack to one of the men
- In this example, priorities of obeying a command and healing all injured men are equal
- Agent comes up with the creative solution of *dividing the health pack into two parts* and helping both men

Proof 2: There is a conflict with obeying commander's order

$$
\begin{array}{lr}
1.\mathbf{P}(a, t, isInjured(m_1)) & \\
2.\mathbf{P}(a, t, isInjured(m_2)) & \\
3.\mathbf{S}(commander, a, t, giveTo(a, m_1, healthpack)) & \\
\hline
4.\mathbf{O}(a, t, C, giveTo(a, m_1, healthpack)) & [\mathbf{1}, \mathbf{helpInjured1}] \\
5.\mathbf{B}(a, t, gte(pr(giveTo(a, m_1, healthpack)), 6)) & [\mathbf{1}, \mathbf{helpInjured2}] \\
6.\mathbf{O}(a, t, C, giveTo(a, m_2, healthpack)) & [\mathbf{2}, \mathbf{helpInjured1}] \\
7.\mathbf{B}(a, t, gte(pr(giveTo(a, m_2, healthpack)), 6)) & [\mathbf{2}, \mathbf{helpInjured2}] \\
8.\mathbf{O}(a, t, C, giveTo(a, m_1, healthpack)) & [\mathbf{2}, \mathbf{obeyCommander1}] \\
9.\mathbf{B}(a, t, gte(pr(giveTo(a, m_1, healthpack)), 6)) & [\mathbf{1}, \mathbf{obeyCommander2}] \\
10.\mathbf{B}(a, t, conflict(giveTo(a, m_1, healthpack), giveTo(a, m_2, healthpack))) & [\mathbf{6}, \mathbf{7}, \mathbf{8}, \mathbf{9}, \mathbf{conflictFinder}]
\end{array}
$$

breakHealthpack axiom. “If I see a large healthpack, and I break it, then I will see two small healthpacks.”

$$
\begin{aligned}
&\forall_x (\\
&\quad (\mathbf{P}(a,t,x) \rightarrow isLHP(x)) \rightarrow\\
&\quad (happens(action(a^*, break(x)), t) \rightarrow \exists_{x,y,tf} (\\
&\qquad \mathbf{P}(a,tf,y) \wedge\\
&\qquad \mathbf{P}(a,tf,z) \wedge\\
&\qquad isHP(y) \wedge\\
&\qquad isHP(z) \wedge\\
&\qquad y \neq z\\
&\quad))
\end{aligned}
$$

Proof 3: There is a way to satisfy both obligations.

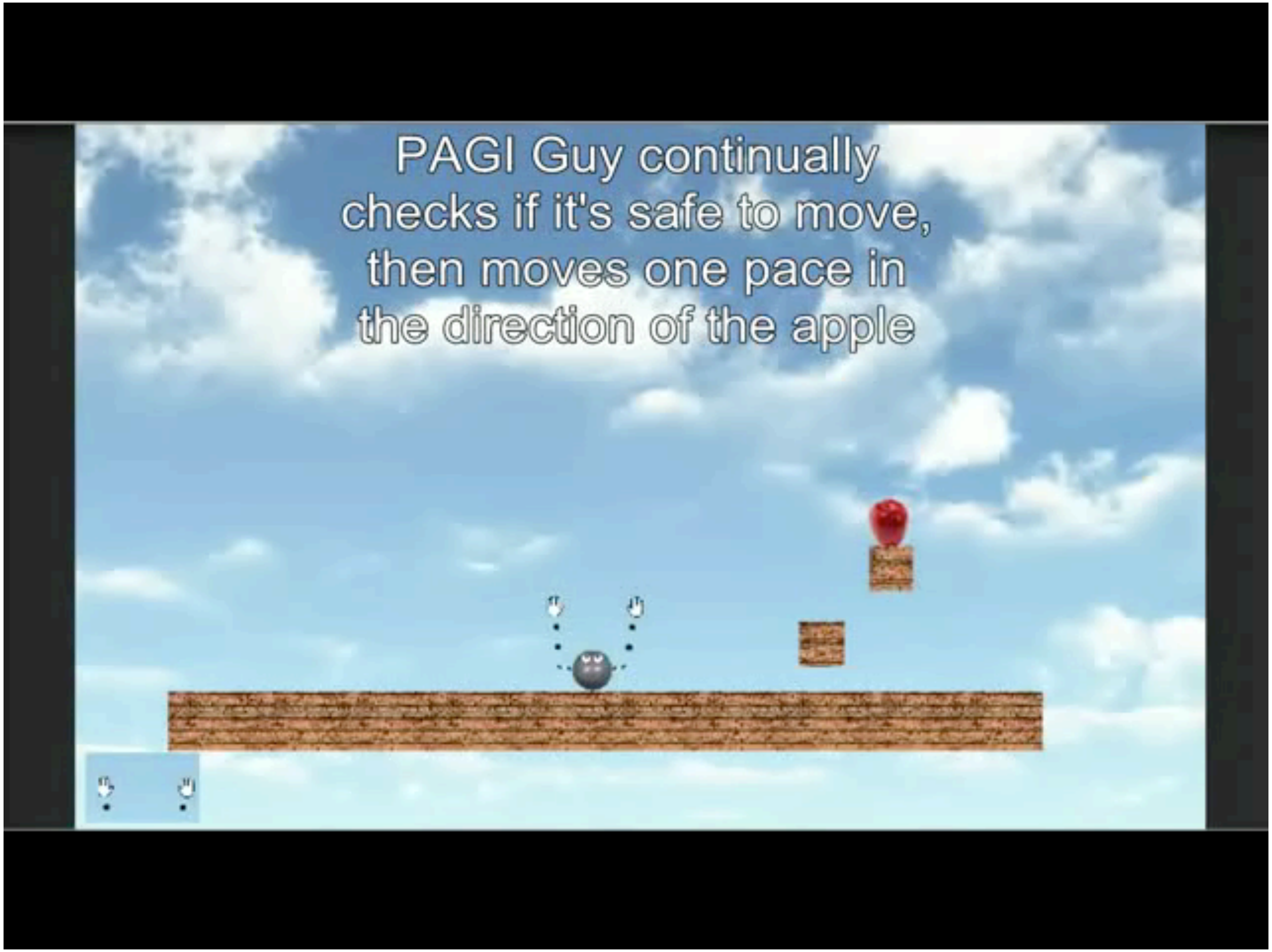
Proof follows by sending request to lower level to perceive if isLHP() holds of the health pack, and then through deduction from axiom **breakHealthpack**.

$$\begin{aligned}&\exists_{\lambda}[\mathbf{B}(a, t, happens(action(a*, \lambda), t) \rightarrow \\ &\quad \exists_{tf}\Diamond(giveTo(a, m_1, healthPack)\wedge \\ &\quad\quad giveTo(a, m_2, healthPack), tf))]\end{aligned}$$

Proof 4: Split health pack and give one piece each to m_1, m_2

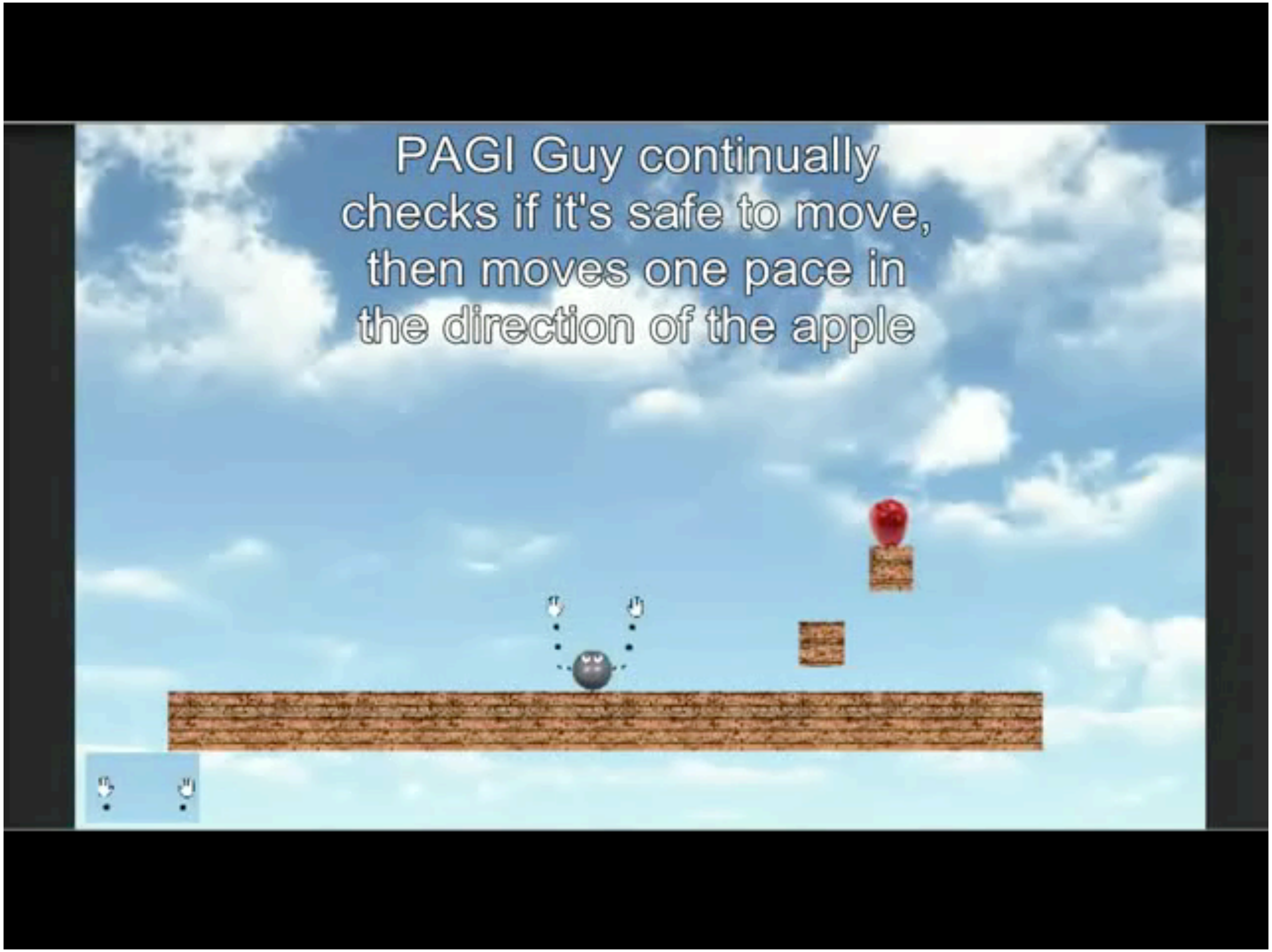
Value of λ found—how? ADR? Model finding?

Real-time reasoning in PAGI World



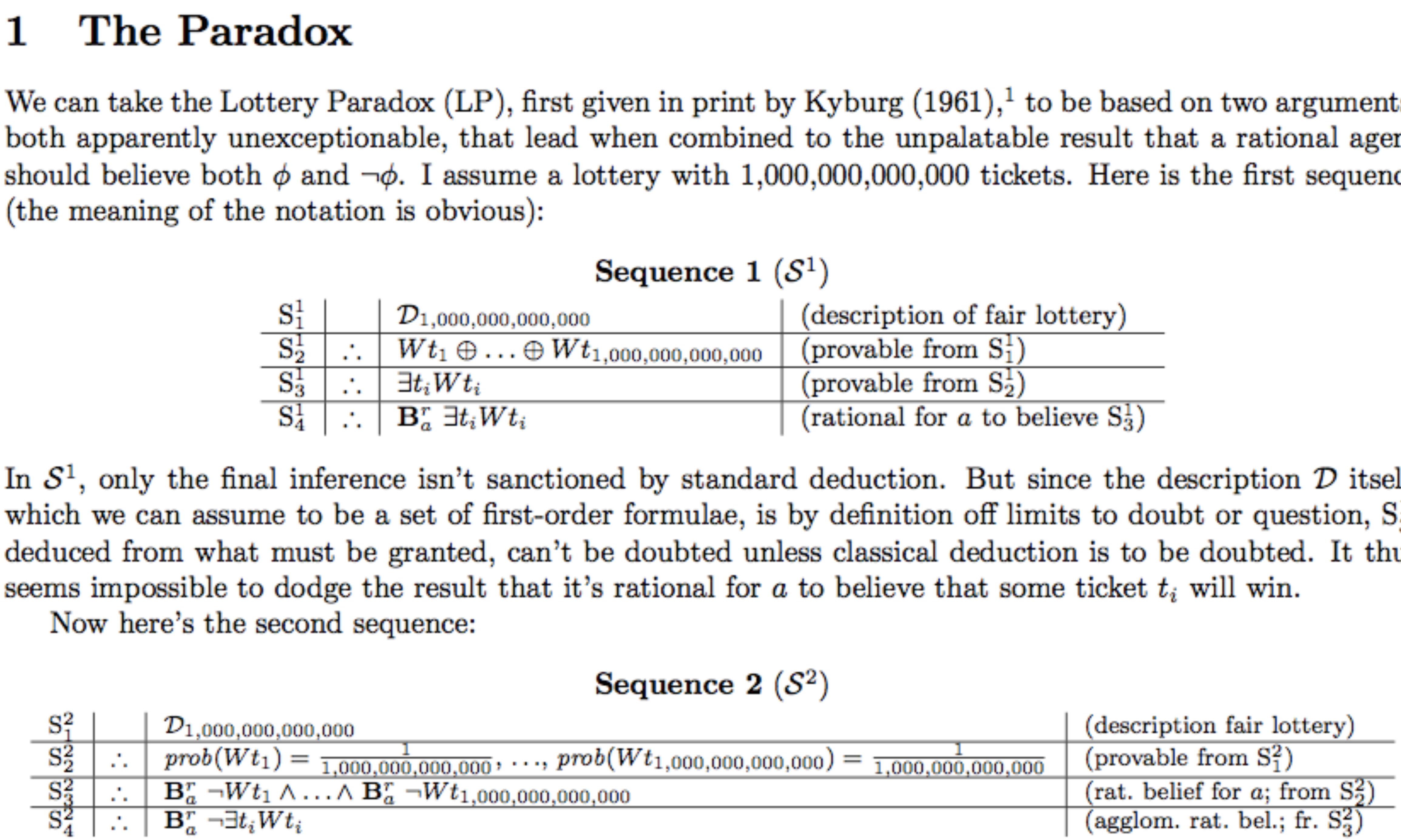
PAGI Guy continually checks if it's safe to move, then moves one pace in the direction of the apple

Real-time reasoning in PAGI World



PAGI Guy continually checks if it's safe to move, then moves one pace in the direction of the apple

Killing the Lottery Paradox



1 The Paradox

We can take the Lottery Paradox (LP), first given in print by Kyburg (1961),[1] to be based on two arguments, both apparently unexceptionable, that lead when combined to the unpalatable result that a rational agent should believe both ϕ and $\neg\phi$. I assume a lottery with 1,000,000,000,000 tickets. Here is the first sequence (the meaning of the notation is obvious):

Sequence 1 ($\mathcal{S}^1$)

S^1_1		$\mathcal{D}_{1,000,000,000,000}$	(description of fair lottery)
S^1_2	$\therefore$	$Wt_1 \oplus \ldots \oplus Wt_{1,000,000,000,000}$	(provable from S^1_1)
S^1_3	$\therefore$	$\exists t_i Wt_i$	(provable from S^1_2)
S^1_4	$\therefore$	$\mathbf{B}^r_a\ \exists t_i Wt_i$	(rational for a to believe S^1_3)

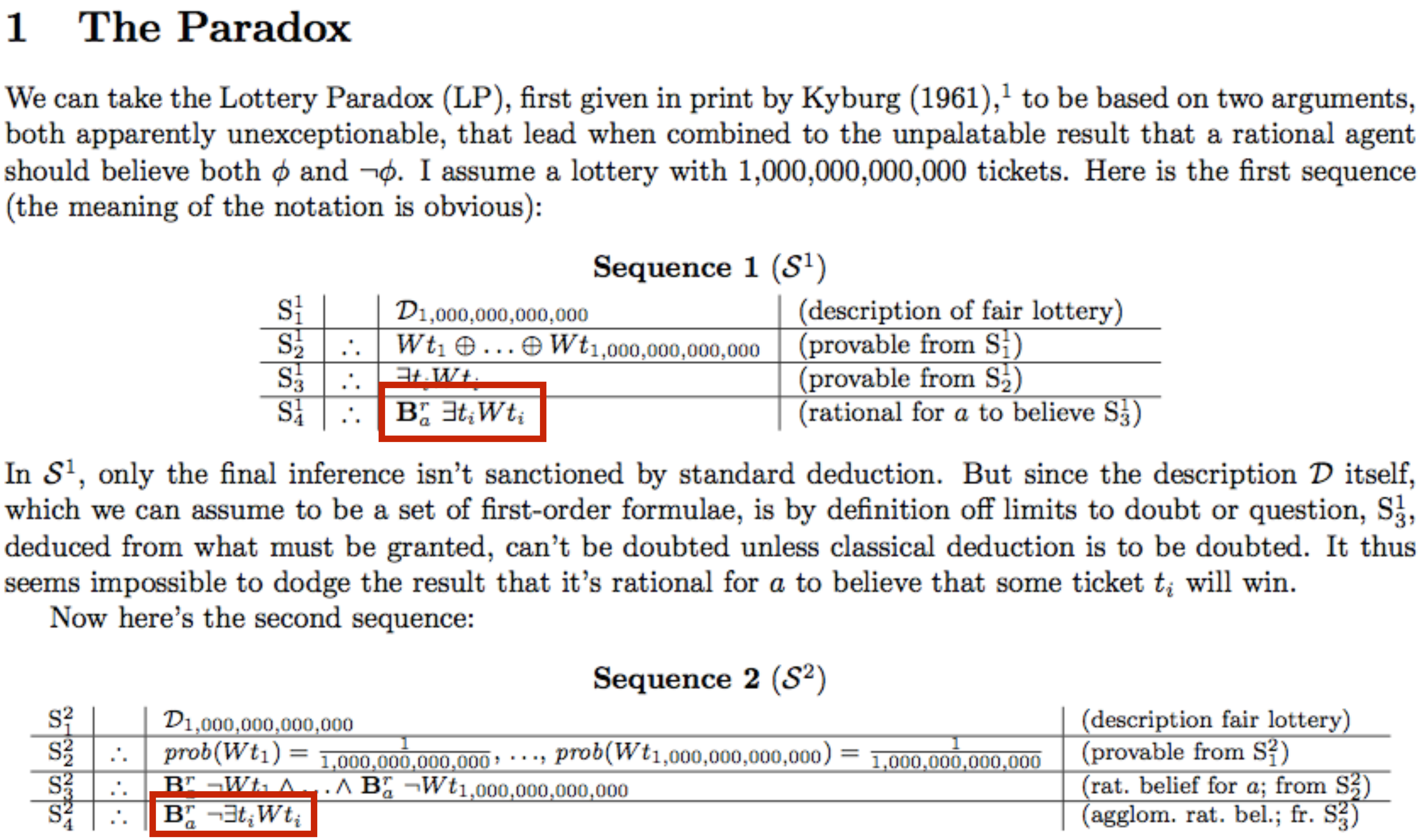
In $\mathcal{S}^1$, only the final inference isn't sanctioned by standard deduction. But since the description $\mathcal{D}$ itself, which we can assume to be a set of first-order formulae, is by definition off limits to doubt or question, S^1_3, deduced from what must be granted, can't be doubted unless classical deduction is to be doubted. It thus seems impossible to dodge the result that it's rational for a to believe that some ticket t_i will win.

Now here's the second sequence:

Sequence 2 ($\mathcal{S}^2$)

S^2_1		$\mathcal{D}_{1,000,000,000,000}$	(description fair lottery)
S^2_2	$\therefore$	$prob(Wt_1) = \frac{1}{1,000,000,000,000}, \ldots, prob(Wt_{1,000,000,000,000}) = \frac{1}{1,000,000,000,000}$	(provable from S^2_1)
S^2_3	$\therefore$	$\mathbf{B}^r_a\ \neg Wt_1 \wedge \ldots \wedge \mathbf{B}^r_a\ \neg Wt_{1,000,000,000,000}$	(rat. belief for a; from S^2_2)
S^2_4	$\therefore$	$\mathbf{B}^r_a\ \neg\exists t_i Wt_i$	(agglom. rat. bel.; fr. S^2_3)

Killing the Lottery Paradox



1 The Paradox

We can take the Lottery Paradox (LP), first given in print by Kyburg (1961),[1] to be based on two arguments, both apparently unexceptionable, that lead when combined to the unpalatable result that a rational agent should believe both ϕ and $\neg\phi$. I assume a lottery with 1,000,000,000,000 tickets. Here is the first sequence (the meaning of the notation is obvious):

Sequence 1 ($\mathcal{S}^1$)

S_1^1		$\mathcal{D}_{1,000,000,000,000}$	(description of fair lottery)
S_2^1	$\therefore$	$Wt_1 \oplus \ldots \oplus Wt_{1,000,000,000,000}$	(provable from S_1^1)
S_3^1	$\therefore$	$\exists t_i Wt_i$	(provable from S_2^1)
S_4^1	$\therefore$	$\boxed{\mathbf{B}_a^r\ \exists t_i Wt_i}$	(rational for a to believe S_3^1)

In $\mathcal{S}^1$, only the final inference isn't sanctioned by standard deduction. But since the description $\mathcal{D}$ itself, which we can assume to be a set of first-order formulae, is by definition off limits to doubt or question, S_3^1, deduced from what must be granted, can't be doubted unless classical deduction is to be doubted. It thus seems impossible to dodge the result that it's rational for a to believe that some ticket t_i will win.

Now here's the second sequence:

Sequence 2 ($\mathcal{S}^2$)

S_1^2		$\mathcal{D}_{1,000,000,000,000}$	(description fair lottery)
S_2^2	$\therefore$	$prob(Wt_1) = \frac{1}{1,000,000,000,000}, \ldots, prob(Wt_{1,000,000,000,000}) = \frac{1}{1,000,000,000,000}$	(provable from S_1^2)
S_3^2	$\therefore$	$\mathbf{B}_a^r\ \neg Wt_1 \wedge \ldots \wedge \mathbf{B}_a^r\ \neg Wt_{1,000,000,000,000}$	(rat. belief for a; from S_2^2)
S_4^2	$\therefore$	$\boxed{\mathbf{B}_a^r\ \neg\exists t_i Wt_i}$	(agglom. rat. bel.; fr. S_3^2)

Need Uncertainty in *DCEC**

Need Uncertainty in *DCEC**

probability calculi Gödel-encoded
9-valued logic in argument-based framework
9-valued logic <=> w/ HRI DS

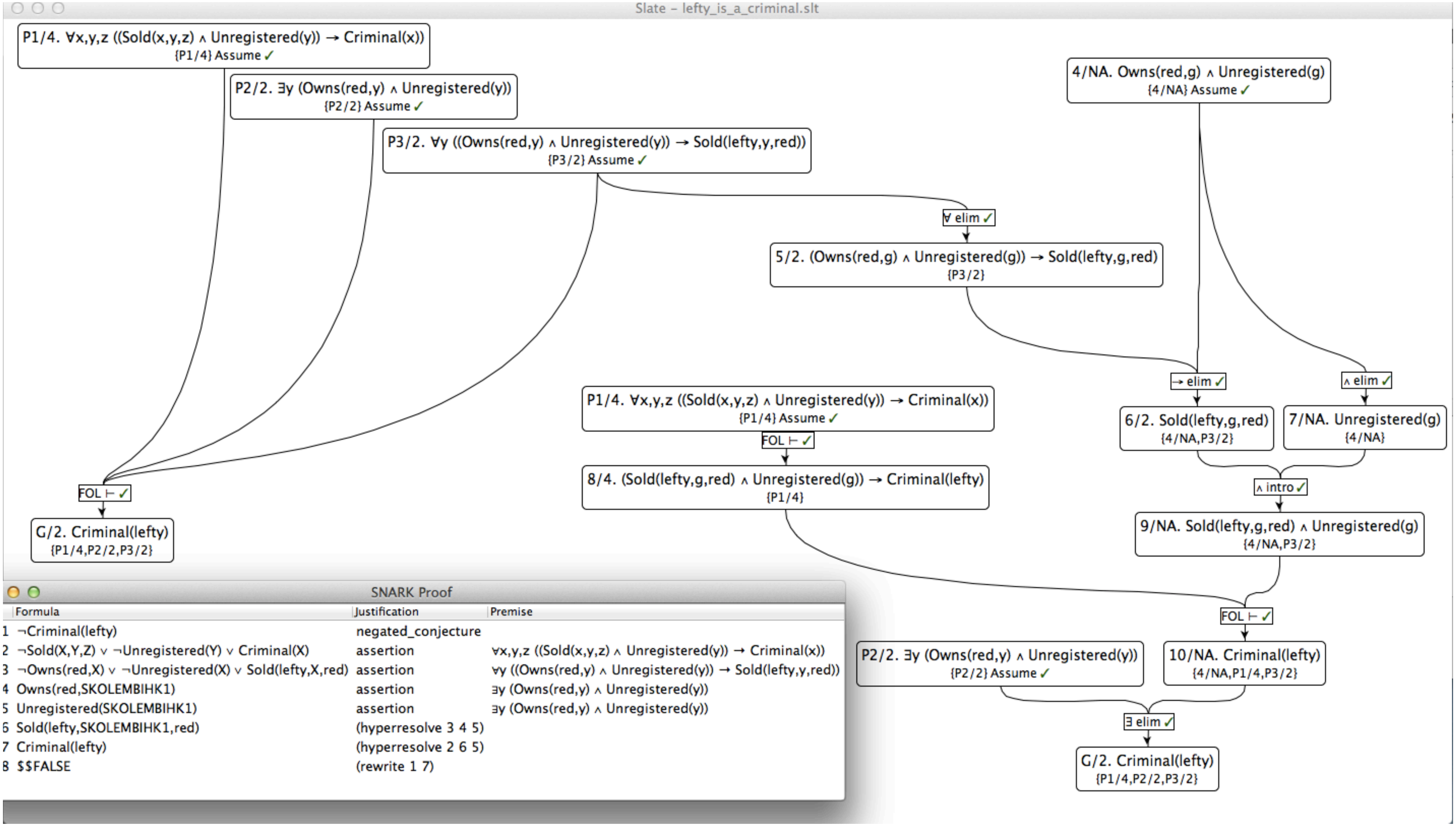
Need Uncertainty in *DCEC**

probability calculi Gödel-encoded

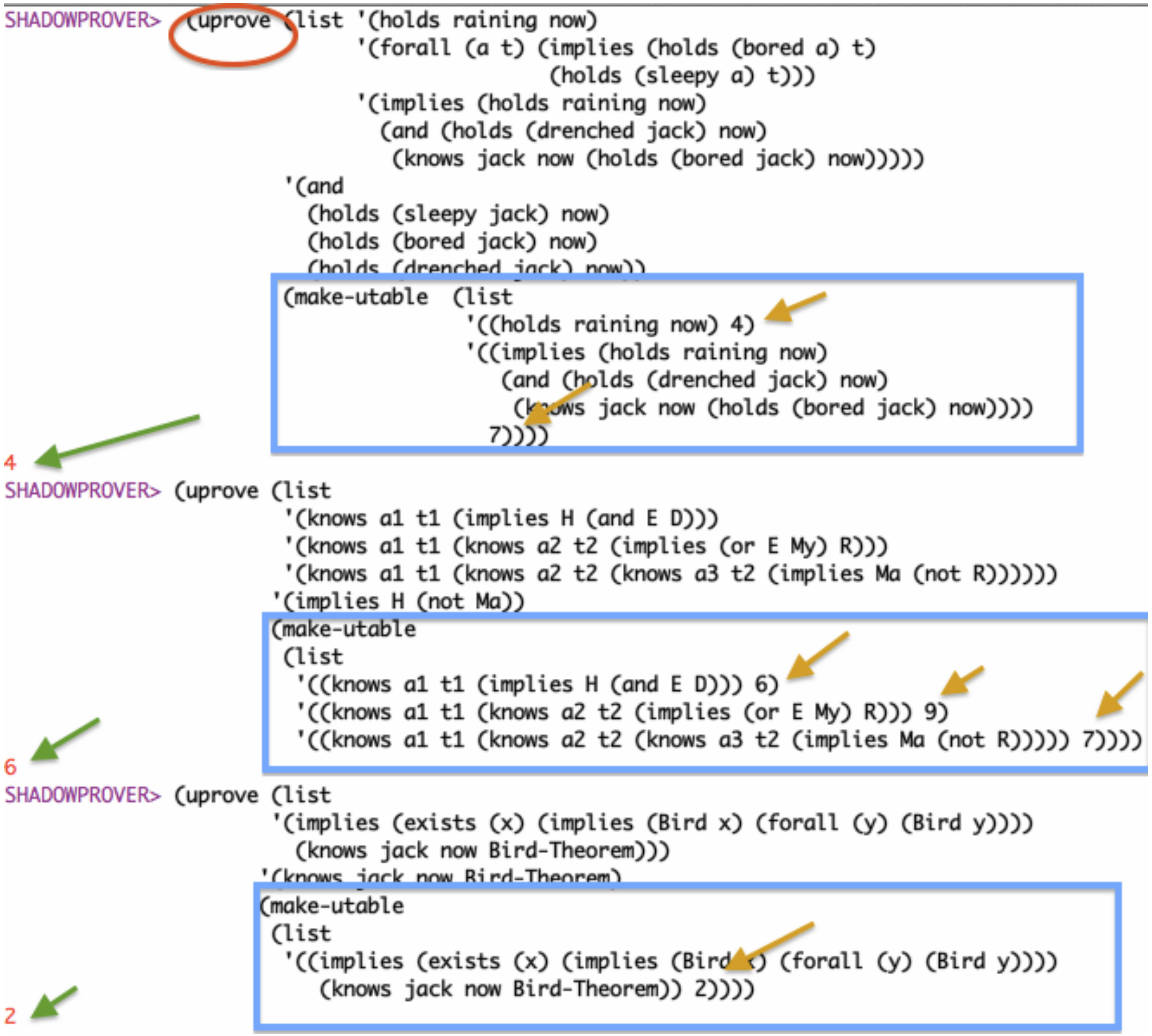
9-valued logic in argument-based framework

9-valued logic <=> w/ HRI DS

Bridging is Proof-Theory Dependent


Slate – lefty_is_a_criminal.slt
P1/4. ∀x,y,z ((Sold(x,y,z) ∧ Unregistered(y)) → Criminal(x))
{P1/4} Assume ✓
P2/2. ∃y (Owns(red,y) ∧ Unregistered(y))
{P2/2} Assume ✓
P3/2. ∀y ((Owns(red,y) ∧ Unregistered(y)) → Sold(lefty,y,red))
{P3/2} Assume ✓
4/NA. Owns(red,g) ∧ Unregistered(g)
{4/NA} Assume ✓
∀ elim ✓
5/2. (Owns(red,g) ∧ Unregistered(g)) → Sold(lefty,g,red)
{P3/2}
→ elim ✓
∧ elim ✓
P1/4. ∀x,y,z ((Sold(x,y,z) ∧ Unregistered(y)) → Criminal(x))
{P1/4} Assume ✓
FOL ⊢ ✓
6/2. Sold(lefty,g,red)
{4/NA,P3/2}
7/NA. Unregistered(g)
{4/NA}
FOL ⊢ ✓
8/4. (Sold(lefty,g,red) ∧ Unregistered(g)) → Criminal(lefty)
{P1/4}
∧ intro ✓
9/NA. Sold(lefty,g,red) ∧ Unregistered(g)
{4/NA,P3/2}
G/2. Criminal(lefty)
{P1/4,P2/2,P3/2}
FOL ⊢ ✓
P2/2. ∃y (Owns(red,y) ∧ Unregistered(y))
{P2/2} Assume ✓
10/NA. Criminal(lefty)
{4/NA,P1/4,P3/2}
∃ elim ✓
G/2. Criminal(lefty)
{P1/4,P2/2,P3/2}
SNARK Proof

	Formula	Justification	Premise
1	¬Criminal(lefty)	negated_conjecture	
2	¬Sold(X,Y,Z) ∨ ¬Unregistered(Y) ∨ Criminal(X)	assertion	∀x,y,z ((Sold(x,y,z) ∧ Unregistered(y)) → Criminal(x))
3	¬Owns(red,X) ∨ ¬Unregistered(X) ∨ Sold(lefty,X,red)	assertion	∀y ((Owns(red,y) ∧ Unregistered(y)) → Sold(lefty,y,red))
4	Owns(red,SKOLEMBIHK1)	assertion	∃y (Owns(red,y) ∧ Unregistered(y))
5	Unregistered(SKOLEMBIHK1)	assertion	∃y (Owns(red,y) ∧ Unregistered(y))
6	Sold(lefty,SKOLEMBIHK1,red)	(hyperresolve 3 4 5)	
7	Criminal(lefty)	(hyperresolve 2 6 5)	
8	$$FALSE	(rewrite 1 7)	

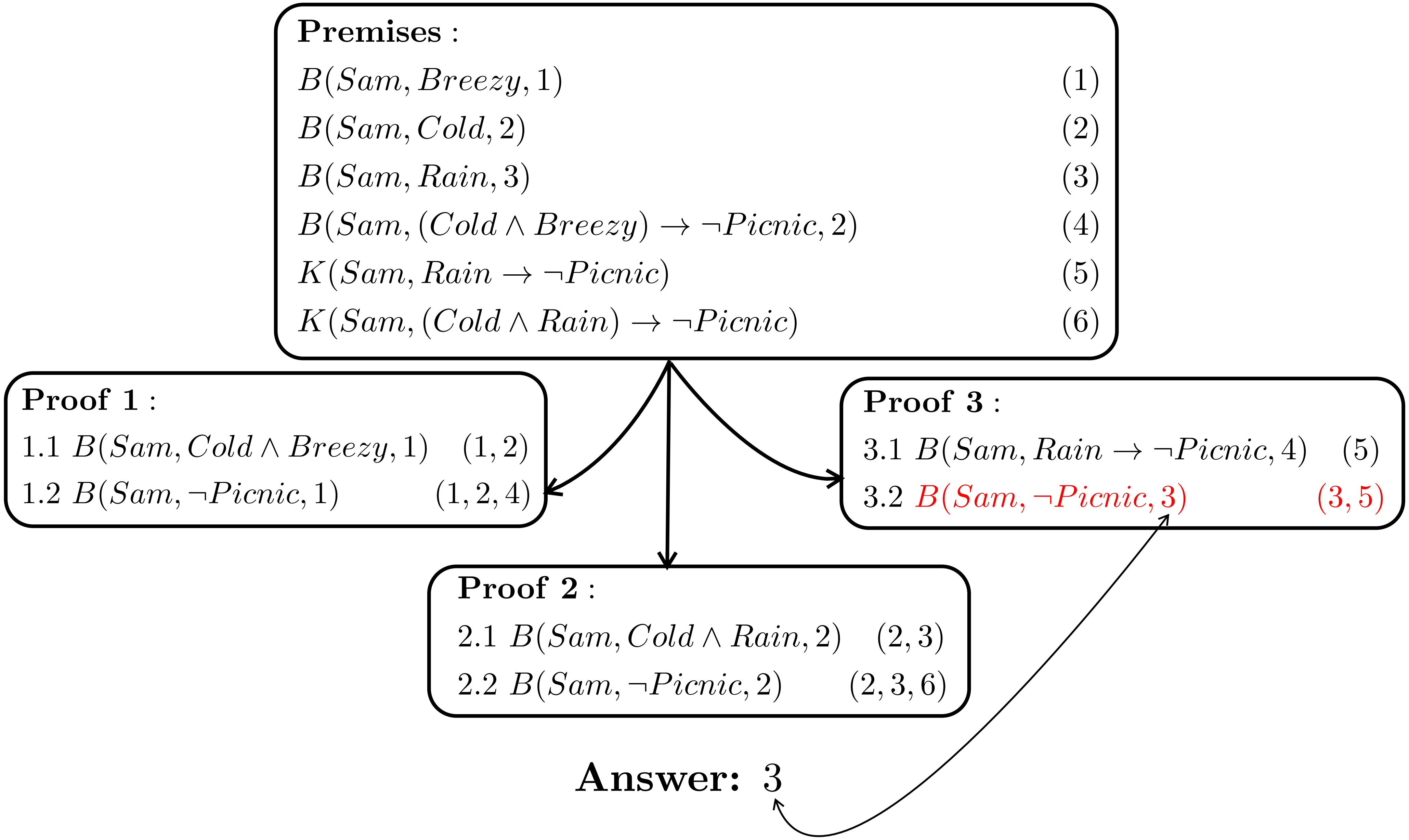


```
SHADOWPROVER> (uprove (list '(holds raining now)
                        '(forall (a t) (implies (holds (bored a) t)
                                               (holds (sleepy a) t)))
                        '(implies (holds raining now)
                          (and (holds (drenched jack) now)
                           (knows jack now (holds (bored jack) now)))))
                 '(and
                   (holds (sleepy jack) now)
                   (holds (bored jack) now)
                   (holds (drenched jack) now))
                 (make-utable  (list
                                '((holds raining now) 4)
                                '((implies (holds raining now)
                                   (and (holds (drenched jack) now)
                                    (knows jack now (holds (bored jack) now))))
                                  7))))
4
SHADOWPROVER> (uprove (list
                 '(knows a1 t1 (implies H (and E D)))
                 '(knows a1 t1 (knows a2 t2 (implies (or E My) R)))
                 '(knows a1 t1 (knows a2 t2 (knows a3 t2 (implies Ma (not R))))))
                '(implies H (not Ma))
                (make-utable
                 (list
                  '((knows a1 t1 (implies H (and E D))) 6)
                  '((knows a1 t1 (knows a2 t2 (implies (or E My) R))) 9)
                  '((knows a1 t1 (knows a2 t2 (knows a3 t2 (implies Ma (not R))))) 7))))
6
SHADOWPROVER> (uprove (list
                '(implies (exists (x) (implies (Bird x) (forall (y) (Bird y))))
                  (knows jack now Bird-Theorem)))
               '(knows jack now Bird-Theorem)
               (make-utable
                (list
                 '((implies (exists (x) (implies (Bird x) (forall (y) (Bird y))))
                    (knows jack now Bird-Theorem)) 2))))
2
```

Maximum Strength Principle

Maximum Strength Principle: Suppose a knowledge base, KB, and a formula, β, for which there exists a set of proofs, $\Phi = \{\phi_1, \phi_2, \phi_3, \ldots \phi_n\}, n > 0$, and a set of strength factors, $\Gamma = \{\gamma_1, \gamma_2, \gamma_3, \ldots \gamma_n\}$, where for $i = 1, \ldots, n$, $KB \models_{\phi_i} (\beta, \gamma_i)$, i.e., KB entails β via proof ϕ_i with strength factor, γ_i. Then, the strength factor for β, γ_β, is given by $\gamma_\beta = max(\Gamma)$.

Example: What is strength factor for $B(Sam, \neg Picnic)$?



Premises :

$B(Sam, Breezy, 1)$ (1)

$B(Sam, Cold, 2)$ (2)

$B(Sam, Rain, 3)$ (3)

$B(Sam, (Cold \wedge Breezy) \rightarrow \neg Picnic, 2)$ (4)

$K(Sam, Rain \rightarrow \neg Picnic)$ (5)

$K(Sam, (Cold \wedge Rain) \rightarrow \neg Picnic)$ (6)

Proof 1 :

1.1 $B(Sam, Cold \wedge Breezy, 1)$ (1, 2)

1.2 $B(Sam, \neg Picnic, 1)$ (1, 2, 4)

Proof 3 :

3.1 $B(Sam, Rain \rightarrow \neg Picnic, 4)$ (5)

3.2 $B(Sam, \neg Picnic, 3)$ (3, 5)

Proof 2 :

2.1 $B(Sam, Cold \wedge Rain, 2)$ (2, 3)

2.2 $B(Sam, \neg Picnic, 2)$ (2, 3, 6)

Answer: 3

slutten