Church's Thesis and Hume's Problem:
Justification as Performance

Kevin T. Kelly
Department of Philosophy
Carnegie Mellon University
kk3n@andrew.cmu.edu

Further Reading

(With C. Glymour) **"Why You'll Never Know Whether Roger Penrose is a Computer",** *Behavioral and Brain Sciences*, 13: 1990.

- *The Logic of Reliable Inquiry*, Oxford: Oxford University Press, 1996.
- (with O. Schulte) **"Church's Thesis and Hume's Problem,"** *in Logic and Scientific Methods, M. L. Dalla Chiara, et al., eds. Dordrecht: Kluwer, 1997.*
- (with O. Schulte and C. Juhl) **"Learning Theory and the Philosophy of Science",** *Philosophy of Science 64: 1997.*

"The Logic of Success", *British Journal for the Philosophy of Science, 51:2000.*

"Uncomputability: The Problem of Induction Internalized," *Theoretical Computer Science 317: 2004.*

Computability and Learnibility, Textbook manuscript, 2005.

Fateful First Cut

Relations of Ideas	Matters of fact
Analytic	Synthetic
Certain	Uncertain
A Priori	A Posteriori
Philosophy of Math	Philosophy of Science
Proofs and algorithms	Confirmation
Truth-finding	"Rationality"
Computability	Probability

Painful Progress

Relations of Ideas	Matters of fact
Analytic	Synthetic
Certain	Uncertain
A Priori	A Posteriori
Philosophy of Math	Philosophy of Science
Proofs and algorithms	Confirmation
Truth-finding	"Rationality"
Computability	Probability

Slight Rearrangement

Philosophy of Math	Philosophy of Science
Certain	Uncertain
A Priori	A Posteriori
Proofs and algorithms	Confirmation
Truth-finding	"Rationality"
Computability	Probability

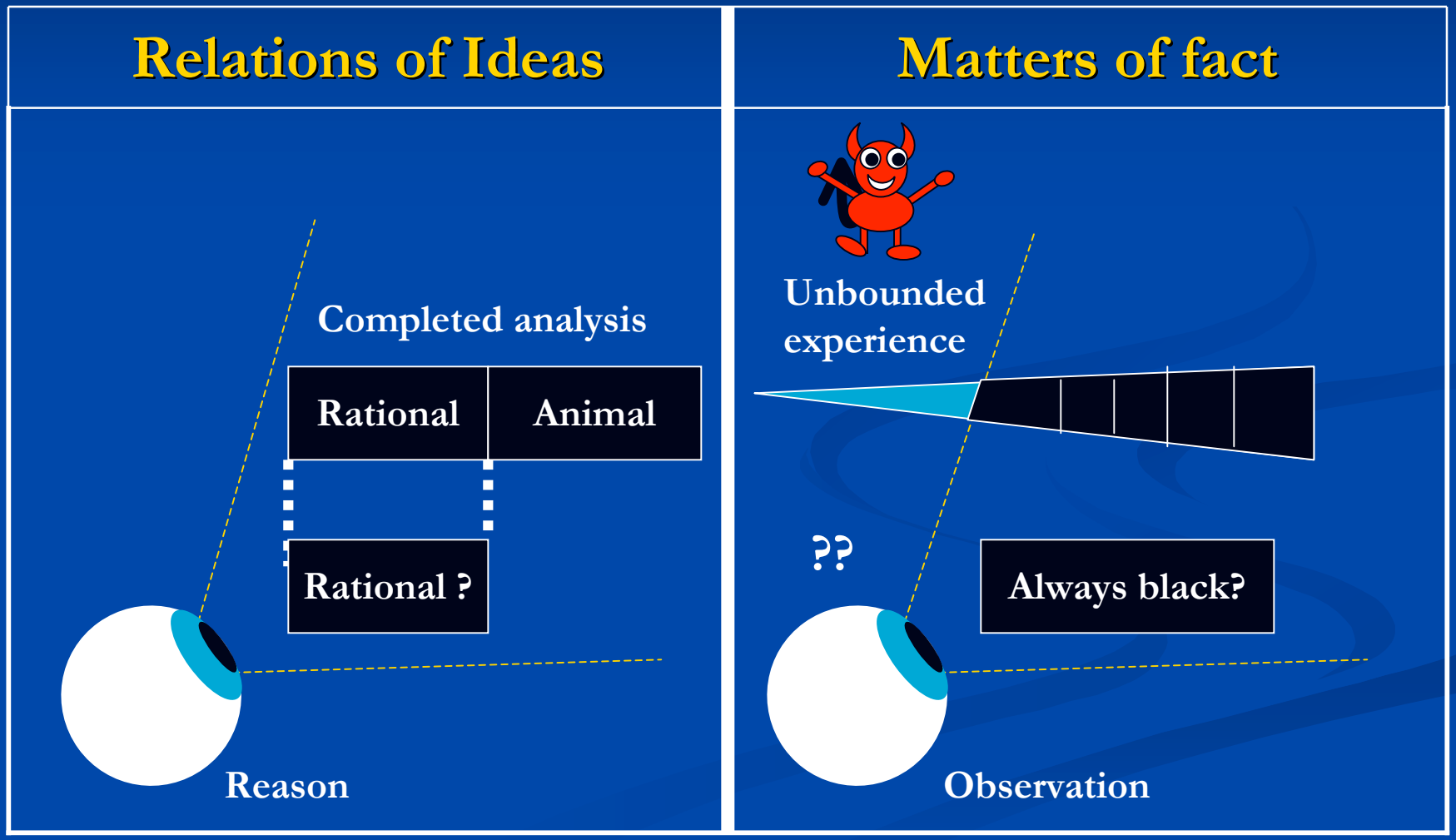
Philosophy of Science 101

Formal Questions	Scientific Questions
	Unverifiable "The sun will never stop rising"
Verifiable "The given number is even"	

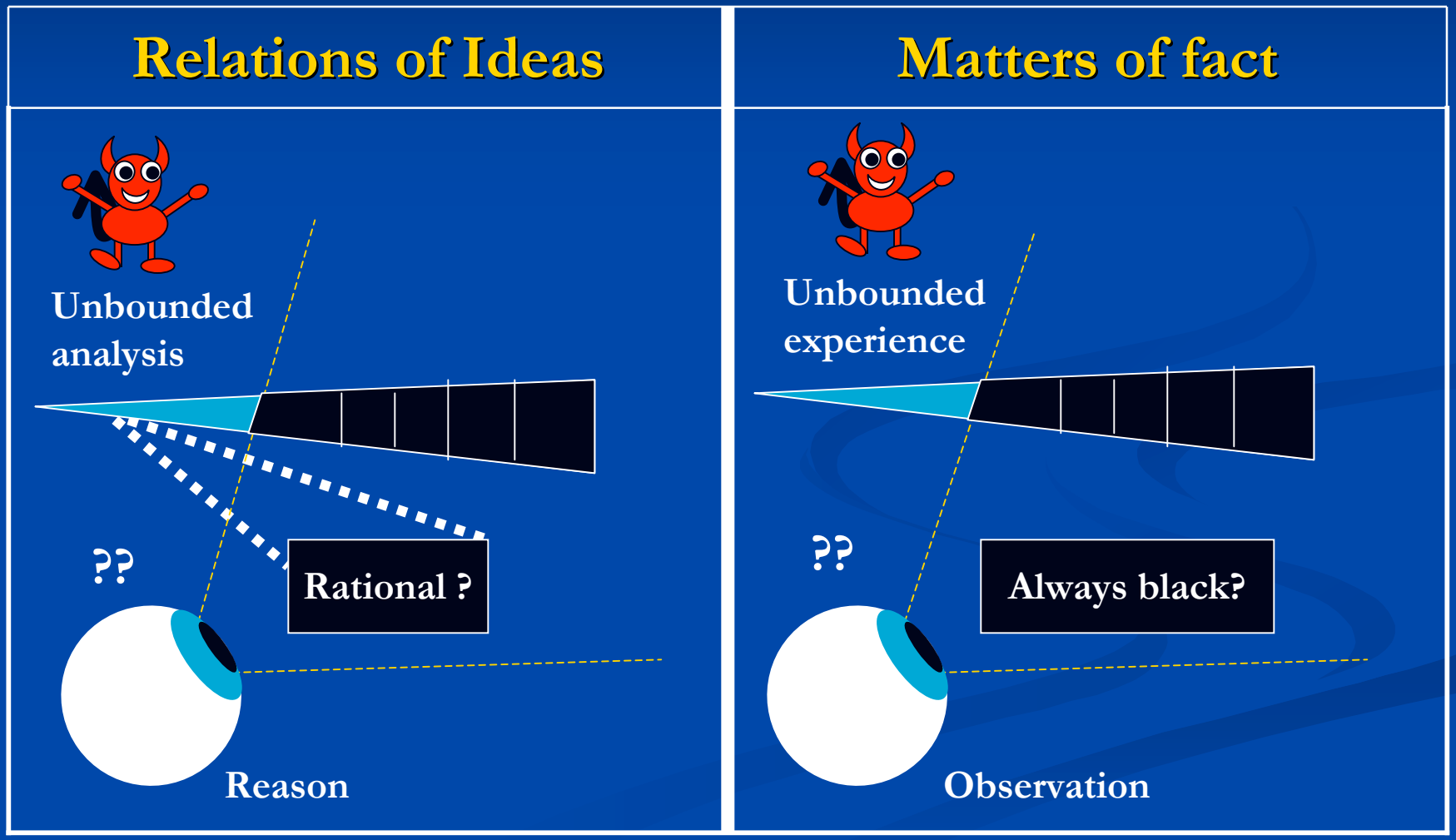
Wrong Twice!

Formal Questions	Scientific Questions
Unverifiable "The given computation will never halt"	**Unverifiable** "The sun will never stop rising"
Verifiable "The given number is even"	**Verifiable** "The next emerald is green"

Classical Excuse

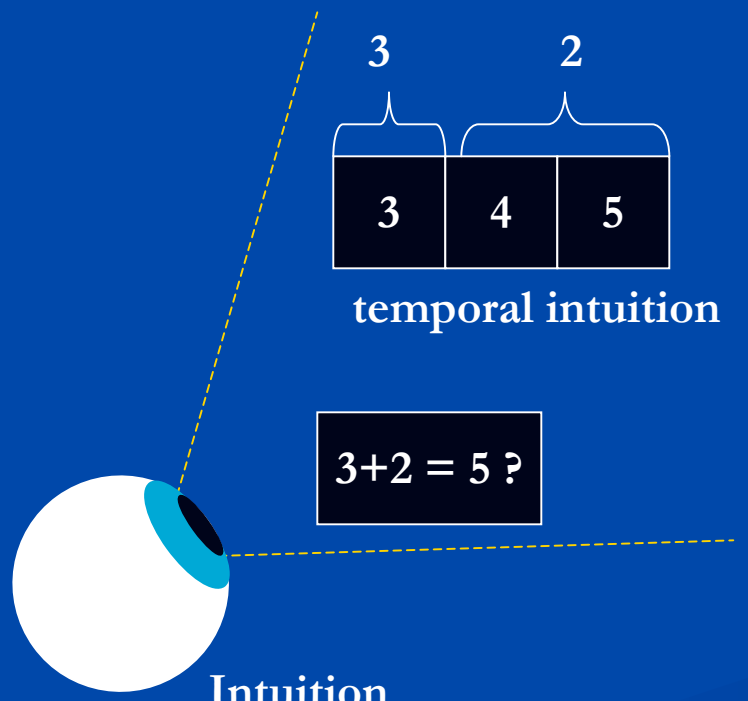


But Maybe…

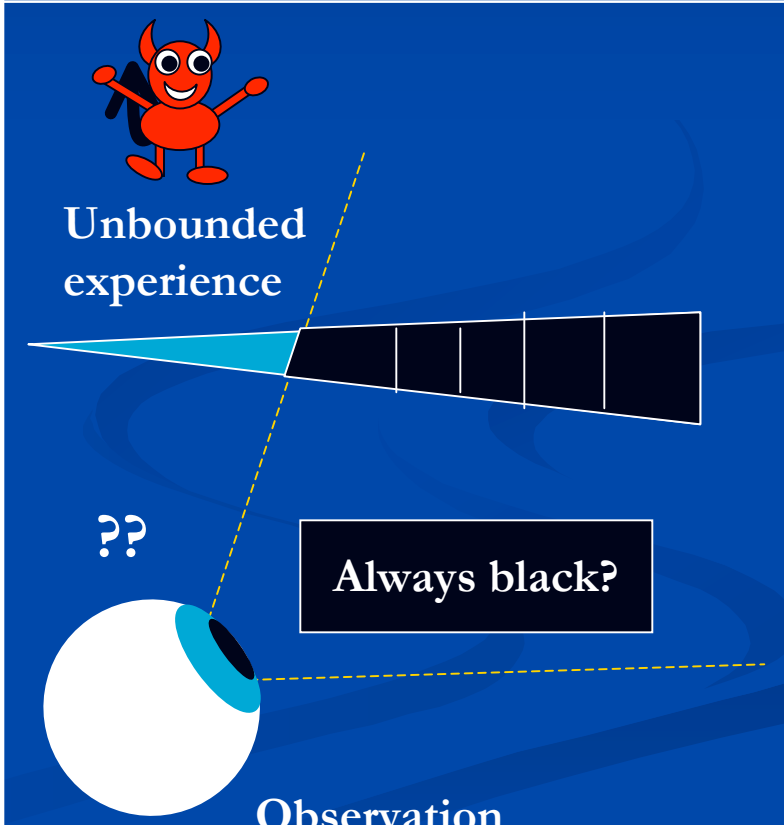

Relations of Ideas
Matters of fact
Unbounded analysis
??
Rational ?
Reason
Unbounded experience
??
Always black?
Observation

Kant

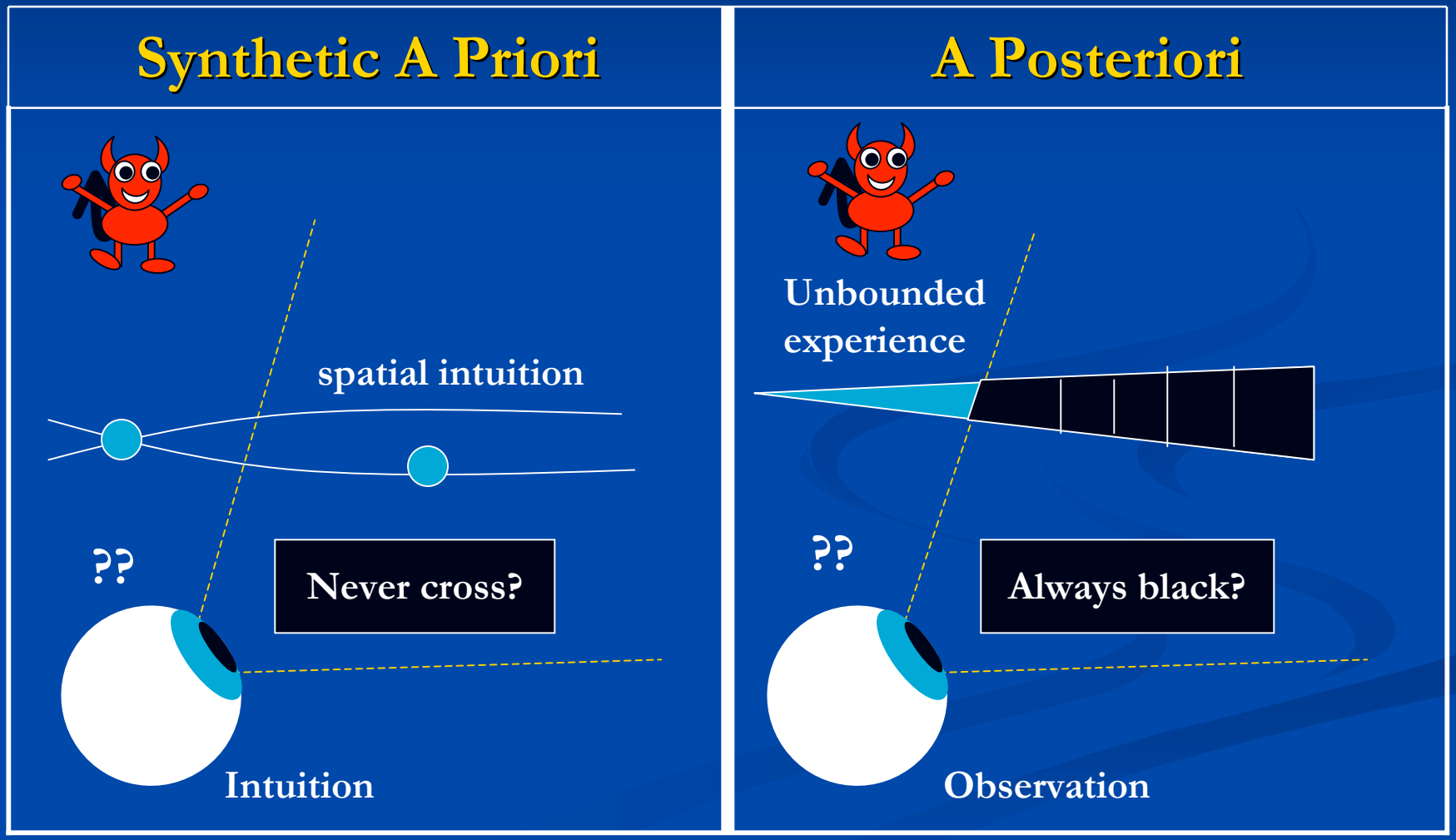
Synthetic A Priori



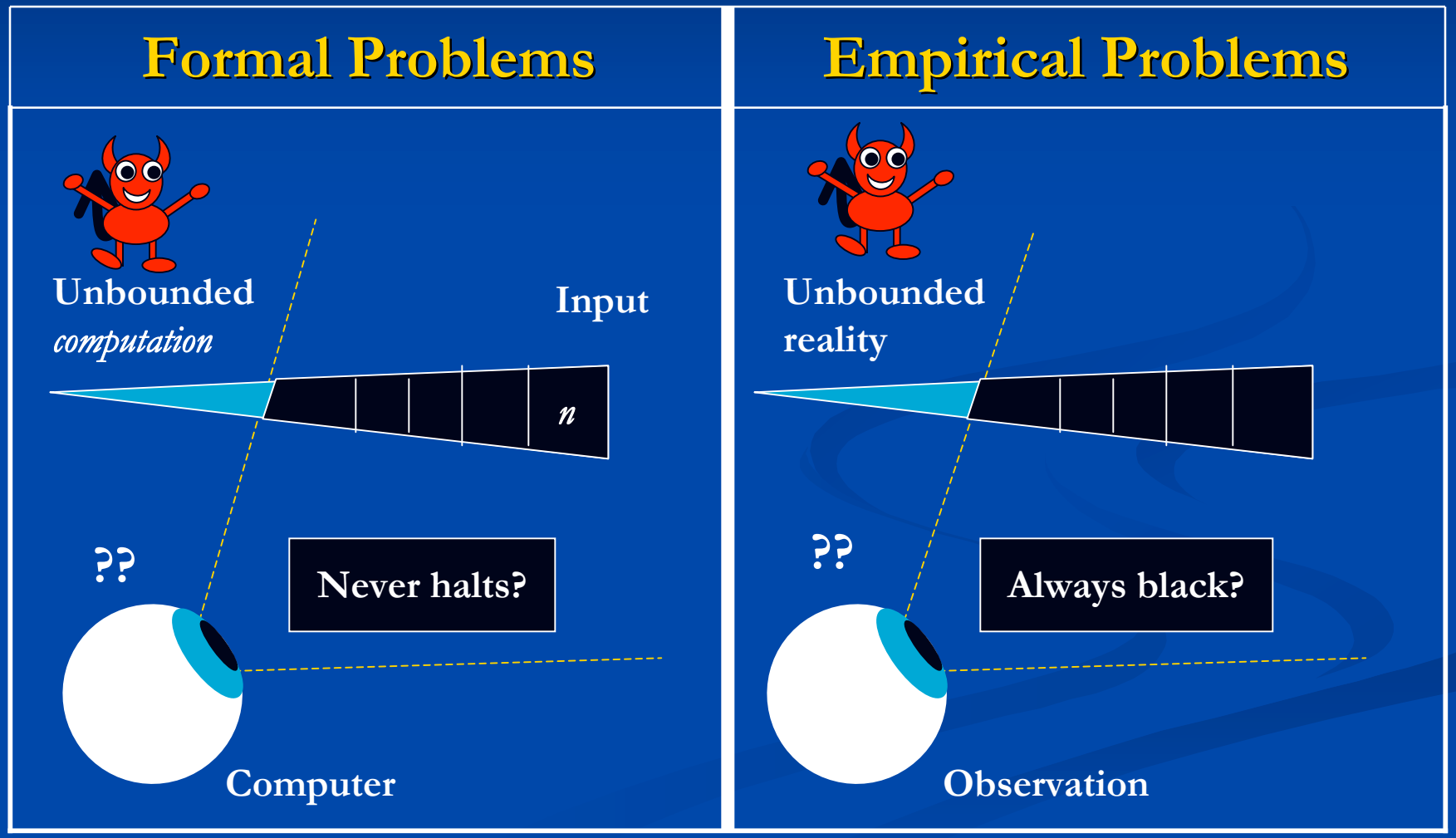
A Posteriori



Oops!


Synthetic A Priori
A Posteriori
spatial intuition
??
Never cross?
Intuition
Unbounded
experience
??
Always black?
Observation

After Turing



Bad First Cut

Formal Questions	Scientific Questions
Unverifiable "The given computation will never halt"	**Unverifiable** "The sun will never stop rising"
Verifiable "The given number is even"	**Verifiable** "The next emerald is green"

Good First Cut

Formal Questions	Scientific Questions
Unverifiable "The given computation will never halt"	**Unverifiable** "The sun will never stop rising"
Verifiable "The given number is even"	**Verifiable** "The next emerald is green"

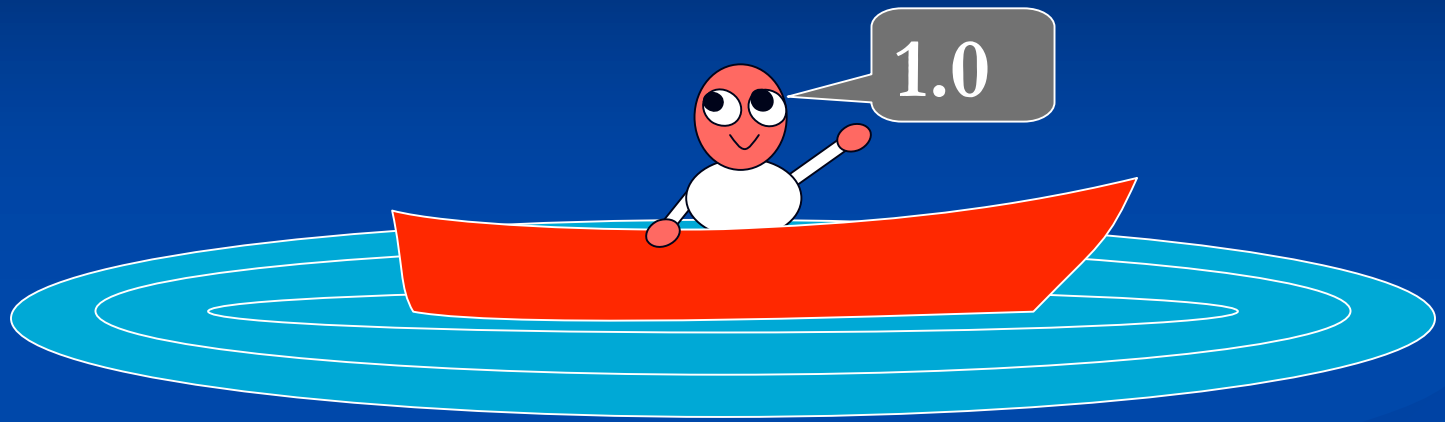
Computational Epistemology

For formal **and** empirical questions:

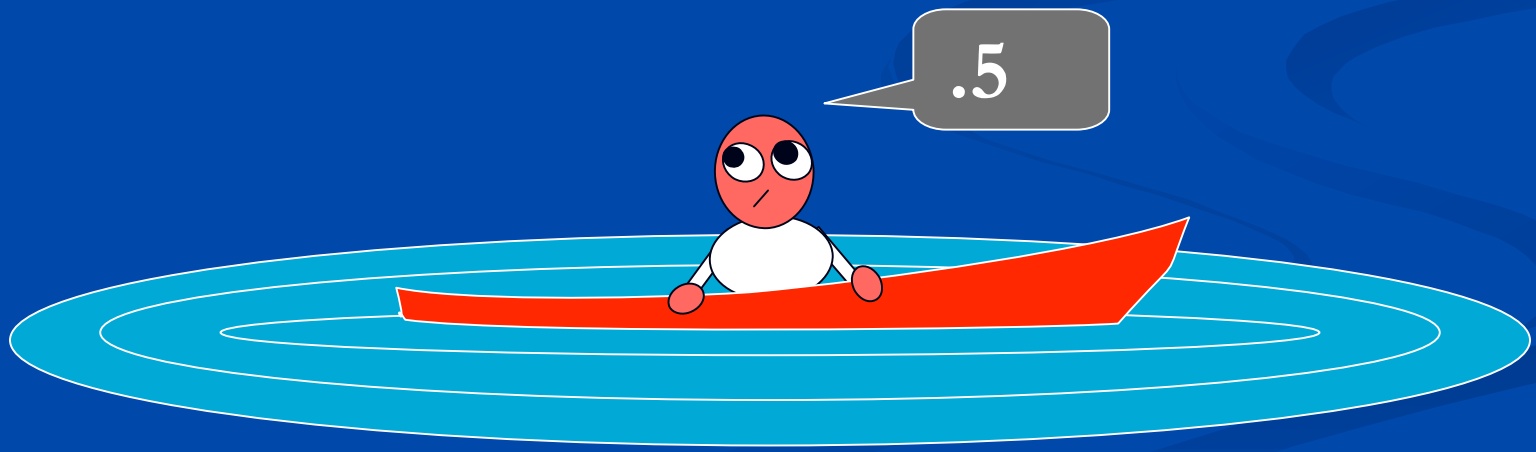
Justification = **truth-finding performance**

- Find the truth in the **best possible sense**
- And then as **efficiently as possible.**

Verification as "Support"

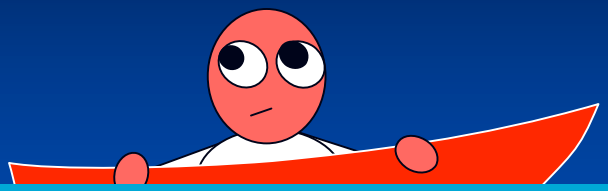


Deductive verification



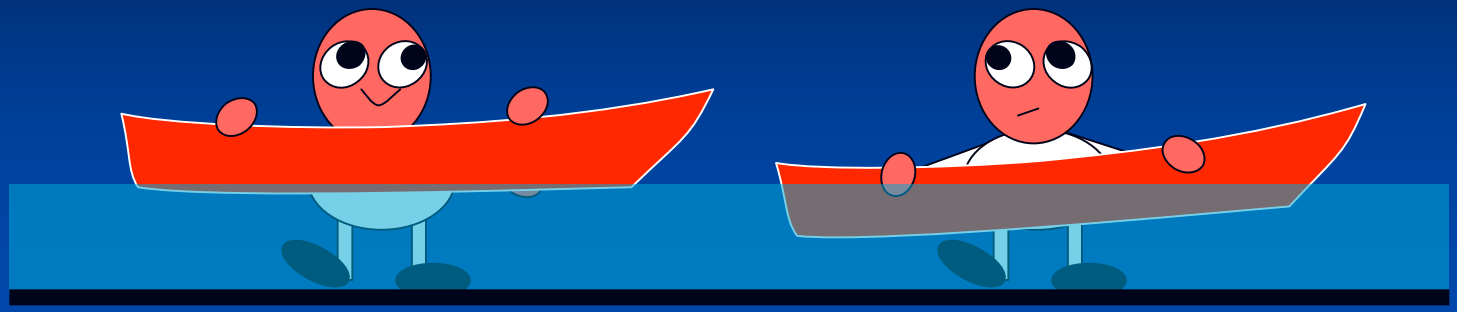
Inductive verification

Paradigms Evolve



- Objective support by evidence.

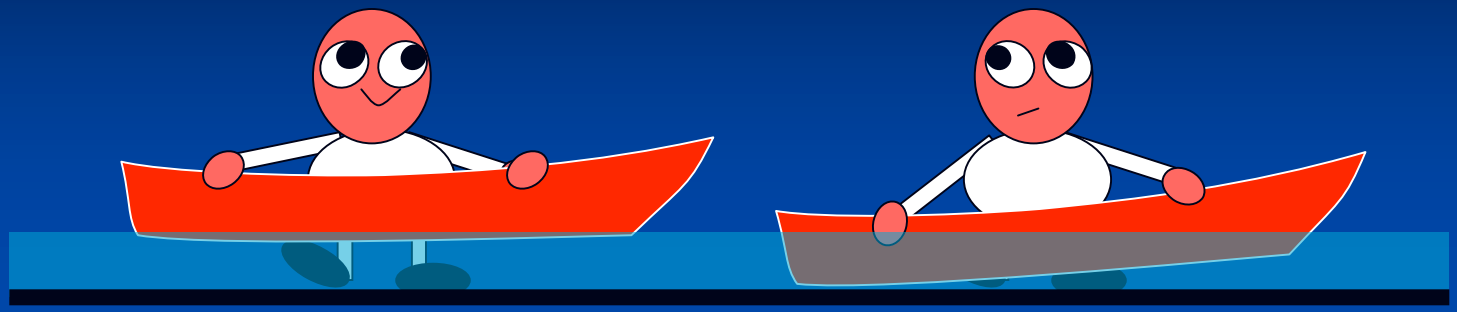
Paradigms Evolve…



- Objective support by evidence.

- Coherent personal opinion.
- "Rational" update.

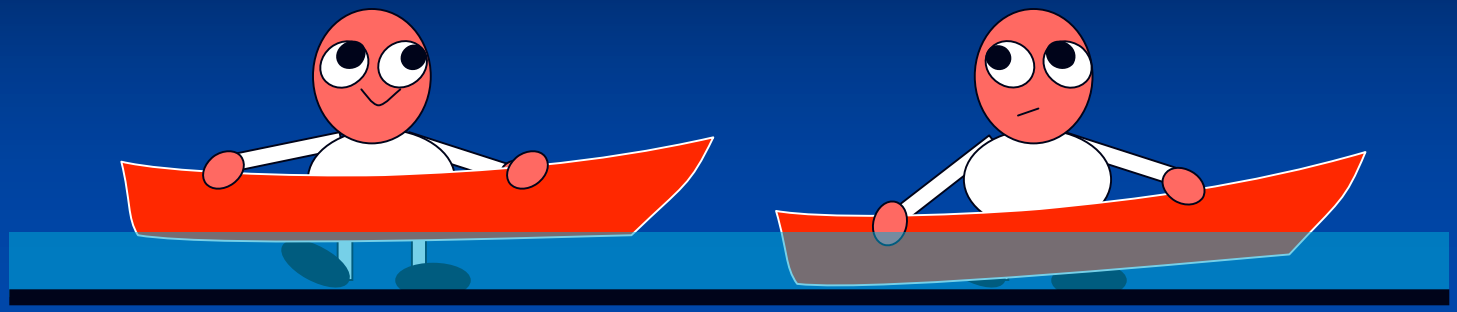
Paradigms Evolve…



- Objective support by evidence.

- Coherent personal opinion.
- "Rational" update.

But Leading Metaphors Matter

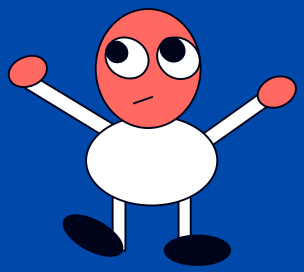


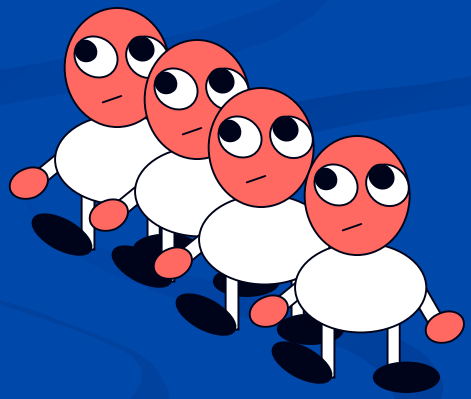
- Myopic
- Bedrock = "rationality" intuitions
- Success = being "rational"
- Truth-finding performance deferred or ignored
- Problem complexity ignored
- Little resemblance to computability

Verification as Halting

Conclusion

Data

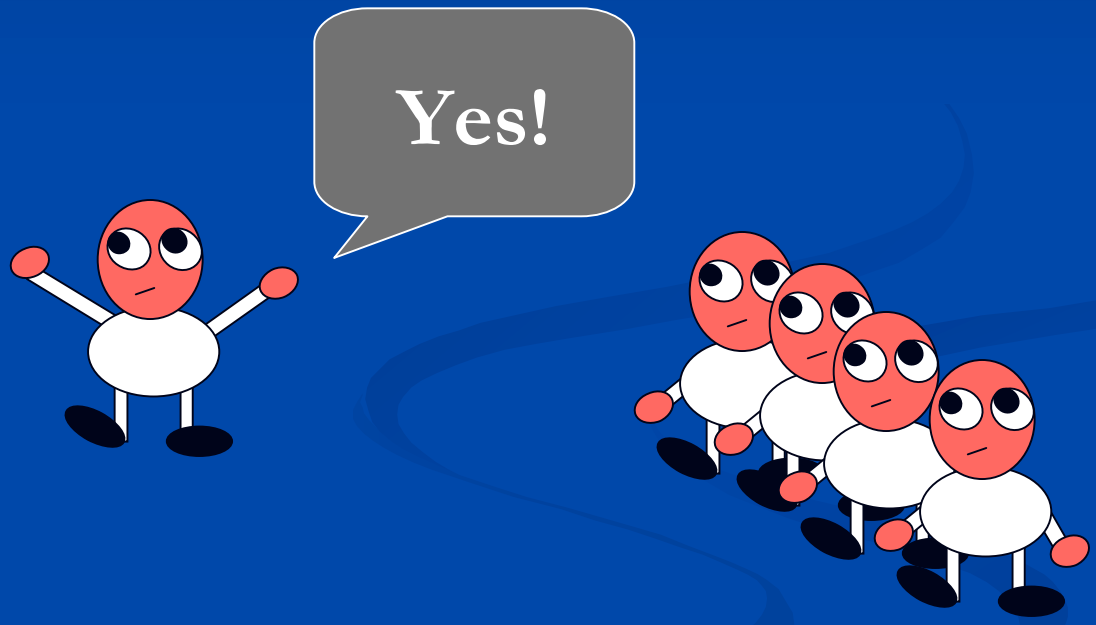




Verification as Halting

Conclusion

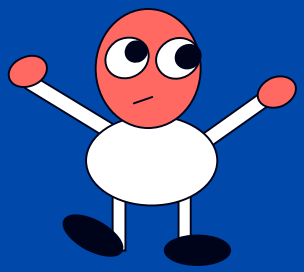
Data

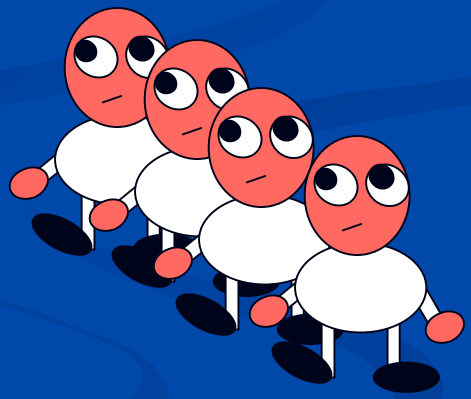


Verification as Halting

Conclusion

Data

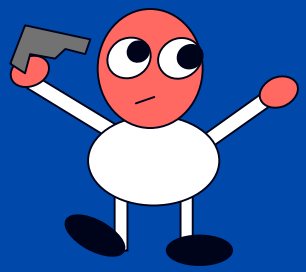


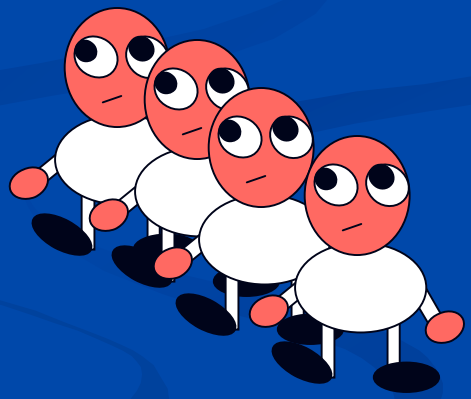


Verification as Halting

Conclusion

Data

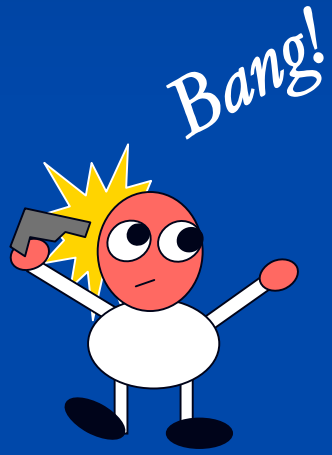


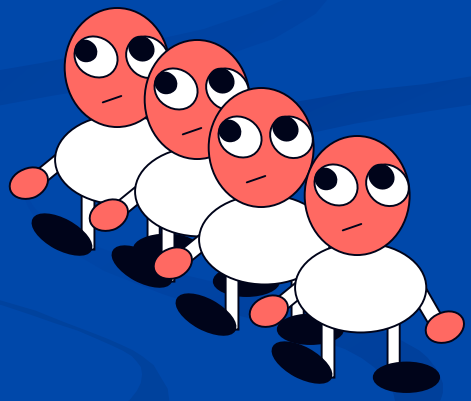


Verification as Halting

Conclusion

Data

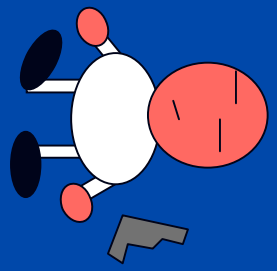


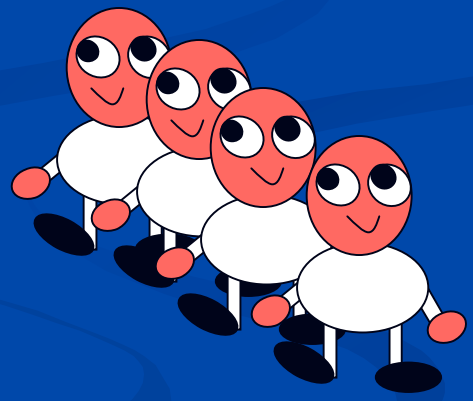


Verification as Halting

Conclusion

Data

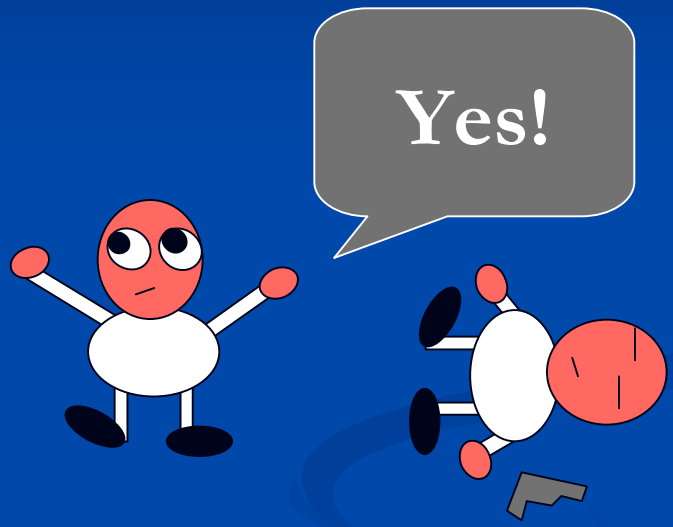




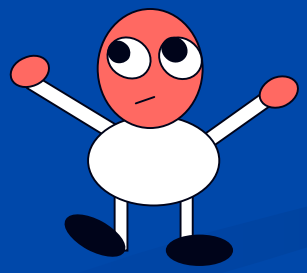
Can't take it back.

Verifiability

Conclusion true

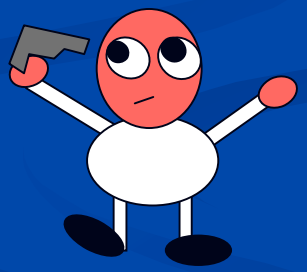


Conclusion false



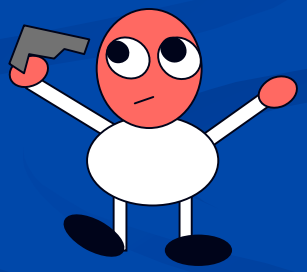
Laws are Not Verifiable

Always green?



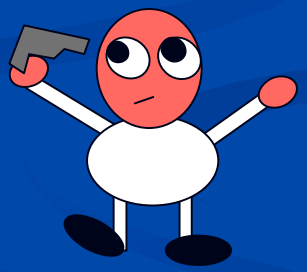
Laws are Not Verifiable

Always green?

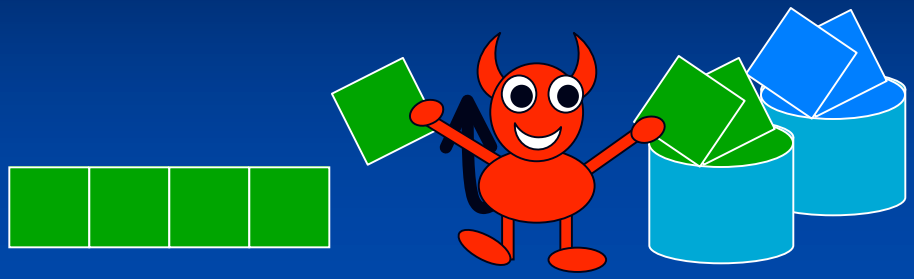


Laws are Not Verifiable

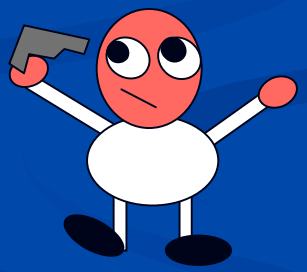
Always green?



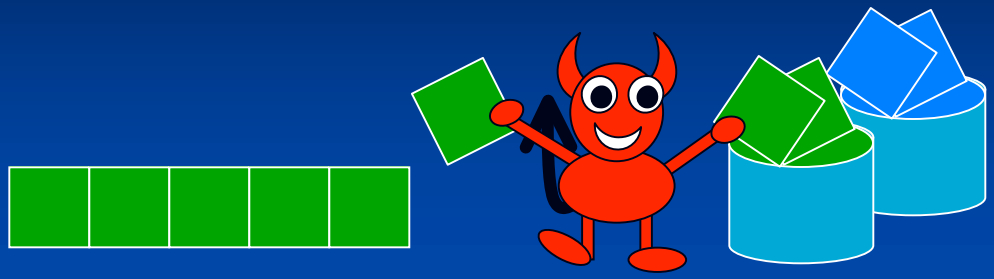
Laws are Not Verifiable



Always green?



Laws are Not Verifiable



Always green?

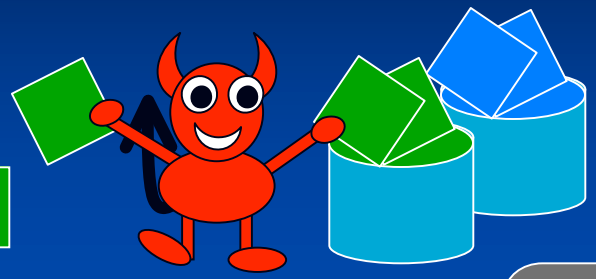
You must halt with "yes" if it's true!

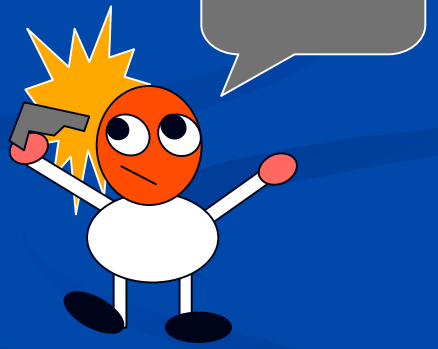
Laws are Not Verifiable

Always green?

You must halt with "yes" if it's true!

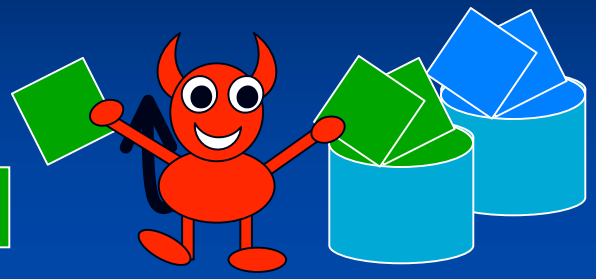
Laws are Not Verifiable



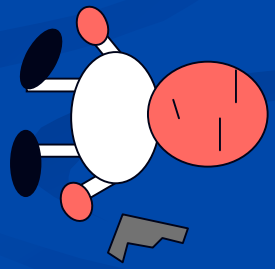


Always green?

Laws are Not Verifiable

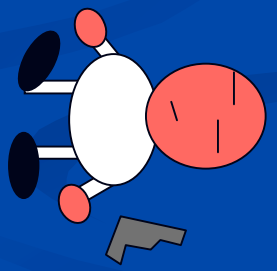


Always green?



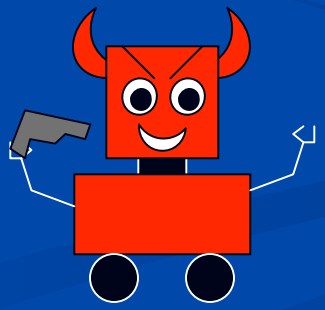
Laws are Not Verifiable

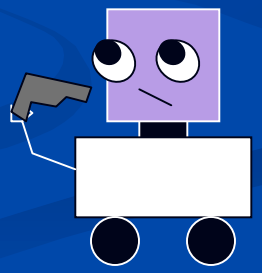
Always green?



A Purely Formal Problem

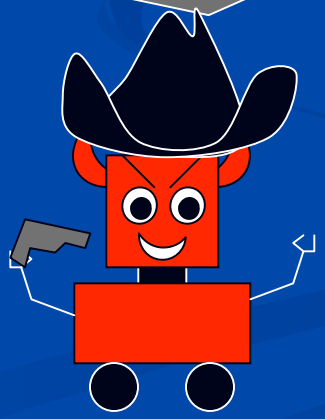
Never shoots?

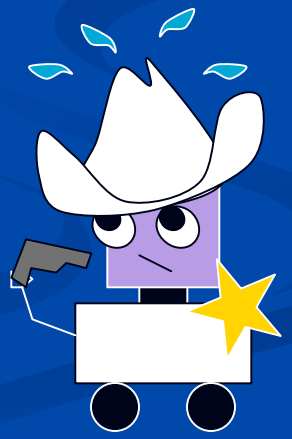




Analogy?

I ain't a' shootin', yeller belly!

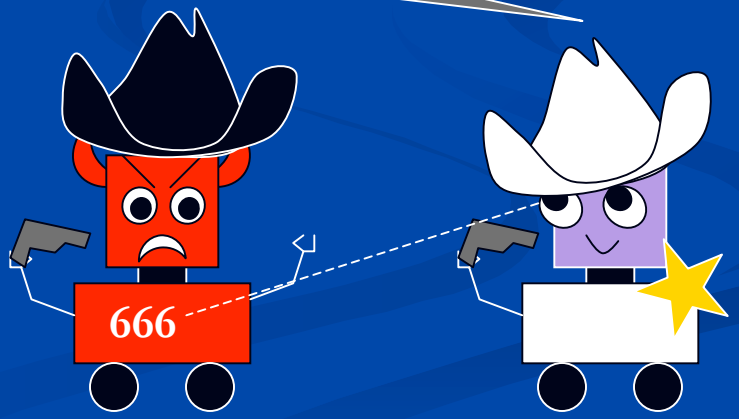




Never shoots?

Apparently Not

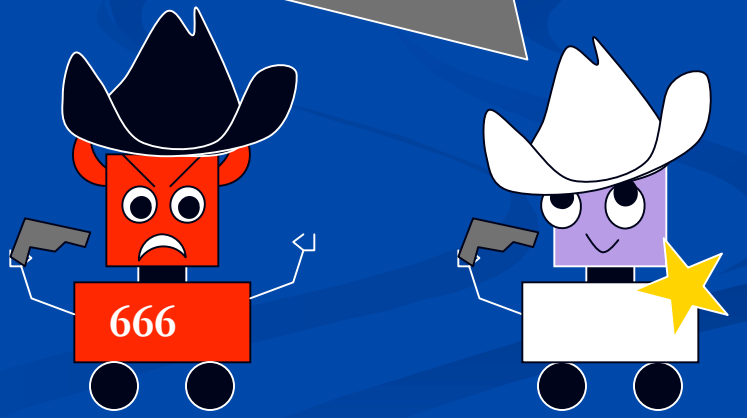
Game's up, J.R.!
Yer program's showin'!



Never shoots?

Apparently Not

Now shaddup! I'm calculatin' a-preeorey-like what yer gonna do.



Never shoots?

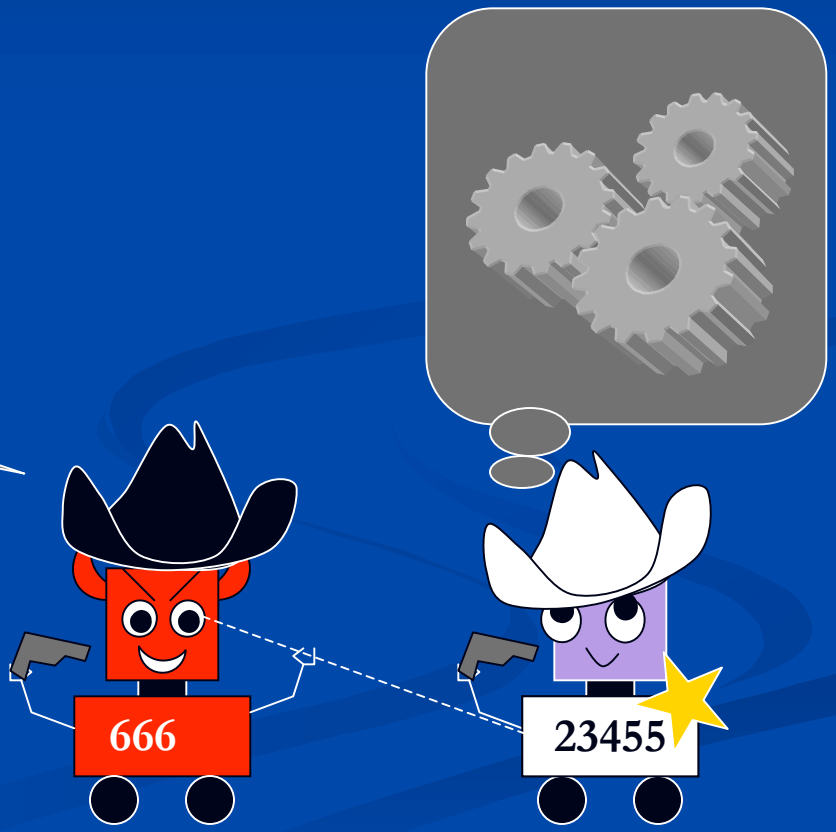
Apparently Not

Never shoots?

But Then Again…

I got it! Two kin play that opry oary game!

Never shoots?

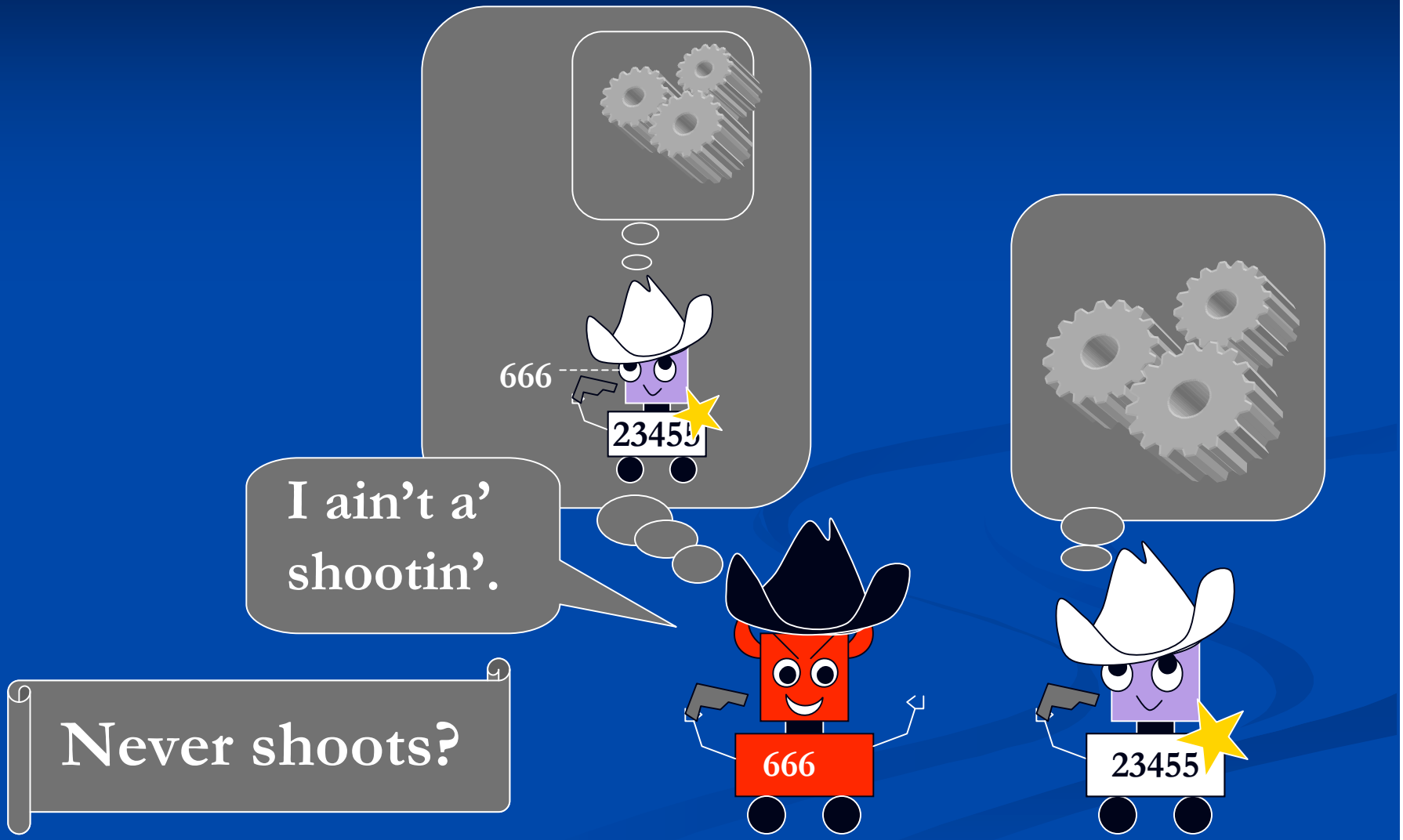


But Then Again…

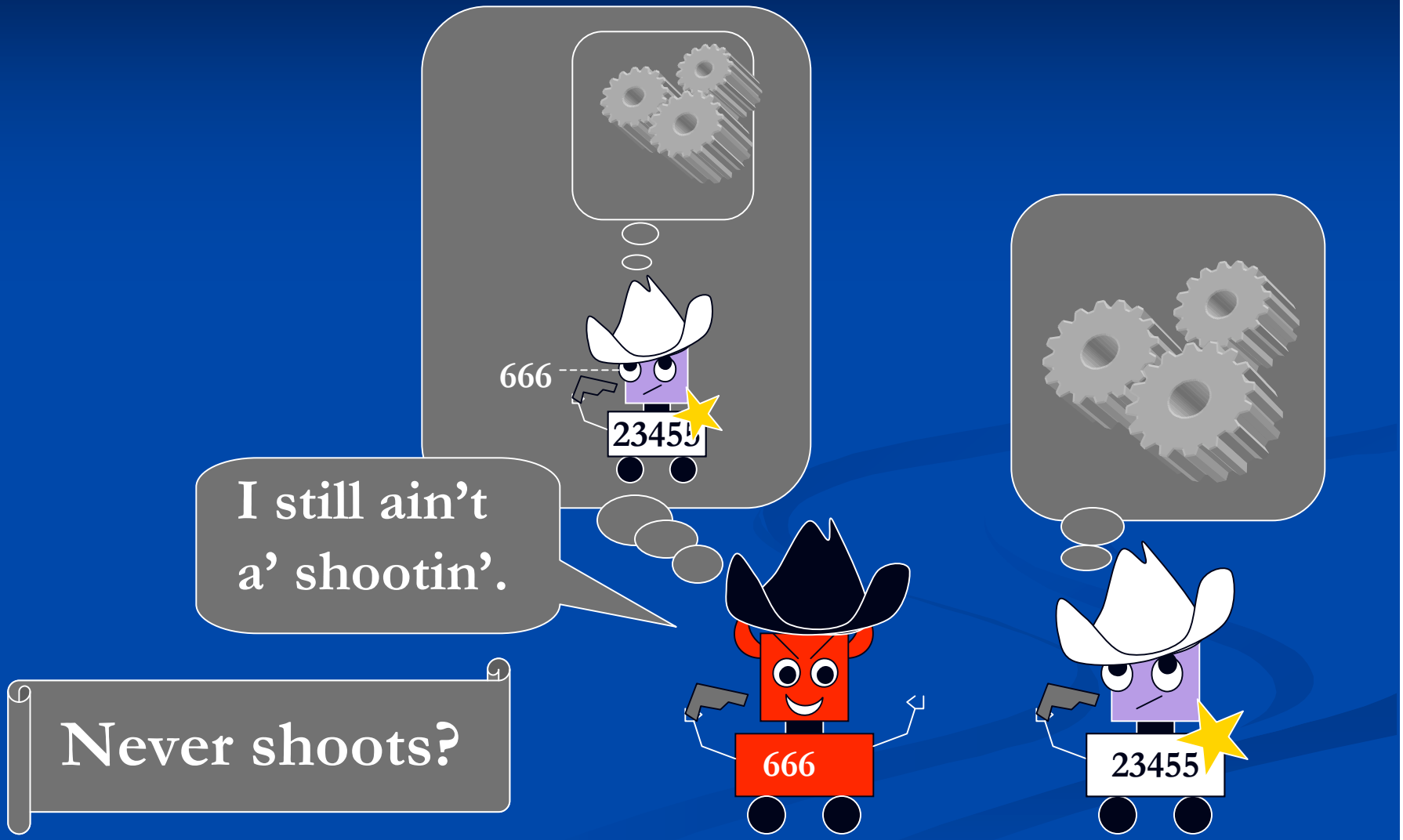
I won't
shoot 'til I
see him shoot
in this here
opry oary
sim-yooo-laytion!

Never shoots?

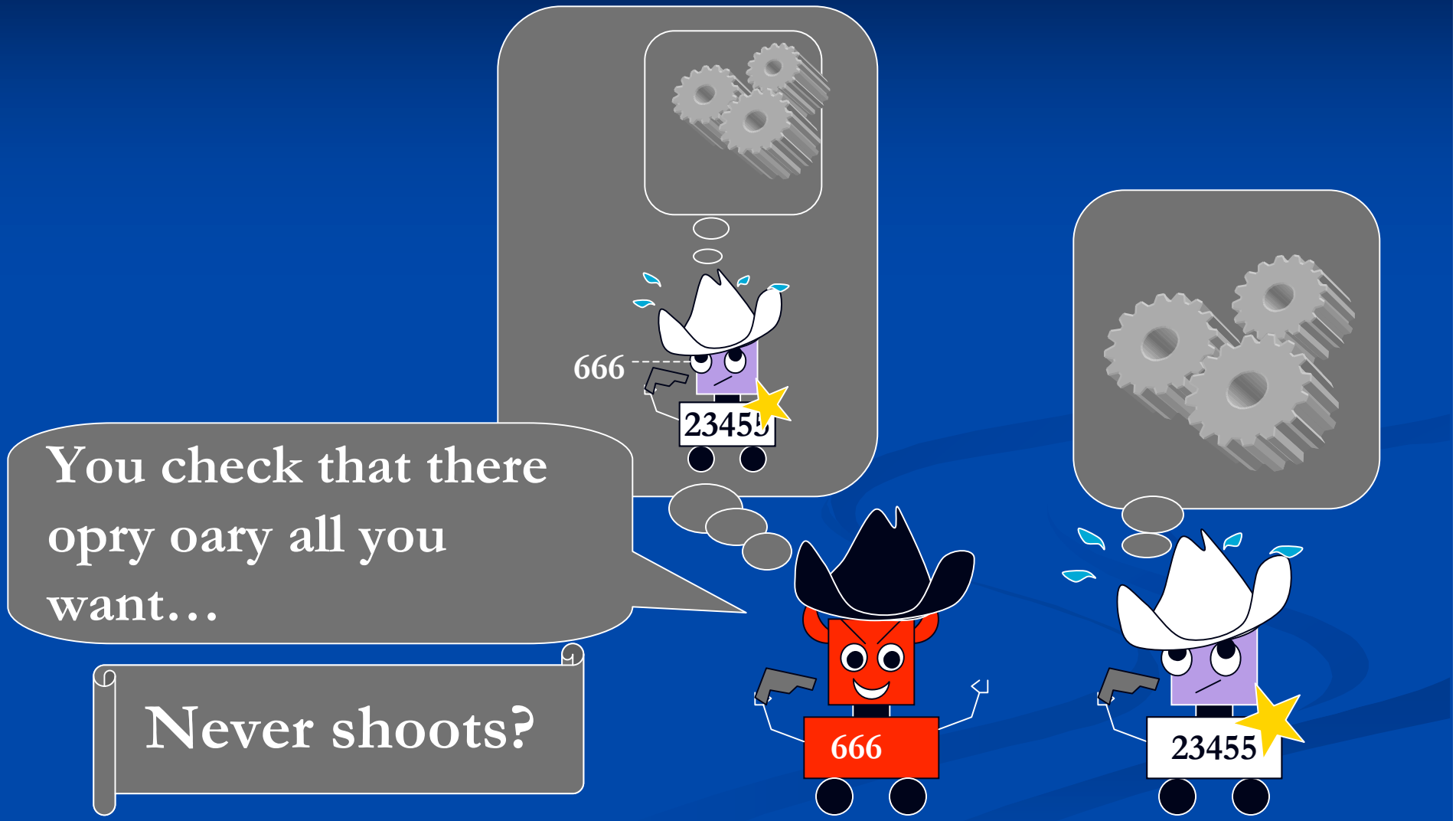
But Then Again…



But Then Again…



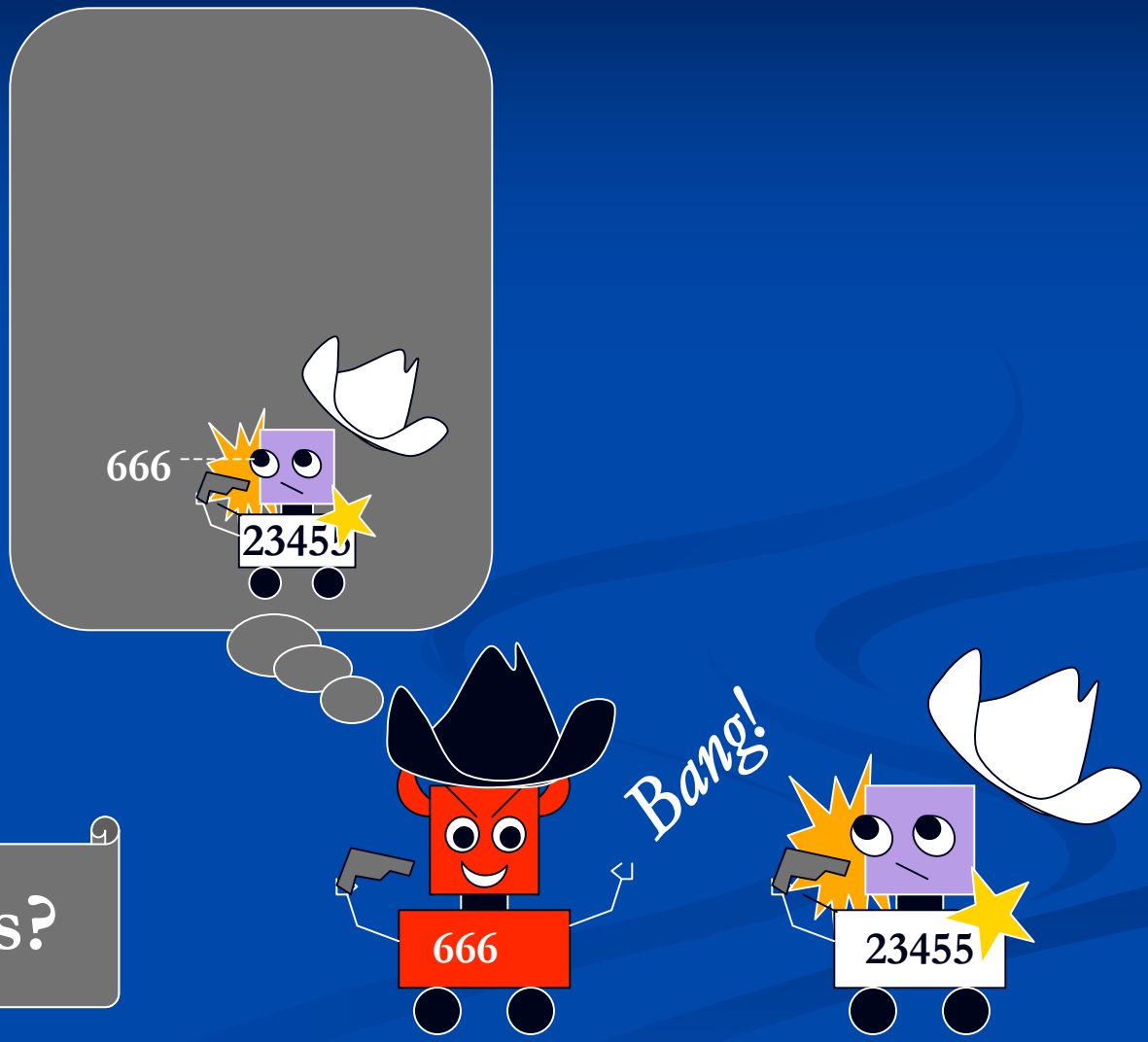
But Then Again…



But Then Again…

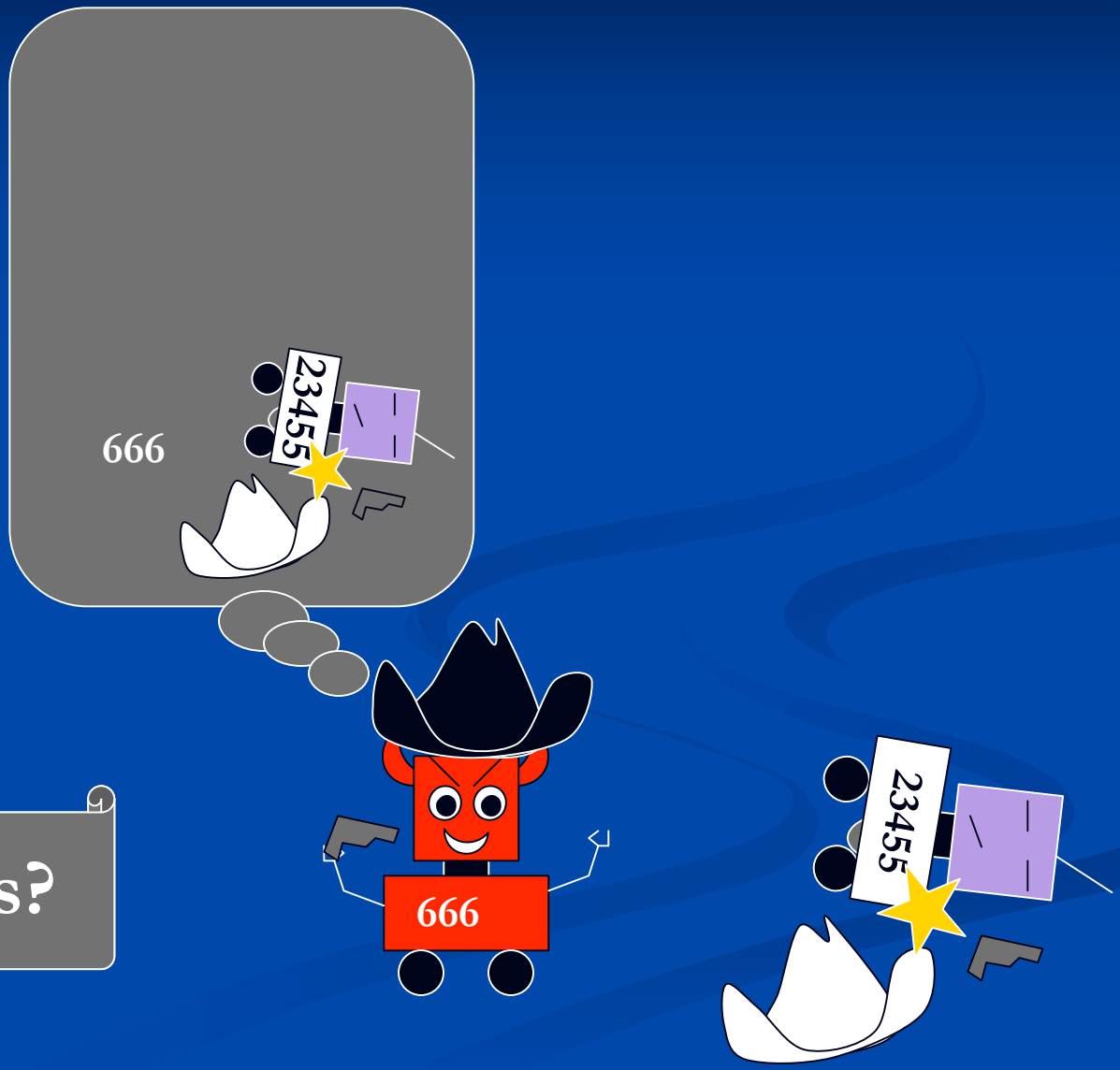
But Then Again…

But Then Again…



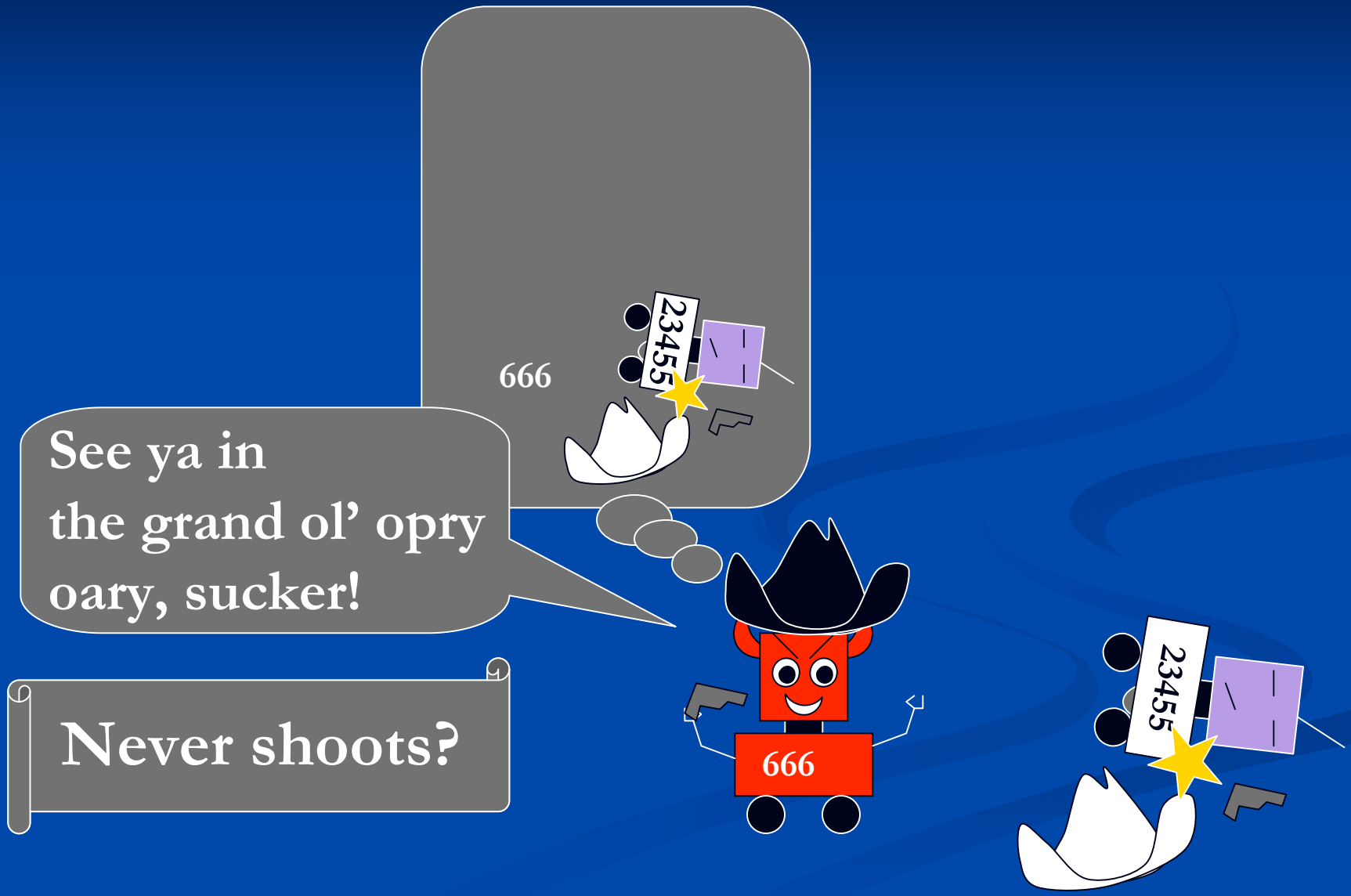
Never shoots?

But Then Again…



Never shoots?

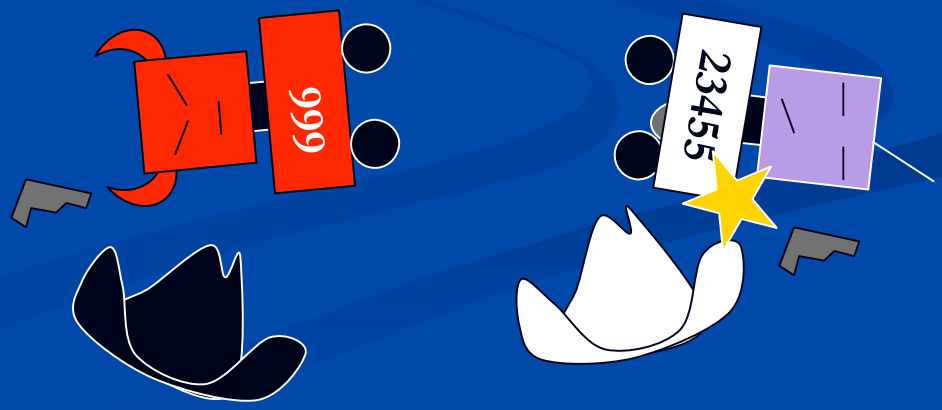
But Then Again…



But Then Again…

But Then Again…

Never shoots?

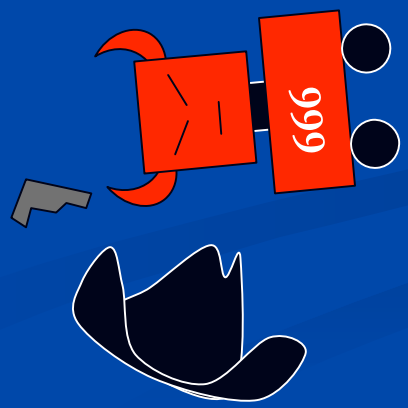


But Then Again…

ROBO
SHERIFF

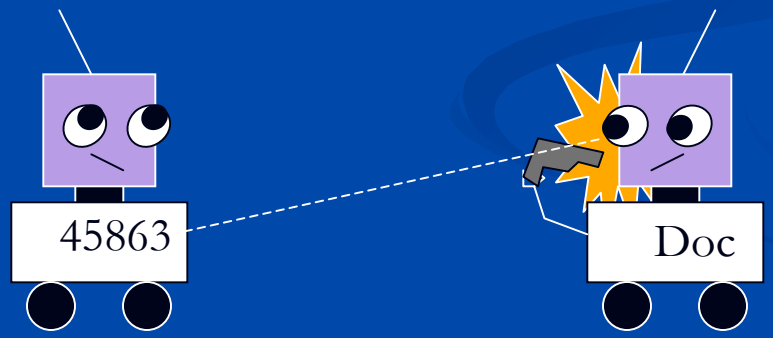
Fooled
A priori

Never shoots?



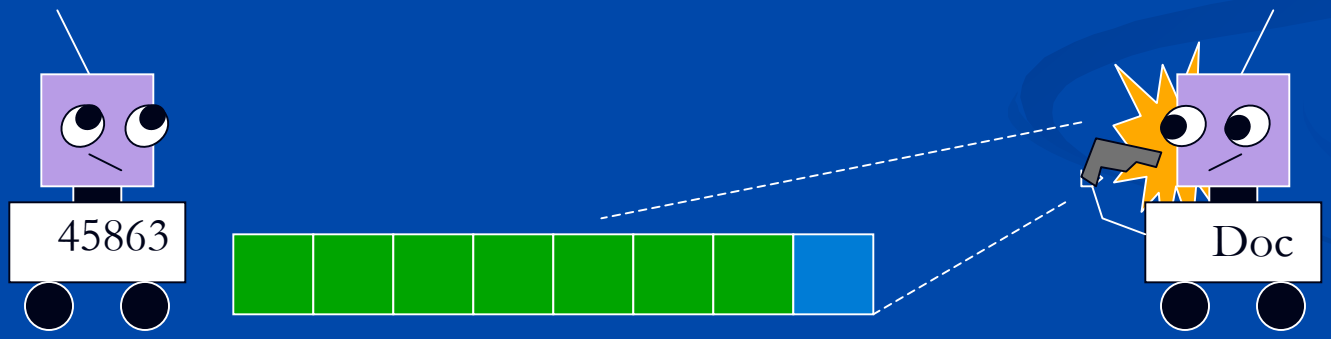
Rice-Shapiro Theorem

Whatever a computable cognitive scientist could verify about an arbitrary computer's input-output behavior by formally analyzing its program…

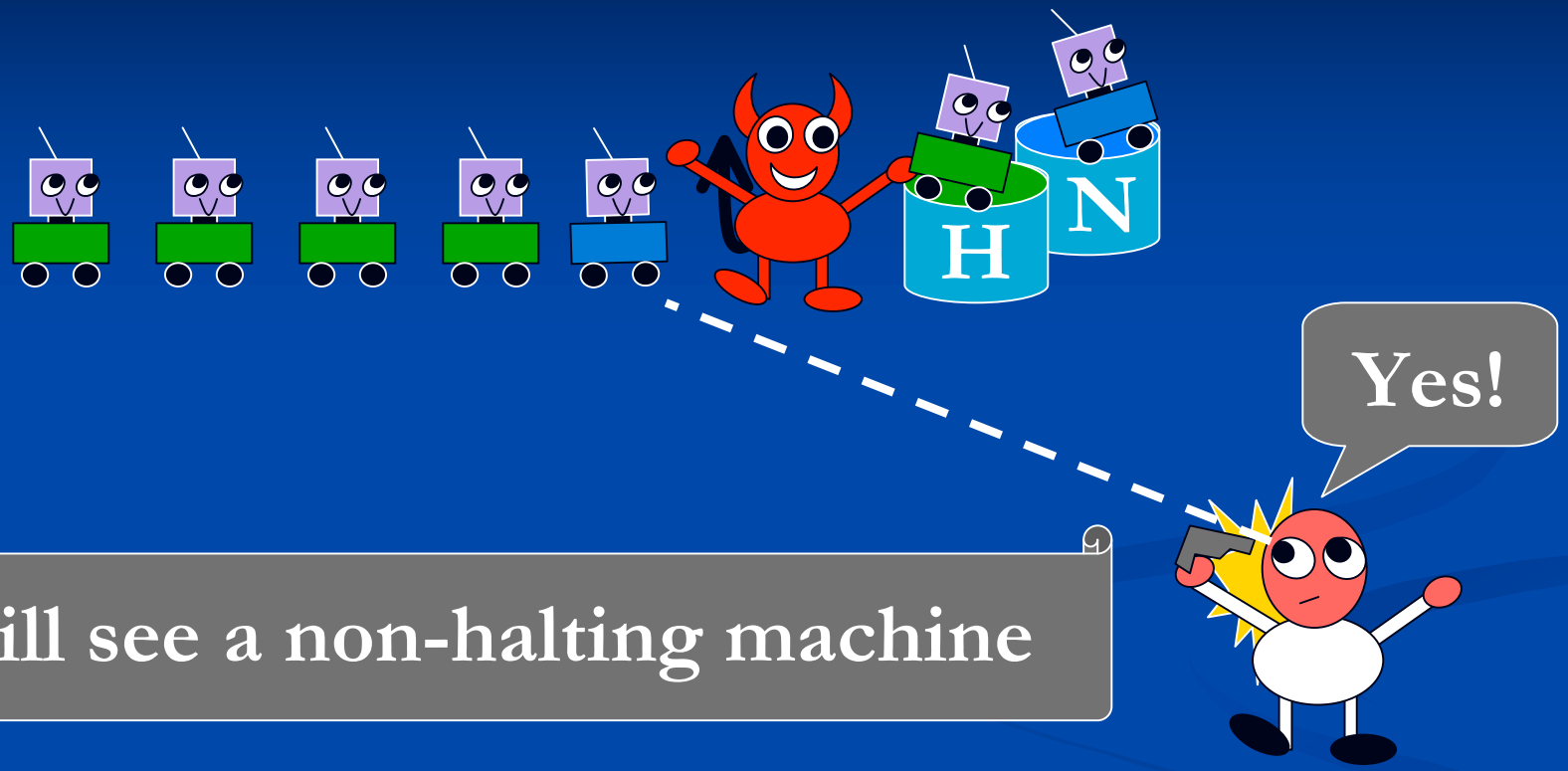


Rice-Shapiro Theorem

…could also have been verified by a behaviorist computer who empirically studies only the arbitrary computer's input-output behavior!



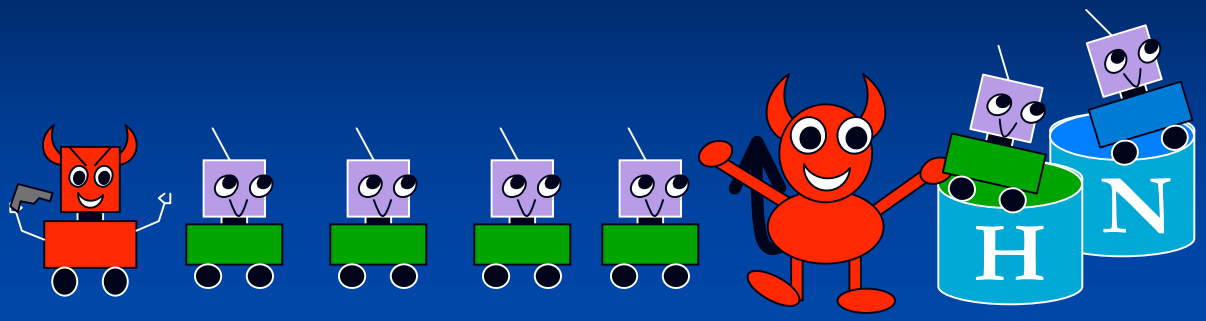
Uncomputable Induction



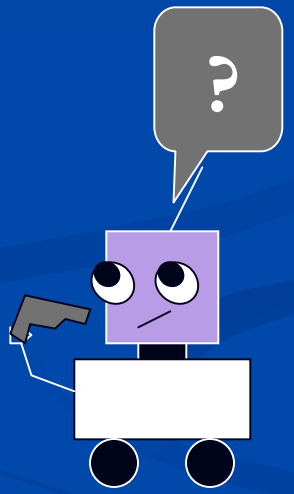
Will see a non-halting machine

Verifiable by “ideal agent”.

Uncomputable Induction



Will see a non-halting machine



Not verifiable by computable agent.

Similarity

Halting Problem	Universal Law
Unverifiable	Unverifiable
Demon can fool **computable** agent	Demon can fool **ideal** agent
Answer runs beyond **formal** experience	Answer runs beyond **empirical** experience

A Difference

Halting Problem	Universal Law
Input ends	Input never ends
Can handle some examples a priori… "Internal" problem of induction eventually catches up!	Problem of induction

Give and Take

Formal input	Empirical input
Jaw Breaker	Noodle
Fits in mouth but may never melt	May never fit in mouth
May never swallow	May never swallow

Another Difference

Formal Science	Empirical Science
Incompleteness	Problem of induction

Not Necessarily

Formal Science	Empirical Science
Incompleteness	Problem of induction
Add more powerful axiom	Conjecture a theory
Who knows if new axiom is consistent with old?	Who knows if theory is consistent with data?
Assume it is, keep checking, and hope	Assume it is, keep checking, and hope

Thesis

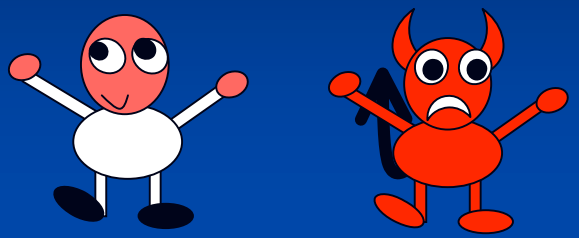
The problem of induction and the problem of uncomputability are essentially similar.

So the two should be **understood similarly**.

From the **ground upward**.

A House Divided

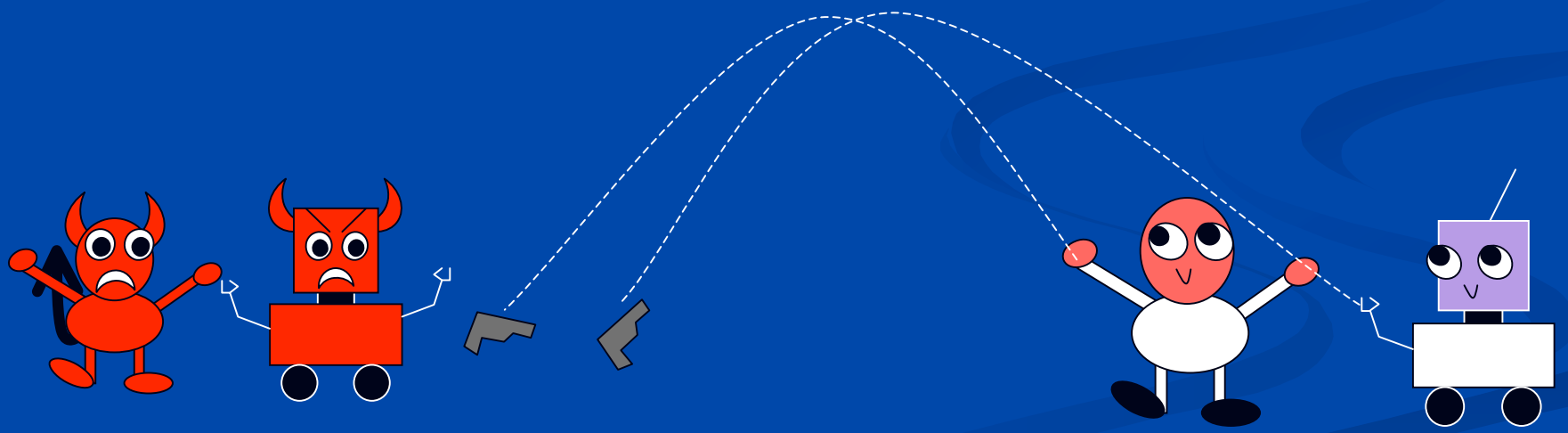
- **Unverifiability**
  - "Confirmation"
  - "Support"
  - "Coherence"
  - "Rational" update, etc.



- **Uncomputability.**
  - Suck it up
  - Logic vs. engineering
  - Try to approximate, etc.

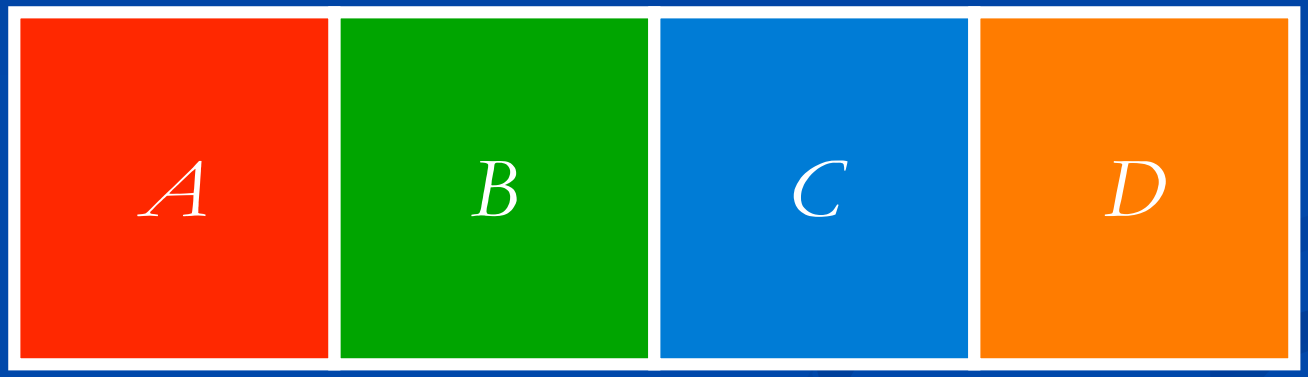
Unified Approach: Gun Control

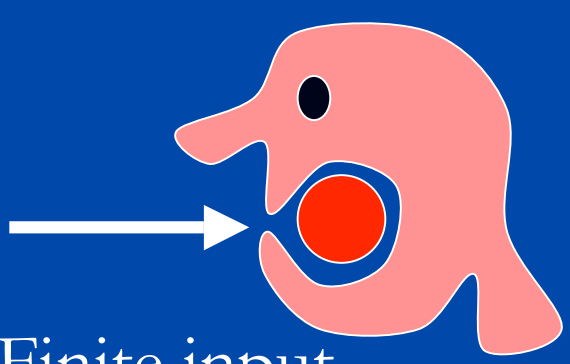
- Find the truth in the best feasible sense.
- So drop the halting requirement if infeasible.



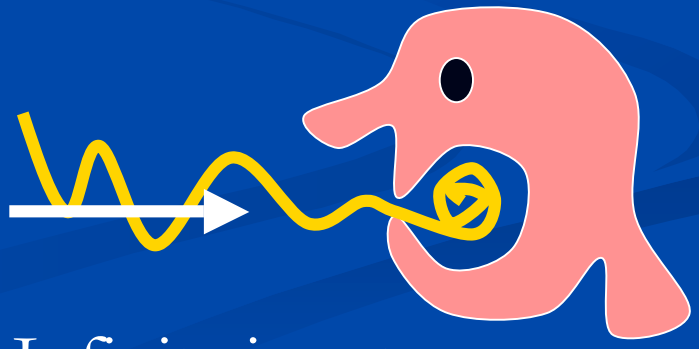
Start with Problems, Not Methods

Partition Q over set K of inputs





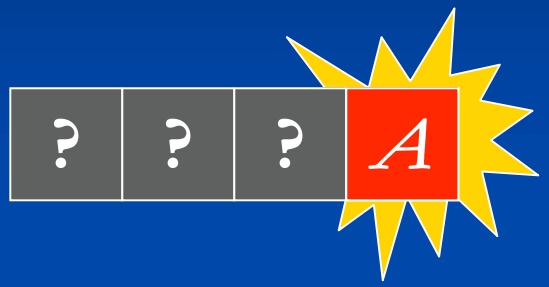
Finite input



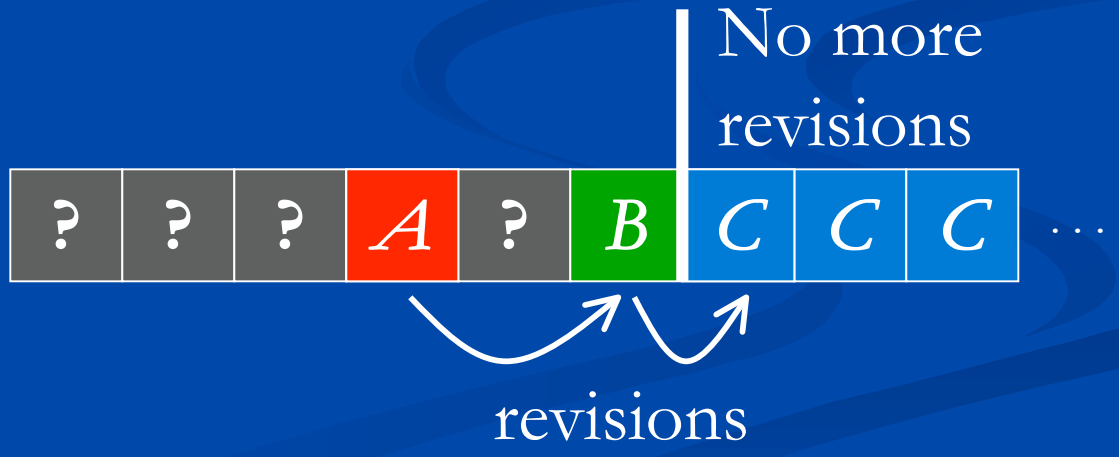
Infinite input

Convergence

Convergence with halting

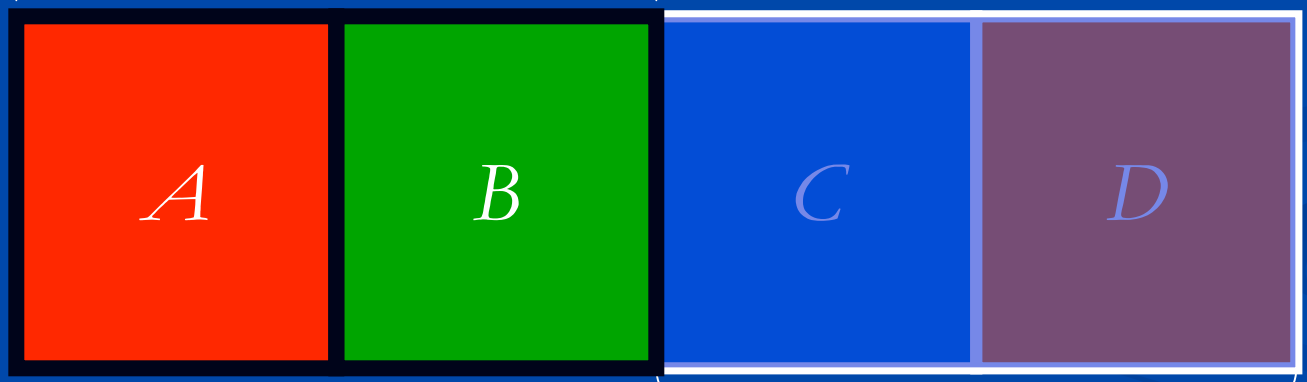


Convergence in limit
(in limit)



Success

Converge to right answer

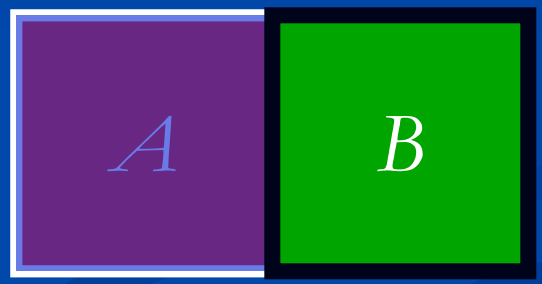


Don't converge to wrong answer

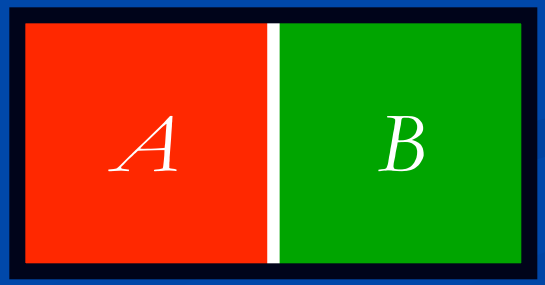
Special Cases

Verify A

Refute A

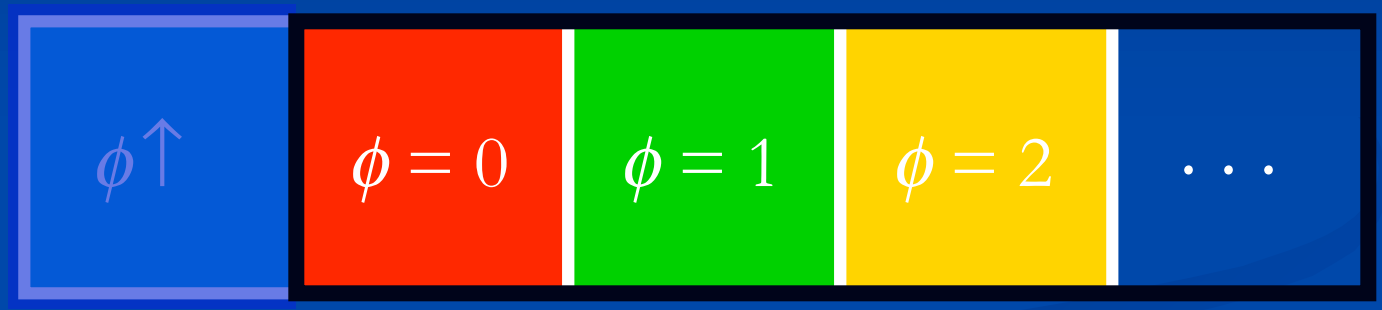


Decide A



Special Cases

Compute partial ϕ



Theory selection

Solutions and Optimality

- **Solution:** method that succeeds on each input.
- **Optimal solution:** solves in best possible sense
- **Solvable problem**: has solution.
- **Problem complexity:** best sense of solvability

Halting Bounds Revisions

Verifiability ≡

convergence with 1 revision ending with “no”.

Refutability ≡

convergence with 1 revision ending with “yes”.

Decidability ≡

convergence with 0 revisions.

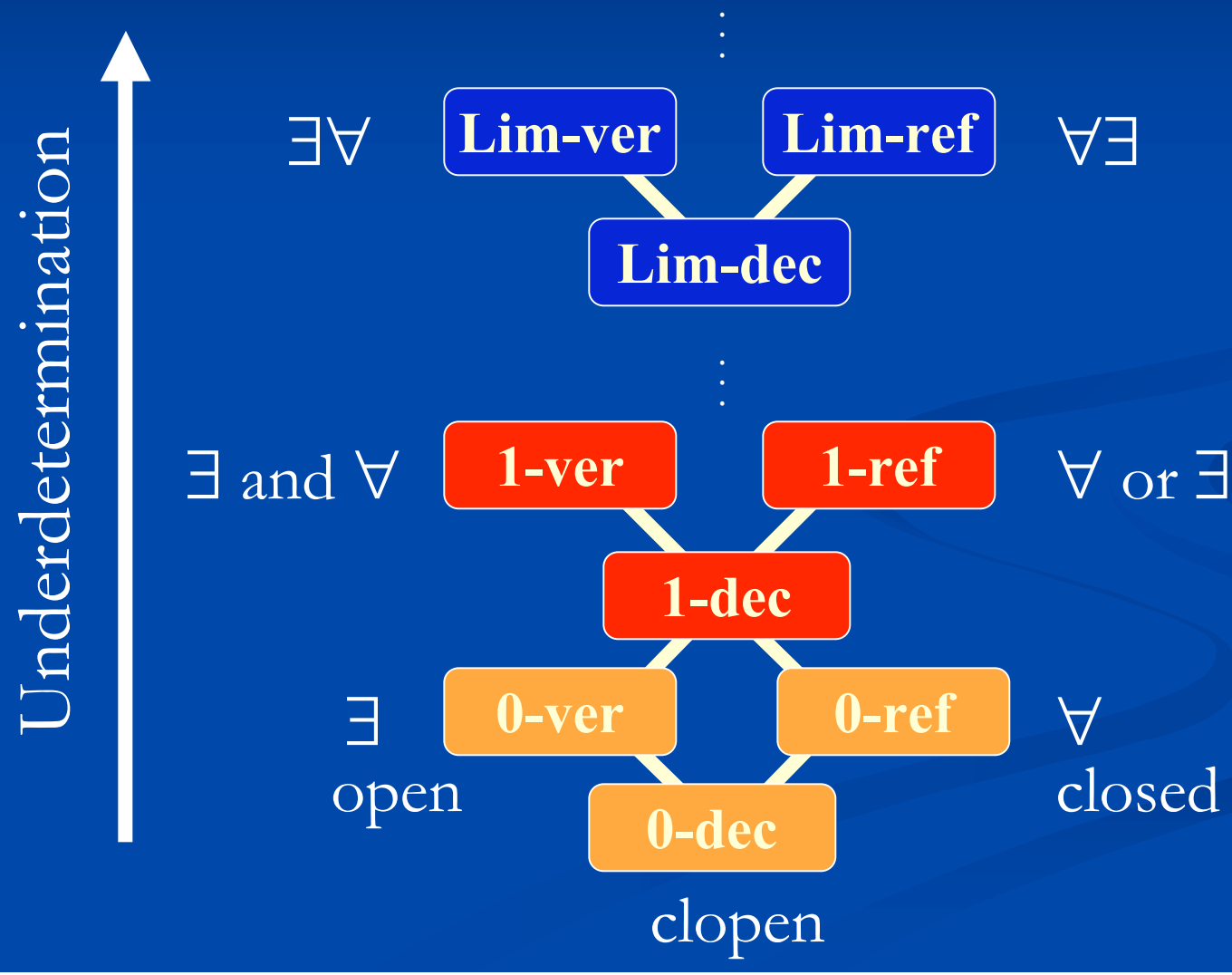
Generalization to *n* Revisions

***n*-Refutation =**
n*+1 rev ending with -*A

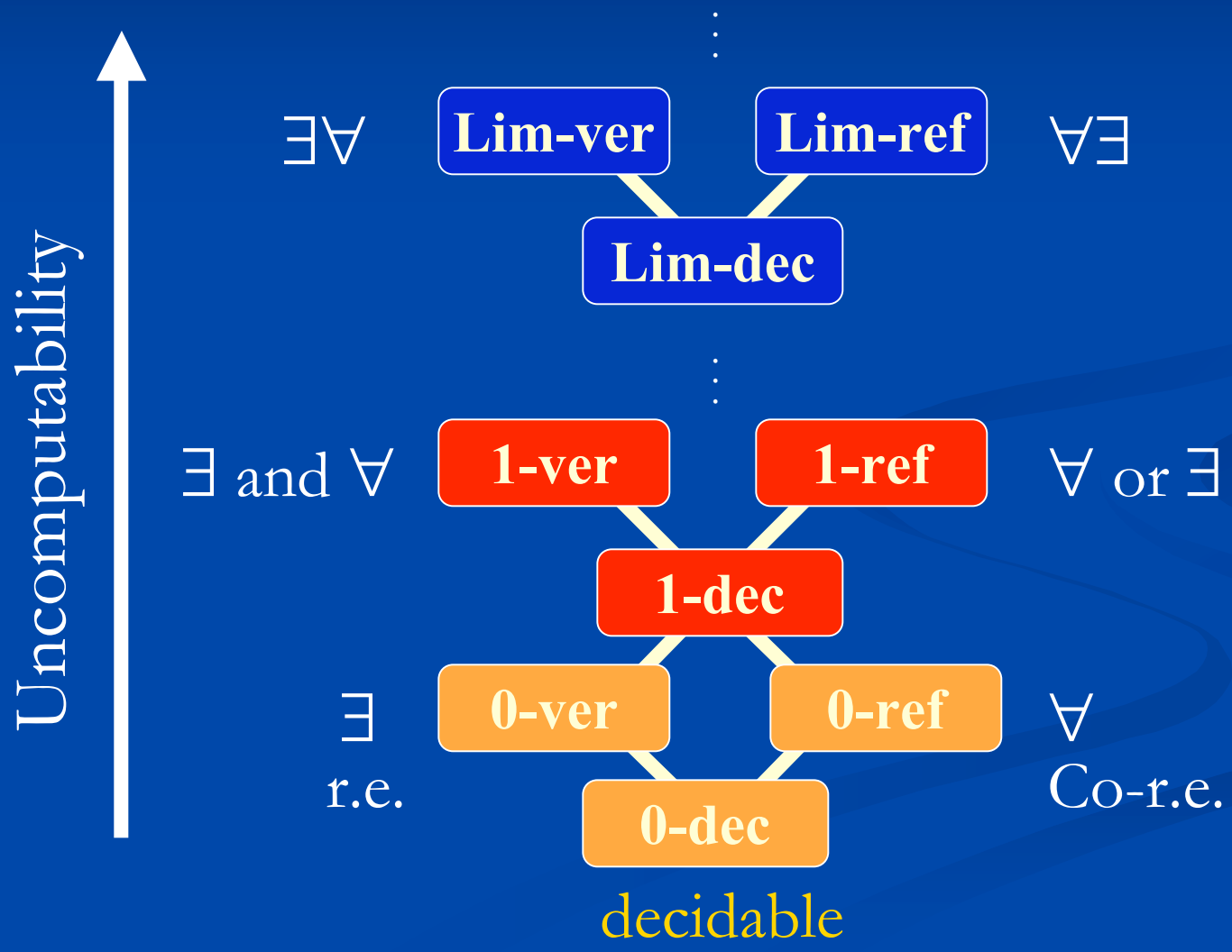
***n*-Verification =**
n*+1 rev ending with *A

***n*-Decision =**
***n* rev**

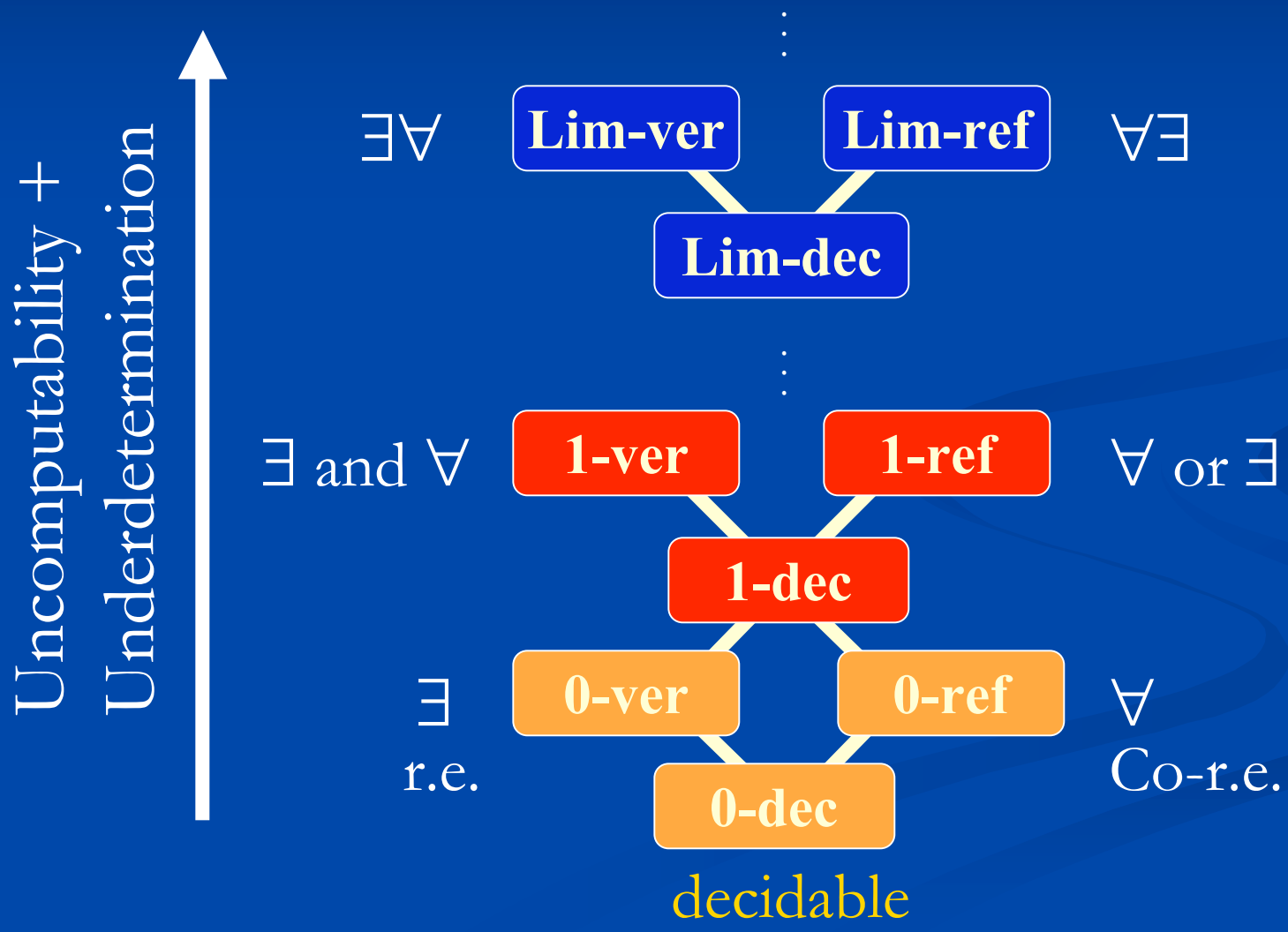
Empirical Complexity



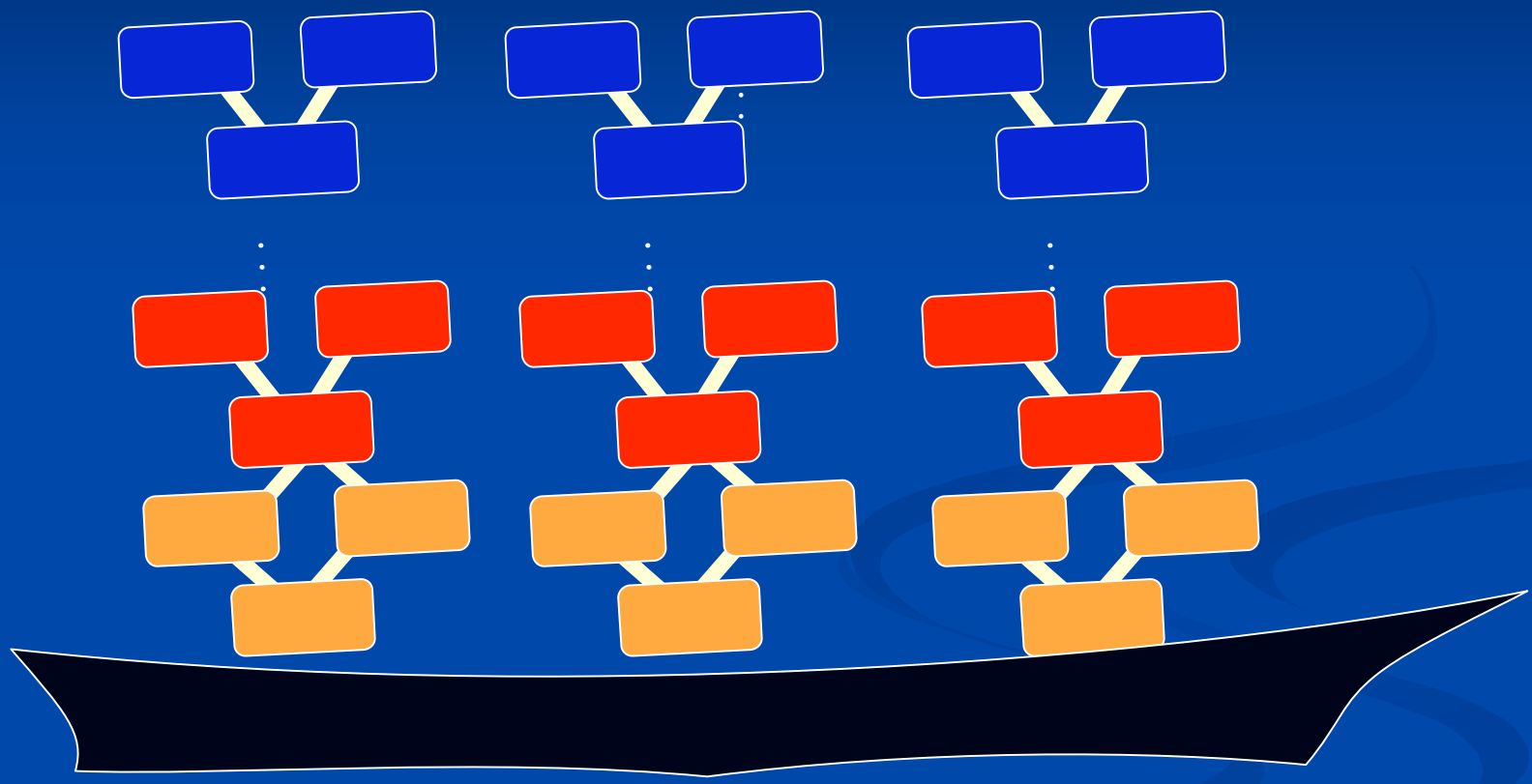
Formal Complexity



Formal + Empirical Complexity



Ship-shape Epistemology

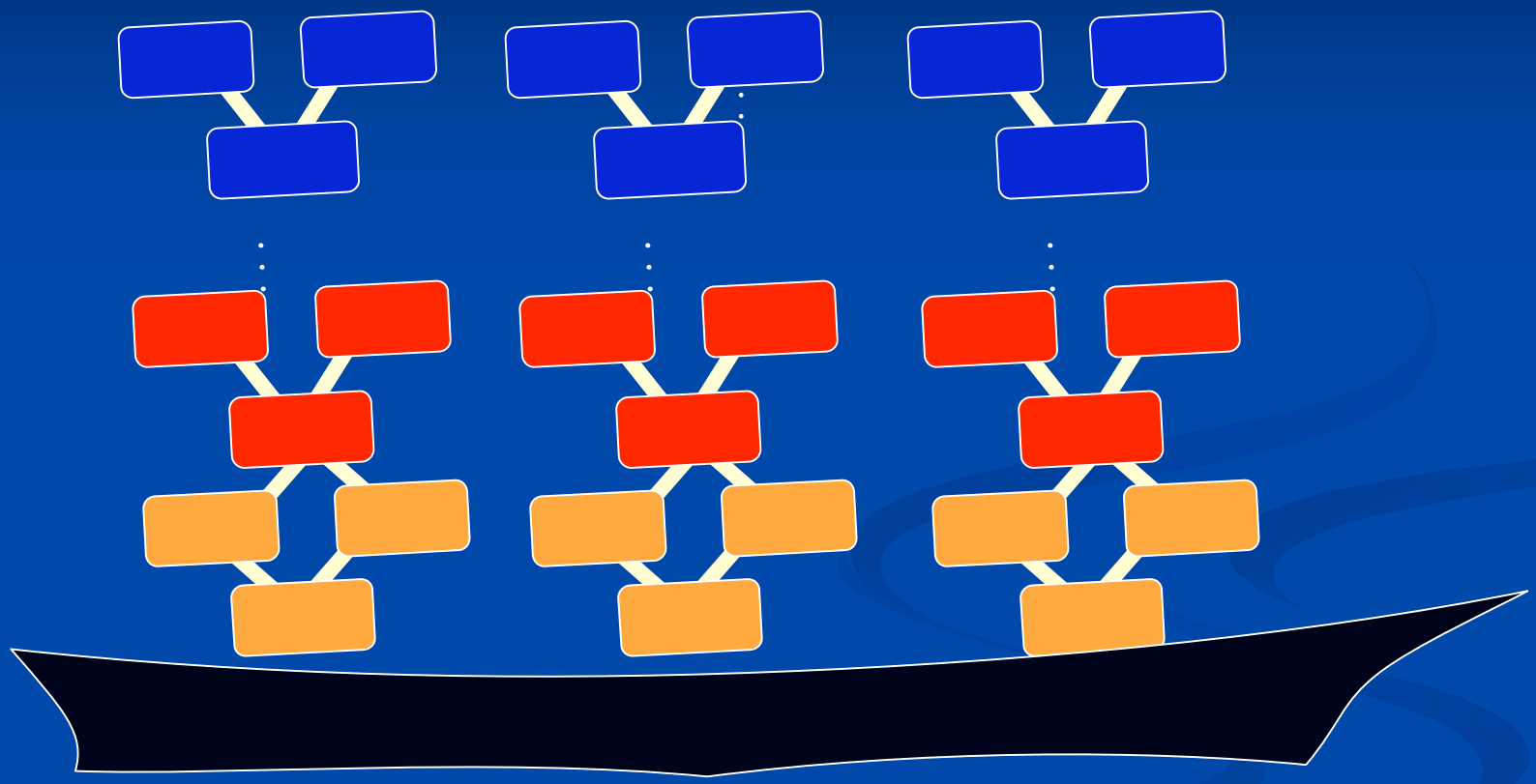


Ideal empirical

Effective empirical

Effective formal

Ship-shape Epistemology

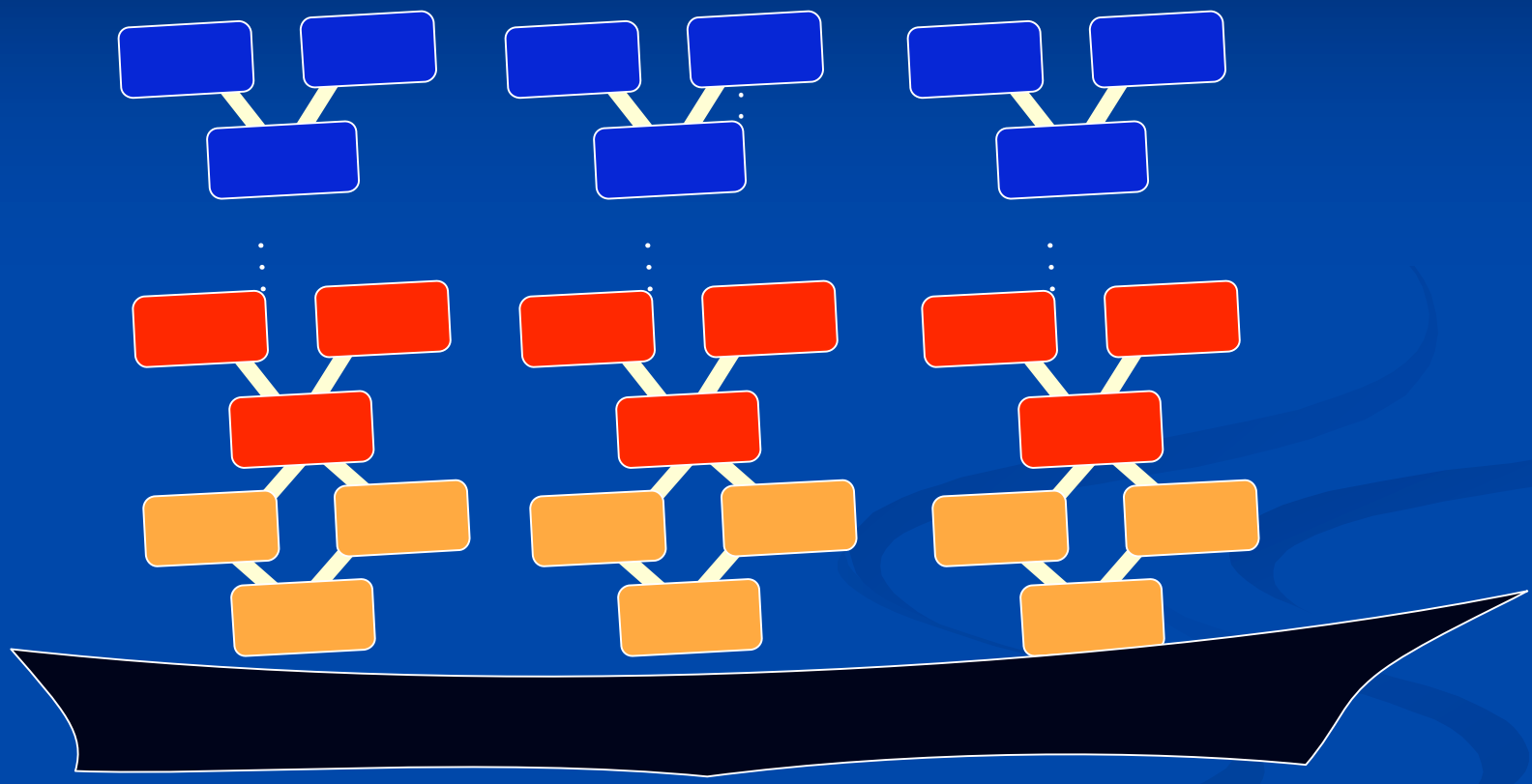


Topological invariants

Recursive invariants

Recursive invariants

Ship-shape Epistemology

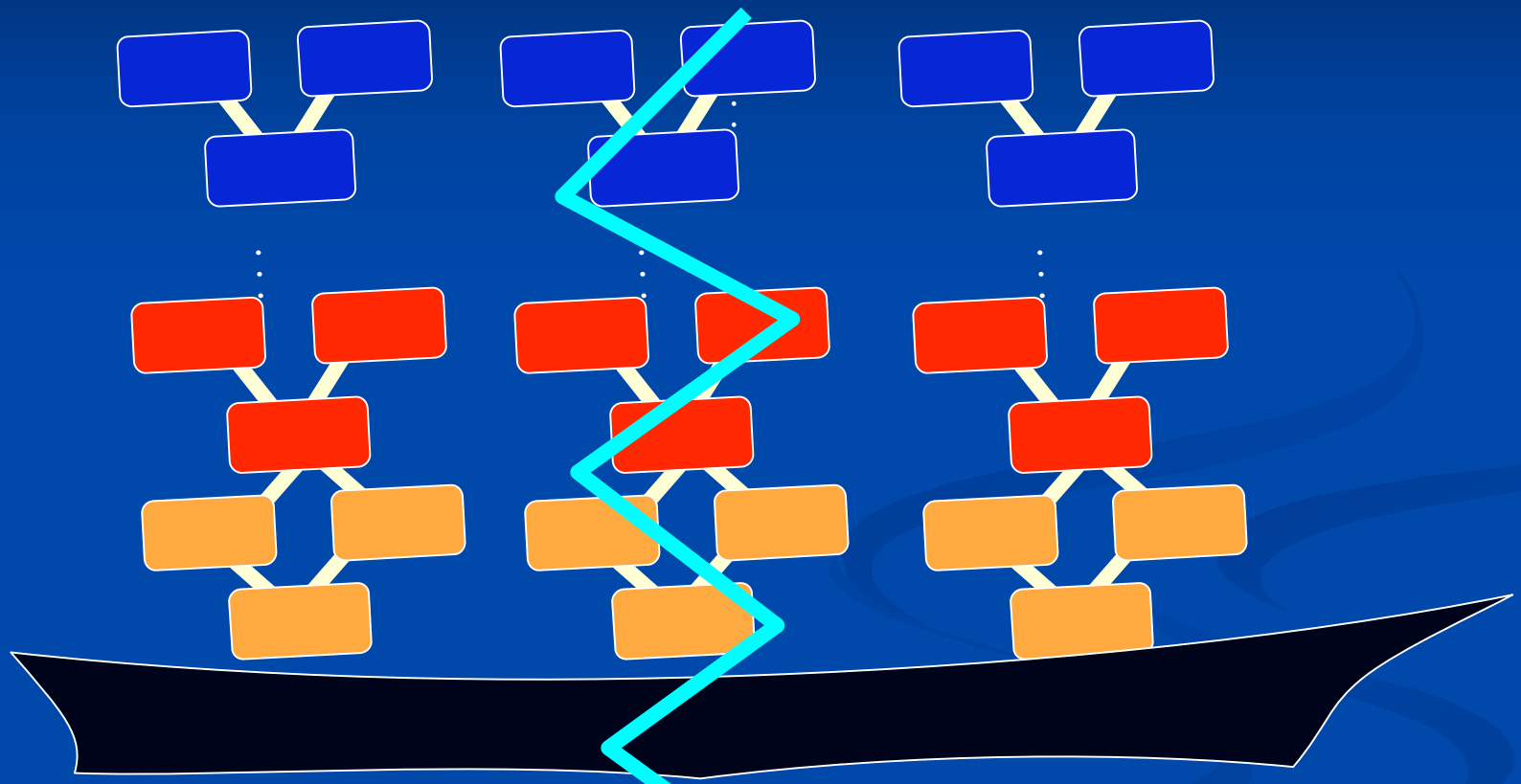


Combination

Lower Bound

Lower Bound

Messy, Odd Distinction

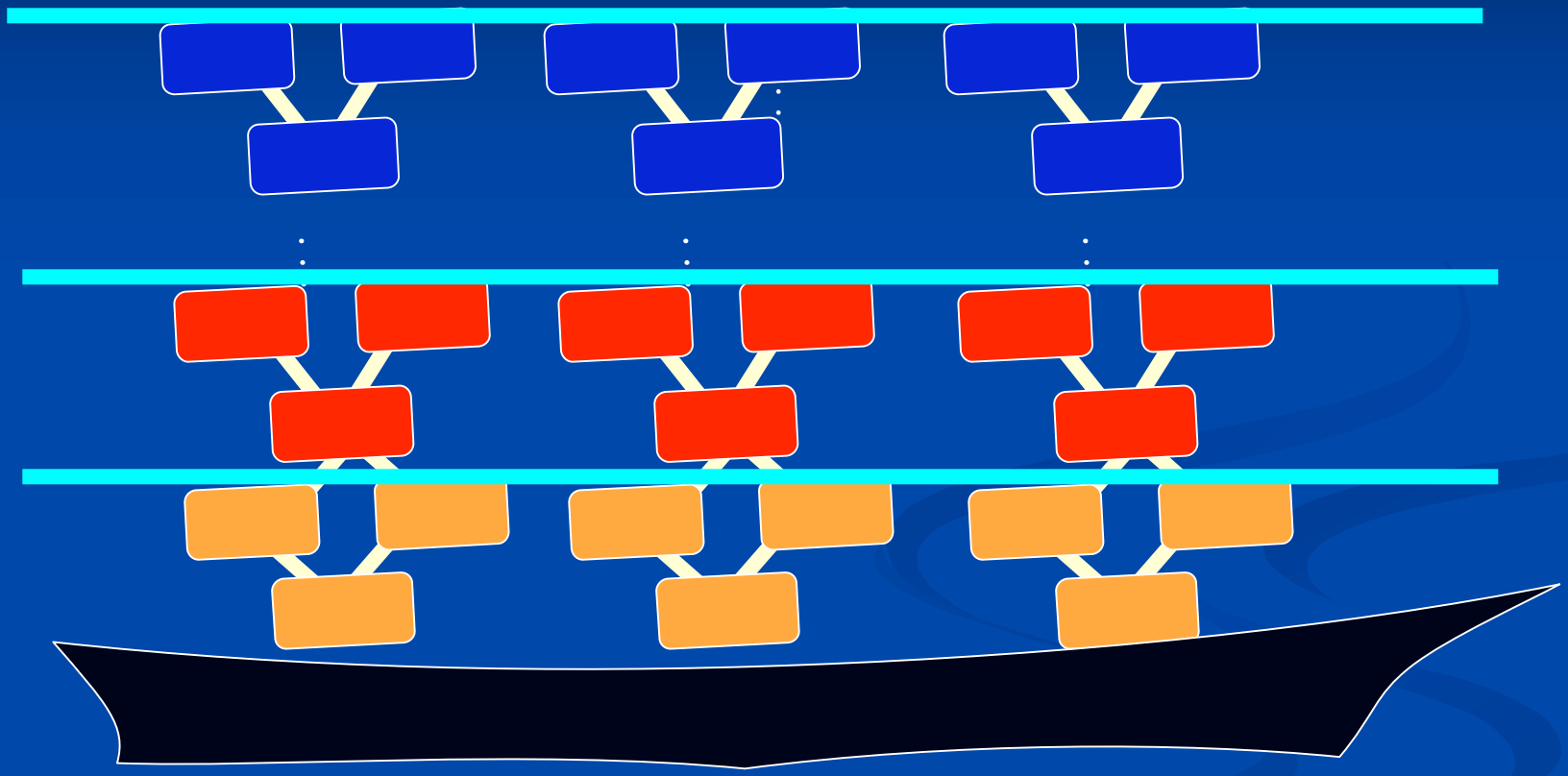


Empirical
Borel

Mixed
effective
Borel

Formal
arithmetical

Neat, Natural Distinctions

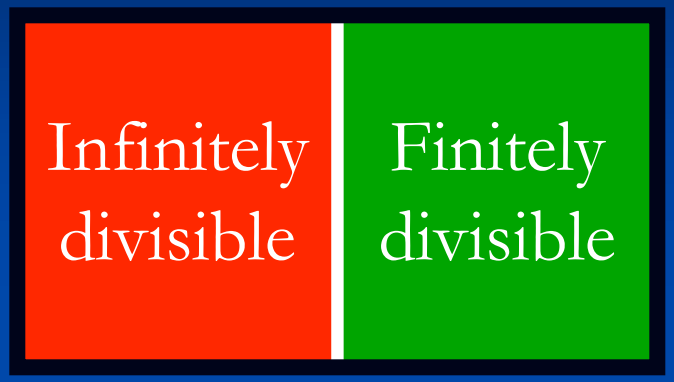


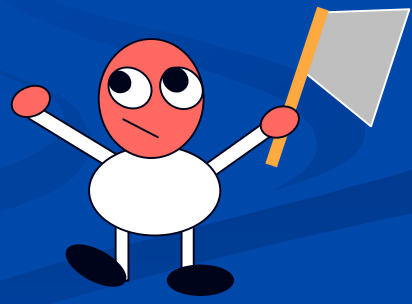
Empirical
Borel

Mixed
effective
Borel

Formal
arithmetical

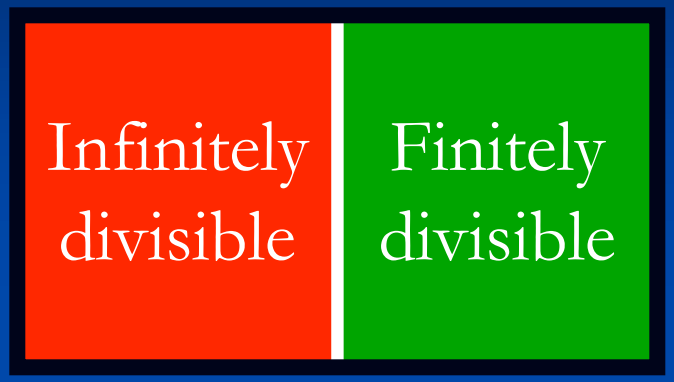
Example: Kantian Antinomy



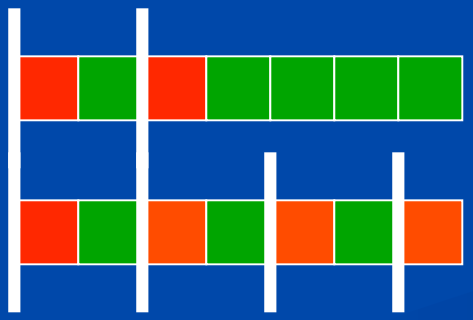


Upper Complexity Bound

Lim-ref



Lim-ver

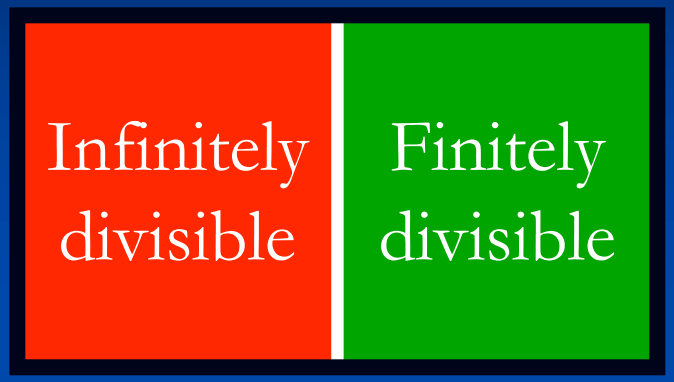


… fin → ↓fin

… inf → ↑fin

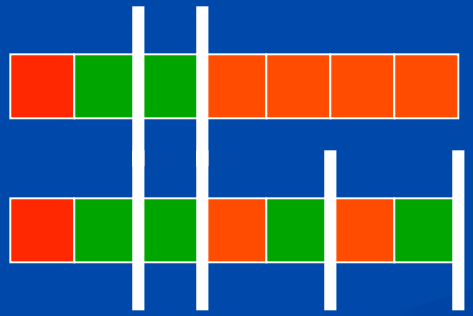
Lower Complexity Bound

Lim-ref
-Lim-ver



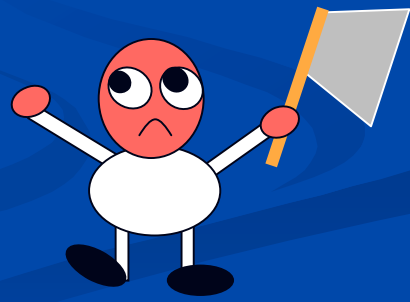
Lim-ver
-Lim-ref

I let you cut when you say **fin.**



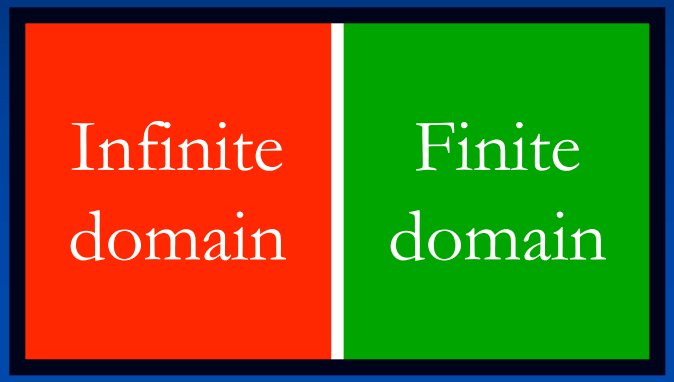
… ↓ **inf** → **fin**

… ↑ **inf** → **inf**

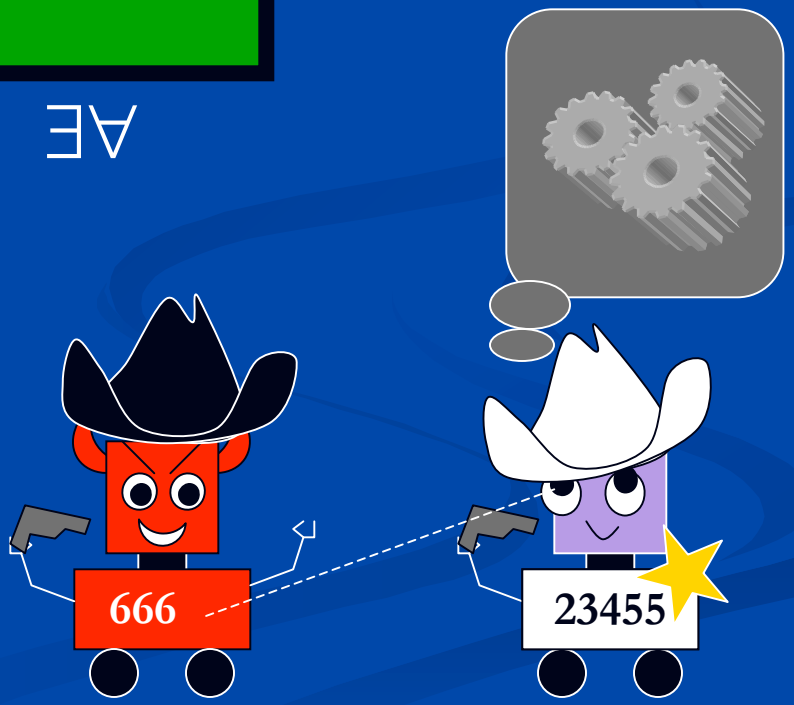


Purely Formal Analogue

Lim-ref
-Lim-ver

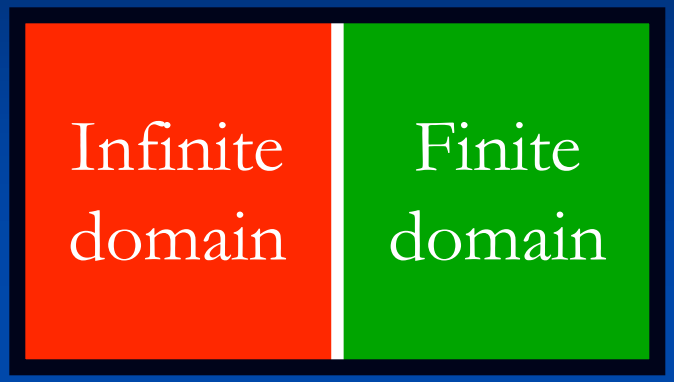


Lim-ver
-Lim-ref



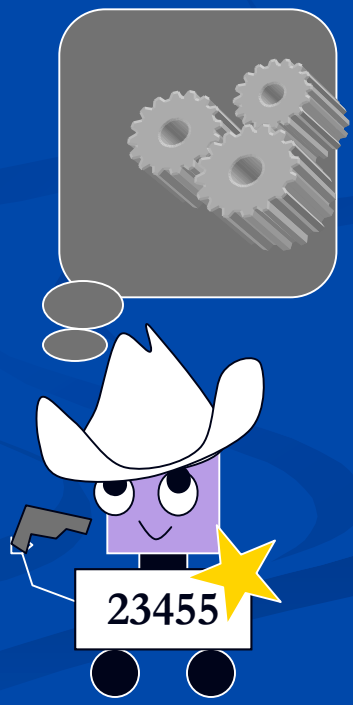
Purely Formal Analogue

Lim-ref
-Lim-ver

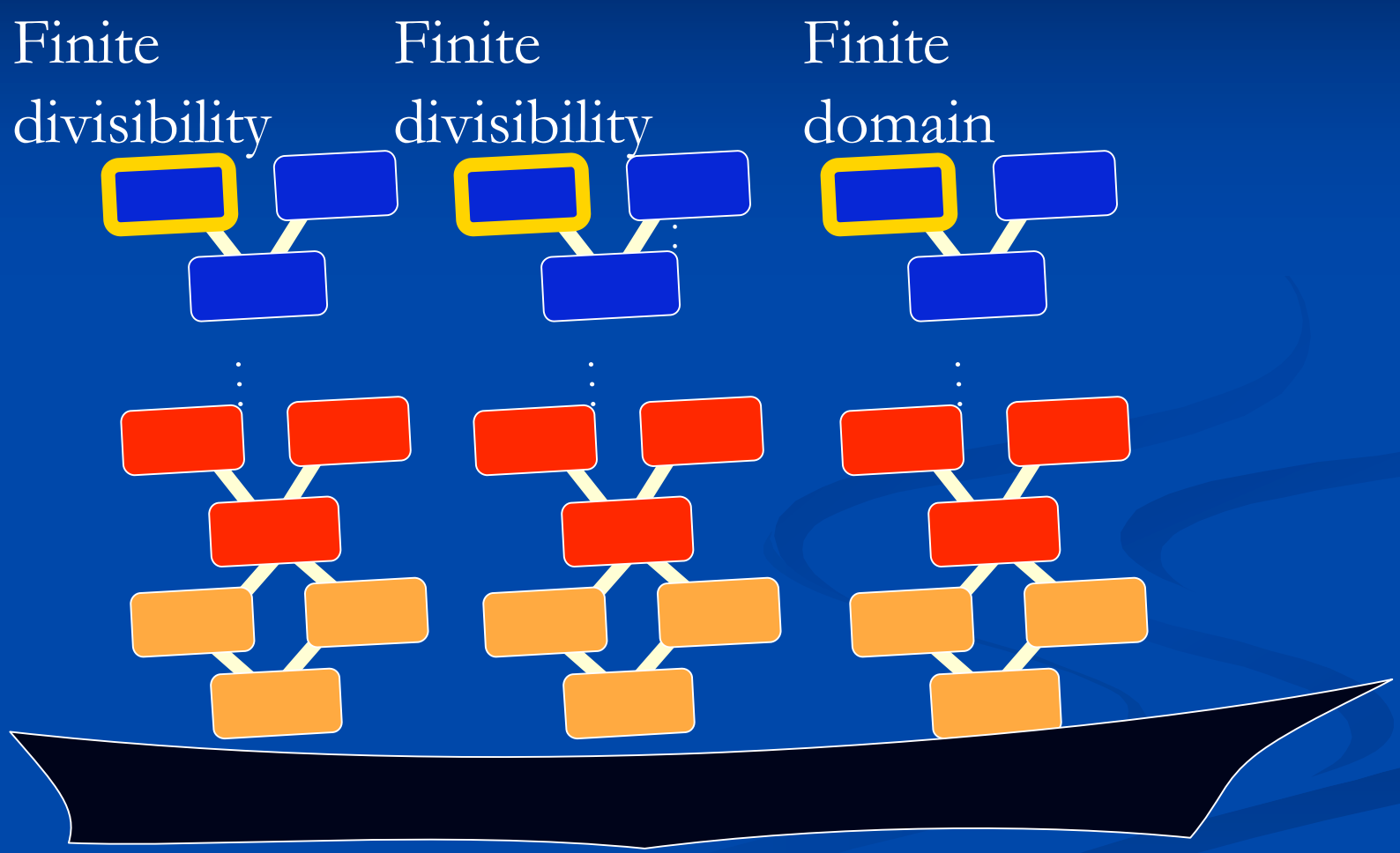


Lim-ver
-Lim-ref

I halt on another input each time he says **fin** in my opry oary simulation of him ganderin' at my program.



Analogies


Finite divisibility
Finite divisibility
Finite domain
Ideal empirical
Effective empirical
Effective formal

Mixed Example: Penrose

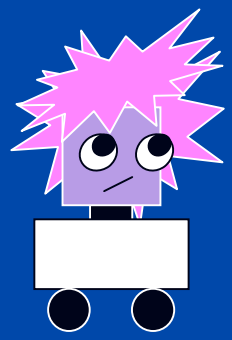
Computable	Uncomputable

Human subject Cognitive scientist

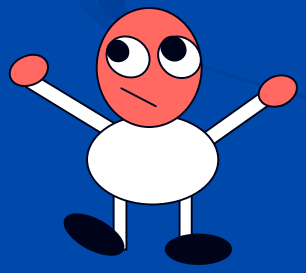
Empirical Irony

Computable	Uncomputable

You can verify human computability in the limit…



Human subject

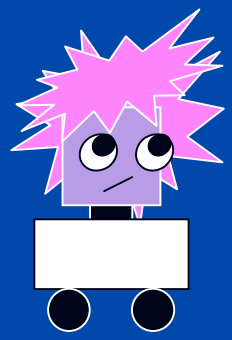


Cognitive scientist

Empirical Irony

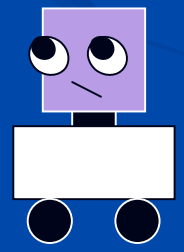
Computable	Uncomputable

But only if you aren't computable!



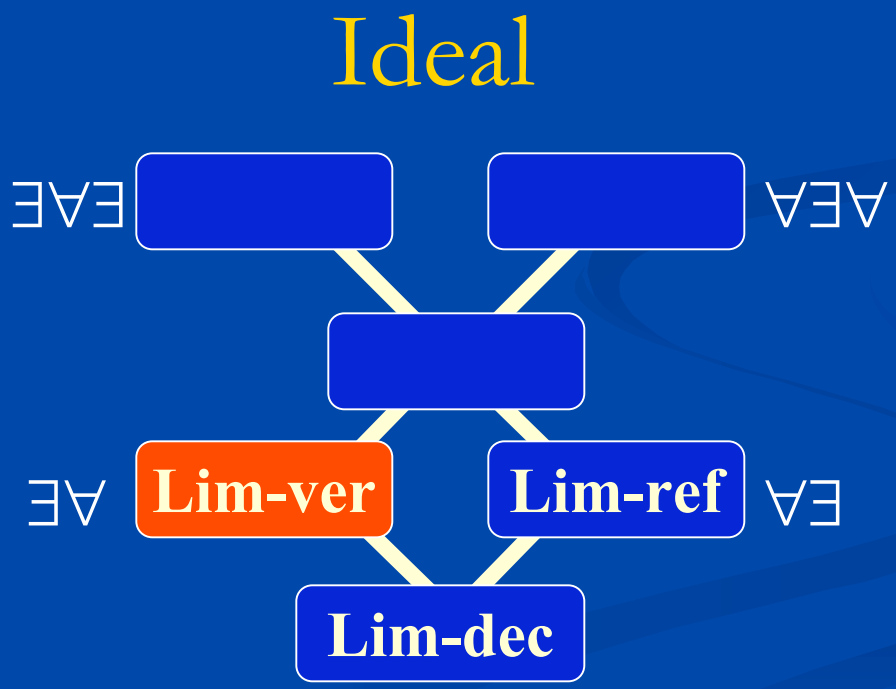
Human subject

UCUCUCCUCCU…



Cognitive scientist

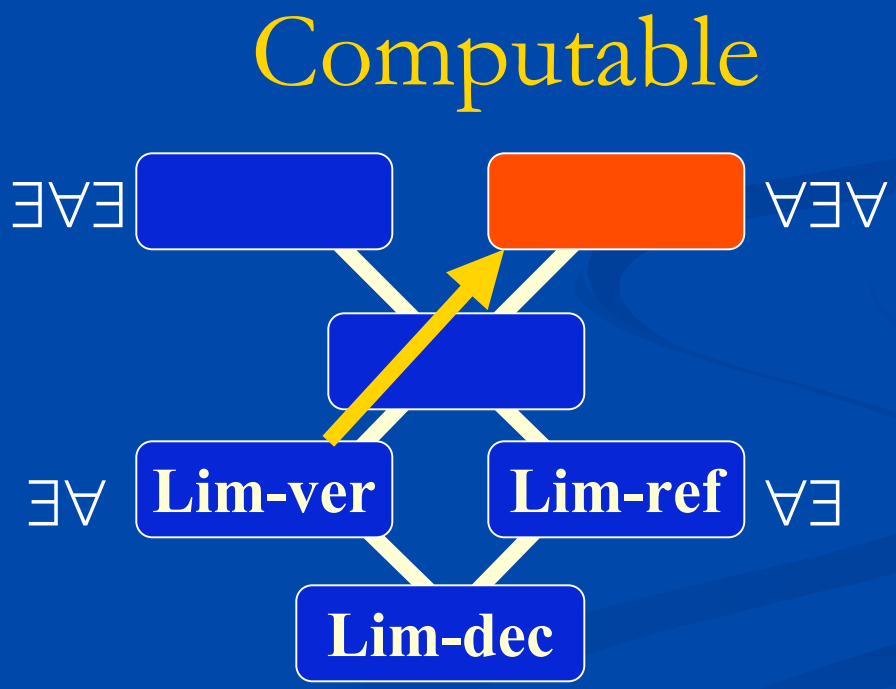
Computability



Computability

We can verify our computability in the limit…

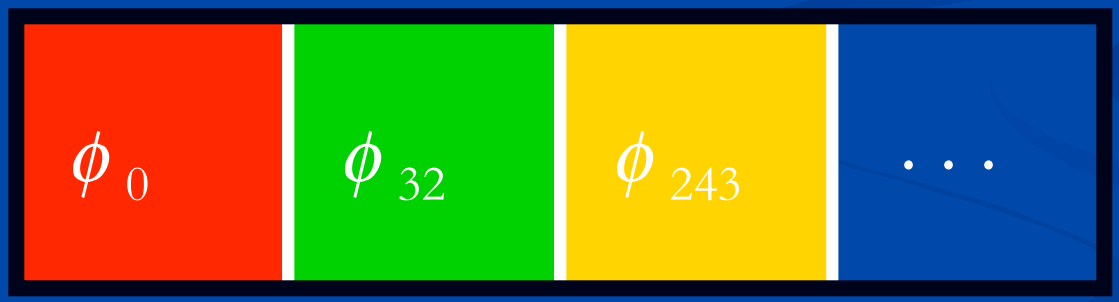
only if we are not computable!



Gold/Putnam

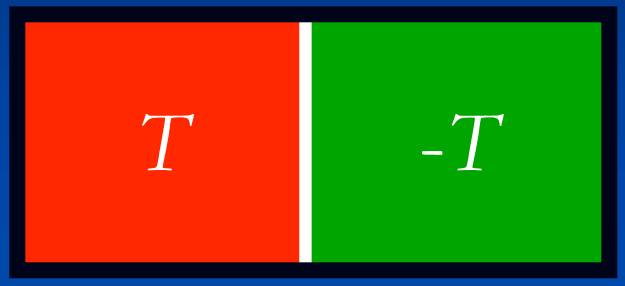
Even assuming computability,
you can converge to the true computable behavior only if you are not computable!

Total computable functions



Uncomputable Predictions

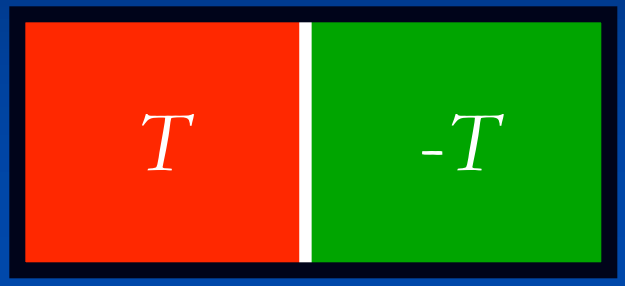
(with Oliver Schulte)



- There exists *T* such that
  - *T* makes a unique prediction at each stage;
  - The predictions are very uncomputable…

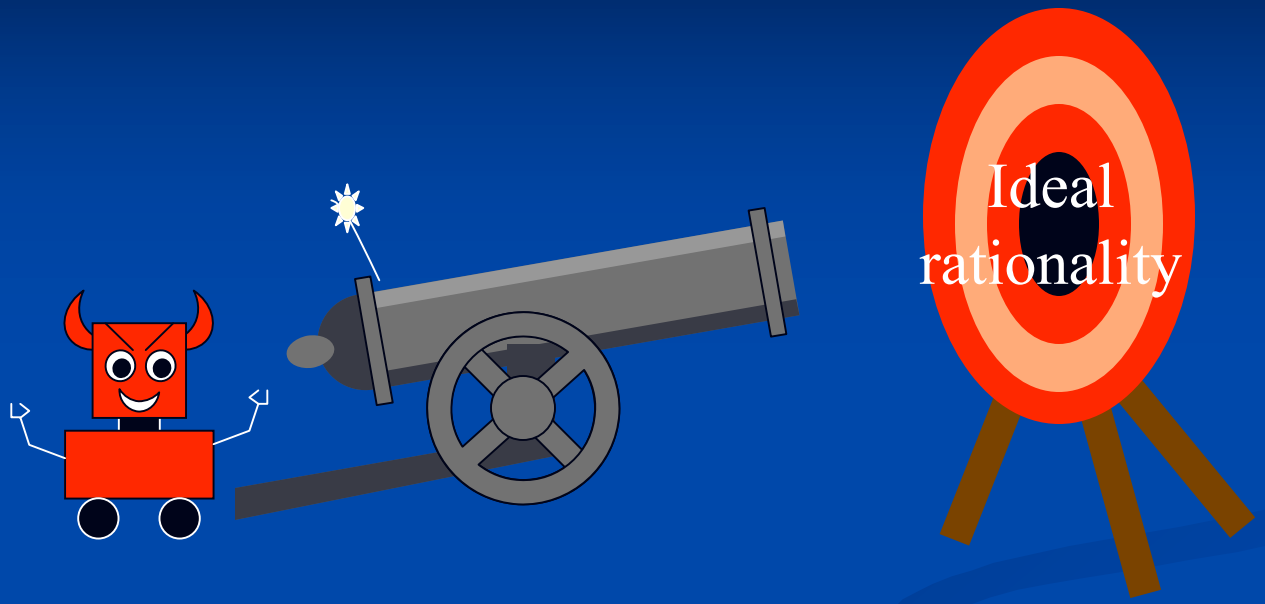
Uncomputable Predictions

(with Oliver Schulte)



- There exists *T* such that
  - *T* makes a unique prediction at each stage;
  - The predictions are very uncomputable…
  - But T is refutable by a computable method!

Moral 1: Rational Hobgoblins

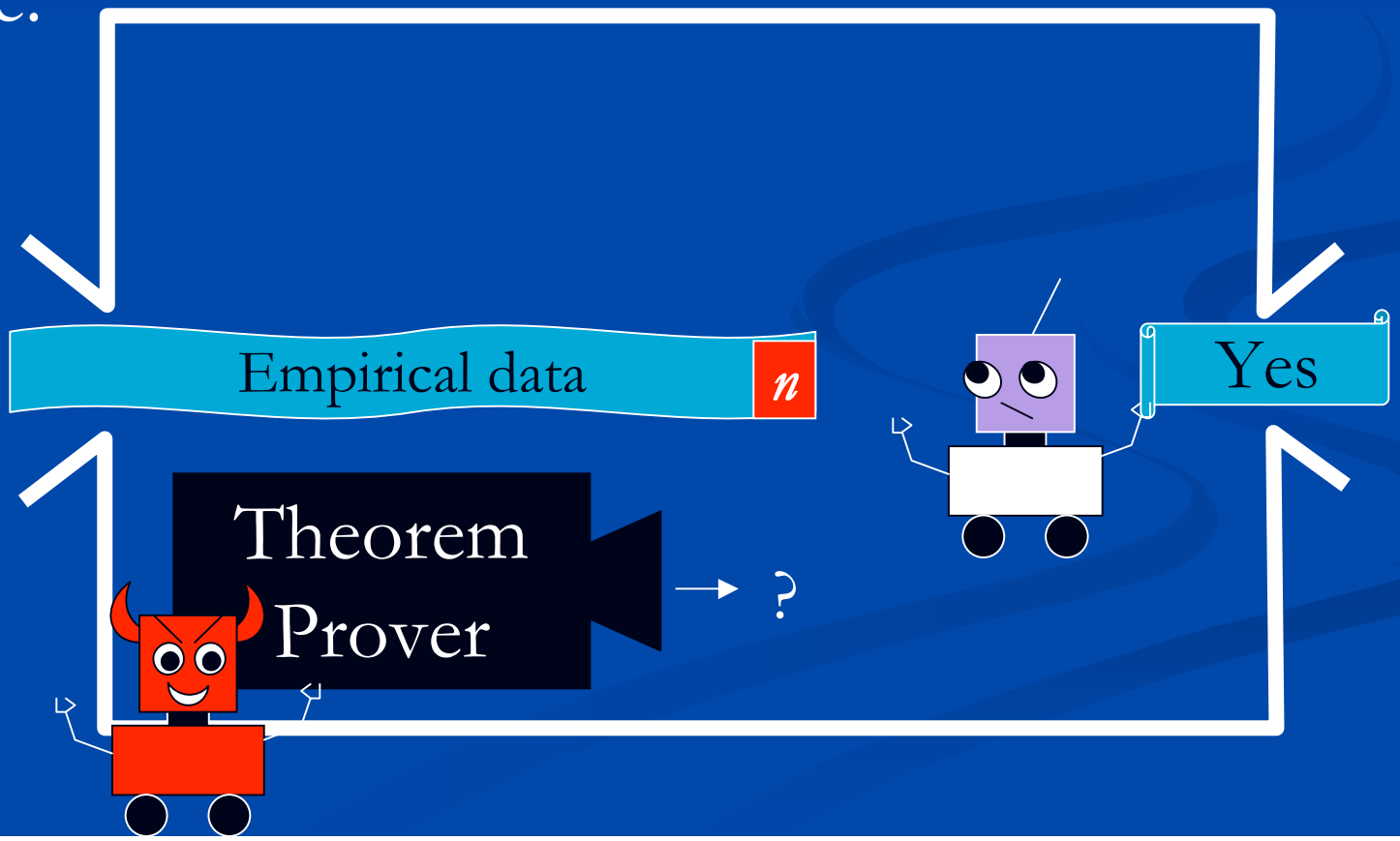


No	consistent	computable	convergent	method.
Exists	inconsistent	computable	refuting	method.
Exists	consistent	computable	non-convergent	method.

So "a foolish consistency" precludes truth-finding!

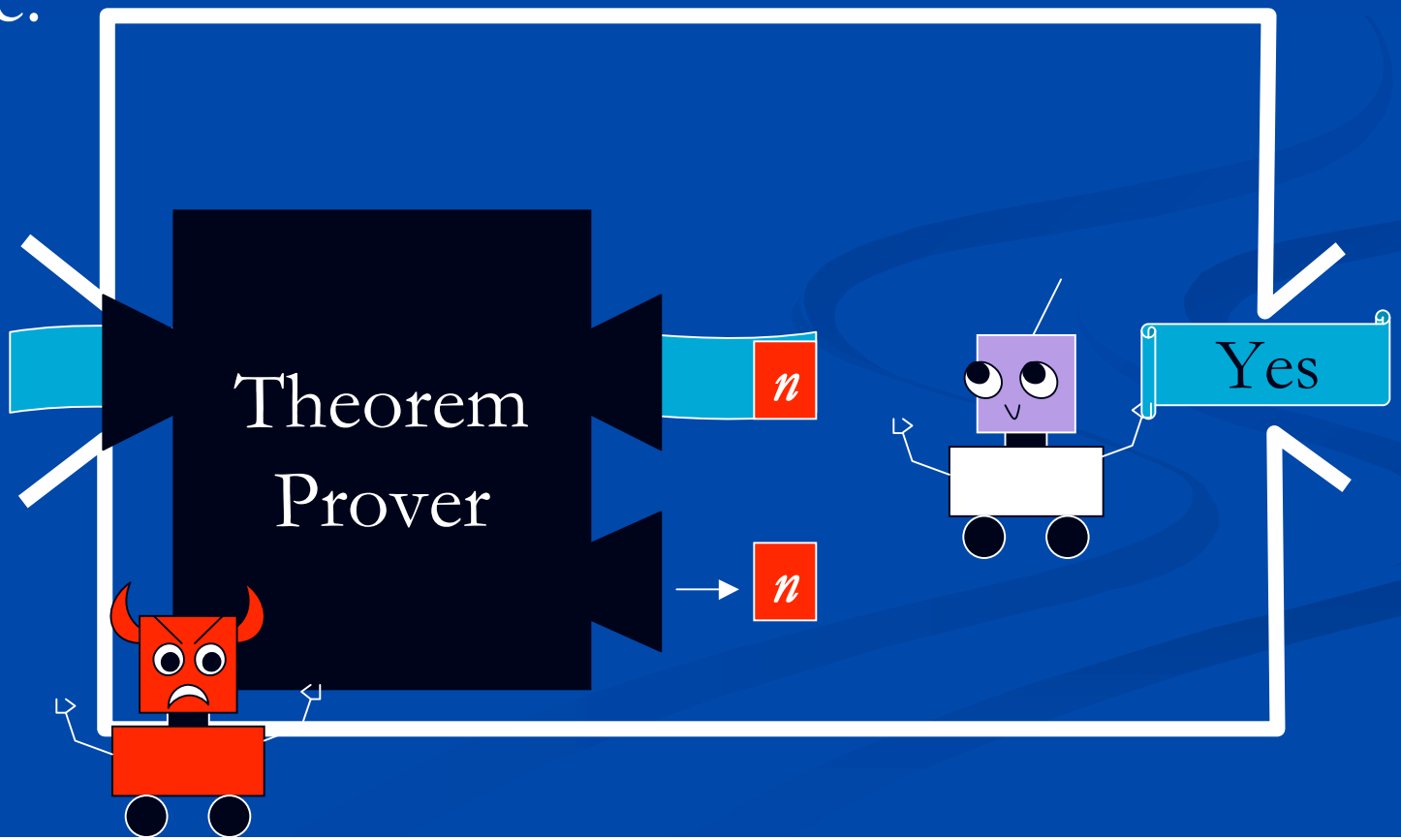
Moral 2: Coup de Grace

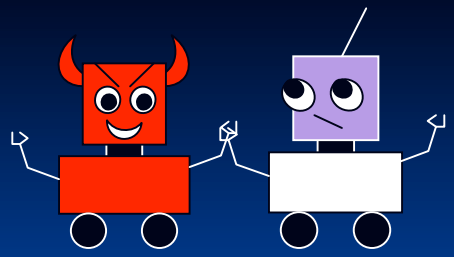
•The very idea of insulating deduction from empirical data restricts the truth-finding power of effective science.



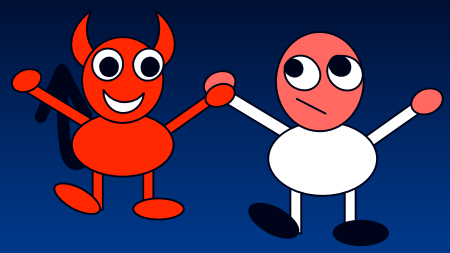
Moral 2: Coup de Grace

- The very idea of insulating deduction from empirical data restricts the truth-finding power of effective science.



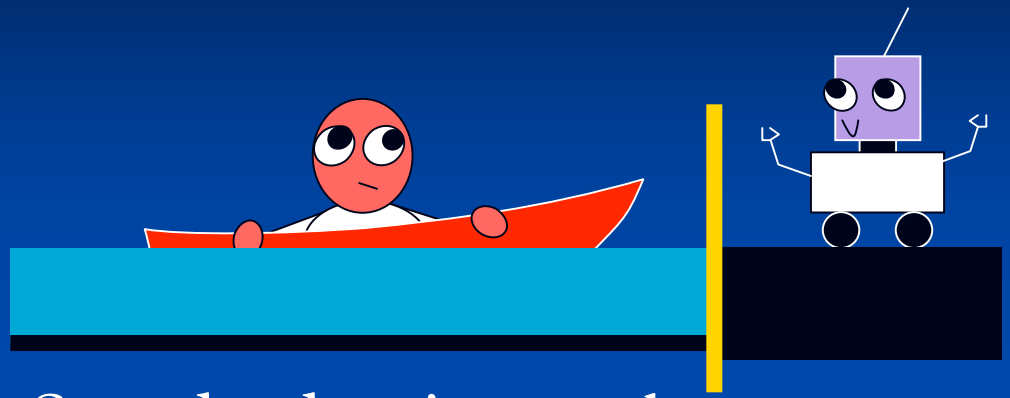


Conclusions

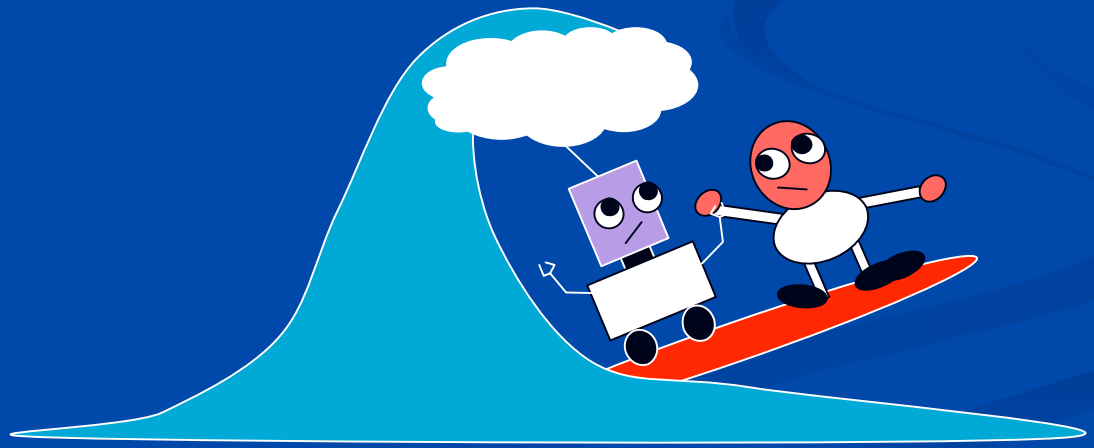


- Justification = truth-finding performance.
- Performance depends on problem complexity.
- Formal and empirical complexity are similar.
- Computational epistemology is unified
- Standard epistemology is divided.
- Insistence on division weakens truth-finding performance of effective science.

Closing Image



Standard epistemology

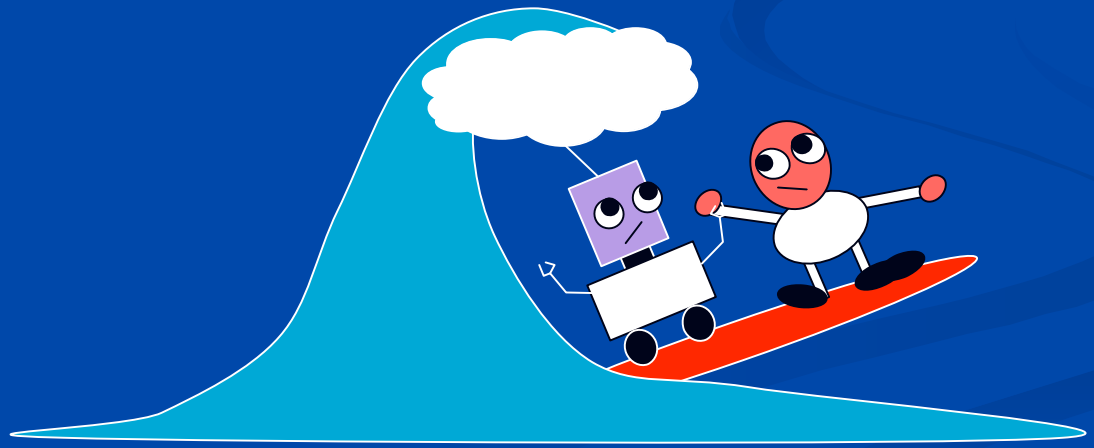


Computational epistemology

Closing Image

I'm rational

Standard epistemology

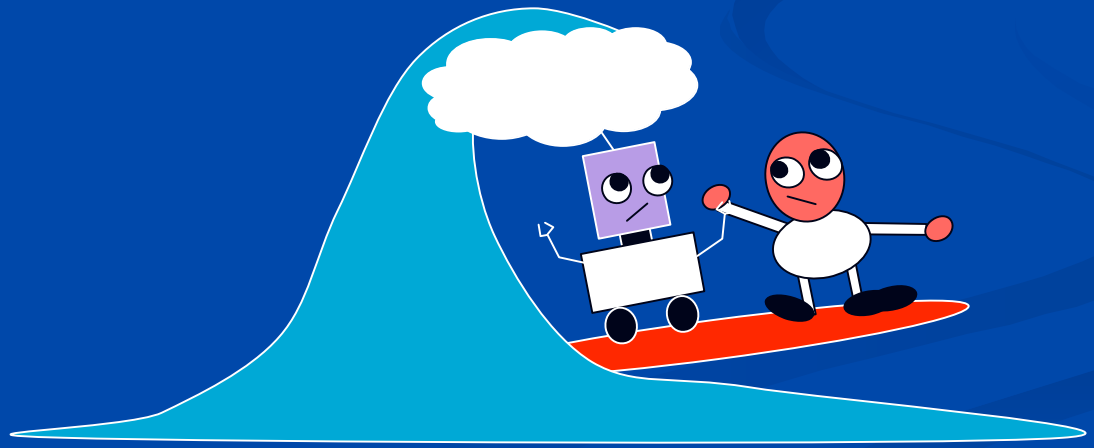


Computational epistemology

Closing Image

I'm rational

Standard epistemology



Computational epistemology

THE END