Verification of Autonomous Systems

Alessandro Pinto

United Technologies Research Center, Berkeley, CA

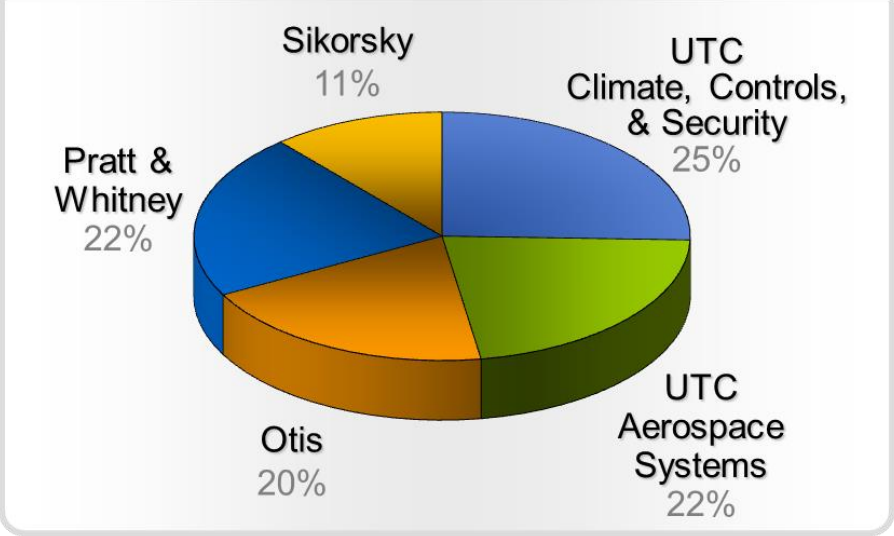
Outline

- Introduction
- Verification and autonomous systems
- Thesis: Verify what the system knows and understands rather than what the system does
- Our contribution in the DARPA ALIAS program

United Technologies Business Units

Otis

2014 Sales: $65.1 billion
Segments : 45% Commercial & Industrial, 55% Aerospace



UTC Climate,
Controls & Security

Pratt & Whitney

UTC Aerospace Systems

Applications

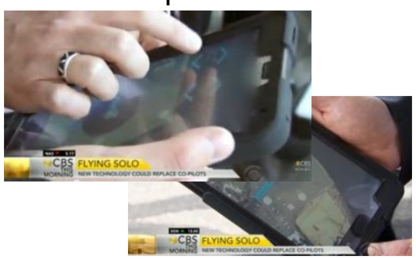
UTRC interest

Autonomous rotorcraft

SARA

Pilot

Field Operator



GCS

ISR missions

Source wikipedia

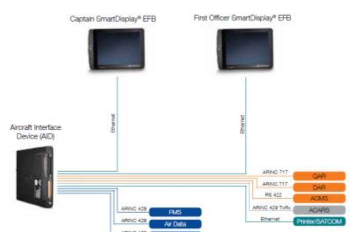
Source cloudcaptech.com

Source cloudcaptech.com

Source cloudcaptech.com

Source cloudcaptech.com

UTAS EFB



Source : utasaerospacesystems.com

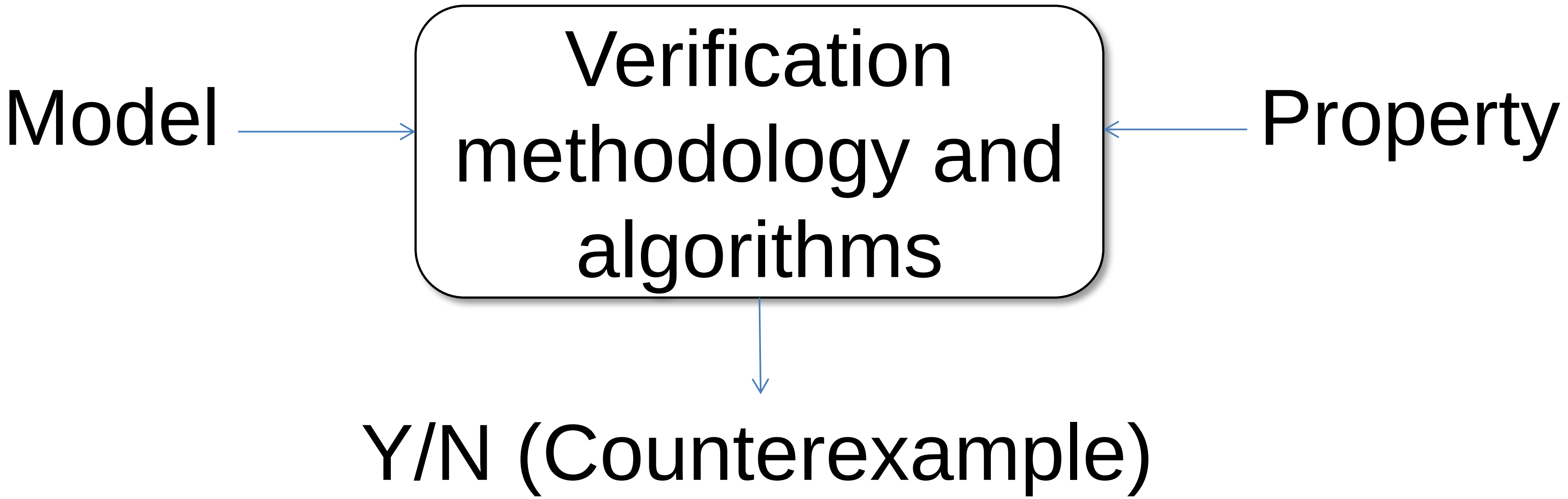
Intelligent Buildings

Source : [ccs.utc.com](ccs.utc.com)

A. Pinto, UTRC, pintoa@utrc.utc.com

Verification

General definition



Verification

The known case

Model
- Simulink
- Code
- …

Verification methodology and algorithm
- Model checking
- Testing
- …

Property
- Requirements
- Common sense
- …

Y/N (Counterexample)

Verification

Autonomous systems

Model?
(Environment?,
Perceptions?,
Decision Making?)

Verification methodology and algorithm?
(can't be define until we know Model and Property)

Property?
(Not designed to do one thing)

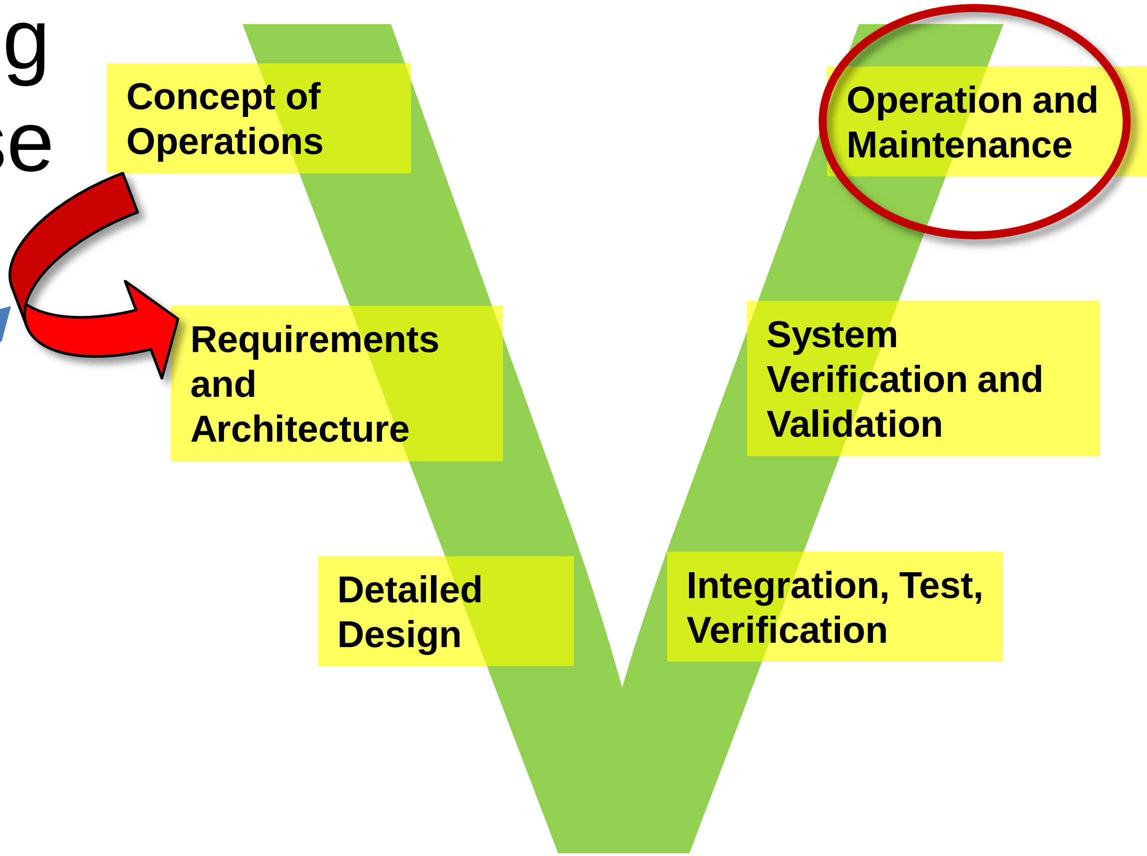
Y/N (Counterexample)

Traditional System Design

From CONOPS to Operations

Running system

Scenarios describing environment and use of the system



Experts + Systems engineers

Autonomous Systems

Operations on CONOPS

Mission objectives, constraints, relevant facts (function of time)

Concept of Operations

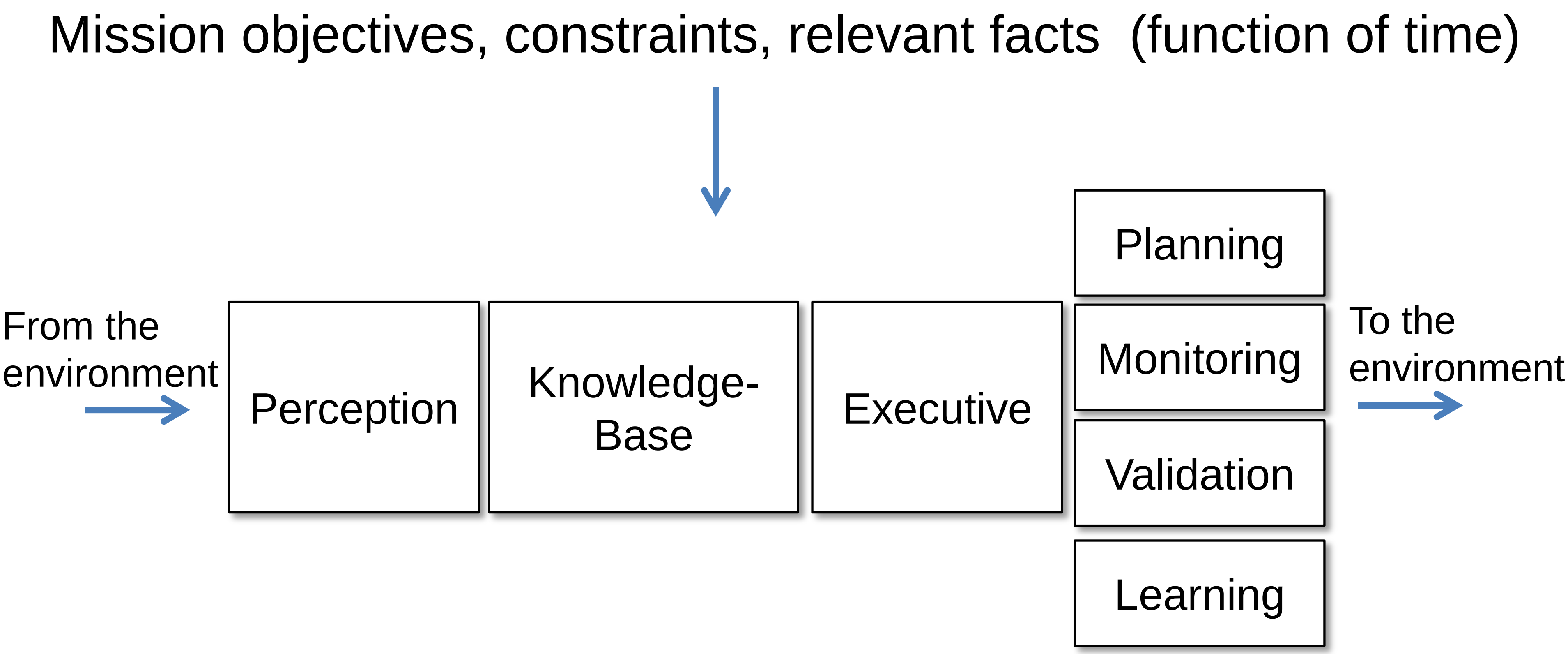
Operation and Maintenance

V

Needs to know/understand the subject matter

Knowledge-Based Systems

Archetype



Knowledge Capturing and Verification

Presented by Alessandro Pinto

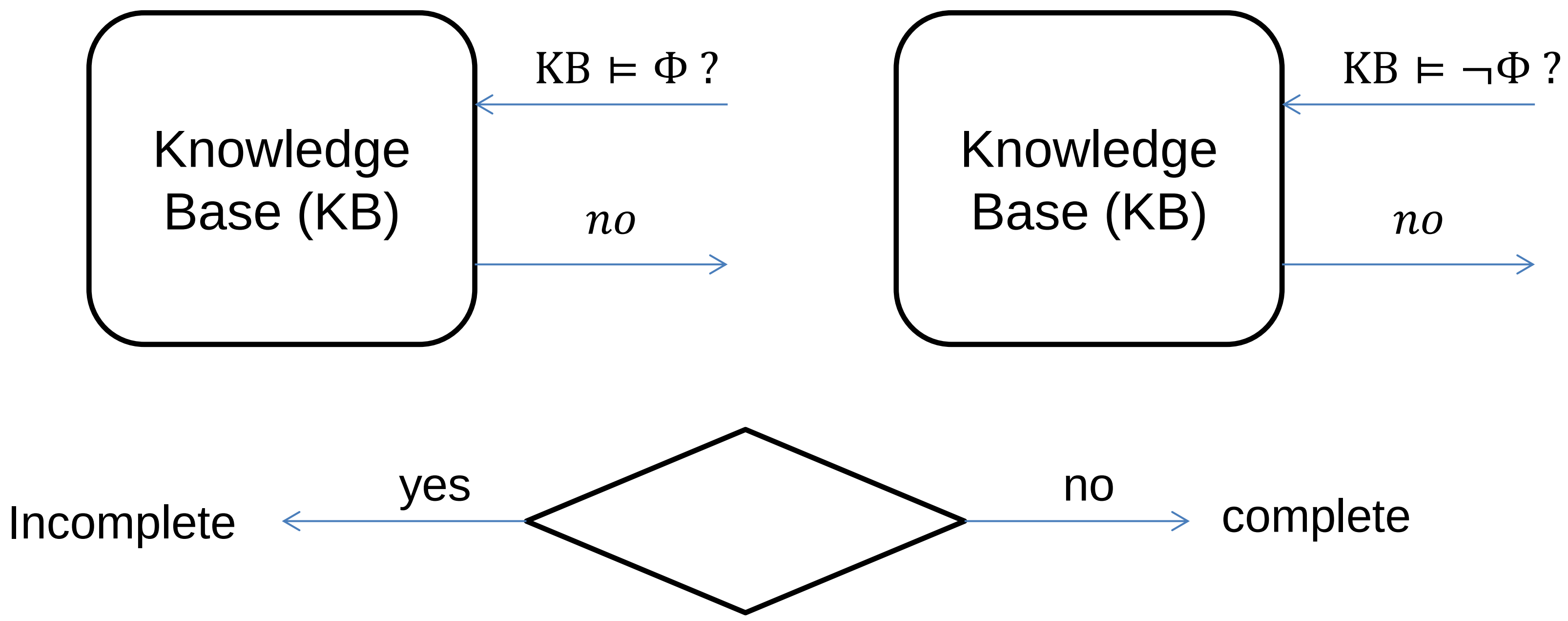
United Technologies Research Center, Berkeley, CA

This research was funded by the Defense Advanced Research Projects Agency (DARPA) Aircrew Labor In-Cockpit Automation System (ALIAS) program. The views, opinions and/or findings expressed are those of the author(s) and should not be interpreted as the official views or policies of the Department of Defense or the U.S. Government.

Completeness

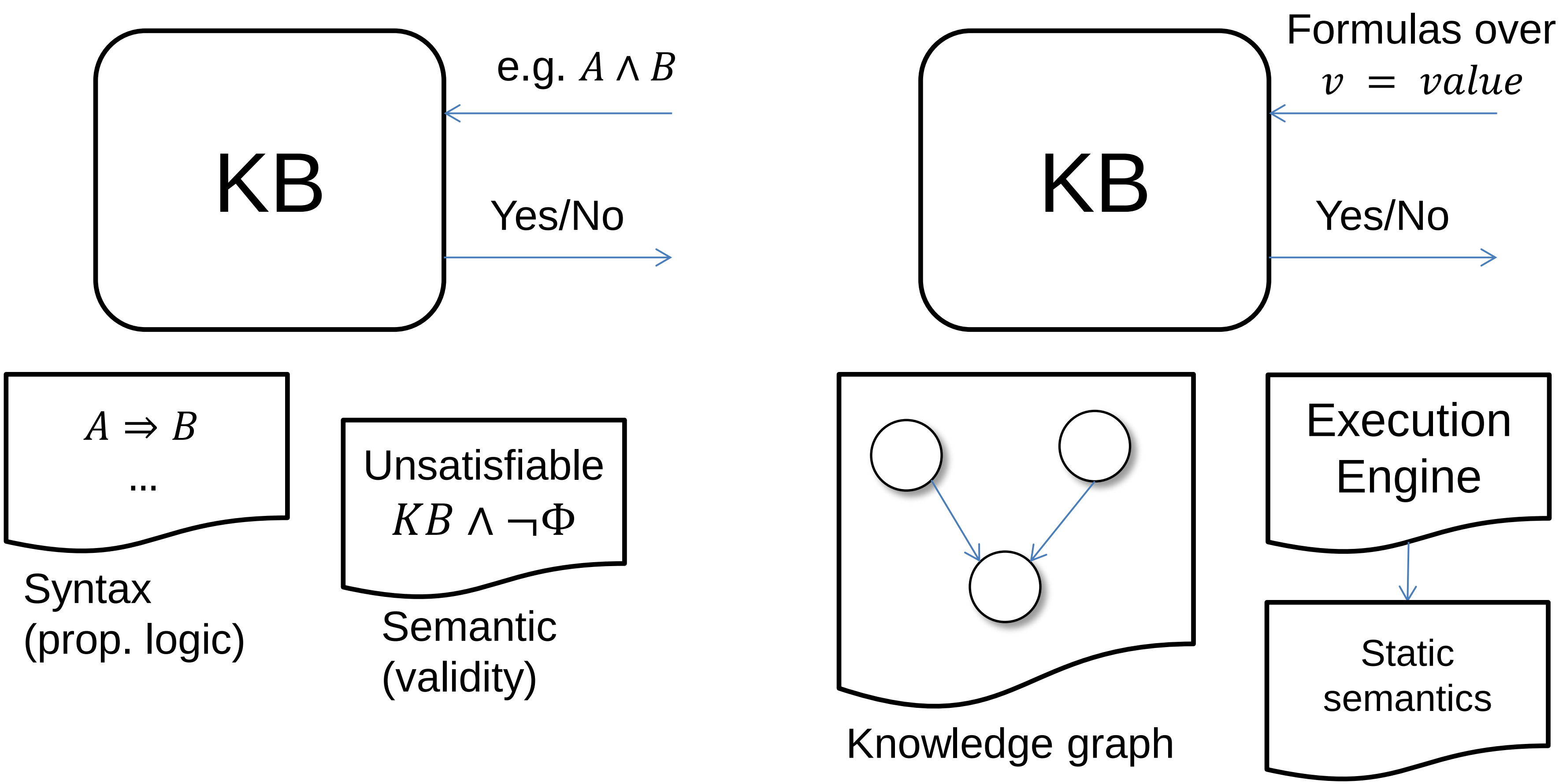
General definition

Is there a query Φ such that:



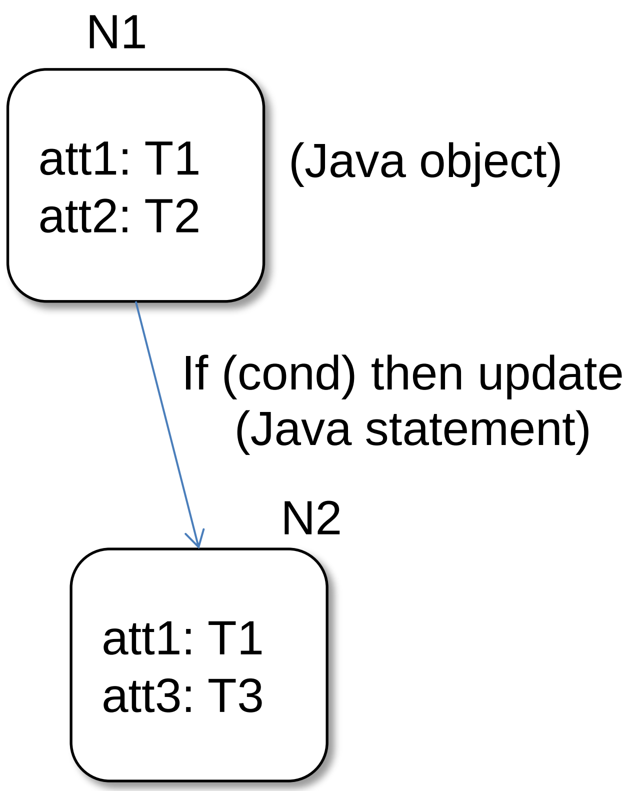
Completeness

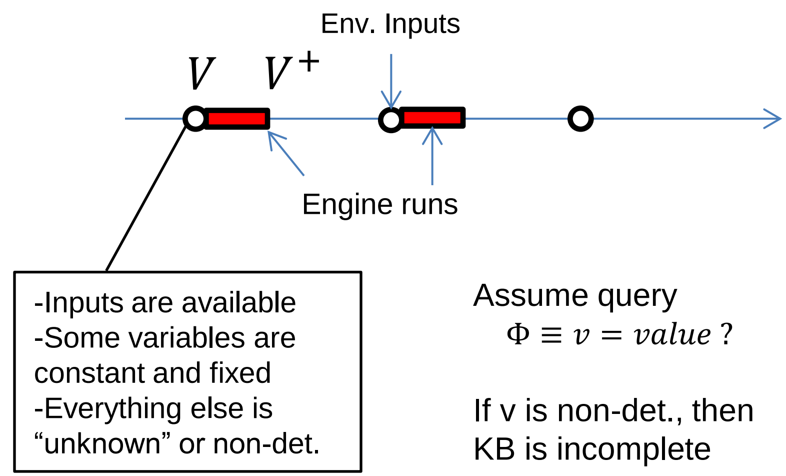
Examples of knowledge bases



Completeness of Knowledge Graphs

Syntax, semantics, and structural rules





Assume query
$\Phi \equiv v = value$?

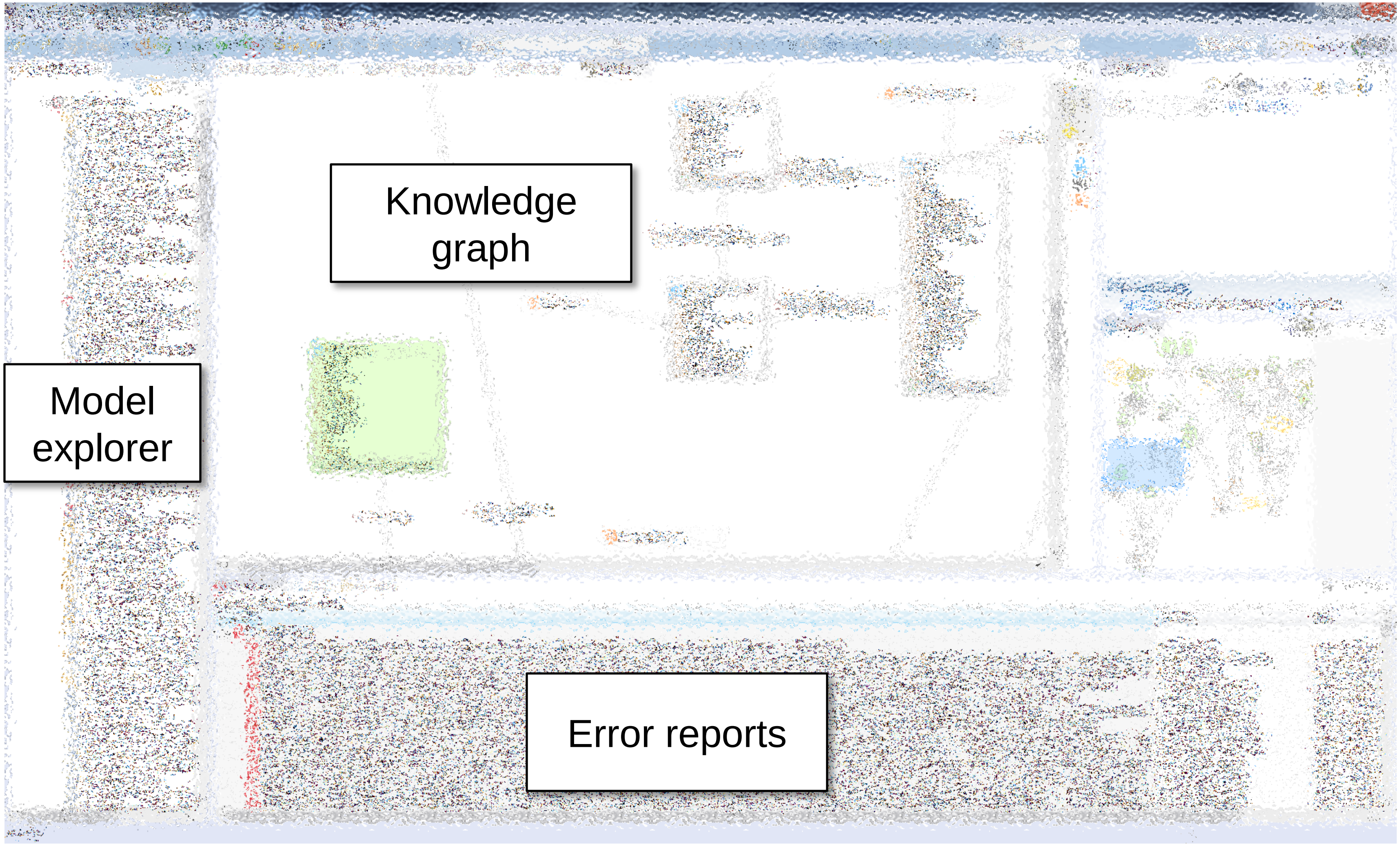
If v is non-det., then KB is incomplete

Rules:
- No cycles
- No leaf nodes
- Nodes with common child update different variables
- All variables are updated by some node
- All variables are read
- …

Analyze dependencies → Build abstract graph → Check rules → Generate report

Completeness of Knowledge Graphs

Integration in the knowledge modeling tool

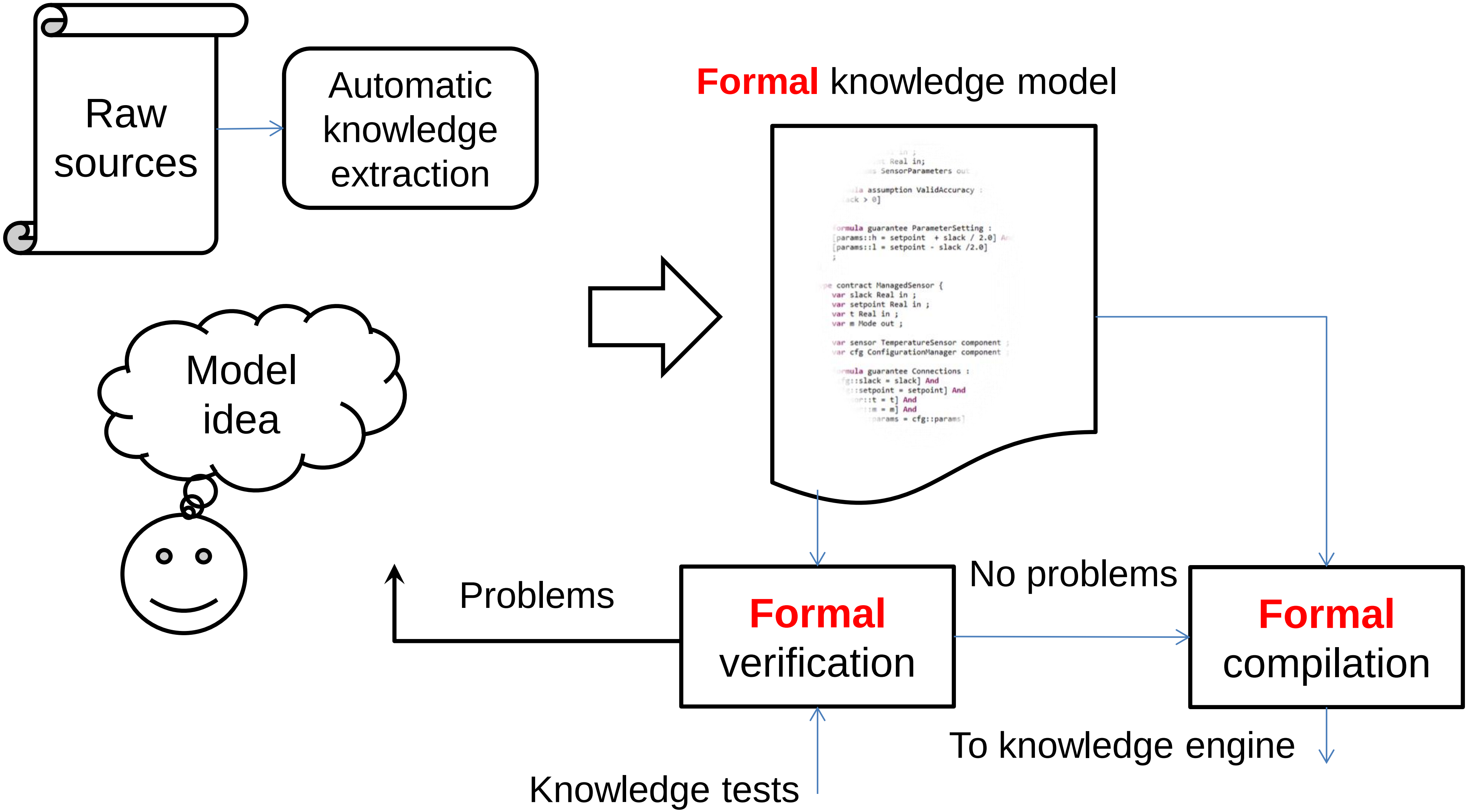


Lessons learned

- Knowledge graphs easy to understand by software engineers
- Not as powerful as declarative knowledge
- Verification is hard on generic code leading to spurious error reports (conservative abstraction)
- If fast, verification can be integrated in the knowledge modeling process
- Verification needs to be precise: avoid confusing users and loosing trust

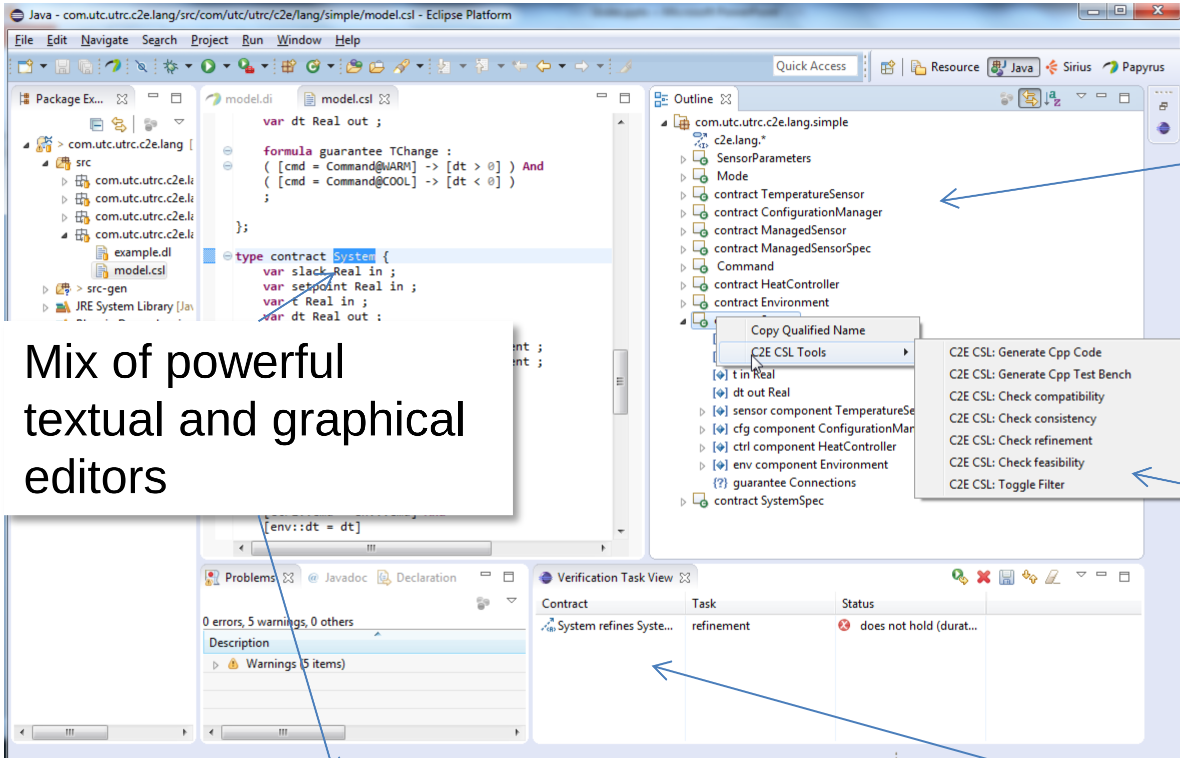
Process

From knowledge sources to execution



Tools

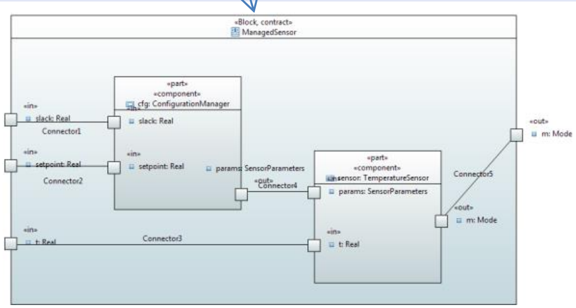
An integrated environment for knowledge modeling and verification



Model browser

Mix of powerful textual and graphical editors

Rapid access to verification and code generation tools



Verification task manager for model debugging

Conclusions

- Autonomous systems solve new class of problems
- A new design and verification method is needed for autonomous systems
- Verification should focus on what these systems know and understand