Autonomous Systems - A Rigorous Architectural Characterization

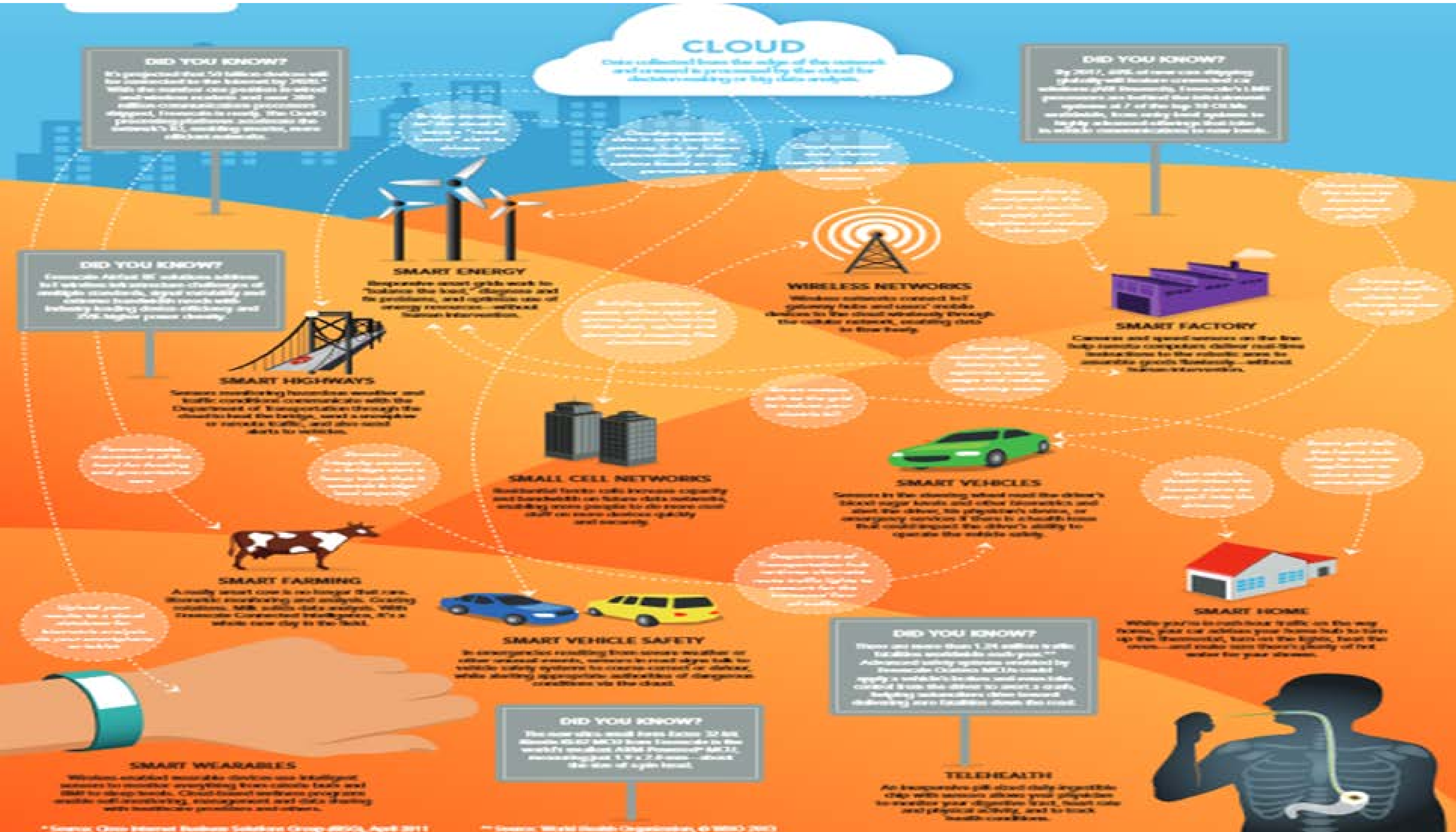
2019 IEEE SERVICES Congress

Milano, July 9, 2019

Joseph Sifakis
Verimag Laboratory

Next-generation autonomous systems – The IoT Vision

The IoT allows objects to be sensed or controlled remotely across a network infrastructure, achieving more direct integration of the physical world into computer-based systems, and resulting in improved efficiency and predictability.



Next-generation autonomous systems – The IoT Vision

The Internet of Things

Industrial IoT

Autonomous

Rules can be changed, but human-driven changes are external to normal behavior

Autonomous transport systems
Industry 4.0
Smart grids

Human IoT

Interactive

People's explicit or arbitrary actions dynamically trigger control sequences or rule changes

Intelligent services
Semantic web

Next-generation autonomous systems – Main Characteristics

Next-generation autonomous systems emerge from the needs to further automate existing complex organizations by progressive and incremental replacement of human agents by autonomous agents.

Such systems exhibit “broad intelligence” by using and producing knowledge in order to

- Manage dynamically changing sets of potentially conflicting goals – this reflects the trend of transitioning from “narrow” or “weak” AI to “strong” or “general” AI.
- Cope with uncertainty of complex and unpredictable environments
- Harmoniously, collaborate with human agents e.g. “symbiotic” autonomy.

The dystopian AI myth

Innovations

Elon Musk: ‘With artificial intelligence we are summoning the demon.’

When should we trust machines that can make mistakes and are not accountable for their behavior?

Next-generation autonomous systems – Current limitations

- Criticality requirements for next-generation autonomous systems cannot be achieved under the current state of the art
  - <u>poor trustworthiness</u> of infrastructures and systems e.g. impossibility to guarantee safety and security;
  - <u>impossibility to guarantee response times</u> in communication thus timeliness which is essential for autonomous reactive systems;
  - <u>Integration of mixed-criticality systems</u> is hard to achieve because critical systems and best-effort systems are developed following two completely different and diverging design paradigms;

- New practices emerge
  - Extensive use of learning-enabled components breaking with the traditional critical systems engineering practice – end-to-end AI-based solutions;
  - In contrast with the current systems engineering practice (*), critical software is customized by <u>updates</u> – Tesla cars software may be updated on a monthly basis.

(*) An aircraft is certified as a product that cannot be modified including all its components even HW – aircraft makers purchase and store an advance supply of the microprocessors that will run the software, sufficient to last for the estimated 50 year production!

Next-generation autonomous systems – Facing the challenge

Systems Engineering comes to a turning point moving from small size centralized non evolvable automated systems to next-generation autonomous systems

- We need a general reference semantic model that could be a basis for evaluating system autonomy - *Not just a list of "self"-prefixed terms e.g. as Self-healing, Self-optimized, Self-protected, Self-aware, Self-organized, etc.*

- What are the technical solutions for enhancing a system's autonomy?
  For each enhancement, what are the implied technical difficulties and risks?

- There is a strong and urgent need to lay out a common <u>engineering foundation</u> for the development of next-generation autonomous systems.
  Essential issues to be addressed:

  1. integration of model-based and data-driven techniques in "hybrid" design flows allowing to determine trade offs between trustworthiness and performance;

  2. means for faithful modeling and simulation of a system in its physical environment (which includes humans);

  3. combine empirical and proof-based validation for assessing trustworthiness and performance – open the way for new standards.

- Autonomous Systems
  - The concept of autonomy
  - Should we trust autonomous systems?
- In Search of a Foundation
  - “Hybrid” design flows
  - Modeling and Simulation
  - Validation
- Discussion
  - Valuing knowledge
  - The way forward

O
V
E
R
V
I
E
W

The Concept of Autonomy – Basic Definitions

An autonomous system involves two different types of components, agents and objects, operating in a common environment so that their coordinated collective behavior meets some global goals

- An agent is a reactive system (controller) interacting with components of its environment so that specific goals are met; It can monitor objects and from their environment and change their states and can coordinate its actions with other agents.

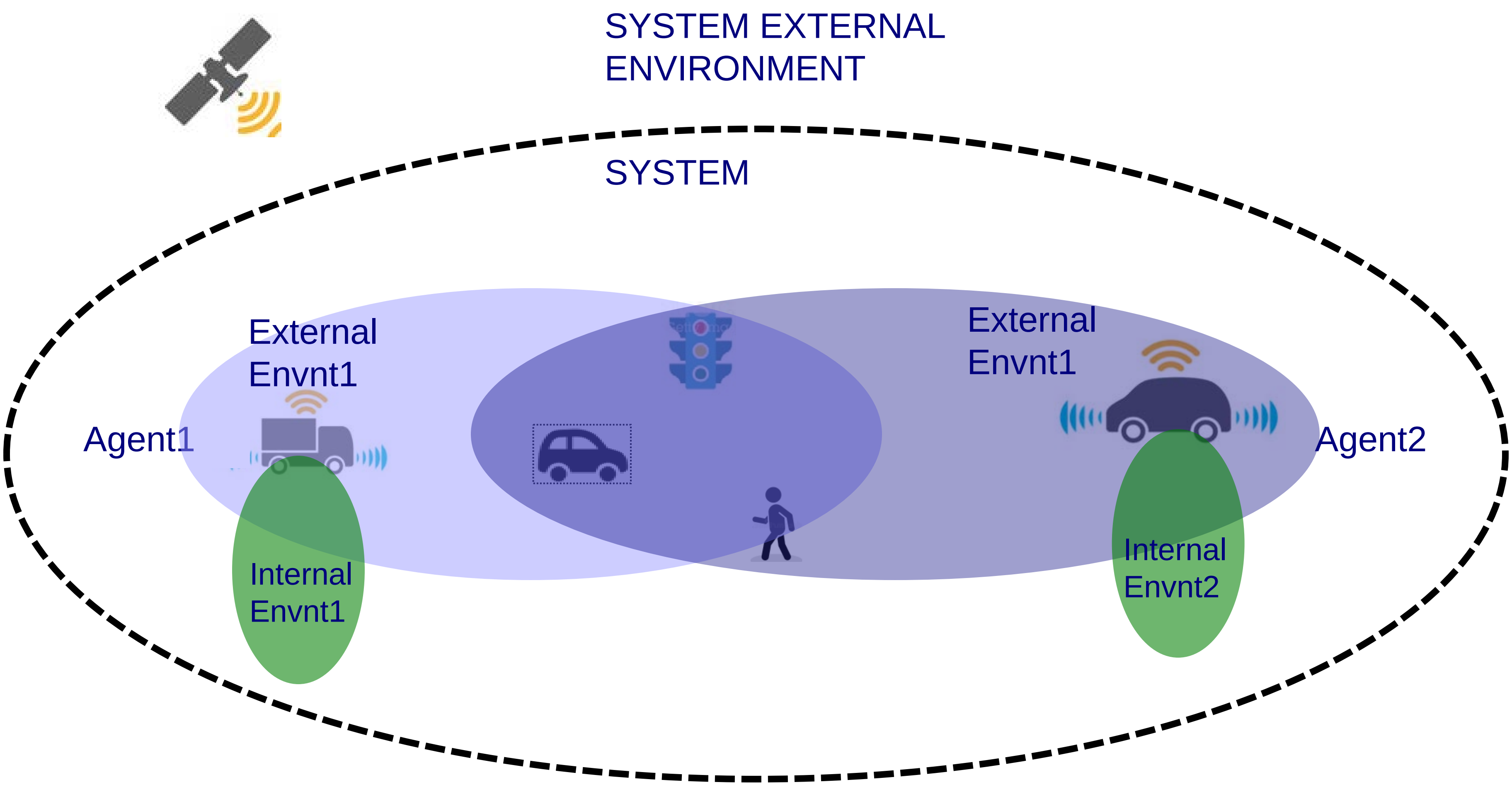
- An object is a physical or virtual component whose behavior can be controlled by system agents i.e. it is integrated as such when the system is designed

- The environment consists of the elements of the physical and virtual infrastructure of the system that are used for the coordination between components (agents and objects) e.g. geographic coordinates to determine connectivity relationships, available communication infrastructure, devices for observability/controllability of objects

Note that

- A component may be agent or object depending on its role in the system
- It is an interesting question indeed how are related system and agent goals

The Concept of Autonomy – Basic Definitions



SYSTEM= Agents + Objets + System_Environment

Agents=Agent1+Agent2

Objects= Traffic_light+Pedestrian+ Human_Driven_car

System_Environment = (External_Envnt1+External_Envnt2)x(Internal_Envnt1+Internal_Envnt2)

The Concept of Autonomy – Find the Differences

Thermostat

Automatic train shuttle

Chess-playing robot

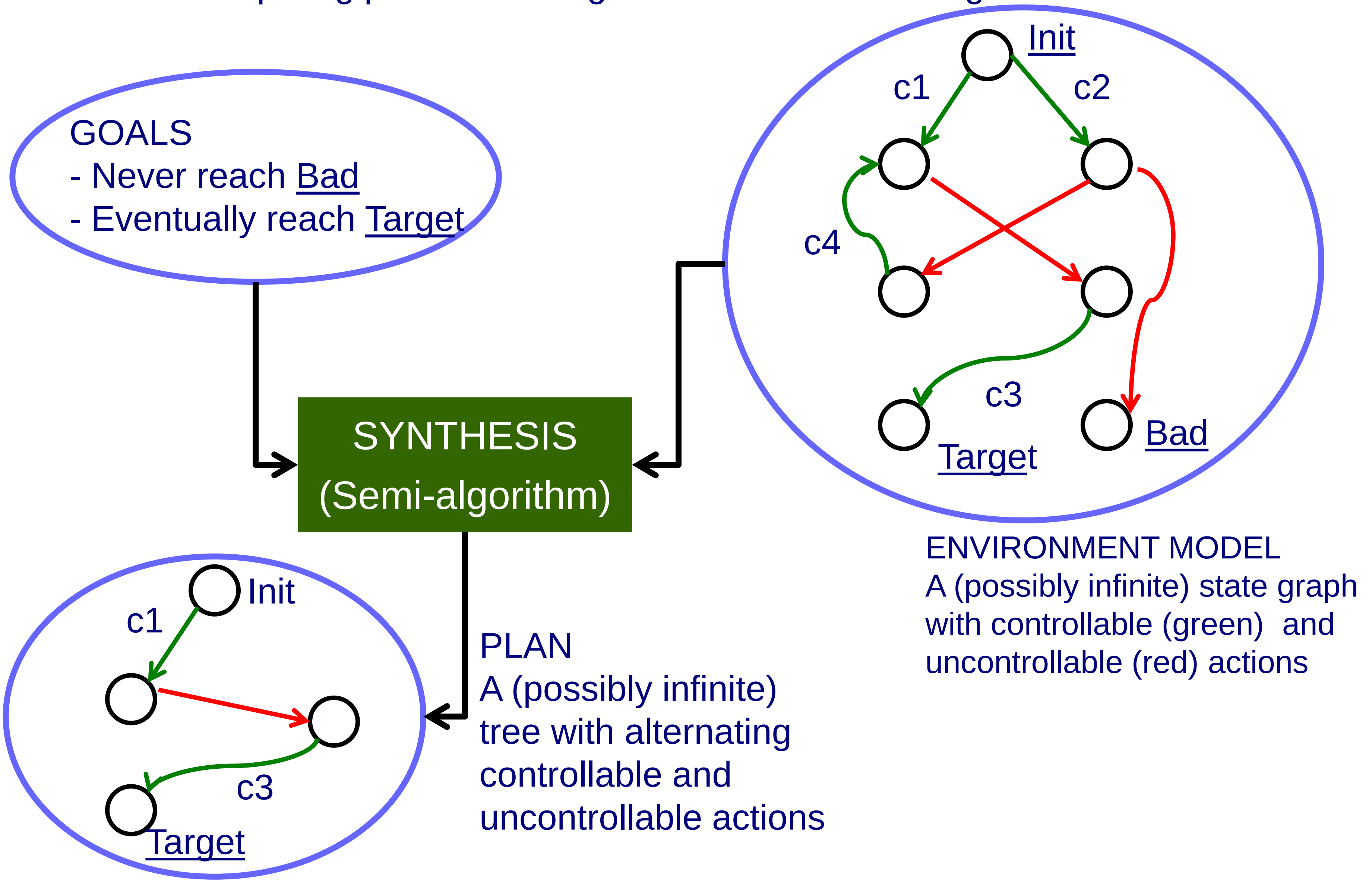
Soccer-playing robot

Robocar

Each system consists of agents acting as controllers on their environment and pursuing individual goals so that the collective behavior meets the system global goals.

The Concept of Autonomy – Meeting Goals

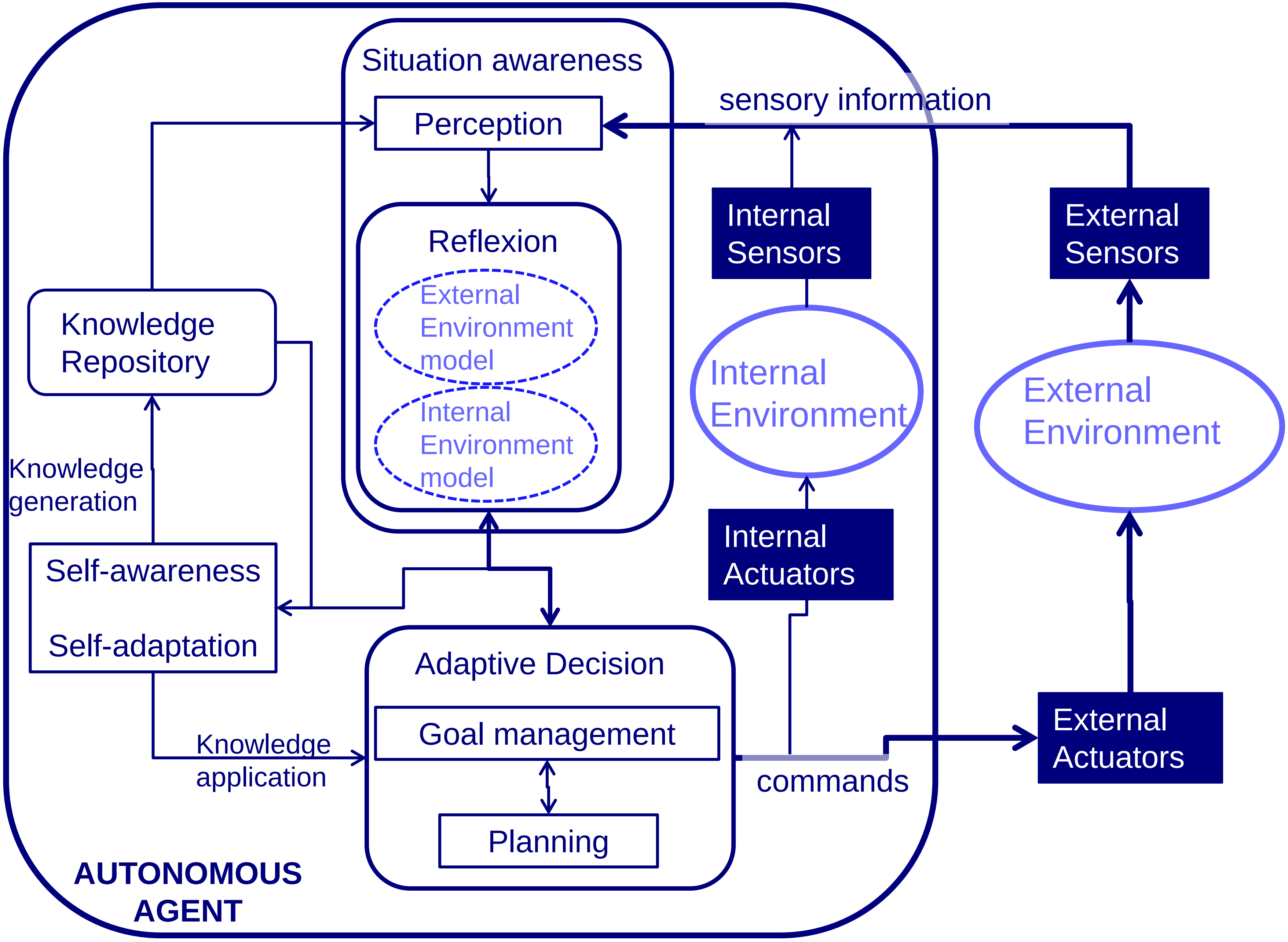
Given a set of goals and the model of an environment to be controlled, there are methods for computing plans enforcing the satisfaction of the goals.



The Concept of Autonomy – From Automation to Autonomy

	Environment	Stimuli	Meeting Goals
Thermostat	Room + Heating/cooling device	Temperature	Explicit controller Single goal
Shuttle	Cars + Passengers+ equipment	Dynamic configuration of cars+ State of equipment	Explicit controller + on line adaptation Many fixed goals
Chess robot	Chess board + pawns	Static configuration of pawns	On-line planning+ stored knowledge Dyn. Changing goals
Soccer robot	Regions in the field + Players + Ball	Dynamic configuration of players/ball	On-line planning+ stored/generated knowledge Dyn. changing goals
Robocar	Vehicles/obstacles + Road/communication equipment	Dynamic configuration of vehicles/obstacles + State of equipment	On-line planning+ stored/generated knowledge Dyn. changing goals

The Concept of Autonomy – Architectural Characterization



The Concept of Autonomy – Architectural Characterization

- Autonomy is the capacity of an agent to achieve a set of coordinated goals by its own means (without human intervention) adapting to environment variations. It combines five complementary functions:

  - Perception e.g. interpretation of stimuli, removing ambiguity from complex input data and determining relevant information;

  - Reflection e.g. building/updating a faithful environment run-time model from which strategies meeting the goals can be computed;

  - Goal management e.g. choosing among possible goals the most appropriate ones for a given configuration of the environment model;

  - Planning to achieve a particular goal;

  - Self-awareness/adaptation e.g. the ability to create new situational knowledge and new goals through learning and reasoning

- These functions are implementation-agnostic
- Insights on
  - Automation vs. Autonomy;
  - Human-assisted vs. Machine Empowered autonomy

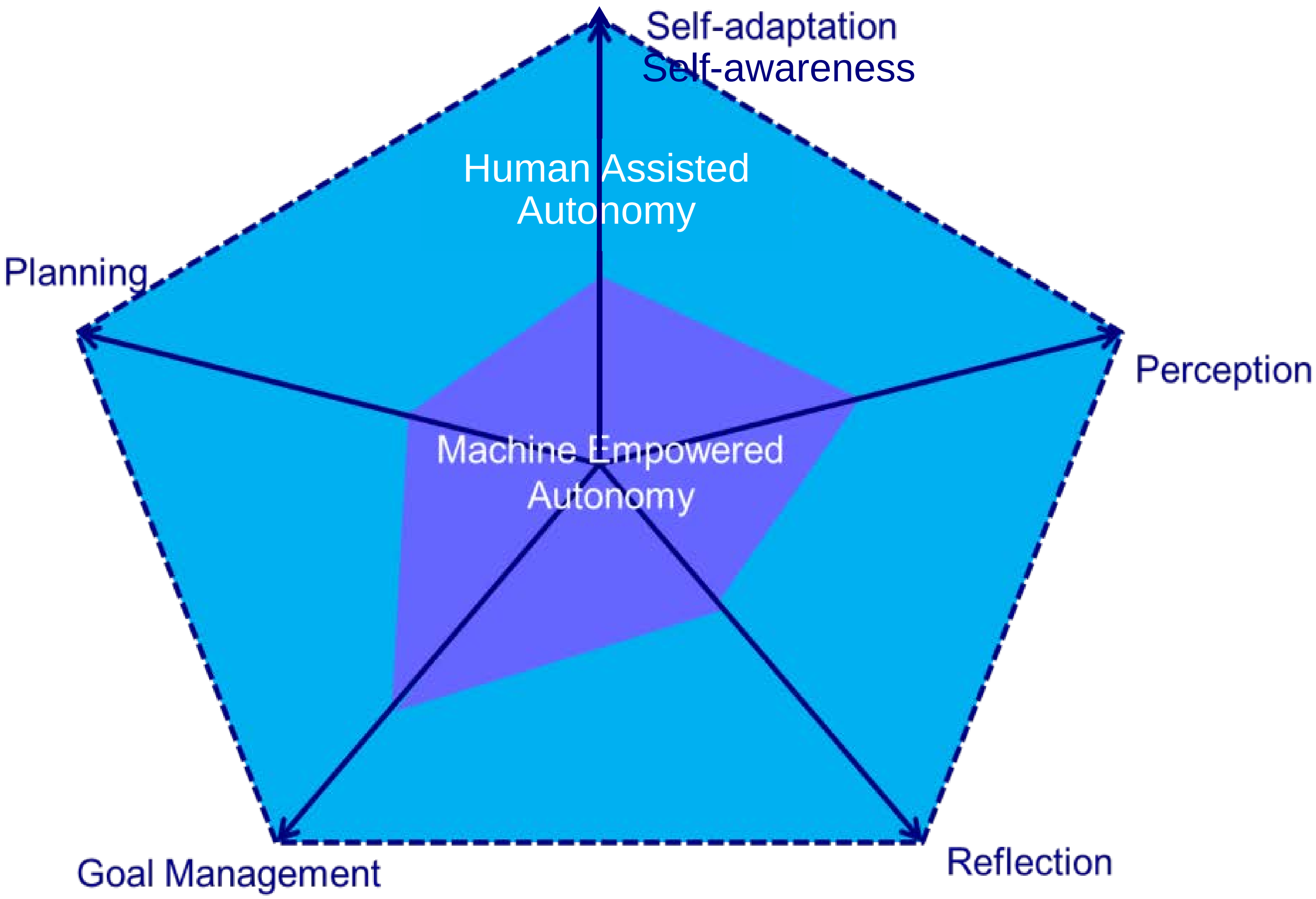
- Autonomous Systems
  - The concept of autonomy
  - Should we trust autonomous systems?
- In Search of a Foundation
  - “Hybrid” design flows
  - Modeling and Simulation
  - Validation
- Discussion
  - Valuing knowledge
  - The way forward

O
V
E
R
V
I
E
W

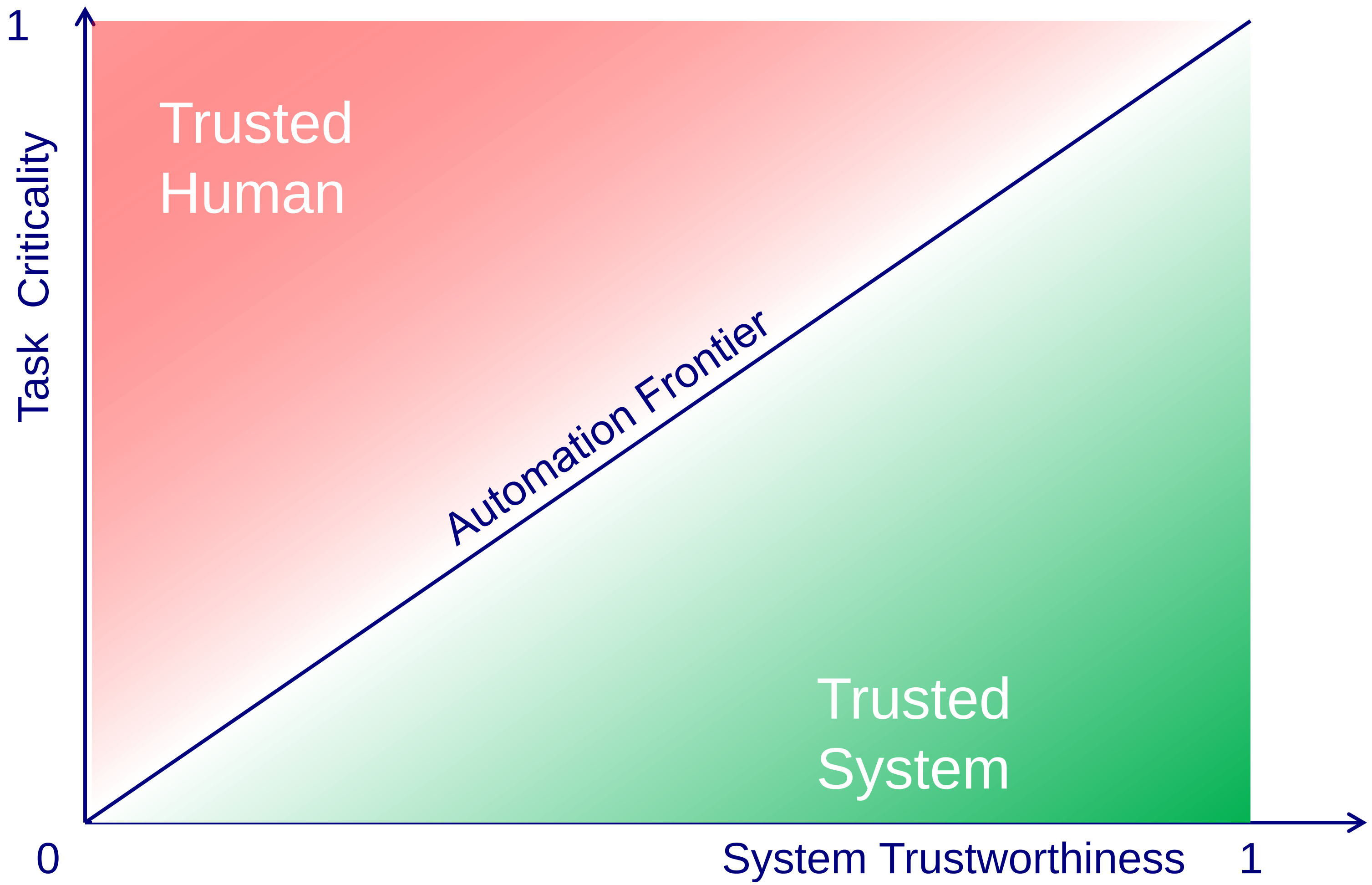
Trusting Autonomous Systems – Autonomy Level

	SAE AYTONOMY LEVELS
Level 0	No automation
Level 1	Driver assistance required (“hands on”)
	The driver still needs to maintain full situational awareness and control of the vehicle e.g. cruise control.
Level 2	Partial automation options available(“hands off”)
	Autopilot manages both speed and steering under certain conditions, e.g. highway driving.
Level 3	Conditional Automation(“eyes off”)
	The car, rather than the driver, takes over actively monitoring the environment when the system is engaged. However, human drivers must be prepared to respond to a "request to intervene”
Level 4	High automation (“mind off”)
	Self driving is supported only in limited areas (geofenced) or under special circumstances, like traffic jams
Level 5	Full automation (“steering wheel optional”)
	No human intervention is required e.g. a robotic taxi

Trusting Autonomous Systems – Autonomy Level



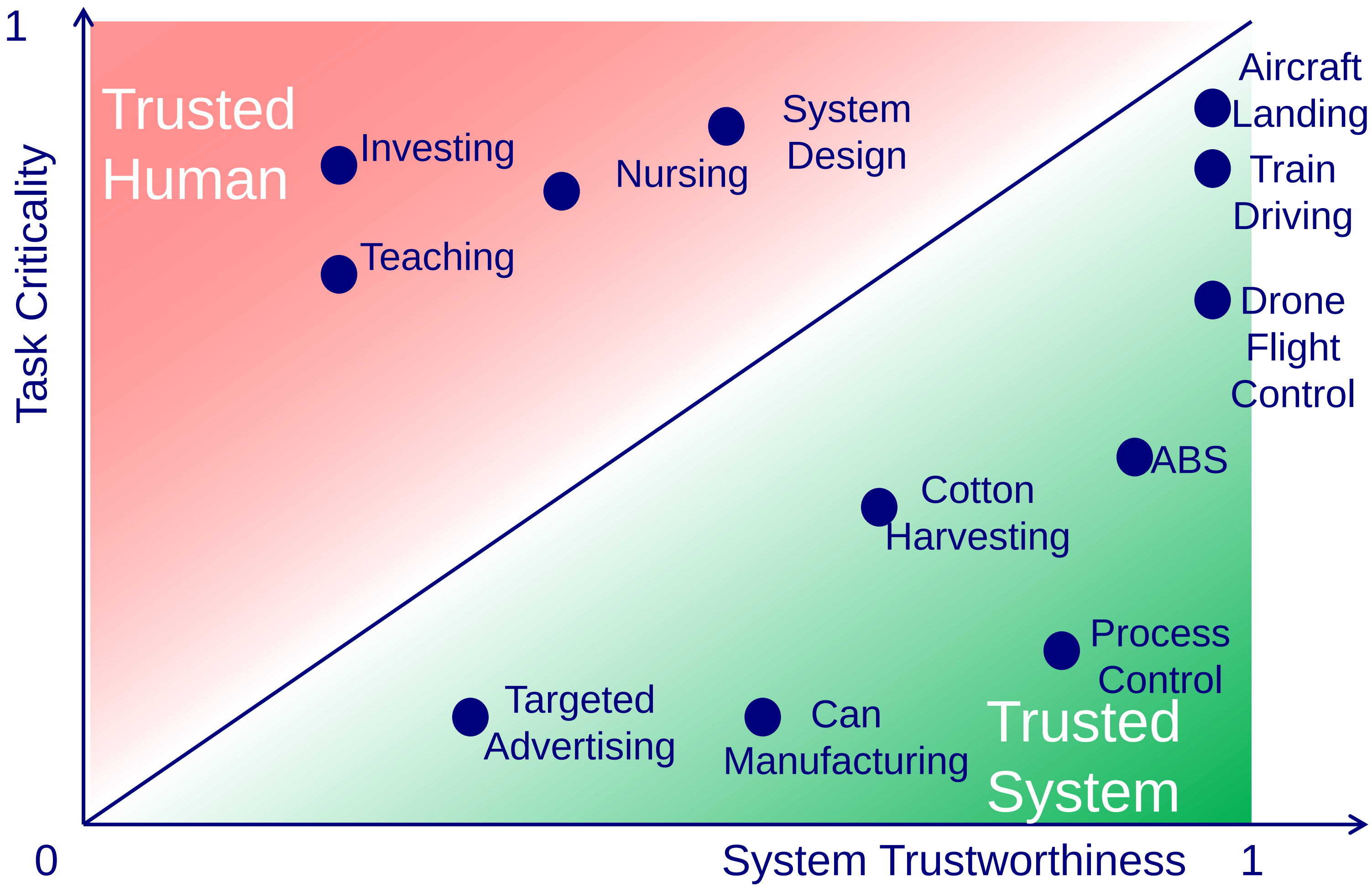
Trusting Autonomous Systems –The Automation Frontier



How we decide whether a System can be trusted for performing a Task:

- SystemTrustworthiness: the system will behave as expected despite any kind of mishaps e.g. resilience to errors, failures, attacks.
- Task Criticality: characterizes the severity of the impact of an error in the fulfilment of the task e.g. driving a car, operating on a patient, nuclear plant control.

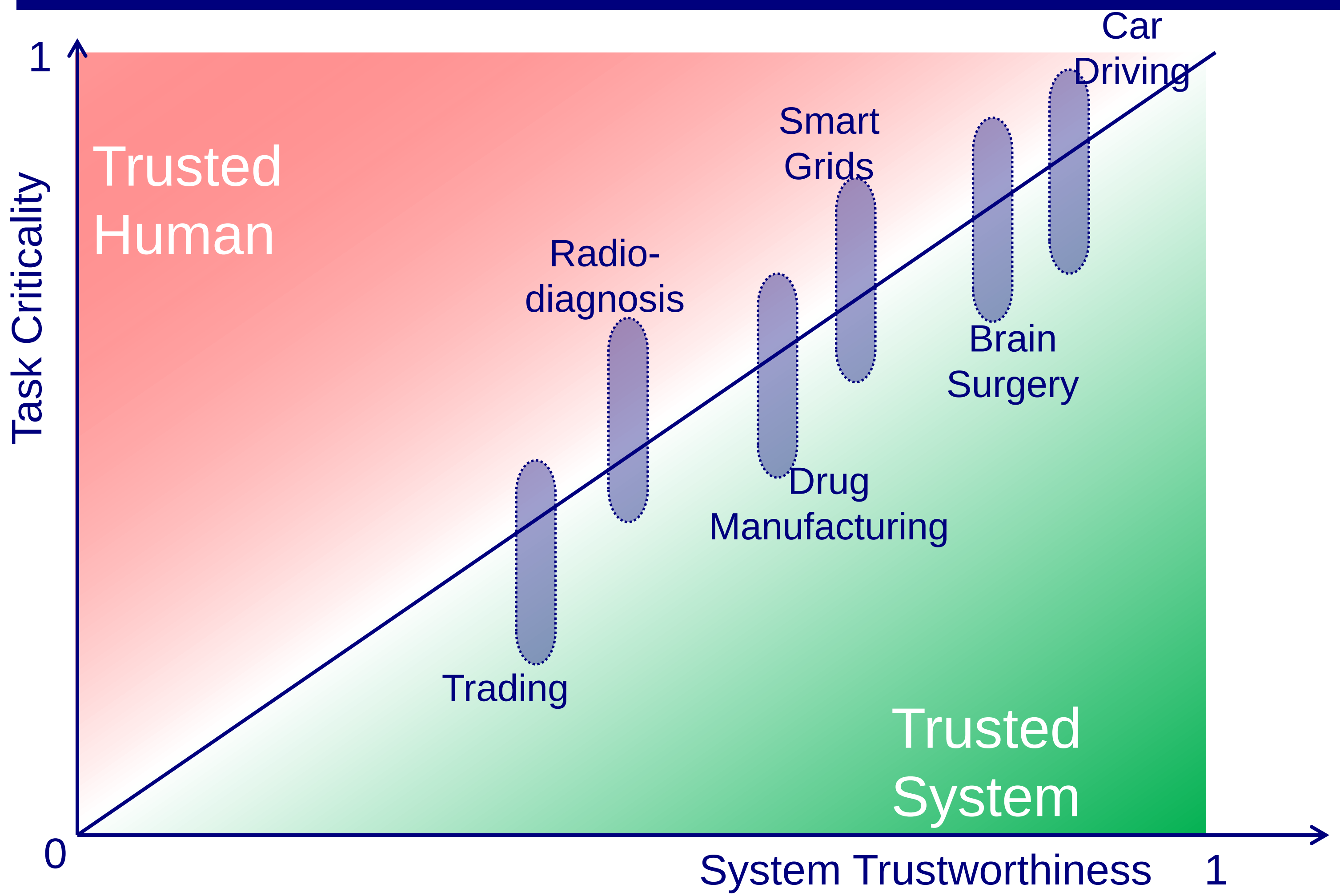
Trusting Autonomous Systems – Automated vs.Non-automated



Automated systems: simple decision process or small impact of failures.

Non-automated systems: require good situation awareness and multiple goal management.

Trusting Autonomous Systems – Symbiotic Systems



- Autonomous systems extensively use knowledge; they cannot be effectively implemented without massive use of AI-based techniques.

- Problem: choose the appropriate <u>degree of autonomy</u> (machine empowered vs. human-assisted operation e.g. SAE degrees of autonomy for vehicles).

Trusting Autonomous Systems – The Role of Institutions

Social acceptance of Truth is a complex process where institutions play an important role

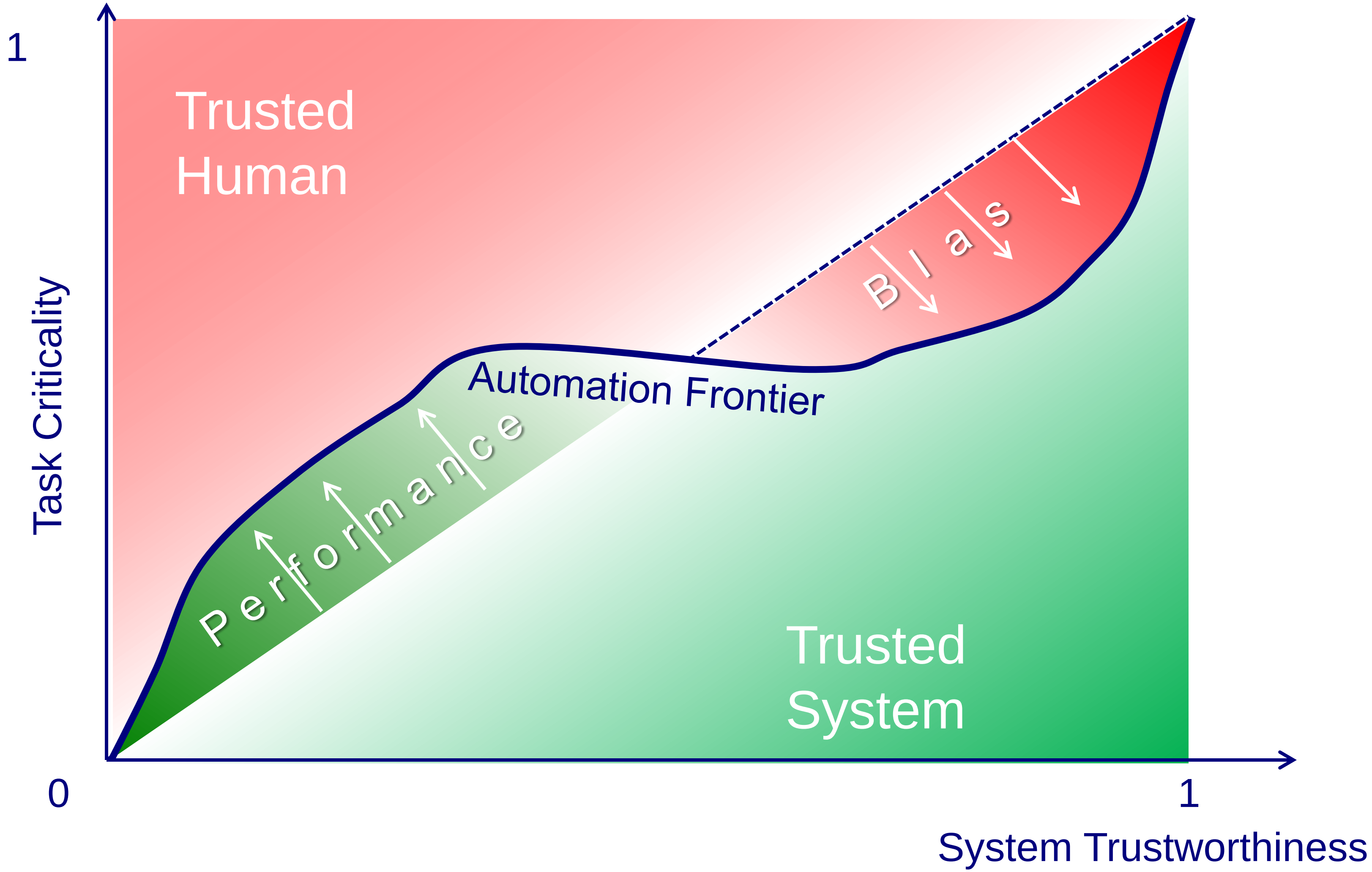
100+ years after

Galileo is WRONG!!

Galileo is RIGHT!!

- ❑ Institutions shape public perceptions about what is TRUE, RIGHT, SAFE, etc…
- ❑ In modern societies independent institutions guarantee trustworthiness of technical infrastructure and common services based on standards and regulations e.g. FDA., FAA, NTHSA, in the US.
- ❑ Most critical systems standards require <u>conclusive model-based evidence</u> e.g. based on the laws of Physics a bridge will not collapse for a century. Such standards not applicable to AI-based systems – self-driving cars are “self-certified”!

Trusting Autonomous Systems – Shaping Factors



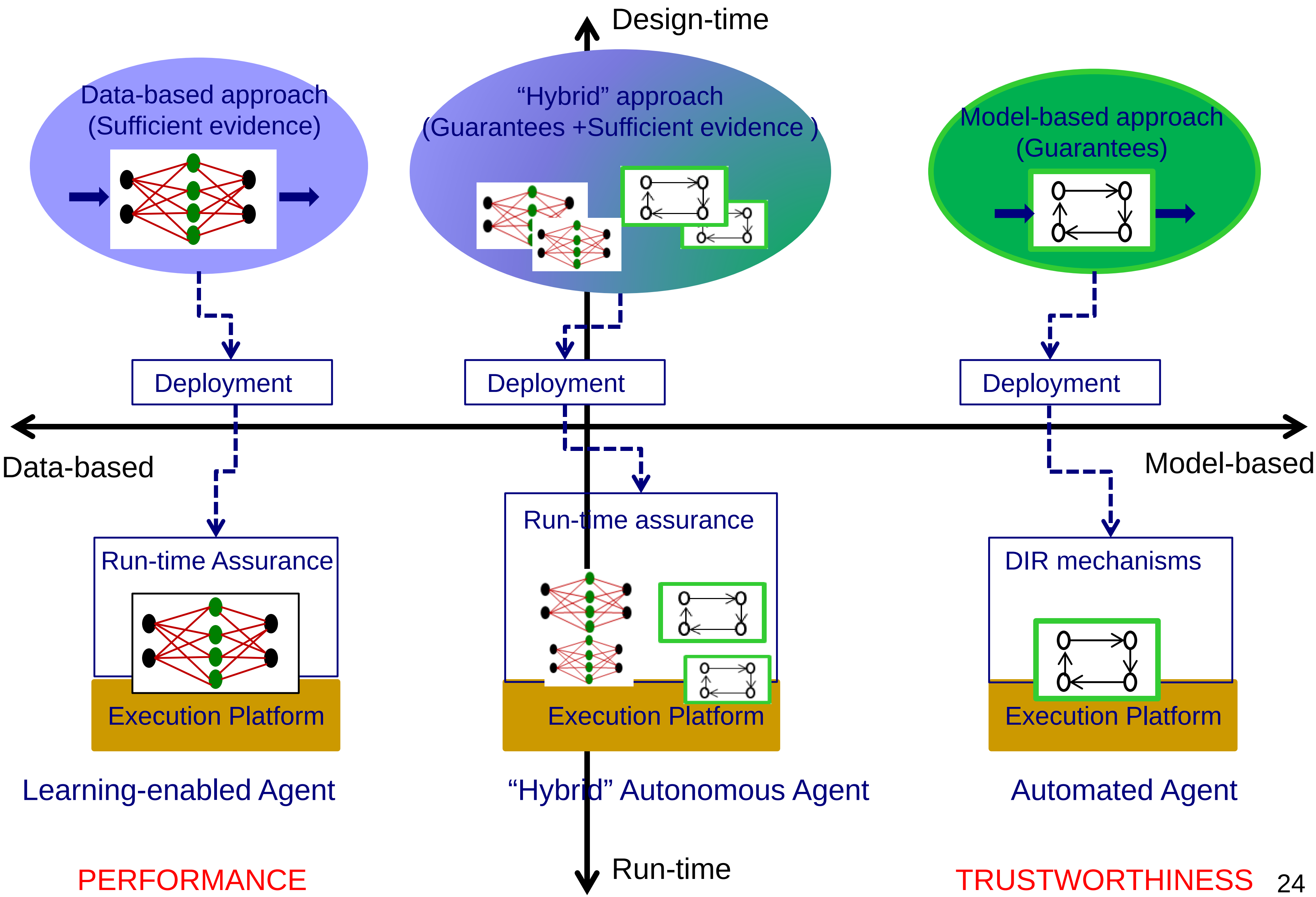
Performance: for low criticality, trade quality of service for performance;

Bias: human error is more acceptable than machine failure.

- Autonomous Systems
  - The concept of autonomy
  - Should we trust autonomous systems?
- In Search of a Foundation
  - “Hybrid” design flows
  - Modeling and Simulation
  - Validation
- Discussion
  - Valuing knowledge
  - The way forward

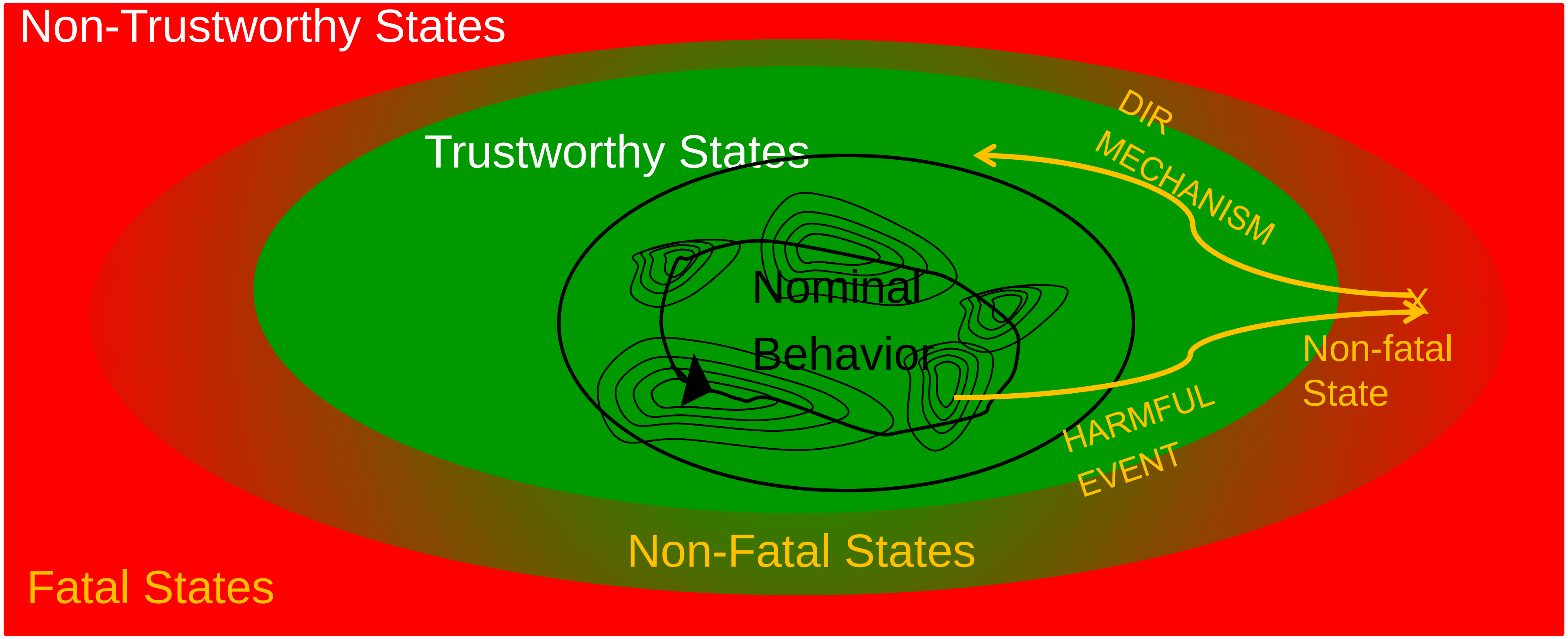
OVERVIEW

Hybrid Design Flows – The Principle



Hybrid Design Flows – Model-based Trustworthiness

- Current approaches guarantee trustworthiness at design time by applying
  - a more or less exhaustive risk analysis that identifies all kind of harmful events
  - techniques guaranteeing tolerance: any single harmful event leads to non-fatal states
  - DIR (Detection, Isolation, Recovery) mechanisms leading from non-fatal states to trustworthy states
- These approaches cannot be directly applied to autonomous systems
  - Lack of predictability and environment complexity make practically impossible identification at design time of all harmful events and corresponding DIR mechanisms
  - Use of learning-enabled components



Hybrid Design Flows – Model-based Trustworthiness

1	Vehicle Failure
2	Control Loss With Prior Vehicle Action
3	Control Loss Without Prior Vehicle Action
4	Running Red Light
5	Running Stop Sign
6	Road Edge Departure With Prior Vehicle Maneuver
7	Road Edge Departure Without Prior Vehicle Maneuver
8	Road Edge Departure While Backing Up
9	Animal Crash With Prior Vehicle Maneuver
10	Animal Crash Without Prior Vehicle Maneuver
11	Pedestrian Crash With Prior Vehicle Maneuver
12	Pedestrian Crash Without Prior Vehicle Maneuver
13	Pedalcyclist Crash With Prior Vehicle Maneuver
14	Pedalcyclist Crash Without Prior Vehicle Maneuver
15	Backing Up Into Another Vehicle
16	Vehicle(s) Turning – Same Direction
17	Vehicle(s) Parking – Same Direction
18	Vehicle(s) Changing Lanes – Same Direction

19	Vehicle(s) Drifting – Same Direction
20	Vehicle(s) Making a Maneuver – Opposite Direction

23	Lead Vehicle Accelerating
24	Lead Vehicle Moving at Lower Constant Speed
25	Lead Vehicle Decelerating
26	Lead Vehicle Stopped
27	Left Turn Across Path From Opposite Directions at Signalized Junctions
28	Vehicle Turning Right at Signalized Junctions
29	Left Turn Across Path From Opposite Directions at Non-Signalized Junctions
30	Straight Crossing Paths at Non-Signalized Junctions
31	Vehicle(s) Turning at Non-Signalized Junctions
32	Evasive Action With Prior Vehicle Maneuver
33	Evasive Action Without Prior Vehicle Maneuver
34	Non-Collision Incident
35	Object Crash With Prior Vehicle Maneuver
36	Object Crash Without Prior Vehicle Maneuver
37	Other

Pre-crash failure typology covering 99.4% of light-vehicle crashes for 5,942,000 cases.
Source: Pre-Crash Scenario Typology for Crash Avoidance Research, DOT HS 810 767, April 2017.

FDIR approaches are not anymore applicable due to overwhelming complexity!

Hybrid Design Flows – Model-based Guarantees

Mobileye's Responsibility-Sensitive Safety: Compute lower bounds of the distance between two cars that guarantee safety. ("*On a Formal Model of Safe and Scalable Self-driving Cars*" Shai Shalev-Shwartz, Shaked Shammah, Amnon Shashua, Mobileye, 2017)

Safe Distance Formula

$$d_{\min} = L + T_f\left[v_r - v_f + \rho\,(a_a + a_b)\right] - \frac{\rho^2 a_b}{2} + \frac{(T_r - T_f)(v_r + \rho\, a_a - (T_f - \rho)a_b)}{2}$$

- L is the average length of the vehicles
- ρ is the response time of the rear vehicle
- v_r, v_f are the velocities of the rear/front vehicles
- a_a, a_b are the maximal acceleration/braking of the vehicles
- T_f is the time for the front car to reach a full stop if it would apply maximal braking
- T_r is the time for the rear car to reach a full stop if it would apply maximal acceleration during the response time, and from there on maximal braking

See also *"The Safety Force Field"* David Nistér, Hon-Leung Lee, Julia Ng, Yizhou Wang, Nvidia White Paper, March 2019

Hybrid Design Flows – Control for Safety and Performance

The general problem:

1. An agent provides critical services and possibly some non-critical services.

2. The agent uses a variable amount of free resources F (measured in space, time, memory, energy, etc.) such that $F_{min} \leq F$ and $|\partial^2 F / \partial t^2| \leq a_{max}$

- F_{min} is sufficient for the system to ensure the critical services
- Critical services should be absolutely ensured (safety)
- The rest of the available resources should be used in the best possible manner to ensure non critical services (performance).

- ❑ Safety cannot be dissociated from performance e.g. overtaking on a two lane road
- ❑ The problem needs to be solved for a humongous number of configurations:
  - use learning-enabled techniques to recognize types of configurations
  - for each identified type, apply a model-based protocol

- Autonomous Systems
  - The concept of autonomy
  - Should we trust autonomous systems?

- In Search of a Foundation
  - “Hybrid” design flows
  - Modeling and Simulation
  - Validation

- Discussion
  - Valuing knowledge
  - The way forward

OVERVIEW

Modeling and Simulation – Basic Modeling Concepts

Currently, most simulation systems use ad-hoc techniques coupling an autonomous monolithic agent to game SW. They lack features for

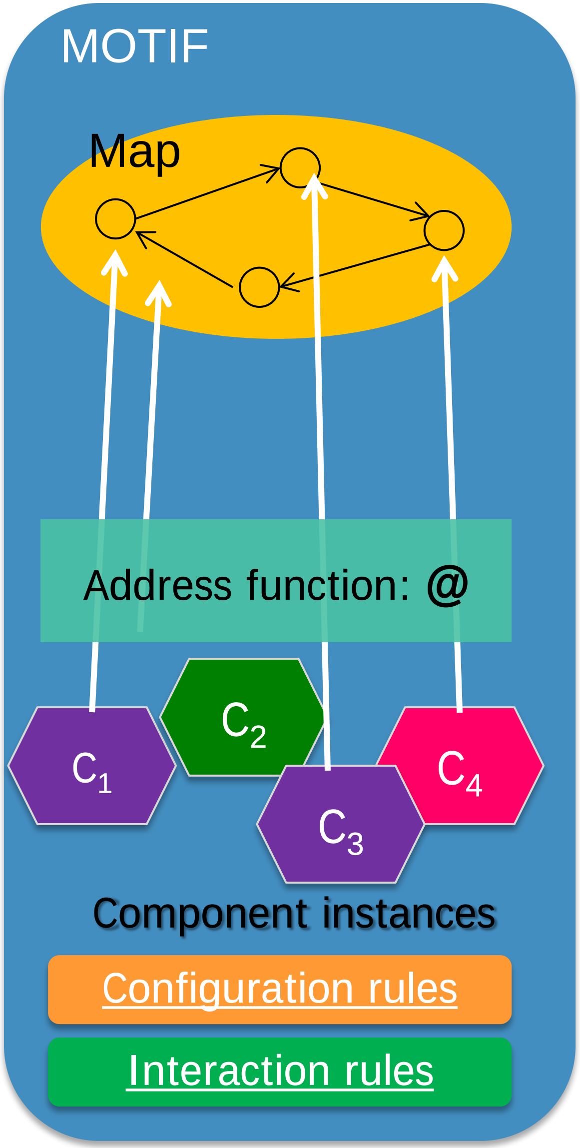
- Building scenarios that capture behavior corner cases and high risk situations
- Building environment models incrementally and compositionally
- Different levels of abstraction from fine grain simulation of cyber physical components to high level simulation

What is the value of results reported by Waymo: 27 000 cars running 24/7, 10 million miles simulated per day, >7 Billion miles of simulation.

We need component-based modeling frameworks integrating:

1. Libraries of component types for both agents and objects, as well as libraries of architecture patterns and protocols;
2. Expressive component coordination primitives supporting parametric description and various types of dynamism such as component creation/deletion and mobility;
3. Self-organization by supporting multi-mode coordination e.g. a component can live in many different “worlds” and migrate according to pursued goals.
4. Knowledge management and application for situational awareness and generation of new goals accordingly.

Modeling and Simulation – State-aware Simulation

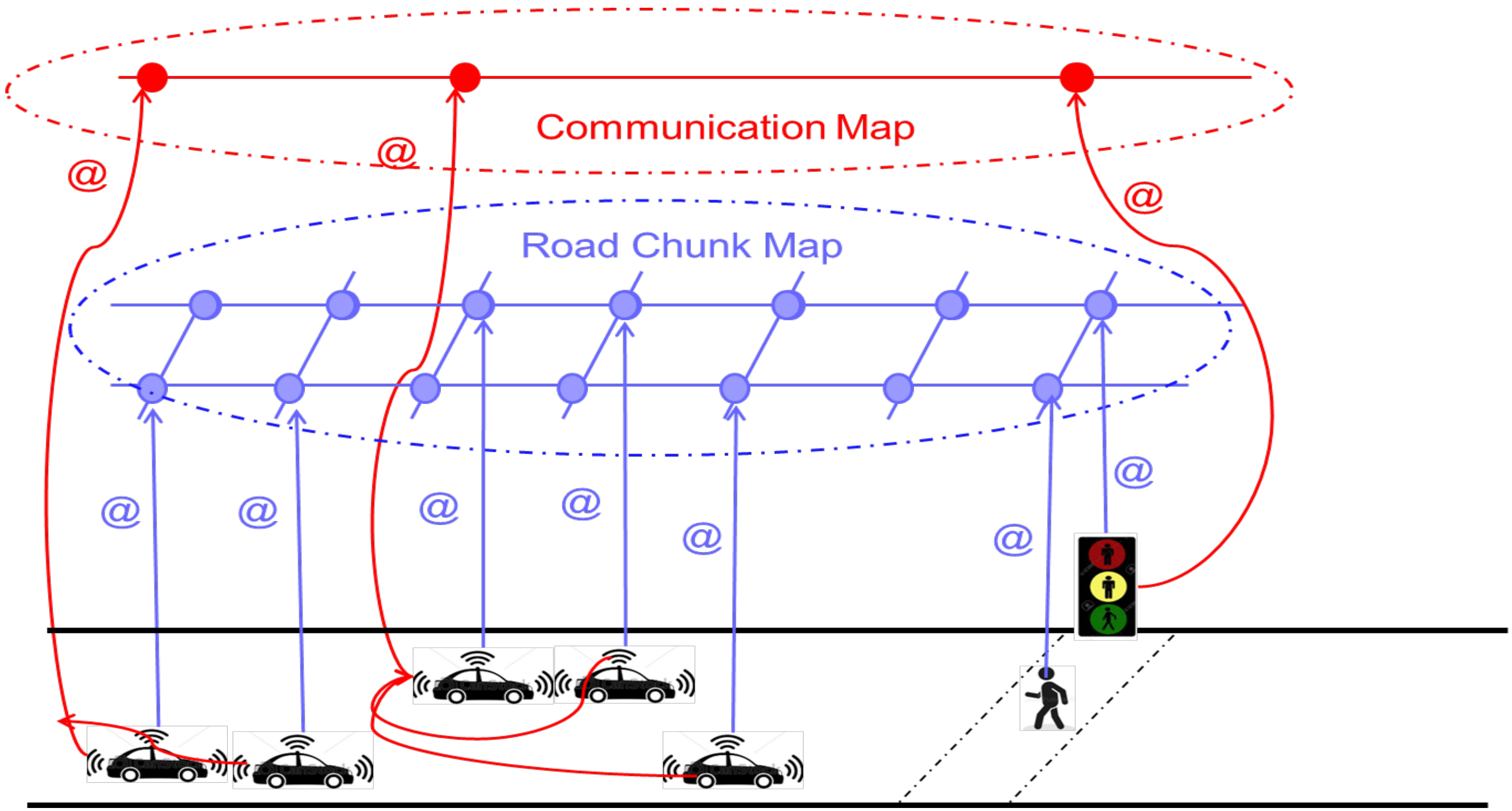


DR-BIP (Dynamic Reconfigurable BIP)

- A system is a set of (architecture) motifs
- A motif is a coordination mode consisting of
  - A set of components, instances of types of agents or objects
  - A map that is a graph (N,E) used to describe relations between components e.g. geographical, organizational, etc.
  - An address function @ mapping components into nodes of the map
  - Interaction rules: define interactions (atomic multiparty synchronization) between components
  - Configuration rules:
    - Mobility of components (change of @)
    - Creation/deletion of components
    - Dynamic change of the map

The meaning of systems models is defined using operational semantics

Model-based Approach – State-aware Simulation



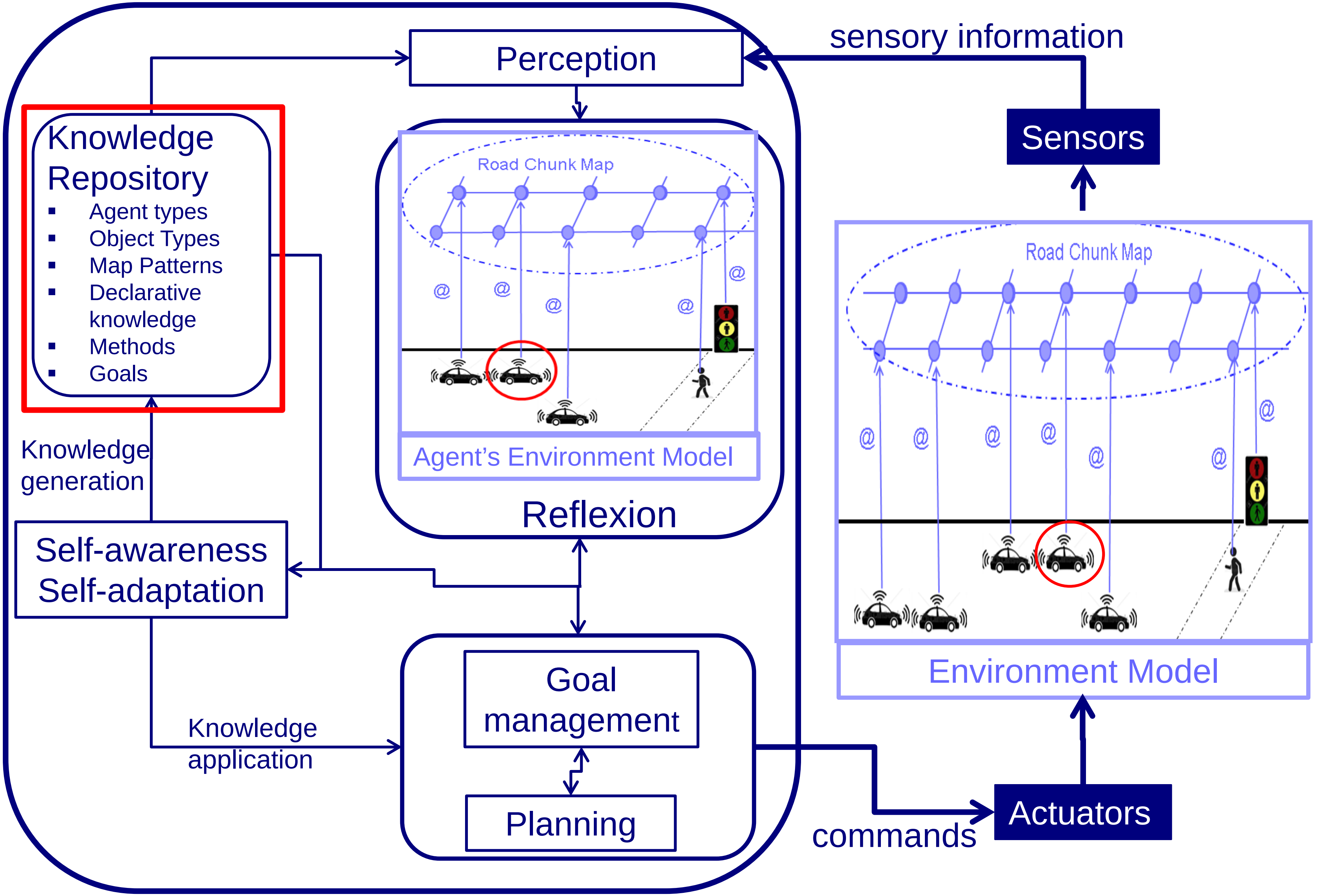
<u>*Interaction rule*</u>*:*

for all a,a':vehicle, if [dist(@(a),@(a'))<l] then exchange(a.speed,a'.speed).

<u>*Mobility rule*</u> *:*

for all a:vehicle if @(a)=n and $@^{-1}$*(n+1)=empty then @(a):=n+1.*

Model-based Approach – Refined Agent Model



- Autonomous Systems
  - The concept of autonomy
  - Should we trust autonomous systems?

- In Search of a Foundation
  - “Hybrid” design flows
  - Modeling and Simulation
  - Validation

- Discussion
  - Valuing knowledge
  - The way forward

OVERVIEW

In Search of a Foundation – Validation

- Machine learning techniques cannot be formally verified as they are not developed based on formal goals e.g. specifying how a dog looks different from a cat - instead, we are showing a whole bunch of pictures so they can learn just like a human learns the differences between a cat and a dog.

- Pushing model-based validation techniques to the limits

- Increasing confidence in ML-models which remain mostly “black boxes”

  - Metamorphic testing: $\exists\ \varphi 1, \varphi 2$ if $y = f(x)$ then $\varphi 2(y) \approx f(\varphi 1(x))$
  - Determining reference models (oracles) i.e. interpretability, explainability, “causal modeling”

- Combining proof-based and empirical validation techniques

<u>Formalization of goals</u> for autonomous systems is extremely hard e.g. “behavioral competencies” for self-driving cars (California PATH)

1. **Detect and Respond to Speed Limit Changes and Speed Advisories**
2. Perform High-Speed Merge (e.g., Freeway)
3. Perform Low-Speed Merge
4. Move Out of the Travel Lane and Park (e.g., to the Shoulder for Minimal Risk)
5. Detect and Respond to Encroaching Oncoming Vehicles
6. **Detect Passing and No Passing Zones and Perform Passing Maneuvers**
7. Perform Car Following (Including Stop and Go)
8. Detect and Respond to Stopped Vehicles
9. Detect and Respond to Lane Changes
10. Detect and Respond to Static Obstacles in the Path of the Vehicle
11. Detect Traffic Signals and Stop/Yield Signs
12. Respond to Traffic Signals and Stop/Yield Signs
13. **Navigate Intersections and Perform Turns**
14. Navigate Roundabouts
15. Navigate a Parking Lot and Locate Spaces
16. Detect and Respond to Access Restrictions (One-Way, No Turn, Ramps, etc.)
17. Detect and Respond to Work Zones and People Directing Traffic in Unplanned or Planned Events
18. **Make Appropriate Right-of-Way Decisions**
19. Follow Local and State Driving Laws
20. Follow Police/First Responder Controlling Traffic (Overriding or Acting as Traffic Control Device)
21. Follow Construction Zone Workers Controlling Traffic Patterns (Slow/Stop Sign Holders).
22. Respond to Citizens Directing Traffic After a Crash
23. Detect and Respond to Temporary Traffic Control Devices
24. Detect and Respond to Emergency Vehicles
25. Yield for Law Enforcement, EMT, Fire, and Other Emergency Vehicles at Intersections, Junctions, and Other Traffic Controlled Situations
26. Yield to Pedestrians and Bicyclists at Intersections and Crosswalks
27. Provide Safe Distance From Vehicles, Pedestrians, Bicyclists on Side of the Road
28. **Detect/Respond to Detours and/or Other Temporary Changes in Traffic Patterns**

Rigorous System Design – Model-based Validation

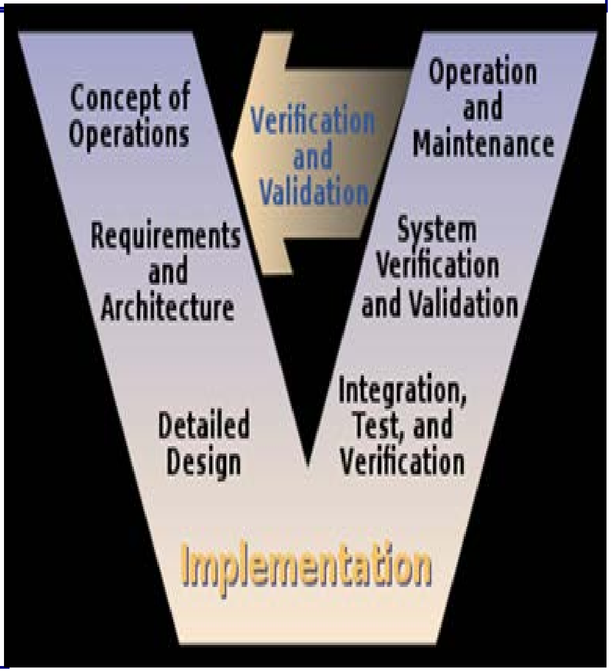
- Formal verification
  - is applicable when goals that can be explicitly formalized as requirements
  - Is tractable for moderate model complexity - only monolithic verification techniques of finite state systems can be automated;
  - Is not enough! Autonomy is about controller synthesis under both safety and optimization constraints;
  - A more natural approach is to achieve correctness by design.

- The V-model, Systems Engineering Process recommended by Safety Standards such as ISO26262

1. assumes that all the system requirements are initially known, can be clearly formulated and understood.

2. assumes that system development is top-down from a set of requirements. Nonetheless, systems are never designed from scratch; they are built by incrementally modifying existing systems and component reuse.

3. considers that global system requirements can be broken down into requirements satisfied by system components.



- Autonomous Systems
  - The concept of autonomy
  - Should we trust autonomous systems?

- In Search of a Foundation
  - "Hybrid" design flows
  - Modeling and Simulation
  - Validation

- Discussion
  - Valuing knowledge
  - The way forward

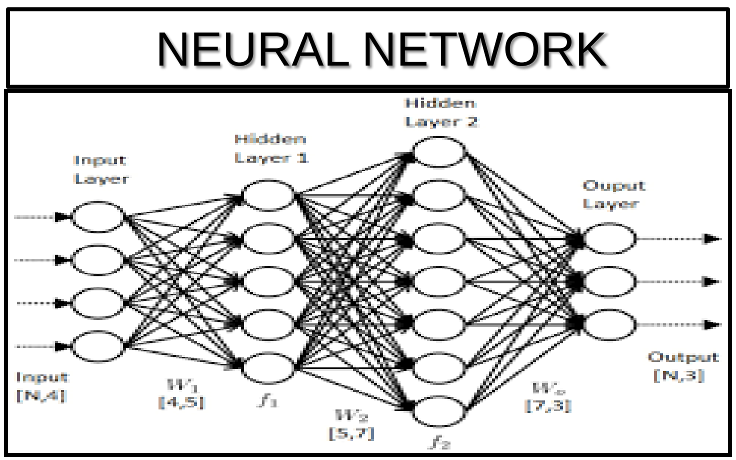
Discussion – An Interesting Analogy

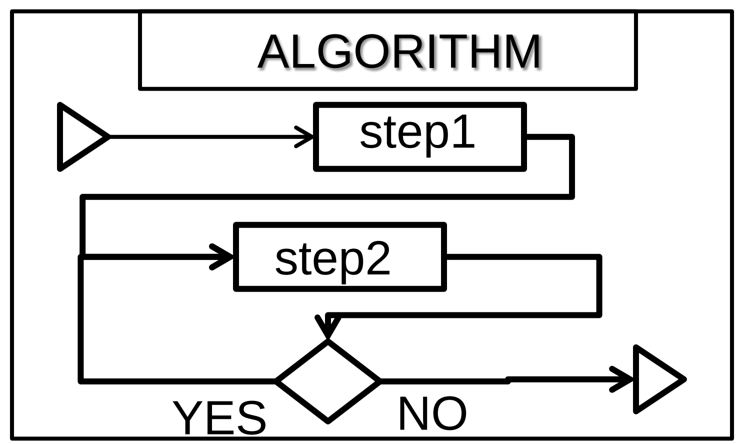
Fast thinking vs. Slow thinking (D. Kahneman's "Thinking Fast and Slow")

System 1: "Fast" Thinking
- Non-conscious – automatic – effortless;
- Without self-awareness or control;
- Handles all kind of empirical implicit knowledge e.g. walking, speaking, playing the piano

System 2: "Slow" Thinking
- Conscious – controlled– effortful;
- With self-awareness and control
- Is the source of any reasoned knowledge e.g. mathematical, scientific, technical.

Neural Networks vs. Conventional Computers

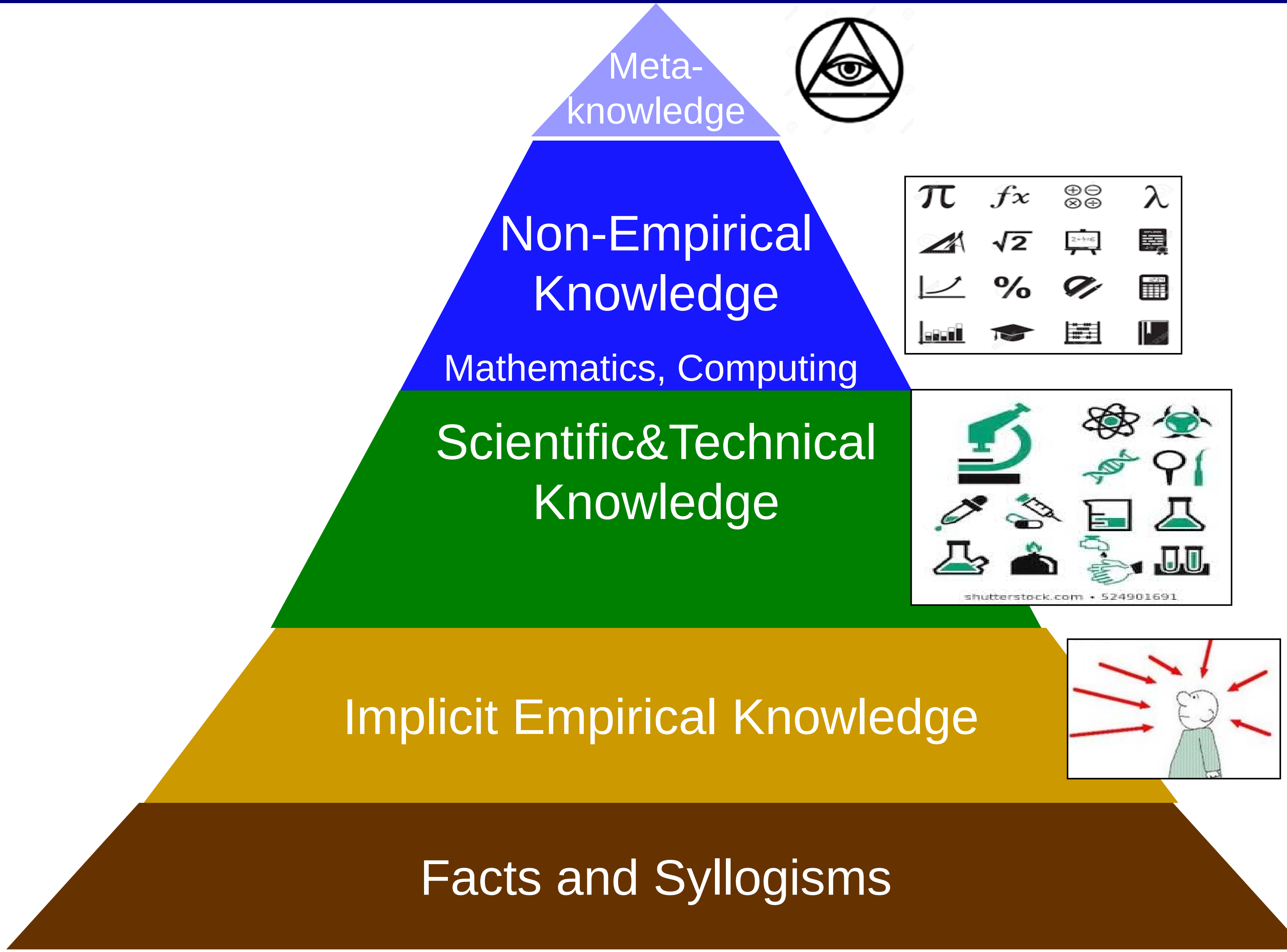




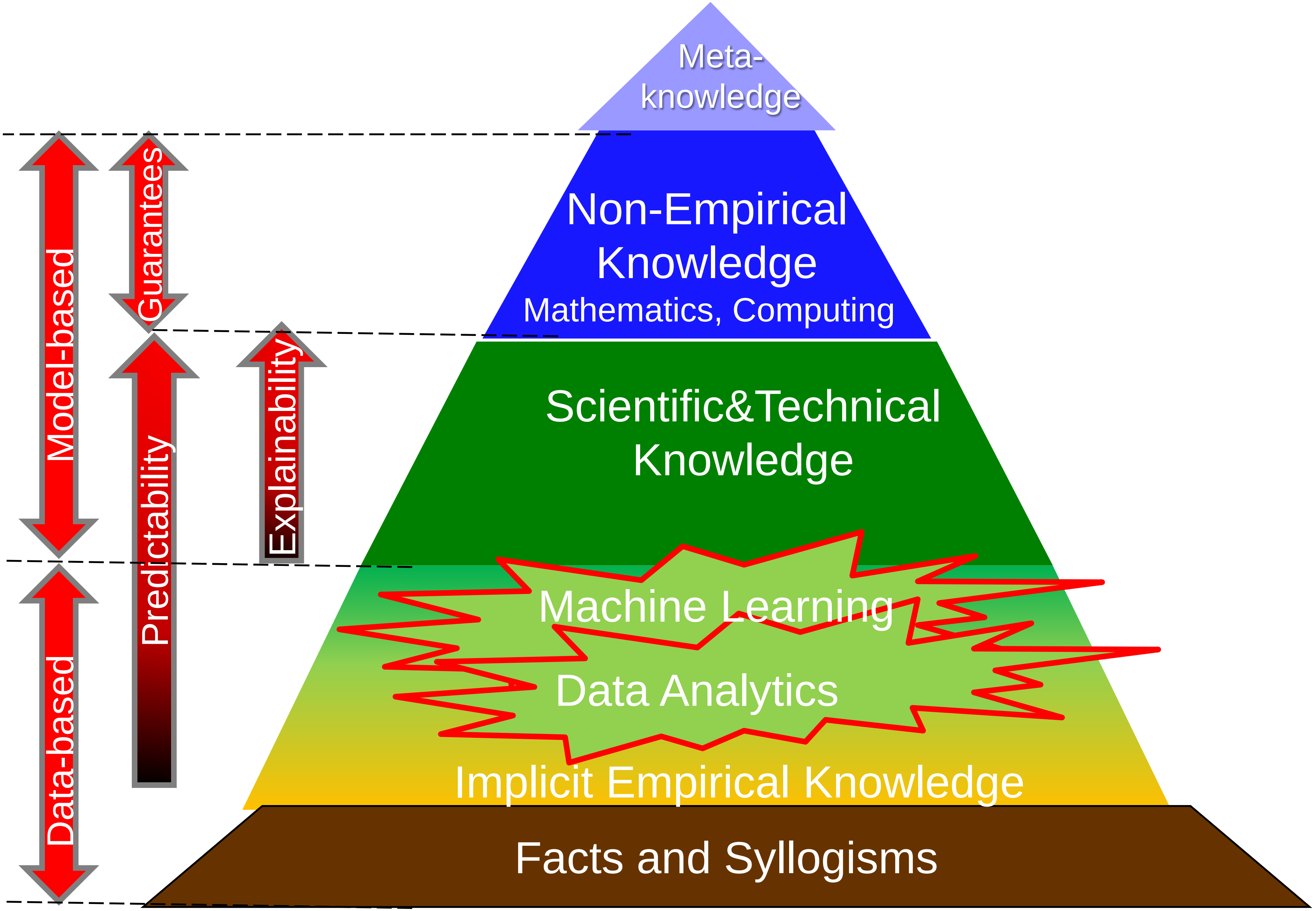
- Generate empirical knowledge after training (<u>Data-based</u> knowledge).
- Distinguish "cats from dogs" exactly as kids do – Cannot be verified!

- Execute algorithms (<u>Model-based</u> knowledge) .
- Deal with explicitly formalized knowledge – Can be verified!

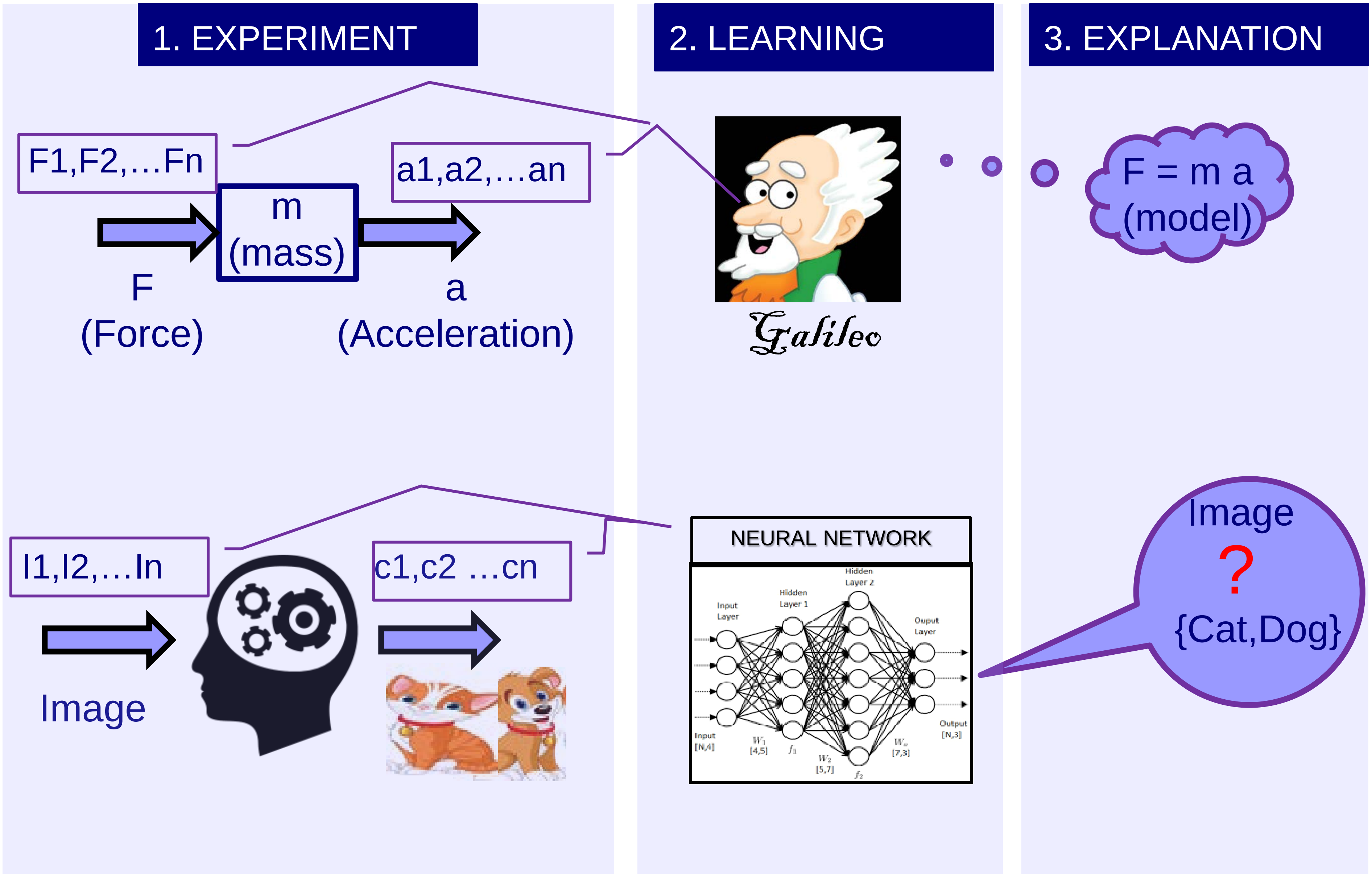
Discussion – The Knowledge Hierarchy (Before)



Discussion – The Knowledge Hierarchy (After)



Discussion – Scientific vs. ML-generated Knowledge



Discussion – Scientific vs. Machine-generated Knowledge

Limitations of the scientific approach

1. Phenomena are explainable provided we have the adequate mathematical model
2. Cognitive complexity: there is a limit in the size of the relations that human mind can deal with: relations of rank five (one predicate + four arguments)
3. We are “lucky”: basic physical laws are easy to understand !!
   BUT our lack of understanding of complex phenomena does not necessarily mean that they are not subject to laws – Simply their complexity exceeds our cognitive capabilities

Can computers help overcome these limitations?

Geophysical Research Letters

RESEARCH LETTER

10.1002/2017GL074677

Machine Learning Predicts Laboratory Earthquakes

Bertrand Rouet-Leduc[1,2], **Claudia Hulbert**[1], **Nicholas Lubbers**[1,3], **Kipton Barros**[1], **Colin J. Humphreys**[2], and **Paul A. Johnson**[4]

[1]Theoretical Division and CNLS, Los Alamos National Laboratory, Los Alamos, NM, USA, [2]Department of Materials Science and Metallurgy, University of Cambridge, Cambridge, UK, [3]Department of Physics, Boston University, Boston, MA, USA, [4]Geophysics Group, Los Alamos National Laboratory, Los Alamos, NM, USA

Key Points:

- Machine learning appears to discern the frictional state when applied to laboratory seismic data recorded during a shear experiment
- Machine learning uses statistical

OVERVIEW

- Autonomous Systems
  - The concept of autonomy
  - Should we trust autonomous systems?

- In Search of a Foundation
  - “Hybrid” design flows
  - Modeling and Simulation
  - Validation

- Complexity Issues
  - Autonomic Complexity
  - Design Complexity

- Discussion
  - The value of knowledge
  - The way forward

Discussion – Standards for Next-Gen Autonomous Systems

- Autonomy should be associated with functionality and not with specific techniques – while ML is essential it is not only way to build perceptors and adaptive controllers.

- Current trends render obsolete conventional critical systems engineering principles and standards such as such as ISO26262 and DO178B , that require <u>*conclusive evidence*</u> that the system can cope with any type of harmful event.
  - <u>they cannot</u> handle machine learning components;
  - <u>they cannot</u> handle design flows for autonomous systems – they give a system credit for a human assistant ultimately being responsible for safety.
  - they require guarantees at design time and stringent predictability that are impossible to provide IoT autonomous systems.

- Consequently, there is no Independent safety certification for autonomous systems!

  - Automotive and medical products are self-certified by their manufacturers according to guidelines that determine how to provide <u>*sufficient evidence*</u> that the developed system is reliable enough.

Discussion – Should be worried about dystopian AI futures?

The role of AI systems will depend on choices we make about when we trust them and when we do not. Making these choices wisely

1. is a matter of social awareness and of sense of political responsibility:
   - When machines use knowledge in critical decision processes make sure that it is truthful, unbiased, neutral, fair, etc. (<u>precautionary principle</u>).
   - Always question motives, objectives and biases of existing systems.

2. requires <u>new scientific foundations</u> allowing the development of trust evaluation tools
   - We need a “new kind of scientific approach” based on a « hybrid » model-based and data-based approach seeking tradeoffs between trustworthiness and performance.
   - We should develop and apply rigorous regulations and standards for the development and use of such systems (as for all artifacts from toasters to bridges and aircraft).
     <u>No self-regulation</u>, <u>no self-certification</u> !!

<u>**Building trustworthy next-generation autonomous systems goes for far beyond the current AI challenge.**</u>

Thank You

Joseph Sifakis
Autonomous Systems -- An Architectural Characterization,
November 2018
https://arxiv.org/abs/1811.10277