Autonomous Systems

Académie des Technologies
24 Juin 2025

Joseph Sifakis
Verimag

Where We Are? – The AI Vision and Its Goals

At present, there's a great deal of confusion about the ultimate goal of AI, fuelled by the media and Big tech companies, who, through grandiose large-scale projects, spread opinions suggesting that human-level AI is only a matter of years away.

Opinions are divided between two very different positions.

- Some AI research and companies such as OpenAI and DeepMind see AGI, as the ultimate goal ***" AGI is a system that can perform any intellectual task a human can, at human-level or beyond*** " S. Altman Feb. 2025.
  - AGI definitions misleadingly suggest that human intelligence can be defined as the ability to perform an undefined set of distinct tasks. Which tasks, exactly how many tasks?
  - AGI definitions suggest that it can be achieved through ML and its further developments i.e ML is the "end of the story" Scaling up model size (parameters, data, compute) inevitably would lead to higher intelligence.

It is surprising that such a poorly defined vision has been blindly adopted by experts who, even take the risk of making predictions about its imminent arrival (it's already here, tomorrow, in five years...).

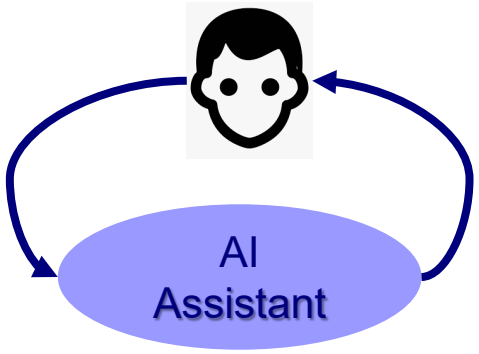
- Others see the goal of AI as building machines with Human-level Intelligence, which requires agreement on what human intelligence is and, more importantly, on methods for comparing human and machine intelligence.
  - According to the Oxford dictionary, intelligence is defined as
    "the ability to learn, understand and think in a logical way about things; the ability to do this well"
  - Machines can perform impressive tasks, outperforming humans in their execution, but there is no evidence that they can surpass humans in terms of situational awareness, adaptation to changes in their environment, and creative thinking.

Without a clear idea of what intelligence is, we cannot develop a theory of how it works!

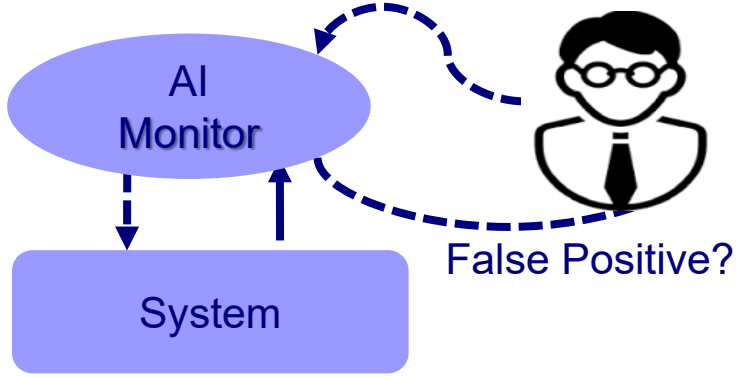
Where We Are? – From Conversational to Autonomous AI

- AI is still in its infancy, despite impressive results culminating in the arrival of generative AI,
  - it only gives us the elements to build intelligent systems, but we don't have the principles and techniques to synthesise them, for example in the way we construct bridges and buildings.
  - It mainly focuses on assistants, while future applications require continuous interaction with little or no human intervention.

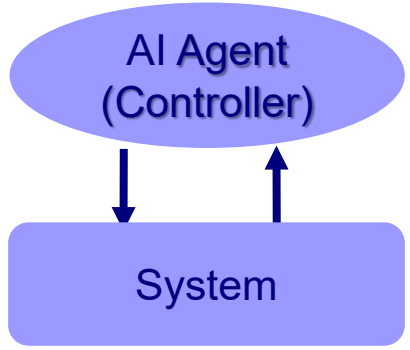
- Three different ways to use AI systems:
  1. Assistants that in interaction with a user, provide a given service;
  2. Monitors of a system behavior synthesizing knowledge to detect or predict critical situations;
  3. Agents (controllers) of a system so that its behavior meets a given set of requirements, e.g. autopilot of autonomous car.



Conversational AI



Predictive/Analytical AI



Autonomous AI

- The impact of AI on the real economy remains limited compared to the enormous potential of AI applications
  - The AI industry revolution has only just begun!
  - Its realisation depends largely on our ability to develop AI-enabled agents to build autonomous systems.

Where We Are? – The Vision for Autonomous and Trustworthy AI

- Autonomous systems stem from the need to replace human operators in complex organizations as envisioned by the IoT e.g, self-driving systems, smart grids, smart factories, autonomous telecommunication systems
  - are composed of agents each pursuing their own goals (individual intelligence) while coordinating to meet global system goals (collective intelligence);
  - are able to perceive and predict changes of their environment and adapt to the constantly changing environments and user requirements by managing possible conflicting goals;
  - are highly complex, often critical distributed dynamic reconfigurable systems that never stop and evolve.

- A key issue concerning the extensive use of AI systems, reputed to be black boxes, is to guarantee their trustworthiness.
  - Safe AI has been the subject of international summits and UN deliberations, marked by naive optimism and ignorance of the fact that LLM safety cannot be guaranteed in the same way as that of an elevator, an airplane, or a car.
  - Human-centric cognitive properties, in addition to behavioral properties, are the subject of numerous studies.
    - "Responsible AI" implies that AI systems meet properties such as fairness, reliability, safety, privacy and security, inclusiveness, transparency, and accountability, difficult, if not impossible, to assess.
    - "AI alignment" meaning alignment with human ethical values while we do not even understand how human will emerges and the associated value-based decision-making system works.

But all this work lacks foundation, because it ignores a basic epistemic principle: *any claim that a system satisfies a property must be backed up by a rigorous method of validation.*

Overview

- Important Clarifications
- Agent Reference Architecture
- Agent Implementations Issues
- Validation of AI Systems
- Where Are We Going?

Important Clarifications – Technical vs. Non-technical Systems

- Technical systems: their I/O relation can be unambiguously characterized as a relationship between mathematical domains.

  Traditional ICT systems are technical systems
  - Their trustworthiness can be formulated as predicates P(x,y) that can be validated by testing.
- Non-technical systems: their input or output domains are defined in terms of sensory or linguistic data.
  - These are systems that mimic human functions not amenable to formalization, e.g. ChatGPT, image analysis system.
  - Their behavioral properties cannot be validated as rigorously as those of technical systems., e.g, safe GPT.
  - Their trustworthiness encompasses human-centric properties determining their degree of alignment with human values.

Type of System / I/O relation	Transformer	Predictor	Analyzer	Controller (Agent)	
		TRADITIONAL SYSTEMS		Static	Dynamic
Technical system Mathematically well-defined	Functions, Combinatorial circuits	Predictive statistical models, e.g. weather forecast	Static analyzers, Model-checker Symbolic solver	Thermostat, Lift controller, Flight controller	Chess playing system, MPC system, Dynamic decision making.
Non technical system Involves Linguistic or Sensory data	Conversational system, Classifier	Regression models, Time series analysis, Decision trees	Root cause analysis systems, Clustering system	Language-controlled manipulation, Object searching	Collaborative agent, Football robot, Self-driving autopilot. Drones
		AI SYSTEMS			AUTONOMOUS SYSTEMS

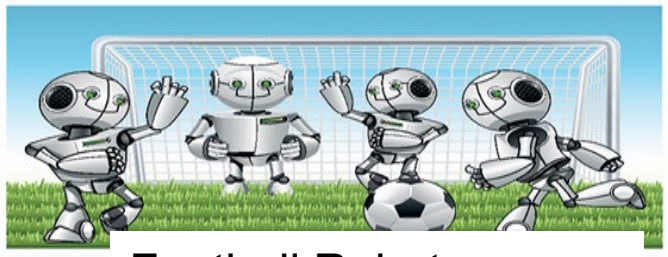
- AI enables the construction of technical and non-technical systems that would be impossible to achieve with ICT alone.!

Important Clarifications – Autonomous vs. Automated Systems

Thermostat

Automated Shuttle

Chess Robots



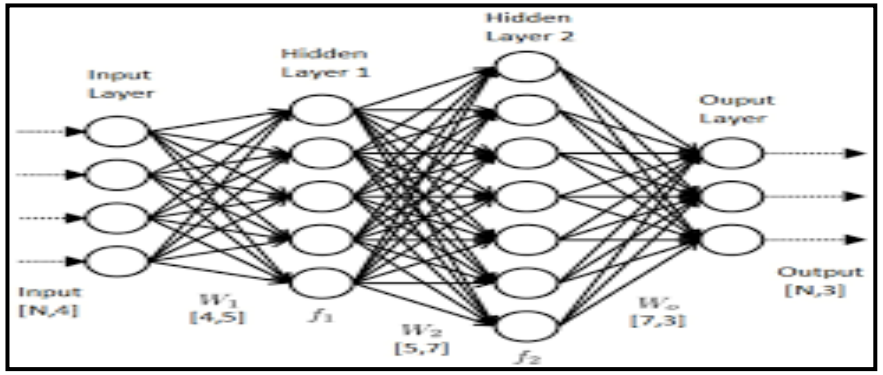
Football Robots

Autonomous Car

	Thermostat	Shuttle	Chess Robot	Football Robot	Autonomous Car
SITUATION AWARENESS	Temperature (number)	Distance from next stop (number)	Pawns on the board (static image)	Players on the pitch (dynamic image)	Obstacles on the road (dynamic image)
DECISION MAKING	Static goals <18 → ON >22 → OFF	Static goals ▪ Stop ▪ Accelerate ▪ Decelerate	Static well-defined goals Dynamic planning of goals	Dynamic multiple goals Dynamic planning of goals	Dynamic multiple conflicting goals Dynamic planning of goals

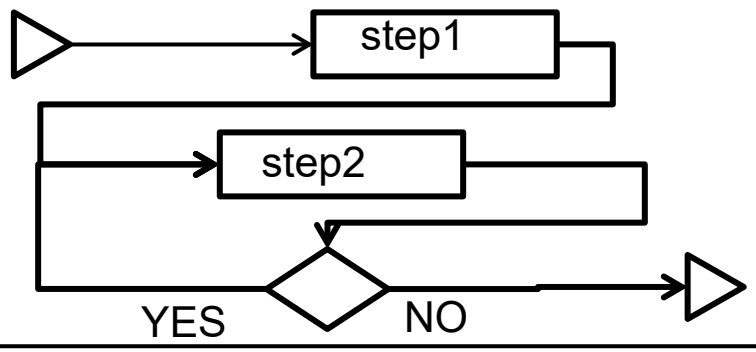
Important Clarifications – Neural Networks vs. Traditional Digital Systems

NEURAL NETWORK



- Generate empirical knowledge after training (Data-based knowledge)
- Are “black-box” not explainable.

COMPUTER



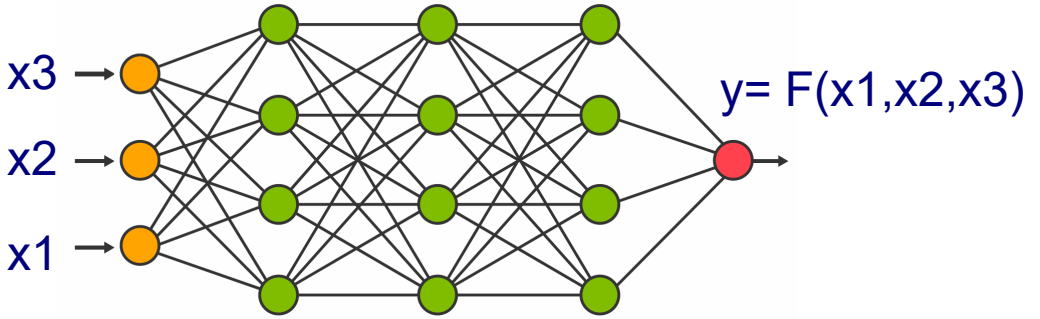
- Execute algorithms.
- Deal with explicit model-based knowledge.
- Can be understood and verified!

- Neural networks are artifacts, not models! Models are
  - representations of things that we use to explain and understand them.
  - essential for science and engineering: they enable us to reason about the things represented.
- Neural Networks do not execute algorithms, we use algorithms to train them!
- There is a remarkable analogy between the two computing paradigms and Kahneman's two systems of thinking:
  - System 1: fast automated thinking, dealing with implicit knowledge;
  - System 2: slow conscious thinking, dealing with explicit knowledge.

Important Clarifications – AI Explainability

- A system is explainable if its behavior can be described by a model that lends itself to reasoning and analysis. Models are usually built following a compositionality principle:
  - In scientific disciplines, explainability is based on mathematical models, such as differential equations and statistical models.
  - For traditional digital systems, explainability is usually based on discrete models, such as transition systems.

- NN explainability : characterize the I/O behavior of a NN by a model obtained as the composition of the behavior of its elements.

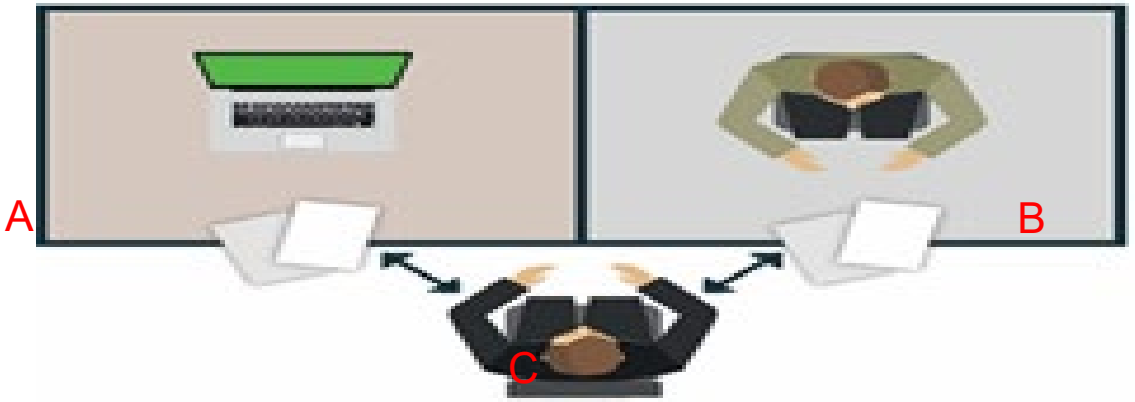


  - For feed-forward networks, it is theoretically possible to calculate the output as a function F of the inputs, given the functions calculated by each node: *φ(weighted_sum_of_inputs),* where φ is an activation function.
  - However, the approach does not scale up for NN's in real-life applications. Only for classes of small feed-forward NNs with simple activation functions, approximations of F can be computed.

Note: Other, weaker notions of explainability fail to provide rigorous characterization sufficient to guarantee safety properties, e.g., extracting a textual description of behavior or decomposing into informally specified elements.

Important Clarifications – Behavioral Intelligence Tests

- Turing Test (Imitation Game):
  1. C sends questions to A and B who, in turn, provide a corresponding answer to each question.
  2. If C cannot tell which is the computer and which the person, then A and B are equally intelligent.



- Criticism:
  - Success depends on human judgement (subjective) and the choice of the test cases (questions).
  - The test cannot be a question/answer game - much of human intelligence is expressed by interaction with the environment (speech, movement, social behavior, etc.)

- Replacement test: *An agent A (indifferently machine or human) is as intelligent as an agent B performing a given task characterized by given well-founded success criteria, if A can successfully replace B. e.g.*
  - a machine is as intelligent as a human driver is if it can successfully replace the driver.
  - a human is as intelligent as a janitor robot if it can successfully replace the robot according to given cleaning criteria.

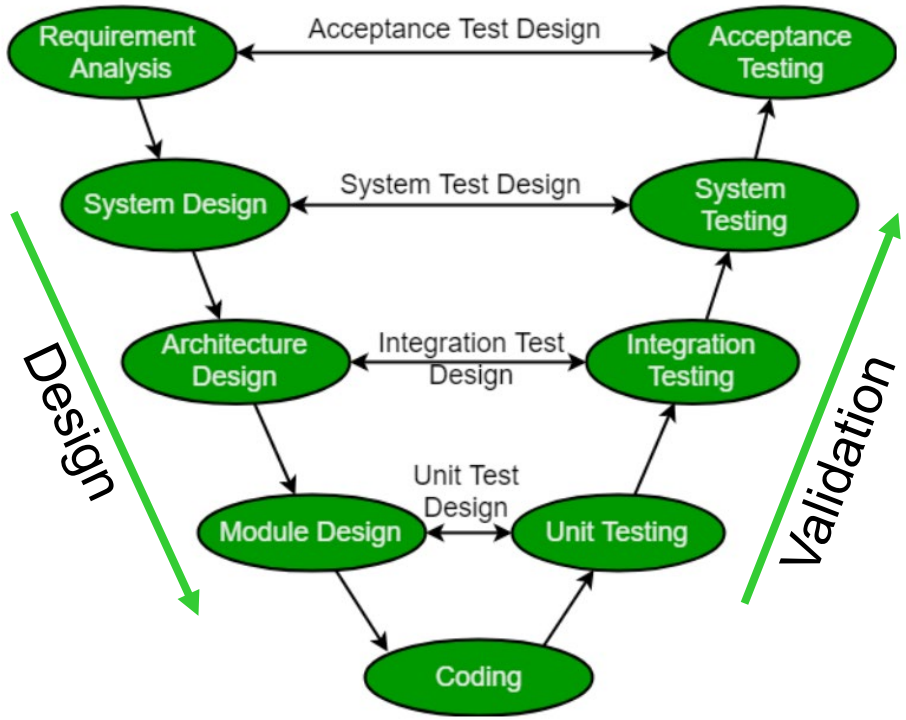
Note that the Turing test is a special case, where the task is a conversation game.

- Behavioral tests are necessary but not sufficient for comparing human and machine intelligence as they cannot take into account cognitive properties – see John Searle's Chinese Room Argument (1980).

Important Clarifications – Traditional vs. AI System Development

❑ Traditional system development

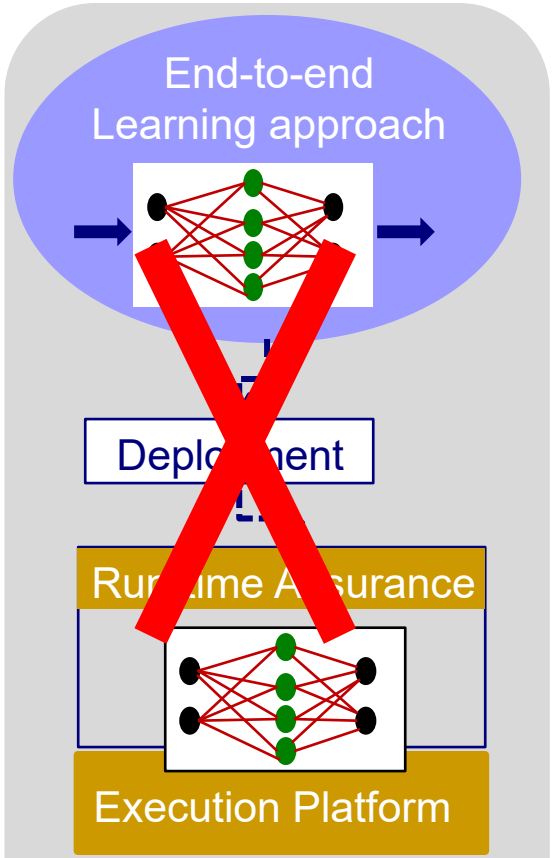
- Requirements are analyzed and broken down into properties satisfied by the system constituents and characterizing system trustworthiness.
- The design flow involves well defined steps leading to an architecture integrating components that are amenable to mathematical analysis.
- Validation techniques combine verification and testing; they allow estimating system trustworthiness e.g. 10^{-9} failures per hour of flight .



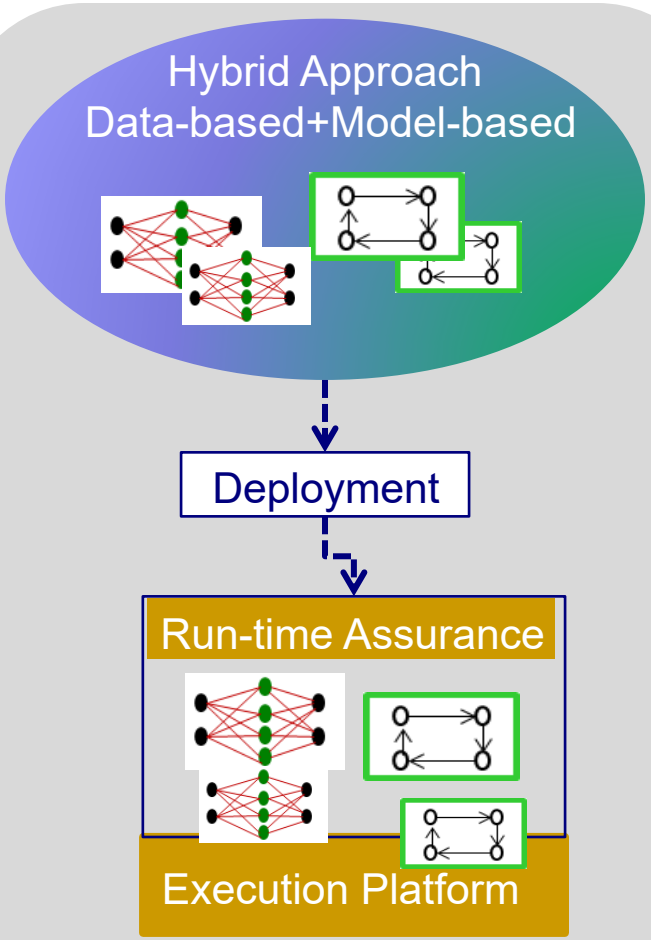
❑ AI development

- holistic empirical approach that aims at mimicking a human function by approximating relations in multidimensional data.
- involves 1) data acquisition and preparation; 2) system development and training; 3) system evaluation and improvement; 4) deployment.
- can be hardly understood and analyzed as the composition of components – non explainability.

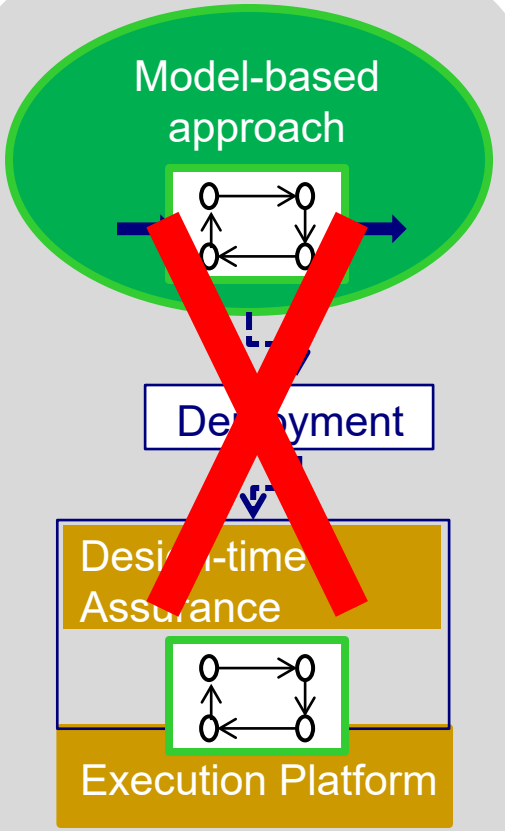
Important Clarifications – Hybrid Architectures



ML-ENABLED SYSTEM
(PERFORMANCE)



HYBRID SYSTEM
(TRADE OFF: PERFORMANCE vs.
TRUSTWORTHINESS)



AUTOMATED SYSTEM
(TRUSTWORTHINESS)

Overview

- Important Clarifications
- Agent Reference Architecture
- Agent Implementations Issues
- Validation of AI Systems
- Where Are We Going?

Agent Reference Architecture – AI Agents

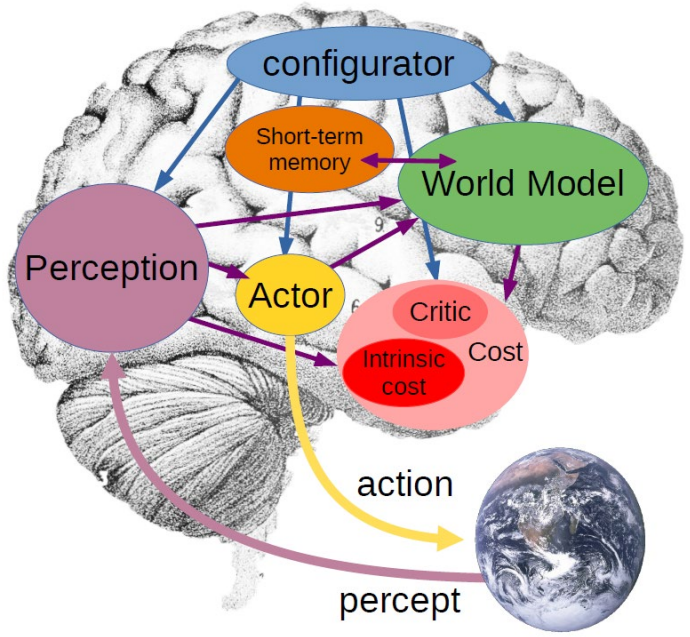
- With the advent of LLMs, AI agents have attracted growing interest, used to provide a service by
  - Perceiving changes in their world (environment) and estimating its state;
  - Predicting and planning actions that change the state of the world so as to satisfy given goals.

Currently, AI agents can provide low-reliability services, mainly by interacting with static digital worlds, e.g. business processes, which is insufficient to cover the needs for autonomous agents.

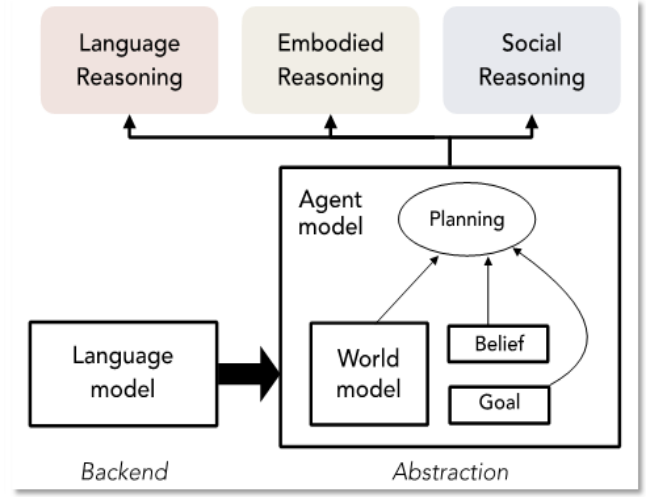
- Although there is broad consensus that symbolic computation is essential to building agents, there is no agreement on the approach to take.
  - Some believe that common sense (superintelligence) can emerge through learning experience e.g. with increasingly powerful machines (*scale is all you need!*), e.g. primarily solutions based on DL and RL (without LLMs).
    - World models are internal representations that an AI system uses to analyse, simulate, predict, and reason about its environment.
  - Others hold that symbolic reasoning must be a core function from the outset, advocating neurosymbolic AI to bridge the gap between neural networks and symbolic knowledge representation, primarily solutions that improve LLMs.
    - LLMs using symbolic engines such as AlphaGeometry, WolframAlpha, simulators, probabilistic programming tools;
    - LLMs using knowledge stored in long-term memory, e.g. Retrieval-Augmented Generation (RAG).

- Regardless of the approach taken, current approaches
  - are moving away from monolithic end-to-end solutions;
  - are based on architectures that integrate functional elements ensuring basic cognitive functions and, as needed, short- and long-term memory.

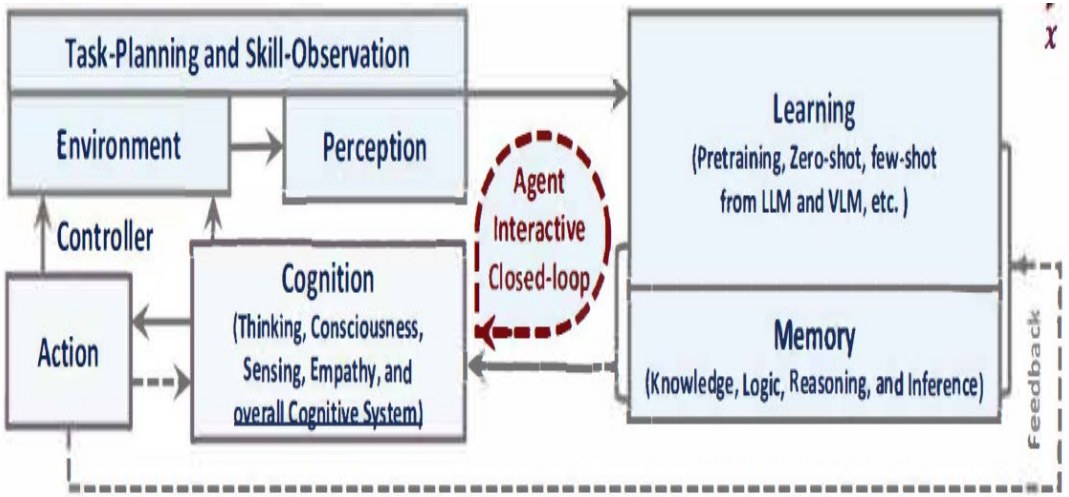
Agent Reference Architecture – Agent Architectures



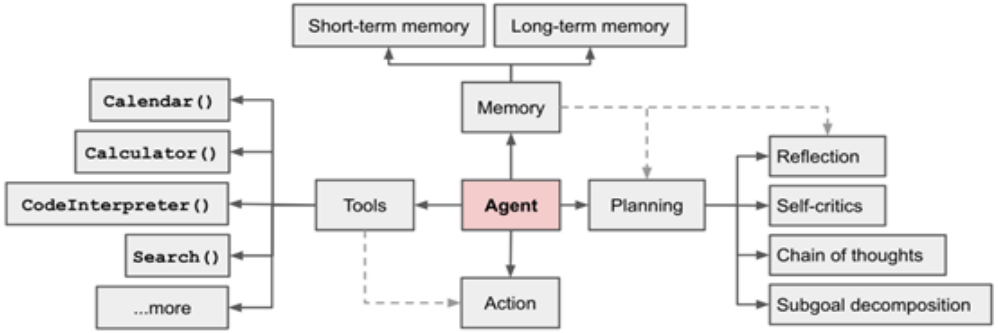
(a)



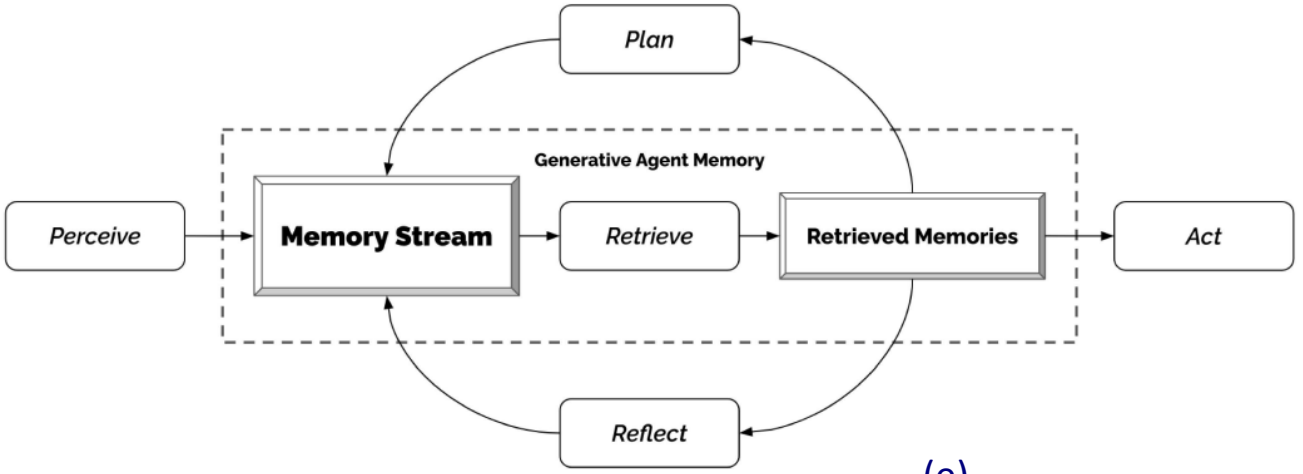
(b)



(c)



(d)



(e)

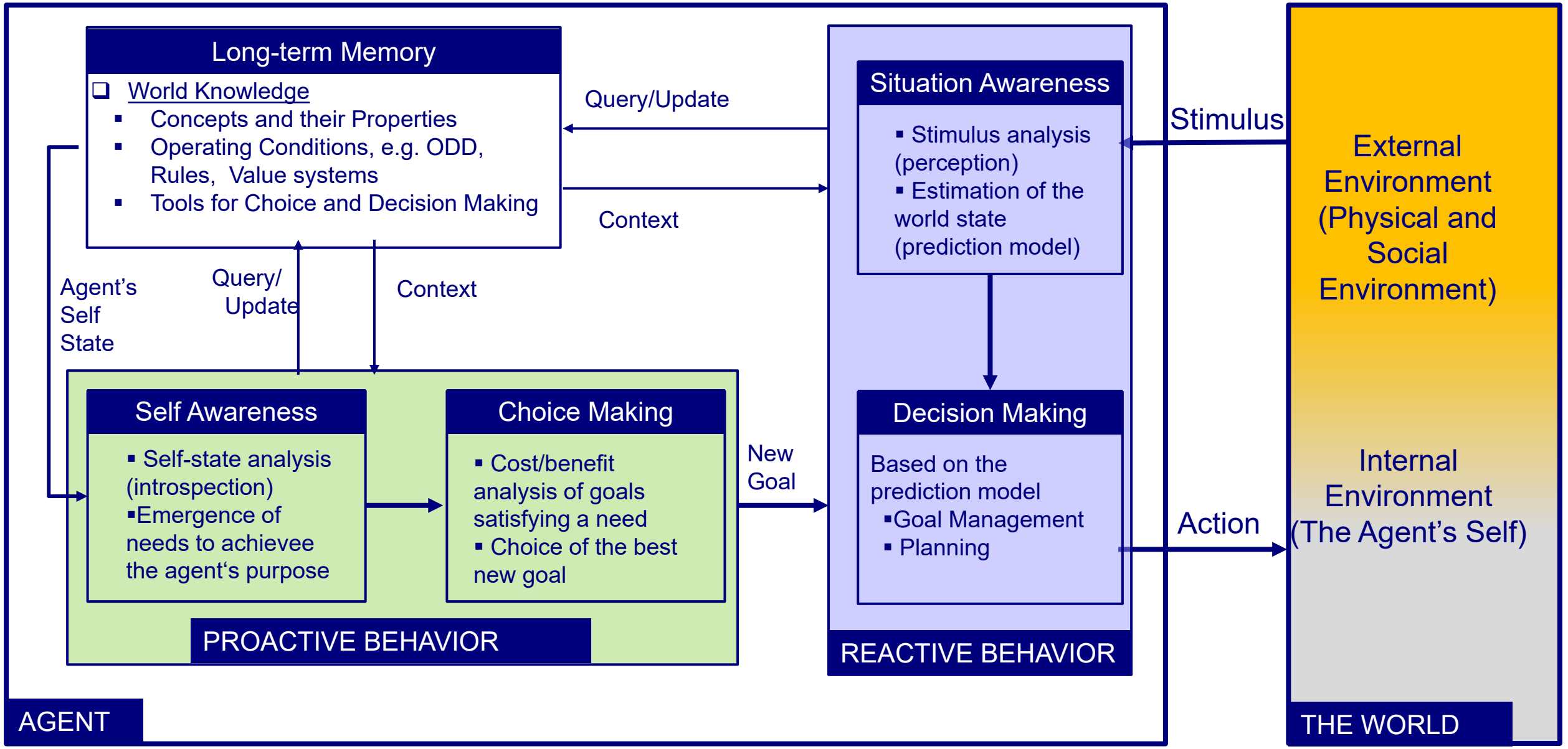
Agent Reference Architecture – Principles

- In accordance with a fundamental paradigm of systems engineering, we use an Agent Reference Architecture that generalizes existing agent solutions,
  - integrating a set of basic cognitive functions that are mutually independent and mathematically defined, without reference to any particular implementation;
  - showing sufficiently general and complete behavior, covering as far as possible various aspects of human mental processes, in particular Kahneman's two systems of thinking.

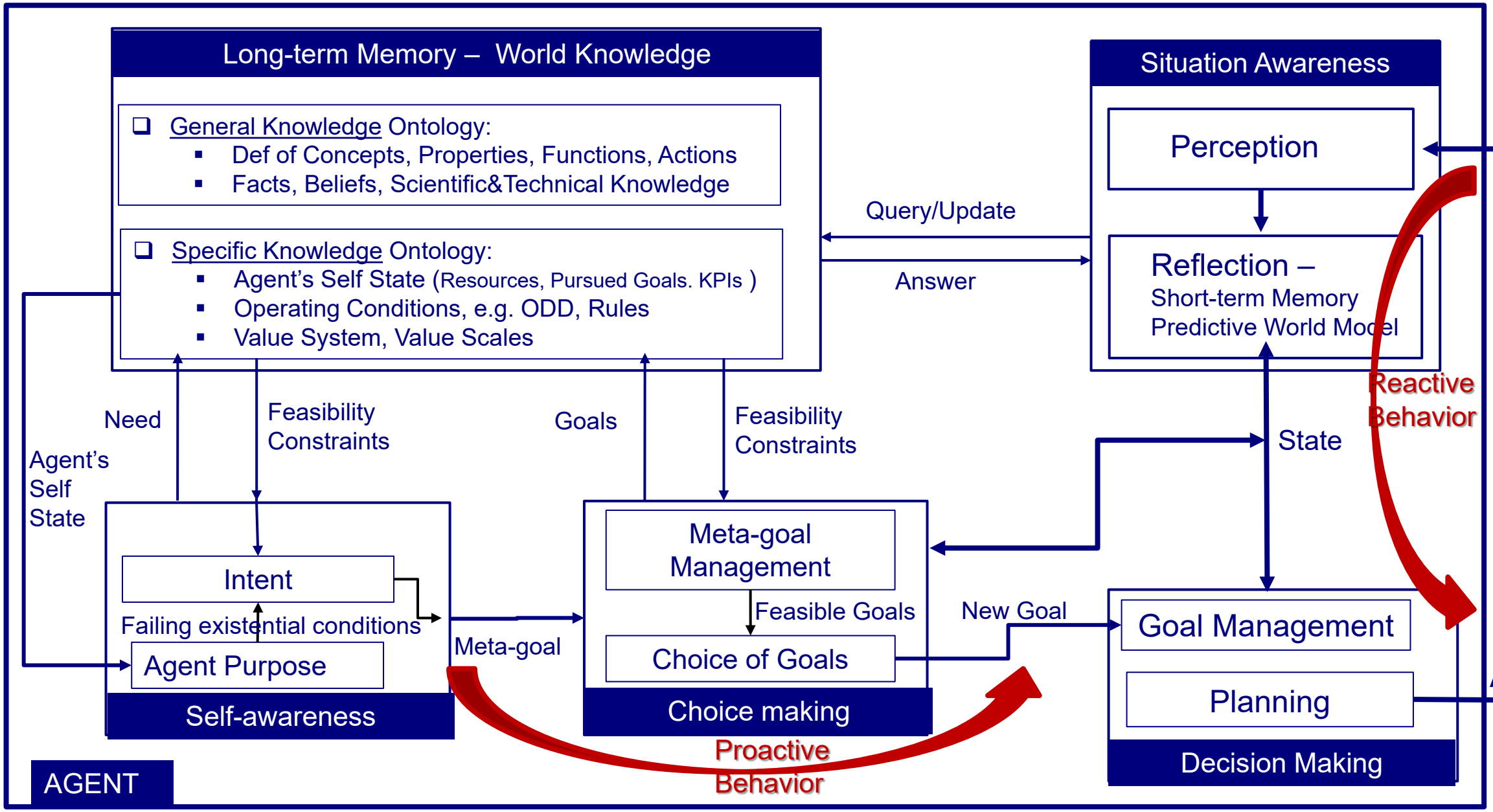
- An agent Reference Architecture could serve as a basis for
  - characterizing autonomy as the composition of functions from which can emerge an intelligent behavior;
  - the comparison of various agent solutions and their degree of coverage of model features
    - ranging from end-to-end machine learning to hybrid architectures;
    - targeting specific application domains such as ADS, Robotics, business decision support, gaming, etc.
  - addressing agent correctness through compositional reasoning and separate validation of the agent's functions.

- Furthermore, the Reference Architecture must enable comparisons between artificial agents and humans by modeling and analyzing mental processes with their underlying cognitive mechanisms as computational processes.
  - How do goals emerge from the intent to satisfy needs?
  - Which factors guide the resolution of conflicts between goals and actions?
  - What is the role of knowledge, beliefs, value systems, normative rules?
  - What it means to act ethically, responsibly, rationally, etc.

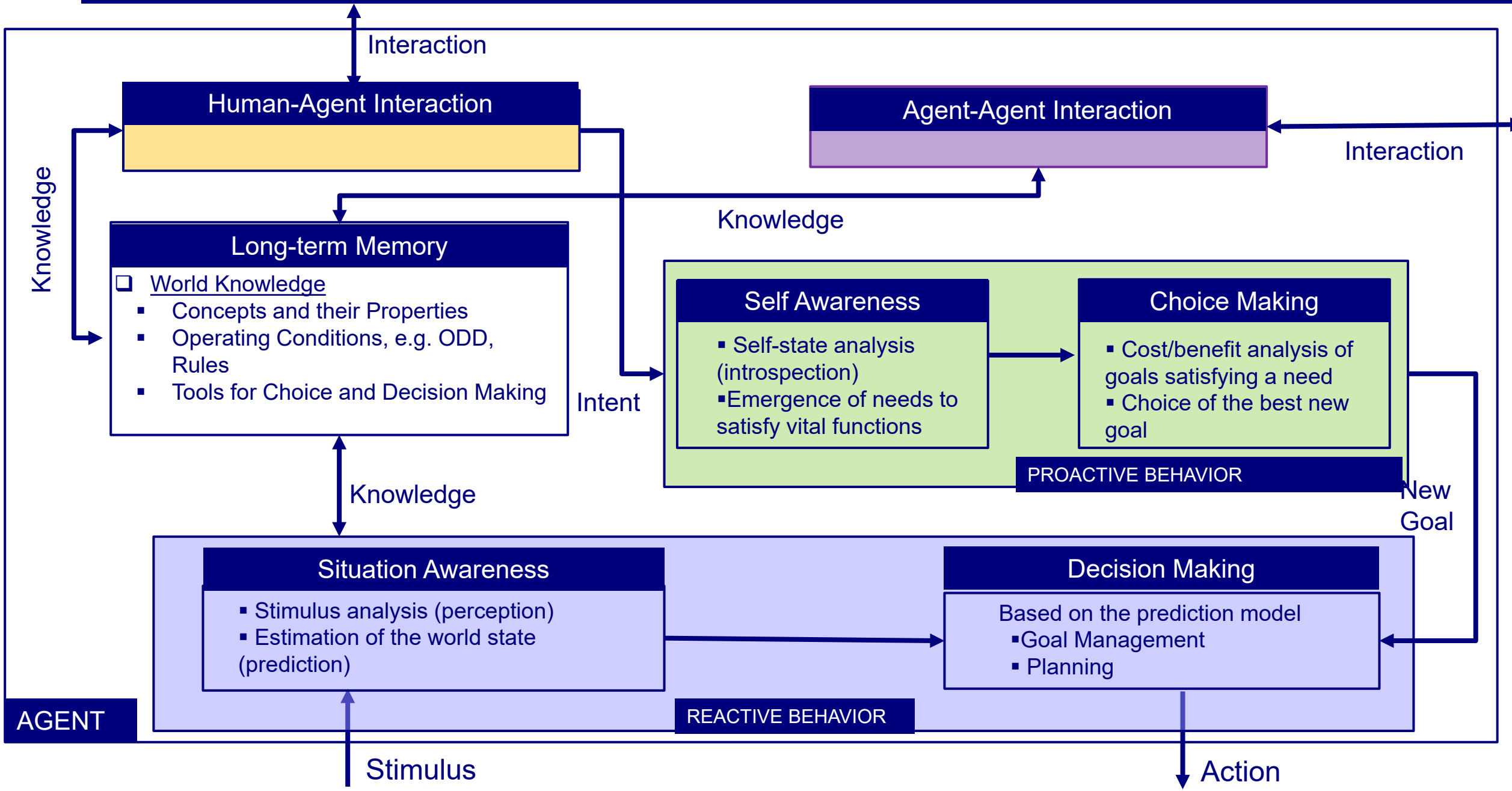
Agent Reference Architecture – General View



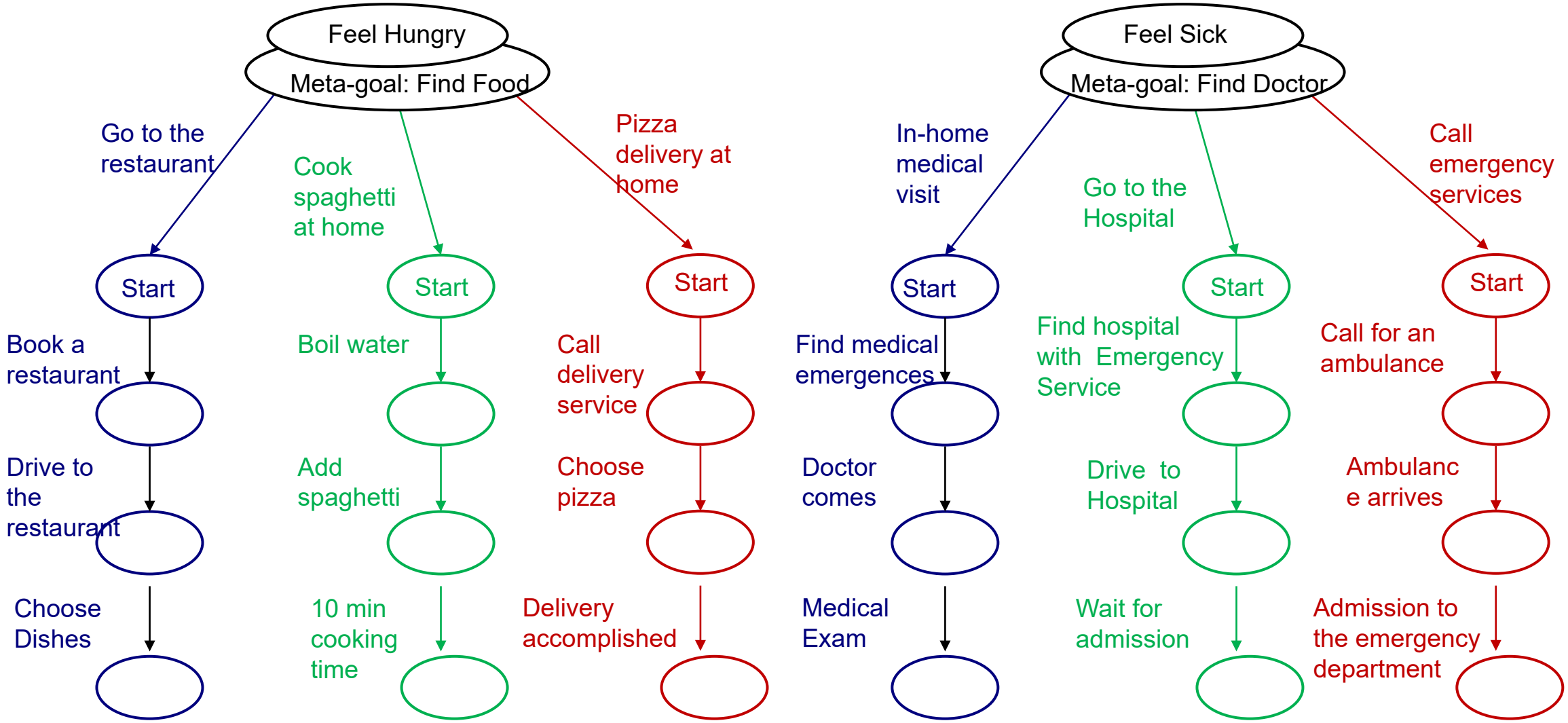
Agent Reference Architecture – Detailed View (1)



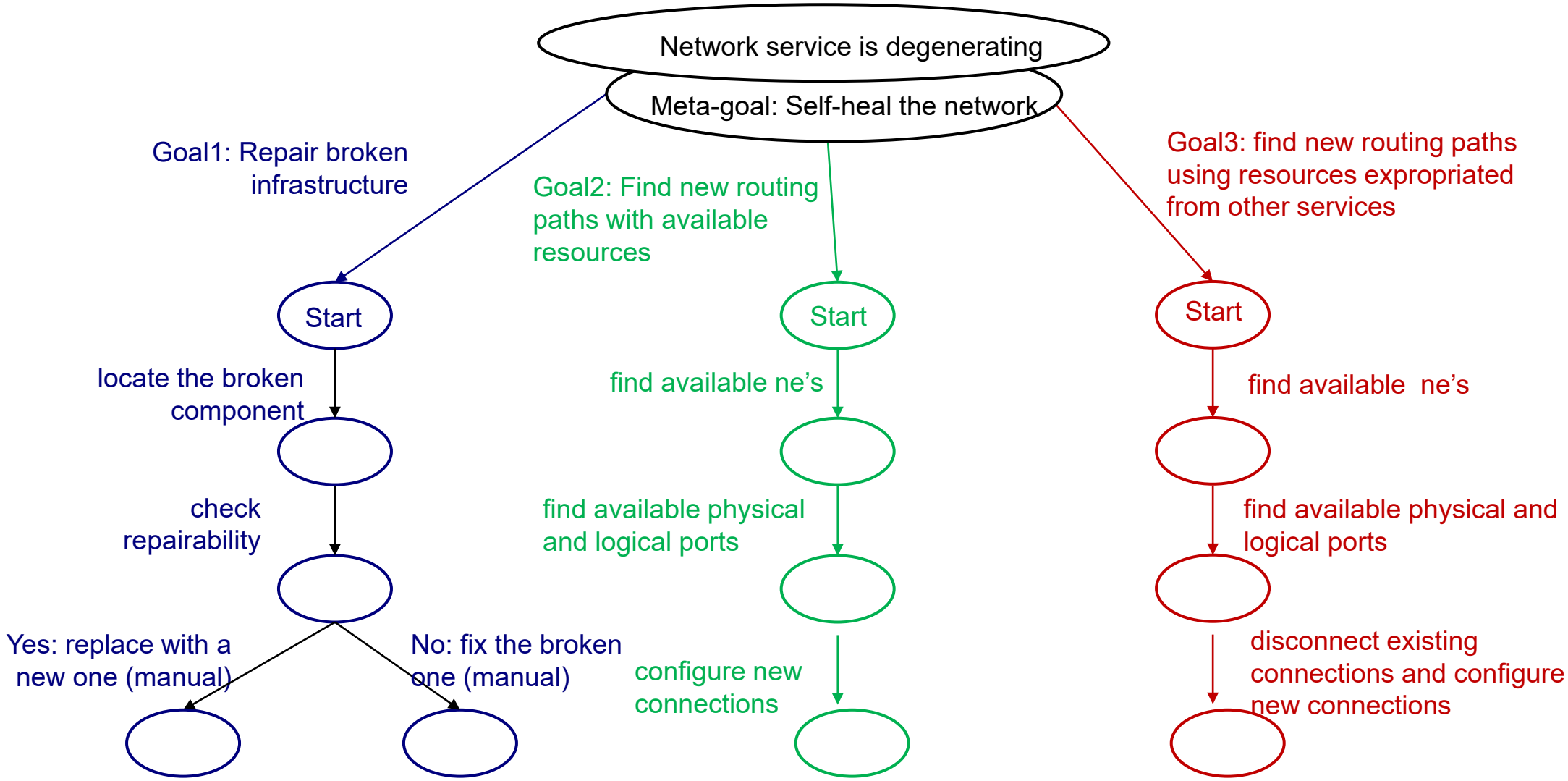
Agent Reference Architecture – Communicating Agent



Agent Reference Architecture – Proactive Behavior: Meta-goals and Goals



Agent Reference Architecture – Proactive Beahvior: Meta-goals and Goals

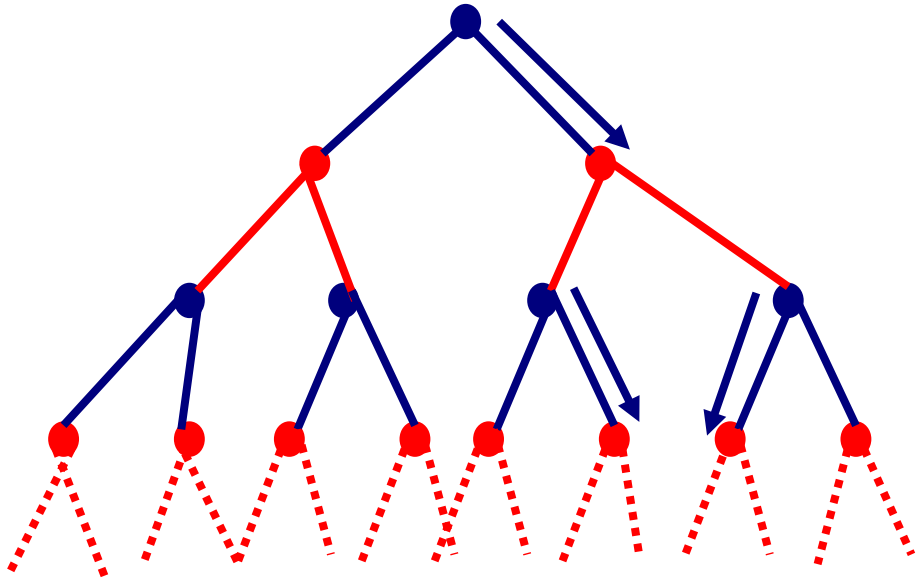


Overview

- Important Clarifications
- Agent Reference Architecture
- Agent Implementations Issues
- Validation of AI Systems
- Where Are We Going?

Agent Implementations Issues – Decision Making

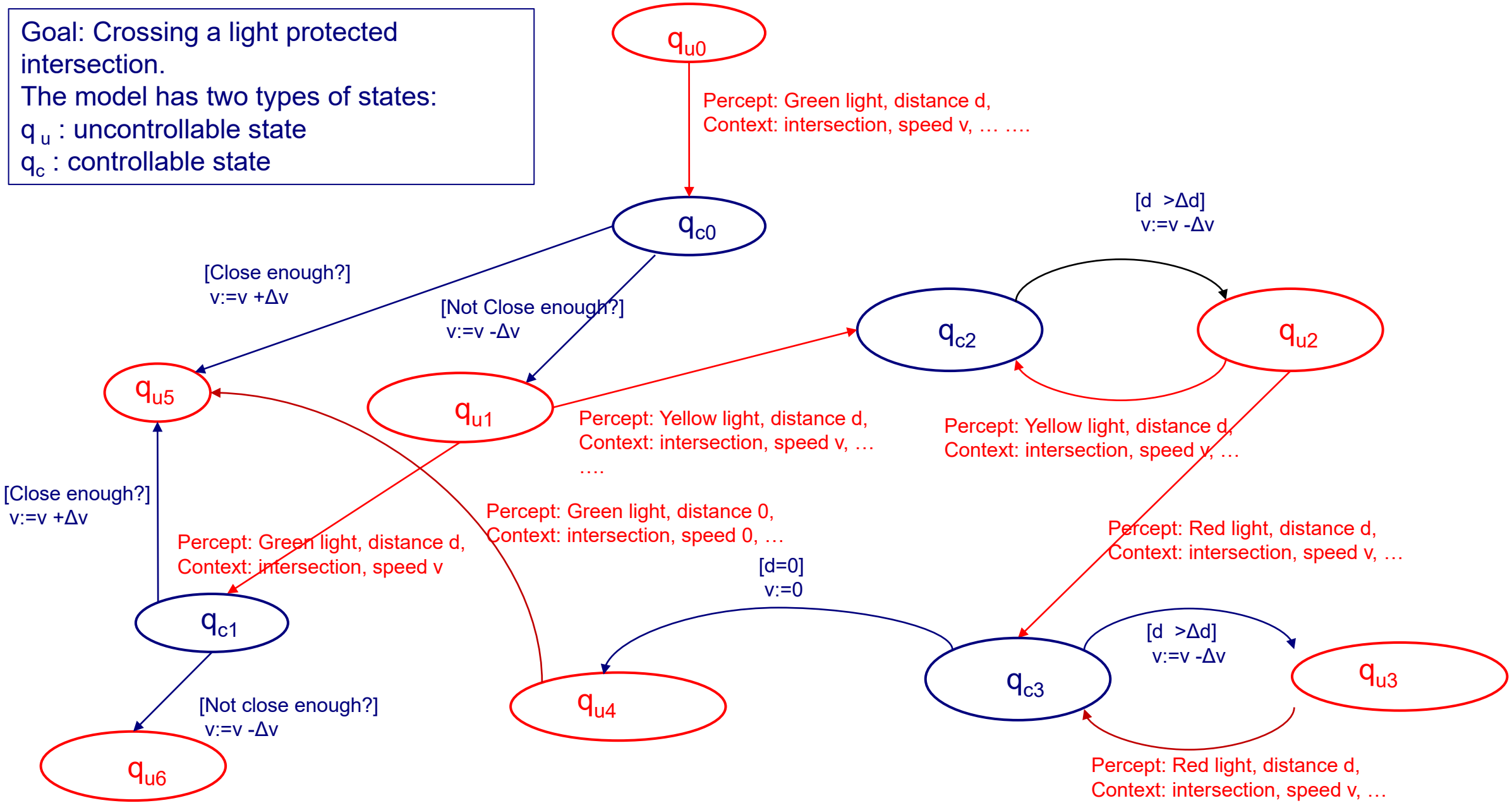
- Machine learning techniques are largely inadequate for agent reactive goal management and planning - it is naïve to try to solve control or coordination problems using CoT or ToT.
  - The agent predictive model is a state transition system, a lookahead tree, intended to represent the step by step interaction between the agent and its environment. Its behavior models all the possible interactions as a two-player game.
    - Building the predictive model involves exponential complexity – only when the rules of the game are well-defined, we can compute approximation e.g. by using Monte Carlo Tree Search (MCTS) algorithms as in AlphaGo.
    - Explicitly computing winning policies (sub-trees of the predictive model) requires a very expensive algorithmic analysis, similar to model checking – approximations will not work for critical applications where precision is required e.g. self-driving cars.



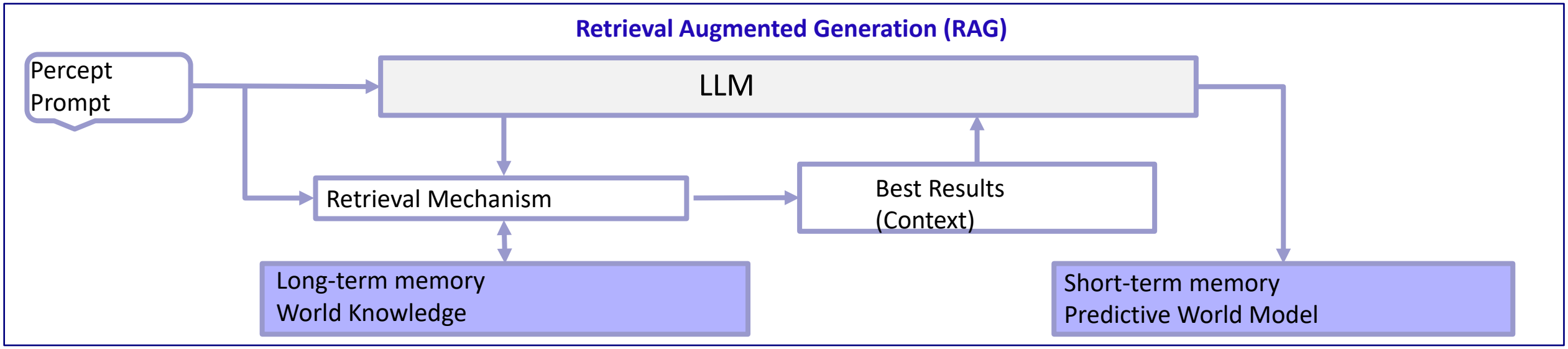
- Many works overlook the very important fact that achieving goals implies satisfaction of both safety and reachability/optimization properties - RL is a powerful tool for optimization problems, but it has a critical limitation: it cannot guarantee safety or avoid dangerous situations.
- We need to develop reactive decision-making components integrated into cyber-physical environments designed for different application domains, where planning is performed by real-time software based on knowledge provided by AI.

Agent Implementations Issues – Fragment of Controller for Self-driving Vehicle

Goal: Crossing a light protected intersection.
The model has two types of states:
q_u : uncontrollable state
q_c : controllable state



Agent Implementations Issues – Augmenting LLMs with Knowledge



- A central element of most agent architectures is the integration of an LLM with a long-term memory containing domain-specific knowledge in order to improve and semantically control the LLM's responses. Key problems:
  - Efficient and semantically rich knowledge representation
    - Embedding techniques prove to be largely non sufficient to account for semantic subtleties;
    - Graph based techniques e.g, knowledge graphs, inspired from ontologies.
  - Efficient retrieval mechanisms based on various similarity relations including perceptual, semantic, thematic, and functional.
  - Knowledge management techniques in particular for checking consistency of updates and modifications.

The RAG paradigm is essential for creating LLM-based agents, with a significant trend towards ontology-based knowledge graphs.

Agent Implementations Issues – Knowledge Classification

Knowledge

Agent-dependent (subjective) Knowledge: Cognitive and Deontic

Agent-independent (intersubjective) Knowledge: Scientific and Technical

Declarative
Implicit empirical
Mathematical and Scientific Theories
Properties, Declarative rules

States – State Predicates

Procedural
Empirical Methods
Design and System development flows
Coordination principles, Imperative rules

Actions – Action Predicates

Implicit human knowledge
Machine learning knowledge

Data-based

Model-based

Explainable by models e.g.
formal or simply conceptual

Objects, Physical or Artefacts
Agents, natural or artificial

Concrete

Concepts

Abstract

A posteriori e.g. Scientific and Technical
A priori e.g. Mathematical, Logical
Self-concepts e.g. Intention, Belief, Feeling

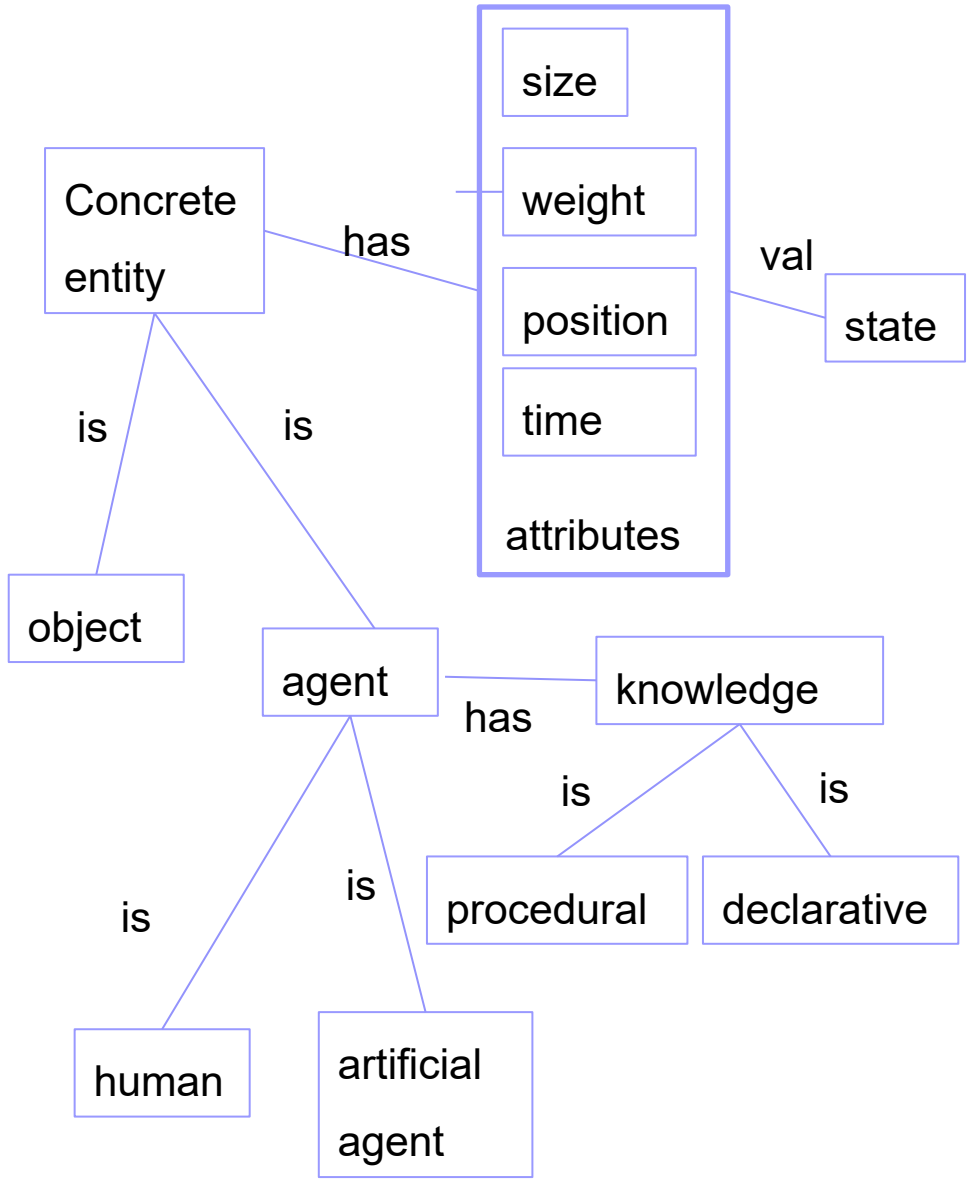
Types of knowledge, depending on their degree of validity, generality and domain and mode of use

Agent Implementations Issues – Ontology-based Knowledge Representation

- Ontologies that are hierarchically structured sets of entities in a subject area or domain that shows their properties and the relations between them.
- Structuring relations
  - Subtyping relations specified using the verb "is" : car "is" vehicle
  - Attribute relations specified using the verb "has" :
    object "has" weight, size, position etc.
    Attributes have domains – their valuations are states.
  - Predicates expressing relations between entities and their attributes :
    1) State predicates; 2) Action predicates.

- A state of the world is a set of entities with their attribute valuations e.g.:
  - {car1, car2. car3 | car.pos(i)=(xi,yi), car.speed(i)=vi, time=10}
  - {George, Mary, Chris| George.children ={Mary, Chris}}

- An action is a state change
  - {car1, car2. car3 | car.pos(i)=(xi,yi), car.speed(i)=vi, time=10} → {car1, car2. car3 | car.pos(i)=(xi',yi'), car.speed(i)=vi', time=12}
  - {George, Mary, Chris| George.children ={Mary, Chris}} → {George, Mary, Chris. Leo| George.children ={Mary, Chris, Leo}}



Agent Implementations Issues – Linking Natural Language to Ontologies

WORLD KNOWLEDGE

- Basic knowledge about the world characterizes the world states that satisfy atomic predicates P(x,y, ….).
- Temporal knowledge about the world characterizes state sequences *seq= state1 state2, .., state_n* using formulas with quantification over sequence and their states
  - *always P(x,y,..) = ∀seq∈SEQ ∀i P(seq(i)(x,y,…))*
  - *inevitable P(x,y,..) = ∀seq∈SEQ ∃i P(seq(i)(x,y,…))*
- Spatial knowledge about the world characterizes relations between positions of the entities in a space: "*a follows b*" means that *distance(b.pos(t)-a.pos(t)) ≤ d for all t*.
- Epistemic knowledge about the world uses the modality k_x to express the fact that agent *x* knows a property *P* e.g. $k_x(P)$
- Deontic knowledge about obligations, permissions to perform actions.

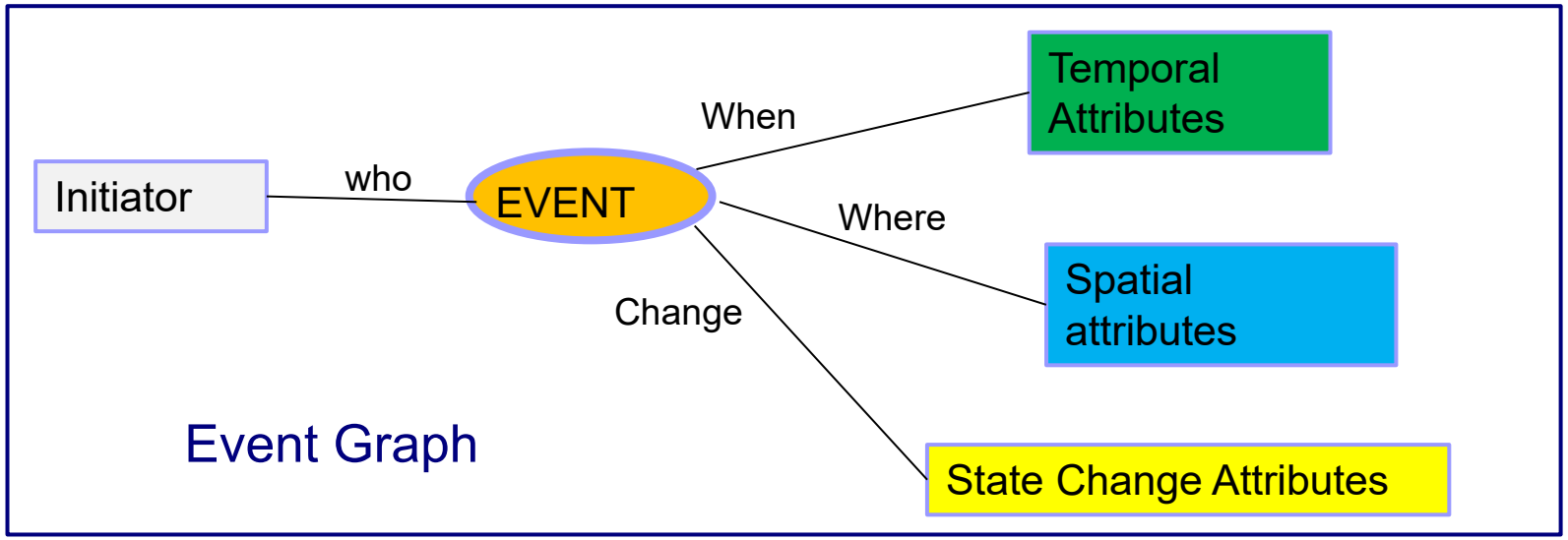
Verbalization

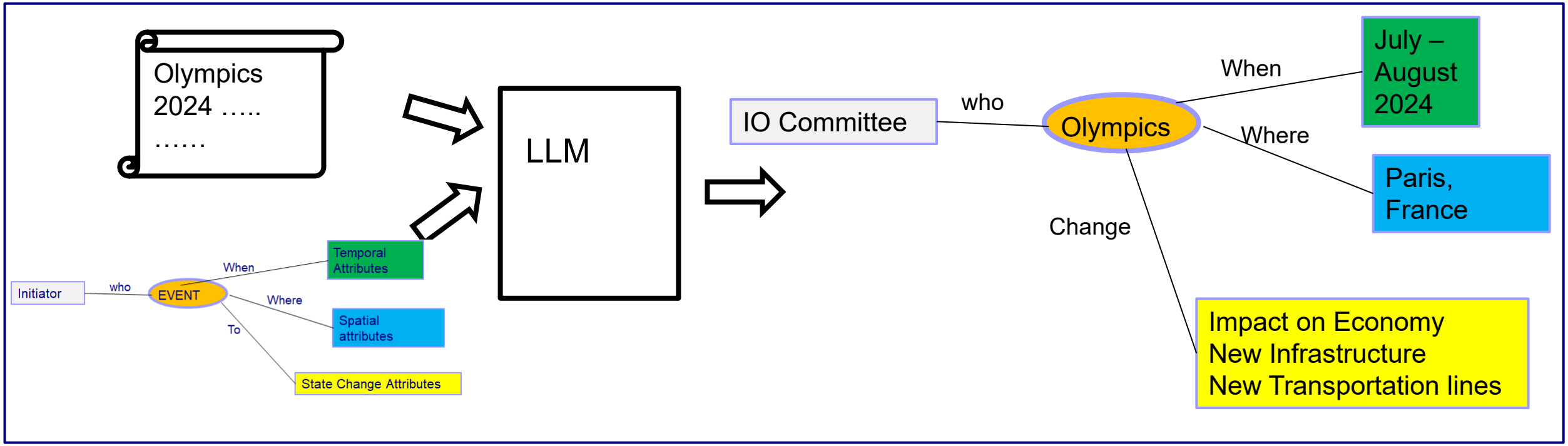
Formalization

NATURAL LANGUAGE

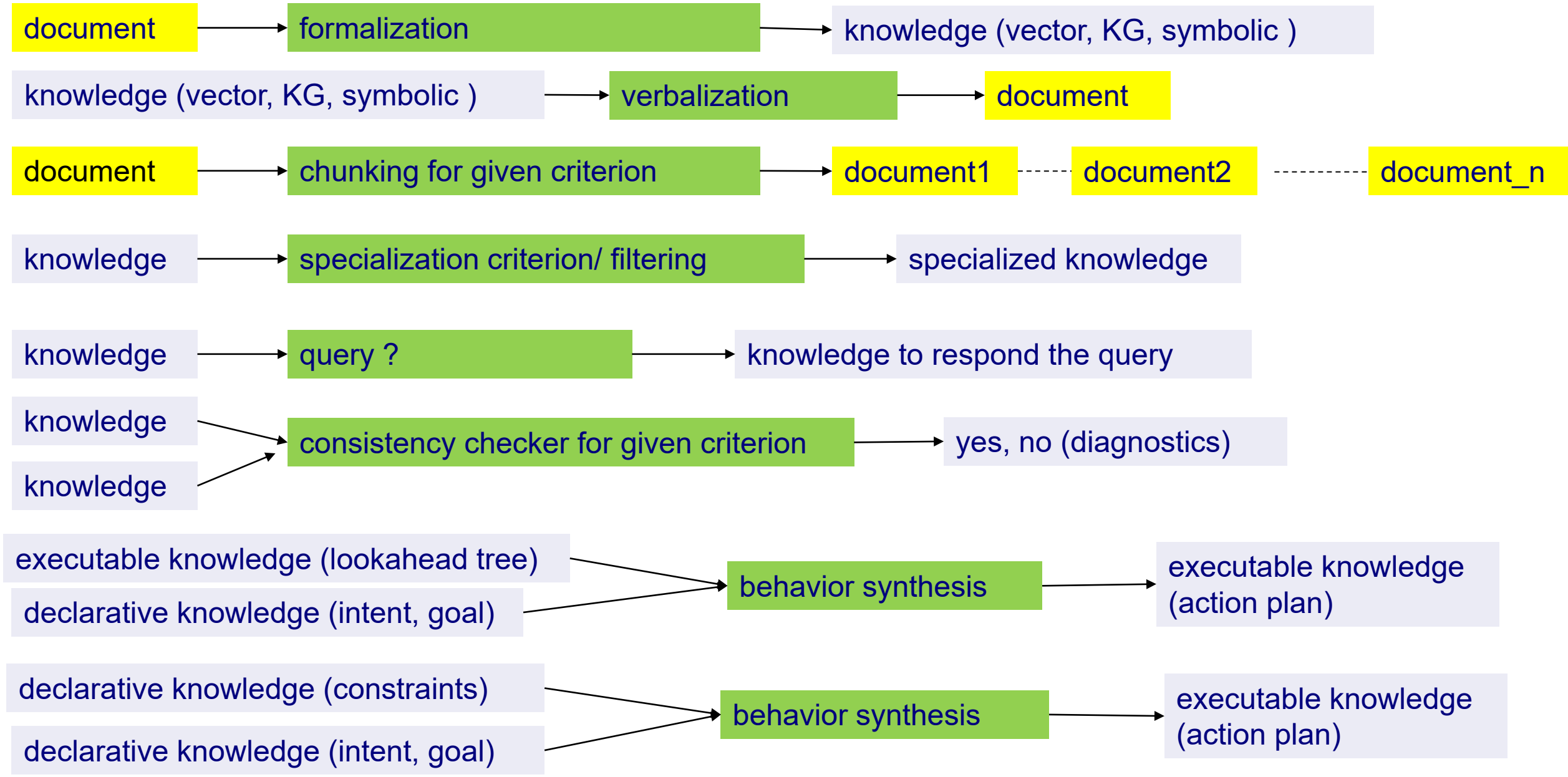
- Establish correspondence between Ontologies and Natural Language requires specific tokenization techniques :
  - Contexts characterizing states or sets of states of the world as relationships between concepts and their attributes involving "is" and "has".
  - Actions that correspond to change of contexts expressed by verbs denoting change, intention to do with two modalities: do(action) and say(text).
  - Temporal modalities, such as always, eventually, possibly, ever, maybe, may, might, after, before ….
  - Spatial modalities such as above, below, left, right, follows, precedes, between, containment relation.
  - Epistemic modalities, such as know, believe, think, …
  - Deontic modalities, such as must, have to, obliged to, mandatory that, imposed on,

Agent Implementations Issues – Extracting Knowledge from Document

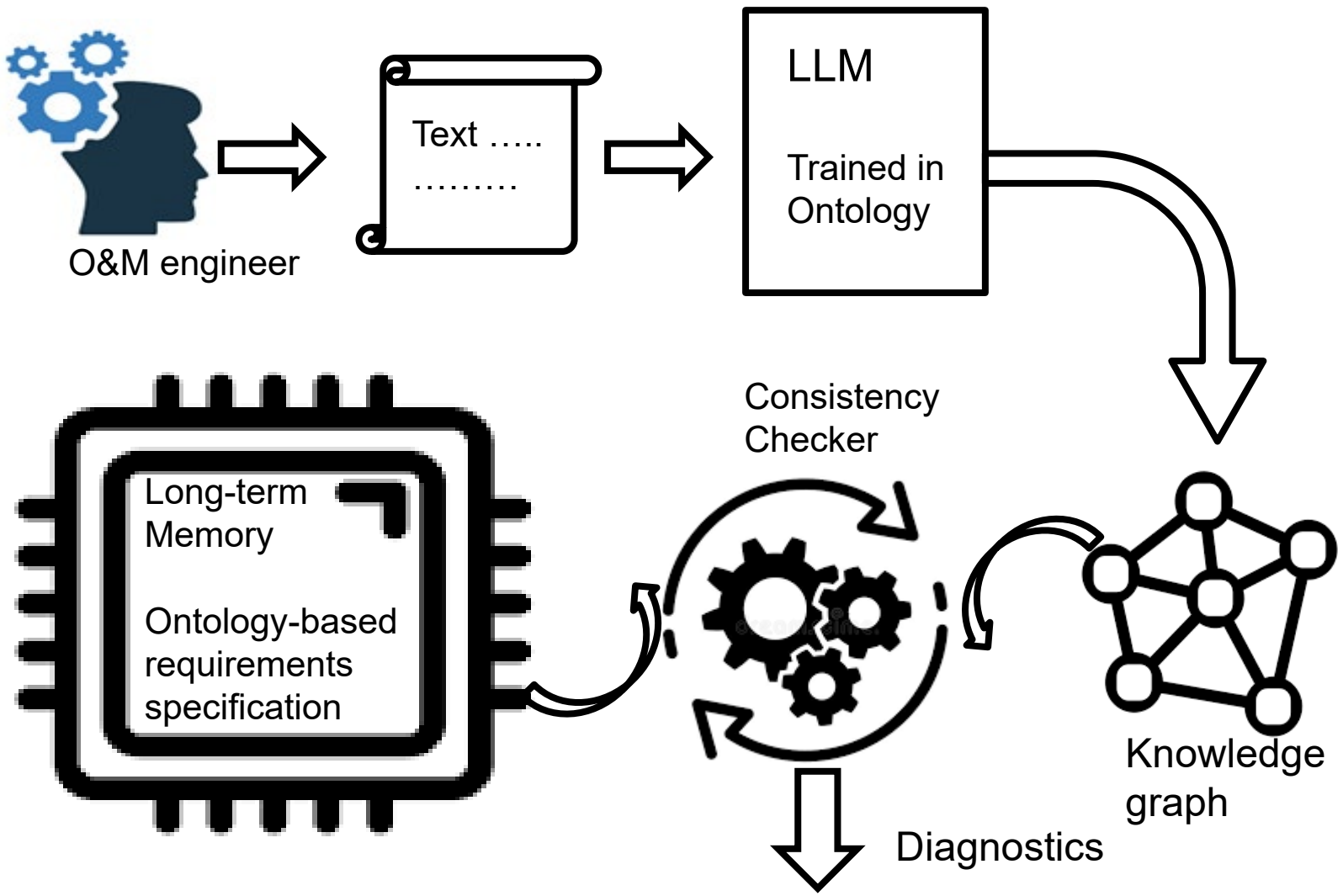




Agent Implementations Issues – Additional Problems To Be Solved



Agent Implementations Issues – Consistency Checking (Safeguarded AI)



Use case :

- An O&M engineer writes a scenario describing the steps for configuring an autonomous network.
  Before applying the scenario, they want to ensure that the generated configuration will not affect the essential requirements of the autonomous network, ranging from connectivity to dynamic load balancing, energy efficiency, and quality of service (QoS) guarantees.
- The long term memory contains an ontology-based requirements specification for ANs
- Use a Consistency Checker to compare the knowledge graphs with the ontology stored in a Memory and generate validation results, possibly diagnostics pointing out inconsistency.

Overview

- Important Clarifications
- Agent Reference Architecture
- Agent Implementations Issues
- Validation of AI Systems
- Where Are We Going?

Validation of Technical AI Systems – When a Self-driving Car is Safe Enough?

Waymo has now driven 10 billion autonomous miles in simulation

Darrell Etherington @etherington / 11:17 pm CEST • July 10, 2019

Comment

- The inability to build formal models for autonomous driving systems, limits their validation to simulation and testing.
  - Simple simulation is not enough - how a simulated mile is related to a "real mile" ?
  - We need evidence, based on coverage criteria, that the simulation deals fairly with the many different situations, e.g., different road types, traffic conditions, weather conditions, etc.

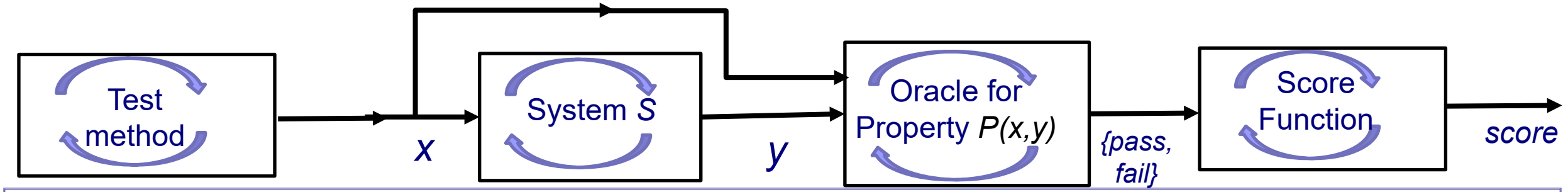
- We sorely lack testing methods for AI systems similar to those applied to software and hardware systems.
  - Sampling theory: methods for constructing samples that adequately cover real-world situations.
  - Repeatability: for two samples with the same degree of coverage, the estimated confidence levels are approximately the same.

Even in this case, it is impossible to obtain reliability guarantees of the order of 10^{-8} failures per hour of operation required for critical systems.

Validation of Technical AI Systems – Testing Basics

- Testing allows providing experimental evidence that a system $y=S(x)$ satisfies a property $P(x,y)$ using a framework:
  1. System *S*: the system under test e.g. a physical system, artifacts like autopilots and AI components;
  2. Property *P*:a predicate (hypothesis) characterizing the I/O behavior of *S*;
  3. Oracle: is an agent that can decide logically or empirically whether $P(x,y)$ holds producing verdicts *pass* or *fail.*

"S satisfies P" means that for any possible input x of S and corresponding y, the property P(x,y) is satisfied.



- Test method: How to choose among the possible test cases and decide whether the process is successful or not?

  1. Coverage Function: such that *coverage*(X)$\in$[0,1] measures the extent to which the set of test cases *X* explores the characteristics of the system's behavior in relation to the property *P*
  2. Score Function: such that *score*(X,Y) measures for a test set *(X,Y)* the likelihood that *S* meets *P* .

Reproducibility: If *(X1,Y1), (X2,Y2)* are two sets of tests then:

coverage(X1)=*coverage*(X2) implies *score*(X1,Y1) ~ *score*(X2,Y2)

Validation of Technical AI Systems – Applicability of Test Methods

System S	Property P (Hypothesis)	Test method	Oracle for P	Results **Evidence** that S satisfies P / **Reproducibility** of results
Solar System	Newton's Theory (Mathematical model for S)	Model-based coverage criteria	Measurements to check Newton's laws	Conclusive evidence/ Objectivity
Flight Controller	Safety properties (Mathematical model for S)	Model-based coverage criteria	Automated analysis of system runs	Conclusive evidence/ Objectivity
Population	Response to a medical treatment e.g. vaccine	Statistics-based clinical tests and setting	Expert analysis of clinical data	Statistical evidence/ Statistical reproducibility
Image classifier	Relation → ⊆ IMAGES×{cat,dog}	Test method for IMAGES?	Human oracle/justifiable unambiguous criteria.	Statistical evidence? / Statistical reproducibility?
Simulated Self-driving systems	Formally specified properties e.g. Traffic rules	Test method for driving scenarios?	Automated Analysis of system runs	Statistical evidence? / Statistical reproducibility ?
ChatGPT	Q/A relations in natural language	Test method for natural languages?	Human Oracle Subjective criteria	No objective evidence

- The application of rigorous test methods to AI systems
  - is limited to behavioral properties of technical systems for which numerous technical difficulties remain to be overcome due to the non-reproducibility of results, e.g., non-robustness, adversarial examples
  - excludes non-technical systems and LLMs in particular, for which other methods should be developed taking into account cognitive properties

Validation of Non-technical AI Systems – Alignment with Human Values

- What do artificial agents lack to approach the characteristics of human behavior?
  - Can an agent be made credible in such a way that it gives the impression of human behavior?
  - What are the distinctive human features that machines have difficulty reproducing?
- Many consider that outperforming humans in behavioral Q&A tests is proof of machine intelligence. But is it enough?
  - There is every reason to believe that an LLM will be able to pass the final medical exams as successfully as the students.
  - Does that mean the LLM should be allowed to practice as a medical doctor?

Certainly NO! We trust humans because we know that they know the rules of a value system and are bound by them.

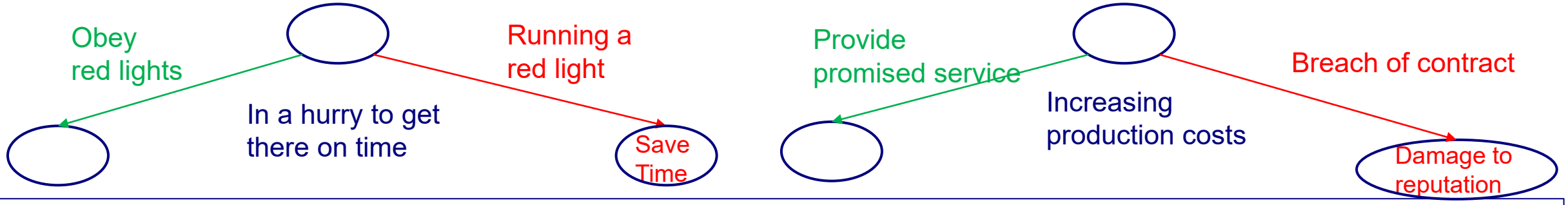
System operation aspect / Requirements	Behavioral Properties		Cognitive Properties
Risk-related properties	Safety	Security	Normative (ethical/law-enforced)
Usefulness properties	Functionality, Performance, Efficiency, User-friendliness		Intent and Goal-directed, Rationality properties

- The comparison between human agents and AI agents must consider
  - behavioral properties characterizing the interaction patterns observed between the agent and the world;
  - cognitive properties depend on the agent's knowledge, particularly its value system.
    - Normative properties characterize the degree of satisfaction of normative rules restricting agent choices;
    - Intent and Goal-directed properties characterize the way the agent chooses its goals and acts for their satisfaction.

Validation of Non-technical AI Systems – Acting Ethically and Rationally

- An agent has a value system, a set of rules and scales of values that enable it to estimate the costs and benefits resulting from the execution of actions for itself and its environment. It acts ethically if
  - It is aware of conflicting actions and can assess the impact of its actions;
  - It makes the choice the most in line with the rules and scales of the value system.

- Unlike behavioral properties, ethical properties cannot be decided without having access to the agent's world knowledge:
  - saying that "the earth is flat" can be a lie or ignorance;
  - non awareness that I am doing something wrong does not imply my responsibility.



- Rationality is a distinctive feature of human thinking that covers a variety of goal-directed properties, such as
  - Optimal decision-making and choice (quantitative reasoning);
  - Coherence, i.e. problems posed in logically equivalent situations admit similar solutions. (similarity of situations);
  - Competence levels: passing a test at a certain level implies passing a test at a lower level! (analogical thinking).

- Rationality simplifies the understanding, analysis and validation of an agent's properties.

  Experimental results show that AI agents are not rational, which makes their testing problem unsolvable!

Validation of Non-technical AI Systems – Formalizing Cognitive Properties

- Consider formulas built from the set of atomic predicates below, where x,y are agents, p is knowledge, and α is an action.
  - *do_x(α)*: agent x executes action α;
  - *say_x(p)*: agent x asserts that p is true;
  - *k_x(p):* agent x knows (believes) that p is true.
  - *vl_y(do_x(α)):* value generated for agent y, according to its value system, by action α executed by x,
  - *wr_y(do_x(α)):* agent y considers it incorrect (contrary to its normative rules) for x to perform action α;

- The following normative properties can be expressed using the above predicates:
  - Dishonest: *say_x(p) and k_x(not p)*
  - Irresponsible*: do_x(α) and k_x(wr_x(α))* *// x performs action α knowing that it violates normative rules;*
    Not responsible: do_x(α) and not(k_x(wr_x(α))) *// x performs α without knowing that it violates normative rules;*
  - Selfish: *do_x(α) and k_x(wr_x(α)) and vl_x(do_x(α))>>0) and vl_y(do_x(α))<<0)*
    //x knowingly performs wrong action α, beneficial to him and detrimental to y.
  - Generous: *do_x(α) and vl_x(do_x(α))<0) and vl_y(do_x(α))>>0)* *// x performs α, detrimental to him, but beneficial to y.*
  - Stupid: *do_x(α) and vl_x(do_x(α))<0) and v_y(do_x(α))<0)* *// x performs action α detrimental to him and to y*
  - *Trust: k_x(vl_y(do_y(α)) > vl_y((vl_y(not do_y(α))) // x trusts y to do action α because x believes it is beneficial to y.*

- The validation of cognitive properties presupposes the evaluation of atomic predicates on the agent's knowledge,

Overview

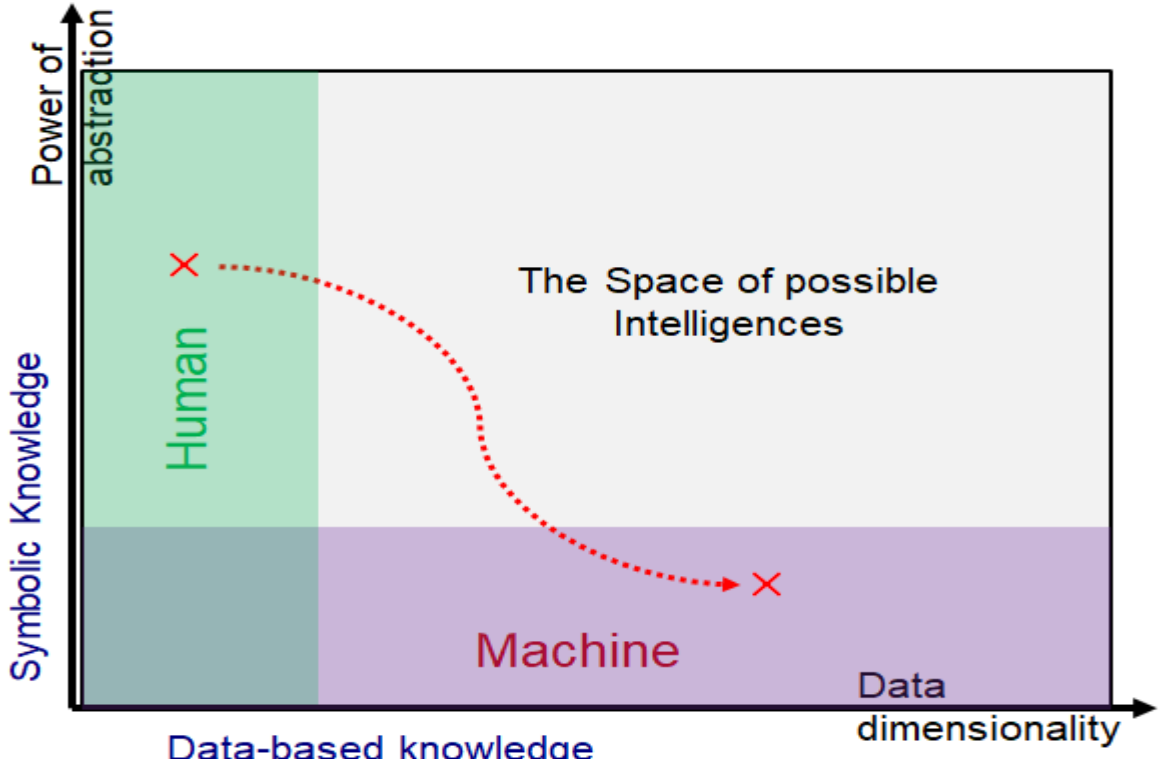
- Important Clarifications
- Agent Reference Architecture
- Agent Implementations Issues
- Validation of AI Systems
- Where Are We Going?

Where Are We Going? – The Space of Possible Intelligences

- Autonomous systems encompass a multi-faceted concept of intelligence.
  - There are multiple intelligences, each characterizing the ability to perform a task in a given context;
    To say that "S1 is smarter than S2" is meaningless without specifying the task and the criteria for success.
  - Human intelligence is not a theoretical concept, it is the result of historical evolution in a given physical environment.
    If human intelligence is the benchmark, AI should be able to perform/coordinate a set of tasks characterizing human skills.

- The space of possible intelligences: equivalent systems may use very different creative processes.
  - Humans are limited in analysis of multidimensional data, but are capable of common sense, abstraction and creativity.
  - AI systems outperform humans in learning multidimensional data, but fail to link symbolic to data-based knowledge.

- We need to explore the vast space of intelligences, particularly by delving into the various aspects of human symbolic intelligence and their relationship to data-driven intelligence.
  - Can we bridge the gap between symbolic and concrete knowledge exclusively by using neural networks?
  - Is it possible to trade symbolic reasoning capability for data-based learning as shown by LLM's opening the way to efficient solutions to symbolic reasoning problems e.g. MathPrompter



Where Are We Going? – Multi-agent Systems

- Current multi-agent systems, which follow on from the multi-agent systems promoted by IBM in the early 2000s and based on symbolic AI and logic, are more decentralized, AI-driven, and scalable. Two protocols are mainly used to create MA applications
  - Anthropic's Model Context Protocol (MCP) is an open standard designed to facilitate seamless integration between LLM applications and external data sources and tools.
  - Google's Agent-to-Agent (A2A) protocol provides an infrastructure for creating decentralized, autonomous AI agent ecosystems that communicate, negotiate, and collaborate without centralized control.
    - 1) semantic analysis of message content to find intent or goals;
    - 2) failure detection and self-healing;
    - 3) agent self-optimization and coordination to achieve global system goals;
    - 4) online monitoring and validation techniques to ensure cognitive properties such as trust, rationality, accountability.

- MA Systems' vision is based on the premise that AI will be a game changer by offering greater flexibility in specifications and greater efficiency in problem solving, but it often overlooks the fact that the price to pay is a serious lack of rigor and semantic control. Unfortunately, the major trends do not bode well
  - Entrusting LLMs with the role of orchestrator poses problems of reliability and lack of responsiveness/adaptability: only traditional software orchestrators could work.
  - It is naive to think that a system becomes smarter by adding features: intelligence is not just about the ability to do more things. **Doing more is different from doing better.**

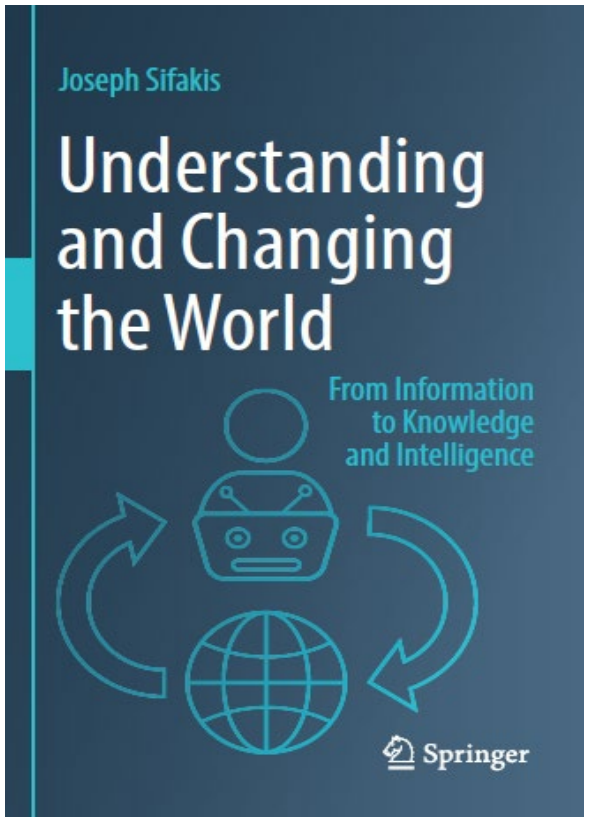
MA Systems' vision requires linking ML to symbolic AI, distributed algorithms, and network technology.

Where Are We Going? – AI Meets Systems Engineering

- The development of autonomous systems requires a marriage between ICT and AI, which poses non-trivial technical problems, as new trends are disrupting traditional systems engineering.
  - How can reliable systems be built from unreliable components using hybrid architectures that integrate ICT components and unexplainable AI components, while getting the best out of each?
  - How to link symbolic and non-symbolic knowledge e.g. sensory information and models used for decision-making.
  - How to move from correctness at design time to correctness at runtime to achieve adaptation?

- System validation is marked by an irreversible shift from rationalism to empiricism due AI's lack of explainability.
  - We must strive to compensate for the lack of solid guarantees of trustworthiness by using explicit knowledge about the world stored in long-term memory.
  - We need technical standards that provide methods for risk assessment and reliability certification.
    - Standards do not hinder innovation; on the contrary, they challenge us to find innovative solutions.
    - Absence of regulation leads to poorly engineered systems, increase technical debt that compromises the future.

- We need to elaborate a broad technical vision covering a wide range of system types and domain-specific technologies.
  - Human intelligence has many facets and can only be achieved by combining different types of AI and ICT, including prestored World Knowledge, symbolic, traditional ML and LLM, e.g. a chess playing robot, cannot drive a car.
  - The setbacks experienced by the autonomous car industry show that there is still a long way to go to bridge the gap between automation and autonomy.


Joseph Sifakis
Understanding and Changing the World
From Information to Knowledge and Intelligence
Springer

Merci