Artificial Cognitive Systems

Module 3: Cognitive Architectures

Lecture 4: Example cognitive architectures: CRAM

David Vernon
Carnegie Mellon University Africa

www.vernon.eu

CRAM

- CRAM: Cognitive Robot Abstract Machine
- Hybrid cognitive architecture (symbolic and sub-symbolic representations and processes)
- Introduced by Michael Beetz in 2010
  - developed significantly since then based on several research projects
- Designed to address robot manipulation tasks in everyday activities
  - tasks that would typically be carried out by people in household settings, e.g., in a kitchen.

The Robot Household Marathon aka the EASE Robot Day Demonstrator

Gayane Kazhoyan, Simon Stelter, Ferenc Balint-Benczedi,
Franklin Kenghagho Kenfack, Sebastian Koralewski and Michael Beetz

Everyday Activity Science and Engineering
www.ease-crc.org

Artificial Intelligence
Universität Bremen

https://www.youtube.com/user/agkiunibremen

The Robot Household Marathon aka the EASE Robot Day Demonstrator

Gayane Kazhoyan, Simon Stelter, Ferenc Balint-Benczedi,
Franklin Kenghagho Kenfack, Sebastian Koralewski and Michael Beetz

Artificial Intelligence

Universität Bremen

Design Principles

Implicit-to-explicit manipulation: “fetch the spoon and put it on the table”

Vaguely-stated
high-level goal

Specific low-level motions
required to accomplish the goal

Design Principles

Implicit-to-explicit manipulation: “fetch the spoon and put it on the table”

Vaguely-stated high-level goal → Specific low-level motions required to accomplish the goal

Mapping is accomplished using a generative model:
Tightly-coupled symbolic and sub-symbolic knowledge representations
of the robot, tasks it is performing, objects it is acting on and environment in which it is operating

Design Principles

Implicit-to-explicit manipulation: “fetch the spoon and put it on the table”

Vaguely-stated high-level goal → Specific low-level motions required to accomplish the goal

Requested by another agent or self-generated

Constraints at the general high-level are propagated to the low-level execution in the mapping

Success in accomplishing the goal is evaluated in the same space as that in which the goal was formulated, i.e. the perceptual space

Design Principles

CRAM focusses on abstract specification of robot actions that are underdetermined

- The action specifications are framed in terms without all knowledge required to complete the action

  e.g "fetch the milk and pour it in the bowl"

- The knowledge required to complete the action is resolved at run-time during plan execution
- by querying in real-time a multi-element knowledge-base
  - Prior knowledge
  - Current world states
  - Robot's sensorimotor state

Design Principles

The control program is stated as a generalized action plan

- One plan for each category of underdetermined action description, e.g. fetch, place, pour, cut, …
- The plan can be executed
- The plan can be reasoned about and transformed
  - Self—programming
  - Development and self-improvement through automatic generation of new plans

Design Principles

The control program is stated as a generalized action plan

Generalized Action Plan

Motion Plan

This is the key concept in CRAM

3-step refinement process:
contextualization
Identify the values of the parameters to the motion plan that maximize the likelihood that the associated body motions successfully accomplish the desired action

The motion parameter values are provided by the generative model

Design Principles

The control program is stated as a generalized action plan

Generalized Action Plan

Motion Plan

This is the key concept in CRAM

The motion parameter values are provided by the generative model

Equivalent

- Perform an underdetermined action description
- Execute (i.e. interpret) a generalized action plan
- Resolve a high-level action designator in a generalized action plan (explained in next few slides)

Control strategies in object manipulation tasks

J Randall Flanagan[1], Miles C Bowman[1] and Roland S Johansson[2]

The remarkable manipulative skill of the human hand is not the result of rapid sensorimotor processes, nor of fast or powerful effector mechanisms. Rather, the secret lies in the way manual tasks are organized and controlled by the nervous system. At the heart of this organization is prediction. Successful manipulation requires the ability both to predict the motor commands required to grasp, lift, and move objects and to predict the sensory events that arise as a consequence of these commands.

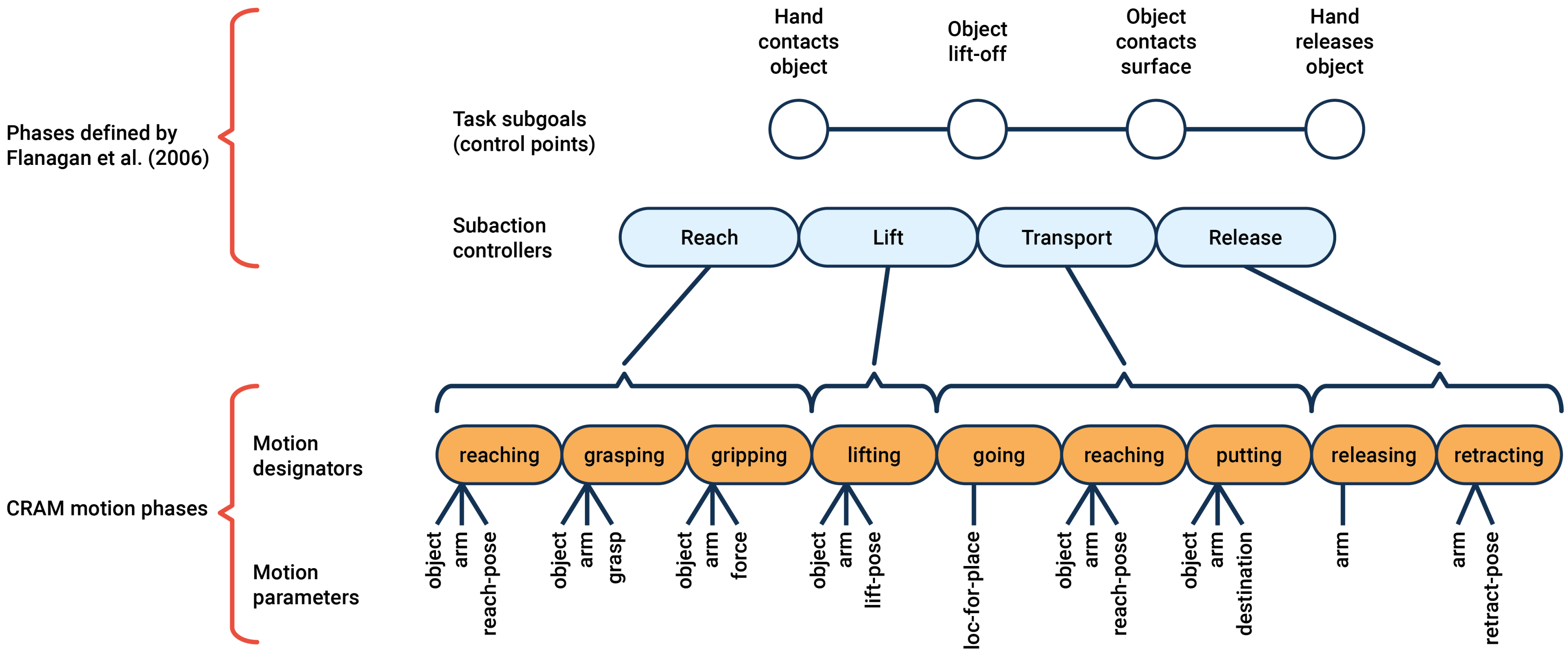
Addresses
[1] Department of Psychology and Centre for Neuroscience Studies, Queen's University, Kingston, ON, K7L 3N6, Canada
[2] Section for Physiology, Department of Integrative Medical Biology, Umeå University, SE-901 87 Umeå, Sweden

Corresponding author: Flanagan, J Randall

and another object or surface. Importantly, these contact events give rise to discrete and distinct sensory events, each characterized by a specific afferent neural signature. Because these sensory events provide information related to the functional goals of successive action phases, they have a crucial role in the sensory control of manipulations. In object manipulation, the brain not only forms action plans in terms of series of desired subgoals but also predicts the sensory events that signify subgoal attainment in conjunction with the generation of the motor commands. By comparing predicted sensory events with the actual sensory events, the motor system can monitor task progression and adjust subsequent motor commands if errors are detected. As discussed further below, such adjustments involve parametric adaptation of fingertip actions to the mechanical properties of objects, triggering

J Randall Flanagan, Miles C Bowman, and Roland S Johansson. Control strategies in object manipulation tasks. *Current opinion in neurobiology*, 16(6):650–659, 2006.

Motion Plan

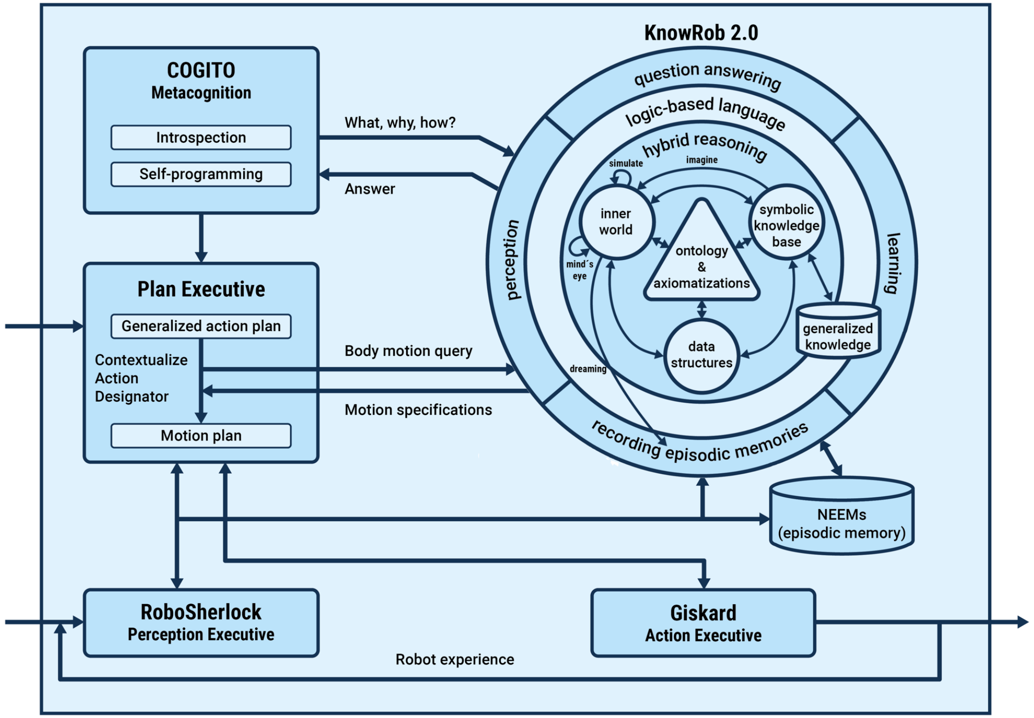


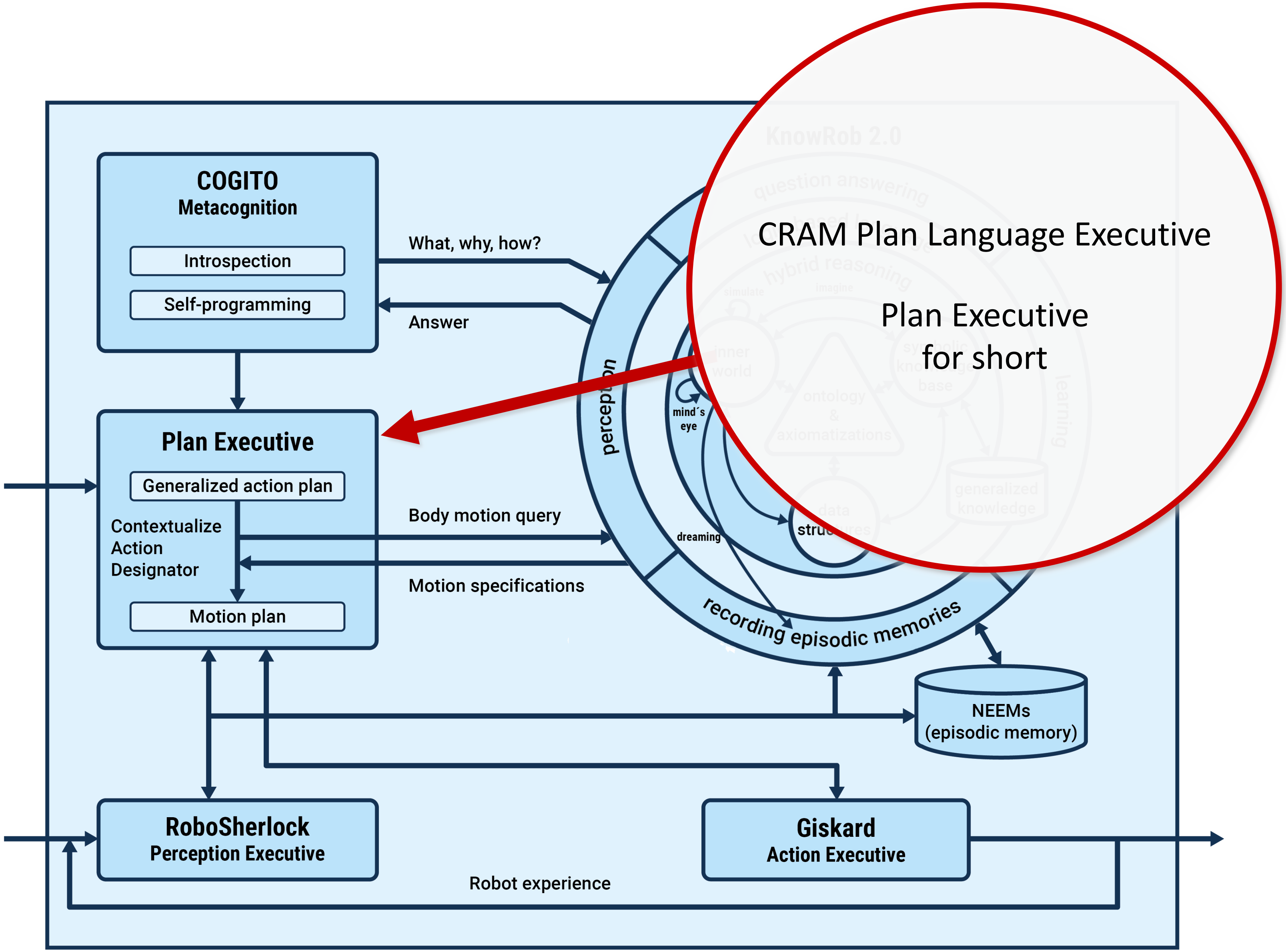
J Randall Flanagan, Miles C Bowman, and Roland S Johansson. Control strategies in object manipulation tasks. *Current opinion in neurobiology*, 16(6):650–659, 2006.

The CRAM Cognitive Architecture

CRAM has five core elements:

1. CRAM Plan Language (CPL) executive
2. KnowRob2.0 knowledge-bases and associated reasoning processes
3. RoboSherlock, the perception executive
4. Giskard, the action executive
5. COGITO, a metacognition system




COGITO
Metacognition
Introspection
Self-programming
What, why, how?
Answer
Plan Executive
Generalized action plan
Contextualize
Action
Designator
Motion plan
Body motion query
Motion specifications
perception
mind´s
eye
dreaming
recording episodic memories
NEEMs
(episodic memory)
RoboSherlock
Perception Executive
Giskard
Action Executive
Robot experience
CRAM Plan Language Executive
Plan Executive
for short

CRAM Plan Language (CPL) Executive

- Tasks are accomplished by executing plans written in the CRAM Plan Language (CPL)
- CPL is an extension of Lisp
- A CPL plan represents all key aspects of the plan as persistent first-class objects in a first-order logic
  - Plans themselves can be reasoned about, even at runtime
  - Particularly relevant for the meta-cognition system, COGITO

CRAM Plan Language (CPL) Executive

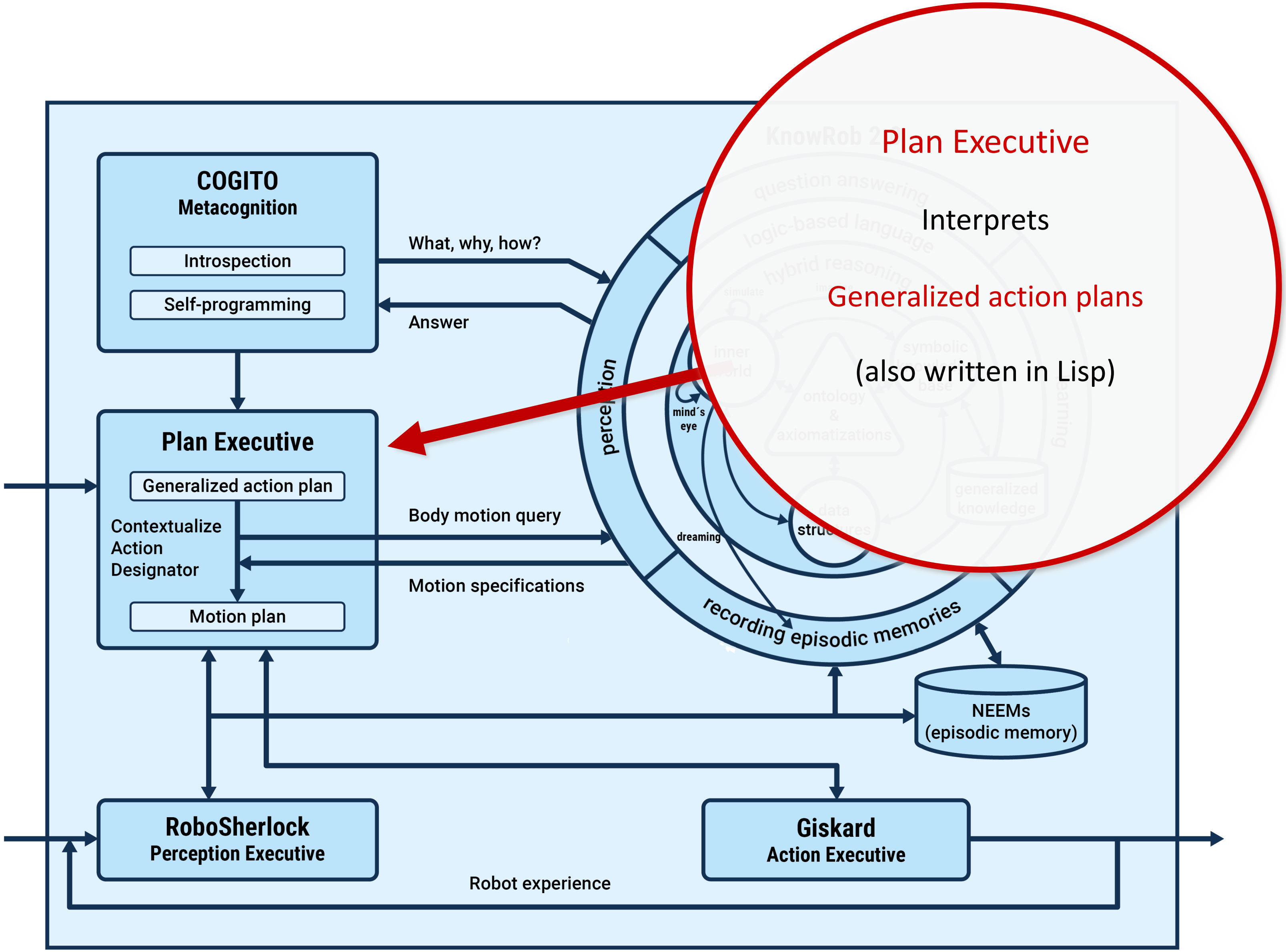
- Plans specify how the robot should respond to
  - Sensory events
  - Changes in belief states
  - Failures in plans
- All of which can be queried, inspected, and reasoned about

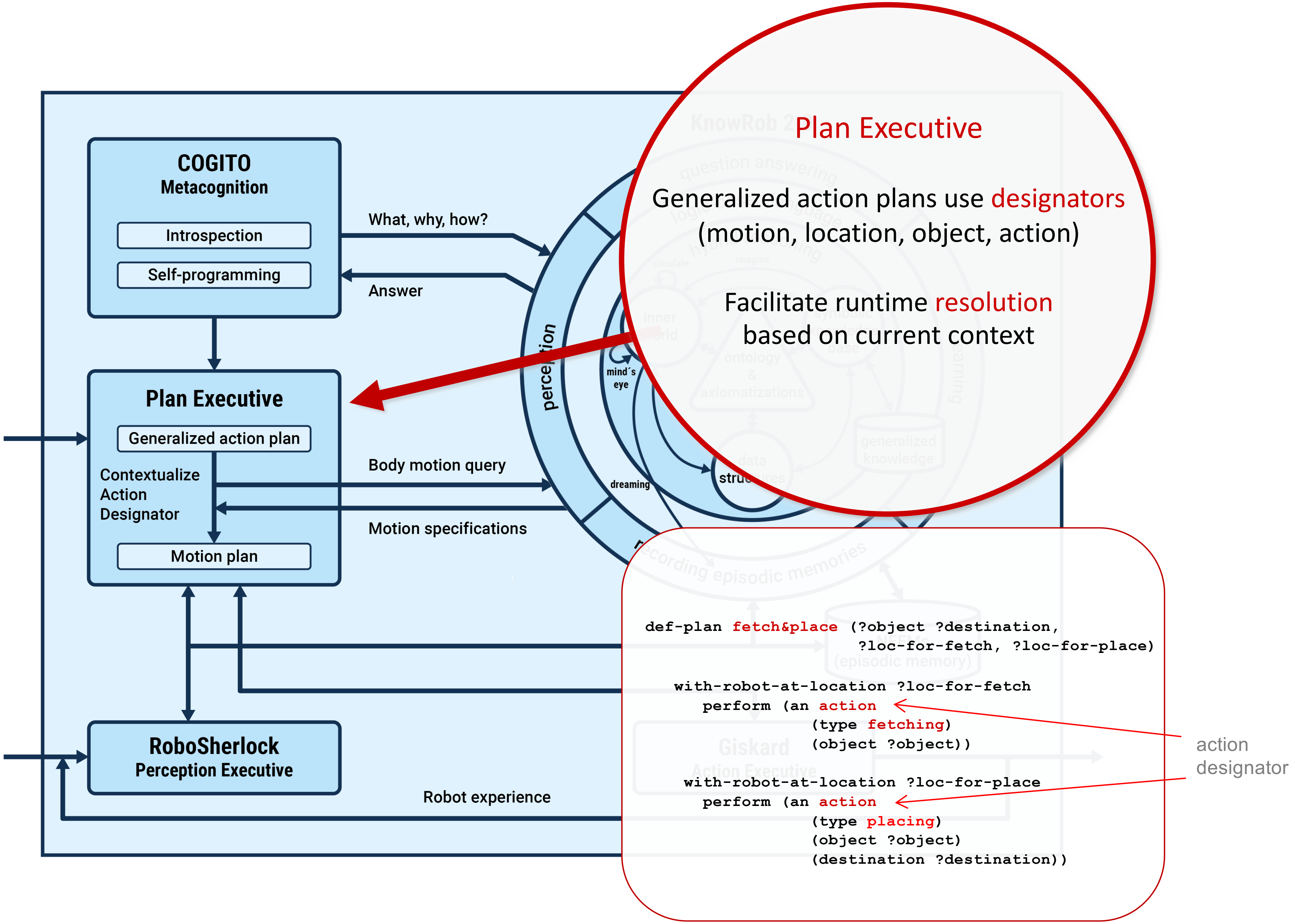
CRAM Plan Language (CPL) Executive

- A plan comprises set of abstract plan designators for
  - actions
  - objects
  - locations
  - motions (i.e. elementary movements)

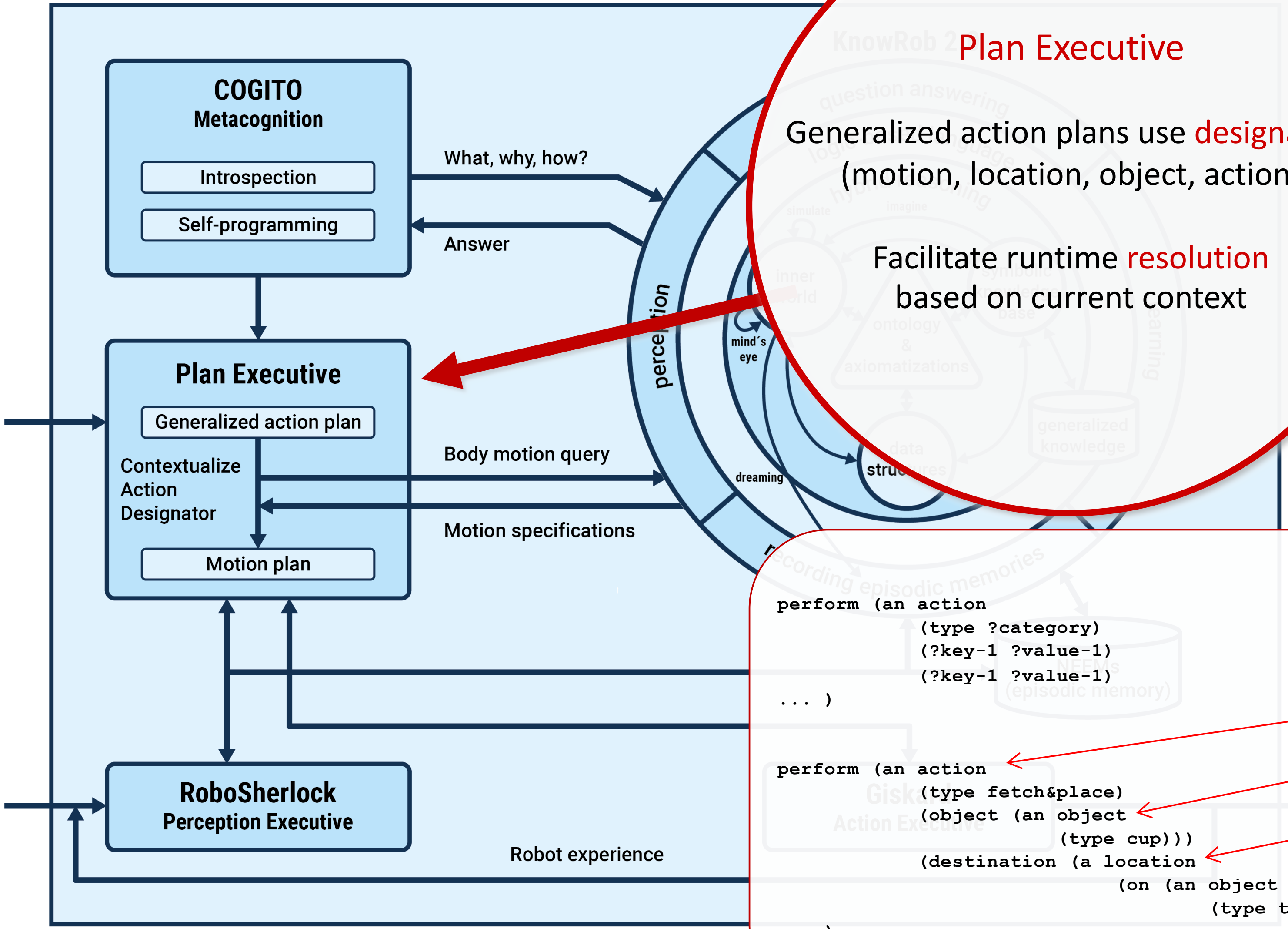
Designators are effectively placeholders

require runtime resolution based on the current context of the task action


COGITO
Metacognition
Introspection
Self-programming
What, why, how?
Answer
Plan Executive
Generalized action plan
Contextualize
Action
Designator
Motion plan
Body motion query
Motion specifications
perception
mind´s eye
dreaming
recording episodic memories
NEEMs
(episodic memory)
RoboSherlock
Perception Executive
Giskard
Action Executive
Robot experience
Plan Executive
Interprets
Generalized action plans
(also written in Lisp)




```
def-plan fetch&place (?object ?destination,
                      ?loc-for-fetch, ?loc-for-place)

  with-robot-at-location ?loc-for-fetch
    perform (an action
                (type fetching)
                (object ?object))

   with-robot-at-location ?loc-for-place
    perform (an action
                (type placing)
                (object ?object)
                (destination ?destination))
```



Plan Executive

Generalized action plans use designators (motion, location, object, action)

Facilitate runtime resolution based on current context

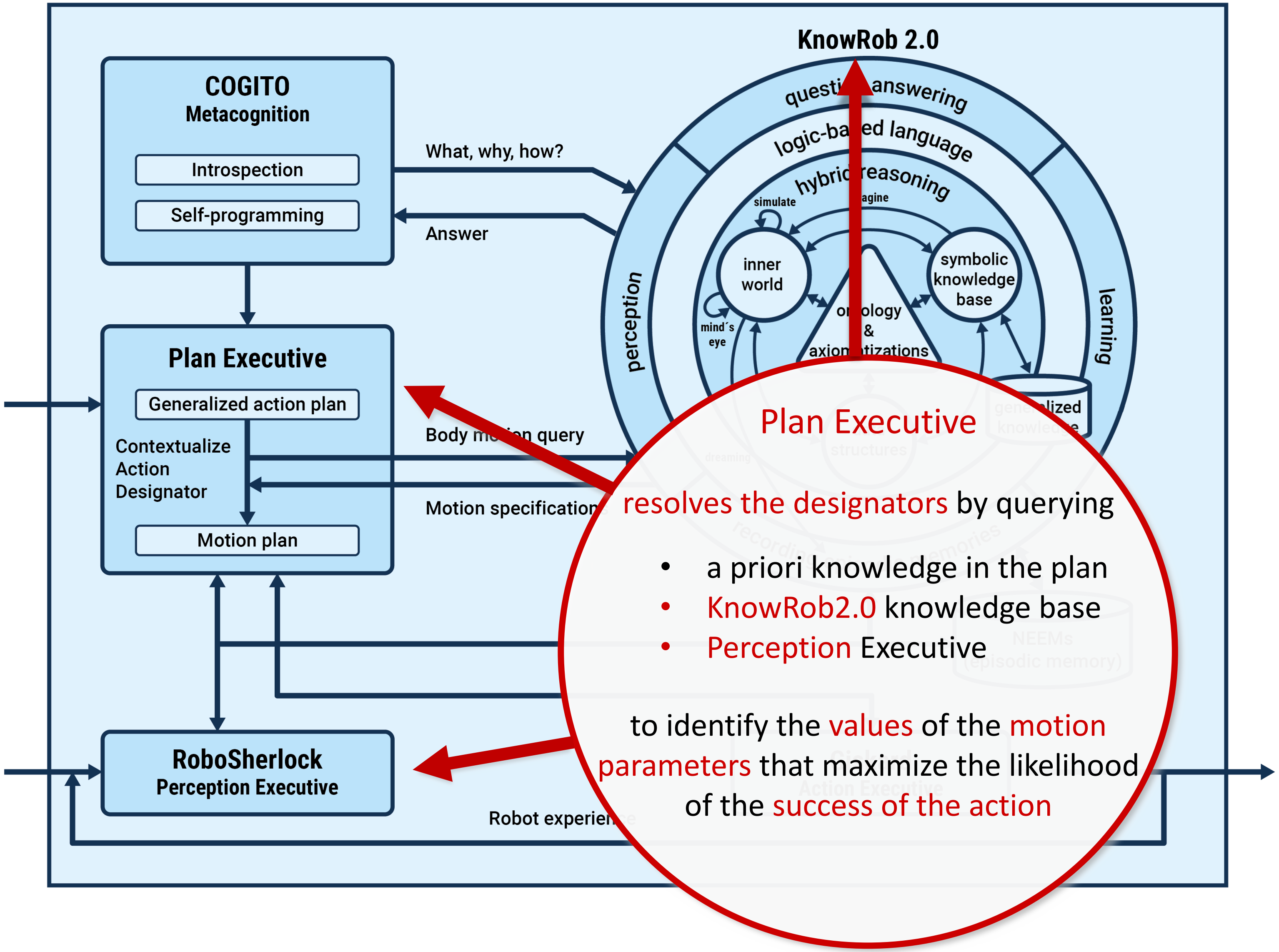
```
perform (an action
            (type ?category)
            (?key-1 ?value-1)
            (?key-1 ?value-1)
... )

perform (an action
            (type fetch&place)
            (object (an object
                        (type cup)))
            (destination (a location
                             (on (an object
                                     (type table)))))
... )
```

action designator

object designator

location designator

object designator



Plan Executive

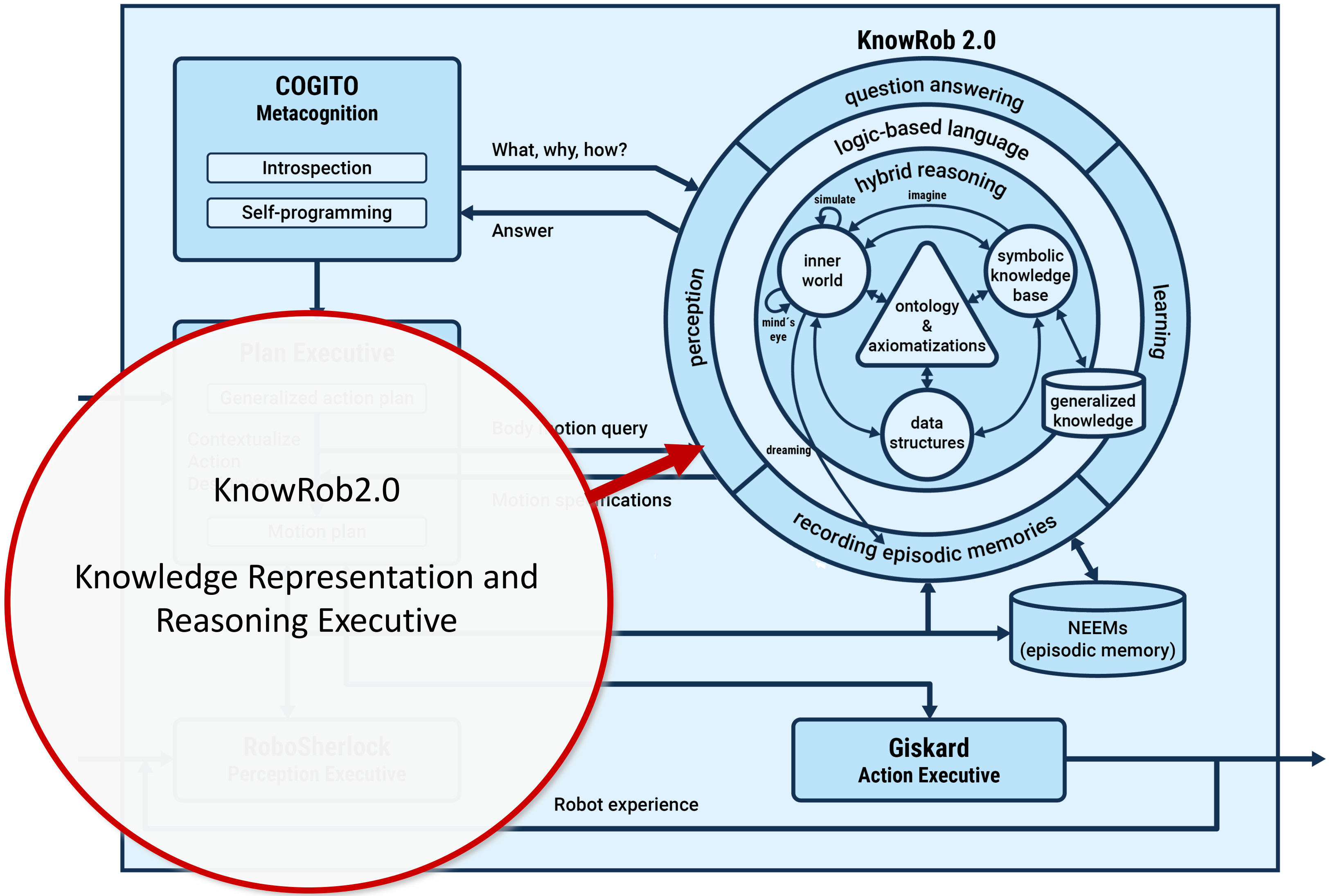
resolves the designators by querying

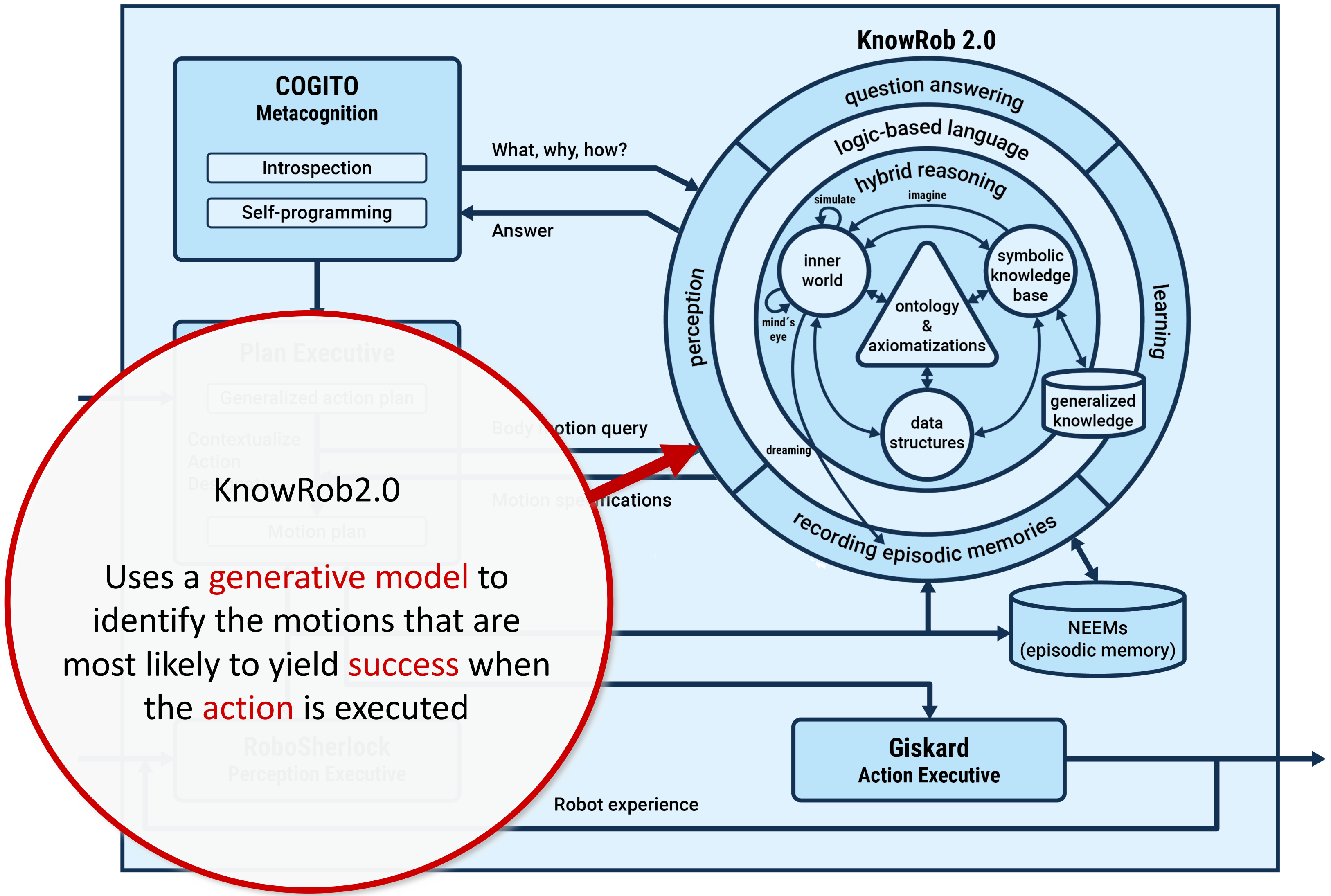
- a priori knowledge in the plan
- KnowRob2.0 knowledge base
- Perception Executive

to identify the values of the motion parameters that maximize the likelihood of the success of the action

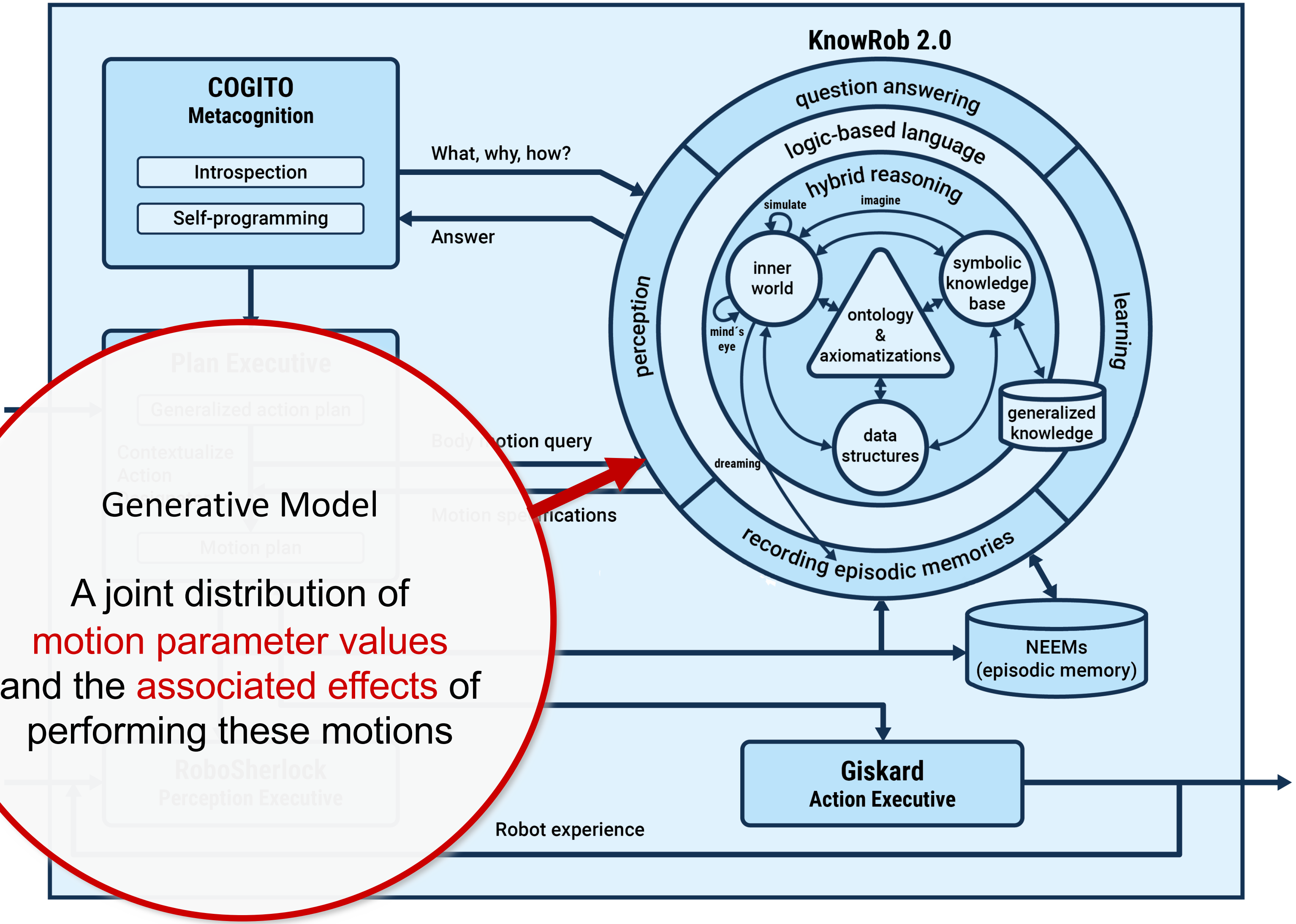
Programming Robotic Agents with Action Descriptions

Gayane Kazhoyan and Michael Beetz


KnowRob 2.0
COGITO
Metacognition
Introspection
Self-programming
What, why, how?
Answer
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
ontology & axiomatizations
mind´s eye
data structures
generalized knowledge
dreaming
perception
learning
recording episodic memories
otion query
ications
NEEMs (episodic memory)
Giskard
Action Executive
Robot experience
KnowRob2.0
Knowledge Representation and Reasoning Executive

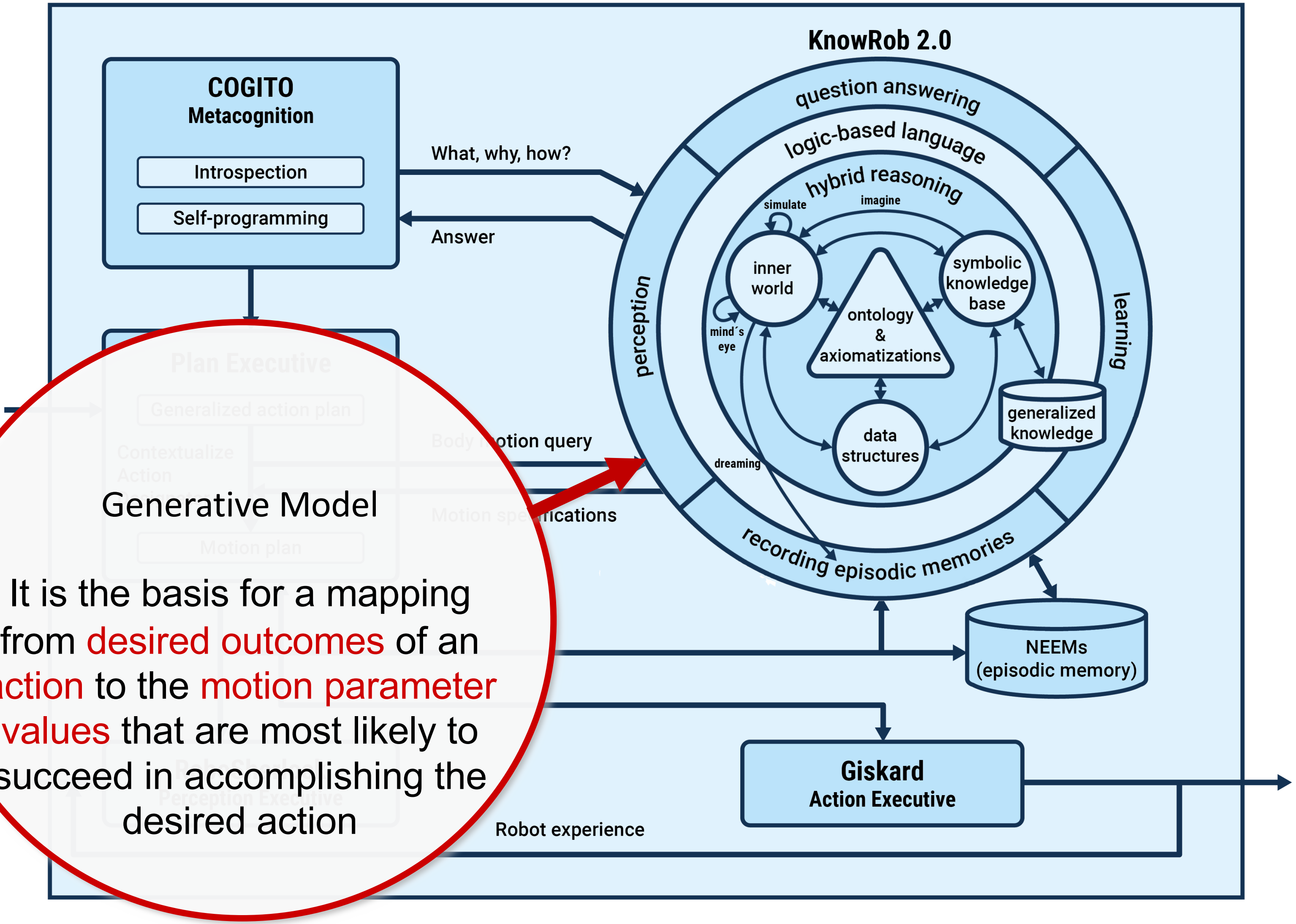


KnowRob 2.0
COGITO
Metacognition
Introspection
Self-programming
What, why, how?
Answer
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
ontology & axiomatizations
mind´s eye
data structures
generalized knowledge
dreaming
perception
learning
recording episodic memories
NEEMs (episodic memory)
Giskard
Action Executive
Robot experience
otion query
ications
KnowRob2.0
Uses a generative model to identify the motions that are most likely to yield success when the action is executed



Generative Model

A joint distribution of
motion parameter values
and the associated effects of
performing these motions



Generative Model

It is the basis for a mapping from desired outcomes of an action to the motion parameter values that are most likely to succeed in accomplishing the desired action

KnowRob2 Knowledge Base

- Knowledge representation and reasoning framework for robotic agents
  - Implemented in Prolog

- Exposed as a conventional first-order time interval logic knowledge-base

- However, many logic expressions are constructed on-demand from sensorimotor data computed in real-time

KnowRob2 Knowledge Base

- Provides the background commonsense intuitive-physics knowledge required by the CPL executive to implement its goal-directed under-determined task plans, e.g.
  - How to grasp an object (depending on the object's shape, weight, softness, and other properties)
  - How it has to be held while moving it (e.g. upright to avoid spilling its contents)
  - Where the object is normally located.

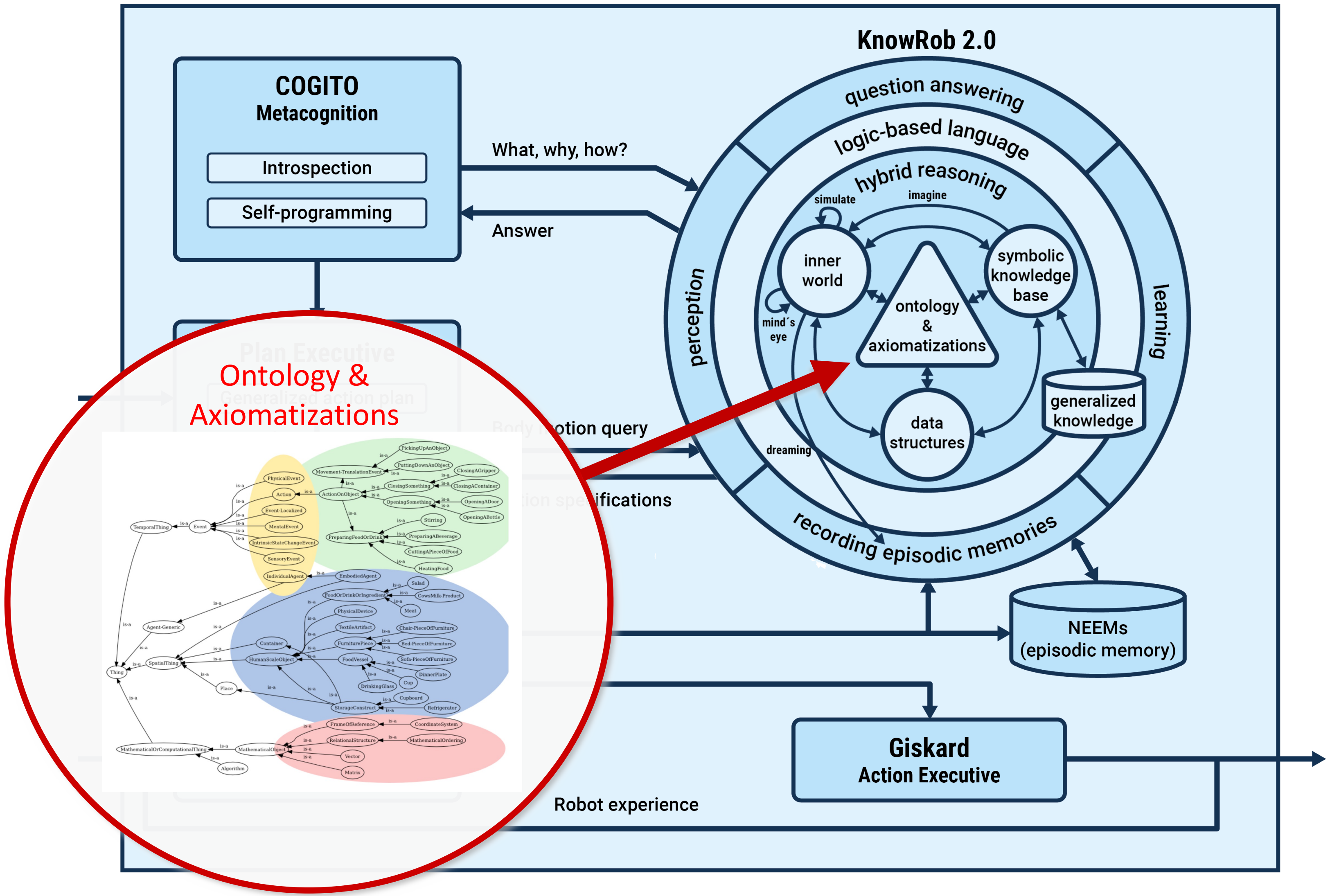
KnowRob2 Knowledge Base

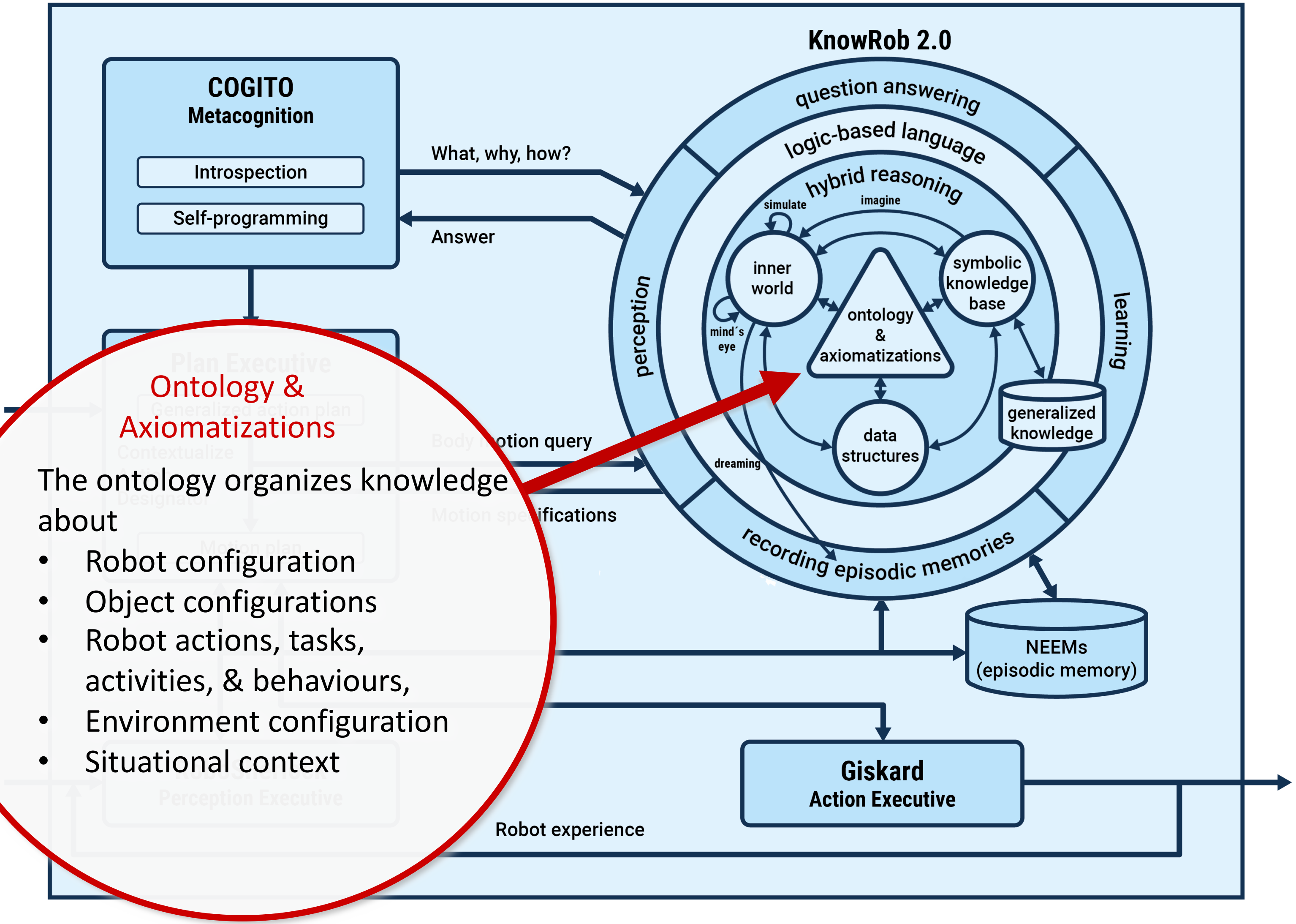
- Source of knowledge:
  - Some is specified a priori
  - Some is derived from experience
  - Some is the result of simulated execution of candidate actions using a high-fidelity virtual reality physics engine simulator
- All represented by a first-order time interval logic expression, and reasoned about as needed.

KnowRob2 Knowledge Base

KnowRob2 comprises five core elements

- embedded in a hybrid multi-formalism reasoning shell
- exposed through a logic-based language layer

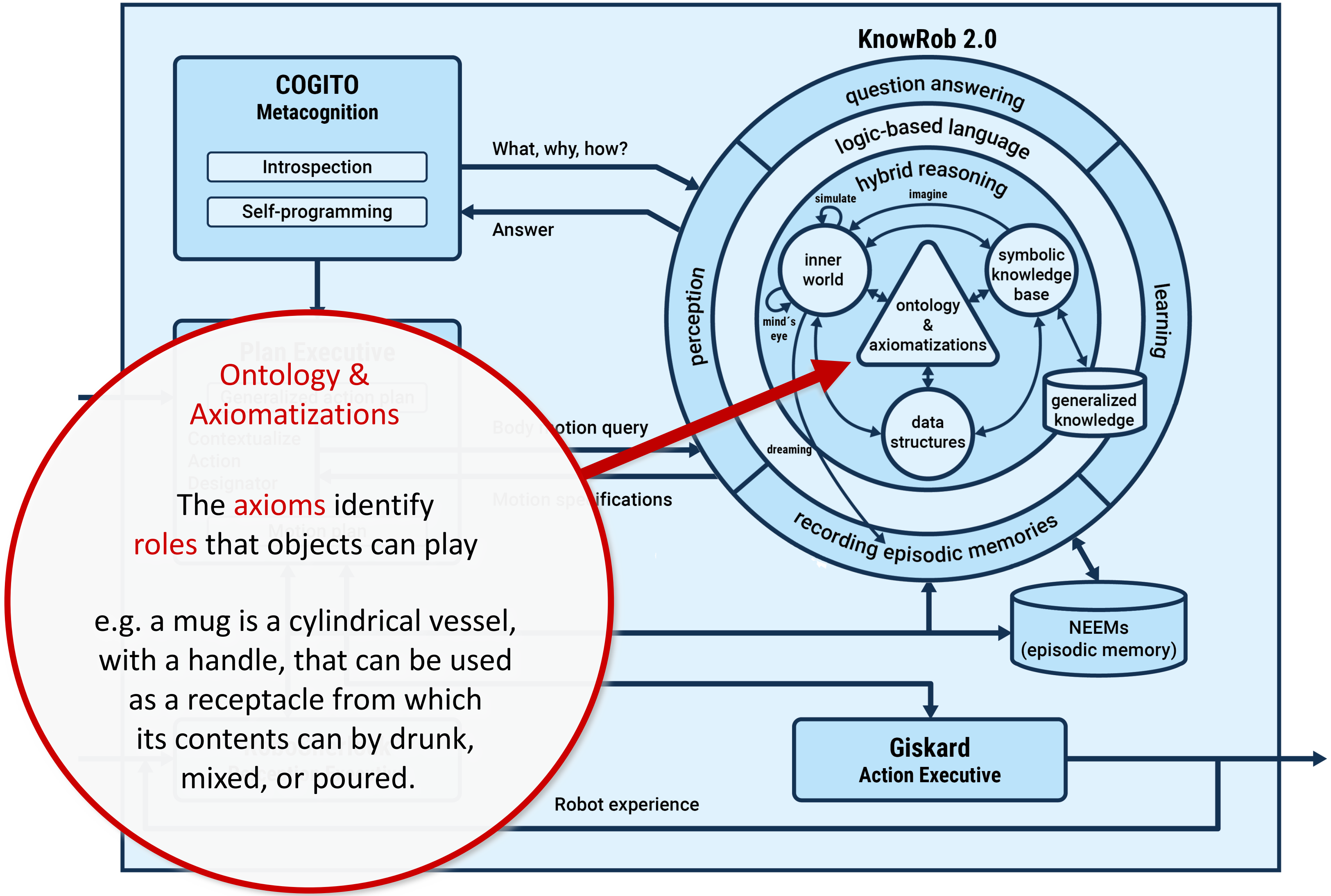

KnowRob 2.0
COGITO
Metacognition
Introspection
Self-programming
What, why, how?
Answer
Robot experience
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
ontology & axiomatizations
mind´s eye
data structures
generalized knowledge
dreaming
perception
learning
recording episodic memories
NEEMs (episodic memory)
Giskard
Action Executive
Ontology &
Axiomatizations

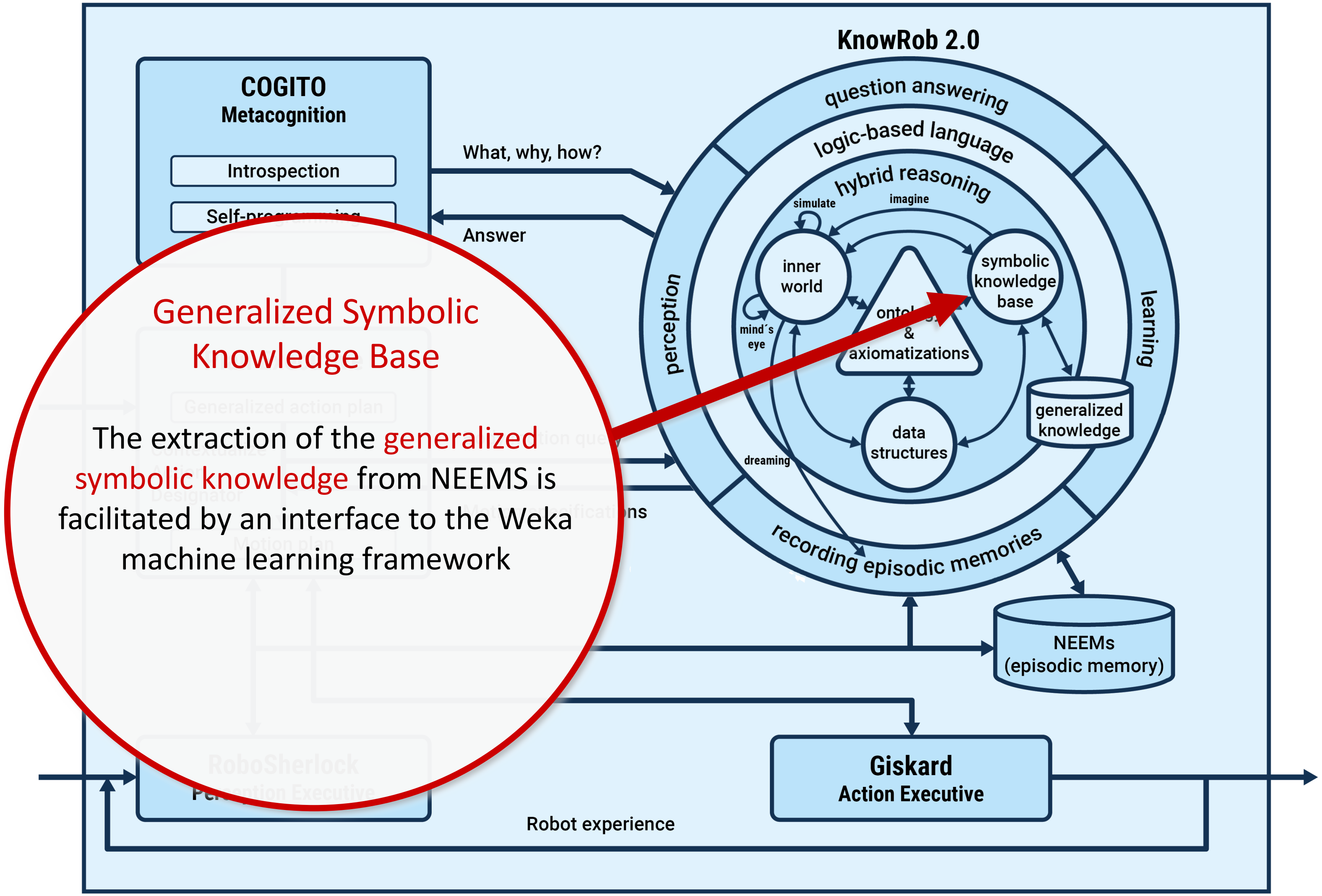


Ontology & Axiomatizations

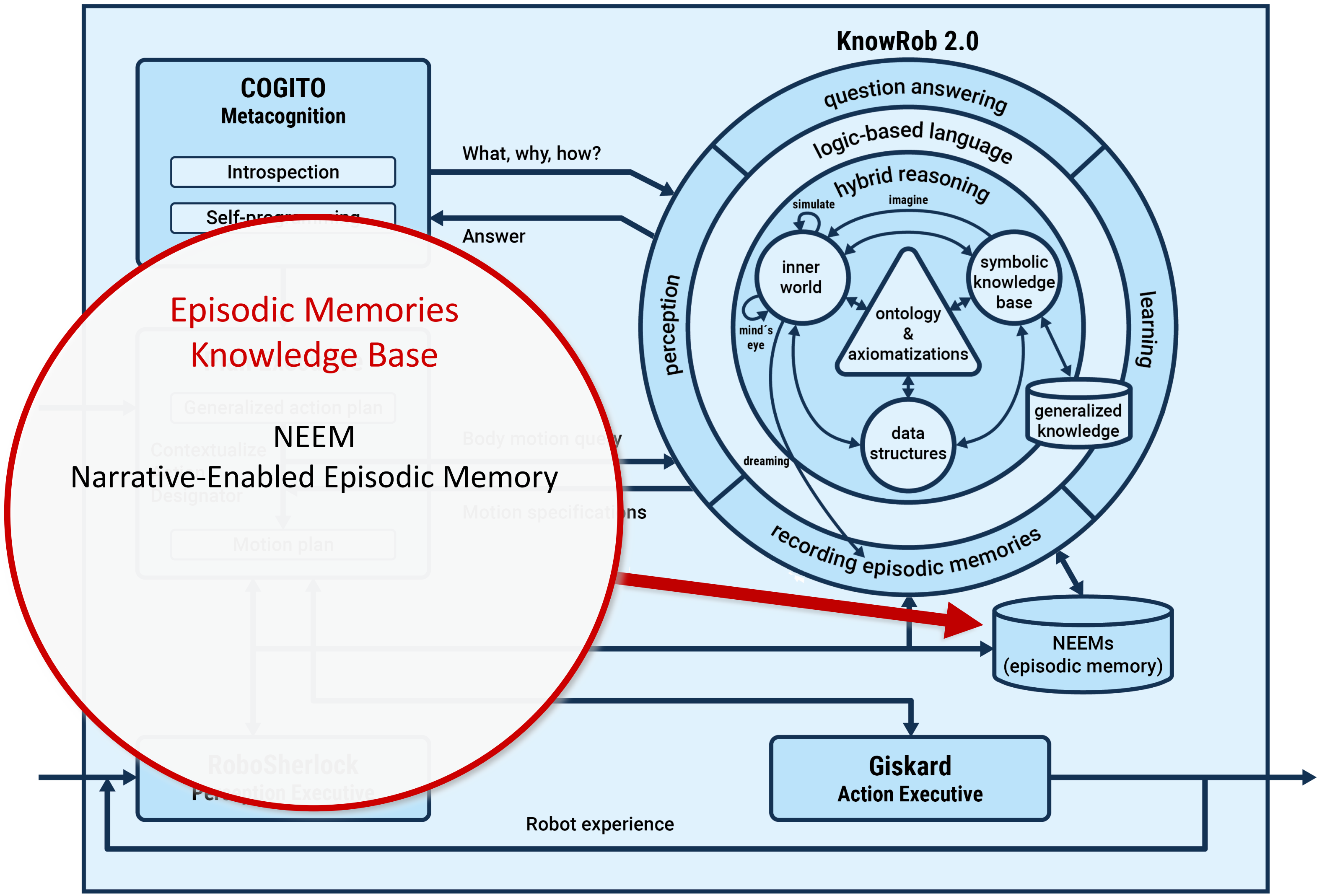
The ontology organizes knowledge about

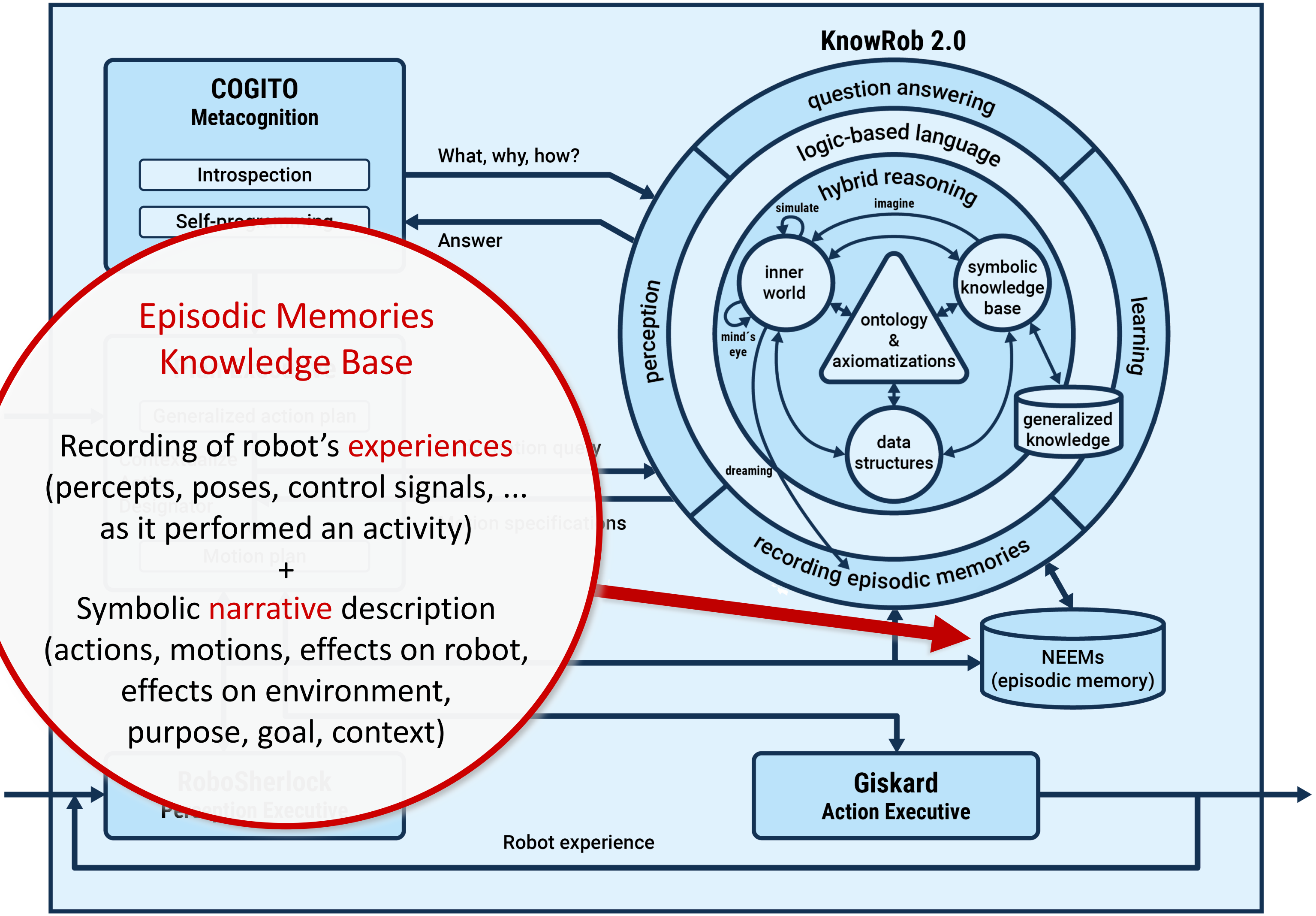
- Robot configuration
- Object configurations
- Robot actions, tasks, activities, & behaviours,
- Environment configuration
- Situational context


KnowRob 2.0
COGITO
Metacognition
Introspection
Self-programming
What, why, how?
Answer
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
ontology & axiomatizations
mind´s eye
perception
learning
generalized knowledge
data structures
dreaming
recording episodic memories
NEEMs (episodic memory)
Giskard
Action Executive
Robot experience
otion query
ifications
Ontology & Axiomatizations
The axioms identify roles that objects can play
e.g. a mug is a cylindrical vessel, with a handle, that can be used as a receptacle from which its contents can by drunk, mixed, or poured.



KnowRob 2.0
COGITO
Metacognition
Introspection
What, why, how?
Answer
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
mind´s eye
axiomatizations
perception
learning
generalized knowledge
data structures
dreaming
recording episodic memories
NEEMs (episodic memory)
Giskard
Action Executive
Robot experience
Generalized Symbolic Knowledge Base
The extraction of the generalized symbolic knowledge from NEEMS is facilitated by an interface to the Weka machine learning framework


KnowRob 2.0
COGITO
Metacognition
Introspection
Self-programming
What, why, how?
Answer
Episodic Memories
Knowledge Base
NEEM
Narrative-Enabled Episodic Memory
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
ontology & axiomatizations
mind´s eye
data structures
dreaming
generalized knowledge
perception
learning
recording episodic memories
NEEMs (episodic memory)
Giskard
Action Executive
Robot experience

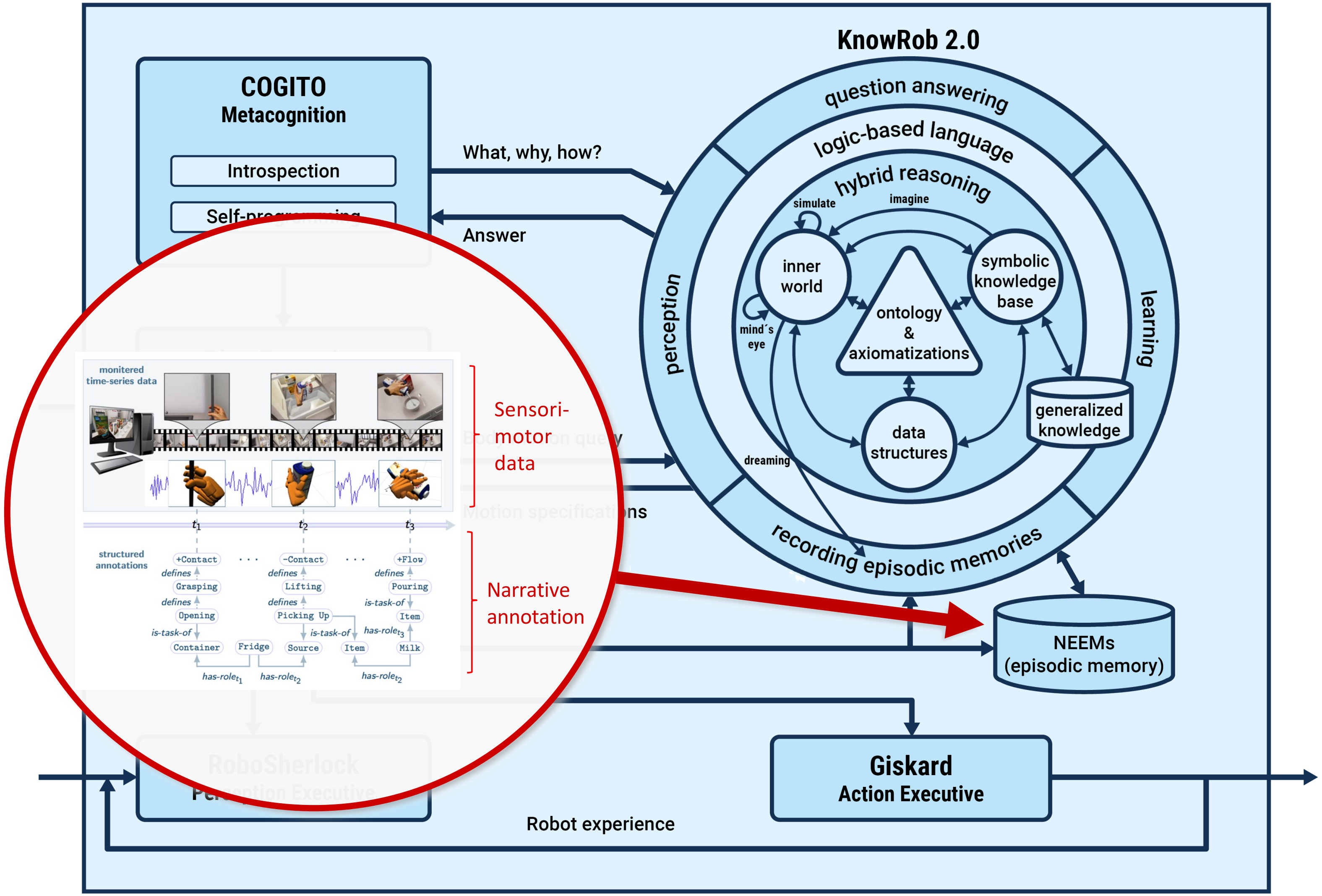


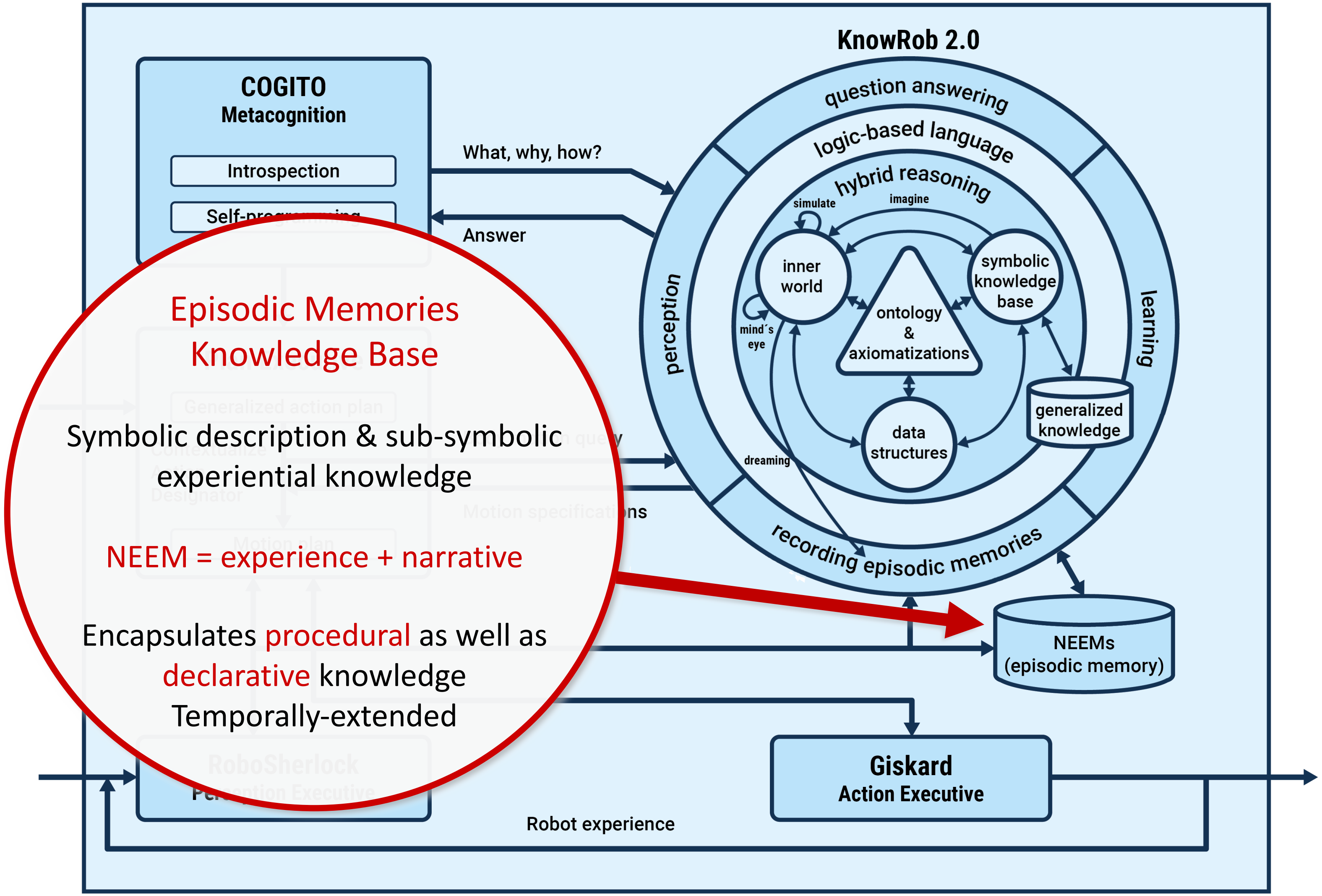
Episodic Memories Knowledge Base

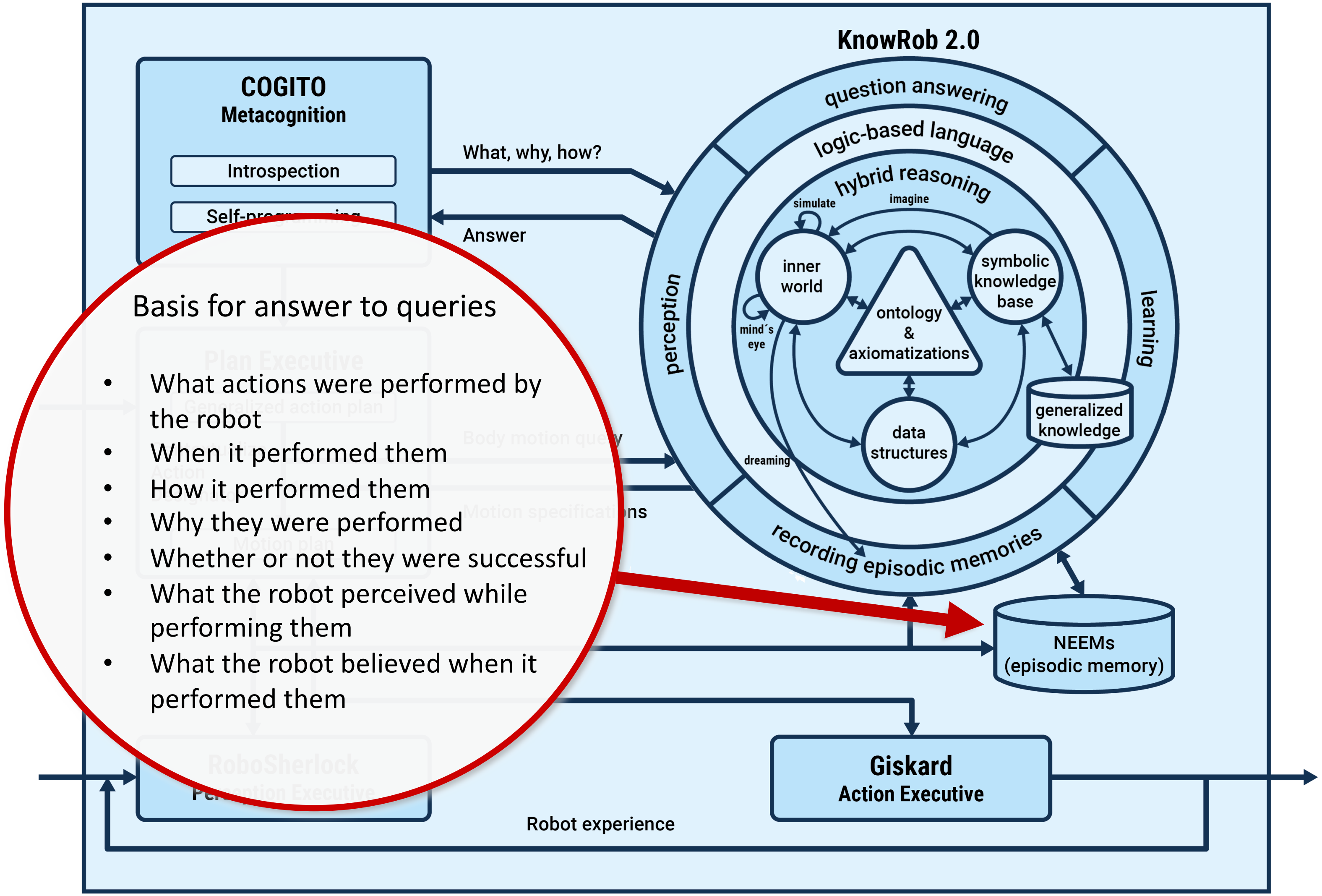
Recording of robot's experiences (percepts, poses, control signals, ... as it performed an activity)

\+

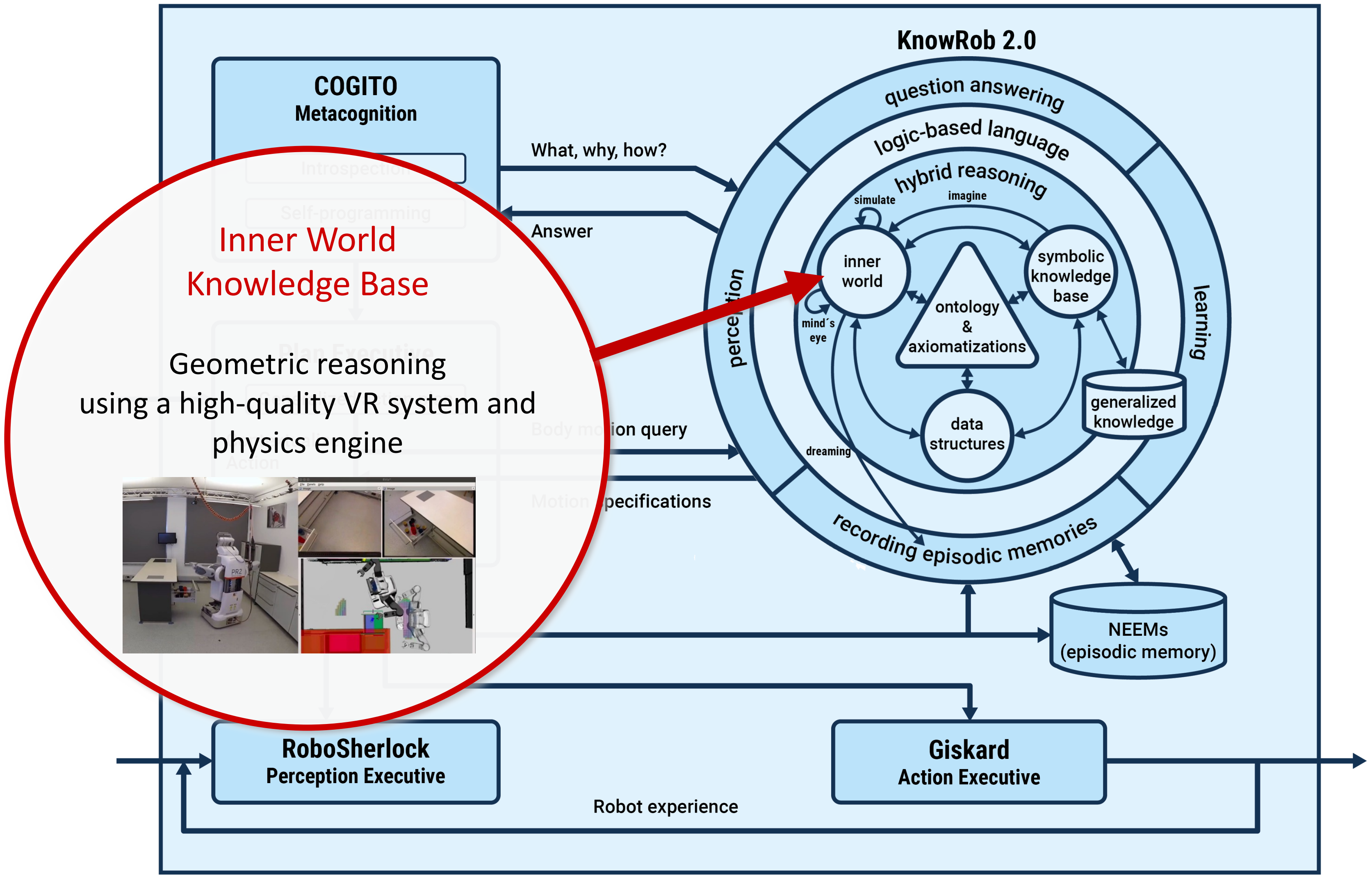
Symbolic narrative description (actions, motions, effects on robot, effects on environment, purpose, goal, context)

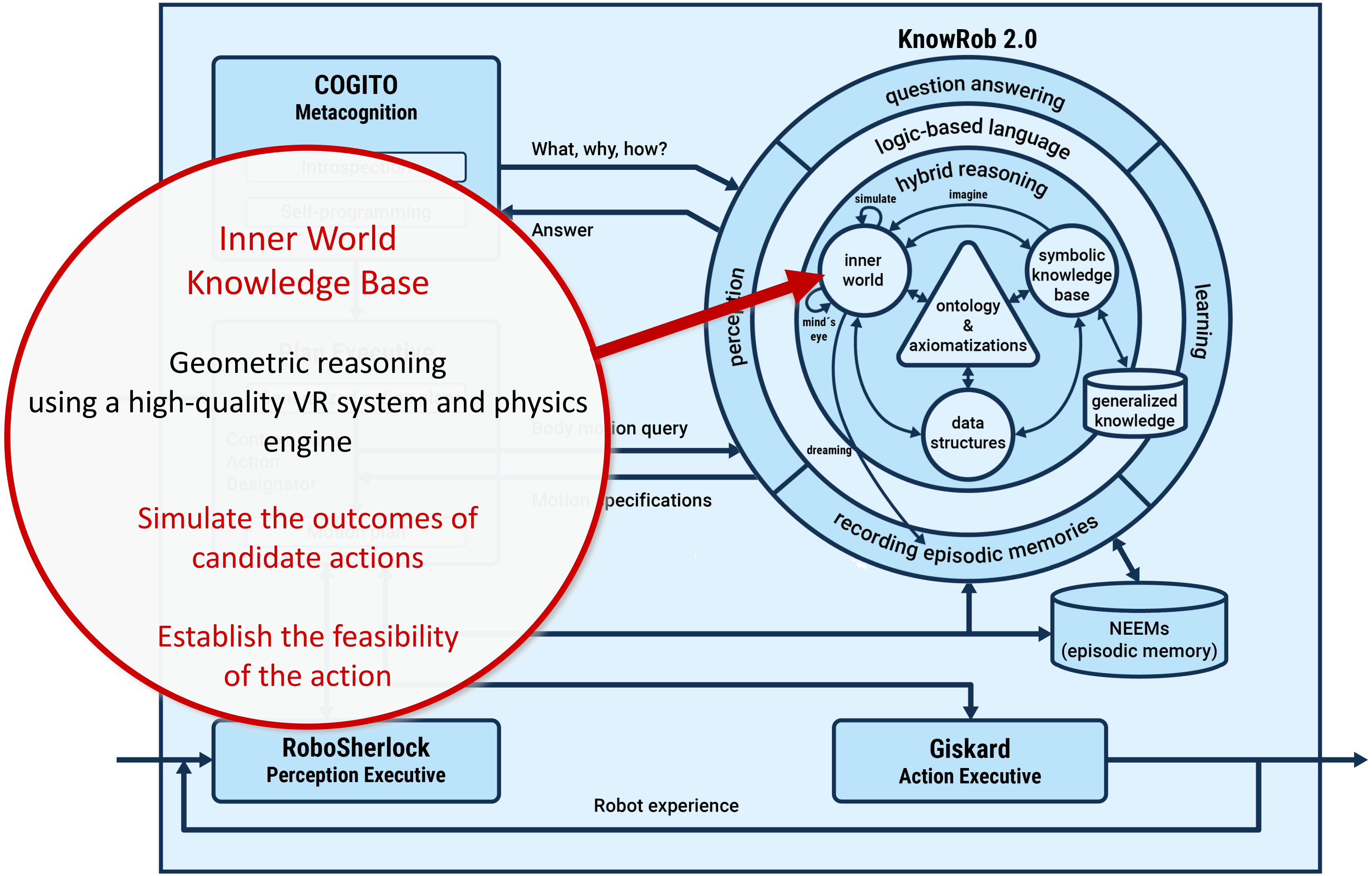

KnowRob 2.0
COGITO
Metacognition
Introspection
What, why, how?
Answer
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
ontology & axiomatizations
mind´s eye
data structures
generalized knowledge
dreaming
perception
learning
recording episodic memories
NEEMs (episodic memory)
Giskard
Action Executive
Robot experience
monitered time-series data
structured annotations
Sensori-motor data
Narrative annotation
+Contact
-Contact
+Flow
defines
Grasping
Lifting
Pouring
Opening
Picking Up
Item
is-task-of
has-role
Container
Fridge
Source
Milk


KnowRob 2.0
COGITO
Metacognition
Introspection
What, why, how?
Answer
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
ontology & axiomatizations
mind´s eye
data structures
dreaming
generalized knowledge
perception
learning
recording episodic memories
NEEMs (episodic memory)
Giskard
Action Executive
Robot experience
Episodic Memories
Knowledge Base
Symbolic description & sub-symbolic experiential knowledge
NEEM = experience + narrative
Encapsulates procedural as well as declarative knowledge
Temporally-extended



COGITO
Metacognition
Introspection
What, why, how?
Answer
KnowRob 2.0
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
ontology & axiomatizations
mind´s eye
data structures
generalized knowledge
dreaming
perception
learning
recording episodic memories
NEEMs (episodic memory)
Giskard
Action Executive
Robot experience
Basis for answer to queries
• What actions were performed by the robot
• When it performed them
• How it performed them
• Why they were performed
• Whether or not they were successful
• What the robot perceived while performing them
• What the robot believed when it performed them

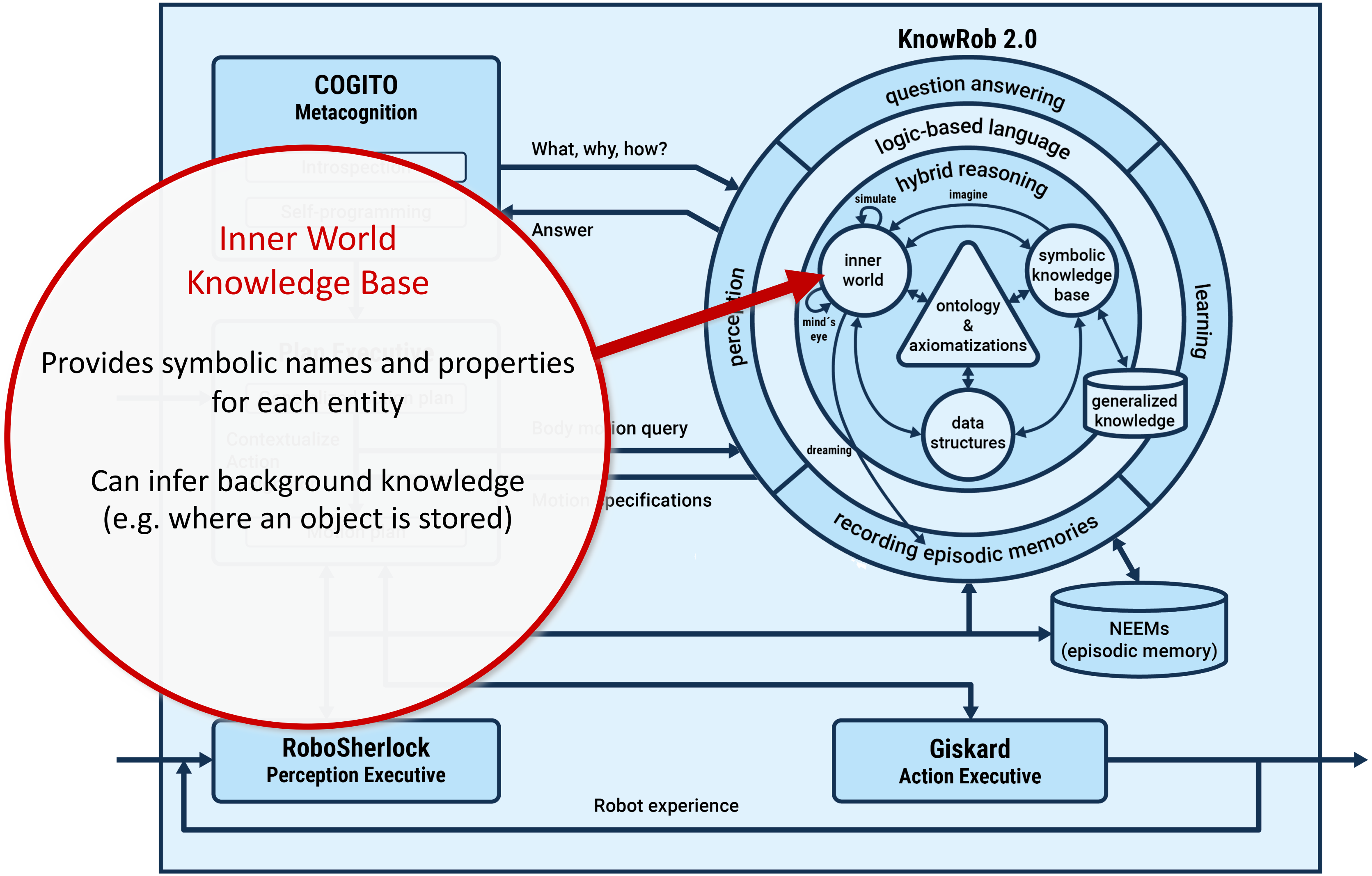

KnowRob 2.0
COGITO
Metacognition
What, why, how?
Answer
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
ontology & axiomatizations
mind´s eye
perception
learning
generalized knowledge
data structures
dreaming
recording episodic memories
NEEMs (episodic memory)
Inner World Knowledge Base
Geometric reasoning using a high-quality VR system and physics engine
ion query
pecifications
RoboSherlock
Perception Executive
Giskard
Action Executive
Robot experience


KnowRob 2.0
COGITO
Metacognition
What, why, how?
Answer
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
ontology & axiomatizations
mind´s eye
data structures
generalized knowledge
dreaming
perception
learning
recording episodic memories
NEEMs (episodic memory)
RoboSherlock
Perception Executive
Giskard
Action Executive
Robot experience
Inner World
Knowledge Base
Geometric reasoning
using a high-quality VR system and physics engine
Simulate the outcomes of candidate actions
Establish the feasibility of the action

Inner World Knowledge Base

Provides symbolic names and properties for each entity

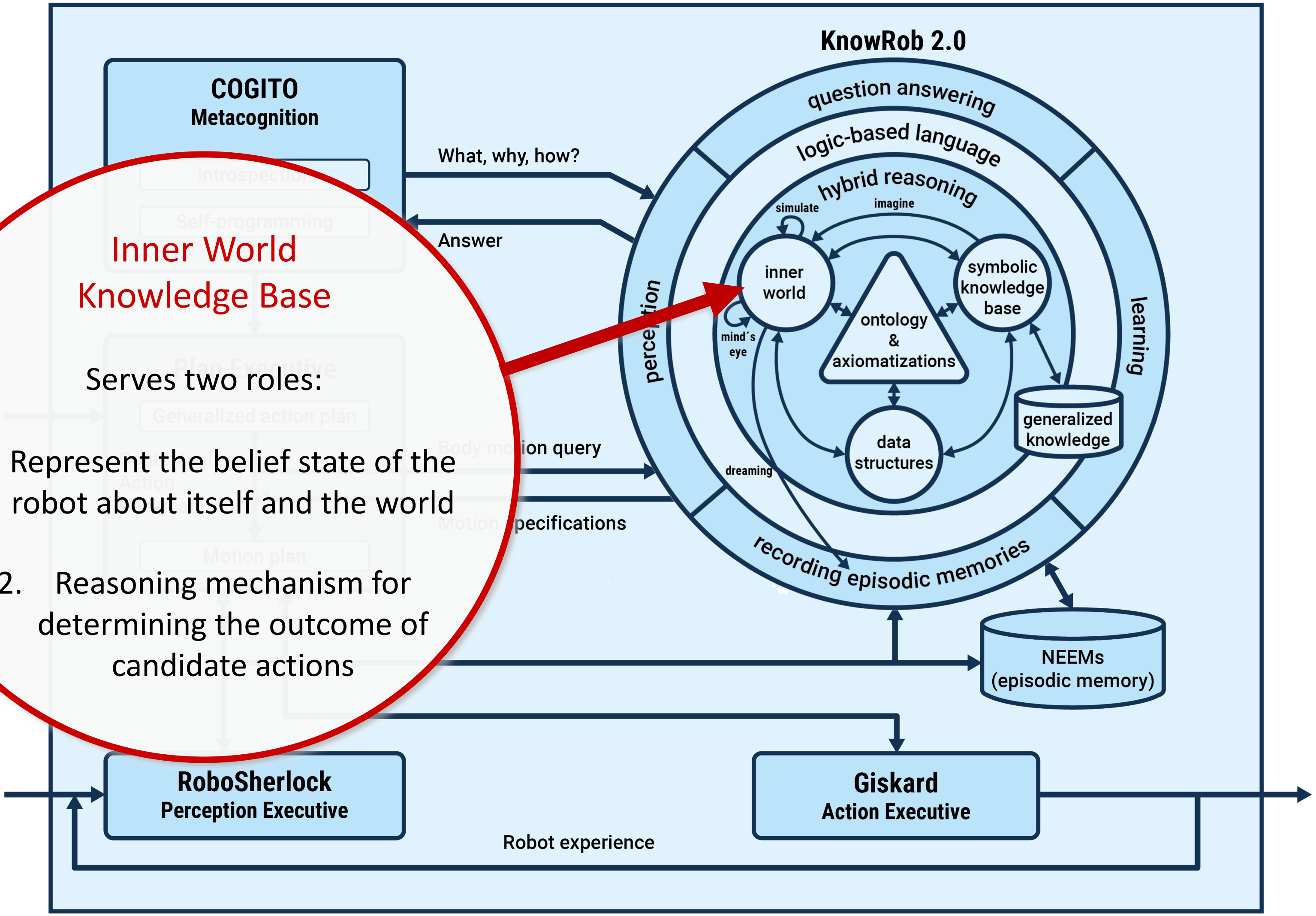
Can infer background knowledge (e.g. where an object is stored)



Inner World Knowledge Base

Serves two roles:

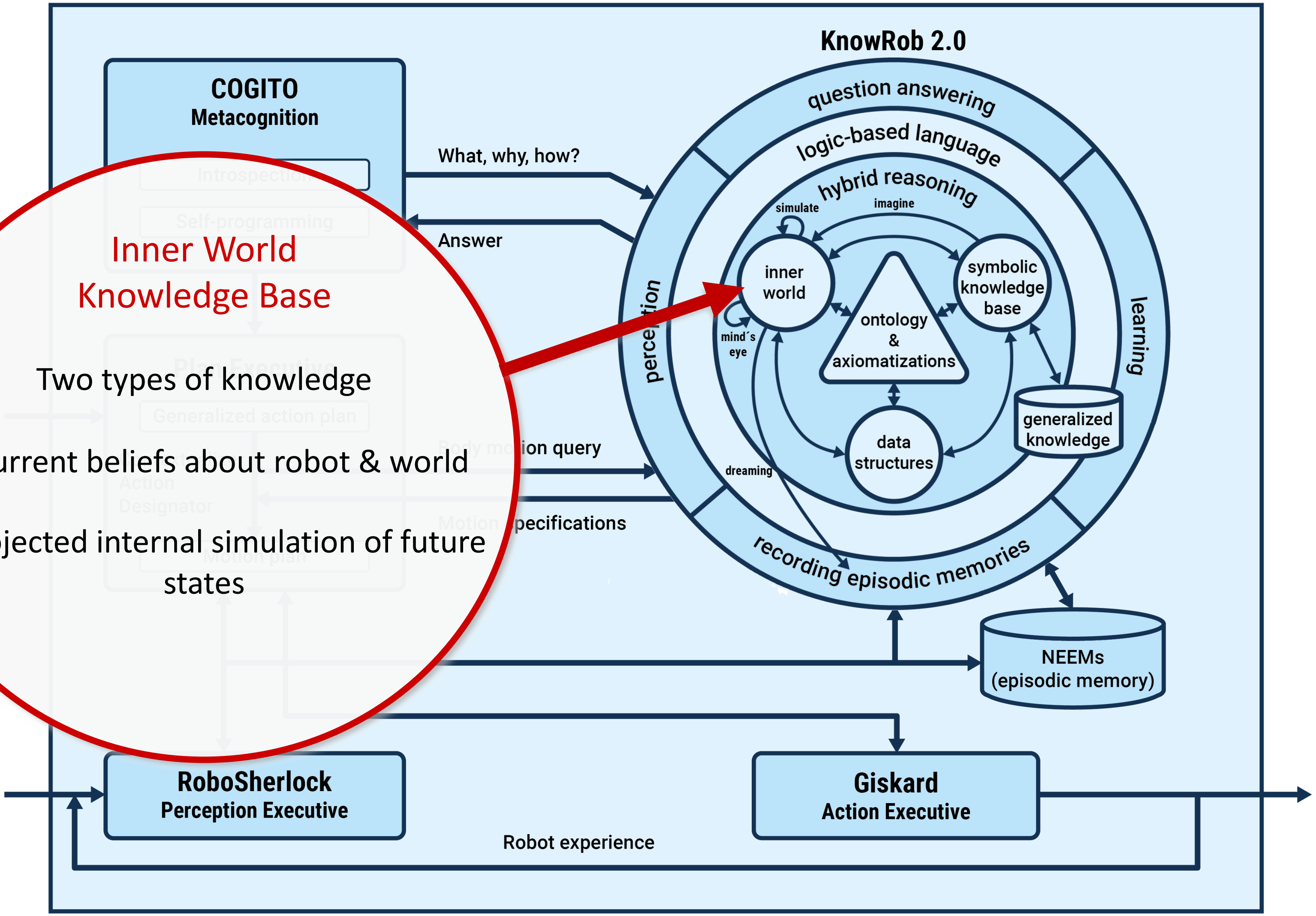
1. Represent the belief state of the robot about itself and the world
2. Reasoning mechanism for determining the outcome of candidate actions



Inner World Knowledge Base

Two types of knowledge

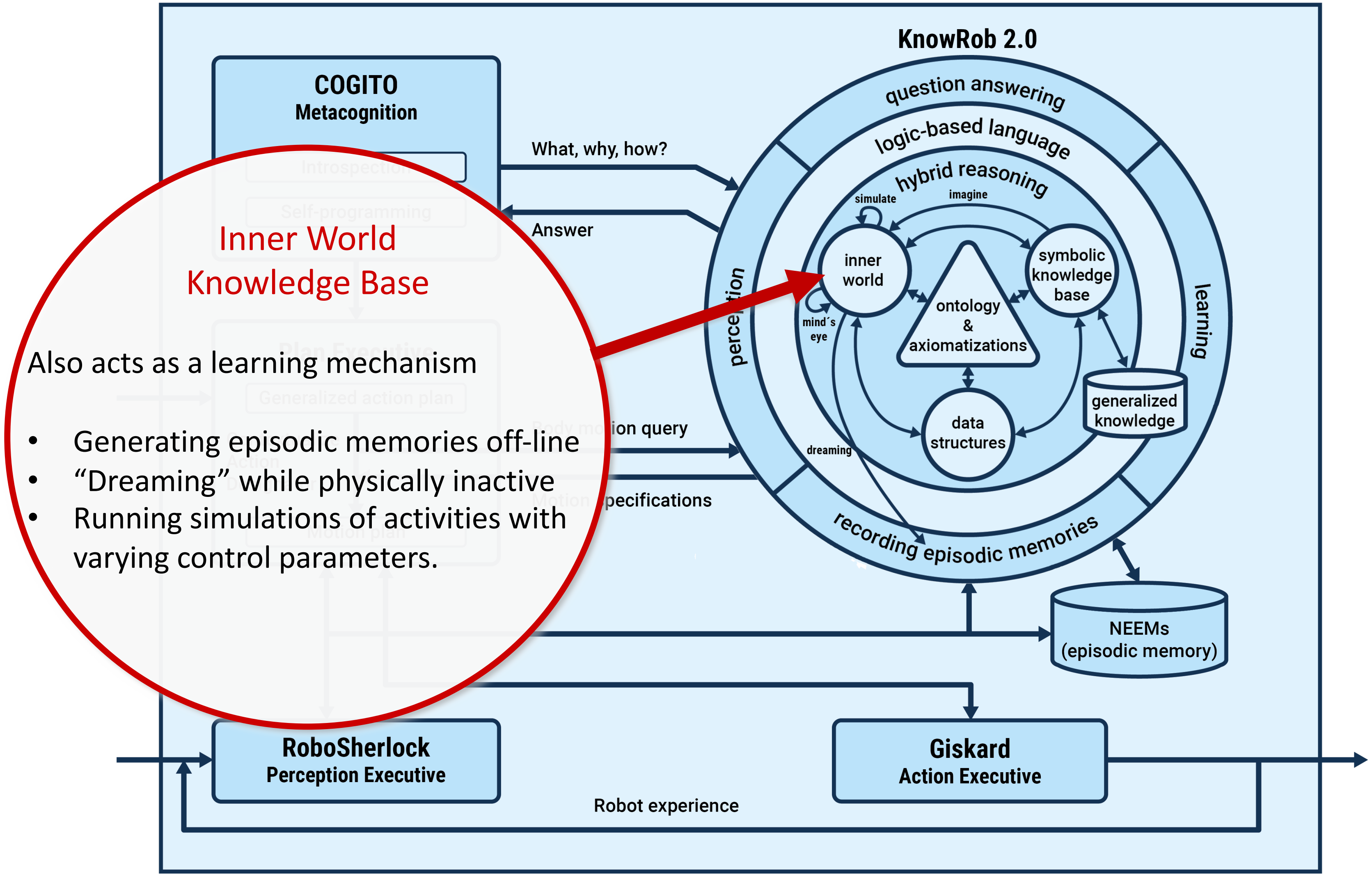
1. Current beliefs about robot & world
2. Projected internal simulation of future states



Inner World Knowledge Base

Also acts as a learning mechanism

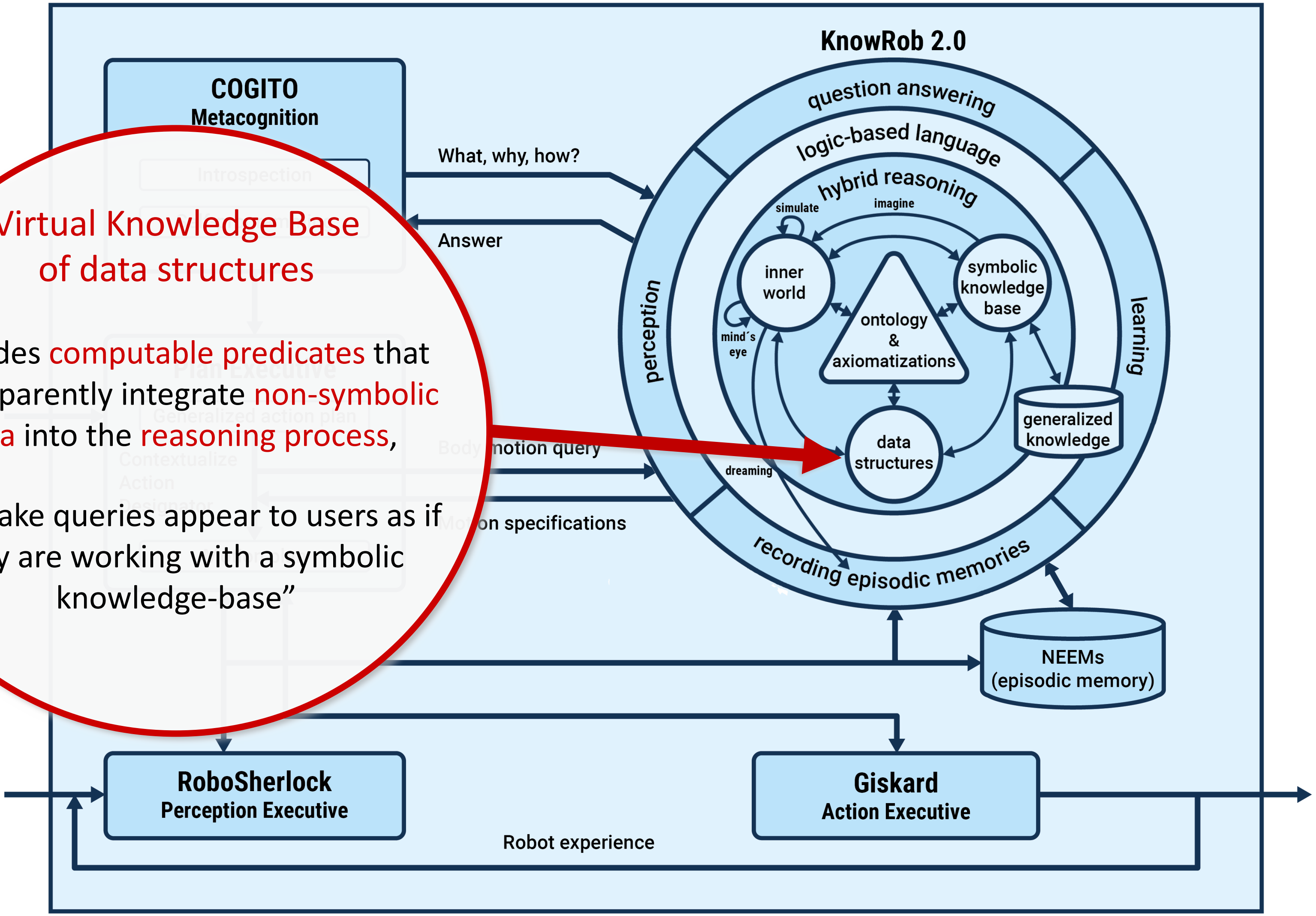
- Generating episodic memories off-line
- “Dreaming” while physically inactive
- Running simulations of activities with varying control parameters.

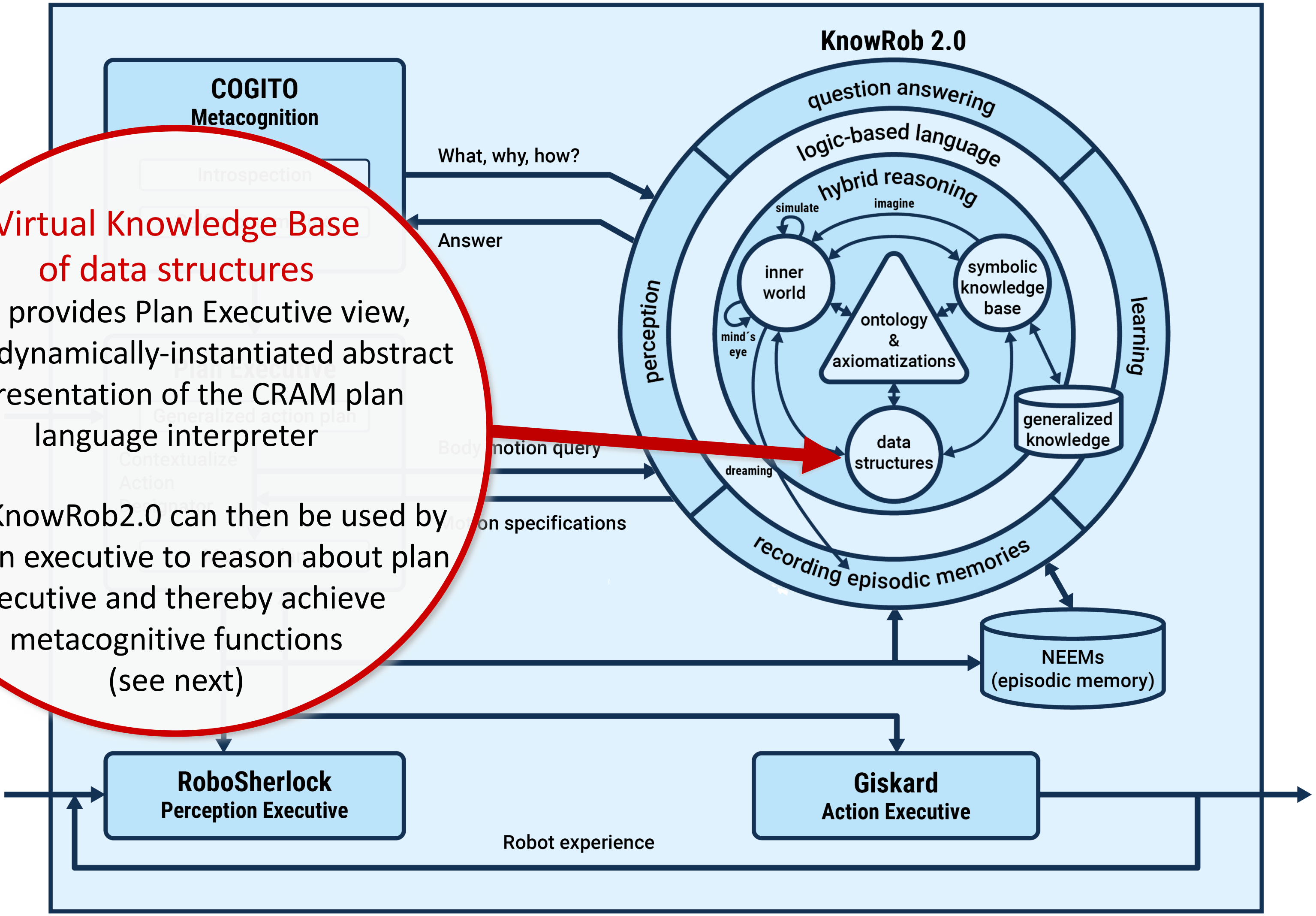


Virtual Knowledge Base of data structures

Provides computable predicates that "transparently integrate non-symbolic data into the reasoning process,

and make queries appear to users as if they are working with a symbolic knowledge-base"

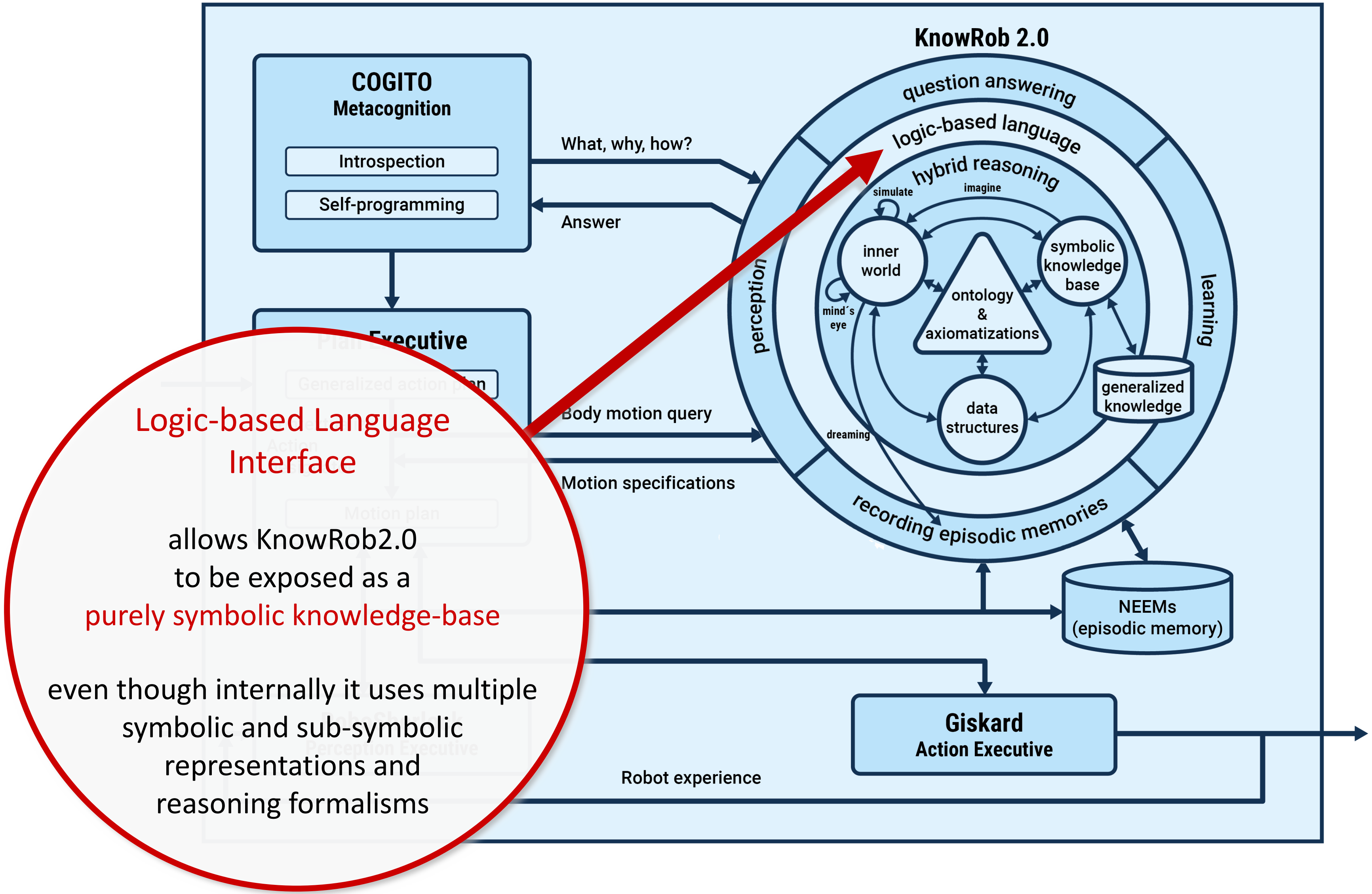




Virtual Knowledge Base of data structures

Also provides Plan Executive view, i.e. a a dynamically-instantiated abstract representation of the CRAM plan language interpreter

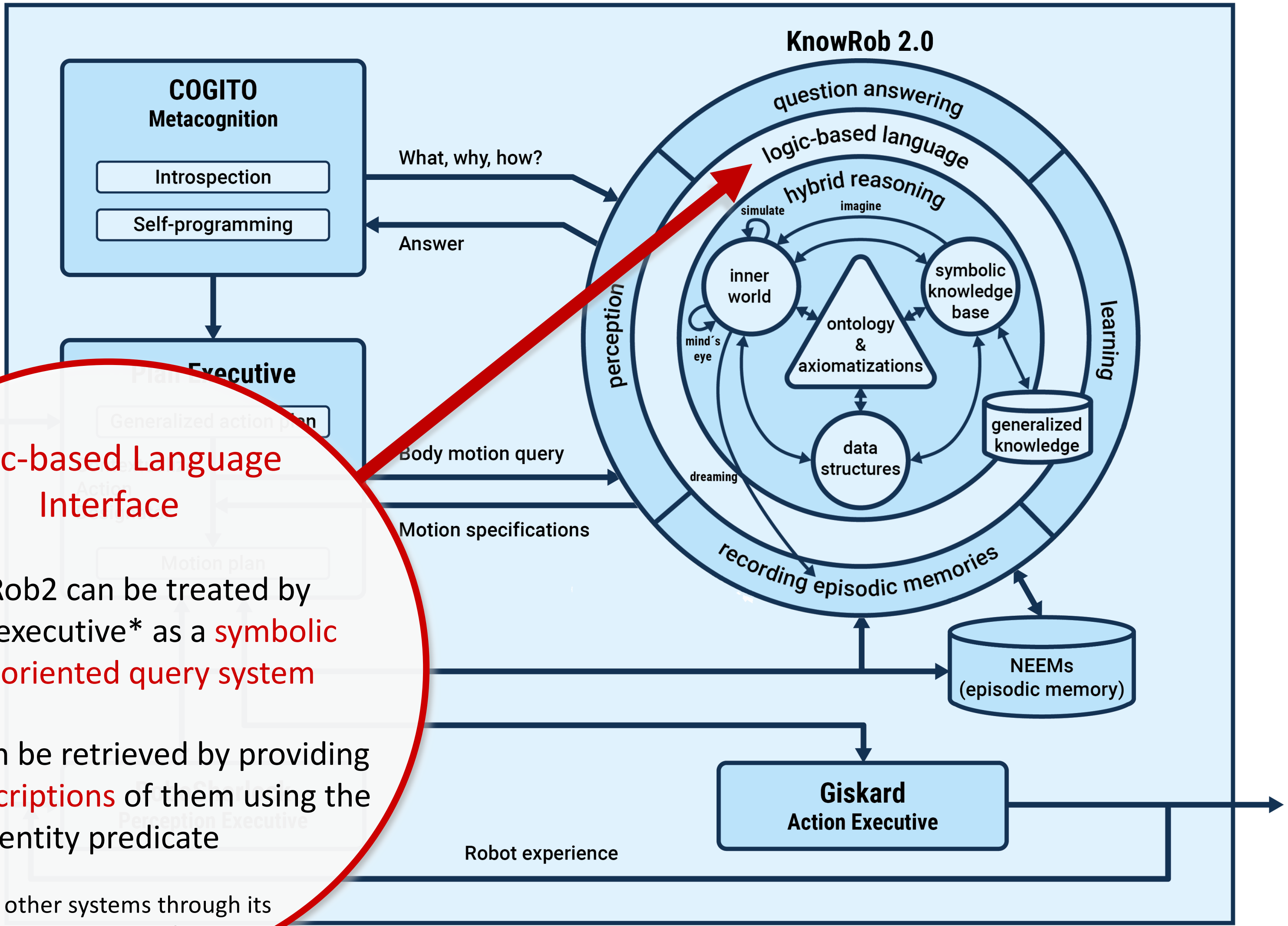
Thus, KnowRob2.0 can then be used by the plan executive to reason about plan executive and thereby achieve metacognitive functions (see next)



Logic-based Language Interface

allows KnowRob2.0 to be exposed as a purely symbolic knowledge-base

even though internally it uses multiple symbolic and sub-symbolic representations and reasoning formalisms

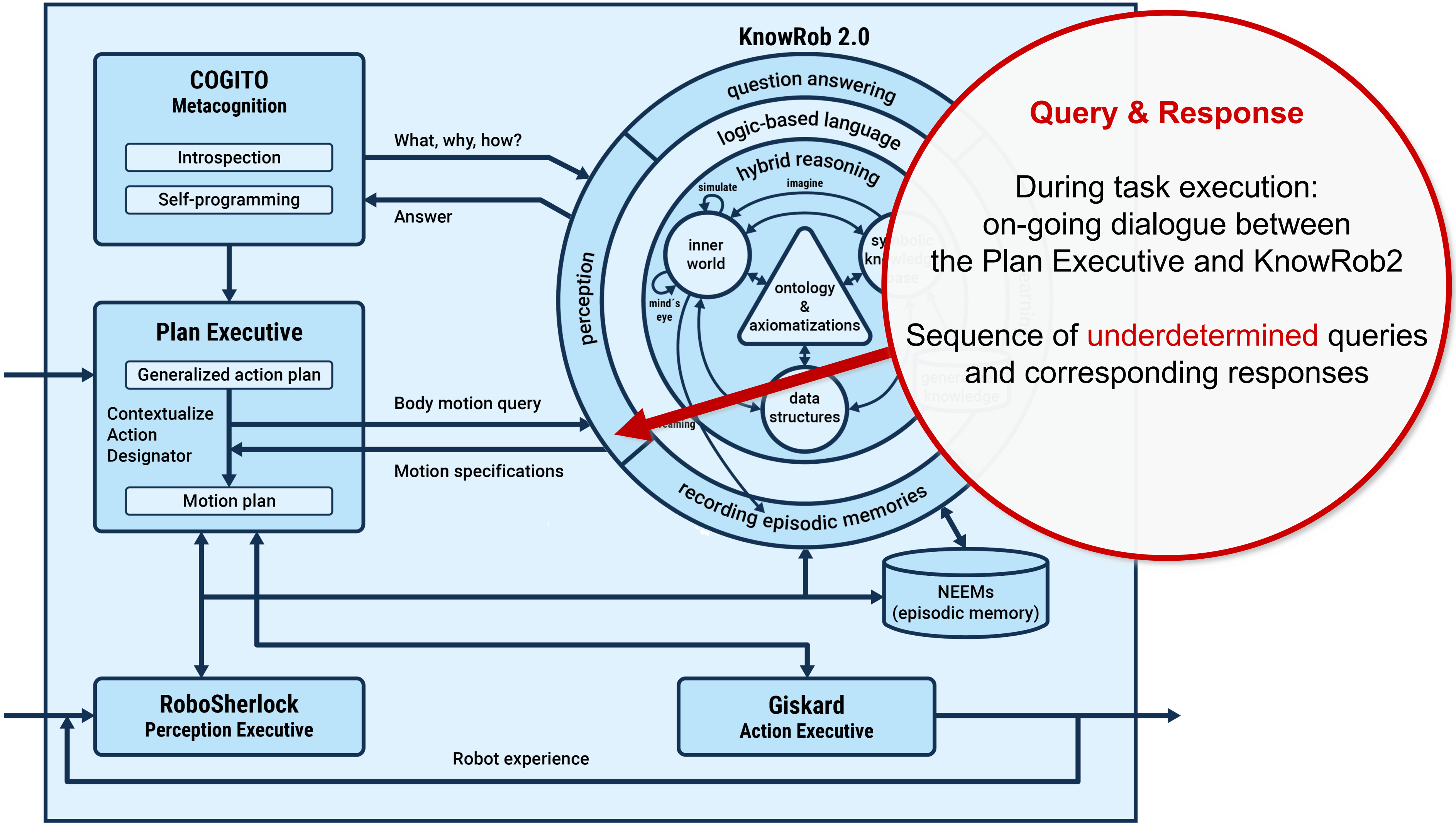


Logic-based Language Interface

KnowRob2 can be treated by the CPL executive* as a symbolic object-oriented query system

Entities can be retrieved by providing partial descriptions of them using the entity predicate

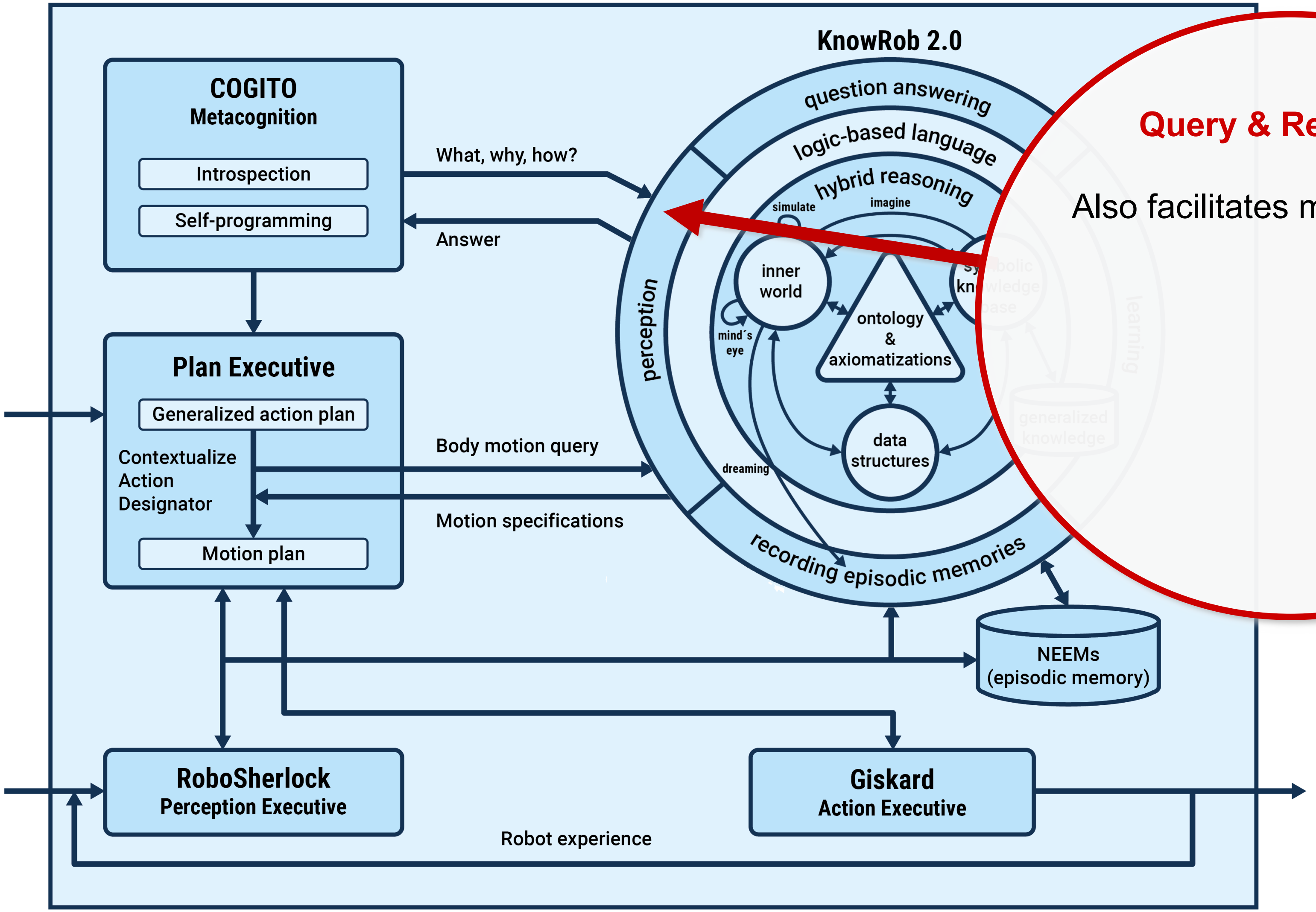
*and other systems through its OpenEASE interface



Query & Response

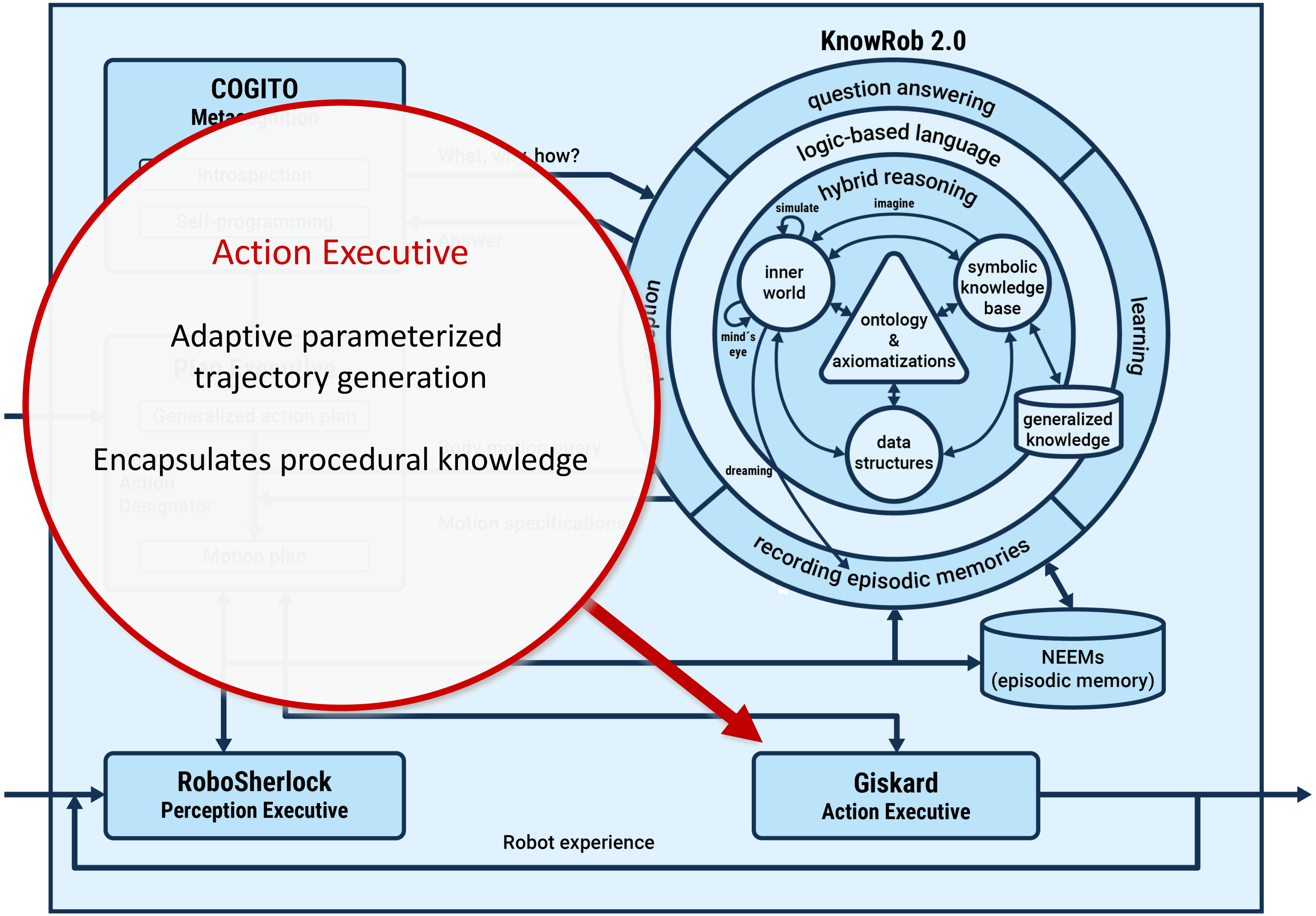
During task execution:
on-going dialogue between
the Plan Executive and KnowRob2

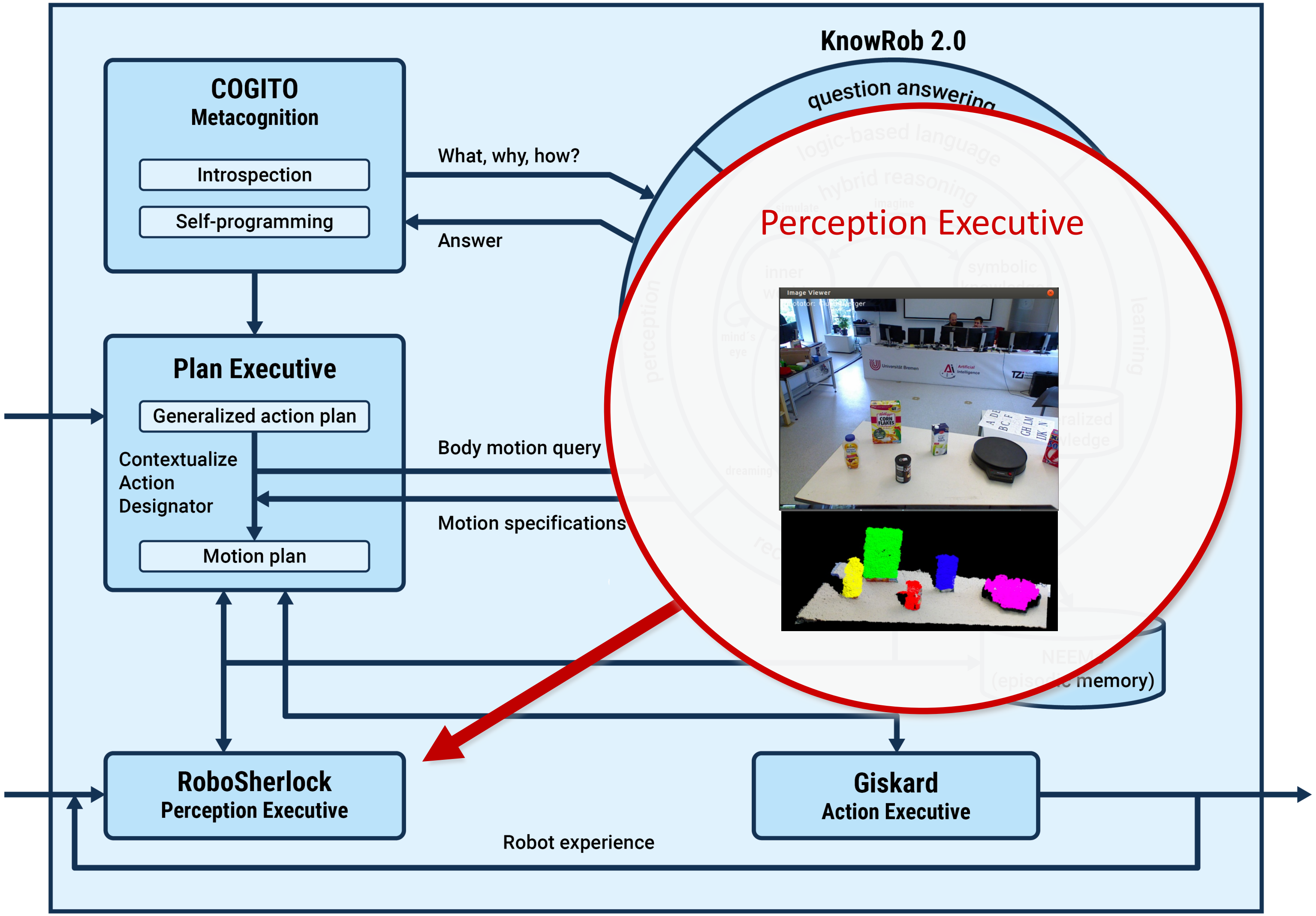
Sequence of underdetermined queries
and corresponding responses

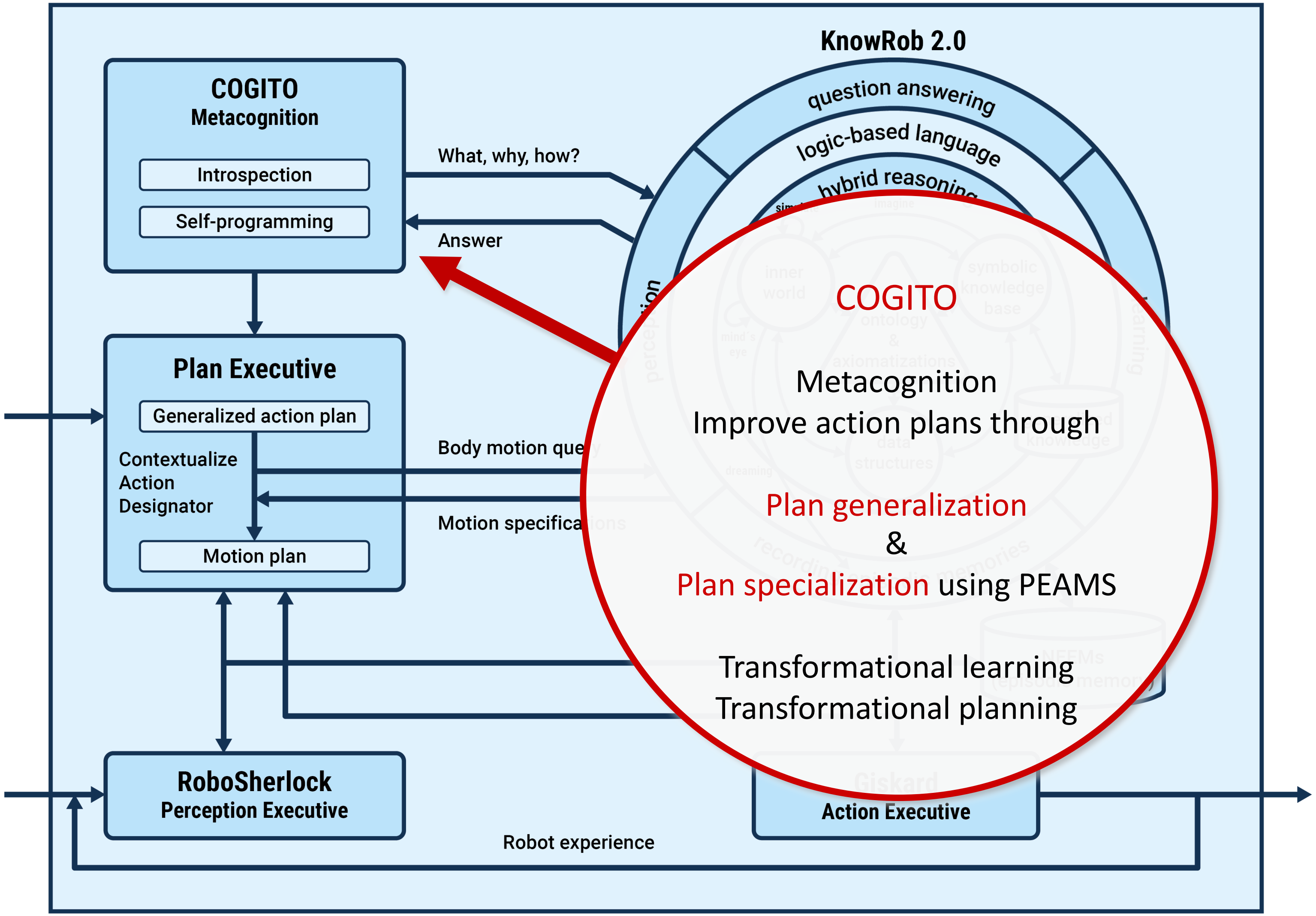


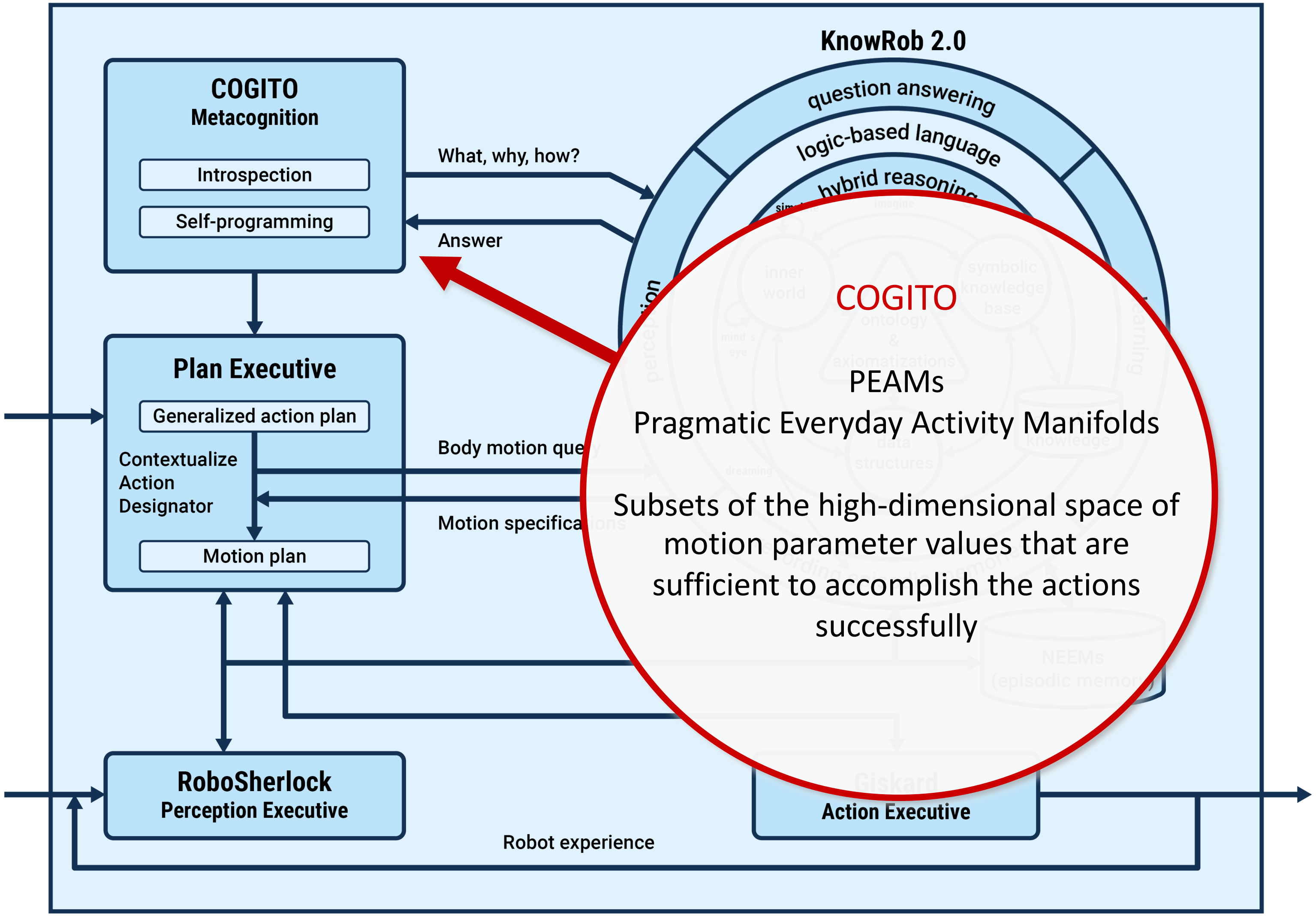
Query & Response

Also facilitates metacognition

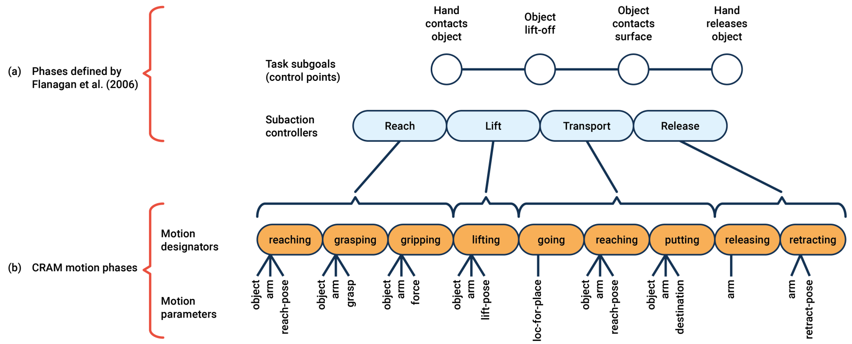

KnowRob 2.0
question answering
logic-based language
hybrid reasoning
simulate
imagine
inner world
symbolic knowledge base
mind´s eye
ontology & axiomatizations
generalized knowledge
data structures
dreaming
learning
recording episodic memories
NEEMs (episodic memory)
COGITO
What, why, how?
Action Executive
Adaptive parameterized trajectory generation
Encapsulates procedural knowledge
RoboSherlock
Perception Executive
Giskard
Action Executive
Robot experience


KnowRob 2.0
question answering
COGITO
Metacognition
Introspection
Self-programming
What, why, how?
Answer
Perception Executive
Plan Executive
Generalized action plan
Contextualize
Action
Designator
Motion plan
Body motion query
Motion specifications
memory)
RoboSherlock
Perception Executive
Giskard
Action Executive
Robot experience


KnowRob 2.0
question answering
logic-based language
hybrid reasoning
COGITO
Metacognition
Introspection
Self-programming
What, why, how?
Answer
Plan Executive
Generalized action plan
Contextualize
Action
Designator
Motion plan
Body motion que
Motion specifica
RoboSherlock
Perception Executive
Action Executive
Robot experience
COGITO
Metacognition
Improve action plans through
Plan generalization
&
Plan specialization using PEAMS
Transformational learning
Transformational planning


KnowRob 2.0
question answering
logic-based language
hybrid reasoning
COGITO
Metacognition
Introspection
Self-programming
What, why, how?
Answer
Plan Executive
Generalized action plan
Contextualize
Action
Designator
Motion plan
RoboSherlock
Perception Executive
Robot experience
Action Executive
COGITO
PEAMs
Pragmatic Everyday Activity Manifolds
Subsets of the high-dimensional space of motion parameter values that are sufficient to accomplish the actions successfully

Walk through the execution of a generalized action plan



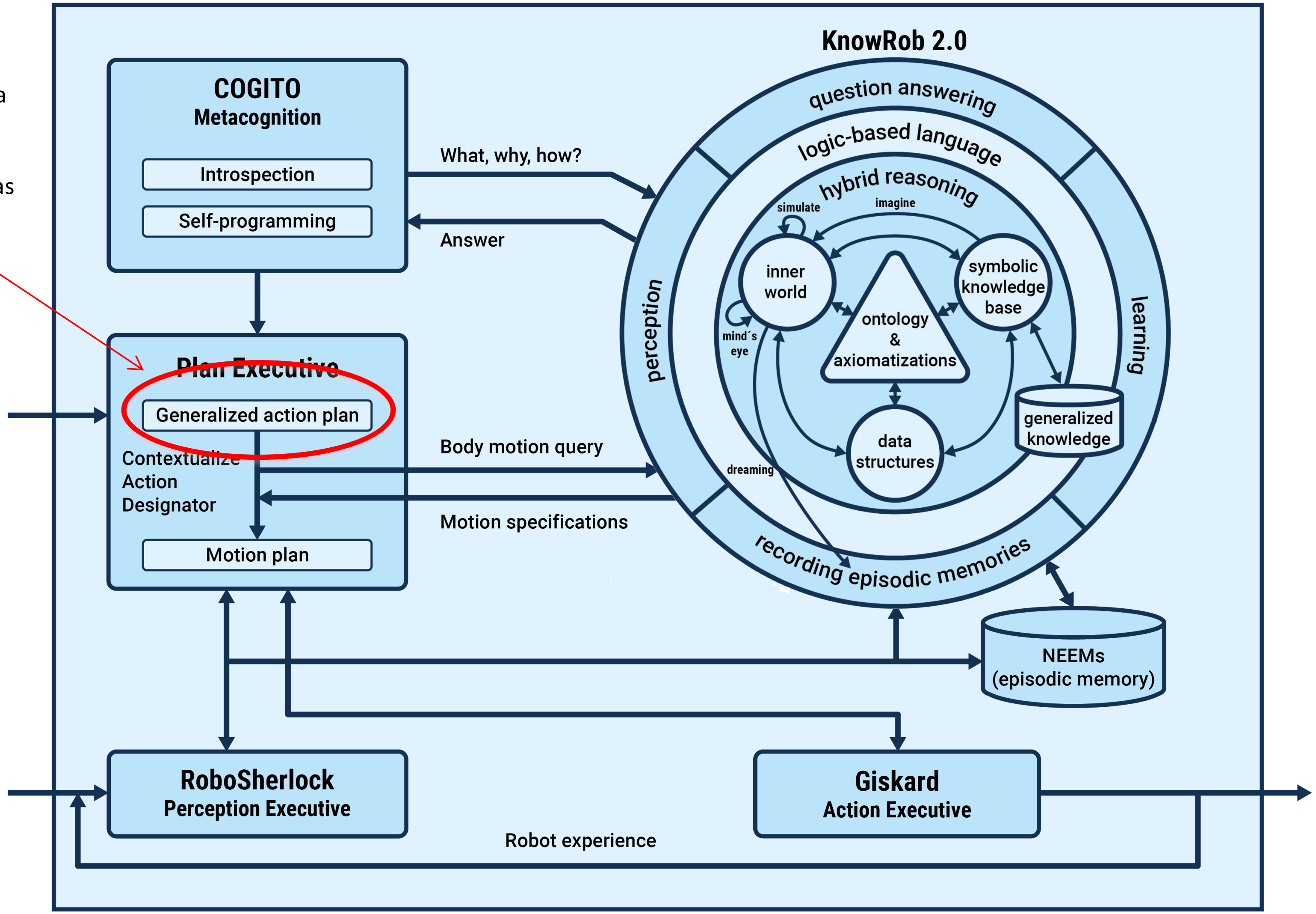
Generalized Action Plan

Motion Plan

The motion parameter values are provided by the generative model

3-step refinement process:
contextualization
Identify the values of the parameters to the motion plan that maximize the likelihood that the associated body motions successfully accomplish the desired action

Recall:
The robot agent is equipped with a generalized action plan for each action category, which typically corresponds to action verbs such as fetch, place, pour, and cut.

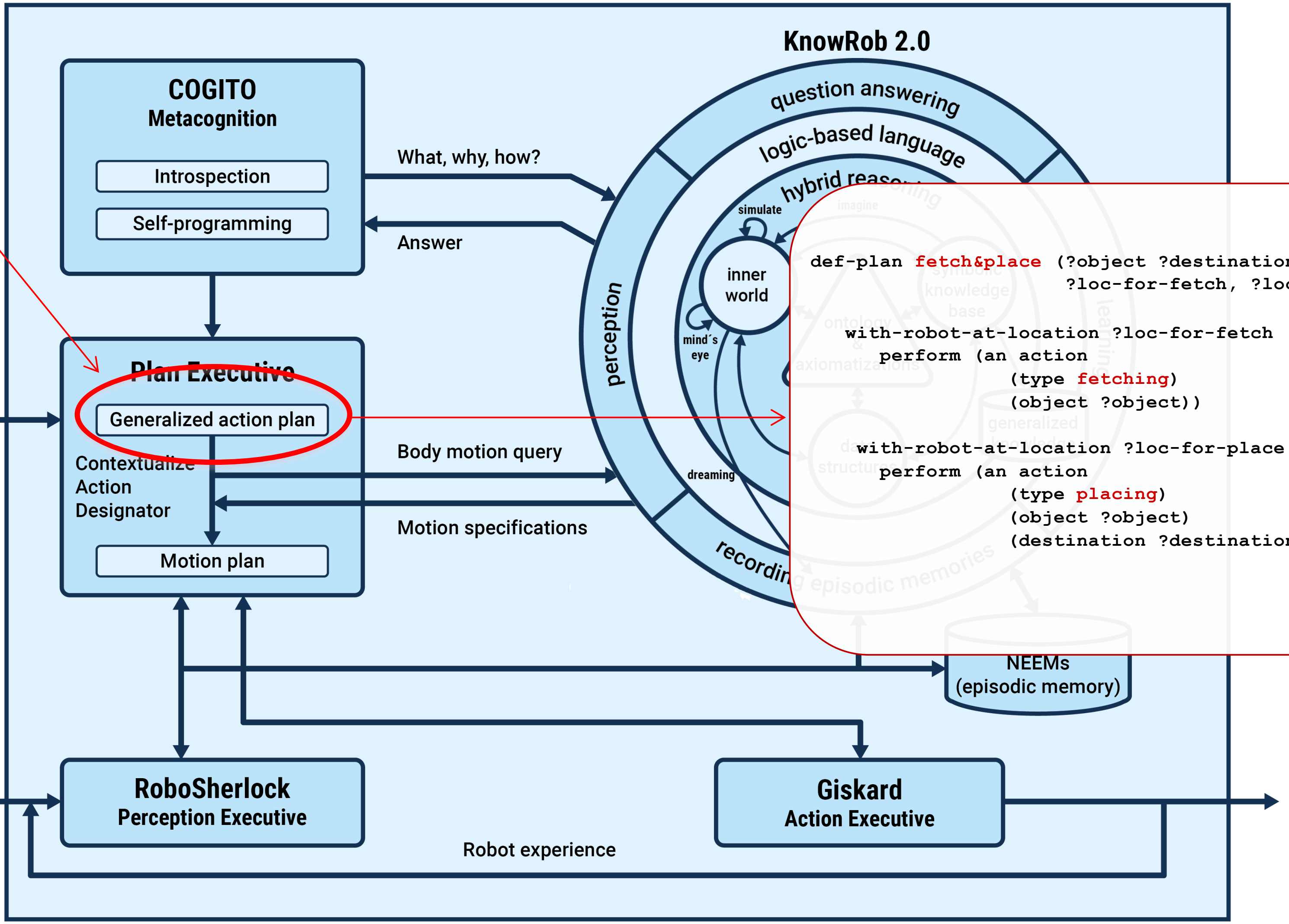


An action plan is invoked with
a request to perform
an underdetermined action description,

e.g. *fetch the cup and place it on the table*

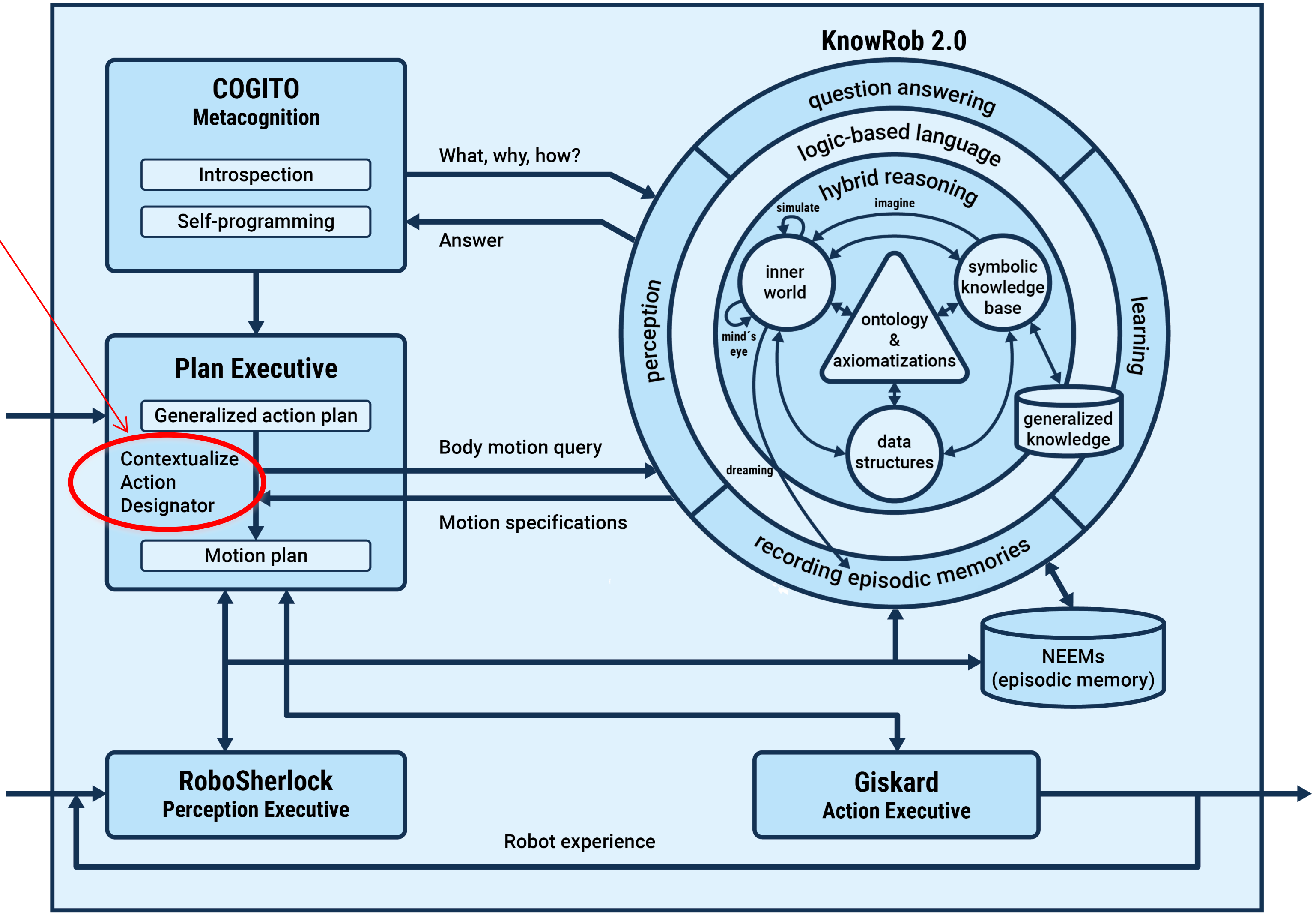
by selecting the generalized action plan
for the action category
corresponding to the action description

(e.g. see the spoon challenge at https://www.youtube.com/watch?v=hrnoY6J8ddE&feature=youtu.be)




```
def-plan fetch&place (?object ?destination,
                      ?loc-for-fetch, ?loc-for-place)

  with-robot-at-location ?loc-for-fetch
    perform (an action
                (type fetching)
                (object ?object))

   with-robot-at-location ?loc-for-place
    perform (an action
                (type placing)
                (object ?object)
                (destination ?destination))
```

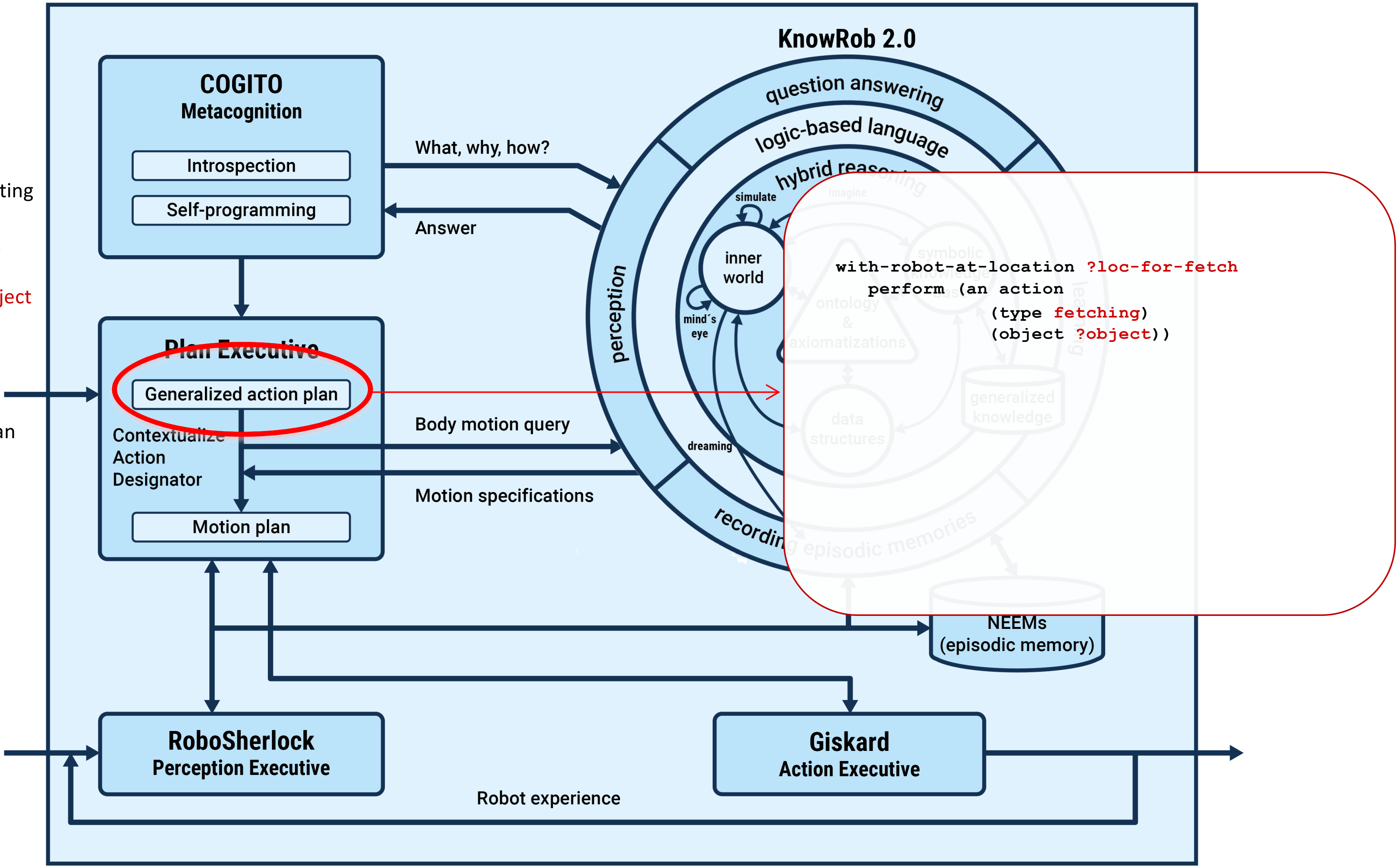
The Plan Executive interprets the generalized action plan in process referred to as contextualization in three steps:



The Plan Executive interprets the generalized action plan in process referred to as contextualization in three steps:

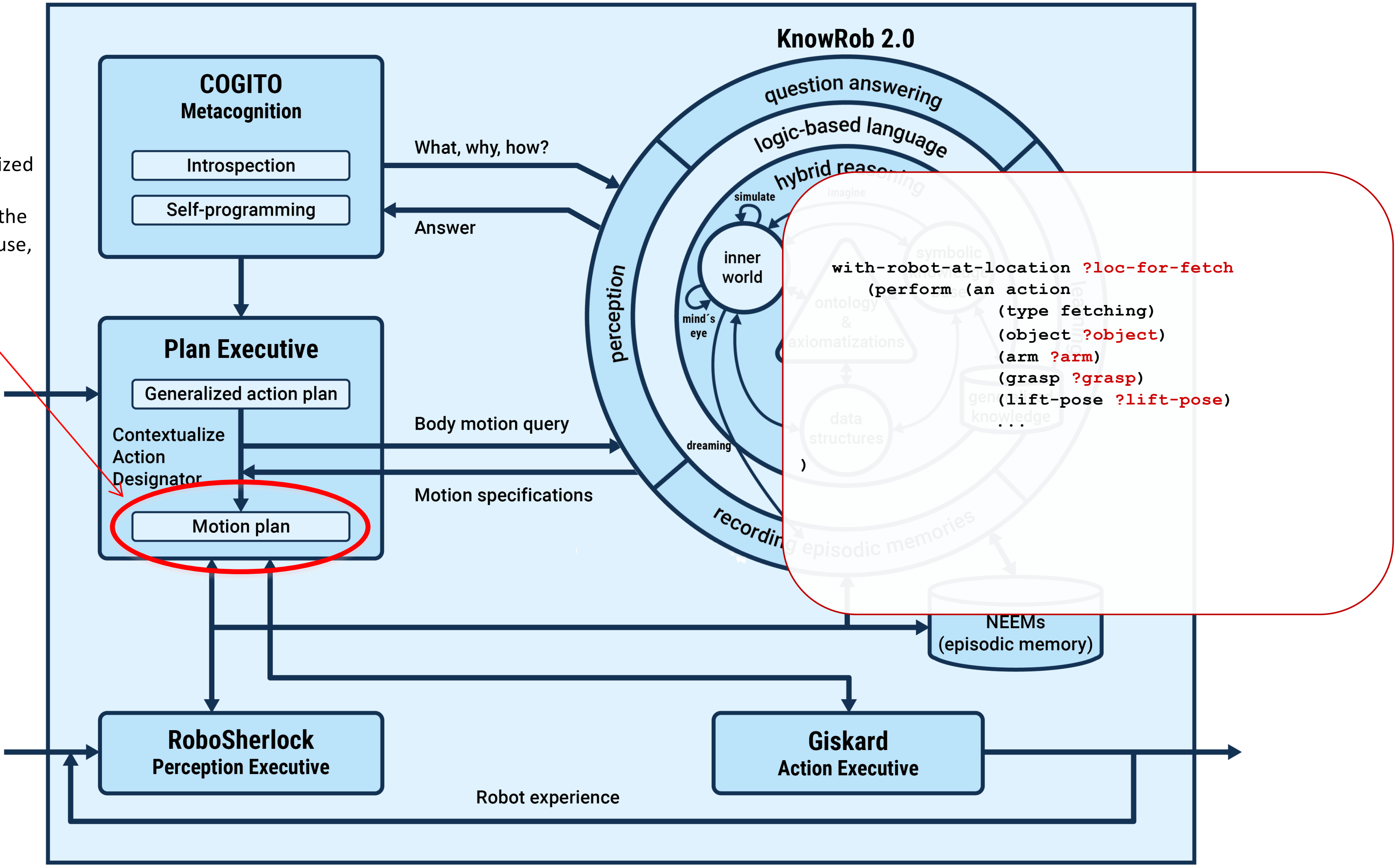
1. Instantiate the selected generalized action plan by inserting the arguments required for the specific action to be performed.

- For example, the type of the object to be manipulated or the destination location.
- These arguments are typically designators of some kind, e.g., an action, object, or location designator.



The Plan Executive interprets the generalized action plan in process referred to as contextualization in three steps:

2. Extend the instantiated generalized action plan by adding the parameters needed to execute the motion plan, e.g. which arm to use, what grasp pose to use.

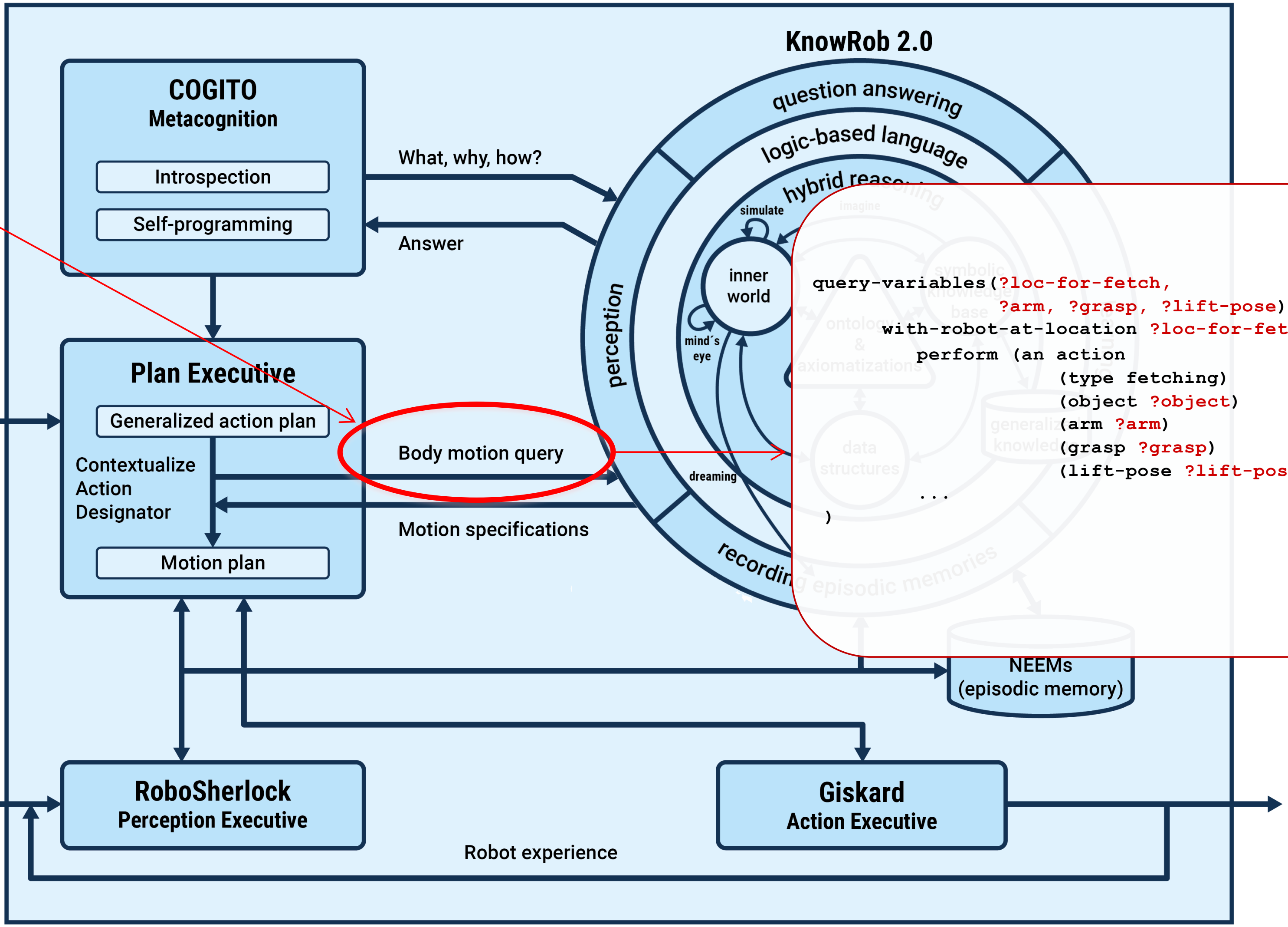



```
with-robot-at-location ?loc-for-fetch
  (perform (an action
            (type fetching)
            (object ?object)
            (arm ?arm)
            (grasp ?grasp)
            (lift-pose ?lift-pose)
            ...
)
```

The Plan Executive interprets the generalized action plan in process referred to as contextualization in three steps:

3. Create a query for the values of these parameters

(that would produce robot body motions to achieve the goal of the underdetermined action description and, equivalently, the associated instantiated extended generalized action plan).

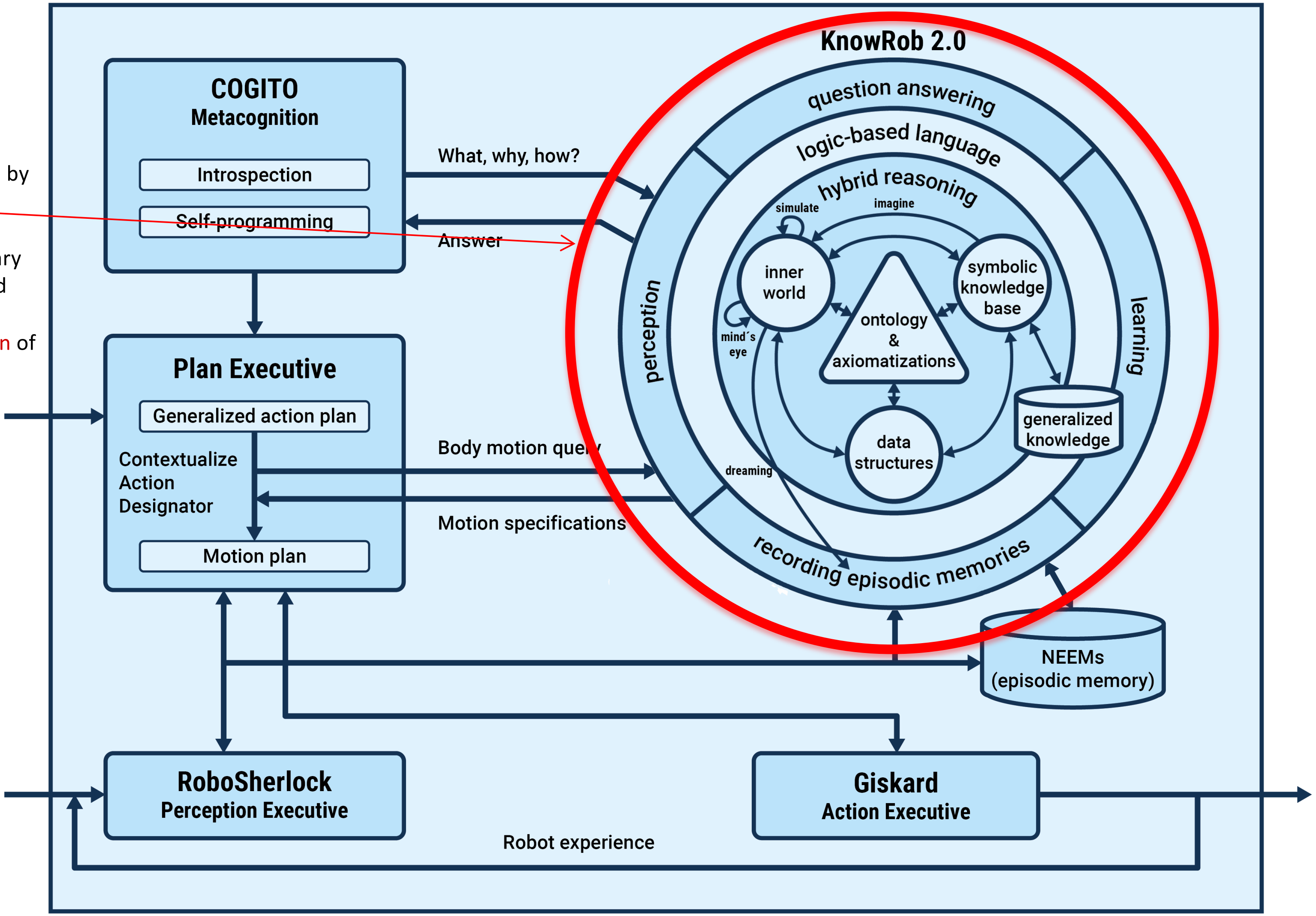



```
query-variables(?loc-for-fetch,
                ?arm, ?grasp, ?lift-pose) to-succeed(
   with-robot-at-location ?loc-for-fetch
     perform (an action
                (type fetching)
                (object ?object)
                (arm ?arm)
                (grasp ?grasp)
                (lift-pose ?lift-pose)
      ...
)
```

The Plan Executive interprets the generalized action plan in process referred to as contextualization in three steps:

The contextualization is accomplished by KnowRob2.0.

The motion parameter values necessary to carry out the action are determined using a **generative model** effectively sampling a joint distribution of the motion parameter values and the associated outcome.

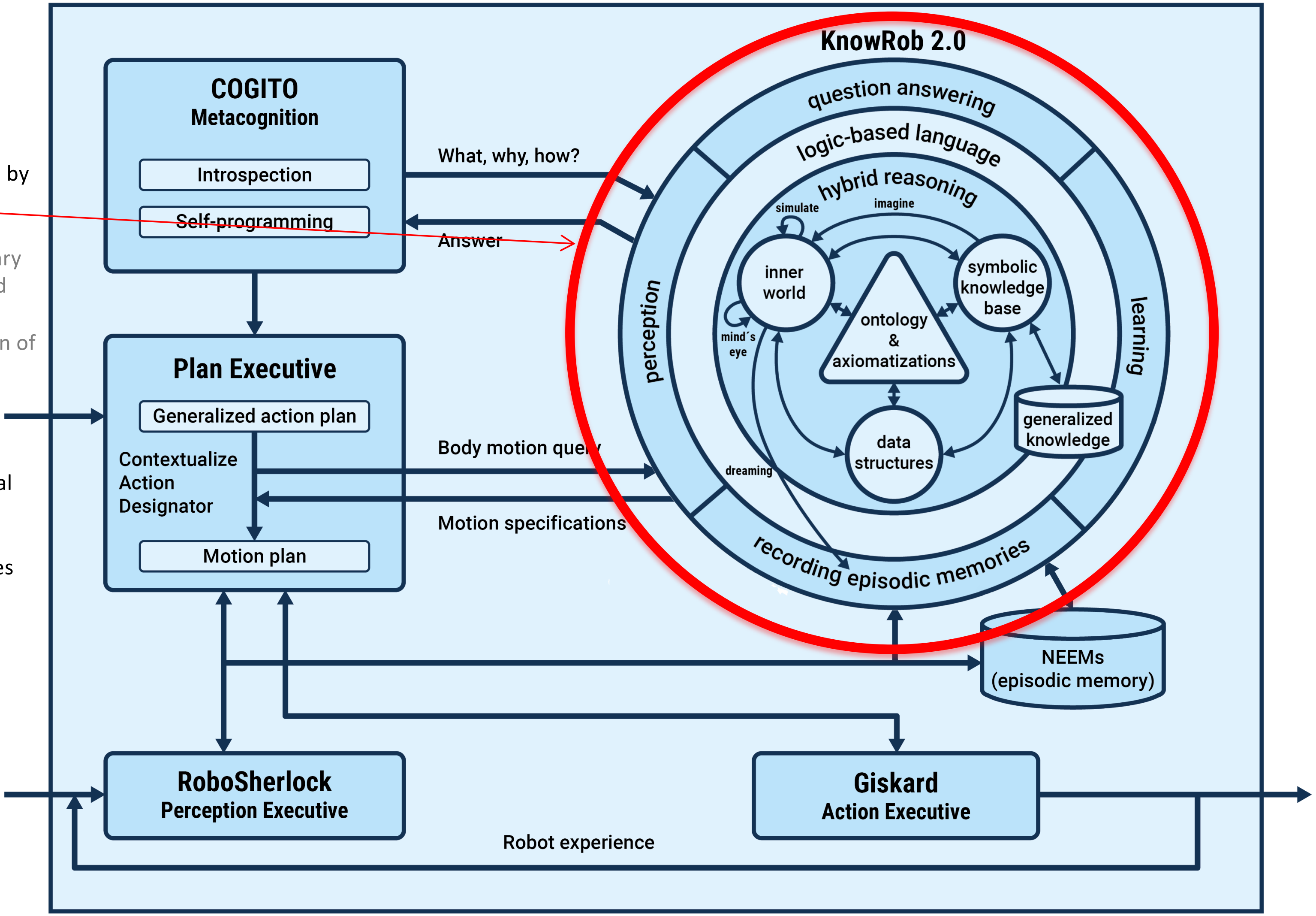


The Plan Executive interprets the generalized action plan in process referred to as contextualization in three steps:

The contextualization is accomplished by KnowRob2.0.

The motion parameter values necessary to carry out the action are determined using a generative model effectively sampling a joint distribution of the motion parameter values and the associated outcome.

It uses knowledge and reasoning, exploiting the constraints of contextual knowledge and current perceptual information, and prospection, to maximize the likelihood that the values identified are most likely to result in a successful action.



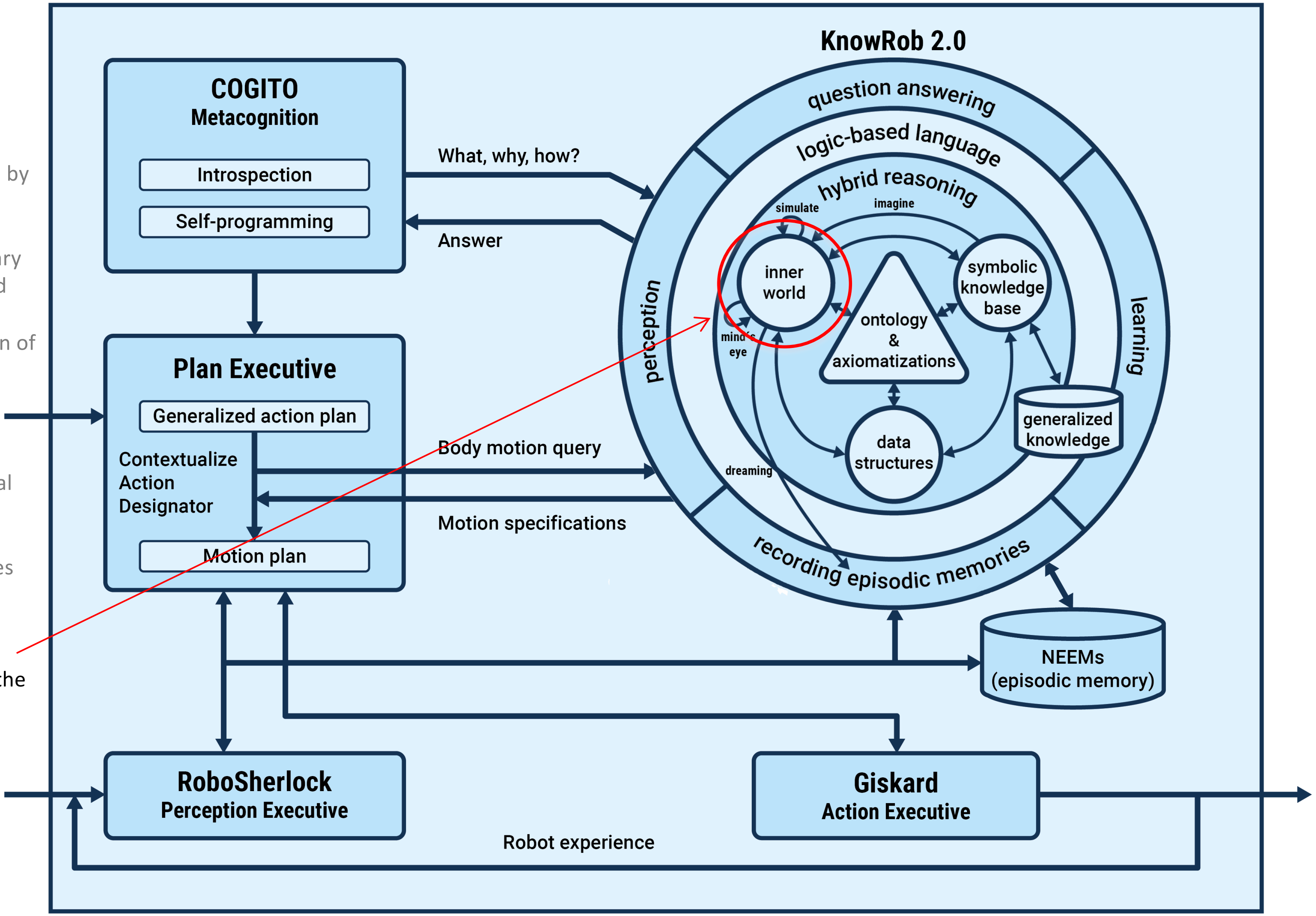
The Plan Executive interprets the generalized action plan in process referred to as contextualization in three steps:

The contextualization is accomplished by KnowRob2.0.

The motion parameter values necessary to carry out the action are determined using a generative model effectively sampling a joint distribution of the motion parameter values and the associated outcome.

It uses knowledge and reasoning, exploiting the constraints of contextual knowledge and current perceptual information, and prospection, to maximize the likelihood that the values identified are most likely to result in a successful action.

It accomplishes prospection by using the robot's inner world to simulate plan execution



The Plan Executive interprets the generalized action plan in process referred to as contextualization in three steps:

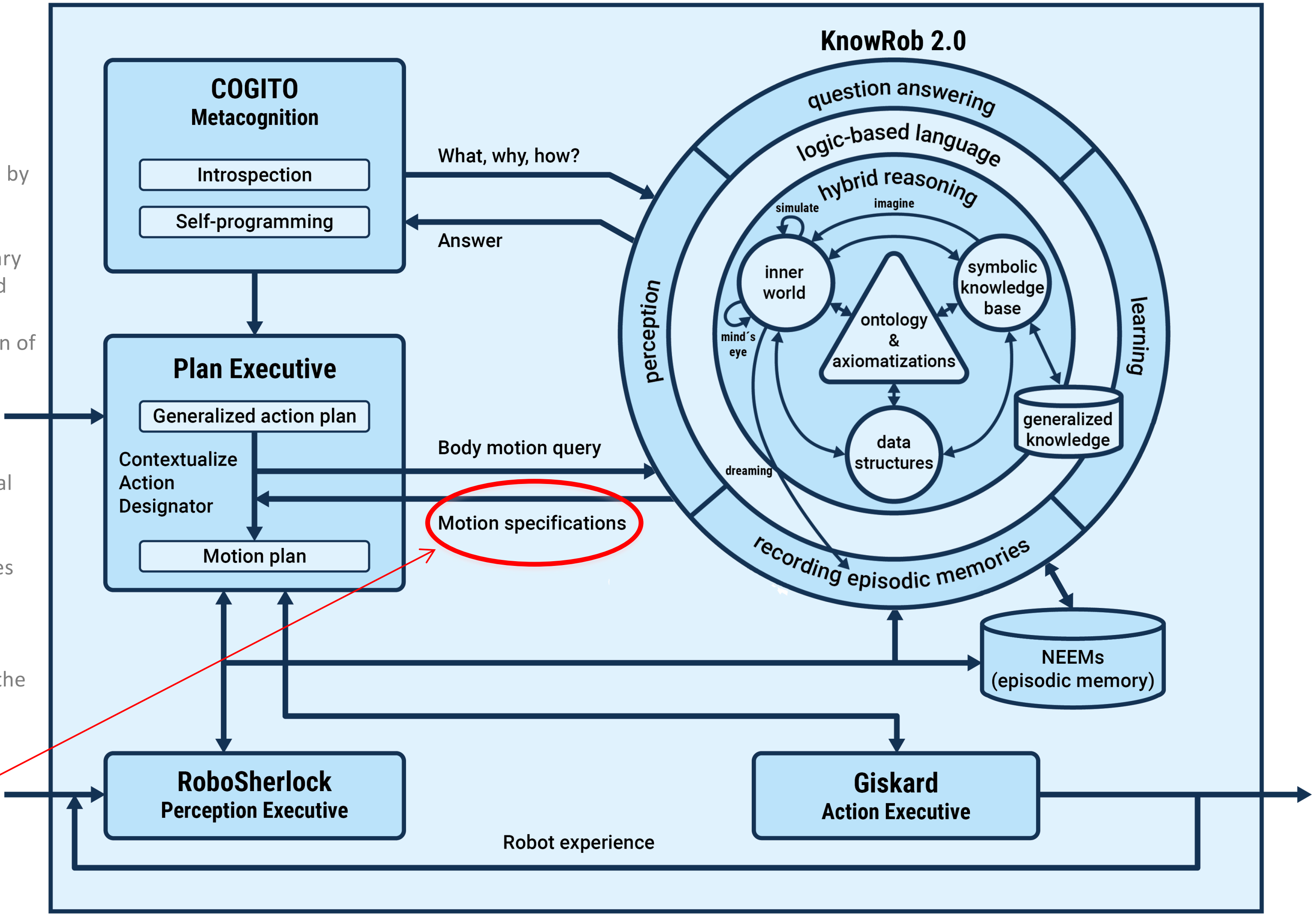
The contextualization is accomplished by KnowRob2.0.

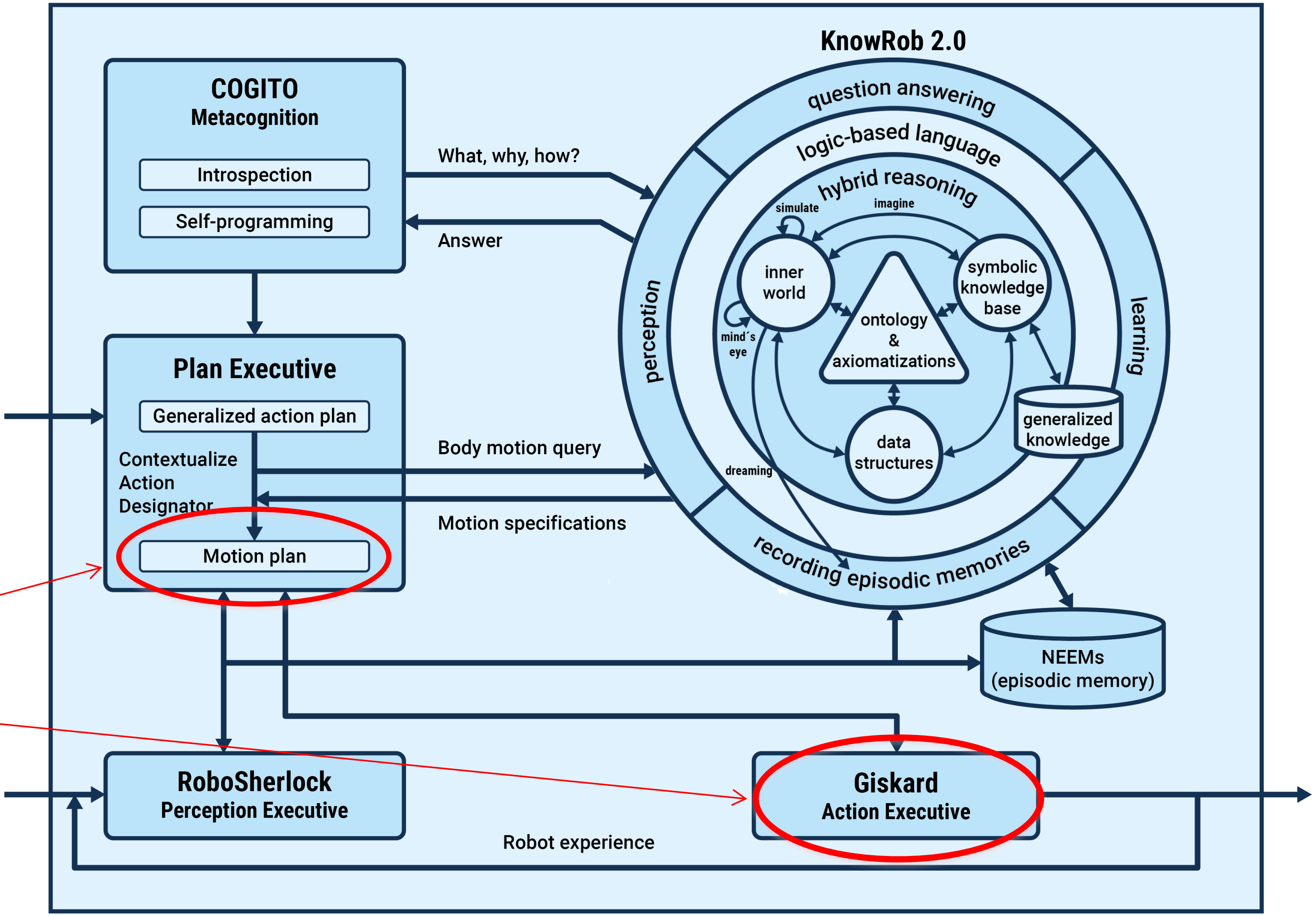
The motion parameter values necessary to carry out the action are determined using a generative model effectively sampling a joint distribution of the motion parameter values and the associated outcome.

It uses knowledge and reasoning, exploiting the constraints of contextual knowledge and current perceptual information, and prospection, to maximize the likelihood that the values identified are most likely to result in a successful action.

It accomplishes prospection by using the robot's inner world to simulate plan execution

The motion parameter values are returned to the Plan Executive





The motion plan is then executed by the Action Executive

Reading

D. Vernon, "Cognitive Architectures", in Cognitive Robotics, A. Cangelosi and M. Asada (Eds.), MIT Press, Chapter 10, 2022, Section 10.6.1.

Further Reading

D. Vernon, J. Albert, M. Beetz, S.-C. Chiou, H. Ritter, and W. X. Schneider, "Action Selection and Execution in Everyday Activities: A Cognitive Robotics & Situation Model Perspective", Topics in Cognitive Science, pp. 1-19, 2021.