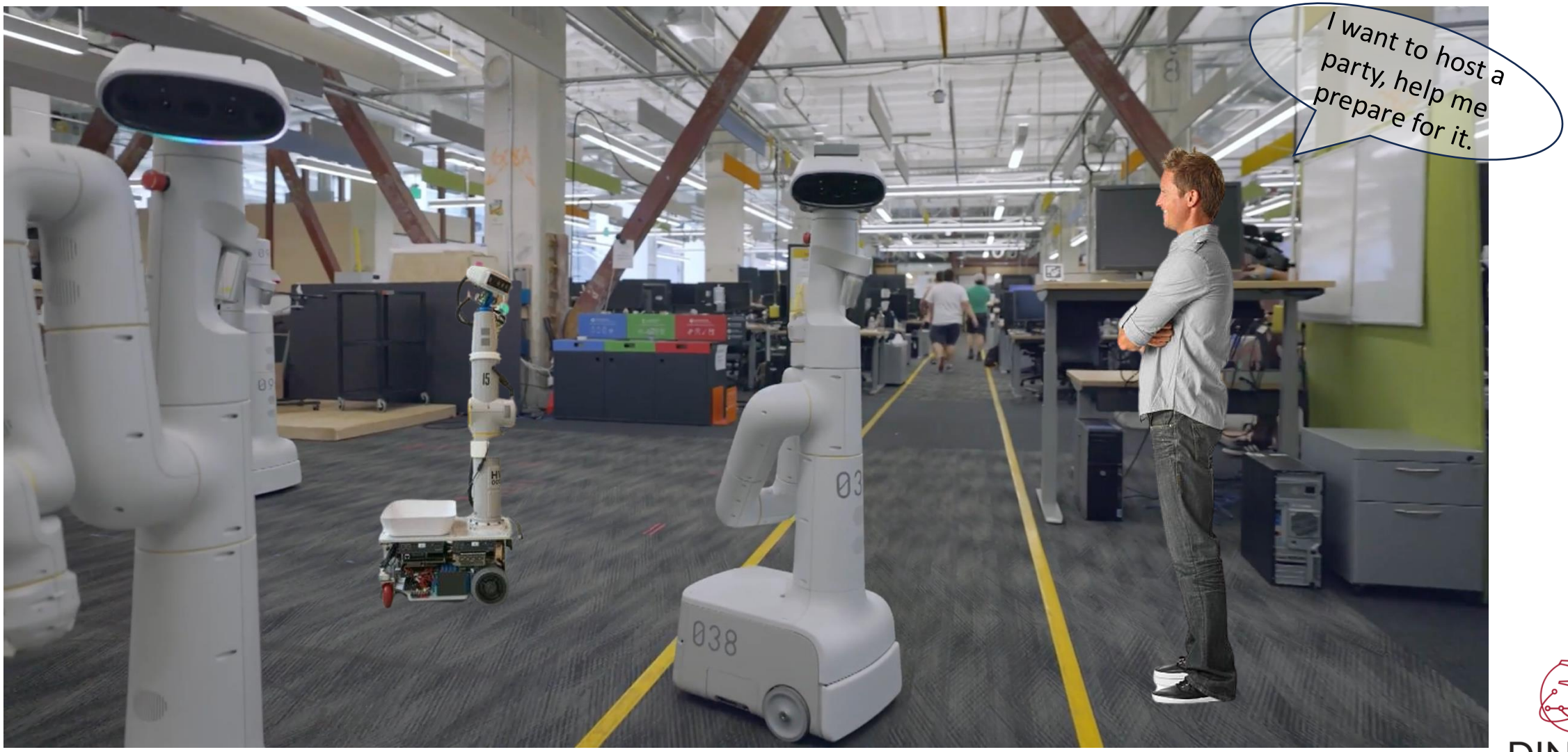
LLaMAR: Long-Horizon Planning for Multi-Agent Robotics in Partially Observable Environments

Siddharth Nayak*, Adelmo Orozco*, Marina Have, Vittal Thirumalai, Jackson Zhang, Darren Chen, Aditya Kapoor, Eric Robinson, Karthik Gopalakrishnan, James Harrison, Brian Ichter, Anuj Mahajan, Hamsa Balakrishnan

Motivation

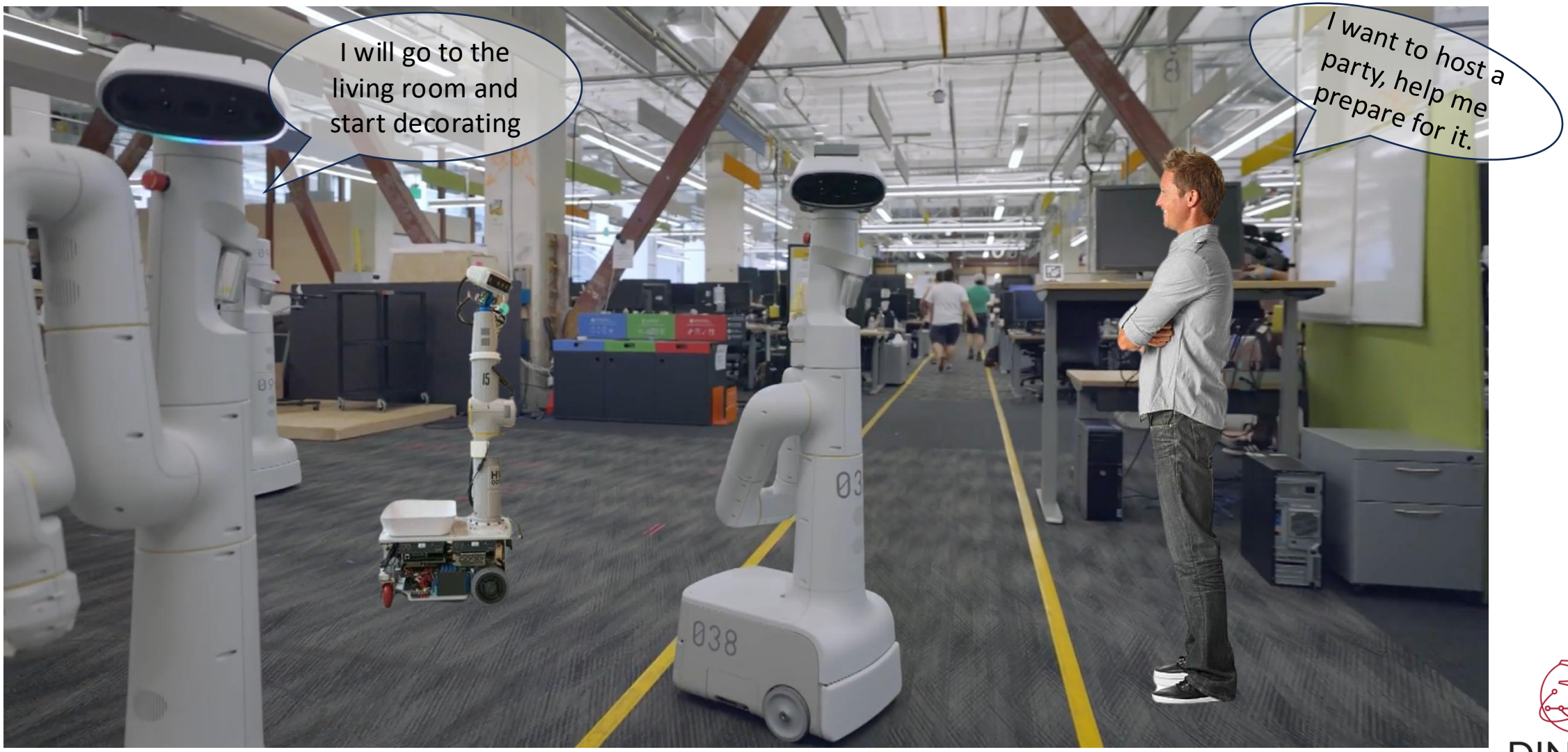
DINaMo

Motivation



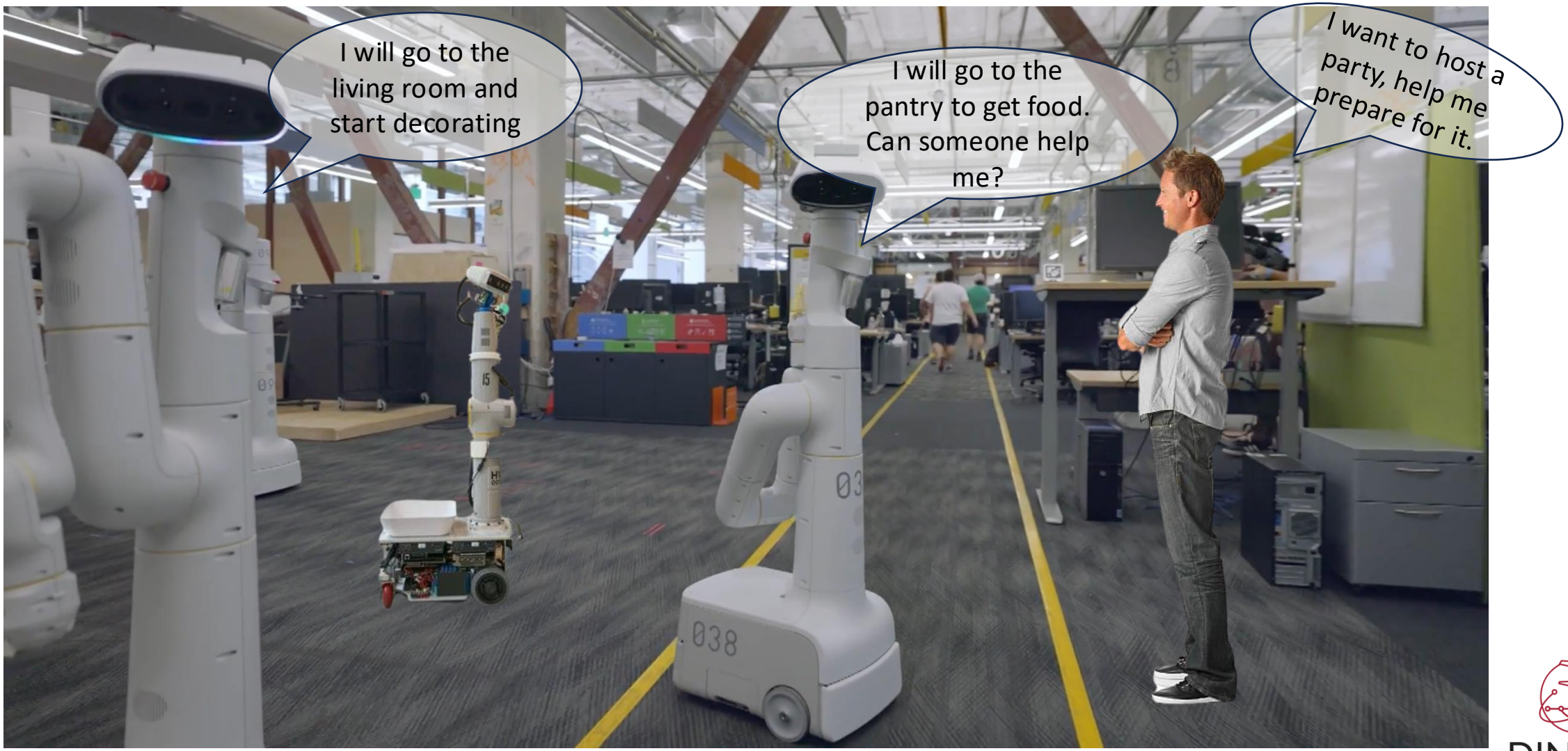
DINaMo

Motivation



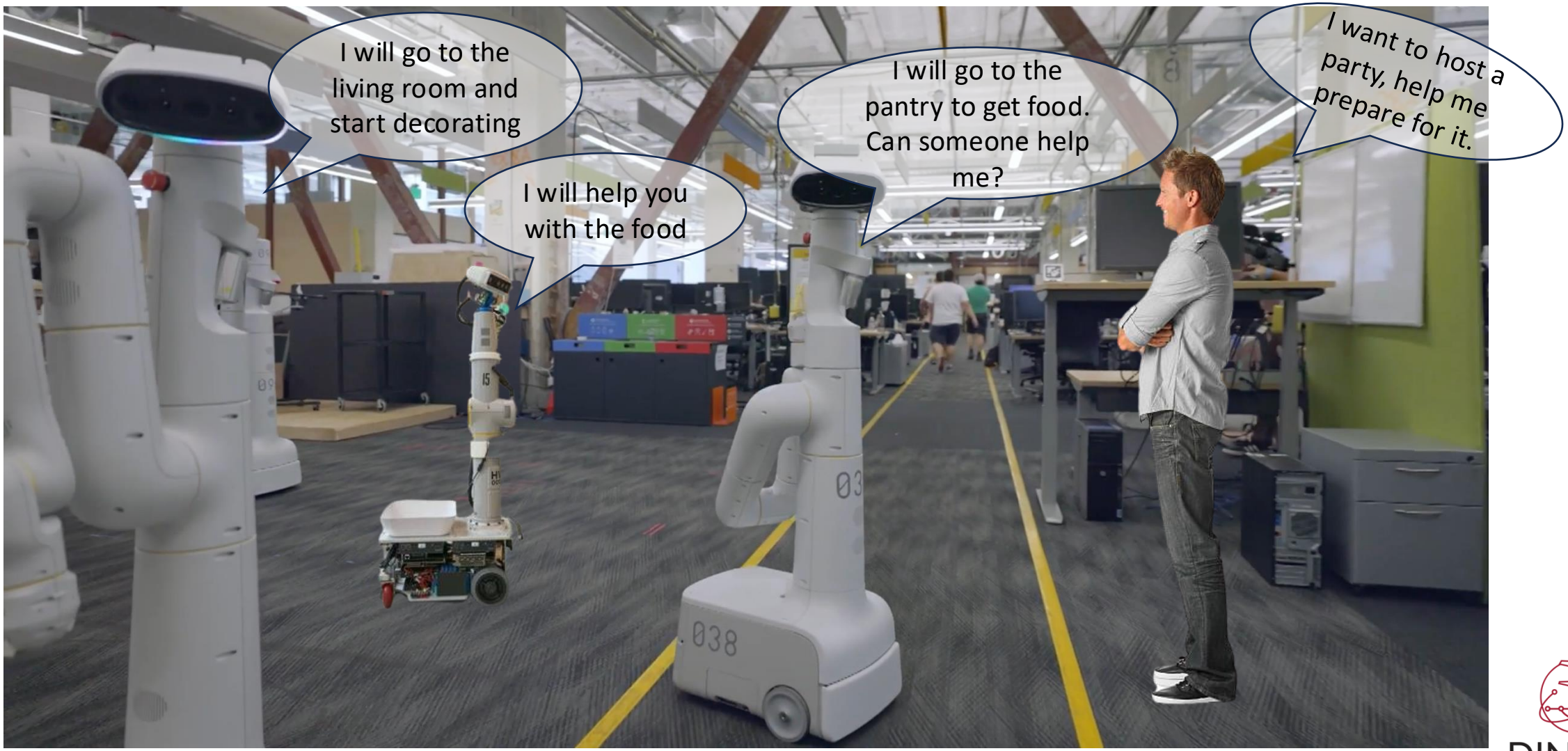
DINaMo

Motivation



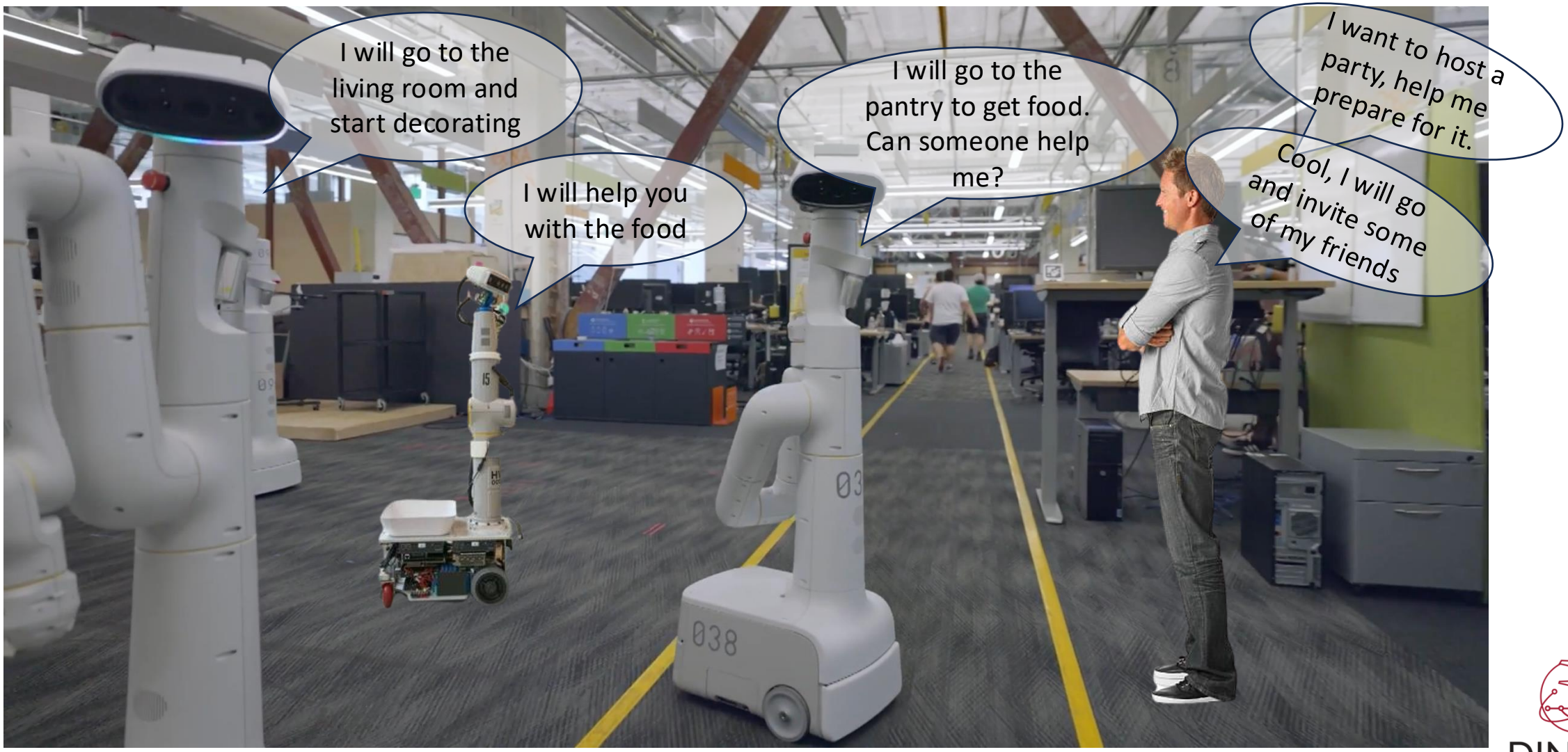
DINaMo

Motivation



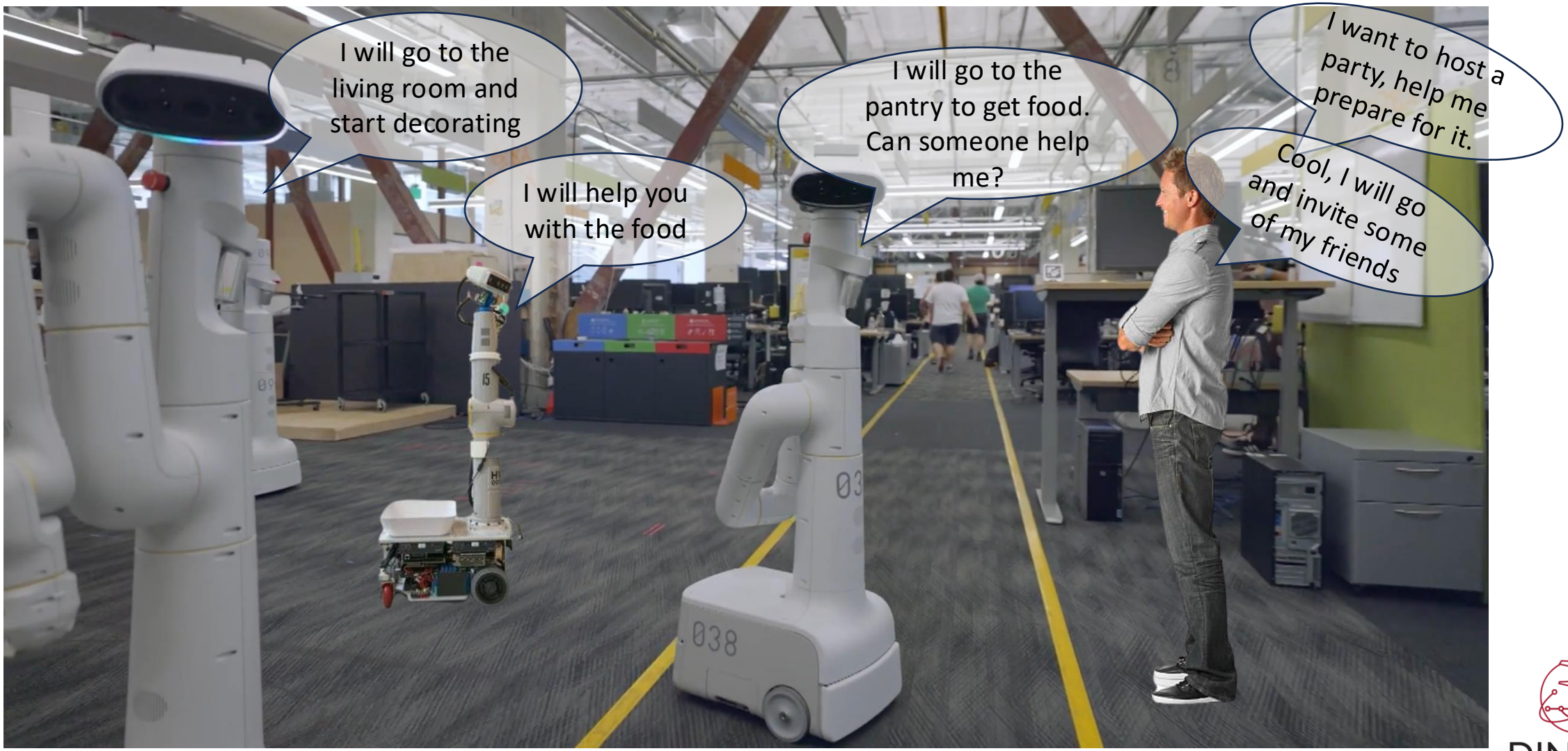
DINaMo

Motivation



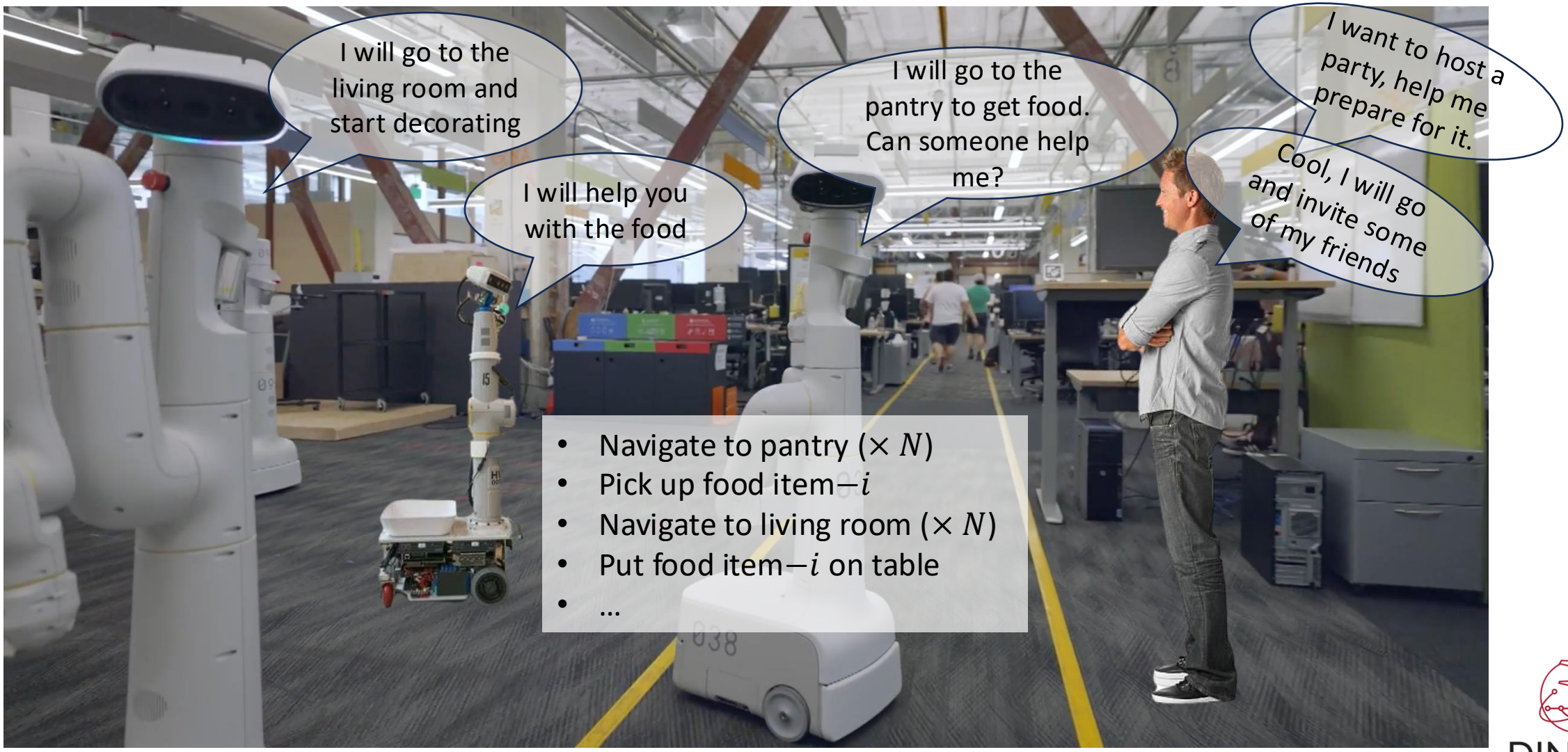
DINaMo

Long-Horizon Planning

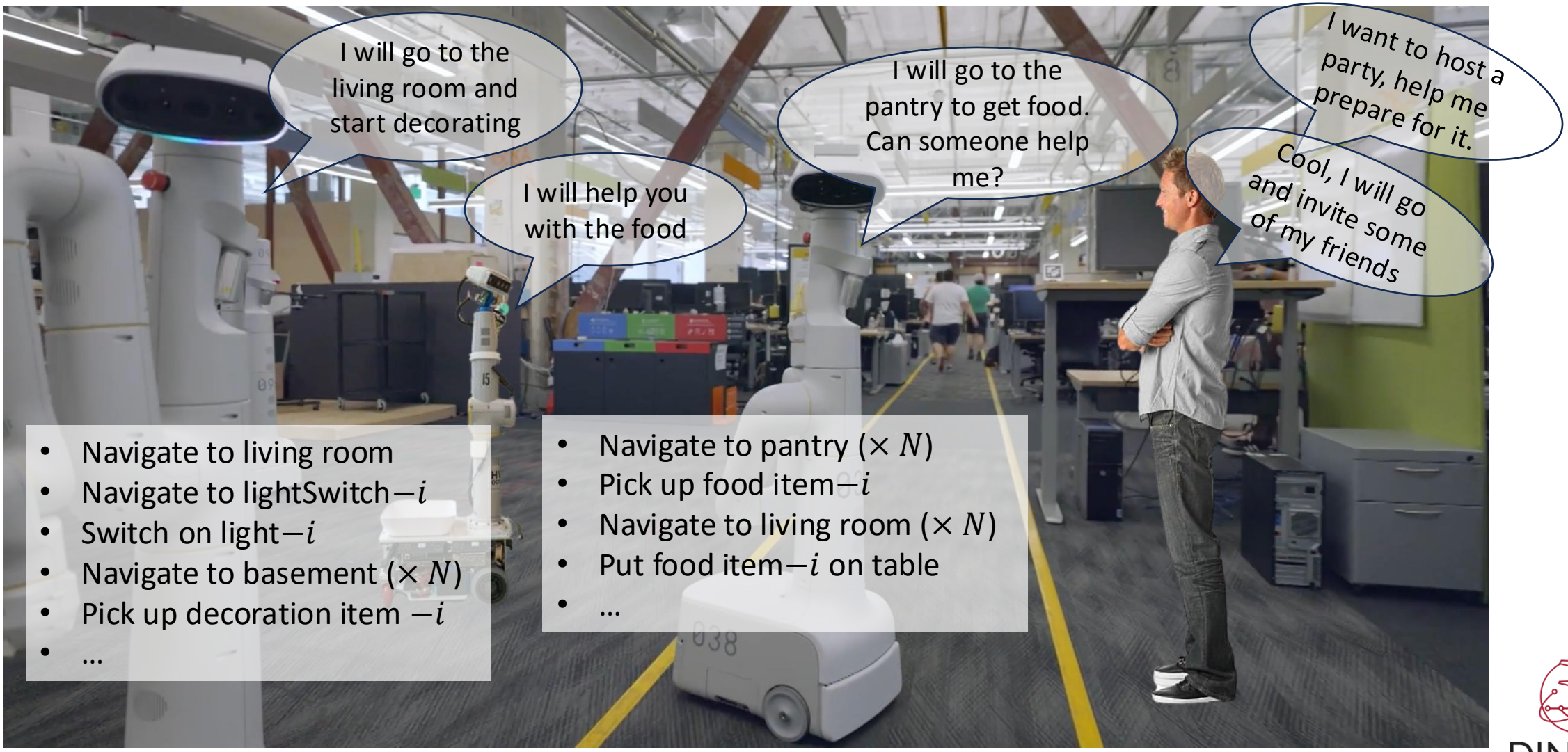


DINaMo

Long-Horizon Planning

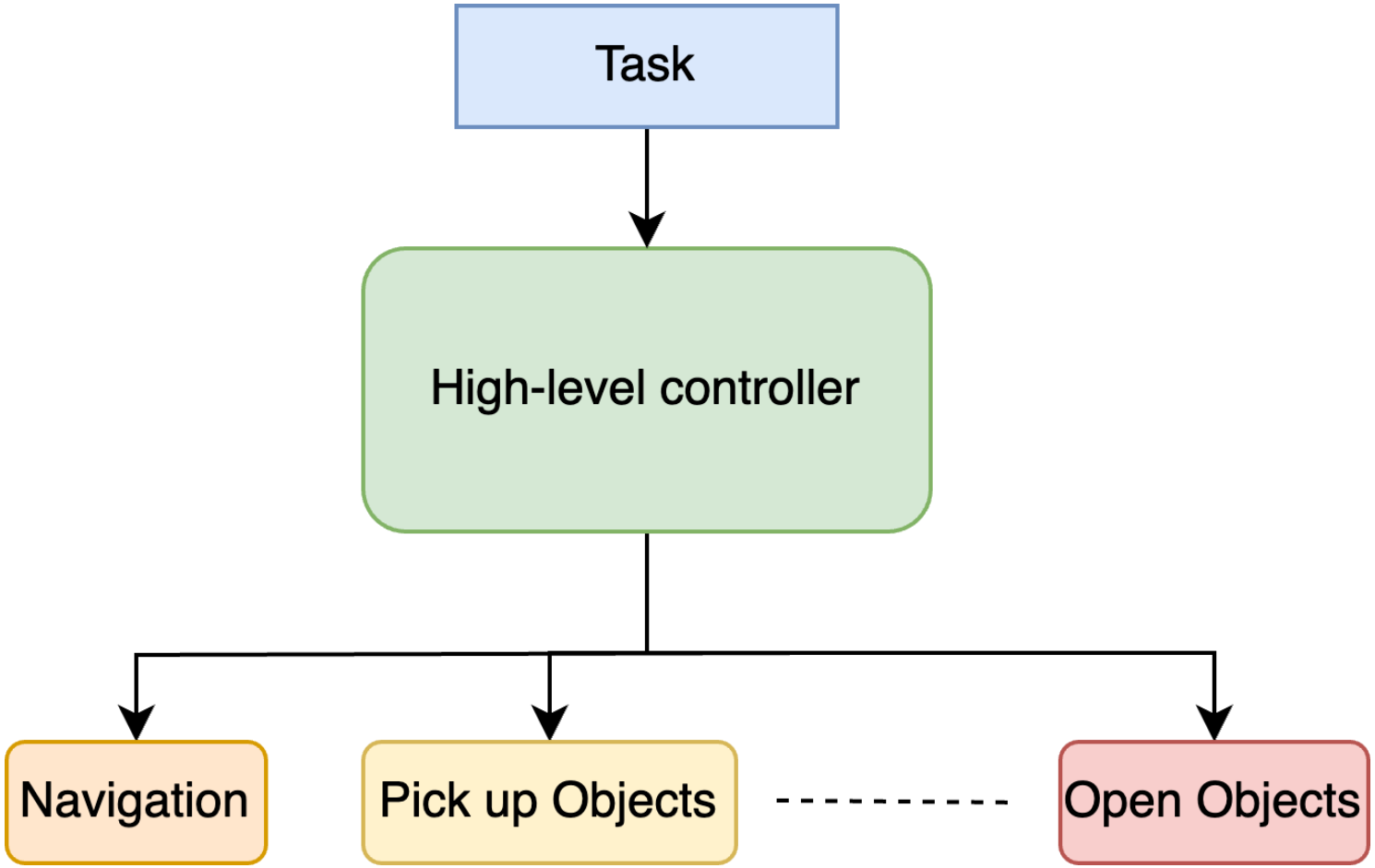


Long-Horizon Planning

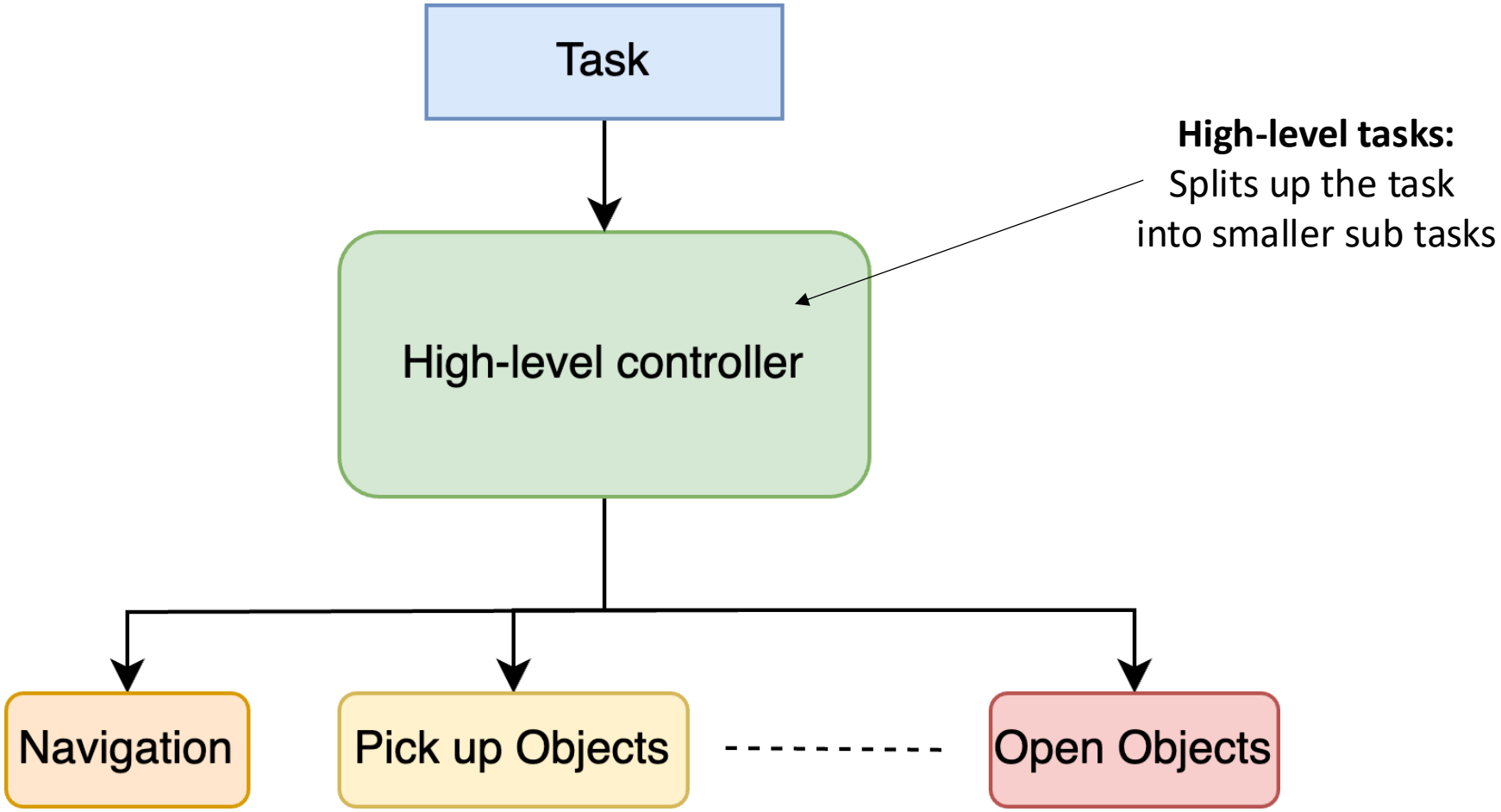


DINaMo

Hierarchical Planning

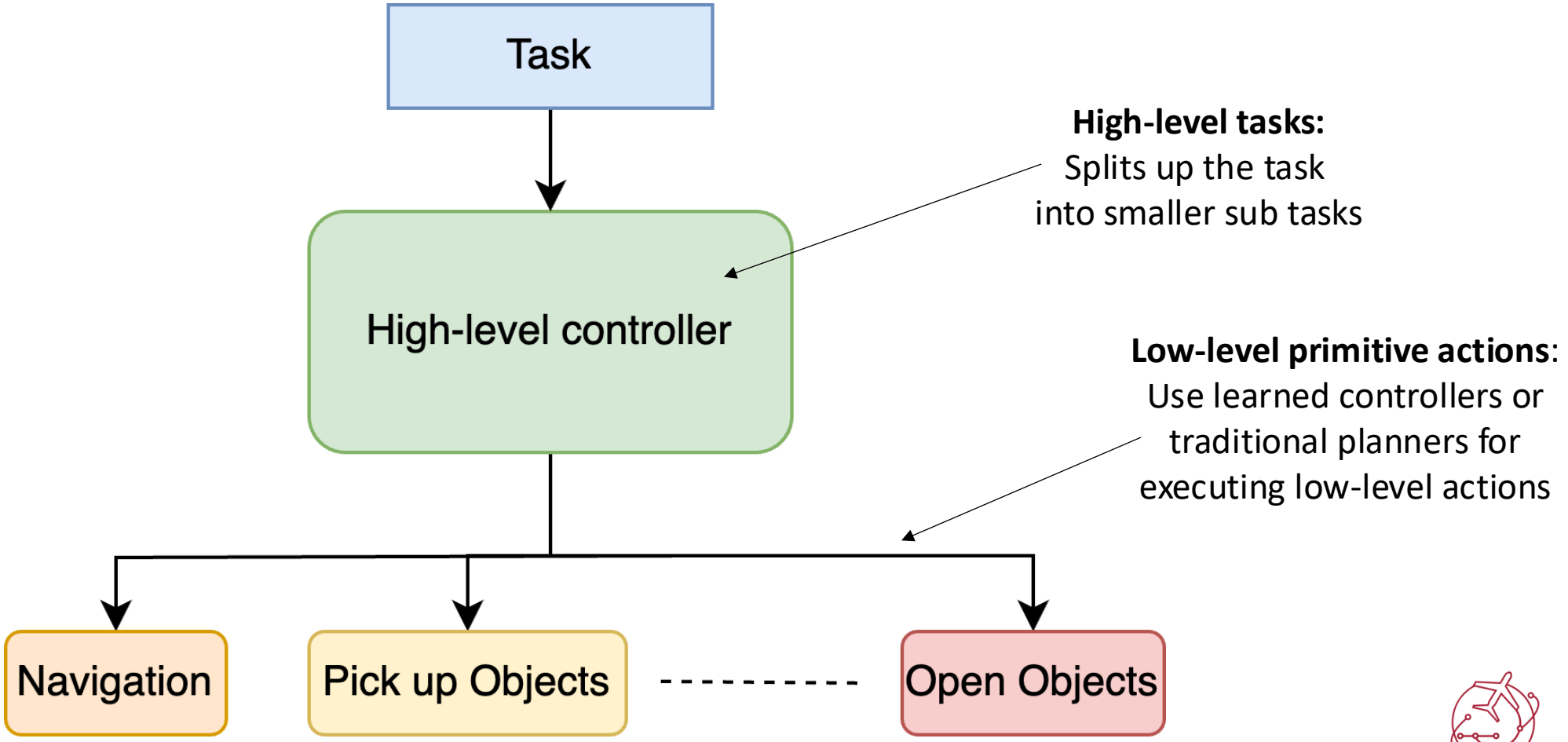


Hierarchical Planning

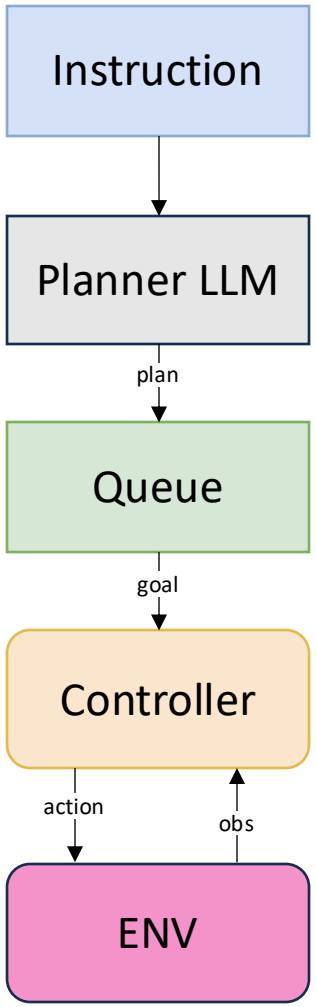


High-level tasks:
Splits up the task
into smaller sub tasks

Hierarchical Planning

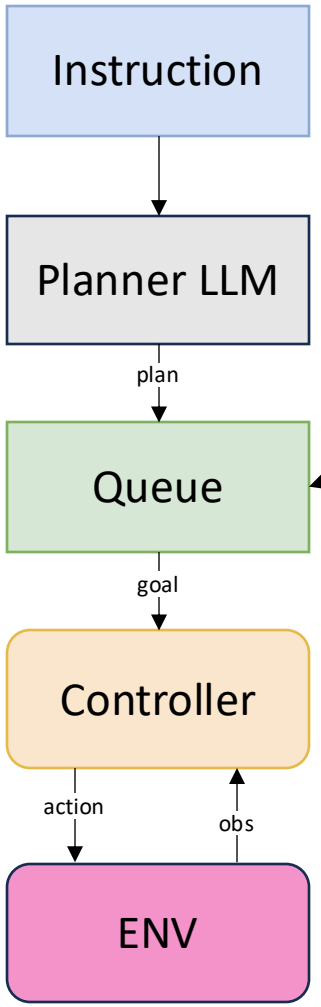


LLMs for planning



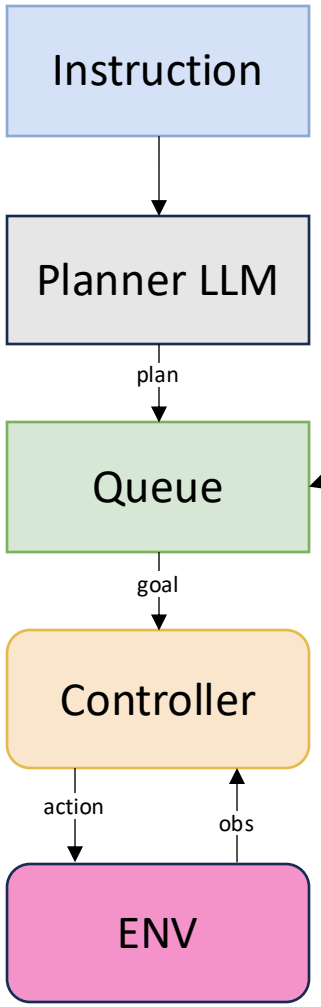
DINaMo

LLMs for planning



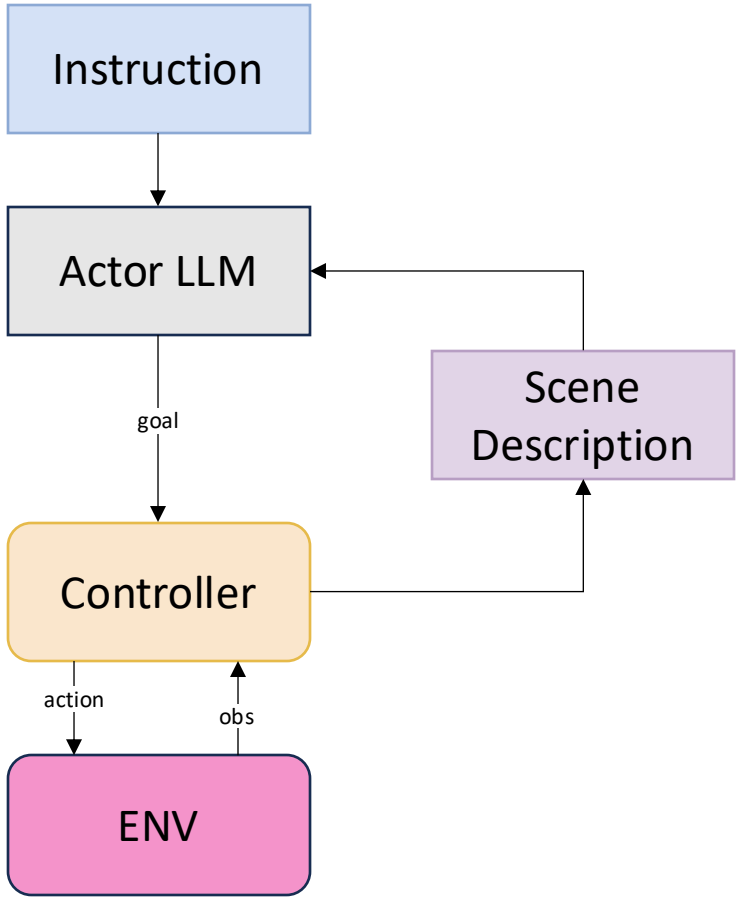
- Does not accommodate for uncertainties

LLMs for planning

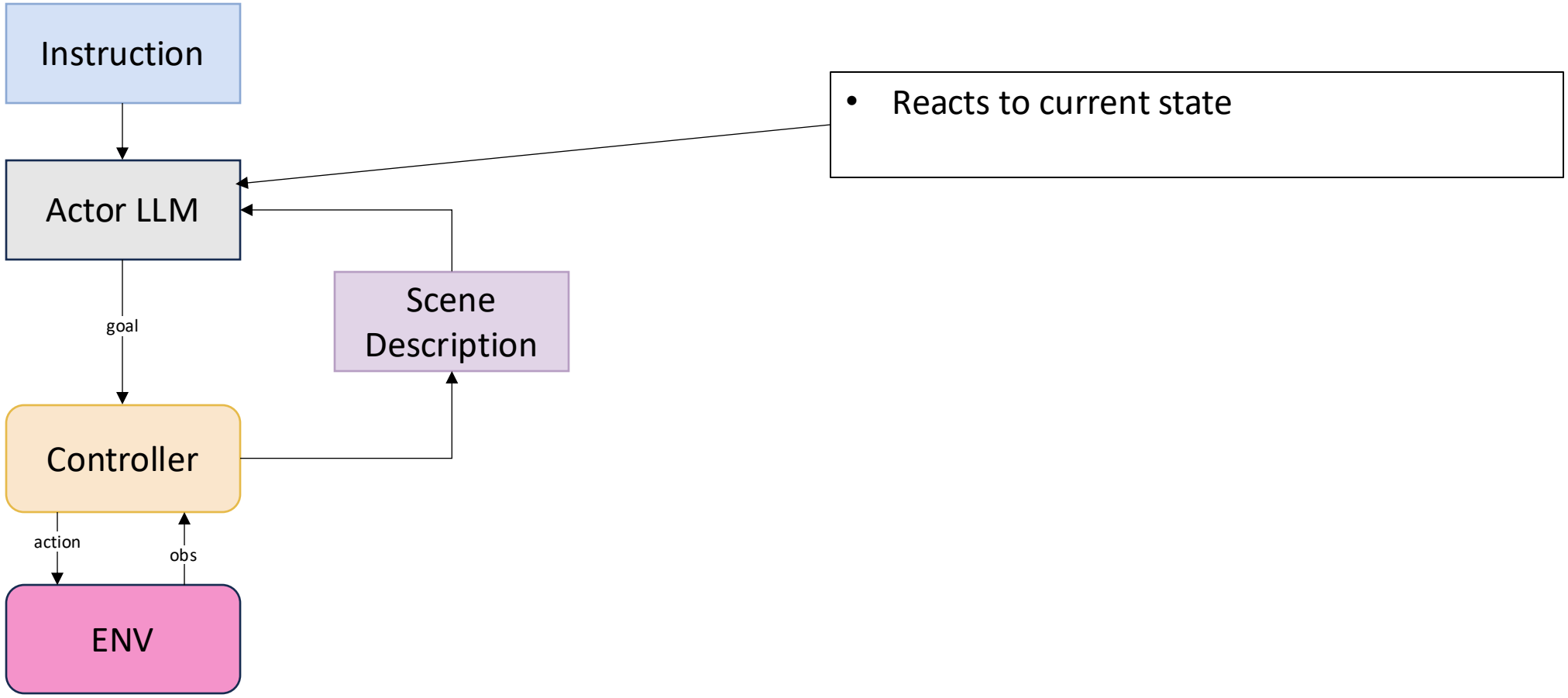


- Does not accommodate for uncertainties
- Requires prior knowledge about subtasks (global information)

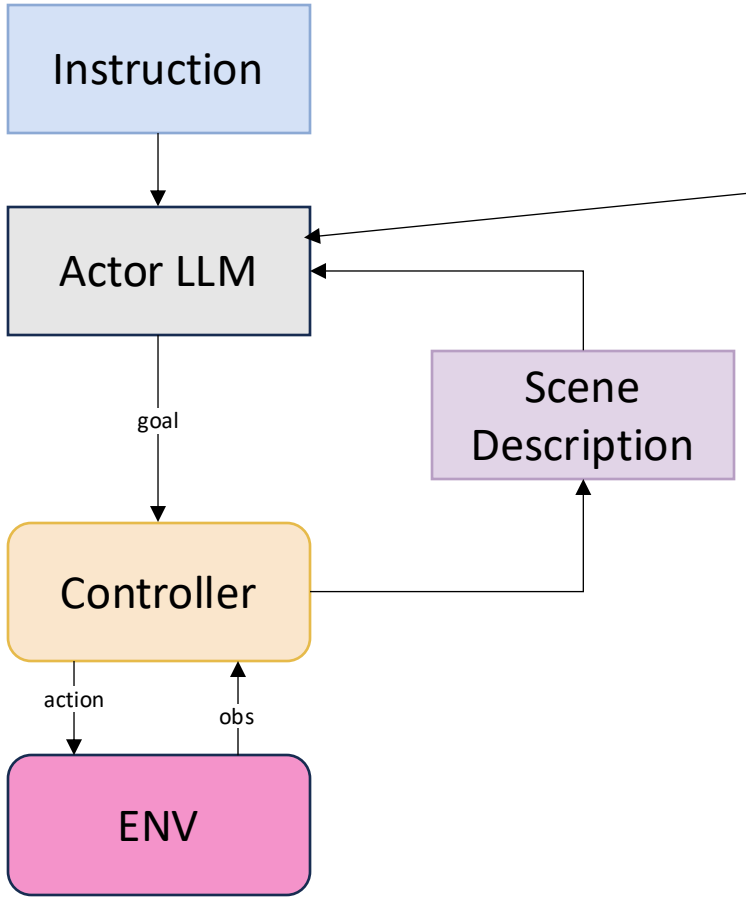
LLMs for Reactive Planning



LLMs for Reactive Planning

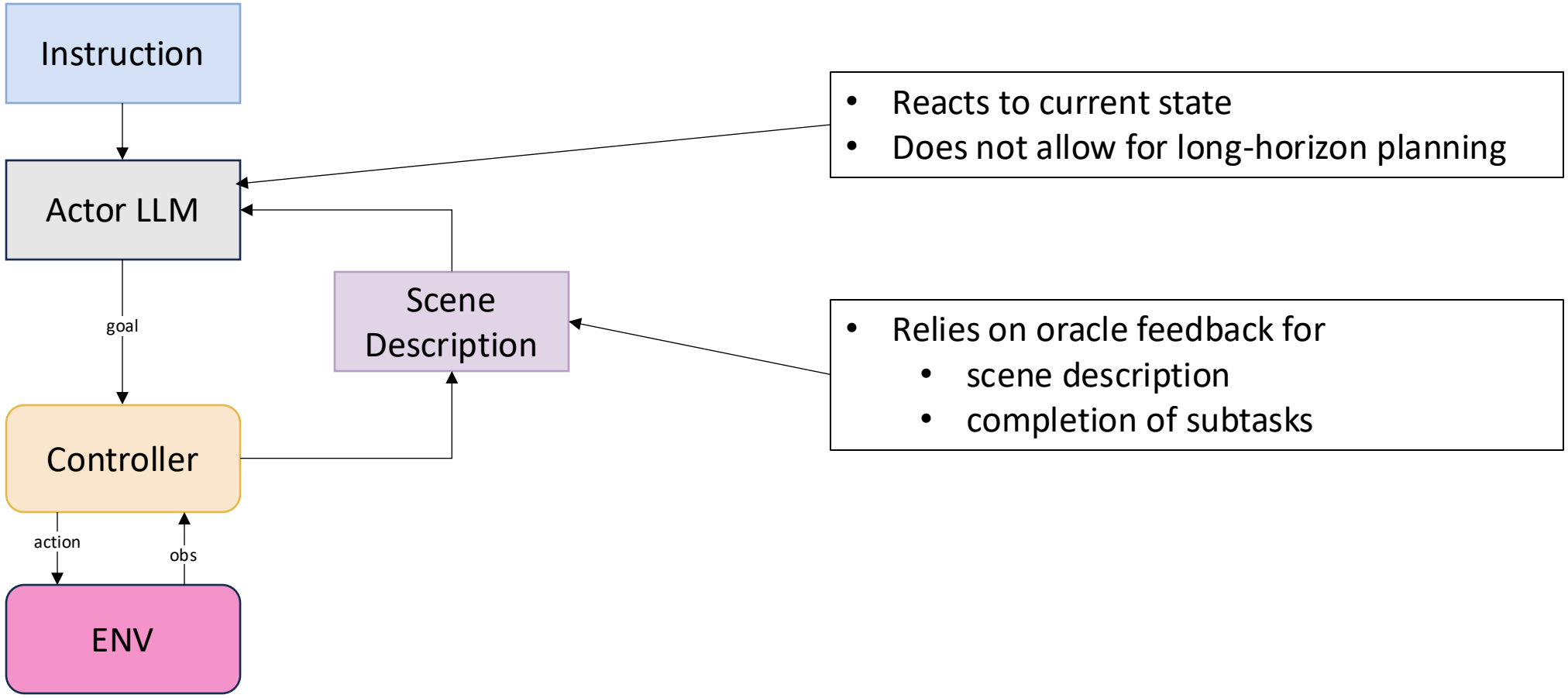


LLMs for Reactive Planning

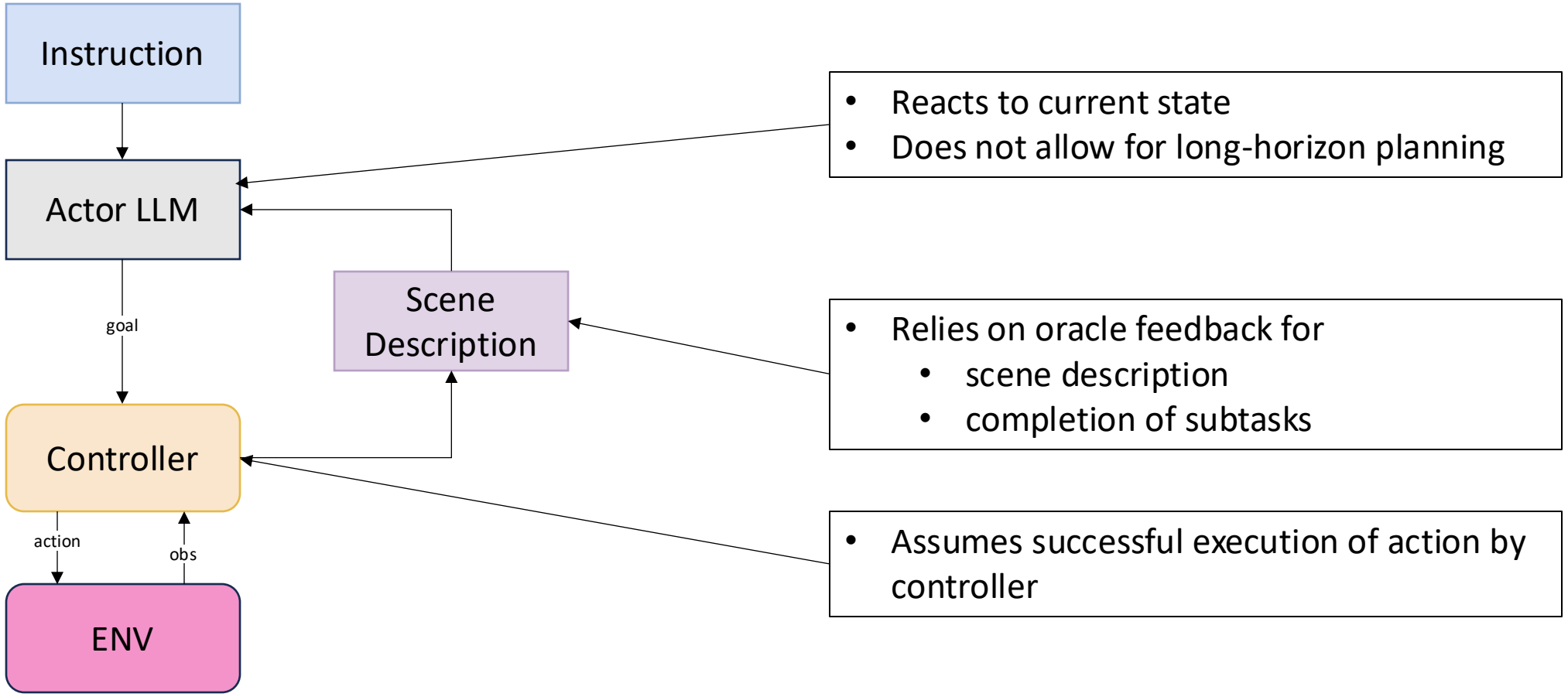


- Reacts to current state
- Does not allow for long-horizon planning

LLMs for Reactive Planning



LLMs for Reactive Planning



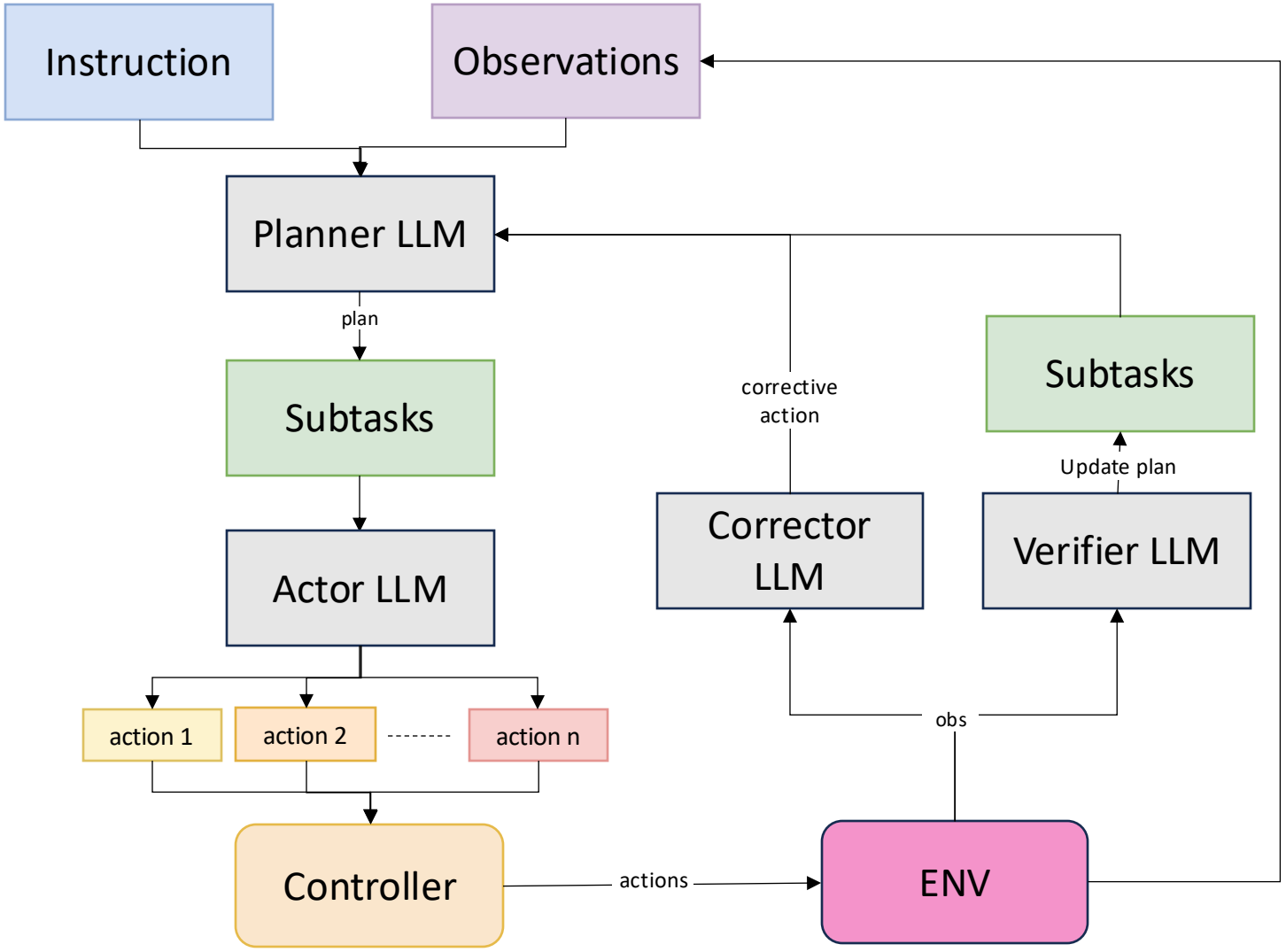
- Reacts to current state
- Does not allow for long-horizon planning

- Relies on oracle feedback for
  - scene description
  - completion of subtasks

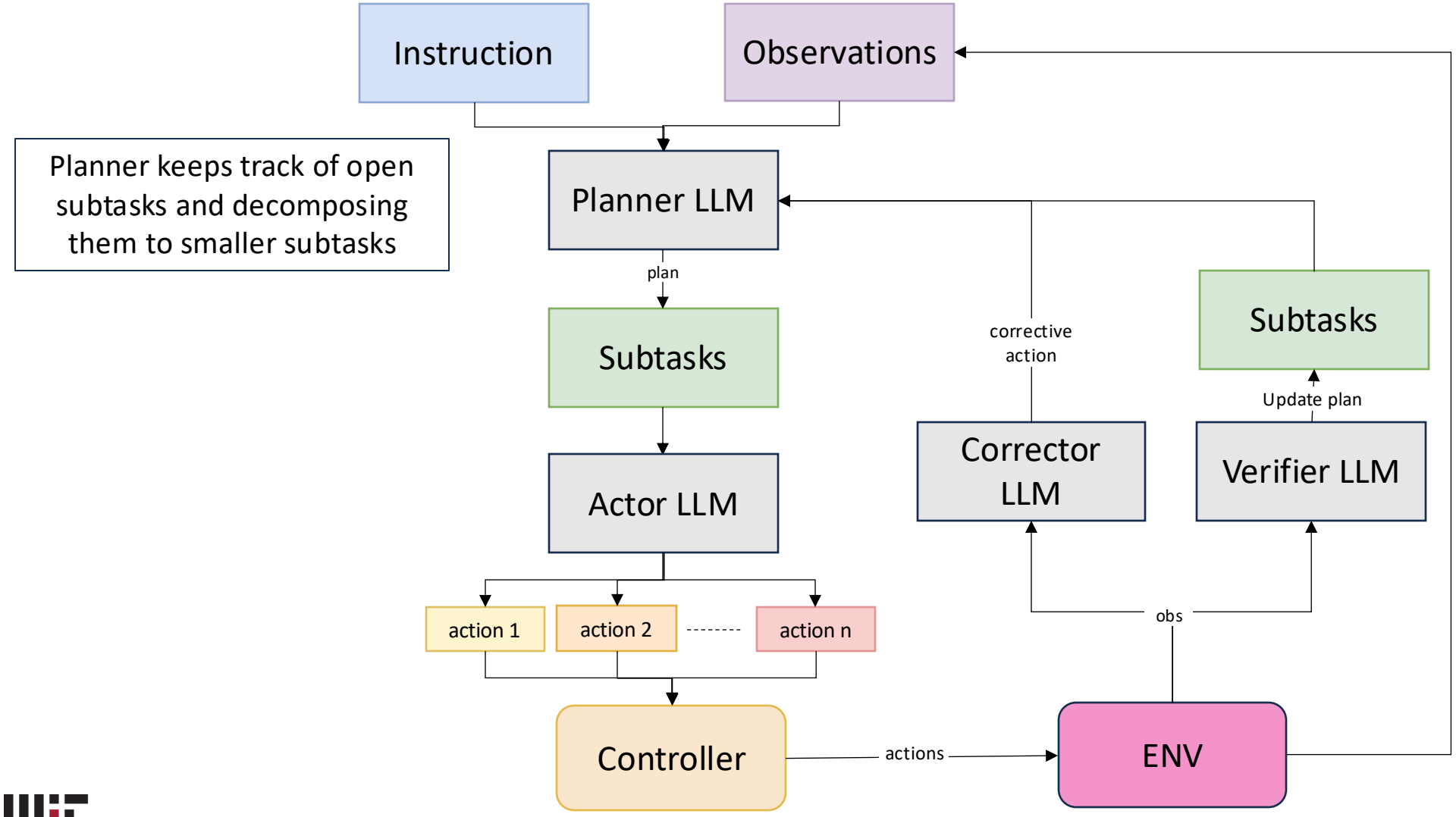
- Assumes successful execution of action by controller

DINaMo

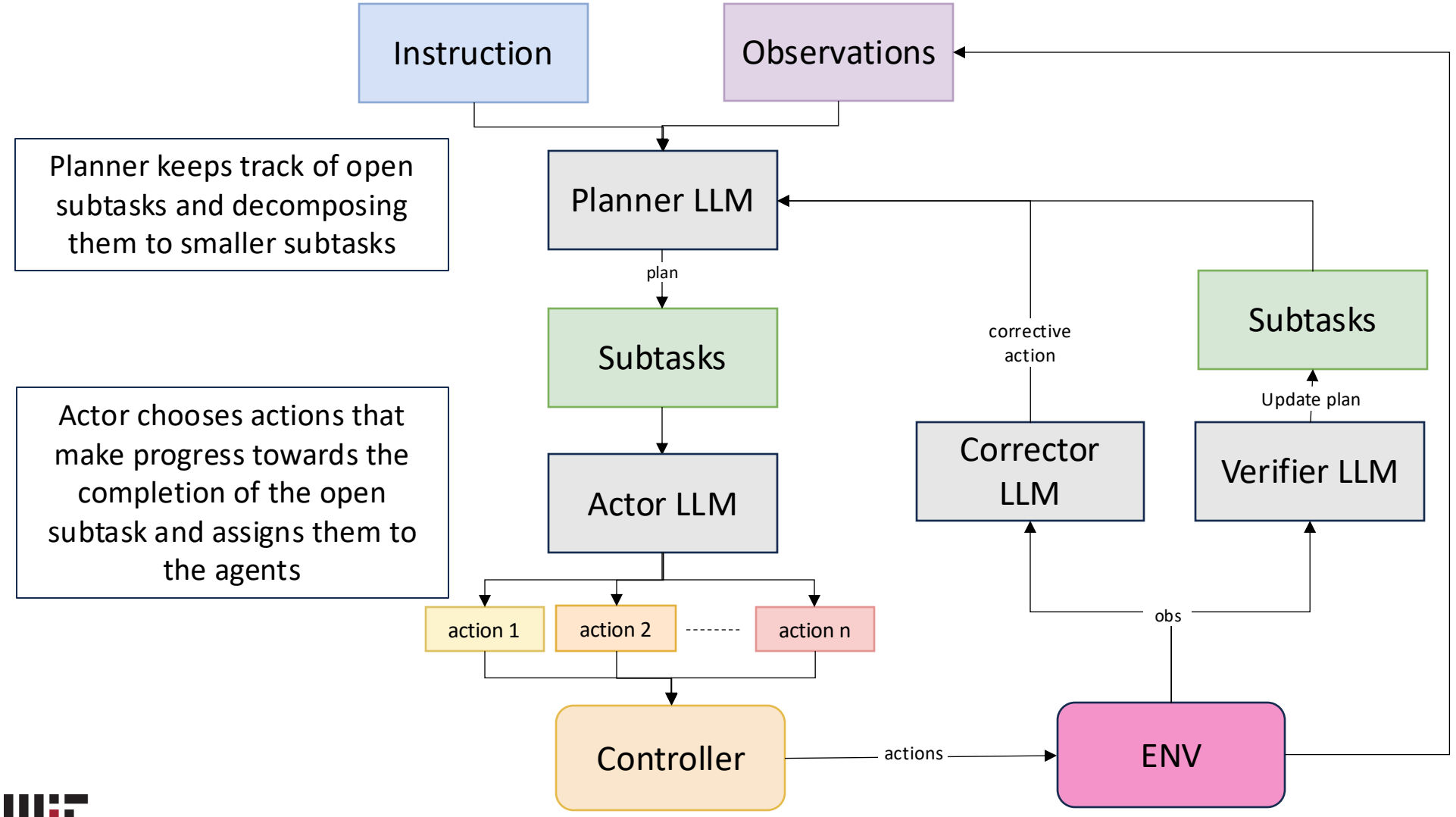
LLaMAR


Instruction
Observations
Planner LLM
plan
Subtasks
Actor LLM
action 1
action 2
action n
Controller
actions
ENV
obs
corrective action
Corrector LLM
Subtasks
Update plan
Verifier LLM

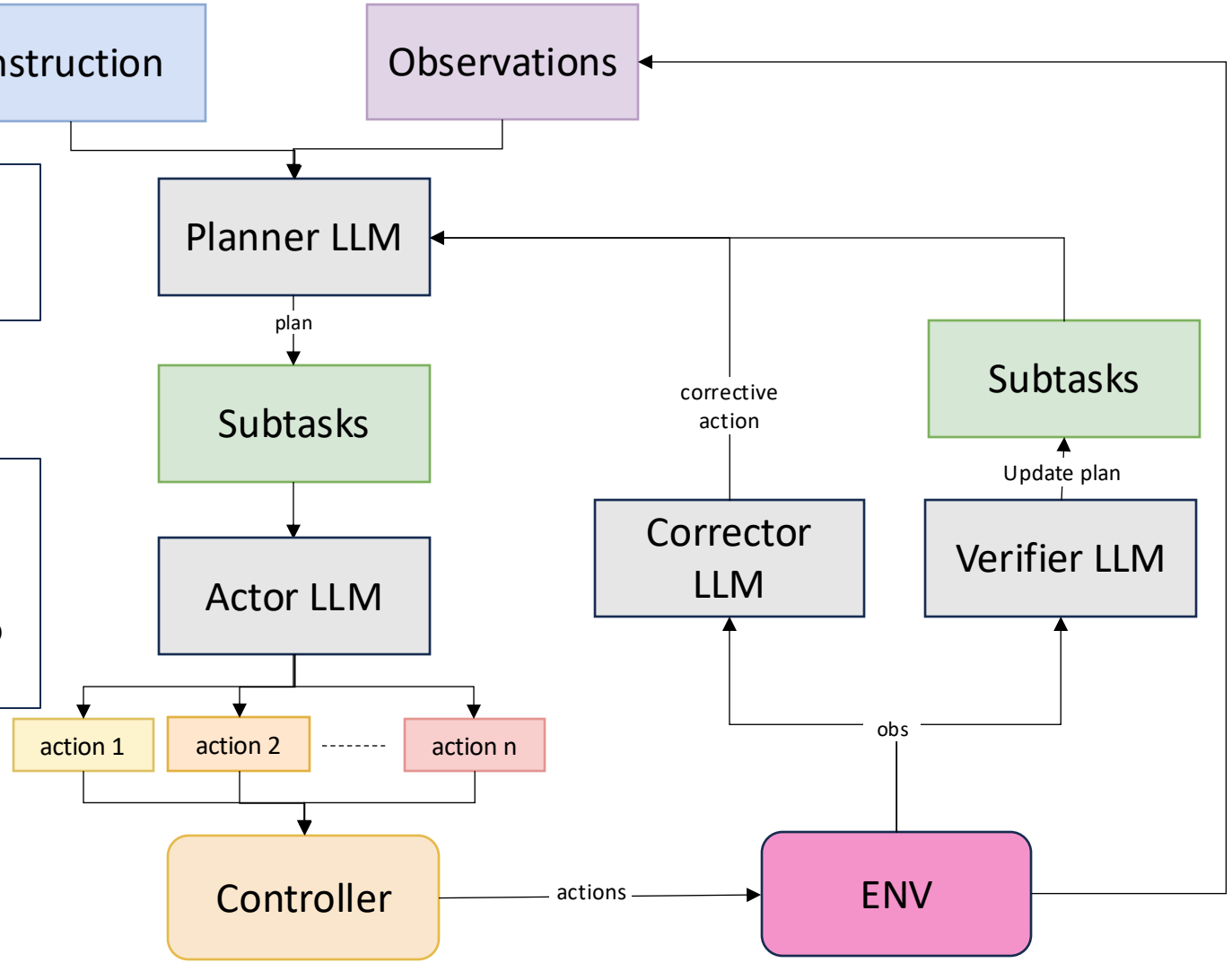
LLaMAR


Instruction
Observations
Planner keeps track of open subtasks and decomposing them to smaller subtasks
Planner LLM
plan
Subtasks
Actor LLM
action 1
action 2
action n
Controller
actions
ENV
obs
corrective action
Corrector LLM
Subtasks
Update plan
Verifier LLM

LLaMAR


Instruction
Observations
Planner keeps track of open subtasks and decomposing them to smaller subtasks
Planner LLM
plan
Subtasks
Actor chooses actions that make progress towards the completion of the open subtask and assigns them to the agents
Actor LLM
action 1
action 2
action n
Controller
actions
ENV
corrective action
Corrector LLM
Subtasks
Update plan
Verifier LLM
obs

LLaMAR

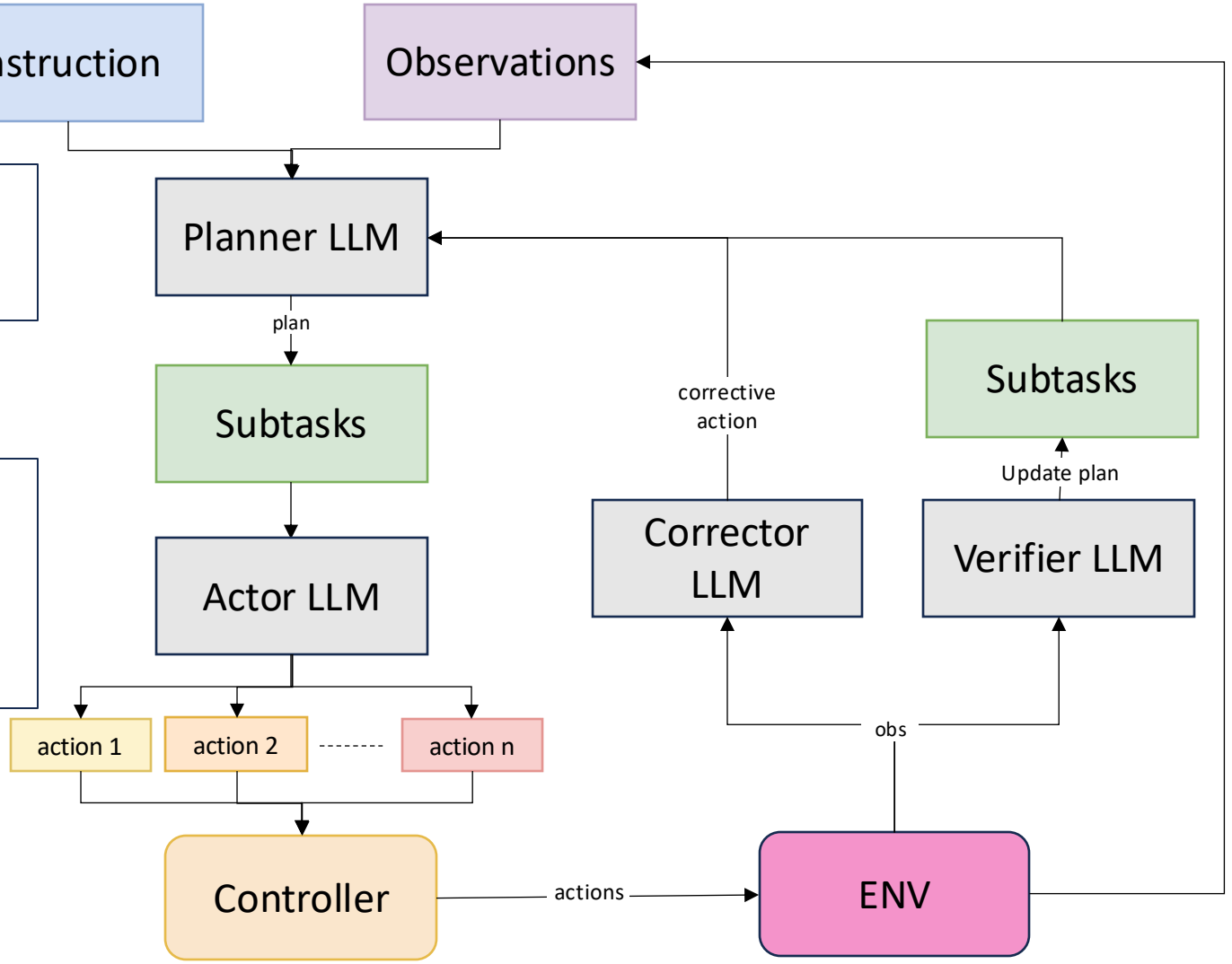


Planner keeps track of open subtasks and decomposing them to smaller subtasks

Actor chooses actions that make progress towards the completion of the open subtask and assigns them to the agents

Verifier keeps track of completed subtasks allowing us to not rely on an oracle for feedback

LLaMAR



Planner keeps track of open subtasks and decomposing them to smaller subtasks

Actor chooses actions that make progress towards the completion of the open subtask and assigns them to the agents

Corrector reasons on why a particular action failed and suggests a corrective action

Verifier keeps track of completed subtasks allowing us to not rely on an oracle for feedback

DINaMo

MAP-THOR

Tasks

MAP-THOR

45 tasks
each with 5 seeds

Tasks

MAP-THOR

45 tasks
each with 5 seeds

Tasks

Categorized
based on
difficulty of tasks

- Explicit item type, quantity and target
  E.g., Put bread, lettuce, tomato in the fridge
- Explicit item type and target, implicit quantity
  E.g., Put all the apples in the fridge
- Explicit target, implicit item type and quantity
  E.g., Put all groceries in the fridge
- Implicit item type, quantity and target
  E.g., Clear the floor by placing the
  items at their appropriate positions

DINaMo

MAP-THOR

45 tasks
each with 5 seeds

Tasks

Metrics

Categorized based on difficulty of tasks

- Explicit item type, quantity and target
  E.g., Put bread, lettuce, tomato in the fridge
- Explicit item type and target, implicit quantity
  E.g., Put all the apples in the fridge
- Explicit target, implicit item type and quantity
  E.g., Put all groceries in the fridge
- Implicit item type, quantity and target
  E.g., Clear the floor by placing the items at their appropriate positions

DINaMo

MAP-THOR

45 tasks
each with 5 seeds

Tasks

Metrics

Benchmark

Categorized
based on
difficulty of tasks

Explicit item type, quantity and target E.g., Put bread, lettuce, tomato in the fridge
Explicit item type and target, implicit quantity E.g., Put all the apples in the fridge
Explicit target, implicit item type and quantity E.g., Put all groceries in the fridge
Implicit item type, quantity and target E.g., Clear the floor by placing the items at their appropriate positions

Demo

Human Instruction: “I want to put the groceries in the fridge

Alice’s POV

Bob’s POV

DINaMo

Key Takeaways

Key Takeaways

- LLaMAR performs 30% better than other LLM-based methods for multi-agent long-horizon planning

Key Takeaways

- LLaMAR performs 30% better than other LLM-based methods for multi-agent long-horizon planning
- Vision Language Models perform better than pure text-based language models

Key Takeaways

- LLaMAR performs 30% better than other LLM-based methods for multi-agent long-horizon planning
- Vision Language Models perform better than pure text-based language models
- A modular cognitive architecture with distinct roles helps with solving complex tasks.

Key Contributions:

Key Contributions:

- A **modular cognitive architecture** called LLaMAR which integrates planning, acting, correcting and self-verification through distinct LLM roles

Key Contributions:

- A **modular cognitive architecture** called LLaMAR which integrates planning, acting, correcting and self-verification through distinct LLM roles
- MAP-THOR a **test-suite and a benchmark** on language-based multi-agent robotic planning based on AI2THOR

Key Contributions:

- A **modular cognitive architecture** called LLaMAR which integrates planning, acting, correcting and self-verification through distinct LLM roles
- MAP-THOR a **test-suite and a benchmark** on language-based multi-agent robotic planning based on AI2THOR
- LLaMAR can create **performant long-horizon planning** in multi-agent tasks by creating subtasks and assigning them to different agents