Practical AI for Autonomous Robots

Day 1: Introduction to Robotic Software and Architectures

Dr. Timothy Wiley
School of Computing Technologies
RMIT University

ESSAI July 2024

RMIT UNIVERSITY

Acknowledgement of Country

RMIT University acknowledges the people of the Woi wurrung and Boon wurrung language groups of the eastern Kulin Nation on whose unceded lands we conduct the business of the University.

RMIT University respectfully acknowledges their Ancestors and Elders, past and present.

RMIT also acknowledges the Traditional Custodians and their Ancestors of the lands and waters across Australia where we conduct our business.

Artwork 'Luwaytini' by Mark Cleaver, Palawa

About me: Dr. Timothy Wiley

PhD, Online Learning of Robotic Behaviours,
UNSW Australia

Lecturer, Artificial Intelligence,
School of Computing Technologies
STEM College, RMIT University

Manager,
Artificial Intelligence Innovation Lab

Research Interests:

- Autonomous Robotics
- Reinforcement Learning
- Qualitative Reasoning
- Uncrewed Ariel Systems
- Sim2Real

Robot Soccer

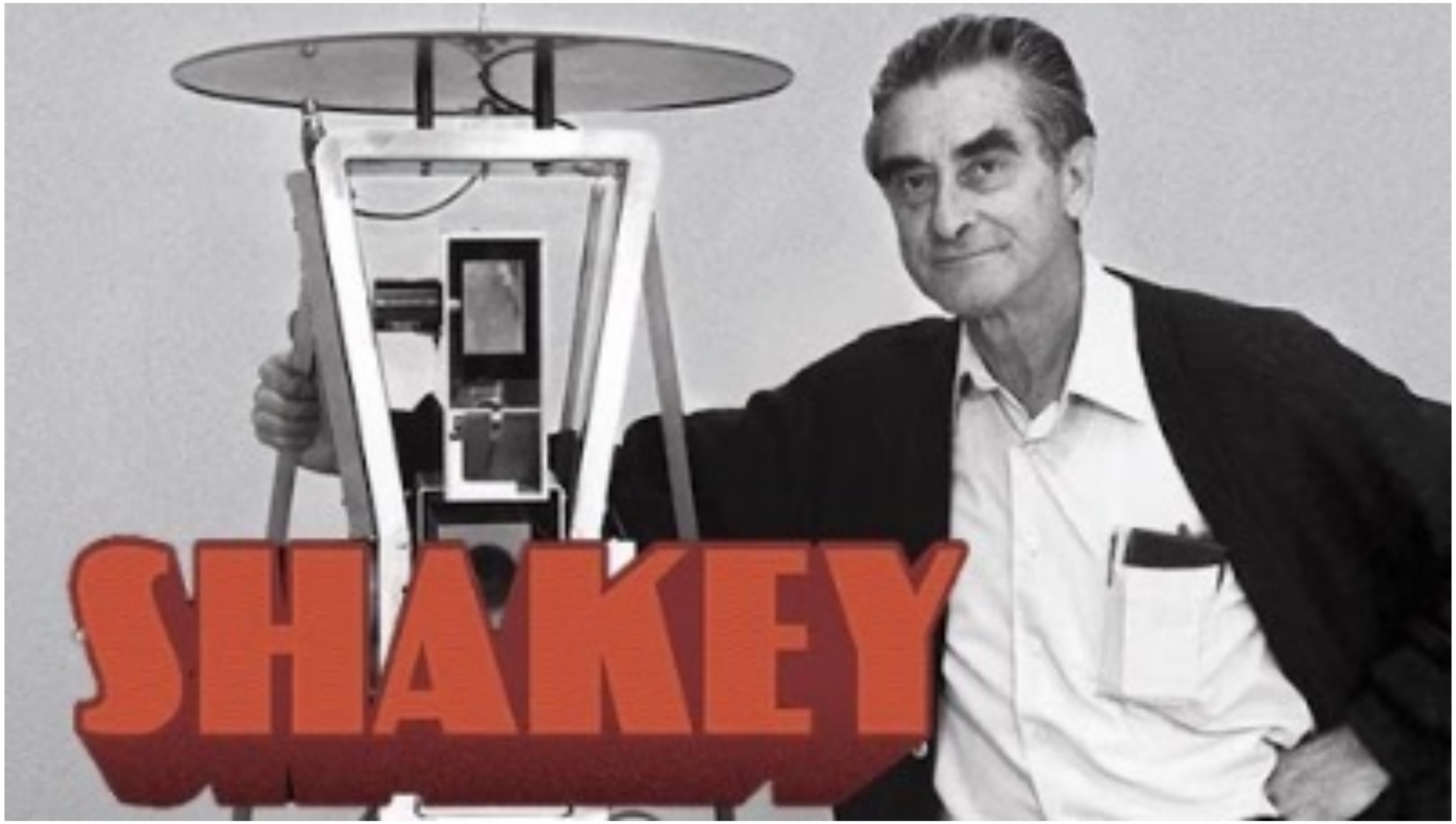
Motivation

ESSAI July 2024

What defines an Autonomous Robot?

What defines an autonomous robot to you?

Shakey the Robot

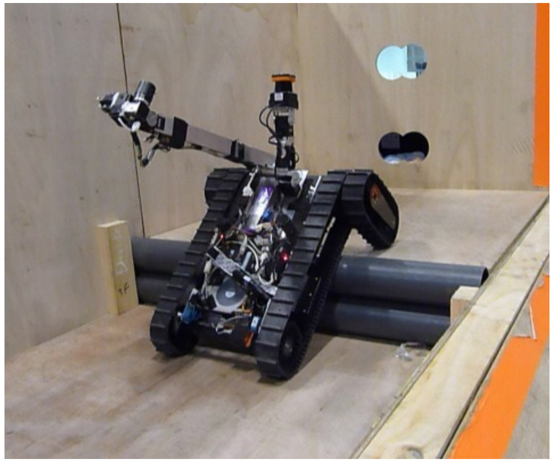


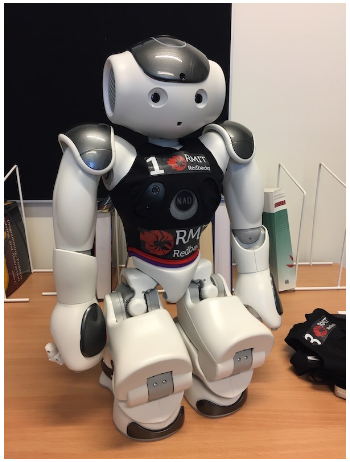
Nilsson, N. J. (1982). Principles of Artificial Intelligence. Berlin Heidelberg: Springer-Verlag

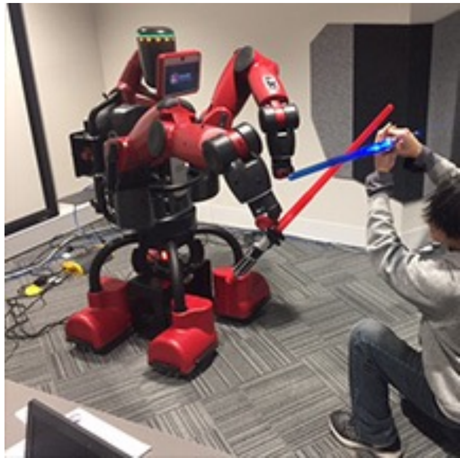
What defines an Autonomous Robot?

Aside from the obvious “act autonomously” that is, “think-and-act for themselves”:

- Perceive the environment
- Interact with the environment
- Internal reasoning





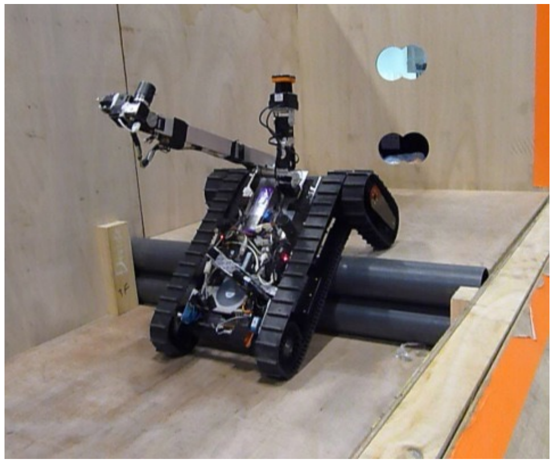


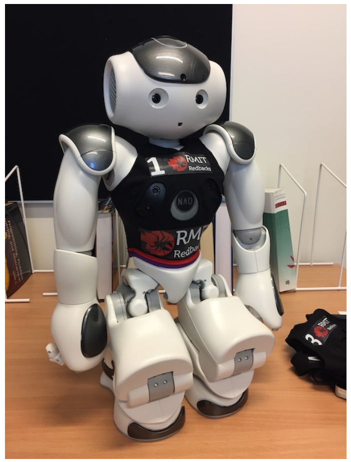
What defines an Autonomous Robot?

Aside from the obvious “act autonomously” that is, “think-and-act for themselves”:

- Perceive the environment
- Interact with the environment
- Internal reasoning ← This is where AI sits, but…

What separates robotics from other fields of AI is the physical input/output





About the Course

ESSAI July 2024

This Course is about…

- Overview of Autonomous Robotics
- Give a breadth of knowledge rather than a depth
- Provide operational knowledge on autonomous robotic system for later in-depth study
- Applied AI/ML
- Hopefully convince you to work in robotics 😛

Topics

Day 1: Introduction to Robotic Software and Architectures

Day 2: Control and Manipulation

Day 3: Localisation, Mapping and Navigation

Day 4: Robotic Vision

Day 5: Task Planning & Miscellaneous topics

This Course is presumes…

- Some familiarity with common AI techniques such as:
  - Graph algorithms
  - Heuristic Search (A*)
  - Random Sampling
  - Machine Learning fundamentals
  - Neural Network structures

Autonomous Challenges

ESSAI July 2024

Problem: Maze

Problem: Maze

Problem: RoboCup Soccer

Elements of Robots

ESSAI July 2024

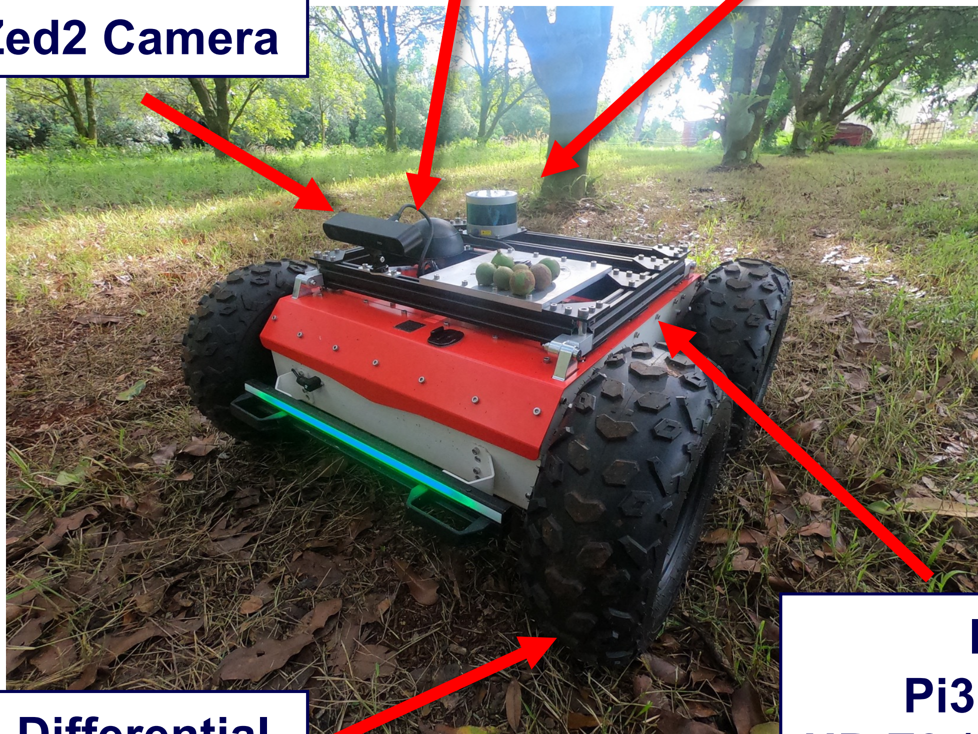
ROSbot

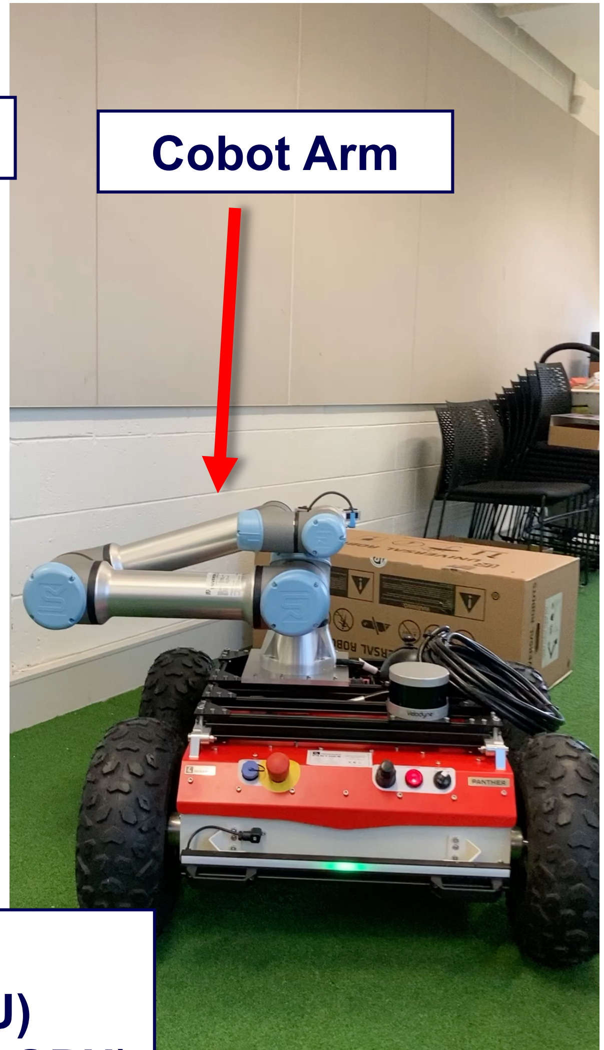
ROSbot

2D LIDAR
RGB-D Camera
Wi-fi
1D Ustrasound
IMU UpBoard (CPU)
Differential Drive

Panther

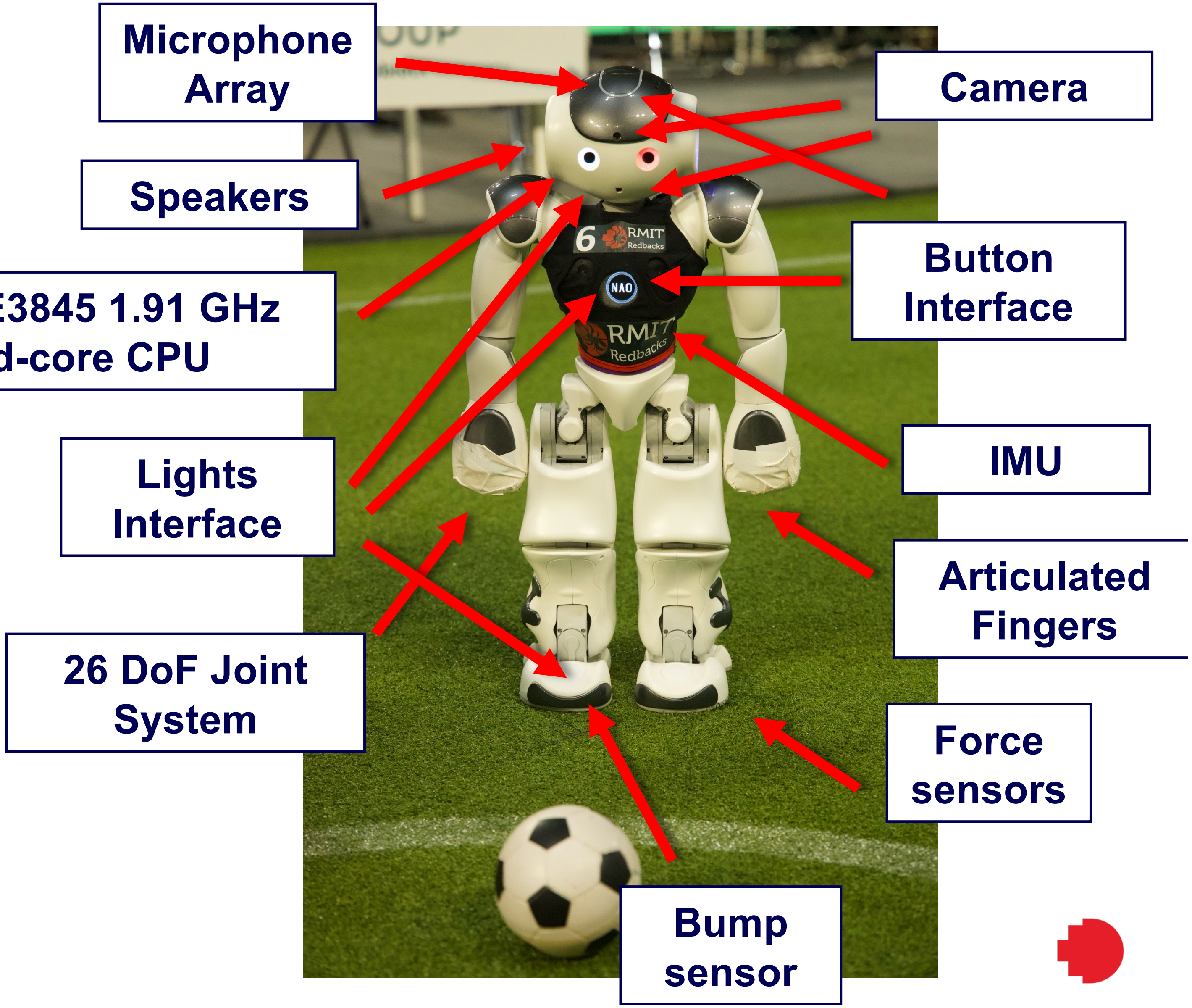
Panther


RGB-D AI
Zed2 Camera
Wi-fi x2
GPS
3D LIDAR
IMU
Pi3 (CPU)
HP-Z2 (CPU+GPU)
Differential
Drive


Cobot Arm

Nao V6

Nao V6



Microphone Array
Camera
Speakers
Button Interface
ATOM E3845 1.91 GHz Quad-core CPU
IMU
Lights Interface
Articulated Fingers
26 DoF Joint System
Force sensors
Bump sensor

Autonomous Software Components

ESSAI July 2024

Autonomous Robot Software Components

World Modelling

- Localisation
- Mapping

Navigation

- Path Planning
- Execution and Search
- Obstacle Avoidance

Behaviour Decision Making Planning

Sensor Processing

- Vision
- Audio

Control

- Locomotion
- Grasping
- Motion Planning

Autonomous Robot Software Components

All of these can be performed at different levels of abstraction:

- Raw sensor signals and raw actuator operations
- Quantised numerical processing
- Sub-symbolic representations & reasoning
- Symbolic representations & reasoning

ROSBot Maze

World Modelling

- 2D SLAM (Localisation & Mapping)
- Hazard Marker Record
- Traversed Path Tracking
- Local & Global Modelling

Navigation

- Unknown Space Exploration
- Object Search

Search + Recognition

Sensor Processing

- Hazard Marker Recognition

Control

- Non-Holonomic (Differential) Drive

RoboCup Soccer

World Modelling

- Localisation
- Ball / Robot persistence
- Team Coordination & Shared State
- Moving object trajectory prediction

Navigation

- Path Planning
- Dynamic obstacle avoidance
- Prediction of kicks

Soccer Skills

Sensor Processing

- Ball Detection
- Field Feature Detection
- Robot Detection

Control

- Bi-pedal Walk & Balance
- Head & Arm Control
- Kicks
- Get-ups

Autonomous Agent Model

ESSAI July 2024

AI Agent Models

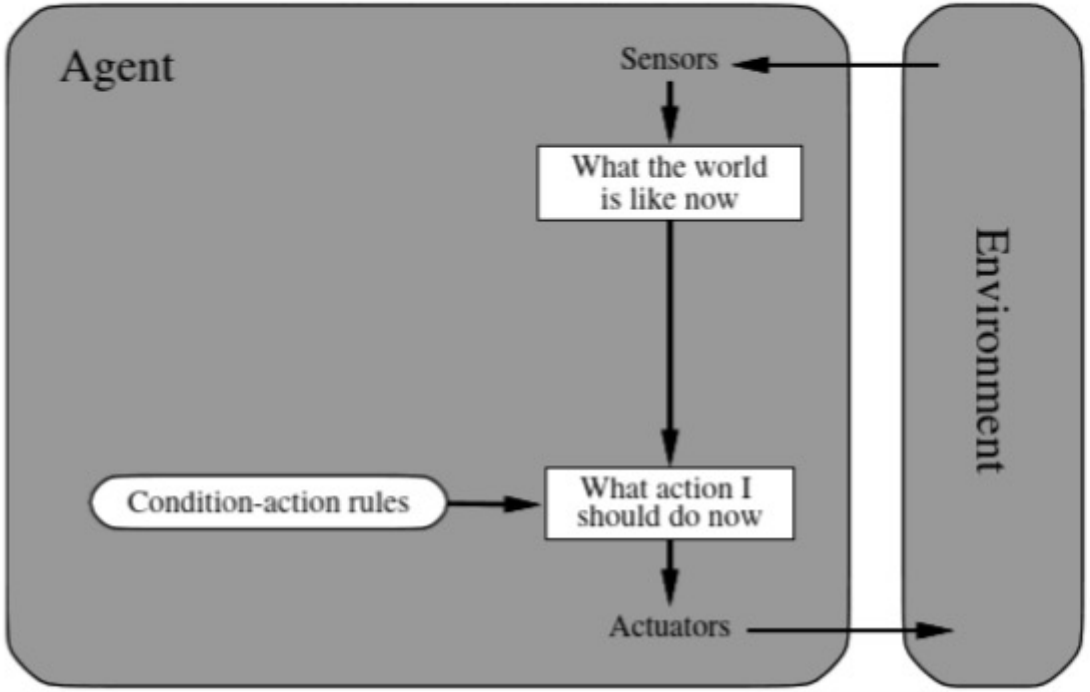


Figure 2.9 Schematic diagram of a simple reflex agent.

Russel & Norvik. Artificial Intelligence: A Modern Approach (2017),

AI Agent Models

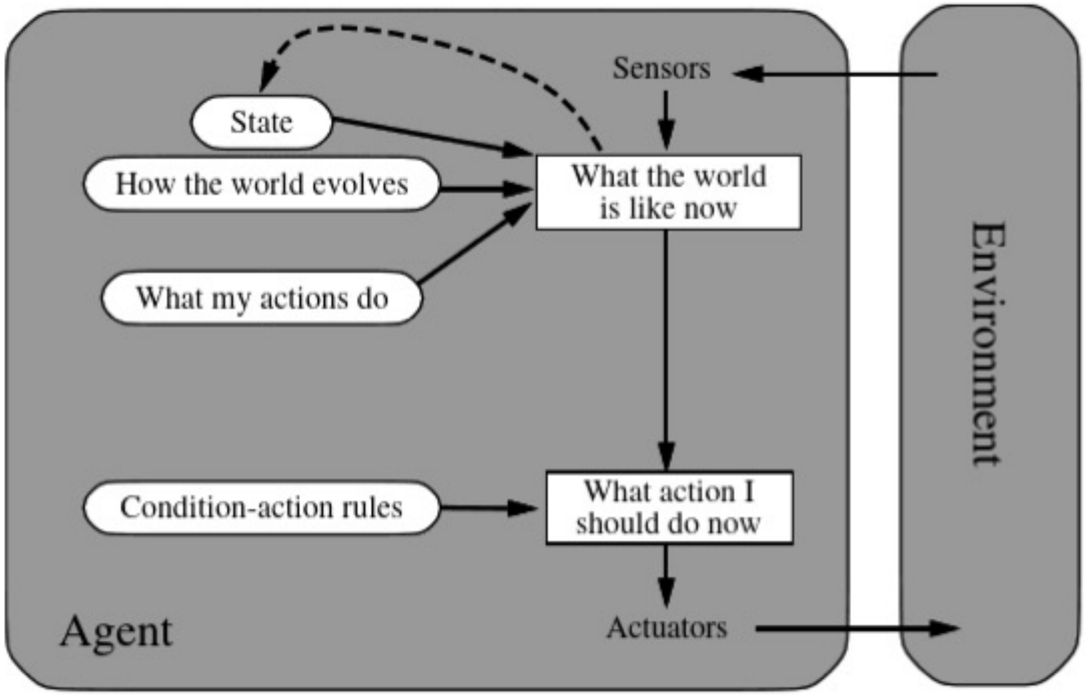


Figure 2.11 A model-based reflex agent.

Russel & Norvik. Artificial Intelligence: A Modern Approach (2017),

AI Agent Models

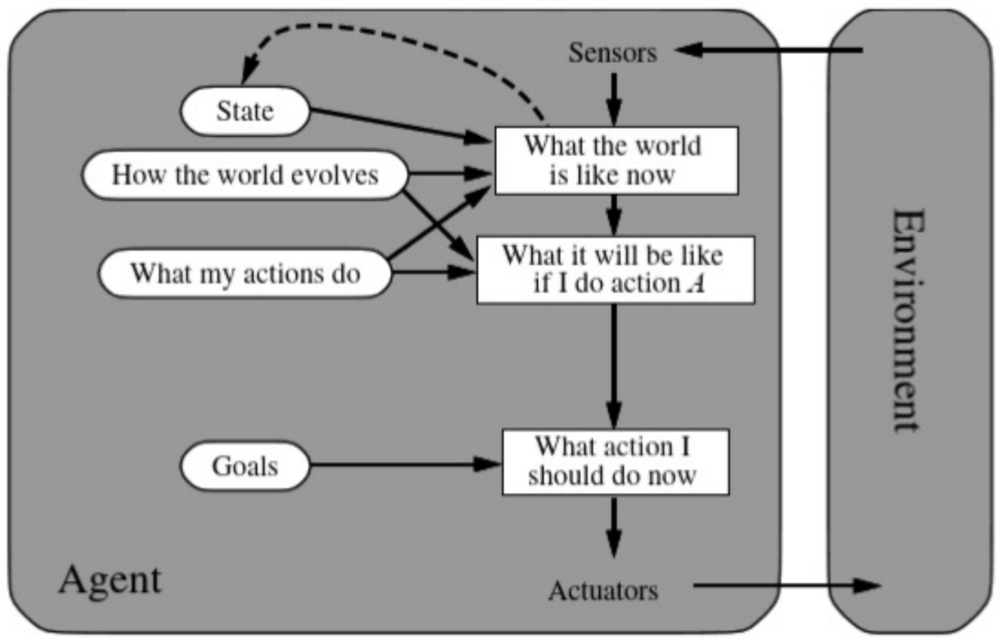


Figure 2.13 A model-based, goal-based agent. It keeps track of the world state as well as a set of goals it is trying to achieve, and chooses an action that will (eventually) lead to the achievement of its goals.

Russel & Norvik. Artificial Intelligence: A Modern Approach (2017),

AI Agent Models

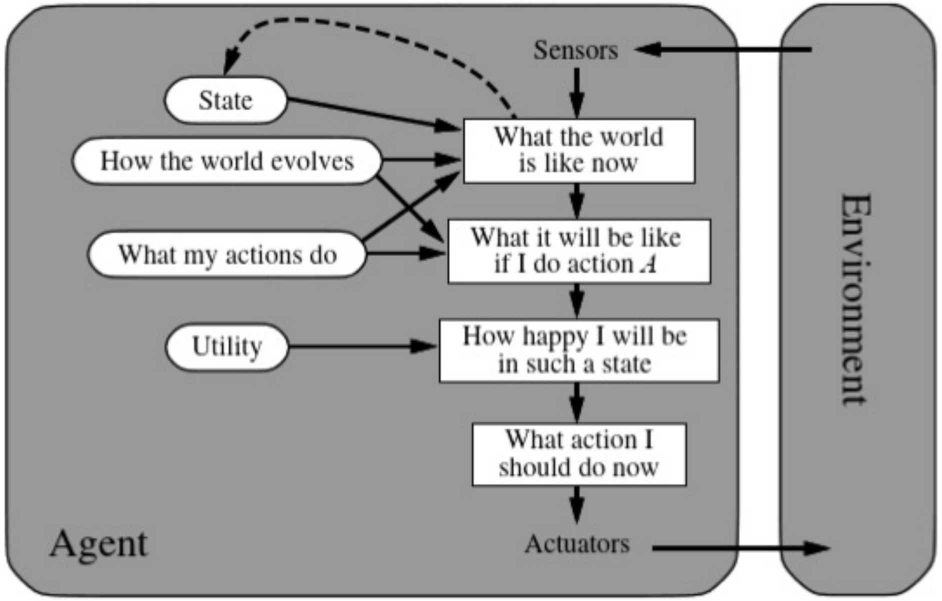


Figure 2.14 A model-based, utility-based agent. It uses a model of the world, along with a utility function that measures its preferences among states of the world. Then it chooses the action that leads to the best expected utility, where expected utility is computed by averaging over all possible outcome states, weighted by the probability of the outcome.

Russel & Norvik. Artificial Intelligence: A Modern Approach (2017),

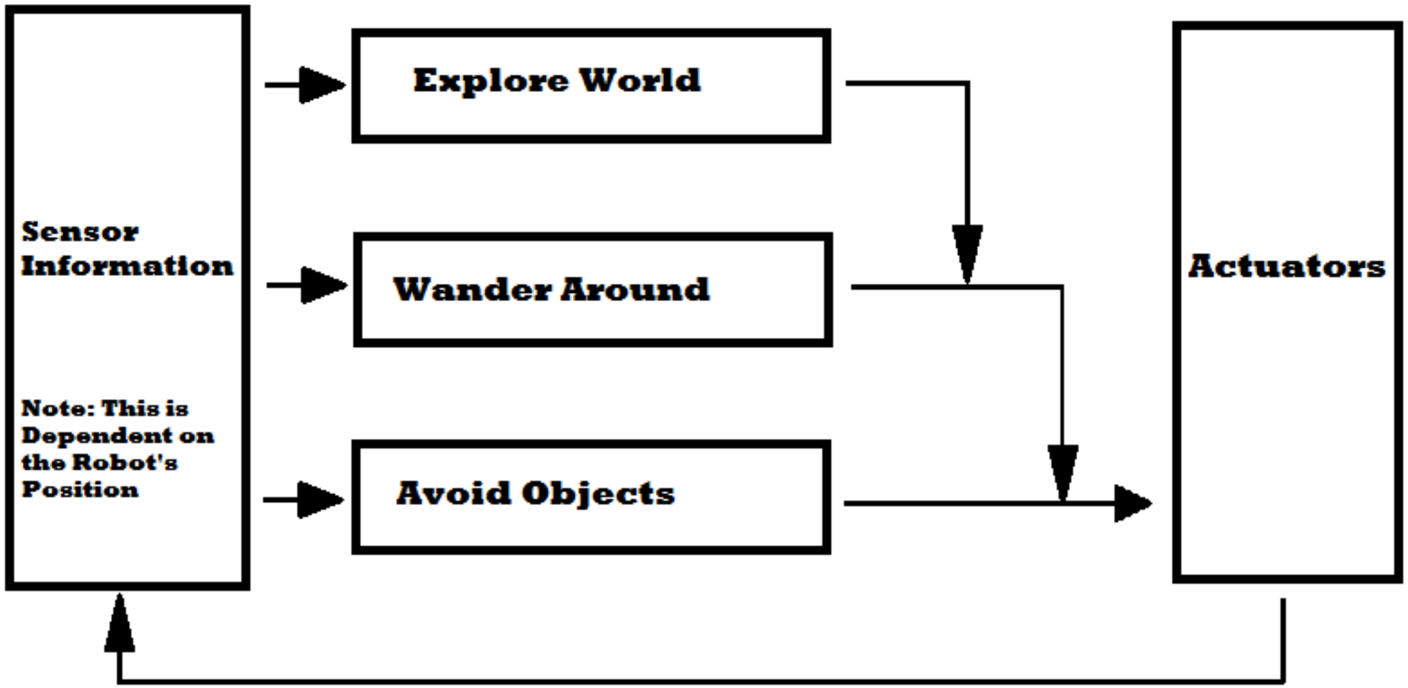
Software Architectures

ESSAI July 2024

Software Architectures: Sense-Plan-Act

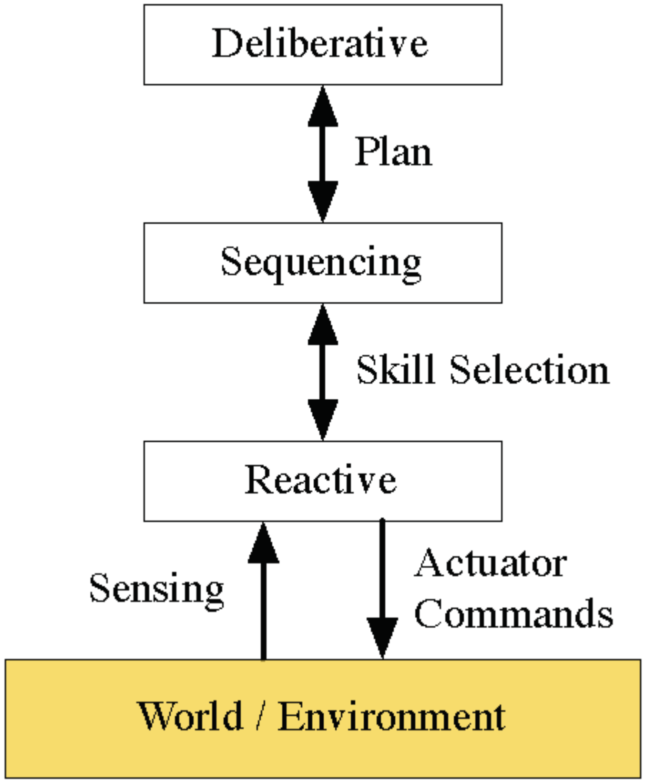
Nilsson, N. J. (1982). Principles of Artificial Intelligence. Berlin Heidelberg: Springer-Verlag

Software Architectures: Subsumption



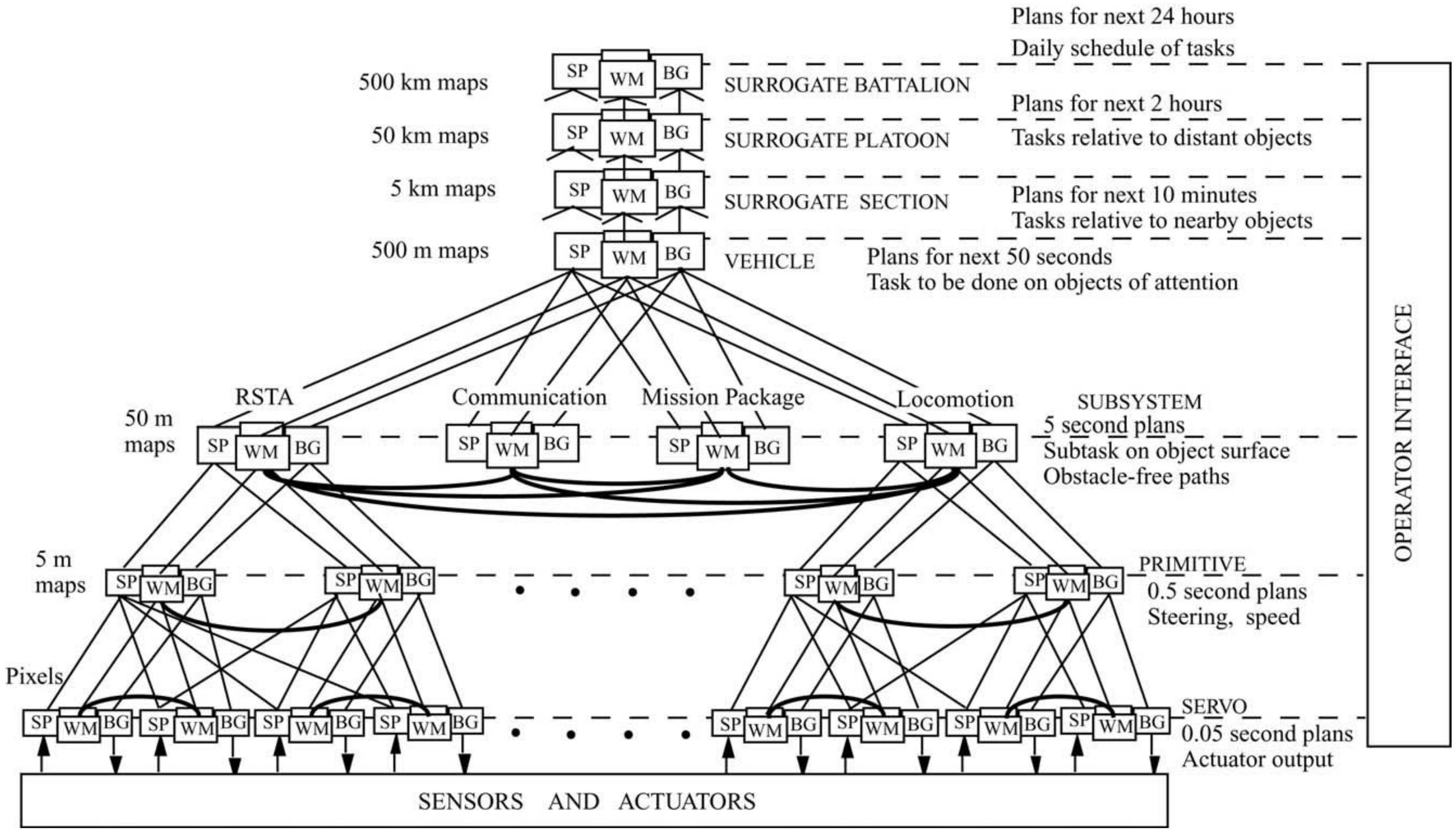
R. A. Brooks (1986), "A Robust Layer Control System for a Mobile Robot", IEEE Journal of Robotics and Automation RA-2, 14-23

Software Architectures: Three-Layer



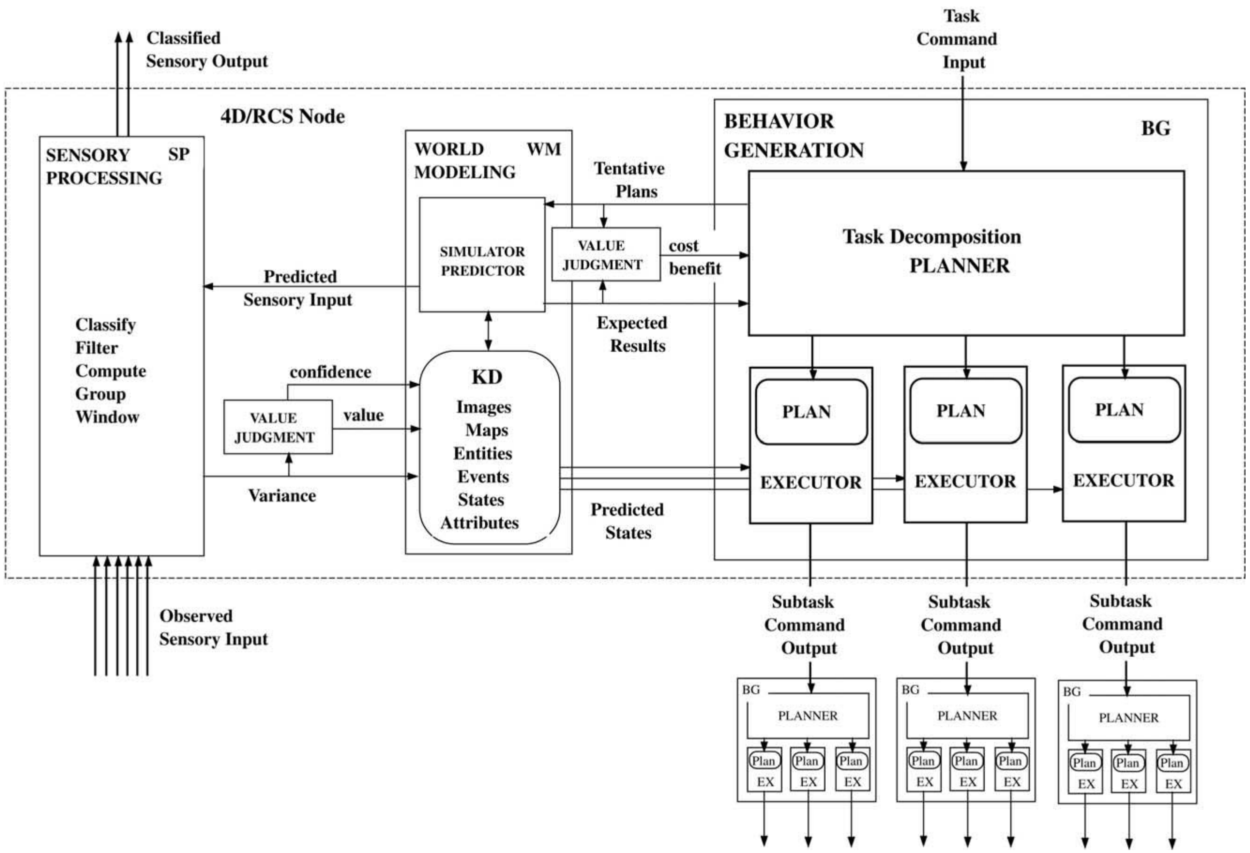
Bonasso, P. et. al. (1997). Experiences with an architecture for intelligent, reactive agents. Journal of Experimental & Theoretical Artificial Intelligence 9(2-3):237– 256

Software Architectures: RCS



Albus, J. S. & Barbera, A. J. (2005) RCS: A cognitive architecture for intelligent multi-agent systems. Annual Rev Control 29, 87–99.

Software Architectures: RCS



Albus, J. S. & Barbera, A. J. (2005) RCS: A cognitive architecture for intelligent multi-agent systems. Annual Rev Control 29, 87–99.

ROS

Robot Operating System

ESSAI July 2024

ROS: Robot Operating System

ROS is a standard platform for developing robotic software.

ROS provides:

1. Consistent Communication Framework
2. Transparent Network handling
3. Common software packages
4. Open Source Package library

ROS: Robot Operating System

ROS was first created by Willow Garage in 2010.

Now supports:

- Annual version tied to the xx.04 Ubuntu annual release
- ROS (v1)
  - Original ROS framework
  - Large package library
  - Centralised structure with “roscore”
- ROS2 (v2)
  - Revised & improved networking stack
  - Decentralised structure
  - Package library less well supported

ROS Concepts: Nodes

Standalone container of single software executable:

- Asynchronous execution & communication
- By default, each node is run is a separate thread
- Can be given configuration parameters
- Named by:
  - /<namespace>/<node_name>
- Executed by either:
  - rosrun
  - roslaunch

ROS Concepts: Topics

Define a communication stream. The communication is defined by:

- Topic name: /<namespace>/<topic>
- Messages are sent via the topic between nodes
- Topic type: The ROS message type of the communication, such as string, image, geometry, transform, map, etc.
    - Types can be simple, to as complex as needed
- Stateless communication

Any node can:

- Publish a message to a topic
- Subscribe to a topic, to receive all messages published to the topic

ROS Concepts: Topics

The ROS infrastructure:

- Manages all networking regardless of where nodes are running.
- Established direct node-to-node links between all pairs of codes that publish/subscribe to a topic
- Uses TCP connections for nodes on different machines
  - 🚨 ROS networking is particular about having correct configuration, especially in ROS1.
  - ROS chooses random TCP ports to establish connections, and these must be open for bi-directional communication

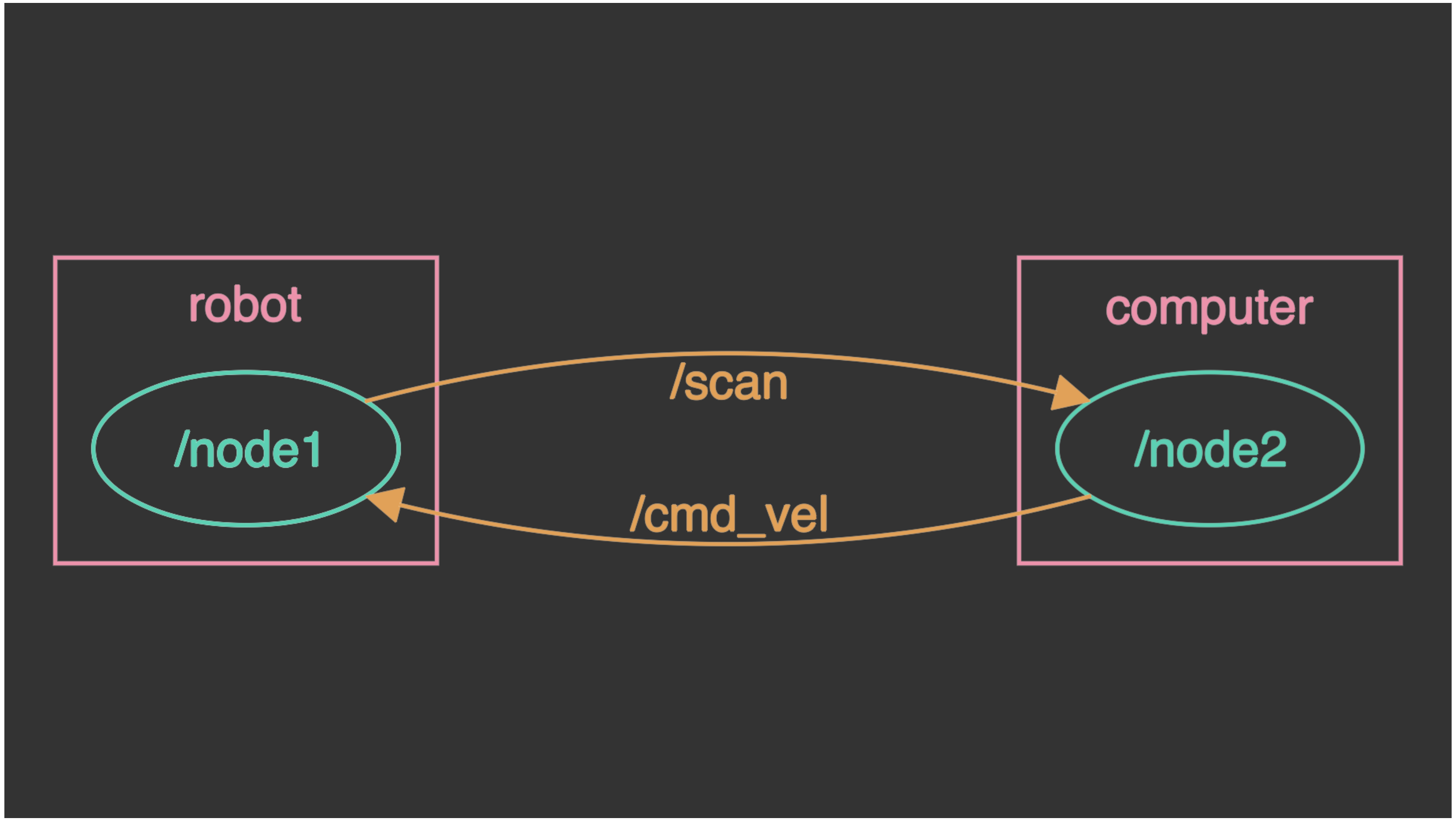
ROS Concepts: Services

Topics are a publisher/subscriber model, so have no state.

ROS Services allow for callable methods:

- Use a client-server model
- A service call includes:
  - Request from the client to the server
  - Reply from the server to the client
  - Status updates from the server to the client

ROS Overview



TheConstruct

Online ROS Platform

ESSAI July 2024

Noon Gudgin

Thank you

Day 2:Control and Manipulation

ESSAI July 2024