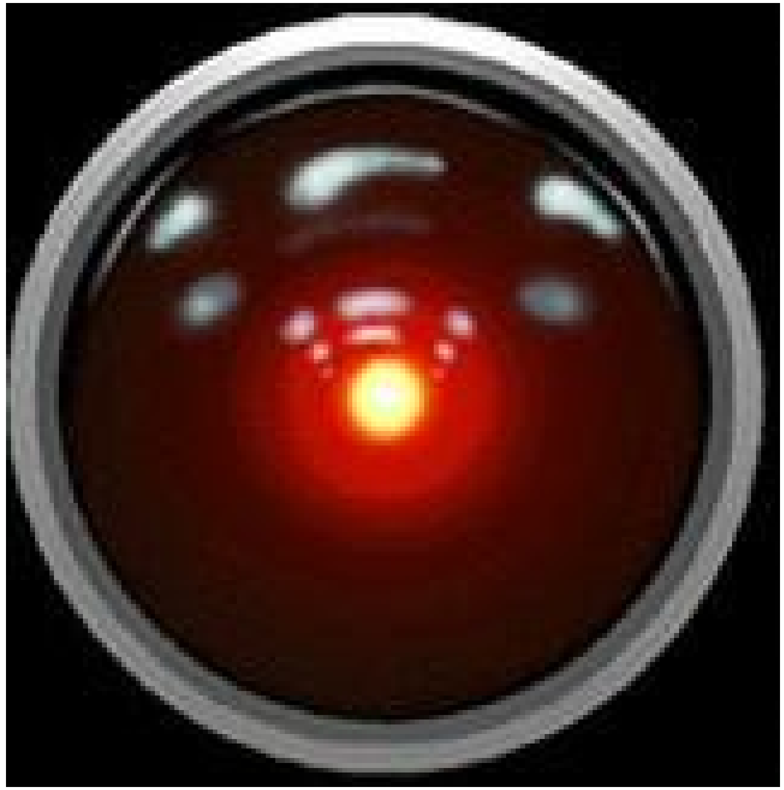


CS 378: Autonomous Intelligent Robotics

Instructor: Jivko Sinapov

http://www.cs.utexas.edu/~jsinapov/teaching/cs378/

Announcements

FRI Summer Research Fellowships:

https://cns.utexas.edu/fri/beyond-the-freshman-lab/fellowships

Applications are due March 1st but apply now!

Funding is available for 4-5 students per FRI stream

Announcements

Homework 3 is due Friday "night"

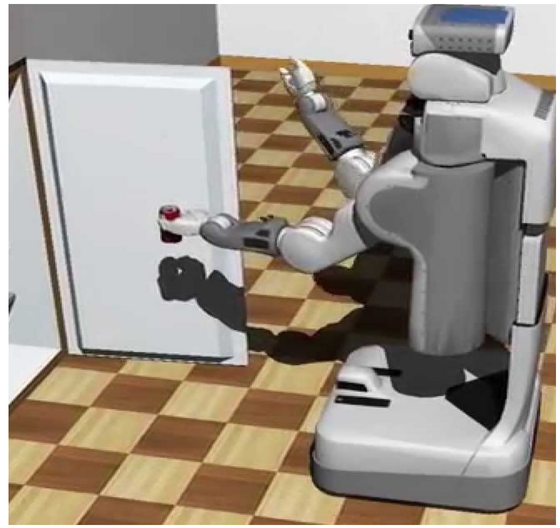
Readings for this week: Behavior-based robotics

R. Brooks (1986). ``A Robust Layered Control System for a Mobile Robot'', MIT AI Memo 864, Vol RA-2, No. 1. p. 14-23

R. Brooks (1991). "Intelligence Without Representation", Artificial Intelligence, Volume 47 , Issue 1-3

Progression

2D simulation



3D simulation

Real World

The Gazebo 3D simulator

- Install gazebo_ros package:

  sudo apt-get install ros-indigo-gazebo-ros

- Run the simulator:

  roslaunch gazebo_ros rubble_world.launch

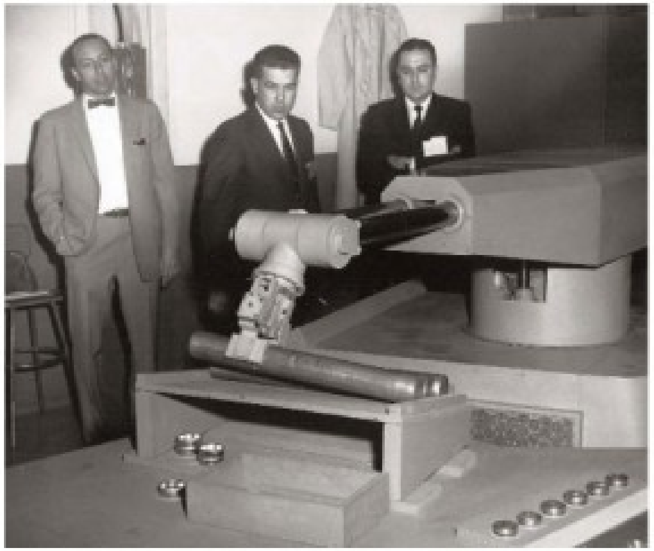
Today

1) Behavior-Based Robotics

2) ROS Services (part 2)

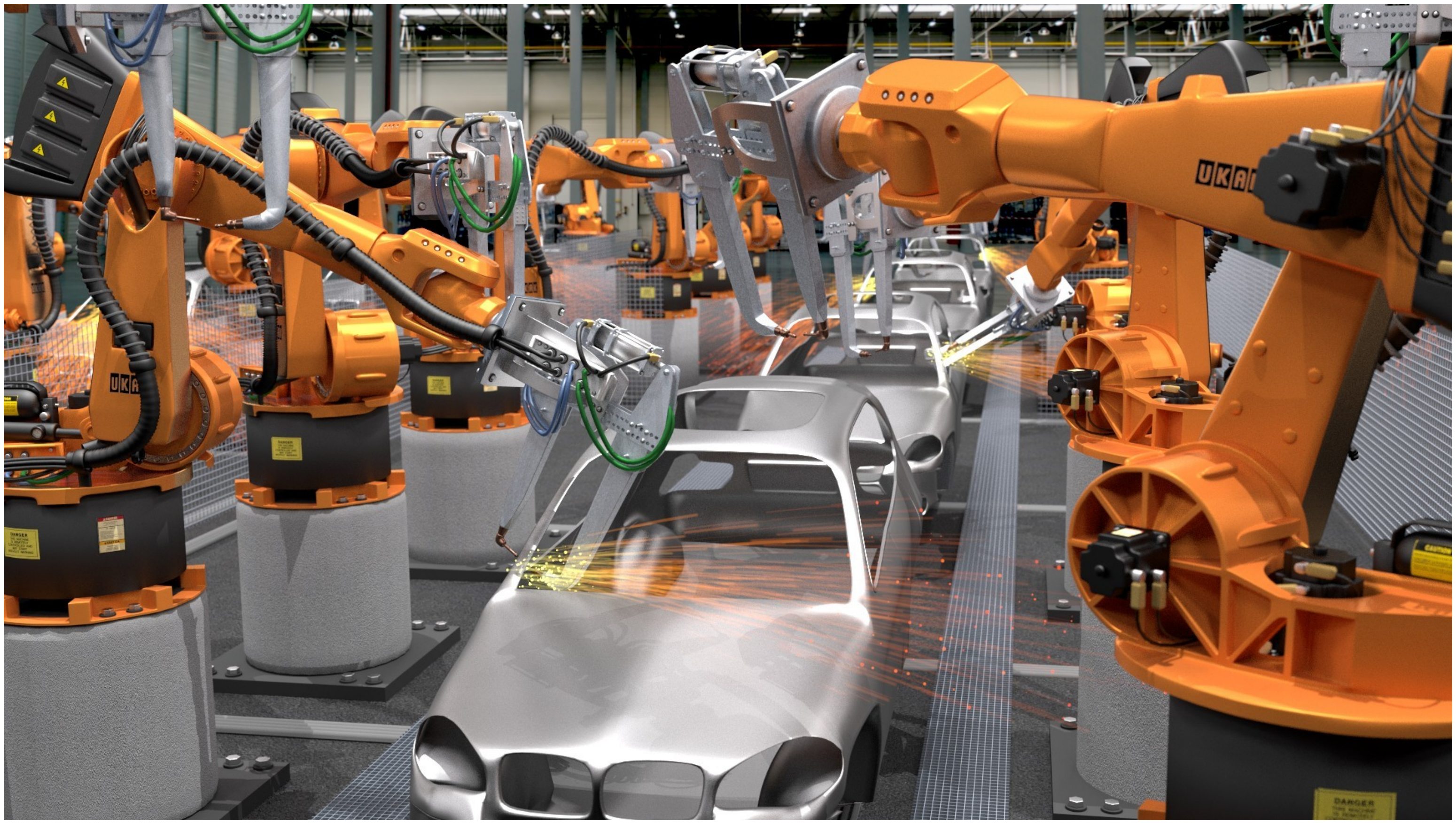
Behavior-Based Robotics

A Bit of History

First Industrial Robot (~60s)

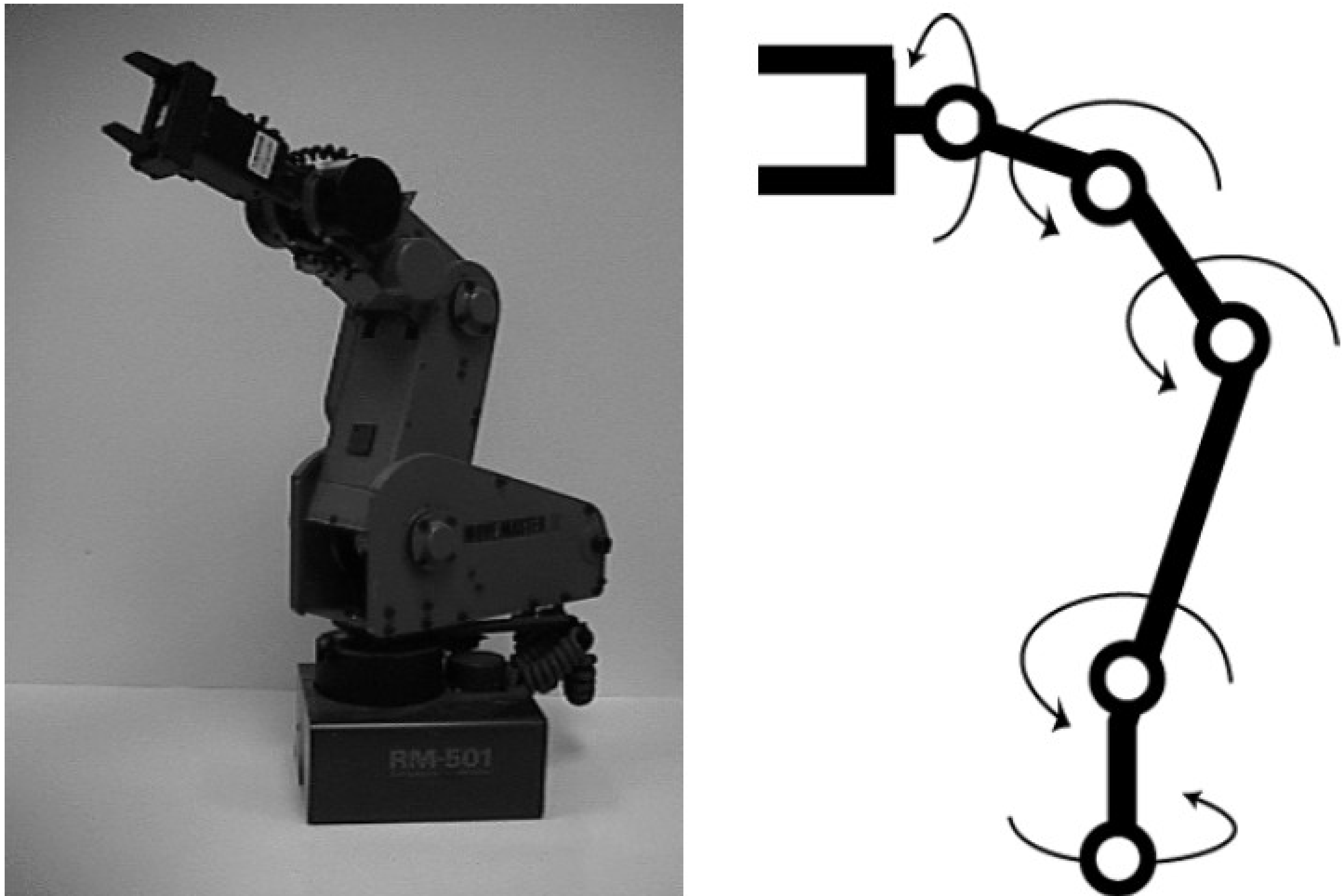
Modern Industrial Robots



Teleoperation

Figure 1.2 A Model 8 Telemanipulator. The upper portion of the device is placed in the ceiling, and the portion on the right extends into the hot cell. (Photograph courtesy Central Research Laboratories.)

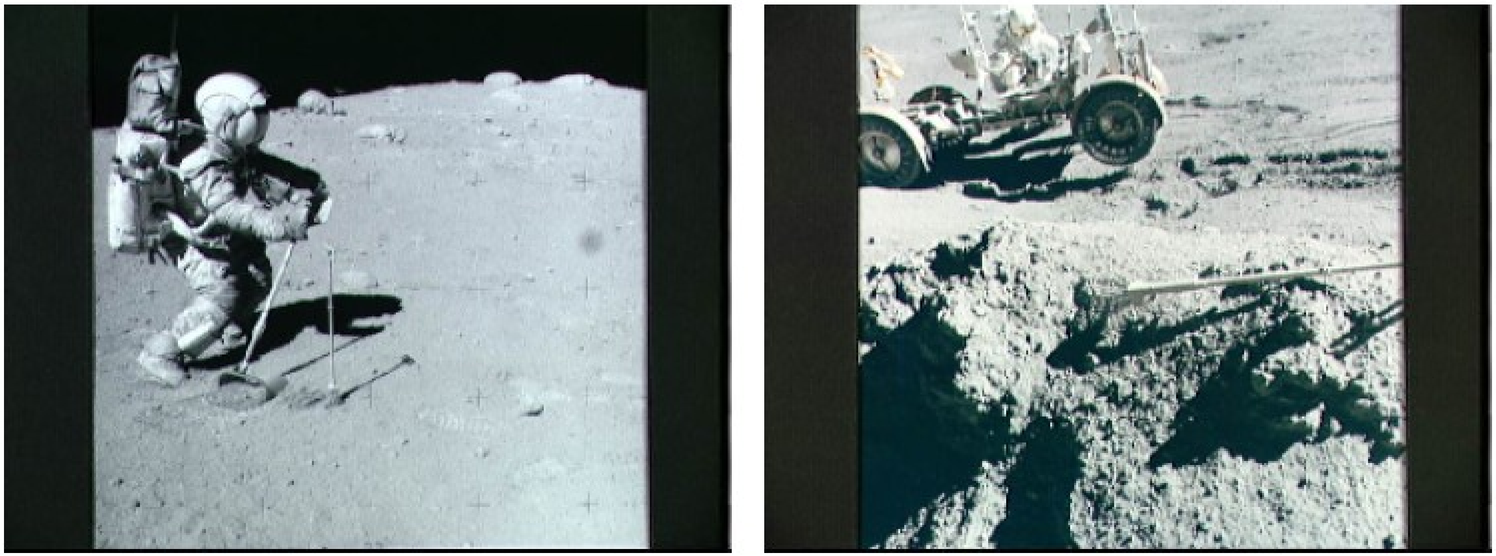
Teleoperation



a. b.

Figure 1.4 A MOVEMASTER robot: a.) the robot arm and b.) the associated joints.

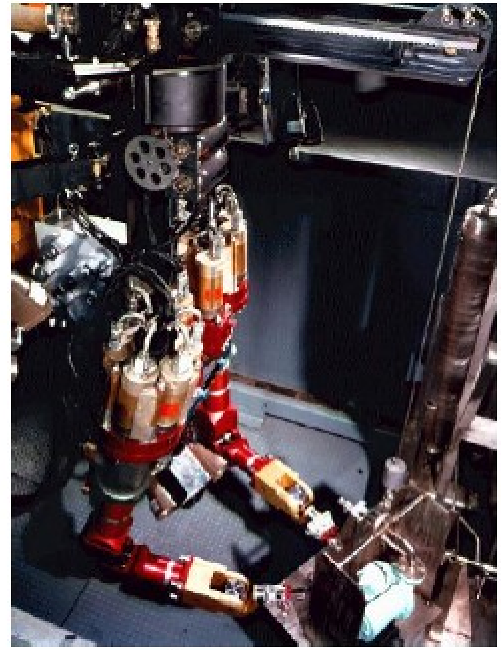
Teleoperation

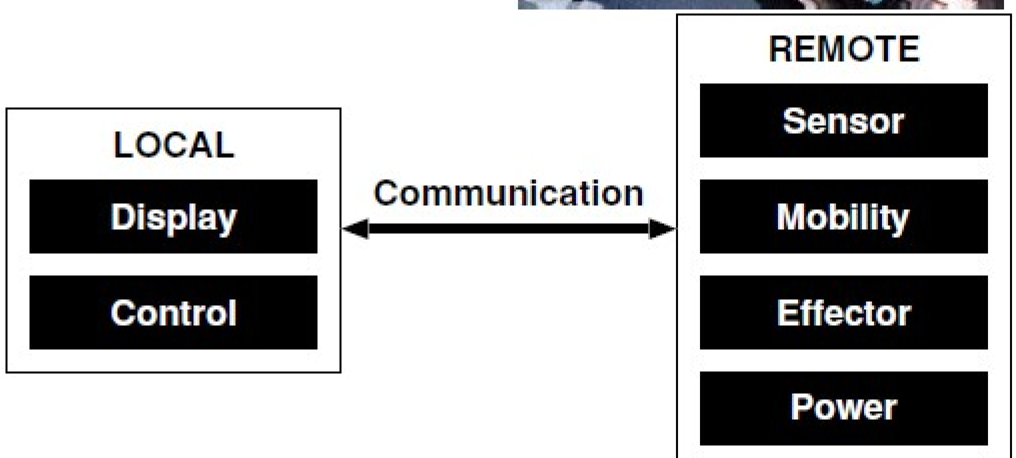


a. b.

Figure 1.5 Motivation for intelligent planetary rovers: a.) Astronaut John Young awkwardly collecting lunar samples on Apollo 16, and b.) Astronaut Jim Irwin stopping the lunar rover as it slides down a hill on Apollo 15. (Photographs courtesy of the National Aeronautics and Space Administration.)

Teleoperation





Teleoperation

Figure 1.7 Sojourner Mars rover. (Photograph courtesy of the National Aeronautics and Space Administration.)

Teleoperation

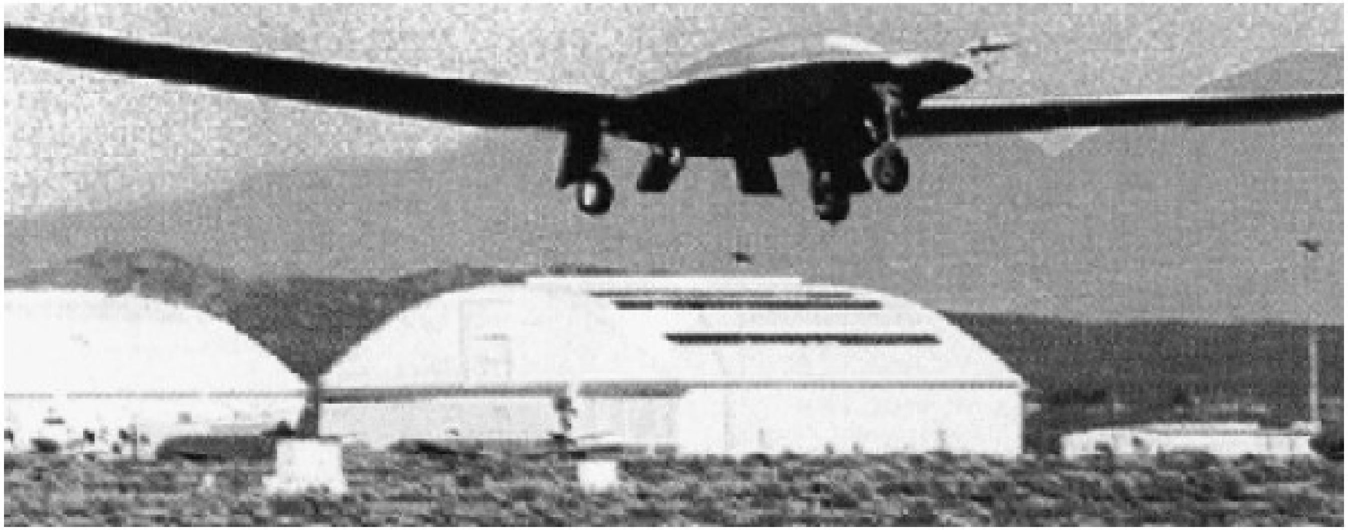


Figure 1.8 Dark Star unmanned aerial vehicle. (Photograph courtesy of DefenseLink, Office of the Assistant Secretary of Defense-Public Affairs.)

Robotics Timeline

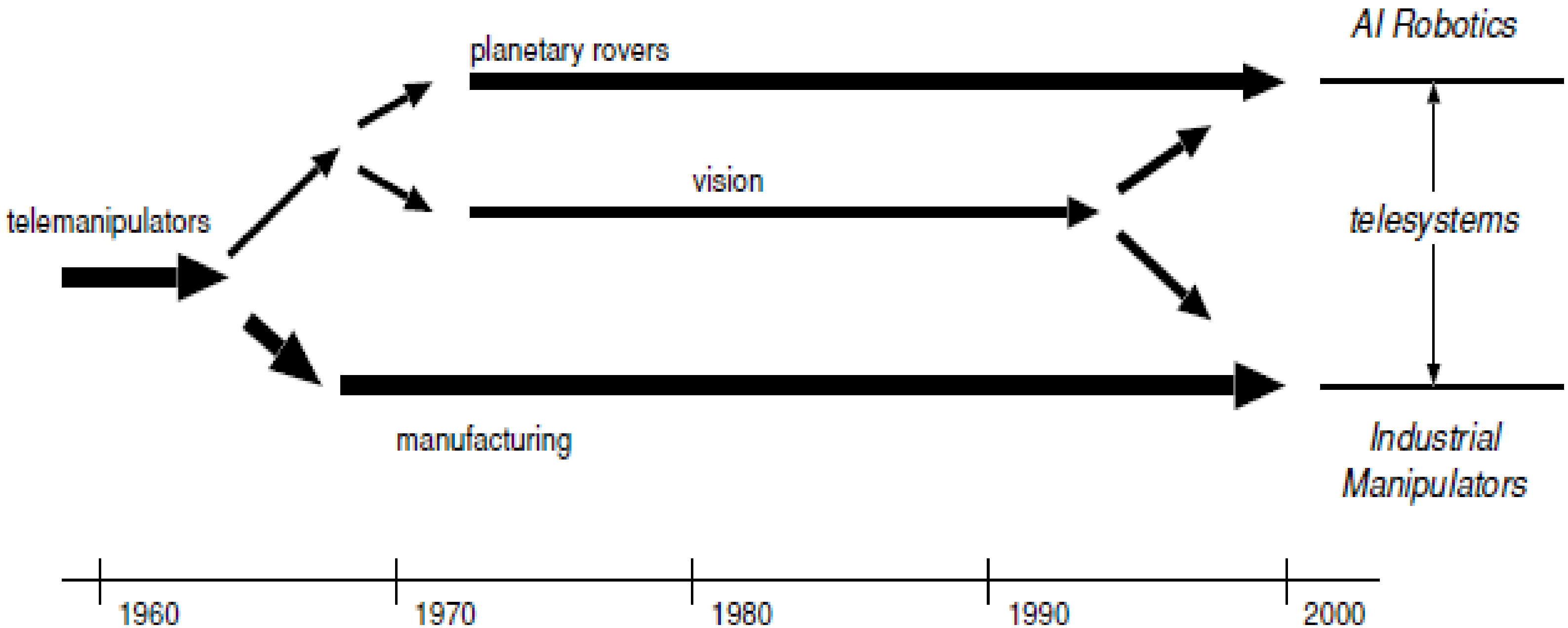


Figure 1.1 A timeline showing forks in development of robots.

Teleoperation vs Telepresence

- An early attempt to improve teleoperation was to add more cameras / displays
- Telepresence aims for placing the operator in a *virtual reality* that mimics the robot's surroundings

Telepresence Robots

http://www.pilotpresence.com/wp-content/uploads/2011/01/remote-presence-systemsv2.jpg

Telepresence Robots

Example:
https://www.youtube.com/watch?v=bVe66UW2XUU

The need for (semi-) autonomy

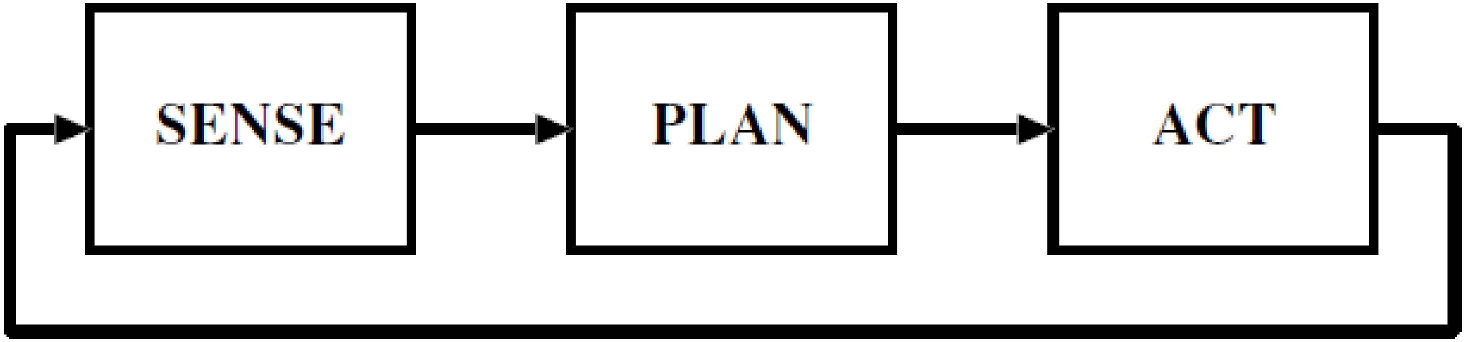
How should autonomy be achieved and organized?

Robot Primitives

ROBOT PRIMITIVES	INPUT	OUTPUT
SENSE	Sensor data	Sensed information
PLAN	Information (sensed and/or cognitive)	Directives
ACT	Sensed information or directives	Actuator commands

Figure I.2 Robot primitives defined in terms of inputs and outputs.

The Early Answer (1967): Sense-Plan-Act

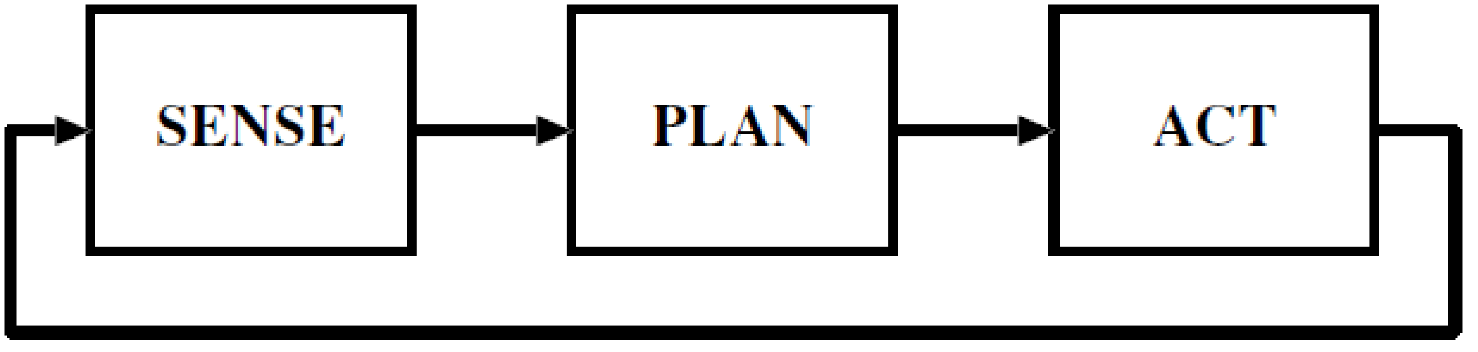


The Early Answer (1967): Sense-Plan-Act

ROBOT PRIMITIVES	INPUT	OUTPUT
SENSE	Sensor data	Sensed information
PLAN	Information (sensed and/or cognitive)	Directives
ACT	[redacted] directives	Actuator commands

Figure I.4 Another view of the Hierarchical Paradigm.

The Early Answer (1967): Sense-Plan-Act



Early Example of S-P-A

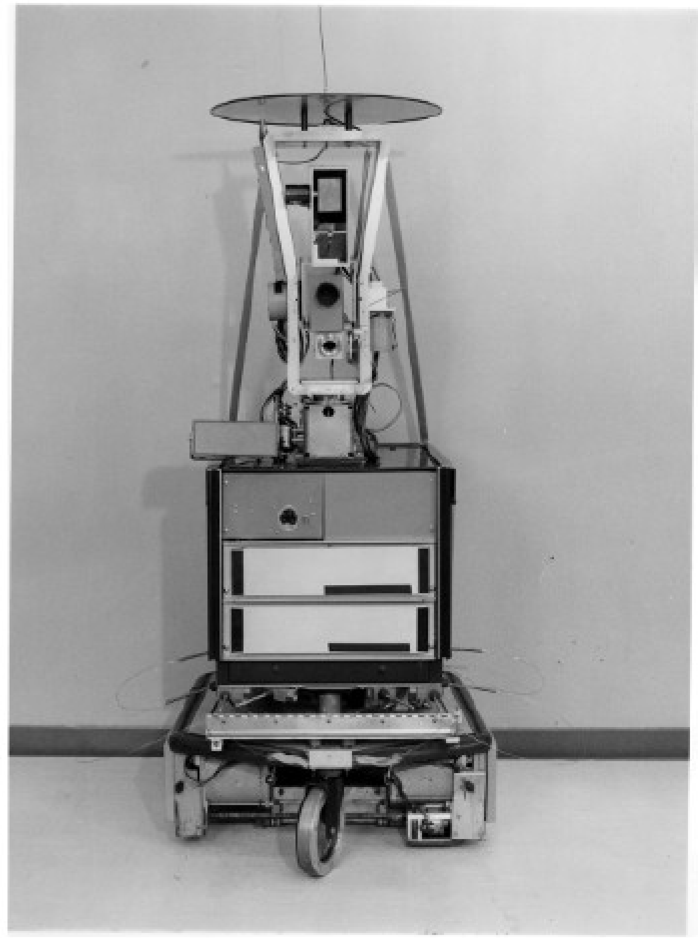


Figure 2.1 Shakey, the first AI robot. It was built by SRI for DARPA 1967–70. (Photograph courtesy of SRI.)

Early Work on Planning

`initial state:`	Tampa, Florida (0,0)
`goal state:`	Stanford, California (1000,2828)
`difference:`	3,000

Early Work on Planning

initial state:	Tampa, Florida (0,0)
goal state:	Stanford, California (1000,2828)
difference:	3,000

difference	operator
$d \geq 200$	fly
$100 < d < 200$	ride_train
$d \leq 100$	drive
$d < 1$	walk

Early Work on Planning

initial state:	Tampa, Florida (0,0)
goal state:	Stanford, California (1000,2828)
difference:	3,000

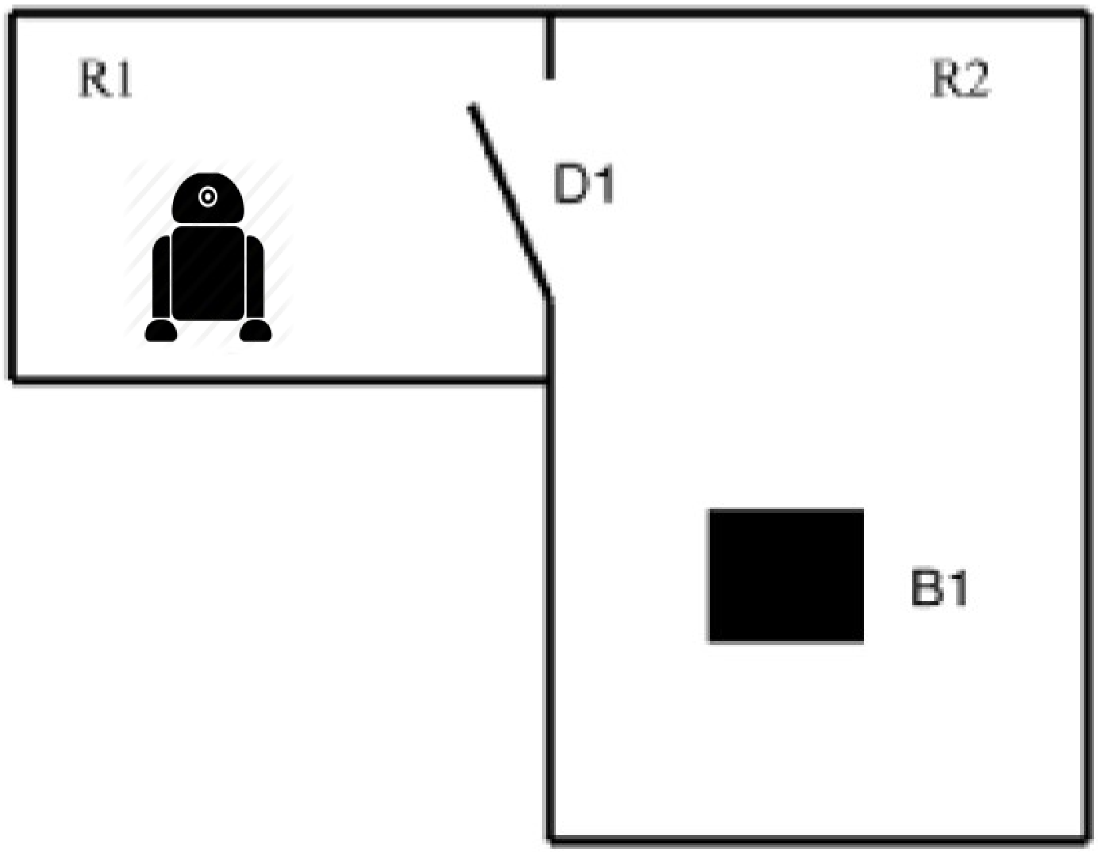
difference	operator	preconditions
d≤200	fly	
100<d<200	ride_train	
d≤100	drive_rental	at airport
	drive_personal	at home
d<1	walk	

Early Work on Planning

difference	operator	pre-conditions	add-list	delete-list
d≤200	fly		at Y at airport	at X
100<d<200	train		at Y at station	at X
d≤100	drive_rental	at airport		
	drive_personal	at home		
d<1	walk			

A More Realistic Example

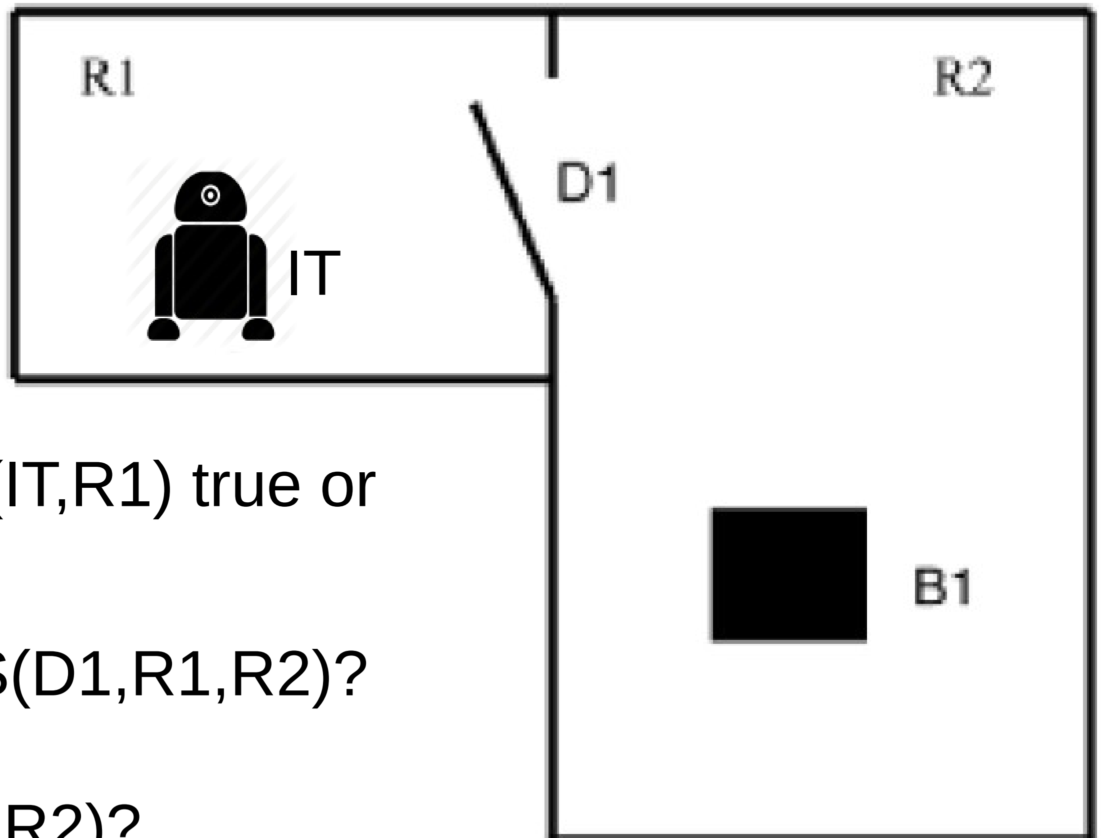
A More Realistic Example



A More Realistic Example

`INROOM(x, r)`	where `x` is an object of type `movable_object`, `r` is type `room`
`NEXTTO(x, t)`	where `x` is a `movable_object`, `t` is type `door` or `movable_object`
`STATUS(d, s)`	where `d` is type `door`, `s` is an enumerated type: `OPEN` or `CLOSED`
`CONNECTS(d, rx, ry)`	where `d` is type `door`, `rx, ry` are the `room`

A More Realistic Example

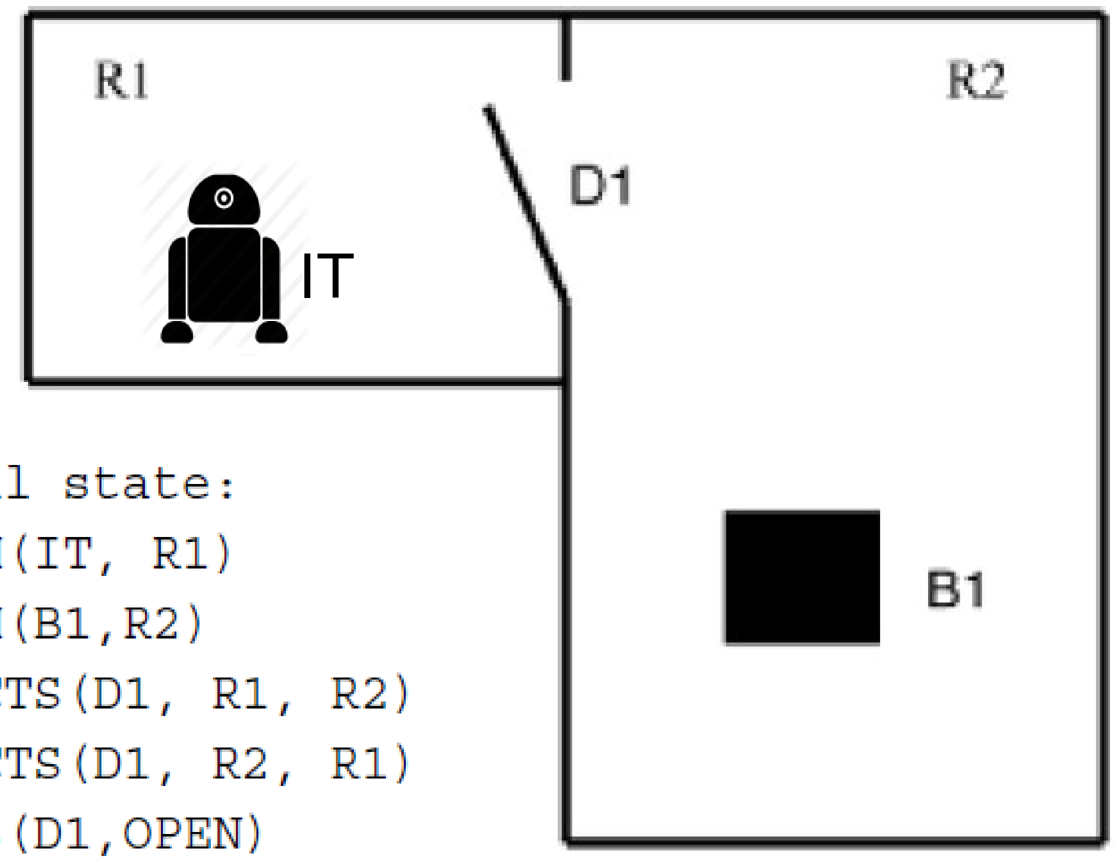


Is INROOM(IT,R1) true or false?

CONNECTS(D1,R1,R2)?

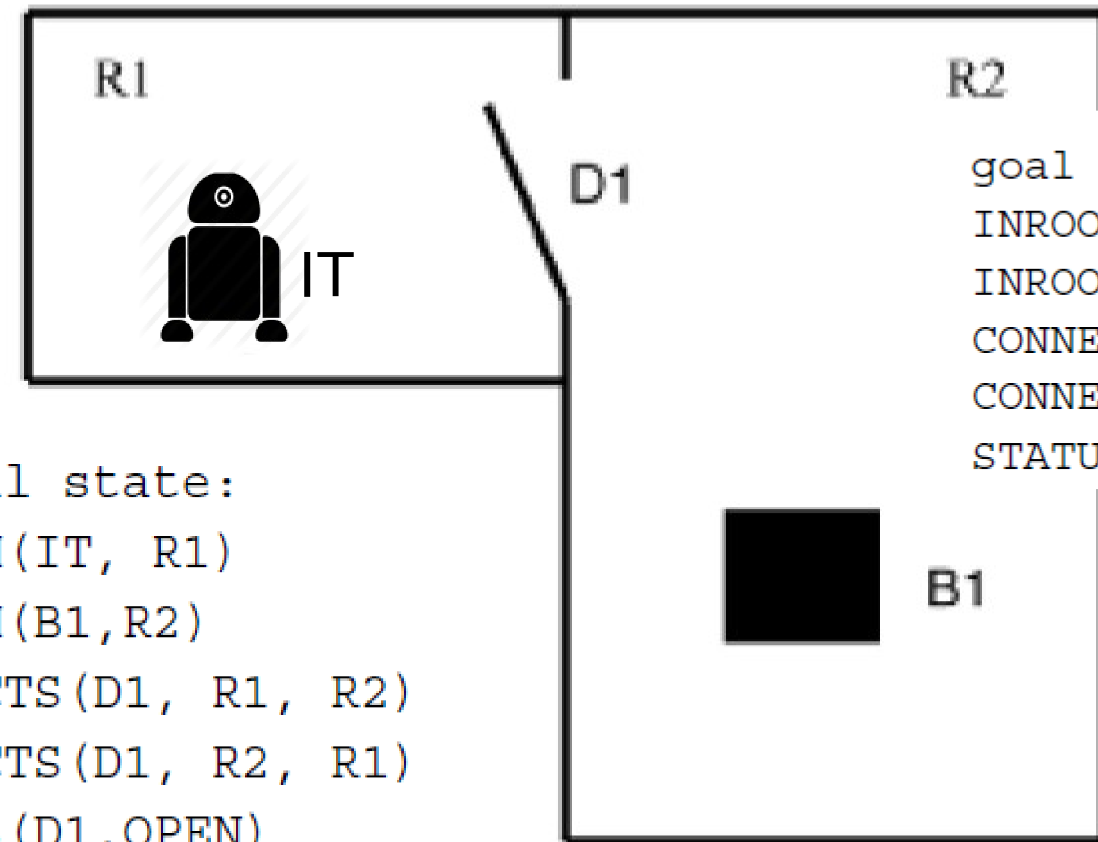
INROOM(IT,R2)?

Representing Initial State




```
initial state:
INROOM(IT, R1)
INROOM(B1,R2)
CONNECTS(D1, R1, R2)
CONNECTS(D1, R2, R1)
STATUS(D1,OPEN)
```

Representing Goal State



goal state:
INROOM(IT,R2)
INROOM(B1,R2)
CONNECTS(D1, R1, R2)
CONNECTS(D1, R2, R1)
STATUS(D1,OPEN)

initial state:
INROOM(IT, R1)
INROOM(B1,R2)
CONNECTS(D1, R1, R2)
CONNECTS(D1, R2, R1)
STATUS(D1,OPEN)

The “difference” table

operator	preconditions	add-list	delete-list
OP1: GOTODOOR(IT,dx)	INROOM(IT,rk) CONNECT(dx,rk,rm)	NEXTTO(IT,dx)	
OP2: GOTHRUDOOR(IT,dx)	CONNECT(dx,rk,rm) NEXTTO(IT,dx) STATUS(dx, OPEN) INROOM(IT,rk)	INROOM(IT,rm)	INROOM(IT,rk)

Logical Difference

goal state:
INROOM(IT,R2)
INROOM(B1,R2)
CONNECTS(D1, R1, R2)
CONNECTS(D1, R2, R1)
STATUS(D1,OPEN)

—

initial state:
INROOM(IT, R1)
INROOM(B1,R2)
CONNECTS(D1, R1, R2)
CONNECTS(D1, R2, R1)
STATUS(D1,OPEN)

=

¬INROOM(IT, R2)

or

INROOM(IT, R2)=FALSE

Eliminating the Difference

operator	preconditions	add-list	delete-list
OP1: GOTODOOR(IT,dx)	INROOM(IT,rk) CONNECT(dx,rk,rm)	NEXTTO(IT,dx)	
OP2: GOTHRUDOOR(IT,dx)	CONNECT(dx,rk,rm) NEXTTO(IT,dx) STATUS(dx, OPEN) INROOM(IT,rk)	INROOM(IT,rm)	INROOM(IT,rk)

INROOM(IT, R2)=FALSE

Eliminating the Difference

operator	preconditions	add-list	delete-list
OP1: GOTODOOR(IT,dx)	INROOM(IT,rk) CONNECT(dx,rk,rm)	NEXTTO(IT,dx)	
OP2: GOTHRUDOOR(IT,dx)	CONNECT(dx,rk,rm) NEXTTO(IT,dx) STATUS(dx, OPEN) INROOM(IT,rk)	INROOM(IT,rm)	INROOM(IT,rk)

INROOM(IT, R2)=FALSE

Eliminating the Difference

operator	preconditions	add-list	delete-list
OP1: GOTODOOR(IT,dx)	INROOM(IT,rk) CONNECT(dx,rk,rm)	NEXTTO(IT,dx)	
OP2: GOTHRUDOOR(IT,dx)	CONNECT(dx,rk,rm) NEXTTO(IT,dx) STATUS(dx, OPEN) INROOM(IT,rk)	INROOM(IT,rm)	INROOM(IT,rk)

INROOM(IT, R2)=FALSE

Discussion

- What are some limitations of planning with STRIPS?

- Where do the predicates, operators, etc. come from?

Then comes Rodney Brooks...

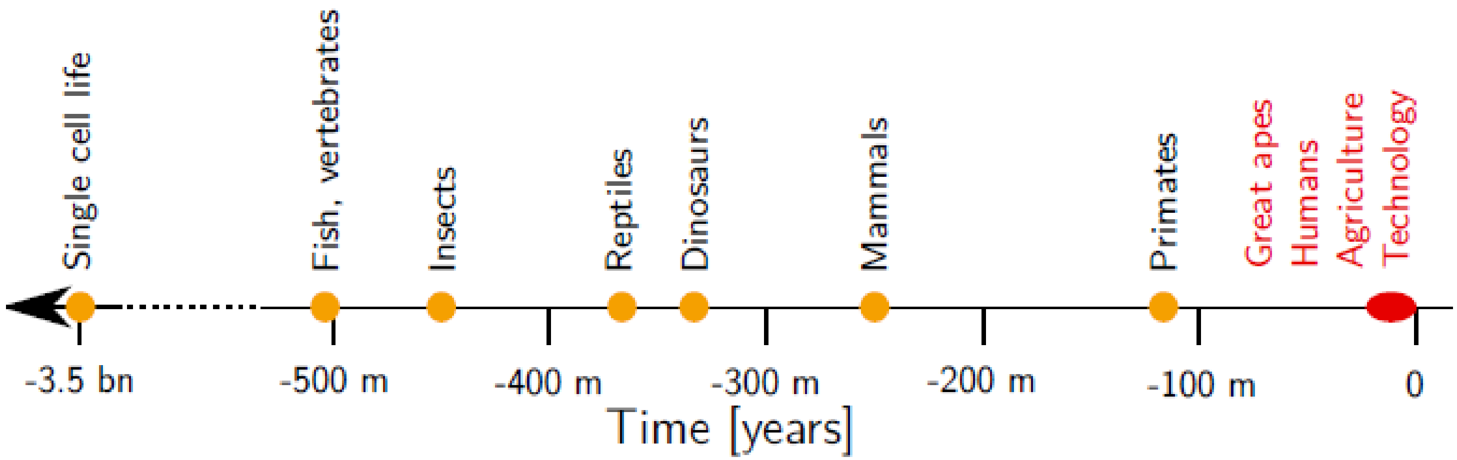
- ⋆ 1954 in Adelaide (Australia)
- Degree in mathematics and computer science
- Positions: CMU, MIT, Stanford
- Professorship: MIT, head of AI Lab
- Companies: iRobot, Heartland robotics, ...
- Contributions: Behavior-based AI, robotics, ...
- Several awards
- Tons of papers

[I1]

GOFAI

- **GOFAI**: good old-fashioned artificial intelligence
- Typically implemented as a central planner operating on a set of symbols (predicates)
- **Tools**: logic, predicate logic, PROLOG, Search algorithms, etc.
- **Solution:** sense → model → plan → act

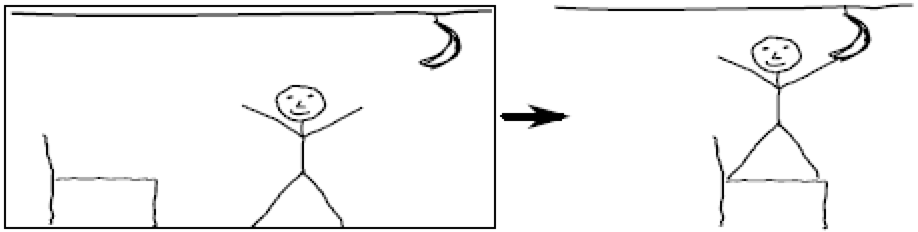
Brooks' opinion: GOFAI failed



Conclusion:

- Complex/intelligent skills appear simple, once the prerequisites are available
- Skills: problem-solving behavior, language, expert knowledge, reasoning
- Prerequisites: mobility, acute sensing, survival and reproduction in dynamic environments

Abstraction is a dangerous weapon



GOFAI:

- Requires abstraction
- Handcrafted decomposition: PERSON, CHAIR, BANANA
- Basic concepts / representation
- Planner (search algorithm)

Reality:

- Intuitive interpretation & solution

Conclusion:

- Over-simplification of GOFAI
- Intelligence includes interpretation & abstraction

Toy worlds vs. Real worlds

GOFAI:

- Use of toy worlds
- Human interpreter for abstraction/simplification
- Static (prepared) environments
- Planning/perception with limited "field of view"

Behavior-based AI:

- Real worlds
- No human assistance, robot should operate on its own
- Dynamic environments without simplifications
- Full bandwidth of intelligent behavior

Conclusion:

Autonomous mobile robots in real-world
$\Rightarrow$ artificial intelligent systems

Toy worlds vs. Real worlds

GOFAI:

- Limited applicability
  small subset of real-world
- Top-down approach
- Engineering decomposition:
  solution $\rightarrow$ decomposition
- Central locus of control

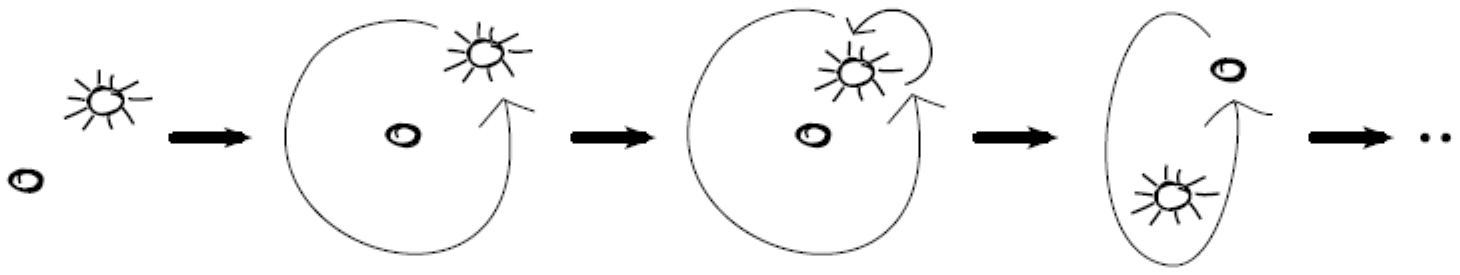
Behavior-based AI:

- Vast repertoire of capabilities, experience, and knowledge
- Bottom-up approach
- Incremental decomposition:
  decomposition $\rightsquigarrow$ solution
- No central control instance

Conclusions:

Intelligent systems as composition of independent sub-systems

Brook's opinion: GOFAI failed



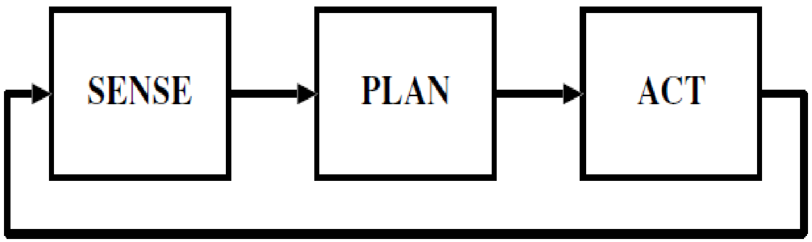
Conclusion:

This cycle will also happen to AI

Overall conclusions:

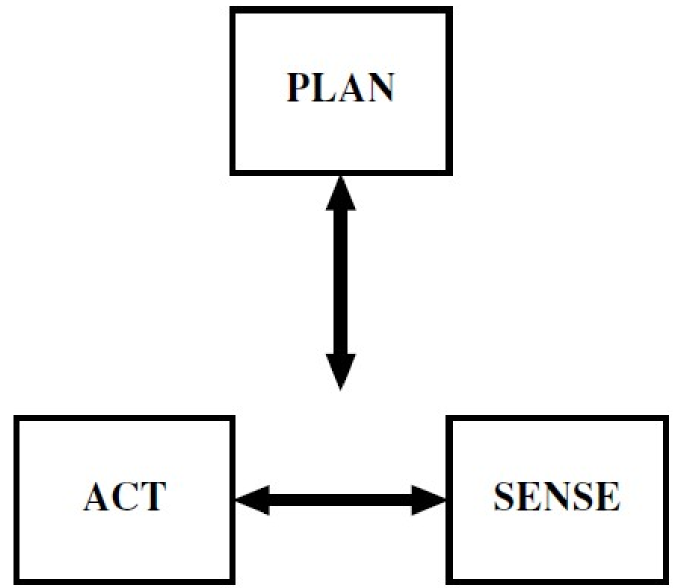
- GOFAI: Not sufficient to explain intelligent behavior
- Hindsight: current (1986) AI work will appear useless
- Change of paradigm: *"towards process, away from state"*

Alternatives to Sense-Plan-Act



Sense-Plan-Act

Reactive



Hybrid

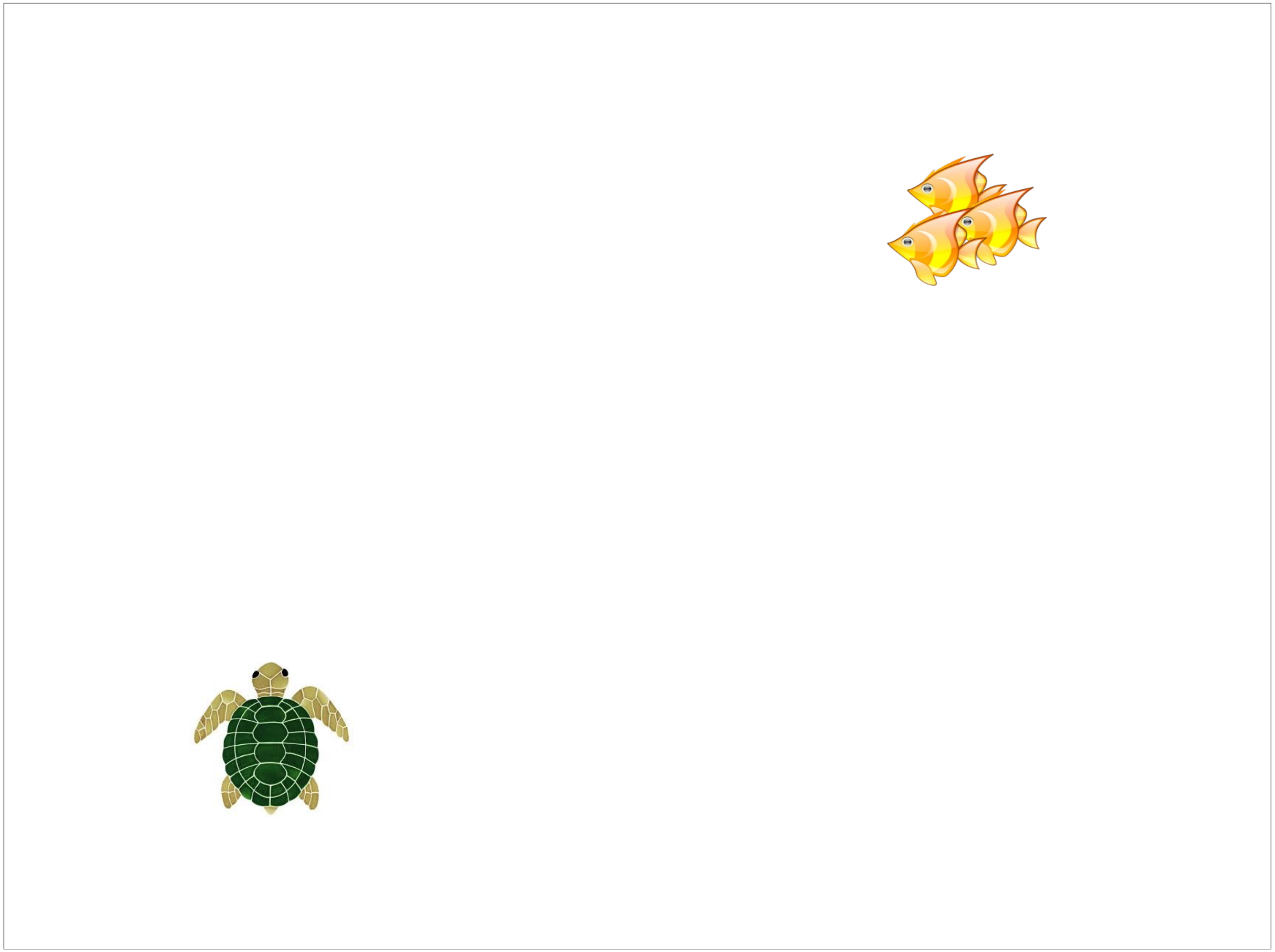
Reactive Paradigm

ROBOT PRIMITIVES	INPUT	OUTPUT
SENSE	Sensor data	Sensed information
[redacted]		
ACT	Sensed information [redacted]	Actuator commands

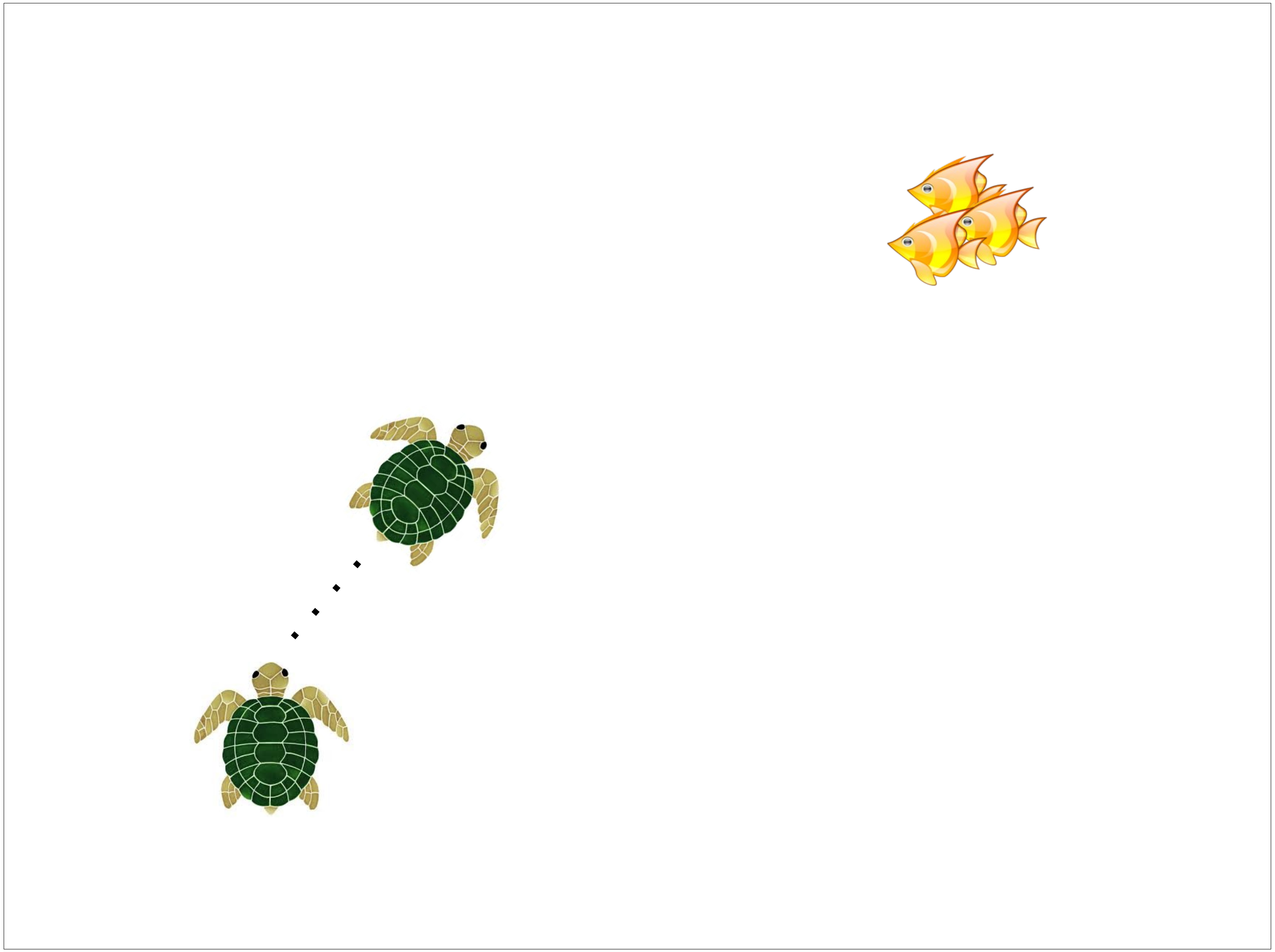
Figure I.5 The reactive paradigm.

Reactive Paradigm Example

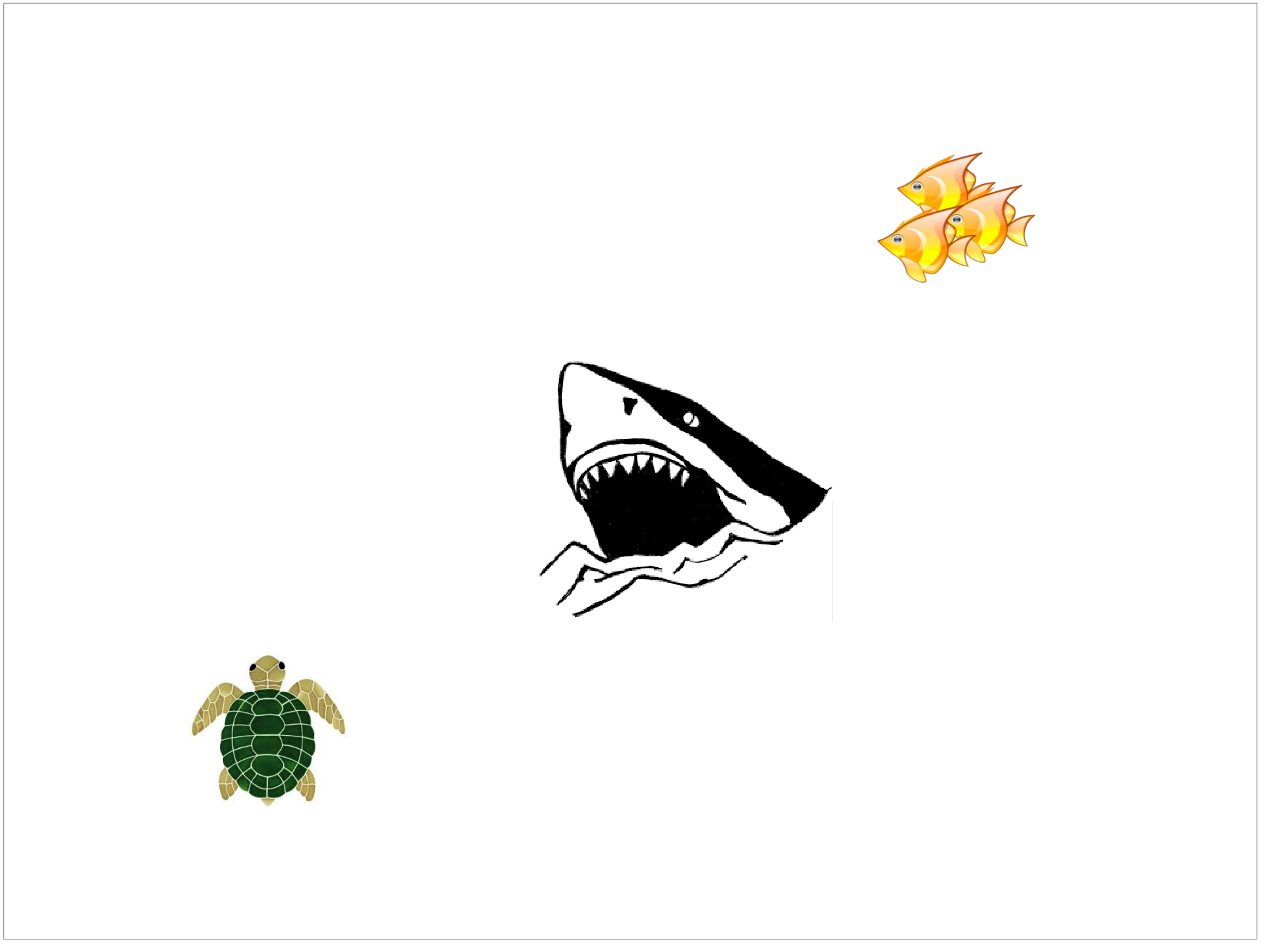
Reactive Paradigm Example



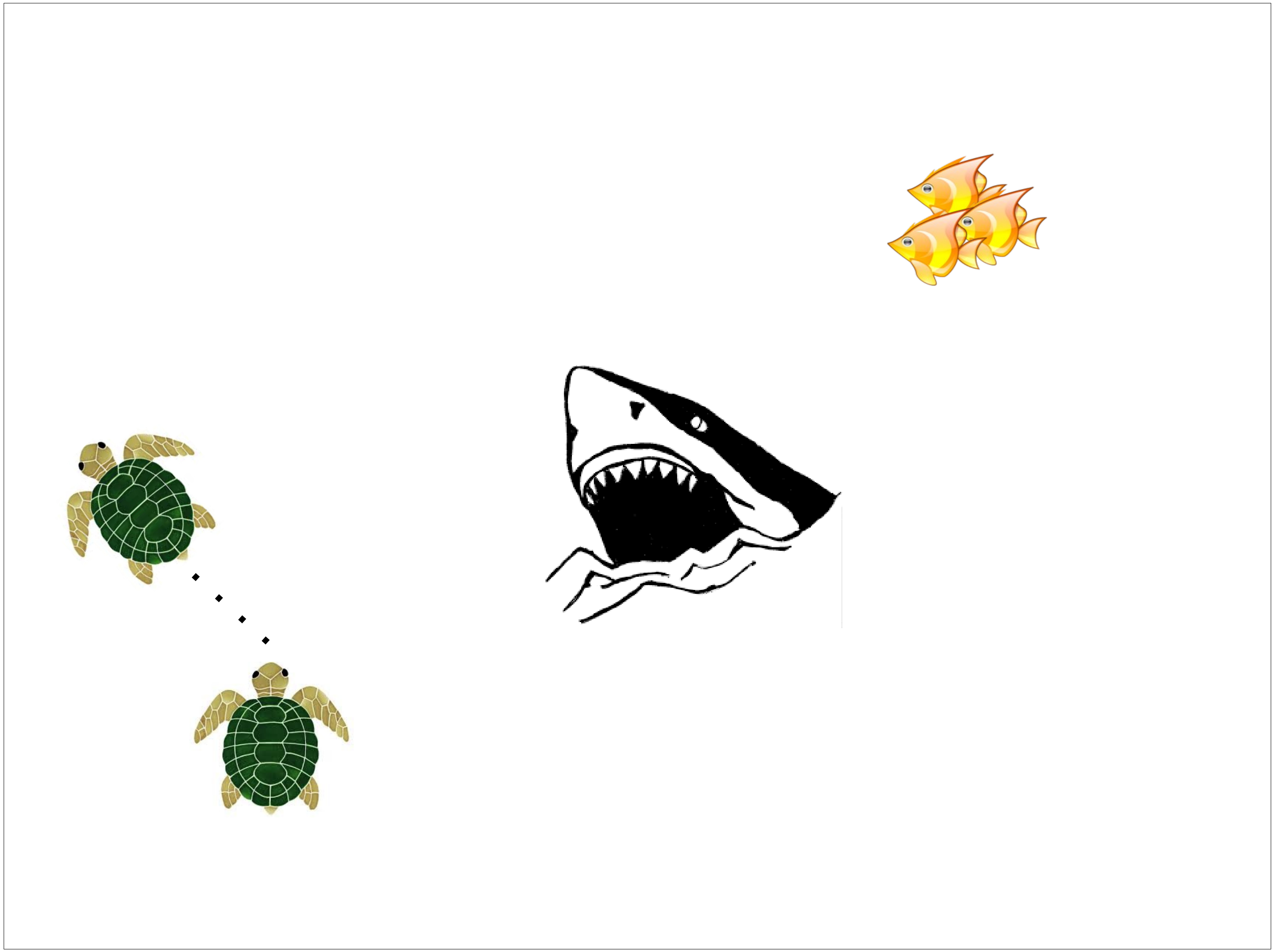
Reactive Paradigm Example



Reactive Paradigm Example



Reactive Paradigm Example



The Hybrid Paradigm

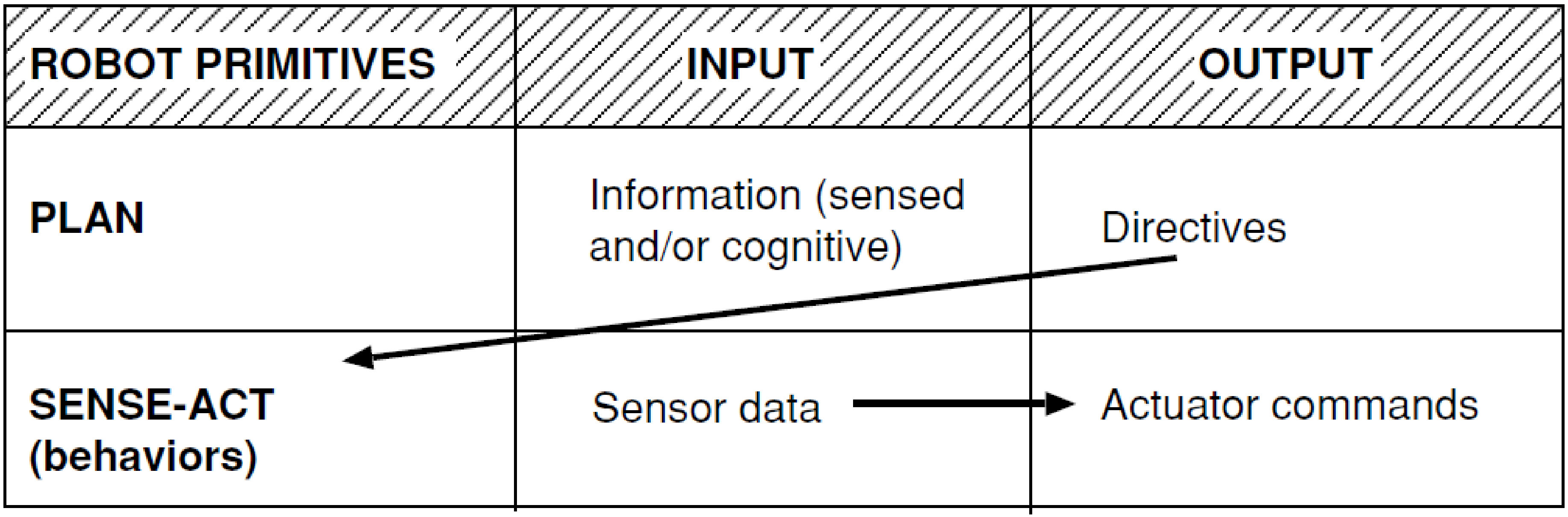
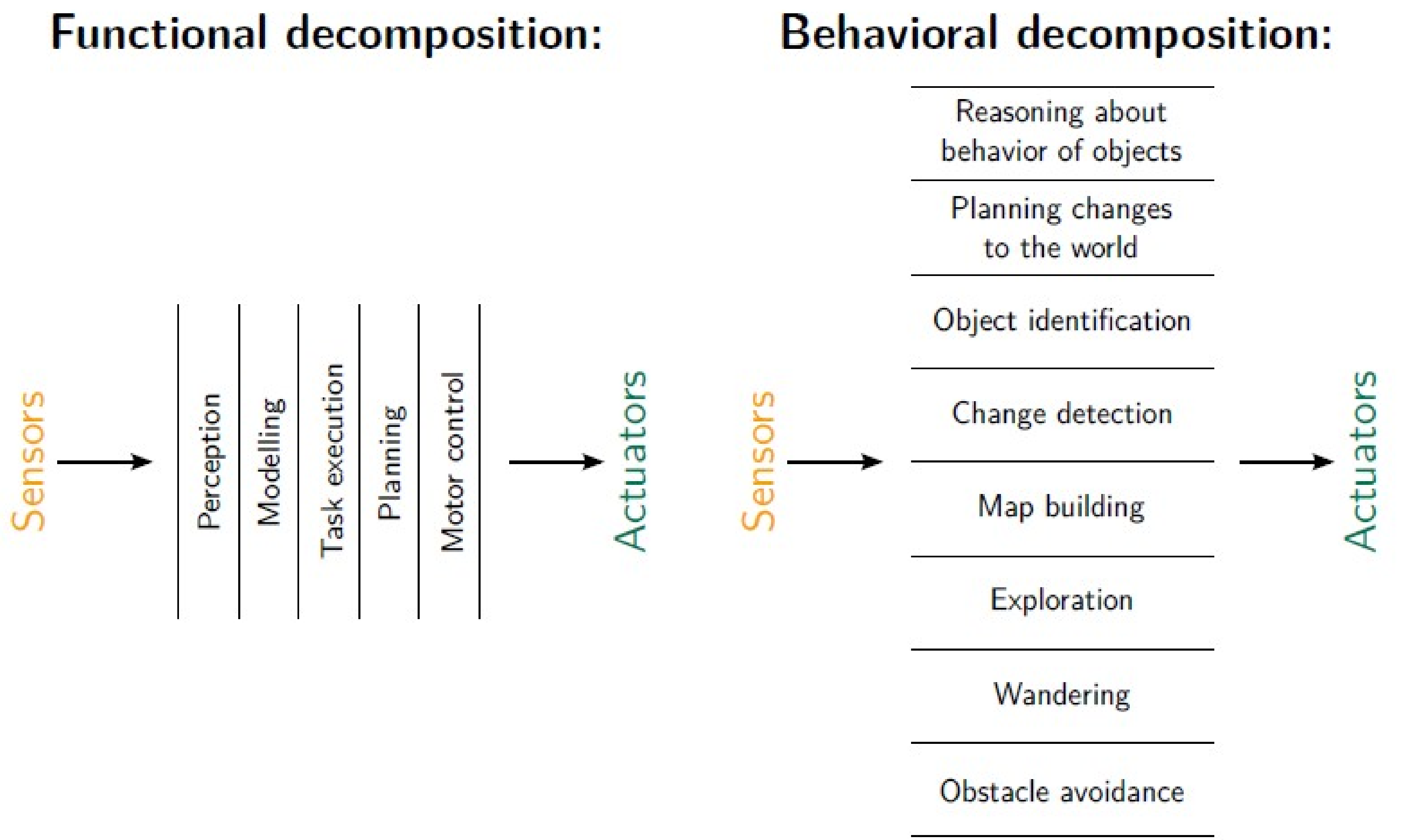


Figure I.6 The hybrid deliberative/reactive paradigm.

Functional vs. Behavioral Decomposition



Words of Wisdom

"When we examine very simple level intelligence, we find that explicit representations and models of the world get in the way."

"It turns out to be better to use the world as its own model."

"Representation is the wrong unit of abstaction in building the bulkiest parts of intelligent systems."

Where is Brooks now?

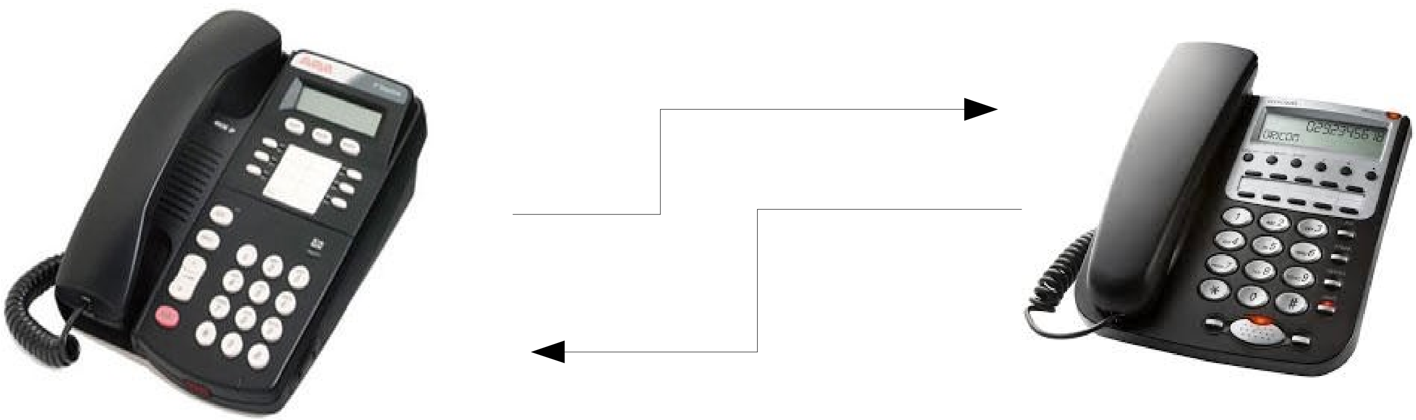
Credits

- “Introduction to AI Robotics” by Robin Murphy
- Slides by Lorenz Hillen from Universität Bielefeld

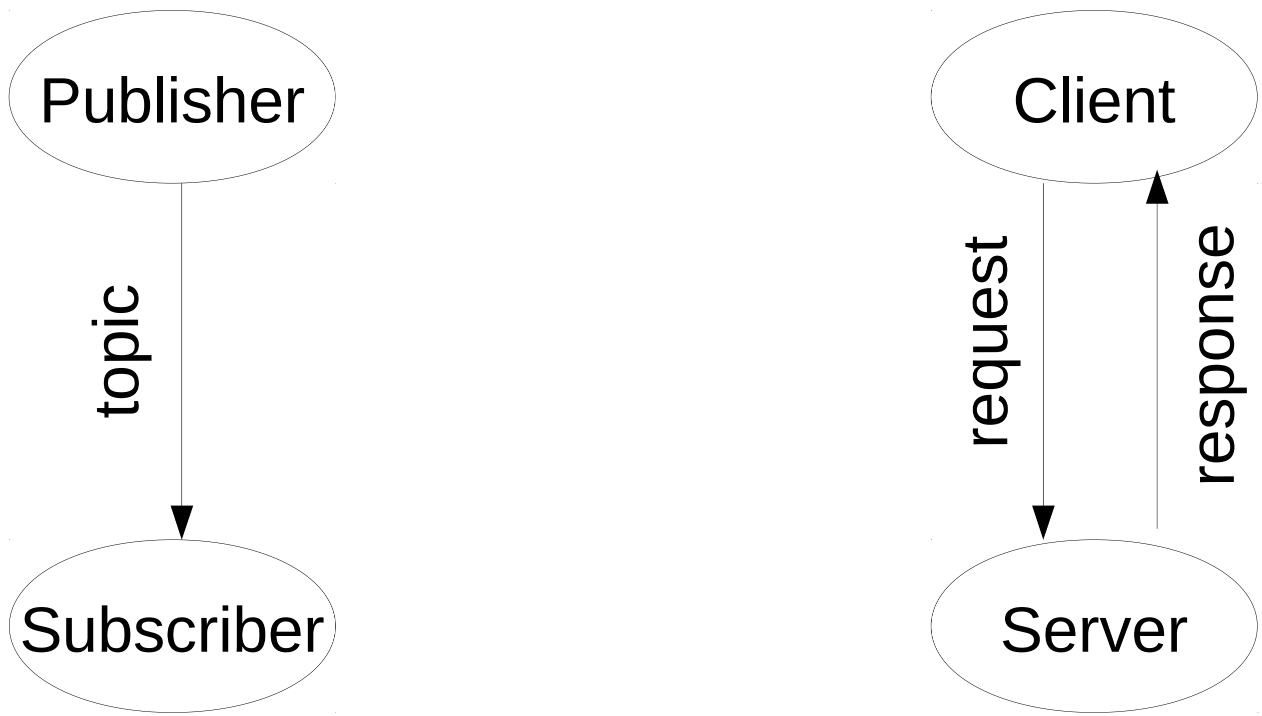
ROS Services

Messages vs. Services

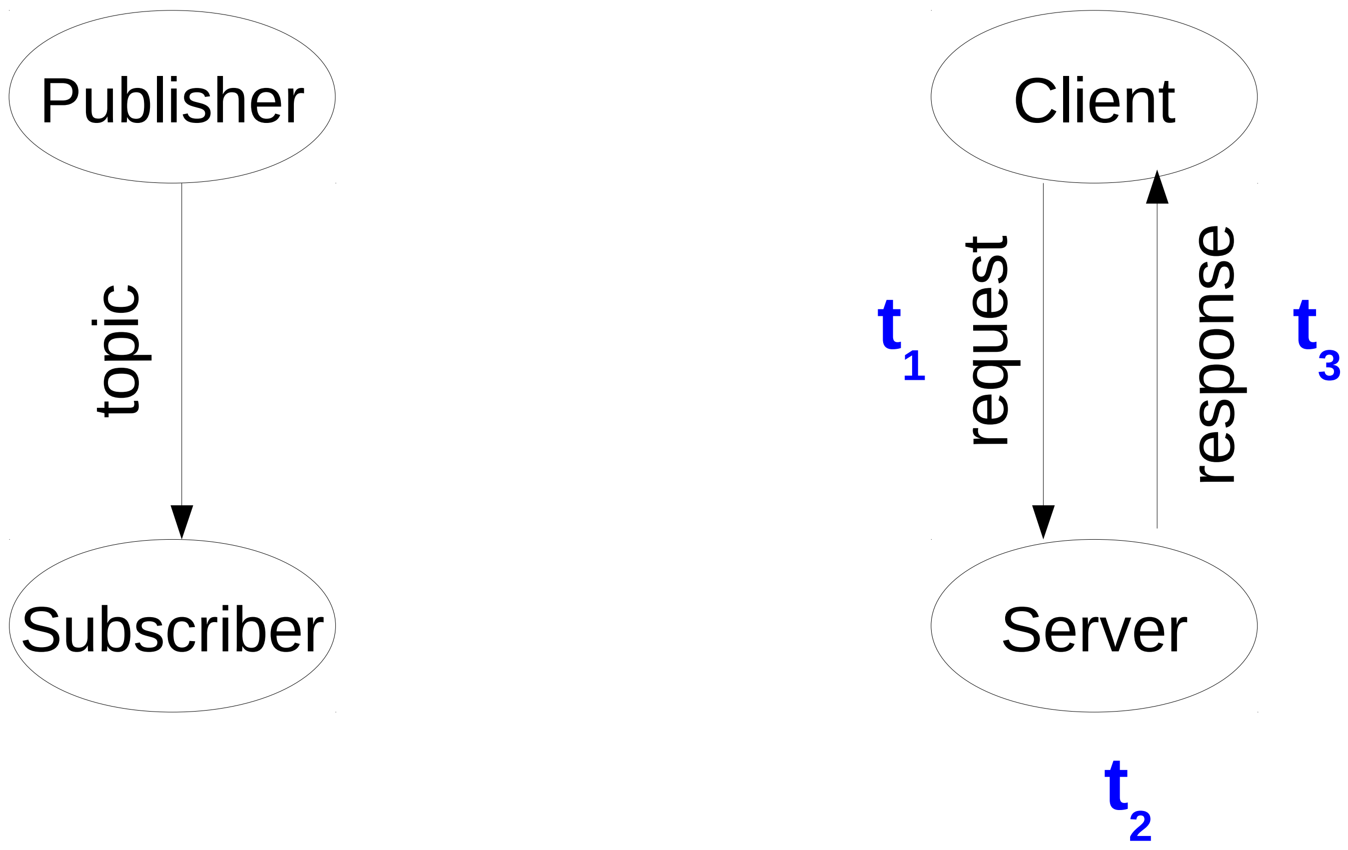
Messages vs. Services



Messages vs. Services



Messages vs. Services



Calling Services in ROS

1) From the command line:

rosservice call <service_name> <request>

2) From code

THE END