Principles of Autonomy and Decision Making

Brian Williams and
Nicholas Roy
16.410/16.413
September 8th, 2004

Today’s Assignment

- Read Chapters 1 and 2 of AIMA
  - “Artificial Intelligence: A Modern Approach” by Stuart Russell and Peter Norvig
  - 2nd Edition (not 1st Edition!!)
  - AIMA is available at the Coop

Outline

- Objectives
- Agents and Their Building Blocks
- Agent Paradigms
- Principles for Building Agents:
  - Modeling Formalisms
  - Algorithmic Principles
- Building an Agent: Fall 03 Projects

Course Objective 1: Principles of Agents

16.410/13: To learn the modeling and algorithmic building blocks for creating reasoning and learning agents:

- To formulate reasoning problems.
- To describe, analyze and demonstrate reasoning algorithms.
- To model and encode knowledge used by reasoning algorithms.

Course Objective 2: Building Agents

16.413: To appreciate the challenges of building a state of the art autonomous explorer:

Fall 03, goals were:

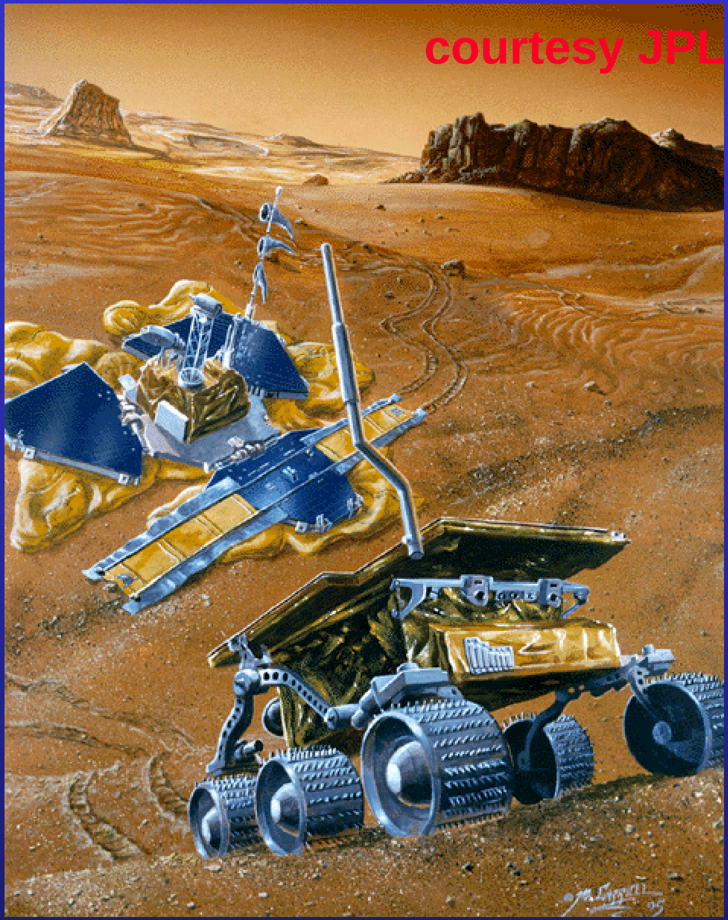
- To model and encode knowledge needed to solve a state of the art challenge.
- To work through the process of autonomy systems integration.
- To assess the promise, frustrations and challenges of using (b)leading art technologies.

Fall 04, stay tuned.

Outline

- Objectives
- Agents and Their Building Blocks
- Agent Paradigms
- Principles for Building Agents:
  - Modeling Formalisms
  - Algorithmic Principles
- Building an Agent: Fall 03 Projects

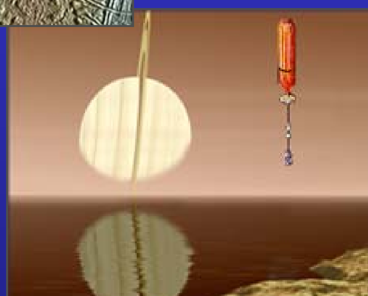
1. Mission-Oriented Agents



``Our vision in NASA is to open the Space Frontier . . . We must establish a virtual presence, in space, on planets, in aircraft and spacecraft.'' - Daniel S. Goldin, NASA Administrator, May 29, 1996

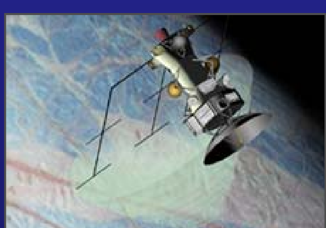
Inner and Outer Planets Missions

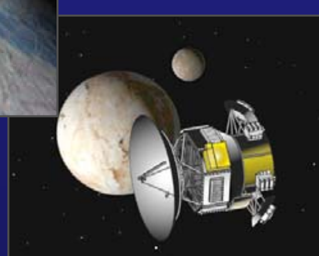
Europa Lander



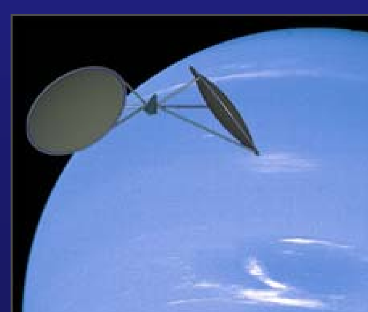
Titan Explorer

Europa Orbiter





Pluto/Kuiper Express

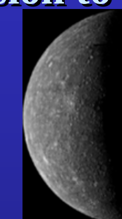


Neptune Orbiter

Comet Nucleus Sample Return

Venus Sample Return

MESSENGER mission to Mercury



Primitive Bodies Missions

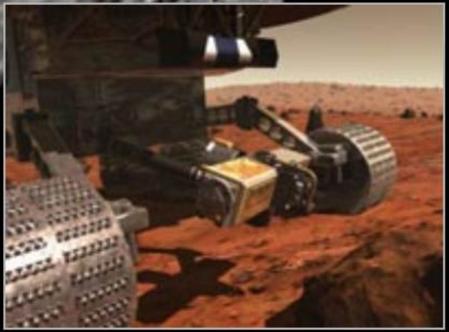
[redacted]

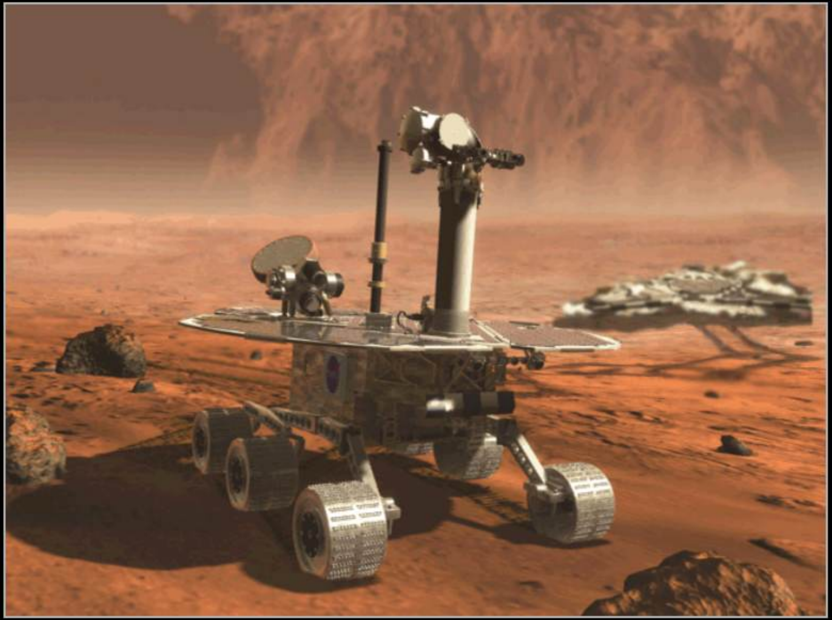
Courtesy of JPL

2003 Twin Mars Exploration Rovers

- Will learn about the climate on Mars and scout for regions where mineralogical evidence of water has been found.
- The rover twins will determine the geologic record of the landing site, what the planet's conditions were like when the Martian rocks and soils were formed, and help us learn about ancient water reservoirs.

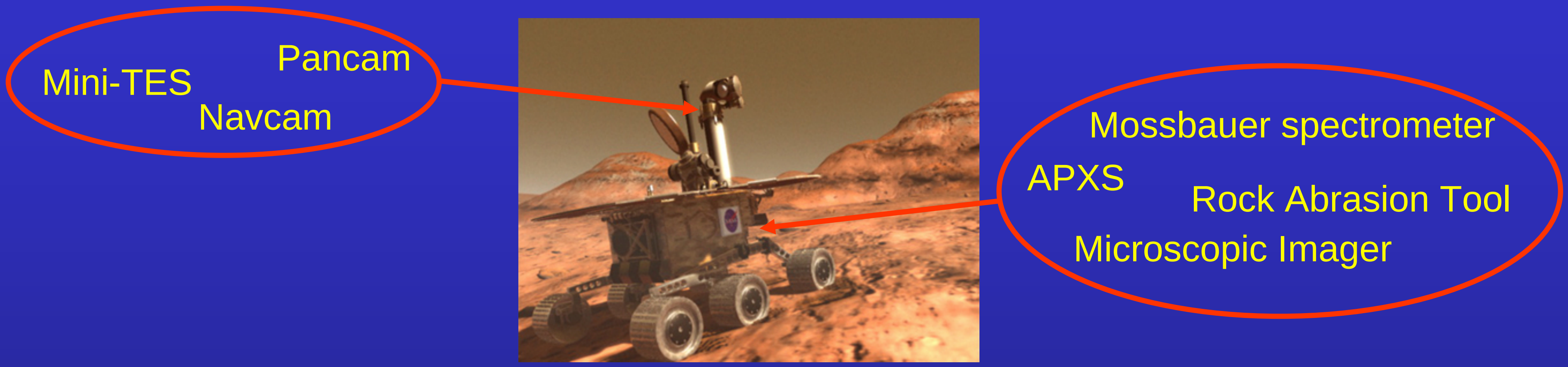
First microscopic view of Mars





Rover 1:	Launch:	May 30, 2003
	Landing:	January 4, 2004
Rover 2:	Launch:	June 27, 2003
	Landing:	January 25, 2004

Mars Exploration Rovers – Jan. 2004



Mission Objective: Learn about ancient water and climate on Mars.

- For each rover, analyze a total of 6-12 targets
  - Targets = natural rocks, abraded rocks, and soil
- Drive 200-1000 meters per rover
- Take 1-3 panoramas both with Pancam and mini-TES
- Take 5-15 daytime and 1-3 nighttime sky observations with mini-TES

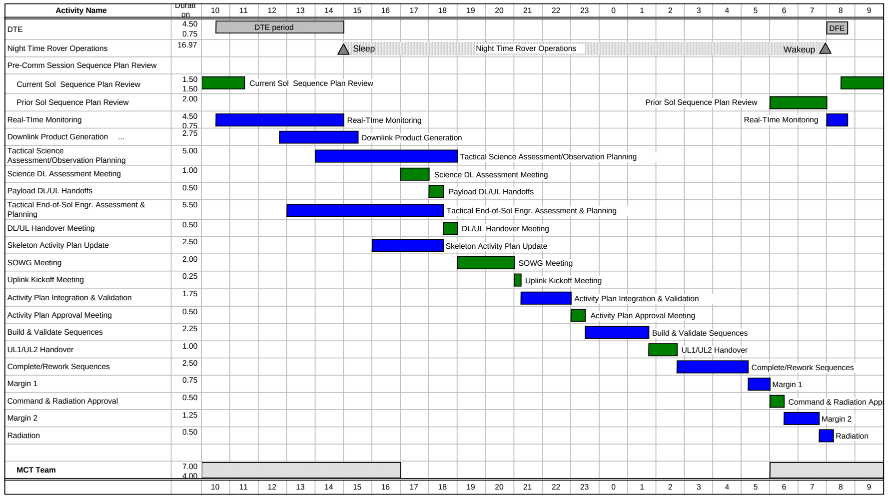
One day in the life of a Mars rover

Model-based Embedded & Robotic Systems

rover operation (7 hr)

tactical sci assess & obs planning (5 hr)

tactical end-of-sol eng assess (5 hr)



SOWG mtg (2 hr)

activity integration & validation (3.5 hr)

sequence development (5.5 hr)

sequence integration & validation (4 hr)

Agent Building Blocks

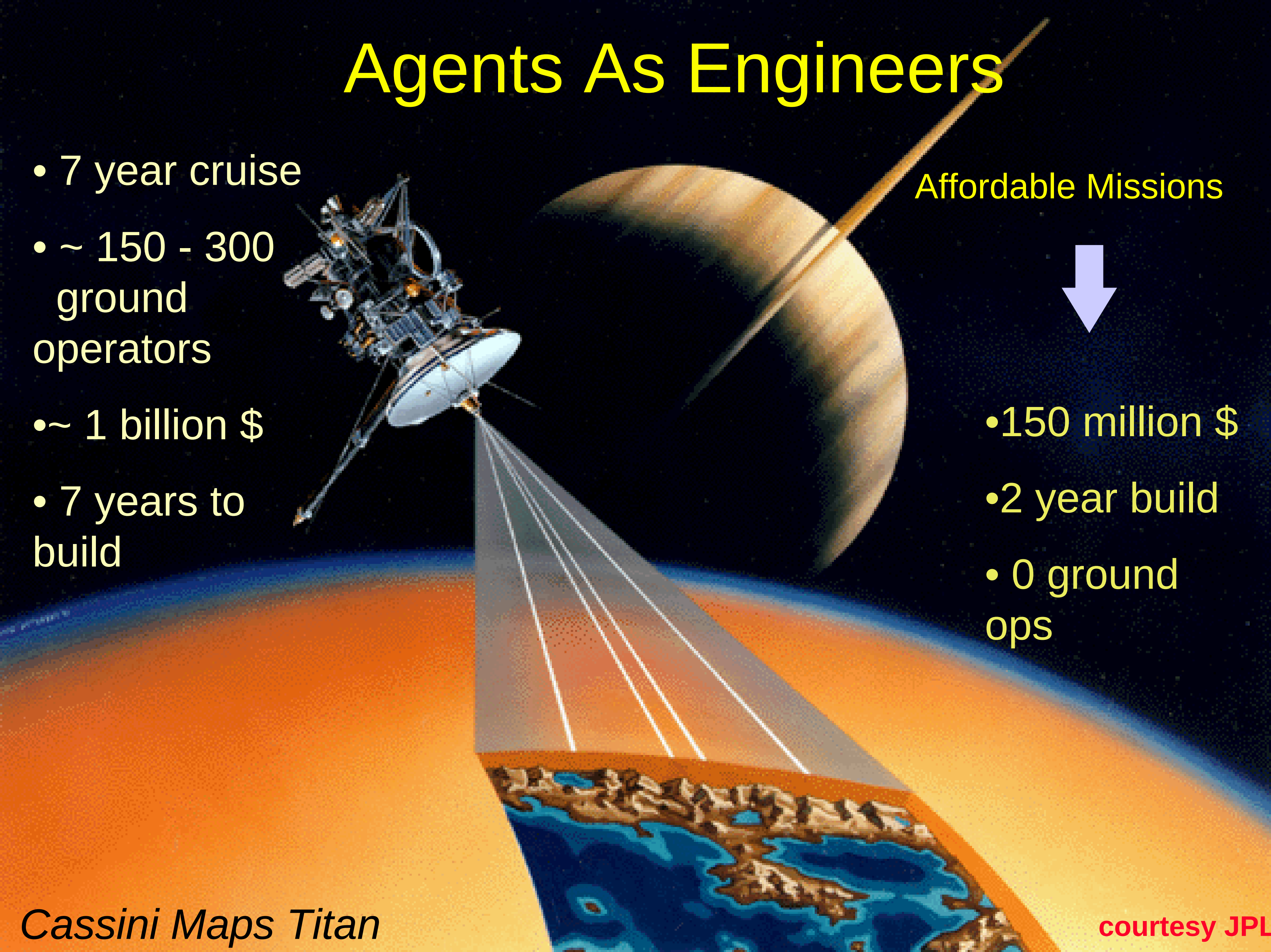
- Activity Planning
- Execution/Monitoring

Agents As Engineers

- 7 year cruise
- ~ 150 - 300 ground operators
- ~ 1 billion $
- 7 years to build

Affordable Missions

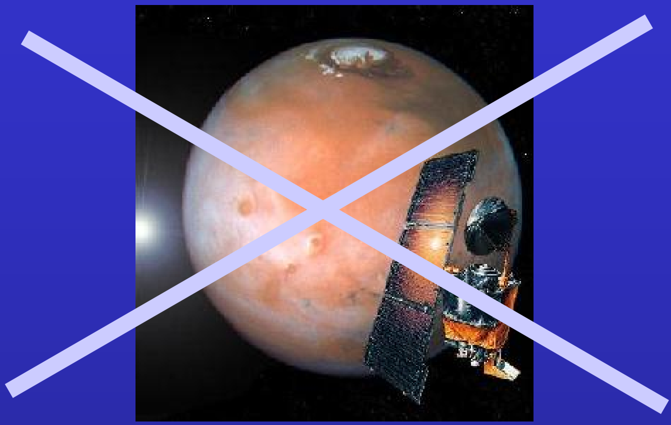
- 150 million $
- 2 year build
- 0 ground ops



Cassini Maps Titan

courtesy JPL

Four launches in 7 months



Mars Climate Orbiter: 12/11/98

Mars Polar Lander: 1/3/9

Stardust: 2/7/99

QuickSCAT: 6/19/98

courtesy of J

Mars Polar Lander

Spacecraft require a good physical commonsense…

Launch: 1/3/99

courtesy of JPL

Traditional spacecraft commanding

GS,SITURN,490UA,BOTH,96-355/03:42:00.000;

```
CMD,7GYON,  490UA412A4A,BOTH,     96-355/03:47:00:000, ON;
CMD,7MODE,  490UA412A4B,BOTH,     96-355/03:47:02:000, INT;
CMD,6SVPM,  490UA412A6A,BOTH,     96-355/03:48:30:000, 2;
CMD,7ALRT,  490UA412A4C,BOTH,     96-355/03:50:32:000, 6;
CMD,7SAFE,  490UA412A4D,BOTH,     96-355/03:52:00:000, UNSTOW;
CMD,6ASSAN, 490UA412A6B,BOTH,     96-355/03:56:08:000, GV,153,IMM,231,
                                                       GV,153;
CMD,7VECT,  490UA412A4E,BOTH,     96-355/03:56:10.000, 0,191.5,6.5,
                                                       0.0,0.0,0.0,
                                                       96-350/
                                                       00:00:00.000,MVR;
SEB,SCTEST, 490UA412A23A,BOTH,    96-355/03:56:12.000, SYS1,NPERR;
CMD,7TURN,  490UA412A4F,BOTH,     96-355/03:56:14.000, 1,MVR;
MISC,NOTE,  490UA412A99A,,        96-355/04:00:00.000, ,START OF TURN;,
CMD,7STAR,  490UA412A406A4A,BOTH  96-355/04:00:02.000, 7,1701,
                                                       278.813999,38.74;
CMD,7STAR,  490UA412A406A4B,BOTH, 96-355/04:00:04.000, 8,350,120.455999,
                                                       -39.8612;
CMD,7STAR,  490UA412A406A4C,BOTH, 96-355/04:00:06.000, 9,875,114.162,
                                                       5.341;
CMD,7STAR,  490UA412A406A4D,BOTH, 96-355/04:00:08.000, 10,159,27.239,
                                                       89.028999;
CMD,7STAR,  490UA412A406A4E,BOTH, 96-355/04:00:10.000, 11,0,0.0,0.0;
CMD,7STAR,  490UA412A406A4F,BOTH, 96-355/04:00:12.000, 21,0,0.0,0.0;
```

Whats a better paradigm?

Houston, we have a problem ...


courtesy of NASA

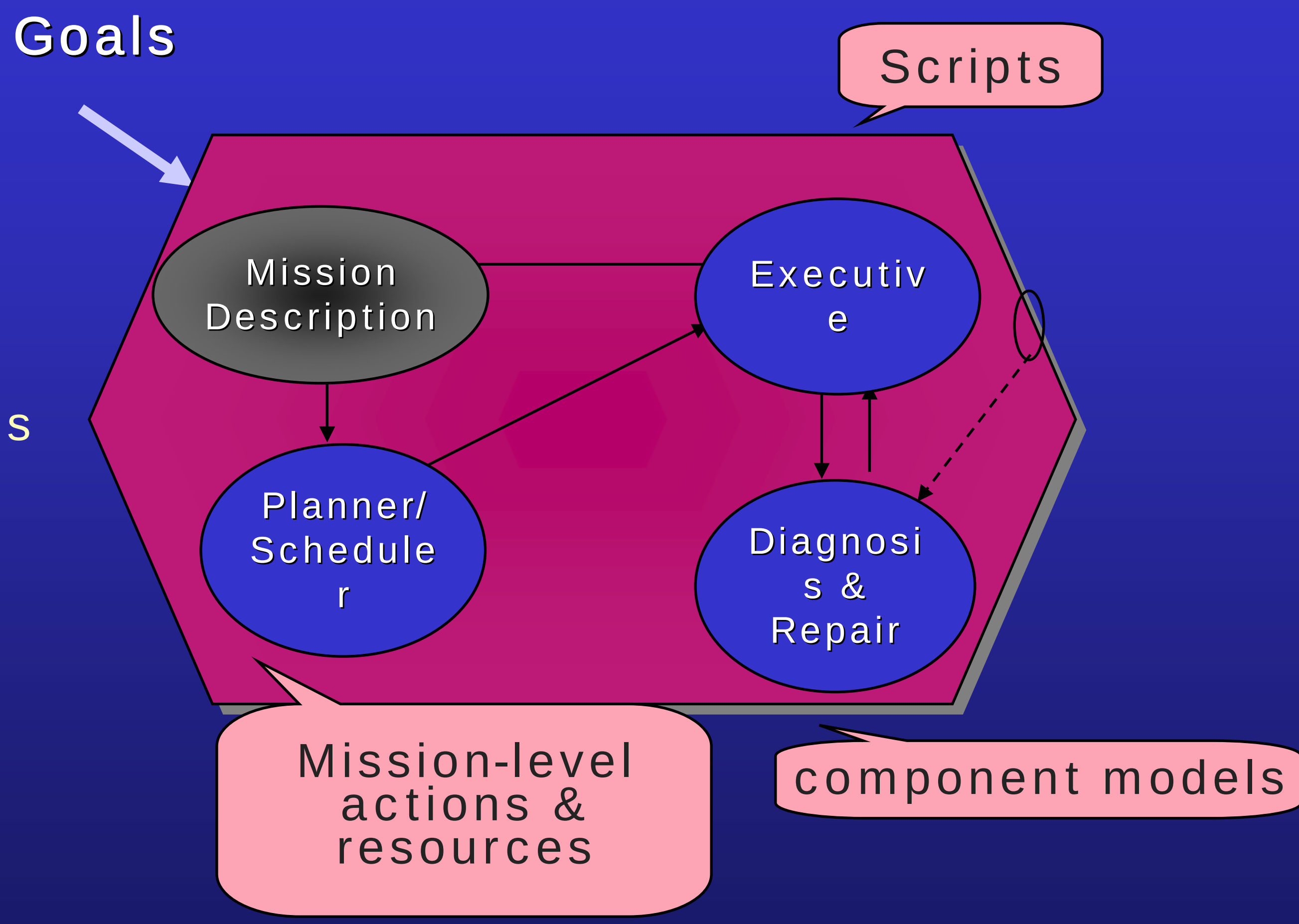
- Quintuple fault occurs (three shorts, tank-line and pressure jacket burst, panel flies off)
  - Diagnosis.
- Mattingly works in ground simulator to identify new sequence handling severe power limitations.
  - Planning & Resource Allocation
- Mattingly identifies novel reconfiguration, exploiting LEM batteries for power.
  - Reconfiguration and Repair
- Swaggert & Lovell work on Apollo 13 emergency rig lithium hydroxide unit.
  - Execution

Remote Agent on Deep Space 1

Started: January 1996
Launch: Fall 1998

Remote Agent

- Goal-directed
- First time correct
    - projective
    - reactive
- Commonsense models
- Heavily deductive



Remote Agent Experiment

See rax.arc.nasa.gov

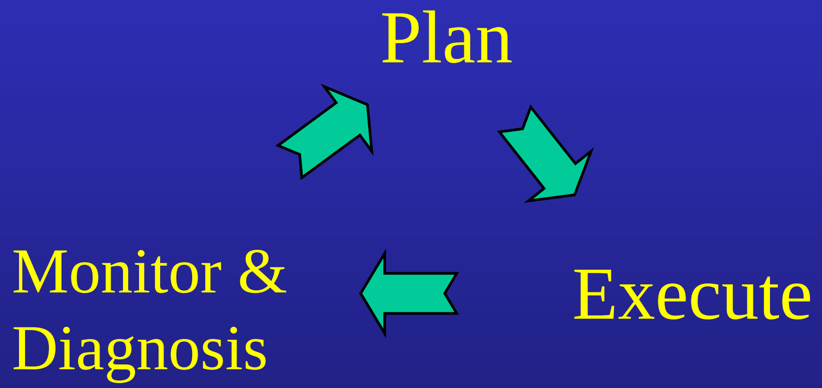
May 17-18th experiment

- Generate plan for course correction and thrust
- Diagnose camera as stuck on
  - Power constraints violated, abort current plan and replan
- Perform optical navigation
- Perform ion propulsion thrust

May 21th experiment.

- Diagnose faulty device and
  - Repair by issuing reset.
- Diagnose switch sensor failure.
  - Determine harmless, and continue plan.
- Diagnose thruster stuck closed and
  - Repair by switching to alternate method of thrusting.
- Back to back planning

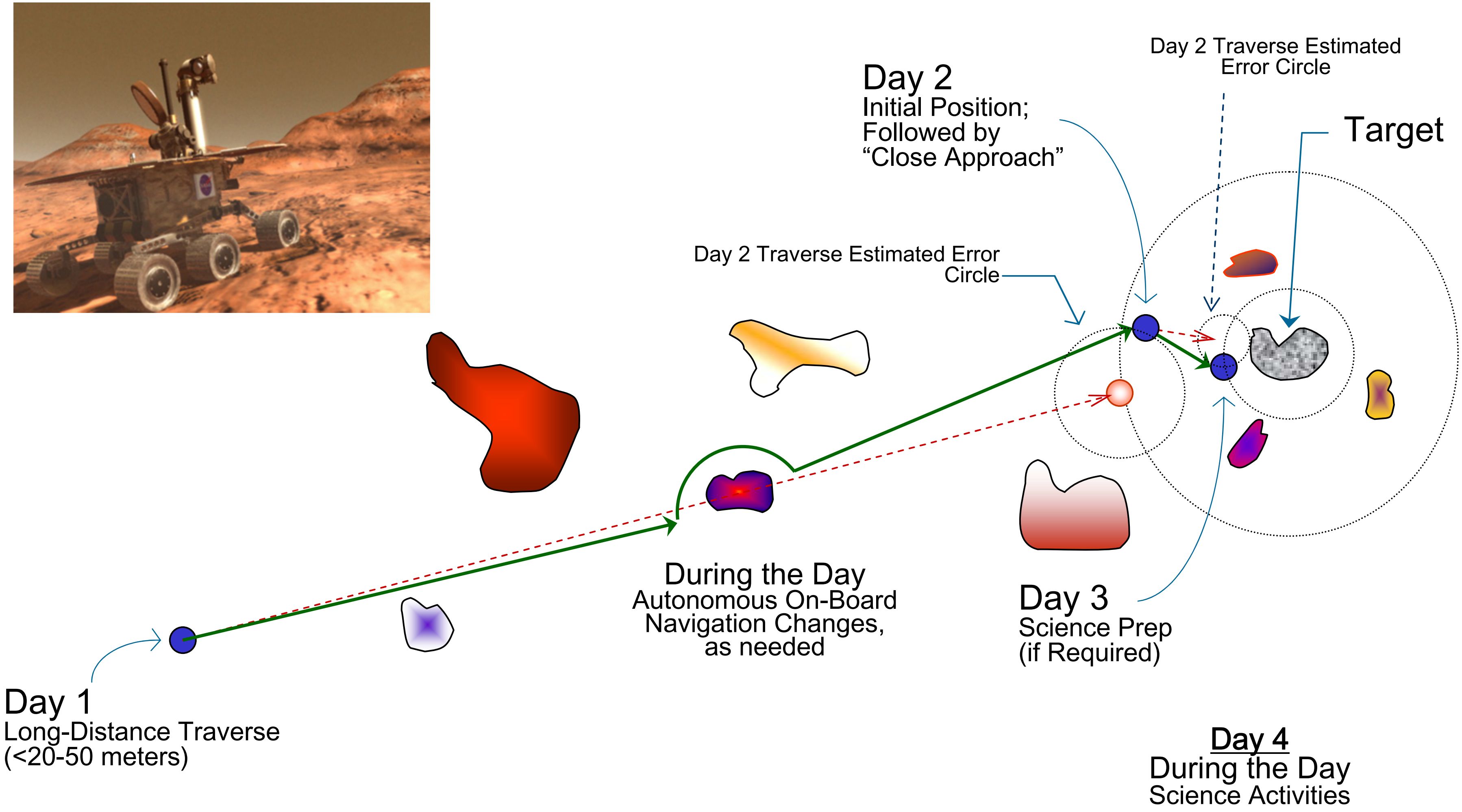
Course Objective 2



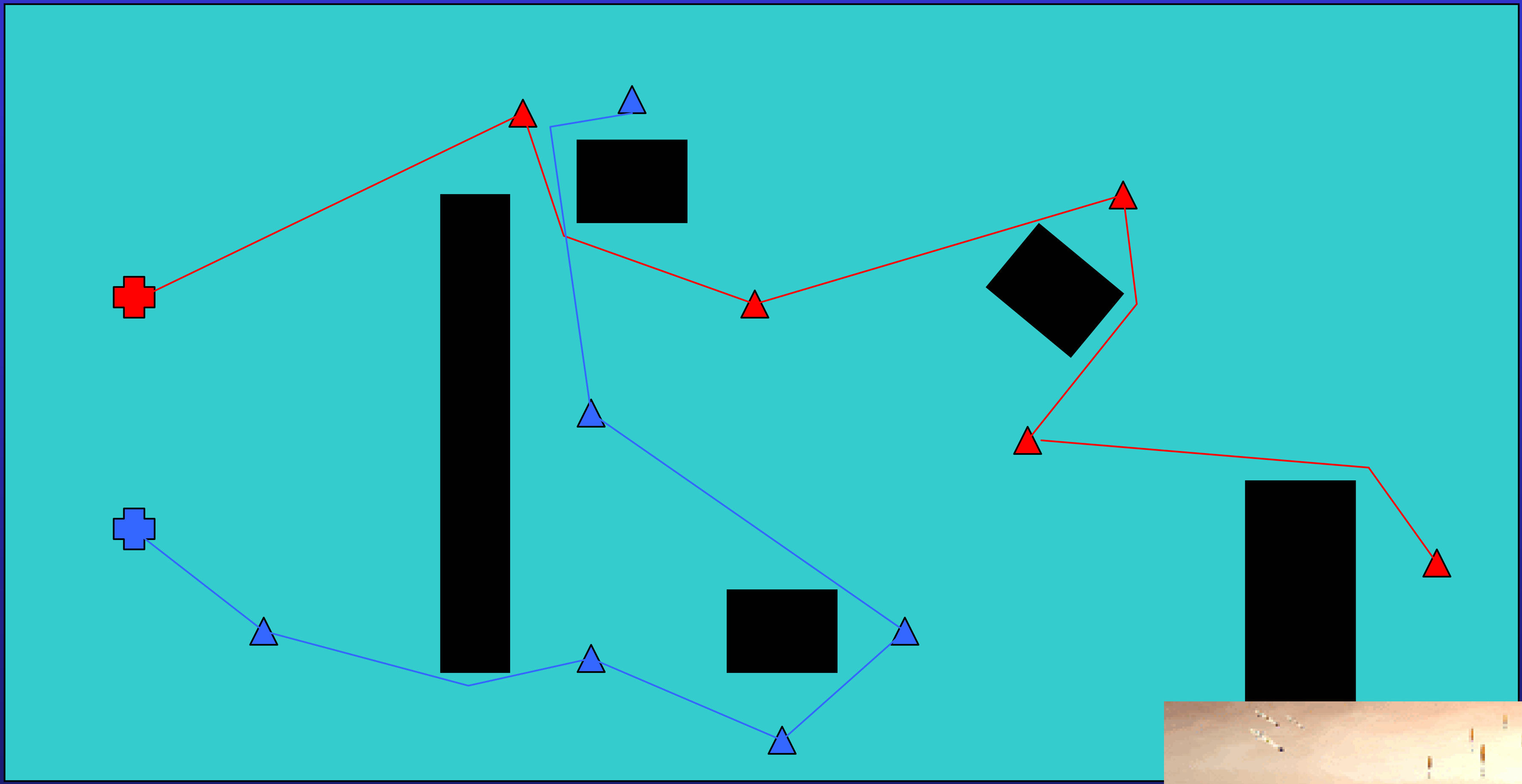
Agent Building Blocks

- Activity Planning
- Execution/Monitoring
- Diagnosis
- Repair
- Scheduling
- Resource Allocation

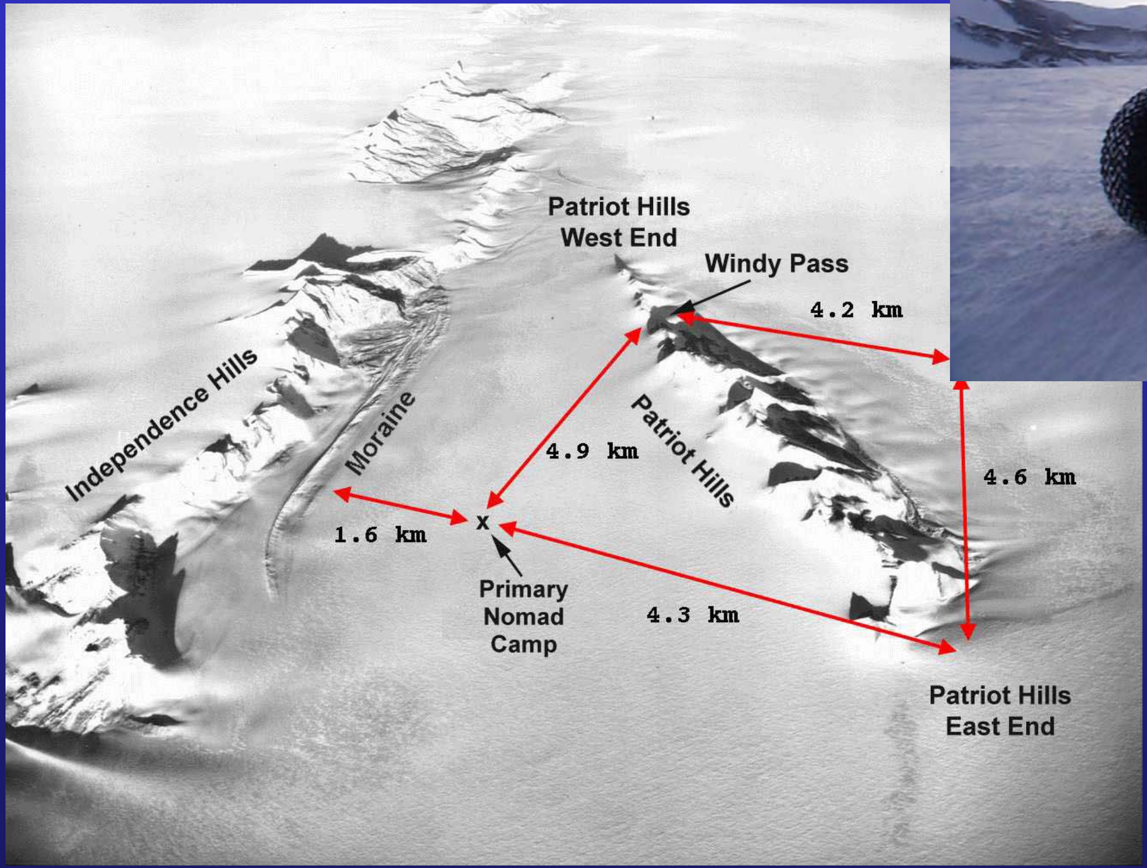
2. Mobile Agents



Multi-Vehicle Path Planning



Nomad Antarctic Explorer

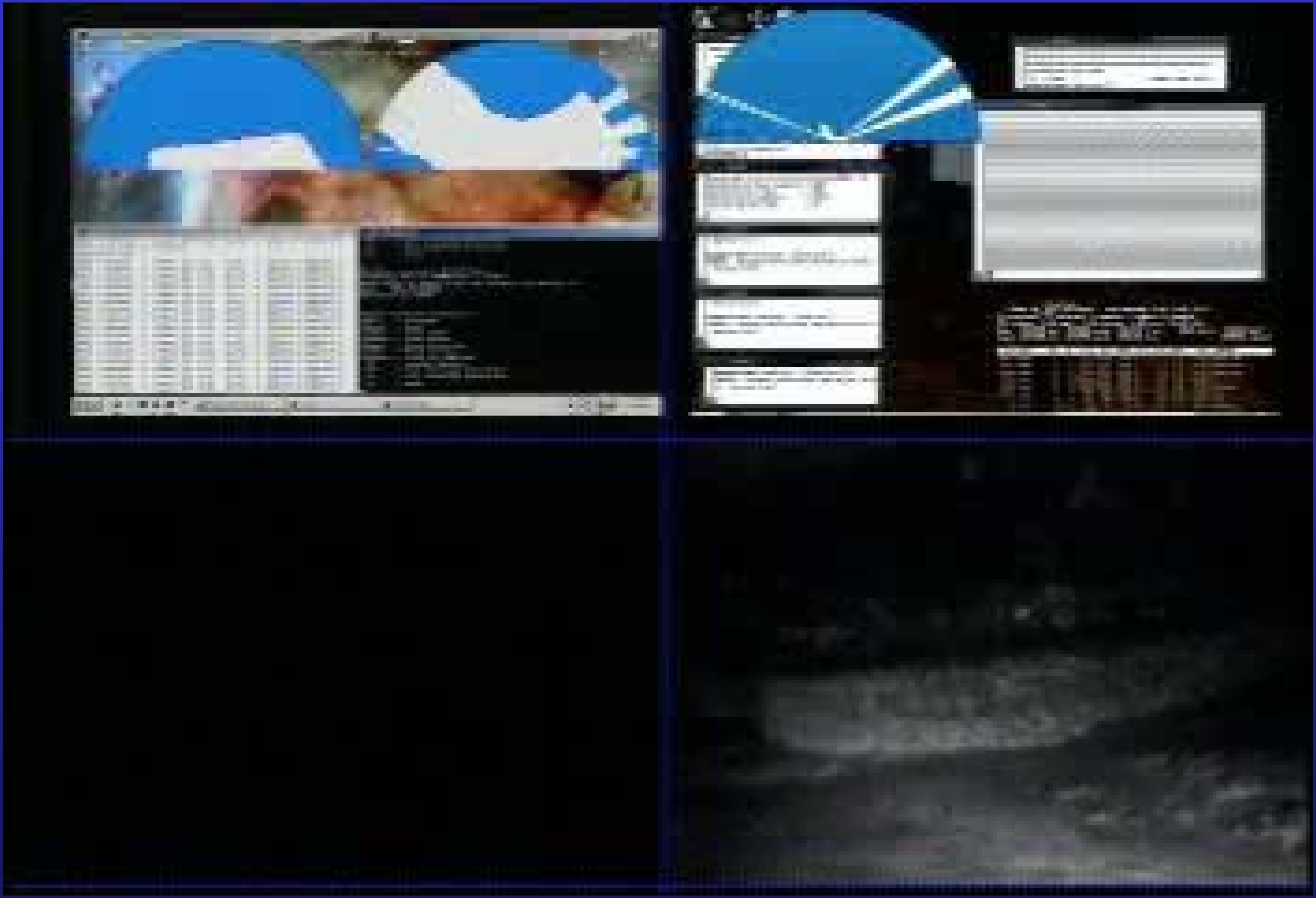


Of 100 rock samples, Nomad correctly classified 3 as meteorites and incorrectly classified a 4th.

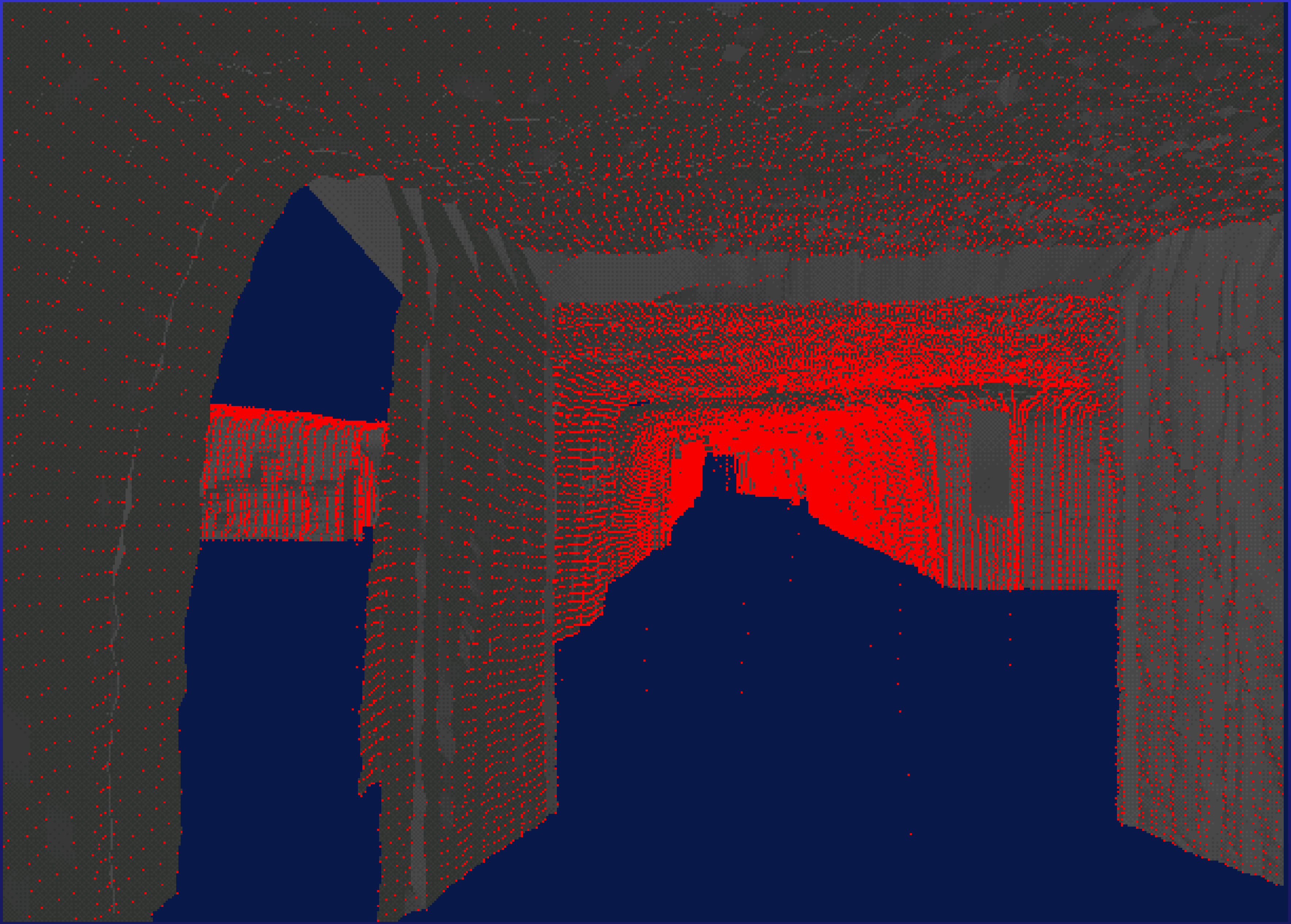
Images courtesy of D. Apostopolous, CMU

Groundhog

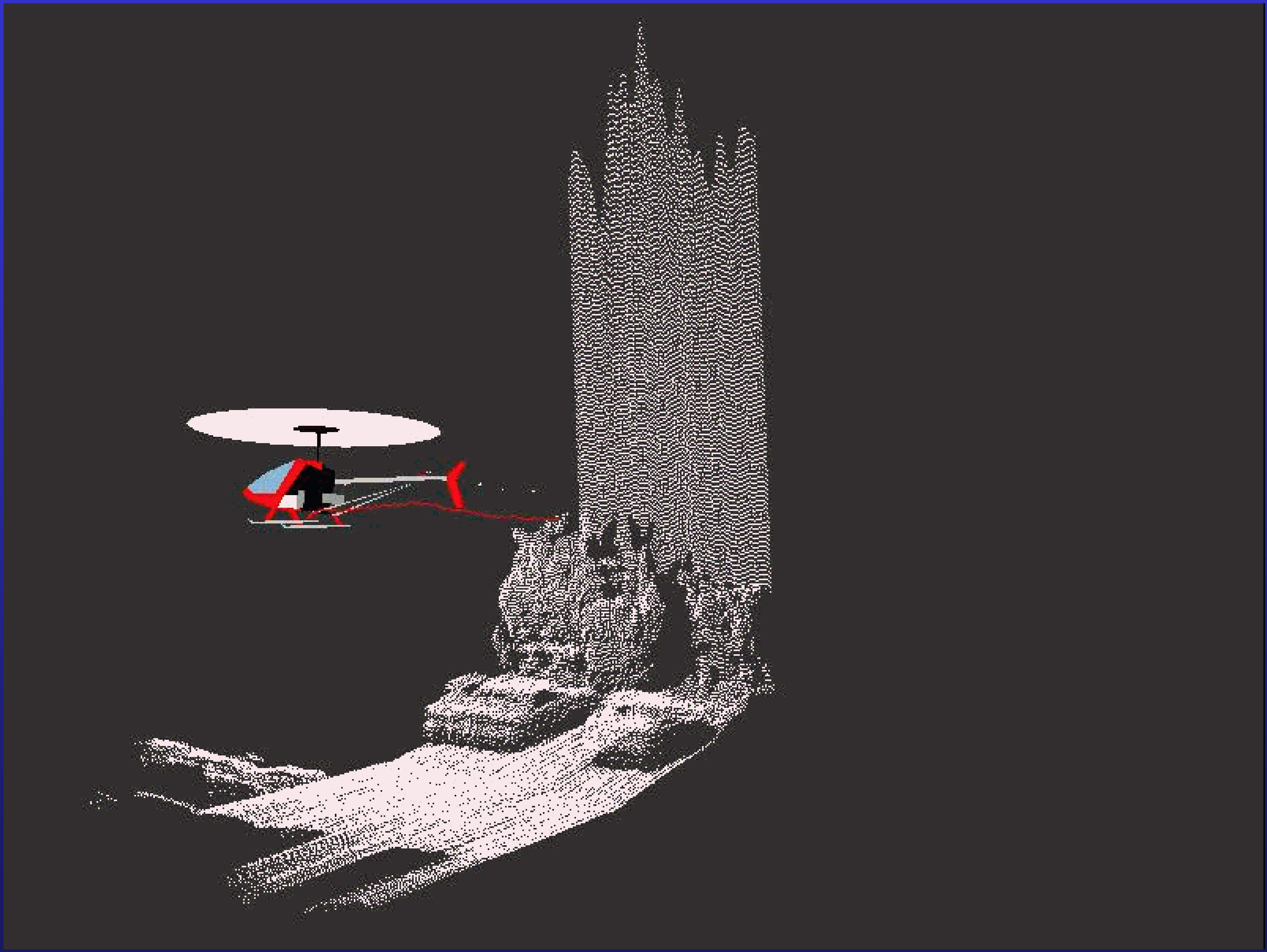
Movie courtesy of S. Thrun, Stanford University



Movie courtesy of S. Thrun, Stanford University



Movie courtesy of S. Thrun, Stanford University

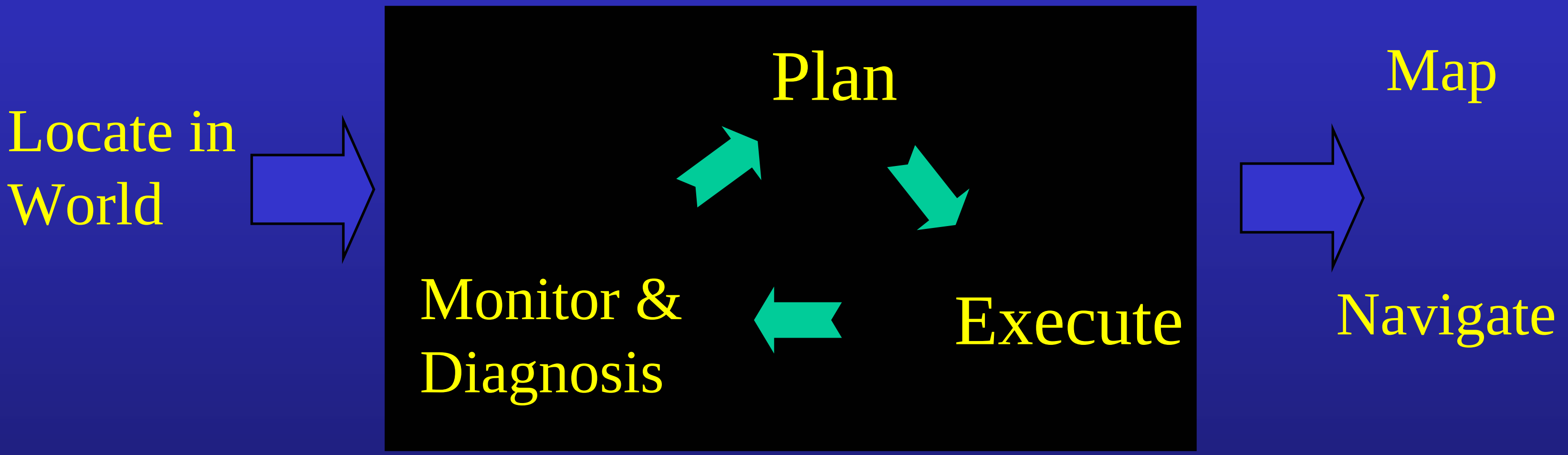



The Minerva Experience

Agent Building Blocks

- Activity Planning
- Execution/Monitoring
- Diagnosis
- Repair
- Scheduling
- Resource Allocation

- Path Planning
- Localization
- Map Building


Locate in World
Plan
Execute
Monitor & Diagnosis
Map
Navigate

3. Agile Agents


JUN 21 98

Agent Building Blocks

- Activity Planning
- Execution/Monitoring
- Diagnosis
- Repair
- Scheduling
- Resource Allocation

- Path Planning
- Localization
- Map Building
- Trajectory Design
- Policy Construction

3. Human-Robot Interaction

Nursebot Pearl

**Assisting Nursing
Home Residents**

Longwood, Oakdale, May 2001
CMU/Pitt/Mich Nursebot Project

Agent Building Blocks

- Activity Planning
- Execution/Monitoring
- Diagnosis
- Repair
- Scheduling
- Resource Allocation
- Path Planning
- Localization
- Map Building
- Trajectory Design
- Policy Construction
- Plan Adaptation
- Dialogue Management
- People Tracking

NASA Exploration Initiative

- NASA has developed a bold vision focused on robotic and combined human-robotic exploration
  - Response to critical need to augment human presence in space missions with automated, closely cooperating robotic devices
  - Significant cost reduction and safety improvement





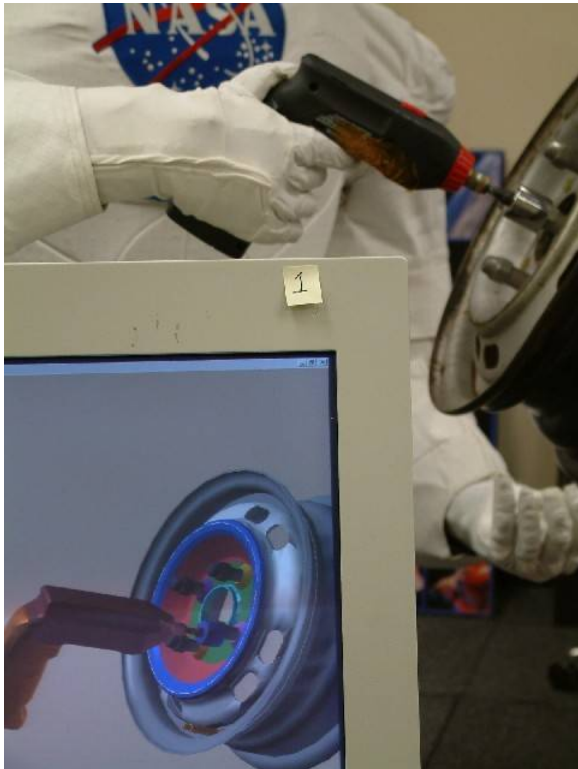
Challenge

- Autonomous humanoid robots
  - Can execute tasks intended for humans

- Human-robot interaction
  - Understand human tasks

Example: orbit assembly and repair

- Robonaut – Humanoid robot for EVA assistance



Example: surface exploration

- ERA – EVA robotic assistant follows astronaut and helps with sample collection, instrument placement

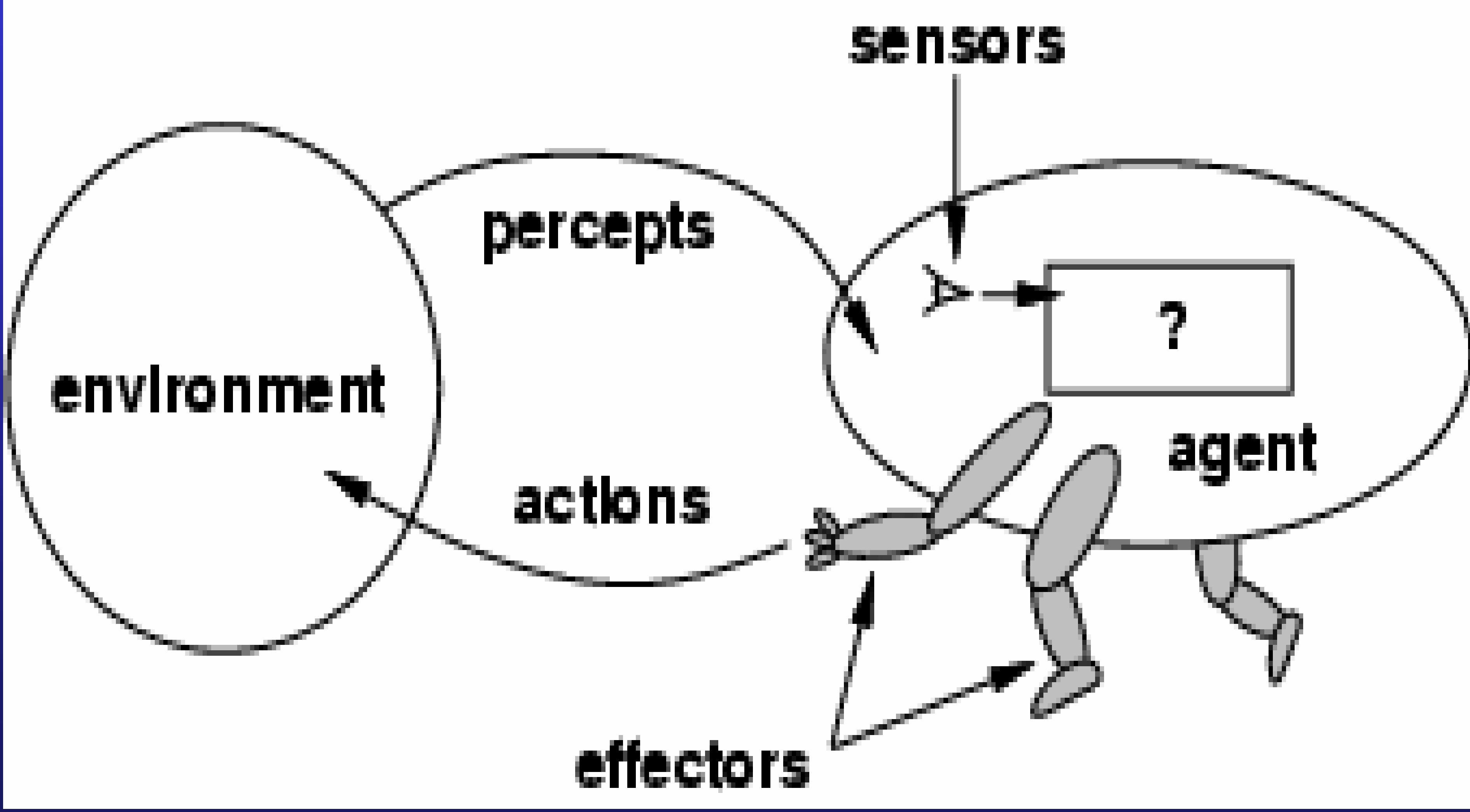
Example Mission Scenario: Task Execution

- Robot walks to its sample area
- Begins collecting samples
- Walks back to astronaut
  - Stumbles over unseen rock along the way, but recovers using appropriate limb motions

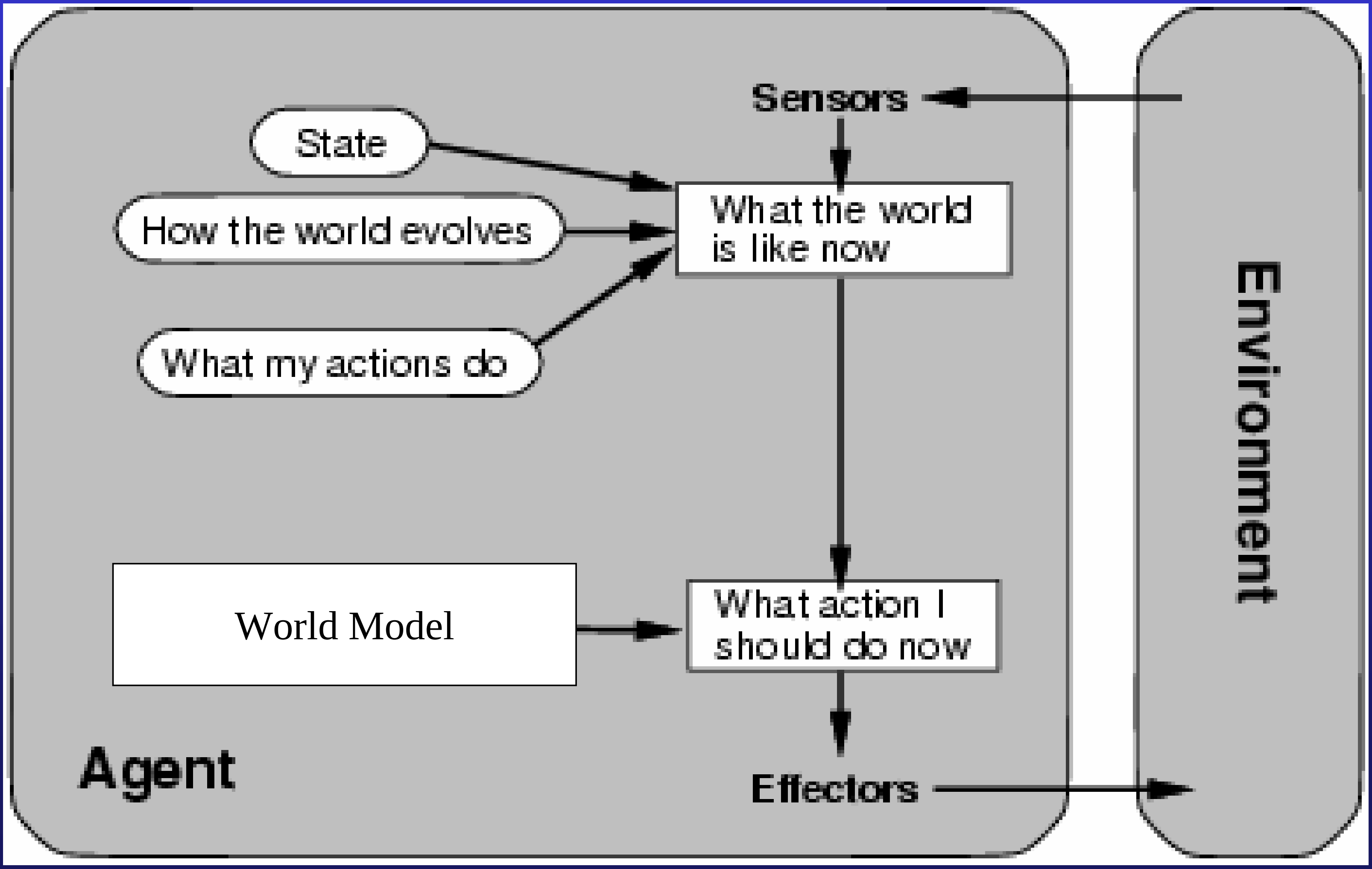
Outline

- Objectives
- Agents and Their Building Blocks
- Agent Paradigms
- Principles for Building Agents:
  - Modeling Formalisms
  - Algorithmic Principles
- Building an Agent: Fall 03 Projects

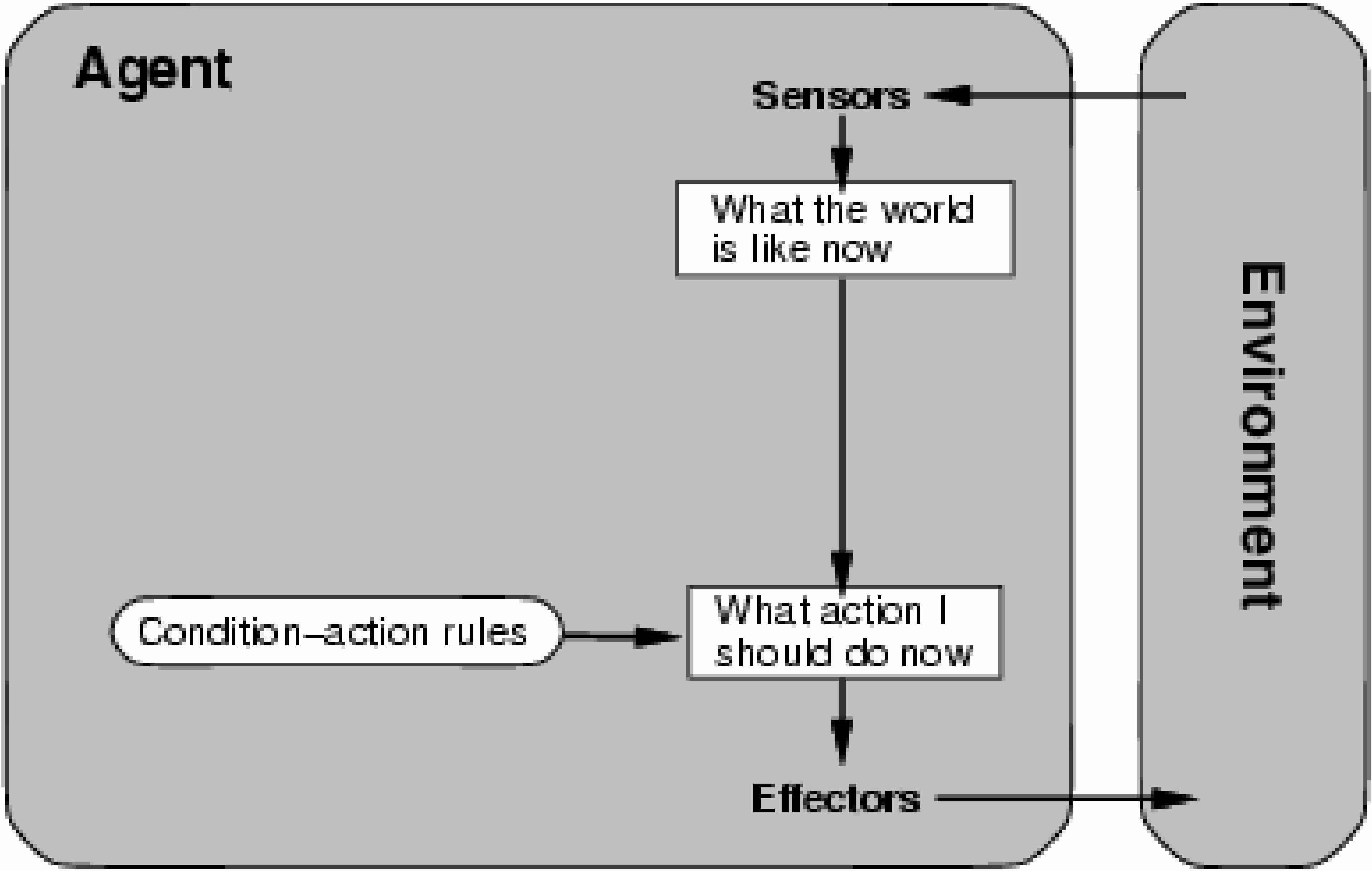
Agent Paradigms



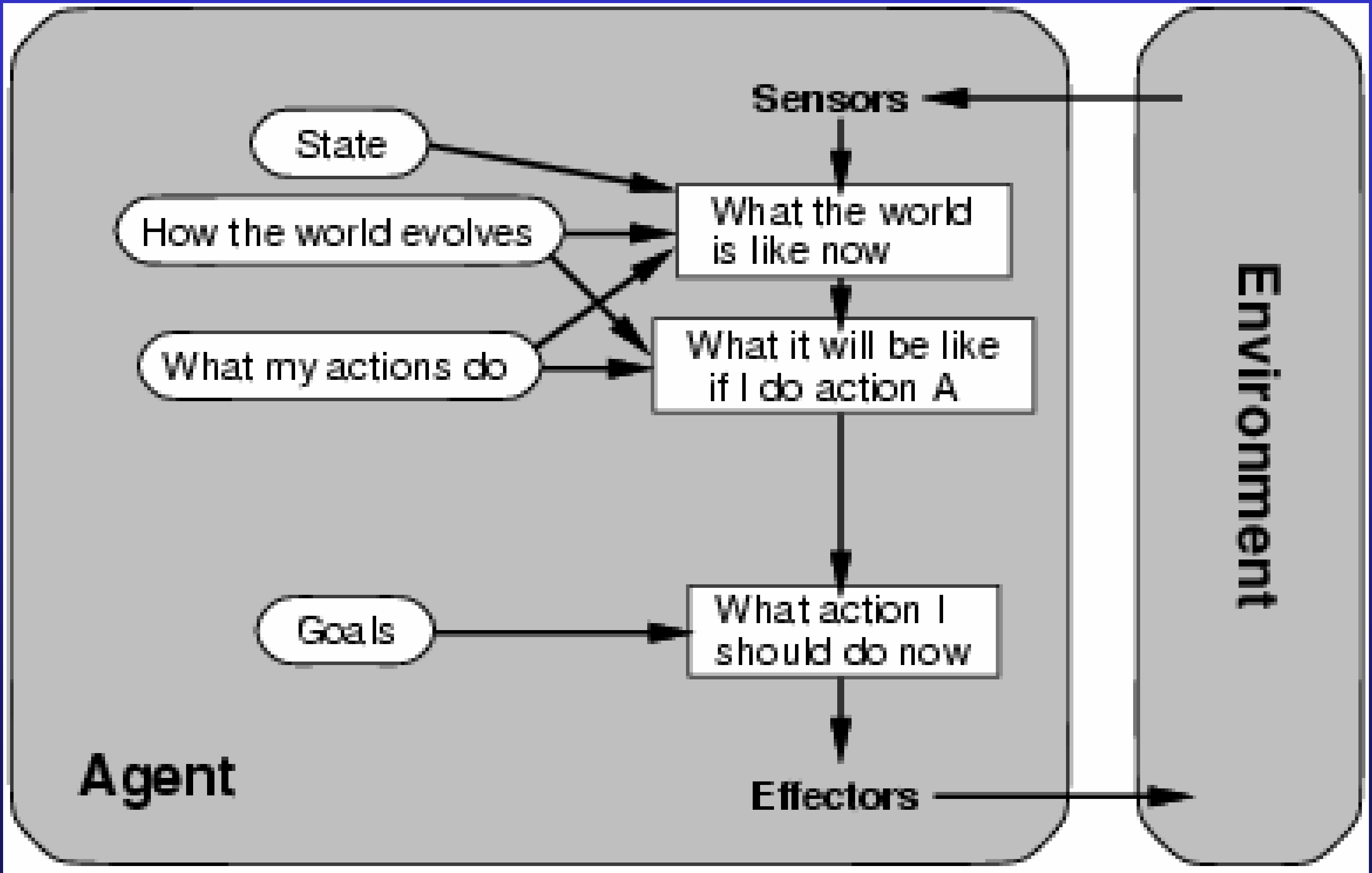
Model-based Agents



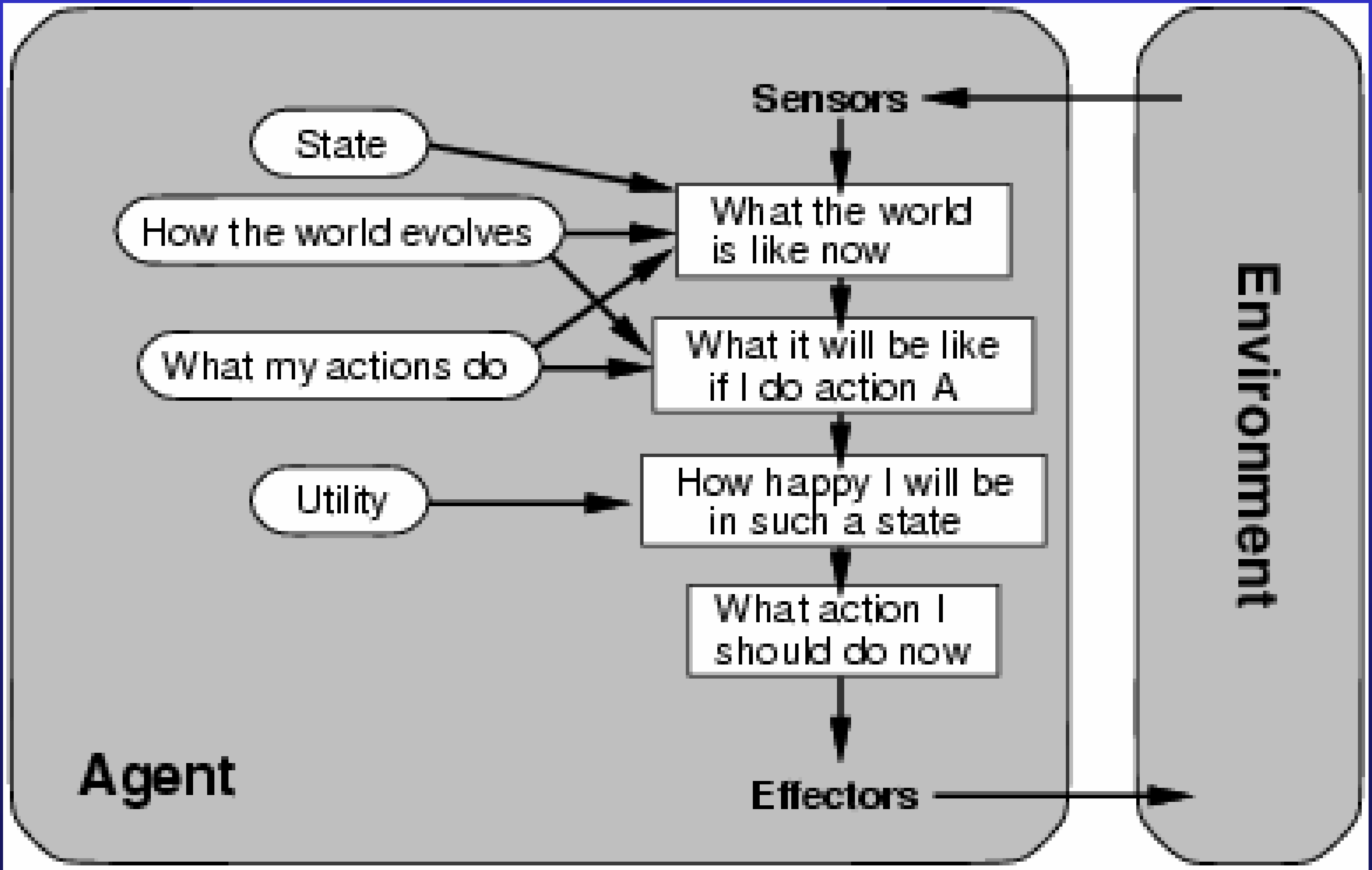
Reflexive Agents



Goal-Oriented Agents



Utility-Based Agents



Outline

- Objective
- Agents and Their Building Blocks
- Agent Paradigms
- Principles for Building Agents:
  - Modeling Formalisms
  - Algorithmic Principles
- Building an Agent: Fall 03 Projects

Building Blocks for Agent Paradigms

- Extensive Reasoning
- Extensive Optimization
- Extensive Learning

Models Underlying The Building Blocks

Building Blocks:

- Activity Planning
- Execution/Monitoring
- Diagnosis
- Repair
- Scheduling
- Resource Allocation
- Global Path Planning
- Task Assignment
- Trajectory Design
- Policy Construction

Models:

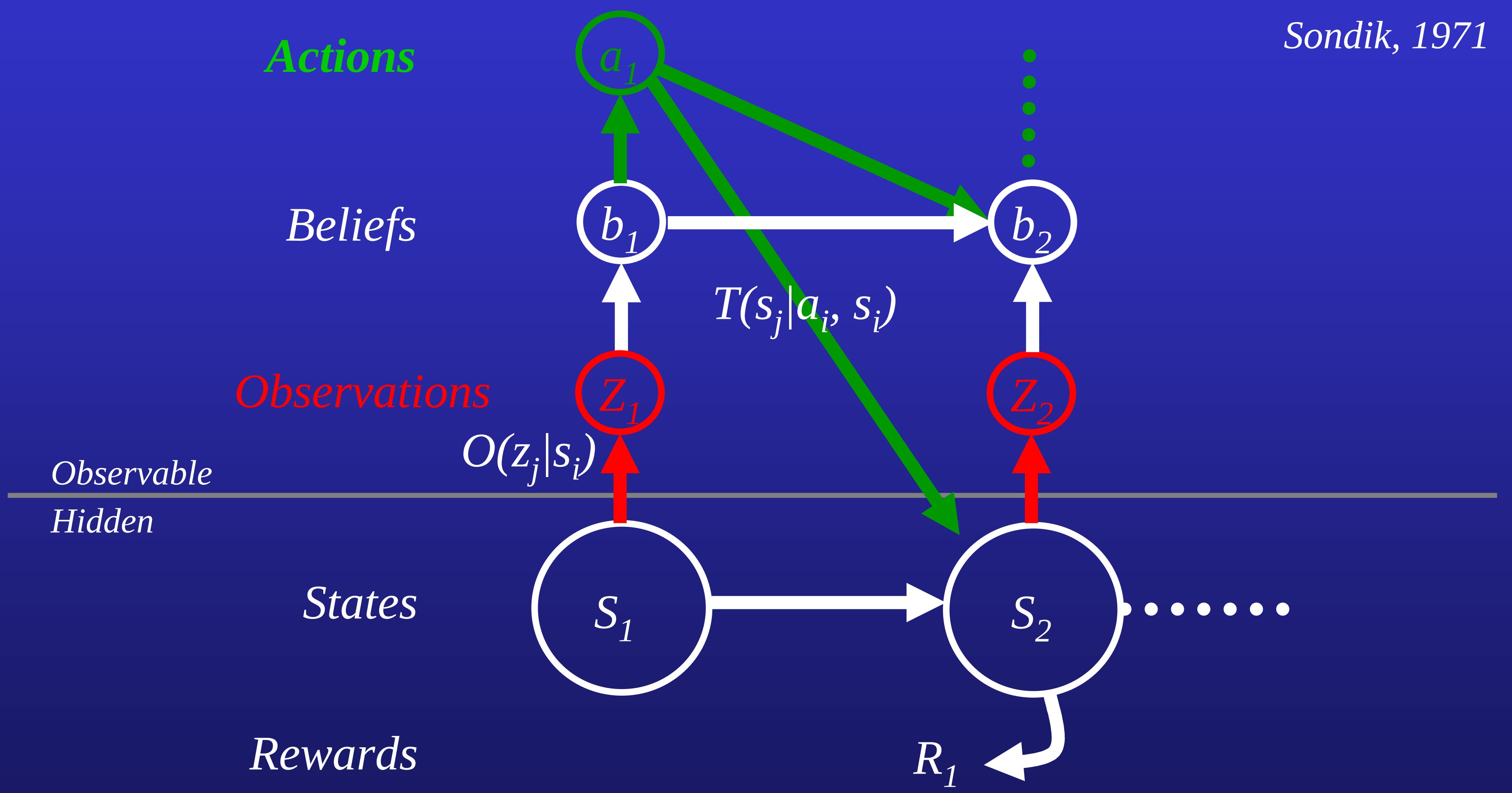
Consistency-based:

- State Spaces
- Rules, First Order Logic
- Strips Operators
- Constraint Networks
- Propositional Logic

Probabilistic & Utility-based:

- Weighted Graphs
- Linear Programs
- Mixed Integer Programs
- Markov Decision Processes
- Graphical Models

Probability and Decision Theory

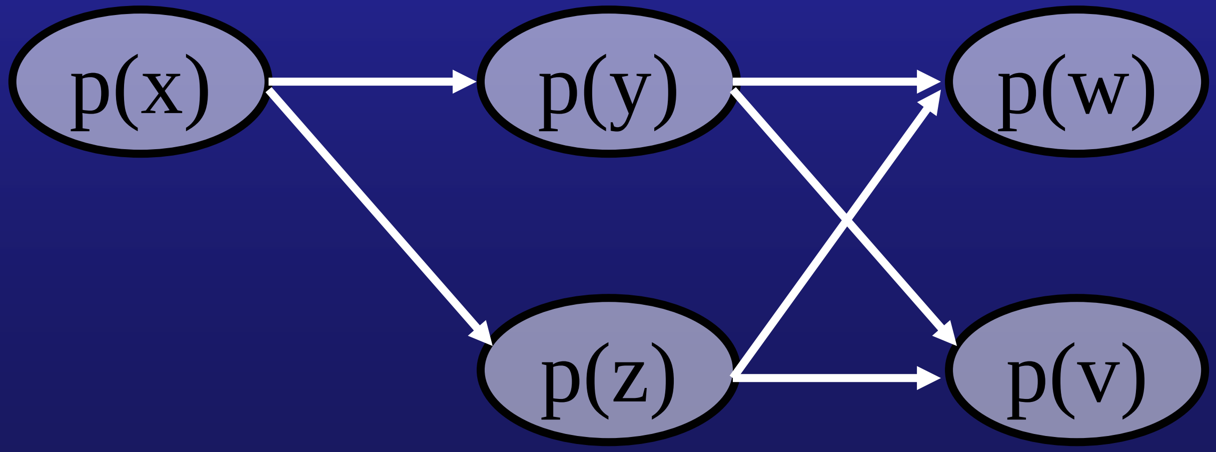


Probability Models

- Bayes Rule

$$p(x \mid z) = \frac{p(z \mid x)\, p(x)}{p(z)}$$

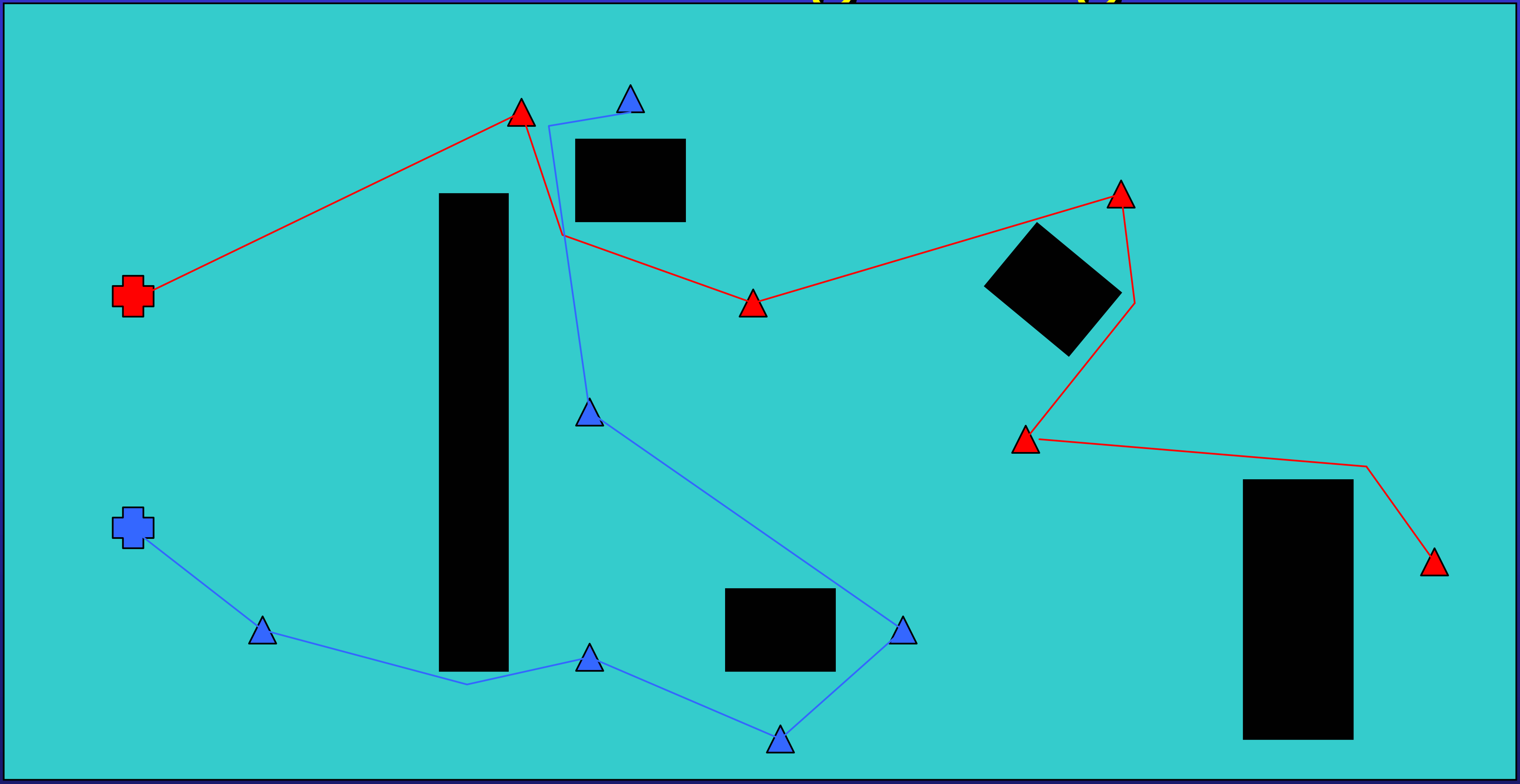
- Graphical Models



Algorithm Instances Of Building Blocks

- Activity Planning
  - Graphplan, SatPlan, Partial Order Planning
- Execution/Monitoring
- Diagnosis
  - Constraint Suspension
- Repair
  - Rule-based
- Scheduling
  - CSP-based
- Resource Allocation
  - LP-based
- Global Path Planning
  - Roadmap
- Task Assignment
- Trajectory Design
  - MILP
- Policy Construction
  - MDP
  - Reinforcement Learning

Modeling Cooperative Path Planning as a Mixed Integer Program



Cooperative Path Planning: MILP Encoding: Fuel Equation

total fuel calculated over all time instants i

past-horizon terminal cost term

$$\min_{w_i,\, v_i} = J_T = \min_{w_i,\, v_i} \sum_{i=1}^{N-1} q' w_i + \sum_{i=1}^{N-1} r' v_i + p' w_N$$

slack control vector

weighting vectors

slack state vector

Cooperative Path Planning: MILP Encoding: Constraints

- $s_{ij} <= w_{ij}$, etc. State Space Constraints
- $\mathbf{s}_i + 1 = \mathbf{A}\mathbf{s}_i + \mathbf{B}\mathbf{u}_i$ State Evolution Equation
- $x_i <= x_{min} + Mt_{i1}$

  $-x_i <= -x_{max} + Mt_{i2}$

  $y_i <= y_{min} + Mt_{i3}$ Obstacle Avoidance

  $-y_i <= -y_{max} + Mt_{i4}$ (for all time i)

  $\Sigma\, t_{ik} <= 3$ (t introduce IP element)
- Similar equation for Collision Avoidance (for all pairs of vehicles)

Algorithm Principles
Underlying Building Blocks

<u>Uninformed Search:</u>

- Depth First, Breadth First
- Iterative Deepening.
- Backtrack Search
- Backtrack w Forward checking
- Conflict-directed Search

<u>Informed Search:</u>

- Single Source Shortest Bath
- Best First Search
  (A*, Hill Climbing, …)
- Simplex
- Branch and Bound

Algorithm Principles
Underlying Building Blocks

<u>Deduction:</u>

- Unification
- Unit Clause Resolution
- Arc Consistency.
- Gaussian Elimination

<u>Relaxation</u>

- Value Iteration
- Reinforcement Learning

<u>Divide and Conquer</u>

- Branching
- Sub-goaling
- Variable Splitting
- Dynamic Programming
- Uninformed & Informed

<u>Abstraction:</u>

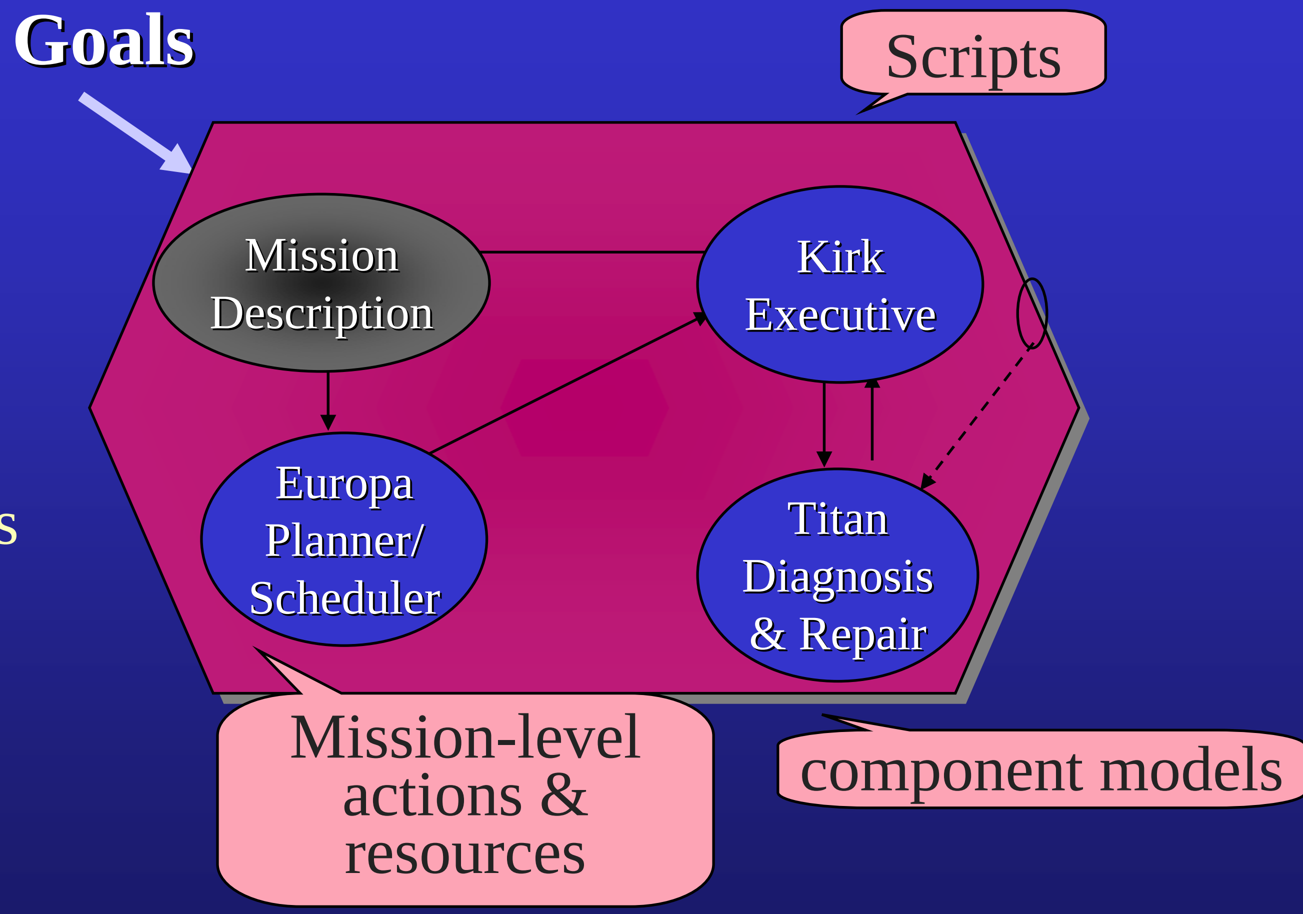
- Conflicts
- Bounding

Outline

- Objectives
- Agents and Their Building Blocks
- Agent Paradigms
- Principles for Building Agents:
  - Modeling Formalisms
  - Algorithmic Principles
- Building an Agent: Fall 03 Projects

16.413 Project: Example of a Model-based Agent:

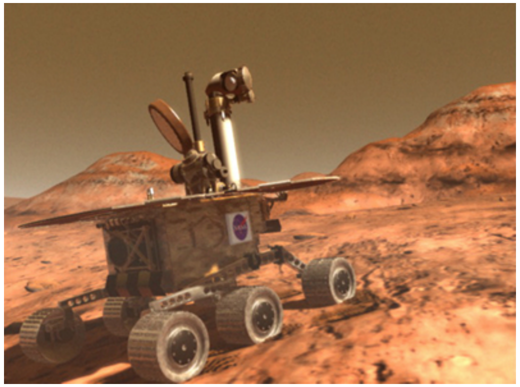
- Goal-directed
- First time correct
  - projective
  - reactive
- Commonsense models
- Heavily deductive



16.410/3 Student Teams

Model-based Embedded & Robotic Systems

- Ground Science Planning for Rovers
  - Jessica Marquez and Julie Arnold
- Onboard Planning and Execution on Rovers
  - Stephen Licht, Andrew Vaughn, Steve Paschall
- **Model Based Diagnosis and Execution on Rovers**
  - **Lars Blackmore, Steve Block, Thomas Leauté, Emily Fox**

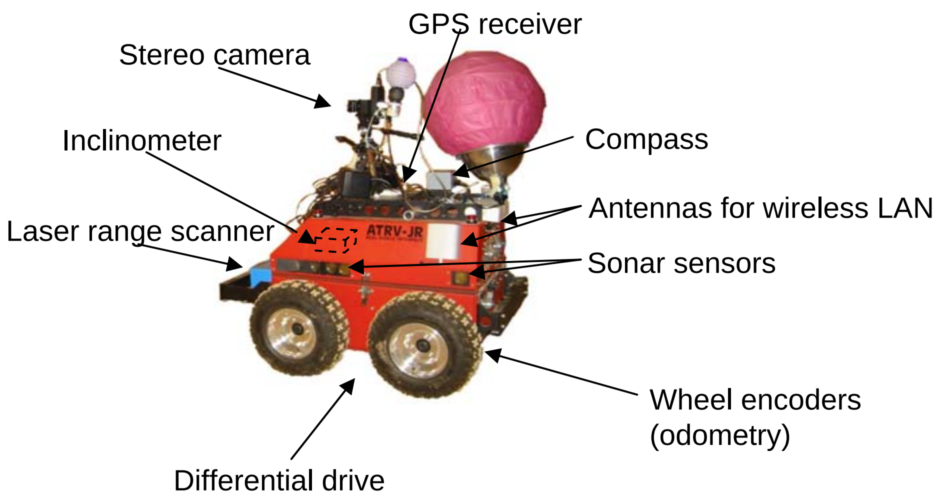


MER

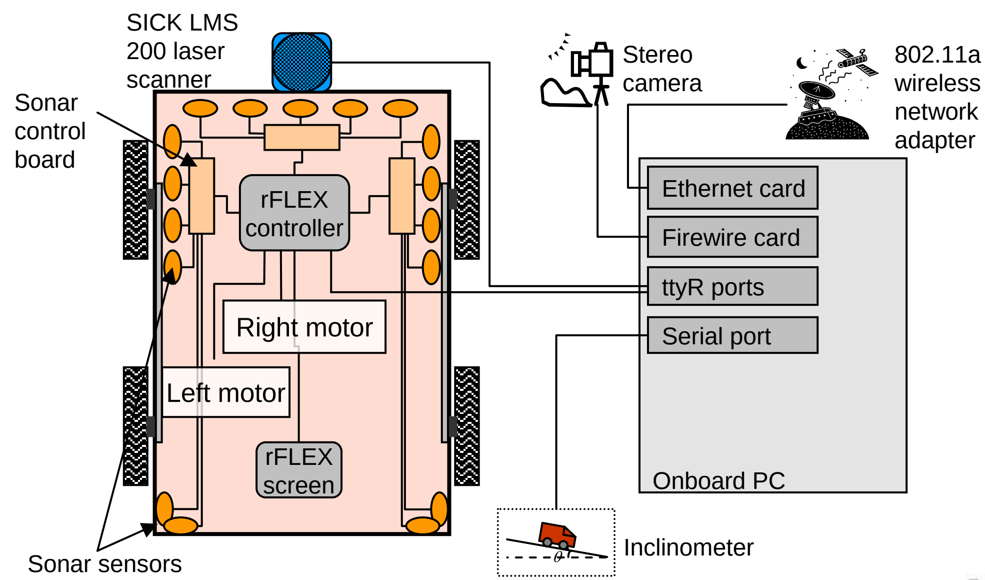
- Model Based Execution on SPHERES I
  - Mehdi Alighanbari, Tsoline Mikaelian, Martin Ouimet, Mike Voightmann
- Model Based Execution on SPHERES II
  - Robert Effinger, Jacomo Corbo, Jonathon Histon, Sameera Ponda

SPHERES

Rover Testbed Setup



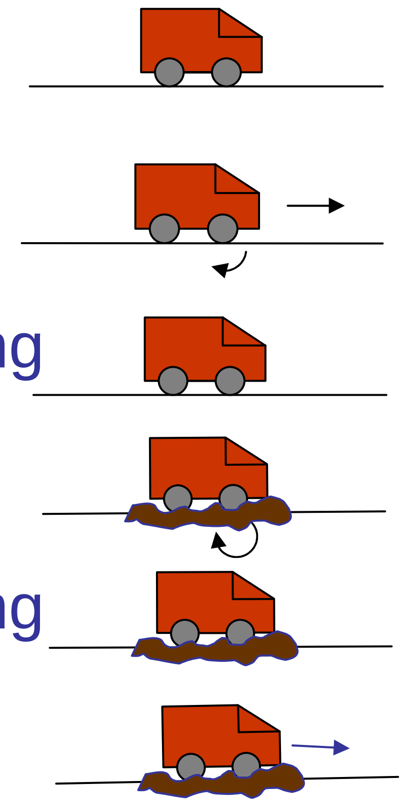
- Sensors give information on motion and environment.
- Onboard PC allows for real-time computation and command processing.



Figures from Seung Chung's Project Description handout

Simple Slipping Scenario

- Initialize in all-stop state
- Command ‘go’ : successful driving
- Command ‘stop’ : successful stopping
- Command ‘go’ : slips
- Command ‘stop’ : successful stopping
- Command ‘go’ : successful driving



Course Objective 1: Principles of Agents

16.410/13: To learn the modeling and algorithmic building blocks for creating reasoning and learning agents:

- To formulate reasoning problems.
- To describe, analyze and demonstrate reasoning algorithms.
- To model and encode knowledge used by reasoning algorithms.