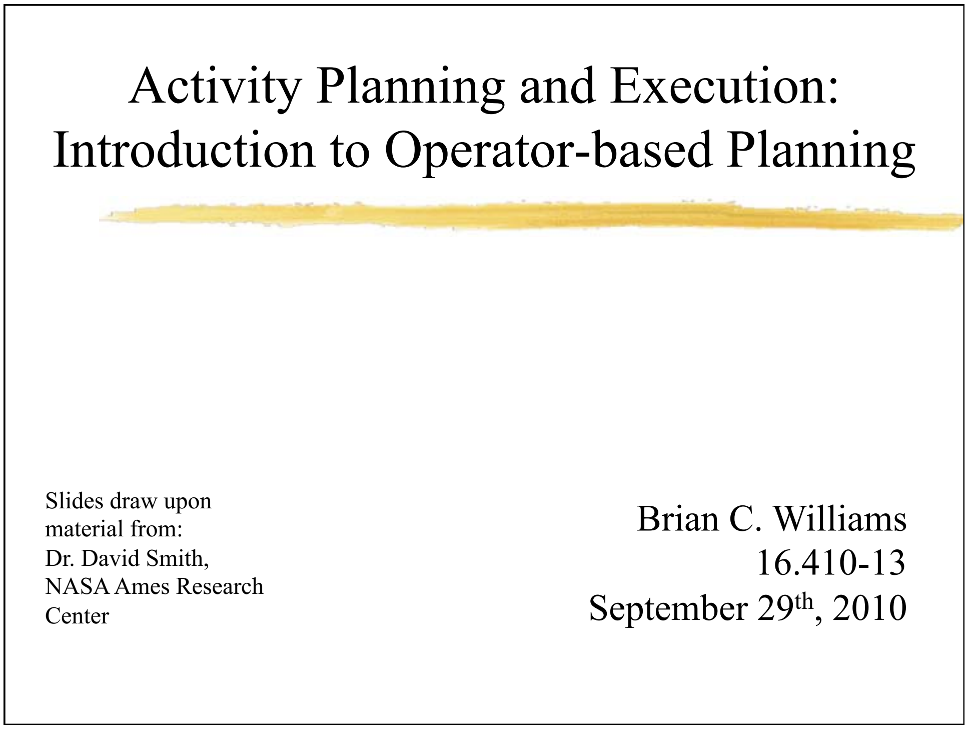
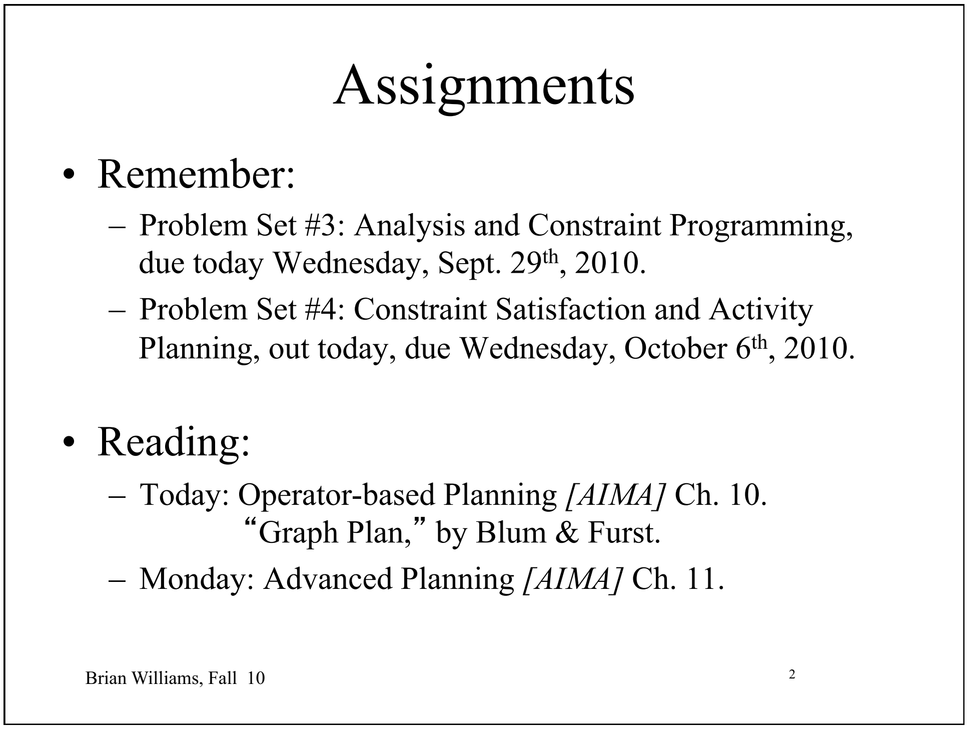


Activity Planning and Execution: Introduction to Operator-based Planning

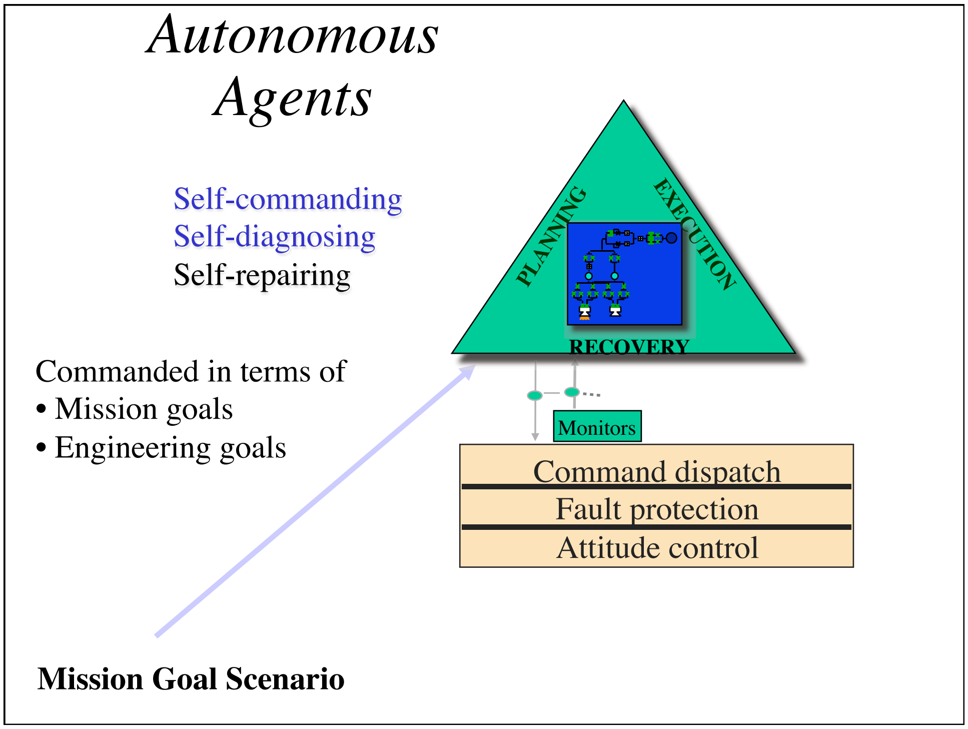
Slides draw upon
material from:
Dr. David Smith,
NASA Ames Research
Center

Brian C. Williams
16.410-13
September 29th, 2010



Assignments

- Remember:
  - Problem Set #3: Analysis and Constraint Programming, due today Wednesday, Sept. 29th, 2010.
  - Problem Set #4: Constraint Satisfaction and Activity Planning, out today, due Wednesday, October 6th, 2010.
- Reading:
  - Today: Operator-based Planning *[AIMA]* Ch. 10.
    “Graph Plan,” by Blum & Furst.
  - Monday: Advanced Planning *[AIMA]* Ch. 11.

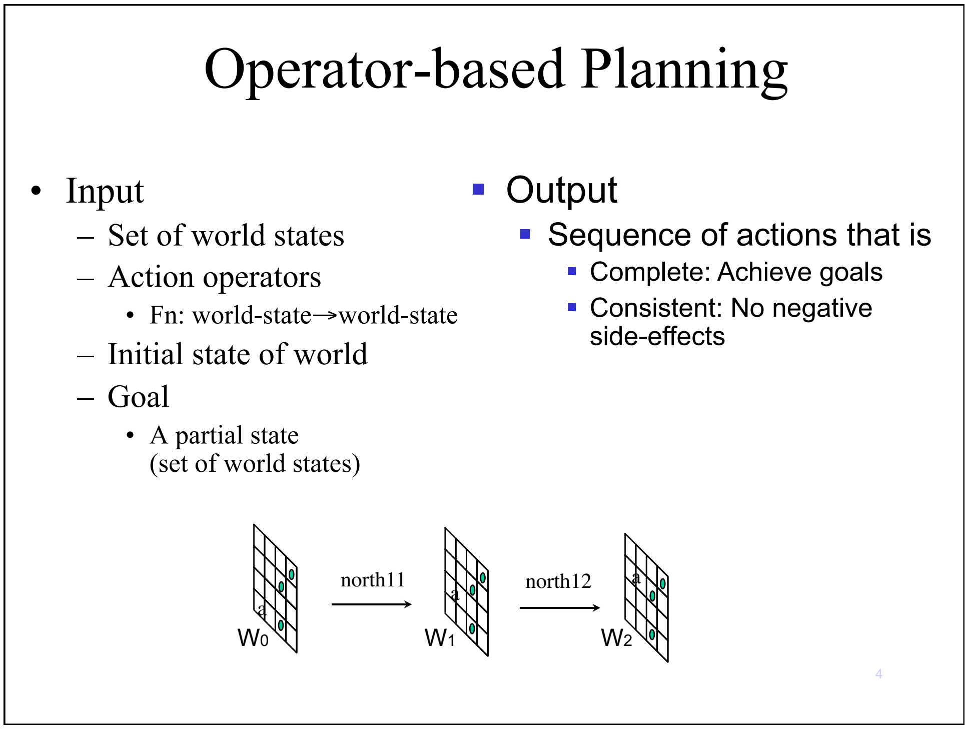


Autonomous Agents

Self-commanding
Self-diagnosing
Self-repairing

Commanded in terms of
- Mission goals
- Engineering goals

Mission Goal Scenario



Operator-based Planning

- Input
  - Set of world states
  - Action operators
    - Fn: world-state→world-state
  - Initial state of world
  - Goal
    - A partial state
      (set of world states)
- Output
  - Sequence of actions that is
    - Complete: Achieve goals
    - Consistent: No negative side-effects

Operators in STRIPS Representation

```
Turn (?target):
    Pre: Pointing(?direction),
         ?direction ≠ ?target
    Eff: ¬Pointing(?direction),
         Pointing(?target)
```

Outline

- Example: DS1 and MER Mission-Planning
- Overview of Operator-based Planning
- Graph Plan

Mars Rover slides removed due to copyright restrictions.

Outline

- Example: MER Mission Planning
- Overview of Operator-based Planning
- Graph Plan

Planning

Find:

program of actions that achieves the objective

Planning

Find:

program of actions that achieves the objective

partially-ordered set of actions

typically unconditional

no loops

goals

Paradigms

Classical planning

(STRIPS, operator-based, first-principles)

“generative”

Hierarchical Task Network planning

“practical” planning

MDP & POMDP planning

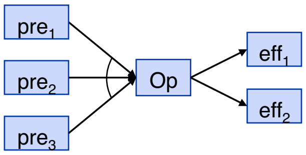
planning under uncertainty

Classical Problem Statement

Propositions: P_i

Initial Conditions: P_1 P_2 P_3 P_4

Operators:



Goals: $Goal_1$ $Goal_2$ $Goal_3$

Simple Spacecraft Problem

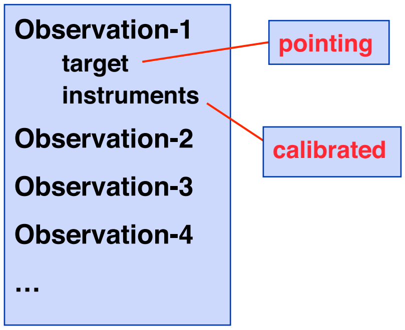


Image credit: NASA.

Propositions: Target Pointed To, Camera Calibrated?, Has Image?

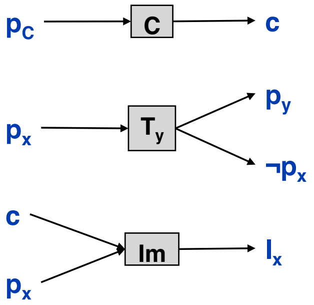
Operators: Calibrate, Turn to Y, and Take Image.

Example

<u>Init</u>

p_C

<u>Actions</u>

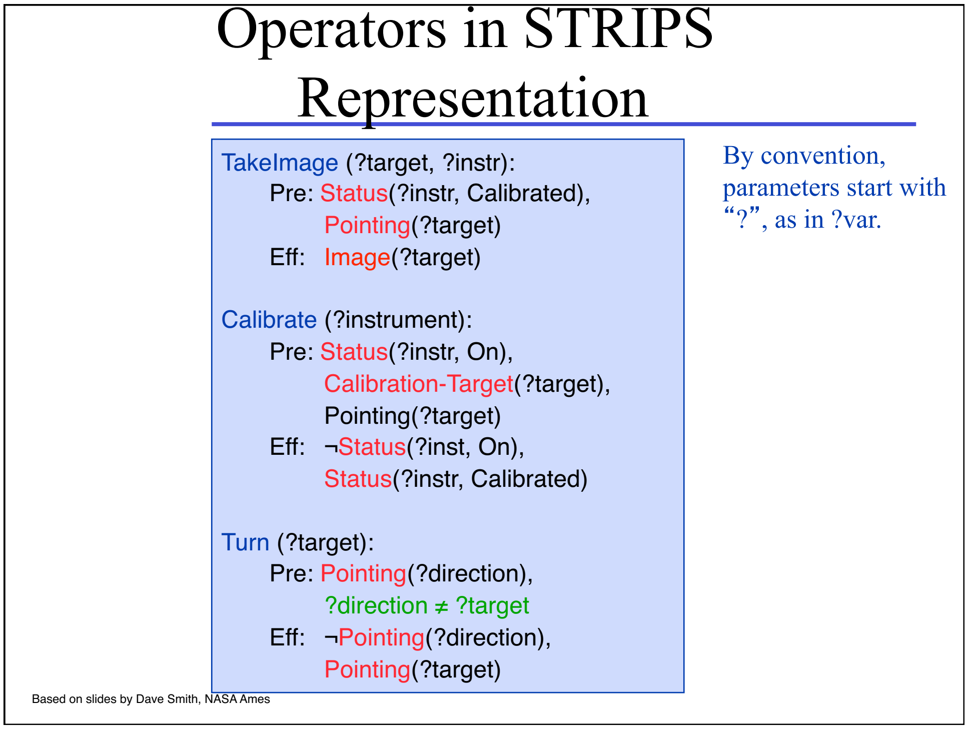


<u>Goal</u>

I_A

Propositions: Target Pointed To, Camera Calibrated?, Has Image?

Operators: Calibrate, Turn to y, and Take Image.

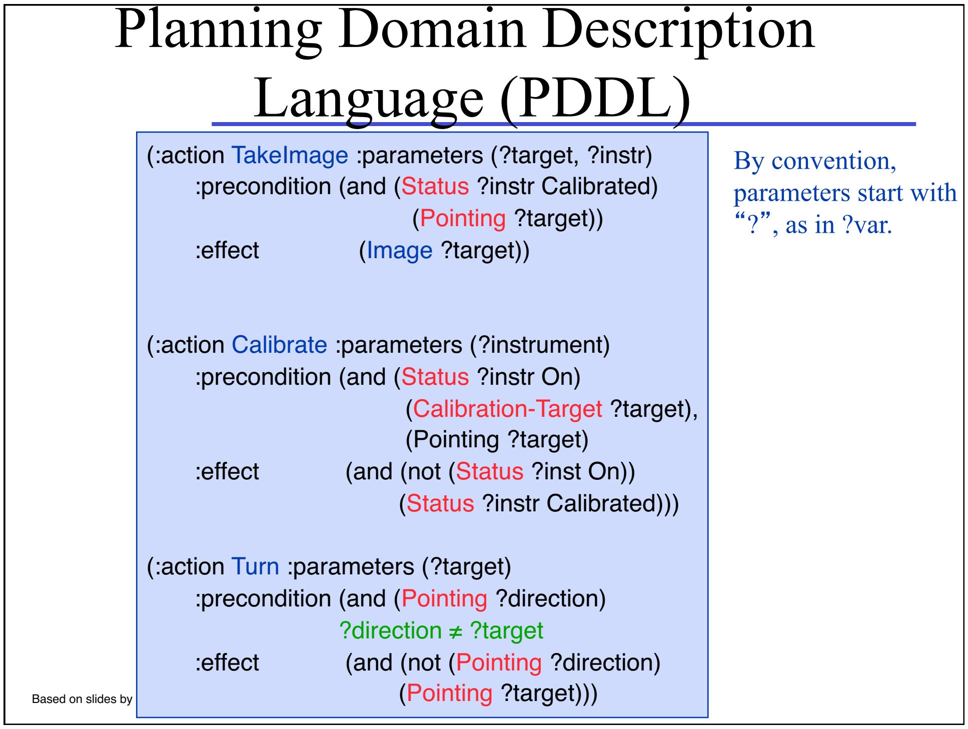


Operators in STRIPS Representation

```
TakeImage (?target, ?instr):
     Pre: Status(?instr, Calibrated),
           Pointing(?target)
     Eff:  Image(?target)

Calibrate (?instrument):
     Pre: Status(?instr, On),
           Calibration-Target(?target),
           Pointing(?target)
     Eff:  ¬Status(?inst, On),
           Status(?instr, Calibrated)

Turn (?target):
     Pre: Pointing(?direction),
           ?direction ≠ ?target
     Eff:  ¬Pointing(?direction),
           Pointing(?target)
```

By convention, parameters start with "?", as in ?var.

Based on slides by Dave Smith, NASA Ames



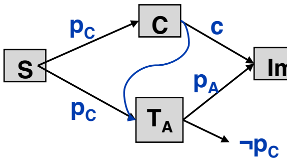
Planning Domain Description Language (PDDL)

```
(:action TakeImage :parameters (?target, ?instr)
     :precondition (and (Status ?instr Calibrated)
                        (Pointing ?target))
     :effect            (Image ?target))


(:action Calibrate :parameters (?instrument)
     :precondition (and (Status ?instr On)
                        (Calibration-Target ?target),
                        (Pointing ?target)
     :effect           (and (not (Status ?inst On))
                            (Status ?instr Calibrated)))

(:action Turn :parameters (?target)
     :precondition (and (Pointing ?direction)
                        ?direction ≠ ?target
     :effect           (and (not (Pointing ?direction)
                            (Pointing ?target)))
```

By convention, parameters start with "?", as in ?var.

Based on slides by

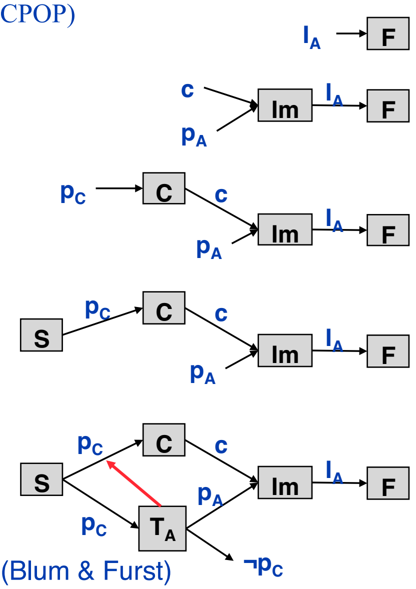
Partial Order Plan

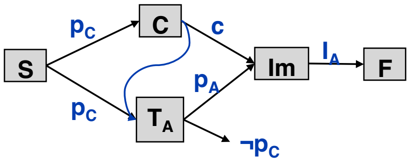


Planning in the Past:
Partial Order Causal Link Planning

(SNLP, UCPOP)

1. Select an open condition
2. Choose an op that can achieve it
   - Link to an existing instance or
   - Add a new instance
3. Resolve threats





Current Planning: Graph-based Planners (Blum & Furst)

What assumptions are implied by the STRIPS representation?

```
TakeImage (?target, ?instr):
      Pre: Status(?instr, Calibrated),
           Pointing(?target)
      Eff:    Image(?target)

Calibrate (?instrument):
      Pre: Status(?instr, On),
           Calibration-Target(?target),
           Pointing(?target)
      Eff:  ¬Status(?inst, On),
            Status(?instr, Calibrated)

Turn (?target):
      Pre: Pointing(?direction),
           ?direction ≠ ?target
      Eff:  ¬Pointing(?direction),
            Pointing(?target)
```

STRIPS Assumptions:

- Atomic time.
- Agent is omniscient (no sensing necessary).
- Agent is sole cause of change.
- Actions have deterministic effects.
- No indirect effects.

The Simple Spacecraft Revisited: Complications

Observation-1
- **priority**
- **time window**
- **target**
- **instruments**
- **duration**

Observation-2

Observation-3

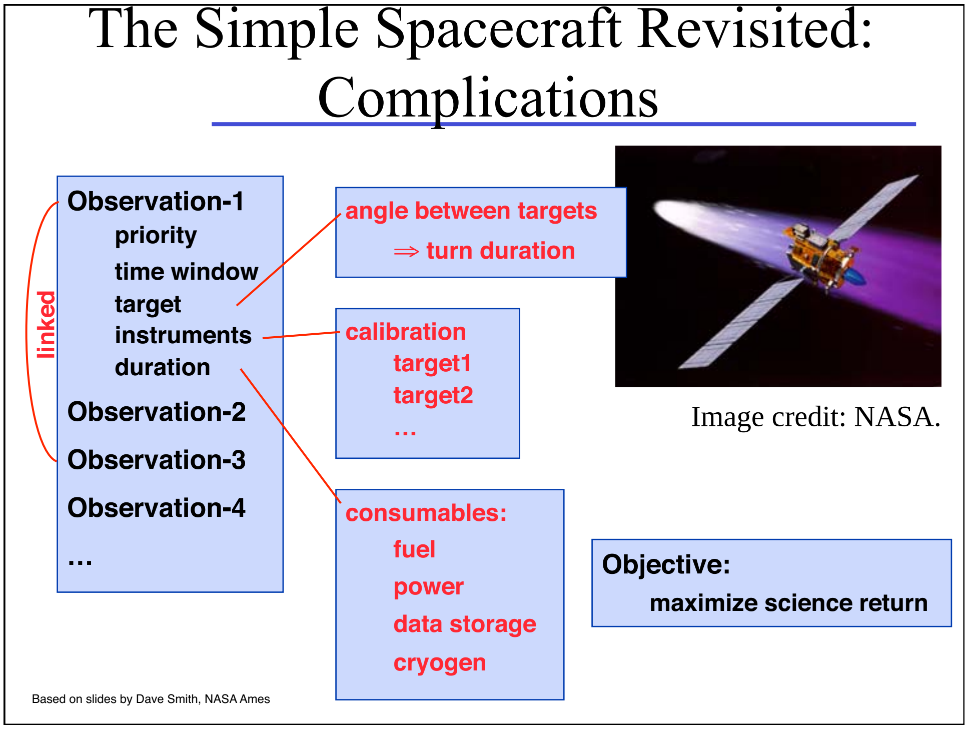
Observation-4

…

Image credit: NASA.

Objective:
maximize science return

Based on slides by Dave Smith, NASA Ames



The Simple Spacecraft Revisited: Complications

linked

Observation-1
- priority
- time window
- target
- instruments
- duration

Observation-2

Observation-3

Observation-4

…

angle between targets

⇒ turn duration

calibration
- target1
- target2
- …

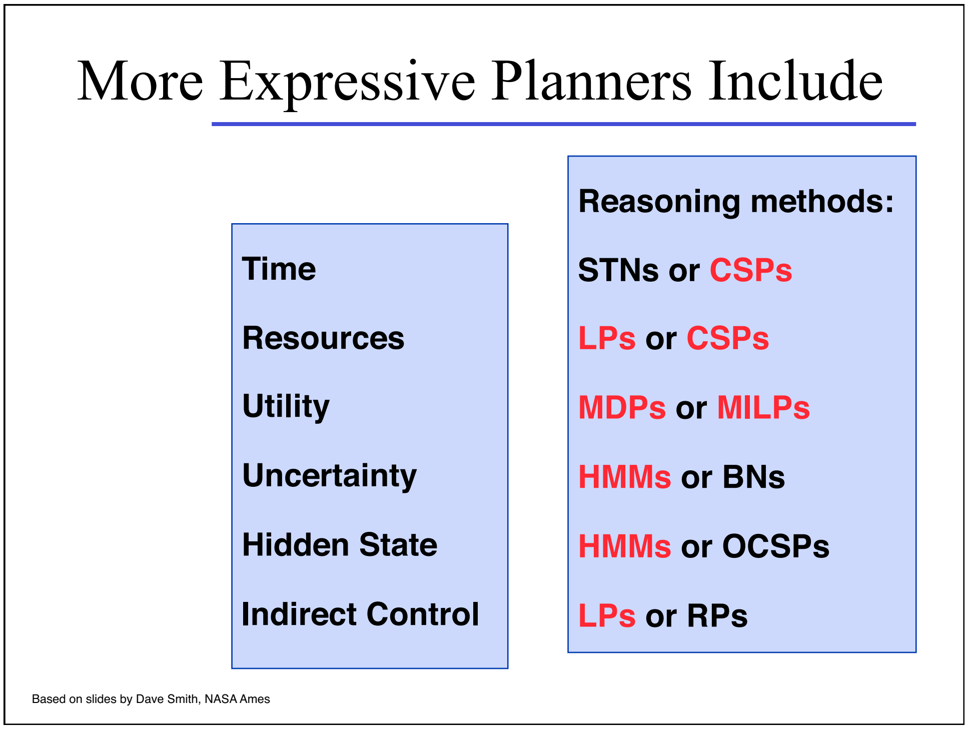
consumables:
- fuel
- power
- data storage
- cryogen

Image credit: NASA.

Objective:

maximize science return

Based on slides by Dave Smith, NASA Ames



More Expressive Planners Include

	Reasoning methods:
Time	STNs or CSPs
Resources	LPs or CSPs
Utility	MDPs or MILPs
Uncertainty	HMMs or BNs
Hidden State	HMMs or OCSPs
Indirect Control	LPs or RPs

Based on slides by Dave Smith, NASA Ames