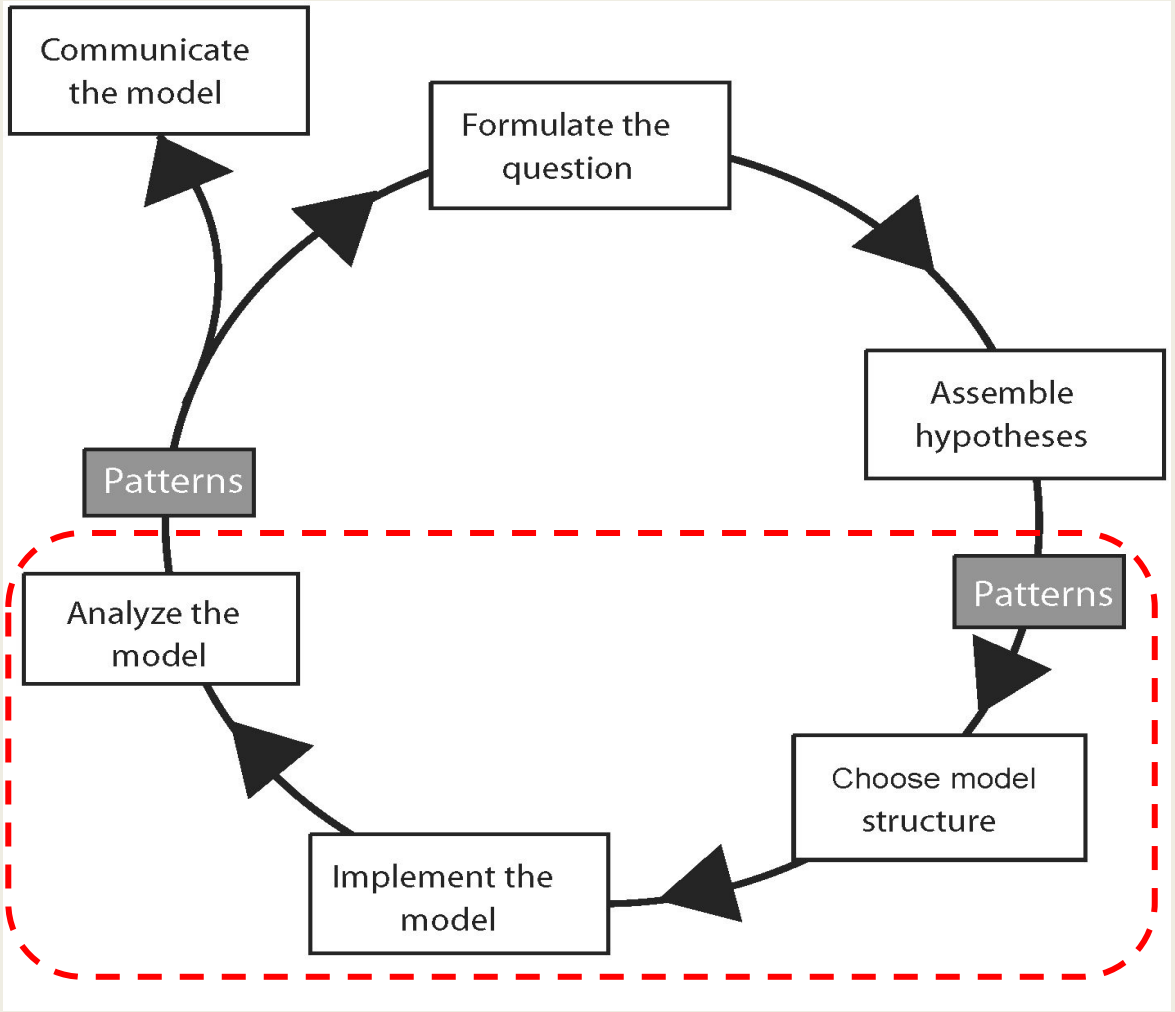
SIMULATION

And the modeling cycle

Simulation

- Simulation is a particular type of modelling
- Modeling is a well recognized way of understanding the world:
  - *something we subconsciously do all the time,*
- A model is a simplification – smaller, less detailed, less complex, or all of these together – of some other structure or system.
  - *Gives us the power of prediction*

Doing Science with Models and Simulations



1. Formulate the Question

- We need to start with a very clear research question
- clear question requires a clear focus
- For complex systems, getting focused can be difficult
- Very often, even our **questions are only experimental** and later we might need to **reformulate** the question

2. Assemble hypotheses

- Agent based modeling directly represent the agents and their behavior
- We create the agents, put them in a virtual environment, and then run the simulation.
- We have to formulate many hypotheses for **what processes and structures** are essential
  - *What factors have strong influence in the phenomenon of interest?*
  - *Are these factors independent or interacting?*
  - *Are they affected by important other factors?*

3. Choose model Structure

- **Scales**
  - *Distance, time and other relevant units mapped to the simulated world*
- **Entities**
  - *Agents and environment (you can think of environment as immobile agents)*
- **State variables**
  - *Variables to capture states of agents and environments*
- **Processes**
  - *Agent actions for every time-step based on state and parameters*
  - *Agent-to-agent interactions based on state and parameters*
  - *Agent-to-environment interactions and parameters*
- **Parameters**
  - *Initial Setup and randomization parameters. Independent/input variables*

4. Implement the model

- **Setup** procedures
- **Agent** procedures
- **World/Observer** procedures
- **Monitors and reporters**
- **Interfaces** – parameters & plots
  - *This is what you will learn this week*

Agent Based Modeling (Recap)

- The idea that the world can be modeled using

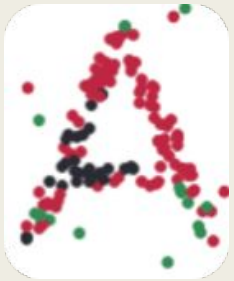
  Agents *– individuals with specific characteristics: location, behavior, etc.*

  Environment *– The simulated world where these agents are deployed*

  Agent-to-Agent Interaction *- How agents interact (cooperate/compete/coexist) with other agents*

  Agent-to-Environment Interaction *– How agents are affected-by/reacts-to the environment and How the environment is affected by the action(s) of the agent(s)*

Agent Based Modeling Toolkits



Agents.jl

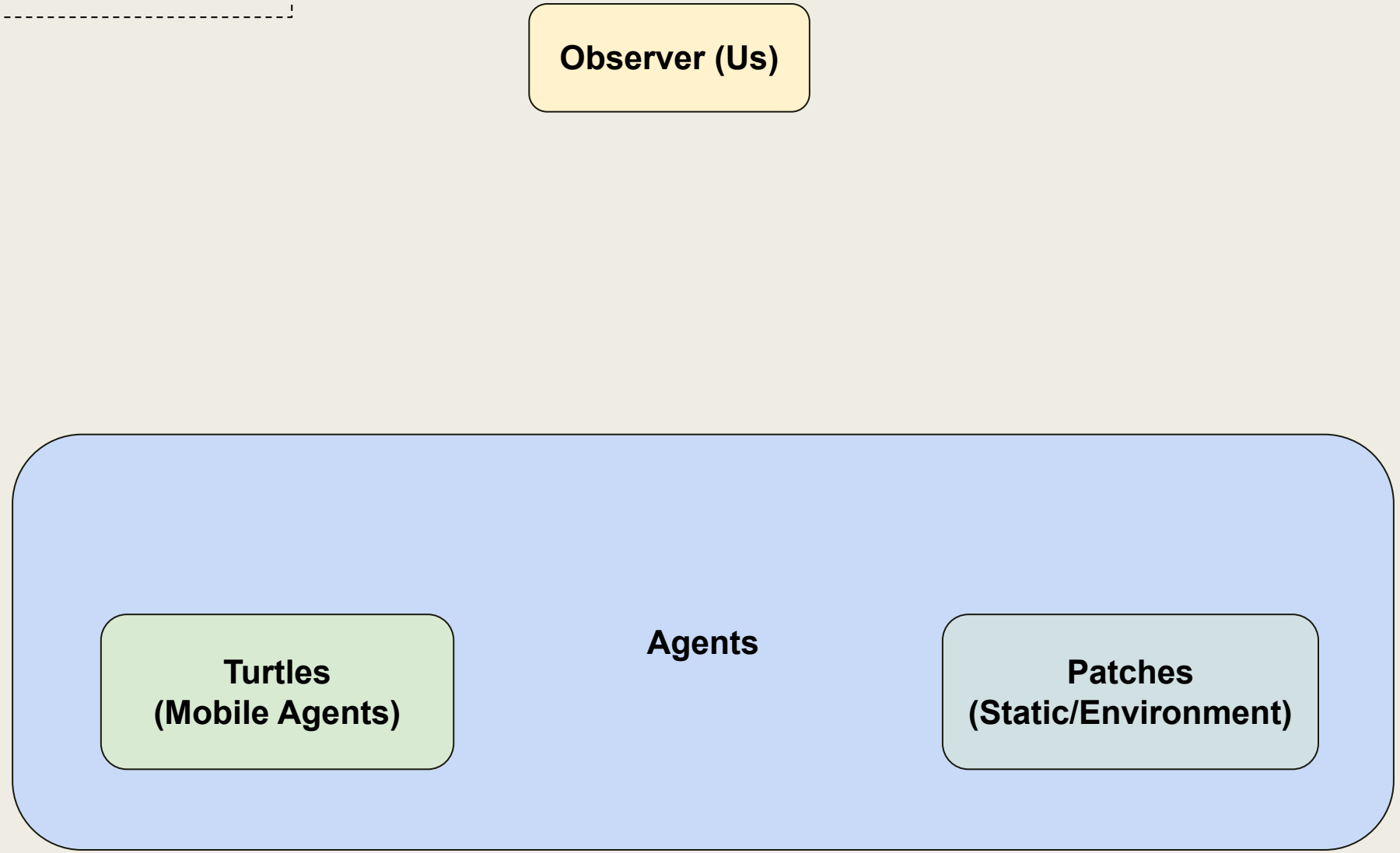
MASON

This course will introduce you to NetLogo

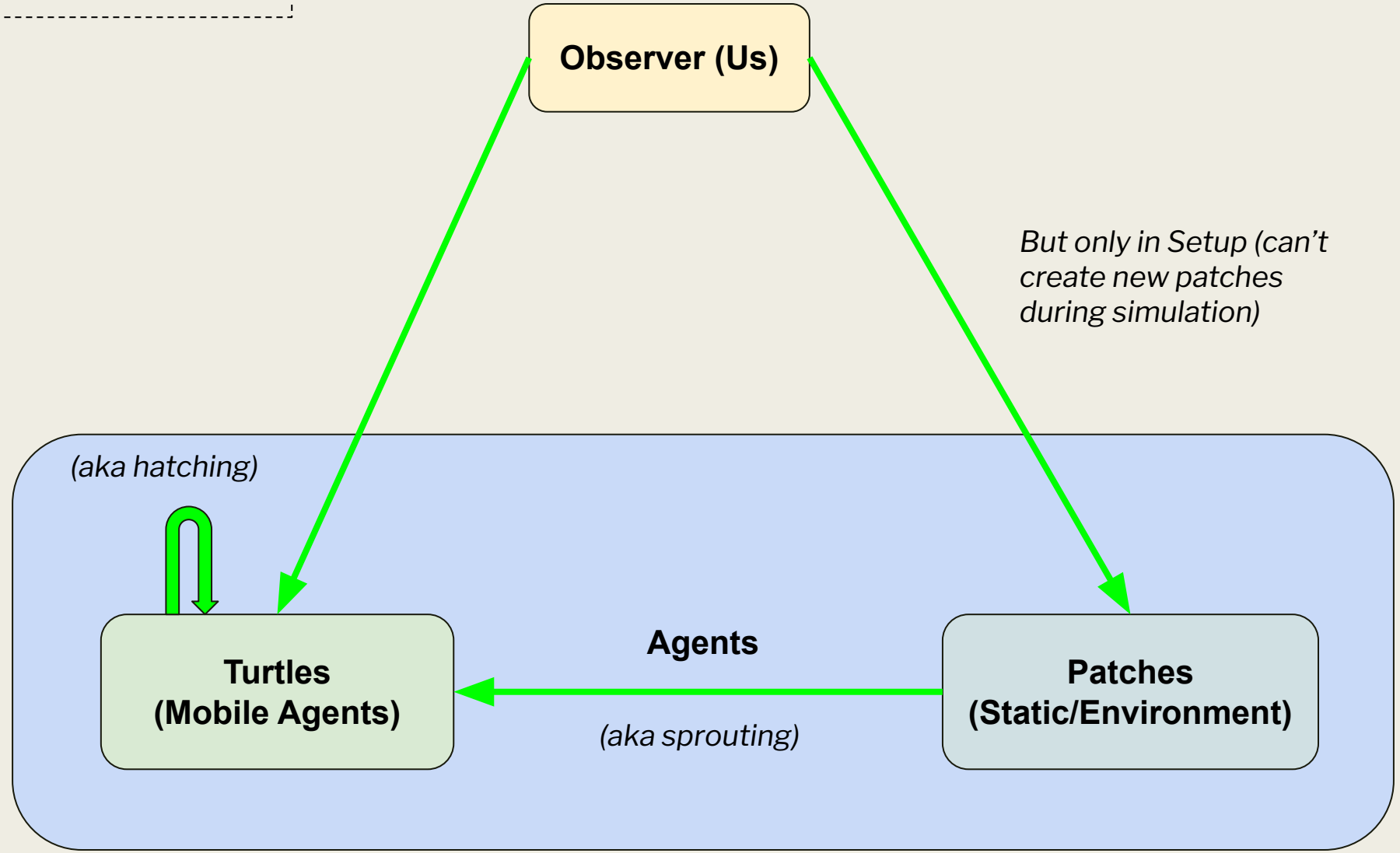
Why NetLogo?

- Pros
  - ***Low threshold*** *– Easy to learn and to start modeling*
  - ***High Ceiling*** *- Supports Highly complex (Professional/Scientific/Industrial-scale) models*
  - *Successfully used in many domains and scientific publications*
- Cons
  - *You have to learn a new programming language (But a very simple language)*
  - *Under-the-hood investigation (or improvement) of the platform requires understanding of Scala*

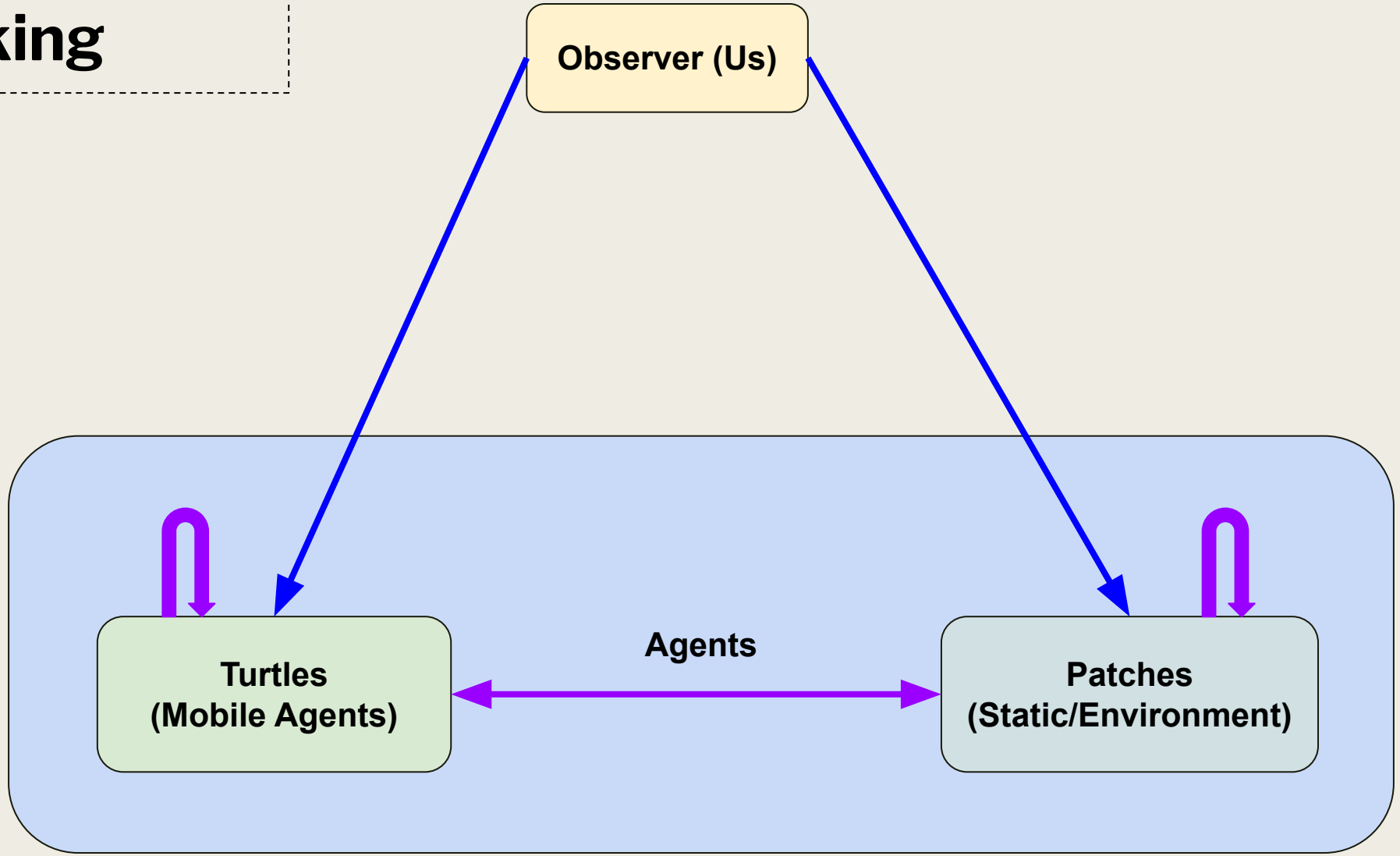
The Setup


Observer (Us)
Agents
Turtles
(Mobile Agents)
Patches
(Static/Environment)

Creation


Observer (Us)
But only in Setup (can't create new patches during simulation)
(aka hatching)
Turtles (Mobile Agents)
Agents
(aka sprouting)
Patches (Static/Environment)

Commanding /Asking



Context

- Every piece of code executed has a “context”
- Put simply: “context” represents “who executes it”
- The default context is the observer context
- You can use ask to switch context
  - *You can also establish context in command-center & directly set context in buttons (But it’s very uncommon)*

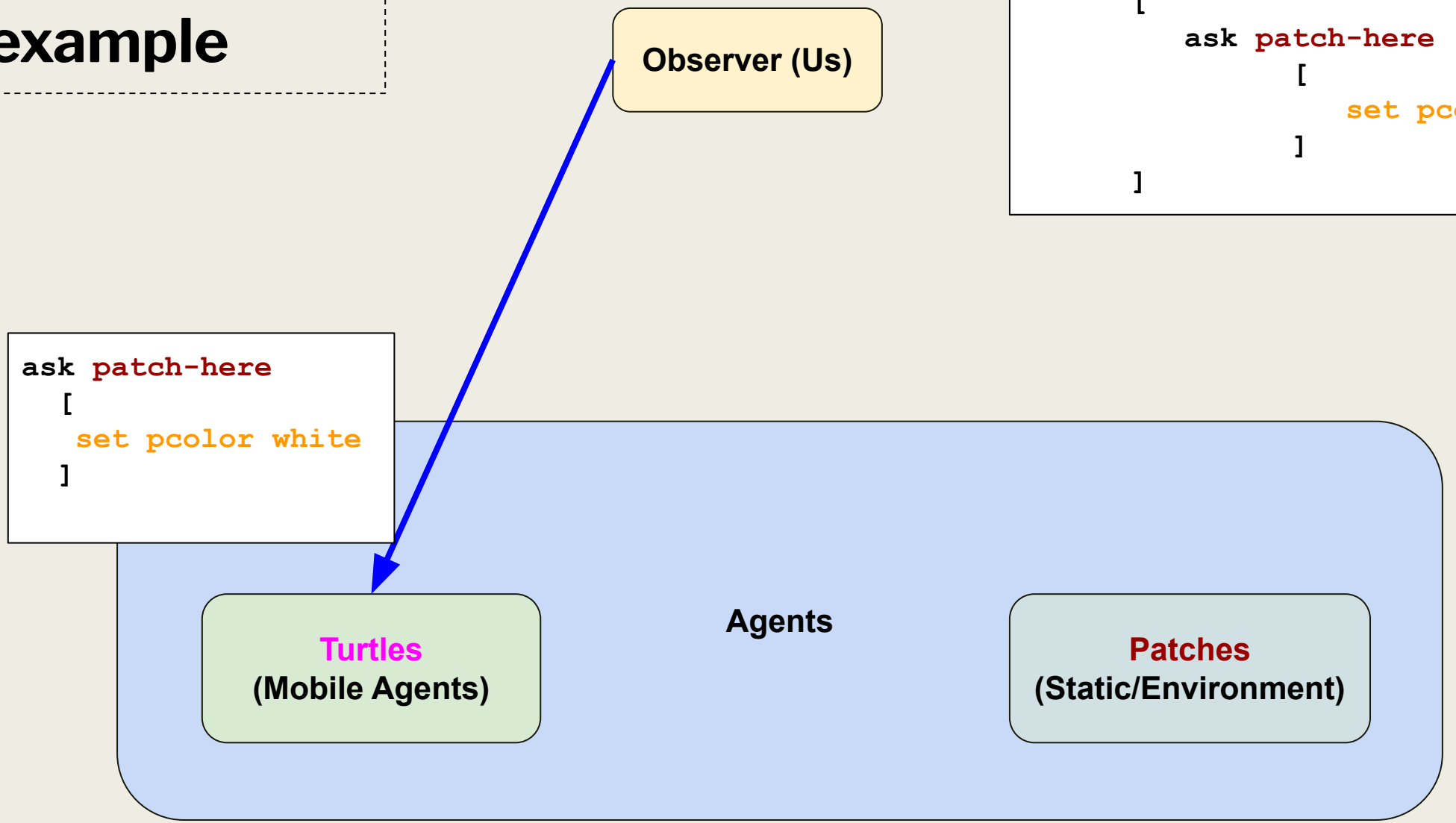
Context example

Observer (Us)

```
ask turtles
      [
         ask patch-here
               [
                  set pcolor white
               ]
      ]
```

Current Context: Observer

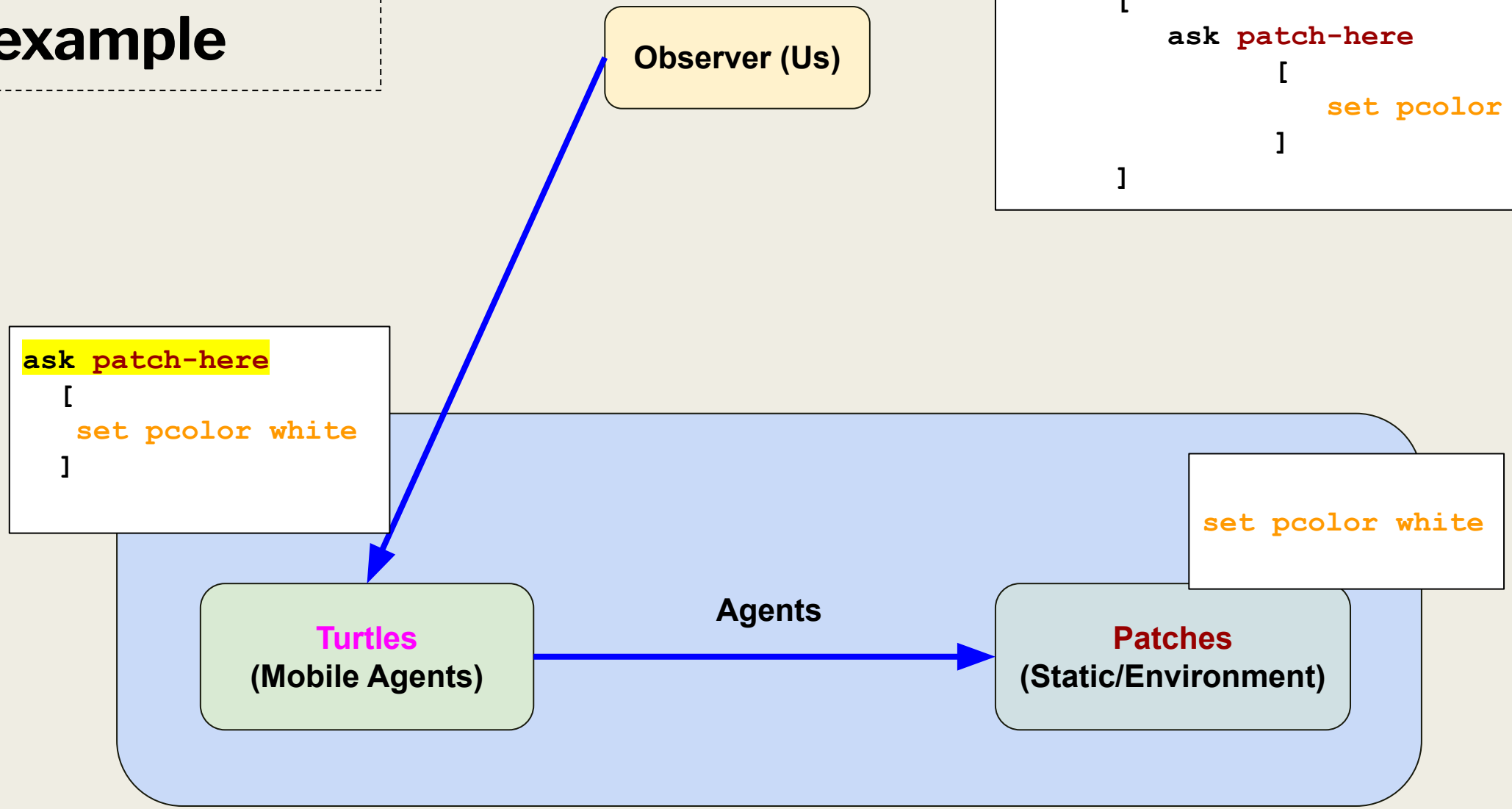
Context example



(Context Switched)
Current Context: Turtles

Context example

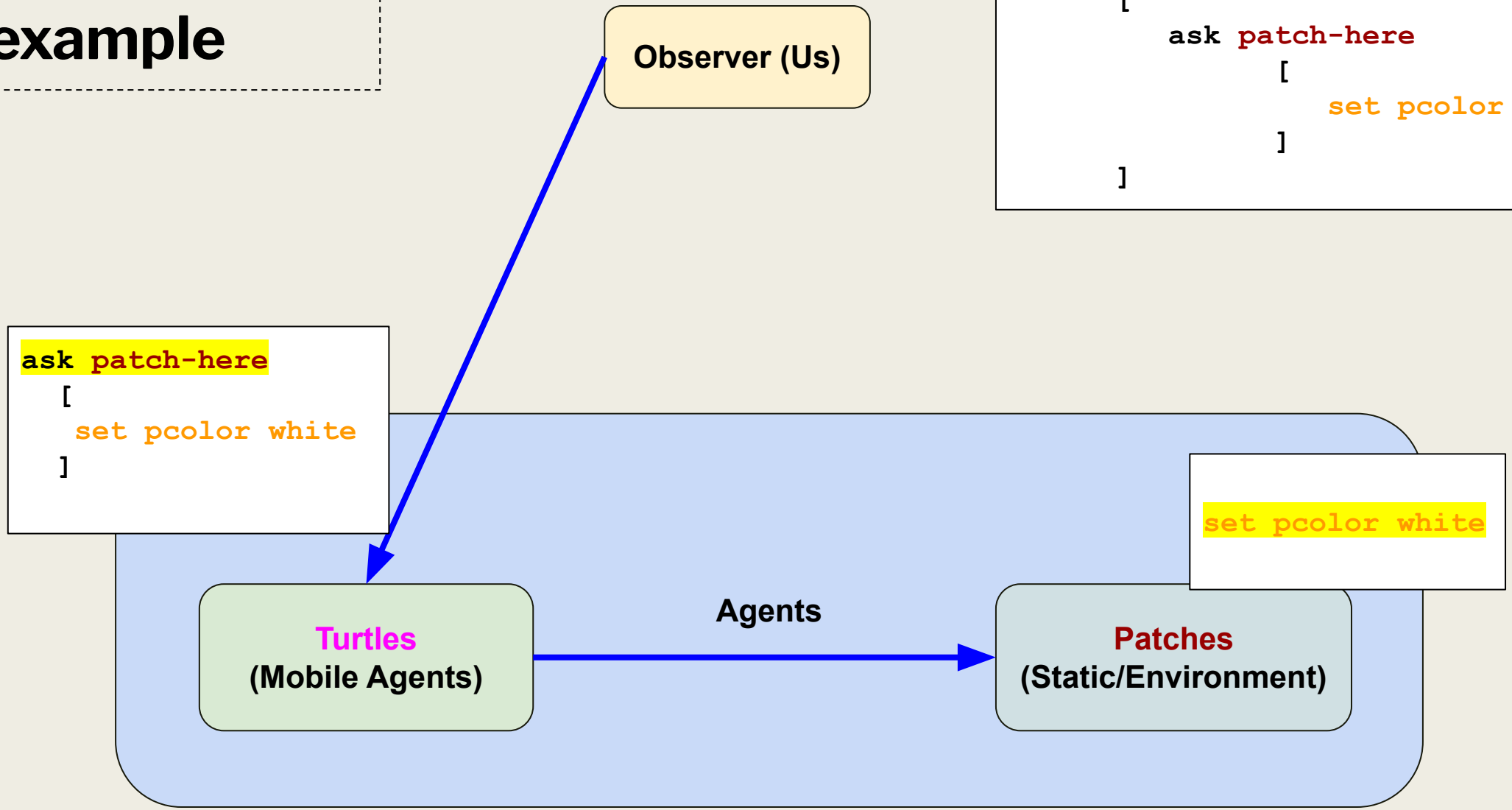
```
ask turtles
      [
         ask patch-here
               [
                  set pcolor white
               ]
      ]
```



(Context Switched)
Current Context: Patches

Context example

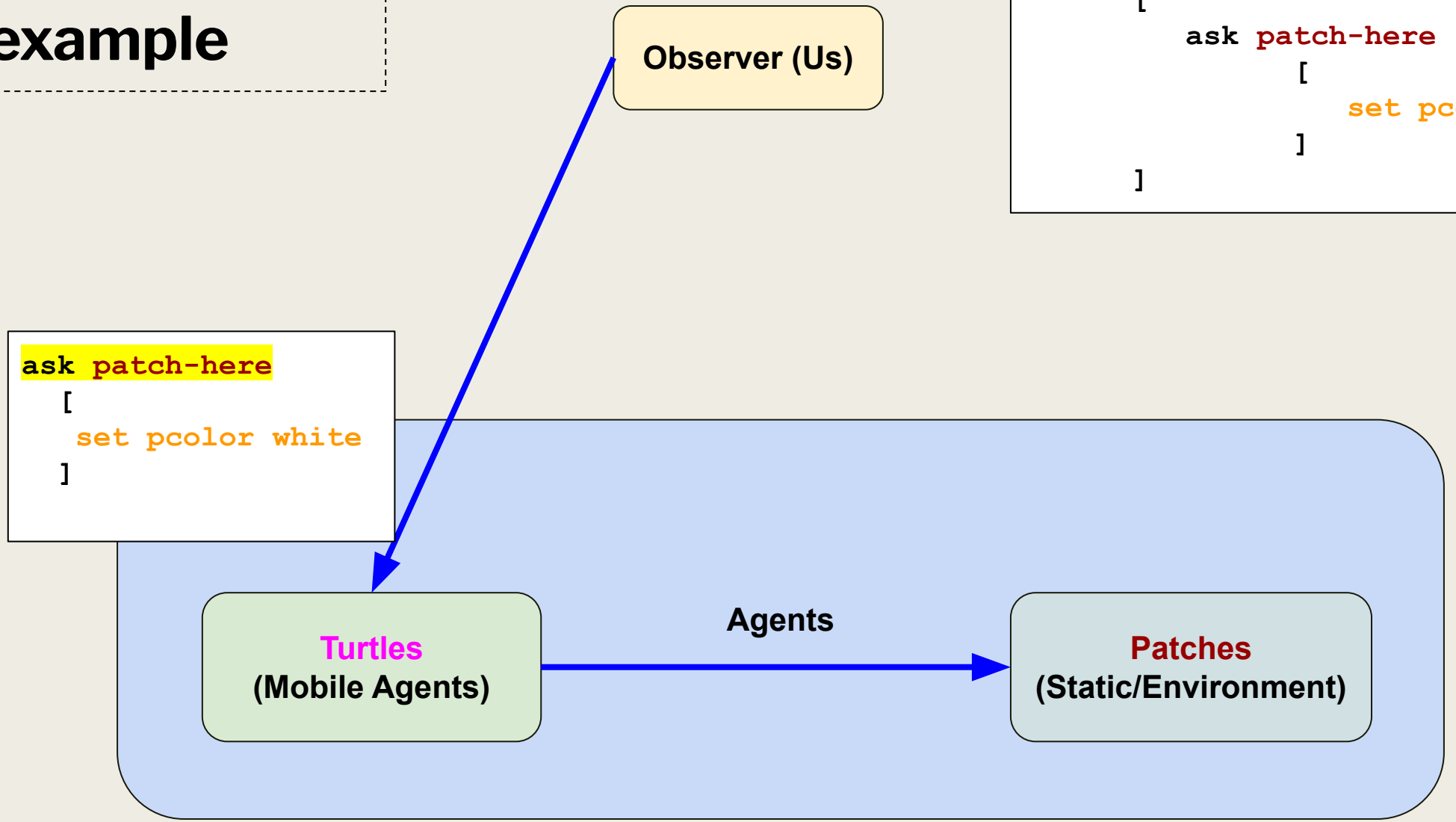
```
ask turtles
      [
         ask patch-here
               [
                  set pcolor white
               ]
      ]
```



Command executed in Patches context

Context example

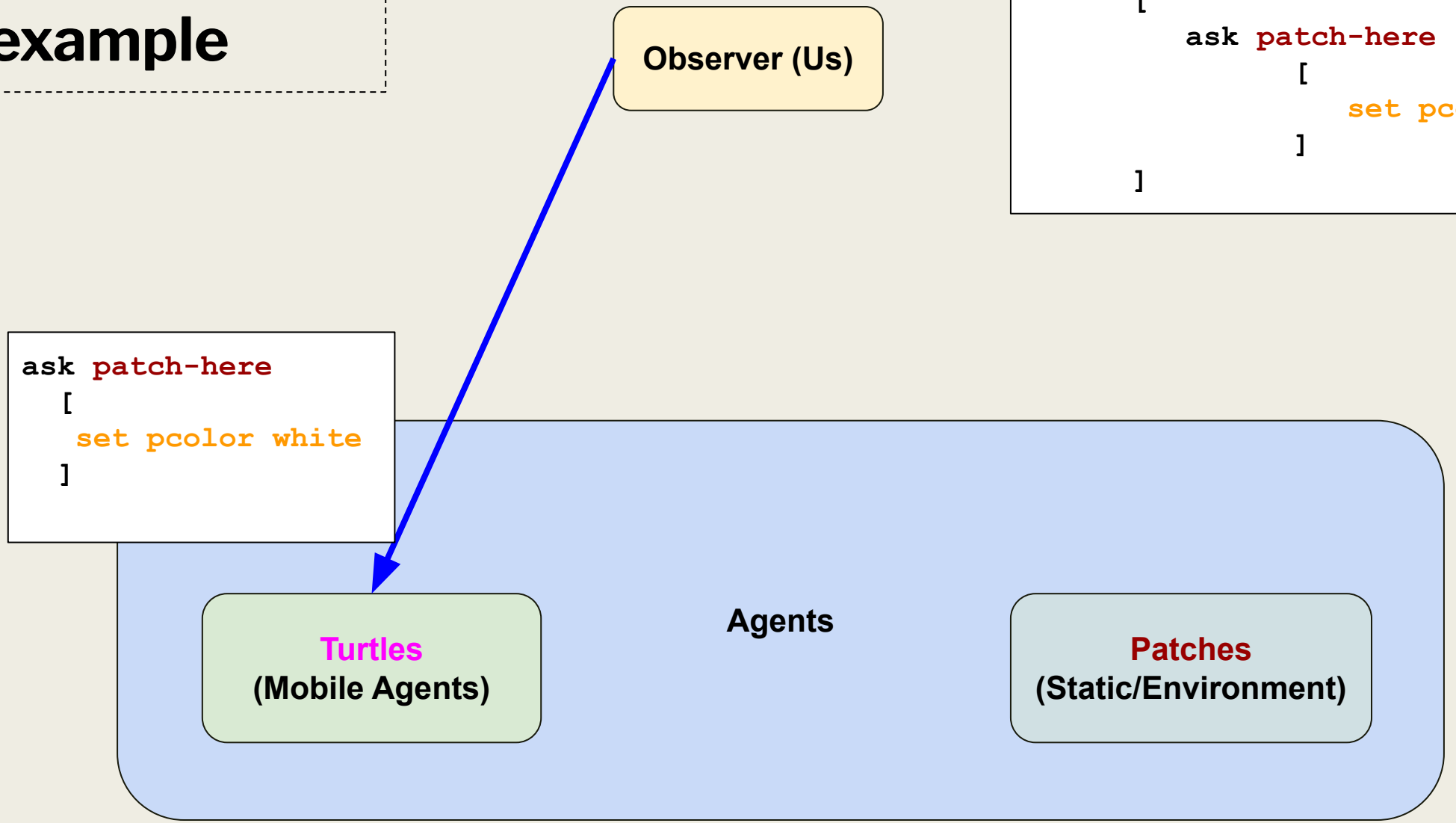
```
ask turtles
      [
         ask patch-here
               [
                  set pcolor white
               ]
      ]
```



Execution completed - pcolor is now white

Context example

```
ask turtles
      [
         ask patch-here
               [
                  set pcolor white
               ]
      ]
```



Execution completed - turtle command is completed

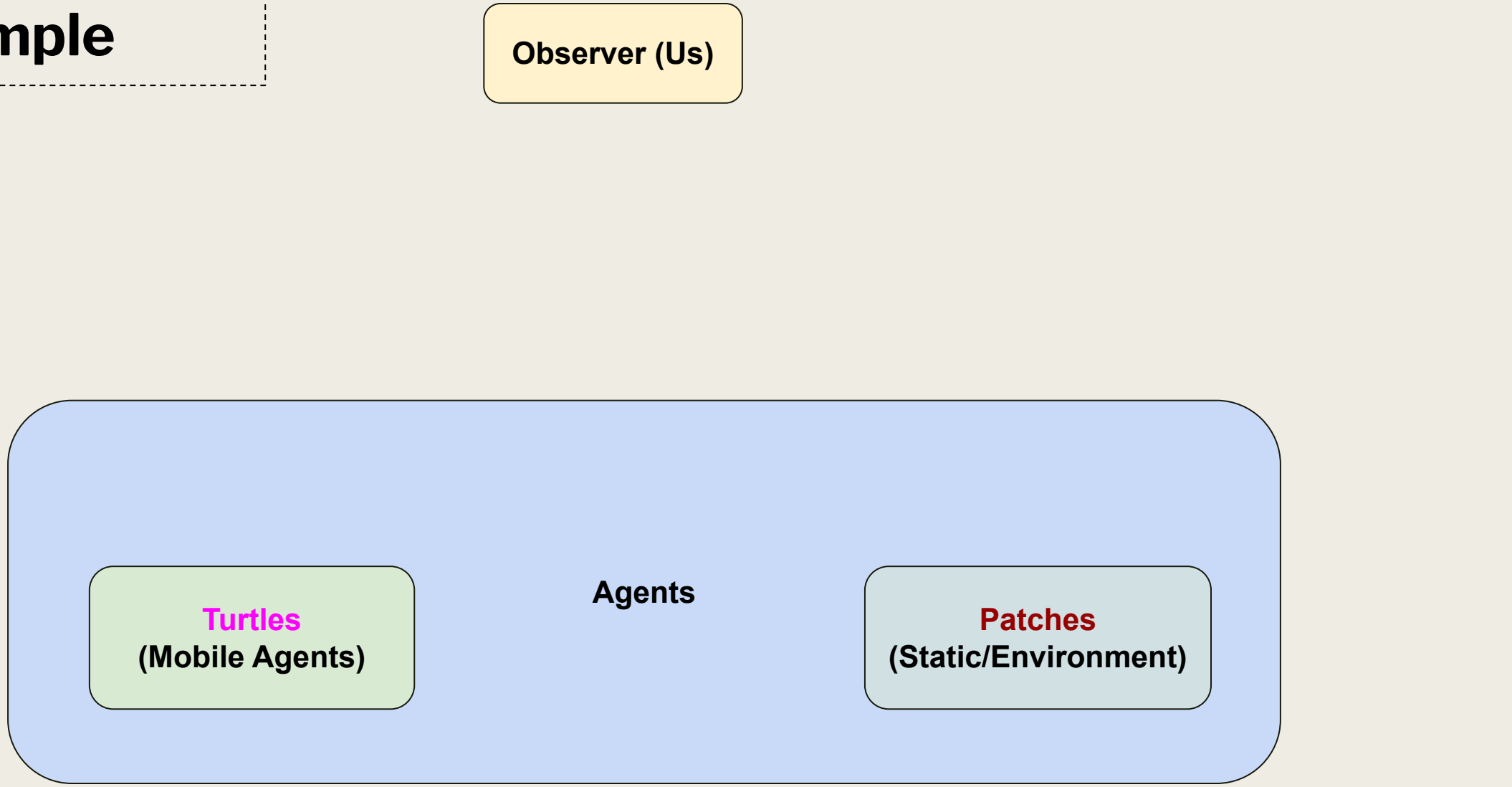
Context example

Observer (Us)

```
ask turtles
      [
         ask patch-here
               [
                  set pcolor white
               ]
      ]
```

Execution completed - turtle command is completed

Context example



Execution completed - observer command is now completed

NetLogo Commands, Reporters and Procedures

1. **Commands**
   a. *directives given to an agent: turtle(s)/patch(es)/link(s)*
2. **Reporters**
   a. *queries (request for report) given to an agent/observer, which in turn returns the answer for the query*
3. **Procedures (user defined)**
   a. ***command-procedure**: A sequence of commands*

      starts with ***to*** and concludes with ***end***

   b. ***reporter-procedure**: A sequence of computation which returns a value*

      starts with ***to-report*** and concludes with **end**

Lets see a demo!

5. Analyze the model

- Subcomponent or model validation
  - *Isolated sub-models to test specific behavior*
- Experimental runs and interrupts to catch extreme behavior
- Revise the model as necessary
- If satisfied, run the experiments and capture results

6. Communicate the model and the results

- **Info tab** should explain each element of the interface and what happens underneath
- Document the model (using **ODD** or other formulation)
  - *Detailed enough for replicating the model*
- Make the model available for other researchers
  - *To scrutinize*
  - *To expand*