cadCAD modelling process

This document uses/references sources <https://community.cadcad.org/> <https://github.com/BlockScience/cadCAD> and is intended to help [cadCAD Applied](https://community.cadcad.org/c/cadcad-applied)

Ideally, the applied researcher creates a thread for their case in cadCAD Applied, makes a copy of this template, and works through it, replacing the generic with the domain specifics.

Any improvements needed that surface as part of that process should be discussed under the [Process Template topic](https://community.cadcad.org/t/cadcad-modeling-process-template/53/3).

# Requirements

### Design Thinking

# The System

## System Purpose/Goal - Problem statement

Before starting to design or redesign the problem statement should be reviewed; what is the purpose of this system and why does it need to exist? In linear business model terms, this is the vision/mission. This is the starting point of a highly iterative systems design process

Or if you have an established architecture, but your system doesn’t “behave as intended”. That’s also why you would use cadCAD to figure out the dynamics of the parts of the system, e.g. to discover wrong assumptions about one part in your system, change them, and improve, or do a [parameter sweep](https://community.cadcad.org/t/on-modeling-agents-in-dynamic-systems/23) and [Monte-Carlo simulations](https://community.cadcad.org/t/introduction-to-simulation-configurations/34). In [future versions of cadCAD](https://community.cadcad.org/t/on-modeling-agents-in-dynamic-systems/23/2) we will see reinforcement learning coupled to the real system by [connecting to its online data](https://github.com/BlockScience/cadCAD/tree/distroduce) pipeline.

*Note: The cadCAD framework as of today already allows you to capture micro-dynamics, and macro-dynamics of your system -* [*it’s a multiscale modeling & simulation framework*](https://community.cadcad.org/t/introduction-to-cadcad/15)*. “Modeling system level properties using differential equations is called System Dynamics (SD) whereas modeling the way agents behaviors interact is called Agent Based Modeling (ABM). cadCAD is the first framework that unites both.”*

## Systems Requirements

Mapping out a robust set of systems requirements is one of the most important steps in the design process. Even if you have an existing system, already designed, it would worth reevaluating what

## System Overview

Draw a diagram of the ecosystem if the system architecture is known to you..

If not, start with discovering, detailing and designing your ecosystem. It might be necessary to redesign a system if the emergent behaviours have too many unintended or unwanted consequences :

Here are a few templates that can help you [link to empty canvas & matrix]. You can think of this part of the process as taking a walk on a nice little patio with three stepping stones:

1. Map all the stakeholders and influencers of your system that you can think of on the canvas
2. Map value exchange between the ecosystem members, both economic and non-economic. The matrix primes your mind to think in terms of Value Creation and Value Sharing.
3. Back on the canvas draw the most fundamental value flows between the stakeholders to reach the system goal and/or fulfill its purpose.

You might see that your system has multiple purposes, it has multiple dimensions (than you’ll have a cube and tensors instead of a matrix ;). These dimensions relate to different types of values flowing in the ecosystem (you can still flatten them into 2D - by fixing some of the dimensions in a given order (and hope reality behaves like that order), or handle them in their multidimensional beauty). No one said, it’s easy, we’re here to model complex adaptive systems. Our goal is to put you at ease with the complexity that you are facing, because now you have a process and a framework.

After you’re finished with the first iteration of the canvas, take a step back and take a look:

* Can you now name/refine the system purpose(s)/goal(s)?
* Which of those goals are complementing, compounding, which are contradicting
* Contradictions shouldn’t be ignored: by keeping those values, flows, stakeholders in the system you are weakening your system - don’t take our word for it! Model, simulate and compare the outcomes. Then refine the system purpose and it’s boundaries:
  + If you cannot find mechanisms to align contradicting goals, you might need to exclude those stakeholders that do not share the purpose - but keep them in the model as external influences perturbing the system.

## Agents

“Since we are interested in modeling social and economic systems we are constantly asking [what our agents will do](https://community.cadcad.org/t/on-modeling-agents-in-dynamic-systems/23).” When you are looking to model beyond digital only, but cyber-physical systems that your humans interact with, your machines that are automated (or autonomous) should also be modelled as agents. In order to decide how to capture their logic, you have to layout their motivations and mechanics, their value-adds and output, their needs and input. Then you will be able to decide on one of the following [ways to approach the behavioral models](https://community.cadcad.org/t/on-modeling-agents-in-dynamic-systems/23) of your agents most appropriately:

1. encoded with Heuristic strategies derived from game theoretic, psychological decision sciences and/or behavioral economics literature.
2. machine learned from past data where the feature space is some characterization of the agent and system states, and the labels are the actions taken.
3. inherently adaptive strategies by encoding them as reinforcement learning agents who will learn to do whatever they can to achieve their goals within the bounds of the action space.

Models of all three types can be implemented in cadCAD; it is even possible for all of them to be used in the same model.

### Human Agents (Motivations/Value-adds/Needs)

In the following list all human agents and add a coherent description of your assumptions and knowledge about them. Their goals can be encoded with heuristic strategies derived from game theoretic, psychological decision sciences and/or behavioral economics literature or machine learned from past data where the feature space is some characterization of the agent and system states, and the labels are the actions taken.

### Machine Agents (Mechanics/Output/Input)

In the following provide a description of our machine agents. Machines can and should be modelled as agents, if they have some ability to 'observe state and make decisions' e.g. anything with a decision/control system (which includes sensors). The more data-driven, and adaptive the control logic of your machine agent is in reality (e.g. smart grids, autonomous cars) the more it will make sense to include them in the system model.

*Note: In future versions, when cadCAD gets more capabilities to plug-in data from the real agents the more useful your modelling, simulation, and optimization will be to your product development and deployment.*

The model should be granular enough, or else you might want to opt to model the effects of these machine agents as environmental processes (see next section). Their goals can be derived from their control-theoretic models, or they can have inherently adaptive strategies by encoding them as reinforcement learning agents who will learn to do whatever it can to achieve its goals within the bounds of the action space.

## Environment

A system is delineated by its spatial and temporal boundaries, surrounded and influenced by its environment, described by its structure and purpose, and expressed through its functioning. A complex system is a system where the behavior of the whole within a given environment cannot be understood separately by understanding the behavior of the parts, the behavior of the whole, and the environment as separate additive entities. Rather, the system’s behavior is deeply dependent on the relations among the parts. Complex adaptive systems, in which in addition to this non-linearity, non-ergodicity is captured - that is when something happens, then the possible outcome, even the environment that sets the system boundary is changed.

In the canvas from the System Overview, you will have identified external variables, internal or external agents that will affect agents in your model and the system as a whole but not the other way round. You would model them in cadCAD as an environmental process.

[Environmental Process Model](https://community.cadcad.org/t/differential-specification-syntax-key/31) is a modelled generator providing an update to state variables using external input signals from an environmental process driver. User action behavior and control functions should not affect this model. The affected state variables should be purely exogenously based, but may be blended from the internal state if necessary to model the system. The exogenous state variables are the link between the changes from the outside world germane to the modeled system, and provides the information necessary for users and controllers to make decisions.

[Environmental Process Drivers](https://community.cadcad.org/t/differential-specification-syntax-key/31) are data feed that informs an exogenous process. This may be real or assumed data to provide updated input for a model representing a change of external conditions.

As a token engineer you need to take factors into account that will perturb your system, and the token mechanisms that can be designed to increase resilience when faced with these factors.

But It may also be that you want to simplify some of the agents’ behaviors by fixing them to show one realization of observable state, in order to simulate higher leverage points in the model that show the token economy’s resilience. Again this is when you would take in a series of real system/agent behavior data or synthesize and create an environmental process driver.

# Stock and Flow Diagram

The ecosystem matrix above, helped to make value creation (stocks) and value sharing (flows) more explicit. These might need to be elaborated more, before high leverage points of the systems can be identified. As we identify and model more subsystems, we will have an increasingly better understanding of the outcomes the system dynamics creates - and be able to zoom in and out as needed to explain.

*Note: In cadCAD submodels can be wired together to get a “bigger picture.” It’s also possible to plugin models created with other tools (via environmental process model).*

Stock & flow diagrams then help in describing high leverage subsystems/concerns more concisely, improve and logical fallacies and fill in the gaps in discussions with fellow system/token engineers. They are a great step for handover into more cadCAD specific differential specification, which maps directly to coding blocks in cadCAD.

[Paste example]

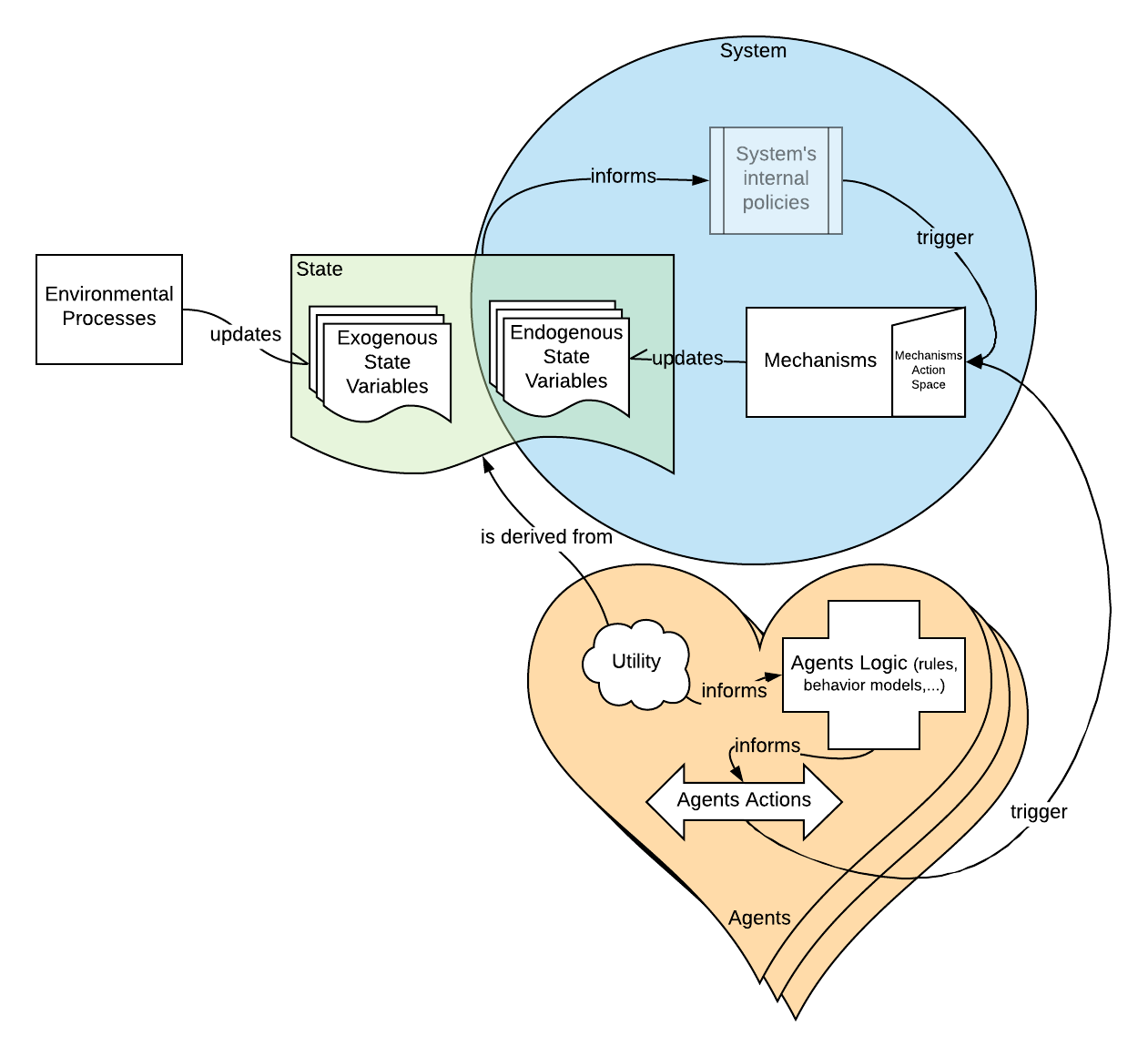
[Template]

# Differential Specification

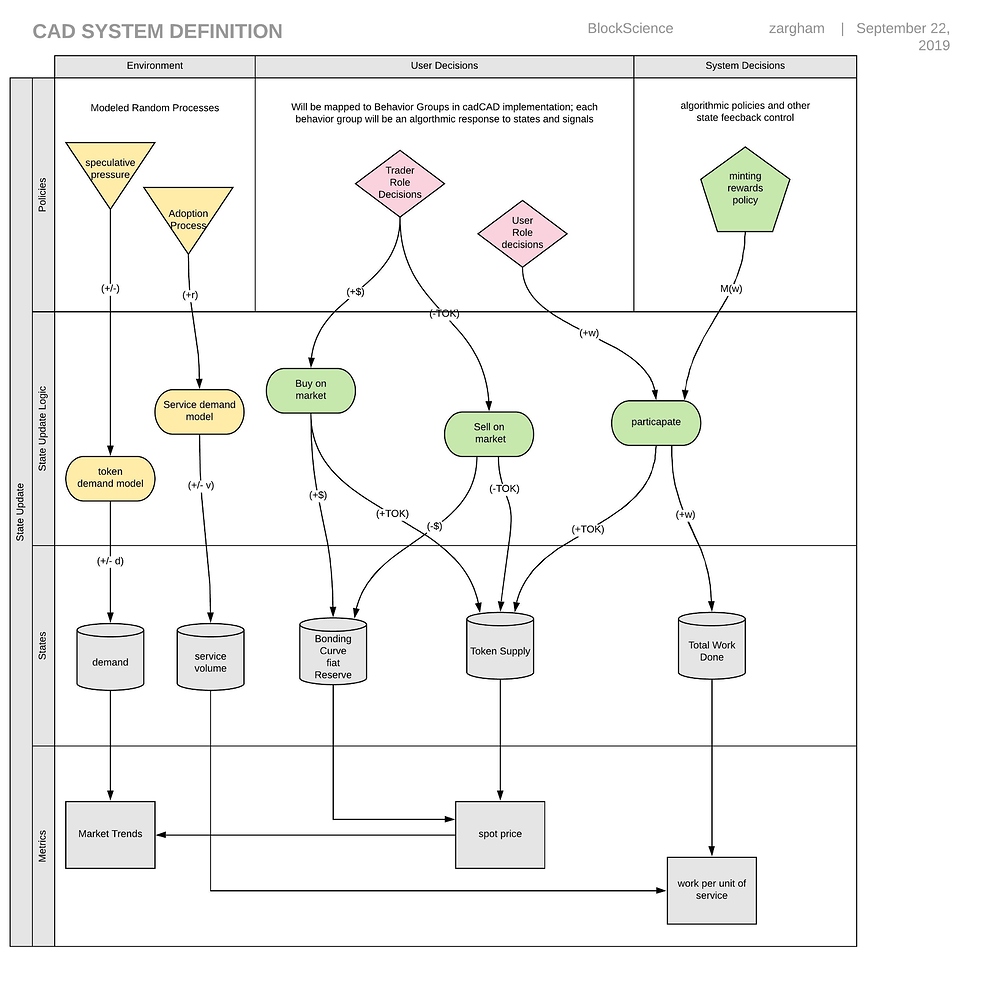
The differential specification is effectively described through a system modeling syntax. The symbols referenced in the differential specification diagram are described here. Note that all information flows described are feed-forward, feedback of state to any process is implicit. Furthermore, the differential specification is laid on a grid where the rows are:

* The **Policies** determine the inputs to the system dynamics regardless of whether they come from the outside environment, user behavior or algorithmic decision systems.
* The **State Update Logic** determines how the state of the system changes as a result of the inputs provided. Note that not all outcomes are the outcomes intended by the actor (system or agent) when the input decision was made due to the interdependence of these inputs.
* The **State** is flattened and summarized in order to keep track of the the interdependence of the mechanisms on the same state variables. Note that the full data model for the state space is generally a network object for which the shown states are mid-level aggregates or summary statistics.
* The **Metrics** are values computed from the state variables in order to assess the economic system. In a computational model, metrics may be used for governance decisions, algorithmic policies or to inform the behavior of individual agents.

The columns in a differential specification refer to the **Partial State Update Block** also called *substeps*, which break the logic up into an ordered sequence in order to support second order dependencies. For example, there may be a set of user behavior functions that resolve a decision to participate and incur some costs, and then a set of environmental process drivers that determine the rewards those participants receive as a result of all of their decisions and the events which transpire there after. A simple differential specification will have one or two substeps, whereas a complex economic system could have 8 to 10 substeps.

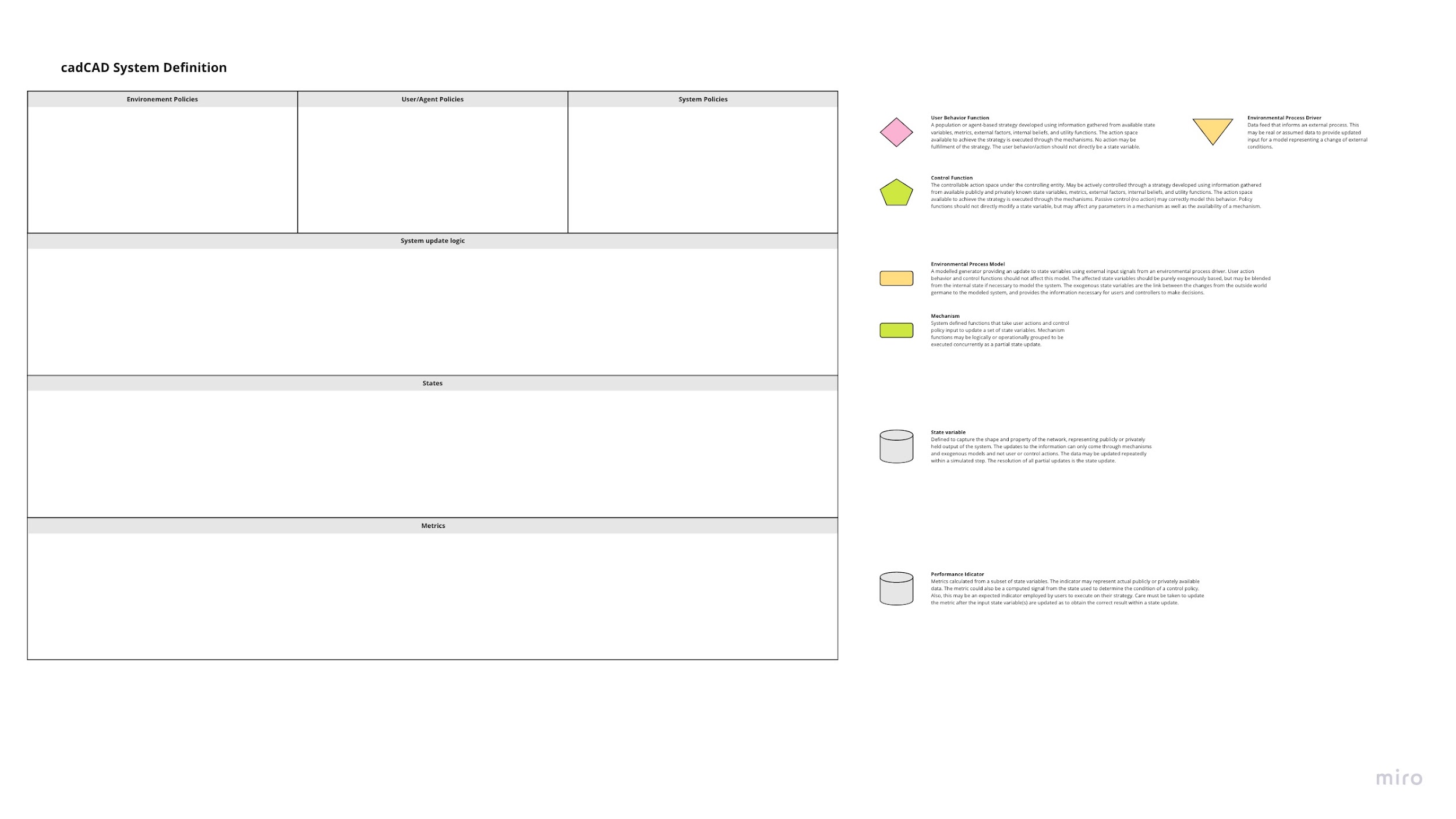


## Example

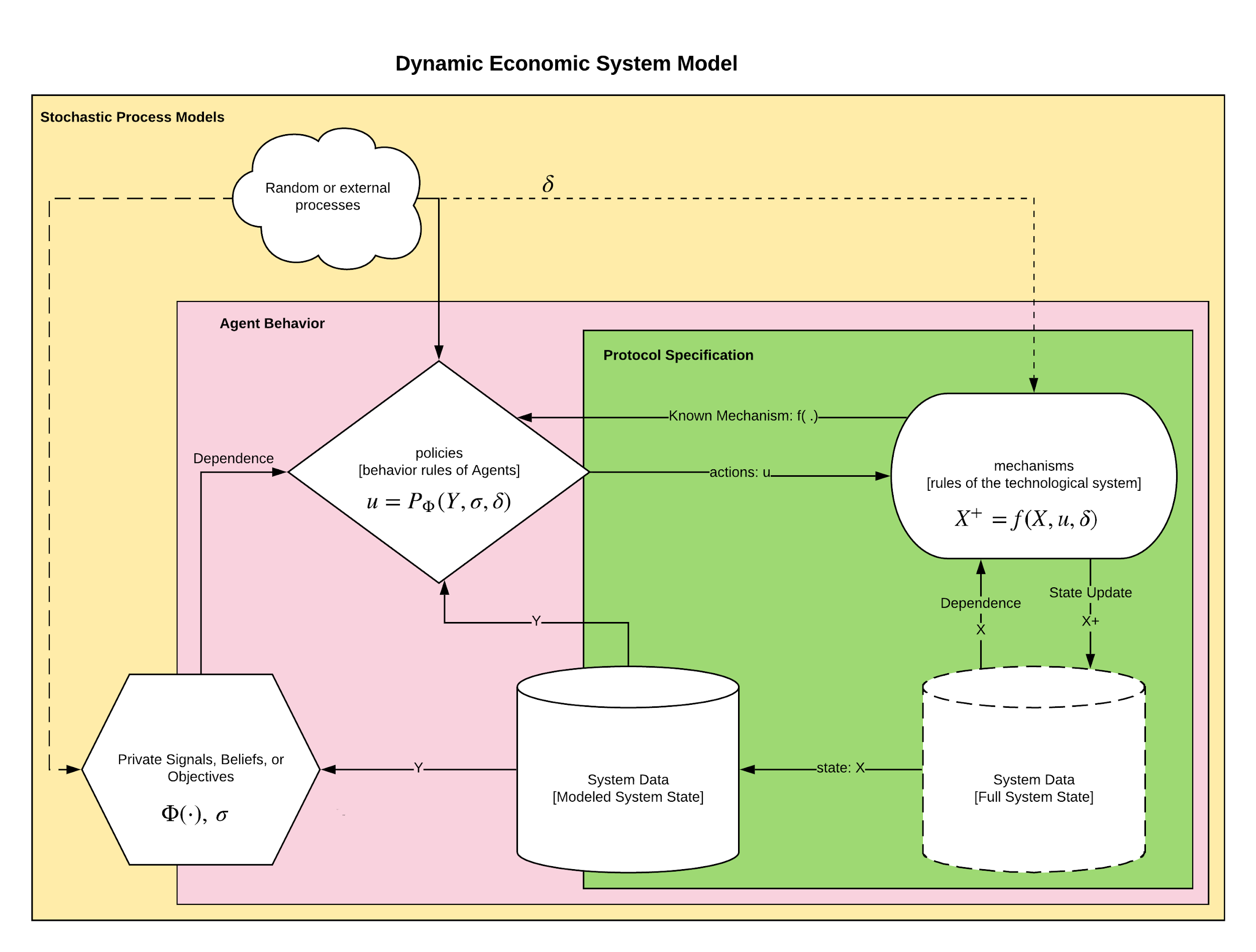


## Template

There is a Miro [template](https://miro.com/welcomeonboard/bYehGqVX7XXtEPx5myNu9ZogT2dkq0Zf73NAuH4XOmbW75isYXjTv4PcOBrsheBD) you can use to get you started, just copy the board to your own board and edit from there.



# Partial Differential Equations describing system dynamics



# Coding the model in cadCAD