



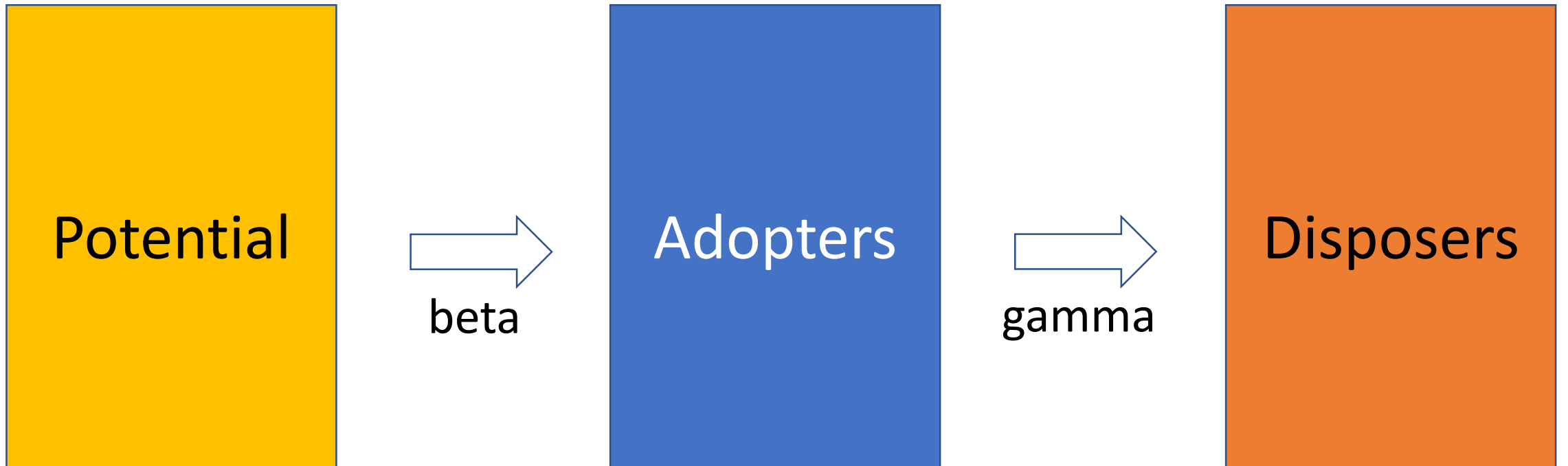
04-3 Python Application Diffusion of Innovation Analysis

CSI 500

Modeling Diffusion of Innovation in Python

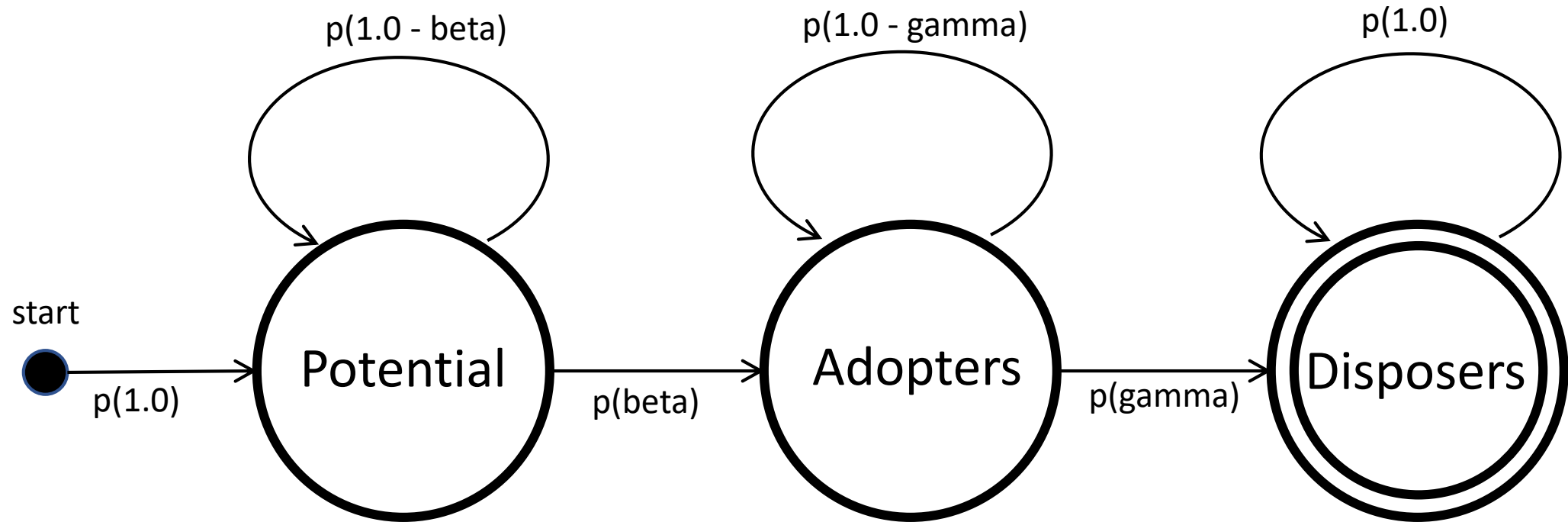
- Let's return to our semester theme on Diffusion of Innovation
 - We can build a model of a population of individuals exposed to a new concept or technological innovation
 - Individuals decide to adopt, then later discard, the innovation
- How to model this?
 - We can use a 3-cell compartment model: potential, adopters, disposers
 - Set parameters for total population size N
 - set parameter for adoption rate β
 - set parameter for disposal rate γ
 - set parameter for maximum time range \max_time
 - save data recording the population dynamics over time
- Major simplifications
 - the individuals don't have autonomy - their behavior is aggregated statistically
 - this type model is widely used in epidemiology, social networks, and marketing/business

Model Overview: Block Diagram



Total population size = N

Model Overview: Finite State Machine



Model functional spec - initialization

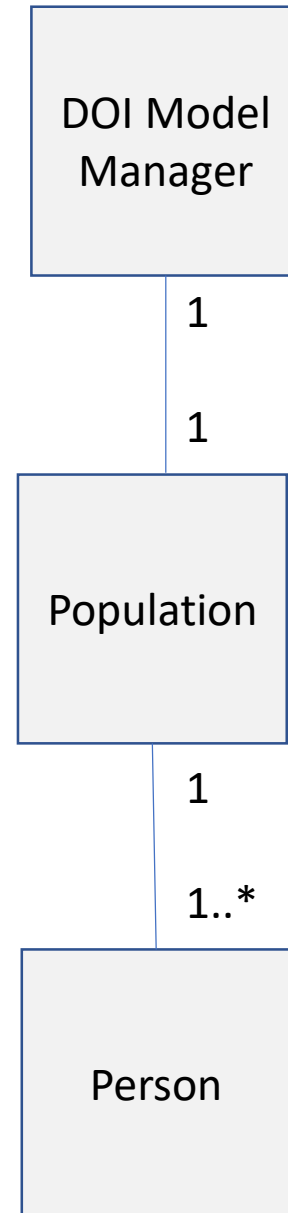
- To configure the model, we select a value for *beta* to manage flow of potential persons to adopter persons
- We also select a value for *gamma* to manage flow of adopter persons to disposer persons
- We select a value for initial adopters, say K . We then set the initial size of the potential pool to $N-K$
- For convenience, we assume there are no initial disposers

Model functional spec - run time

- Simulate a user specified period of time steps *max_time*
- At each time step, compute how many persons change state and move them to their appropriate new state
 - from potential to adopter: $\beta * \text{number of potentials}$
 - from adopter to disposer: $\gamma * \text{number of adopters}$
- Provide capability to graphically render these quantities
 - use a Python plotting library to make a diagram of quantities over time

Model class design

- Let's consider the things we'll need
 - A Person model to manage the individuals in the simulation
 - A Population model to manage the Persons as a group
 - A Diffusion of Innovation (DOI) model to run the clock and manage the Population
- We can use Python classes to express these constructs



Summary

- Before coding, analyze the problem space
 - block diagrams
 - finite state machines
 - class models
 - high-level functional specifications
- Python classes used to create a diffusion of innovation model
 - Person class for a person
 - Population class for a population
 - DOI_Model class manages the simulation
- Next step is to start developing the classes we'll need