THE SHEPHERD, THE SHEEP AND THE ALIEN

Story

- Human is on the brink of extinction
- Alien’s attack devastate every civilization
- You are the last hope to unlock the key to survive for mankind

Gameplay

- Lead the people to safe area
- Chase away the aliens
- People/Sheep: mindless drones, follow the player everytime and run away from the aliens.
- Alien/Hunter: work in group and hunt down any people they see bruthlessly.

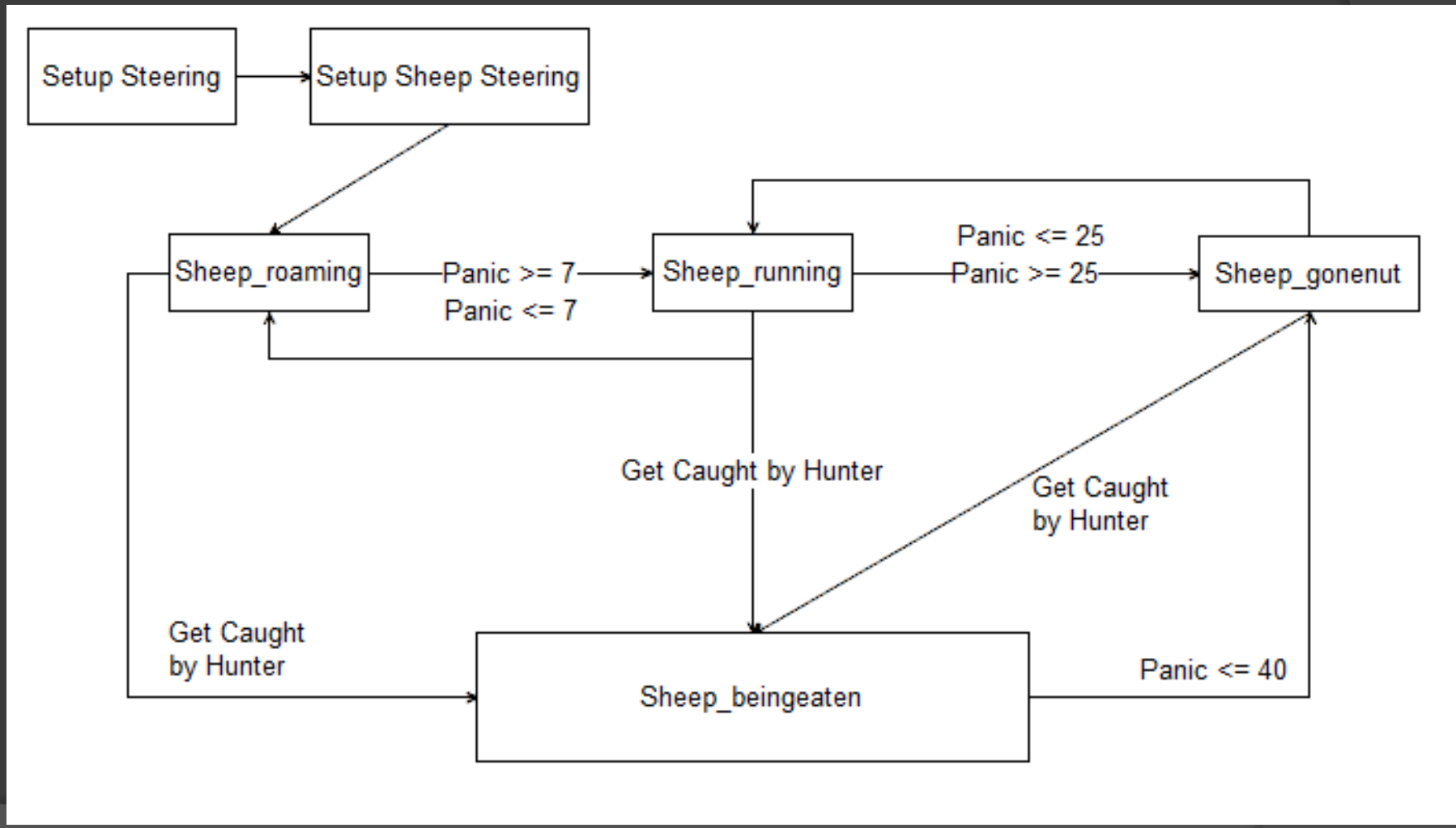
Finite State Machine

- Each state is built on an interface.
- Enter, Run, Exit
- Enter: setting up steering behaviour, connect Brain, characteristic.
- Run: run every frame, check for triggers to change State
- Exit: prepare to exit state.

Finite State Machine (cont)

- Sheep:
  - Roaming: roam around. Panic level none
  - Running: panic level low. If the sheep runs too much, its panic increases overtime.
  - Gone nuts: the sheep is so scared to run and accept death. Panic level is high.
  - Being eaten: the hunter has caught the sheep and about to eat it. Panic is at maximum

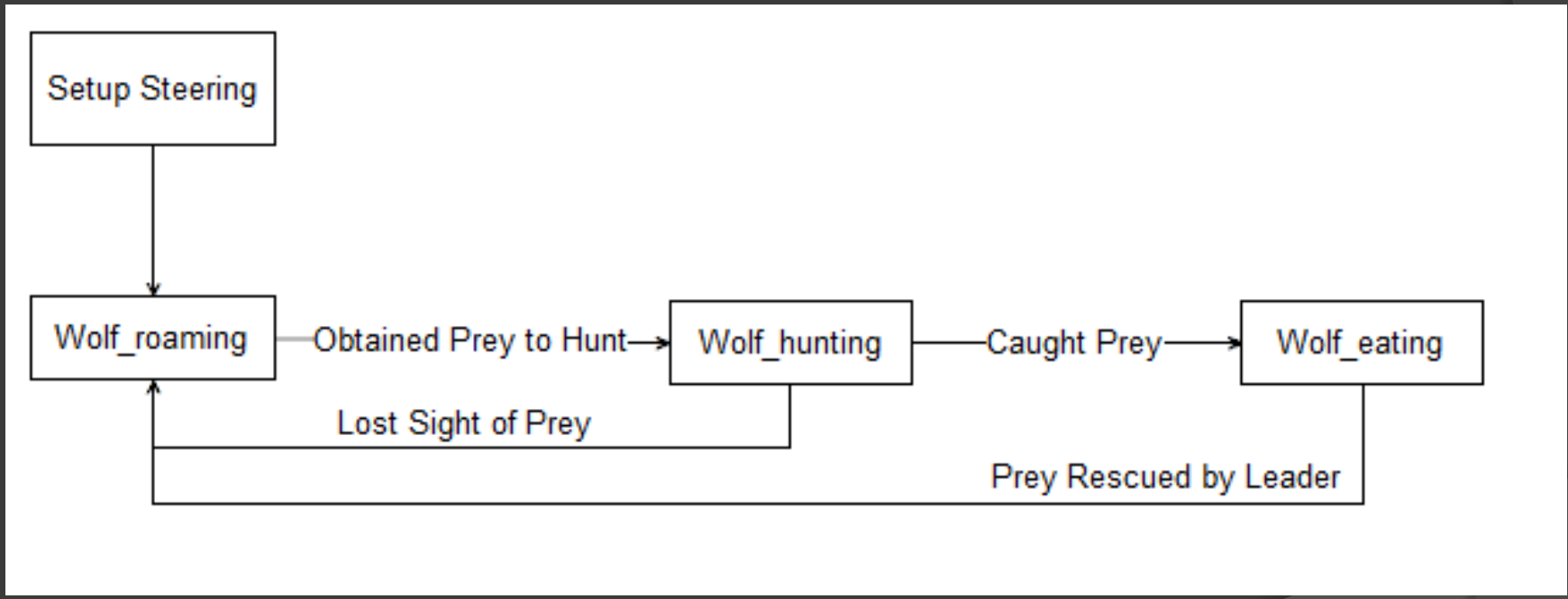
Finite State Machine (cont)



Finite State Machine (cont)

- Hunter:
  - Roaming: finding for sheep, pick target and send command to others
  - Hunting: hunt and seek the sheep
  - Eating: eat the sheep. Player can chase the hunter away to rescue the sheep. If player fails to chase the hunter away, the sheep will be eaten in an amount of time based on its panic level.

Finite State Machine (cont)



Steering Behaviour

- It outputs a desired velocity
- Has 2 weights: internal and external to manage the behaviour as desired
- Complex steering (flocking) can be created by combining multiple simpler behaviours.
- Using grid to steer the agent to visit each grid square along the path

A* Pathfinding

- Manhattan distance as the heuristic
- The algorithm piggybacks off the Grid data structure
- How it works:
  - Initialise a map to store any A* nodes.
  - Initialise a map of Booleans as the closed set
  - Initialise a minimum priority queue for the open set

A* Pathfinding (cont)

- How it works:
  - Add the current position to the open set
  - While there is still a node in the open set
  - Mark the node as processed in the closed set and remove it from the open set.
  - If the current node is the target node, exit.
  - For every neighbour of the node, skip it if it's in the closed set