Element.AI Official Graphic Documentation

ELEMENT.AI

Saturday, February 18th 11AM - 6:30PM

Note: This documentation is meant to explain the Element.AI environment. Please see ***quickstart.md*** *for cli commands and other code-heavy explanations.*

Table of Contents

Overview

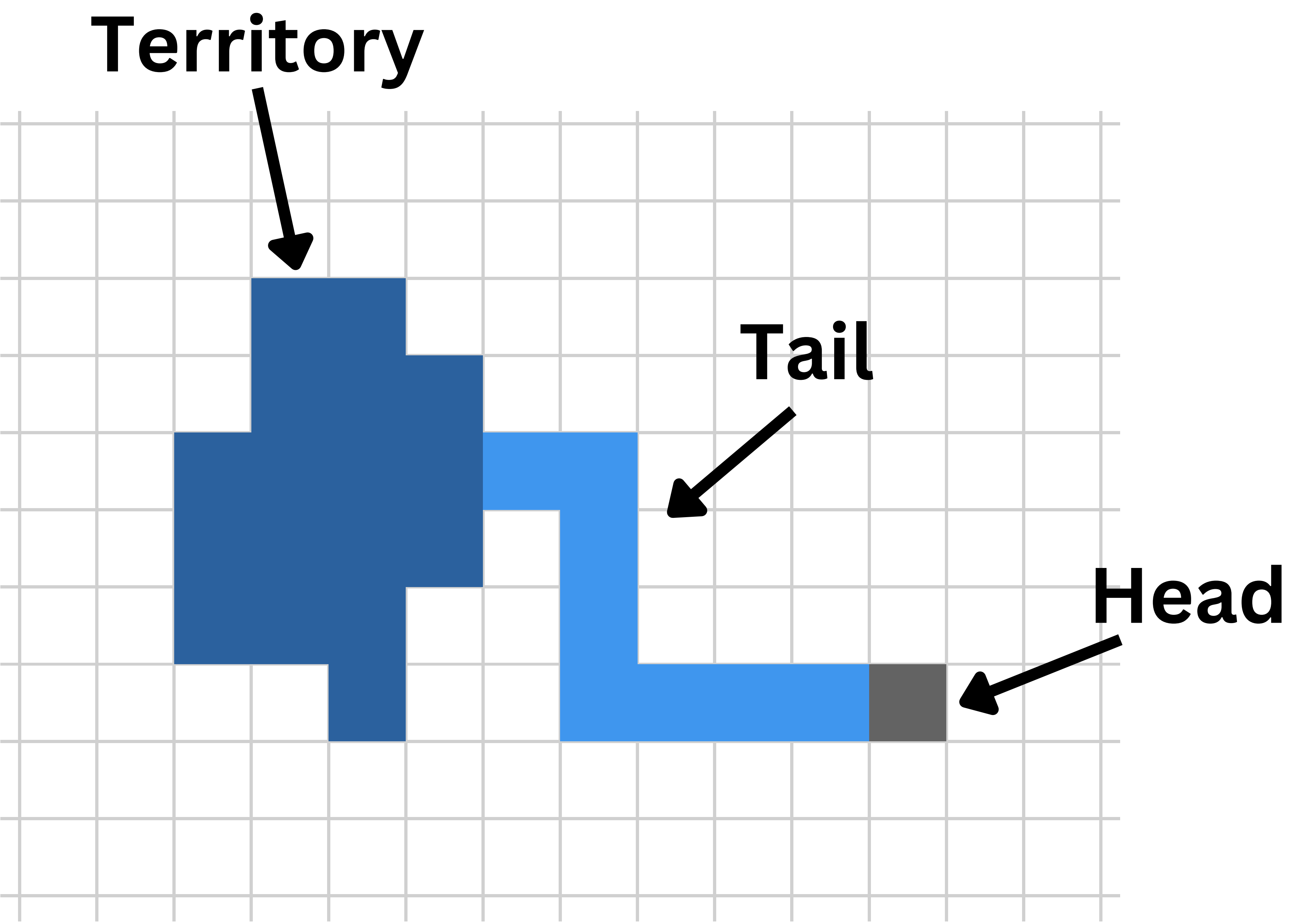
Basic Rules - 1 of 2

Your bot will take **observations** and return an **action**:

1. -1 to move left
2. 0 to move forward
3. 1 to move right

If your bot is not allowed to move, any action it returns will be ignored.
If your bot does not return a valid action, it will move forward by default.

The bot plays in a square grid. It has three components:



There are three ways a player can die:

1. Run into *any* player's tail (including their own) or *any* player's head
2. Run off the map
3. Run into a bomb

If a player dies, they will respawn, but their progress (territory, speed, etc) will be reset.

See **Environment Rules > Respawning** for more details on respawning process and penalties.

Overview

Basic Rules - 2 of 2

Your bot also has **energy** and **speed**. You may increase your energy by collecting **boosts** (1 boost = 1 energy).

Energy	0	2	4	11	20
Speed	1	2	3	4	5

You may gain territory by enclosing space. The definition of enclosing space is as follows:

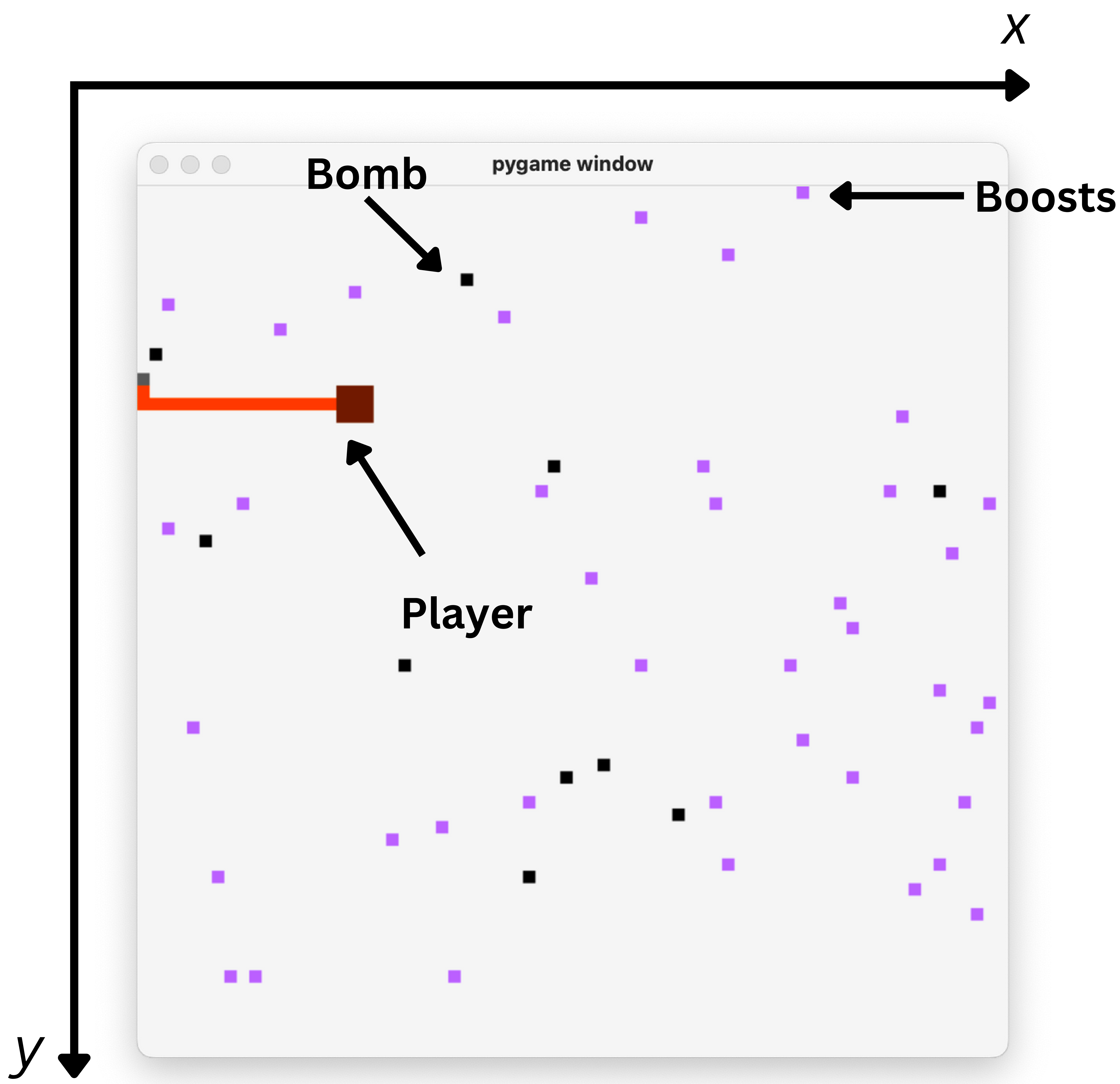
- Bot lands on a square marked as its own territory
- Its tail will be converted to territory
- Any tiles fully enclosed by its tail will also be converted to territory

The number of tiles in your territory will be added to your total score at each replenishment (see **Runtime Loop > Replenishing** and **Runtime Loop > One Iteration** for information on replenishment).

See **quickstart.md** for information on running the environment and action/observation structure.

Overview

Game Board



2p = 70x70
3p = 85 x 85
4p = 100x100

max bombs = 10
max boosts = 40

The board will is an nxn board (depending on the number of players). 2 players = 70x70 board, 3 players = 85x85 board, and 4 players = 100x100 board. All other board constants remain the same regardless of the number of players.

Note: the board's pixels are indexed from the *top left* (as with most displays).

Runtime Loop

Elements of Environment

Player Bot

Element.AI Environment

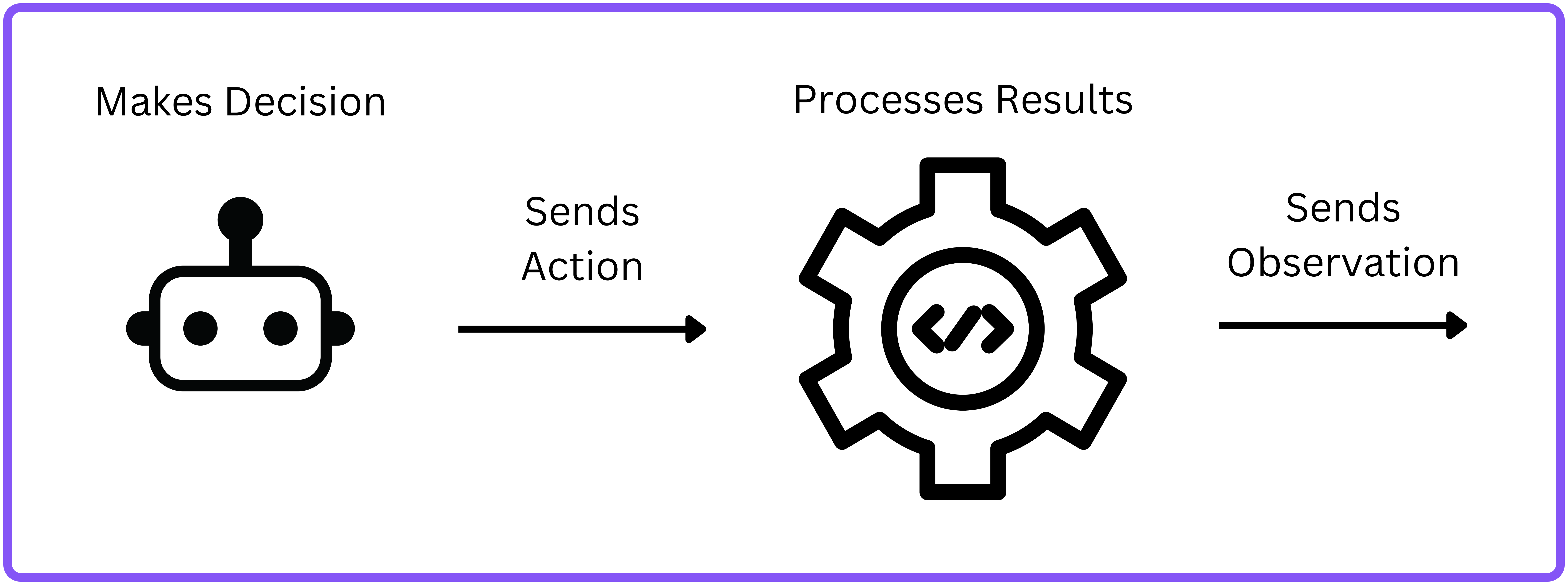
You may edit your bot under kits/python or kits/java
You may edit agent.py/agent.java, or add any files you wish

- Do *not* edit main.py or tools/

The environment is saved locally for testing under paperio/, but *do not* edit this. You may explore the environment code if you wish, but it likely won't help much.

Runtime Loop

One Step



Your agent received an **observation** from a previous iteration, and uses this information to output an **action**.

The environment will process this action (as well as other bots' actions), update the environment state, and output a new observation.

See the **quickstart.md** for more information on action and observation structure.

Runtime Loop

Replenishing

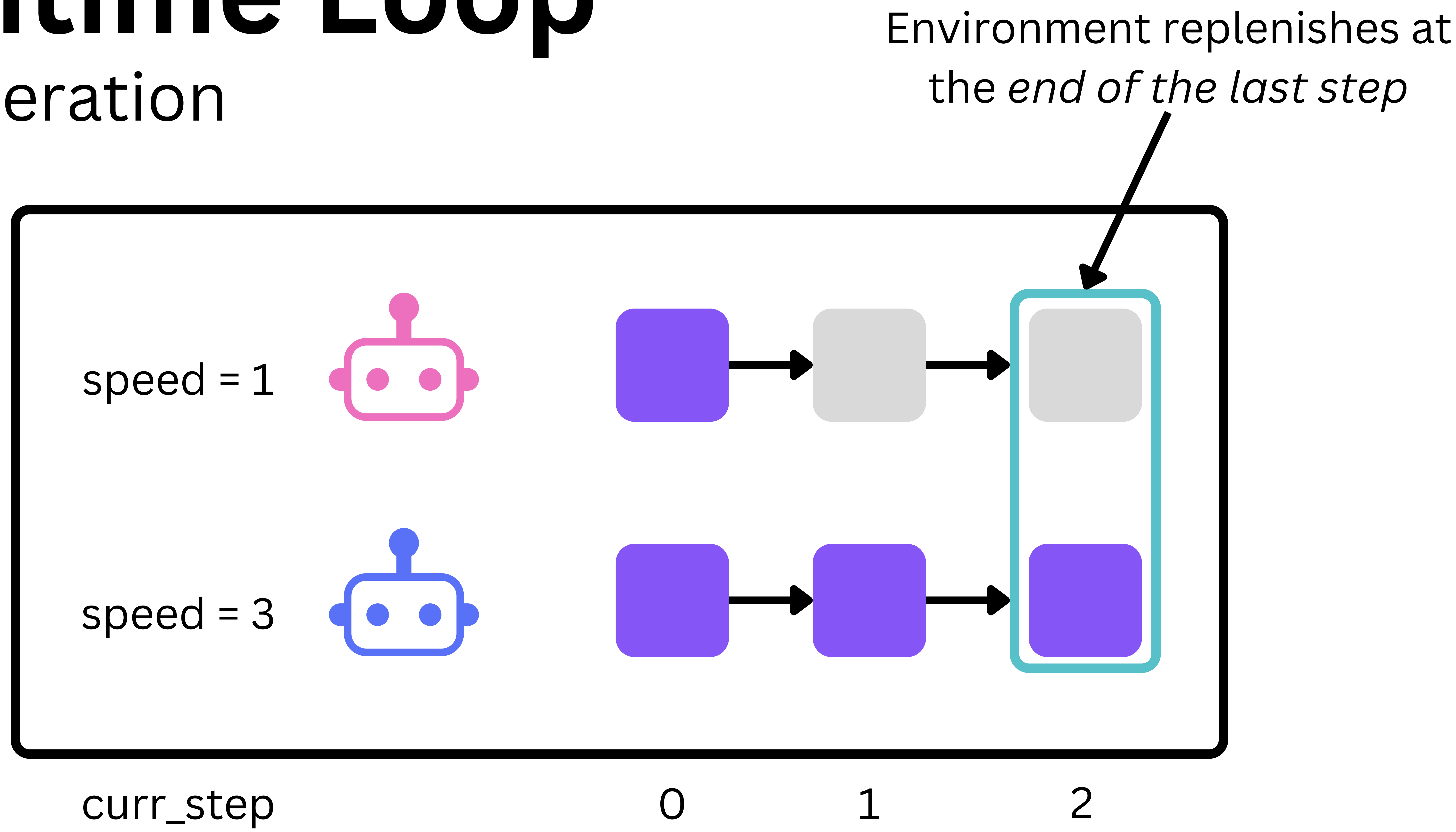
- Scores (rewards) tallied
- Speeds updated
- Players respawn (if allowed)
- Bombs and Boosts added (if any were taken during the iteration) to a max of 2 boosts and 1 bomb

Replenishment happens at the end of an iteration.

See **Environment Rules** for information on how score and speed are calculated, when players may respawn after dying, and how many bombs/boosts there are.

Runtime Loop

One Iteration



In the above example, player 0 (pink) has speed = 1, and player 1 (blue) has speed = 3. The iteration will, therefore, last 3 steps.

Note the greyed out squares. Since player 0 has speed = 1, they may only move during step 0. During steps 1 and 2, they will still be provided the updated observations (as this may be helpful for predictive algorithms), but their actions will be ignored by the environment.

Environment Rules

Energy and Speed

Energy	0	2	4	11	20
Speed	1	2	3	4	5

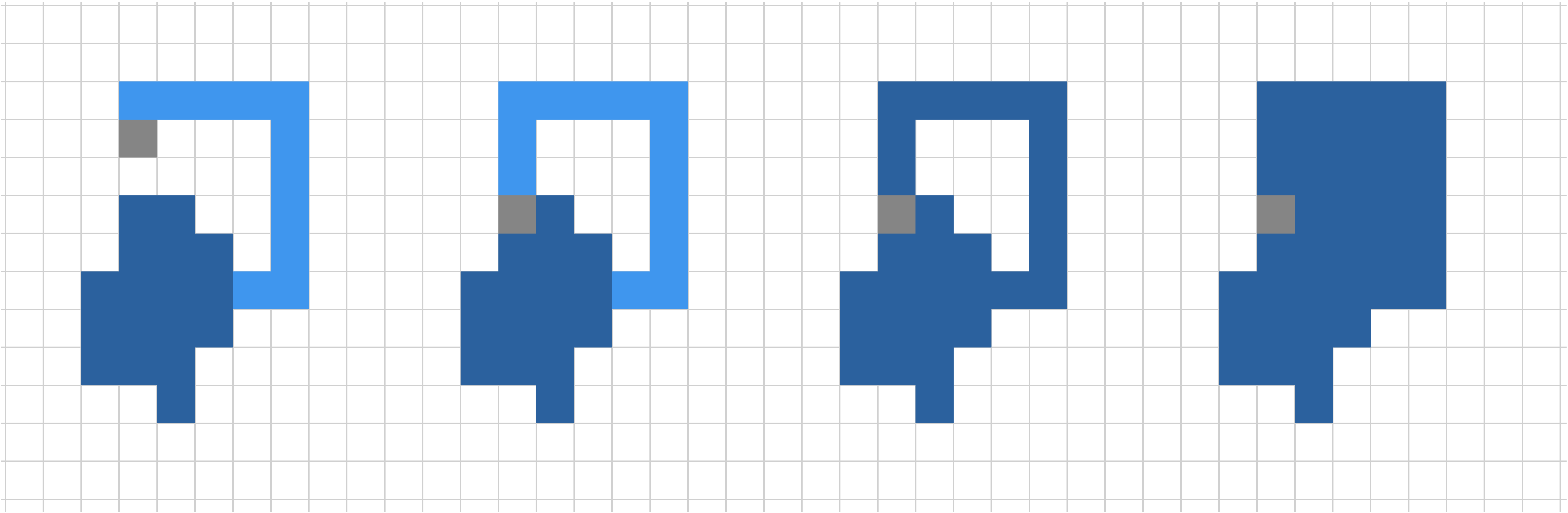
Collecting a boost increases the energy by 1.

After reaching a certain amount of energy, player speed will increase. The maximum speed is 5.

Increased speed allows the player to move multiple steps in one iteration (see **Runtime Loop**).

Environment Rules

Enclosing Territory - 1 of 3



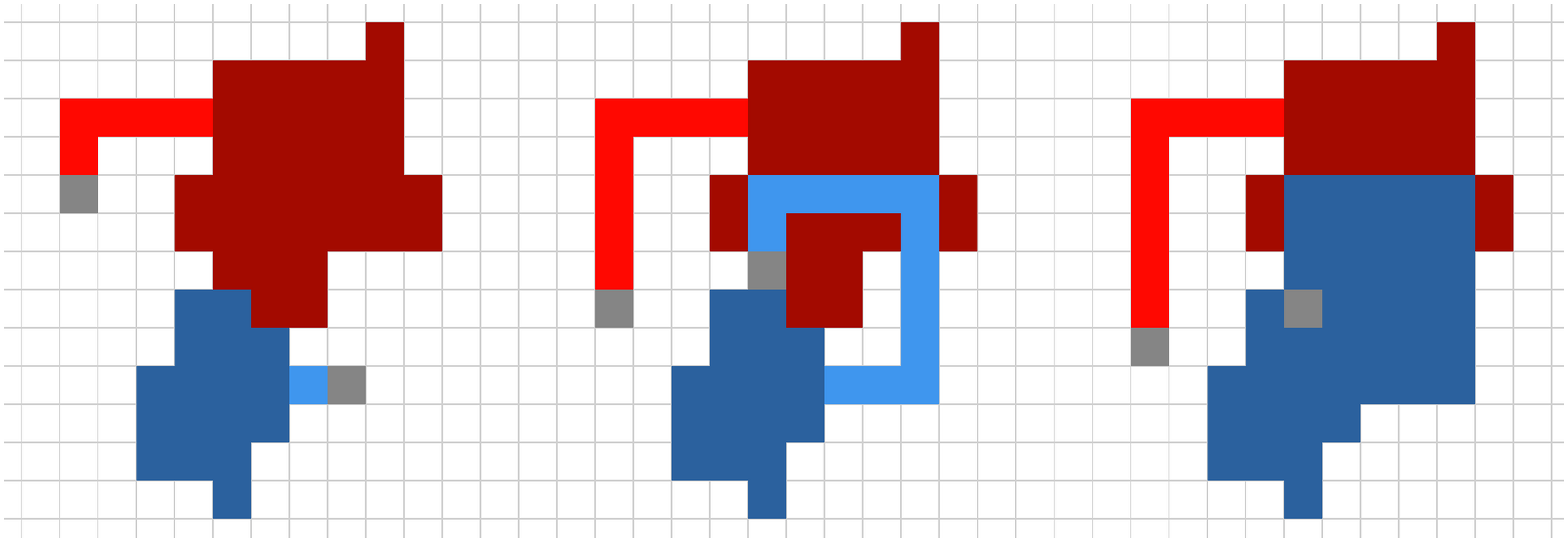
You may gain territory by enclosing space. The definition of enclosing space is as follows:

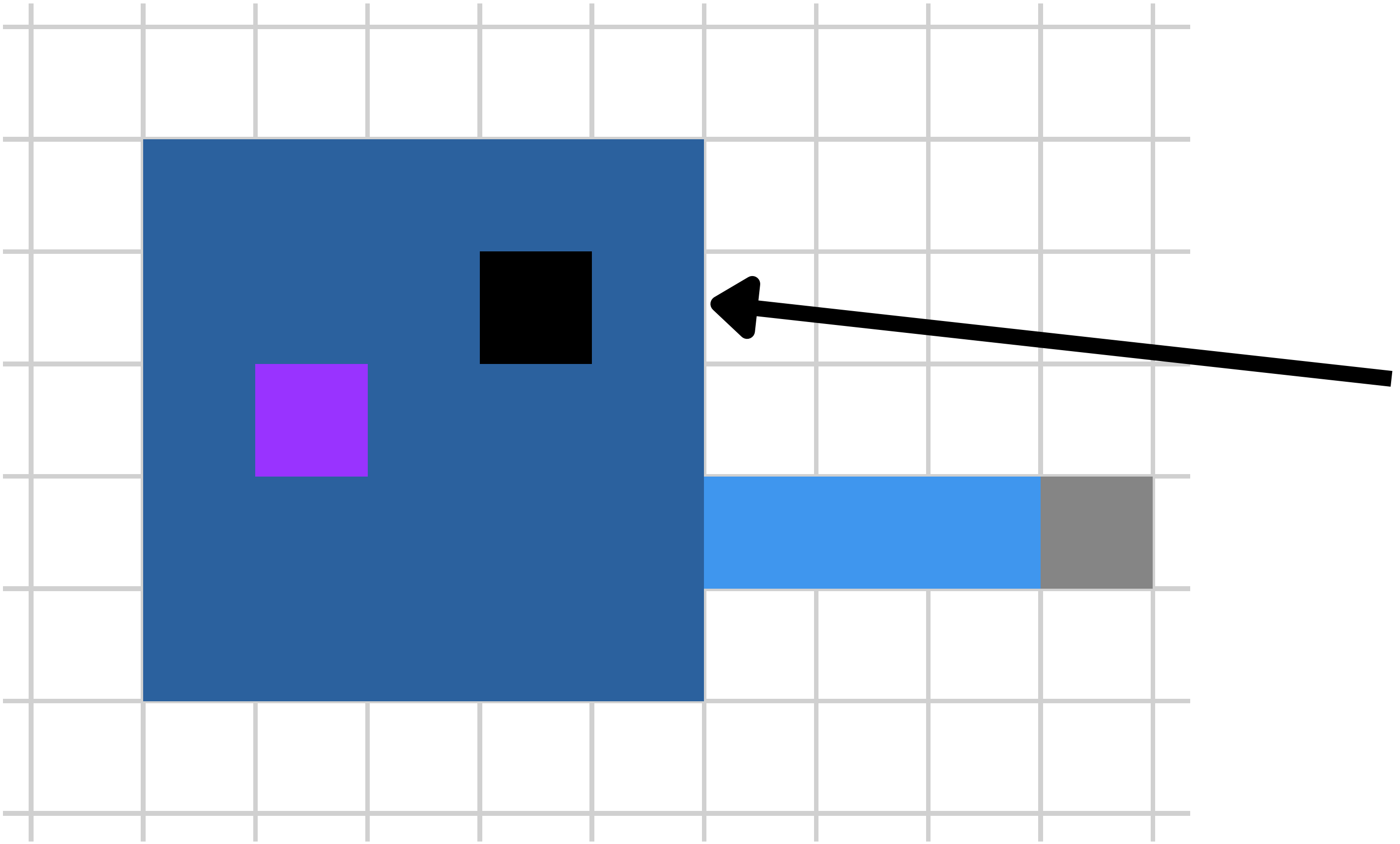
- Bot lands on a square marked as its own territory
- Its tail will be converted to territory
- Any tiles fully enclosed by its tail will also be converted to territory

The number of tiles in your territory will be added to your total score at each replenishment (see **Runtime Loop > Replenishing** and **Runtime Loop > One** Iteration for information on replenishment).

Environment Rules

Enclosing Territory - 2 of 3





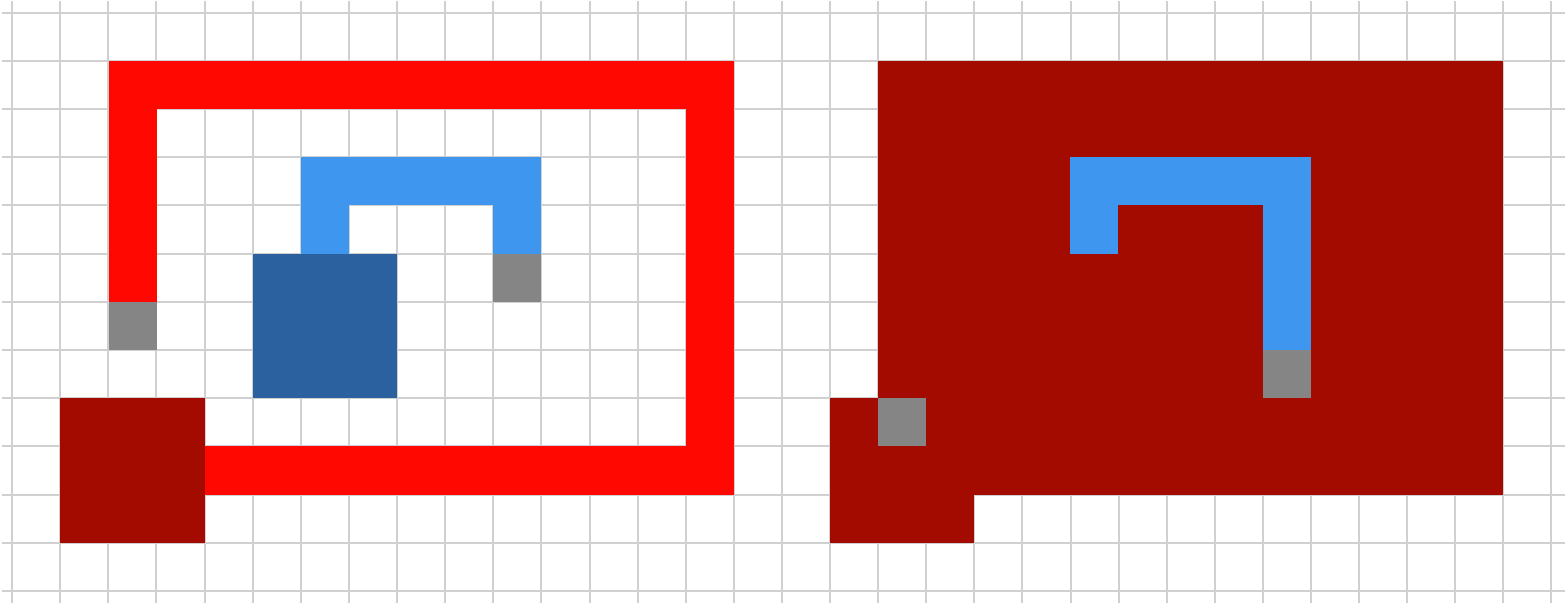
# tiles in your territory = 25

When your bot enters an opponent's territory, you will remove that territory from the opponent. Furthermore, if you enclose an opponent's territory, it becomes yours.

Furthermore, though not colored differently, bombs and boosts in your bot's territory will be counted as 'your' territory.

Environment Rules

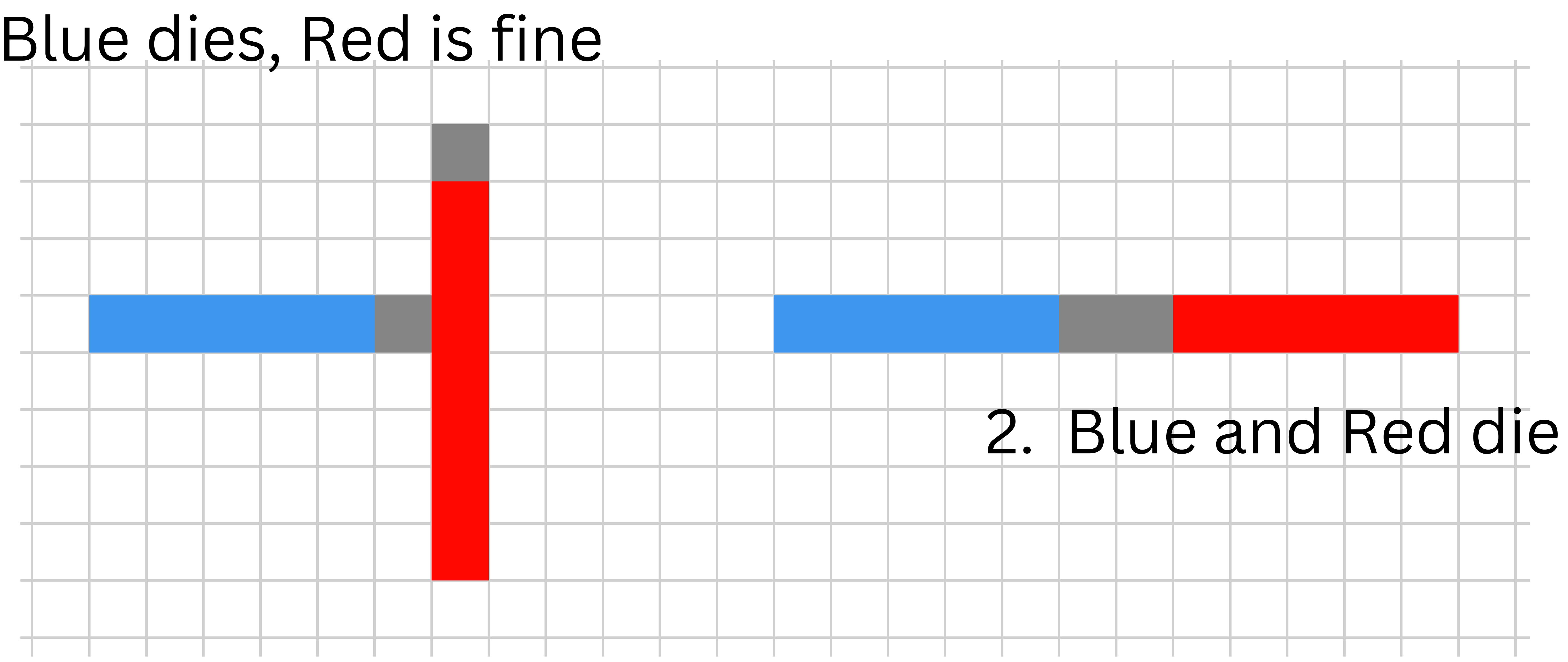
Enclosing Territory - 3 of 3



Note you cannot kill another player by enclosing them, as this is not one of the three means of dying mentioned in **Quick Start > Basic Rules - 1 of 2**. However, as explained above, their territory will become yours when you enclose them.

Environment Rules

Collisions



To kill another player, you must make them crash into your bot's tail.

In the event of a head-on collision, both players will die.

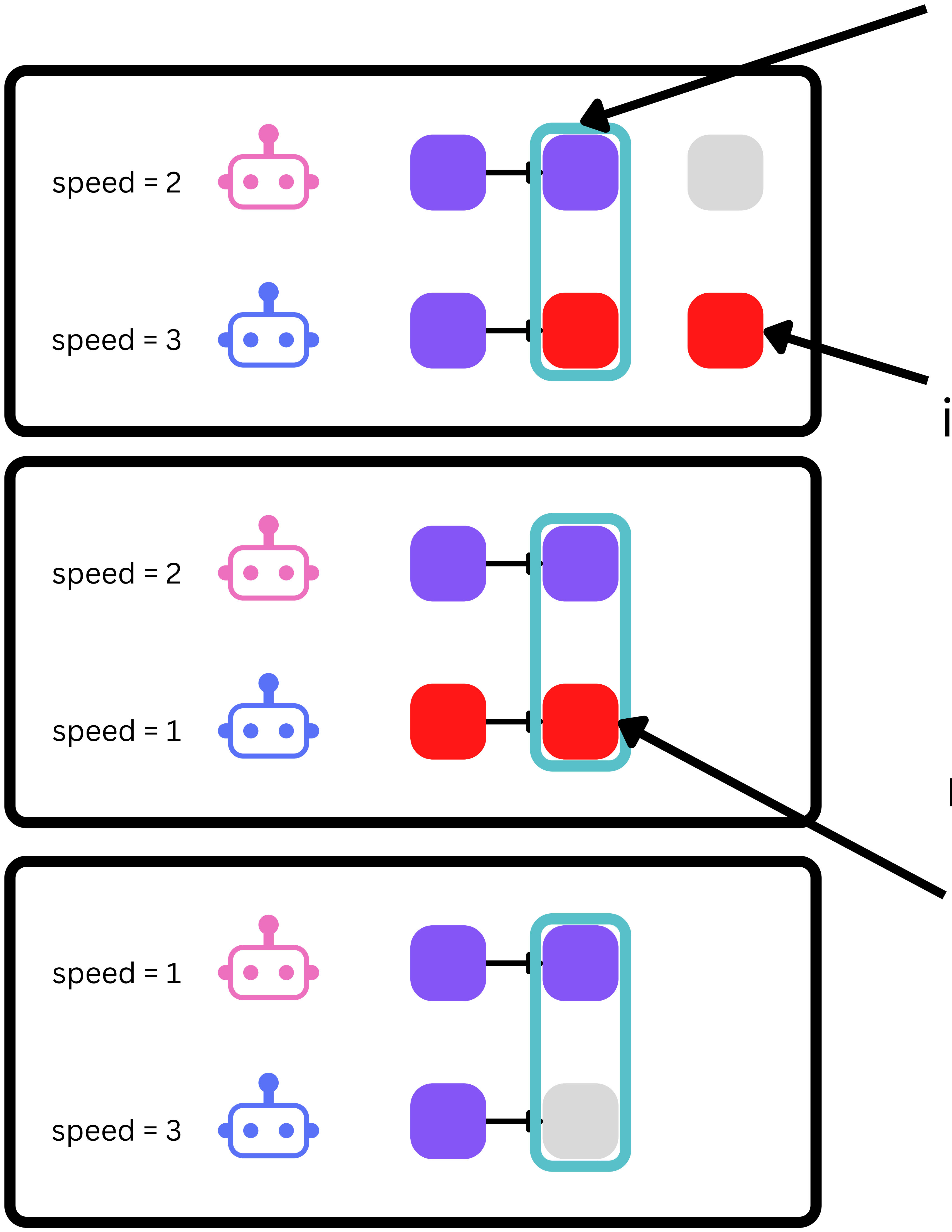
Environment Rules

Respawning - 1 of 2

iteration *n*

iteration *n+1*

iteration *n+2*



dies in step 1 of iteration *n*

pink cannot move and blue is dead, so step 2 is ignored

respawns at end of iteration *n+1* during replenishing

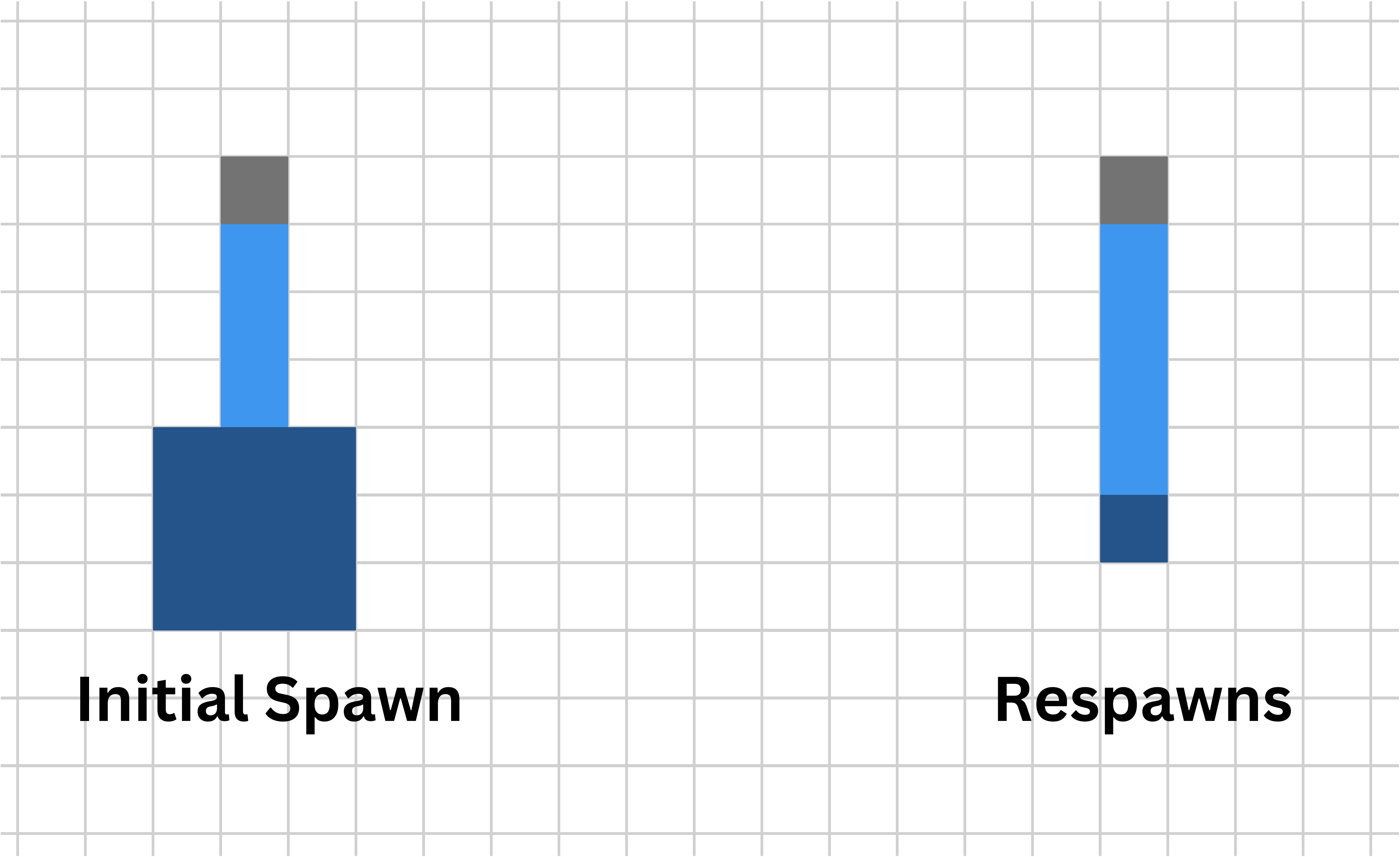
Respawning works as follows:

1. Player dies during iteration *n*. Speed and energy are reset.
2. Player must wait all of iteration *n+1*.
3. The player will respawn, and may progress normally during iteration *n+2*.

Environment Rules

Respawning - 2 of 2

Respawning occurs during replenishing (see **Runtime Loop**)



When initially spawning into the environment, the player will be given 3x3 starting territory.

When respawning, the player will only be given a 1x1 starting territory.