Multi-Agent Reinforcement Learning

Different types of multi-agent environments:

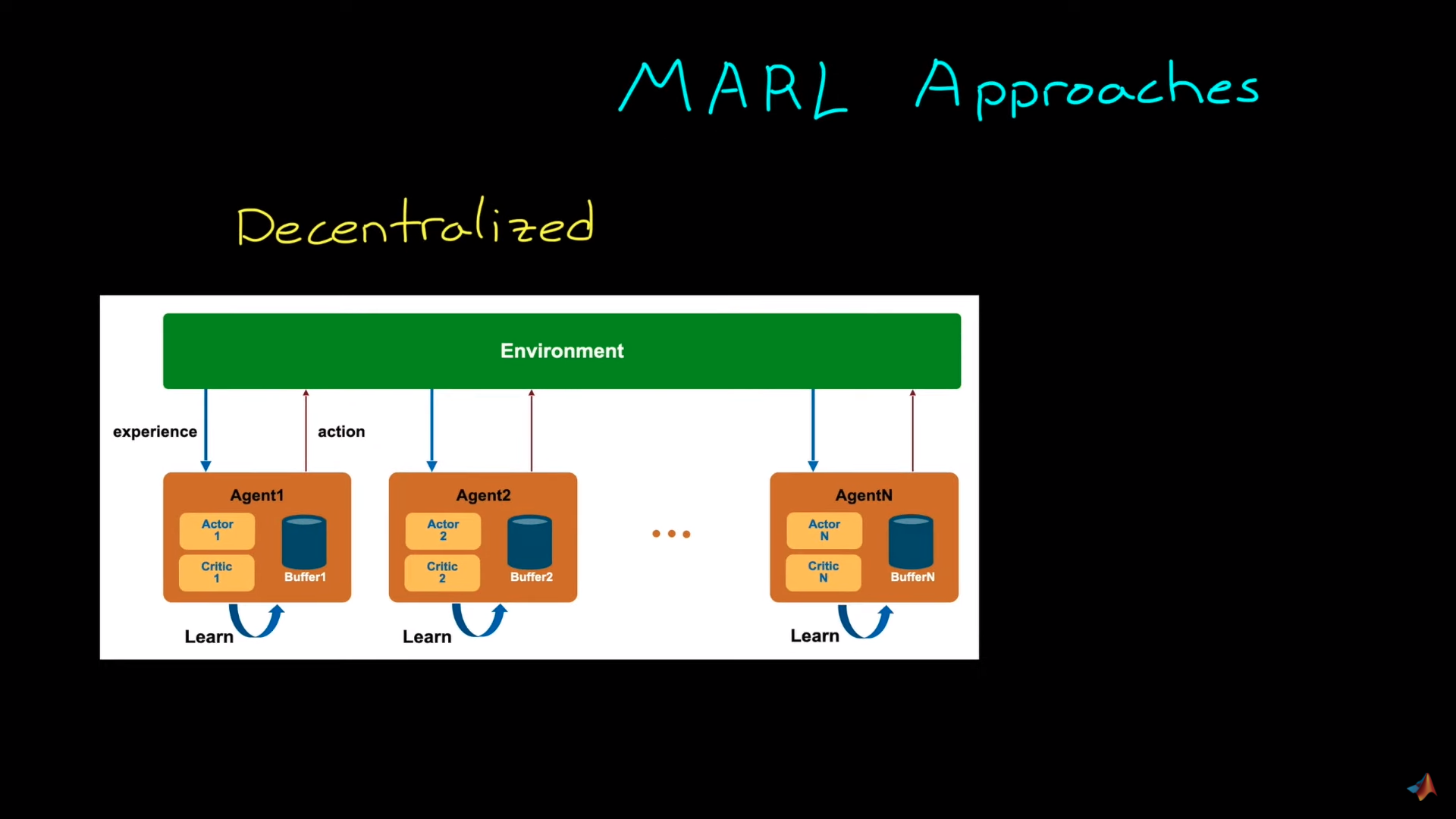
1. **Cooperative environments**: where your agents need to maximize the common benefits.
2. **Competitive/Adversarial environments**: in this case, your agent wants to maximize its benefits by minimizing the opponent’s.
3. **Mixed of both adversarial and cooperative**: like in our ‘SoccerTwos’ environment, two agents are part of a team (blue or purple): they need to cooperate with each other and beat the opponent team.

Designing Multi-Agents systems

We have two solutions to design this multi-agent reinforcement learning system (MARL):

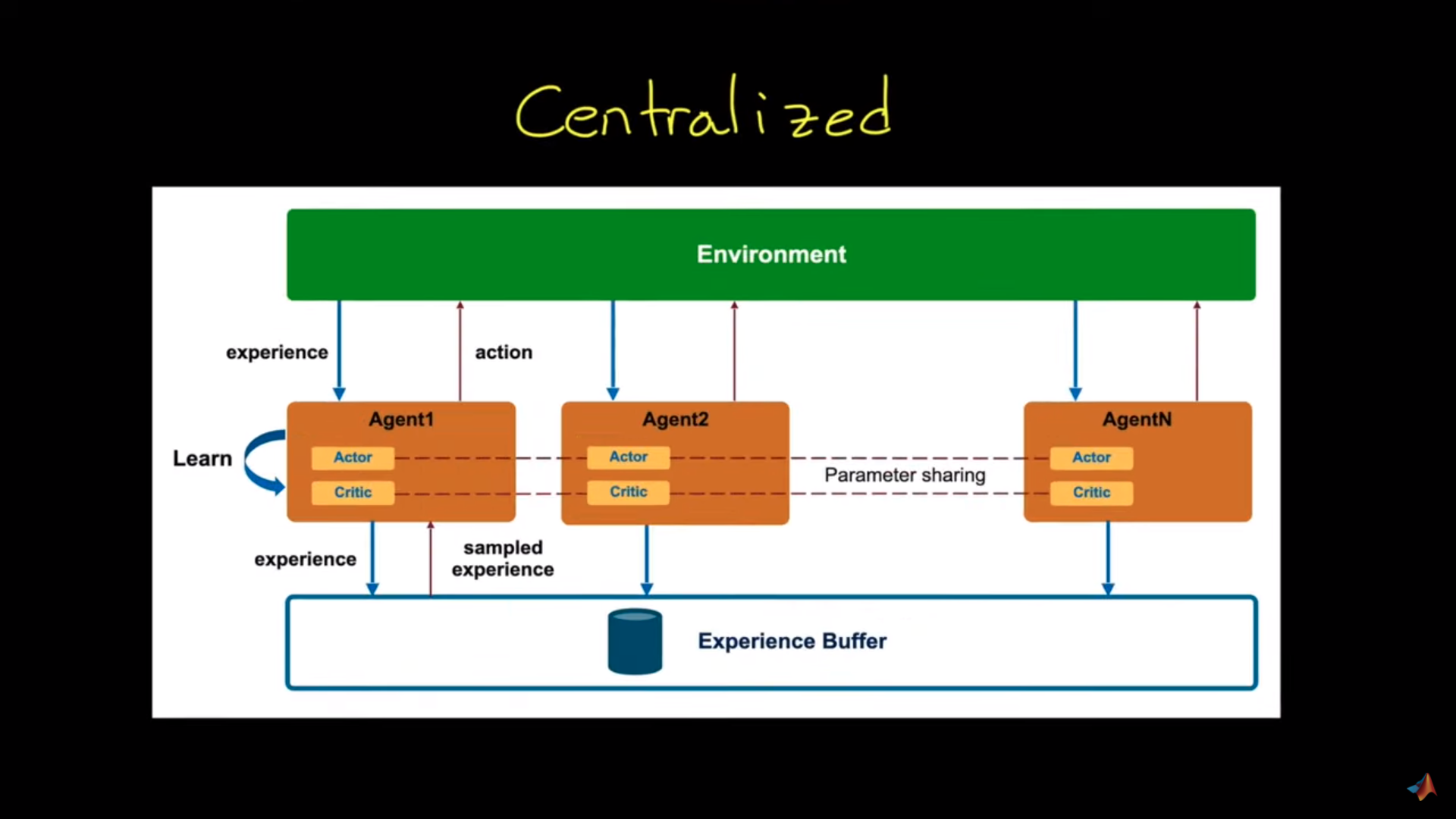
1. **Decentralized system:**

* Each agent is trained independently from the others.
* Since no information is shared between agents, these agents can be designed and trained like we train single agents.
* Our training agent will consider other agents as part of the environment dynamics.
* The big drawback of this technique is that it will make the environment non-stationary and this is problematic for many Reinforcement Learning algorithms that can’t reach a global optimum with a non-stationary environment.



1. **Centralized system**:

* We have a high-level process that collects agents’ experiences: the experience buffer. And we’ll use these experiences to learn a common policy.
* Takes as input the present state of an environment and the policy outputs joint actions.
* The reward is global.



**Self-Play**

* We start with a copy of our agent as an opponent, this way the opponent is on a similar level.
* We learn from it and, when we acquire some skills, we update our opponent with a more recent copy of our training policy.

**Elo Score**

ELO rating system (named after Arpad Elo) calculates the relative skill level between 2 players from a given population in a zero-sum game.

In a zero-sum game: one agent wins, and the other agent loses. It’s a mathematical representation of a situation in which each participant’s gain or loss of utility is exactly balanced by the gain or loss of the utility of the other participants. We talk about zero-sum games because the sum of utility is equal to zero.

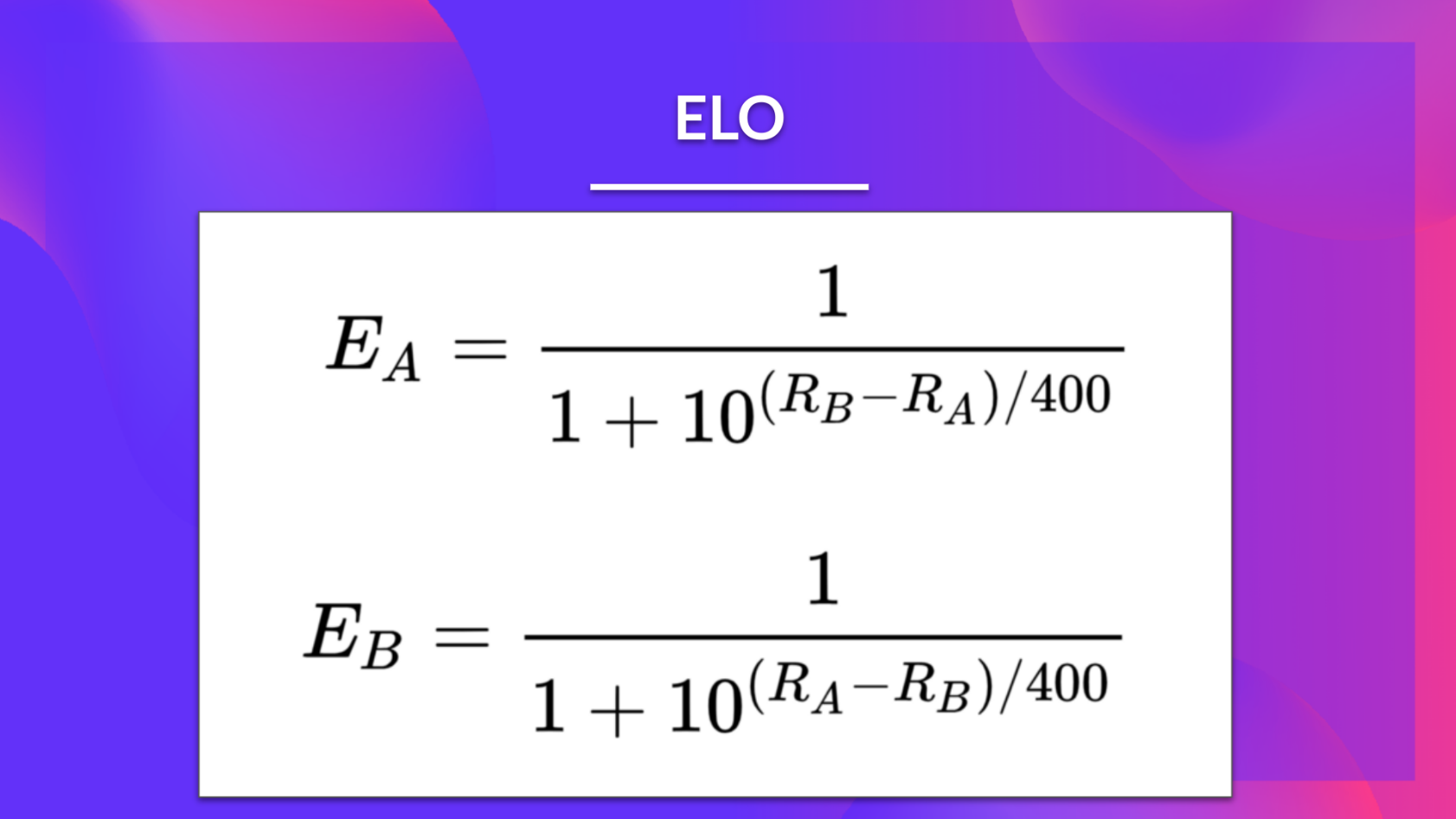
The Elo system is inferred from the losses and draws against other players. It means that player ratings depend on the ratings of their opponents and the results scored against them.

The difference in rating between 2 players serves as the predictor of the outcomes of a match. If the player wins, but the probability of winning is high, it will only win a few points from its opponent since it means that it is much stronger than it.

After every game:

* The winning player takes points from the losing one.
* The number of points is determined by the difference in the 2 players ratings (hence relative).
* If the higher-rated player wins → few points will be taken from the lower-rated player.
* If the lower-rated player wins → a lot of points will be taken from the high-rated player.
* If it’s a draw → the lower-rated player gains a few points from the higher.

If A and B have rating Ra and Rb, then the expected scores are given by:



Then, at the end of the game, we need to update the player’s actual Elo score. We use a linear adjustment proportional to the amount by which the player over-performed or under-performed.

We also define a maximum adjustment rating per game: K-factor.

* K=16 for master.
* K=32 for weaker players.

If Player A has Ea points but scored Sa points, then the player’s rating is updated using the formula:

