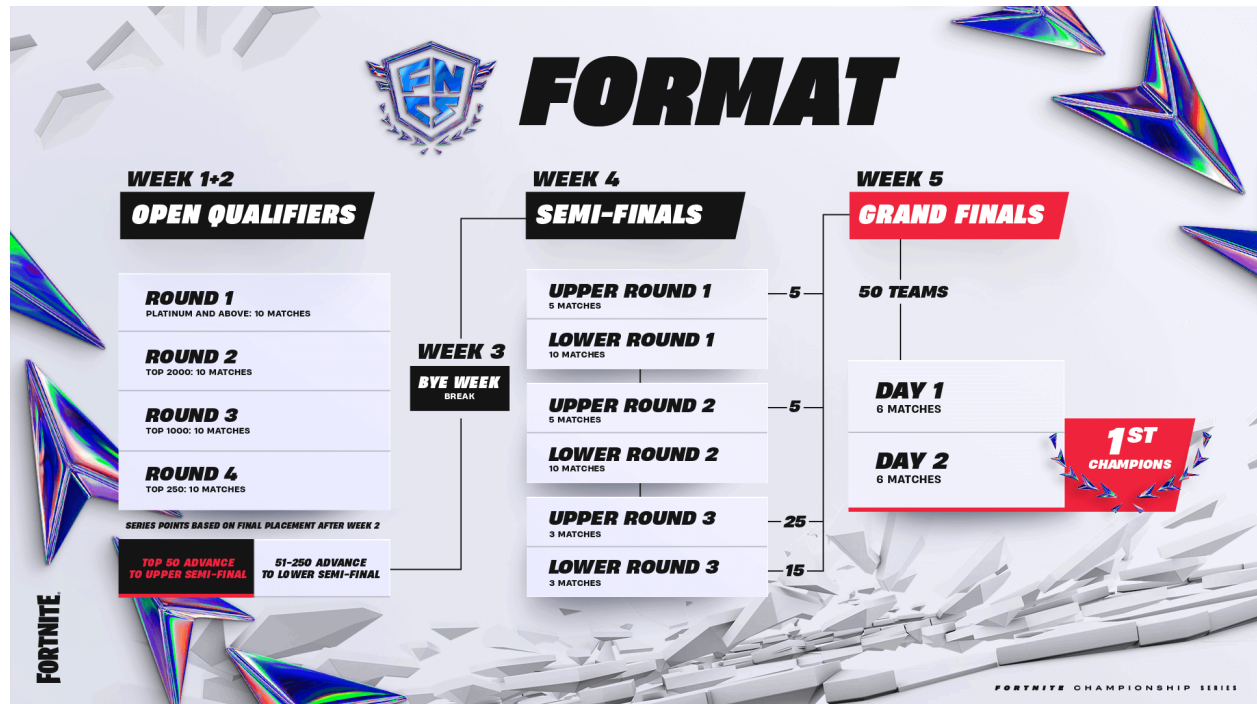


I've recently been watching professional esports for the game Fortnite but considering the scale of the games played, up to 100 players, it can be difficult to track any specific teams performance across matches and the tournament as a whole.



Once a team qualifies for the championship, they play a group of matches called rounds where they earn points based on their amount of eliminations and placement (how many teams they outlasted). The team's points are then totaled up across all of the matches in the round and the teams with the highest amount of points move on to the next round.

Rules:

- Every team's performance in a match must be tracked
- Each match must belong to a round, every round to a tournament

Nouns:

- Round
- Match
- Team
- Tournament
- Performance
- Player

Tables:

- Tournament
- Round (A series of matches)
- Match
- MatchPerformance (A team's performance during a match)
- Team (2 players)
- Player

Redis:

I will make the matches and match performance tables into redis structures

Implementation:

I will implement the matches table with a redis list that contains the matchID's. I will implement the matchPerformance table with hashes, where the key to the hash is the matchID and the fields in the hash are the team's id and the value is the team's placement and eliminations.

Commands:

To initialize/update the matches list I would use:

*LSET matches match:matchID*

To initialize/update the matches list I would use:

*HSET match:matchID*

*matchPerformance:teamID*

*matchPerformance:placement.toString() + " " +*

*matchPerformance:eliminations.toString()*

To get a match performance:

*HGET match:matchID matchPerformance:teamID*

To get all of the matchIDs:

*LRANGE matches 0 -1*