

To: Mattias
From: codemanistan/dare-rebr
Subject: Observer

An Observer is an independent component that should interface with the tournament manager and receive updates for game state for specified games. The Observer should be able to request that it observes a specified game, and it should receive the updated board state for that Game (being handled by a Referee) as moves and build within that game occur. The Observer should be able to receive the list of currently-running Games, and it should be able to request continuous updates for a given Game, at which point it will receive updates from the tournament manager until that game is over.

Observer should have the following interface methods:

- `GetObservableGames(target TournamentManager) []int`
 - o Returns a list of gameID's representing the games available for observation.
- `ObserveGame(gameID int)`
 - o Observe a game by ID (int) between two players
- `receiveBoard(board Board)`
 - o Receive an update to the state of the game being observed
- `gameOver(winner string, board Board)`
 - o when the game is won, the tournament manager/administrative component sends a string with the name of the winner and the final board state which is received by the observer.

Where `receiveBoard` and `gameOver` is a push update called from the tournament manager when there is a new Board state.

This will require the tournament manager to receive updates from the referee, if that referee's game is being observed. This requires a slight tweak in the