

## Santorini Tournament

### Server/Tournament Manager

### Client

Set-Up

1) Listen for Connection

Listen

Attempt Connection

3) Accept/Refuse the Connection

Accept Connection

4) Read Configuration Files and  
create all Players

2) Attempt to Connect to Server

Run Games

5) Generate each Round of Games  
Match and Assign Referees to Players

6) Have each Referee run their Game

Prompt Client for Actions in Prompt Format

Send Actions in Action, JSON Format

8) Save Game Results and Communicate  
Result to Players and Observers

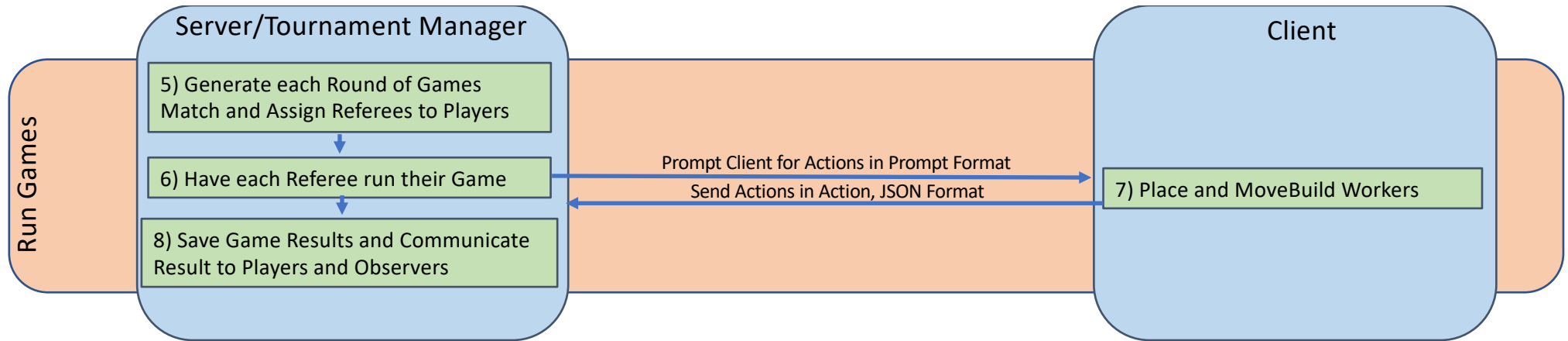
7) Place and MoveBuild Workers

Shut Down

9) Compute Winner and Send  
Results to all Clients

Send Game Results Back in Tournament Result Format

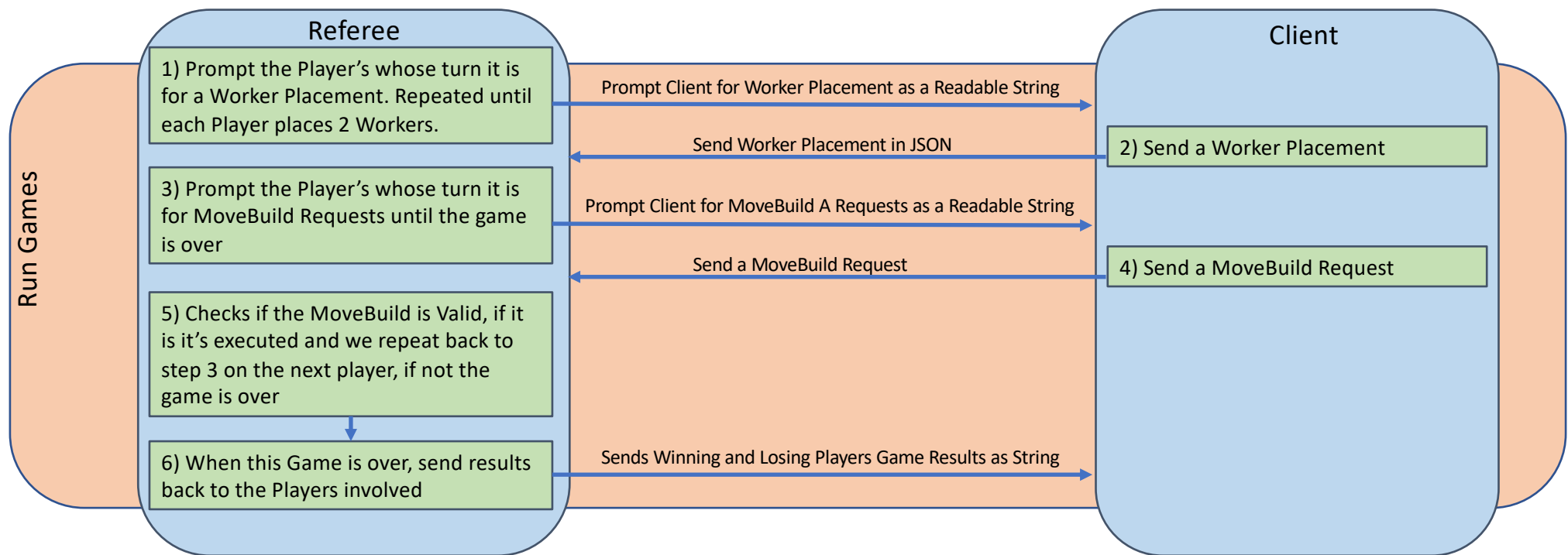
10) Closes all Connections



### Run Games Stage

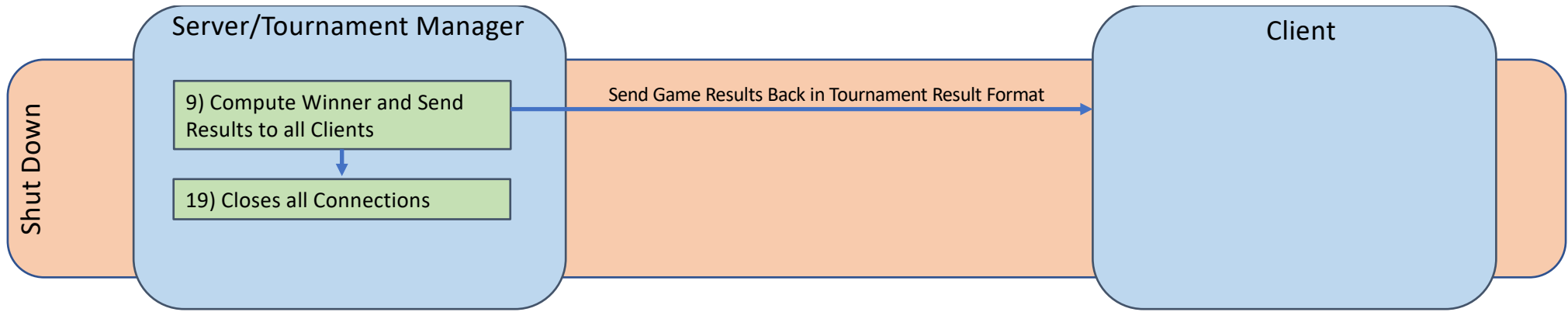
In this stage, the Tournament Manager facilitates the pairing of Players for Santorini Games and Creation of Referees, delegating game play to each referee. Within each round, the Tournament Manager matches each Player with a Player they haven't played yet. Each Referee is stored within the Tournament Manager to store game result information. All interaction between clients and games of Santorini are handled within each Referee. This stage is repeated until all Players have played against every other Player in the Tournament.

See Single Game diagram and description for communication formatting.



### Run A Single Game Communications

- 1) Prompts as Readable String: Will only be used for Client(human) Players so they know what kind of request is needed at this stage in the game, AI Players will ignore
- 2) Worker Placement Request JSON: [*"place", x, y*],  
→ x, y are integer coordinates to place workers
- 3) MoveBuild Request JSON, request to move and build from a Worker: [*"move", Worker, Direction*] [*"+build", Direction*]  
→ Worker is the String name of the Worker the Client wants to move  
→ Direction is [EastWest, NorthSouth] where EastWest is one of: "EAST" "PUT" "WEST" and NorthSouth is one of: "NORTH" "PUT" "SOUTH"
- 4) Game Results: "Winner: *Winning Player*, Loser: *Losing Player*"  
→ Winning Player and Losing Player are the String name of each Player



### Shut Down Stage

When the Tournament has finished and all Players have faced every other Player, the Tournament will compute the winner(s) and send the results to the client which include all disqualified players and results of all games played.

#### Results:

- 1) JSON Array of JSON Strings of disqualified players names, ordered by time disqualified.
  - ex. If "p1" was kicked out round 1 and "p2" was kicked out round 2, output would be: ["p1", "p2"]
- 2) JSON Array of Game Results
  - A Game Result is a JSON array of [*winning player name*, *losing player name*]
  - ex. [["p1", "p2"], ["p3", "p4"]] means p1 won against p2 and p3 won against p4