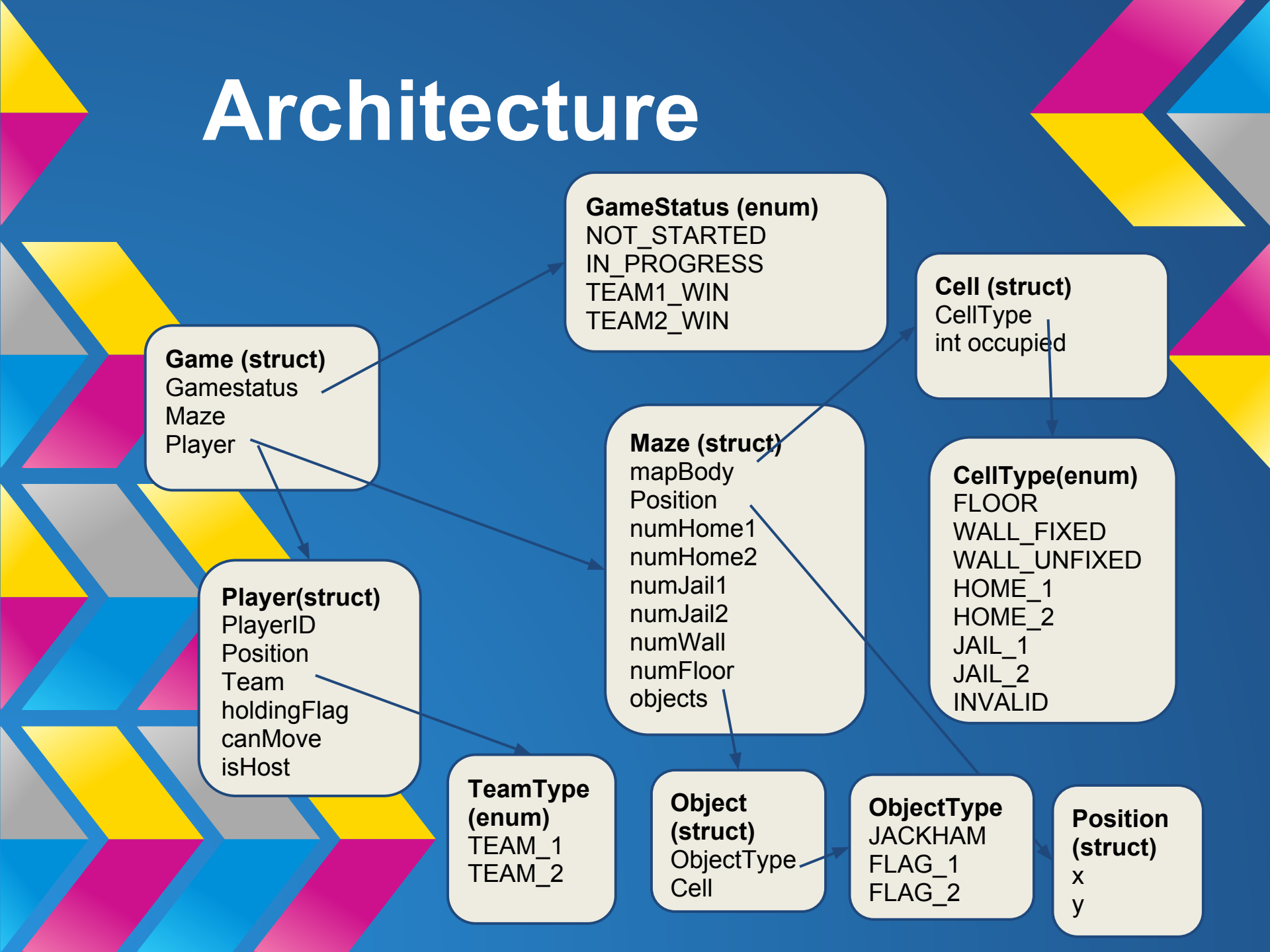
Da Game

Documentation

Architecture



Architecture

(To distinguish different tiles in the map)
CellType (enum) - {
Floor1,
Floor2
NonFixedWall,
FixedWall,
Home_1,
Home_2,
Jail_1,
Jail_2,
Invalid,
}

Architecture

Team (enum) - {
Team_1,
Team_2
}

(to determine winner)
GameStatus (enum) - {
NOT_STARTED,
IN_PROGRESS,
TEAM1_WIN,
TEAM2_WIN
}

Architecture

```
Position{
      int x;
       int y;
}

Cell{
     CellType type, (ex. wall, jail);
     int occupied;
     int fog;//0=no one, 1=team1, 2=team2, 3= both teams
}
```

Architecture

(The general map and the map's info)

```
Maze {
    Cell **mapBody;
    Position dimension;
    int numHome1;
    int numHome2;
    int numJail1;
    int numJail2;
    int numWall;
    int numFloor;
}
```

Architecture

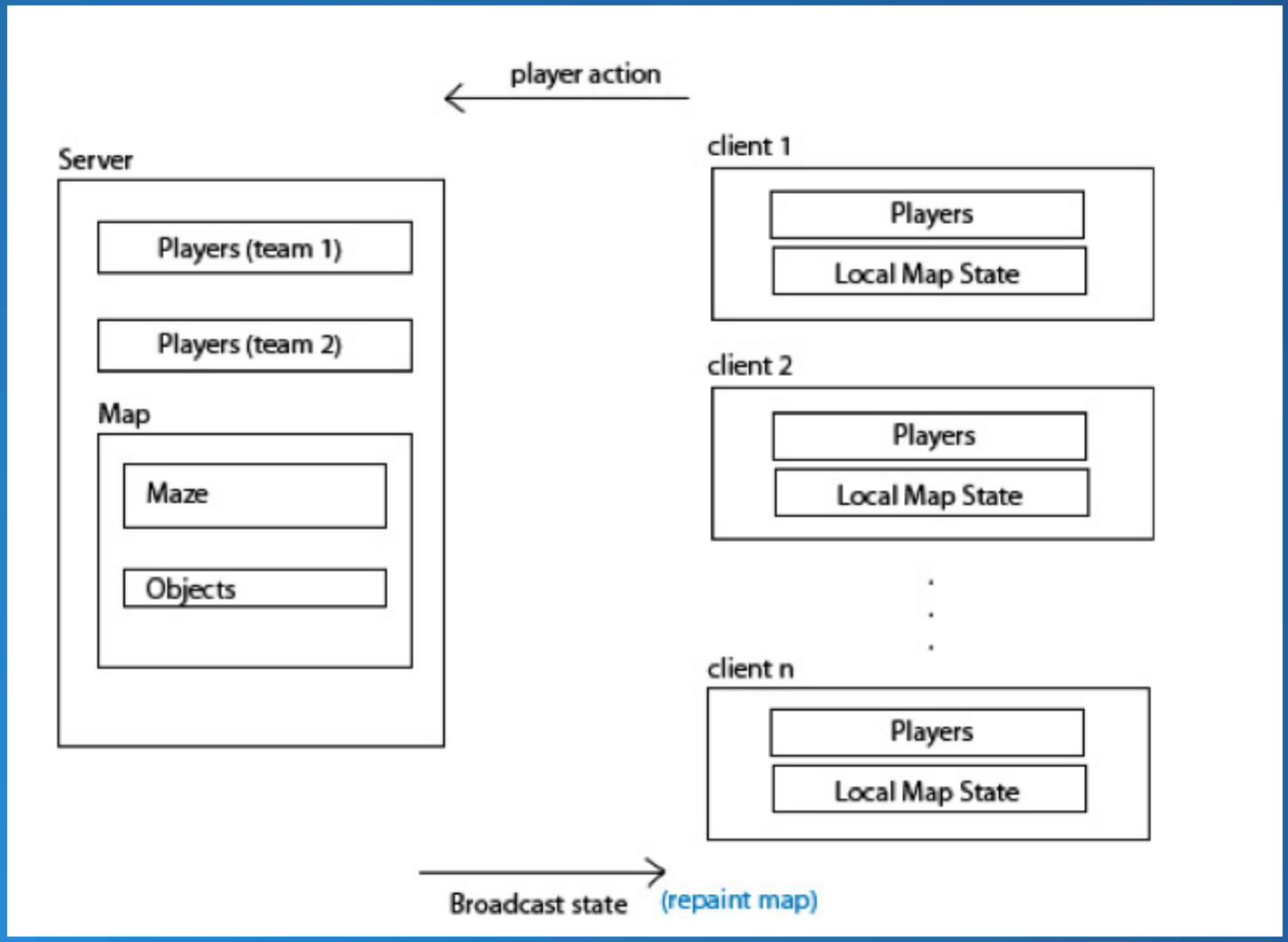
```
Player {
    Cell cellPosition;
    TeamType team;
    Item itemHeld; (can only hold one item)
    int canMove;
    int playerID;
    int isHost;
}

Item (enum) {
    NONE,
    JACK_HAMMER1,
    JACK_HAMMER2,
    FLAG_1,
    FLAG_2,
}
```

Architecture

```
Game {
    Player Team1_Players[MAX_NUM_PLAYERS];
    Player Team2_Players[MAX_NUM_PLAYERS];
    Maze map;
    GameStatus state;
}
```

Architecture



Synchronization

Mutex Lock

Game Struct

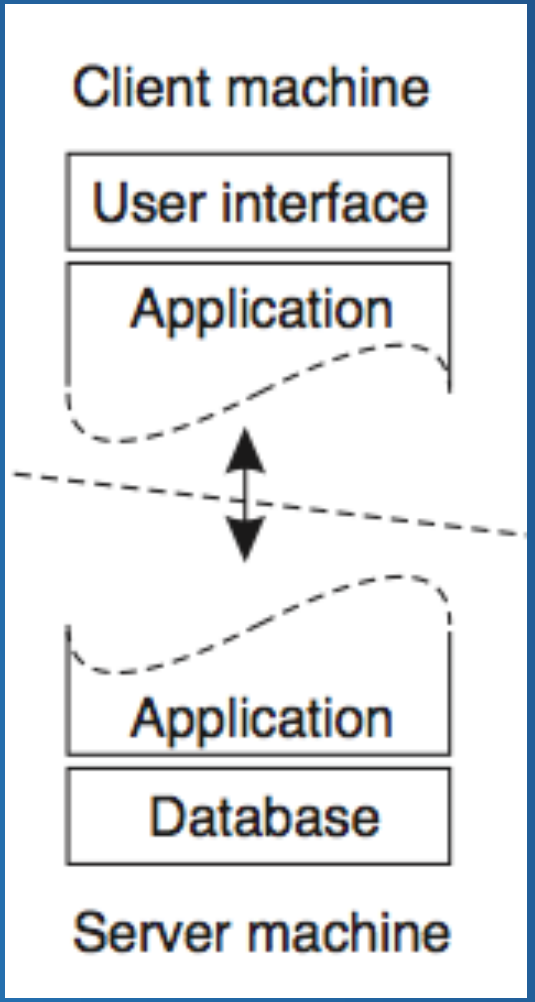
Modified Game State

Mutex Unlock

Game Semantics

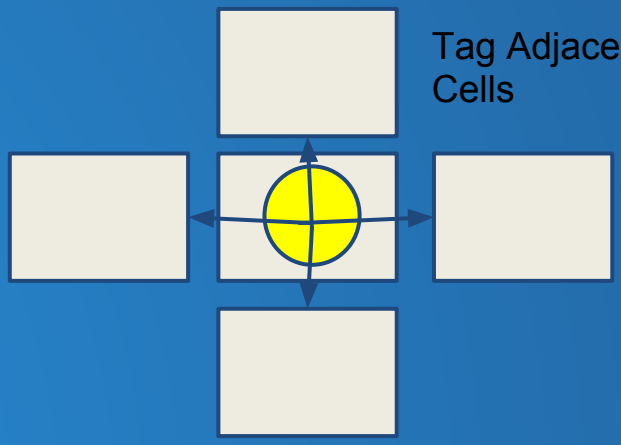
Player Moving

- Moving is partially split between server and client.
- Client will do a basic check to see if it is trying to move into an indestructible wall. If valid it will send the RPC, and handle the rest of the move logic server side.
- Two players cannot occupy the same cell.
- Server will check the other details of the move - if other player is there, if inner wall is there..etc.



Tagging

- After a valid move action occurs, the server will then tagging logic.
- A player tags an enemy if the player's new location is on its own side, and the enemy is located in any of the four blocks adjacent to the player.
- This checking is done after a move action, so a separate tag action is not needed.
- If a tag takes place, the player that is tagged is sent to a position in the opposing team's jail.



Jail

The new position of a player going to jail is determined by randomly selecting an unoccupied jail cell.

- If a player moves into the enemy's jail, and he is not tagged, then all of the player's teammates that are currently in jail are teleported back to their home base (positions selected at random)

Picking up/dropping

- A player can only hold one object at a time (flag1, flag2, jackhammer1, jackhammer2)
- An object is picked up when a player is on the current position of that object and the pickup key is pressed.
- After an object is picked up, the player will now "hold" this object. (ItemHeld will be set to that item's enum)
- Since a player can only hold one object at a time, when that pickup key is pressed again, it will drop the object it is currently holding. (ItemHeld = none)
- The server will check with the player when a pickup/drop key is pressed. If the player is holding an item it will drop it, if nothing is held it will attempt a pickup action.

Flags

- Each flag is generated randomly on its respective side at initialization.
- The flag cannot be generated in a home, or jail cell.
- If a player picks up its own flag on the enemies side, it will continue to hold it until the player reaches their own side.
- If a player moves onto their own side while holding their flag, the flag is then again randomly placed on that side.
- A player is NOT allowed to pick up its own flag if the flag is on that player's side.
- If a player is holding a flag, and is tagged the player gets sent to jail and the flag will be dropped in the location of the tagged player.

Jackhammer

- When a player sends a move action, if that player is holding a jackhammer the game logic for that move will change.
- If the player sends a move action that would place them on a destructible wall cell, then this wall will be destroyed.
- The players location will also replace where that wall used to be.
- After a destruction occurs, the map is updated and the player no longer hold a jackhammer.
- After the jackhammer destroys a wall, the jackhammer will be now reset back at the jackhammer spawn position.

Winning/Losing

- On every move action, after all other updates have been made (new position, changes in map..etc). Check to see if this new version of the GameState results in a win/lose
- Winning occurs if all players on the same team are on their respective side, with both flags on their side as well.
- Both of the flags need to be dropped on that side, therefore if an enemy player is holding a flag on the side the game does not end yet.

Disconnect

- If a user disconnects (either through a dropped connection or a quit command) the game continues.
- If the disconnected user was holding an object, that object will be dropped where the user was standing.
- If all users in the game have disconnected then the game ends.