

# Multi Robot Task Distribution

## Shared Memory Module – Architecture & Design

### Functionality

- Acts as a centralized data repository for storing shared information such as task assignments, robot statuses, and other relevant data.
- Provides services for querying and updating shared data.
- Ensures synchronization and consistency of shared data accessed by multiple nodes.

### Components

1. Data Storage: In-memory data structures to store real-time information. A log to maintain historical data in the form of csv file/log file.
2. Service Interface: Defines ROS services for querying and updating shared data. These services handle requests from other nodes to read or modify shared information.
3. Synchronization Mechanisms: ROS services allow nodes to communicate by sending requests and receiving responses in a synchronous manner.
4. Communication Mechanisms:
  - a. ROS Services: Provides ROS services for interactions with other nodes. These services include functionalities such as querying the current list of available robots, updating task assignments, or updating robot status.
  - b. ROS Topics: Optionally, ROS topics can be used for broadcasting updates on shared data. For example, a topic could be used to publish real-time updates on robot positions.

### Data Structures

- Task Assignments: Stores information about tasks assigned to each robot, including task IDs, descriptions, priorities, and status (e.g., pending, in progress, completed).
- Robot Statuses: Maintains the current status of each robot, including position, target, battery level, and operational status.

### Interaction with Other Nodes

- Load Balancer Node: The load balancer interacts with the shared memory module through ROS services to query the current status of robots and update task assignments based on workload and availability.
- Robot Nodes: Robot nodes communicate with the shared memory module to update their status (e.g., position, target) and retrieve task assignments.

### Possible Inclusions/modifications upon initial implementation

- Use of Database instead of In-Memory data structures
- Use of Grpc for communication instead of ROS Services
- Estimation of ETA of each robot and traffic considerations