## **GAS ClassDiagram**

## **Traverse**

- + Traverse(ros::NodeHandle)
- + status:bool
- + bill\_y:double
- + bill\_x:double
- + X:vector<double>
- + Y:vector<double>
- + Z:vector<double>
- + W:vector<double>
- + goals : vector<move base msgs::MoveBaseGoal>
- node:ros::NodeHandle
- + move\_next(): void
- + to goal(double,double):void
- + reach\_bill():void

## **Detect**

- + Detect(ros::NodeHandle)
- + laser\_dist: double
- + aligned: bool
- + get dist: bool
- + pos\_x : double
- + orienitation : double
- + rotate : int
- + out : int
- + spotted : bool
- + pos : vector<int>
- + camsub : ros::Subscriber
- + scansub : ros::Subscriber + odomsub: ros::Subscriber
- + motorpub : ros::Publisher
- + pos\_y : double
- n : ros::NodeHandle
- + process\_image\_callback(sensor\_msgs::Image): void
- + LaserCallback(sensor\_msgs::LaserScan::ConstPtr):void
- + odomCallback(nav\_msgs::Odometry::ConstPtr):void
- + drive\_robot(float,float):void
- + startdetect():void
- + spot\_image(cv::Mat)
- + robot\_motion(cv::Mat)

## Collect

- nh: ros::NodeHandle
- spawn\_model\_client : ros::ServiceClient
- delete\_model\_client : ros::ServiceClient
- pose : geometry\_msgs::Pose + checkr : bool
- + checks : bool
- + spawn(char\*, double, double, double,int):void
- + remove\_ob(char\*):void