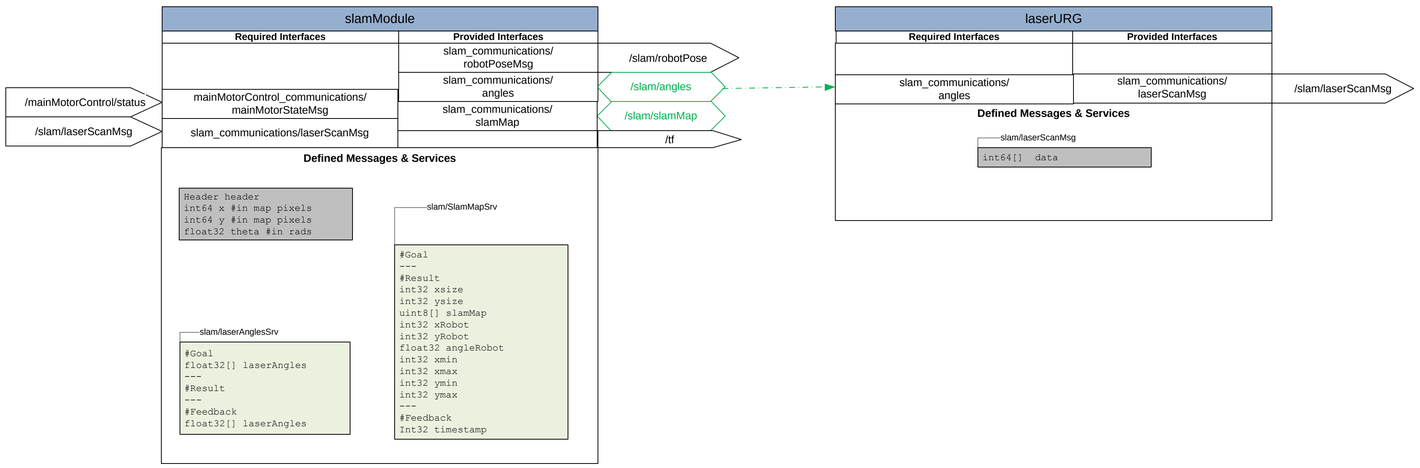
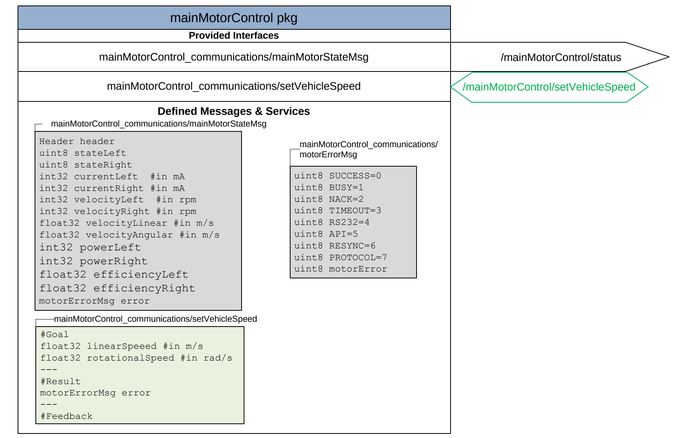
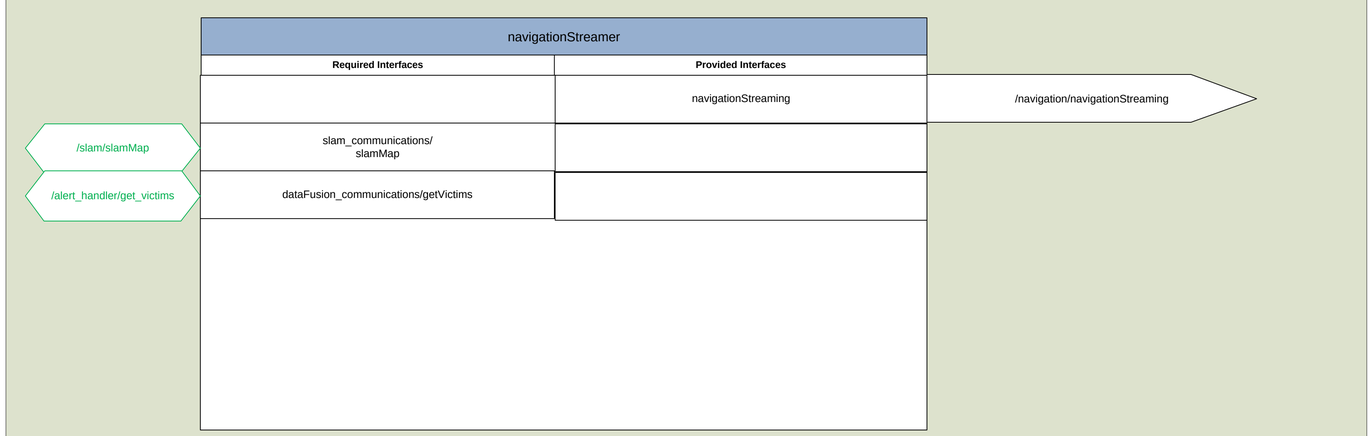
Slam package



Main Motor Control package



Map Streamer



Controllers and Sensors package

controllersAndSensors

Required Interfaces	Provided Interfaces	
	controllersAndSensors_communications/ gripperControlSrv	/sensors/gripperControl
	controllersAndSensors_communications/ controlSwitchSrv	/sensors/ledControl
	controllersAndSensors_communications/ controlServo	/sensors/moveHead
	controllersAndSensors_communications/ sonarMsg	/sensors/sonar
	controllersAndSensors_communications/ headSonarMsg	/sensors/headSonar
	controllersAndSensors_communications/ compassMsg	/sensors/compass
	controllersAndSensors_communications/ irMsg	/sensors/ir
	controllersAndSensors_communications/ headIrMsg	/sensors/headIr
	controllersAndSensors_communications/ tpaMsg	/sensors/tpa/*(0,1,2)
	controllersAndSensors_communications/ co2Msg	/sensors/co2
	controllersAndSensors_communications/ butterflyMsg	/sensors/butterfly
	controllersAndSensors_communications/ mlxMsg	/sensors/mlx

Defined Messages & Services

controllersAndSensors_communications/ gripperControlSrv

```
uint8 STATE_OFF=0
uint8 STATE_OPEN=1
uint8 STATE_CLOSE=2
uint8 state
---
uint8 ERROR_OK=0
uint8 ERROR_COMMUNICATING=1
uint8 error
```

controllersAndSensors_communications/ controlSwitchSrv

```
uint8 STATUS_OFF=2
uint8 STATUS_ON=1
uint8 status
---
uint8 OK=0
uint8 ERROR_COMMUNICATING=1
uint8 error
```

controllersAndSensors_communications/ controlServo

```
#Goal
uint8 position
uint8 positionSet=2
---
#Result
uint8 error
uint8 OK=0
uint8 ERROR_COMMUNICATING=1
---
#Feedback
```

controllersAndSensors_communications/ tpaMsg

```
Header header
int32 ambientTemp #Celcius
int32[8] pixelTemp #Celcius
uint8 id
uint8 CENTER=0
uint8 LEFT=0
uint8 RIGHT=0
uint8 LEFT_REAR=0
uint8 RIGHT_REAR=0
```

controllersAndSensors_communications/ butterflyMsg

```
Header header
float32[2] voltage #in volts
```

controllersAndSensors_communications/ co2Msg

```
Header header
uint32 ppm
```

controllersAndSensors_communications/ sonarMsg

```
Header header
int32[5] distance #in mm
```

controllersAndSensors_communications/ compassMsg

```
Header header
float32  pitch # in rads
float32 roll  # in rads
```

controllersAndSensors_communications/ irMsg

```
Header header
int32[5] distance #in mm
```

controllersAndSensors_communications/ headIrMsg

```
Header header
int32 distance #in mm
```

controllersAndSensors_communications/ mlxTempMsg

```
Header header
float32[2] mlxTemp #Celcius
```

controllersAndSensors_communications/ headSonarMsg

```
int32[5] distance
```

Vision package

vision

Required Interfaces	Provided Interfaces	
	vision_communications/ hazmatIdentificationMsg	/vision/hazmats
controllersAndSensors_communications/ controlSwitchSrv	vision_communications/ lineColorMsg	/vision/arenaColors
	vision_communications/ victimIdentificationDirectionMsg	/vision/victimDirection
	vision_communications/ victimIdentificationPositionMsg	/vision/victimPosition
	imageTransport	/vision/image
	std_srvs::Empty	/vision /motionArmService

/sensors/ledControl

Defined Messages & Services

vision_comunications/ victimIdentificationPositionMsg

```
Header header
  uint32 seq
  time stamp
  string frame_id
int32[3] position
float32[3] normalVector
float32 probability
uint64 holeId
```

vision_comunications/ hazmatIdentificationMsg

```
Header header
  uint32 seq
  time stamp
  string frame_id
float32 x
float32 y
uint8 patternType
```

vision_comunications/lineColorMsg

```
uint8 COLOR_YELLOW=1
uint8 COLOR_RED=2
uint8 COLOR_ORANGE=3
uint8 ORIENTATION_NONE=0
uint8 ORIENTATION_HORIZONTAL=1
uint8 ORIENTATION_VERTICAL=2
Header header
  uint32 seq
  time stamp
  string frame_id
float32 fromAngle
float32 toAngle
uint8 color
uint8 orientation
```

vision_comunications/ victimIdentificationDirectionMsg

```
uint8 FACE=1
uint8 HOLE=2
uint8 SKIN=3
uint8 MOTION=4
Header header
  uint32 seq
  time stamp
  string frame_id
float32 x
float32 y
float32 probability
int32 area
uint8 type
uint64 holeId
```

GUI package

gui

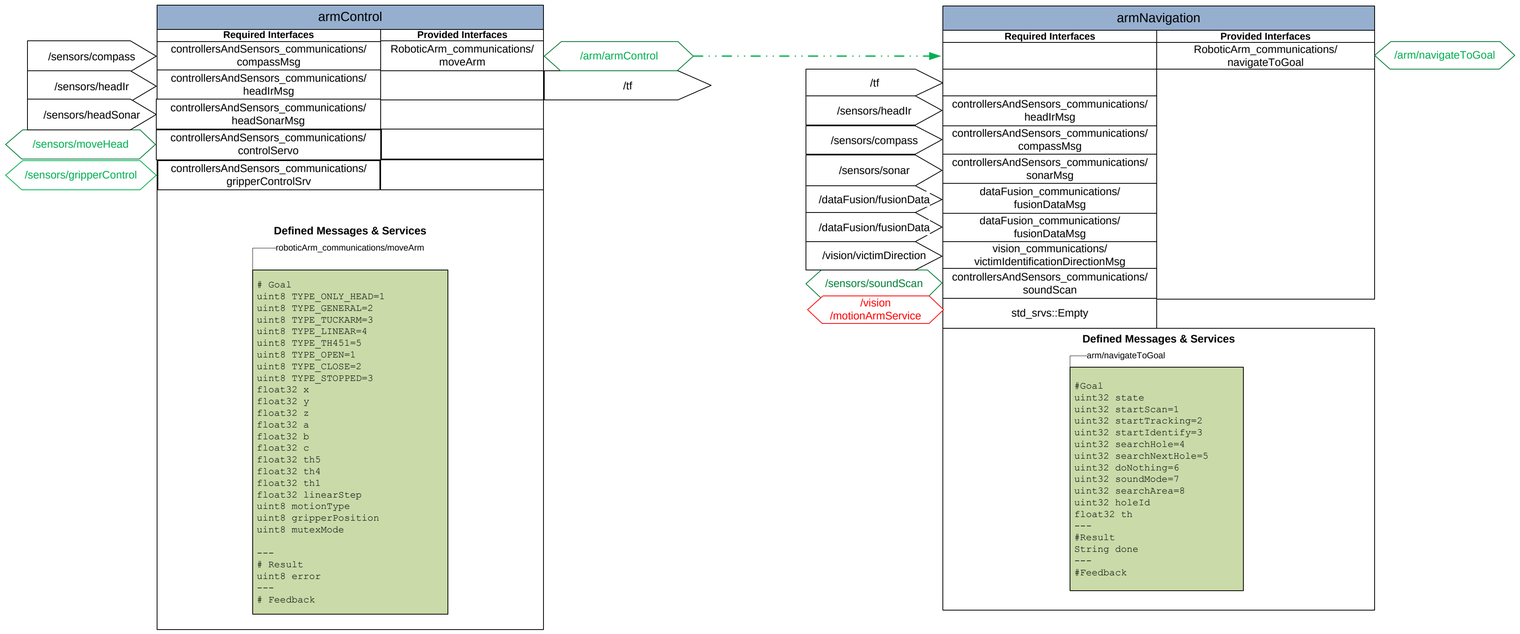
Required Interfaces	Required Interfaces	
		gui_communications/targetPosition	/guiMsgs/semiAutonomous
/mainMotorControl/ setVehicleSpeed	mainMotorControl/setVehicleSpeed	
/arm/armControl	RoboticArm_communications/moveArm	
/sensors/mlx	controllersAndSensors_communications/mlxMsg	
/tf		
/sensors/soundExistence	controllersAndSensors_communications/soundExistenceMsg	
/sensors/ir	/gui/sensors/ir	controllersAndSensors_communications/irMsg
/sensors/headIr	/gui/sensors/headIr	controllersAndSensors_communications/headIrMsg
/sensors/tpa/*(0,1,2)	/gui/sensors/tpa/*(0,1,2)	controllersAndSensors_communications/tpaMsg
/sensors/co2	/gui/sensors/co2	controllersAndSensors_communications/co2Msg
/sensors/butterfly	/gui/sensors/butterfly	controllersAndSensors_communications/butterflyMsg
/sensors/sonar	/gui/sensors/sonar	controllersAndSensors_communications/sonarMsg
	/gui/sensors/headSonar	controllersAndSensors_communications/headSonarMsg
/dataFusion/fusionGlobal	dataFusion_communications/fusionGlobalMsg	
/navigation/victimsData	navigation_communications/victimsMsg	
/vision/image2	imageTransport	
/navigation/navigationStreaming	imageTransport	
/mainMotorControl/status	mainMotorControl_communications/mainMotorStateMsg	
/robot/state	stateManager_communications/robotModeMsg	

Defined Messages & Services

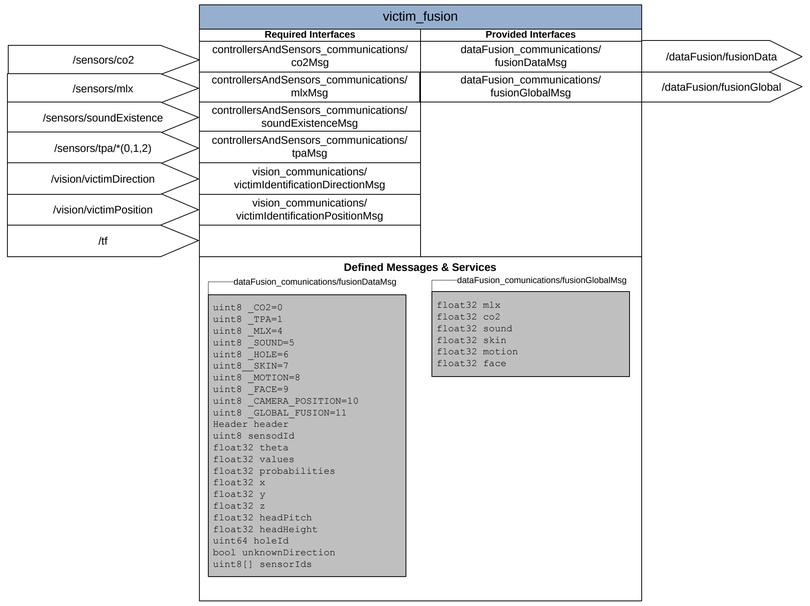
gui_communications/targetPosition

```
float32 x
float32 y
int32[] sensorsIDs
```

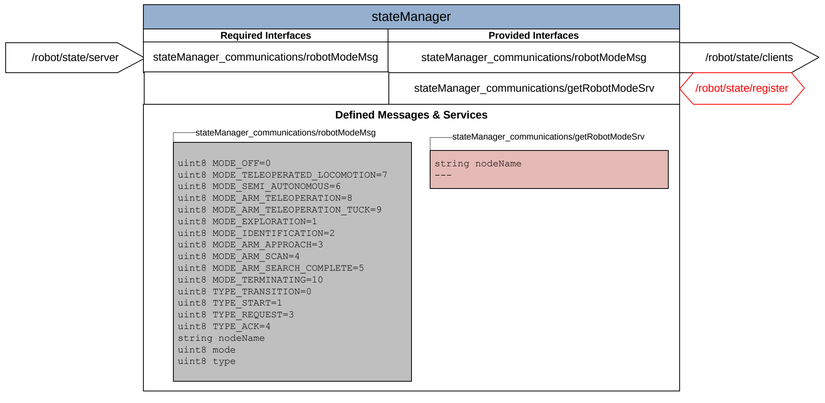
Robotic Arm package



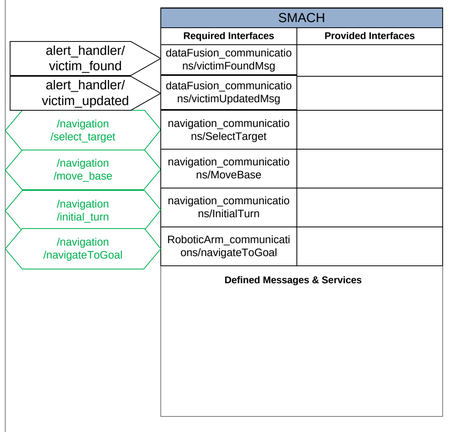
Data Fusion package

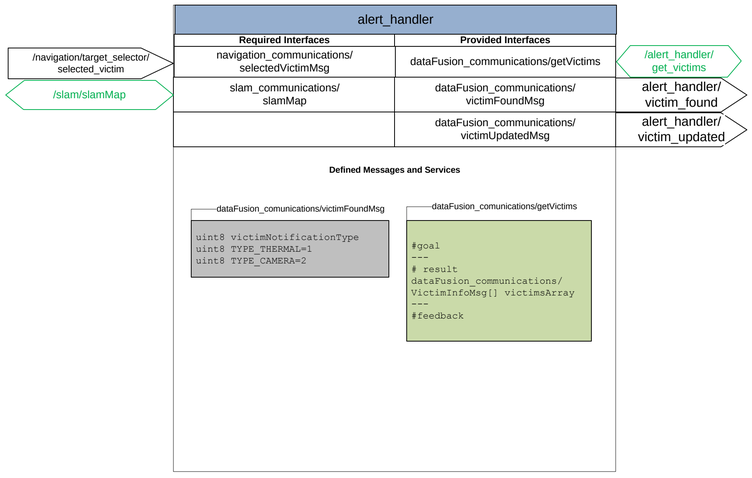
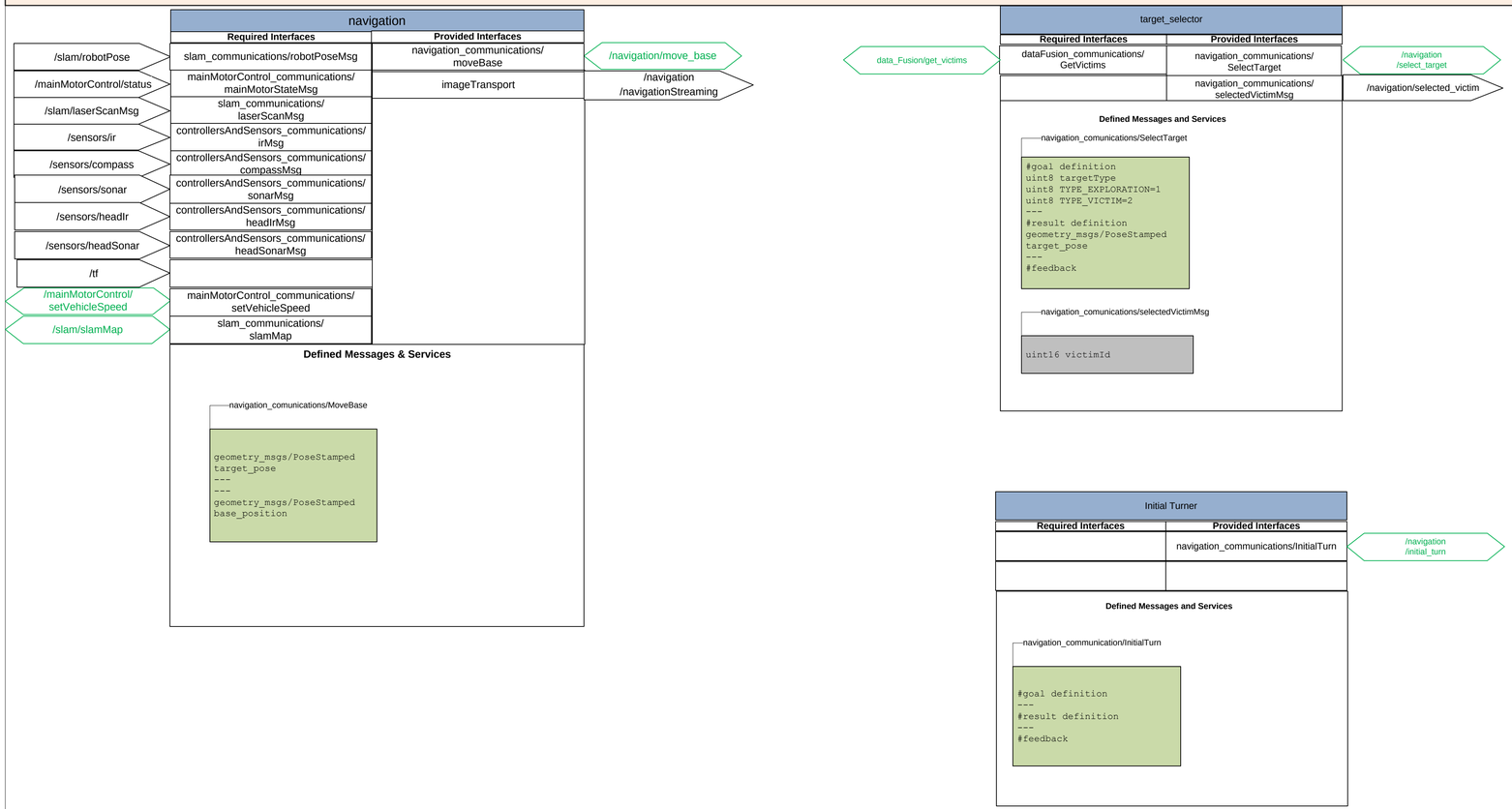


State Manager package



FSM





Navigation package