Class ownership diagram for OMPL.

"A → B" means A is owned by B.

Only when planning with controls

ControlManifold

Represents the control space the planner uses to represent inputs to the model of the system planning is performed for.
Provides propagation of the system to be planned for using sampled inputs.

ControlSampler

Implements sampling of controls for a specific ControlManifold.

StateManifold

Represents the state space in which planning is performed. Implements topologically specific functionality:

distance()
Interpolate()
alloc/free/copy states
...

ManifoldStateSampler

Implements uniform and Gaussian sampling of states for a specific StateManifold.

ProjectionEvaluator

Computes projections from states of a specific state manifold to a small dimensional Euclidean space.

SpaceInformation

Provides routines typically used by motion Planners. Combines the functionality of classes it depends on is a useful fashion.

MotionValidator

Provides the ability to check the validity of path segments, In accordance to the interpolation provided by the state manifold.

StateValidityChecker

Decides whether a given state from a specific state manifold is valid or not.

ValidStateSampler

Provides the ability to sample valid states.

Planner

Solves a motion planning problem.

ProblemDefinition

Specifies the instance of the planning problem to be solved. This implies definition of start states and goal representation.

Goal

Representation of a goal.

Path

Representation of a path. Usually used to represent solutions.

SimpleSetup

Provides a simple way of setting up all needed classes for planning without limiting functionality.

User code

Legend

User must instantiate an instance of this class

User must instantiate an instance of this class unless SimpleSetup is used (in which case SimpleSetup instantiates it)

User can instantiate an instance of this class, but reasonable defaults are provided