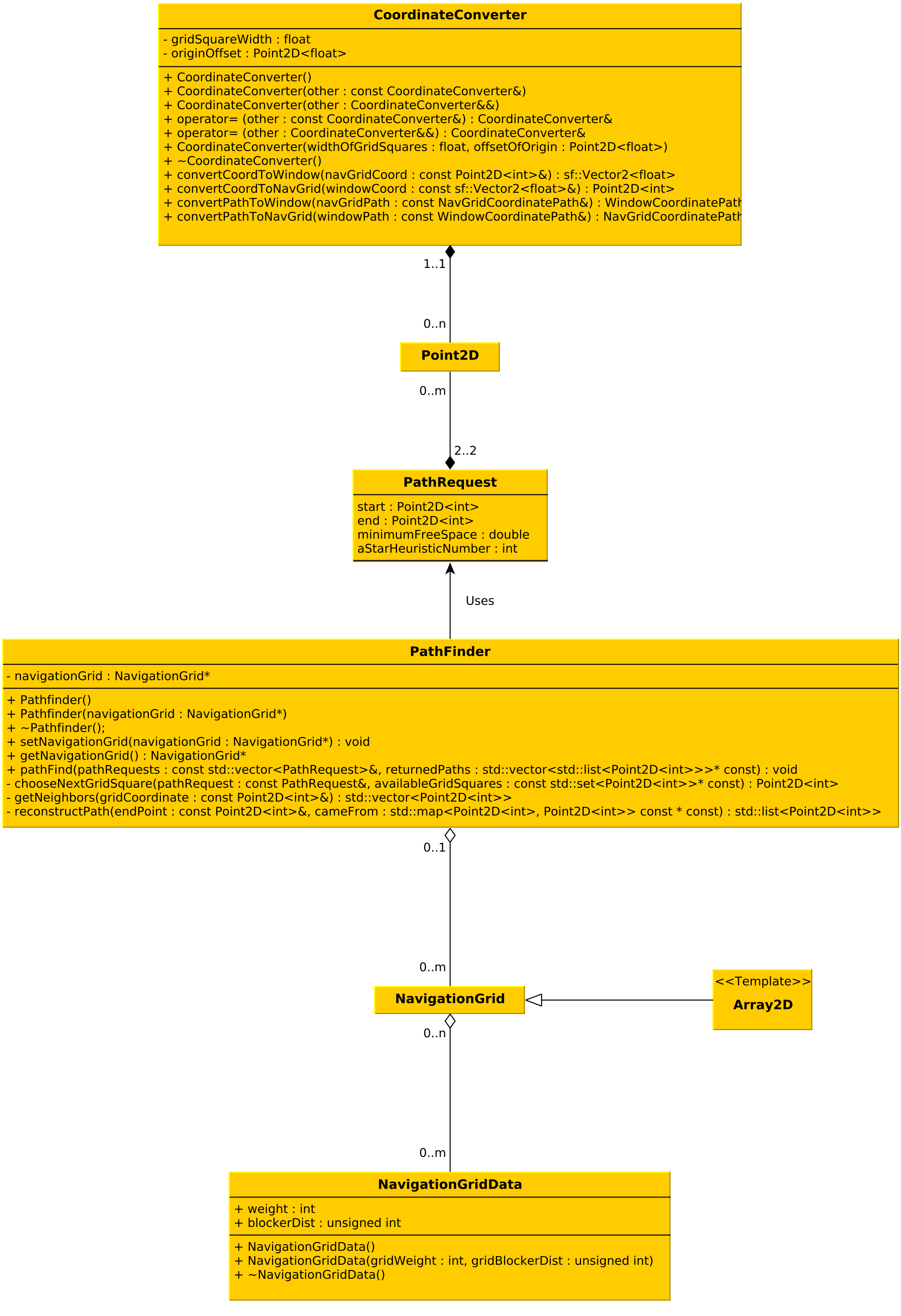
CoordinateConverter
- gridSquareWidth : float
- originOffset : Point2D<float>
+ CoordinateConverter()
+ CoordinateConverter(other : const CoordinateConverter&)
+ CoordinateConverter(other : CoordinateConverter&&)
+ operator= (other : const CoordinateConverter&) : CoordinateConverter&
+ operator= (other : CoordinateConverter&&) : CoordinateConverter&
+ CoordinateConverter(widthOfGridSquares : float, offsetOfOrigin : Point2D<float>)
+ ~CoordinateConverter()
+ convertCoordToWindow(navGridCoord : const Point2D<int>&) : sf::Vector2<float>
+ convertCoordToNavGrid(windowCoord : const sf::Vector2<float>&) : Point2D<int>
+ convertPathToWindow(navGridPath : const NavGridCoordinatePath&) : WindowCoordinatePath
+ convertPathToNavGrid(windowPath : const WindowCoordinatePath&) : NavGridCoordinatePath
1..1
0..n
Point2D
0..m
2..2
PathRequest
start : Point2D<int>
end : Point2D<int>
minimumFreeSpace : double
aStarHeuristicNumber : int
Uses
PathFinder
- navigationGrid : NavigationGrid*
+ Pathfinder()
+ Pathfinder(navigationGrid : NavigationGrid*)
+ ~Pathfinder();
+ setNavigationGrid(navigationGrid : NavigationGrid*) : void
+ getNavigationGrid() : NavigationGrid*
+ pathFind(pathRequests : const std::vector<PathRequest>&, returnedPaths : std::vector<std::list<Point2D<int>>>* const) : void
- chooseNextGridSquare(pathRequest : const PathRequest&, availableGridSquares : const std::set<Point2D<int>>* const) : Point2D<int>
- getNeighbors(gridCoordinate : const Point2D<int>&) : std::vector<Point2D<int>>
- reconstructPath(endPoint : const Point2D<int>&, cameFrom : std::map<Point2D<int>, Point2D<int>> const * const) : std::list<Point2D<int>>
0..1
0..m
NavigationGrid
0..n
<<Template>>
Array2D
0..m
NavigationGridData
+ weight : int
+ blockerDist : unsigned int
+ NavigationGridData()
+ NavigationGridData(gridWeight : int, gridBlockerDist : unsigned int)
+ ~NavigationGridData()