**User stories of know-how acquisition**

## Story 4.1 Repast

To look further into the possibilities of using Repast we have to do extra exploration work. Repast is used throughout the BW4T codebase, but is essential for the simulation to work. Right now Repast is mostly used to keep track of objects and their locations in the continuous space. Furthermore, the “moveTo” method of the continuous space is indirectly used to move the robots (they are drawn by using the location on the grid), but the path ﬁnding isn’t done by Repast. In fact, Repast does not have a pathﬁnder. In Repast agents are moved by using either the “moveTo”, “moveByDisplacement” or “moveByVector” methods. In a nutshell, Repast is currently used as a collection of all objects in the simulation space, and to map objects to a certain point in the simulation space. The fact that Repast isn’t needed that much at the moment, doesn’t mean that we should get rid of it. It oﬀers some features such as the RandomCartesianAdder class which allows for adding objects at a random location in a space, and which could therefore be used to generate random maps. In addition it has support for getting speciﬁc objects or agents from neighbourhood grid spaces. Therefore we should further explore the possibilities of Repast in improving our system.

## Story 4.2 Existing code analysis

The code we have received to work with and refactor was not written by us and therefore requires our time to analyze what is already in the code and what should be changed or left out. We know that the current version of the source code is unstructured, difficult to work with and uses an enormous hashmap. Before we can refactor the current code we should first look in to code architecture and understand it.