

# Codebase Documentation

Daniel Swaab  
4237455

Sille Kamoen  
1534866

May 16, 2014

This document will contain a simple discription about the codebase of BW4T. We will use som UML-diagrams to make relations clear. Use this when you want to refactor/understand the code. It took us hours to comprehend the code before we could start refactoring, we hope this will allow other people to get an understanding of the code in a shorter time period.

## The begin

The project is split up into 3 sub-projects: *Core*, *Client* and *Server*.

- The Server project contains all classes needed to run a BW4T-ServerInstance.
- The Core project contains classes that are shared by both client and server. It can be seen as a shared library.
- The Client project contains all classes to run a BW4T-ClientInstane.

## EIS

A lot of classes make use of the *EIS* library. *EIS* is an Environment Interface Standard. Consider this a black box that you use to implement environments. It contains function/methods/interfaces to run enviroments.

Sille Kamoen: "You dont know what it is, but you do know what it does"

***INSERT IMAGE***

## Server

You would think this is a client-server architecture but it's more complicated. The central class is the Singleton *BW4TEnvironment.java*. This is a class that contains the main method to start the server entity. From javadoc:

The central environment which runs the data model and performs actions received from remote environments through the server. Remote environments also poll percepts from this environment. Remote environments are notified of entity and environment events also using the server.

The environment uses a "BW4T-server" instance to communicate with the connecting clients. The name "server" can be a little confusing because of this. The environment extends an *AbstractEnvironment* from the "EIS" library.