Performance Report

Modular AI Unity package

Benjamin Wharton

Issues encountered when integrating the modular complex system into the test application.

* Implementing delegate systems had some limitations
  + There was a limit on the number of parameters which can be used
    - For the delegation of methods to work correctly, the container class had to specify a set number of parameters that a method must have in order to be used
    - This meant that when dealing with them in the unity inspector, I was unable to provide the user the option to enter pre-defined parameters. This is often important for interrupt conditions, for example, the timer condition would ideally be able to take a pre-defined float number that represents a time in seconds.
    - A solution to this was creating a separate type of delegate container class which completely removed the use of delegates and stored parameters for use at invocation.
    - In order to receive parameters from the inspector, a new container had to be created which contains a set of basic data types which can be set. Through the inspector, the method parameter types were checked, and corresponding input fields displayed for any of the basic types in the container class. At initialisation, the container class could then check for the type being used and return that back to the delegate class for use in invocation.

Be sure to outline any changes required when implementing your system that were different to the details included in your initial project brief.

The performance of the system.  
How you measure performance will be dependent on the type of complex system implemented. However, you should aim to benchmark your system against similar implementations.

Any required changes for the system to function as intended.