DARBS v2: Development of an Open-Source Blackboard System

Programmer: Joe Turner (joe@oampo.co.uk)
Principal Investigator: Prof. Adrian Hopgood, Faculty of Technology, De Montfort University (aah@dmu.ac.uk)
Co-investigator: Prof. Andrew Hugill, Institute of Creative Technologies, De Montfort University

Open Source

The source code for this software system has been made freely available under an open-source licence at
     http://darbs.sourceforge.net/
to provide a central location and to build a community of users and developers.
Please use it!

What is DARBS?

DARBS (Distributed Algorithmic and Rule-based Blackboard System) is a software tool for building multi-agent intelligent systems. It enables a variety of agents to work collaboratively, sharing data and knowledge via a server known as the Blackboard (Fig.1). The agents can run simultaneously, either on one computer or many across the Internet. DARBS is particularly well suited to complex problems that require a combination of software techniques.
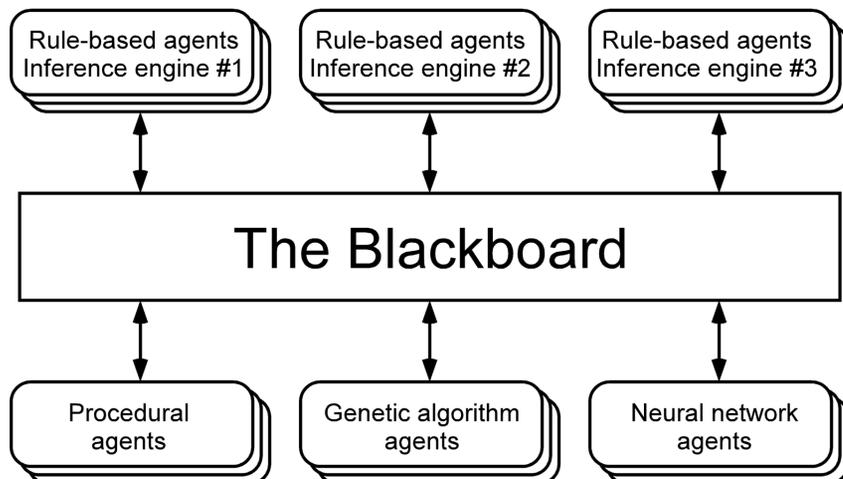
Unifying DARBS

The prior development of DARBS by multiple researchers had led to four different versions, each with different features. The different versions of DARBS have now been unified, taking the improvements from each into a common version. Boost libraries were used, reducing maintenance and removing all platform-dependency.

Moving to XML

Older versions of DARBS had used text strings for Blackboard information and messages, but one version had an additional layer that translated messages into XML format. The translation was inefficient, so the textual string format has now been completely replaced with an XML format (Fig. 2), with the following advantages:
 - easier and faster to parse;
 - easier to extend;
 - allows nesting of requests;
 - widely supported;
 - standardised.

Web-Based Interface

Previous versions of DARBS offered command-line tools and a graphical user interface (GUI) based on an obsolete and platform-specific toolkit. A new GUI has been implemented that is web-based and platform-independent.

The GUI (Fig. 3) gives a continually updated display of the contents of the Blackboard, along with the status of any active agents. From the GUI, the Blackboard can be manually edited and agents can be started, stopped and debugged. A novel feature is that the GUI itself is an agent, communicating with other agents via the Blackboard.
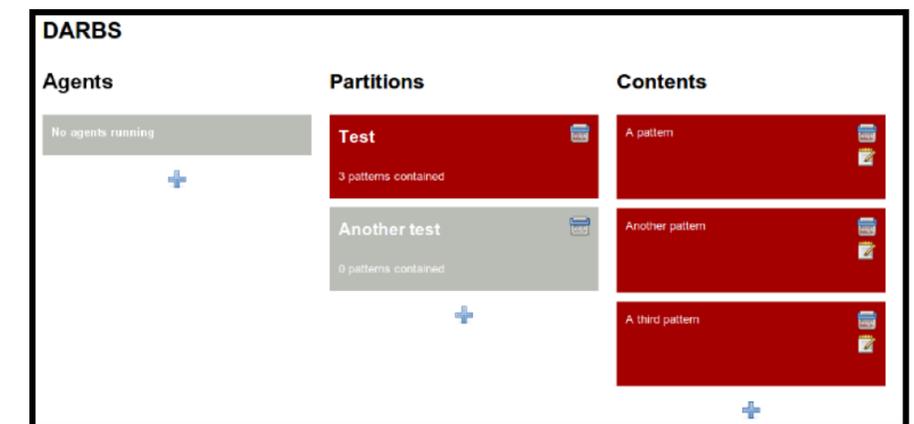

Fig. 3 - DARBS graphical user interface

Fig. 1 - DARBS structure

Old text-string syntax

```
KS Square

   KS_TYPE procedural_KS

   INFERENCE_MODE MI_Forwardchain

   FIRABILITY_FLAG true

IF
[
   [not_on_partition [Square is fired] ControlChars]
]

THEN
[
   [run_algorithm[libsquare.so square[2.34] result]]
   AND
   [add [Square is fired] ControlChars]
   AND
   [report [Square of 2.34 is ~result]]
]

END
```

New XML syntax

```
<ks type="procedural">
   <precondition type="not">
      <request type="present">
         <partition name="control">
            <pattern>
               Square is fired
            </pattern>
         </partition>
      </request>
   </precondition>

   <action>
      <request type="run" library="libsquare.so"
                  function="square" variable="result">
         <pattern>2.34</pattern>
      </request>

      <request type="report">
         <pattern>
            The square of 2.34 is ~result
         </pattern>
      </request>

      <request type="add">
         <partition name="control">
            <pattern>
               Square is fired
            </pattern>
         </partition>
      </request>
   </action>
</ks>
```

Fig. 2 - Old and new DARBS syntax