

};

The array describes a graph with four states s1-s4, and five arcs a1-a5. From node s0 we can take arc a1 to state s1 or arc a2 to state s4 or alternative  
 5 arc a3 to state s4. When more than one can be traversed from a node, the arcs are tried in the order in which they are presented in the graph description.

The code with which a user defines a node is simply required to implement an *exec()* and a *backtrack()* method. For arcs the code should implement an *exec()* method which returns a boolean result of true or false.

10

### 2.3.2 Coordination Mechanisms

Referring to Figure 11, in CABS agents every coordination mechanism is specified in terms of a 14-stage framework where in each stage at least one state  
 15 process function should be implemented. The 14-stage framework can be considered an "executive summary" of the detailed logic of the co-ordination engine 210 set out in code above. Generic atomic process functions for the fourteen stages are listed below. Figure 11 describes in schematic form the stages listed below.

20

#### Resource Phase

In this phase, an agent A2 has been triggered by an incoming message from another Agent A1 which for instance needs to delegate tasks. The incoming  
 25 message will be sent to the co-ordination engine and reasoning system 210 of A2 which will call on the planner and scheduler 220, and on the various databases 230, 225, 215, 245 in building and running a co-ordination graph in order to respond to agent A1. In the resource phase, agent A2 will use the task database 230 to see what resources a task requires, and the resource and commitment  
 30 databases 225, 245 to see if it has those resources available.

#### Stage One

Verify resource availability and determine actions for situations of sufficient/insufficient resources;