

Assignment 10

SE1 Proseminar, winter 2007/2008

Stefan Resmerita

stefan.resmerita@cs.uni-salzburg.at

In this exercise, you are required to use the Abstract Factory design pattern to allow creation of two types of teams: autonomous teams and dependent teams. An autonomous team decides whether a robot member accepts a job or not, without consulting its parent team (if it has one). A dependent team always asks its parent team if a member robot may accept a job or not. If a dependent team has no parent, we make the convention that any job is accepted (i.e., the team has no objection for the robot member to take the job). You do not need to implement the decision making procedure in this assignment.

Implement the following command:

TeamType <type>

where <type> can be either *autonomous* or *dependent*. After the execution of this command, all the new teams created by subsequent **CreateTeam** commands will have the type given by <type>, until the next **TeamType** command is issued. The initial team type is specified in the configuration file. For uniformity, let us set it to *dependent*.

The team type should be displayed by the **ShowContents** command, as in the following example.

```
>CreateTeam TransportRobot=2
2001
>CreateTeam PaintingRobot=1 CleaningRobot=1
2002
>TeamType autonomous
>CreateTeam PaintingRobot=2 2001 2002
2003
>ShowContents 2003
Begin Team 2003 autonomous
  Begin Team 2001 dependent
    TransportRobot 101
    TransportRobot 102
  End Team 2001
  Begin Team 2002 dependent
    PaintingRobot 103
    CleaningRobot 104
  End Team 2002
PaintingRobot 105
PaintingRobot 106
End Team 2003
```

The checking interface for this assignment should be the same as for the previous assignment.