# Assignment 4

SE1 Proseminar, winter 2007/2008

Stefan Resmerita
stefan.resmerita@cs.uni-salzburg.at

## The Composite Pattern

In this exercise, you are required to include teamwork features to the factory floor simulator.

**A.** Change the implementation of the **`PowerOn`** command such that it returns a unique identification tag associated to the robot. As usual, the command parameter is the robot type. For example, this can be ***`TransportRobot, PaintingRobot, Cleaning Robot,`*** etc.

**B.** Implement the following commands:

```
1. CreateTeam <(robot_type1=n1)|(Team_id1)> [<(robot_type2=n2)|(Team_id2)> ...]
```

A new team is created, consisting of already existing teams and/or new robots of specified types. A unique team identification tag must be displayed at the standard output as a result of this command. In case of an error, an adequate error message should be displayed.

Examples:

```
CreateTeam TransportRobot=2
```

Two transportation robots are powered on and grouped in a new team. Assume that the returned team identification tag is *2001*.

```
CreateTeam PaintingRobot=1 CleaningRobot=1
```

One painting robot and one cleaning robot are powered on and grouped in a new team. Assume that the returned team identification tag is *2002*.

```
CreateTeam PaintingRobot=2 2001 2002
```

Two painting robots are powered on. A new team composed of two painting robots, plus team *2001* and team *2002* is created. Consider that the id of the new team is *2003*.

A team may be contained in at most one other team. For example, the following commands should fail:

```
CreateTeam 2001 2003
CreateTeam 2002 TransportRobot=2
```

because teams *2001* and *2002* are members of team *2003*.

### 2. AddToTeam *<team_id> <id1> [<id2> ...]*

The robots or teams specified by ***<id1>, <id2>,...*** are added to the team given by ***<team_id>*** The command fails if one of the items to be added is already member of some other team.

### 3. GetType *<id>*

If ***<id>*** is the identification tag of a robot, then the type of the robot is displayed. If ***<id>*** refers to a team, then the types of all robots contained by the team are displayed. For each type, the total number of robots in the team is displayed. The robots of a team include the robots contained in all the sub-teams of the team.

Example (continuation from the previous page):

```
> PowerOn TransportRobot                  // assume this returns 101
> GetType 2001
TransportRobot=2

> GetType 101
TransportRobot

> GetType 2003
TransportRobot=2
PaintingRobot=3
CleaningRobot=1

> AddToTeam 2001 101
> GetType 2003
TransportRobot=3
PaintingRobot=3
CleaningRobot=1

> CreateTeam CleaningRobot=4 TransportRobot=1    // assume this returns 2004
> AddToTeam 2002 2004
> GetType 2003
TransportRobot=4
PaintingRobot=3
CleaningRobot=5

> AddToTeam 2003 2004      // fails, because 2004 is already a member of 2002
```

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