The project ENTs
A storyboard demo

Cyril Brom, Ondřej Bojar
Charles University in Prague
Faculty of Mathematics-Physics
brom@ksvi.mff.cuni.cz
obo@cuni.cz
2/2005
Introduction to the project ENTs:

1. What are ents (human-like agents)?
2. What can they do? What they aim at?
3. How can a user interact with them?
4. What is the purpose of the tool?

A world of ents just after startup – the gardener is sitting in a chair and the user-controlled agent is looking at it.
We observe the behaviour of an artificial gardener:

1. After it finds a can, the gardener will start watering dry beds in the garden.

2. The gardener moves from one dry bed to another.

Up to now, the life has been simple…

A user-agent is looking at the gardener who is just watering a bed.
We observe reactive switching in the behaviour of the gardener:

1. When the can becomes empty the gardener must *reactively* switch to another behaviour.

2. The can can become empty at any time… no way to plan this! What else can change anytime? … Whatever! For instance, someone else can water the beds instead of the gardener so suddenly there may be no dry bed left.

3. The gardener chooses the closest basin to refill the can.

The user-agent is looking at the gardener who has just chosen the closest basin and who is going to refill the can there.
Interacting with the gardener...

1. Using a text input field to formulate an utterance (optionally pointing at an object), we ask the gardener to pick up a can.

2. The gardener obeys the command and interrupts hoeing the beds to pick up the selected or described can.
We demonstrate that the gardener carries out diverse set of goals, in particular it must satisfy all its internal needs like thirst and hunger:

1. If hungry, the gardener can pull up a carrot and wash it (or find a bread or other suitable object), wash its hands and eat it.
2. Then it is able to continue any interrupted task.
3. In the meantime, the gardener can be again interrupted by a user-agent or by any relevant change in the world.
Towards prototyping...

We conclude with a real example on how to modify the behaviour of an agent and briefly show a debugger of scripts that control the behaviour.

```plaintext
schedulerRegisterTask("wateringGarden", _, 20, 30, 360, 400, _),
schedulerRegisterTask("hoeGarden", _, 180, 20, 720, 800, _),
schedulerRegisterInternalNeed("hunger", "eatScript", 78, _),
schedulerRegisterInternalNeed("thirst", "drinkScript", 80, _), ...
```

```plaintext
if not inHands("can") then
    { findAndTakeSubGoal(["item" = "can"],
    hCan, "S") OR { FAIL } },
    RERUN
fi,
localHook({ not inHands("can"), "PRIO_MAX",
    RERUN }, interruptCanNotInHands )
, ...
```
The End