Ao Dai: Agent Oriented Design for Ambient Intelligence

Amal El Fallah Seghrouchni, Andrei Olaru, Thi Thuy Nga Nguyen and Diego Salomone

LIP6, University Pierre et Marie Curie, Paris

- Introduction
- Context
- Agents
- CLAIM
- Scenario
- Architecture
- Agent Types
- Interaction
- Experiment
- Conclusion

Ao Dai: Agent Oriented Design for Ambient Intelligence

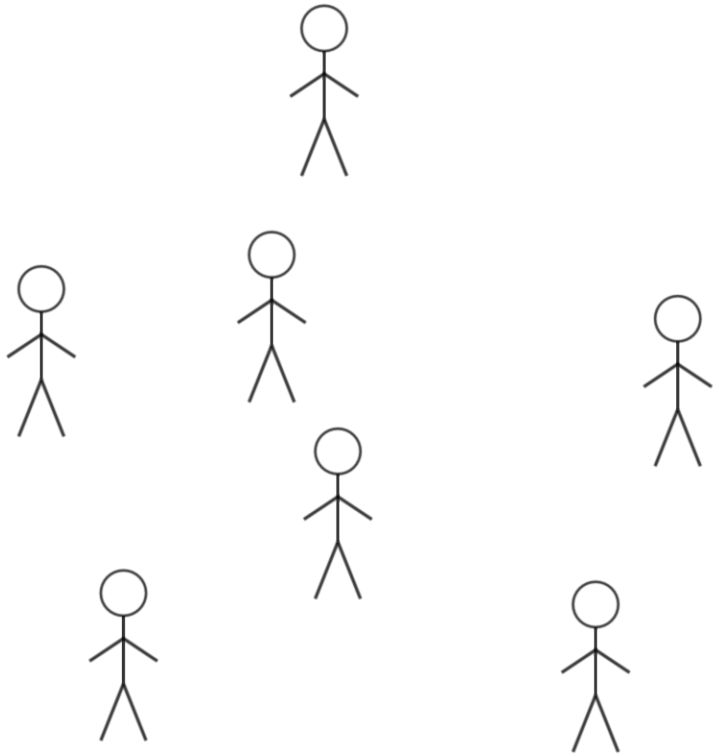
overview

. Amal El Fallah Seghrouchni, Andrei Olaru, Thi Thuy Nga Nguyen and Diego Salomone
. PRIMA 2010
. Kolkata, India,

Ubiquitous electronic environment that supports people in their daily lives, in a proactive, but "invisible" and non-intrusive manner [Ramos et al., 2008, Weiser, 1993]

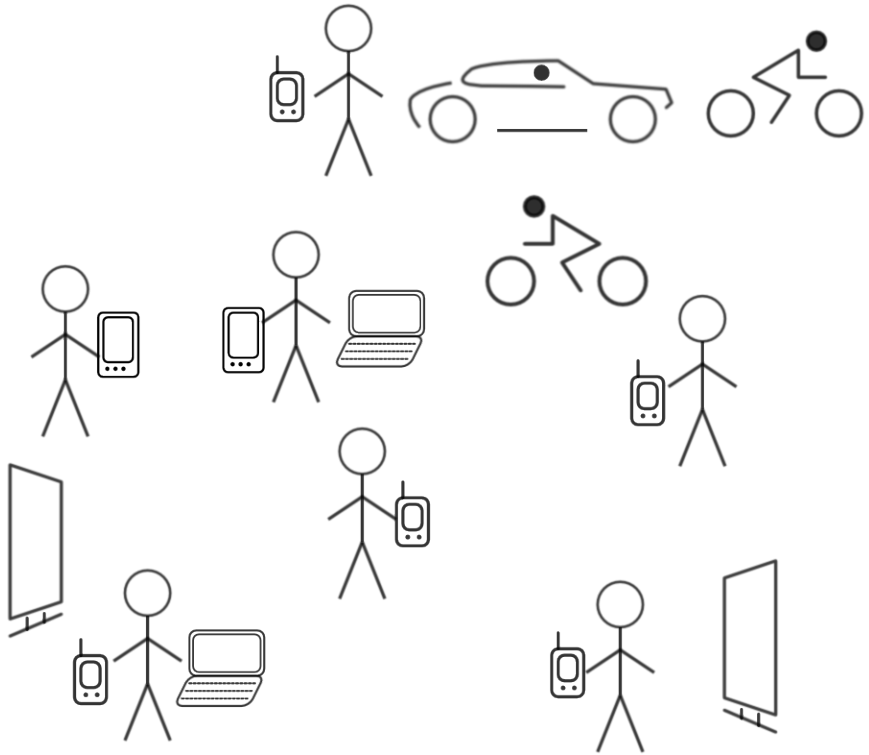
. Amal El Fallah Seghrouchni, Andrei Olaru, Thi Thuy Nga Nguyen and Diego Salomone

Ubiquitous electronic environment that supports people in their daily lives, in a proactive, but "invisible" and non-intrusive manner [Ramos et al., 2008, Weiser, 1993]



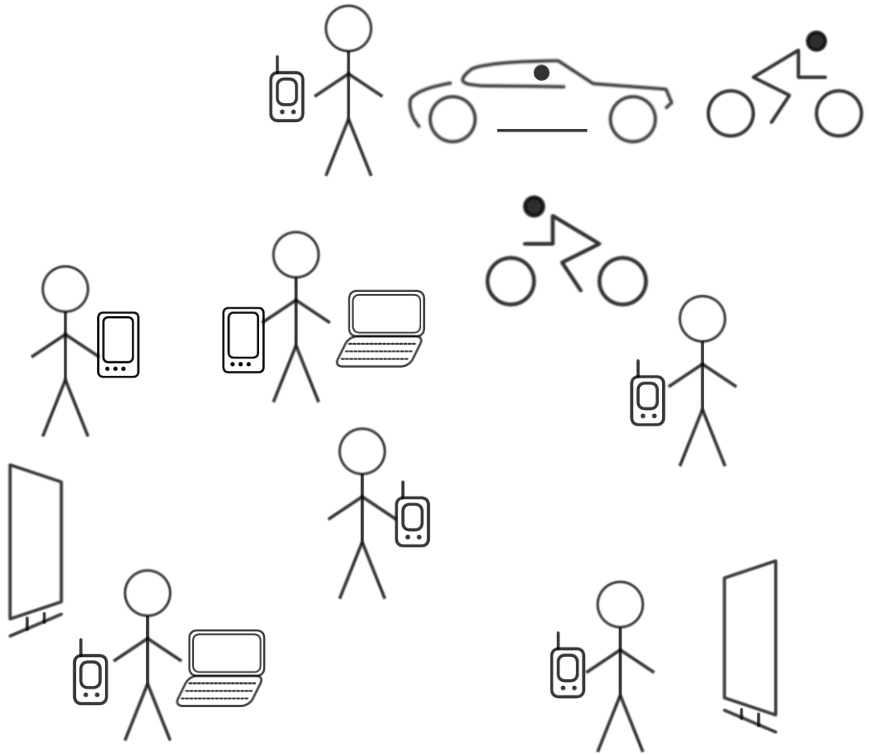
People

Ubiquitous electronic environment that supports people in their daily lives, in a proactive, but "invisible" and non-intrusive manner [Ramos et al., 2008, Weiser, 1993]



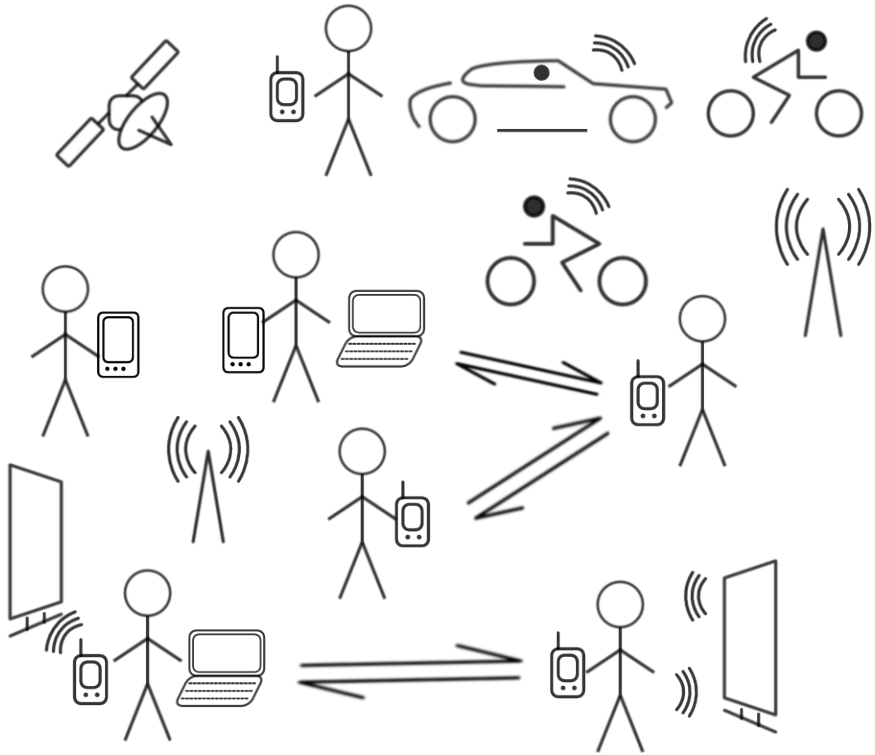
People · Devices

Ubiquitous electronic environment that supports people in their daily lives, in a proactive, but "invisible" and non-intrusive manner [Ramos et al., 2008, Weiser, 1993]



People · Devices · Services

Ubiquitous electronic environment that supports people in their daily lives, in a proactive, but "invisible" and non-intrusive manner [Ramos et al., 2008, Weiser, 1993]



People · Devices · Services · Communication

Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves. [Dey and Abowd, 2000]

Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves. [Dey and Abowd, 2000]

Aspects: [Chen and Kotz, 2000]

- physical aspect (e.g. location)
- user profile and preferences
- computing resources
- associations (e.g. time – place – activity)
- temporal aspect
- activity
- social aspect

Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves. [Dey and Abowd, 2000]

Aspects: [Chen and Kotz, 2000]

- physical aspect (e.g. location)
- user profile and preferences
- computing resources
- associations
  (e.g. time – place – activity)
- temporal aspect
- activity
- social aspect

In the Ao Dai project, we have so far considered:

- the spatial location of the user
- the user's preferences
- the available computing resources

Software agents are an appropriate implementation for AmI, considering they satisfy the needs of AmI in terms of:

- reactivity
- proactivity
- autonomy
- anticipation
- reasoning

Software agents are an appropriate implementation for AmI, considering they satisfy the needs of AmI in terms of:

- reactivity
- proactivity
- autonomy
- anticipation
- reasoning

Agents also offer beliefs, goals, intentions and easier implementation of a human-inspired behaviour.

Software agents are an appropriate implementation for AmI, considering they satisfy the needs of AmI in terms of:

- reactivity
- proactivity
- autonomy
- anticipation
- reasoning

Agents also offer beliefs, goals, intentions and easier implementation of a human-inspired behaviour.

For Ao Dai, we use CLAIM + Sympa as agent-oriented programming language and platform.

· Agent-Oriented programming language created by Alexandru Suna, during his Thesis at LIP6 [Suna and El Fallah Seghrouchni, 2004]

· Eases the programming task involving a Multi-Agent System

CLAIM is based on explicit declaration of agent's characteristics:

- Knowledge
- Goals
- Capabilities
- Procedures
  - Conditions
  - Triggers
  - ...

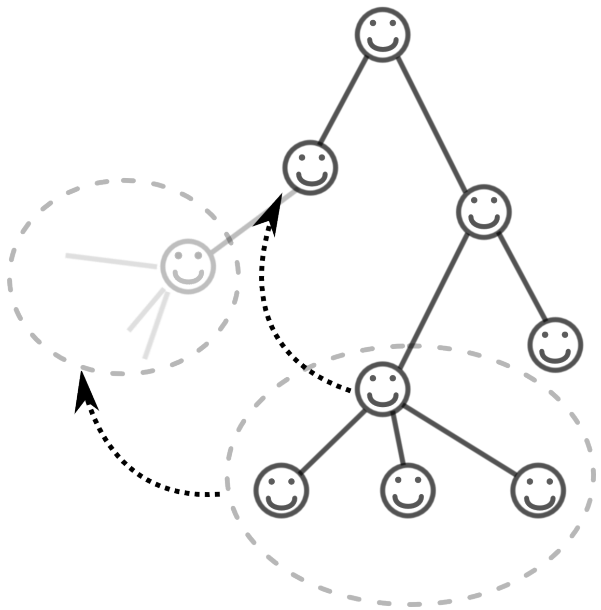
· Agent-Oriented programming language created by Alexandru Suna, during his Thesis at LIP6 [Suna and El Fallah Seghrouchni, 2004]

· Eases the programming task involving a Multi-Agent System

CLAIM is based on explicit declaration of agent's characteristics:

- Knowledge
- Goals
- Capabilities
- Procedures
  - Conditions
  - Triggers
  - ...

· an essential feature – mobility:

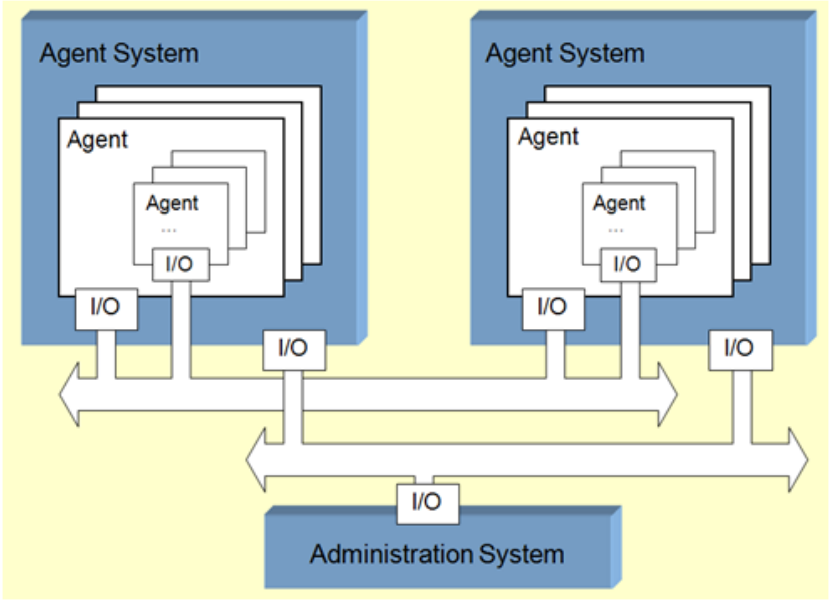


· example:

```
defineAgentClass PDA(?w,?h,?x_i,?y_i){
    authority = null;
    parent = null;
    knowledge = {location(?x_i,?y_i); type(1);}
    goals = null;
    messages = null;
    capabilities = {
        message = PDAatLoc (?name,?x_new,?y_new);
            condition = null;
            do{send(this,migrateTo(?name))}
            effects = null;
        }
        migrate{
            message = migrateTo(?name);
            condition = not(Java(PDA.isParent(this,?name)));
            do{send(this,removeOldNavi(?name))
.moveTo(this,?name).send(this,demandNavi(?name))}
            effects=null;
        }
        ...
    processes={send(this,starting())}
    agents=null;
}
```

· CLAIM agents are managed by the Sympa platform, that executes the CLAIM code

· Sympa is Java-based.



· a researcher comes for the first time to the LIP6 laboratory, for a presentation.

· as he enters the floor and the meeting will start soon, a nearby screen shows the way to the meeting room where he must go.

· in the meeting room, he needs a large screen for a presentation. The system will suggest going to another room where there is a larger screen, more adequate to the user's preferences.

Basic elements:

- physical context (localization)
- computational context (available devices / services)
- user preferences

Idea: map contexts to agents:

· each agent represents a device, or a service, or a location, or a user;

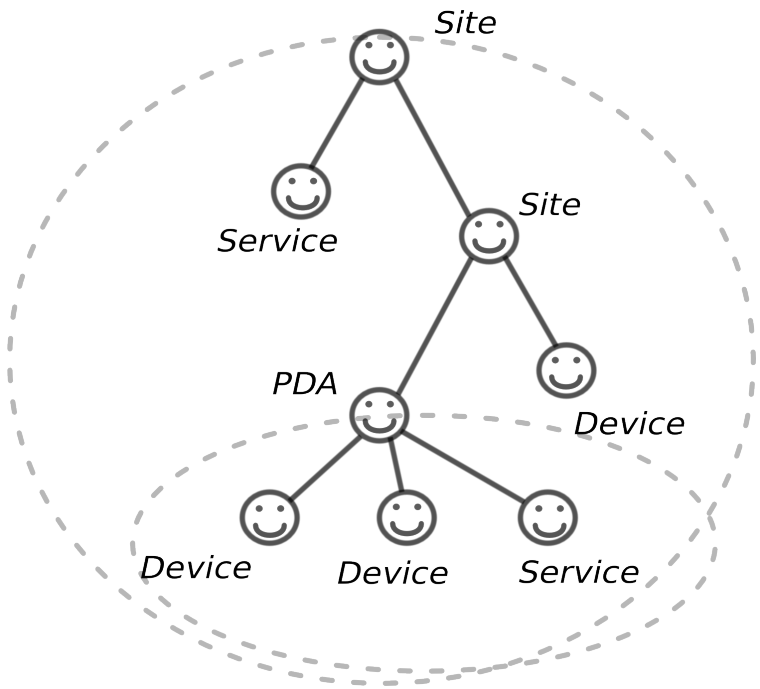
· the agent sub-tree of every agent represents the context of the agent and moves together with it.

Idea: map contexts to agents:

· each agent represents a device, or a service, or a location, or a user;

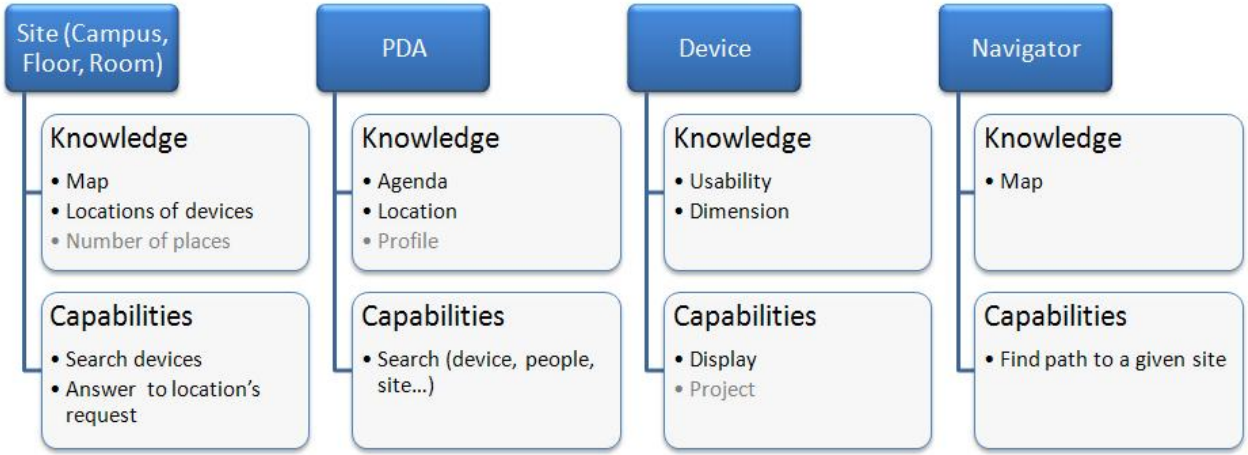
· the agent sub-tree of every agent represents the context of the agent and moves together with it.

Example:



· Types of agents:

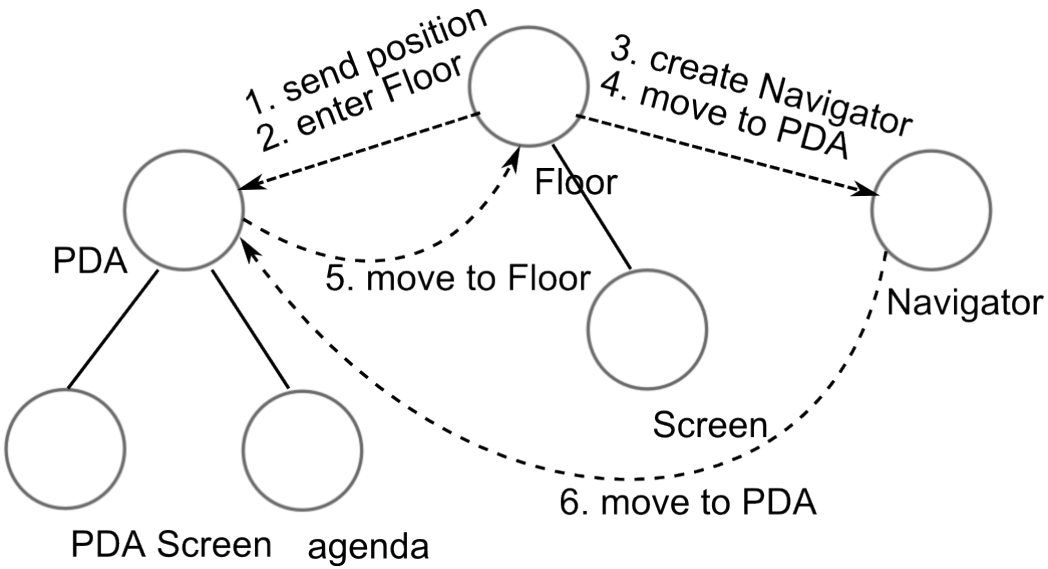
- Site (e.g. Floor, Office) – represents a physical place;
- Device / Service (e.g. Navigator, Screen) – offers a certain capability;
- PDA – directly interacts with the user.



· Agent interacts only with its parent or its children

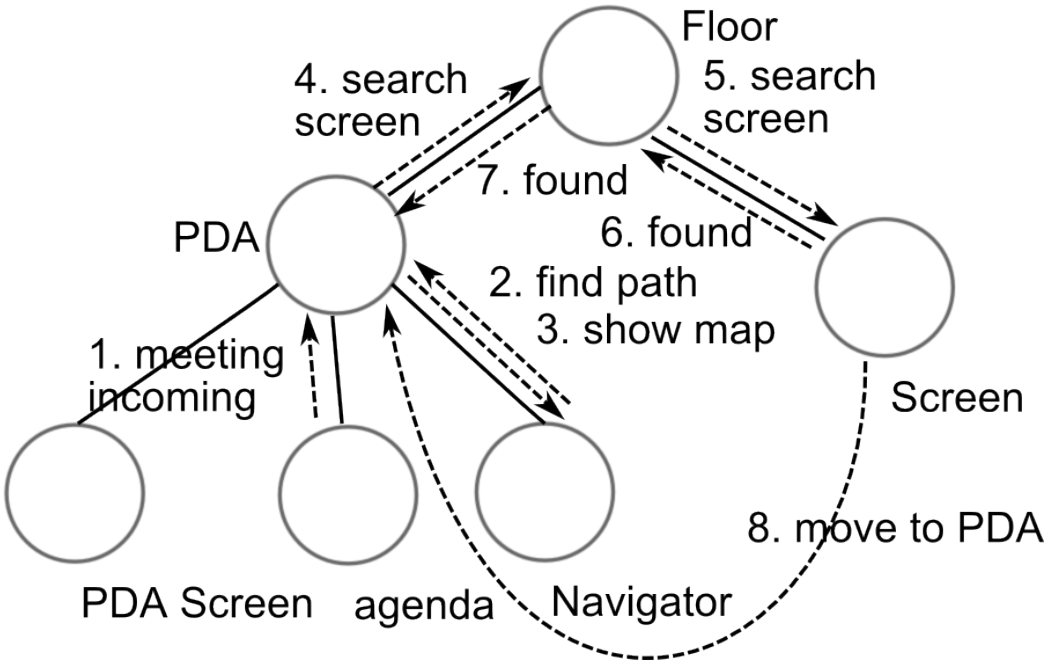
Examples:

User enters floor:



Examples:

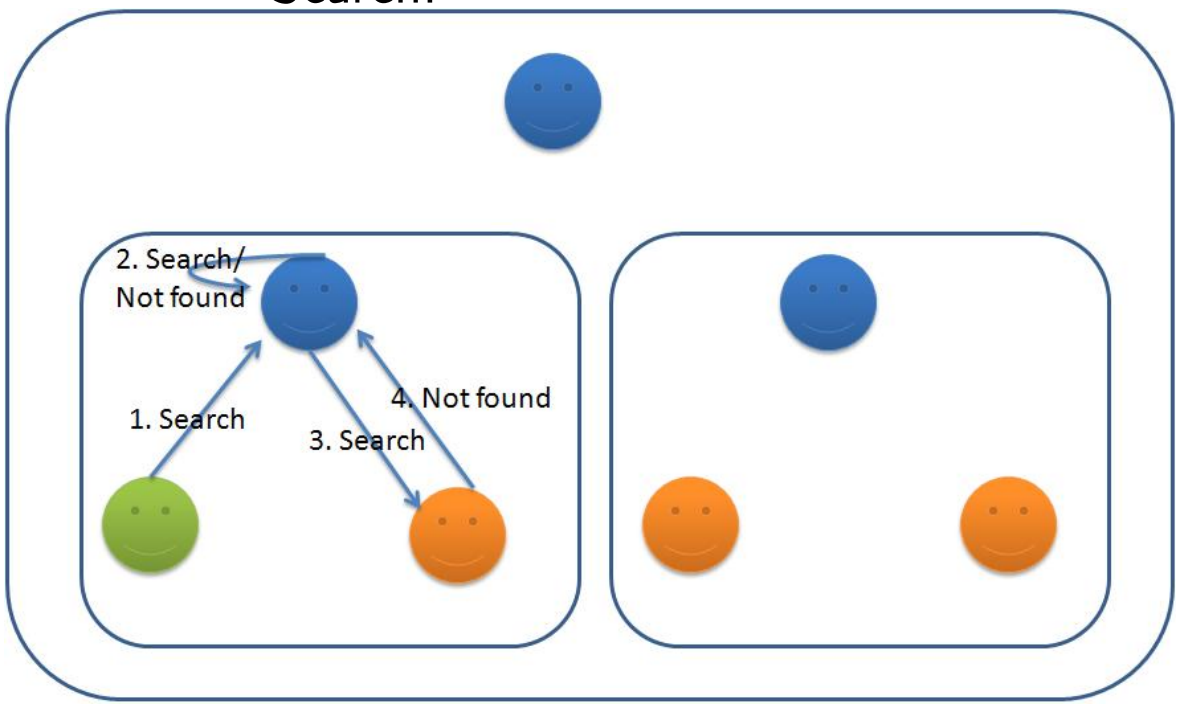
User needs a screen to show the path:



. Amal El Fallah Seghrouchni, Andrei Olaru, Thi Thuy Nga Nguyen and Diego Salomone
. PRIMA 2010
. Kolkata, India,

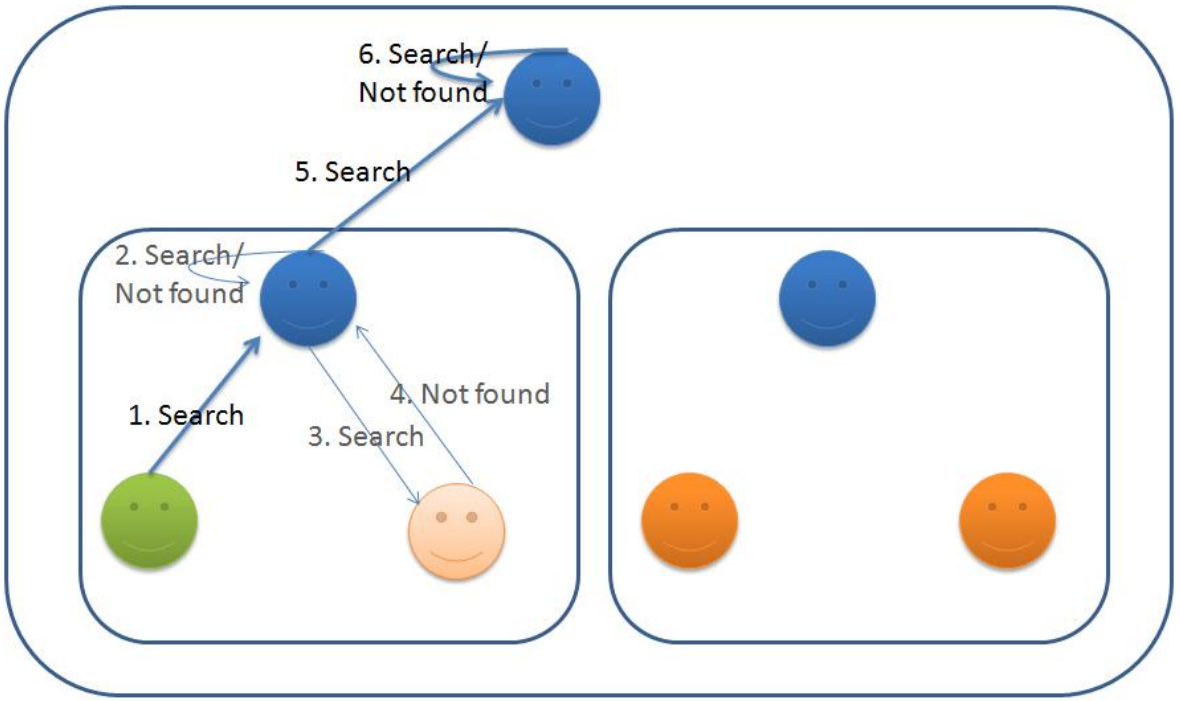
Examples:

Search:



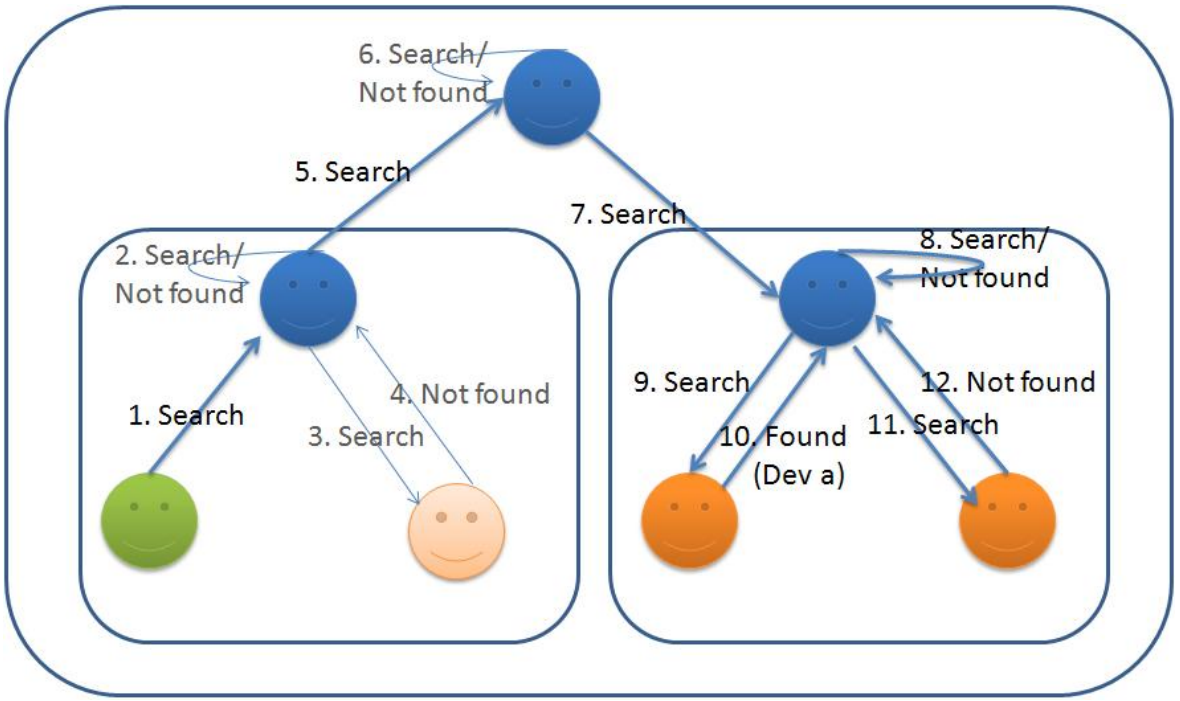
Examples:

Search:



Examples:

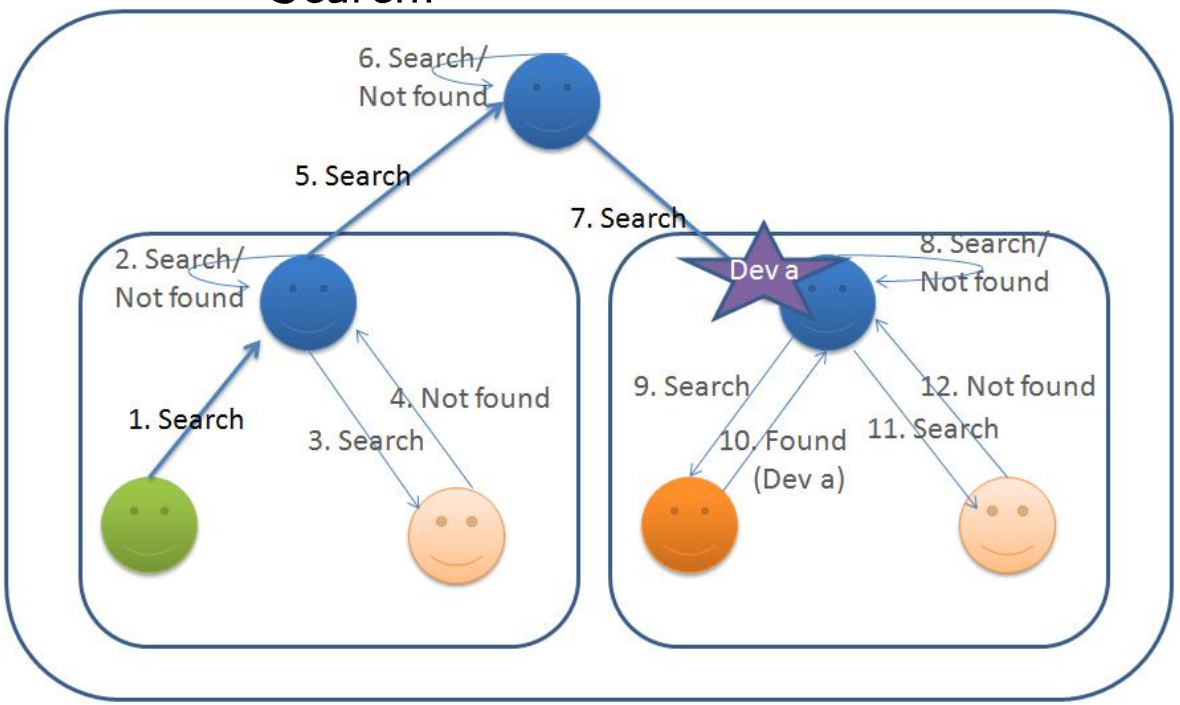
Search:



. Amal El Fallah Seghrouchni, Andrei Olaru, Thi Thuy Nga Nguyen and Diego Salomone
. PRIMA 2010
. Kolkata, India,

Examples:

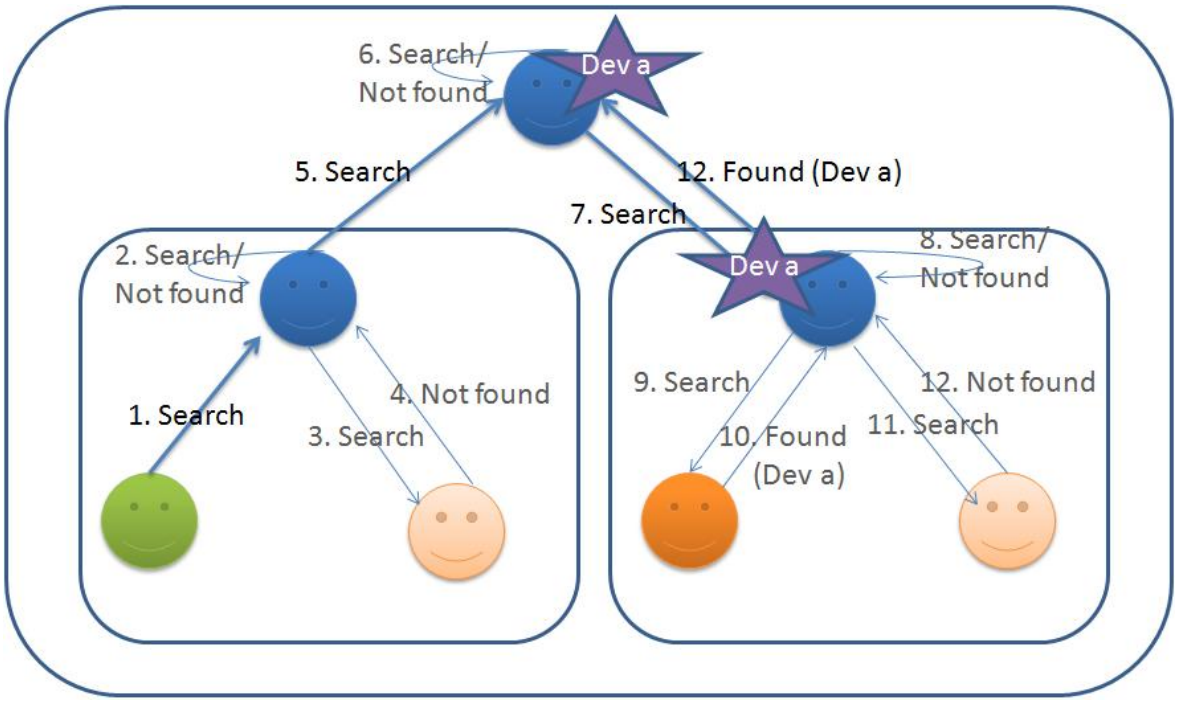
Search:



. Amal El Fallah Seghrouchni, Andrei Olaru, Thi Thuy Nga Nguyen and Diego Salomone
. PRIMA 2010
. Kolkata, India,

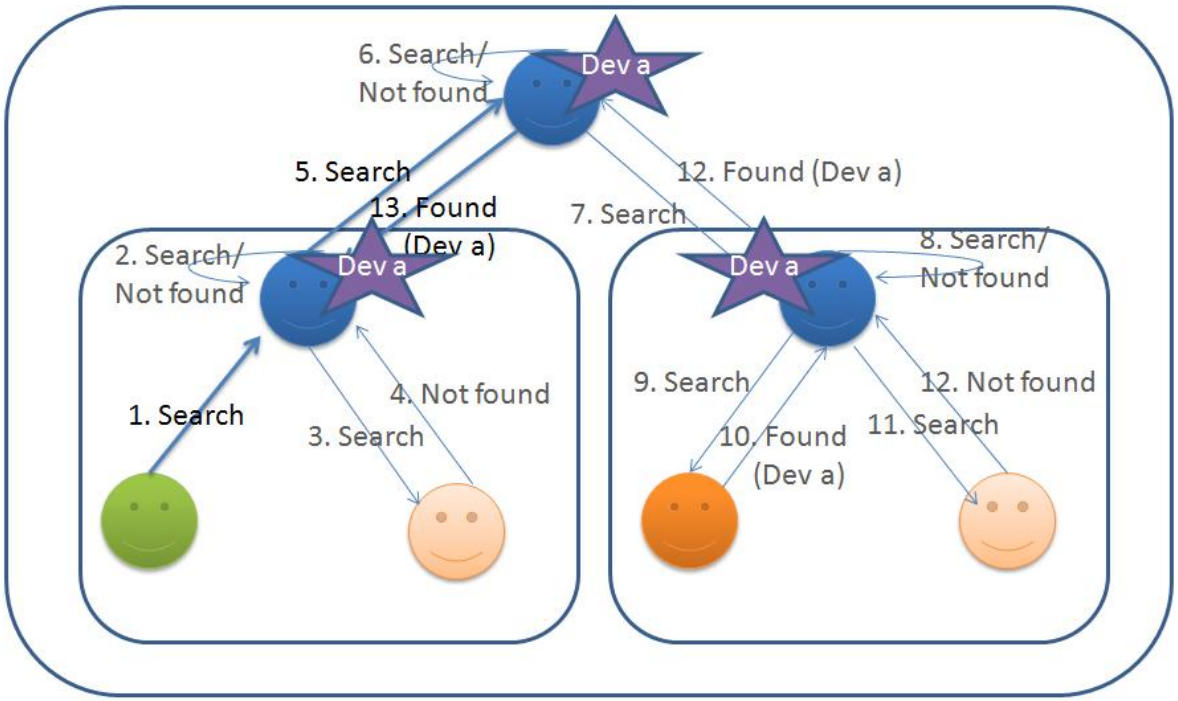
Examples:

Search:



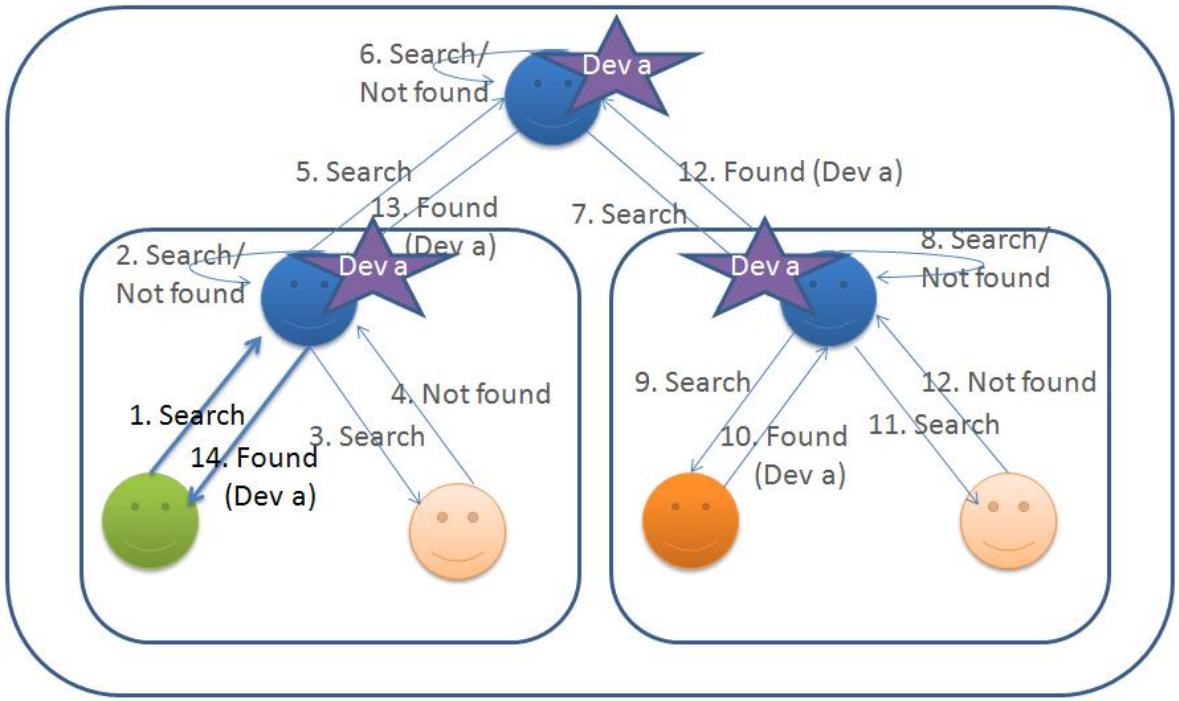
Examples:

Search:



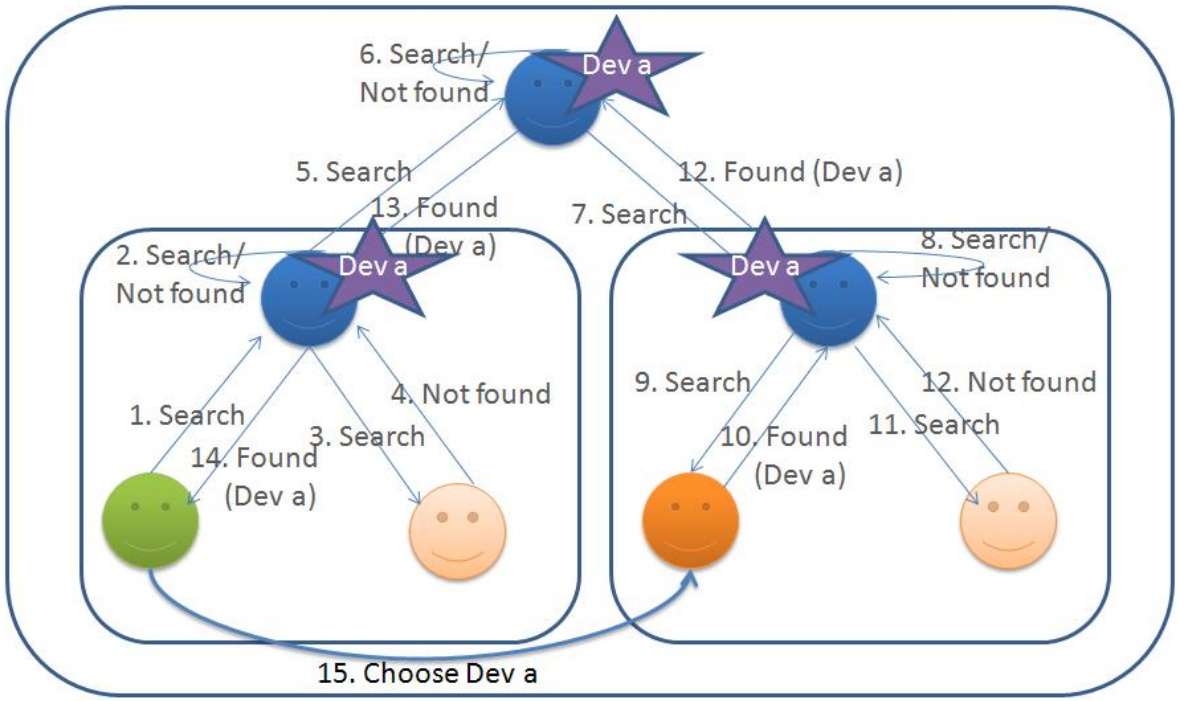
Examples:

Search:



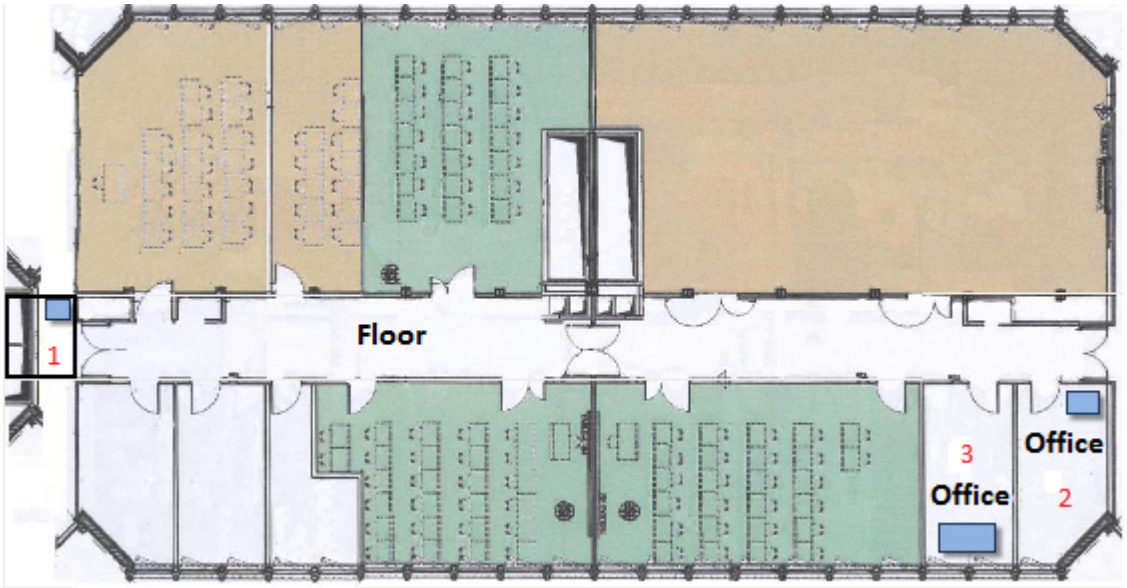
Examples:

Search:



· presented at the 5th NII-LIP6 Workshop, and developed by Thi Thuy Nga Nguyen, Diego Salomone-Bruno and Andrei Olaru, under the supervision of prof. Amal El Fallah Seghrouchni.

· presented at the 5th NII-LIP6 Workshop, and developed by Thi Thuy Nga Nguyen, Diego Salomone-Bruno and Andrei Olaru, under the supervision of prof. Amal El Fallah Seghrouchni.



· the Ao Dai project means implementing the idea of linking the two concepts of context and agent in a hierarchy.

· the project was implemented in CLAIM, that offers to developers an easy way to work with agents and hierarchies of agents, at a higher level.

· the demonstration showed how a simple scenario can be implemented, supporting context-aware actions that support the user.

· future work includes developing the features of agents, a better representation of context, and the extension of the types of context that are supported.

Chen, G. and Kotz, D. (2000).

A survey of context-aware mobile computing research.
Technical Report TR2000-381, Dartmouth College.

Dey, A. and Abowd, G. (2000).

Towards a better understanding of context and context-awareness.
CHI 2000 workshop on the what, who, where, when, and how of context-awareness, pages 304–307.

Ramos, C., Augusto, J., and Shapiro, D. (2008).

Ambient intelligence - the next step for artificial intelligence.
IEEE Intelligent Systems, 23(2):15–18.

Suna, A. and El Fallah Seghrouchni, A. (2004).

Programming mobile intelligent agents: An operational semantics.
Web Intelligence and Agent Systems, 5(1):47–67.

Weiser, M. (1993).

Some computer science issues in ubiquitous computing.
Communications - ACM, pages 74–87.

Thank You!