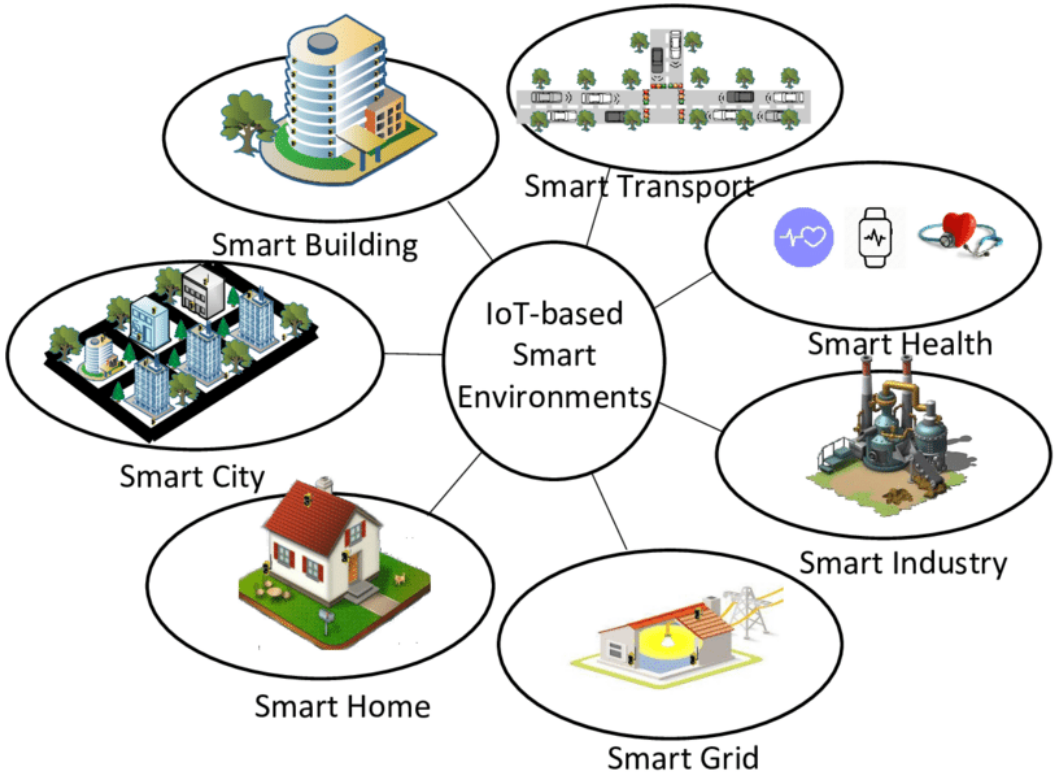
Declarative Goal Mediation in Smart Environments

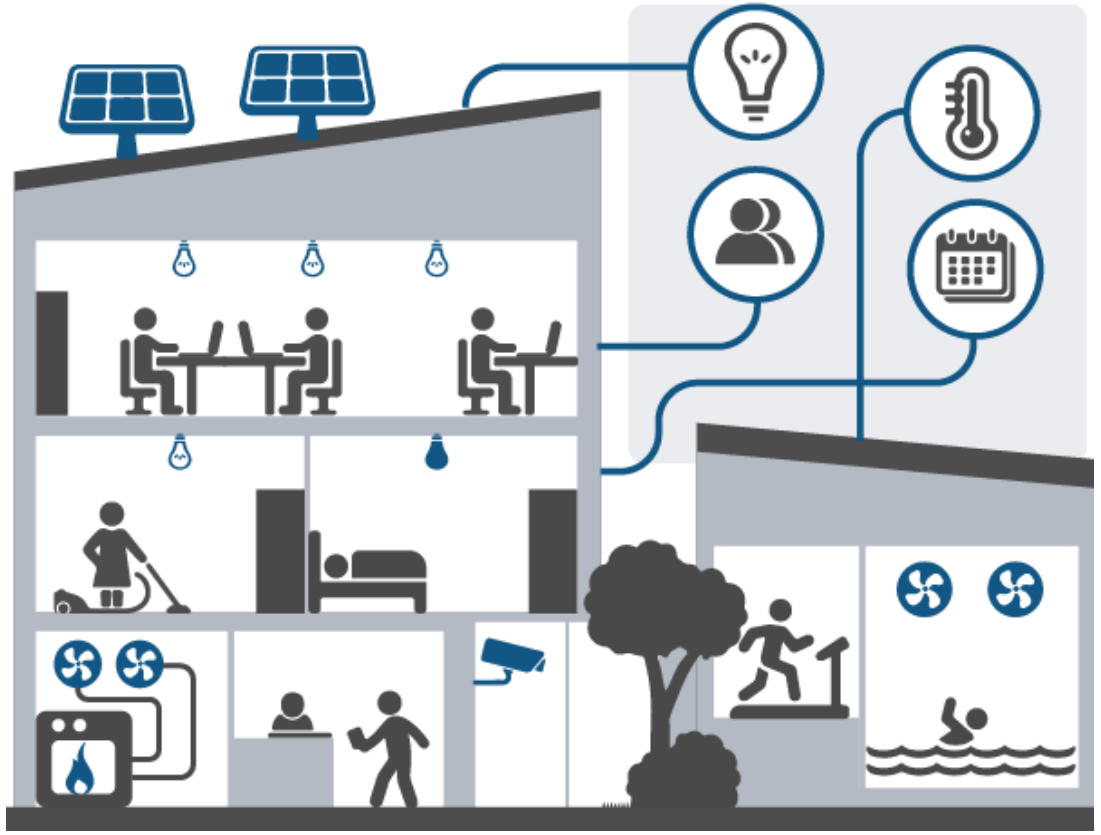
Giuseppe Bisicchia, Stefano Forti and Antonio Brogi

Service-oriented, Cloud and Fog Computing Research Group
Department of Computer Science
University of Pisa, Italy

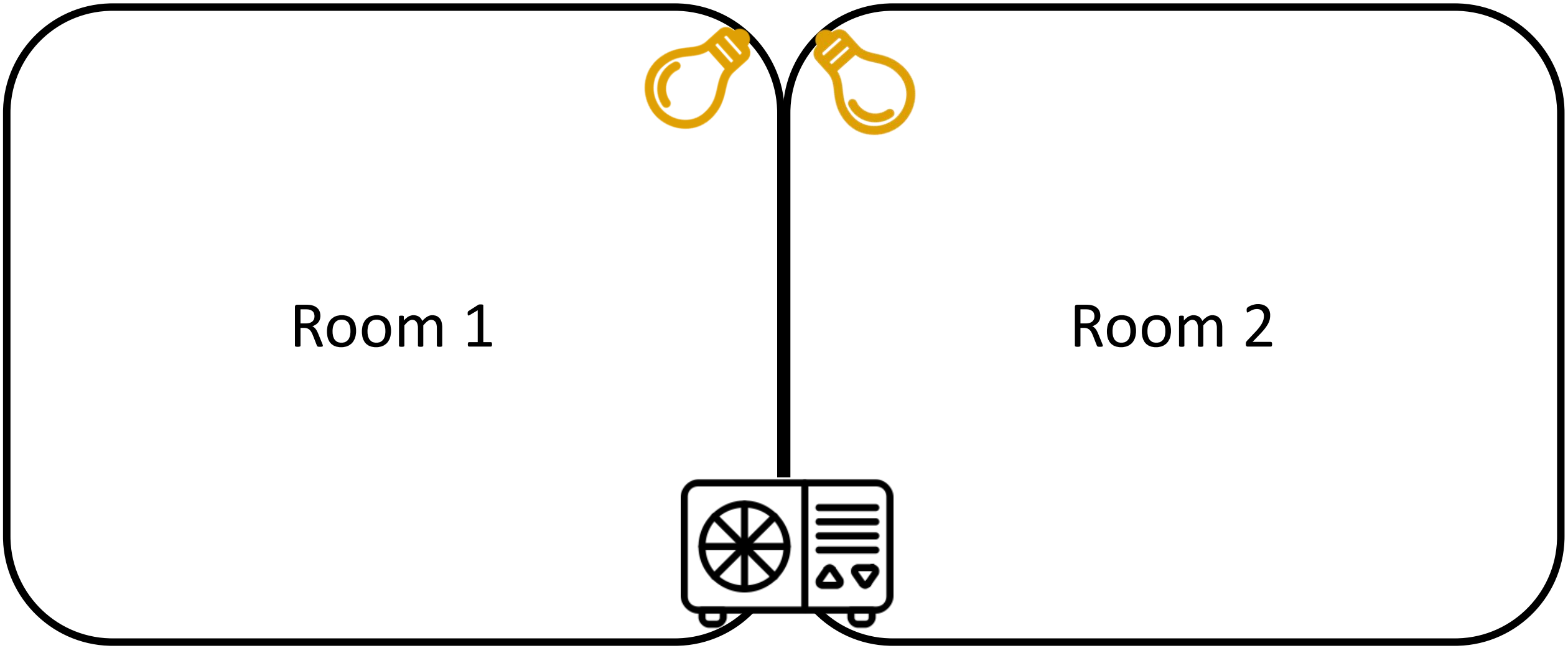
7th IEEE International Conference on Smart Computing (WiP track), 23-27 August 2021

Smart Environments

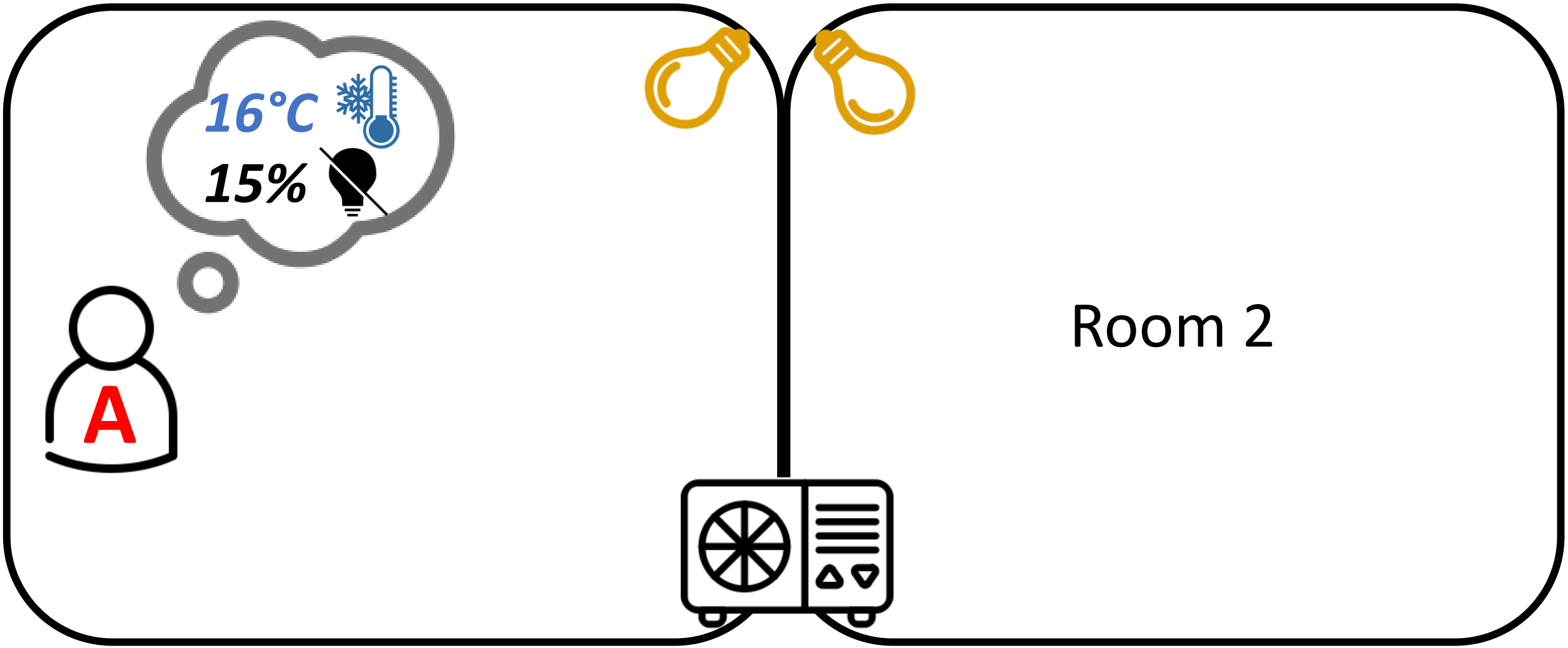




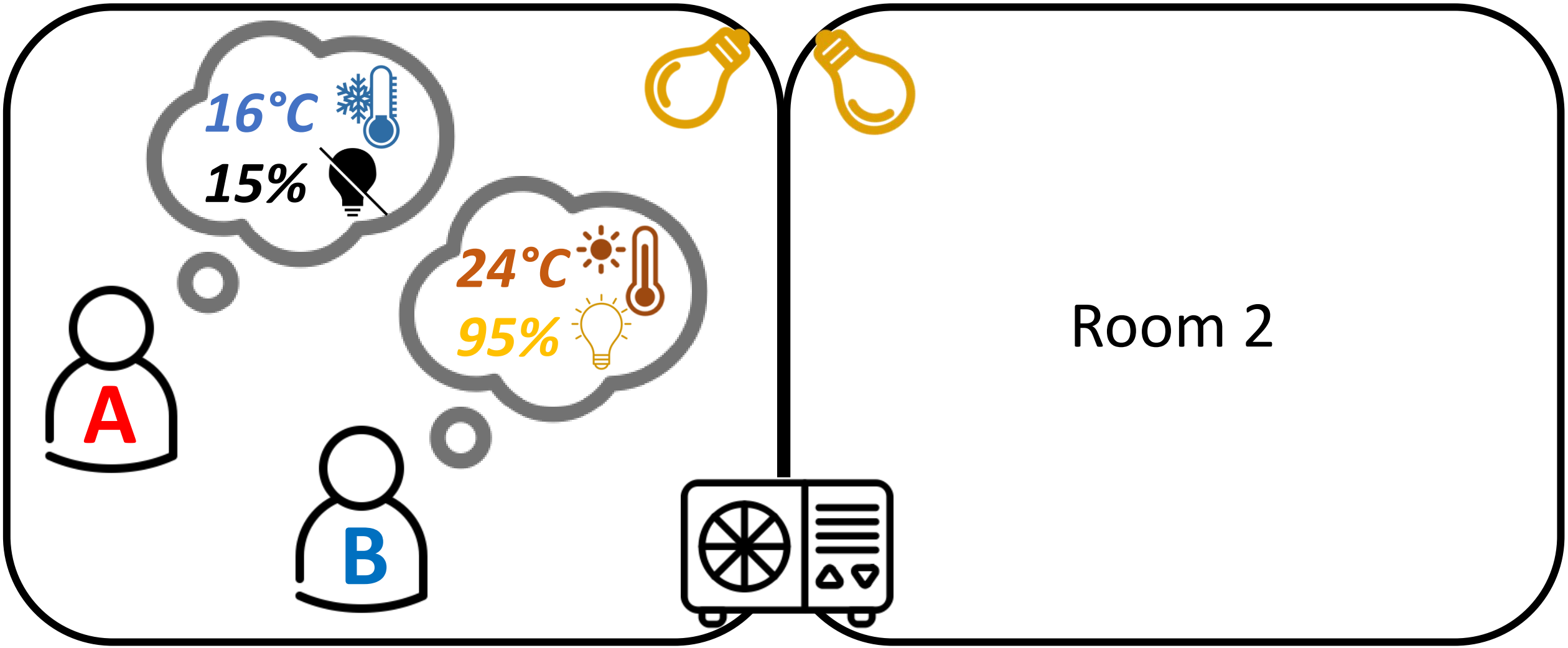
Motivating Scenario



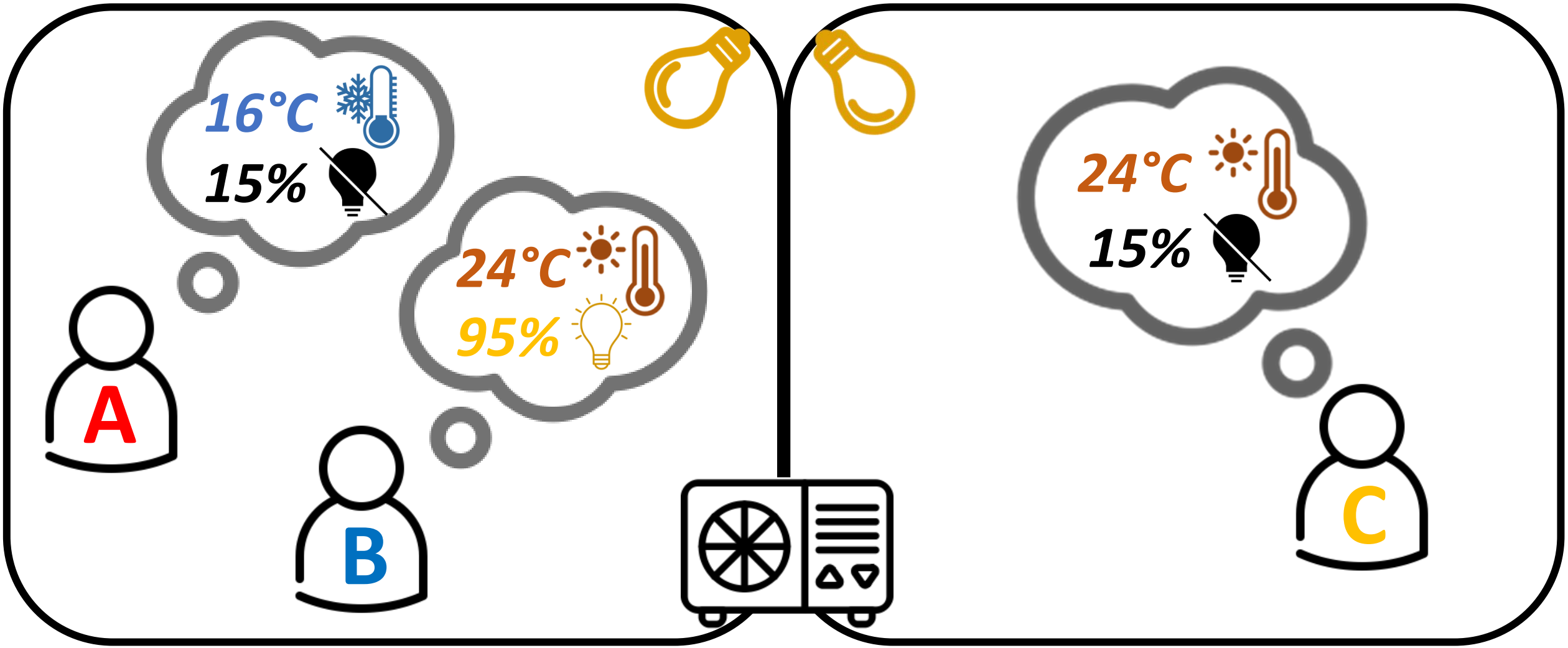
Motivating Scenario



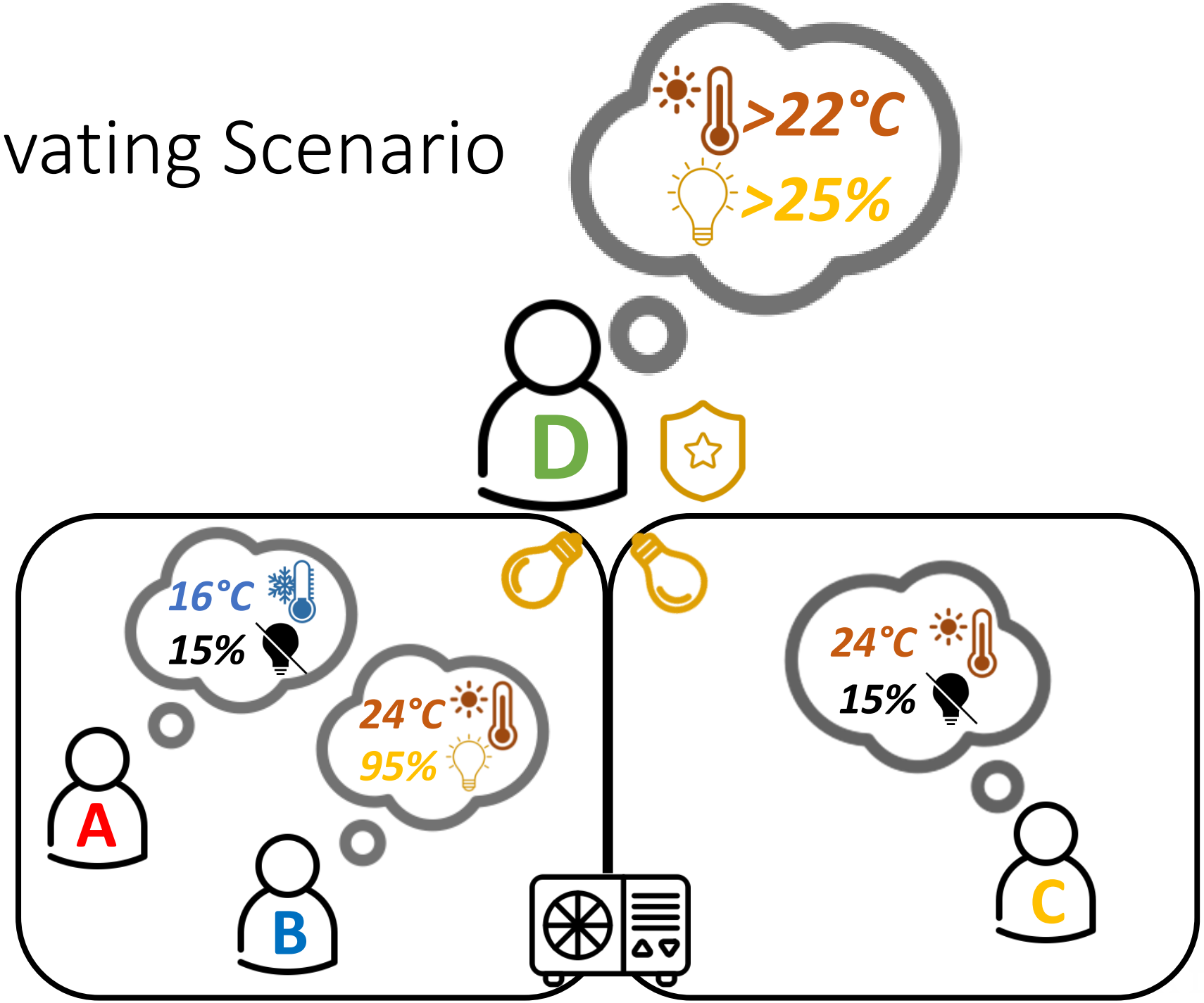
Motivating Scenario



Motivating Scenario



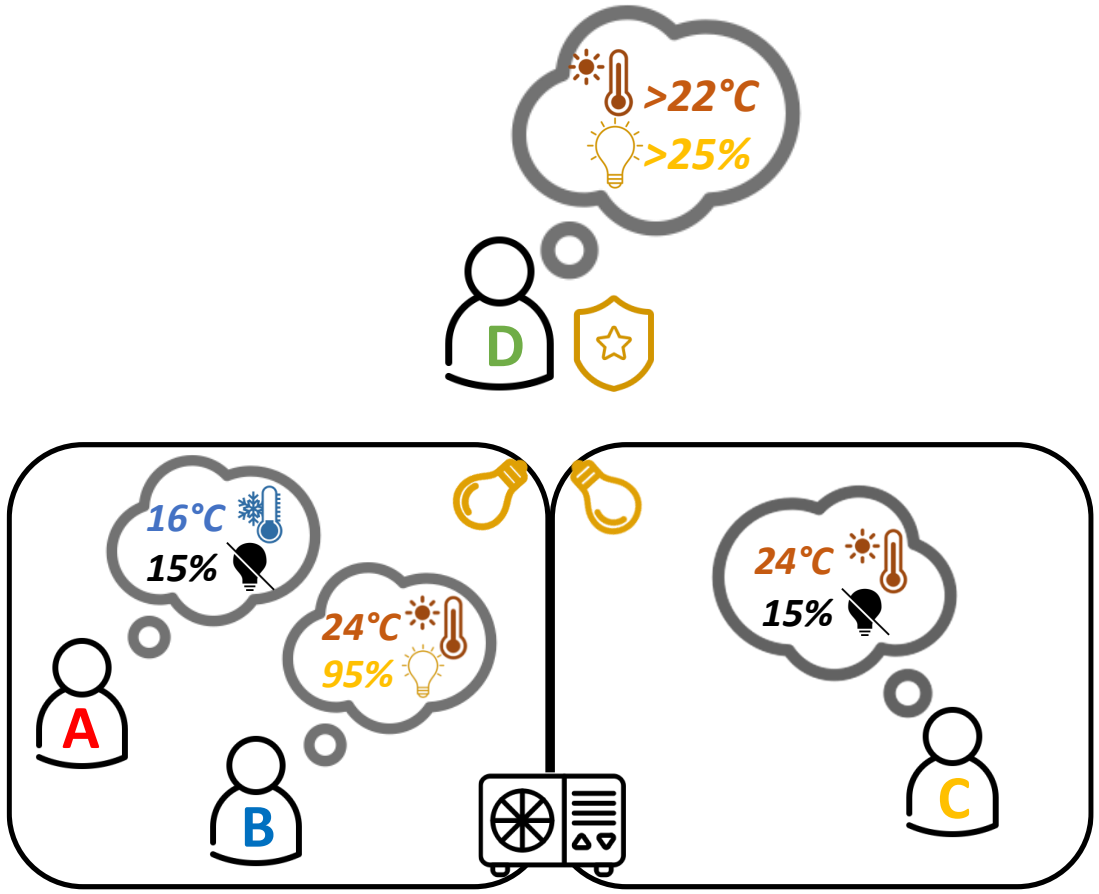
Motivating Scenario



Open Questions

Three types of **conflict** can arise:

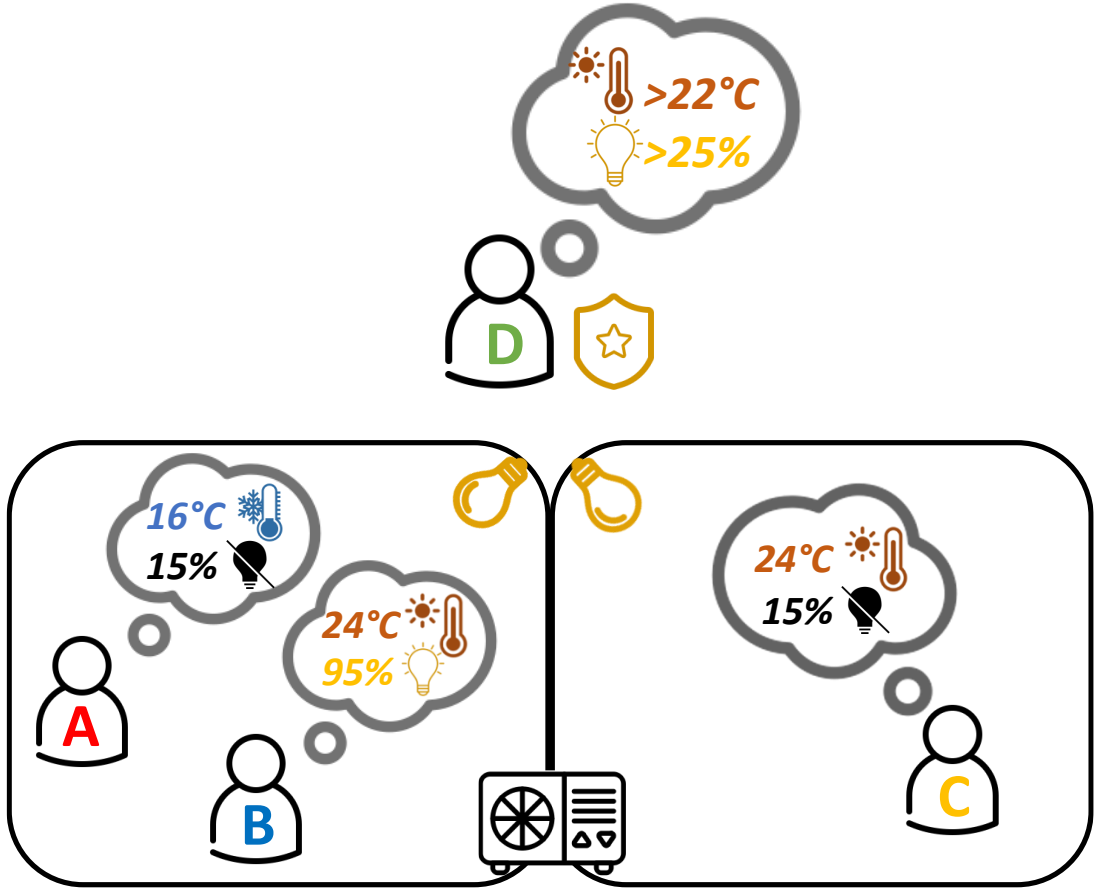
- **[User-User]** *How to **mediate all preferences** to satisfy them in the best possible way?*



Open Questions

Three types of **conflict** can arise:

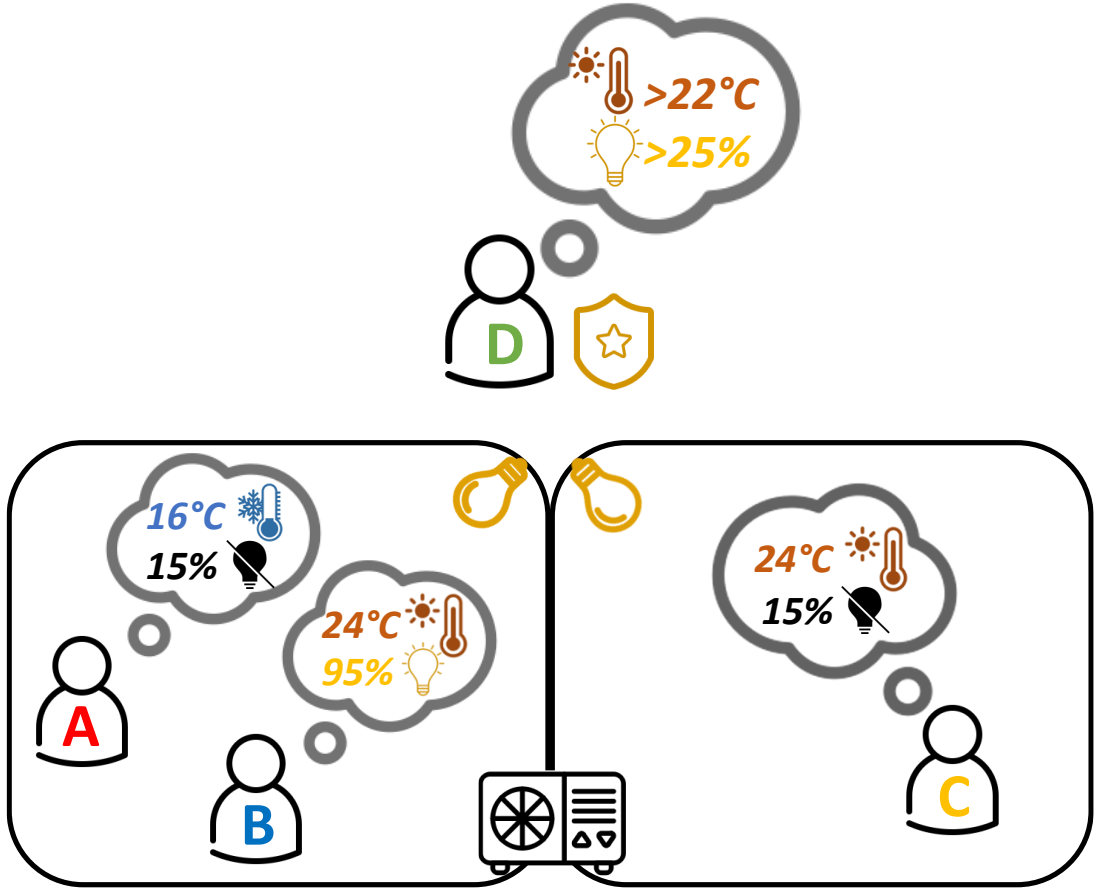
- **[User-User]** *How to **mediate all preferences** to satisfy them in the best possible way?*
- **[User-Admin]** *How to **achieve goals** set by the Sys. Admin. (e.g. energy savings)?*



Open Questions

Three types of **conflict** can arise:

- **[User-User]** *How to **mediate all preferences** to satisfy them in the best possible way?*
- **[User-Admin]** *How to **achieve goals** set by the Sys. Admin. (e.g. energy savings)?*
- **[IoT-IoT]** *How to **reach a mediated target state** by suitably settings the available actuators?*



Our Proposal

A **declarative framework** -- and its prototype **Solomon** -- to specify **customisable mediation policies** for reconciling **contrasting goals** and actuator settings in **smart environments**.

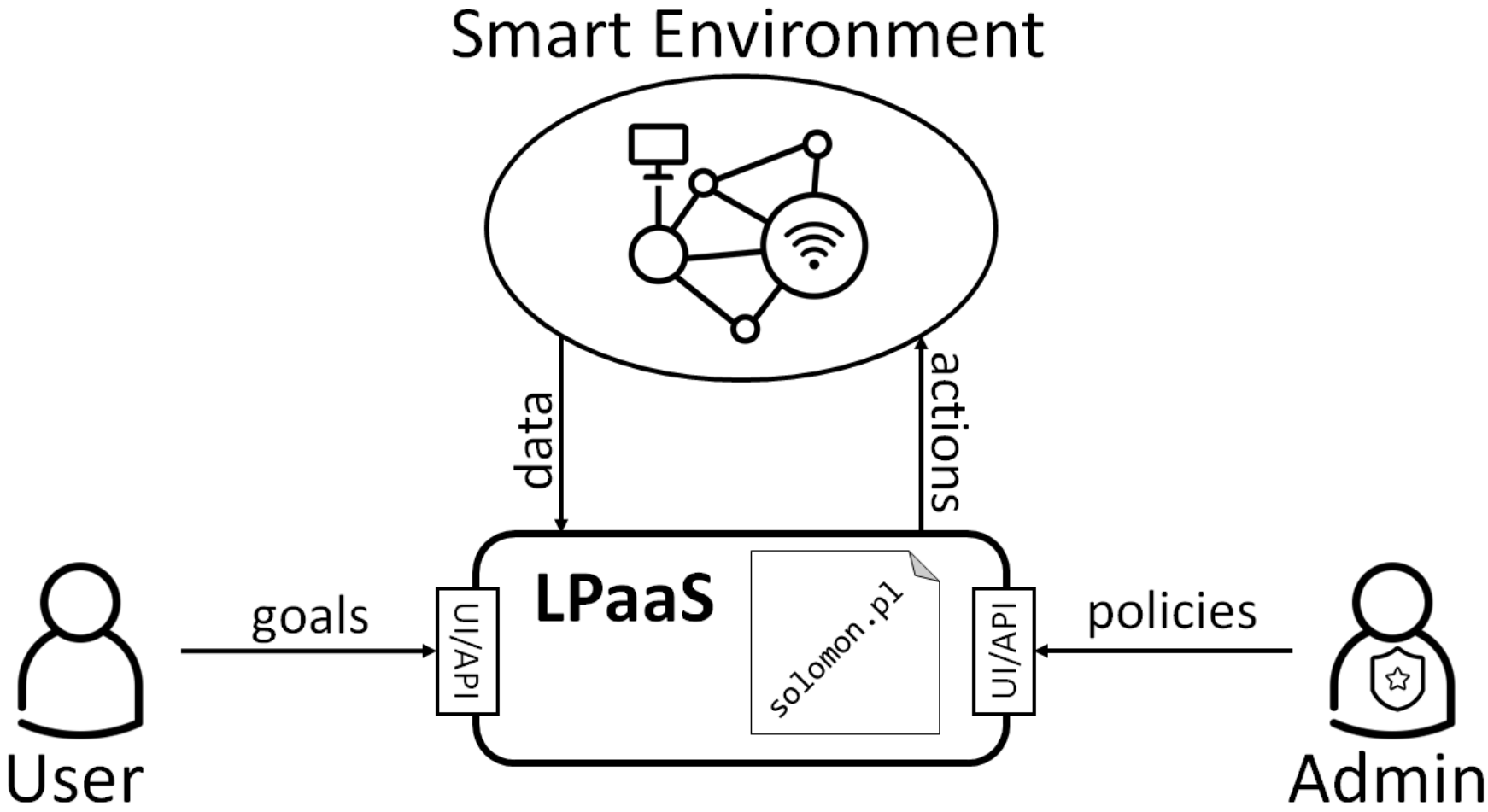
Our Proposal

A **declarative framework** -- and its prototype **Solomon** -- to specify **customisable mediation policies** for reconciling **contrasting goals** and actuator settings in **smart environments**.

How?

- By **reasoning on a model** of the available **IoT infrastructure** and on (possibly contrasting) goals.
- Specifying **ad-hoc mediation policies** for *User-User*, *User-Admin* and *IoT-IoT* conflicts.

Our Prototype



Our **declarative methodology** has been **prototyped in Prolog** and offered **as-a-service** through the **LPaaS paradigm**. The code is **open-source** and available at: https://github.com/di-unipi-socc/Solomon

Solomon Functioning

1. Collecting all user requests.

```
react(Requests, MediatedRequests, Actions) :-
    getRequests(Requests, ValidRequests),
```

Solomon Functioning

1. **Collecting all user requests.**
2. **Mediating the requests.**

```
react(Requests, MediatedRequests, Actions) :-
    getRequests(Requests, ValidRequests),
    mediateRequests(ValidRequests, MediatedRequests),
    validMediation(MediatedRequests),
```

Solomon Functioning

1. **Collecting all user requests.**
2. **Mediating the requests.**
3. **Determining actions** for **individual IoT actuators.**

```
react(Requests, MediatedRequests, Actions) :-
    getRequests(Requests, ValidRequests),
    mediateRequests(ValidRequests, MediatedRequests),
    validMediation(MediatedRequests),
    associateActions(MediatedRequests, Actions),
    validActions(Actions).
```

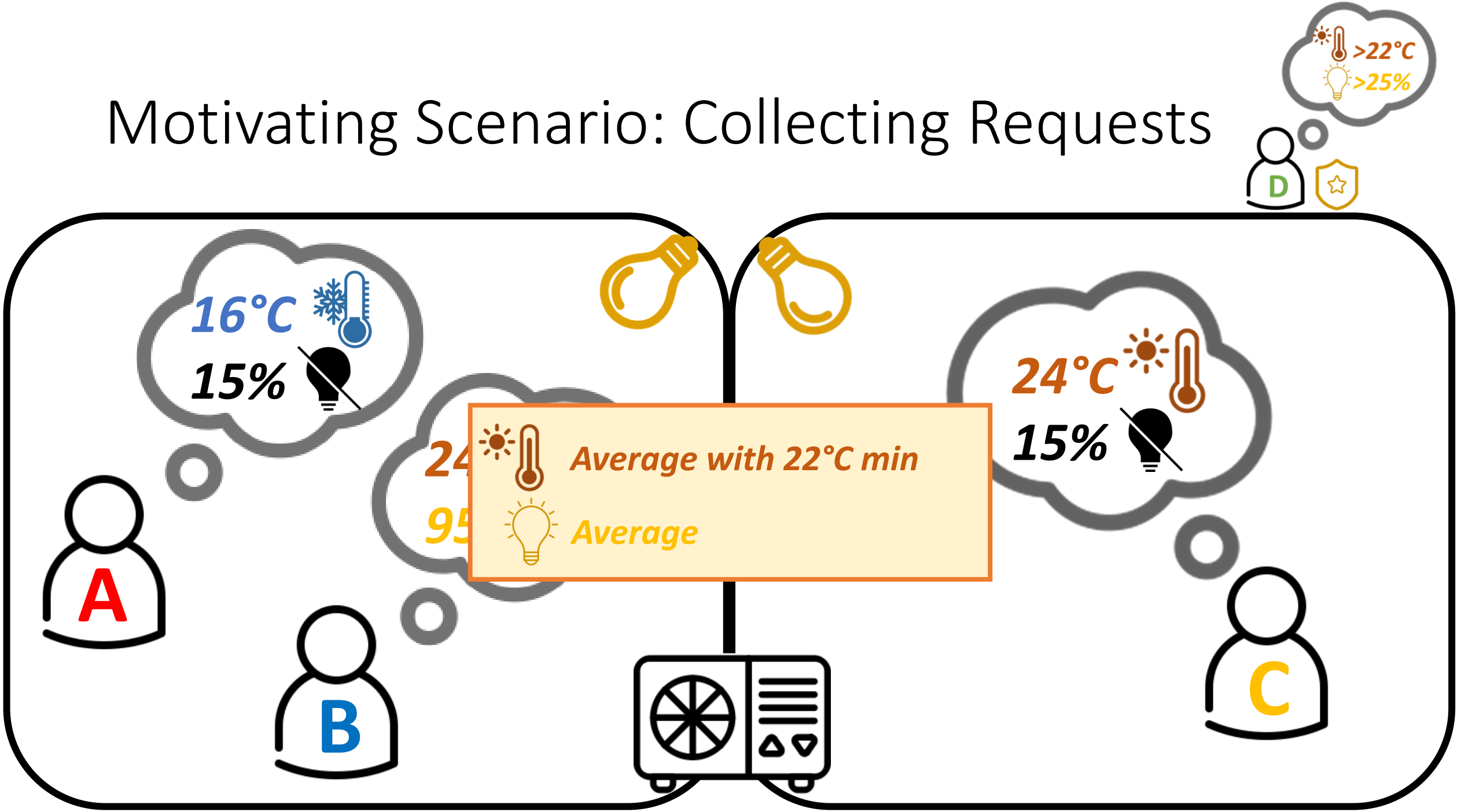
Solomon Functioning

1. **Collecting all user requests.**
2. **Mediating the requests.**
3. **Determining actions** for **individual IoT actuators.**

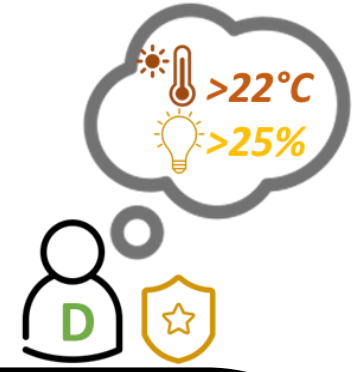
```
react(Requests, MediatedRequests, Actions) :-
    getRequests(Requests, ValidRequests),
    mediateRequests(ValidRequests, MediatedRequests),   % Defined by the Sys. Admin
    validMediation(MediatedRequests),
    associateActions(MediatedRequests, Actions),        % Defined by the Sys. Admin
    validActions(Actions).
```

*Solomon also offers a **library of standard predicates** to implement **mediation policies** (e.g. average, consensus, min/max).*

Motivating Scenario: Collecting Requests



Motivating Scenario: Mediating Requests



22°C

55%

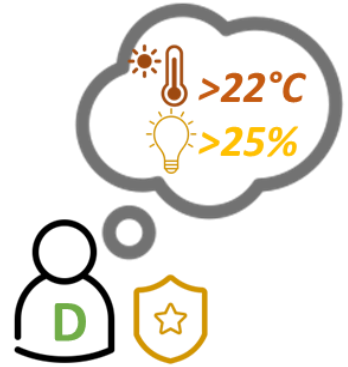
Average with 22°C min

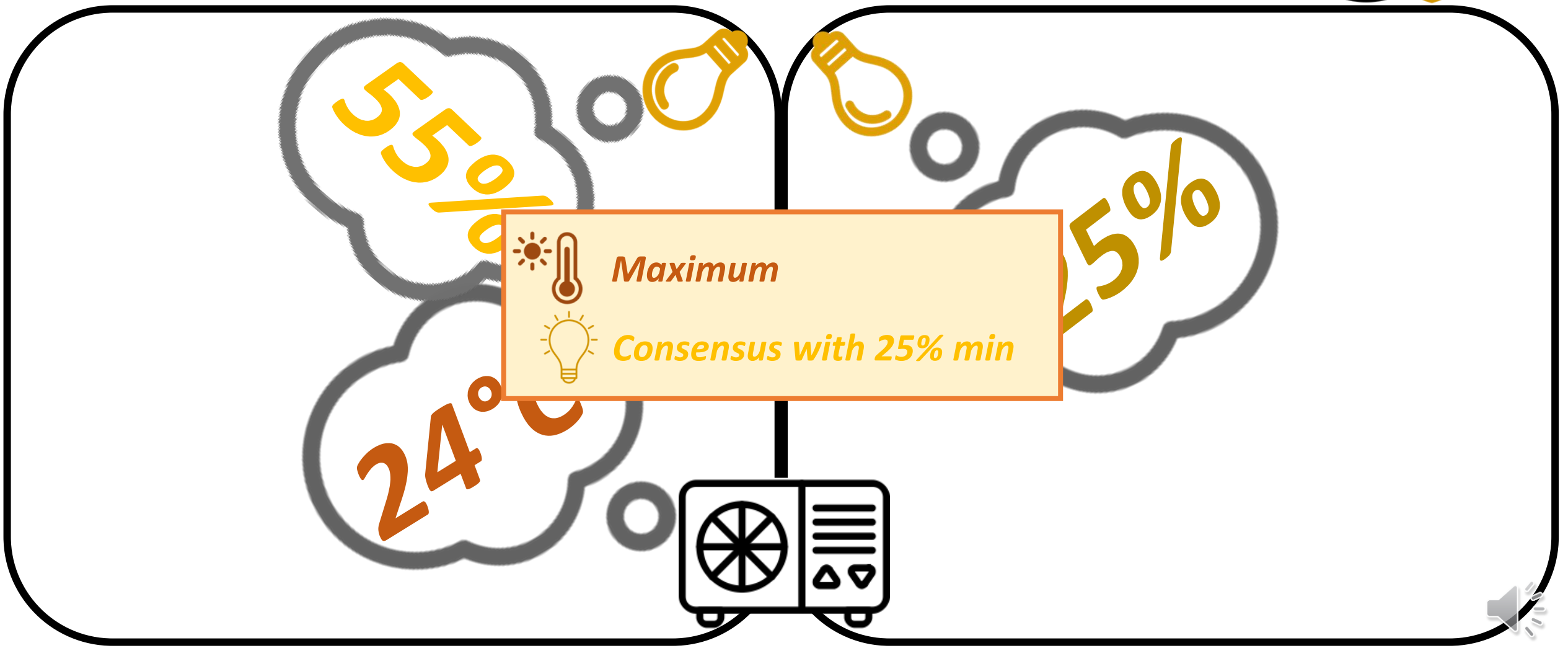
Average

24°C

15%

Motivating Scenario: Determing Actions





Conclusions

goal-driven

It **considers** and **mediates** among them **goals**, from all (human and machine) **stakeholders** involved in a Smart Environment, to reach a **target state**.

customisable

Being **open-source** and **enabling customisation** from its end-users. **Code** and **Docs** at: https://github.com/di-unipi-socc/Solomon

As it is **Prolog** code: **concise** (around **50 sloc**) and featuring a good level of **abstraction** and **flexibility** to **accommodate new emerging needs of Smart Environments**.

declarative

As it features a **well-defined REST API** based on **LPaaS**, it enables **interoperability** with other systems through **remote interactions**.

as-a-Service

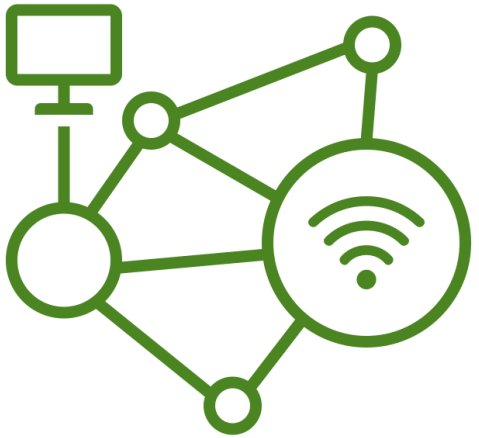
Future Work

New Policies

to propose a set of modular policies to the end-users

Goal Geolocalisation

to predict users' movements so to reduce manual interactions



Web of Things

to exploit Solomon in actual smart environments

Declarative Goal Mediation in Smart Environments

Giuseppe Bisicchia, Stefano Forti and Antonio Brogi

Service-oriented, Cloud and Fog Computing Research Group
Department of Computer Science
University of Pisa, Italy

7th IEEE International Conference on Smart Computing (WiP track), 23-27 August 2021