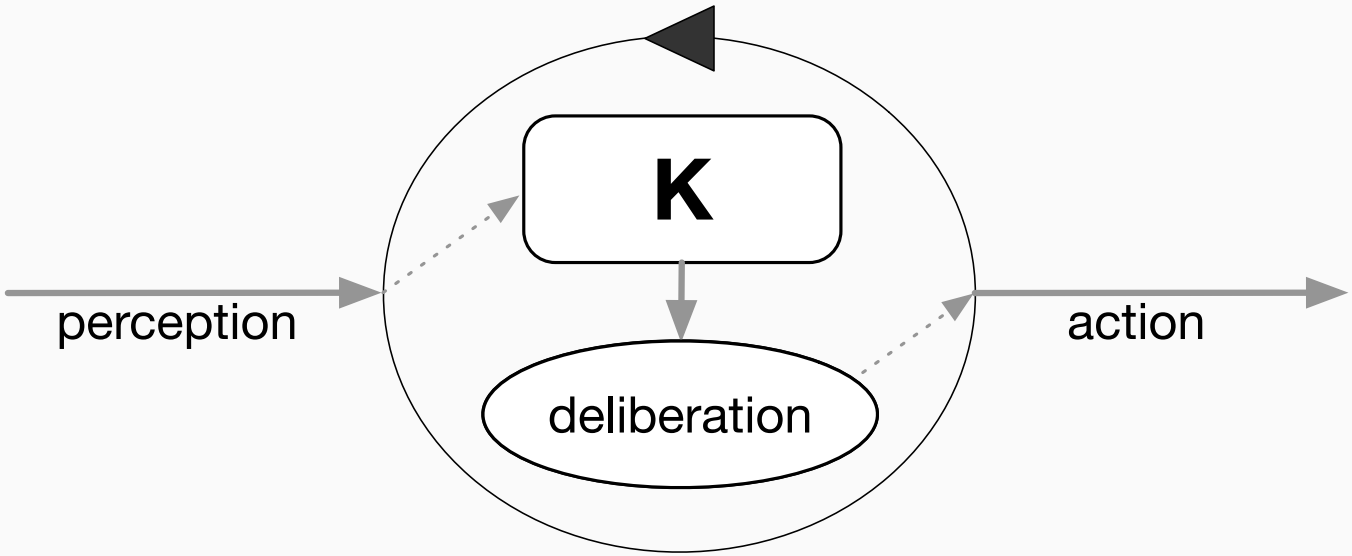
Agent Dimension

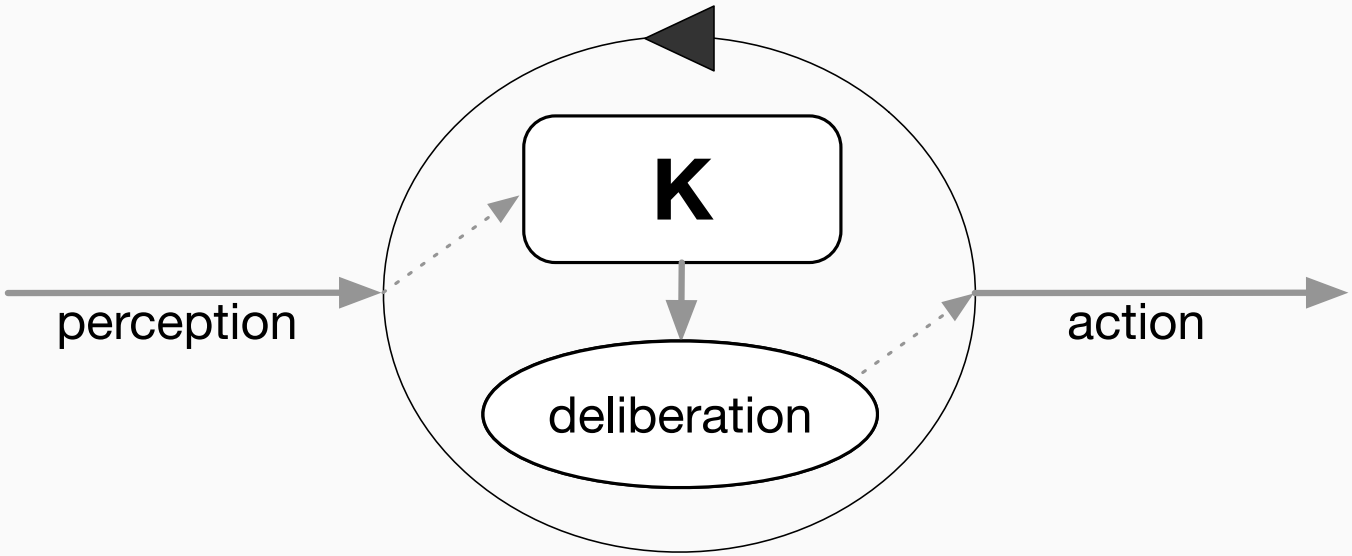
***Credits**: Slides prepared by Jomi F. Hübner*

Agent

Agent



Agent



[reasoning cycle]

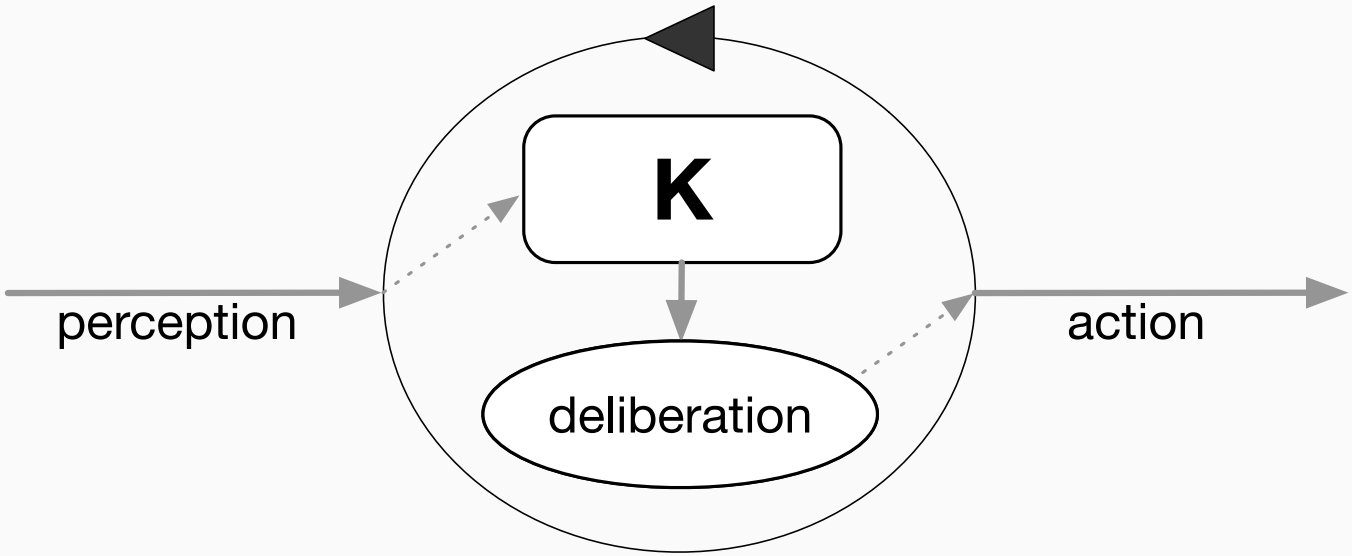
while true do

$K \leftarrow K \pm perception()$

$A \leftarrow deliberation(K)$

$act(A)$

Agent



to **program** an agent is to define K

deliberation ⇝ **autonomy**

Agent Knowledge (in JaCaMo)

Beliefs : information about the environment, other agents, itself, application, ...

```
temperature(20).
happy(bob).
```

Goals : the agent objectives

```
!temperature(20).
!happy(bob).
```

Plans :

Agent Knowledge (in JaCaMo)

Beliefs : information about the environment, other agents, itself, application, ...

```
temperature(20).
happy(bob).
```

Goals : the agent objectives

```
!temperature(20).
!happy(bob).
```

Plans :

Agent Knowledge (in JaCaMo)

Beliefs : information about the environment, other agents, itself, application, ...

```
temperature(20).
happy(bob).
```

Goals : the agent objectives

```
!temperature(20).
!happy(bob).
```

Plans : specifies how goals can be **achieved** by **actions**

```
+!temperature(20) <- startCooling.
+!happy(bob) <- kiss(bob).
```

Agent Knowledge (in JaCaMo)

Beliefs : information about the environment, other agents, itself, application, ...

```
temperature(20).
happy(bob).
```

Goals : the agent objectives

```
!temperature(20).
!happy(bob).
```

Plans : specifies how goals can be **achieved** by **actions**

```
+!temperature(20) <- startCooling.
+!happy(bob) <- kiss(bob).
```

specifies **reactions** to mental state changes

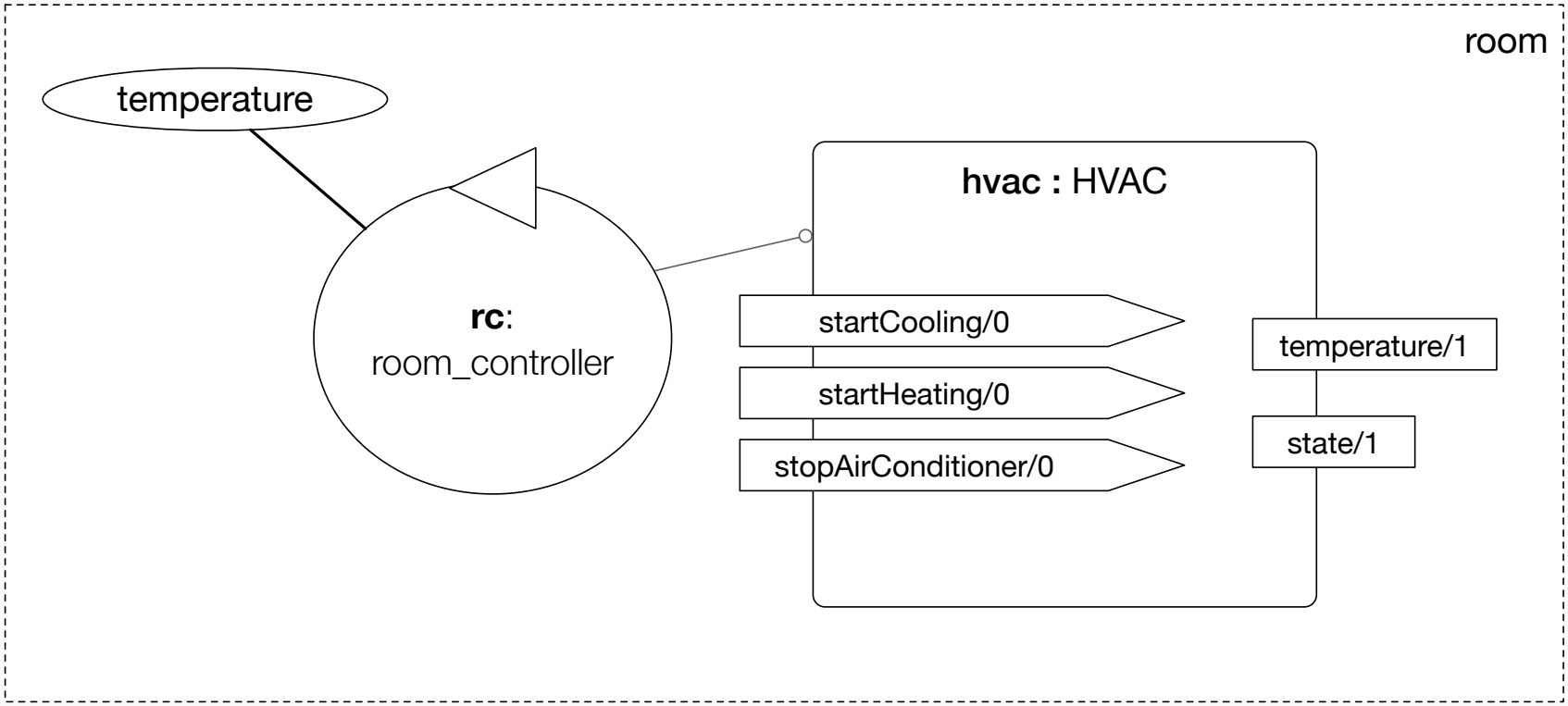
```
+temperature(10) <- !temperature(20).
-happy(bob) <- !happy(bob).
```

Knowledge Sources

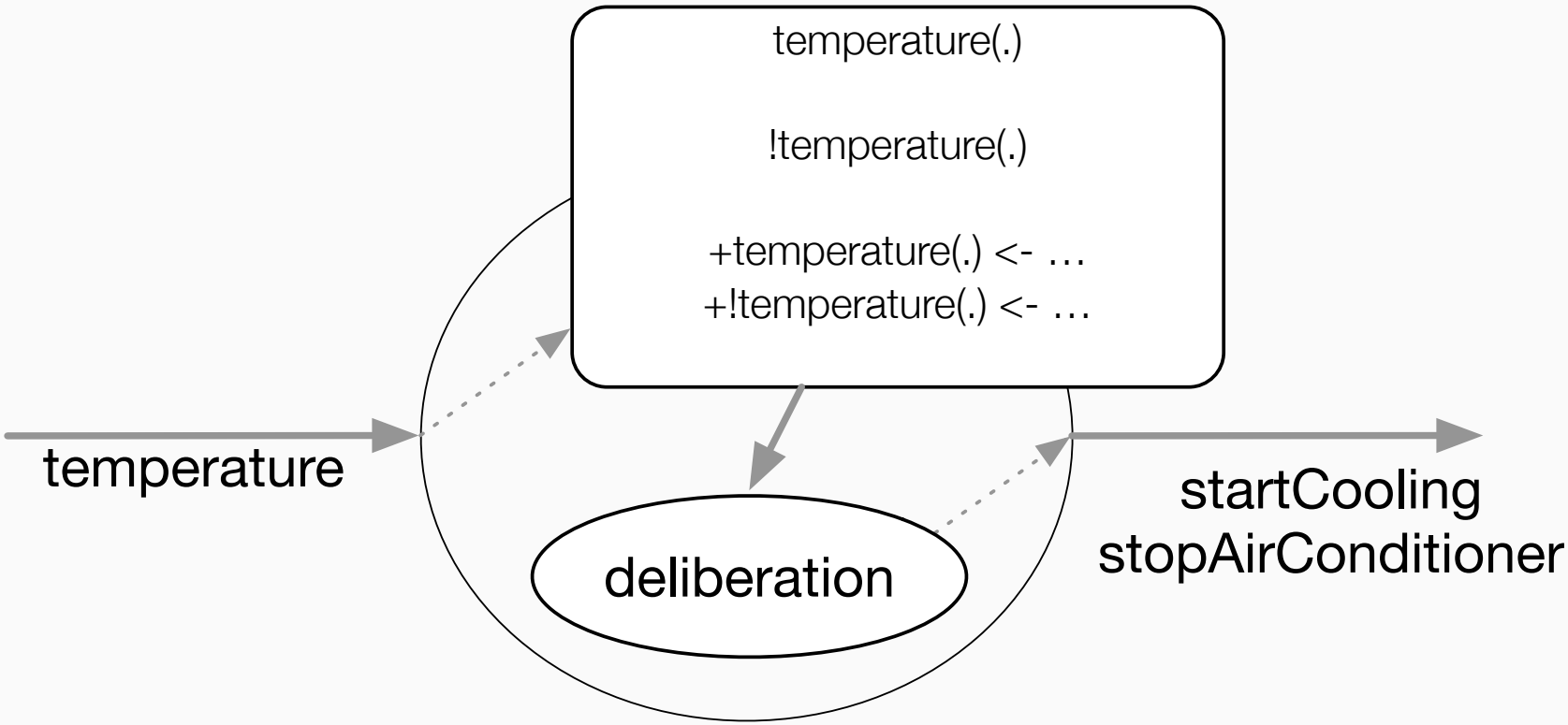
Beliefs, goals, and plans are provided by

- perception: in the case of beliefs
- developers: initial mental state of the agent
- other agents: by communication
- the agent itself: by reasoning or learning

Smart Room Scenario — initial implementation



Agent Programming (in JaCaMo)



Agent Programming (in JaCaMo)

```
+temperature(30)  <- !temperature(20).
+!temperature(20) <- startCooling.
```

Agent Programming (in JaCaMo)

```
+temperature(30)  <- !temperature(20).
+temperature(20)  <- stopAirConditioner.

+!temperature(20) <- startCooling.
```

Agent Programming (in JaCaMo)

```
// initial belief, given by the developer
preference(20).

// reaction to changes in the temperature
+temperature(T)  : preference(P) & math.abs(P-T) > 2
   <- !temperature(P).
+temperature(T)  : preference(T)
   <- stopAirConditioner.

// plans to achieve some temperature
+!temperature(P) : temperature(T) & T >  P
   <- startCooling.
```

Agent Programming (in JaCaMo)

```
// initial belief, given by the developer
preference(20).

// initial goal, given by the developer
!keep_temperature.

// maintenance the goal pattern
+!keep_temperature
    : temperature(T) & preference(P) & T >  P
   <- startCooling;
      !keep_temperature.
+!keep_temperature
    : temperature(T) & preference(P) & T <= P
   <- stopAirConditioner;
      !keep_temperature.
```

Main Features

- **reactivity**: even when achieving some goals
- **pro-activity**: new goals can be created
- **long-term goals**: agents are committed to achieve goals
- **context awareness**: plans are selected based on the circumstances
- **transparency**: we can trace back the reasons for an action
- sound **theoretical background** for agent architectures:
  - practical reasoning [Bratman, 1987]
  - intentions [Cohen and Levesque, 1987]
  - BDI [Rao and Georgeff, 1995]
  - ...

Main Features

- **reactivity**: even when achieving some goals
- **pro-activity**: new goals can be created
- **long-term goals**: agents are committed to achieve goals
- **context awareness**: plans are selected based on the circumstances
- **transparency**: we can trace back the reasons for an action
- sound **theoretical background** for agent architectures:
  - practical reasoning [Bratman, 1987]
  - intentions [Cohen and Levesque, 1987]
  - BDI [Rao and Georgeff, 1995]
  - ...

Main Features

- **reactivity**: even when achieving some goals
- **pro-activity**: new goals can be created
- **long-term goals**: agents are committed to achieve goals
- **context awareness**: plans are selected based on the circumstances
- **transparency**: we can trace back the reasons for an action
- sound **theoretical background** for agent architectures:
  - practical reasoning [Bratman, 1987]
  - intentions [Cohen and Levesque, 1987]
  - BDI [Rao and Georgeff, 1995]
  - ...

Main Features

- **reactivity**: even when achieving some goals
- **pro-activity**: new goals can be created
- **long-term goals**: agents are committed to achieve goals
- **context awareness**: plans are selected based on the circumstances
- **transparency**: we can trace back the reasons for an action
- sound **theoretical background** for agent architectures:
  - practical reasoning [Bratman, 1987]
  - intentions [Cohen and Levesque, 1987]
  - BDI [Rao and Georgeff, 1995]
  - ...

Main Features

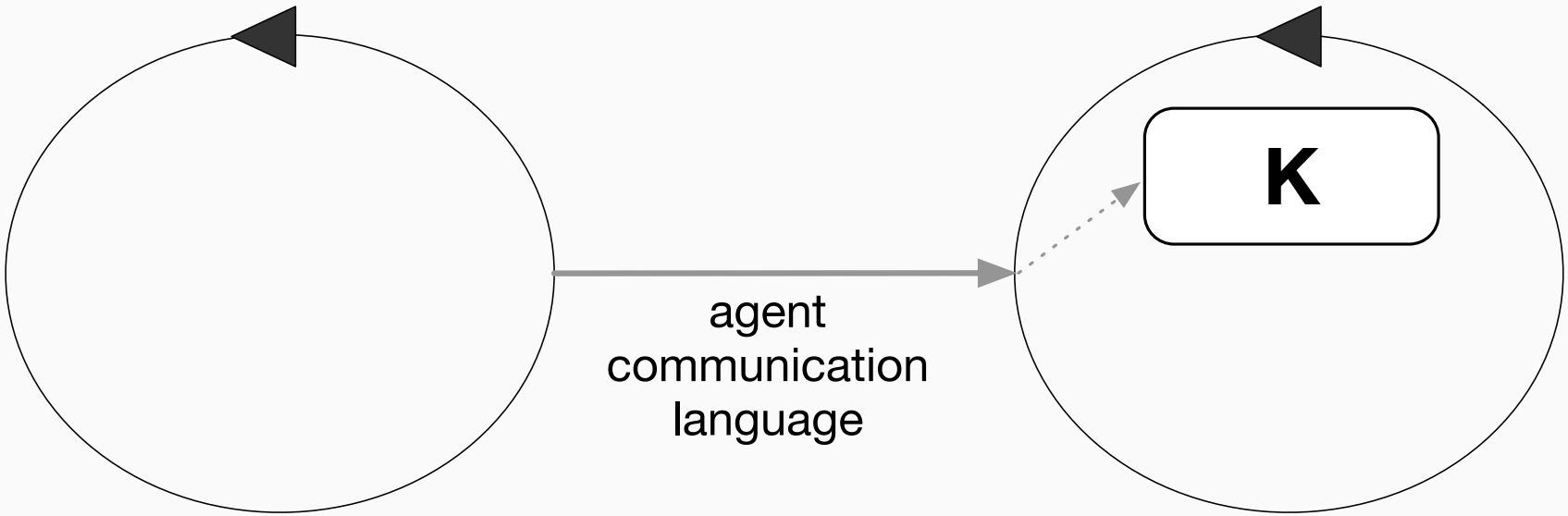
- **reactivity**: even when achieving some goals
- **pro-activity**: new goals can be created
- **long-term goals**: agents are committed to achieve goals
- **context awareness**: plans are selected based on the circumstances
- **transparency**: we can trace back the reasons for an action
- sound **theoretical background** for agent architectures:
  - practical reasoning [Bratman, 1987]
  - intentions [Cohen and Levesque, 1987]
  - BDI [Rao and Georgeff, 1995]
  - ...

Main Features

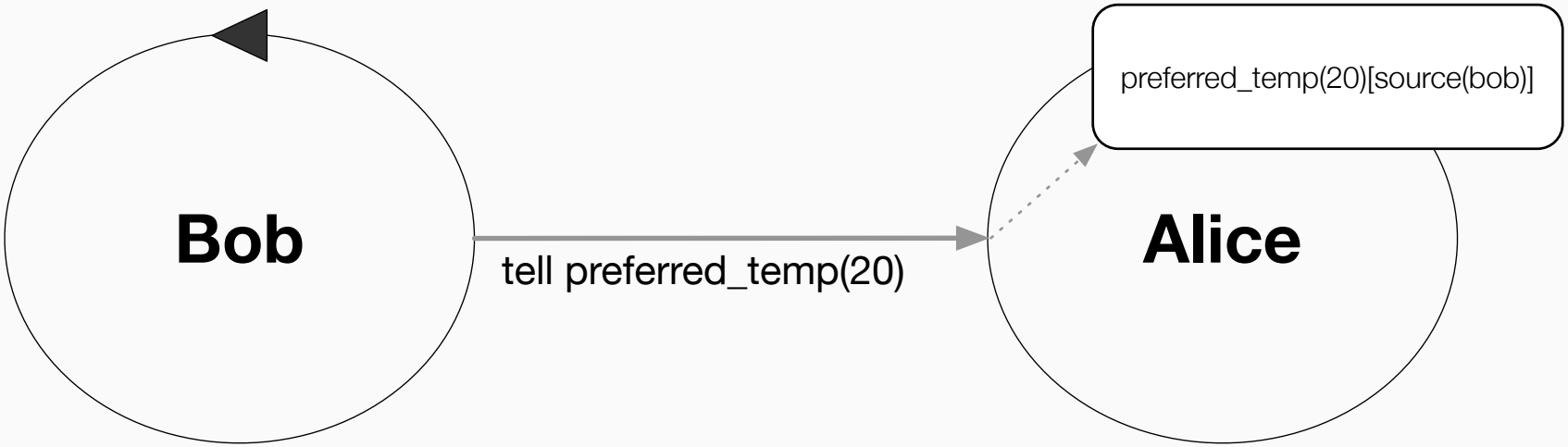
- **reactivity**: even when achieving some goals
- **pro-activity**: new goals can be created
- **long-term goals**: agents are committed to achieve goals
- **context awareness**: plans are selected based on the circumstances
- **transparency**: we can trace back the reasons for an action
- sound **theoretical background** for agent architectures:
  - practical reasoning [Bratman, 1987]
  - intentions [Cohen and Levesque, 1987]
  - BDI [Rao and Georgeff, 1995]
  - ...

Agent Interaction (communication)

Agent–Agent Communication



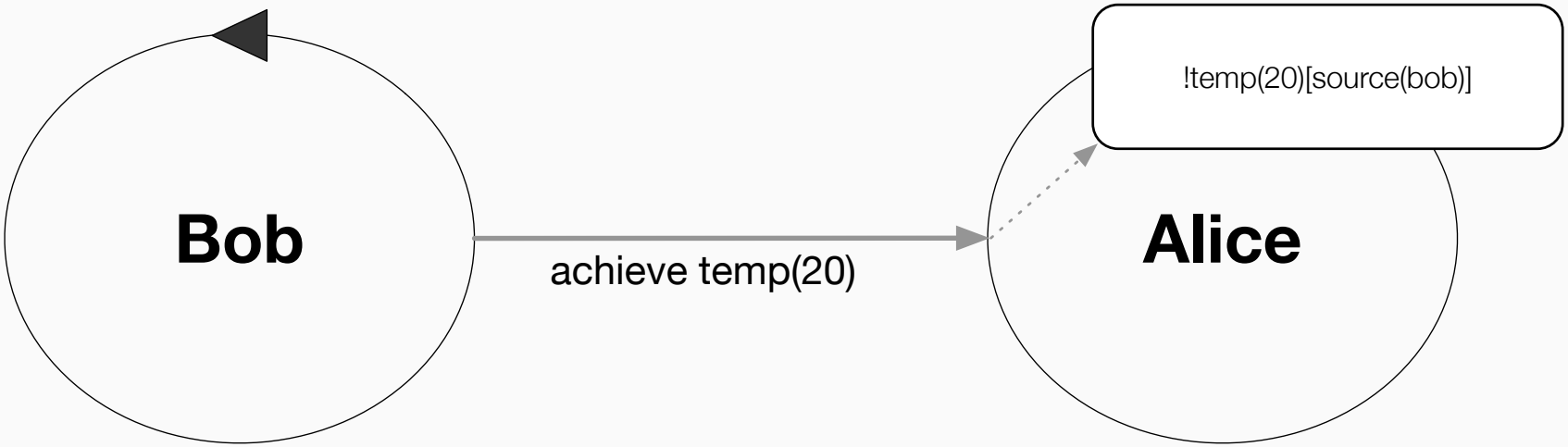
Semantic of messages



A message has:

- an intention (tell, ask, achieve, ...)
- a content (belief, goal, plan)

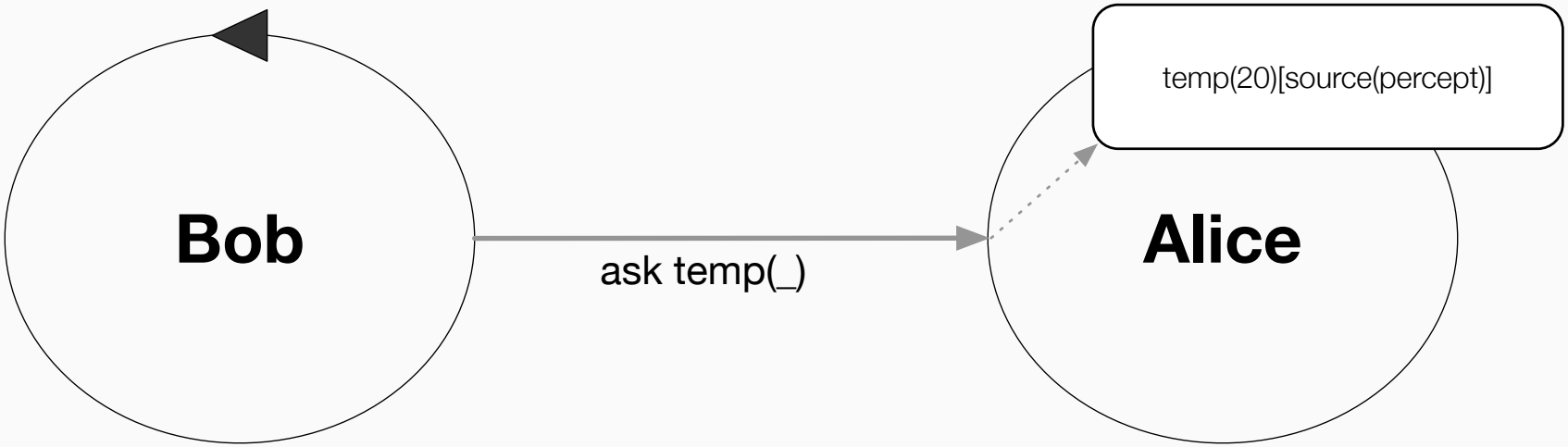
Semantic of messages



A message has:

- an intention (tell, ask, achieve, ...)
- a content (belief, goal, plan)

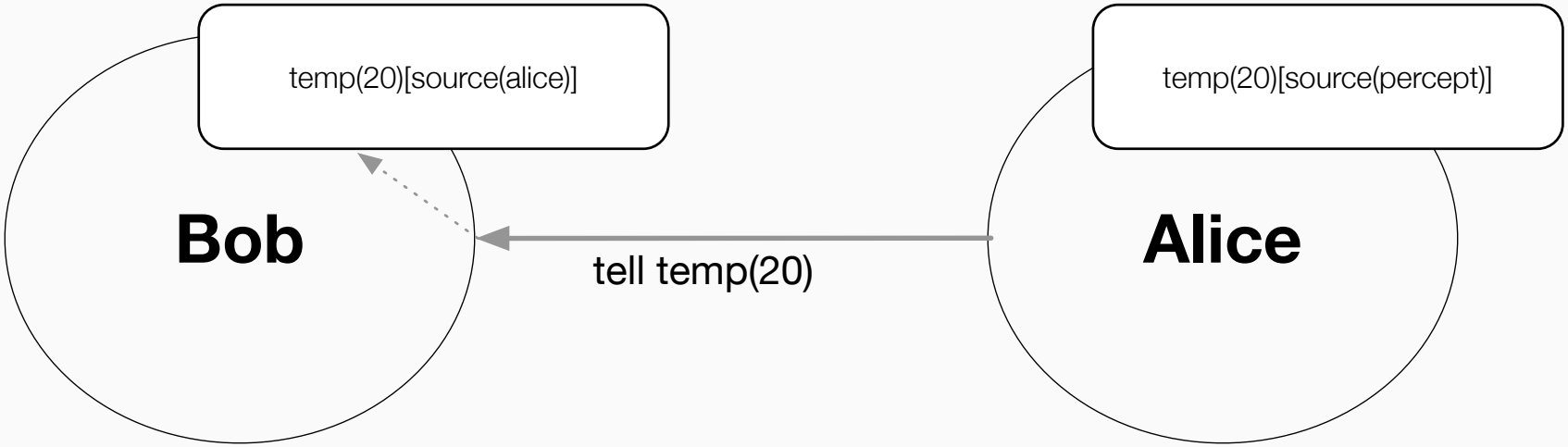
Semantic of messages



A message has:

- an intention (tell, ask, achieve, ...)
- a content (belief, goal, plan)

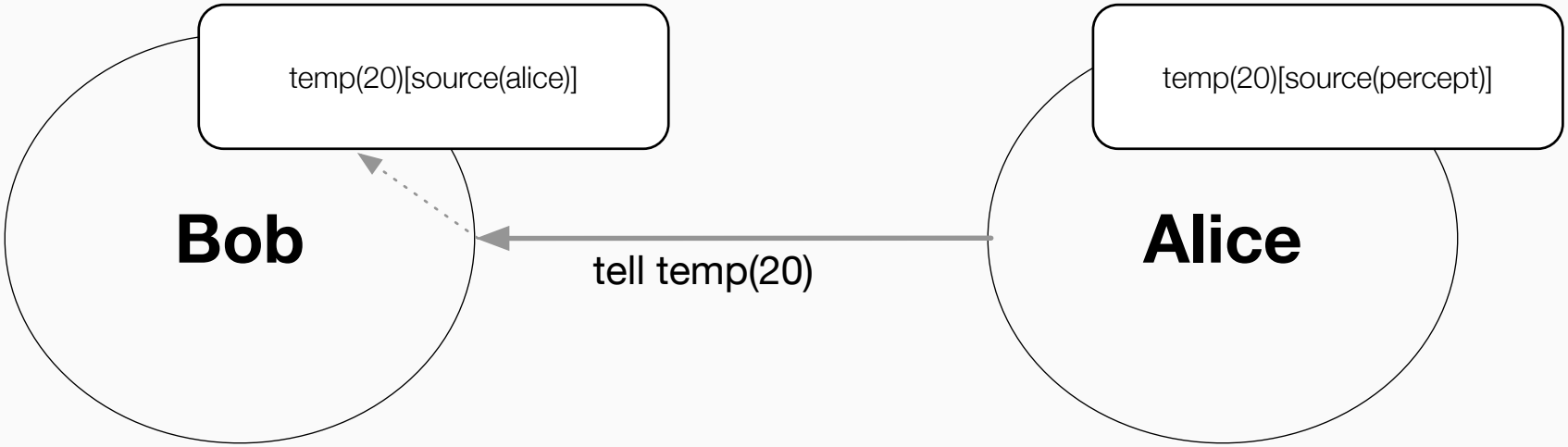
Semantic of messages



A message has:

- an intention (tell, ask, achieve, ...)
- a content (belief, goal, plan)

Semantic of messages



- we are not programming computers,
  we are programming agents, which are based on knowledge
- communication is not about data exchange, but
  knowledge sharing

JaCaMo implementation

Sender: `.send(bob,tell,happy(alice))`

- receiver: agent unique name
- performative: tell, achieve, askOne, askHow, ...
- content: a literal

Receiver

- nothing is needed

Properties

- distributed & support for decentralized
- (usually) asynchronous
- KQML vs FIPA-ACL
- not reduced to method invocation

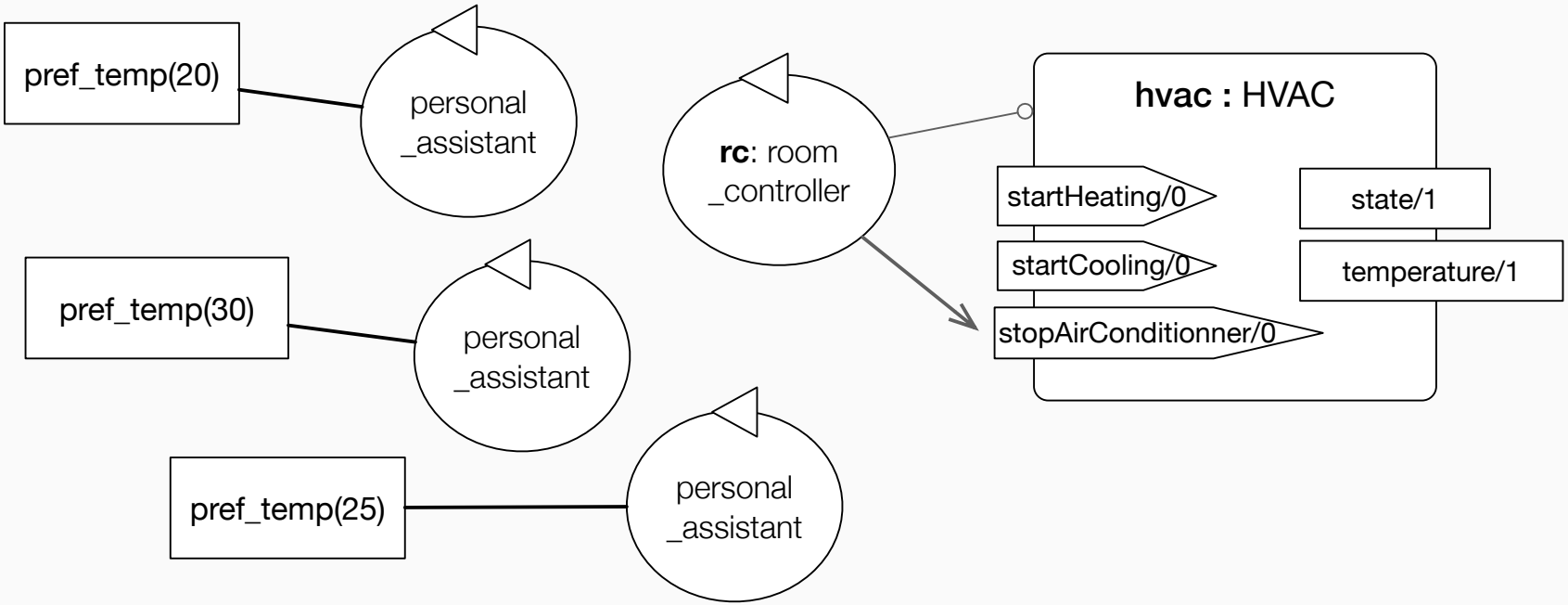
JaCaMo Performatives

- **tell** and untell: change beliefs of receiver
- **achieve** and unachieve: change goals of receiver
- **askOne** and askAll: ask for beliefs of the receiver
- **askHow**, tellHow, and untellHow: exchange plans with other agent
- **signal**: add an event in the receiver

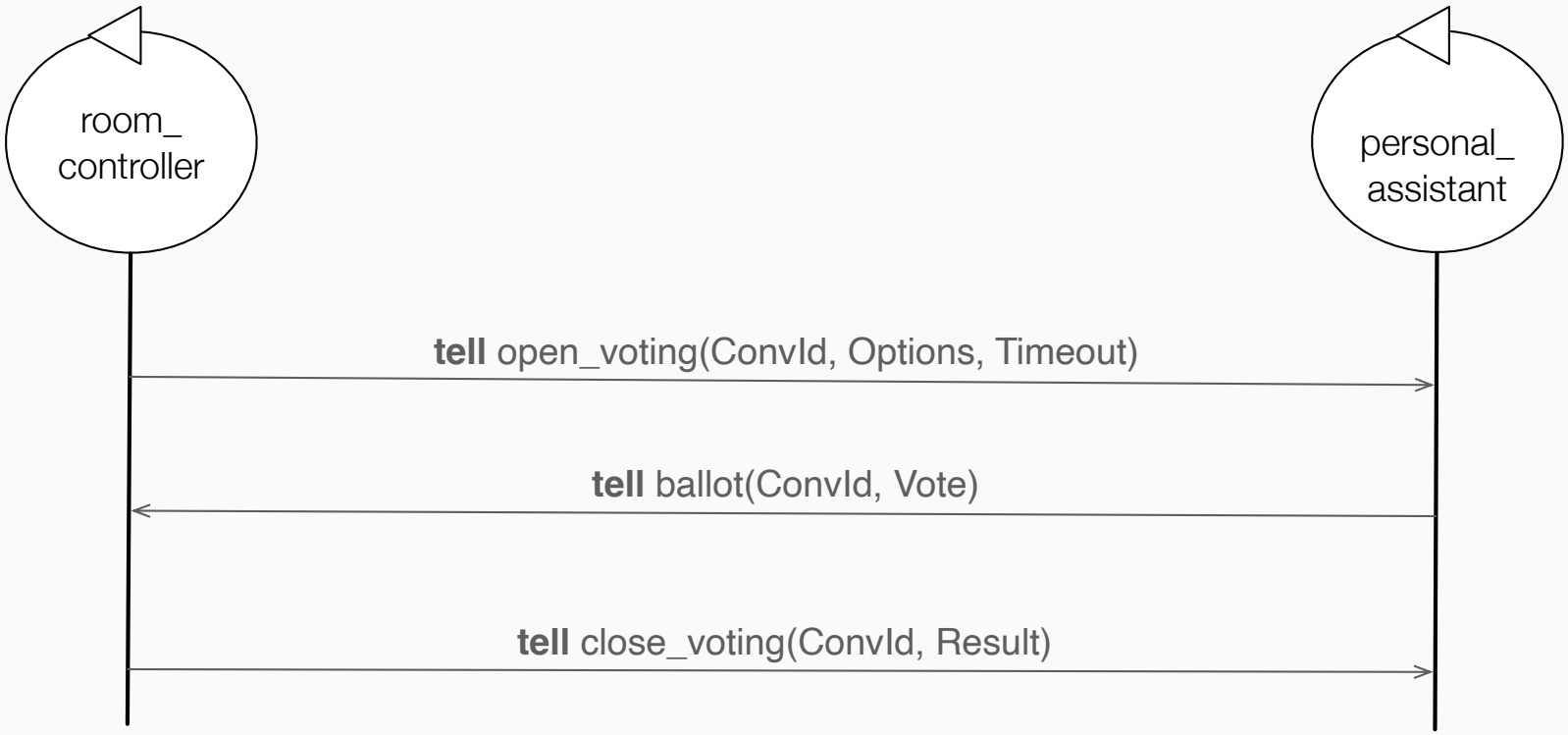
Smart Room Scenario

many users

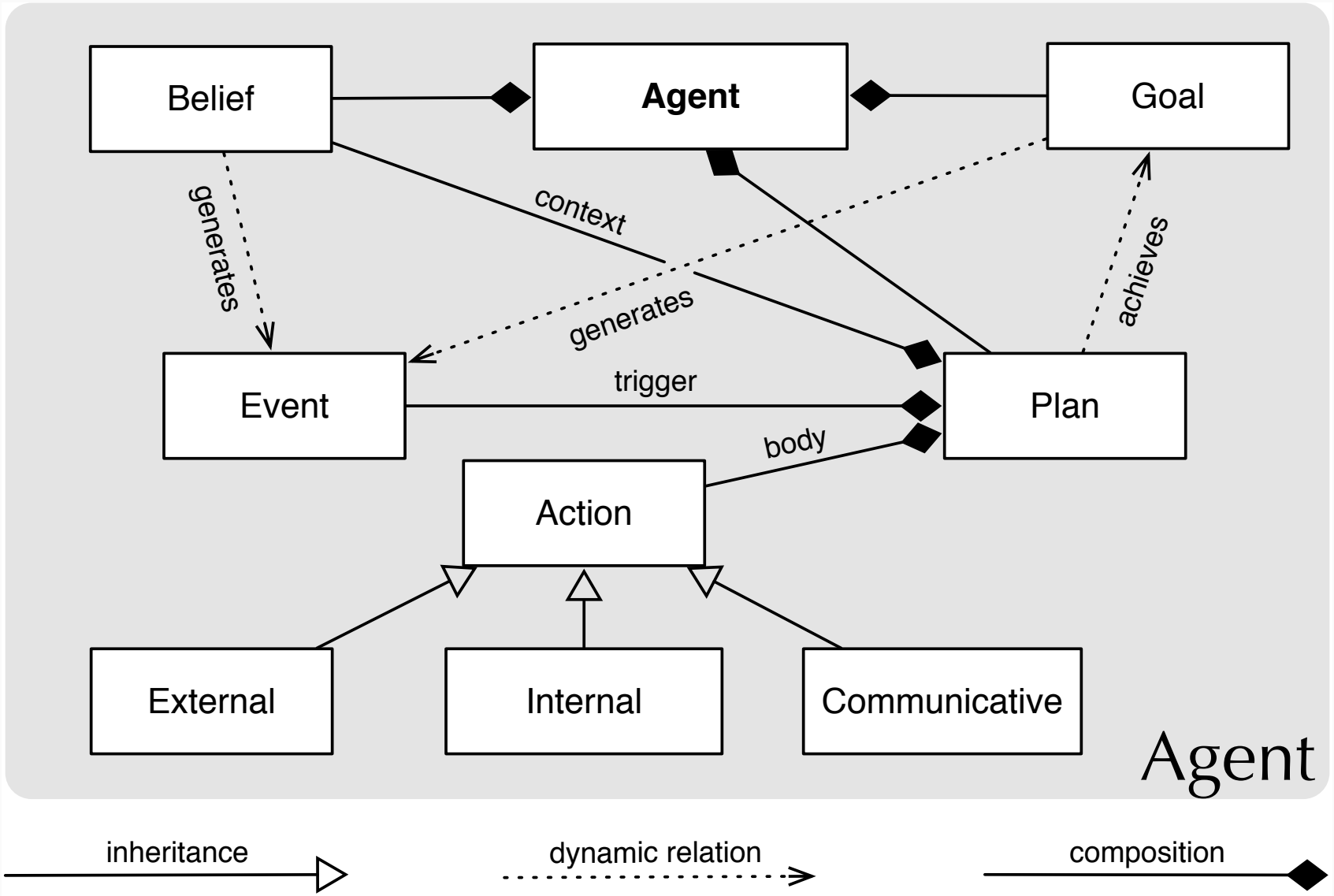
The system have to consider the preference of temperature of many users and use a voting strategy to define the target temperature



Interaction Protocols ⇝ *coordination*



Wrap-up: Agent Model



Wrap-up: Agent Programming

- **AgentSpeak**
  - Logic + BDI
  - Agent programming language
- ***Jason***
  - AgentSpeak interpreter
  - Implements the operational semantics of AgentSpeak
  - Speech-act based communicaiton
  - Highly customisable
  - Useful tools
  - Open source