Reinforcement Learning of Context Models for Ubiquitous Computing

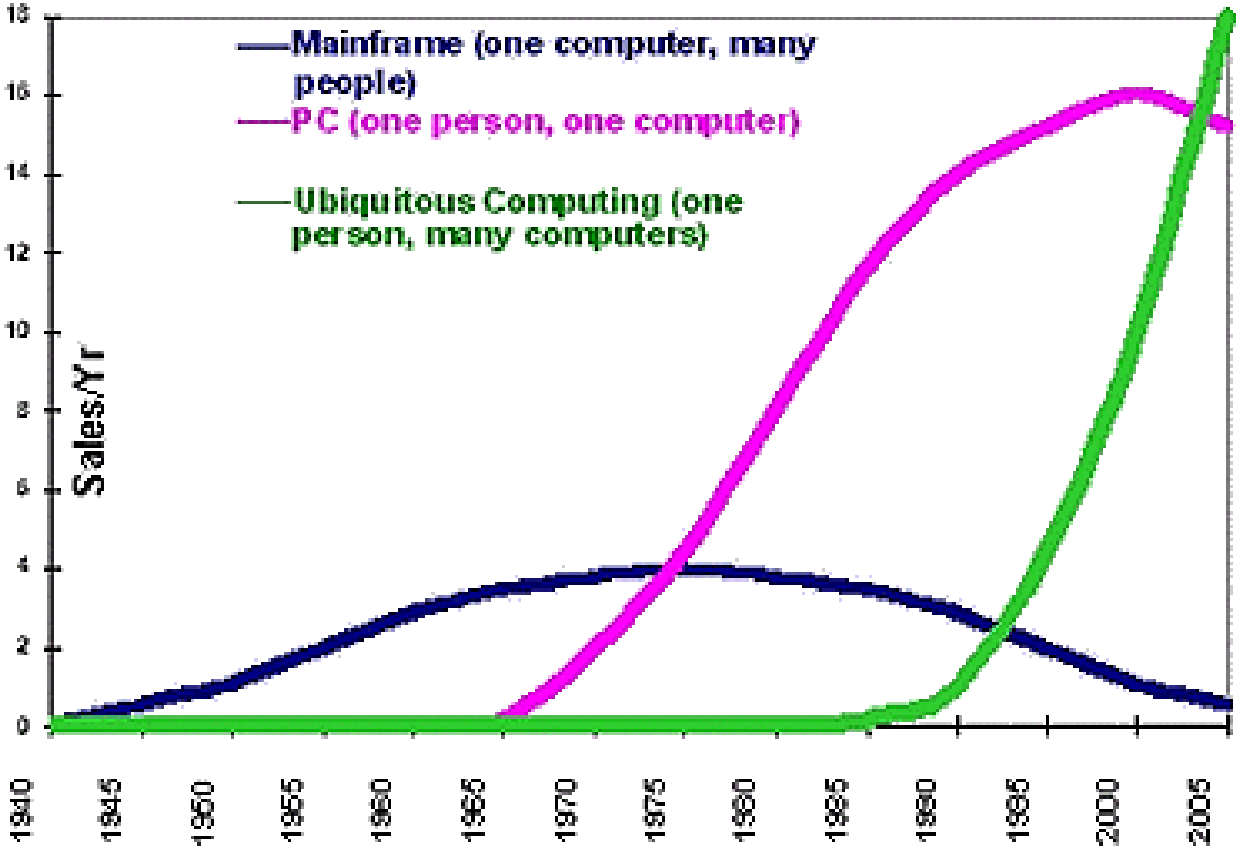
Sofia ZAIDENBERG
Laboratoire d'Informatique de Grenoble
PRIMA Group

Under the supervision of
Patrick REIGNIER and James L. CROWLEY

Ambient Computing

Ubiquitous Computing (ubicomp)

[Weiser, 1991]
[Weiser, 1994]
[Weiser and Brown, 1996]




Wi Fi
RFiD


Wi Fi
RFiD

Ambient Computing

- *« autistic »* devices
  - Independent
  - Heterogeneous
  - Unaware

- Ubiquitous systems
  - Accompany without imposing
  - In periphery of the attention
  - *Invisible*
  - *Calm computing*

15 years later…

- Ubiquitous computing is already here [Bell and Dourish, 2007]
  - It's not exactly like we expected
    - "Singapore, the intelligent island"
    - "U-Korea"
  - Not seamless but messy
  - Not invisible but flashy
  - Characterized by improvisation and appropriation
- Engaging user experiences [Rogers, 2006]

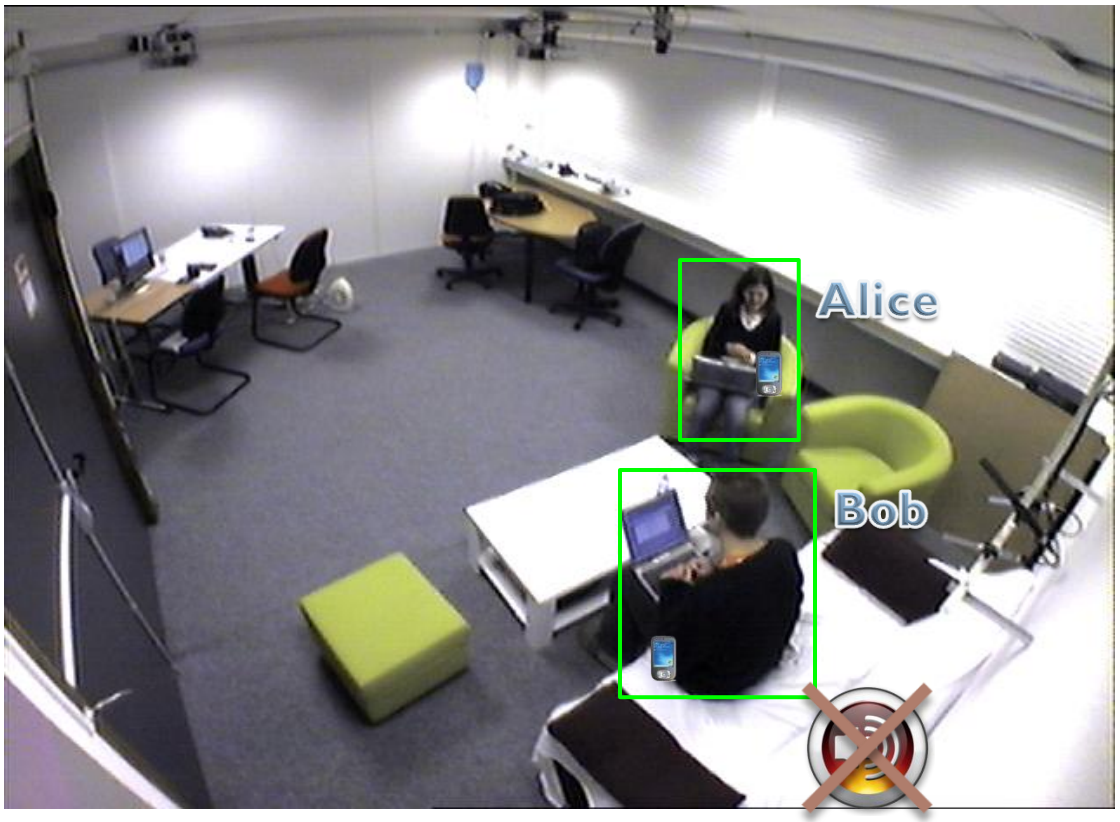
New directions

- Study habits and ways of living of people and create technologies based on that (and not the opposite)
  [Barton and Pierce, 2006; Pascoe *et al.*, 2007; Taylor *et al.*, 2007; Jose, 2008]
- Redefine smart technologies [Rogers, 2006]

- Our goal:
  - Provide an AmI application assisting the user in his everyday activities

Our goal

- **Context-aware computing** + Personalization
  - Situation + user ⇨ action



1. Perception
2. Decision

Proposed solution

Personalization by

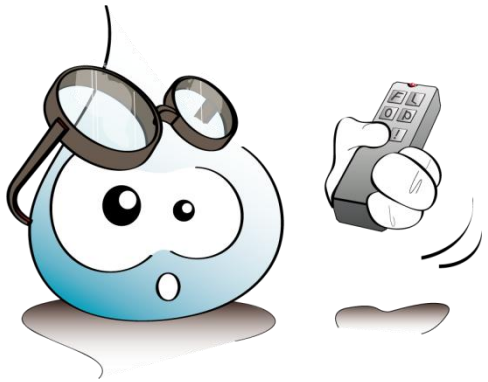
Learning

Outline

- Problem statement
- **Learning in ubiquitous systems**
- User Study
- Ubiquitous system
- Reinforcement learning of a context model
- Experimentations and results
- Conclusion

Proposed system

- A **virtual assistant** embodying the ubiquitous system
- The assistant
  - Perceives the context using its sensors
  - Executes actions using its actuators
  - Receives user feedback for training
  - Adapts its behavior to this feedback *(learning)*



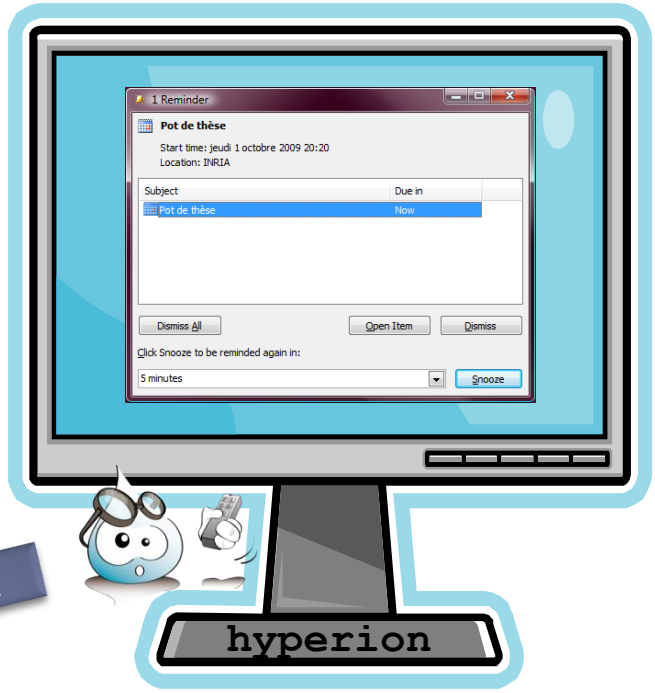
Constraints

- Simple training
- Fast learning
- Initial behavior consistency
- *Life long learning* → The system is adapting to environment and preferences changes
- User trust

- Transparence [Bellotti and Edwards, 2001]
  - Intelligibility
    - System behavior understood by humans
  - Accountability
    - System able to explain itself

Example

Reminder!



J109

J120

Outline

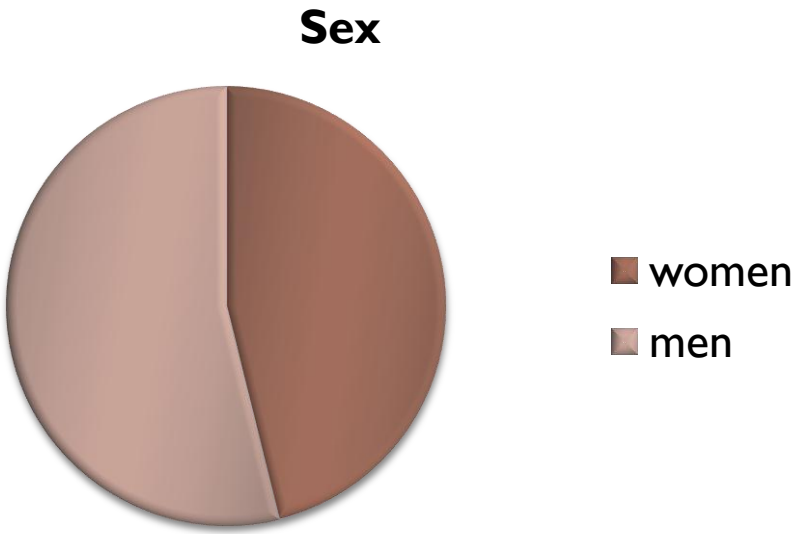
- Problem statement
- Learning in ubiquitous systems
- **User Study**
- Ubiquitous system
- Reinforcement learning of a context model
- Experimentations and results
- Conclusion

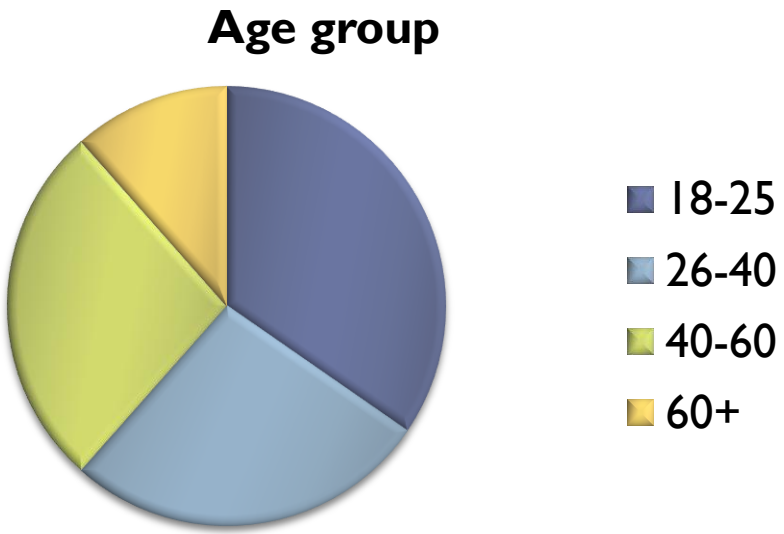
User study

- Why a general public user study?
  - Original Weiser's vision of *calm computing* revealed itself to be unsuited to current user needs
- Objective
  - Evaluate the expectations and needs vis-à-vis "ambient computing" and its usages

Terms of the study

- 26 interviewed subjects
  - Non-experts
  - Distributed as follows:





Terms of the study

- 1 hour interviews with open discussion and support of an interactive model
- Questions about advantages and drawbacks of a ubiquitous assistant
- User ideas about interesting and useful usages

Results

- 44 % of subjects interested, 13 % conquered
- Profiles of interested subjects:
  - Very busy people
  - Experiencing cognitive overload
- *Leaning* considered as a plus
  - More reliable system
  - ✓ Gradual training *vs.* heavy configuration
  - ✓ Simple and pleasant training (“one click”)

Results

- ✓ Short learning phase
- ✓ Explanations are essential

- Interactions
  - Depending on the subject
  - Optional debriefing phase
- Mistakes accepted as long as the consequences are not critical
- Use control
- Reveals subconscious customs
- Worry of dependence

Conclusions

- Constraints
  - Not a black box
    - [Bellotti and Edwards, 2001] Intelligibility and accountability
  - Simple, not intrusive training
  - Short training period, fast re-adaptation to preference changes
  - Coherent initial behavior

⇨ Build a ubiquitous assistant based on these constraints

Outline

- Problem statement
- Learning in ubiquitous systems
- User Study
- **Ubiquitous system**
- Reinforcement learning of a context model
- Experimentations and results
- Conclusion

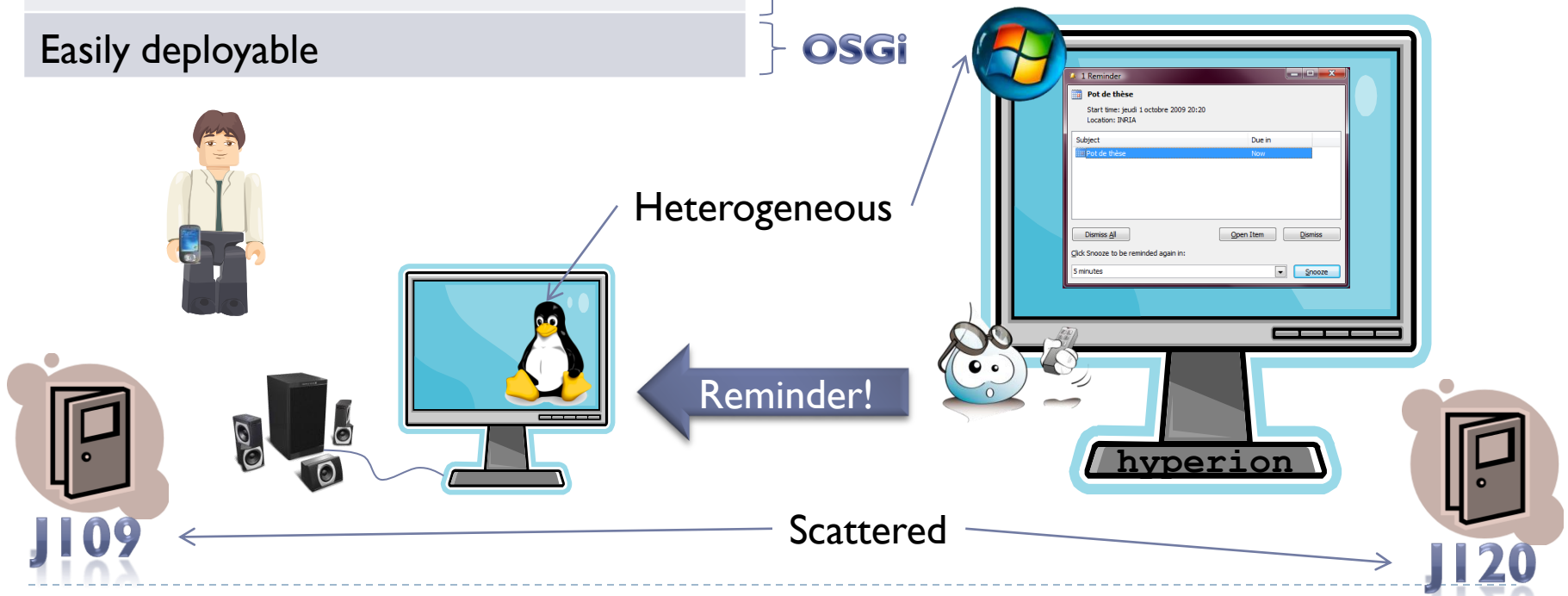
Ubiquitous system

System needs
Multiplatform system
Distributed system
Communication protocol
Dynamic service discovery
Easily deployable

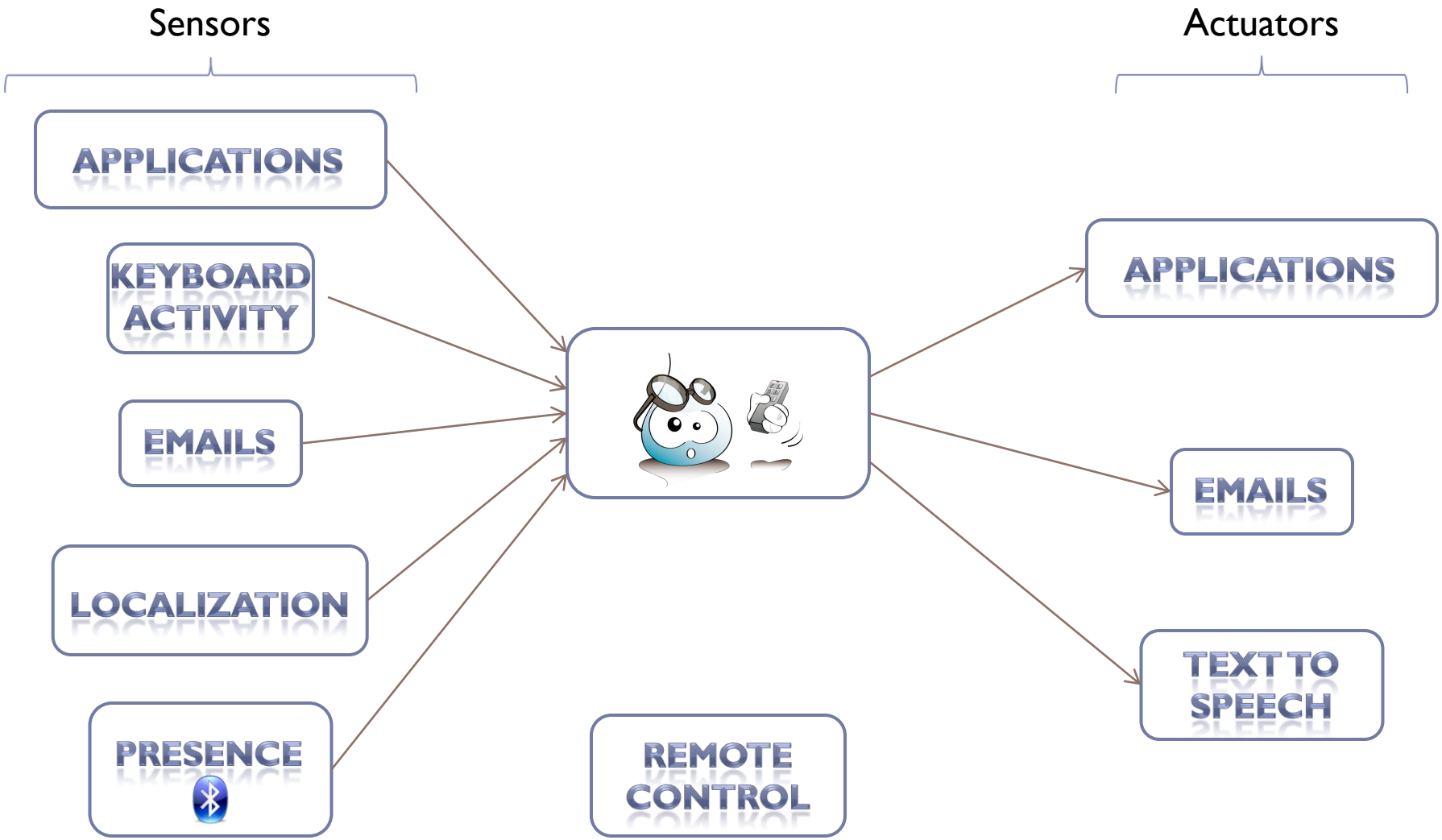
OMiSCID [Emonet *et al.*, 2006]

OSGi

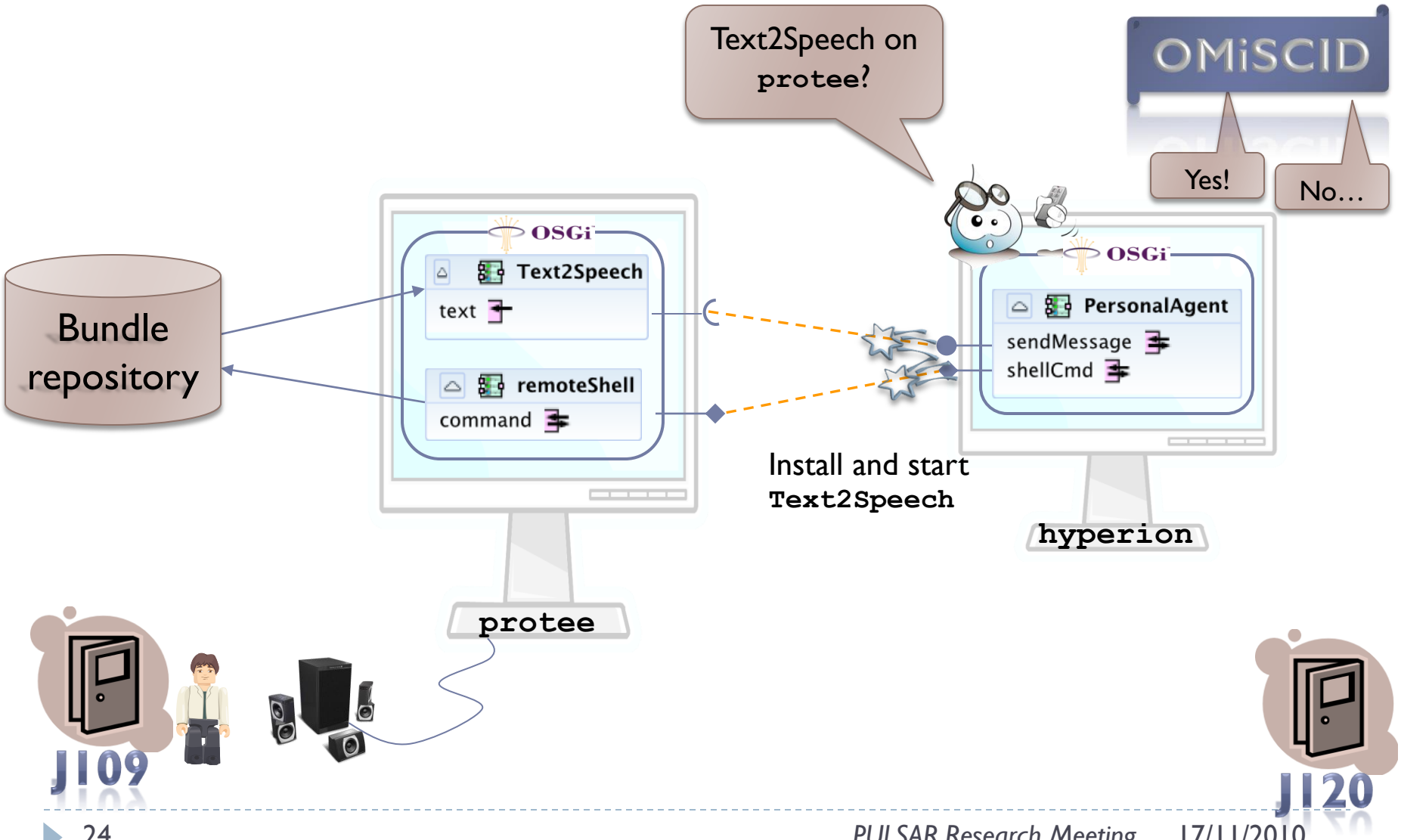
- Uses the existing devices



Modules interconnection



Example of message exchange



Database

- Contains
  - Static knowledge
  - History of events and actions
    - To provide explanations
- Centralized
  - Queried
  - Fed

  } by all modules on all devices
  - Simplifies queries

Outline

- Problem statement
- Learning in ubiquitous systems
- User Study
- Ubiquitous system
- **Reinforcement learning of a context model**
  - Reinforcement learning
  - Applying reinforcement learning
- Experimentations and results
- Conclusion

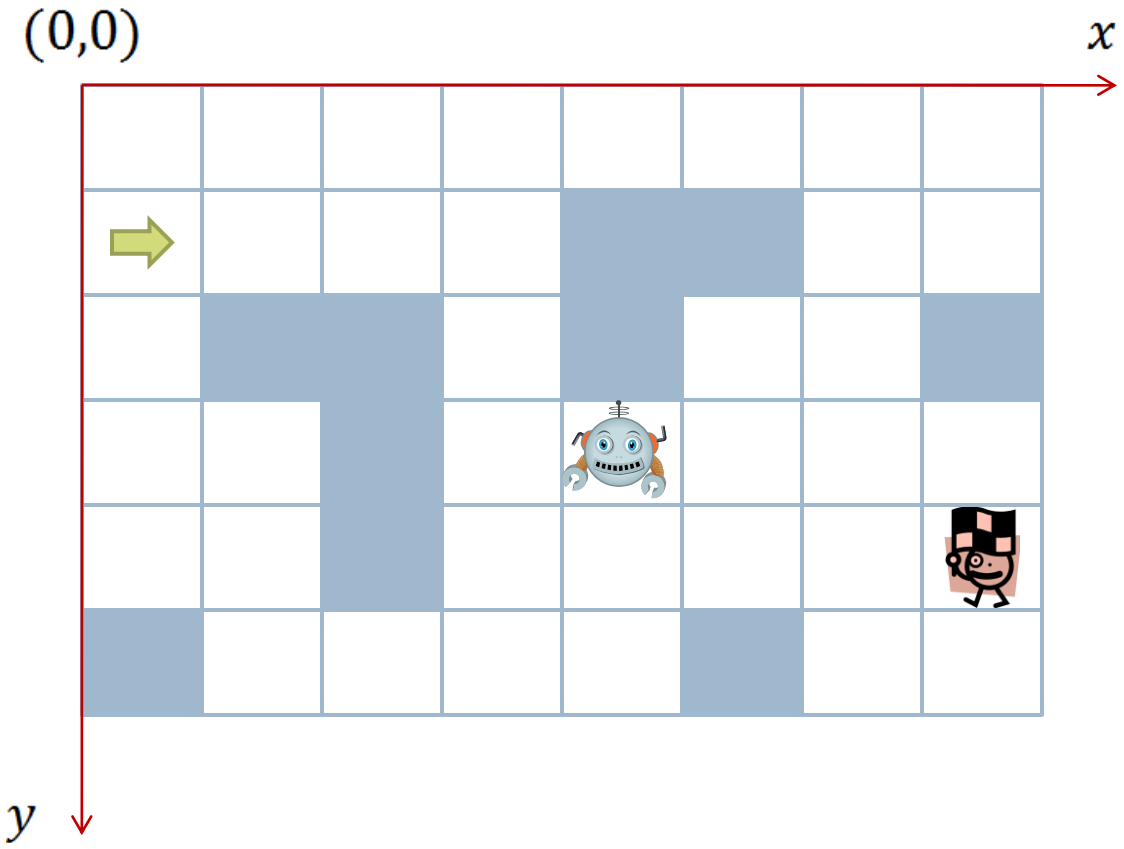
Reminder: our constraints

- Simple training
- Fast learning
- Initial behavior consistency
- Life long learning
- Explanations

Supervised

[Brdiczka *et al.*, 2007]

Reinforcement learning (RL)



$Q(state, action)$

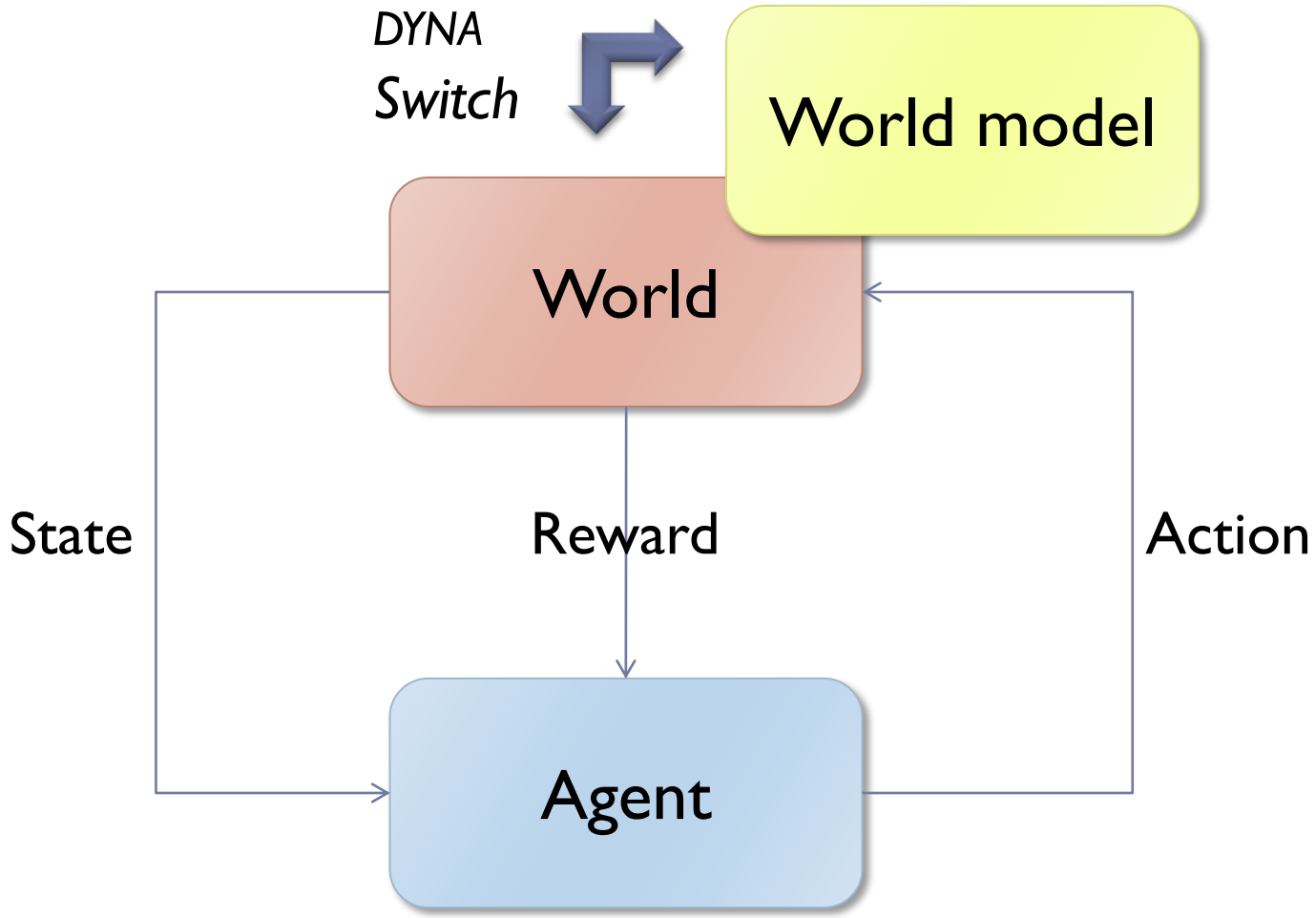
- Markov property
  - The state at time *t* depends only on the state at time *t-1*

Standard algorithm

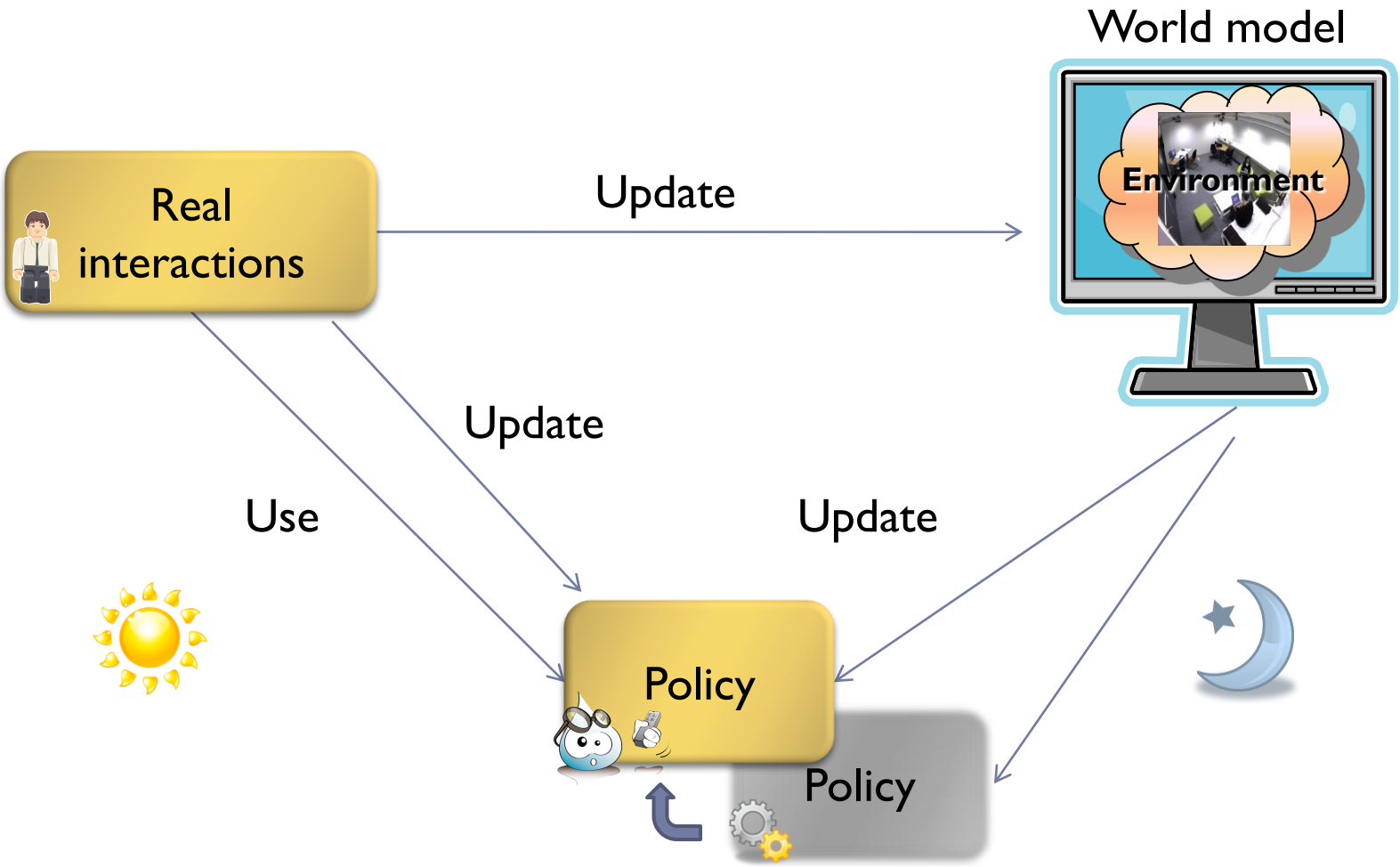
- *Q-Learning* [Watkins, 1989]
  - Updates Q-values on a new experience
    {state, action, next state, reward}
  - Slow because evolution only when something happens
    - Needs *a lot* of examples to learn a behavior

DYNA architecture

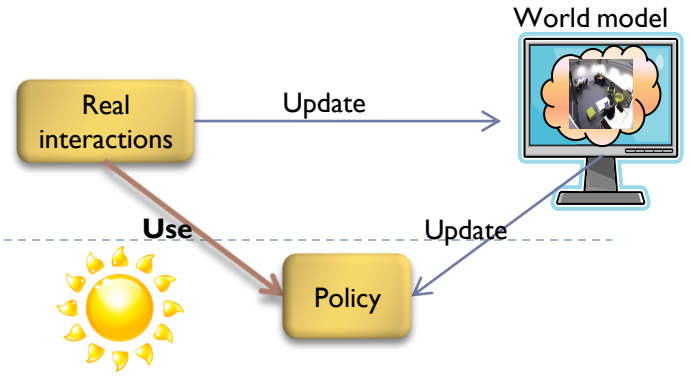
[Sutton, 1991]

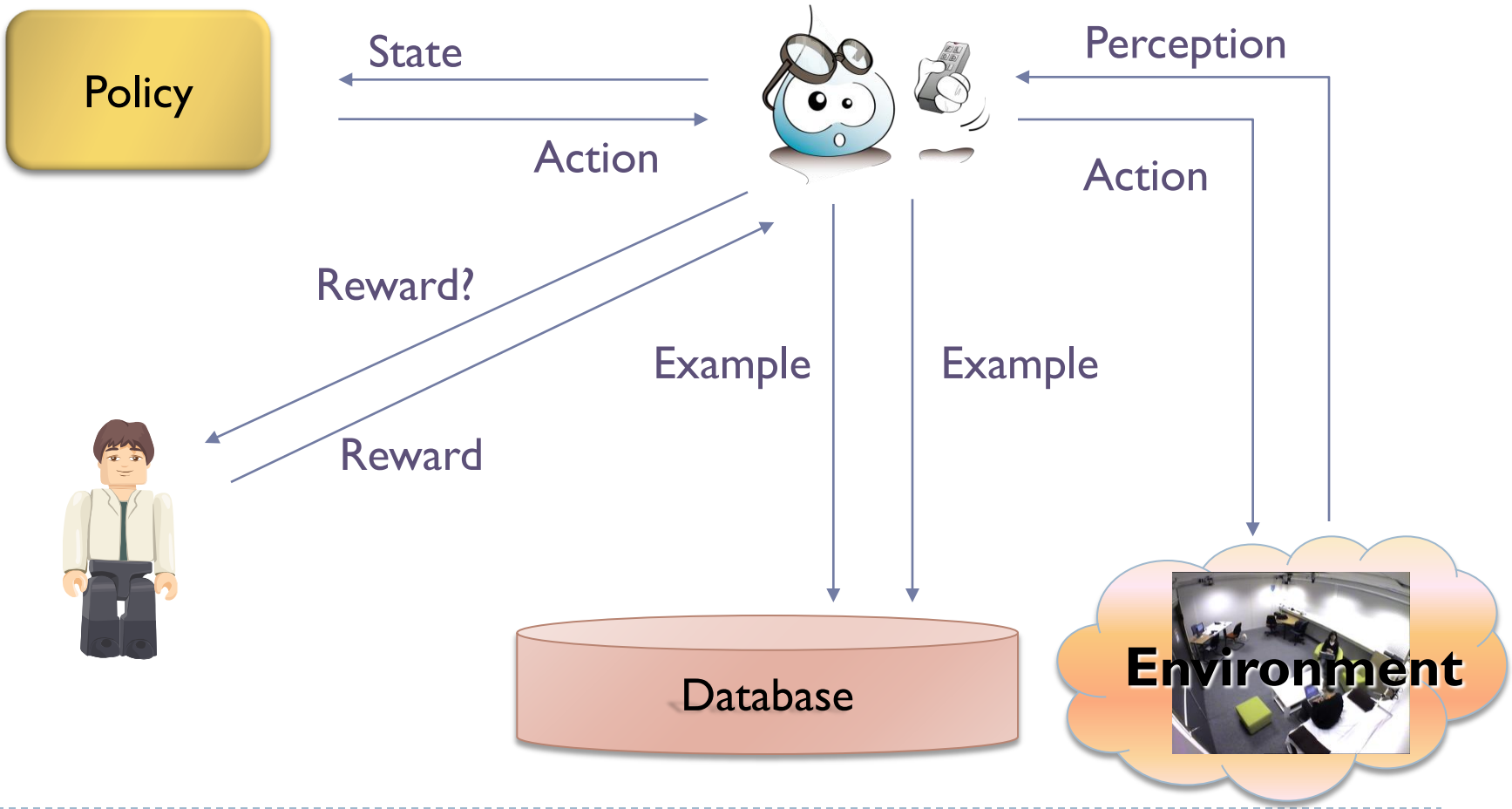


DYNA architecture



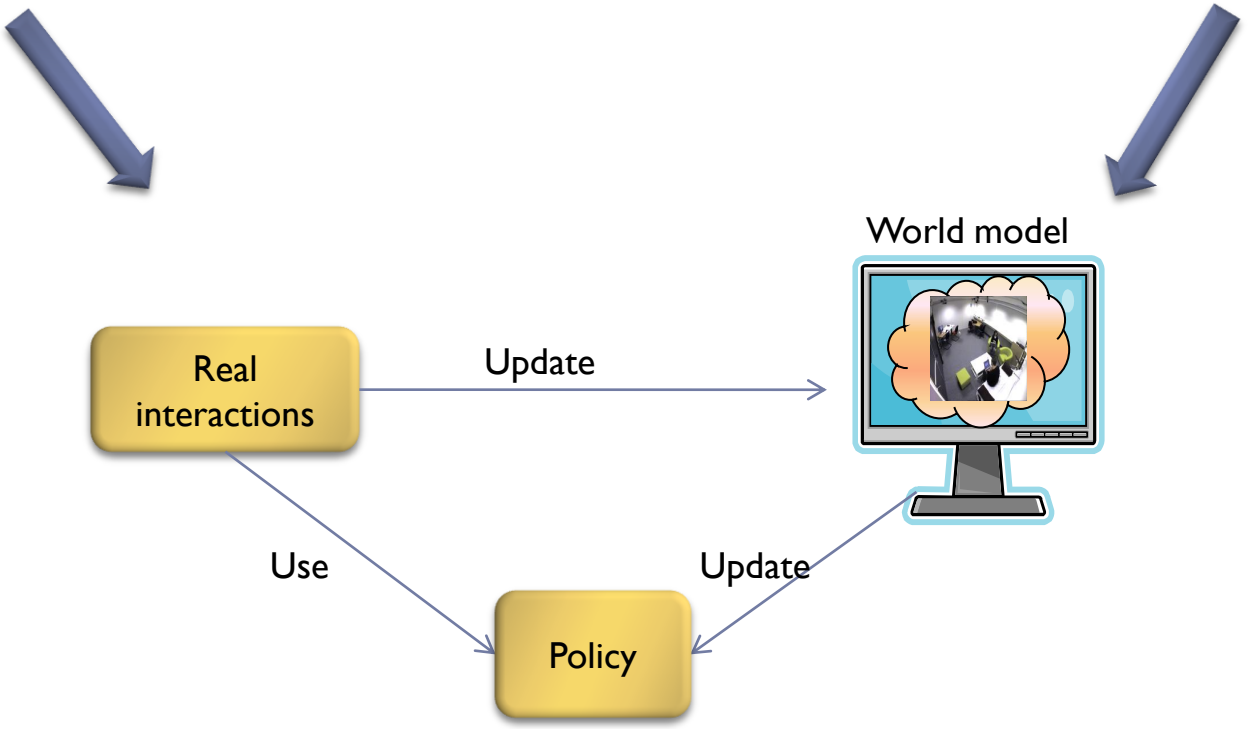
Global system



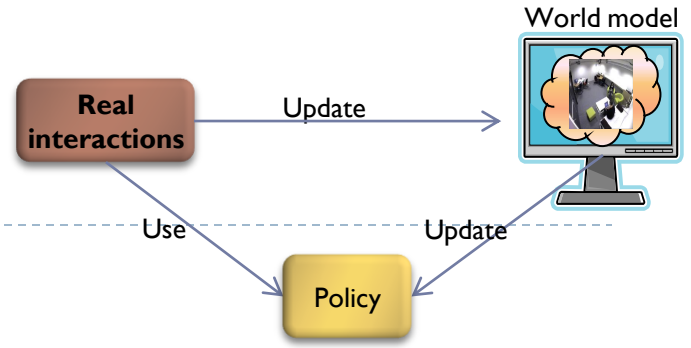


Modeling of the problem

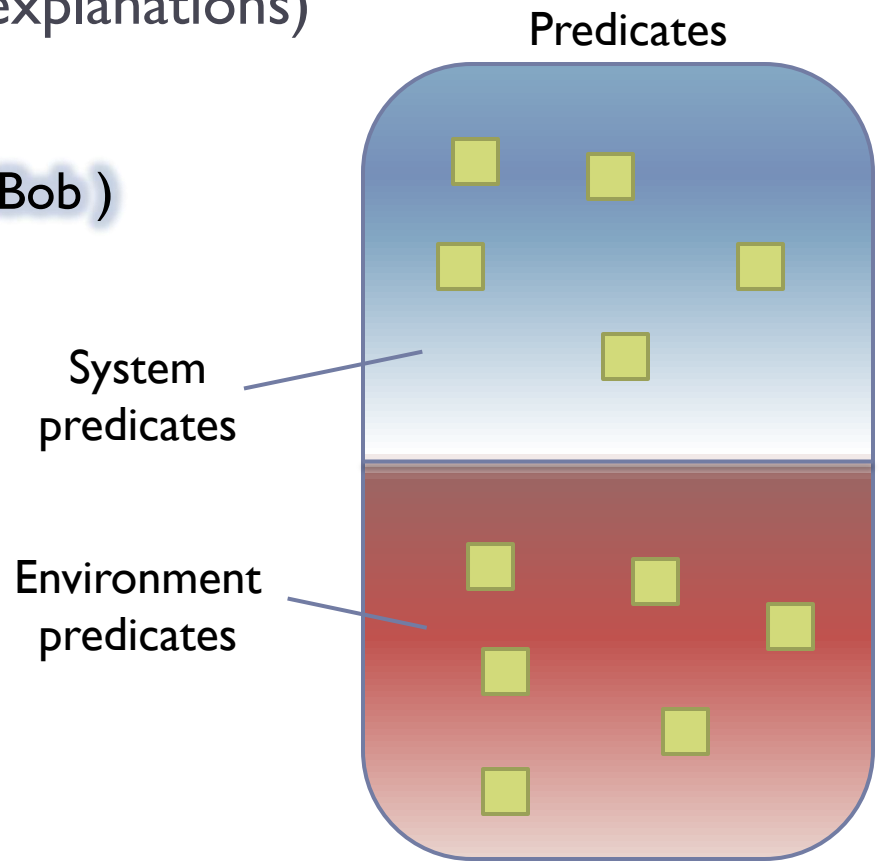
- Components:
  - States
  - Actions
- Components:
  - Transition model
  - Reward model



State space



- States defined by *predicates*
  - Understandable by humans (explanations)
  - Examples :
    - newEmail (from = Marc, to = Bob)
    - isInOffice (John)
- State-action:
  - entrance(~~Karl~~)
  - ⇨ Pause music

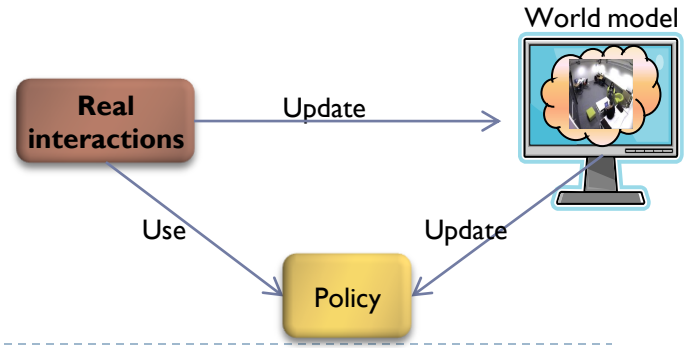


State space

- State split
  - newEmail(from= directeur, to= <+>)
    - ⇨ Notify
  - newEmail(from = newsletter, to= <+>)
    - ⇨ Do not notify

Action space

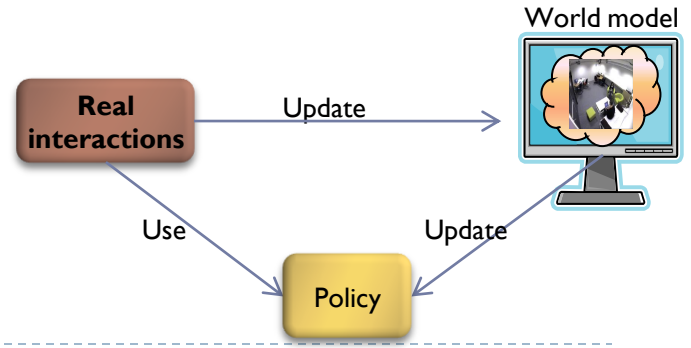
- Possible actions combine
  - Forwarding a reminder to the user
  - Notify of a new email
  - Lock a computer screen
  - Unlock a computer screen
  - Pause the music playing on a computer
  - Un-pause the music playing on a computer
  - Do nothing



Reward

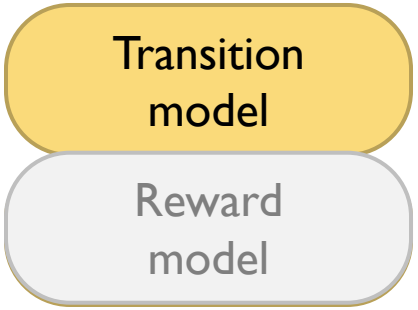
- Explicit reward
  - Through a non-intrusive user interface

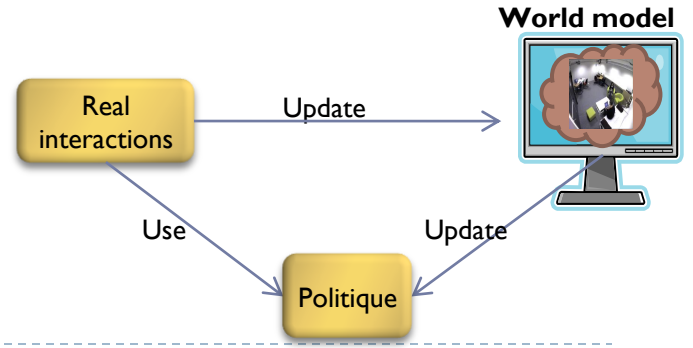
- Problems with user rewards
  - Implicit reward
    - Gathered from clues
      (numerical value of lower amplitude)
  - Smoothing of the model



World model

- Built using supervised learning
  - From real examples
- Initialized using common sense
  - Functional system from the beginning
  - Initial model *vs.* initial Q-values [Kaelbling, 2004]
  - Extensibility

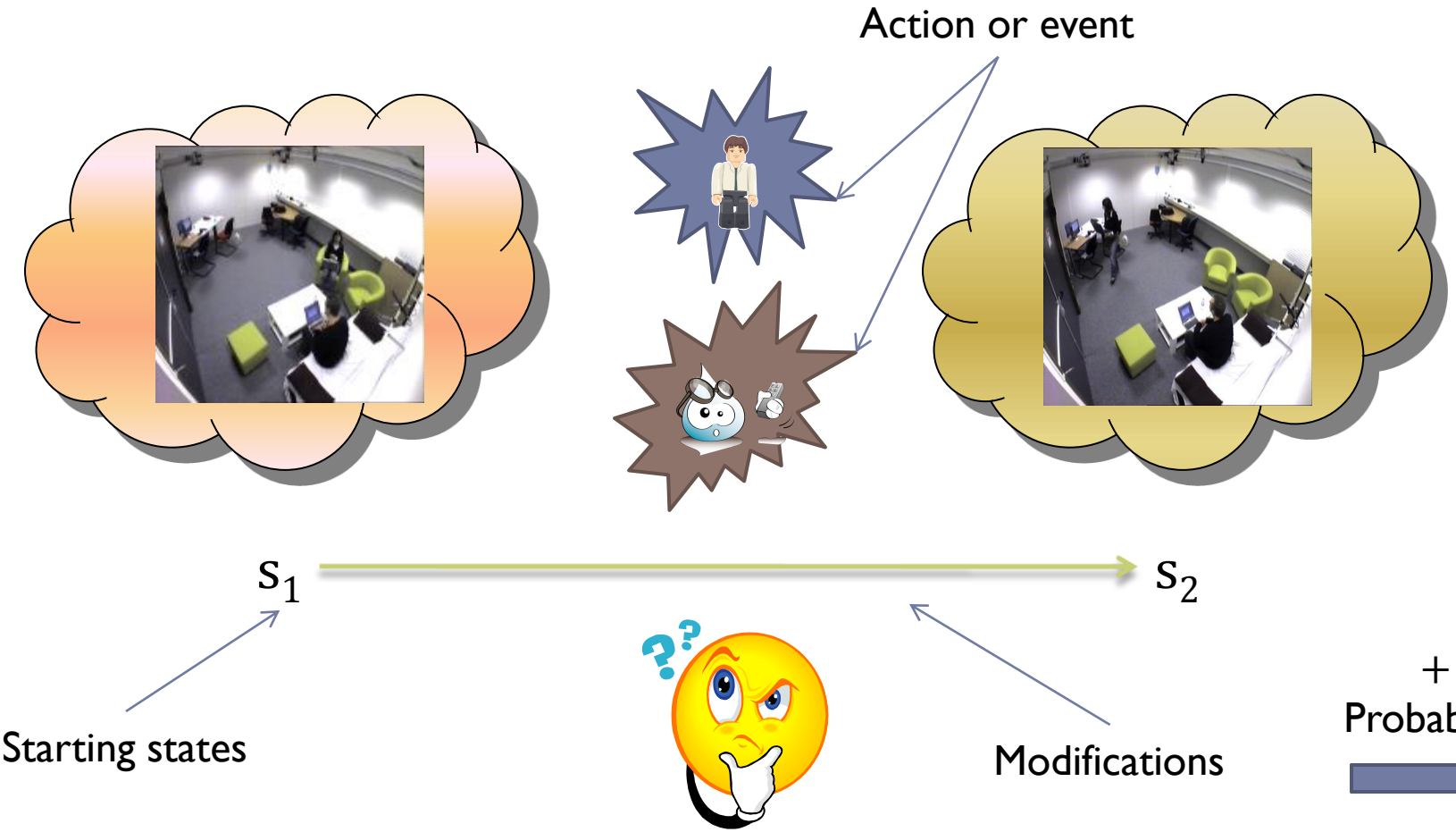




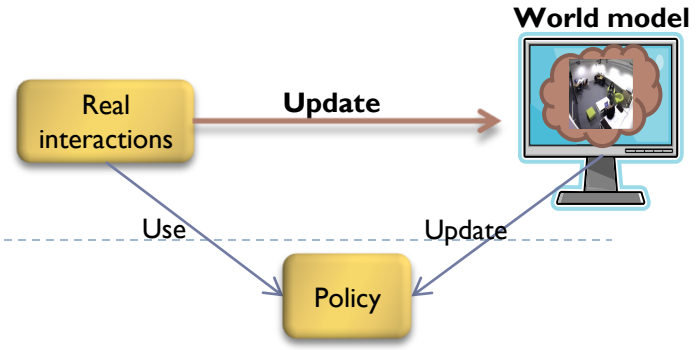
Transition model

Transition model

Reward model

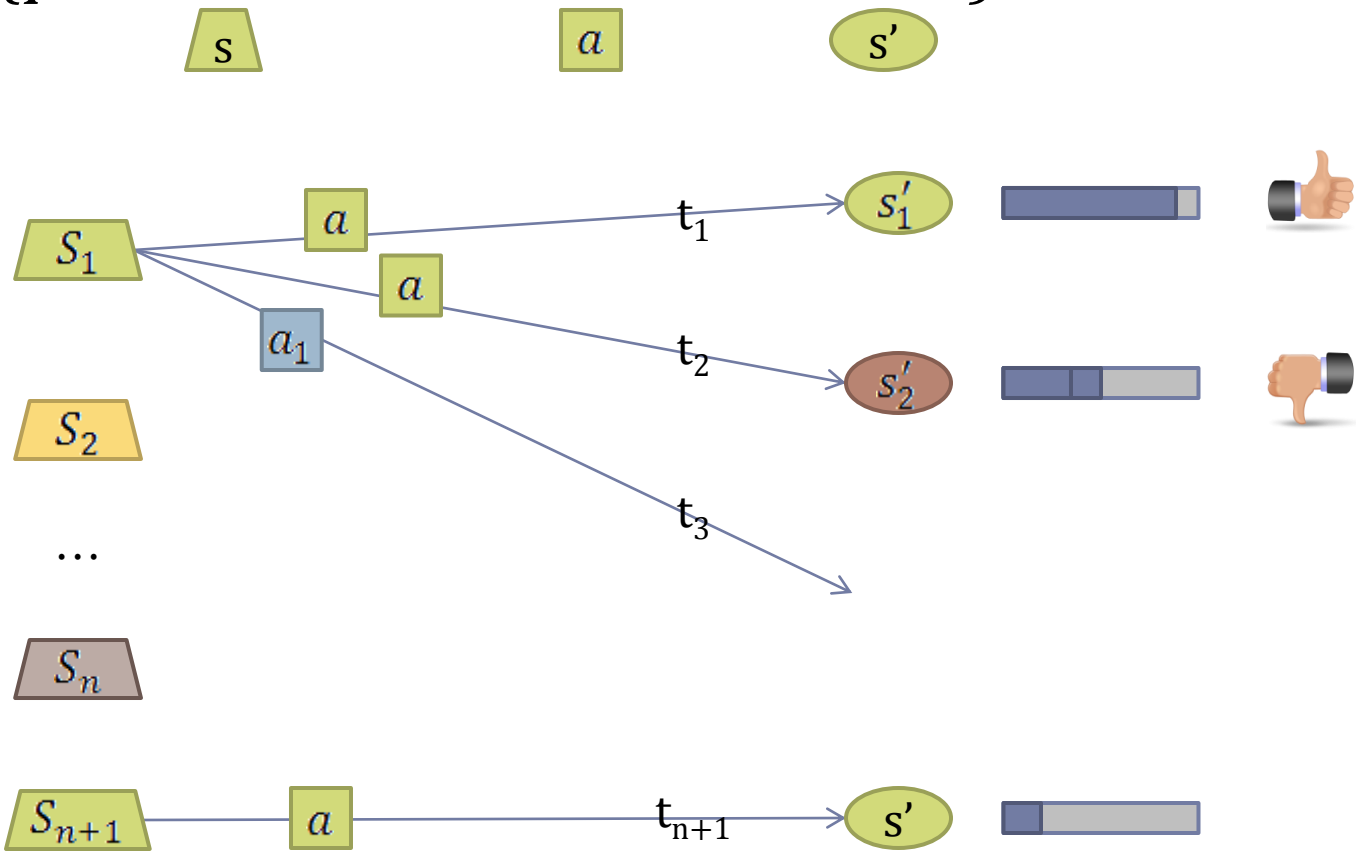


Supervised learning of the transition model

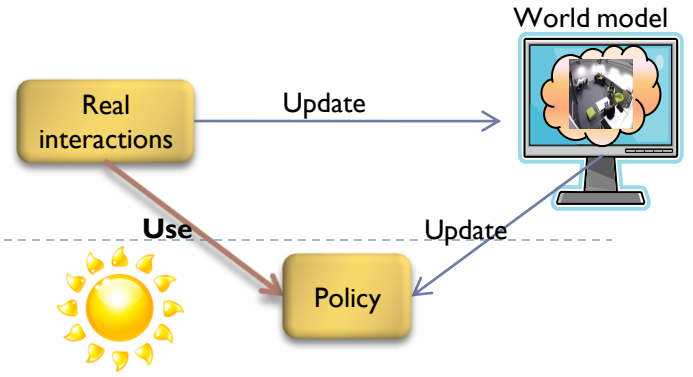


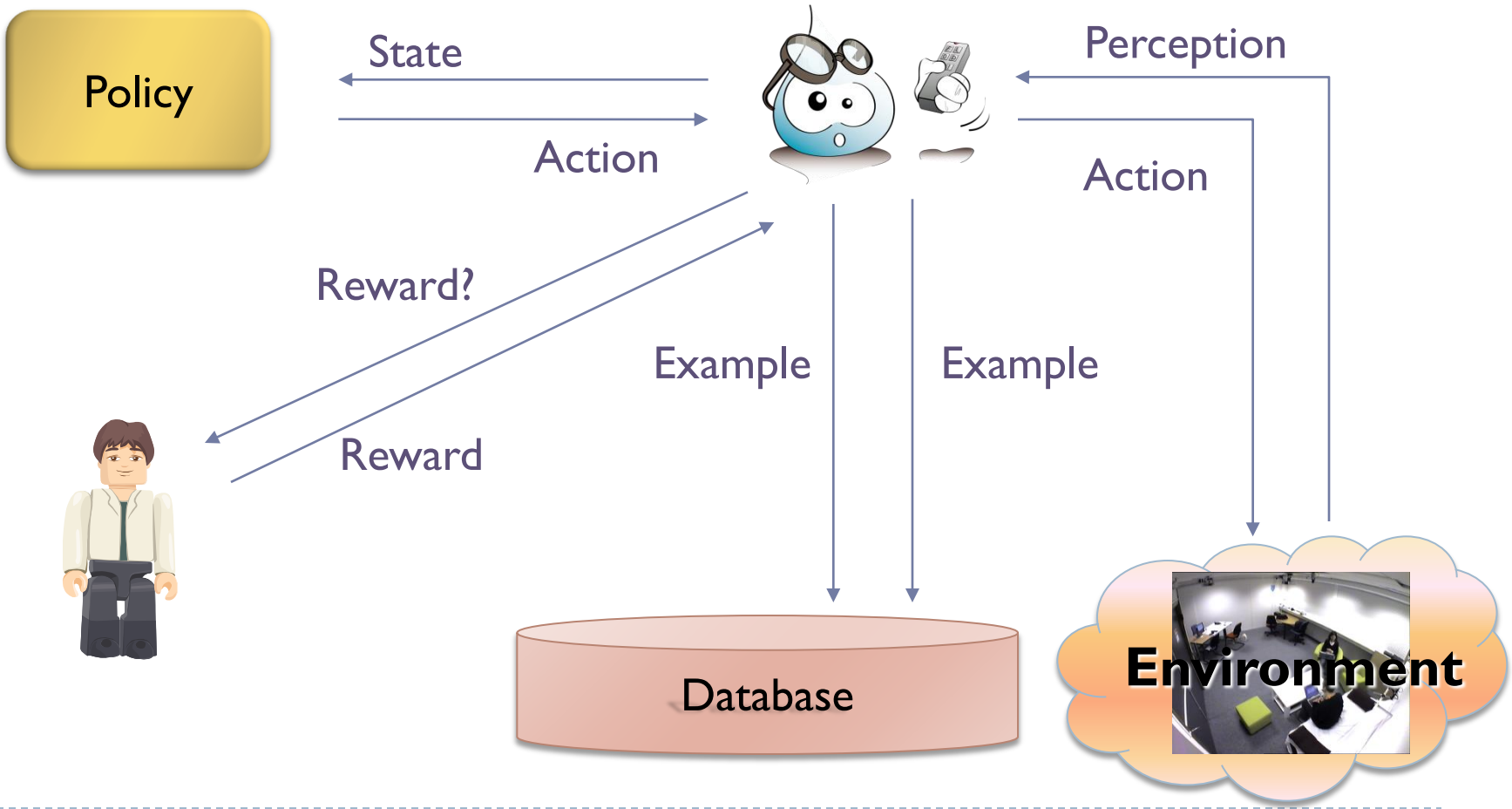
- Database of examples

  {previous state, action, next state}



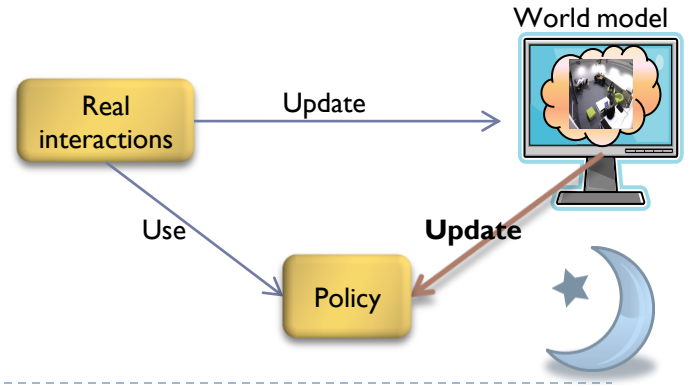
Global system



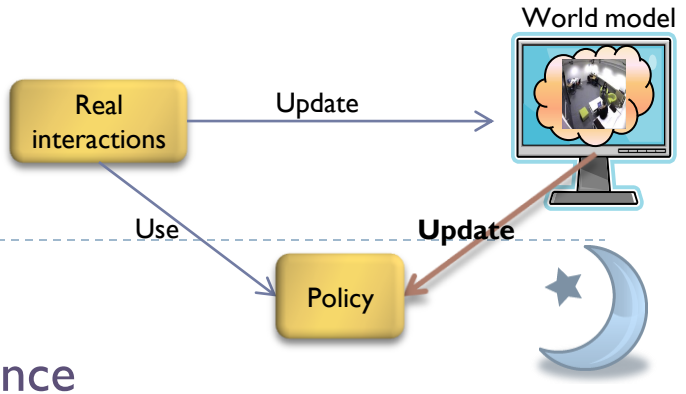


Episode

- Episode steps have 2 stages:
  - Select an event that modifies the state
  - Select an action to react to that event



Episode



Q-Learning updates

$Q(s', a)$

Experience

$\{\ ,\ ,\ ,\ \}$

Policy

s'

a

RL agent

(s', a)

(s'', r)

(s, e)

s'

e

or

e

World model

Environment

Policy

Database

Learned from real interactions

Outline

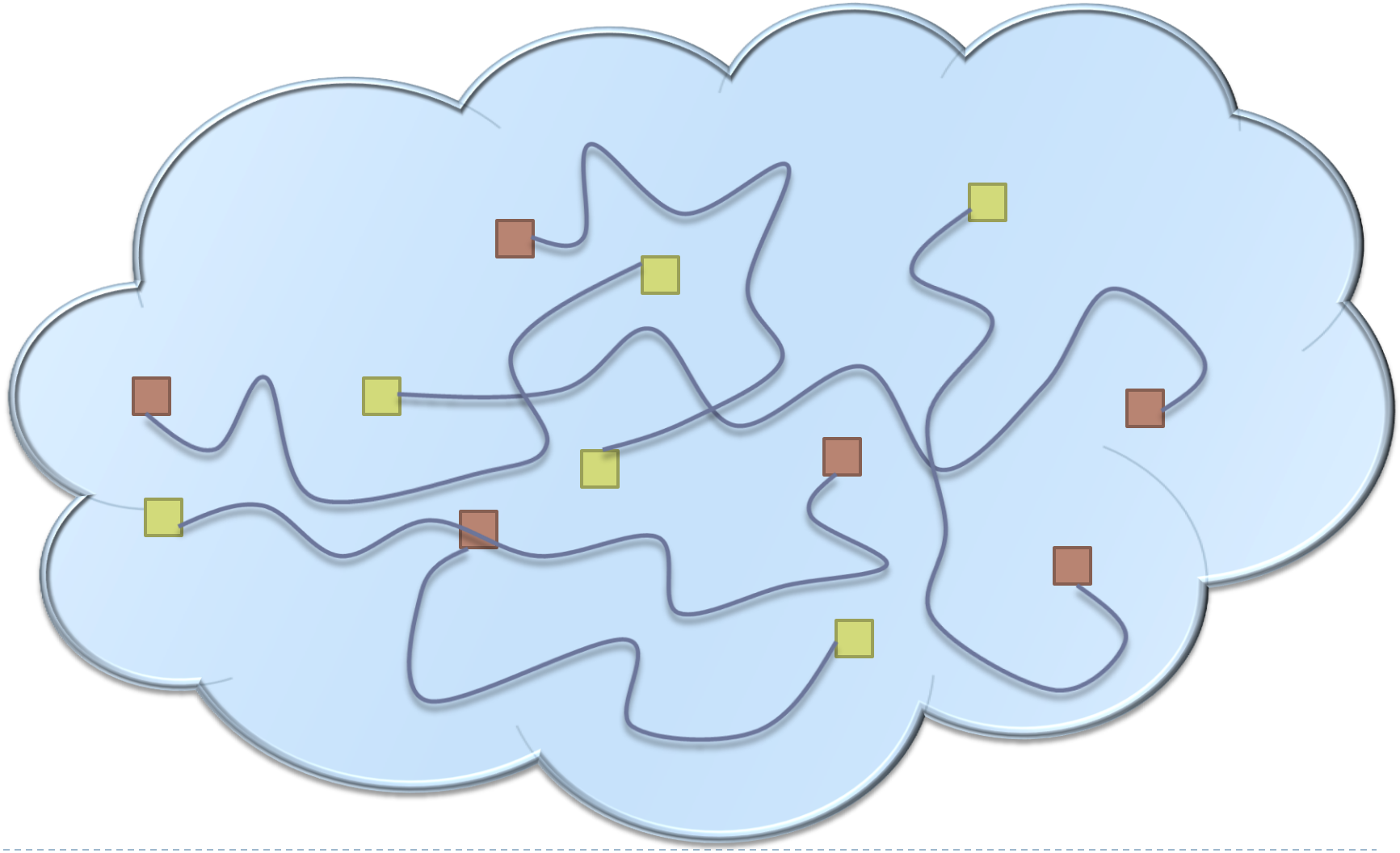
- Problem statement
- Learning in ubiquitous systems
- User Study
- Ubiquitous system
- Reinforcement learning of a context model
- **Experimentations and results**
- Conclusion

Experimentations

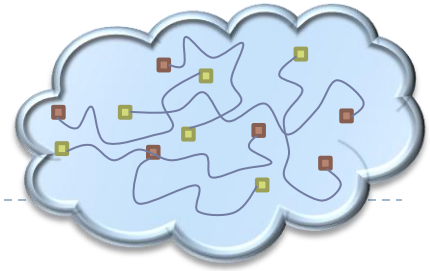
- General public survey → qualitative evaluation
- Quantitative evaluations in 2 steps:
  - Evaluation of the initial phase
  - Evaluation of the system during normal functioning

Evaluation 1

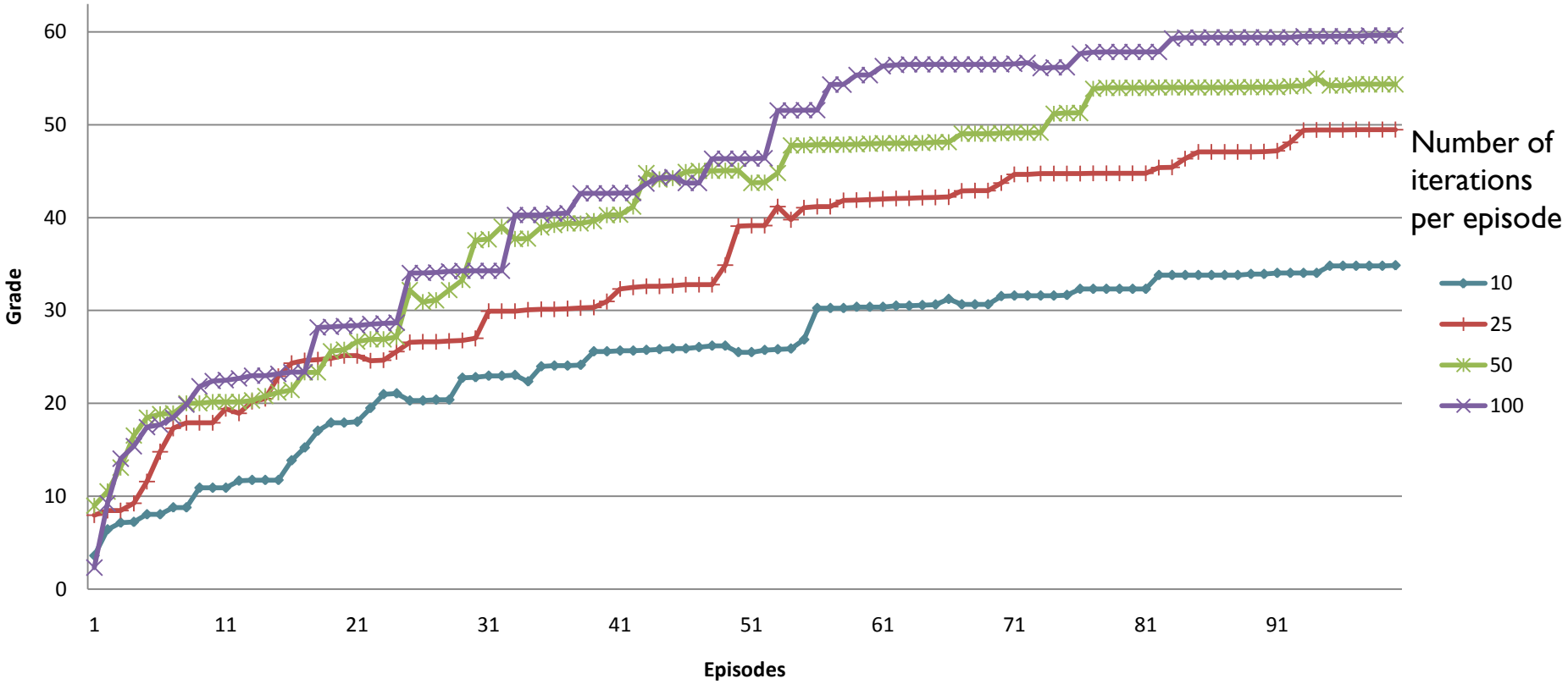
« about initial learning »



Evaluation 1
« about initial learning »

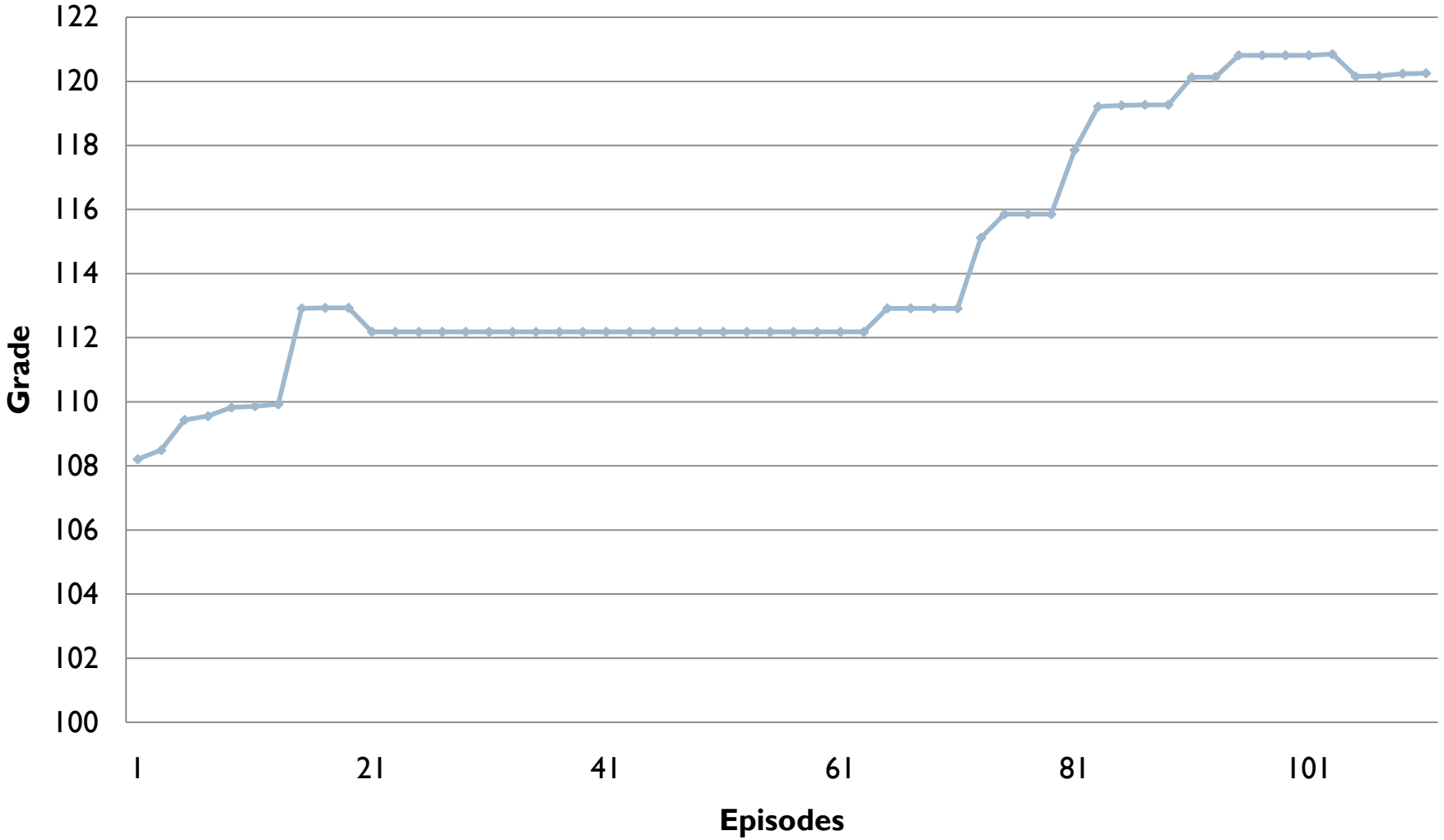


Initial episodes with events and initial states randomly chosen from the database



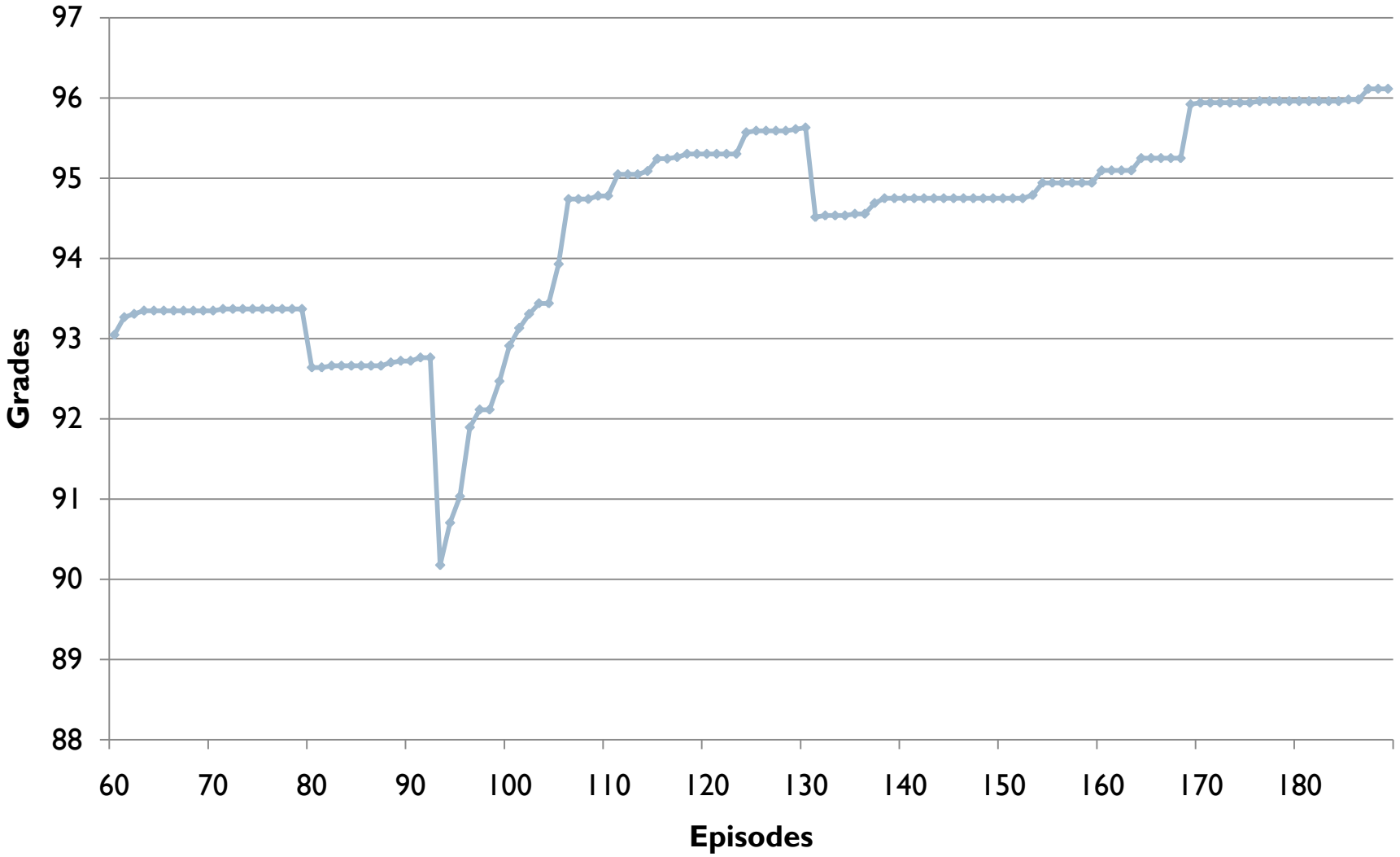
Evaluation 2

« interactions and learning »



Evaluation 2

« interactions and learning »



Outline

- Problem statement
- Learning in ubiquitous systems
- User Study
- Ubiquitous system
- Reinforcement learning of a context model
- Experimentations and results
- **Conclusion**

Contributions

- Personalization of a ubiquitous system
  - Without explicit specification
  - Easy to evolve
- Adaptation of indirect reinforcement learning to a real-world problem
  - Construction of a world model
  - Injection of initial knowledge
- Deployment of a prototype

Perspectives

- Non-interactive analyze of data
- User interactions
  - Debriefing

Conclusion

- The assistant is a means of creating an ambient intelligence application
  - The user is the one making it smart

Thanks for your attention

Questions?

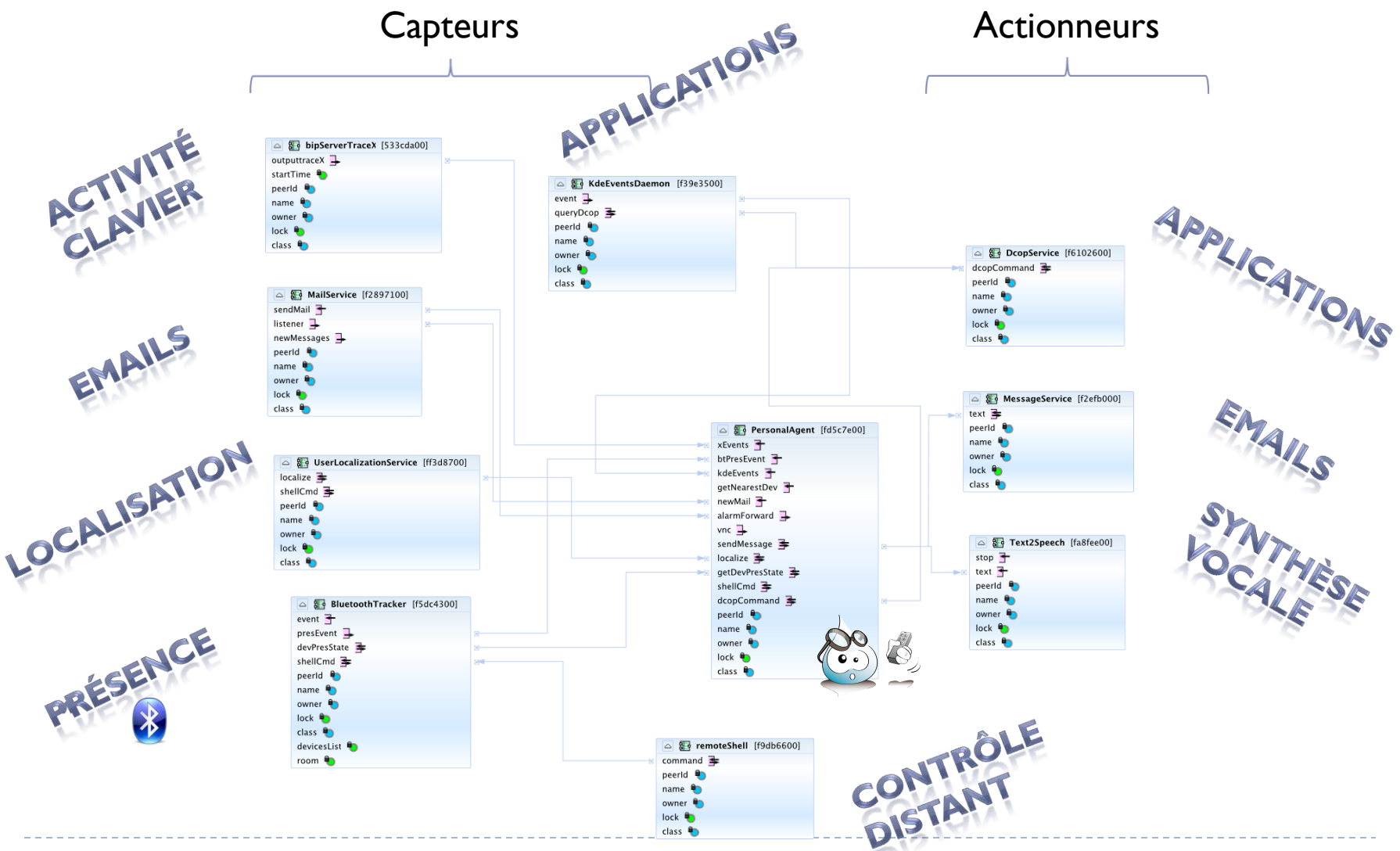
Bibliography

[Bellotti and Edwards, 2001]	Victoria BELLOTTI and Keith EDWARDS. « Intelligibility and accountability: human considerations in context-aware systems ». *In Human-Computer Interaction*, 2001.
[Brdiczka *et al.*, 2007]	Oliver BRDICZKA, James L. CROWLEY and Patrick REIGNIER. « Learning Situation Models for Providing Context-Aware Services ». *In Proceedings of HCI International*, 2007.
[Buffet, 2003]	Olivier Buffet. « Une double approche modulaire de l'apprentissage par renforcement pour des agents intelligents adaptatifs ». Thèse de doctorat, Université Henri Poincaré, 2003.
[Emonet *et al.*, 2006]	Rémi Emonet, Dominique Vaufreydaz, Patrick Reignier and Julien Letessier. « O3MiSCID: an Object Oriented Opensource Middleware for Service Connection, Introspection and Discovery ». *In IEEE International Workshop on Services Integration in Pervasive Environments*, 2006.
[Kaelbling, 2004]	Leslie Pack Kaelbling. « Life-Sized Learning ». Lecture at CSE Colloquia, 2004.
[Maes, 1994]	Pattie MAES. « Agents that reduce work and information overload ». *In Commun. ACM*, 1994.
[Maisonnasse 2007]	Jerome MAISONNASSE, Nicolas GOURIER, Patrick REIGNIER and James L. CROWLEY. « Machine awareness of attention for non-disruptive services ». *In HCI International*, 2007.
[Moore, 1975]	Gordon E. MOORE. « Progress in digital integrated electronics ». *In Proc. IEEE International Electron Devices Meeting*, 1975.

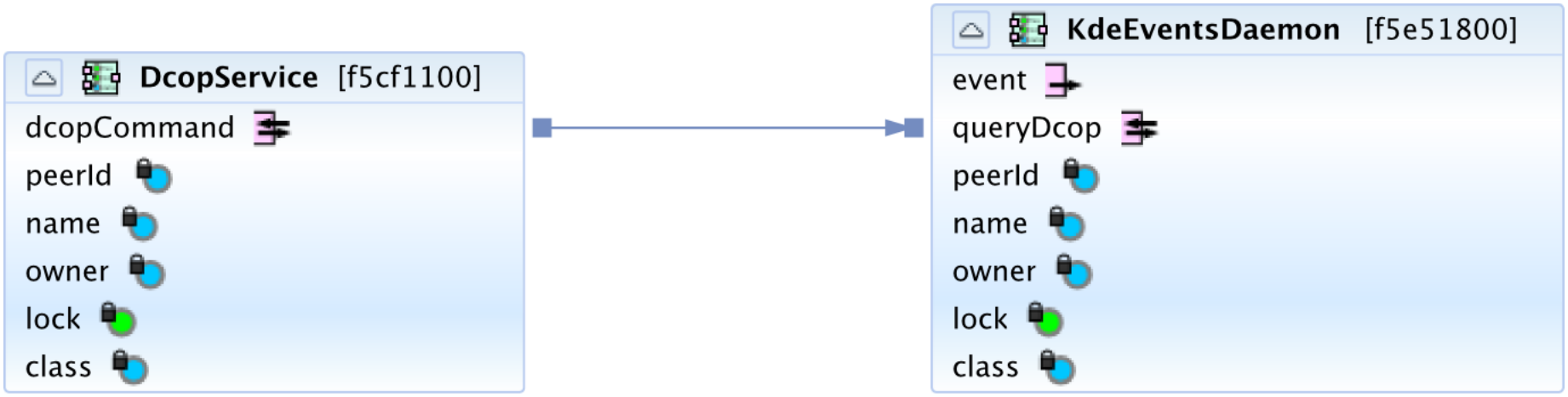
Bibliography

[Nonogaki and Ueda, 1991]	Hajime Nonogaki and Hirotada Ueda. « FRIEND21 project: a construction of 21st century human interface ». *In CHI '91: Proceedings of the SIGCHI conference on Human factors in computing systems*, 1991.
[Roman *et al.*, 2002]	Manuel ROMAN, Christopher K. HESS, Renato CERQUEIRA, Anand RANGANATHAN, Roy H. CAMPBELL and Klara NAHRSTEDT. « Gaia: A Middleware Infrastructure to Enable Active Spaces ». *In IEEE Pervasive Computing*, 2002.
[Sutton, 1991]	Richard S. Sutton. « Dyna, an integrated architecture for learning, planning, and reacting ». *In SIGART Bull,* 1991.
[Weiser, 1991]	Mark WEISER. « The computer for the 21st century ». *In Scientic American*, 1991.
[Weiser, 1994]	Mark WEISER. « Some computer science issues in ubiquitous computing ». *In Commun. ACM*, 1993.
[Weiser et Brown, 1996]	Mark WEISER and John Seely BROWN. « The coming age of calm technology ». http://www.ubiq.com/hypertext/weiser/acmfuture2endnote.htm, 1996.
[Watkins, 1989]	CJCH Watkins. « Learning from Delayed Rewards ». Thèse de doctorat, University of Cambridge, 1989.

Interconnexion des modules



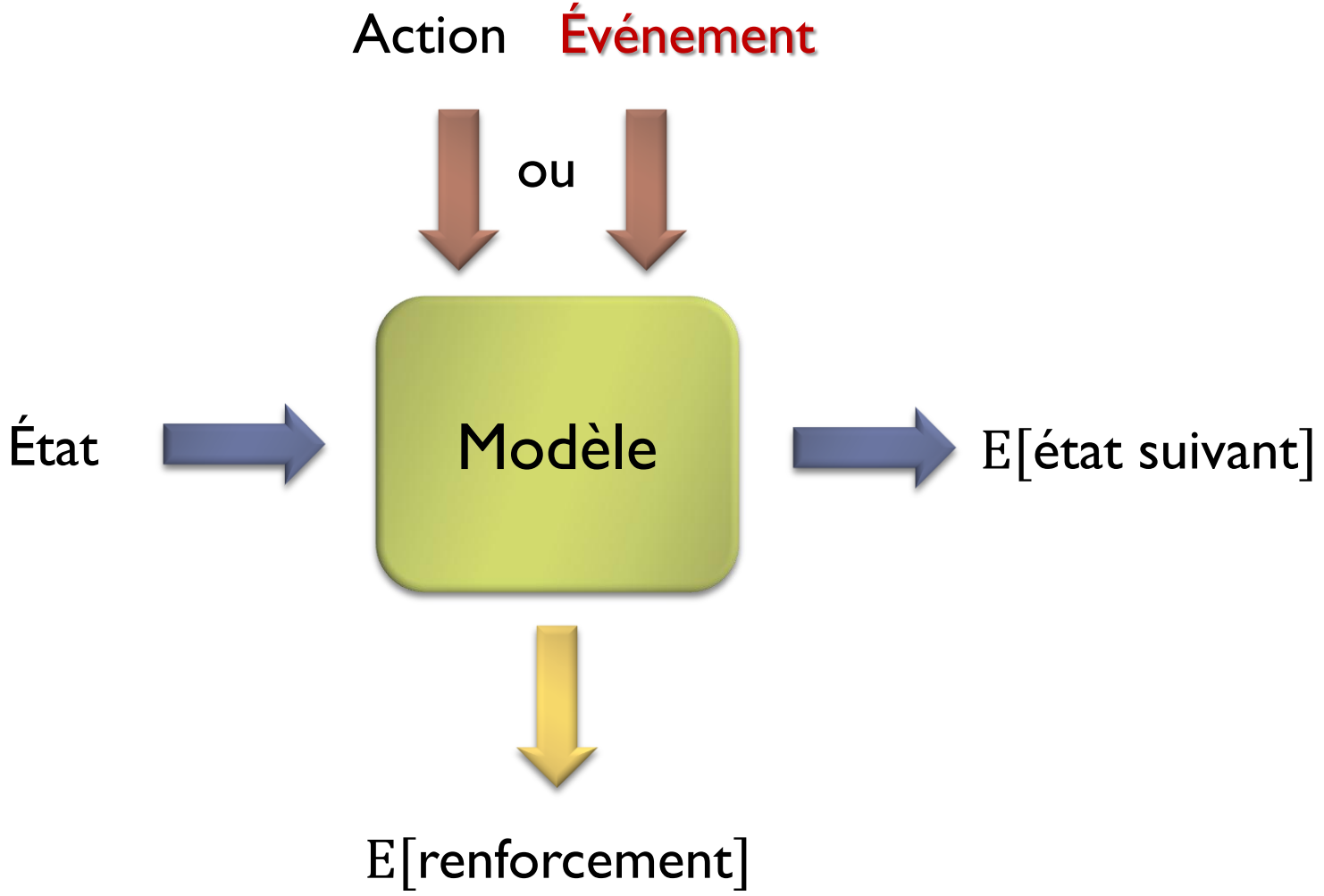
Service OMiSCID



Définition d'un état

Prédicat	Arguments
alarm	title, hour, minute
xActivity	machine, isActive
inOffice	user, office
absent	user
hasUnreadMail	from, to, subject, body
entrance	isAlone, friendlyName, btAddress
exit	isAlone, friendlyName, btAddress
task	taskName
user	login
userOffice	office, login
userMachine	machine, login
computerState	machine, isScreenLocked, isMusicPaused

Modèle de l'environnement



Réduction de l'espace d'états

- Accélération de l'apprentissage
  - Factorisation d'états

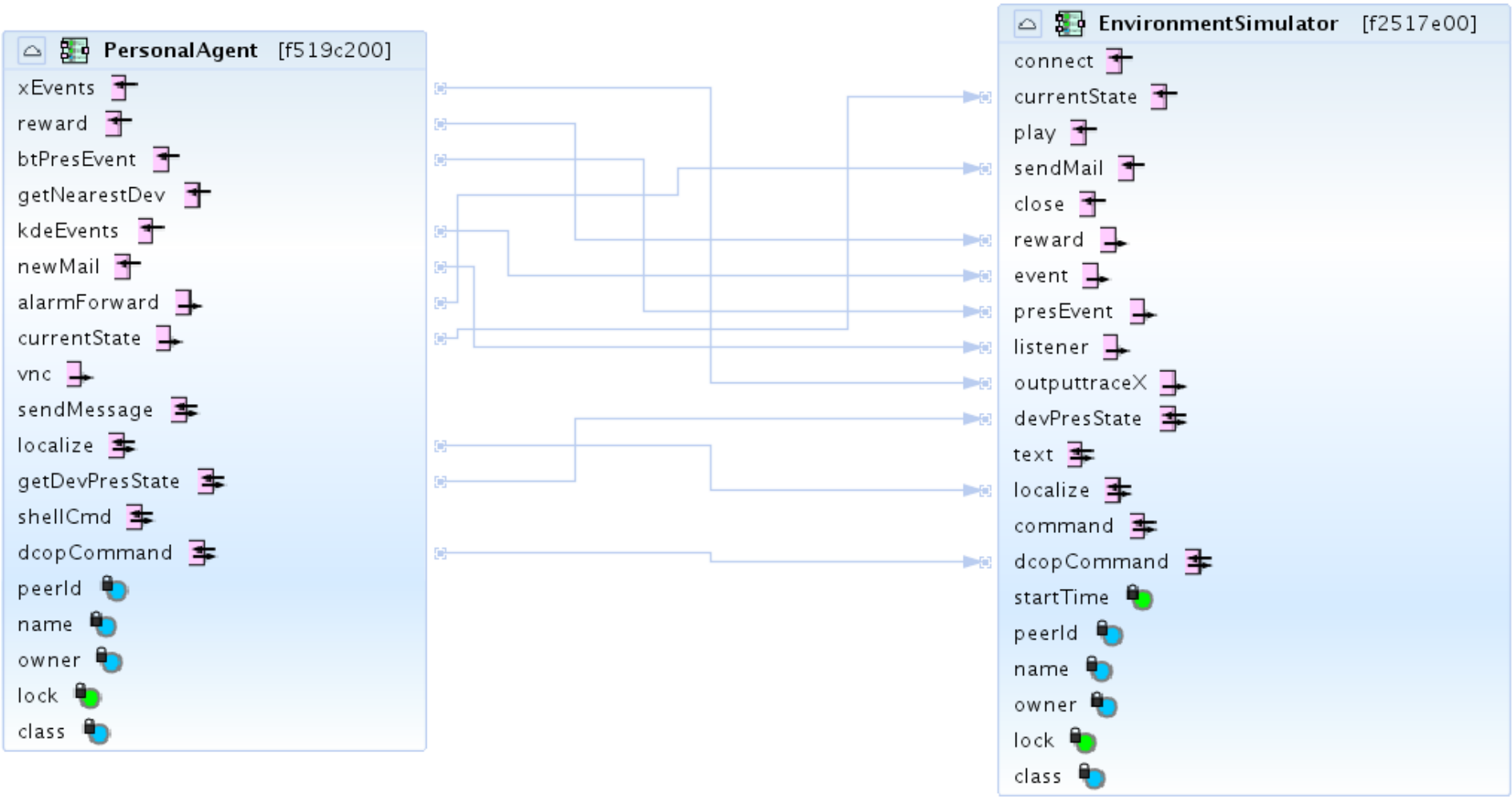
Jokers
<*> et <+>

État	Action	Q-valeur
…entrance(isAlone=true, friendlyName=**<+>**, btAddress=**<+>**)…	pauseMusic	125.3

  - Division d'états

État	Action	Q-valeur
…hasUnreadMail(from=**boss**, to=<+>, subject=<+>, body=<+>)…	inform	144.02
…hasUnreadMail(from=**newsletter**, to=<+>, subject=<+>, body=<+>)…	notInform	105

Le simulateur de l'environnement



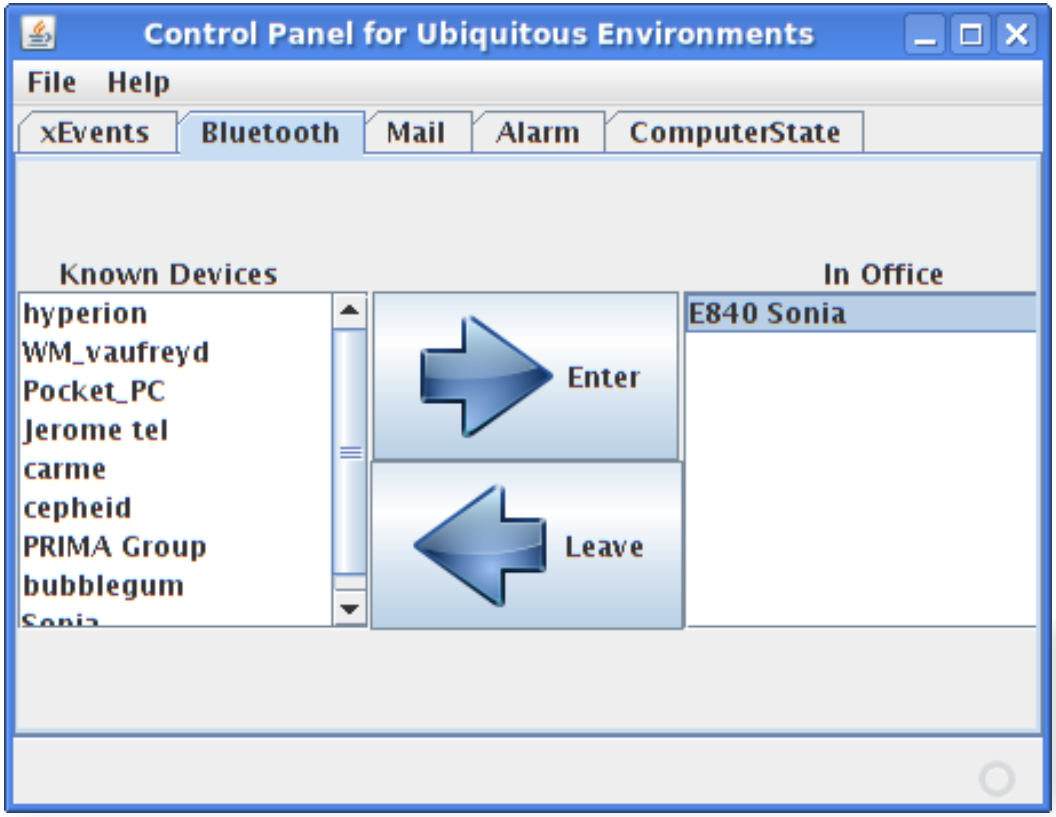
Critère d'évaluation : la note

- Résultat de l'AR : une Q-table
- Comment savoir si elle est « bonne » ?
- Apprentissage réussi si
  - Comportement correspond aux souhaits de l'utilisateur
  - Et c'est mieux si on a beaucoup exploré et si on a une estimation du comportement dans beaucoup d'états

$$note = \frac{1}{13}(10 \times n_{correct} + 2 \times p_{nonNul} + n_{total})$$

« Le tableau de bord »

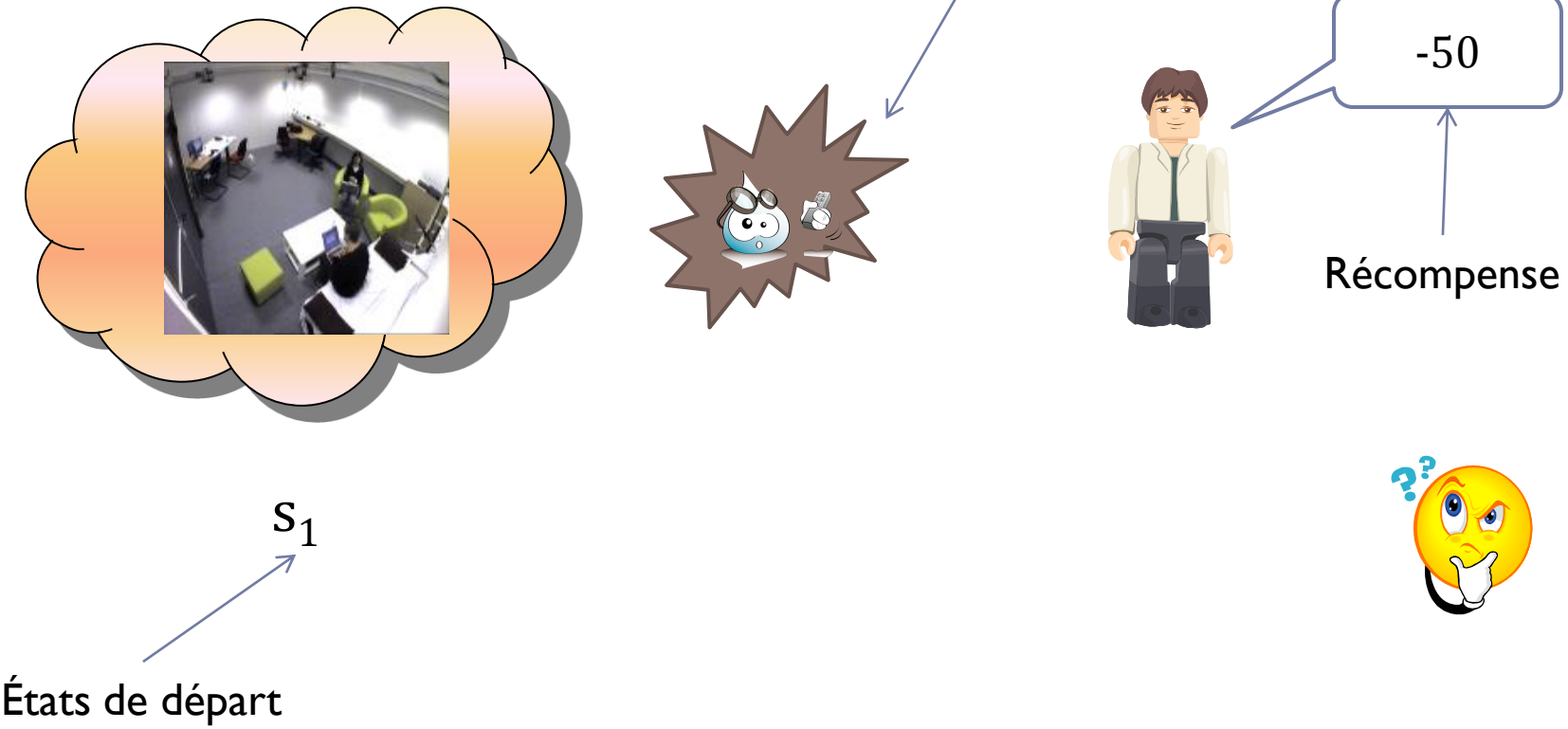
- Permet d'envoyer par un clic les mêmes événements que les capteurs



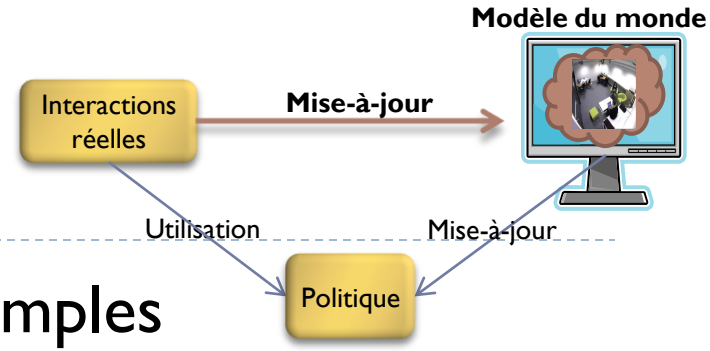
Modèle de récompense

- Ensemble d'entrées spécifiant
  - Des contraintes sur certains arguments de l'état
  - Une action
  - La récompense

Modèle de récompense



Apprentissage supervisé du modèle de récompense



- La base de données contient des exemples
  {état précédent, action, récompense}