Modélisation de la subjectivité et des biais cognitifs pour un agent conversationnel assistant réaliste

François Bouchet, Jean-Paul Sansonnet

LIMSI-CNRS
Université Paris-Sud XI

17 Novembre 2009

Journée du GT ACA

Outline

Outline

Assisting Conversational Agents

Issues of assistance to novice users

- "Paradox of motivation" *(Carroll & Rosson, 1987)*
- Users prefer the help provided by "a friend behind the shoulder" *(Capobianco & Carbonell, 2001)*

A solution: conversational agents for assistance

- "Persona Effect" : an animated agent increases credibility *(Lester, 1997)*
- Natural language : ideal modality when facing cognitive distress *(Carbonell, 2003)*

Realistic Assisting Conversational Agents

To be used, must look like "the friend behind the shoulder" :

- Embodiment : movements, emotions rendering...
  $\longrightarrow$ suitable with its visual realism
  – or risks to fall into the "Uncanny valley" *(Mori, 1970)*

- Cognitively : coherent personality, credible reactions to requests...
  $\longrightarrow$ suitable with its embodiment
  – or risks to reproduce the "Clippy Effect" *(Xiao et al., 2004)*

Related works

- CoJACK : addition of human physiological constraints to JACK agents *(Evertsz et al., 2008)*
- Extra parameters to BDI architectures : to help to model emotional behaviors – fundamental desires, capacities, resources... *(Pereira et al., 2008)*
- Heuristics order : impacts the perception of high level personnality traits *(Dastani, 2002)*

Personality: realism in the behavior expression

Issue: purely rational reasoning and factual answers aren't enough

- Not context-sensitive
  $\longrightarrow$ lack of competence
- No personality
  $\longrightarrow$ lack of realism (compared to user's expectations)
  $\longrightarrow$ lack of consistency (compared to previous reactions)

"How to quit this application?"

Solution : a model of personality

Personality: realism in the behavior expression

Issue: purely rational reasoning and factual answers aren't enough

"How to quit this application?"

- *Neutral* : "Click on the cross in the top right-hand corner"
- *Surprise* : "But the task isn't over!" (pragmatics)
- *Sadness* : "Why leaving me?" (previous interactions)
- *Pleasure* : "Good riddance!" (previous interactions)

Always interpreted in terms of personality *(Reeves & Nass, 1996)*.

Solution : a model of personality

Personality: realism in the behavior expression

Issue: purely rational reasoning and factual answers aren't enough

"How to quit this application?"

Solution : a model of personality

- static parameters : to keep consistency
- dynamic parameters : evolving according to interactions

Cognitive biases: realism in the choice of behavior

Issues

- decisions always intentional : the agent can explain them
- personality factors as an additional layer : the agent can inhibit them
  - accidentally : rules interferences
  - willingly : if self-monitoring to optimize its global behavior

Solution

Particular rules :

- hidden : applied outside from the main engine, agent can't introspect them
- destructive : applied transparently, the original message isn't available anymore

Outline

Actors

Agent $\mathcal{A}$

$\mathcal{A} = <\mathcal{E}, \mathcal{M}, \Psi>$:

- $\mathcal{E}$: set of *agent's engines*, actively processing requests.
- $\mathcal{M}$: set of *agent's memories*, storing knowledge of the agent (learnt or original).
- Ψ: set of *agent's mental states*, psychological parameters.

Interacts with the external *world* $\mathcal{W}$ = users + application.

Information

$\mathcal{W}$, $\mathcal{M}$ and Ψ store information as *entities*.

Entity

Triple associated to an identifier:

$$\#id = H\left[\bigcup_i a_i \rightarrow v_i\right]$$

- $\#id$: identifier
- H: head
- a_i: attribute restricted by H
- v_i: value restricted by a_i: terminal value, other entity, existing identity (identifier)

Communication

- external: $\mathcal{A} \leftrightarrow \mathcal{W}$
- internal: $\mathcal{E} \leftrightarrow \mathcal{M}$ and $\mathcal{E} \leftrightarrow \Psi$

Handled through *messages*.

Message

Requests sent between or within actors:

- `INFORM[recipient, request]`: transmits request, expects nothing in return
- `GET[recipient, value]`: asks value, expects an `INFORM[sender,X]` in return
- `CHECK[recipient, attribute, value]`: asks if the value sent is the one of the attribute, expects `INFORM[sender,T|F|?]`

World $\mathcal{W}$

Definition

Set of entities providing an "objective" description.

Information about a user

```
#user7 = PERSON[
    name   -> "Smith",
    role   -> user,
    age    -> 20,
    gender -> male
]
```

Agent's mental states Ψ

Definition

Psychology of the agent, modeled according to four types taking value in:
$[-1, 1]$ (0 = neutral) or [vlow, low, neutral, high, vhigh]

	Unary	**Binary**
Static	Trait Ψ_T	Role Ψ_R
Dynamic	Mood Ψ_t	Relationship Ψ_r

Agent's mental states – Traits Ψ_T

Definition

Classical "Big Five" *(Goldberg, 1981)* defining the personality

- *Openness*: appreciation for adventure, curiosity
- *Conscientiousness*: self-discipline and achieves goals
- *Extraversion*: strong positive emotions and sociability
- *Agreeableness*: compassion and cooperativeness
- *Neuroticism*: experience negative emotions easily

	1	2
Stat.	Ψ_T	Ψ_R
Dyn.	Ψ_t	Ψ_r

Unary mental state encoding

```
traits[
   openness -> -0.2,
   conscientiousness -> 0.7,
...]
```

Agent's mental states – Moods Ψ_t

Definition

Personality factors changed in time by heuristics and biases

- *Energy*: physical strength
- *Happiness*: physical contentment regarding the situation
- *Confidence*: cognitive strength
- *Satisfaction*: cognitive contentment regarding the situation

	1	2
Stat.	Ψ_T	Ψ_R
Dyn.	Ψ_t	Ψ_r

Agent's mental states – Roles Ψ_R

Definition

Static relationship between the agent and another entity of the world (*e.g.* users)

- *Authority*: right to be directive to X and reciprocally to not accept directive behaviors from X.
  Antisymmetric: `Authority(X,Y) = -Authority(Y,X)`
- *Familiarity*: right to use informal behaviors towards X.
  Symmetric: `Familiarity(X,Y) = Familiarity(Y,X)`

	1	2
Stat.	Ψ_T	Ψ_R
Dyn.	Ψ_t	Ψ_r

Binary mental state encoding

```
roles[
  towards     -> #iduser,
  authority   -> val1,
  familiarity -> val2]
```

Agent's mental states – Relationships Ψ_r

Definition

Dynamic relationships between the agent and another entity (*e.g.* users)

- *Dominance* : power felt towards X.
  Antisymmetric: `Dominance(X,Y) = -Dominance(Y,X)`
- *Affection* : attraction and tendency to be nice to X.
  Not necessarily symmetric.
- *Trust* : feeling one can rely on X.
  Not necessarily symmetric.

	1	2
Stat.	Ψ_T	Ψ_R
Dyn.	Ψ_t	Ψ_r

Agent's memory $\mathcal{M}$

Definition

Stores knowledge learnt through interaction or that the agent originally had.

Content

1. Semantic memory $\mathcal{M}_s$: agent's vision of the world, observed (direct) or created through introspection (indirect).
2. Episodic memory $\mathcal{M}_e$: focused on the agent *i.e.* autobiographical memory *(Tulving, 1983)*.
3. Procedural memory $\mathcal{M}_p$: set of heuristics, *i.e.* rules to apply in some given situations, defining the reactions.

Semantic memory $\mathcal{M}_s$

Definition

Extended subset of the world:

- subset: whole world not available to $\mathcal{A}$ and pieces of information possibly out-dated.
- extended: new facts available through reasoning over the memory content.

World $\mathcal{W}$

```
#object9 = OBJECT[
   name    -> "btnValid2",
   type    -> button,
   label   -> "OK",
   color   -> green
]
```

Semantic memory $\mathcal{M}_s$

```
#object3 = OBJECT[
   type    -> button,
   label   -> "OK",
   color   -> green,
   trigger-> accept();
]
```

Episodic memory $\mathcal{M}_e$

Definition

Set of previous interactions of the agent with the user and the application, distinguishing incoming (INBOX) from outgoing messages (OUTBOX).

INBOX/OUTBOX

```
INBOX[
   from -> [sender],
   time -> [timestamp],
   message -> [message]
]
```

```
OUTBOX[
   to -> [recipient],
   time -> [timestamp],
   message -> [message]
]
```

Procedural memory $\mathcal{M}_p$

Definition

Set of heuristics defining the reaction to an incoming request.

Heuristic

Associates a set of actions to a situation:

- head: regular expression defining classes of requests.
- body: decision tree, where nodes send messages to $\mathcal{M}$ and $\mathcal{W}$ (rationality) or to $\mathcal{M}_s$ (subjectivity). At the end, an answer request is sent to $\mathcal{W}$.

Heuristic example

Forbidden action

```
if conscientiousness ≥ neutral then
    allow ← CHECK[rep, DOABLE[A], true]
end if
if allow = false then
    if agreeableness ≥ neutral then
        if affection(user) ≥ neutral &
        familiarity(user) ≥ neutral then
            ans +← POS[NOTPOSSIBLE[A]];
        else if affection(user) ≤ vlow then
            ans +← NEG[NOTPOSSIBLE[A]];
        else
            ans +← NOTPOSSIBLE[A];
        end if
    end if
end if
if authority(user) ≥ neutral then
    req +← INFORM[memory, forbidden(A)]
    done ← true
else
    done ← false
end if
```

– sequel –

```
if neuroticism ≥ neutral then
    req +← INFORM[memory,
    decrease(satisfaction)]
    if dominance(user) ≥ neutral then
        ans +← UNHAPPY
    end if
end if
if satisfaction ≤ low & familiarity(user) ≥
neutral then
    ans +← NEG[(done?ACK:NACK)]
else if done & satisfaction ≤ vlow then
    ans +← NEG[(done?ACK:NACK)]
else
    ans +← (done?ACK:NACK)
end if

req +← INFORM[user, answer]
return req
```

Output: [not possible] [unhappy] [ack/nack]

Heuristic example

Forbidden action

```
if conscientiousness ≥ neutral then
    allow ← CHECK[rep, DOABLE[A], true]
end if
if allow = false then
    if agreeableness ≥ neutral then
        if affection(user) ≥ neutral &
        familiarity(user) ≥ neutral then
            ans +← POS[NOTPOSSIBLE[A]];
        else if affection(user) ≤ vlow then
            ans +← NEG[NOTPOSSIBLE[A]];
        else
            ans +← NOTPOSSIBLE[A];
        end if
    end if
end if
if authority(user) ≥ neutral then
    req +← INFORM[memory, forbidden(A)]
    done ← true
else
    done ← false
end if
```

– sequel –

```
if neuroticism ≥ neutral then
    req +← INFORM[memory,
    decrease(satisfaction)]
    if dominance(user) ≥ neutral then
        ans +← UNHAPPY
    end if
end if
if satisfaction ≤ low & familiarity(user) ≥
neutral then
    ans +← NEG[(done?ACK:NACK)]
else if done & satisfaction ≤ vlow then
    ans +← NEG[(done?ACK:NACK)]
else
    ans +← (done?ACK:NACK)
end if

req +← INFORM[user, answer]
return req
```

Output: [not possible] [unhappy] [ack/nack]

Heuristic example

Forbidden action

```
if conscientiousness ≥ neutral then
    allow ← CHECK[rep, DOABLE[A], true]
end if
if allow = false then
    if agreeableness ≥ neutral then
        if affection(user) ≥ neutral &
        familiarity(user) ≥ neutral then
            ans +← POS[NOTPOSSIBLE[A]];
        else if affection(user) ≤ vlow then
            ans +← NEG[NOTPOSSIBLE[A]];
        else
            ans +← NOTPOSSIBLE[A];
        end if
    end if
end if
if authority(user) ≥ neutral then
    req +← INFORM[memory, forbidden(A)]
    done ← true
else
    done ← false
end if
```

– sequel –

```
if neuroticism ≥ neutral then
    req +← INFORM[memory,
    decrease(satisfaction)]
    if dominance(user) ≥ neutral then
        ans +← UNHAPPY
    end if
end if
if satisfaction ≤ low & familiarity(user) ≥
neutral then
    ans +← NEG[(done?ACK:NACK)]
else if done & satisfaction ≤ vlow then
    ans +← NEG[(done?ACK:NACK)]
else
    ans +← (done?ACK:NACK)
end if

req +← INFORM[user, answer]
return req
```

Output: [not possible] [unhappy] [ack/nack]

Heuristic example

Forbidden action

```
if conscientiousness ≥ neutral then
    allow ← CHECK[rep, DOABLE[A], true]
end if
if allow = false then
    if agreeableness ≥ neutral then
        if affection(user) ≥ neutral &
        familiarity(user) ≥ neutral then
            ans +← POS[NOTPOSSIBLE[A]];
        else if affection(user) ≤ vlow then
            ans +← NEG[NOTPOSSIBLE[A]];
        else
            ans +← NOTPOSSIBLE[A];
        end if
    end if
end if
if authority(user) ≥ neutral then
    req +← INFORM[memory, forbidden(A)]
    done ← true
else
    done ← false
end if
```

– sequel –

```
if neuroticism ≥ neutral then
    req +← INFORM[memory,
    decrease(satisfaction)]
    if dominance(user) ≥ neutral then
        ans +← UNHAPPY
    end if
end if
if satisfaction ≤ low & familiarity(user) ≥
neutral then
    ans +← NEG[(done?ACK:NACK)]
else if done & satisfaction ≤ vlow then
    ans +← NEG[(done?ACK:NACK)]
else
    ans +← (done?ACK:NACK)
end if

req +← INFORM[user, answer]
return req
```

Output: [not possible] [unhappy] [ack/nack]

Engines $\mathcal{E}$

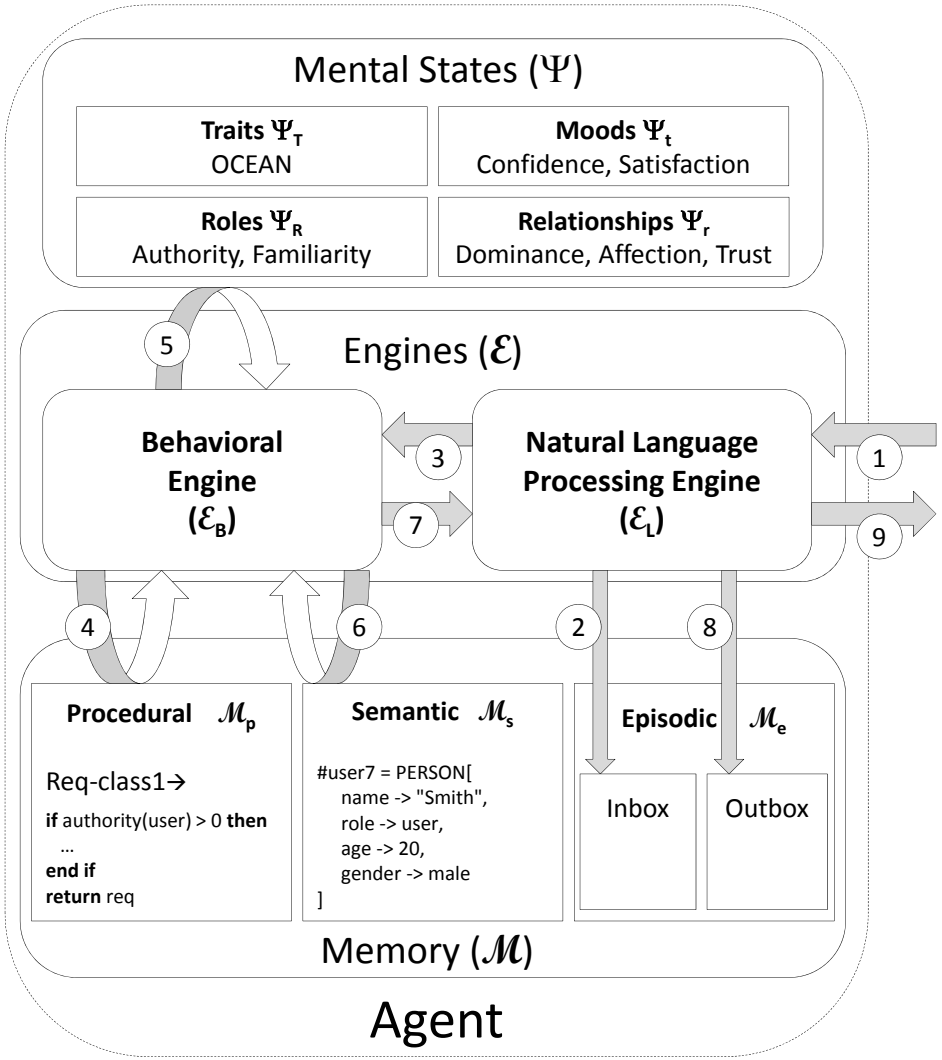
Natural Language Processing Engine $\mathcal{E}_L$

- *Grammatical* analysis: lemmatization, POS tagging, WSD...
- *Semantic* analysis: production of a formal request *(Bouchet & Sansonnet, 2007)*.

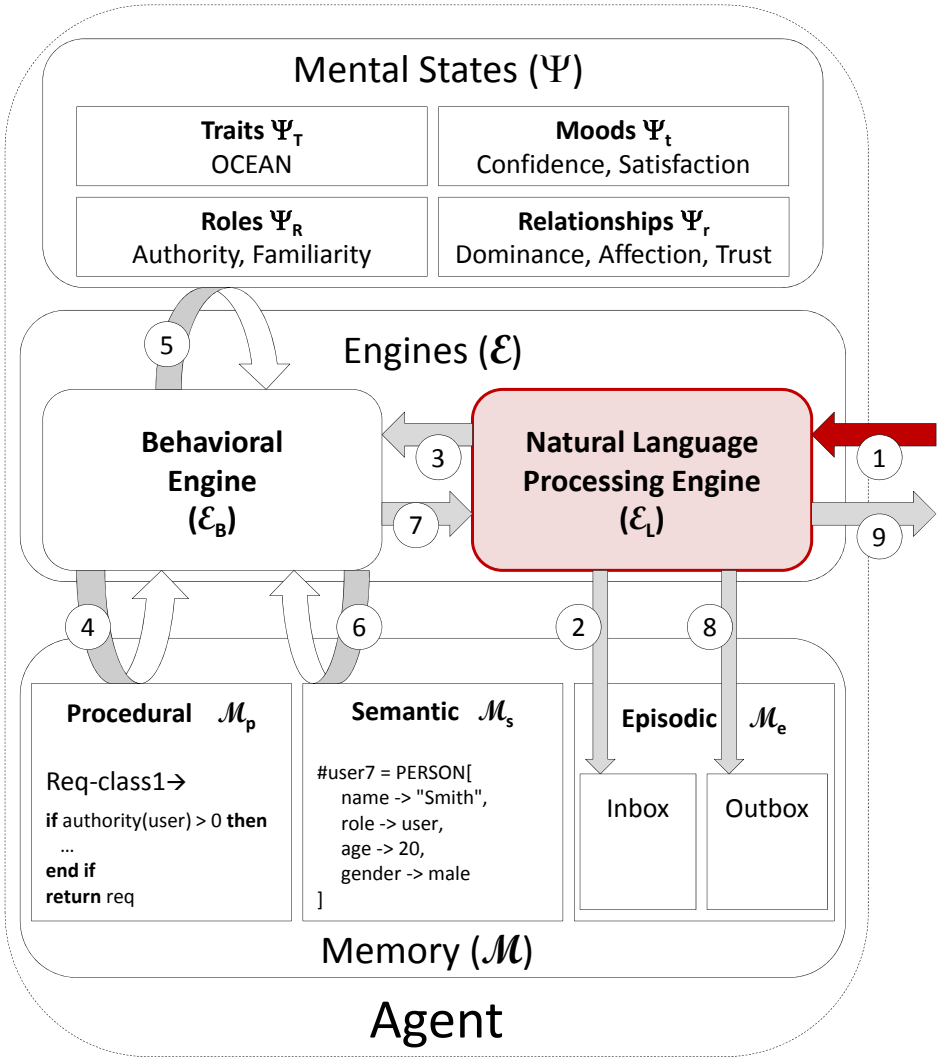
Behavioral Engine $\mathcal{E}_B$

- Centralizes the reception and sending of messages
- Chooses heuristics (from $\mathcal{M}_p$) to be applied
- Computes the reactions from heuristics according to current values of $\mathcal{M}_s$ and Ψ

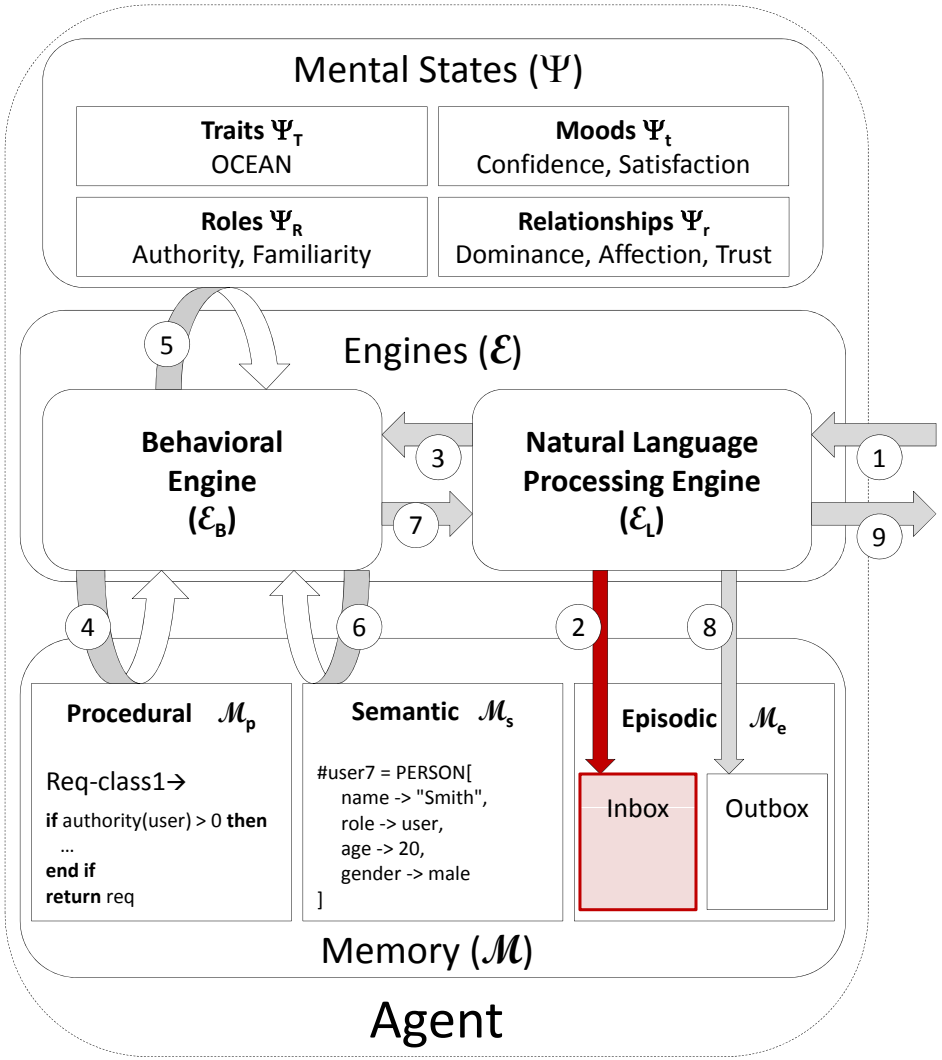
Dynamic functioning



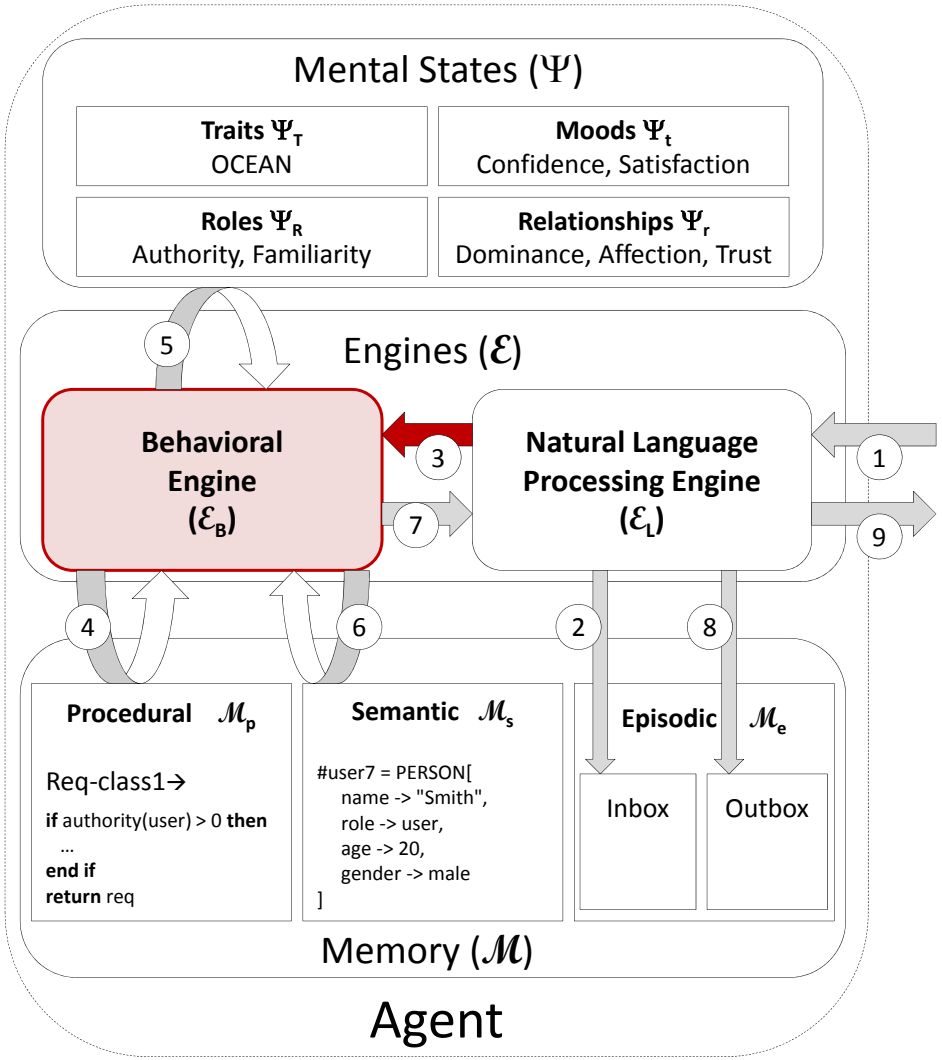
Dynamic functioning



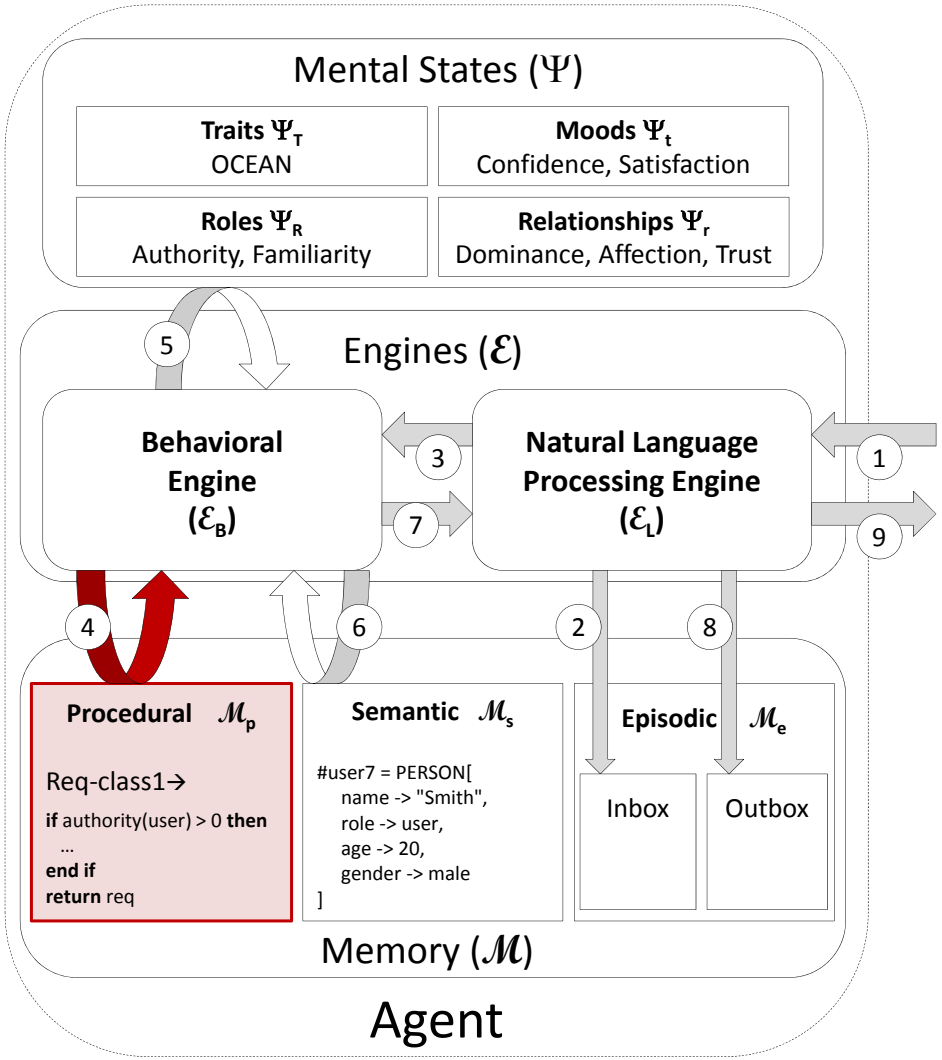
Dynamic functioning



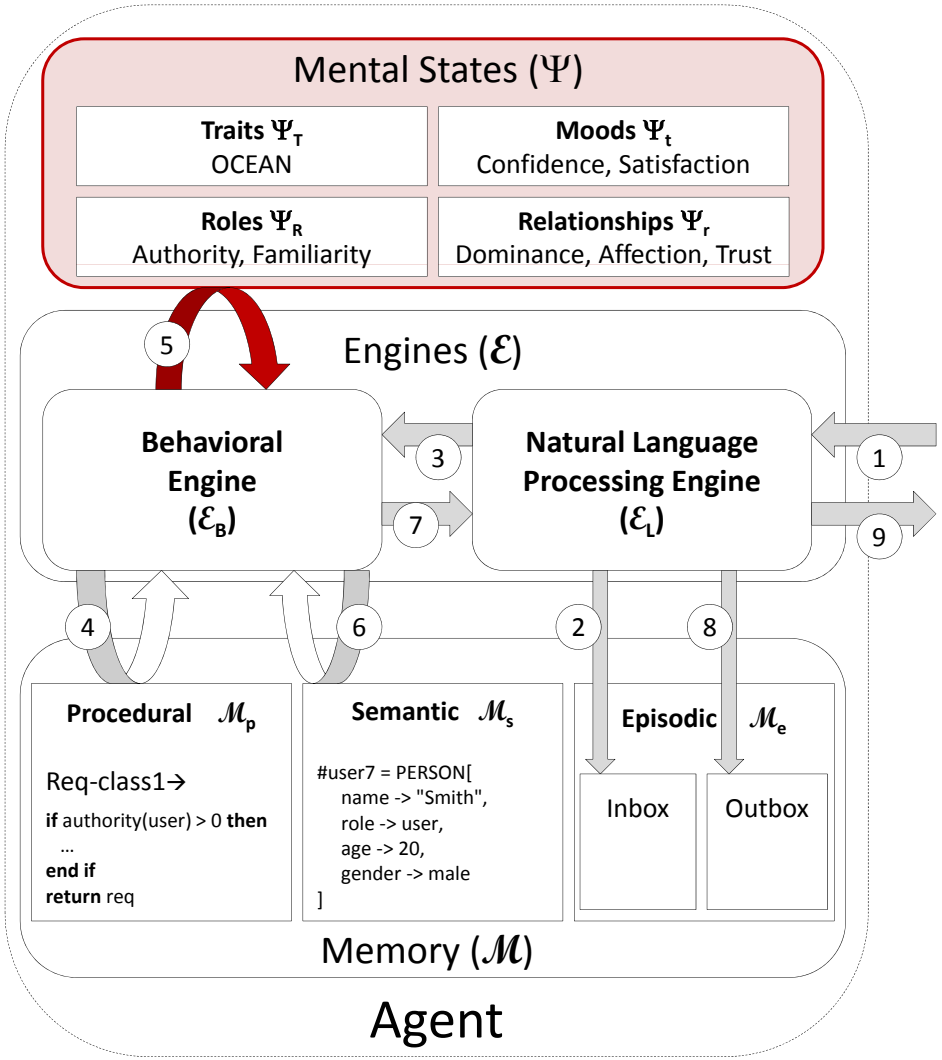
Dynamic functioning



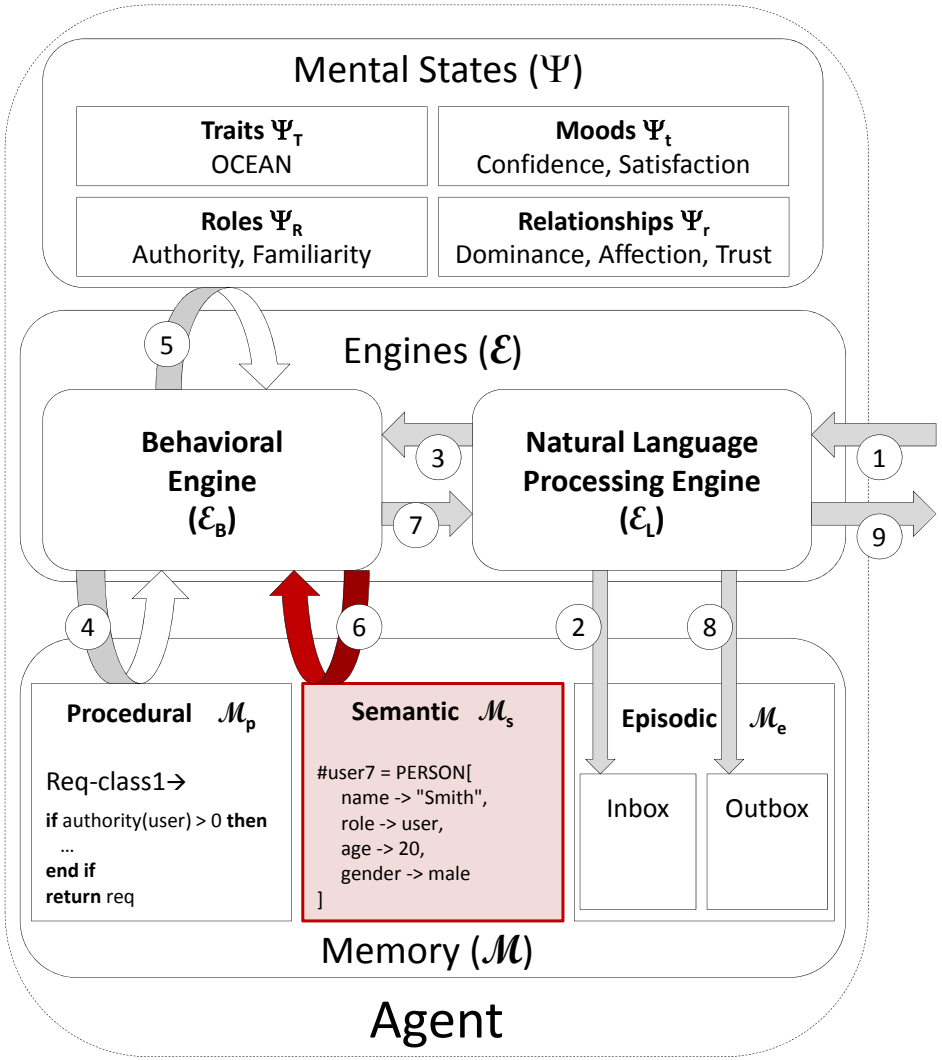
Dynamic functioning



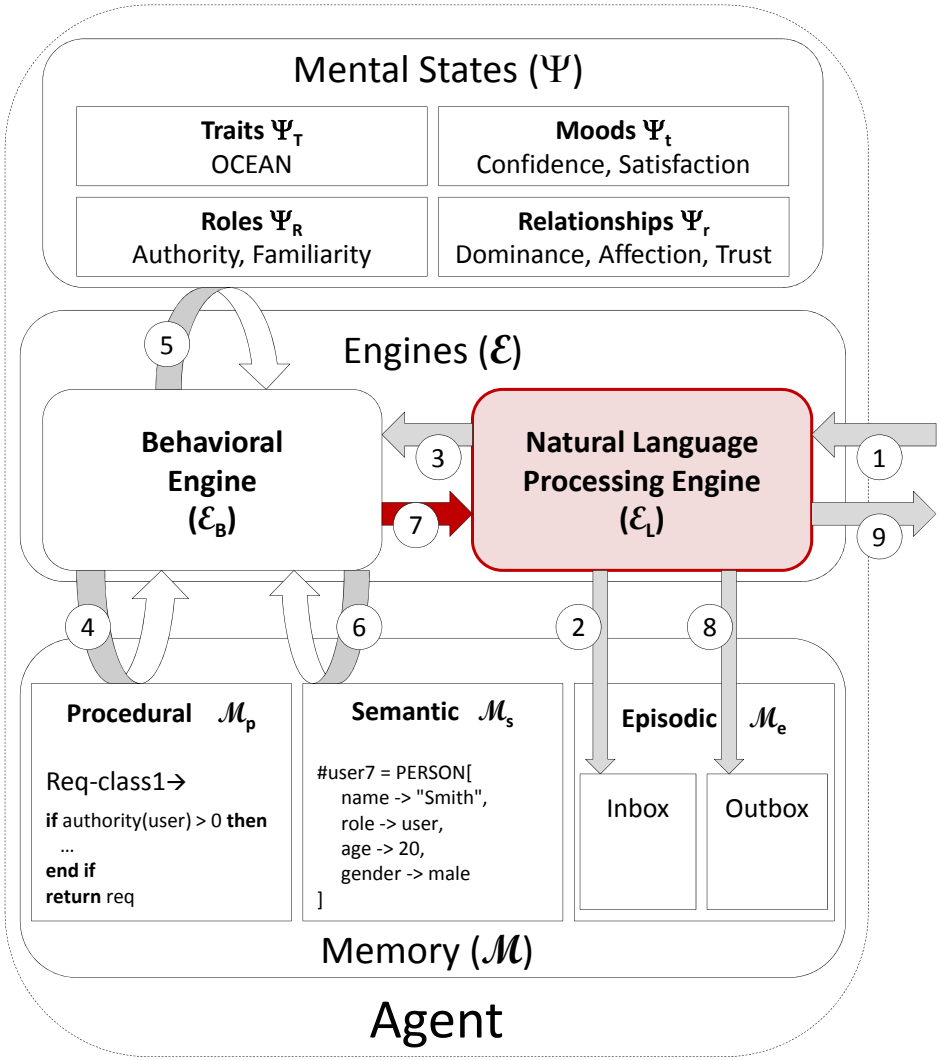
Dynamic functioning



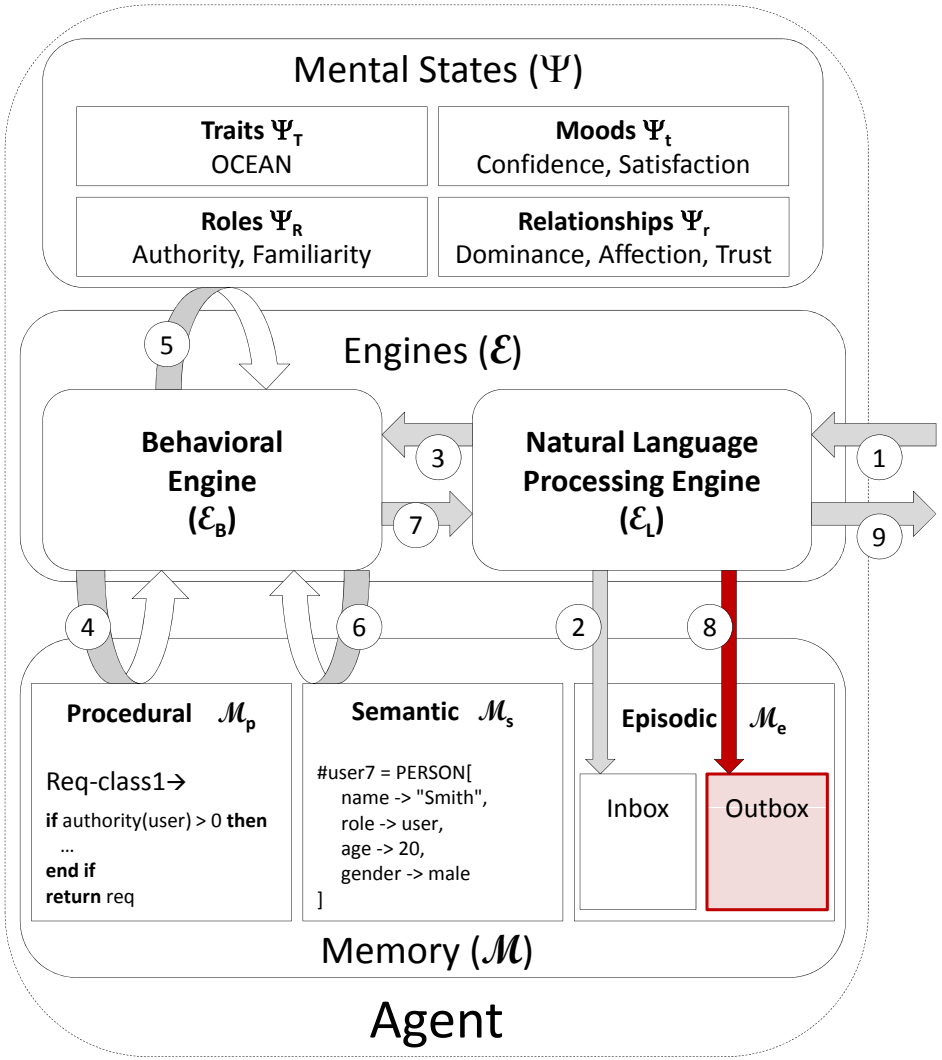
Dynamic functioning



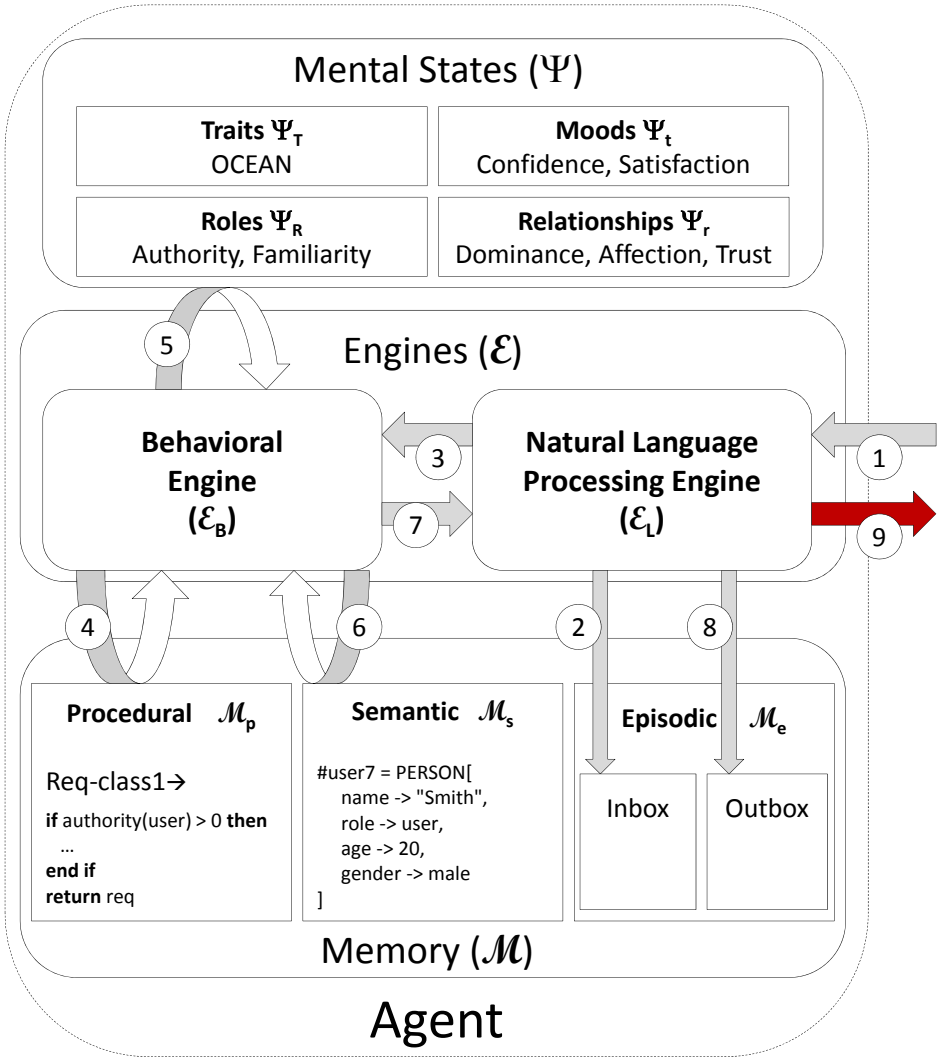
Dynamic functioning



Dynamic functioning



Dynamic functioning



Outline

Cognitive bias concept

Definition

A bias is a transformation rule over messages sent by the agent (within itself or to the world), without the agent's knowledge.

A bias b on a message between X and Y: $X \xrightarrow{b} Y$.

Comparing with heuristics

- Objective: *modifying* a message
- Impact: *any* request between X and Y
- Factors used: Ψ *only*
- Introspection: *impossible*

Representing biases

Formal definition

Same structure as heuristics:

- head: category of the bias.
- body: decisions tree to modify the request, according to Ψ.

Biased perception of a nervous and unhappy agent

```
BIAS[
   description -> "victimization",
   category -> "perceptive"
   body -> {
   if (neuroticism <= low && satisfaction <= vlow):
      output = NEGATIVE[input]
}]
```

Biases categories

Possible channels

- 4 elements ($\mathcal{M}$, Ψ, $\mathcal{E}$, $\mathcal{W}$) $\Rightarrow$ 6 channels
- Bidirectional channels
- 3 types of messages (INFORM, GET, CHECK)

$\Rightarrow 6 \times 2 \times 3 = 36$ biases possible in theory

Restrictions on channels

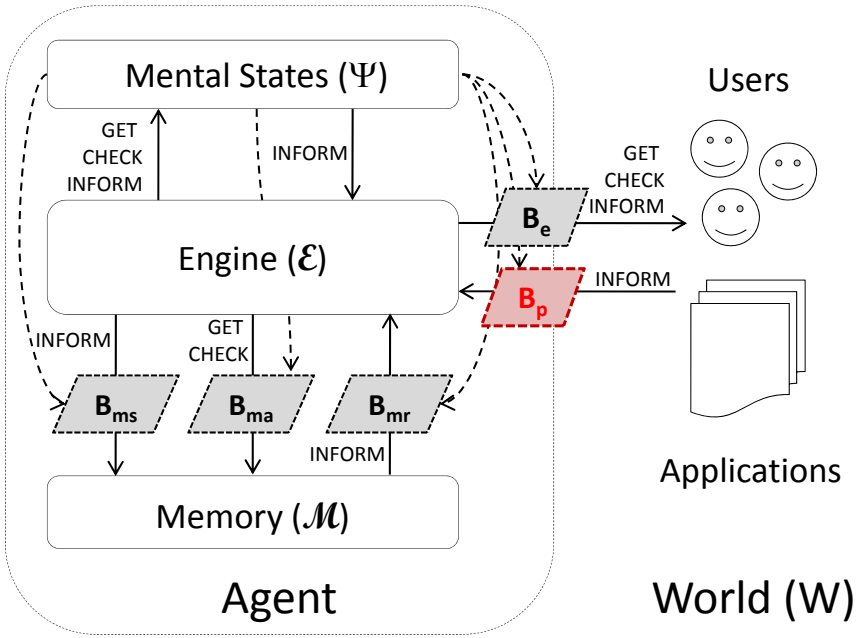
- $\mathcal{E}_B$ is the core of communication of $\mathcal{A}$: it's the only one able to send messages
- Every message isn't relevant for each channel
- The agent can always know its mental states Ψ

$\Rightarrow$ *5 types of biases* left on 7 channels

Perceptive bias $\mathcal{W} \xrightarrow{B_p} \mathcal{E}_B$

Victimization: *cf.* previous example

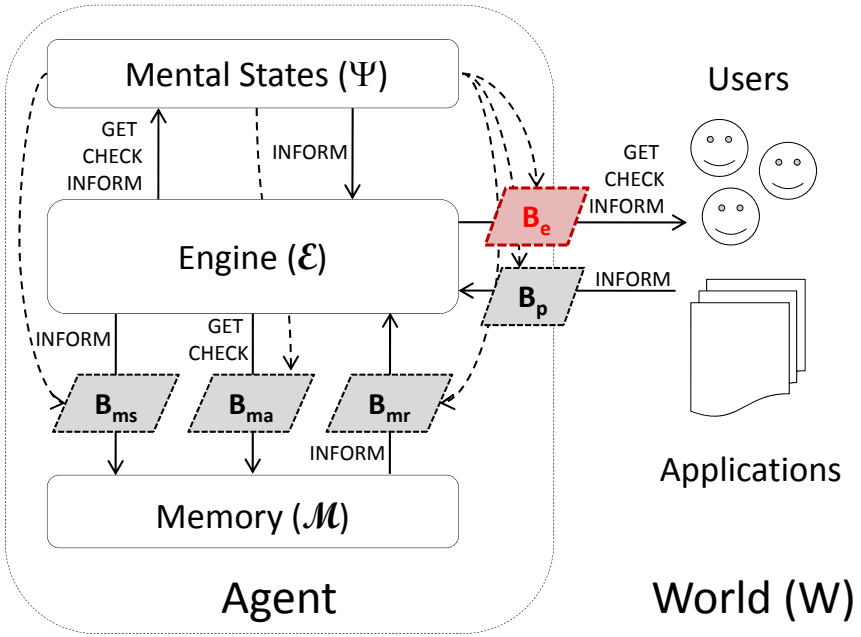
Minimization:
Condition:
`(satisfaction`$\geq$`high && neuroticism`$\leq$`neutral)`
Consequence: tend to ignore negativity in user's NL request.



Expressive bias $\mathcal{E}_B \xrightarrow{B_e} \mathcal{W}$

Stress:
Condition:
$\texttt{authority(A,U)} \leq \texttt{low}$
Consequence: extra uncontrolable nervousness in the answer (independantly from the content of the request).

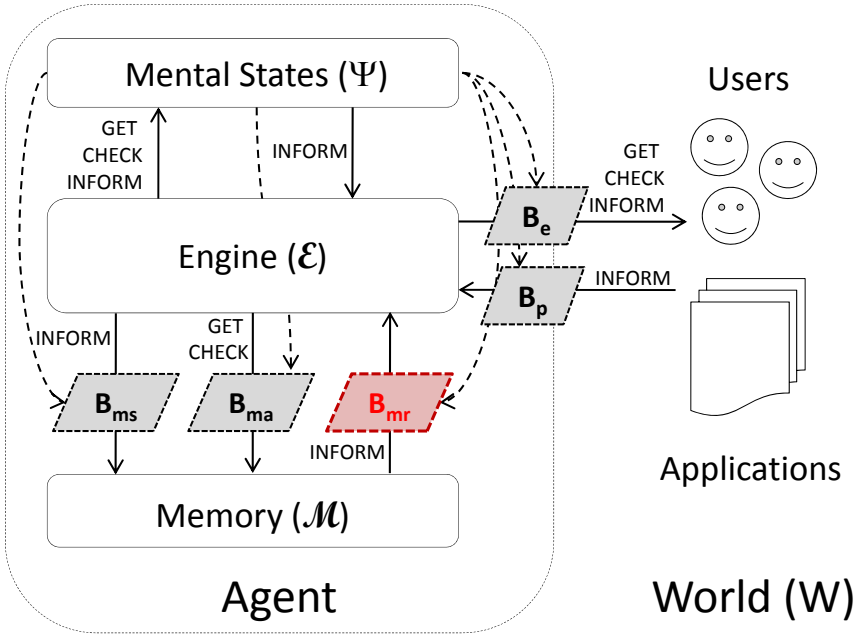
Cheeriness/gloominess:
Condition: $\texttt{extraversion} \geq \texttt{high}$ (resp. $\leq \texttt{low}$)
Consequence: adds positive (resp. negatives) connotations to the answer.



Memory retrieval bias $\mathcal{M} \xrightarrow{B_{mr}} \mathcal{E}_B$

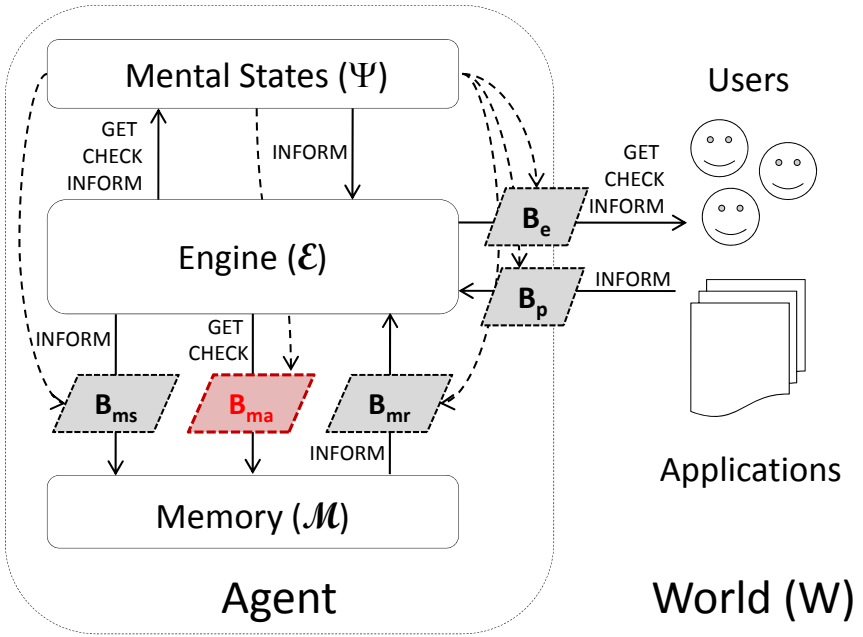
(while answering to a `GET` or `CHECK`)

Doubts:
Condition:
`trust(agent,agent)`$\leq$`neutral && satisfaction`$\leq$`low`
Consequence: Discards or lowers the confidence of the facts retrieved from its memory.



Memory access $\mathcal{E}_B \xrightarrow{B_{ma}} \mathcal{M}$

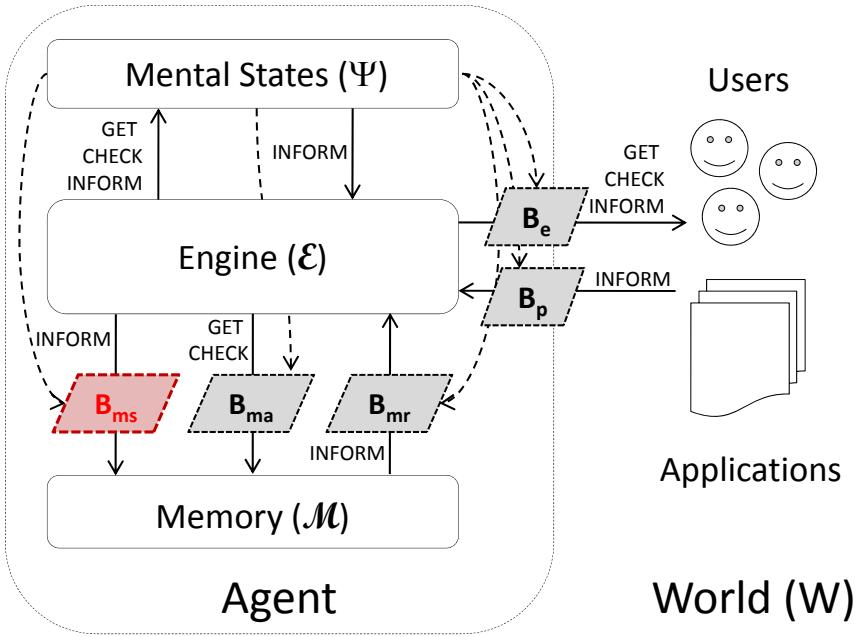
Bad faith:
Condition: `satisfaction`$\leq$`vlow` && `authority(U,A)`$\geq$`high`
Consequence: Introduce mistakes (*e.g.* forgetting a parameter) in messages to $\mathcal{M}_s$. Agent is unaware to have done something else than what it was asked for.



Memory storage bias $\mathcal{E}_B \xrightarrow{B_{ms}} \mathcal{M}$

Tolerance:
Condition: `satisfaction`$\geq$`high && neuroticism`$\leq$`neutral`
Consequence: Do not remember negative comments from the user on the long term (*e.g.* criticisms towards the agent)

Scatterbrain:
Condition: `conscientious`$\leq$`low`
Consequence: Randomly forget to store some messages said or received into $\mathcal{M}_e$: they are lost forever.



Outline

Summary

- Subjectivity allows to design agents adapting their assistance:
  - a priori: to have a personality Ψ_T chosen according to the user's one.
  - dynamically: according to the user's behavior which has modified its mental state (Ψ_t and Ψ_r).
- Cognitive biases allow to mimic human cognitive constraints and give primacy to emotions over rationality.

Perspectives

Evaluating novice users interacting with:

1. a purely rational agent
2. a rational and subjective agent
3. a rational, subjective and biased agent

Expected results:

- realism: $1 < 2 < 3$
- efficiency: $1 \leq 2$ and probably $3 \leq 2$

But which assisting agent would be the most used? The best rated overall?