# Multi-Agent Systems

## Speech Acts

Albert-Ludwigs-Universität Freiburg

UNI FREIBURG

Bernhard Nebel, Rolf Bergdoll, and Thorsten Engesser

Winter Term 2019/20

# Typology of agent interaction

- Non-/Quasi-communicative interaction:
  - Shared environment (interaction via resource/capability sharing)
  - "Pheromone" communication (e.g., ant algorithms)
- Knowledge-level Communication:
  - Information exchange: sharing knowledge, exchanging views and plans
  - Collaboration, distributed planning: optimising use of resources and distribution of tasks, coordinating execution
  - Negotiation: reaching agreement in the presence of conflict
  - Human-machine dialogue, reporting errors, etc.

# OOP vs. AOP

- Objects vs. Agents:
  - “Objects do it for free; agents do it for money."(Jennings, Sycara, Wooldridge, 1998)
  - “Objects do it because they have to; agents because they want to."(Joseph, Kawamura, 2001)
- Communication
  - OOP: Object $o_2$ can communicate with $o_1$ by invoking $o_1$'s method $m1$ (i.e., $o_1.m1(arg)$). I a way, object $o_2$ makes the decision to make $o_1$ to perform some action while $o_1$ has no control about it.
  - AOP: There is no such thing as method invocation. Agent $i$ can try to act in a way to bring about that agent $j$ intends what $i$ intends, viz., by speech acts.

# Speech act theory

- Most multiagent approaches to communication based on speech act theory (started with Austin's book, *How to Do Things with Words*, 1962)
- Basic idea: treat communication in a similar way as non-communicative action
- Austin (1962): Utterances are produced, and may have effects, like "physical" actions: utterances may change the state of the world:
  - *I hereby name this ship H.M.S titanic!*
  - *Bring me some beer!*

    ⇒Not true or false but appropriate or inappropriate
- In MAS speech act theory is used to specify how utterances can be used to achieve intentions

# Speech act theory: Austin

- A speech act can be conceptualised to consist of:
  1. Locutionary act (physical utterance)
  2. Illocutionary act (intended meaning)
  3. Perlocution (effect of the act)

- Two parts of a speech act:
  1. Performative: communicative verb used to distinguish between different “illocutionary forces”
     Examples: promise, request, purport, insist, demand, etc.
  2. Propositional content: what the speech act is about

- Example:
  - Propositional content: “the window is open”
  - Performative: can be a request/inform/enquire

# Speech act theory: Searle

- Searle (1972) identified the following categories of performatives, each corresponding to a different type of speech acts:
  - assertives/representatives (informing, making a claim)
  - directives (requesting, commanding)
  - commissives (promising, refusing)
  - declaratives (effecting change to state of the world)
  - expressives (expressing mental states)

- Ambiguity problems: “The window is open.”
- Detection of performative for natural language understanding:
  - A: “Open the door!” (Directive: Request)
  - B: “Sure!” (Commissive: Promise)

$\Rightarrow$Different types of speech acts require different conditions to hold to be meaningful.

# Speech act theory

Specification of speech acts in terms of the conditions under which they can be successfully completed.

Austin's felicity conditions for *declarations*:

1. Conventionality of procedure: There must be an accepted conventional procedure for the performative
2. Complete execution: The procedure must be executed correctly and completely
3. Sincerity and context: the speaker must be sincere and the context must be appropriate errors or interruptions

Searle's properties for success of a *request*:

1. I/O conditions: ability to hear request
2. Preparatory conditions: requested action can be performed, speaker must believe this, hearer will not perform action anyway
3. Sincerity conditions: speaker wants action to be performed

# Speech acts: Plan-based semantics

- Communication as action: Preconditions and effects. Thus, we can ask for a plan of speech acts to achieve some goal.
- Cohen and Perrault (1979) proposed applying planning techniques to speech acts (STRIPS-style)
- Preconditions and effects describe beliefs, abilities and wants of participants
- Distinction between `can-do` and `want` preconditions
- Identified necessity of mediating acts, since speech acts say nothing about perlocutionary effect
- This has been the most influential approach to using communication in multiagent systems!

# Cohen & Perrault: Request

## Specfications

- Request$(s, h, \alpha)$
  - Cando.pr: $\text{Bel}_s\,\text{Can}_h\,\alpha \wedge \text{Bel}_s\,\text{Bel}_h\,\text{Can}_h\alpha$
  - Want.pr: $\text{Bel}_s\,\text{Want}_s$ *requestinstance*
  - Effect: $\text{Bel}_h\,\text{Bel}_s\,\text{Want}_s\alpha$
- CauseToWant$(a1, a2, \alpha)$
  - Cando.pr: $\text{Bel}_{a1}\,\text{Bel}_{a2}\,\text{Want}_{a2}\alpha$
  - Want.pr: –
  - Effect: $\text{Bel}_{a1}\,\text{Want}_{a1}\,\alpha$

- Guenther wants Jutta to get him some beer. He makes up the following plan:
  1. Request(guenther, jutta, getbeer)
  2. CauseToWant(jutta, guenther, getbeer)

# Cohen & Perrault: Inform

## Specfications

- Inform($s, h, \varphi$)
  - Cando.pr: $\text{Bel}_s \varphi$
  - Want.pr: $\text{Bel}_s\,\text{Want}_s$ *informinstance*
  - Effect: $\text{Bel}_h\,\text{Bel}_s\,\varphi$
- Convince($a1, a2, \alpha$)
  - Cando.pr: $\text{Bel}_{a1}\,\text{Bel}_{a2}\,\varphi$
  - Want.pr: –
  - Effect: $\text{Bel}_{a1}\,\varphi$

- Guenther wants Jutta to believe that the Britains have voted for Brexit. He makes up the following plan:
  1. Inform(guenther, jutta, brexitvoting)
  2. Convince(jutta, guenther, brexitvoting)

# Agent communication languages

- Agent communication languages (ACLs) define standards for messages exchanged among agents
- Usually based on speech act theory, messages are specified by:
  - Sender/receiver(s) of the message
  - Performative to describe intended actions
  - Propositional content in some content language
- Most commonly used languages:
  - KQML/KIF
  - FIPA-ACL (today the de-facto standard)

FIPA: Foundation for Intelligent Physical Agents
KQML: Knowledge Query and Manipulation Language
KIF: Knowledge Interchange Format

# KQML/KIF

- KQML: Knowledge Query and Manipulation Language
- . . . is an “outer” language, defines various performatives
- Example performatives:
  - `ask-if` (“is it true that . . . ”)
  - `perform` (“please do the following action . . . ”)
  - `tell` (“it is true that . . . ”)
  - `reply` (“the answer is . . . ”)
- Message format:

```
(performative
    :sender       <word>   :receiver    <word>
    :in-reply-to  <word>   :reply-with  <word>
    :language     <word>   :ontology    <word>
    :content      <expression>
)
```

# Example: Advertise a capability

```
(advertise
   :sender        Agent1
   :receiver      Agent2
   :in-reply-to   ID1
   :reply-with    ID2
   :language      KQML
   :ontology      kqml-ontology
   :content       (ask-if
                     :sender   Agent1
                     :receiver Agent3
                     :language Prolog
                     :ontology blocks-world
                     :content  "on(X,Y)"
                     )
    )
```

# KQML/KIF

- KQML is not suited as a general content language
- KIF (Knowledge Interchange Format): a logical language to describe knowledge
- … essentially first-order logic with some extensions/restrictions
- Examples:
  - `(forall (?x ?n) (=> (and (real-num ?x) (even-num ?n)) (> (expt ?x ?n) 0)))`
  - `(married guenther jutta)`
  - (forall (?x) (=> (human ?x) (exists ?y (and (amountOfWater ?y) (essentialFor ?y ?x)))))
- KIF can also be used to describe the ontology referred to by interacting agents.

# Example: Dialog

```
(evaluate
    :sender A             :receiver B
    :language KIF         :ontology motors
    :reply-with q1        :content (val (torque m1))
)
(reply
    :sender B             :receiver A
    :language KIF         :ontology motors
    :in-reply-to q1       :content (= (torque m1)
                                      (scalar 12 kgf.m))
)
```

# KQML/KIF

- KQML/KIF were very successful, but also some problems
- List of performatives (up to 41!) not fixed (interoperability problems)
- No formal semantics, only informal descriptions of meaning
- KQML completely lacks commissives, this is a massive restriction!
- Performative set of KQML rather ad hoc, not theoretically clear or very elegant

$\leadsto$ These lead to the development of FIPA ACL

# FIPA ACL

Basic structure is quite similar to KQML:

- performatives: fixed set of 20 performatives in FIPA
- housekeeping: e.g., sender, receiver, message IDs
- content: the actual content of the message

Example:

```
(inform
   :sender   agent1
   :receiver agent3
   :content  (price goodABC 125)
   :language Prolog
   :ontology auction
)
```

# FIPA ACL performatives

| performative | passing info | requesting info | negotiation | performing actions | error handling |
|---|---|---|---|---|---|
| accept-proposal | | | x | | |
| agree | | | | x | |
| cancel | | x | | x | |
| cfp | | | x | | |
| confirm | x | | | | |
| disconfirm | x | | | | |
| failure | | | | | x |
| inform | x | | | | |
| inform-if | x | | | | |
| inform-ref | x | | | | |
| not-understood | | | | | x |
| propose | | | x | | |
| query-if | | x | | | |
| query-ref | | x | | | |
| refuse | | | | x | |
| reject-proposal | | | x | | |
| request | | | | x | |
| request-when | | | | x | |
| request-whenever | | | | x | |
| subscribe | | x | | | |

# Inform and Request

The meaning of inform and request is defined in two parts:

- feasibility pre-condition: what must be true in order for the speech act to succeed
- rational effect: what the sender of the message hopes to bring about

$\langle i, inform(j, \varphi)\rangle$

- feasibility precondition: $B_i\varphi \wedge \neg B_i(B_j\varphi \vee B_j\neg\varphi \vee U_j\varphi \vee U_j\neg\varphi)$
- rational effect: $B_j\varphi$

$\langle i, request(j, \alpha)\rangle$

- feasibility precondition: $B_i Agent(\alpha, j) \wedge \neg B_i I_j Done(\alpha)$
- rational effect: $Done(\alpha)$

# Problems

- Impossible for the speaker to enforce those beliefs on the hearer! No way to verify mental state of agent on the grounds of its (communicative) behavior. This is a fundamental problem of all mentalistic approaches to communication semantics!
- Alternative approaches use the notion of social commitments
  - Idea: "A debtor $s$ is indebted to a creditor $h$ to perform action $\alpha$ or to believe proposition $\varphi$"
    - $[inform(s,h,\varphi)]O_{s,h}B_s\varphi$
    - $[promise(s,h,\alpha)]O_{s,h}I_s\alpha$
  - Often public commitment stores are used to track status of generated commitments
  - Benefit: at least (non)fulfilment of commitments can be verified

# Communication in GOAL

## Moods of messages

- Indicative: **:**
  - “I have run out of milk.”
  - Example: **send:**(amountLeft(milk, 0))
- Declarative: **!**
  - “I want the door to be closed!”
  - Example: **send!**(status(door, closed))
- Interrogative: **?**
  - “What time is it?”
  - Example: **send?**(timeNow(_))

- No inbuilt check of precondition and rational effects.
- Agents may lie, and agents may ignore others’ messages.

# Sending and receiving messages

- Fridge runs out of milk and tells the grocery planner.
  - **if bel**(amountLeft(milk, 0)) **then** (groceryplanner).**send:**(amountLeft(milk, 0))
- Grocery planner receives the message and adopts buying milk as a new goal.
  - **if** (fridge).**sent:**(amountLeft(milk, 0)) **then adopt**(buy(milk)).
- Special agent expressions:
  - all: all agents in the MAS
  - allother: all agents except the sending agent
  - some: select some agent randomly
  - someother: random selection among the other agents
  - self: the sender itself

# Ontologies

- One aspect we have not discussed so far: how can agents ensure the terminology they use is commonly understood?
- A prerequisite for meaningful communication is to agree on a “formal, explicit specification of a shared conceptualization” $\stackrel{\text{def}}{=}$ Ontology!

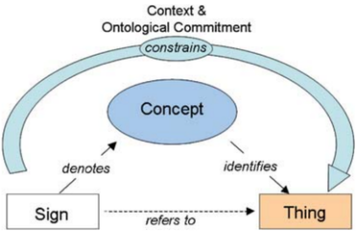
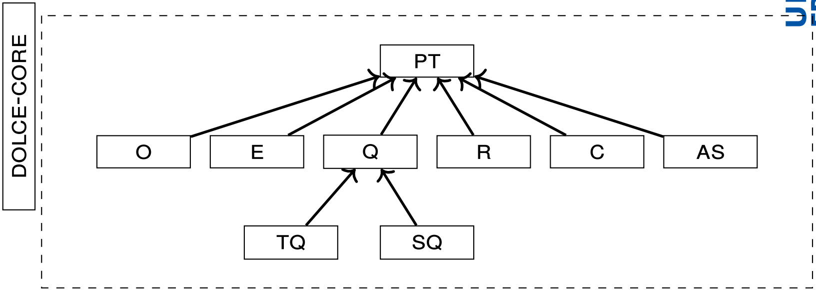



Fig.: Source: [4]

# The DOLCE-CORE Upper-Level ontology (cf., [6])




- Axioms constrain the interpretations. Ideally, the possible interpretations match the intended interpretations.
- Example: $\forall x, y\,[E(x) \rightarrow \exists y\,[TQ(y) \wedge I(y,x)]]$ excludes interpretations in which something is an event but not located in time.

# Ontology engineering

From a domain modeller's perspective, an ontology is a (terminological) knowledge base given by:

- a vocabulary used to describe some given domain
- a specification of the intended meaning of the vocabulary
- possibly, further constraints specifying additional domain knowledge

The aim is:

- to specify a common understanding of the domain
- to have a formal and machine-readable model of the domain
- Linking domain-level ontologies to upper ontologies can help to increase interoperability

# Ontology engineering

An ontology should be:

- meaningful, e.g., all named classes can have instances
- correct, in the sense that domain experts can agree on the meaning of the vocabulary as specified in the ontology
- rich, in the sense that the specified meaning provides a reasonable approximation of the intended meaning of the vocabulary

Existing tools and reasoners (Protege, Fact++, Racer, etc.) can help to build such ontologies, but also to solve several reasoning tasks.
There exists a family of well-defined ontology languages (e.g., OWL-languages) with a solid logical basis (Description Logics).

# Reasoning services

- Concept satisfiability: Is there a model of the ontology such that the concept's interpretation is nonempty?
- Concept subsumption: Does concept *A* subsume concept *B*, i.e., must each (possible) instance of concept *B* be an instance of concept *A*?
- Subsumption hierarchy: Compute the subsumption relations between all pairs of named concepts mentioned in the ontology
- Instance queries: Given a knowledge base of the individuals of the domain, retrieve all instances that match a given query
- Ontology mapping/alignment: Given two ontologies of the same domain, map/align the concepts specified in both

# Summary

- Speech act theory: Communicating as acting
- Formalizations of speech acts in terms of preconditions and effects on agents' mental states …
- …provide semantics for Agent Communication Languages
- Communication on the knowledge level requires agents to share an ontology

# Literature


- A subset of the slides is based on: Dr. Michael Rovatsos, The University of Edinburgh
  `http://www.inf.ed.ac.uk/teaching/courses/abs/abs-timetable.html`
- Michael Wooldridge: An Introduction to MultiAgent Systems, John Wiley & Sons, 2nd edition 2009.
- Cohen, P. R., Perrault, C. R., Elements of a Plan-Based Theory of Speech Acts, Technical Report No. 141, University of Illinois, 1979.
- Guarino, N., Oberle, D., Staab, S., What is an ontology?, In Handbook on Ontologies, Springer Berlin Heidelberg, 2009, pp. 1–17.
- Hindriks, K. V., Communicating agents, In Programming Cognitive Agents in GOAL, 2016, pp. 91–100.
- Borgo, S., Masolo, C., Foundational choices in dolce. In S. Staab, R. Studer (Eds.), Handbook on Ontologies, Second Edition 2009, pp. 361–381. Springer.