Reasoning, Personalization, Adaptation on the Web

Cristina Baroglio
Dipartimento di Informatica
Università degli Studi di Torino

member of REVERSE network of excellence
Outline

- Know your teacher (briefly introducing myself)
- Motivation to this lesson: why should you care about adaptation, personalization and reasoning?
- Technicalities
  - User modeling (concepts)
  - Adaptation by reasoning (mostly reasoning about actions and change)
    - Agents
    - Curriculum sequencing / planning
    - Web services / interaction protocols
- Lunch!
**Who am I?**

**Researcher**
Ph.D. in **Cognitive Sciences**
background: **Artificial Intelligence/Soft Computing**

machine learning (symbolic and neural)
- Agents that acquire a behaviour from experience

reasoning about actions/change
- Adaptation by means of reasoning -> **Semantic Web**

**Applicative fields:**
- robotics (control function approximation)
- image representation and understanding
- curriculum sequencing / e-learning
- web services
Artificial Intelligence / Soft Computing

the dream of most who work in AI:
to see a robot, that you programmed,
that really shows an intelligent behavior

problem:
robots tend to fall down the stairs and get broken ...

software agents are much
less expensive! So why not doing
something that goes on the web?
Is it that different?

A user (maybe a program) aims at finding a web page. The user describes - by keywords - what desired to a search engine (e.g. Google) that finds the proper resources and returns the links to the user. The key: powerful indexing mechanisms.
A user (maybe a program) aims at the execution of a task which involves the use of information/services accessible over the web. The user executes a program, that finds the proper resources and invokes them, in a way that fulfills the user's goals.
Problem

the information on the web is human-oriented

access should be based on keywords

access should be based on content ->
the Semantic Web effort
Resources are annotated by terms that belong to shared ontologies. Descriptions can be “understood” by programs, that perform intelligent queries, and reason about them.
Ontologies and Inferences (example of intelligent info retrieval)

Scenario
a Comp. Sci. Department information service

Query
Find the telephone numbers of researchers

But
telephone numbers associated to people not to what they do!! (e.g. A. Turing 34567)

If
the system knows that researchers are human beings and that telephone numbers are associated to human beings

Then
the system will retrieve the desired information
Ontologies and Inferences
(is-a and part-of)

Relations
ontologies usually define shared vocabularies, often the terms defined by an ontology are related by means of is-a or part-of relations.

is-a
A researcher is a person

part-of
A CPU is a part of a computer

What is a computer?
A conjunction should be expressed
Some other scenarios
Example 1

scenario:
Gigi wants to read the news
he connects to the website of his favourite newspaper

Provider:
the program that retrieves and maybe composes the
information to be presented

Product:
the composed page

Possible forms of adaptation/personalization:
at the level of information retrieval (by provider)
at the level of information presentation
(by provider or by browser)

Technique:
user modeling

Images: http://www.sla.purdue.edu/fll/JapanProj/FLClipart/
Example 2

Scenario:
Lilla wants to buy a computer on-line, she connects to an on-line computer shop

Provider:
the program that interacts with the user for understanding the needs and proposing a product

Product:
the assembled computer

Possible forms of adaptation/personalization:
at the level of computer assembly (by provider)
at the level of information presentation (by provider or by browser)

Technique:
goal-driven reasoning (+ user mod.)
Example 3

Scenario:
An artificial agent is delegated the task of making a reservation at a cinema where they show a given movie in a given town; the user does not want to leave his/her telephone number.

Provider:
the cinema booking service

Product:
the interaction with the provider + the reservation

Possible forms of adaptation/personalization:
at the level of interaction (by the agent)

(The search involves also dealing with a registry)

Technique:
reasoning about behaviour

Images: http://www.sla.purdue.edu/fll/JapanProj/FLClipart/
Example 4

Scenario:
An artificial agent is delegated the task of making a reservation at a restaurant and then at a cinema where they show a given movie in a given town; the user does not want to leave his/her telephone number.

Provider:
the restaurant and the cinema booking services

Product:
the interaction with the provider + the reservation

Possible forms of adaptation/personalization:
at the level of interaction (by the agent)
at the level of composition (by the agent)

(The search involves also dealing with a registry)

Technique:
behaviour composition + reason about behaviour
Difficult tasks!

They require adaptation

Adaptation:
The ability to alter so as to fit for a new use
produce personalized solutions according to:

- The user's characteristics
- The user's goal
- To fit a process specification

Difficult tasks!

Resource annotation is not enough

- **Adaptation** is mostly a form of reasoning

- **Knowledge** is important:
  - Info about the user ('s goal)
  - Info about the domain
  - Constraints

- **Models**
Taking into account the characteristics of the user ... user models
User models

90% of research on adaptation and personalization on the Web is based on user models.

A user model is a description of those characteristics of the user that are relevant to the decision process that leads to the selection and the presentation of the information (e.g. Education, age, personal taste, ...)

Classic approach

the user model is initialized by explicitly asking the user to answer a questionnaire, and is refined connection after connection. Often the initial model is empty.
The user model must be described according to a vocabulary; the same holds for the resources that are to be selected/presented to the user; the system must know the vocabulary as well.
Each prototype has associated a presentation style that rules the way in which the information will be shown to the user. Style sheets are commonly used to this aim. Often different descriptions are directly associated to each resource.
Example 1
(shopping on the web)

In a virtual shop, the same good is presented in different ways to different users

User 11: young, loves computers and science-fiction

Description will contain many technical details, comparison with other products, many numbers and acronyms

User 12: middle-age, loves painting and nature

Description will be at high-level, design and eco-compatibility of the good will be underlined
Example 2
(educational hypermedia)

Users (called “learners”) have access to a set of resources (documents, software) that they should read. Such resources are arranged in a hypermedia, through which they “navigate.”

Problem: many kinds of users! Different backgrounds!

One step look-ahead strategy: whatever the current position, show only the links, that lead to documents about topics, that the user can understand.
Example 2
(educational hypermedia)

- New links are “opened” according to:
  - Which pages have been visited
  - Which tests have been passed
- Often links are not hidden, they are just marked in a different way:
  - red = stop, you cannot understand this page
  - green = go ahead, you are suggested to read this page
Example 2
(educational hypermedia)

- A decision process occurs
- How to decide if a link is to be shown?
- The technique is curriculum sequencing and consists in:
  1) To describe the contents of documents by annotating the learning resources by the terms of an ontology (such terms are called in many ways: knowledge items, knowledge entities, competences, ...)
  2) to capture in some formal way the relations between the knowledge items
  3) To use such descriptions and knowledge about the user (user model + knowledge about what he/she already read) to decide which other topics he/she can understand
  4) Bayesian networks are used to this purpose
Some links

- L. Ardissono and A. Goy, Tailoring the Interaction With Users in Web stores, User Modeling and User-Adapted Interaction, 10(4):251-303
Learning models

The idea is to induce a user model by observing the behaviour of a user

- **Off-line learning**: a training set of examples is gathered while interacting with the user and it is used to learn a model

- **Lazy learning**: the model is learnt/updated while making recommendations
Social behavior
collaborative filtering
(user models built by clustering)

User models can be inferred by studying the behaviour of a population
e.g. By means of unsupervised neural networks

K-means and Self-Organizing Maps
Clustering

Is the task of identifying groups of similar individuals out of a set of learning instances. The obtained groups can be used for classifying new individuals based on a similarity measure.
• Each instance is a point in the so-called input space; each coordinate is a feature of the instance
• Similarity of instances corresponds to their closeness in the input space
• Given a set of similar instances I can compute a prototype
K-means clustering algorithm

- Choose K initial cluster centers
- Assign all data points to the closest cluster
- Recalculate the cluster center by applying the following formula to each center coordinate

\[
\frac{\sum_{j=1}^{N} W_{ij}}{N}
\]

- If the new centers differ from the previous ones go back to the beginning

Other approaches exploit SELF-ORGANIZING MAPS (or Kohonen's maps) to perform the same task. They are an unsupervised neural network model.
Some example systems

**WEBSOM**: documents are organized in a two-dimensional map based on a description of their contents.

**LOGSOM**: organizes web pages on a **SOM** according to user navigation patterns.

**Learning data**: obtained from server log files.
Problems

- Neural networks are mathematical tools for function approximation, their inputs are vectors of numbers.
- Web page content is described in a symbolic way by the terms from some ontology.
- How to turn a discrete, symbolic domain in a continuous, numeric domain?
Problems of collaborative filtering

- **Early-rater problem**: collaborative filtering has no way to suggest a new resource (it has not been considered by users yet)

- **Sparsity problem**: if the number of users with respect to the volume of information in the system user models are not reliable

- **Users whose taste is different than the norm are not modeled**
Some links


- B. Mobasher, R. Cooley, J. Srivastava, Automatic personalization based on web mining, communications of the ACM 43(8), 2000

- M. Perkowitz, O. Etzioni, Adaptive sites, learning from user access patterns, proc. of WWW6, 1997

Not only user models
When UM do not work well

User models are refined according to the behaviour of the user and capture his/her general preferences. They are "past-oriented", in some situations this may be inconvenient.

Example

newsportal: usually Pino checks for sport news but today he heard of a terrible accident and would like to immediately find info about it.
When UM do not work well 2

User models are meaningful only if users visit a web site frequently; this is not always the case.

Often web site visits are “task-oriented”, the reasoning task should focus on helping the user in satisfying his/her goals.

Example

buying a PC: find out what kind of use will be done of the PC (server? Playing games? Browsing the web? Esthetic purposes?)

Recommender systems
When UM do not work well

The user may be a piece of software (e.g. a software agent)

this may be the case of “web services”, hardware or software resources, accessible via the web, that can be automatically retrieved, invoked, composed, etc.

Example

organize a visit to Aussois: find a train, find a hotel, check out the weather forecast, etc.
Are these forms of personalization?

- **Newsportal**: personalize information selection w.r.t. the user's goal
- **Computer shop**: adapt the interaction with the user according to his/her interests
- **Organize trip to Aussois**: compose a set of softwares (web services, developed independently) so to accomplish a task that respects the constraints posed by the user

In all these cases:

personalization is an outcome of processing we will how it can be an outcome of reasoning
What do we need?

A semantic layer associated to resources (documents, software, ... whatever)

Reasoning techniques for performing retrieval, sequencing, composition, ...

Is it necessary to develop new techniques?
No
What we'll see next: reasoning about actions

Two main cases:

- **educational application domain**
  - selection
  - curriculum sequencing
  - validation

- **web services**
  - selection
  - personalization of the interaction with a w.s.
  - composition
Educational Framework
Learning resources as actions

Let us interpret a learning resource LR as an action:

- LR can be executed if the user has some competence C
- by executing LR the user will acquire competence G

C and G are described in the terms given by an ontology

the system state is a representation of the user (supposed) knowledge (its initial knowledge, augmented step by step)

The system adopts the user goal of acquiring some expertise (learning goal) and builds a plan for achieving it

no probability is to be defined
Learning resources as actions

- Module A
  - Prerequisites: processes, concurrency
  - Effects: deadlock

- Module B
  - Prerequisites: processes, concurrency, mutual exclusion
  - Effects: deadlock, deadlock avoidance

Ontology terms

Resources explicitly annotated by ontology terms!
Important!

The *description* of the resources is *not extracted* (e.g. by text processing techniques) as it often happens nowadays.

It is *explicitly added* to the resources by the creators.

It is a description at the *level of knowledge*, given in terms of a shared vocabulary, that can be used by *automatic systems*.
Learning resources representation standards

do I have to invent new formalisms? No

**SCORM** is a standard framework for representing learning resources; it allows the semantic annotation of a resource by means of LOM (standard for learning object metadata)

LOM allows to add to the description of a learning resource a set of “taxons”, taxonomy elements, each of which has a role. Among the possible roles “educational objective” and “prerequisite” can be used to encode preconditions and effects of the resource, interpreted as an action.
Reasoning about learning resources
selection

Selection can be based on a (knowledge level) description of what a resource supplies/teaches.

It can further be constrained by the system state.

User knows processes concurrency deadlock
Want to learn

Module A

Module B

Same as...
Curriculum sequencing

We have mentioned educational application domains and curriculum sequencing: i.e. The task of defining good reading paths in a hyperspace of learning resources.

In Adaptive Hypermedia a one step approach is followed.

At any time a Bayesian network decides which learning resources the user can read with profit; it disables access to the others.

A learning goal is defined: the user will arrive to the goal by different paths, depending on the initial knowledge, and on personal taste.

But I can also ...
Reason about learning resources sequencing

Module B can be selected only if the user knows also something about Mutual Exclusion; the user knows nothing about it.

The system can search for a second module that teaches the missing competence.

User knows
processes concurrency

Deadlock avoidance
Want to learn

Module X

teaches: Mutual Exclusion

Module B
requires: Mutual Exclusion

Multi-step sequencing: planning
Building complete reading sequences

To this aim, **AI planning techniques** can be used. A **plan** is a sequence of atomic actions (resources).

Two main approaches:

- Combine atomic actions (e.g., A*, graphplan).
- Produce plans that respect a schema: e.g., procedural planning (interesting in educational domains: procedures as learning strategies).
What is a learning strategy?

The organization of the material in a lesson or a course is not only up to prerequisites and effects but also to the experience of the lecturer.

**Learning strategy**: overall schedule of the topics the view of teacher of how topics should be sequenced...
My learning strategy for today:

- Know your teacher (briefly introducing myself)
- Motivation to this lesson: why should you care about adaptation, personalization and reasoning?
- Technicalities
  - User modeling (core)
  - Adaptation by reasoning about actions and changes
  - Agents
  - Curriculum sequencing / planning
  - Web services / interaction protocols
- Lunch!!
Moving to the web ...

Task of adaptation and reasoning on the semantic web

identify a set of learning resources that fit in my learning strategy and that are the most suitable to the specific user
Procedural planning

The search space is constrained by allowing only sequences of actions that are executions of a given procedure

\[
\text{plan: procedure execution} \\
\text{procedure: behavior schema}
\]
A specific case: using DyLOG to specify learning strategies

DyLOG, by Baldoni et al., is a language for programming agents, based on a modal approach for reasoning about actions and change.

- **Primitive actions**: preconditions and effects
- **Sensing actions**: interaction with the world
- **Prolog-like procedures**: complex actions

A domain description consists of a set of primitive actions, a set of sensing actions, a set of complex actions, together with a set of initial observations.
Learning resource as a DyLOG atomic action

Module A
- Processes
- Concurrency
- Module A causes deadlock

Module B
- Processes
- Concurrency
- Mutual exclusion
- Module B causes deadlock, deadlock avoidance

moduleA possible if processes, concurrency
moduleA causes deadlock

moduleB possible if
    processes, concurrency, mutual_exclusion
moduleB causes
deadlock, deadlock_avoidance
Learning strategy as a procedure

The meaning of “basic notions of Comp. Sci.” depend on the kind of students you need to explain things to ...

```
comp_science_4_biolists is db, statistical_tools

comp_science_4_web_designers is
    web_programming, computer_networks
```

Given a specific student: the task is to produce a specific plan, i.e. sequence of actual learning resources, that will allow him/her to acquire the desired expertise
Procedural planning in DyLOG
Another educational application validation of the user's choices
Scenario

A user can browse a repository of learning resources and choose some that he/she likes and considers useful for achieving a given learning goal.

The user would like to know if the chosen material will lead him/her to the target.

Forms of reasoning
Plan validation + Explanation of failure
Plan validation + Explanation of failure

**Initial state**

![Diagram of state transitions](image)

A set of assumptions is collected along the way; the plan is incorrect given that those assumptions hold.

- LRi possible if a, b
- LRi causes c

**Conclusion:**
- LRi cannot be applied unless user already knows b

The domain is monotonic; each state includes the previous one; there is no forgetting.
Links

• M. Baldoni, L. Giordano, A. Martelli, V. Patti, Programming rational agents in a modal action logic, Annals of Mathematics and Artificial Intelligence, Special Issue on Logic-based Agent Implementation, to appear
• M. Baldoni, C. Baroglio, e V. Patti, Web-based adaptive tutoring: an approach based on logic agents and reasoning about actions, Artificial Intelligence Review. To appear
Web Services
The Web as a provider of services

The web is evolving into a provider of services, resources that allow one to check/change the state of the world:

- **Information providing services**: weather forecast, flight arrival schedule, ...
- **World-altering services**: ticket booking, purchase, ...

**Web services**: hard-/soft-ware devices accessible over the internet

There's a need for all these devices to interoperate

**So far**: interoperation provided by code hardwired in the APIs for information extraction. APIs are strongly dependant on the structure of documents/resources they must handle, if the latter changes they are to be changed

More flexibility is desirable ...
The Web as a provider of services

The Semantic Web initiative deals also with the problem of representing and remotely invoking web services.

In order to obtain flexibility:
need for a computer-interpretable description of the services

Two main initiatives

- AI agent community: DAML-S/ OWL-S (based on semantics)
- Commercial initiative WSDL/ BPEL4WS (not based on semantics yet)

Once again the action metaphor

Markup languages
Some possible tasks

What do we wish to do (in an automatic way) with web services?

- Discovery
- Execution
- Composition & interoperation

Currently, none of these tasks can fully be executed (do you agree?)
Example: organize the journey to Aussois

Services I can find on the web:
- train ticket / flight
- hotel reservation

Supplied information: school period (June, 21st-25th), ticket class (economy)

My expectation: after task execution I will have a ticket that will allow me to arrive before the beginning of the school, and leave right after its end, and a reservation at some nice hotel

possible problems...
Example: organize the journey to Aussois

Would you trust an agent to do it for you?

- The system finds a place called Aussois in Canada, thus it reserves a flight to Ottawa ... a little far!
- The system does not find a free seat on the 21st, but it finds a free seat on the 19th, it immediately makes a reservation; luckily the hotel has a free room starting from the 21st ... where to sleep before this date?
- Days of arrival and hotel reservation match, however the train arrives in Modane at 21:00 ... school is over
- ...
What kind of markup do we need for...

**Discovery?** What does the service provide to users? Properties that classify the service

**Execution?** Dataflow model / IOPE (inputs, outputs, prerequisites and effects)

**Composition?** Business logic, how the service affects the world
OWL-S descriptions

Kinds of service

- **primitive**: a single device is invoked, little interaction with the user, a simple response is returned
- **complex**: composed of multiple services, a conversation with the user is required, the user can make choices

A service is described by

- **service profile**: used for advertising (and discovering) services
- **process model**: describes the service's operations
- **grounding**: describes how to interoperate with the service by means of messages
Info in the profiles

- **human-readable**: Service name, text description and contact information
- **functionality description**: inputs and outputs (they can be conditional and refer to inputs and outputs of the process model), preconditions and effects (they can be conditional)
- **profile attributes**: category (ontology-based description)
- ...

...
<!-- specification of quality rating for profile -->
<profile:qualityRating>
  <profile:QualityRating rdf:ID="BravoAir-goodRating">
    <profile:ratingName>
      SomeRating
    </profile:ratingName>
    <profile:rating rdf:resource="&concepts;#GoodRating"/>
  </profile:QualityRating>
</profile:qualityRating>

<!-- Specification of the service category using NAICS -->
<profile:serviceCategory>
  <addParam:NAICS rdf:ID="NAICS-category">
    <profile:value>
      Airline reservation services
    </profile:value>
  </addParam:NAICS>
  <profile:code>
    561599
  </profile:code>
</profile:serviceCategory>
Info in the process model 1

- **input/(conditional) output**: information that is necessary for the computation to take place (may be provided by other processes), and that will be processed. + info that is produced

- **preconditions/(conditional) effects**: related to changes occurring to the world by the execution of the service

- **conditions**: no mandatory language to be used (best candidates SWRL and DRS)

- ...
http://www.daml.org/services/owl-s/1.0/BravoAirProfile.owl

<!-- Descriptions of IOPEs -->

<profile:hasInput rdf:resource="&ba_process;#DepartureAirport_In"/>
<profile:hasInput rdf:resource="&ba_process;#ArrivalAirport_In"/>
<profile:hasInput rdf:resource="&ba_process;#OutboundDate_In"/>
<profile:hasInput rdf:resource="&ba_process;#InboundDate_In"/>
<profile:hasInput rdf:resource="&ba_process;#RoundTrip_In"/>
<profile:hasInput rdf:resource="&ba_process;#PreferredFlightItinerary_In"/>
<profile:hasOutput rdf:resource="&ba_process;#AvailableFlightItineraryList_Out"/>
<profile:hasInput rdf:resource="&ba_process;#AcctName_In"/>
<profile:hasInput rdf:resource="&ba_process;#Password_In"/>
<profile:hasInput rdf:resource="&ba_process;#ReservationID_In"/>
<profile:hasInput rdf:resource="&ba_process;#Confirm_In"/>
<profile:hasOutput rdf:resource="&ba_process;#PreferredFlightItinerary_Out"/>
<profile:hasOutput rdf:resource="&ba_process;#AcctName_Out"/>
<profile:hasOutput rdf:resource="&ba_process;#ReservationID_Out"/>
<profile:hasEffect rdf:resource="&ba_process;#HaveSeat"/>
<!- Confirm Reservation (ATOMIC) Confirm selected reservation -->

<process:AtomicProcess rdf:ID="Confirm Reservation">
    <process:hasInput rdf:resource="#ReservationID_In"/>
    ...
    <process:hasOutput rdf:resource="#PreferredFlightItinerary_Out"/>
    ...
    <process:hasEffect rdf:resource="#HaveSeat"/>
</process:AtomicProcess>

<process:Input rdf:ID="ReservationID_In">
    <process:parameterType rdf:resource="&concepts;#ReservationNumber"/>
</process:Input>
...

<process:UnConditionalOutput rdf:ID="PreferredFlightItinerary_Out">
    <process:parameterType rdf:resource="&concepts;#FlightItinerary"/>
</process:UnConditionalOutput>

<process:UnConditionalEffect rdf:ID="HaveSeat">
    <process:ceEffect rdf:resource="&concepts;#HaveFlightSeat"/>
</process:UnConditionalEffect>
Info in the process model 2

- **Atomic processes**: directly invocable, they execute in a single step
- **Simple processes**: not invocable, they are elements of abstraction that present a service as having a one-step execution
- **Composite**: decomposable, possible control constructs:
  - Split (concurrent execution), Unordered, Split+join
  - Sequence
  - Unordered
  - Choice, If-then-else
  - Iterate, repeat-until
BravoAir_Process is a composite process. It is composed of a sequence whose components are 2 atomic processes, GetDesiredFlightDetails and SelectAvailableFlight, and a composite process, BookFlight.

```xml
<bravoairexporting flux>
OWL-S description of a service

Service

- Is described by a Service Model
- It presents a Service Profile
- It supports a Service Grounding

Seeking agent

What the service requires and supplies

How it works

How to access to it

Low level protocol/port/serialization
Emphasis on the seeking agent

Once again these representations are thought for the automatic retrieval, invocation, etc. performed by artificial entities (agents)

Agent main characteristics

- autonomous
- has a state
- can sense the world
- decides how to act based on its state and its perceptions

E.g. a thermostat!

It is not necessarily rational ;-)

The seeking agent

Two-step search

use the profile
profiles stored / handled by registries for quick focus

use the model
used when selection is based on how the service is executed, and for composition

Not necessarily a search engine
Many different agents roam the web with various purposes

It requires reasoning !!
Example

I live out of town, need to move by bus
Agent Jeeves: search if a book is available in some library of Mytown, and reserve it; please, pay attention to the bus schedule!

University library requires: you are a student
pick up during opening time: 9:00 – 16:00
reservation?

Town library requires: you are registered
pick up during opening time: 9:00 – 12:00 / 15:00 -18:00
reservation?

Bus schedule: 6:00 or 13:00
Example

I live out of town, need to move by bus

Agent Jeeves: search if a book is available in the libraries of Mytown, and reserve it; please, pay attention to the bus schedule!

Agent task: build a plan that will allow my master to achieve her goal

Two library services are available, both have the book and my master can go to both ... which to choose?

Interaction with the bus information service leads to choose between the two libraries

The bus!
The seeking agent: how to build it?

**Requisite**: it must be able to draw conclusions about programs described in a declarative way (the web services) and/or by interacting with them.

Web services descriptions based on the action metaphor.

Techniques for reasoning about actions and change are necessary.

**Examples**: situation calculus, agent programming languages such as Golog and DyLOG.
The seeking agent: a sketch

1) download

2) decide if interacting

3) invoke

4) interaction

Is this a multi-agent system?
Comments

The agent community is highly variable, organized on the fly depending on needs.

Agents let their behaviour be visible to the others.

They can reason on each other's behavior and foresee the effects of their actions.

Actions: world-affecting / information-providing

In order to foresee effects of information-providing actions the agents must be able to make assumptions on the “mental state” of their interlocutors.
Comments

Many approaches in the literature are based on variant of modal logic, in which mental attitudes, such as beliefs, goals and intentions, as well as communicative acts are represented by modalities.

Only recently the attention has been moved to formalize those aspects of communication that are related to the conversational context in which communicative acts occur.

Literature ...


What kind of reasoning is interesting?

Proof of existential properties

I can execute a protocol $p$ in a way such that afterwards the provider will not have this piece of information:

$\langle p^\text{me} \rangle B^\text{me} \rightarrow B^\text{other info}$

*Modal logic representation*

There is a possible execution of $p$ such that...

Nested belief

I believe the “other” does not know info
DyLOG, the interaction protocols

**Communication protocol**: procedure that defines the communicative behavior of an agent (or a web service)

It is defined on the basis of predefined speech acts (atomic, communicative actions), for instance FIPA speech acts

**Conversation**: an execution of a communication protocol, a specific sequence of speech acts
DyLOG, example

Customer's view of the protocol in DyLOG:

\[
\langle \text{get\_ticket\_1}_C(Self, WebS, Film) \rangle \varphi \subset \\
\langle \text{yes\_no\_query}_Q(Self, WebS, available(Film)) ; \\
B_{Self} available(Film)? ; get\_info(Self, WebS, cinema(C)); \\
\text{yes\_no\_query}_I(Self, WebS, pay\_by(credit\_card)); \\
B_{Self} pay\_by(credit\_card)? ; inform(Self, WebS, cc\_number); \\
get\_info(Self, WebS, booked(Film)) \rangle \varphi
\]
What kind of reasoning is interesting?

Proof of universal properties

Properties of the multi-agent system as a whole

The overall behavior of the agent system is modeled

Among the various approaches:
use of Dynamic Linear Time Temporal Logic
Links

• S. A. McIlraith, T. C. Son, H. Zeng, Semantic Web Services, IEEE Intelligent Systems, 2001
• S. A. McIlraith, T. C. Son, Adapting Golog for composition of semantic web services
• The OWL Services Coalition, OWL-S: Semantic Markup for Web Services, white paper (http://www.daml.org)
• Linksto web service resources http://lsdis.cs.uga.edu/lib/presentations/SWSP-tutorial-resource.html