Silvia Rossi

Introduzione

Lezione n. 1

Corso di Laurea:
Informatica

Insegnamento:
Sistemi
multi-agente

Email:
silrossi@unina.it

A.A. 2014-2015

Sistemi Multi-Agente

Contatto:

silrossi@unina.it
people.na.infn.it/~srossi

Orario di Ricevimento:

Martedì: 14:00-16:00 – Studio 0D15b
PRISCA Lab

Libri di testo consigliati:

An Introduction to Multi-Agent Systems, M. Wooldridge (W). (Second Edition)

...

WOA 2015

XVI WORKSHOP "DAGLI OGGETTI AGLI AGENTI"

NAPOLI
17-19 GIUGNO 2015

Home | Call for Paper | Per gli Autori | Miniscuola | Programma | Venue | Registrazione

XVI WORKSHOP "DAGLI OGGETTI AGLI AGENTI"

Cinque sviluppi tecnologici stagno influenzando le storia dell’informatica e dell’IA:

ubiquità;

interconnessione;

Intelligenza;

delegazione; e

human-orientation.

Ubiquità

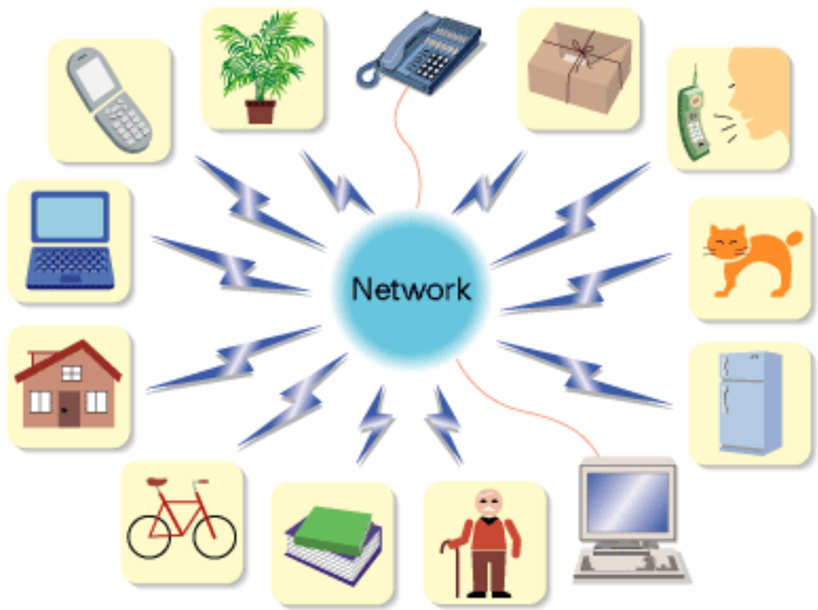
•The continual reduction in cost of computing capability has made it possible to introduce processing power into places and devices that would have once been uneconomic;

•As processing capability spreads, sophistication (and intelligence of a sort) becomes ubiquitous;

•What could benefit from

•having a processor embedded

•in it…?



Ubiquitous computing will enable diverse wireless applications, including monitoring of pets and houseplants, operation of appliances, keeping track of books and bicycles, and much more.

Interconnessione

•Computer systems today no longer stand alone, but are networked into large distributed systems

•The internet is an obvious example, but networking is spreading its ever-growing tentacles…

•Since distributed and concurrent systems have become the norm, some researchers are putting forward theoretical models that portray computing as primarily a process of interaction

•The complexity of tasks that we are capable of automating and delegating to computers has grown steadily;

•If you don't feel comfortable with this definition of "intelligence", it's probably because you are a human.

Delega

- Computers are doing more for us – without our intervention
- We are *giving control* to computers, even in safety critical tasks
- One example: fly-by-wire aircraft, where the machine's judgment may be trusted more than an experienced pilot
- Next on the agenda: fly-by-wire cars, intelligent braking systems, cruise control that maintains distance from car in front…

Human-Orientation

- The movement away from machine-oriented views of programming toward concepts and metaphors that more closely reflect the way we ourselves understand the world
- Programmers (and users!) relate to the machine differently
- Programmers conceptualize and implement software in terms of higher-level – more human-oriented – abstractions

Programming has progressed through:

- machine code;
- assembly language;
- machine-independent programming languages;
- sub-routines;
- procedures & functions;
- abstract data types;
- objects;

to *agents*.

- Delegation and Intelligence imply the need to build computer systems that can act effectively on our behalf
- This implies:
  - The ability of computer systems to act *independently*
  - The ability of computer systems to act in a way that *represents our best interests* while interacting with other humans or systems

Interconnection and Distribution have become core motifs in Computer Science;

But Interconnection and Distribution, coupled with the need for systems to represent our best interests, implies systems that can *cooperate* and *reach agreements* (or even *compete*) with other systems that have different interests (much as we do with other people);

So Computer Science expands…

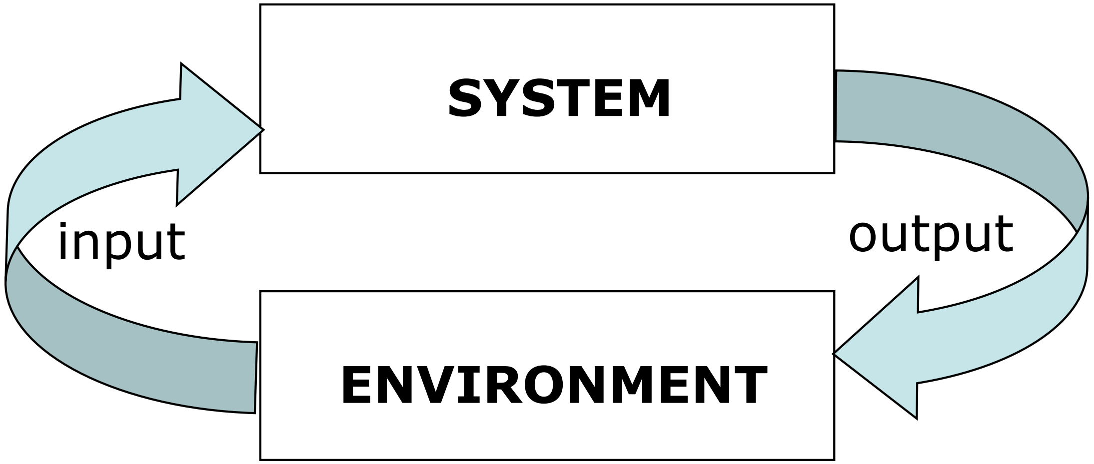
These issues were not studied in Computer Science until recently;

All of these trends have led to the emergence of a new field in Computer Science and IA: *multi-agent systems.*

Agenti Razionali

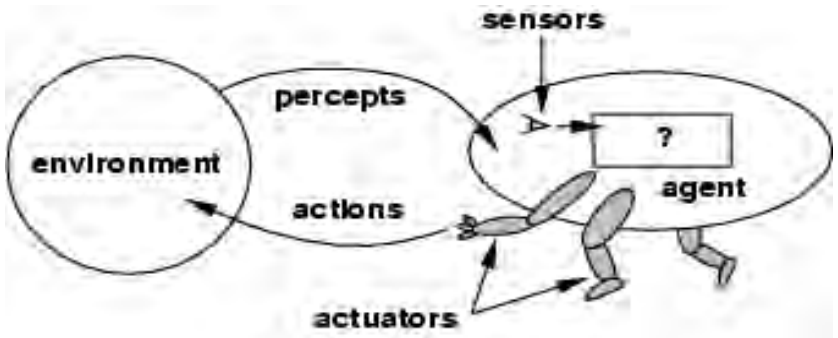
(W: 2.2-2.4)

- An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators



- agent *is a computer system capable of* autonomous action *in some environment in order to meet its* design objectives

Agents and environments



- The agent function maps from percept histories to actions:

  [*f*: P* → A]

- The agent program runs on the physical architecture to produce *f*
- agent = architecture + program

- Rational Agent: For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever built-in knowledge the agent has.

- Rationality is distinct from omniscience (all-knowing with infinite knowledge)
- Agents can perform actions in order to modify future percepts so as to obtain useful information (information gathering, exploration)
- An agent is autonomous if its behavior is determined by its own experience (with ability to learn and adapt)

- Trivial (non-interesting) agents:
  - thermostat
  - UNIX daemon (e.g., biff)
- *An* intelligent agent *is a computer system capable of* flexible *autonomous action in some environment*

By *flexible*, we mean:

- *reactive*
- *pro-active*
- *social*

Reactivity

If a program's environment is guaranteed to be fixed, the program need never worry about its own success or failure – program just executes blindly

Example of fixed environment: compiler

The real world is not like that: things change, information is incomplete. Many (most?) interesting environments are *dynamic*

Software is hard to build for dynamic domains: program must take into account possibility of failure – ask itself whether it is worth executing!

A *reactive* system is one that maintains an ongoing interaction with its environment, and responds to changes that occur in it (in time for the response to be useful)

Proactiveness

Reacting to an environment is easy (e.g., stimulus → response rules)

But we generally want agents to *do things for us*

Hence *goal directed behavior*

Pro-activeness = generating and attempting to achieve goals; not driven solely by events; taking the initiative

Recognizing opportunities

Balancing Reactive and Goal-Oriented Behavior

- We want our agents to be reactive, responding to changing conditions in an appropriate (timely) fashion

- We want our agents to systematically work towards long-term goals

- These two considerations can be at odds with one another
- Designing an agent that can balance the two remains an open research problem

Social Ability

The real world is a *multi*-agent environment: we cannot go around attempting to achieve goals without taking others into account

Some goals can only be achieved with the cooperation of others

Similarly for many computer environments: witness the Internet

Social ability in agents is the ability to interact with other agents (and possibly humans) via some kind of *agent-communication language*, and perhaps cooperate with others

Social Ability: Cooperation

•Cooperation is *working together as a team to achieve a shared goal*.

• Often prompted either by the fact that no one agent can achieve the goal alone, or that cooperation will obtain a better result (e.g., get result faster).

Social Ability: Coordination

- Coordination is *managing the interdependencies between activities*.
- For example, if there is a non-sharable resource that you want to use and I want to use, then we need to coordinate.

Social Ability: Negotiation

- Negotiation is *the ability to reach agreements on matters of common interest*.

- For example: You have one TV in your house; you want to watch a movie, your housemate wants to watch football.
- A possible deal: watch football tonight, and a movie tomorrow.
- Typically involves offer and counter-offer, with compromises made by participants.

Other Properties

Other properties, sometimes discussed in the context of agency:

mobility: the ability of an agent to move around an electronic network

veracity: an agent will not knowingly communicate false information

benevolence: agents do not have conflicting goals, and that every agent will therefore always try to do what is asked of it

learning/adaptation: agents improve performance over time

A Weak Notion of Agency

- If you take any of these attributes away, then you end up with software you already have. . .
- Think of (weak) agents as human-like ‘assistants’ or ‘drones’ that are limited in their abilities:
    - you can give them tasks to do, and they can go away and cooperate with other agents to achieve these tasks;
    - also, they are capable of taking the initiative in a limited way, like a human secretary would.

Are agents just objects by another name?

Object:

- encapsulates some state
- communicates via message passing
- has methods, corresponding to operations that may be performed on this state

Main differences:

agents are autonomous:

agents embody stronger notion of autonomy than objects, and in particular, they decide for themselves whether or not to perform an action on request from another agent

agents are smart:

capable of flexible (reactive, pro-active, social) behavior, and the standard object model has nothing to say about such types of behavior

agents are active:

a multi-agent system is inherently multi-threaded, in that each agent is assumed to have at least one thread of active control

Objects do it for free...

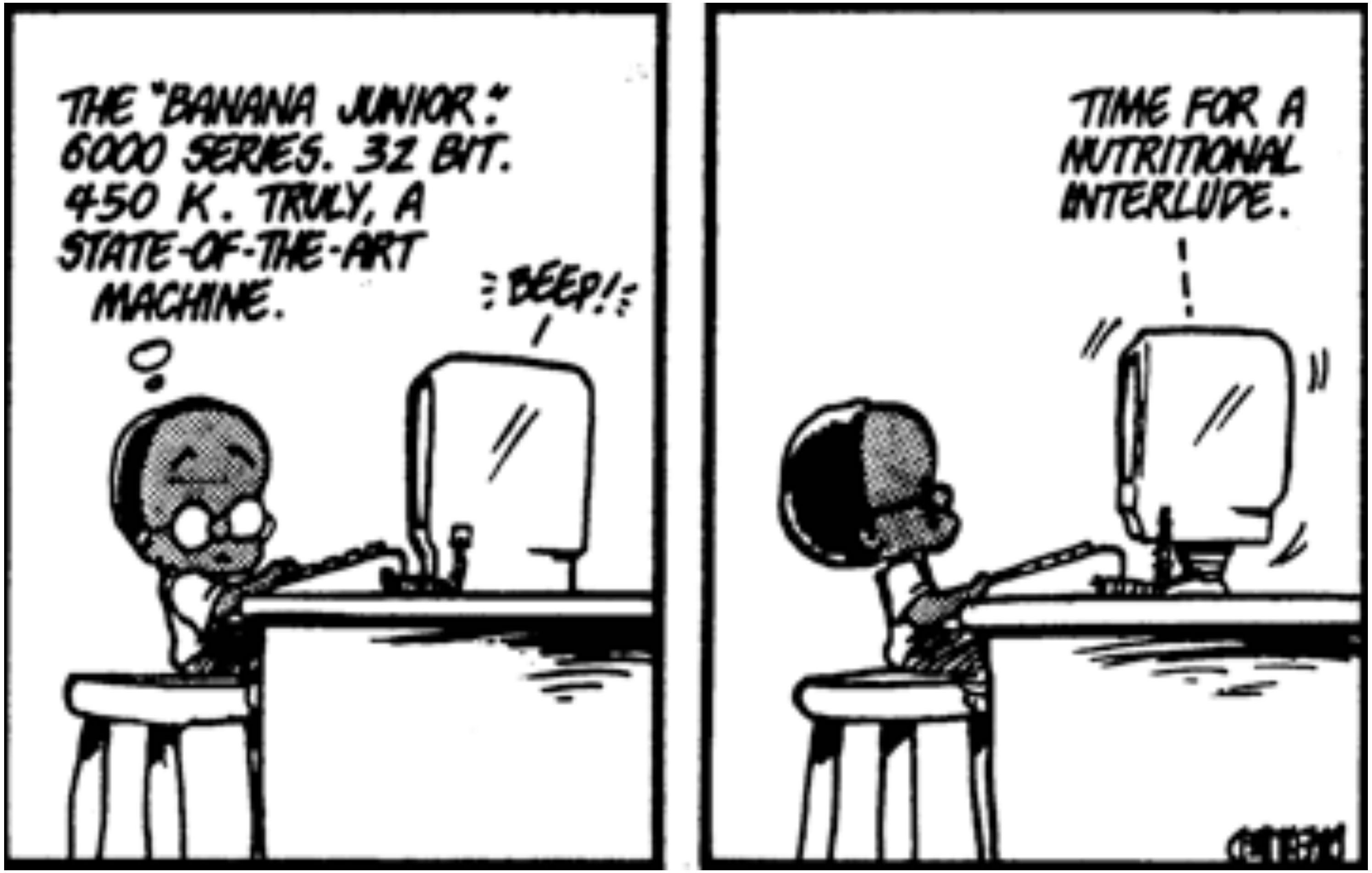
agents do it because they want to

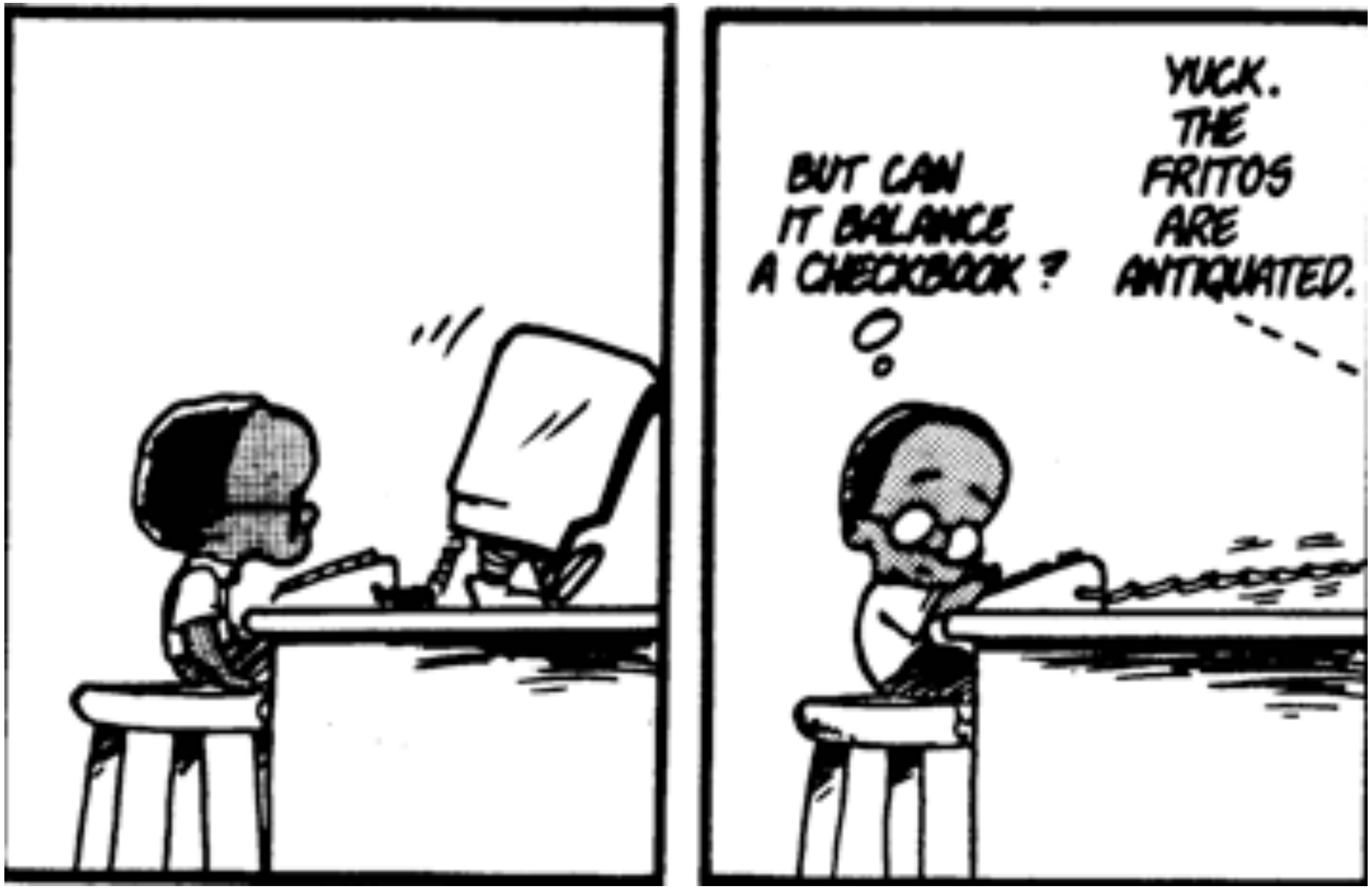
agents do it for money

...Objects


SOME
ASSEMBLY
REQUIRED
BANANA
PC JUNIOR
6000 SERIES
COMPUTER

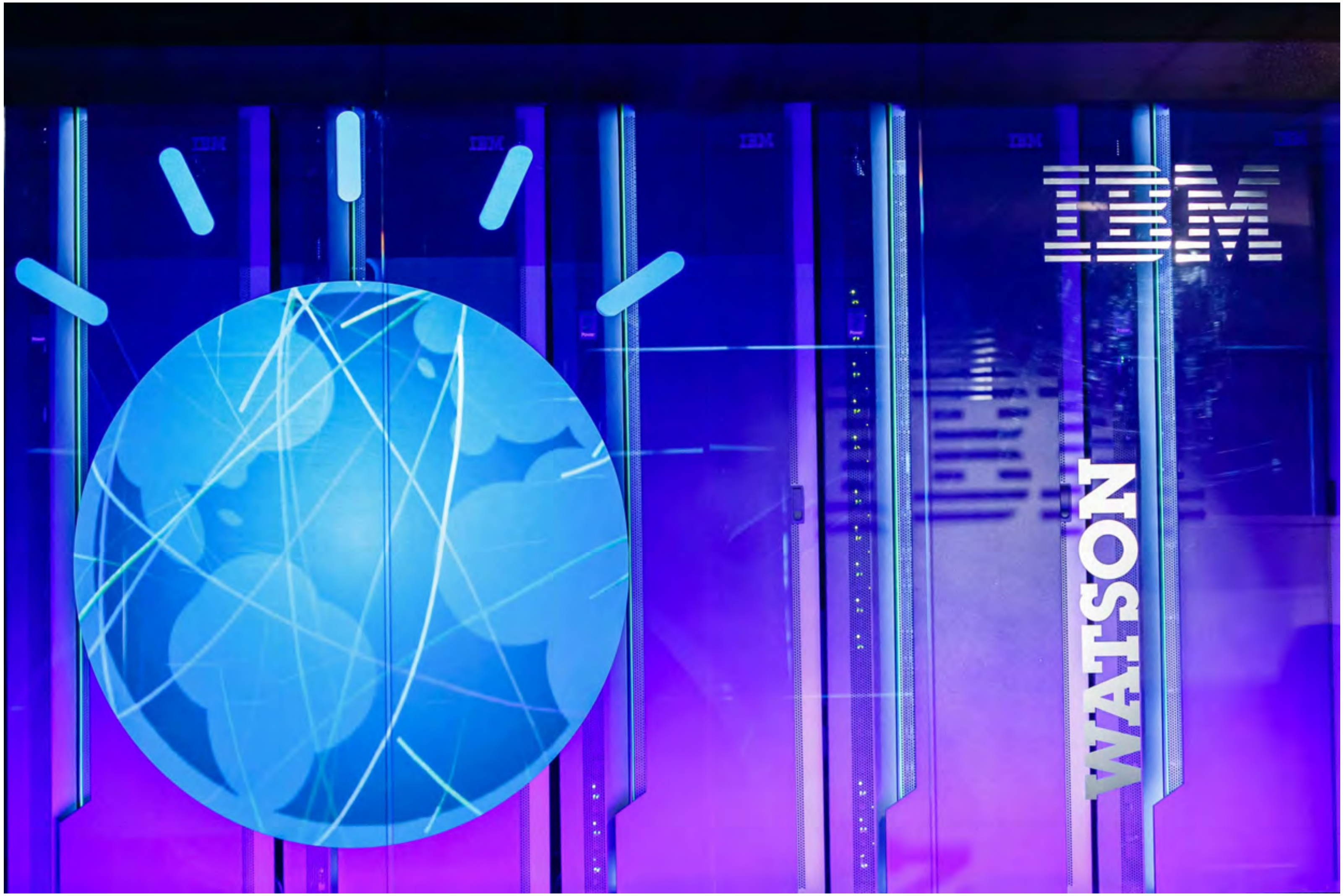
… Agents


THE "BANANA JUNIOR" 6000 SERIES. 32 BIT. 450 K. TRULY, A STATE-OF-THE-ART MACHINE.
BEEP!
TIME FOR A NUTRITIONAL INTERLUDE.


BUT CAN IT BALANCE A CHECKBOOK ?
YUCK. THE FRITOS ARE ANTIQUATED.

- Aren’t agents just expert systems by another name?
- Expert systems typically disembodied ‘expertise’ about some (abstract) domain of discourse (e.g., blood diseases)
- Example: MYCIN knows about blood diseases in humans
  - It has a wealth of knowledge about blood diseases, in the form of rules
  - A doctor can obtain expert advice about blood diseases by giving MYCIN facts, answering questions, and posing queries

- Main differences:
  - agents *situated in an environment:*
    MYCIN is not aware of the world — only information obtained is by asking the user questions
  - agents *act:*
    MYCIN does not operate on patients
- Some *real-time* (typically process control) expert systems *are* agents


IBM
WATSON

- Aren't agents just the AI project?
Isn't building an agent what AI is all about?
- AI aims to build systems that can (ultimately) understand natural language, recognize and understand scenes, use common sense, think creatively, etc. — all of which are very hard
- So, don't we need to solve all of AI to build an agent…?

When building an agent, we simply want a system that can choose the right action to perform, typically in a limited domain

We *do not* have to solve *all* the problems of AI to build a useful agent:

a little intelligence goes a long way!

Oren Etzioni, speaking about the commercial experience of NETBOT, Inc:
"We made our agents dumber and dumber and dumber…until finally they made money."

There’s no such thing as a single agent system.

Multi-agent Systems: a First Definition

- A multi-agent system is one that consists of a number of agents, which *interact* with one-another

- In the most general case, agents will be acting on behalf of users with different goals and motivations

- **To successfully interact, they will require the ability to cooperate, coordinate, and negotiate with each other, much as people do.**

- The course covers two key problems:
  - How do we build agents capable of independent, autonomous action, so that they can successfully carry out tasks we delegate to them?
  - How do we build agents that are capable of interacting (cooperating, coordinating, negotiating) with other agents in order to successfully carry out those delegated tasks, especially when the other agents cannot be assumed to share the same interests/goals?
- The first problem is *agent design*, the second is *society design* (micro/macro)

Key problems

In Multi-agent Systems, we address questions such as:

- How can cooperation emerge in societies of self-interested agents?
- What kinds of languages can agents use to communicate?
- How can self-interested agents recognize conflict, and how can they (nevertheless) reach agreement?
- How can autonomous agents coordinate their activities so as to cooperatively achieve goals?

While these questions are all addressed in part by other disciplines (notably economics and social sciences), what makes the multi-agent systems field unique is that it emphasizes that the agents in question are *computational, information processing* entities.

Multi-agent Systems is Interdisciplinary

The field of Multi-agent Systems is influenced and inspired by many other fields:

- Economics
- Philosophy
- Game Theory
- Logic
- Ecology
- Social Sciences

This can be both a strength (infusing well-founded methodologies into the field) and a weakness (there are many different views as to what the field is about)

This has analogies with artificial intelligence itself

Multi-agent Systems is primarily a search for appropriate theoretical foundations:

We want to build systems of interacting, autonomous agents, but we don’t yet know what these systems should look like.

You can take a “neat” or “scruffy” approach to the problem, seeing it as a problem of *theory* or a problem of *engineering.*

This, too, has analogies with artificial intelligence research.

Over the last two decades, a major Computer Science research topic has been the development of tools and techniques to model, understand, and implement systems **in which interaction is the norm.**

Agents as a paradigm for software engineering: Software engineers have derived a progressively better understanding of the characteristics of complexity in software.

It is now widely recognized that *interaction* is probably the most important single characteristic of complex software

Agents as a tool for understanding human societies:

Multi-agent systems provide a novel new tool for simulating societies, which may help shed some light on various kinds of social processes.

This has analogies with the interest in “theories of the mind” explored by some artificial intelligence researchers.

Isn’t it all just Distributed/Concurrent Systems?

There is much to learn from this community, but:

Agents are assumed to be autonomous, capable of making independent decision – so they need mechanisms to synchronize and coordinate their activities **at run time.**

Agents are (can be) **self-interested**, so their interactions are “economic” encounters.

Isn't it all just AI?

We don't need to solve all the problems of artificial intelligence (i.e., all the components of intelligence) in order to build really useful agents.

Classical AI ignored social aspects of agency.

These are important parts of intelligent activity in real-world settings.

Isn’t it all just Economics/Game Theory?

These fields also have a lot to teach us in multi-agent systems, but:

Insofar as game theory provides *descriptive* concepts, it doesn’t always tell us *how* to compute solutions; we’re concerned with computational, resource-bounded agents.

Some assumptions in economics/game theory (such as a rational agent) may not be valid or useful in building artificial agents.

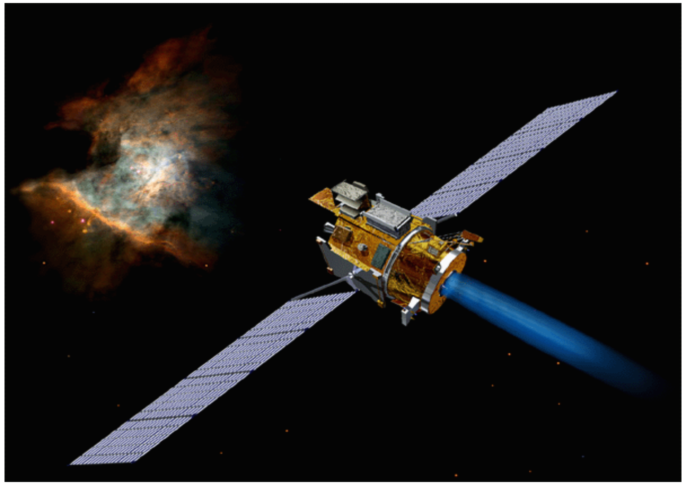
Isn’t it all just Social Science?

We can draw insights from the study of human societies, but there is no particular reason to believe that artificial societies will be constructed in the same way.

Again, we have inspiration and cross-fertilization, but hardly subsumption.

- It's easiest to understand the field of multiagent systems if you understand researchers' vision of the future
- Fortunately, different researchers have different visions
- The amalgamation of these visions (and research directions, and methodologies, and interests, and…) define the field
- But the field's researchers clearly have enough in common to consider each other's work relevant to their own

- When a space probe makes its long flight from Earth to the outer planets, a ground crew is usually required to continually track its progress, and decide how to deal with unexpected eventualities. This is costly and, if decisions are required *quickly*, it is simply not practicable. For these reasons, organizations like NASA are seriously investigating the possibility of making probes more autonomous — giving them richer decision making capabilities and responsibilities.
- *This is not fiction:* NASA's *DS1 has done it!*



- http://nmp.jpl.nasa.gov/ds1/
- "Deep Space 1 launched from Cape Canaveral on October 24, 1998. During a highly successful primary mission, it tested 12 advanced, high-risk technologies in space. In an extremely successful extended mission, it encountered comet Borrelly and returned the best images and other science data ever from a comet. During its fully successful hyperextended mission, it conducted further technology tests. The spacecraft was retired on December 18, 2001." – NASA Web site

Autonomous Agents for specialized tasks

- The DS1 example is one of a generic class
- Agents (and their physical instantiation in robots) have a role to play in high-risk situations, unsuitable or impossible for humans
- The degree of autonomy will differ depending on the situation (remote human control may be an alternative, but not always)

Air Traffic Control

- "A key air-traffic control system…suddenly fails, leaving flights in the vicinity of the airport with no air-traffic control support. Fortunately, autonomous air-traffic control systems in nearby airports recognize the failure of their peer, and cooperate to track and deal with all affected flights."
- Systems taking the initiative when necessary
- Agents cooperating to solve problems beyond the capabilities of any individual agent

- Searching the Internet for the answer to a specific query can be a long and tedious process. So, why not allow a computer program — an agent — do searches for us? The agent would typically be given a query that would require synthesizing pieces of information from various different Internet information sources. Failure would occur when a particular resource was unavailable, (perhaps due to network failure), or where results could not be obtained.

What if the agents become better?

- Internet agents need not simply search
- They can plan, arrange, buy, negotiate – carry out arrangements of all sorts that would normally be done by their human user
- As more can be done electronically, software agents theoretically have more access to systems that affect the real-world
- But new research problems arise just as quickly…

Research Issues

- How do you state your preferences to your agent?
- How can your agent compare different deals from different vendors? What if there are many different parameters?
- What algorithms can your agent use to negotiate with other agents (to make sure you get a good deal)?
- These issues aren't frivolous – automated procurement could be used massively by (for example) government agencies
- The Trading Agents Competition…