LECTURE 6: APPLICATIONS

An Introduction to Multiagent Systems

http://www.csc.liv.ac.uk/˜mjw/pubs/imas/

1 Application Areas

- Agents are usefully applied in domains where *autonomous action* is required.
- *Intelligent* agents are usefully applied in domains where *flexible* autonomous action is required.

  This is not an unusual requirement! Agent technology gives us a way to build systems that mainstream software engineering regards as hard*!*
- Main application areas:
  - distributed/concurrent systems;
  - networks;
  - human-computer interfaces.

1.1 Domain 1: Distributed Systems

- In this area, the idea of an agent is seen as a natural metaphor, and a development of the idea of concurrent object programming.
- Example domains:
  - air traffic control (Sydney airport);
  - business process management;
  - power systems management;
  - distributed sensing;
  - Space shuttle fault diagnosis;
  - factory process control.

1.2 Domain 2: Networks

- There is currently a lot of interest in *mobile* agents, that can move themselves around a network (e.g., the Internet) operating on a user's behalf.
- This kind of functionality is achieved in the TELESCRIPT language developed by General Magic, Inc, for *remote programming*.
- Applications include:
  - hand-held PDAs with limited bandwidth;
  - information gathering.

1.3 Domain 3: HCI

- One area of much current interest is the use of agent in *interfaces*.
- The idea is to move away from the *direct manipulation* paradigm that has dominated for so long.
- Agents sit 'over' applications, watching, learning, and eventually doing things without being told — taking the initiative.
- Pioneering work at MIT Media Lab (Pattie Maes):
  - news reader;
  - web browsers;
  - mail readers.

- One view:

'The 'agent' answers the phone, recognises the callers, disturbs you when appropriate, and may even tell a white lie on your behalf. The same agent is well trained in timing, versed in finding opportune moments, and respectful of idiosyncracies. ' (p150)
'If you have somebody who knows you well and shares much of your information, that person can act on your behalf very effectively. If your secretary falls ill, it would make no difference if the temping agency could send you Albert Einstein. This issue is not about IQ. It is shared knowledge and the practice of using it in your best interests.' (p151)
'Like an army commander sending a scout ahead . . . you will dispatch agents to collect information on your behalf. Agents will dispatch agents. The process multiplies. But [this process] started at the interface where you delegated your desires.' (p158)
(From *Being Digital*, by Nicholas Negroponte, Hodder & Staughton, 1995.)

2 Agents on the Internet

- The *potential* of the internet is exciting.
- The *reality* is often disappointing:
  - the Internet is *enormous* — it is not always easy to *find* the right information manually (or even with the help of search engines);
  - *systematic* searches are difficult:
    - *human factors*: we get bored by slow response times, find it difficult to read the WWW rigorously (it is designed to prevent this!) get tired, miss things easily, misunderstand, and get sidetracked;
    - *organizational factors*: structure on the net is only superficial — there are no standards for home pages, no semantic markup to tell you what a page contains;

  - the amount of information presented to us leads to 'information overload'.

- What we want is a kind of 'secretary': someone who understood the things we were interested in, (and the things we are not interested in), who can act as 'proxy', hiding information that we are not interested in, and bringing to our attention information that *is* of interest.

  This is where agents come in!
- We cannot afford *human* agents to do these kinds of tasks (and in any case, humans get suffer from the drawbacks we mentioned above).
- So we write a program to do these tasks: this program is what we call an agent.

3 A Scenario

- Here is a scenario illustrating the kinds of properties that we hope Internet agents will have:

  > Upon logging in to your computer, you are presented with a list of email messages, sorted into order of importance by your personal digital assistant (PDA). You are then presented with a similar list of news articles; the assistant draws your attention to one particular article, which describes hitherto unknown work that is very close to your own. After an electronic discussion with a number of other PDAs, your PDA has already obtained a relevant technical report for you from an FTP site, in the anticipation that it will be of interest.
- Demonstrator systems *used today*.

4 Another Scenario

'The 'agent' answers the phone, recognizes the callers, disturbs you when appropriate, and may even tell a white lie on your behalf. The same agent is well trained in timing, versed in finding opportune moments, and respectful of idiosyncrasies. ' (p150)
'If you have somebody who knows you well and shares much of your information, that person can act on your behalf very effectively. If your secretary falls ill, it would make no difference if the temping agency could send you Albert Einstein. This issue is not about IQ. It is shared knowledge and the practice of using it in your best interests.' (p151)
'Like an army commander sending a scout ahead . . . you will dispatch agents to collect information on your behalf. Agents will dispatch agents. The process multiplies. But [this process] started at the interface where you delegated your desires.' (p158)

(From *Being Digital*, by Nicholas Negroponte.)

5 Email Reading Assistants

- The staple diet of software agent researchers. . .
- Pattie Maes developed MAXIMS – best known email assistant:

  > 'learns to prioritize, delete, forward, sort, and archive mail messages on behalf of a user . . . '.
- MAXIMS works by 'looking over the shoulder' of a user, and learning about how they deal with email.
- Each time a new event occurs (e.g., email arrives), MAXIMS records the situation $\rightarrow$ action pairs generated.
- Situation characterised by features of event:
  - sender of email;
  - recipients;
  - subject line;

– etc.

- When new situation occurs, MAXIMS matches it against previousy recorded rules.

- Tries to predict what the user will do — generates a *confidence level*.
- Confidence level matched against two threshholds: “tell me” and “do it”.

  Confidence $<$ “tell me”:

  – agent gets feedback.

  “tell me” $<$ confidence $<$ “do it”:

  – agent makes suggestion.

  Confidence $>$ “do it”:

  – agent acts.
- Rules can be “hard coded”; even get help from other users.
- MAXIMS has a simple ‘personality’, (a face icon), communicating its ‘mental state’ to the user.

6 Agents for E-Commerce

- Another important rationale for internet agents is the potential for *electronic commerce*.
- Most commerce is currently done *manually*. But there is no reason to suppose that certain forms of commerce could not be safely delegated to agents.
- A simple example: finding the cheapest copy of Office 97 from online stores.
- More complex example: flight from Manchester to Dusseldorf with veggie meal, window seat, and does not use a fly-by-wire control.
- Simple examples *first-generation* e-commerce agents:
  – Bargain Finder from Andersen;
  – Jango from NETBOT (now EXCITE).

- *Second-generation*: negotiation, brokering, . . . *market systems*.

- Jango (Doorenbos et al, Agents 97) is good example of e-commerce agent.
- Long-term goals:
  1. Help user decide what to buy.
  2. Finding specs and reviews of products.
  3. Make recommendations.
  4. Comparison shopping for best buy.
  5. Monitoring “what’s new” lists.
  6. Watching for special offers & discounts.

- Isn’t comparison shopping impossible? WWW pages all different!
- Jango/ShopBot exploits several *regularities* in merchant WWW sites:
  - *navigation regularity*:
    sites designed so that products easy to find
  - *corporate regularity*:
    sites designed so that pages have same look’n’feel;
  - *vertical separation*:
    merchants use whitespace to separate products.

- Two key components of Jango/ShopBot:
  - *learning vendor descriptions*;
  - *comparison shopping*;

7 Real Soon Now

(Etzioni & Weld, 1995) identify the following specific types of agent that are likely to appear soon:

- Tour guides:
  The idea here is to have agents that help to answer the question ‘where do I go next’ when browsing the WWW. Such agents can learn about the user’s preferences in the same way that MAXIMS does, and rather than just providing a single, uniform type of hyperlink actually indicate the likely interest of a link.
- Indexing agents:
  Indexing agents will provide an extra layer of abstraction on top of the services provided by search/indexing agents such as LYCOS and *InfoSeek*. The idea is to use the raw information provided by such engines, together with knowledge of the users goals, preferences, etc., to provide a *personalised* service.

- FAQ-finders:

  The idea here is to direct users to FAQ documents in order to answer specific questions. Since FAQS tend to be knowledge intensive, structured documents, there is a lot of potential for automated FAQ servers.

- Expertise finders:

  Suppose I want to know about people interested in temporal belief logics. Current WWW search tools would simply take the 3 words 'temporal', 'belief', 'logic', and search on them. This is not ideal: LYCOS has no model of what you *mean* by this search, or what you really *want*. Expertise finders 'try to understand the users wants and the contents of information services', in order to provide a better information provision service.