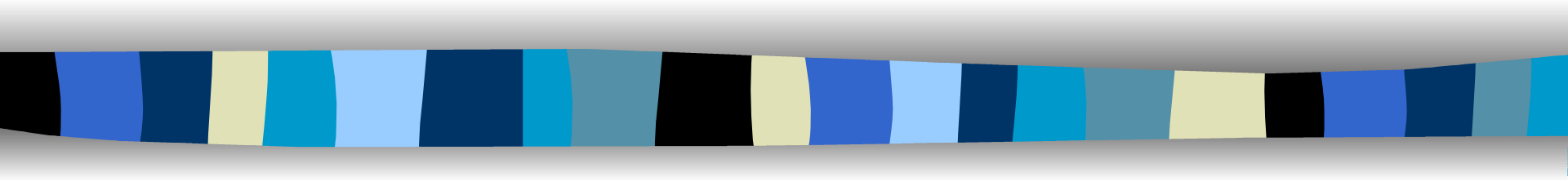
Technologies for E-Commerce Agents and Bots



slide credits:

- Peter McBurney, Univ of Liverpool
- E-commerce 2004, Prentice Hall
- Michael Huhns, Agents as Web Services, 2002

Introduction

- Software agents: -also known as intelligent agents, knowbots, softbots, or bots.
- In this lecture, we will learn more about the general features and operation of software agents in a networked world such as worldwide web.
- We will take a close look at shopbots – autonomous agents which automate part or all of the shopping experience online.

Why software agents for E.C

- Information overload
  - 10 billion (nonspam) e-mail messages were sent per day (in 2001).
  - 35 billion messages per day by 2005.
  - The amount of unique information being produced worldwide is doubling every year.
  - The amount of information in 2002 and 2001 is more information that was accessible in the entire 300,000 years of human history.
- Most data we gather goes unused.

Why software agents for E.C

- For example, according to the Gartner Group (Kyte 2002):
  - The amount of data collected by large enterprises doubles every year.
  - Knowledge workers can analyze only about 5 percent of the data.
  - Most of knowledge workers' efforts are spent trying to discover important patterns in the data (60 percent or more); a much smaller percentage is spent determining what those patterns mean (20 percent or more); and very little time (10 percent or less) is spent actually doing something based on the patterns.
  - Information overload reduces knowledge workers' decision-making capabilities by 50 percent.

What is the Solution to the problem?

- Delegate, Do not navigate
  - *…What we call "agent-based interfaces" will emerge as the dominant means by which computers and people will talk to one another.*

    Being Digital (1995).
  - in the future end users will delegate tasks such as information review and filtering to mobile agents that travel to remote data sources, examine them locally, and return with a summary of the data in a process called information filtering.

Value of the Software agents

- Information access and navigation are the major applications of the software agents in today's internet.
- Other reasons:
  - Mundane personal activity.
    - Shopping for groceries or travel planning.
  - Search and retrieval
  - Repetitive office activity.
  - Decision support.
  - Domain experts.

Definition and concepts

- "An agent is anything that can be viewed as perceiving its environment through sensors and acting on that environment through effectors." (Russell and Norvig 1995, p. 33)
- **software (intelligent) agents**
- Software agents that continuously perform three functions:
  - perception of dynamic conditions in the environment,
  - action to affect conditions in the environment,
  - and reasoning to interpret perceptions, solve problems, draw inferences, and determine actions.

A Comparison..

EXHIBIT D.1 Software Agents Versus Traditional Software Programs

Characteristics	Regular Software	Software Agents
Nature	Static	Dynamic.
Manipulation	Direct: User initiates every action	Indirect: Autonomous.
Interactivity	Noninteractive	Dialogues are fully interactive. Actions may be initiated by either the user or the agent system. Interacts with user and with other agents.
Flexibility	Never changes, unless changed by a human or an error in the program	Adapts, learns.
Temporal continuity	Runs one time, then stops to be run again when called	Persistent: Continues to run over time.
Response	Predictable: Does what you tell it to, even if you didn't mean what you said	Interprets what you mean, not what you say. In the best of circumstances, actions are based on rules, but they may change over time or in reaction to different circumstances.
Autonomy, independence	Follows instructions	May initiate actions, as well as respond to instructions.
Mobility	Stays in one place	May be mobile, traveling to other servers.
Concurrency	Generates process in one dedicated server with limited processing power	Dispatches simultaneously to accomplish various parts of a task in parallel.
Local interaction	NTBL:Accesses data across network using client-server architecture	Can travel and interact with local entities, such as data bases, file servers and stationary agent, through message passing.

Source: Based on Feldman and Yu (1999).

INTELLIGENCE LEVELS

- Definitions of agents are greatly dependent on the agents' levels of intelligence, which are described by
- Lee et al. (2002) as follows:
  - **Level 0 (the lowest).** These agents retrieve documents for a user under straight orders. Popular Web browsers fall into this category. The user must specify the URLs where the documents are. These agents help in navigating the Web.
  - **Level 1.** These agents provide a user-initiated searching facility for finding relevant Web pages. Internet search agents such as Google, Alta Vista, and Lycos are examples. Information about pages, titles, and word frequency is stored and indexed. When the user provides key words, the search engine matches them against the indexed information. These agents are referred to as search engines.

INTELLIGENCE LEVELS

- ‣ **Level 2.** These agents maintain users' profiles. They then monitor the Internet and notify the users whenever relevant information is found. An example of such an agent is WebWatcher (search for WebWatcher at cs.cmu.edu). Agents at this level are frequently referred to as semi-intelligent or software agents.
- **Level 3.** Agents at this level have a learning and deductive component of user profiles to help a user who cannot formalize a query or specify a target for a search. DiffAgent (CMU) and Letizia (MIT) are examples of such agents. Agents at this level are referred to as learning or truly intelligent agents. Similar to the concept of levels is the idea of "agent generation."

Operation of a simple Agent

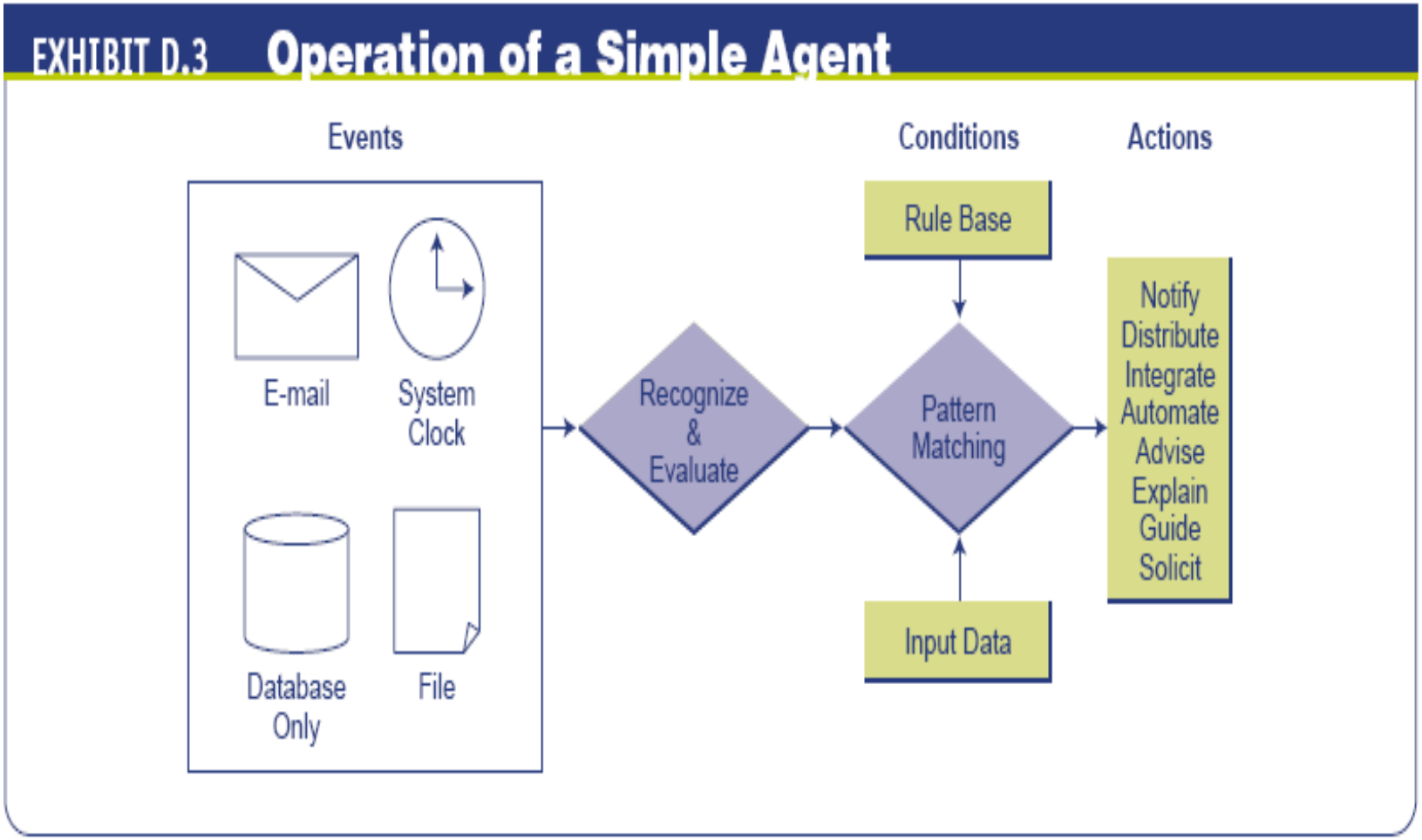
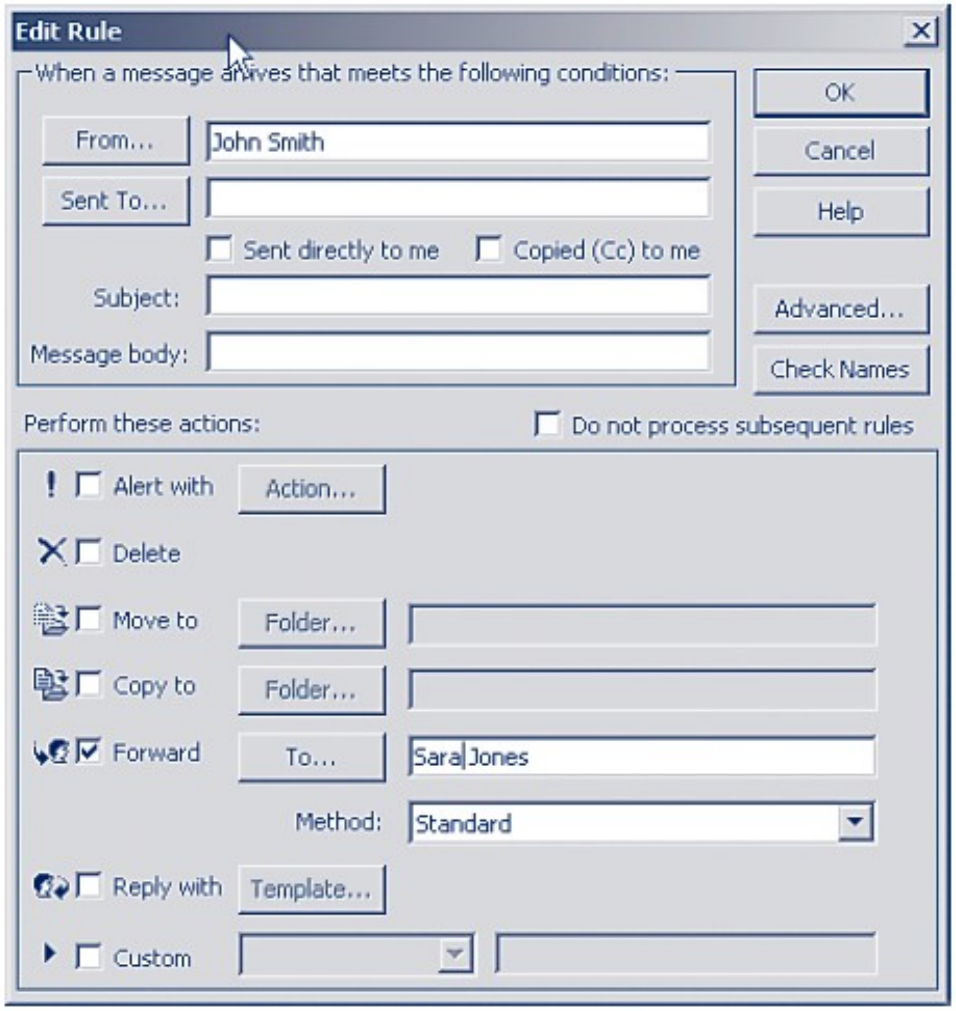


EXHIBIT D.4 Out-of-Office Agent in Microsoft E-Mail



Softbots

- Shopbots are computer programs which visit web-sites:
  - agents, spiders, robots, softbots, bots.
  - The activity of collecting information from web-pages is often called **harvesting**.

- Example applications:
  - To find a particular product:
    - e.g. to find a rare book or CD.
  - To undertake comparison shopping:
    - e.g. to find all the sites selling a particular book and find the cheapest. (http://www.addall.com)
    - e.g. to find all the sites selling a particular book and find the one which can deliver it fastest.
  - To harvest information
    - e.g. for web-page indexing; to gather email addresses; to archive publications.

A classification of bots

- Chatterbots
  - A bot which undertakes a conversation with a human
    - e.g. a front-end to a web-search engine
  - A famous chatter-bot was Eliza (1970s) which gave psychotherapy advice to humans in natural language.

- Commercial intermediaries
  - Bots which act in between 2+ companies:
    - e.g. Brokering deals; Monitoring work (e.g. delivery of goods), etc.

- Government bots
  - Helping citizens with obtaining Government information or fulfilling legal obligations
  - e.g. helping with car licensing or new business registration processes.

A classification of bots (2)

- News bots
  - e.g. Retrieving news stories; Creating and delivering personalized e-newspapers.
- Newsgroup bots
  - Management of newsgroups.
    - e.g. Screening abusive/obscene language; Grouping postings, etc.
- Update bots
  - Alerts users on changes (e.g. to a web-site)
- Web-developer bots
  - e.g. checking if links are still current; tracking hits.

Shopping Bots: a classification

- **Shopbots:**
  - Bots which act on behalf of human users who are potential customers for some product or service.
  - This classification is due to Patti Maes at MIT Media Lab.

- **Product Brokering**
  - Bot alerts user to new releases; or recommends products based on past purchases or user preferences.
  - Usually works with one potential supplier.
  - "people who bought this items also bought…."
  - e.g. recommendations on www.amazon.com.

Classification of shopbots (2)

- **Merchant Brokering**
  - Bot undertakes comparison shopping across a range of potential suppliers, collecting information, collating and analyzing it, and presenting it to the user
  - Works with multiple suppliers

- **Negotiating**
  - Bot enters auctions or undertakes negotiations on behalf of user, within parameters set by user.
  - The counter-parties may be other bots or may be humans.
  - Some Intranet or B2B examples; no B2C examples.
  - E.g. Service provisioning of new telecoms services at British Telecom.

Design decisions for shopbots (1)

- What is the human-computer interface?
  - e.g. Users can be overwhelmed if too much information is presented to them.
- How are preferences elicited from the human customer?
- Which potential sellers to include?
- Are potential sellers hard-wired or is the search done on-the-fly?
  - Most are hard-wired, because:
    - Searches are faster
    - The sellers have to provide information in an agreed format.
  - Whereas, web-searches usually search on-the-fly.
    - They don't keep a list of preferred web-pages ready in case someone asks for them!

Design decisions for shopbots (2)

- When does the shopbot cease waiting to receive an offer?
- How are offers shortlisted?
  - They may use some model of multi-attribute decision-making.
- How are offers displayed?
  - The display can impact the user's choices.
- How much autonomy does the shopbot have?

Design decisions for shopbots (3)

- Can the shopbot learn from experience?
  - e.g. To learn the user's preferences by observing his/her actual purchase decisions.
  - e.g. To learn which suppliers are reliable, etc.
- Can the shopbot predict market trends?
  - e.g. To suggest to human user that prices may fall if he/she waits a week.
- What aspects of the human decision are included?
  - e.g. The cost of waiting; the cost of information processing?

Does the bot need to know XML?

- Related question: Do the vendor sites have to be encoded using XML (or be otherwise semantically annotated)?

- No, because most sites have certain regularities:
  - Navigation regularities (so that consumers can find stuff quickly)
  - Uniform look and feel
  - Vertical separation (new products on a new line, etc)
  - Common symbols (e.g. prices quoted with a "£" symbol in front of them)

- Difficulties arise in some domains:
  - Those without regularities (e.g. entertainment industry sites)
  - Those using text embedded in graphics
    - Hard or impossible to parse the text
    - Common for company logos.

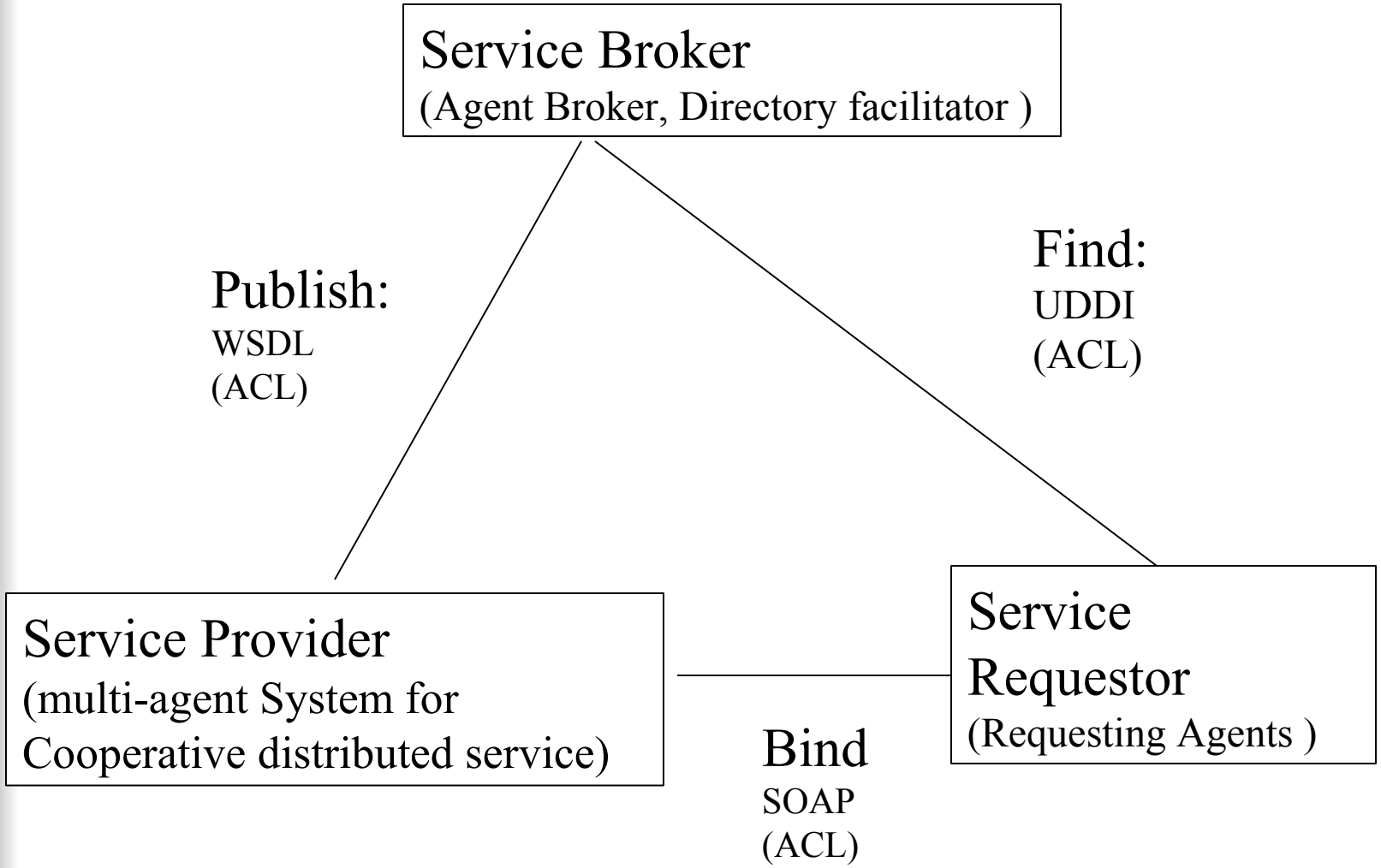
- Parsing may need to be vendor-specific.

Semantic Web

- **semantic Web**
  - A tool that provides a content presentation and organization standard so that content can be shared safely among different software applications, using mutually understandable semantic constructs.

Current Standard for Web Services

- Current Standard for Web Services:
  - XML (extensible markup language) provides the common service providers and requestors need to connect and exchange information.
  - SOAP (The simple object protocol) provides the common protocol systems need to communicate with each other so that they can request services. Example, Schedule appointments, order parts and deliver information.
  - WSDL (the web services description language) describes the services in a machine readable form, where names of functions, their required parameters, and their results can be specified.
  - Finally, UDDI (universal descriptions, discovery, and integration) gives clients- users and businesses –a way to find needed services by specifying a registry or "yellow pages" of services.



Source: Huhns, Agents as Web Services, 2002

ADVANTAGES AND LIMITATIONS OF SEMANTIC WEB

Using semantic Web to design intelligent agents has the following advantages:

- **Easy to understand.** Semantic Web demonstrates objects and their relationships as graphic templates for easy understanding.
- **Easy resource integration.** It is easier to integrate systems and modules designed in semantic Web. This also makes it easier for system analysis and maintenance.
- **Saving development time and costs.** Semantic Web allows incremental ontology creation, enabling more rapid system development and lower development costs.

Advantages…

- **Automatic update of content.** Because agents can easily locate a specific knowledge on semantic Web, they can have functions to update or import contents automatically. This adds the level of intelligence to the software agents.
- **Easy resource reuse.** The ontology-based annotations can turn briefings into reusable resources.

ADVANTAGES AND *LIMITATIONS* OF SEMANTIC WEB

The limitations of semantic Web include the following:

- **The graphical representation may be oversimplified.** For example, using an arrow to represent a relation between two instances is unable to show more complicated multiparty relations.
- **Additional tools for searching content and building references to preexisting instances are needed** for effective use of semantic Webs.

Limitations..

- **Ontologies may not be correctly defined.** In some cases, the outcome of this could be severe. It is still hard to prove the completeness or correctness of a defined ontology.
- When agents deal with a semantic Web containing information that is inconsistent, incorrect, or unreliable, the **agents could become contaminated or be misled.**
- Because the semantic Web allows agents from different systems to communicate and share information, security is a key concern. **Security is always a problem for an open system.**