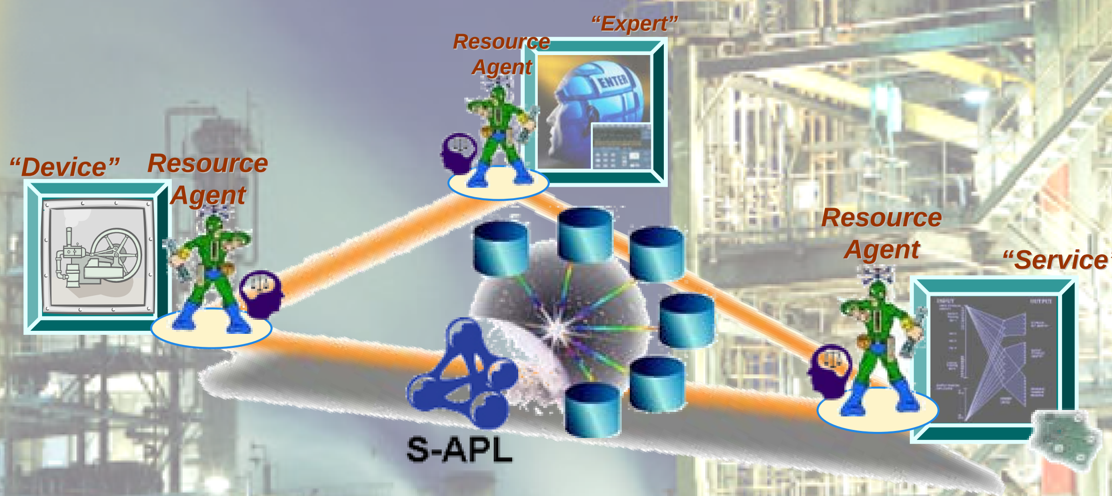
"Advanced IT from Multiagents to Semantic Web"

Towards the "Web of Everything": role of semantic and agent technologies

by Vagan Terziyan



Industrial Ontologies Group
University of Jyväskylä

March-April, 2010

Introduction: about the lecturer and his research team

- Vagan Terziyan brief profile;
- Industrial Ontologies Group

Presenter's short BIO

Vagan Terziyan has got his M.Sc. degree in Applied Mathematics in 1981 from Kharkov National University of Radioelectronics (KNURE) in Ukraine; later in 1985 from same university he got Ph.D. degree in Technical Cybernetics, then in 1993 – the Degree of Doctor of Technical Sciences in Artificial Intelligence. He got academic title of Professor in Software Engineering from Supreme Certifying Commission of Ukraine in 1996. He was acting as Professor since 1994 and as the Head of Artificial Intelligence Department in KNURE since 1997. He has been awarded the title "Distinguished Teacher of Ukraine" and the medal "For Distinguished Service". He was acting on various research and/or teaching positions in the University of Jyväskylä (Departments: MIT, CS&IS, TITU, Agora Center) during 1996 – 2006 due to various grants, temporal contracts and projects. In 2001 he has been nominated as a Docent (AI and Knowledge Management) by MIT Department, University of Jyväskylä; later in October 2006 he has been elected as a Professor (Distributed Systems) and then in 2008 invited to the permanent appointment. He has also been a visiting lecturer in Vrije Universiteit Amsterdam (the Netherlands) and ITIN (France). His research and teaching profile is design of distributed, intelligent and secure Web applications, systems and services, which are: (a) targeted to the needs of industry; (b) able to automatically discover, compose and integrate heterogeneous components; (c) able to manage heterogeneous data sources; and (d) utilizing for that emerging Knowledge-, Agent-, Machine-Learning-, Mobile-, Context-Aware- and Semantic Web- based technologies and tools. He leads a research group (Industrial Ontologies Group), has been a project leader in several Tekes projects and also has much of international project experience. Vagan Terziyan has recently celebrated his 50th Anniversary. Read more in: http://www.cs.jyu.fi/ai/vagan .

Industrial Ontologies Group

University of Jyväskylä

AGORA
HUMAN TECHNOLOGIES CENTER

UBIWARE Team

- **Researchers**
  - **Vagan Terziyan** (Head)
  - **Olena Kaykova**
  - **Oleksiy Khriyenko**
  - **Sergiy Nikitin**
  - **Michal Nagy**
  - **Artem Katasonov**
  - **Michal Szydlowski**
  - **Joonas Kesäniemi**
  - **Michael Cochez**

- **Contact Person:**

Timo Tiihonen

- **e-mails:**
  - *timo.tiihonen@jyu.fi*
  - *vagan.terziyan@jyu.fi*
- **phone:** *+358 14 260 2741*

URL: http://www.mit.jyu.fi/ai/OntoGroup

Group Profile

http://www.mit.jyu.fi/ai/Industrial_Ontologies_Group_booklet_print.doc

Technologies

- Semantic Web, Metadata, Ontology Engineering, Semantic Technology;
- Agents and Multiagent Systems;
- Distributed Systems, Web X.0, Distributed, Autonomic and Proactive Computing;
- Internet of Things, Ubiquitous/Pervasive Computing, Embedded Systems, RFID;
- Web of Services, Service-Oriented Architectures, Cloud Computing;
- Integration, Interoperability, Middleware, Web-Based Portals and Platforms;
- Artificial Intelligence, Data/Web Mining;
- Self-Managed, Context-Aware Systems;
- Software Engineering

Application Areas

- Industrial Automation, Monitoring, Diagnostics, Control, etc.;
- Power and Process Industry;
- Product-Centric Applications and Life-Cycle Management;
- Electronic Commerce;
- Logistics;
- Future Internet;
- Healthcare, eHealth and Wellness;
- Nanotechnology;
- Military;
- Collaborative Traffic;
- Education;
- … etc.

Integrated Profile

"*Design Platform, Semantic Middleware and Execution / Integration / Coordination Environment for Proactive, Self-Managed, Ubiquitous, Web-Based, Distributed Industrial Resources and Systems of Different Nature*"

1. The "Web of Everything" Roadmap

Warning: this presentation contains innovative or/and sensitive material, including personal ideas and future business concepts, or information related to other projects, which are meant to be presented as a lecture to students. This presentation cannot be sent to anybody outside course participants without permission from: vagan.terziyan@jyu.fi

Acknowledgement: author is grateful to all anonymous or named authors of various Web content, slides, texts, pictures, which have been used, compiled into or cited in this presentation.

1. The “Web of Everything” Roadmap

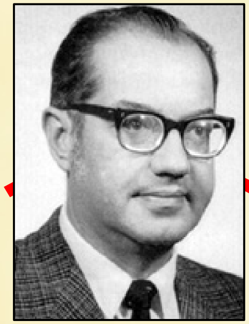
- 1.1. Before the Web (Internet);
- 1.2. Web 1.0 (Web of Shared Information);
- 1.3. Web 2.0 (Social Web);
- 1.4. Web 2.1 (Web of Things);
- 1.5. Web 2.2 (Web of Services);
- 1.6. Web 3.0 (Semantic Web);
  - Semantic Web Basics;
- 1.7. Web 4.0 (Web of Intelligence);
  - Agent Technology Basics;
- 1.8. Web 4.1 (Web of Context);
- 1.9. Web 4.2 (Web of Policies);
- 1.10. Web 4.3 (Web of Configurations);
- 1.11. Web 4.4 (Web of Presentations);
- 1.12. Web 5.0 (Global Understanding Environment);
- 1.13. Beyond Web 5.0 (Human 2.0 ?)

1.1. Before the Web

The Internet

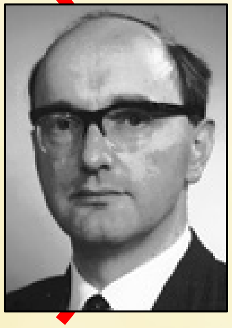
Internet (≈ 1969-1973)

The ***Internet*** is a massive network of networks. It is a networking infrastructure that connects millions of computers together globally, forming a network in which any computer can communicate with any other computer as long as they are both connected to the Internet.

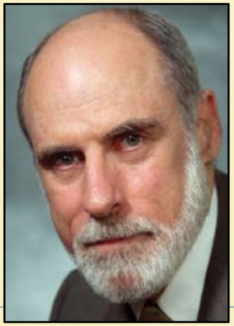


Paul Baran

Leonard Kleinrock



Donald Davies



Vinton Cerf

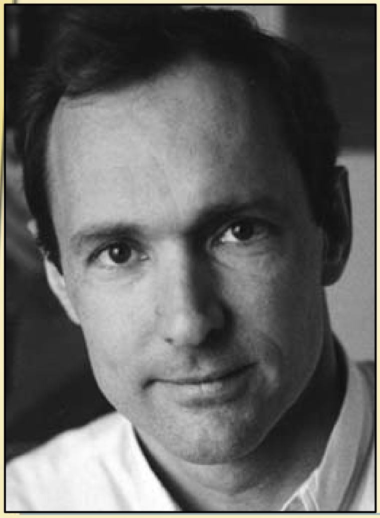
1.2. Web 1.0

Web of Shared Information

Web (World Wide Web) (≈ 1989-1990)

The ***World Wide Web***, or simply ***Web***, is a way of accessing information over the medium of the Internet. It is an information-sharing model that is built on top of the Internet.

Internet vs. Web: The Web is just one of the ways that information can be disseminated over the Internet. The Internet, not the Web, is also used for e-mail, news groups, instant messaging, FTP, etc. So the Web is just a portion of the Internet and the two terms are not synonymous and should not be confused.



Tim Berners-Lee

The World Wide Web is a "killer application" for the Internet !

Current Web (Web 1.0, Syntactic Web)

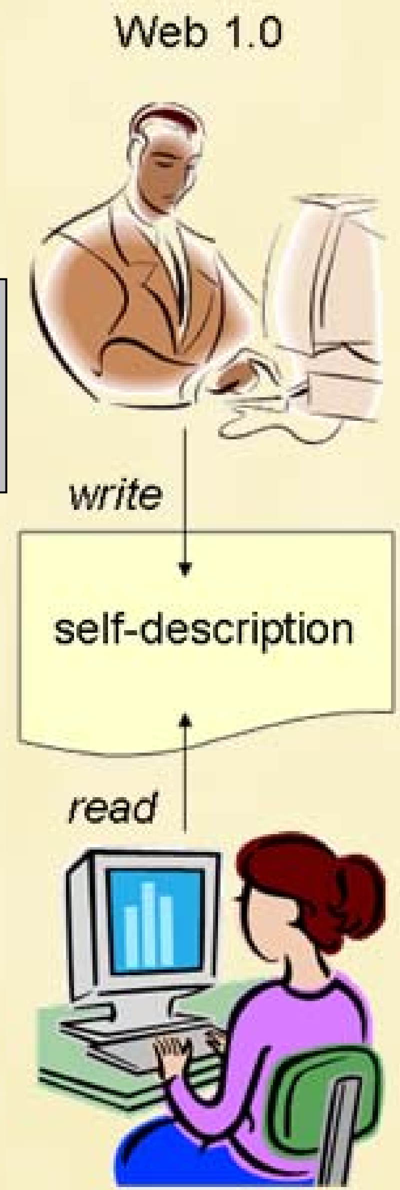
Web 1.0 connects people to the content of static pages published in the World Wide Web.

Facilitates Information-to-Information interaction

In Web 1.0, a small number of writers create Web pages for a large number of readers.

Number of Public Web Pages

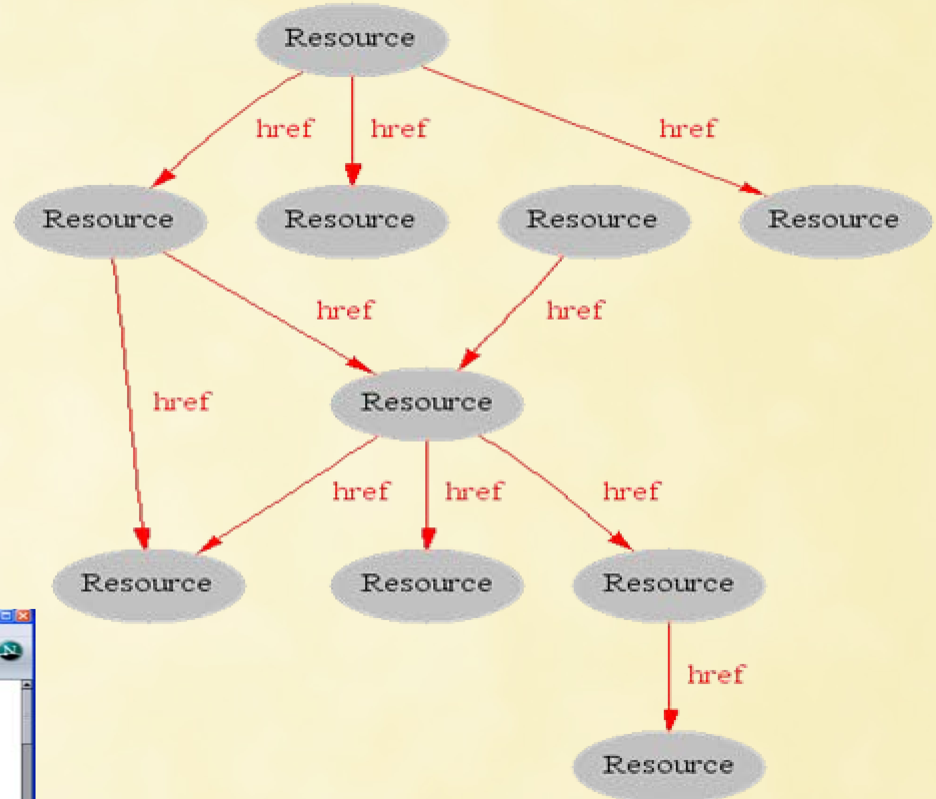
1990: 1
1998: 26 million (~26,000,000)
2008: >1 trillion (~1,000,000,000,000)
… more than the number of neurons in the human brain



Are we happy with it ?

Is it good for real technical applications ?

Web 1.0: Syntactic Web



[Hendler & Miller 02]

Web 1.0 is not enough any more

Sharing
Information is Power!

- However we, engineers, expect Web to be not only a tool for ***information sharing*** but also a tool to support at least:
  - ***Automation, Interoperability and Integration***
  - ***Linking people, data and diverse devices***
  - ***Everyone, everywhere anytime access to everything***
  - ***Collaboration and Coordination***
  - ***Service Provisioning***
  - ***Enabling Business in the Web***
  - ***Intelligence (smart information retrieval and extraction, knowledge discovery, reasoning)***
  - ***…***

"The Web is not just technology but Humanity Connected by Technology and what that technology and those links can do to empower all people" [Steve Bratt, W3C]

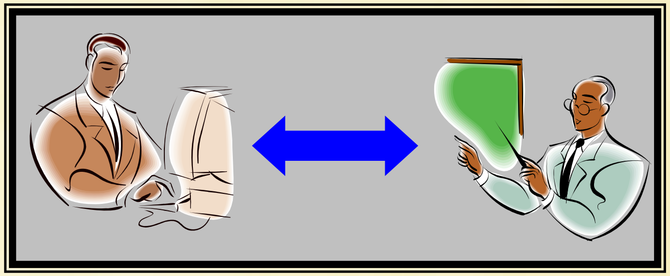
- Therefore there are **several trends** towards next generation Web aimed to meet new challenging requirements

1.3. Web 2.0

Social Web

Web of Humans (Social Web, Web 2.0)

Human Communities



Web 2.0

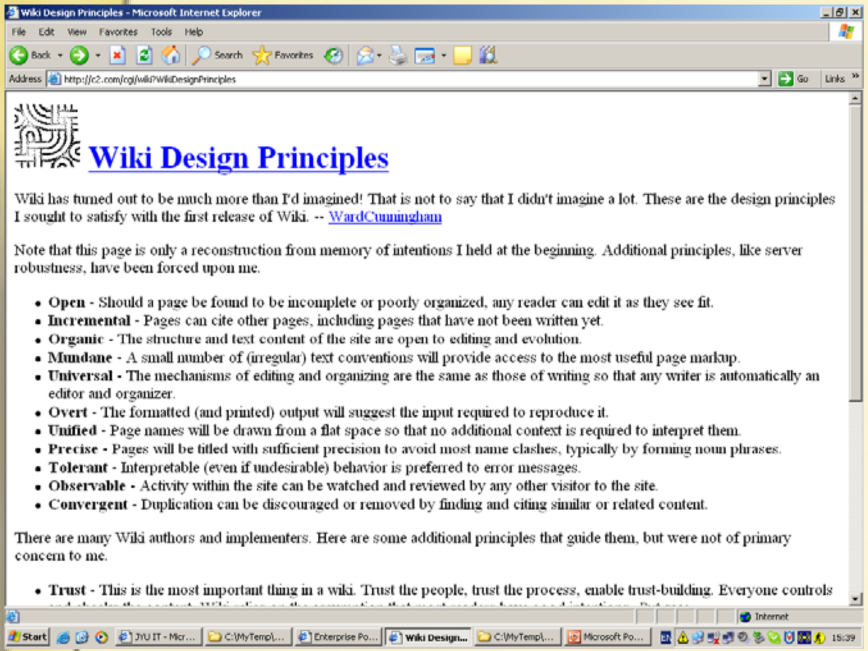
- Wikis
- Blogs
- Mashups
- Social Networks

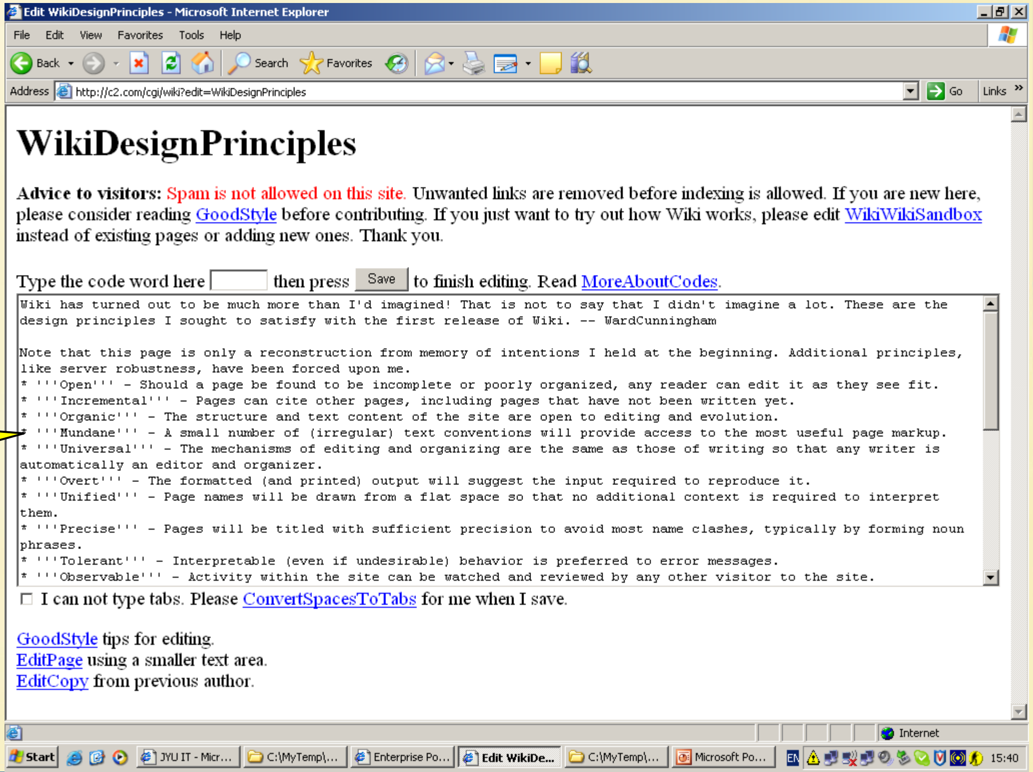
Facilitates Human-to-Human interaction

HaaS: Human-as-a-Service

HaaU: Human-as-a-User

Sample of Wiki Web page



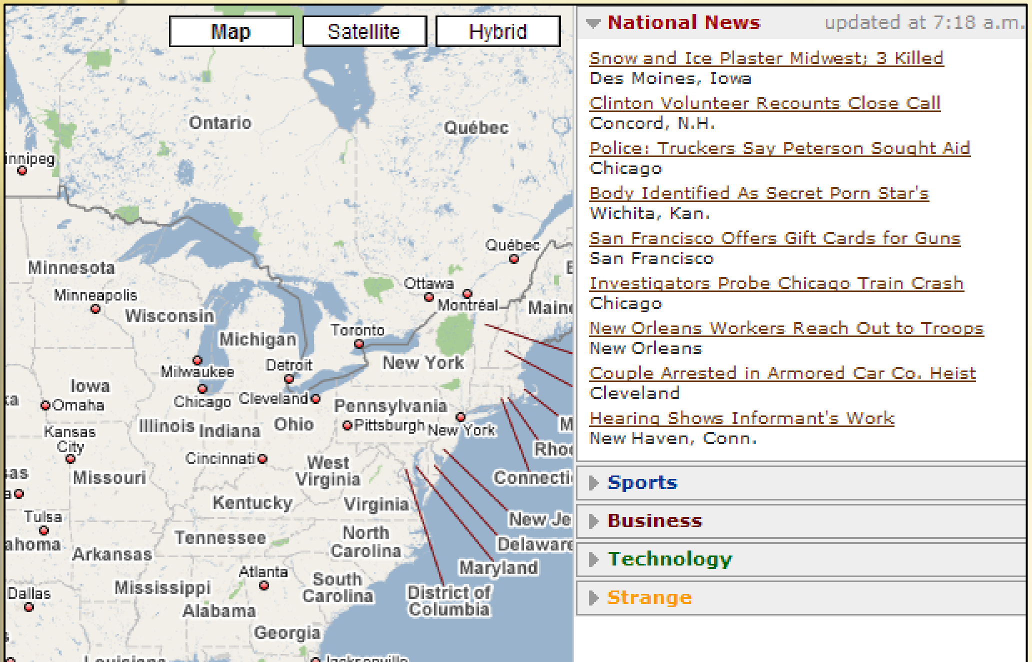


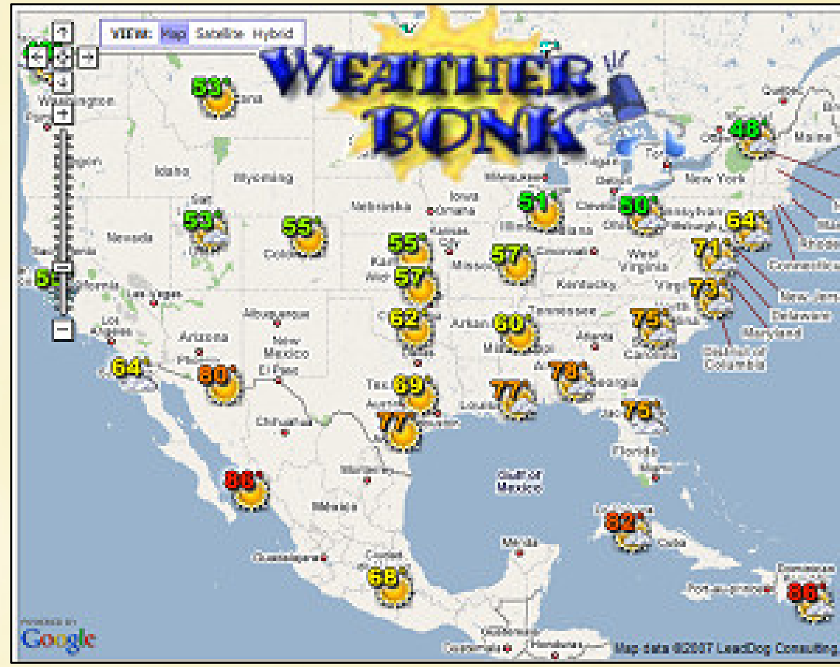
Collaborative editing window

Wikipedia

Wikipedia - Wikipedia, the free encyclopedia - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address http://en.wikipedia.org/wiki/Wikipedia
Go
Links
Sign in / create account
article
discussion
view source
history
Your continued donations keep Wikipedia running!
Wikipedia
The Free Encyclopedia
navigation
Main Page
Community Portal
Featured content
Current events
Recent changes
Random article
Help
Contact Wikipedia
Donations
search
Go
Search
toolbox
What links here
Related changes
Upload file
Special pages
Wikipedia
From Wikipedia, the free encyclopedia
Wikipedia is a multilingual, Web-based free content encyclopedia project. The name is a portmanteau of the words wiki and encyclopedia. Wikipedia is written collaboratively by volunteers, allowing most articles to be changed by almost anyone with access to the Web site. Its main servers are in Tampa, Florida, with additional servers in Amsterdam and Seoul.
Wikipedia was launched as an English language project on January 15, 2001, as a complement to the expert-written and now defunct Nupedia, and is now operated by the non-profit Wikimedia Foundation. It was created by Larry Sanger and Jimmy Wales; Sanger resigned from both Nupedia and Wikipedia on March 1, 2002. Wales has described Wikipedia as "an effort to create and distribute a multi-lingual free encyclopedia of the
Wikipedia
English
Deutsch
Français
Polski
日本語
Nederlands
Svenska
Italiano
Português
Español
URL
http://www.wikipedia.org/
Commercial?
No
Type of site
Internet encyclopedia project
Internet
Start
JYU IT - Signout - Micros...
Wikipedia - Wikipedi...
C:\MyTemp\vagan\Proje...
Microsoft PowerPoint - [...
EN
12:22

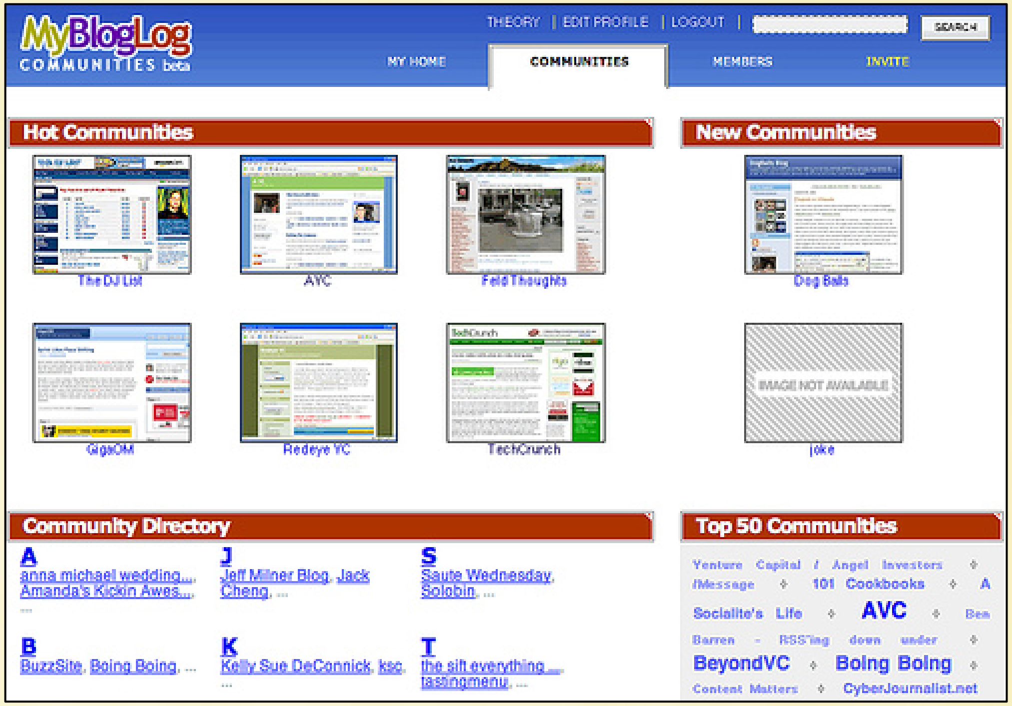
Web 2.0: Mashups





- Mashup is a term that's become popular to describe Web 2.0-ish sites that combine the features or functions of one website with another. Website mashups, created by clever programmers typically feature a high level of interactivity, user input, social networking, and sometimes even encourage people to use them as the basis for derivative works. The most common mashups involve maps, but there are also video mashups, photo mashups, search and shopping mashups, and news mashups. Website developers can use data feeds and application programming interfaces (APIs) provided by established sites such as Google, Yahoo, Microsoft, Amazon, Ebay and others, which are created specifically to encourage mashups.

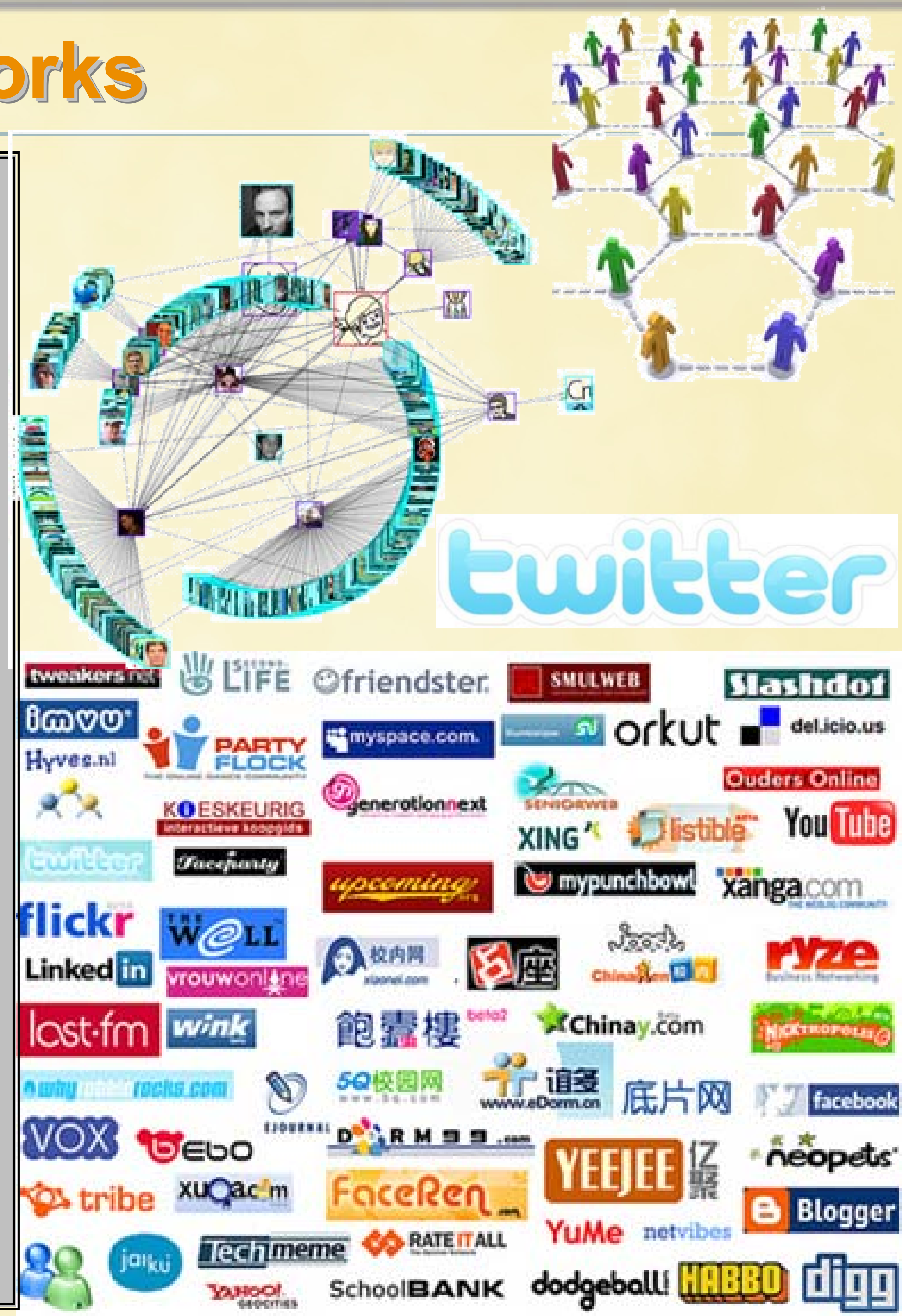
Web 2.0: Blogs



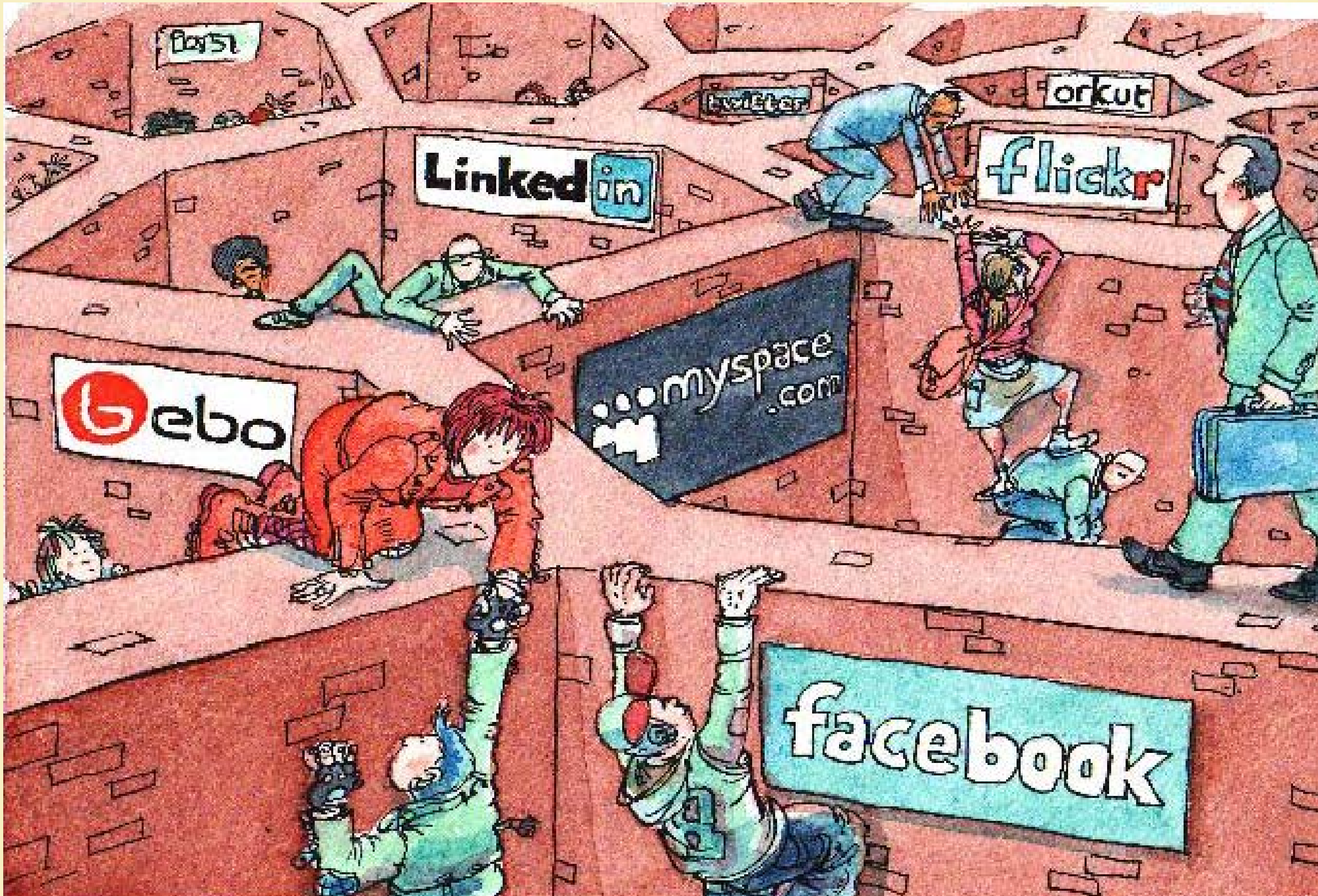
- A **blog** (we**b log**) is a website where entries are written in chronological order and commonly displayed in reverse chronological order. Many blogs provide commentary or news on a particular subject; others function as more personal online diaries. A typical blog combines text, images, and links to other blogs, web pages, and other media related to its topic. The ability for readers to leave comments in an interactive format is an important part of many blogs. Most blogs are primarily textual, although some focus on art (artlog), photographs (photoblog), sketchblog, videos (vlog), music (MP3 blog), audio (podcasting) and are part of a wider network of social media. Micro-blogging is another type of blogging which consists of blogs with very short posts. As of September 2007, blog search engine Technorati was tracking more than 106 million blogs.

Social Networks

A **social network** is a social structure made of individuals (or organizations) called "nodes," which are tied (connected) by one or more specific types of interdependency, such as friendship, kinship, financial exchange, dislike, sexual relationships, or relationships of beliefs, knowledge or prestige. Social network analysis views social relationships in terms of network theory about *nodes* and *ties*. Nodes are the individual actors within the networks, and ties are the relationships between the actors. The resulting graph-based structures are often very complex. There can be many kinds of ties between the nodes. [WIKIPEDIA]



Interoperability issue in social networks: the "*Walled Gardens*" problem



David Simonds, The Economist

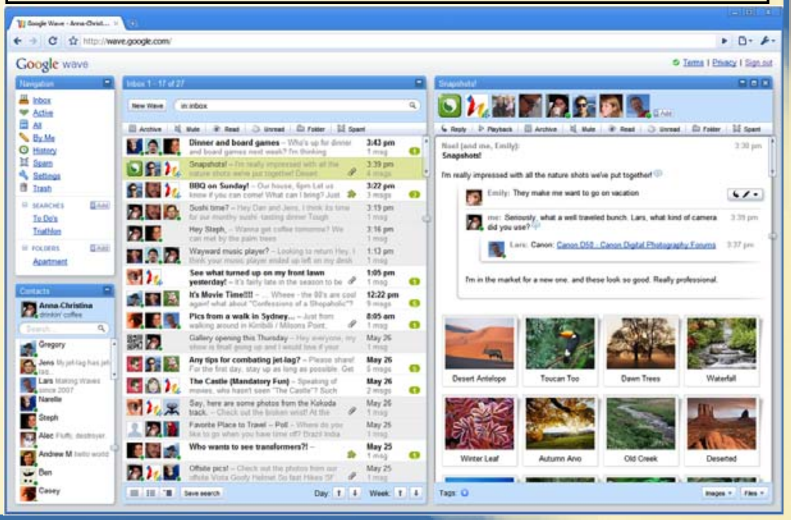
Web-Based Communication and Collaboration

Google Wave is a new model for communication and collaboration on the Web. It will soon become an amazing hub for collaboration, participation and in some way redefine how most of us will communicate online

A **wave** is equal parts conversation and document. People can communicate and work together with richly formatted text, photos, videos, maps, and more. A wave is shared. Any participant can reply anywhere in the message, edit the content and add participants at any point in the process. Then playback lets anyone rewind the wave to see who said what and when. A wave is live. With live transmission as you type, participants on a wave can have faster conversations, see edits and interact with extensions in real-time.

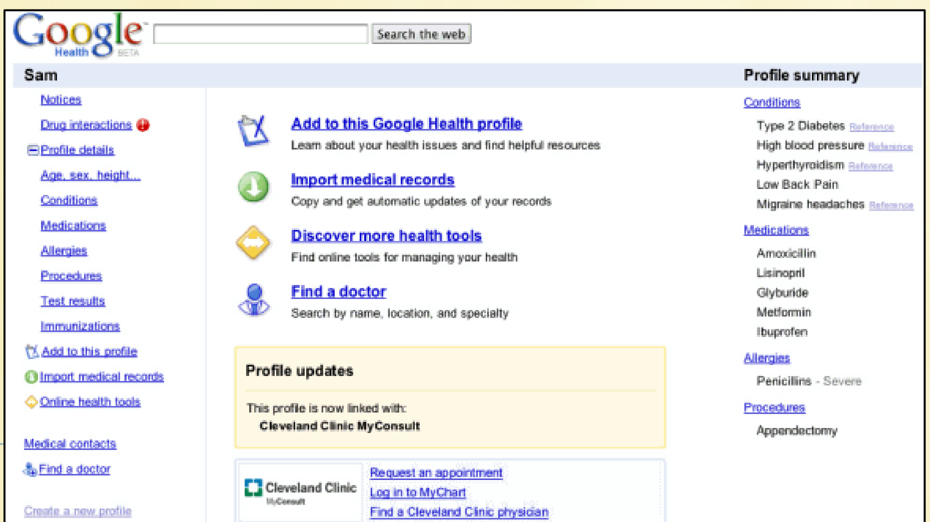
Google wave

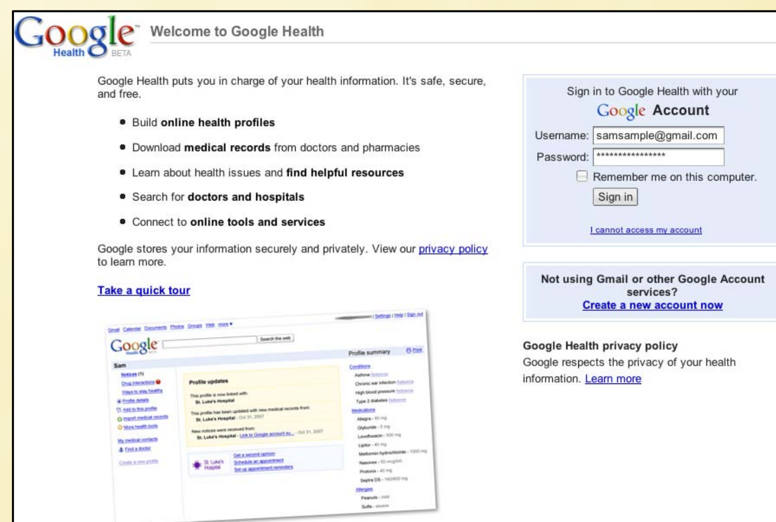
In a nutshell, Google Wave is an effort to combine everyone's growing communication needs (email, chat, photo and video sharing, working with documents and spreadsheets and social networking) into one online platform.



Online Healthcare and Wellness

Google Health stores (securely and privately) and manages all your health information in one place. You can create and save a Google Health profile using your free Google Account. You can enter as little or as much information as you want-for example, conditions, medications, and allergies. You can read about symptoms, causes, and treatments. You can even create additional profiles for your kids, your parents, or anyone you care. System allows you to import your records and prescription history from healthcare providers that treat you. Linking accounts with these partners is secure. Just identify yourself by signing in at the partner's site with the username and login that you have with them, and then confirm that you want to link accounts and transfer data to Google Health. Every time you add new health data to your profile, Google Health will check for potential interactions between your drugs, allergies, and conditions. To better coordinate your care you may share your health records with family members, friends, doctors or anyone else in your care network. You can also print a wallet-sized version of your health profile to share with your doctor or family members in person, or for use in case of emergency. You can get personalized health information based on your profile. You can link external services in the same way you link to other partners to import your medical records. Google has no financial relationship with any of service providers. You decide whether to connect with a service and share your health information with it.

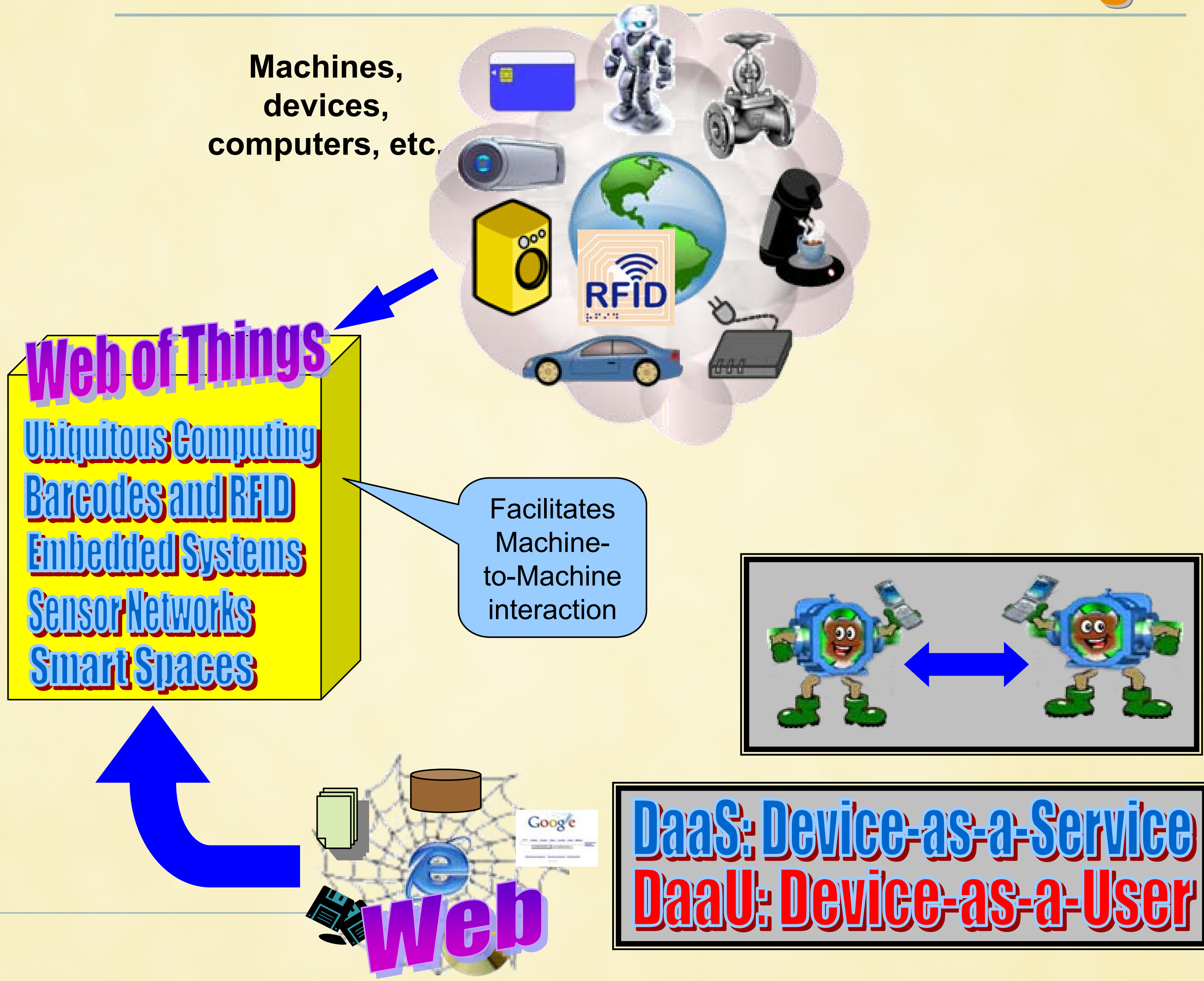




1.4. Web 2.1

Web of Things

Web of Things



Internet of Things (we knew much earlier)

TIME IN PARTNERSHIP WITH CNN

SEARCH

FRIDAY, NOVEMBER 28, 2008

HOME | U.S. | ELECTION '08 | WORLD | BUSINESS & TECH | HEALTH & SCIENCE | ENTERTAINMENT | PHOTOS | PEOPLE | BEST & WORST LISTS | MAGAZINE | TRAVEL

TIME's Best Inventions of 2008

30. The Internet Of Things

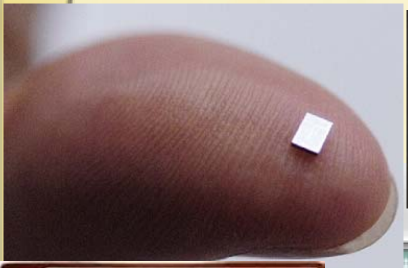
ARTICLE TOOLS
- Print
- Email
- Sphere
- AddThis
- RSS
- Yahoo! Buzz

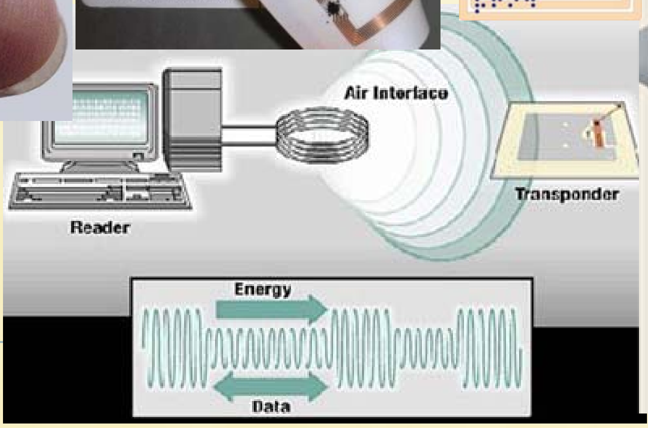
In September, a group of high-tech companies that includes Cisco and Sun formed the IP for Smart Objects Alliance. Simply put, the organization intends to create a new kind of network that will allow sensor-enabled physical objects — appliances in your home, products in a factory, cars in a city — to talk to one another, the same way people communicate over the Internet.

ILLUSTRATION FOR TIME BY CHRISTOPH NIEMANN

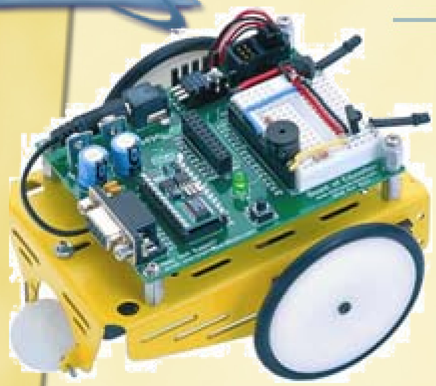
RFID: Radio-Frequency Identifiers

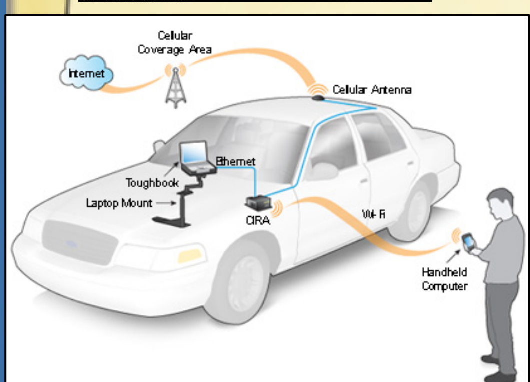
Radio-frequency identification (RFID) “tags”, or microchips with antennas, allow the automatic identification and localization of objects and people using radio waves. Data collection and their registration become possible. RFID are active (with batteries) or passive (needing an outside signal/impulse). They are used in pricing, cashing and inventory of products (increasingly instead of barcodes) , in biometric passports and credit cards, tickets, etc... Widely spread RFID tags reaching the size of a few mm – the size of rice, which implies the possibility for such chips to be embedded for example in ordinary sheets of paper. In 2009 it was demonstrated that RFIDs can be glued to living ants. Thus, RFIDs could soon be embedded in virtually everything bought, worn, driven, or read, enabling tracking everything and everybody wherever they go. As such, RFIDs are considered as an important component of the future “Internet of Things”. A seamless, global network of electronic scanners will be able to scan radio tags in a variety of public settings. Emerging computational RFID tags could analyze and possibly take action depending on situation. Such RFIDs could play a role in the transformation of the Internet from a network of computers to a network of things and even further – to a network of smart things.





Embedded Systems

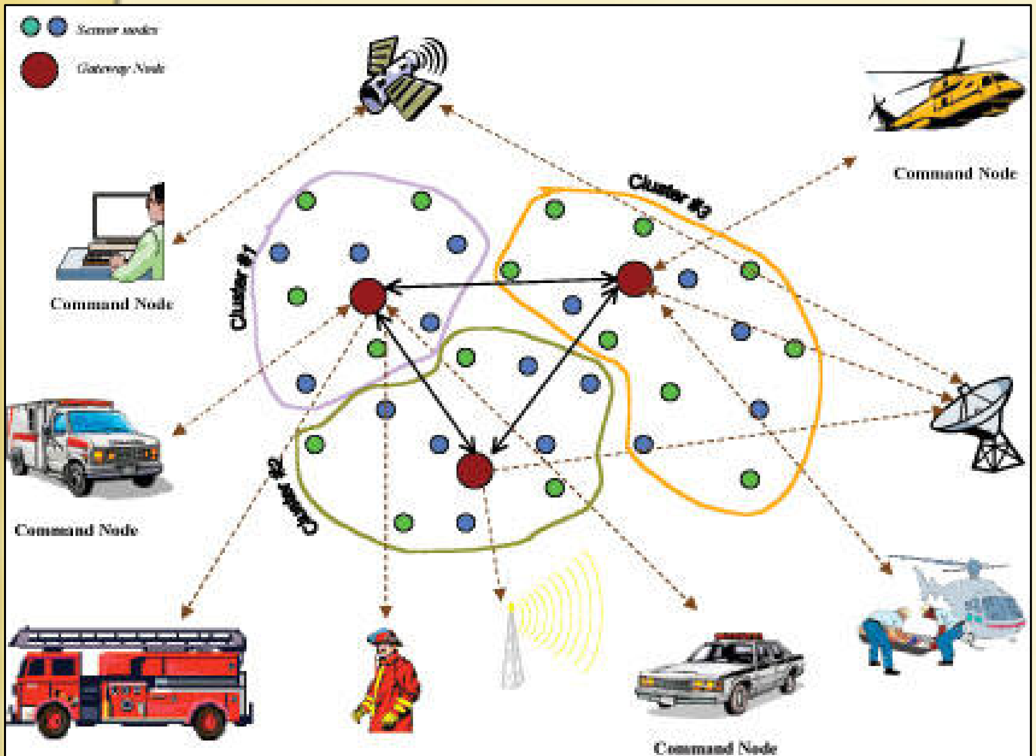


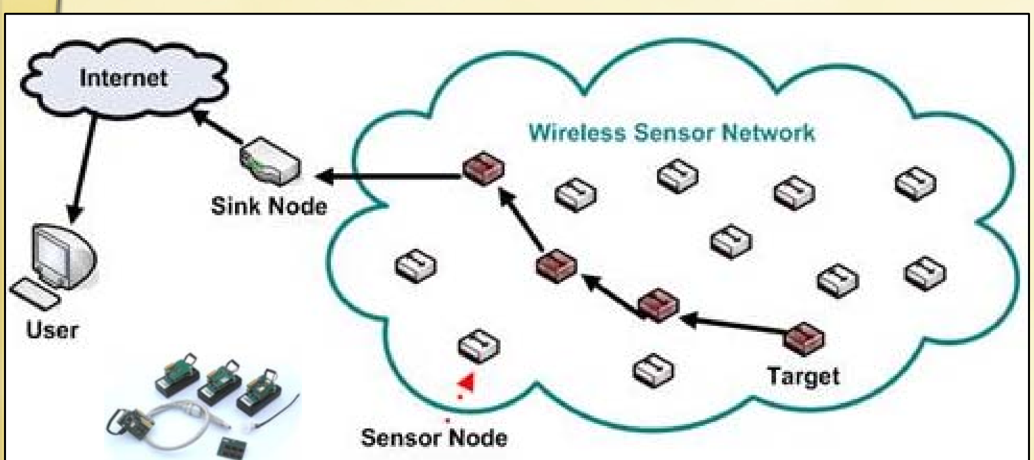


… any electronic system that uses a computer chip, but that is not a general-purpose workstation, desktop or laptop computer. Such systems use microcontrollers or microprocessors, or they may use custom-designed chips. Deployed by the billions each year in myriad applications, the embedded systems market uses the lion's share of all the electronic components in the world. Embedded systems are employed in automobiles, planes, trains, space vehicles, machine tools, cameras, consumer electronics, office appliances, network appliances, video games, cellphones, PDAs, GPS navigation as well as robots and toys.

"Computer"

Sensor Networks



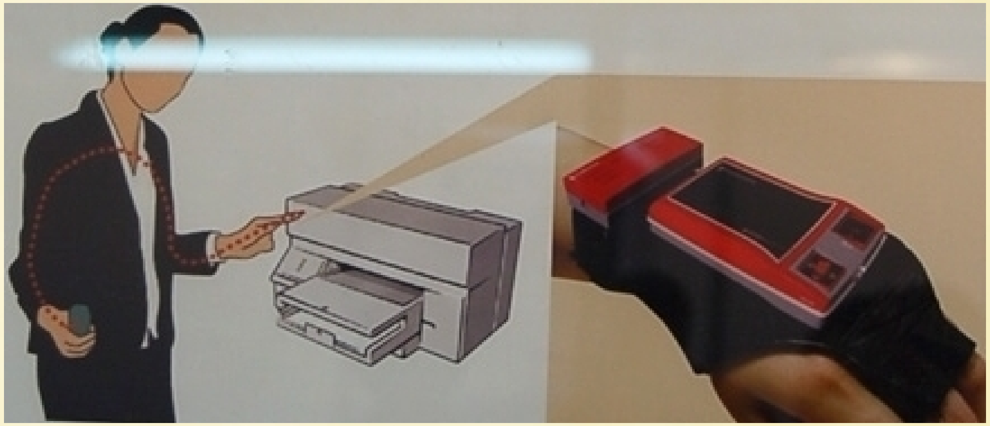


A (wireless) **sensor network** consists of spatially distributed autonomous sensors to cooperatively monitor physical or environmental conditions. Now used in many industrial and civilian application areas, including industrial process monitoring and control, machine health monitoring, environment and habitat monitoring, healthcare applications, home automation, and traffic control. In addition to one or more sensors, each node in a sensor network is typically equipped with a radio transceiver or other wireless communications device, a small microcontroller, and an energy source, usually a battery. Wikipedia

Body Area Networks

WBAN or BAN, short for **Wireless Body Area Network** handles communication between devices using the human body as a medium. WBAN consists of a set of mobile and compact intercommunicating sensors, either wearable or implanted into the human body, which monitor vital body parameters and movements.

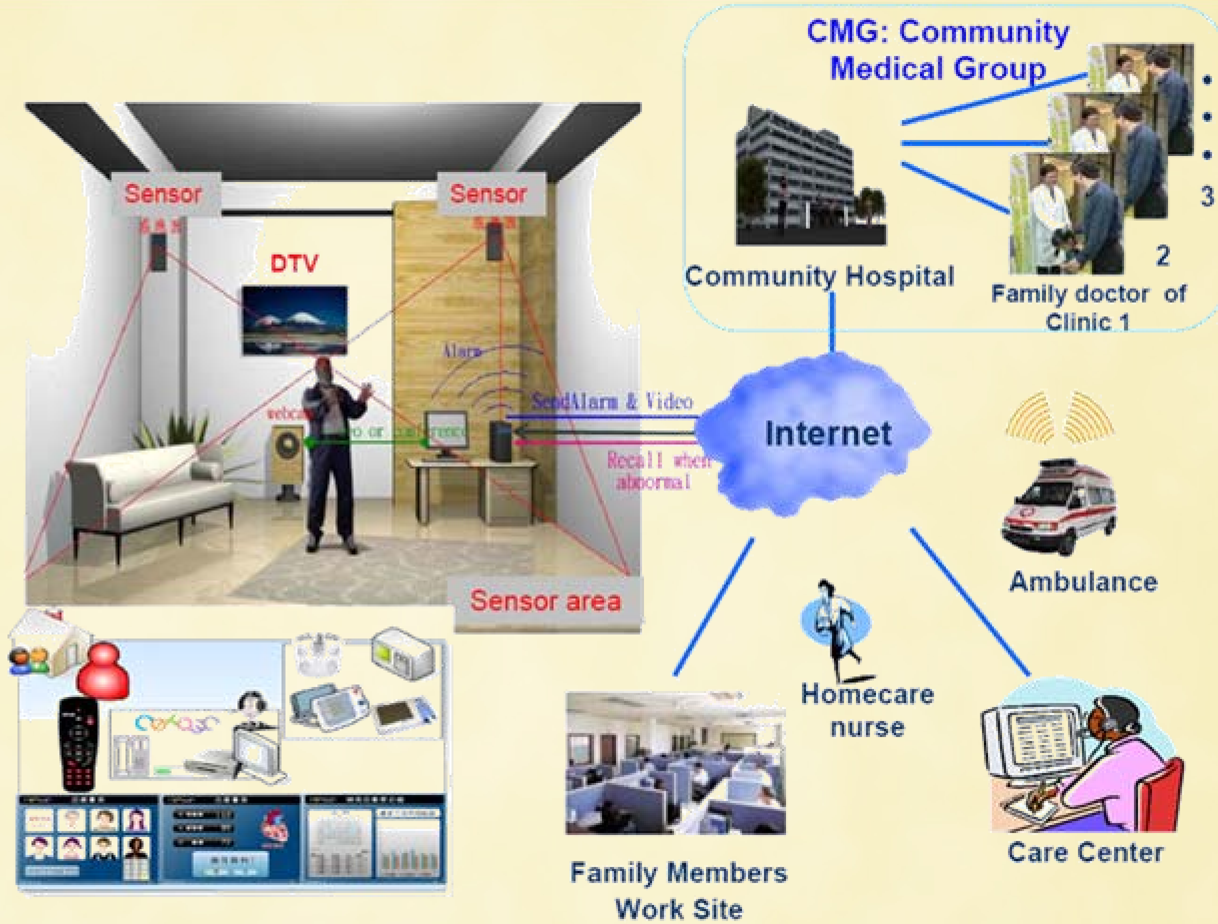
Body Area Network uses the human body as the electrical conduit between devices. One can print content of her mobile device by touching printer.

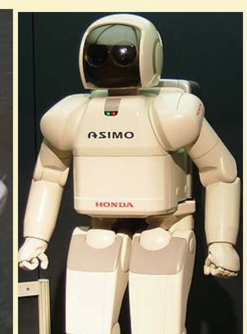


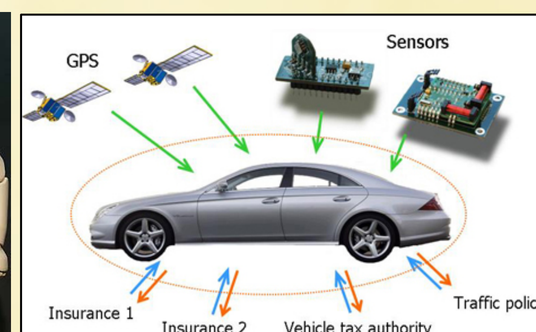
Two people wearing body area networks can exchange electronic business cards via shaking hands

Smart Spaces

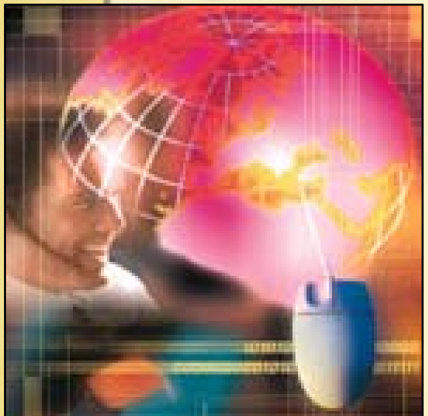
Smart Spaces or smart environments is a technological concept that, according to Mark Weiser, is "a physical world that is richly and invisibly interwoven with sensors, actuators, displays, and computational elements, embedded seamlessly in the everyday objects of our lives, and connected through a continuous network"

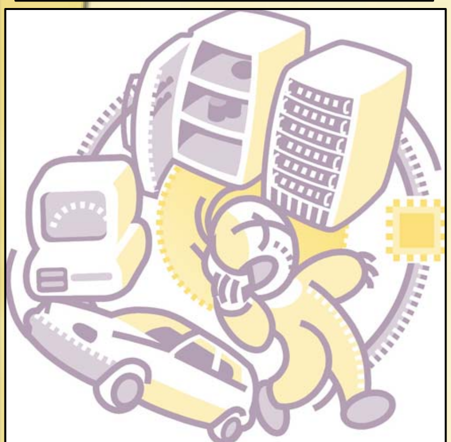






Ambient Intelligence





Ambient Intelligence refers to electronic environments that are sensitive and responsive to the presence of people. In an ambient intelligence world, devices support people in carrying out their everyday life activities, tasks and rituals in easy, natural way using information and intelligence that is hidden in the network connecting these devices. The ambient intelligence is characterized by systems and technologies that are:

- embedded: many networked devices are integrated into the environment
- context aware: these devices can recognize you and your situational context
- personalized: they can be tailored to your needs
- adaptive: they can change in response to you
- anticipatory: they can anticipate your desires without conscious mediation.

Wikipedia

Internet of Things vs Web of Things

The **Web of Things** is a vision inspired from the **Internet of Things** where everyday devices and objects (objects that contain embedded devices) are connected by fully integrating them to the Web. Unlike in the many systems that exist for the Internet of things, the Web of Things is about re-using the Web standards to connect the quickly expending eco-system of embedded devices built into everyday smart objects. Well-accepted and understood standards (such as URI, HTTP, REST, RSS, etc.) are used to access the functionality of the smart objects.
In the Internet of Things, the physical world becomes integrated with computer networks. Web of Things in addition allows real-world devices to be easily combined with other virtual and physical resources.

WoT=IoT+Web standards (HTTP, HTML,...)

1.5. Web 2.2

Web of Services

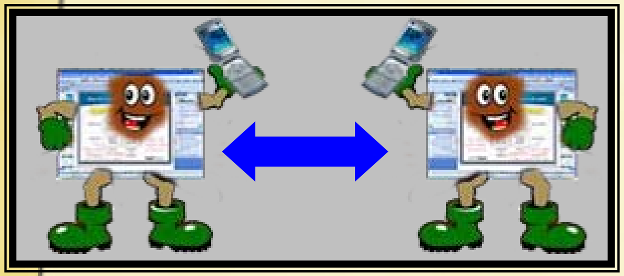
Web of Services

Software and Services

Facilitates Software-to-Software interaction

Web of Services

- Cloud Computing
- SOA
- SOAP, WSDL, UDDI
- SaaS

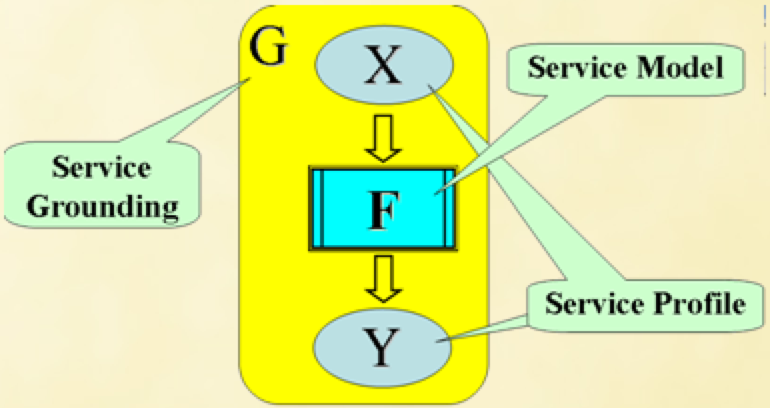


SaaS: Software-as-a-Service

SaaU: Software-as-a-User

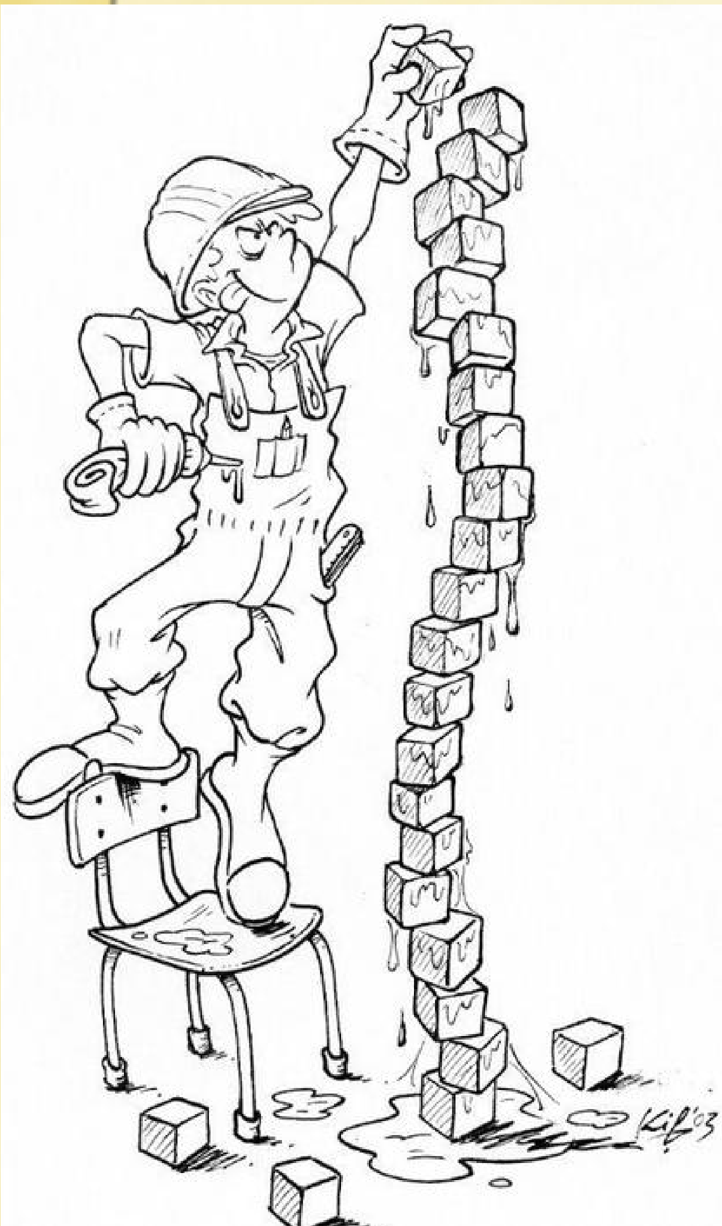
Web Services

A **web service** is defined by the W3C as "a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (Web Services Description Language **WSDL**). Other systems interact with the web service in a manner prescribed by its description using **SOAP** messages, typically conveyed using HTTP with an XML serialization in conjunction with other web-related standards." Web services are frequently just Internet API that can be accessed over a network, such as the Internet, and executed on a remote system hosting the requested services. **UDDI** is an open industry initiative enabling businesses to publish service listings and discover each other and define how the services or software applications interact over the Internet. [WIKIPEDIA]



- *What does the service require of the user or other applications and provides for them?* - **ServiceProfile**
- *How does it work?* - **ServiceModel**
- *How is it used?* - **ServiceGrounding**

What is Service-Oriented Architecture ?



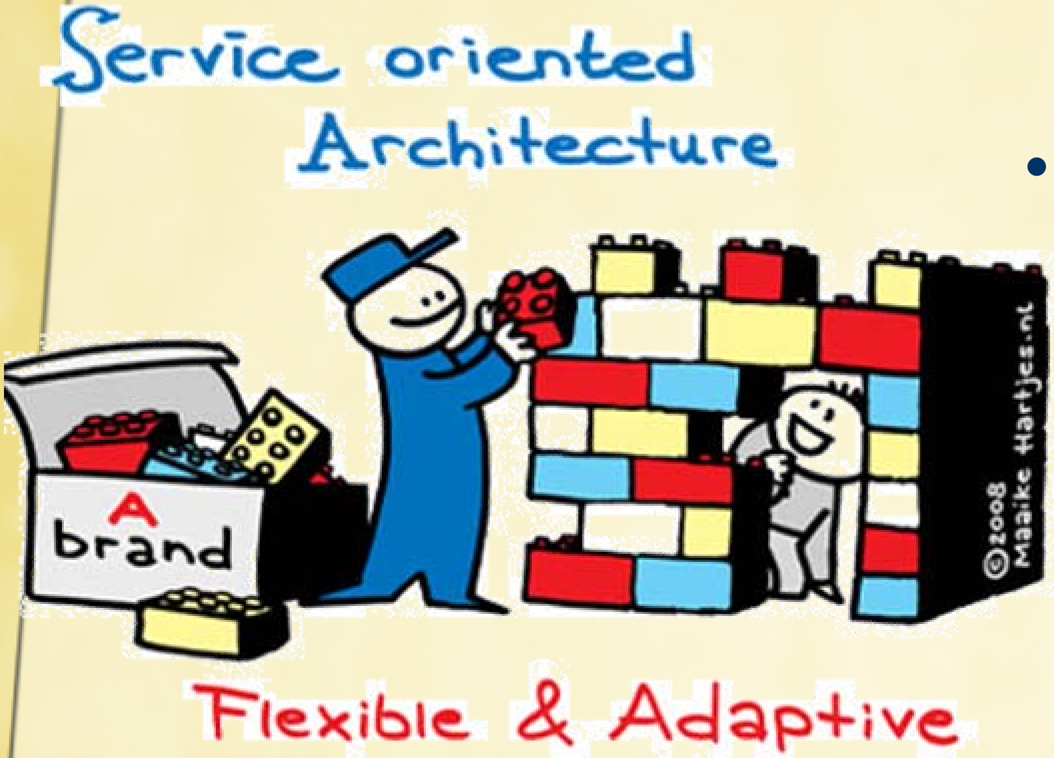
- **SOA** is the practice of sequestering the *core business functions* into independent services that don't change frequently. SOA is a tool for software (as a service) integration. Rather than defining an API, SOA defines the interface to remote Web-based services in terms of protocols and functionality.

Service Oriented Architecture (SOA) is a means of designing and building software. It is a manufacturing model.

Software as a Service (SaaS) is a means of receiving software through an external party to your business similar to telephone or power utilities. It is a sales and distribution model.

[J Natoli, Intel]

Why Service-Oriented Architecture ?



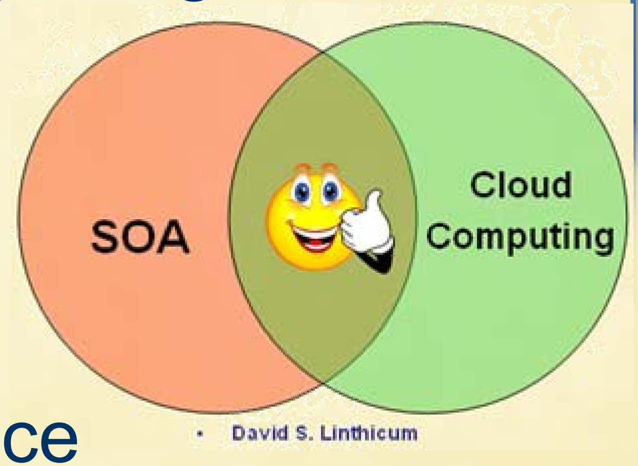
- **SOA has many advantages:**
  - ❑ Ability to couple or decouple functionality without impacting other parts of the system and architecture.
  - ❑ Processes can be orchestrated in a consistent and clear manner.

What is and why Cloud Computing?

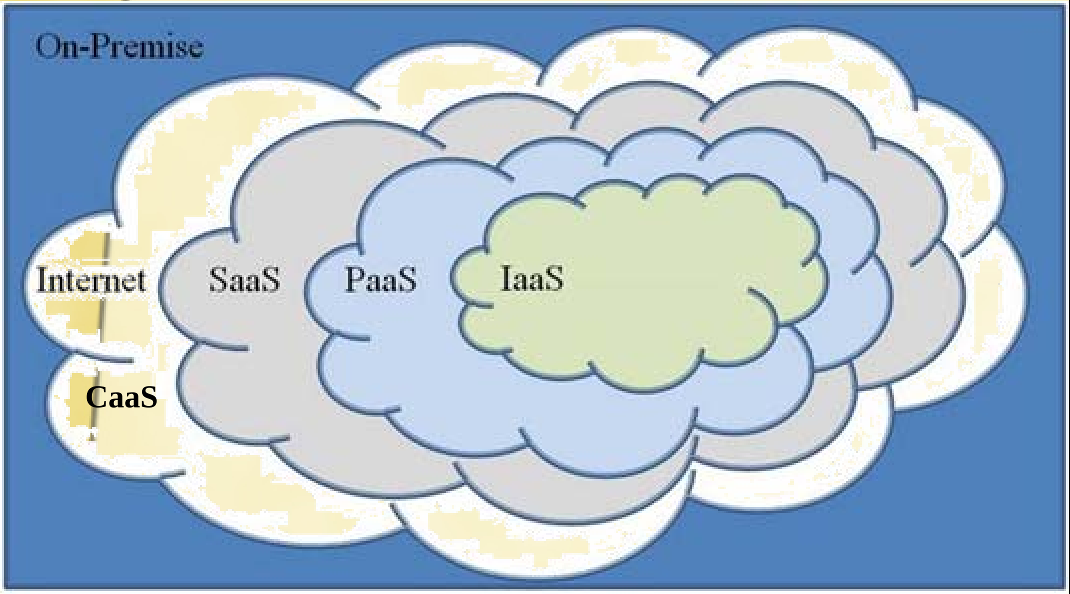
"A pool of abstracted, highly scalable, and managed computational infrastructure capable of hosting end-customer applications and billed by consumption" [Forrester Research]

Some segments of cloud computing:

- SaaS
  - Software as a Service
  - Storage as a Service
- PaaS – Platform as a Service
- IaaS – Infrastructure as a Service



Cloud Computing: off-premise computing

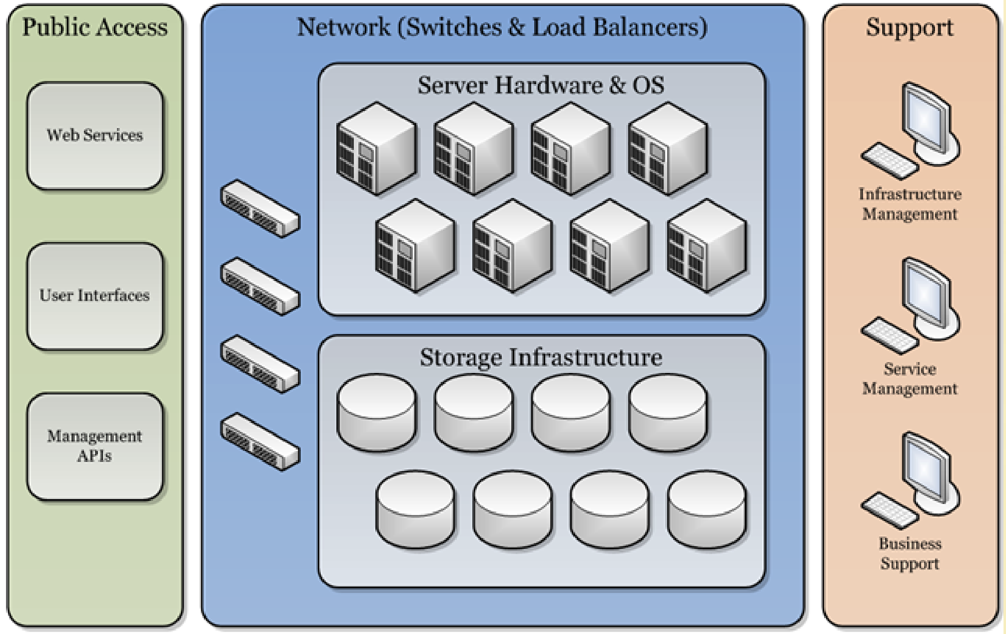


Different types of computing in the cloud

Source: Kavis Technology Consulting

"With SaaS solutions, the underlying infrastructure is hidden from you. With PaaS, you manage the amount of virtual server instances you use but you must use the technologies required by the provider. With IaaS, you manage the resources you use and are free to leverage whatever technologies you choose to deploy on."

Message to Cloud Computing Customers: "*You get the resources you need quicker, and pay less up front for them, compared with procuring your own servers and software*."



Infrastructure as a Service

Infrastructure as a Service (IaaS) is the delivery of computer infrastructure (typically a platform virtualization environment) as a service. This is an example of the ***everything as a service*** trend and shares many of the common characteristics. Rather than purchasing servers, software, data center space or network equipment, clients instead buy those resources as a fully outsourced service. The service is typically billed on a utility computing basis and amount of resources consumed (and therefore the cost) will typically reflect the level of activity. It is an evolution of web hosting and virtual private server offerings.

- **Resources delivered as a service** including servers, network equipment, memory, CPU, disk space, data center facilities;
- **Dynamic scaling** of infrastructure which scales up and down based on application resource needs;
- **Variable cost** service using fixed prices per resource component;
- **Multiple tenants** typically coexist on the same infrastructure resources;
- **Enterprise grade** infrastructure allows mid-size companies to benefit from the aggregate compute resource pools.

Wikipedia

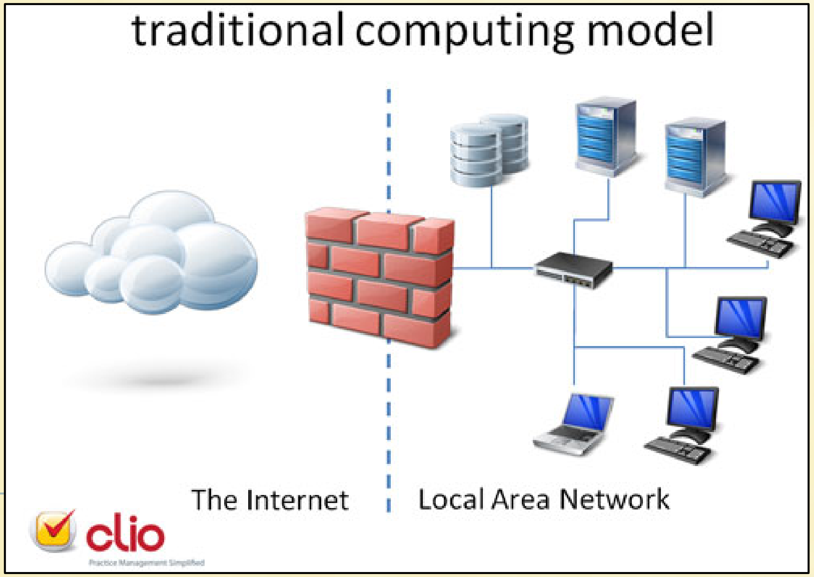
Platform as a Service

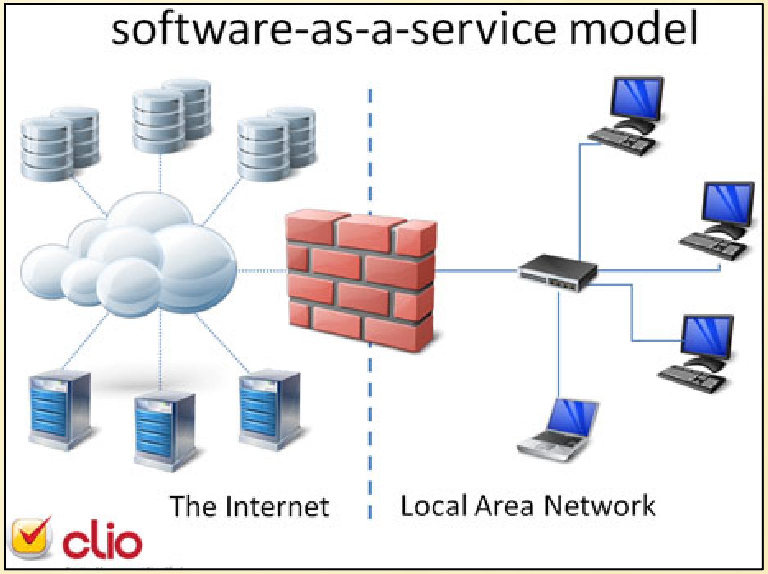
- **Platform-as-a-service (PaaS)** is the delivery of a computing platform and solution stack as a service. It facilitates deployment of applications without the cost and complexity of buying and managing the underlying hardware and software layers, providing all of the facilities required to support the complete life cycle of building and delivering web applications and services entirely available from the Internet with no software downloads or installation for developers, IT managers or end-users. It's also known as *cloudware* .
- Within the und-user context, PaaS can be defined as the concept to deliver a cost-effective cloud based workspace environment – the platform - to the end-user, which integrates work/life environment and facilitates him or/her to work, communicate, interact and play (games) anywhere, anytime, any device in a safe manner based on the roles assigned to the end-user. As such PaaS could also be described as Datacenter Centric Client Based Utility Computing. **Wikipedia**

With PaaS, corporate IT departments can focus on innovation instead of complex infrastructure. By leveraging the PaaS, organizations can redirect a significant portion of their budgets from "keeping the lights on" to creating applications that provide real business value.
This model is driving a new era of mass innovation. For the first time, developers around the world can access unlimited computing power. Now, anyone with an Internet connection can build powerful applications and easily deploy them to users wherever they're located.
SalesForce.com

Software as a Service

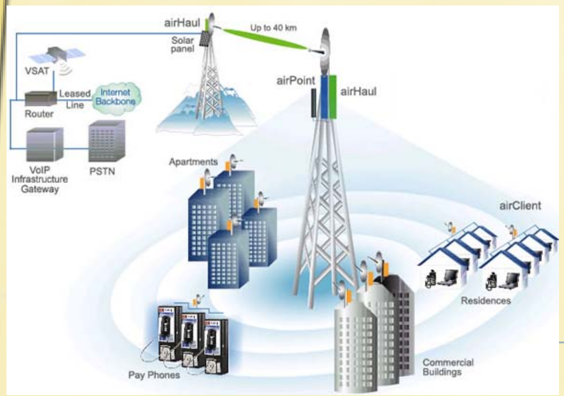
- **Software-as-a-service (SaaS)** is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet. It is a model of software deployment whereby a provider licenses an application to customers for use as a service on demand. SaaS software vendors may host the application on their own web servers or download the application to the consumer device, disabling it after use or after the on-demand contract expires. The on-demand function may be handled internally to share licenses within a firm or by a third-party application service provider sharing licenses between firms .
- According to SaaS, each software service can act as a service provider, exposing its functionality to other applications via public brokers, and can also act as a service requester, incorporating data and functionality from other services. **Wikipedia**





Communication as a Service

- **Communication-as-a-service (CaaS)** is a type of outsourced enterprise communications solution where a third party vendor (known as CaaS vendor) is responsible for the management of hardware and software required for delivering Voice over IP (also known as Voice as a Service), instant messaging, and video conferencing applications using fixed and mobile devices. A synonym for CaaS is Unified communications as SaaS. The CaaS model has evolved in the telecommunication industry in a similar manner to the SaaS model in the field of software delivery. **Wikipedia**



1.6. Web 3.0

Semantic Web / Web of Knowledge

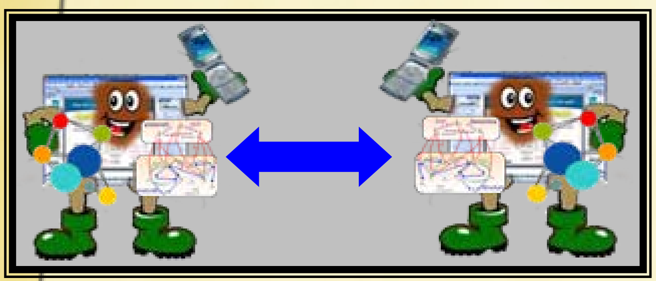
Web of Knowledge (Semantic Web, Web 3.0)

Knowledge and Data Collections

Facilitates Knowledge-to-Knowledge interaction

Web of Knowledge

- Ontologies
- Metadata
- Reasoning
- Semantic Technology
- Integration



KaaS: Knowledge-as-a-Service

KaaU: Knowledge-as-a-User

Web 3.0

Web 3.0 is an evolving extension of the World Wide Web in which web content can be expressed not only in natural language, bu t also in a form that can be understood, interpreted and used by software agents, thus permitting them to find, share and integrate information more easily.

According to Nova Spivack, the CEO of Radar Networks, one of the leading voices of this new-age Internet, "Web 3.0 is a set of standards that turns the Web into one big database."

Web 3.0 in terms of Semantic Web is the third generation of World Wide Web in which machines can read sites similar to human being and also follows your instructions. For example if you order to check your schedule against the schedules of all the dentists and doctors within a 10-mile radius if follows tour order and provide the appropriate information.

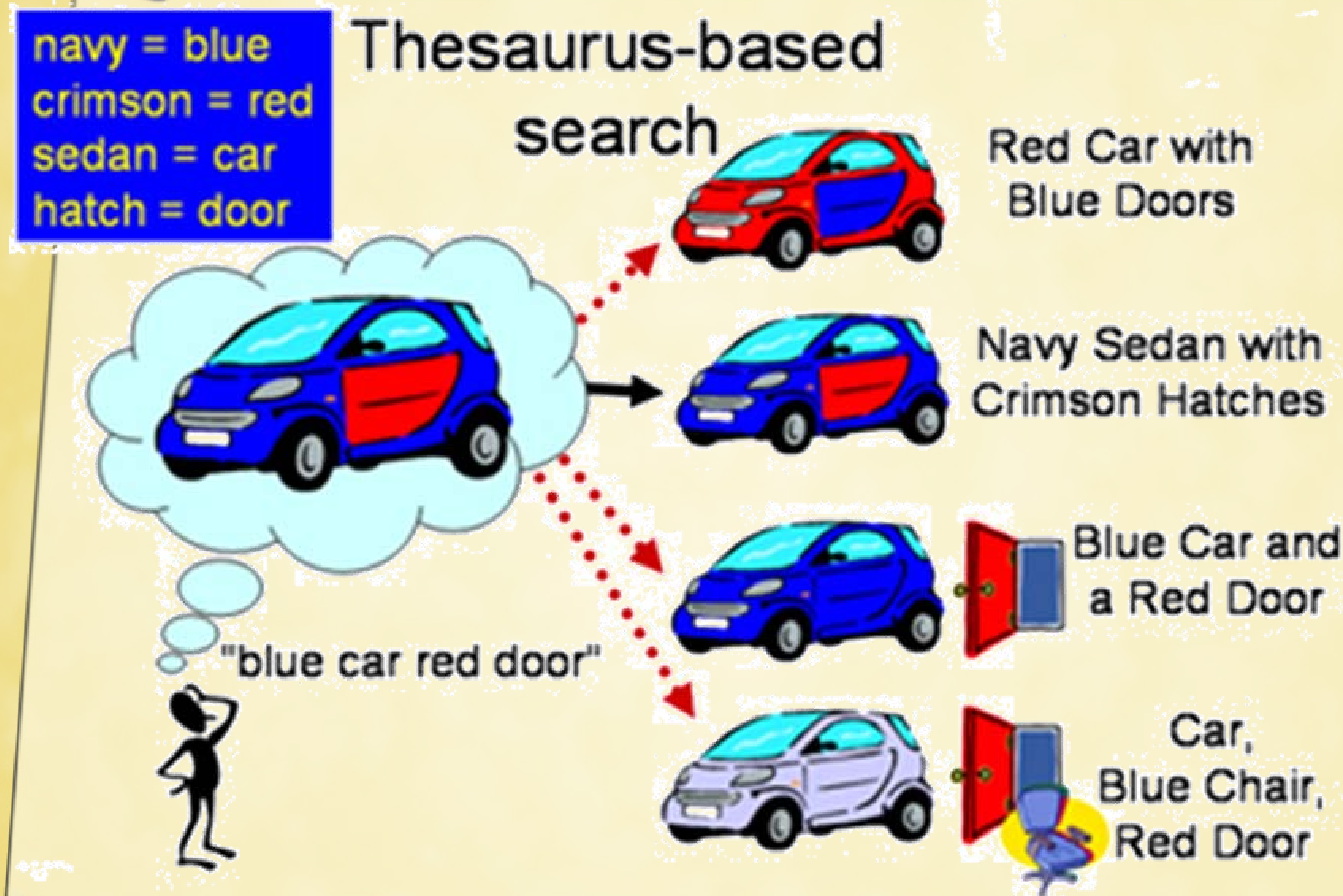
Web 3.0 has also been associated to a possible hub of SOA (Service Oriented Architecture) and Semantic web.

From: http://www.javajazzup.com/issue3/JavaJazzUp.pdf

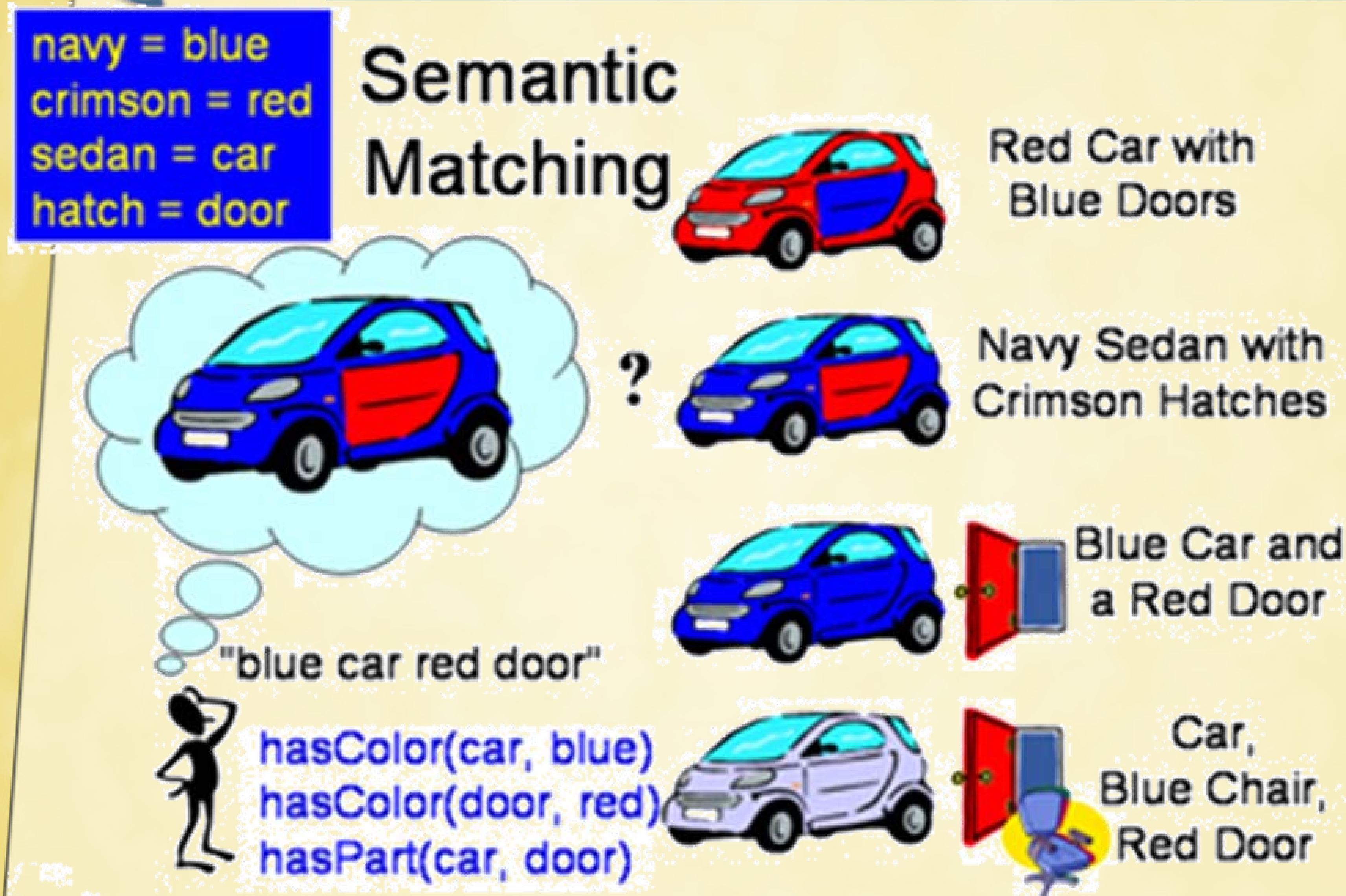
Semantic Web Motivation (1)

Simple word-matching

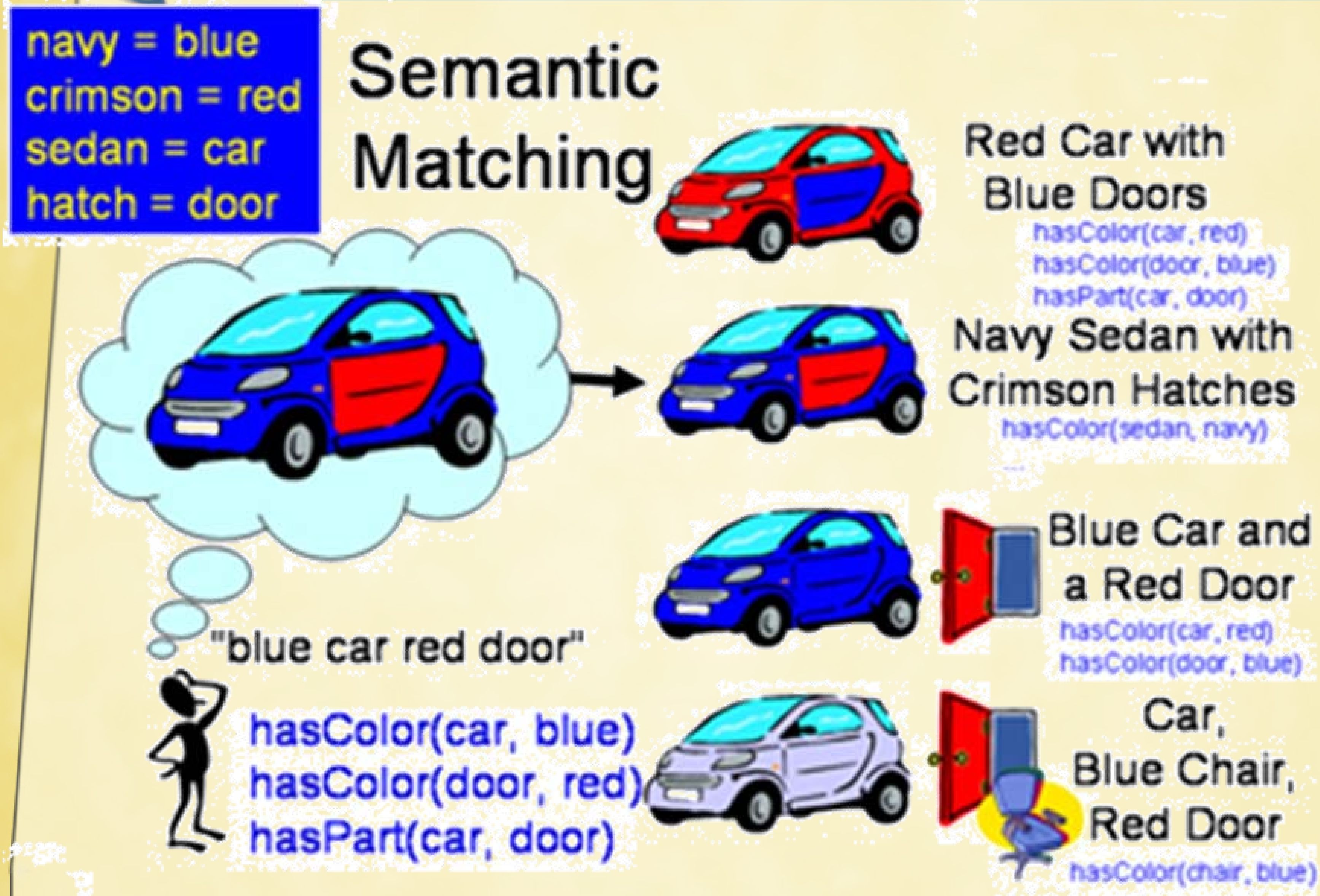
Semantic Web Motivation (2)



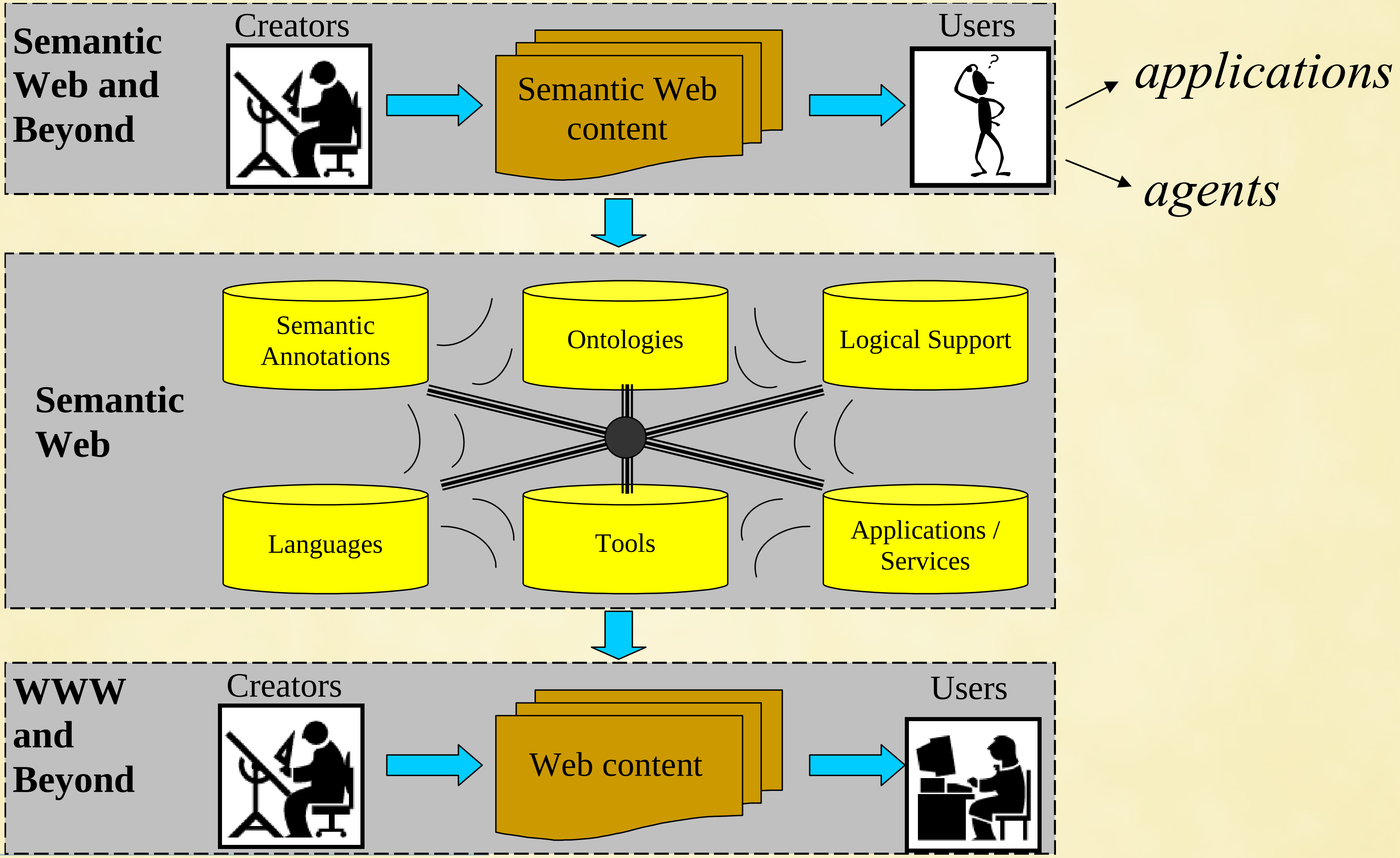
Semantic Web Motivation (3)



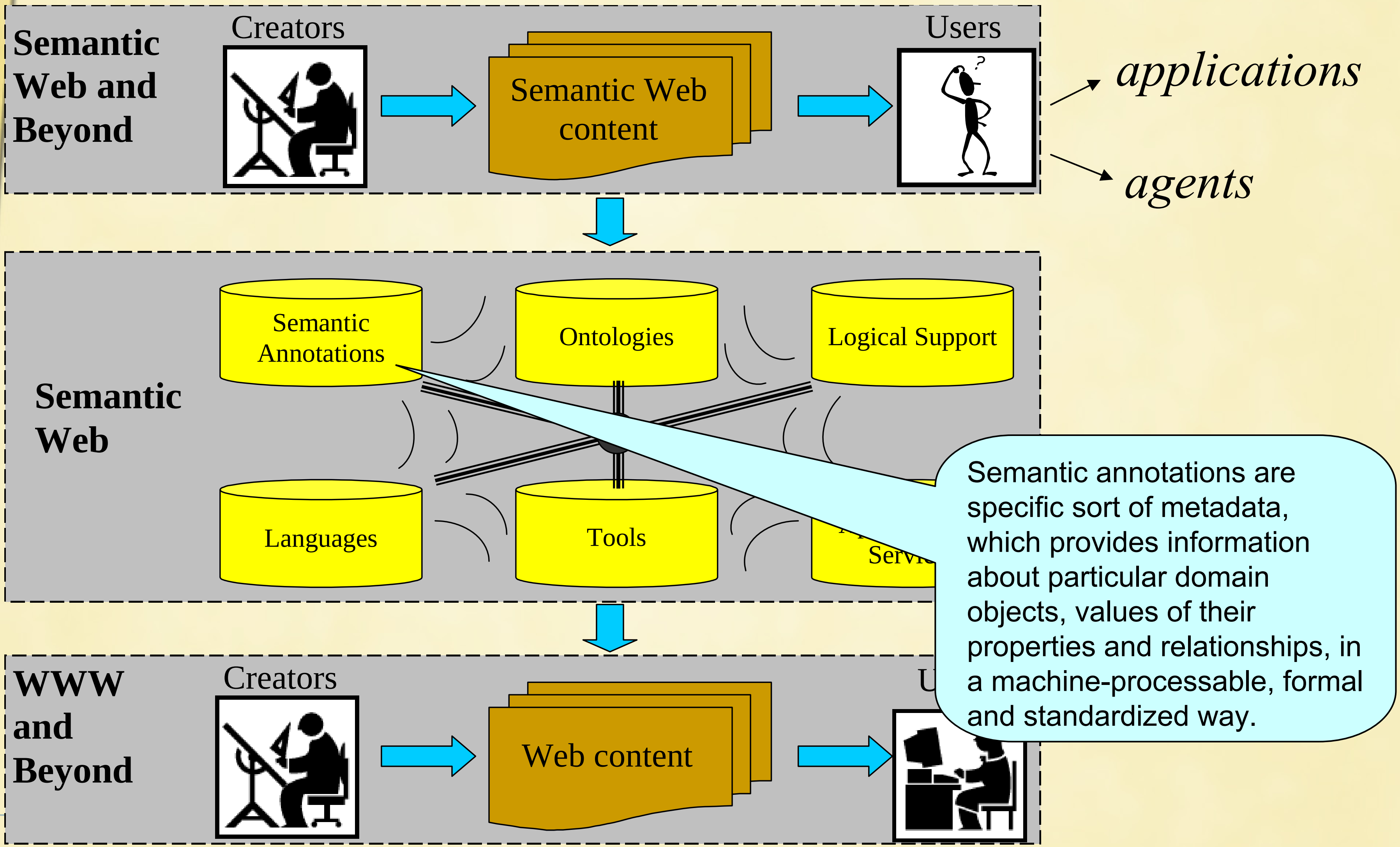
Semantic Web Motivation (4)



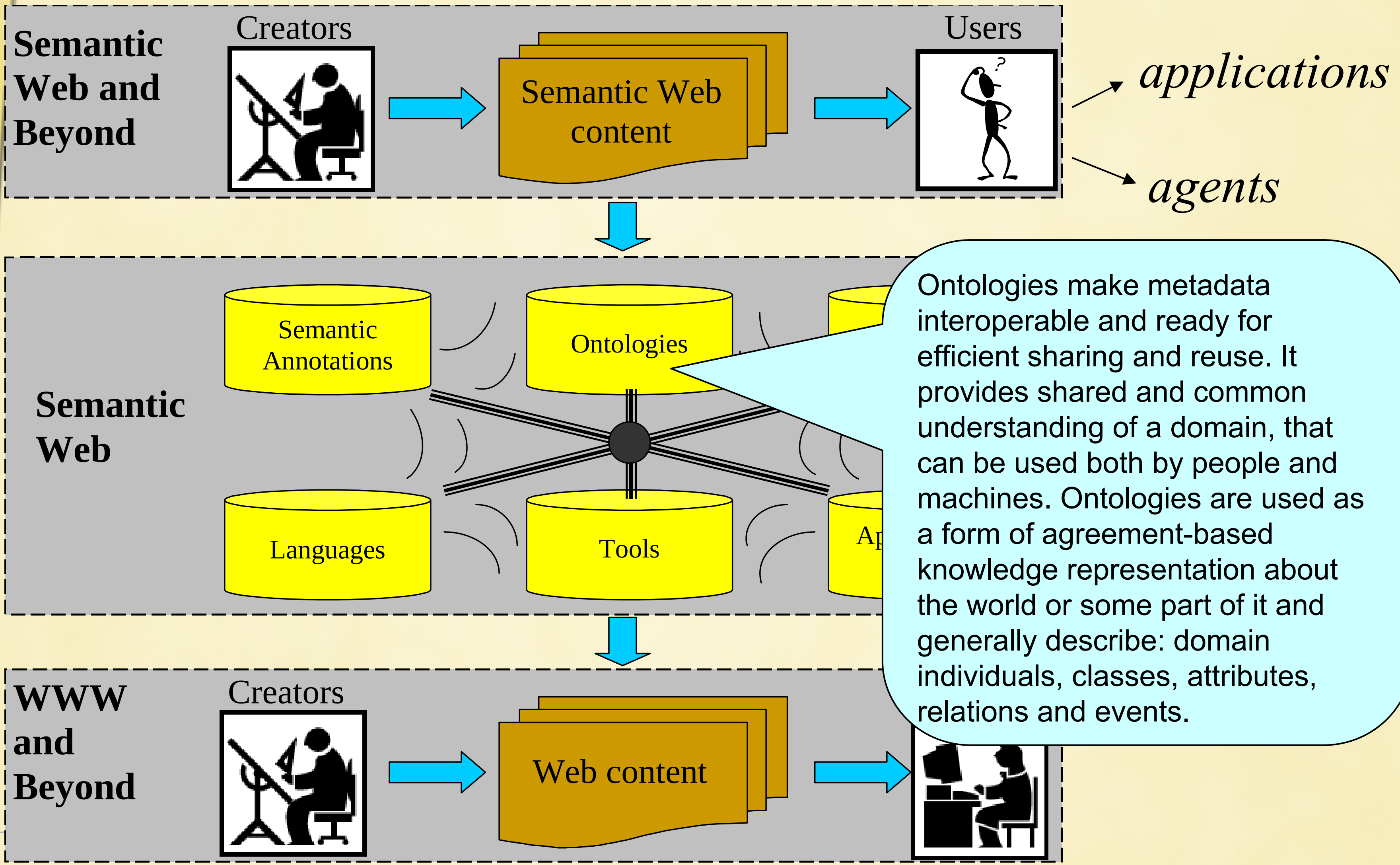
Semantic Web: New "Users"



Semantic Web: Annotations

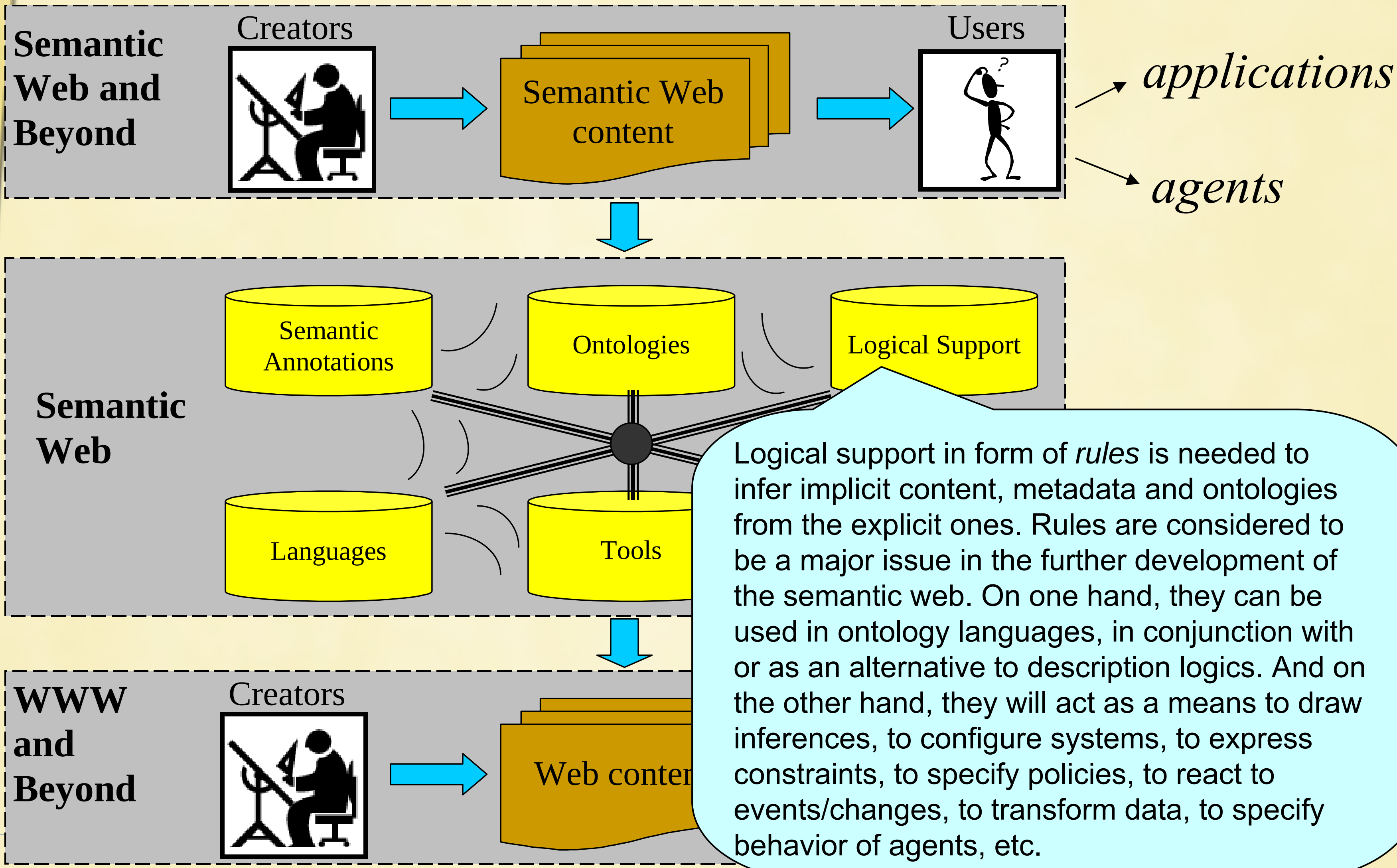


Semantic Web: Ontologies



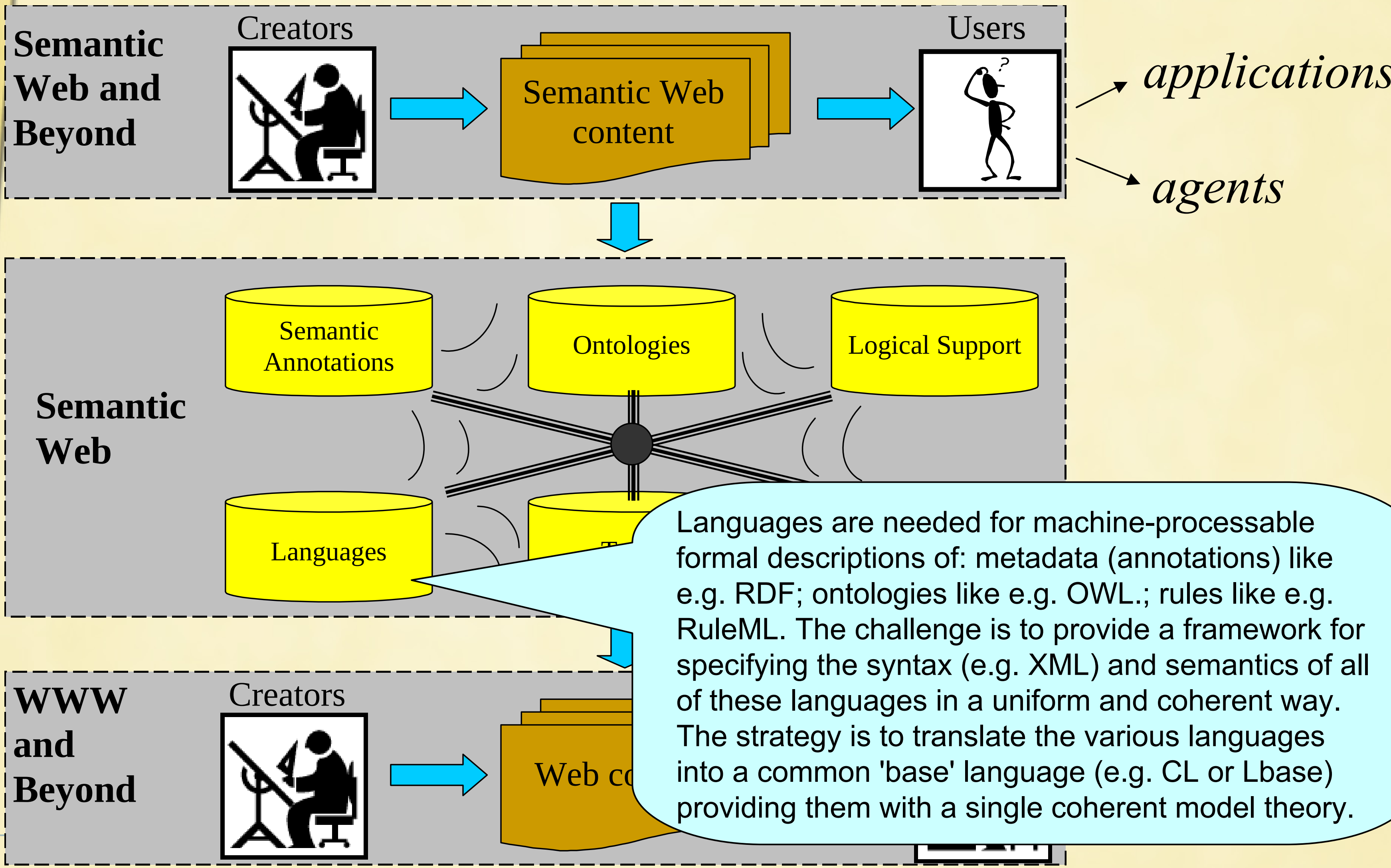
Ontologies make metadata interoperable and ready for efficient sharing and reuse. It provides shared and common understanding of a domain, that can be used both by people and machines. Ontologies are used as a form of agreement-based knowledge representation about the world or some part of it and generally describe: domain individuals, classes, attributes, relations and events.

Semantic Web: Rules



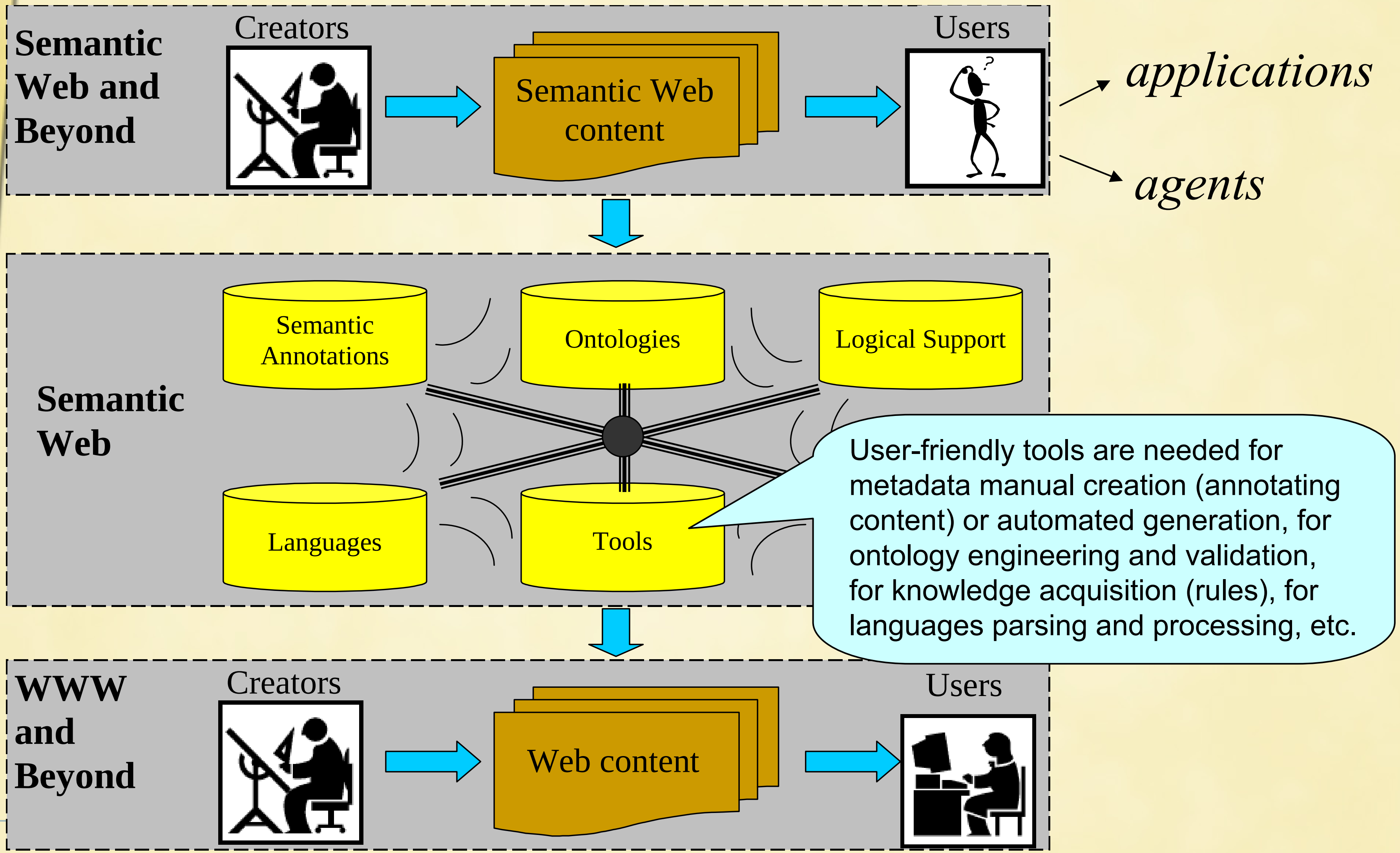
Logical support in form of *rules* is needed to infer implicit content, metadata and ontologies from the explicit ones. Rules are considered to be a major issue in the further development of the semantic web. On one hand, they can be used in ontology languages, in conjunction with or as an alternative to description logics. And on the other hand, they will act as a means to draw inferences, to configure systems, to express constraints, to specify policies, to react to events/changes, to transform data, to specify behavior of agents, etc.

Semantic Web: Languages

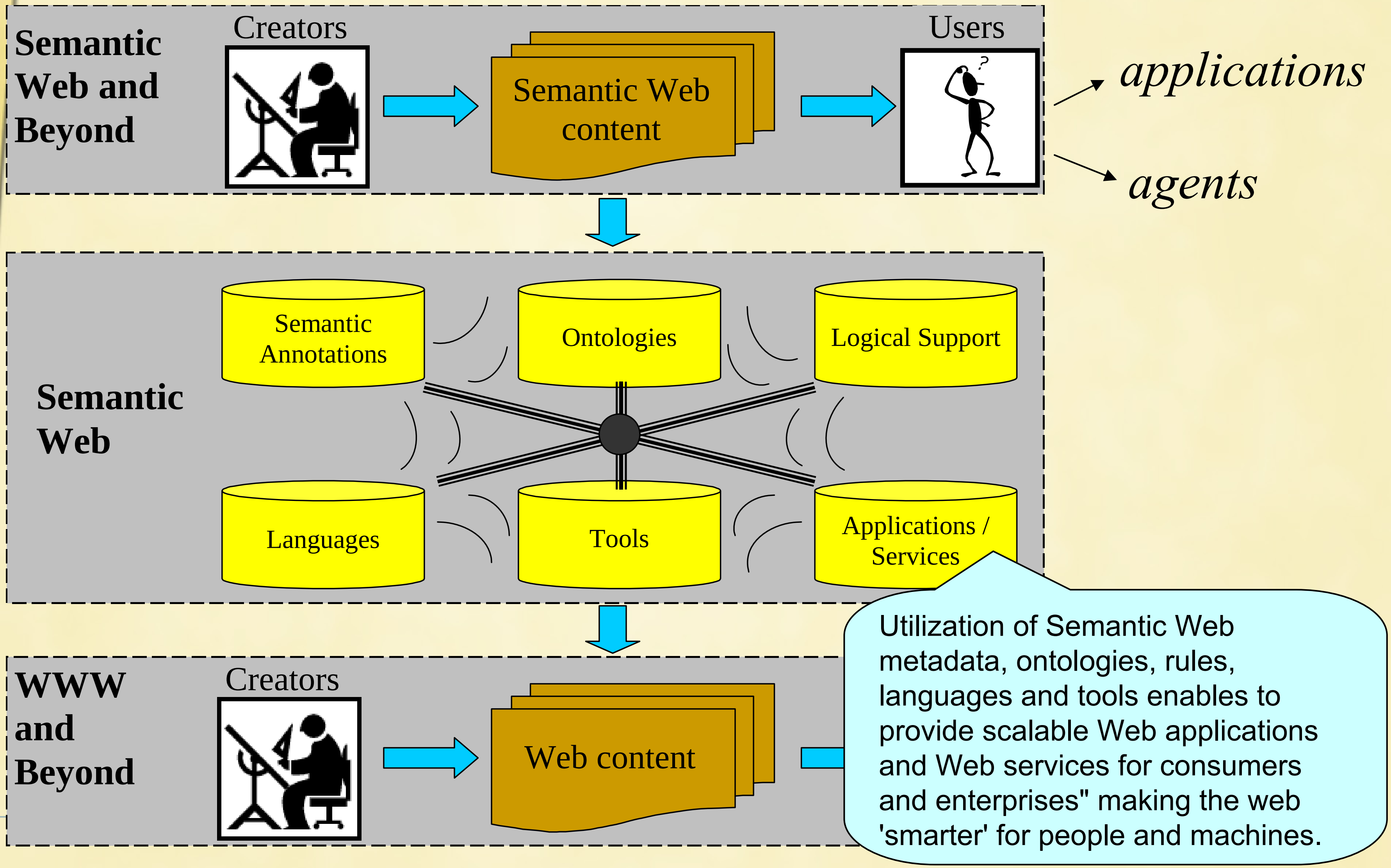


Languages are needed for machine-processable formal descriptions of: metadata (annotations) like e.g. RDF; ontologies like e.g. OWL.; rules like e.g. RuleML. The challenge is to provide a framework for specifying the syntax (e.g. XML) and semantics of all of these languages in a uniform and coherent way. The strategy is to translate the various languages into a common 'base' language (e.g. CL or Lbase) providing them with a single coherent model theory.

Semantic Web: Tools



Semantic Web: Applications and Services



Vision 2006: "Real Semantic Web"

- *Semantic data generation vs. reuse* (the ability to operate with the semantic data that already exist, i.e. to exploit available semantic markup);
- *Single-ontology vs. multi-ontology systems* (the ability to operate with huge amounts of heterogeneous data, which could be defined in terms of many different ontologies and may need to be combined to answer specific queries);
- *Openness with respect to semantic resources* (the ability to make use of additional, heterogeneous semantic data, at the request of their user);
- *Scale as important as data quality* (the ability to explore, integrate, reason and exploit large amounts of heterogeneous semantic data, generated from a variety of distributed Web sources);
- *Openness with respect to Web (non-semantic) resources* (the ability to take into account the high degree of change of the conventional Web and provide data acquisition facilities for the extraction of data from arbitrary Web sources);
- *Compliance with the Web 2.0 paradigm* (the ability to enable *Collective Intelligence* based on massively distributed information publishing and annotation initiatives by providing mechanisms for users to add and annotate data, allowing distributed semantic annotations and deeper integration of ontologies;
- . *Open to services* (the ability applications integrate Web-service technology in applications architecture).

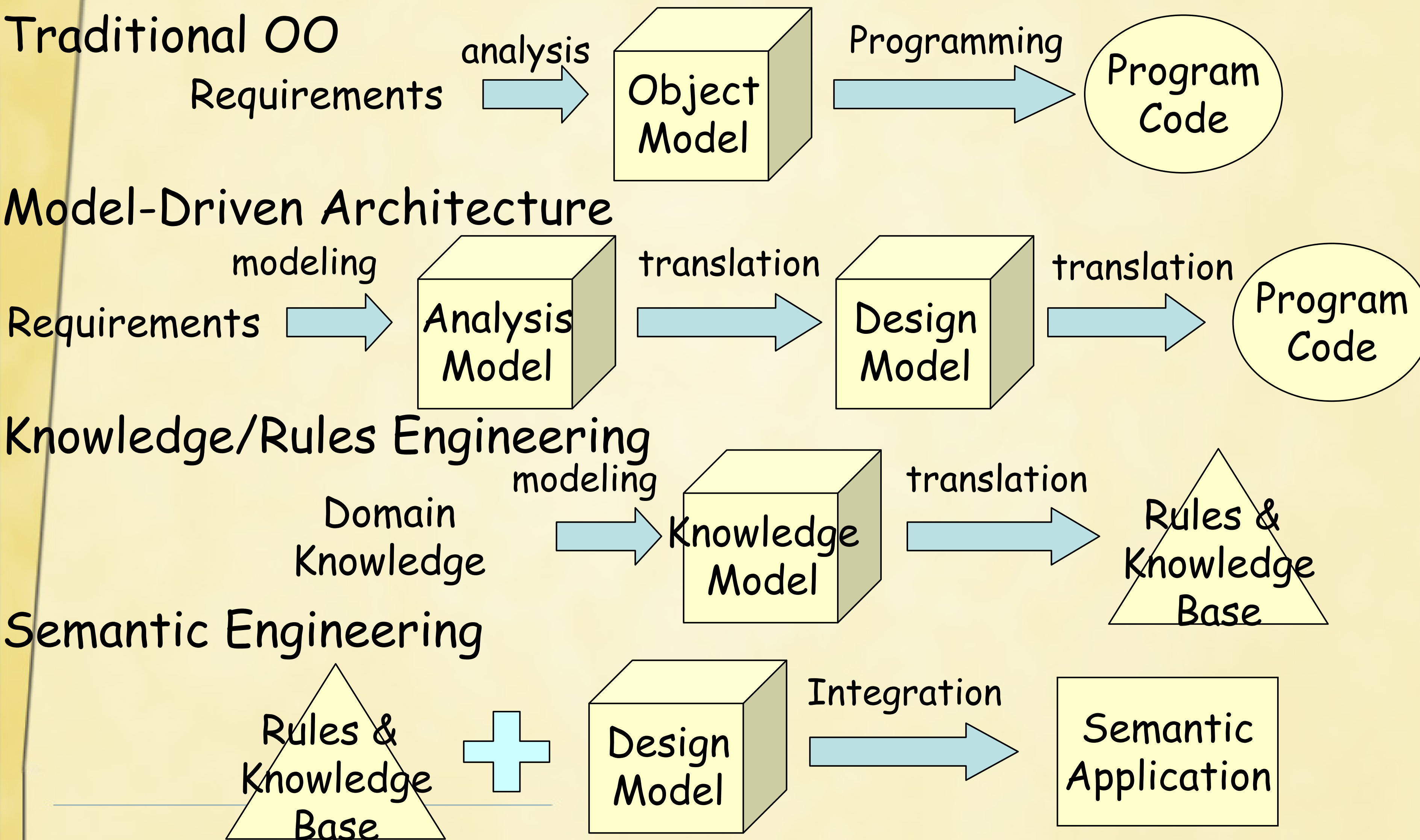
Motta and Sabou, 2006

Semantic Technology

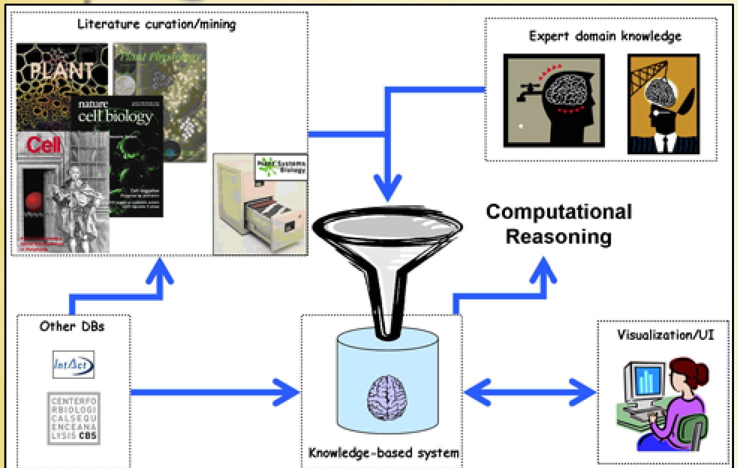
Semantic technologies are digital tools that represent meanings and knowledge (e.g., knowledge of something, knowledge about something, and knowledge how to do something, etc.) separately from content or behavior artifacts such as documents, data files, and program code. This knowledge is encoded in a digital form that both people and machines can access and interpret.

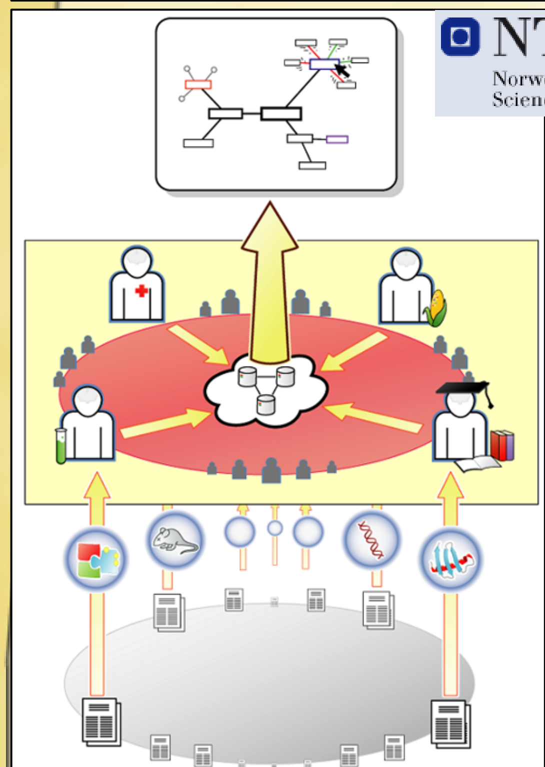
Semantic technology as a **software technology** allows the meaning of information to be known and processed at execution time. For a semantic technology there must be a **knowledge model** of some part of the world that **is used by one or more applications** at execution time.

Contrasting Semantic and Other Technologies

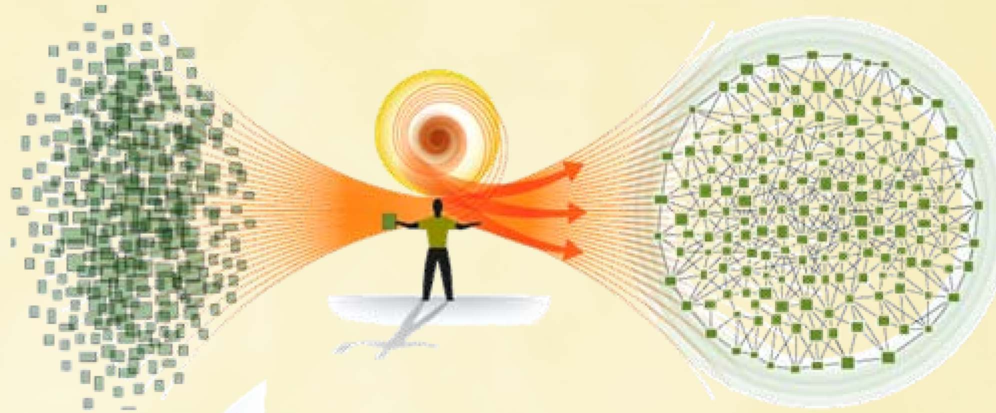


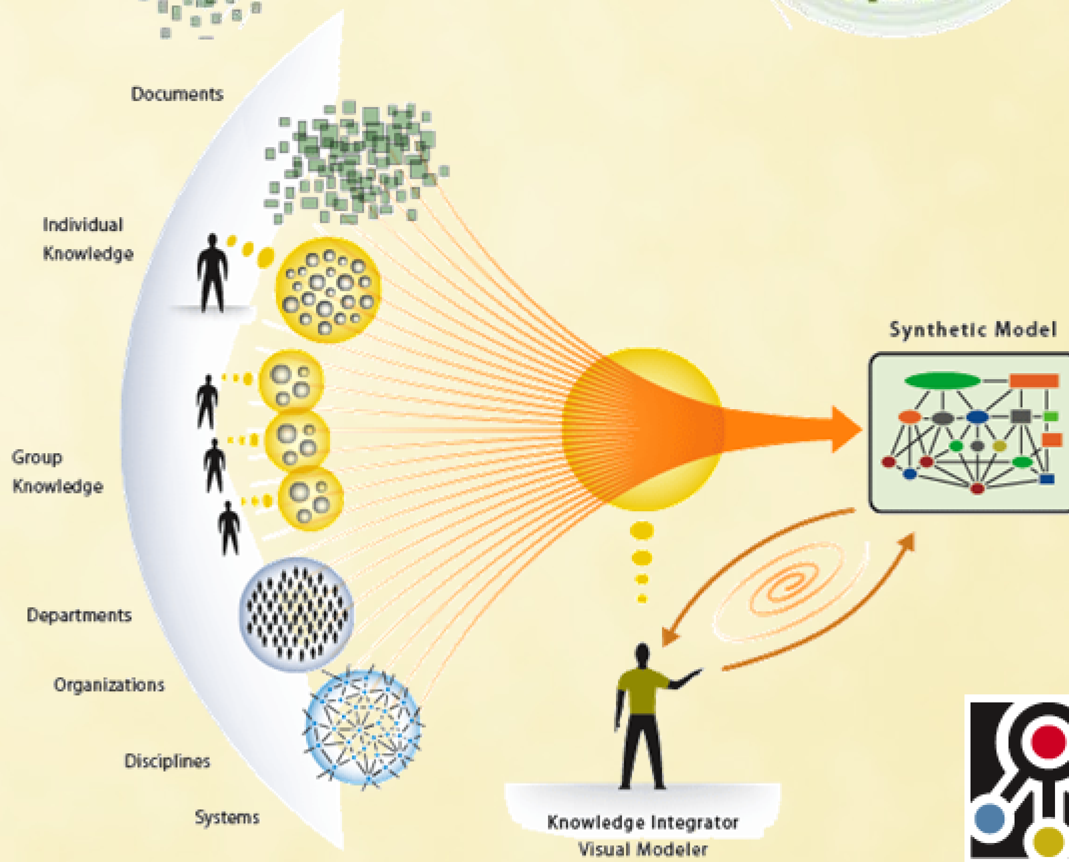
Knowledge Integration





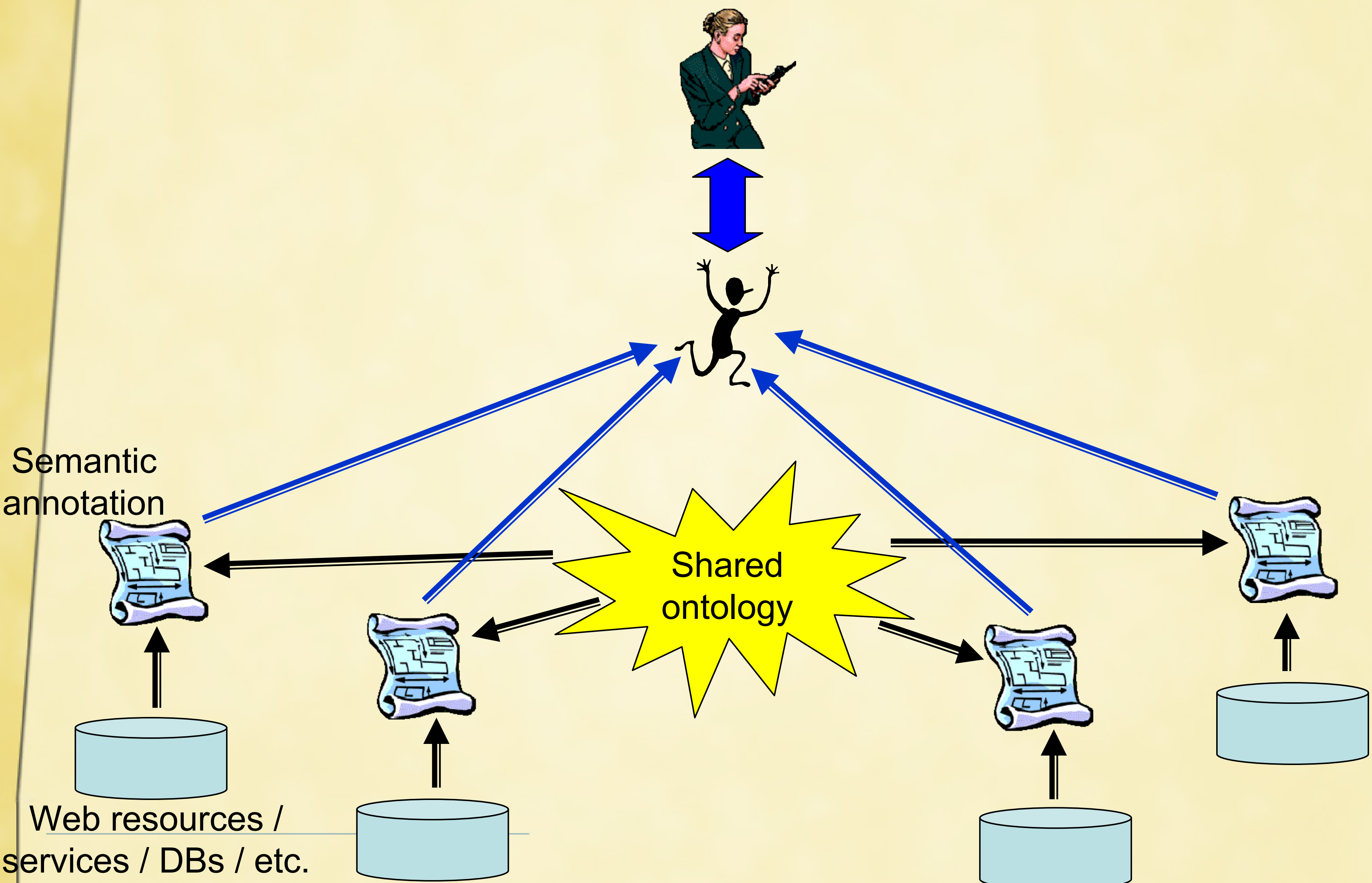
NTNU
Norwegian University of Science and Technology





Idiagram

Semantic Web: Resource Integration



Semantics in Social Computing (Mills Davis)

Semantic instant messaging — Use semantic technology for online messages, chat, and conference to understand conversations; keep track of people, topics & history; search by concept; act on messages.

Semantic email — Use semantic technology to understand messages. Models & tags people, profiles, threads, contents, and addresses; Searches semantically. Links messages to other information. Performs actions according to a semantic model.

Semantic blog — Enhance web journal with machine interpretable annotations and models & personal ontologies to harvest, link, and search information of interest by concepts and relationships.

Semantic desktop and webtop — Use natural language understanding, ontologies, data space concepts, and semantic processing to manage every piece of information a person encounters.

Semantic bookmarking & tag clouds — Associate links to web resources with concepts represented in an external ontology. Use semantic auto-tagging to Map folksonomy + semantic relationships between tags, users, and site resources.

Semantic social networks — Web of people, content, sites, and profiles that machines help build, interrelate, communicate with, and enjoy.

Semantic Collaboration — Collaboration tools enable groups to read, write, edit, and present information, coordinate their activities, share information and manage knowledge together. Semantic collaboration adds a layer of knowledge representation and meanings that enrich the collaborative experience and utility of its results.

Semantic wikis — Read-write web site that includes an underlying model of the knowledge described in its pages. Features include concept- rather than language-based searching; richly structured content navigation (multiple views, perspectives, levels of abstraction); context-specific visualization and presentation; mining of relationships; linking with external repositories, feeds, and systems.

Semantic Mash-Ups

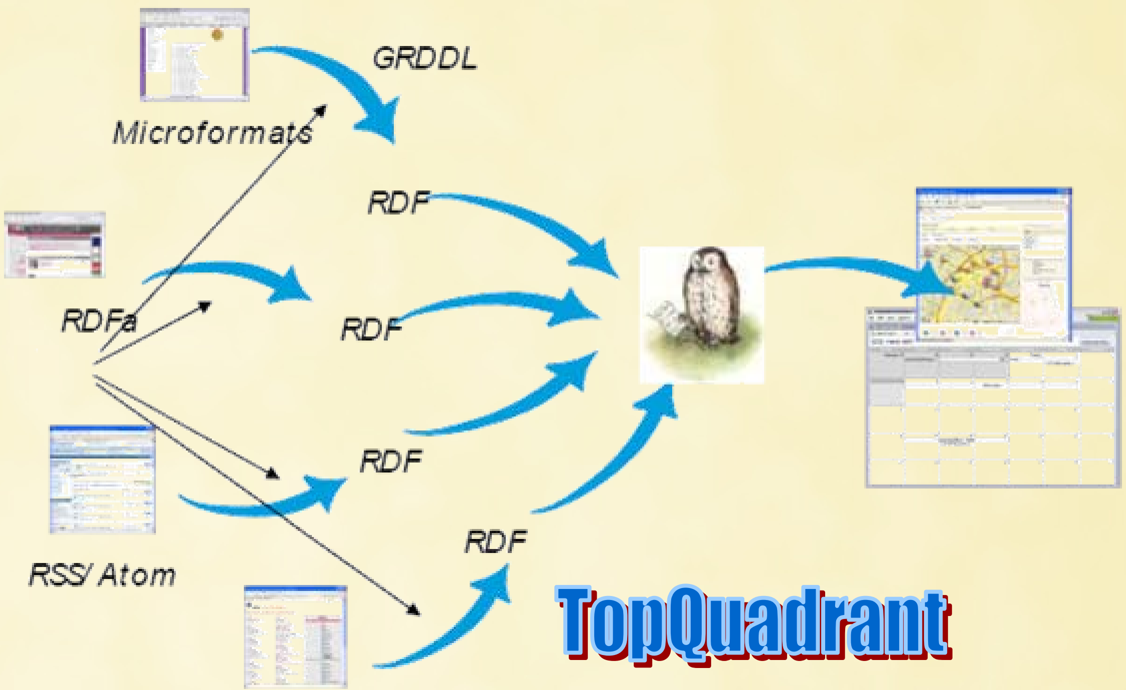
NASA about SMash-Up

Enable efficient expertise location by:

- integrating already existing but disparate data sources,
- providing a dynamic UI for exploring the information integration,
- visualizing social networks to facilitate communication,
- supporting incremental integration.

Integrate information from disjoint data sources, ad hoc'ly, to solve customer needs.

- Without upsetting delicate info-ecologies (data owners, curators, extant policies & procedures).
- Without requiring major investment in time or $$.

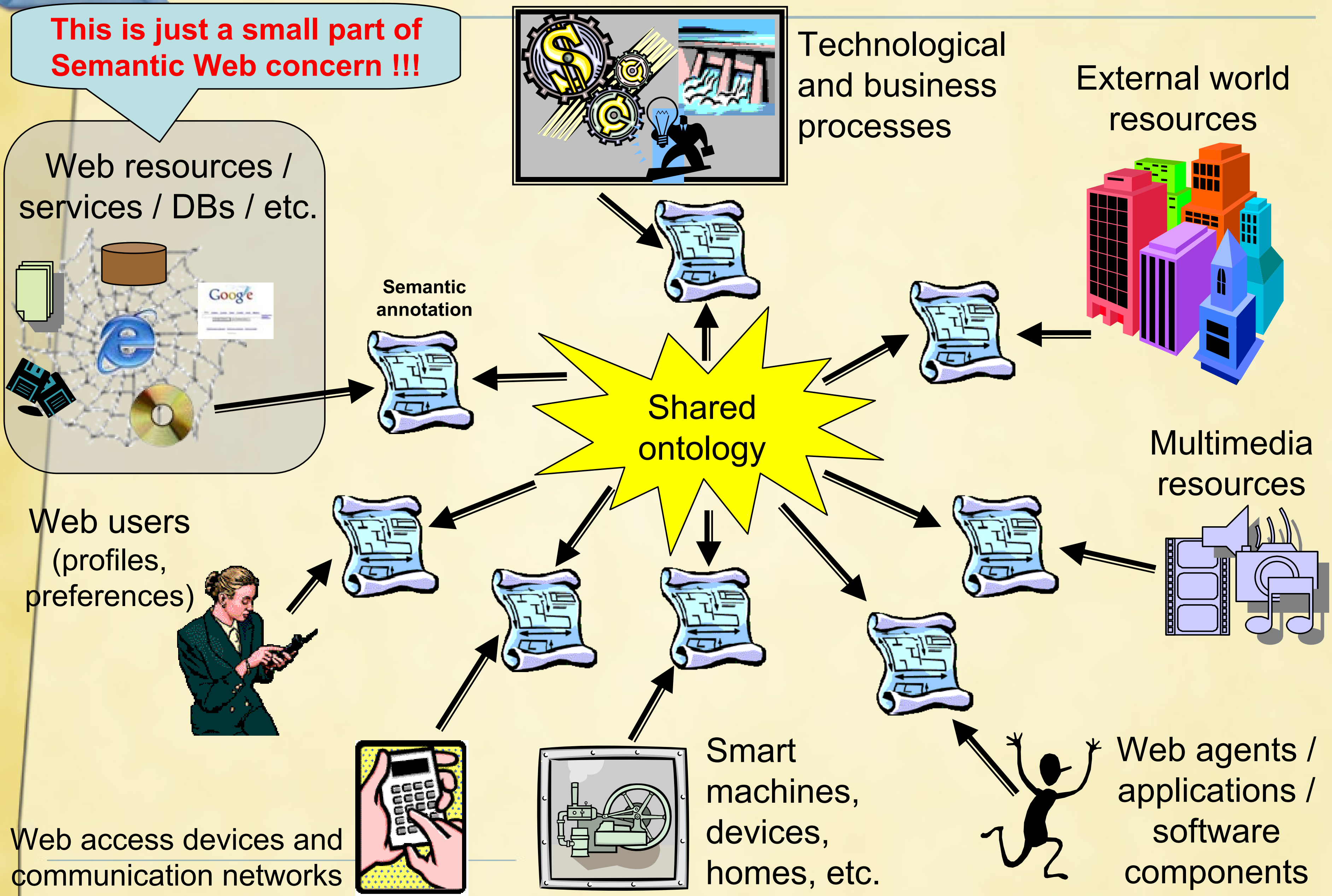


TopQuadrant

"In the idea of a semantic mash-up, the mash-up program is a model-driven architecture. This puts the structure of the mash-up under model control, rather than program control. It is still necessary to translate each information source into a semantic structure (i.e., RDF), but once that has been done, the structure of the mash-up is specified by a model, rather than by program code" [TopQuadrant Inc, June 2007].

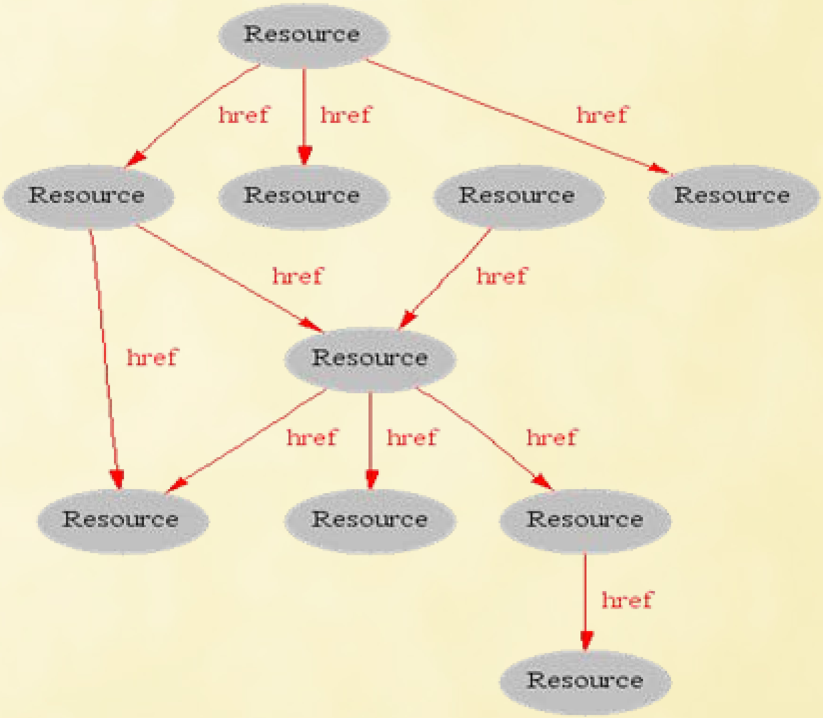
http://jazoon.com/jazoon07/en/conference/presentationdetails.html?type=sid&detail=870

Semantic Web: which resources to annotate ?



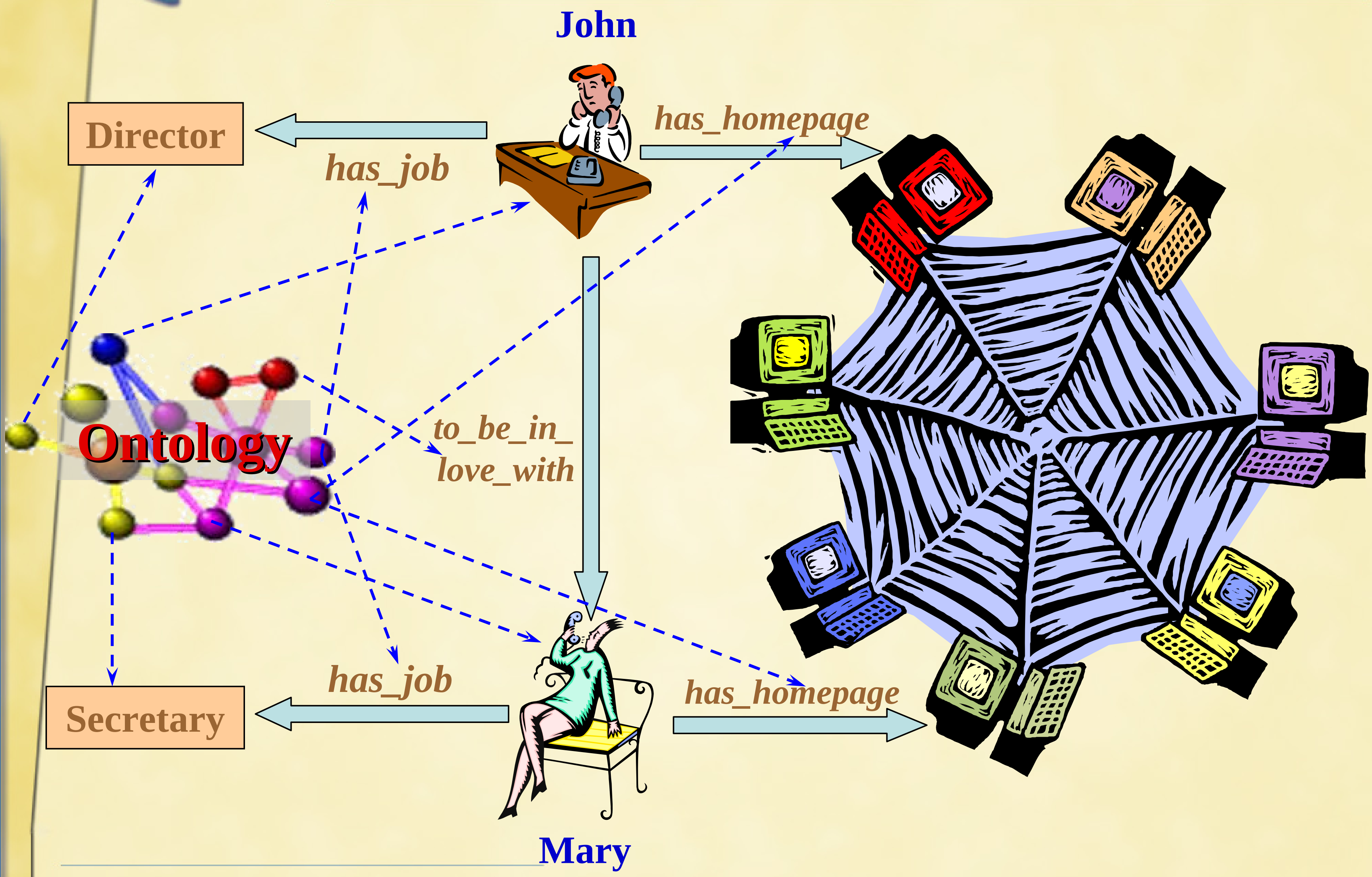
Semantic Web basics

Where we are Today: the Syntactic Web



[Hendler & Miller 02]

RDF – Semantic Network over Web Resources



Resources

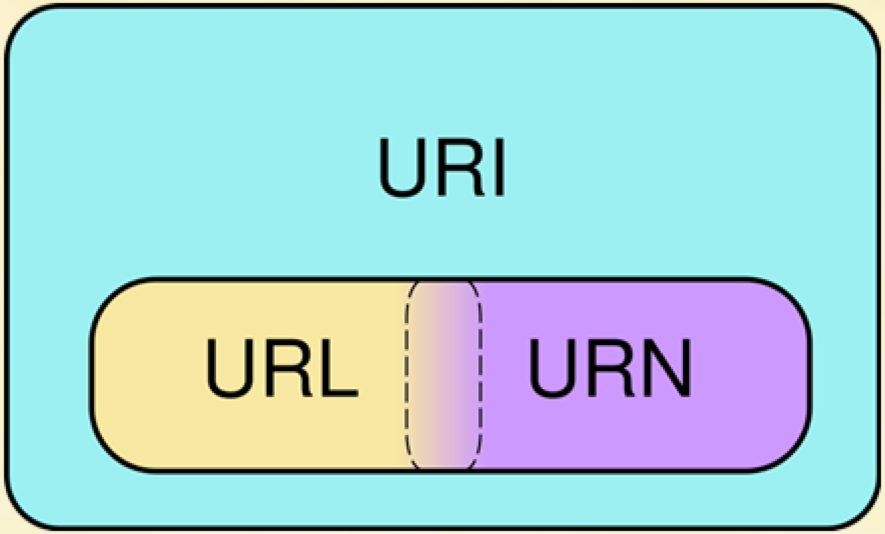
- All things being described by RDF expressions are called *resources*:
  - entire Web page;
  - a specific XML element;
  - whole collection of pages;
  - an object that is not directly accessible via the Web.

Resources and URIs

- A resource can be anything that has identity
- *Uniform Resource Identifiers* (URI)* provide a simple and extensible means for identifying a resource
- Not all resources are network "retrievable"; e.g., human beings, corporations, and books in a library can also be considered resources

* The term "Uniform Resource Locator" (URL) refers to the subset of URI that identify resources via a representation of their primary access mechanism (e.g., their network "location"), rather than identifying the resource by name or by some other attribute(s) of that resource.

URI: Uniform Resource Identifier



Venn diagram of Uniform Resource Identifier (URI) scheme categories. Schemes in the URL (**locator**) and URN (**name**) categories both function as resource IDs, so URL and URN are subsets of URI. They are also, generally, disjoint sets. However, many schemes can't be categorized as strictly one or the other, because all URIs can be treated as names, and some schemes embody aspects of both categories – or neither.

RDF Statement

- ***Subject*** of an RDF statement is a resource
- ***Predicate*** of an RDF statement is a property of a resource
- ***Object*** of an RDF statement is the value of a property of a resource

Example of RDF Statement

Ora Lassila is the creator of the resource http://www.w3.org/Home/Lassila.

Subject (resource)	http://www.w3.org/Home/Lassila
Predicate (property)	Creator
Object (literal)	“Ora Lassila”

RDF Example (subject of statement)

Ora Lassila is the creator of the resource http://www.w3.org/Home/Lassila.

Subject

```
<rdf:RDF>
  <rdf:Description about=
       "http://www.w3.org/Home/Lassila">
    <s:Creator>Ora Lassila</s:Creator>
  </rdf:Description>
</rdf:RDF>
```

RDF Example (predicate of statement)

Ora Lassila is the creator of the resource http://www.w3.org/Home/Lassila.

```
<rdf:RDF>
  <rdf:Description about=
       "http://www.w3.org/Home/Lassila">
    <s:Creator>Ora Lassila</s:Creator>
  </rdf:Description>
</rdf:RDF>
```

Predicate

RDF Example (object of statement)

Ora Lassila is the creator of the resource http://www.w3.org/Home/Lassila.

```
<rdf:RDF>
  <rdf:Description about=
       "http://www.w3.org/Home/Lassila">
    <s:Creator>Ora Lassila</s:Creator>
  </rdf:Description>
</rdf:RDF>
```

Object

RDF Example (reference to ontology)

Ora Lassila is the creator of the resource http://www.w3.org/Home/Lassila.

```
<rdf:RDF>
  <rdf:Description about=
       "http://www.w3.org/Home/Lassila">
    <s:Creator>Ora Lassila</s:Creator>
  </rdf:Description>
</rdf:RDF>
```

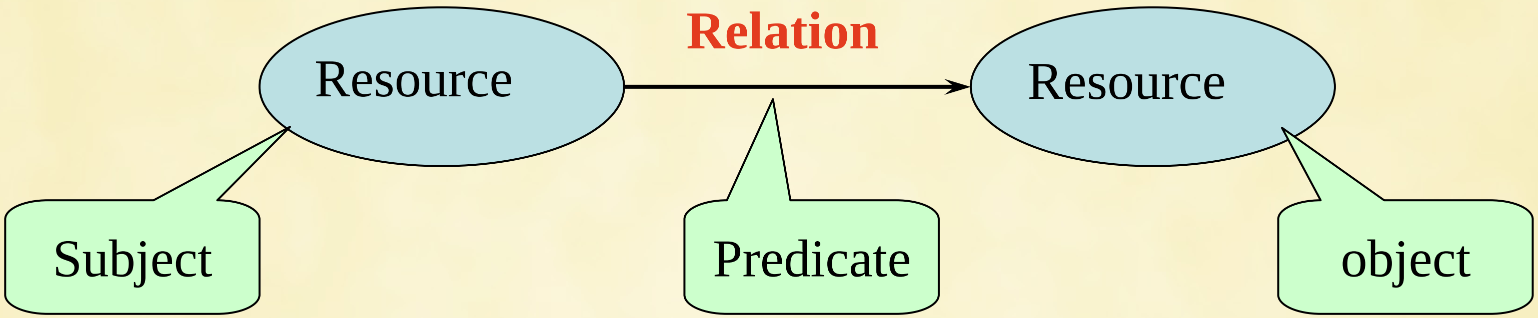
a specific namespace prefix as reference to ontology where predicates are defined, e.g. xmlns: s="http://description.org/schema/"

Full XML Document for the Example

Namespaces as attributes of "RDF" element in XML

```
<?xml version="1.0"?>
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:s="http://description.org/schema/">
  <rdf:Description about=
       "http://www.w3.org/Home/Lassila">
    <s:Creator>Ora Lassila</s:Creator>
  </rdf:Description>
</rdf:RDF>
```

Semantic Relation as RDF statement



http://www.cs.jyu.fi/ai/vagan/index.html

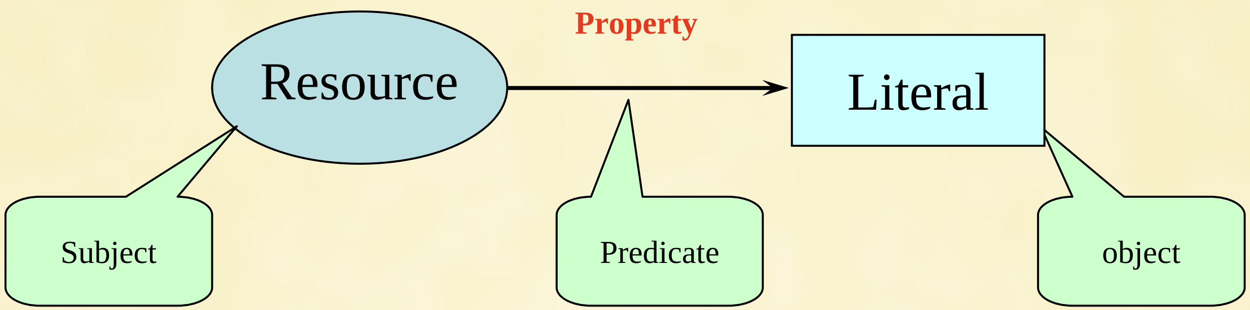
http://www.jyu.fi/agora-center/indexEng.html

employed_by

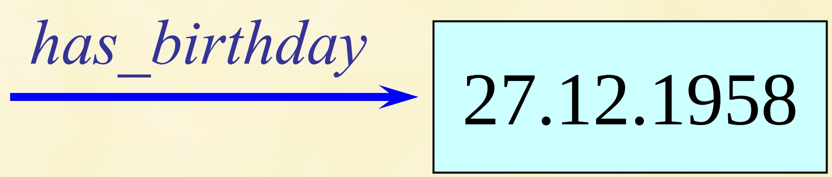
Personal web page of Terziyan V.

Web page of Agora Center

Semantic Property as RDF statement



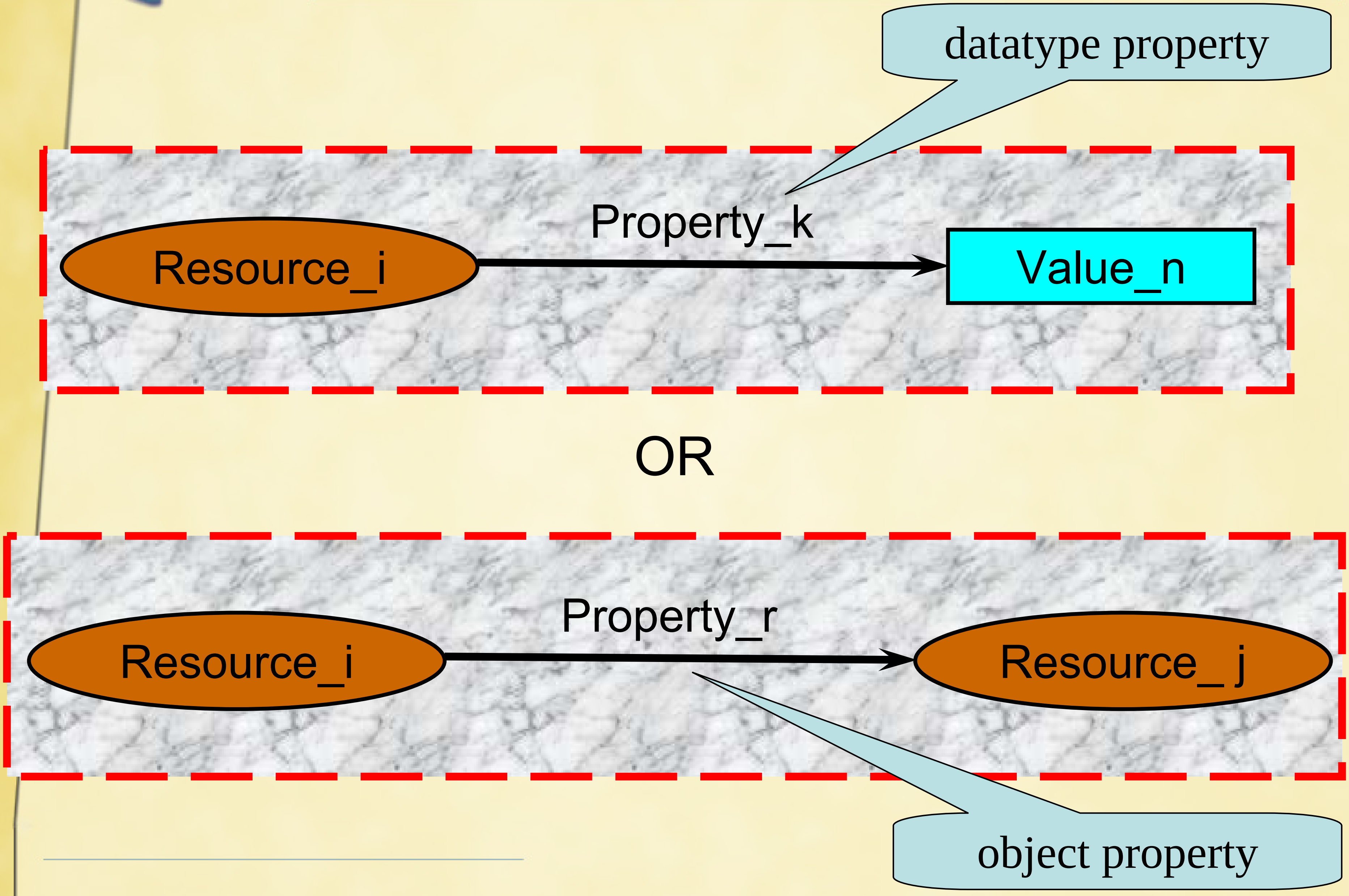
http://www.cs.jyu.fi/ai/vagan/index.html



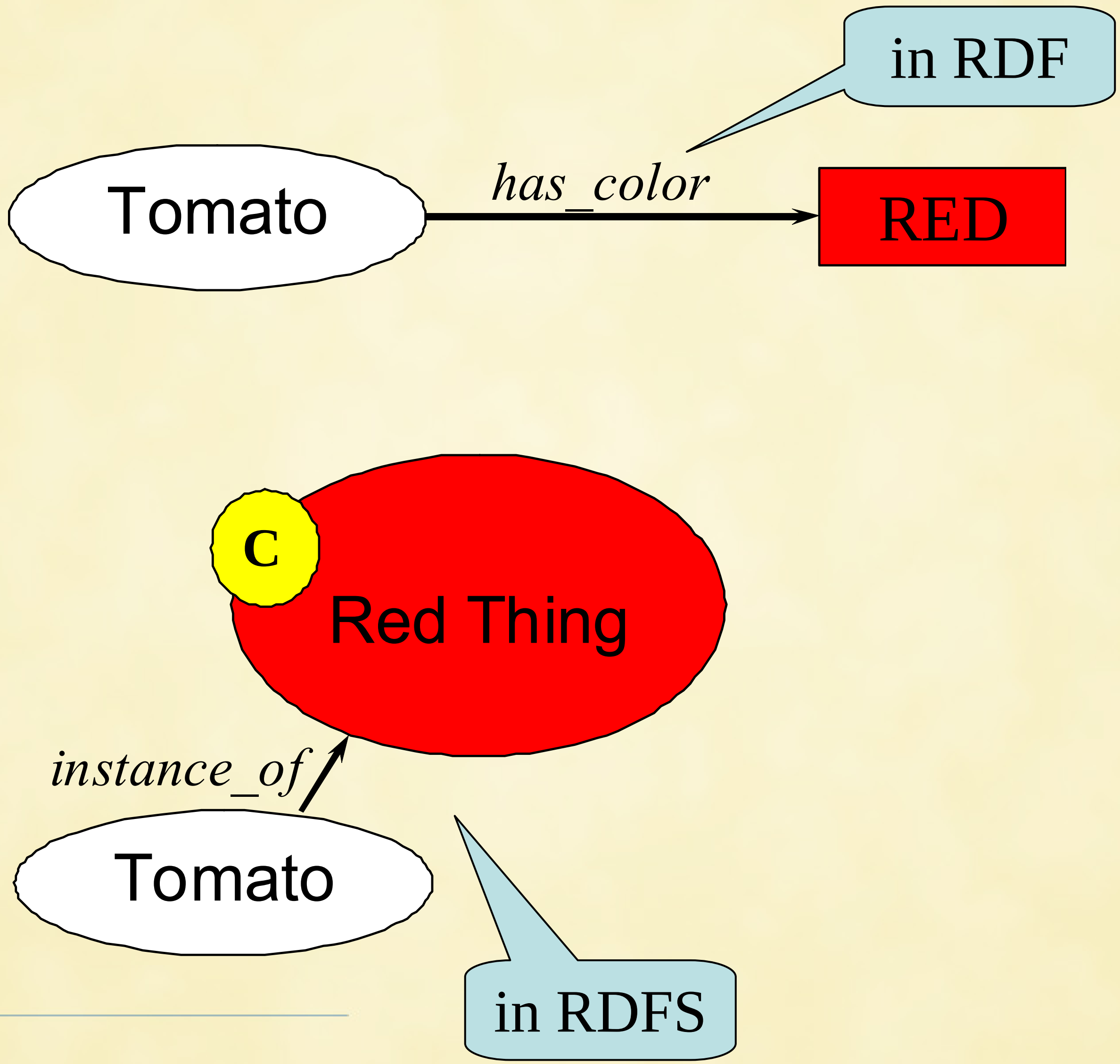
Literal

Personal web page of Terziyan V.

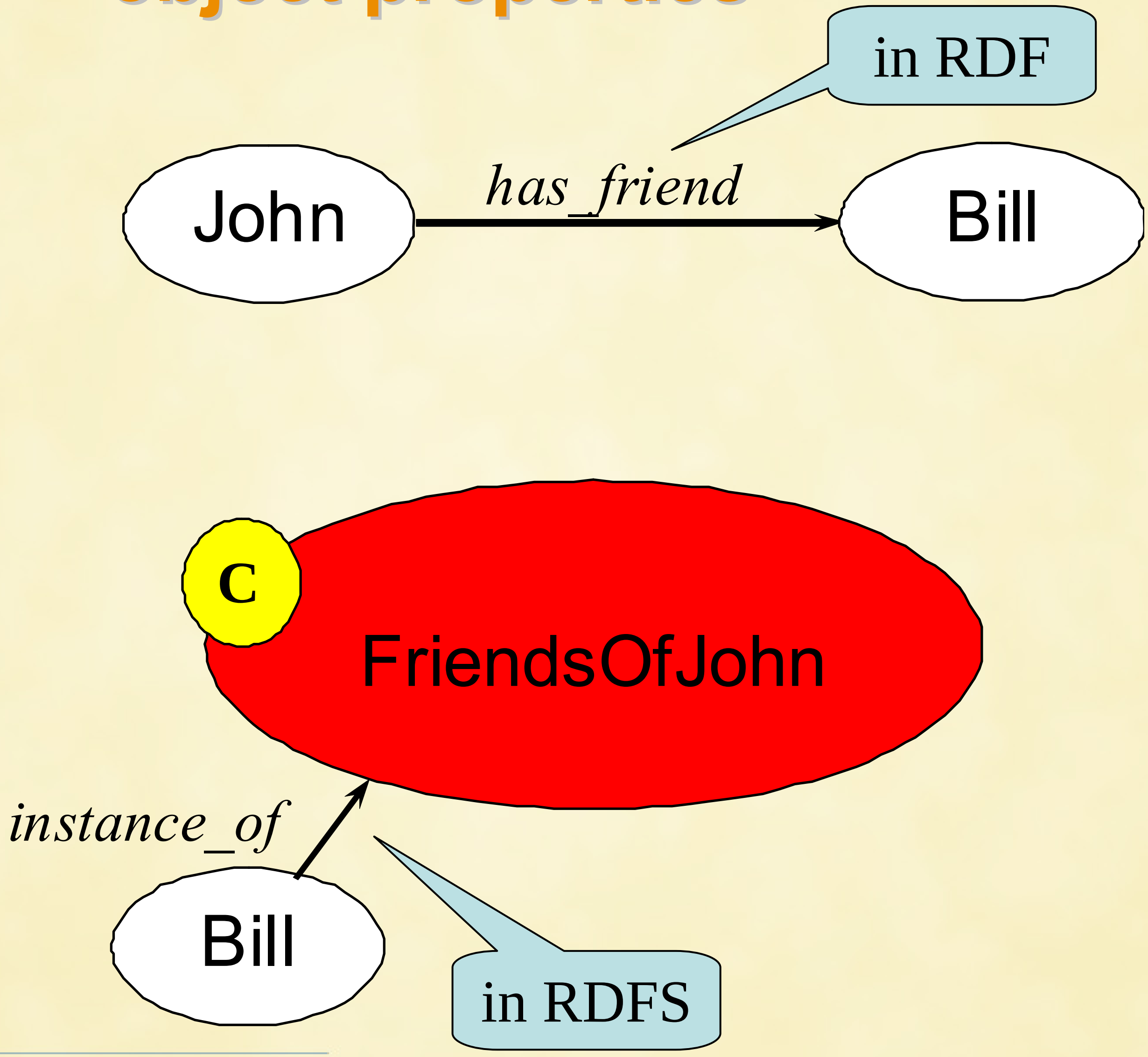
RDF Statement



Different Ways to Represent datatype properties

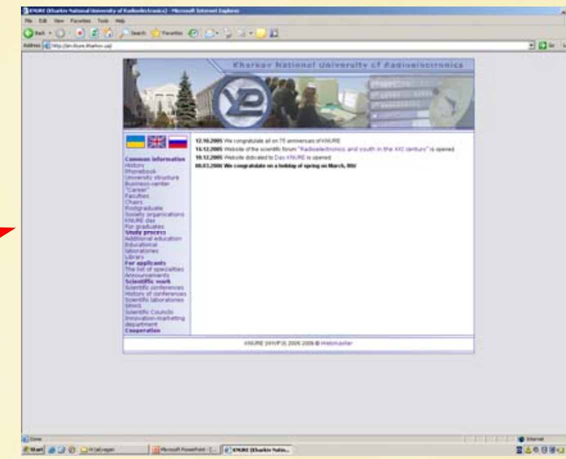


Different Ways to Represent object properties



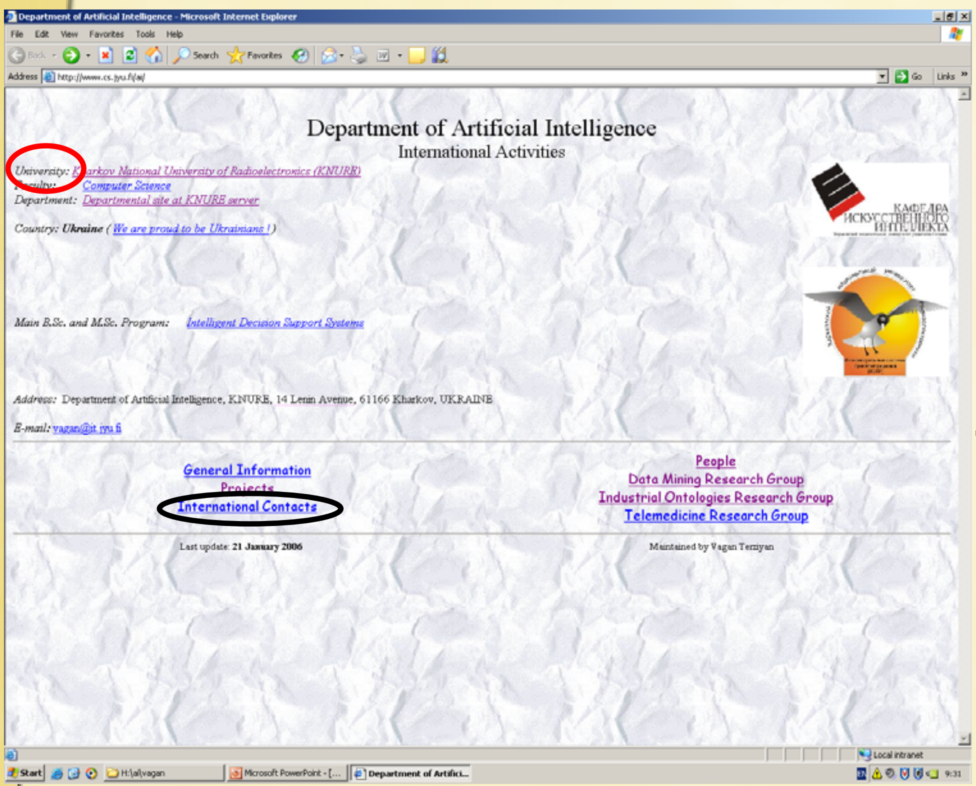
From Hyperlinks to Semantic Web

http://www.kture.kharkov.ua/



university

http://www.cs.jyu.fi/ai/



international
_contacts

http://www.cs.jyu.fi/ai/contacts.html

Semantic Web as the "Web of Trust"

(RDF Reification example)

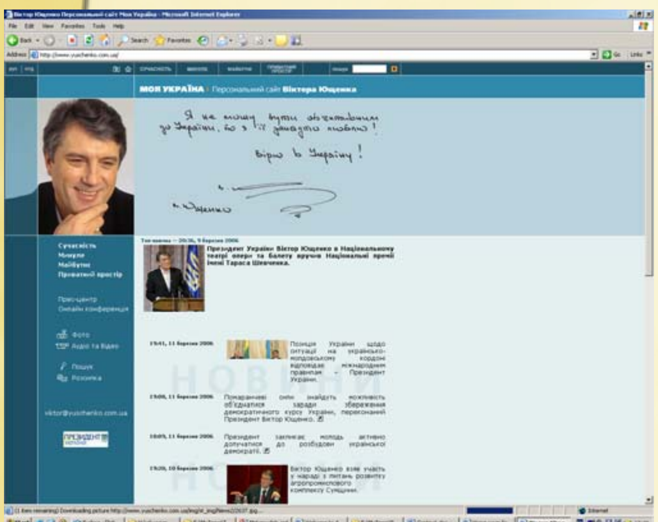
Source: "*Ukrayinska Pravda*"

agrees

has_polytical_opponent

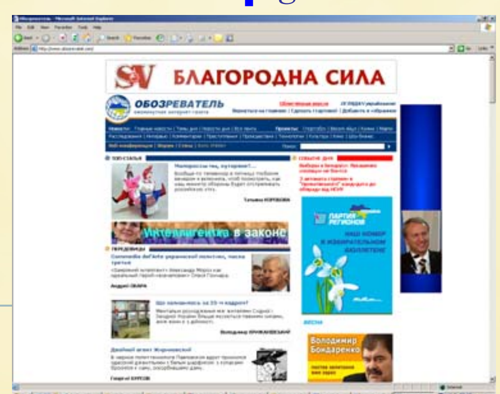
has_coalition_partner

agrees



V. Yuschenko

V. Yanukovich



Source: "*Obozrevatel*"

RDF N3 examples

- Simple statement

```
:John :Loves :Mary
```

- Reified statement

```
{:John :Loves :Mary} :accordingTo :Bill
```

- Goal statement:

```
:I :want {:John :Loves :Mary}
```

Dublin Core

- A set of fifteen basic properties for describing generalised Web resources: http://dublincore.org/documents/dces/
- ISO Standard 15836-2003 (February 2003):

  http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=52142

The Dublin Core Metadata Initiative is an open forum engaged in the development of interoperable online metadata standards that support a broad range of purposes and business models.

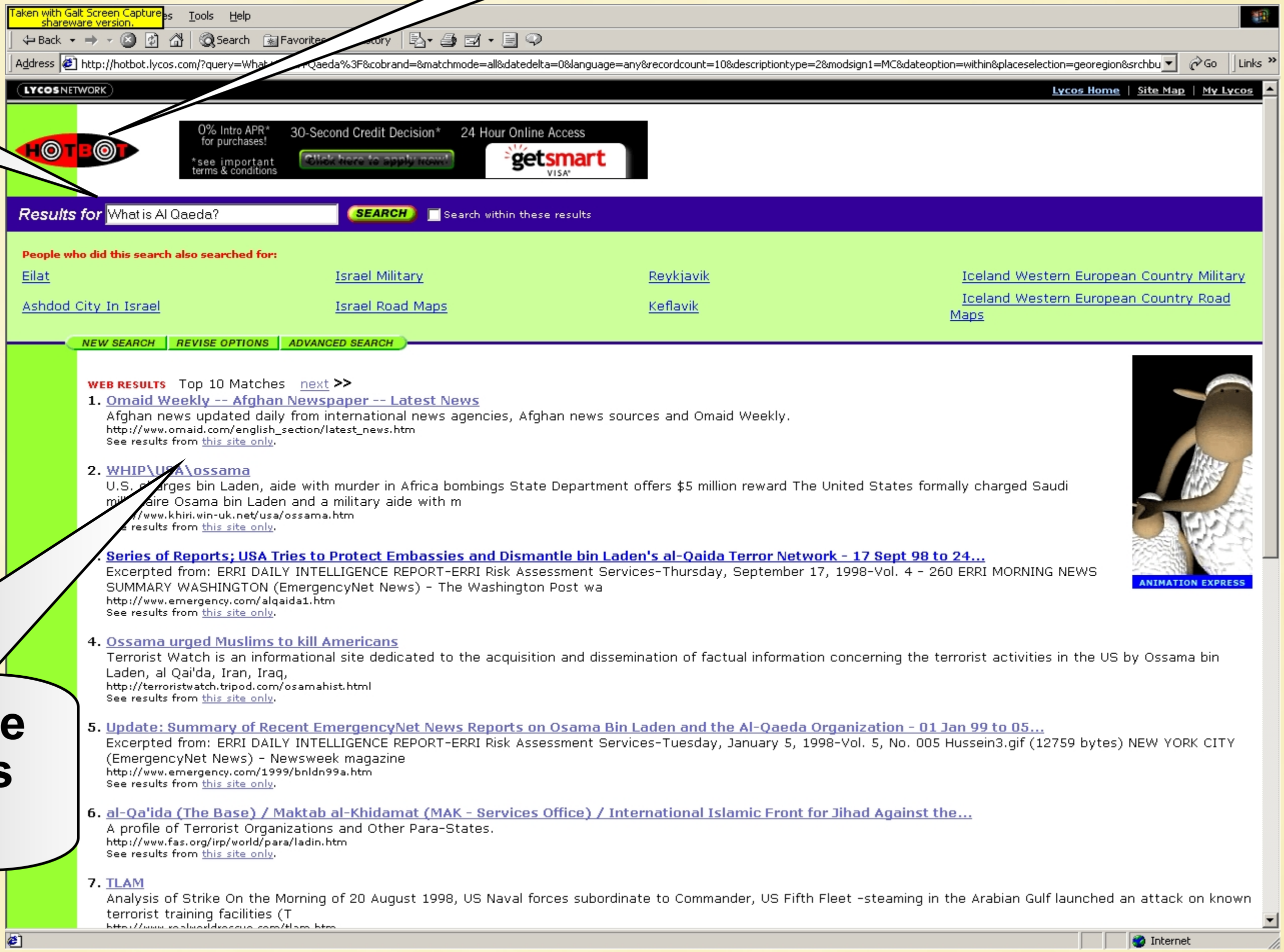
Dublin Core (15 basic properties):

- Title
- Creator
- Subject
- Description
- Publisher
- Contributor
- Date
- Type
- Format
- Identifier
- Source
- Language
- Relation
- Coverage
- Rights

Query Today

WWW Hotbot

What is Al Qaeda?



The answer may be somewhere in this list of URLs

Semantic Query

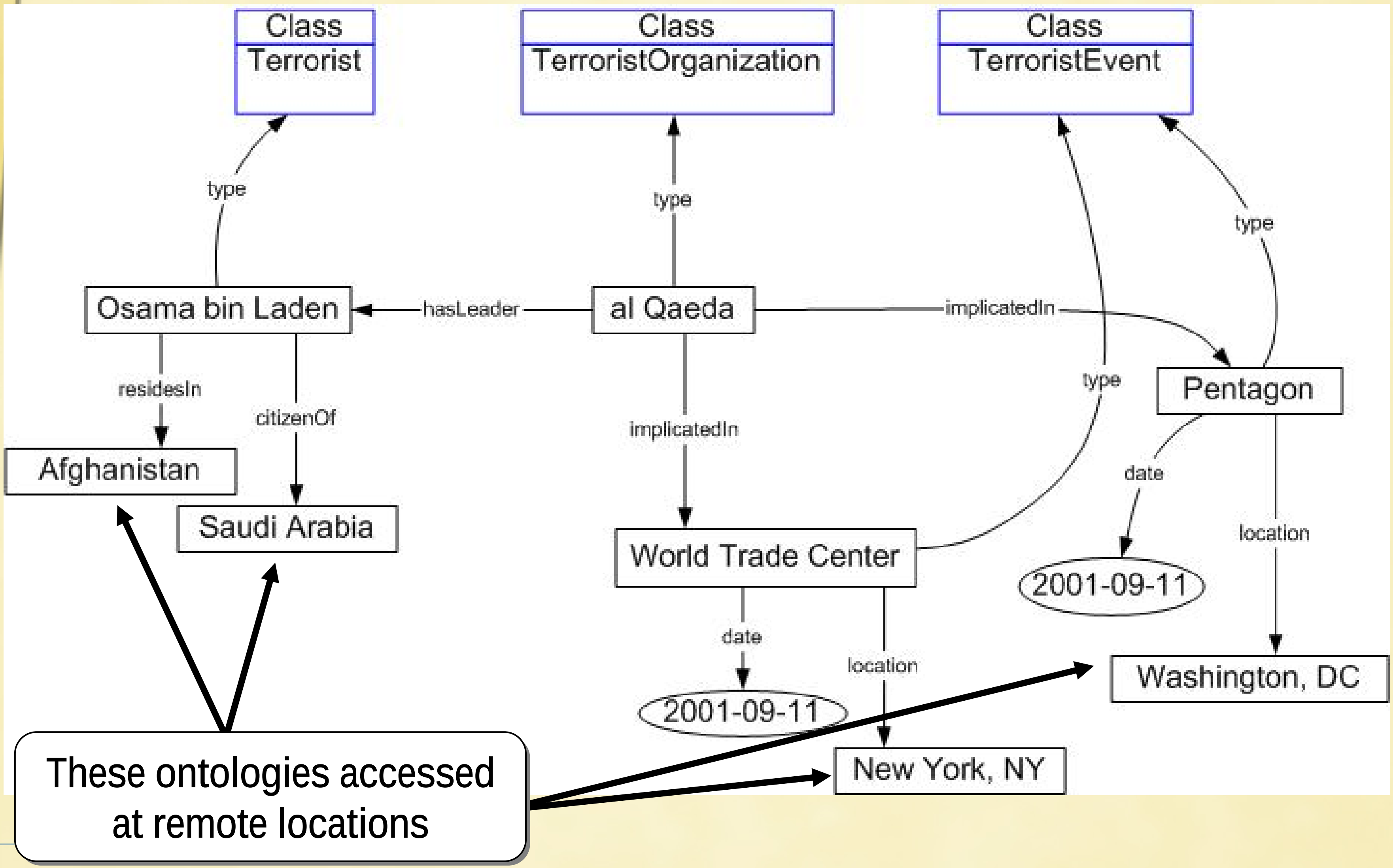
What is Al Qaeda?

A terrorist organization

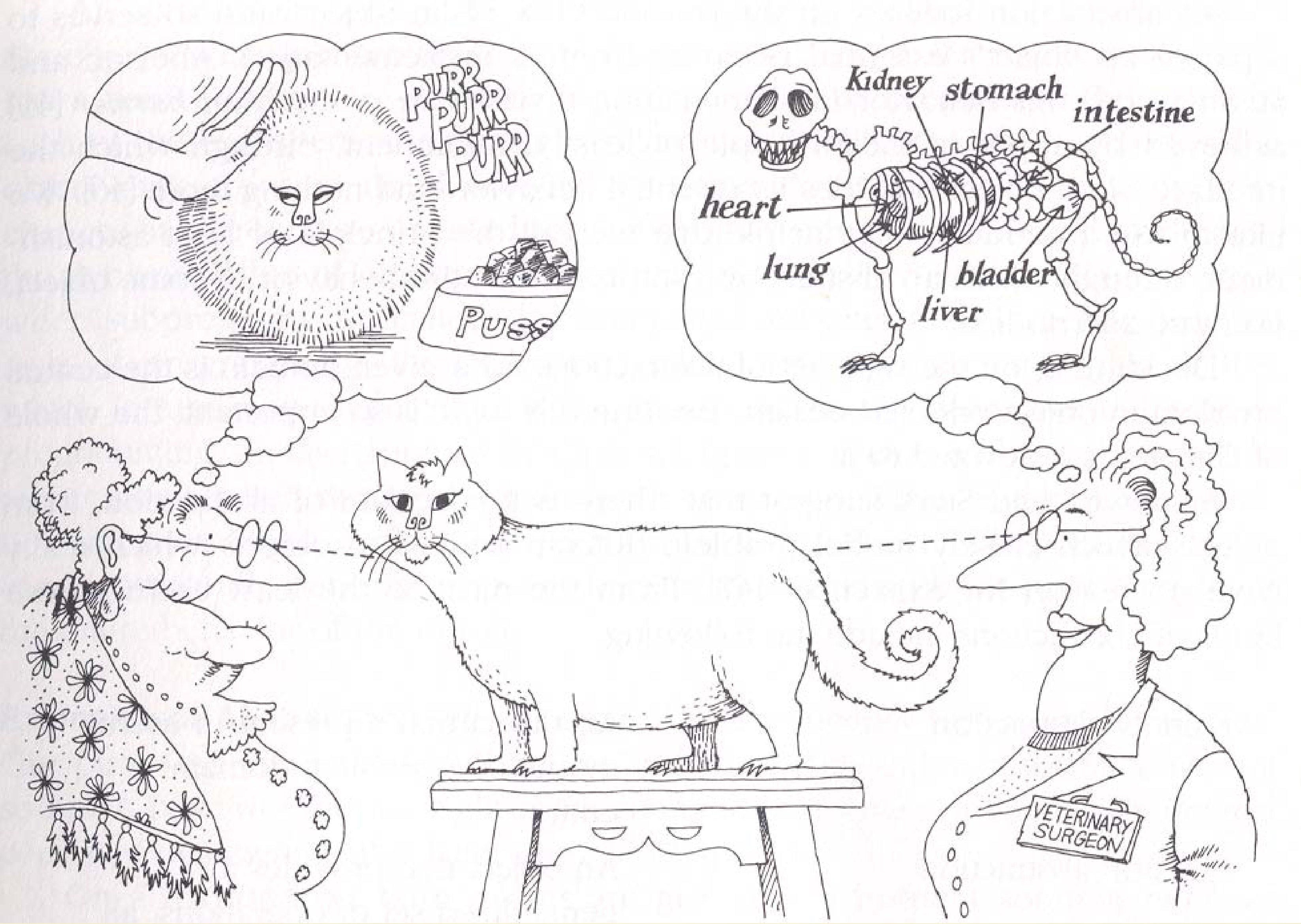
Would you like additional information on?

- Membership
- Locations
- Structure
- Finances
- Tactics
- Other terrorist organizations

Example Ontology



Communication between people



From: Ian Horrocks "OWL 2: The Next Generation"

What is an Ontology?

From: Ian Horrocks "OWL 2: The Next Generation"

What is an Ontology?

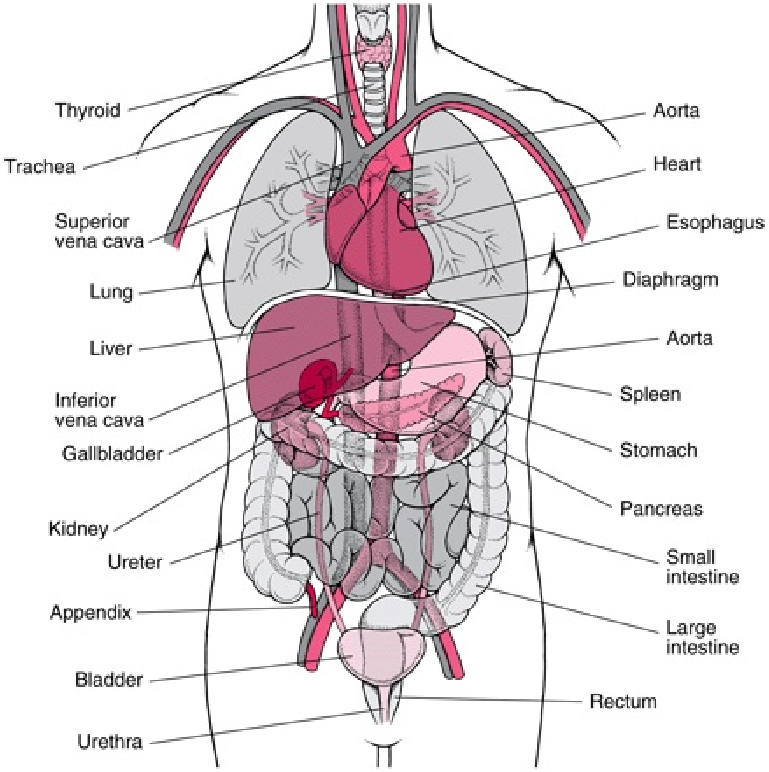
A model of (some aspect of) the world

From: Ian Horrocks "OWL 2: The Next Generation"

What is an Ontology?

A model of (some aspect of) the world

- Introduces **vocabulary** relevant to domain, e.g.:
  - Anatomy

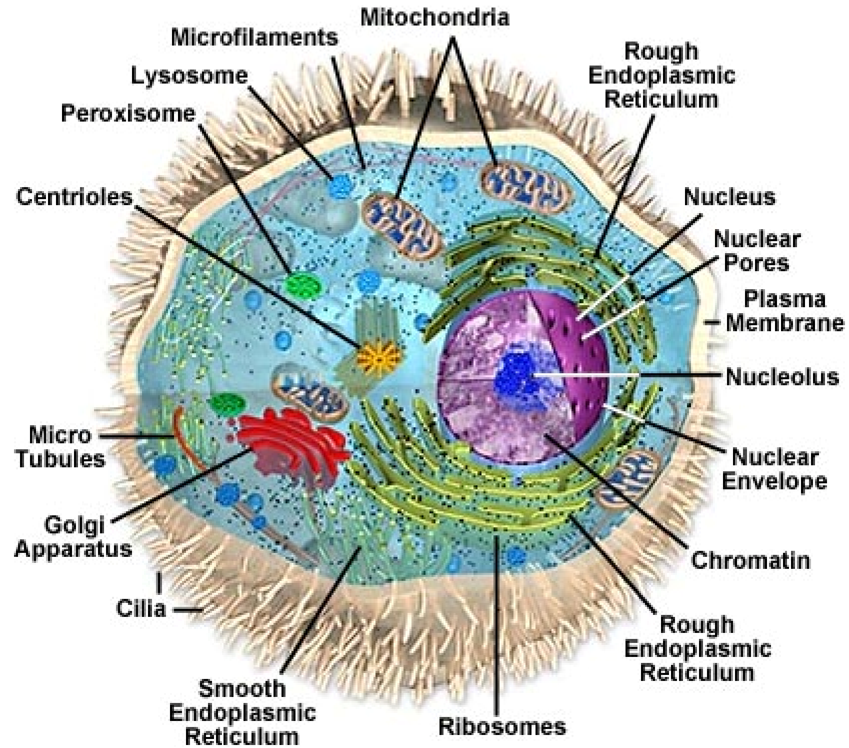


From: Ian Horrocks "OWL 2: The Next Generation"

What is an Ontology?

A model of (some aspect of) the world

- Introduces **vocabulary** relevant to domain, e.g.:
  - Anatomy
  - Cellular biology

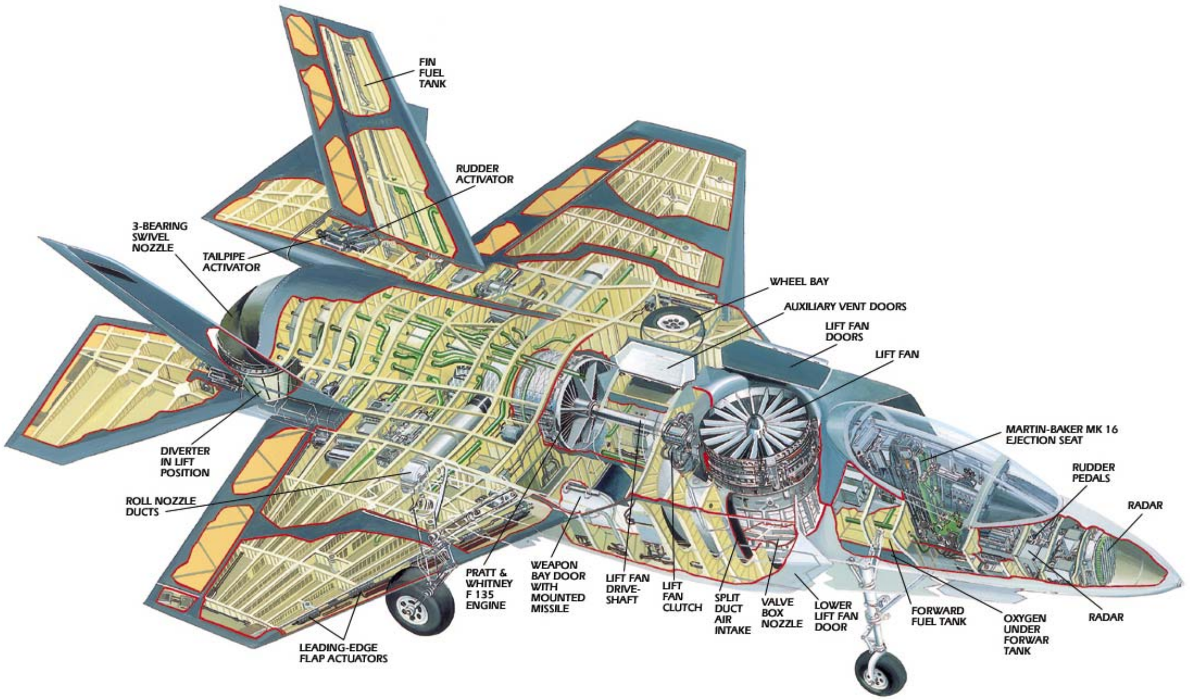


From: Ian Horrocks "OWL 2: The Next Generation"

What is an Ontology?

A model of (some aspect of) the world

- Introduces **vocabulary** relevant to domain, e.g.:
  - Anatomy
  - Cellular biology
  - Aerospace

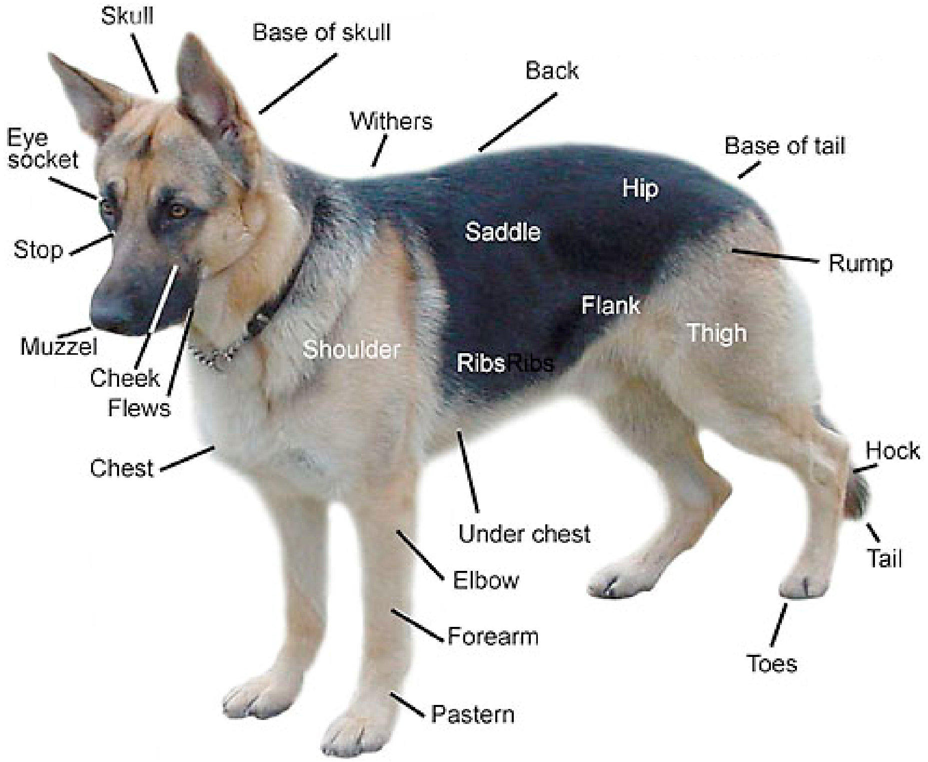


From: Ian Horrocks "OWL 2: The Next Generation"

What is an Ontology?

A model of (some aspect of) the world

- Introduces **vocabulary** relevant to domain, e.g.:
  - Anatomy
  - Cellular biology
  - Aerospace
  - Dogs

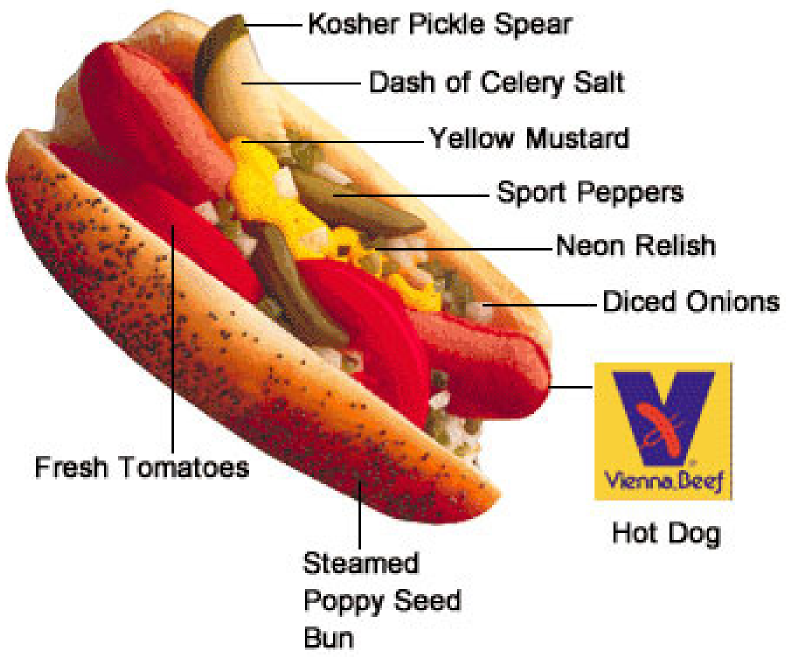


From: Ian Horrocks "OWL 2: The Next Generation"

What is an Ontology?

A model of (some aspect of) the world

- Introduces **vocabulary** relevant to domain, e.g.:
  - Anatomy
  - Cellular biology
  - Aerospace
  - Dogs
  - Hotdogs
  - …



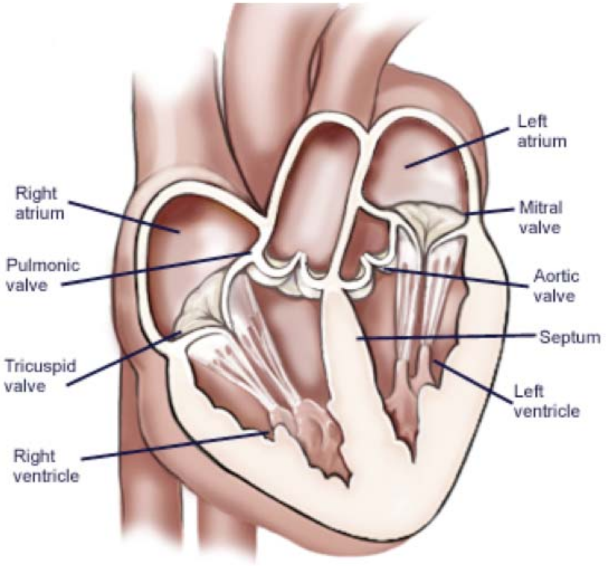
From: Ian Horrocks "OWL 2: The Next Generation"

What is an Ontology?

A model of (some aspect of) the world

- Introduces **vocabulary** relevant to domain
- Specifies **meaning** of terms

Heart is a muscular organ that is part of the circulatory system



From: Ian Horrocks “OWL 2: The Next Generation”

What is an Ontology?

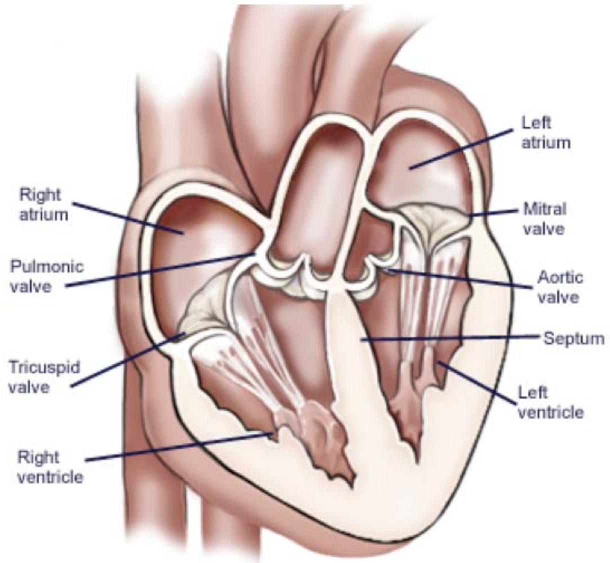
A model of (some aspect of) the world

- Introduces **vocabulary** relevant to domain
- Specifies **meaning** of terms

  Heart is a muscular organ that is part of the circulatory system

- **Formalised** using suitable logic

$$\forall x.[\mathsf{Heart}(x) \rightarrow \mathsf{MuscularOrgan}(x) \wedge \exists y.[\mathsf{isPartOf}(x,y) \wedge \mathsf{CirculatorySystem}(y)]]$$



DL Semantics

From: **Ian Horrocks “OWL: A Description Logic Based Ontology Language”**

Semantics given by standard FO model theory:

Interpretation function I | Interpretation domain Δ^{I}

Individuals i^{I} 2 Δ^{I}

Individual Resources

John

Mary

Concepts C^{I} µ Δ^{I}

Classes of Resources

Lawyer

Doctor

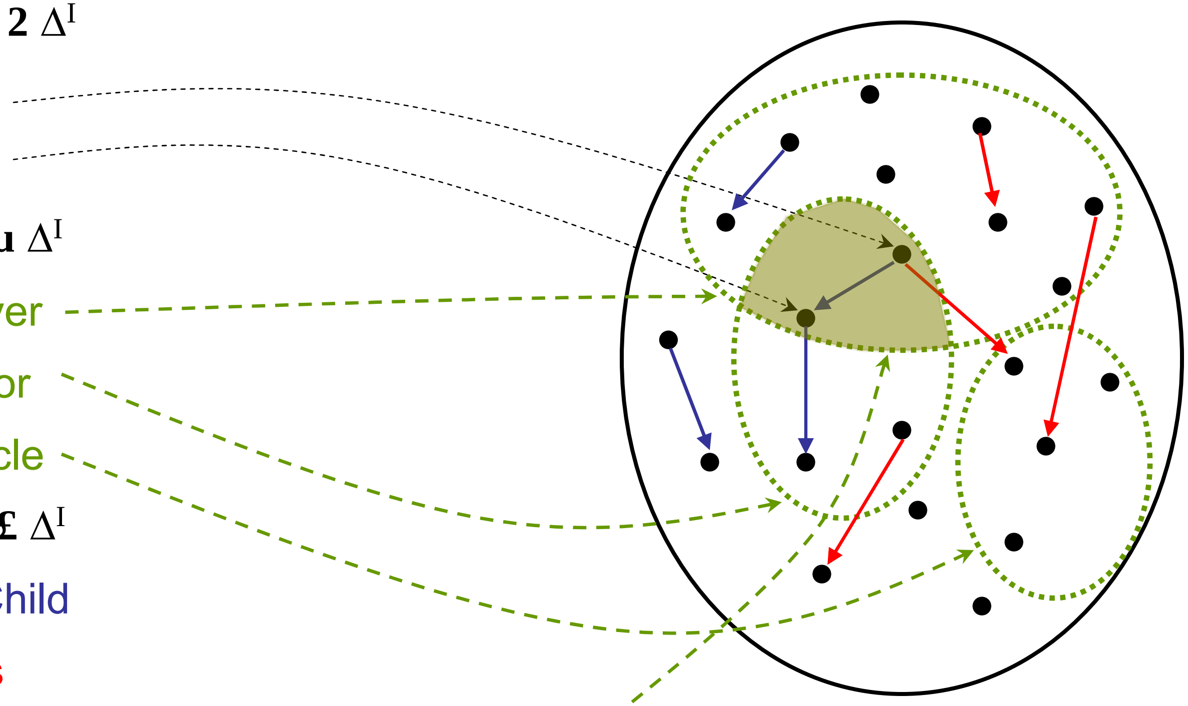
Vehicle

Roles r^{I} µ Δ^{I} £ Δ^{I}

Properties of Resources

hasChild

owns



(Lawyer u Doctor)

What is an ontology?

Studer(98): Formal, explicit specification of a shared conceptualization

Machine readable

Consensual knowledge

Concepts, properties, functions, axioms are explicitly defined

Abstract model of some phenomena in the world

Ontology Elements

- Concepts(classes) + their hierarchy
- Concept properties (slots/attributes)
- Property restrictions (type, cardinality, domain)
- Relations between concepts (disjoint, equality)
- Instances

How to build an ontology?

Steps:

- determine domain and scope
- enumerate important terms
- define classes and class hierarchies
- define slots
- define slot restrictions (cardinality, value-type)

Step 1: Determine Domain and Scope

Domain: geography

Application: route planning agent

Possible questions:

- Distance between two cities?
- What sort of connections exist between two cities?
- In which country is a city?
- How many borders are crossed?

Step 2: Enumerate Important Terms

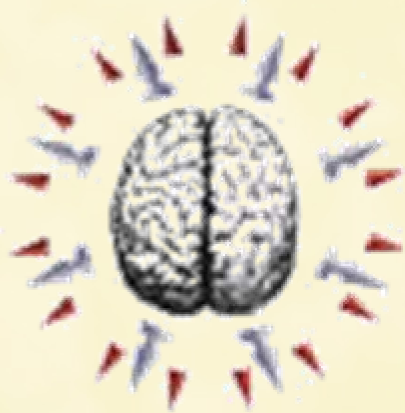
city

capital

Connection_on_land

country

border



road

railway

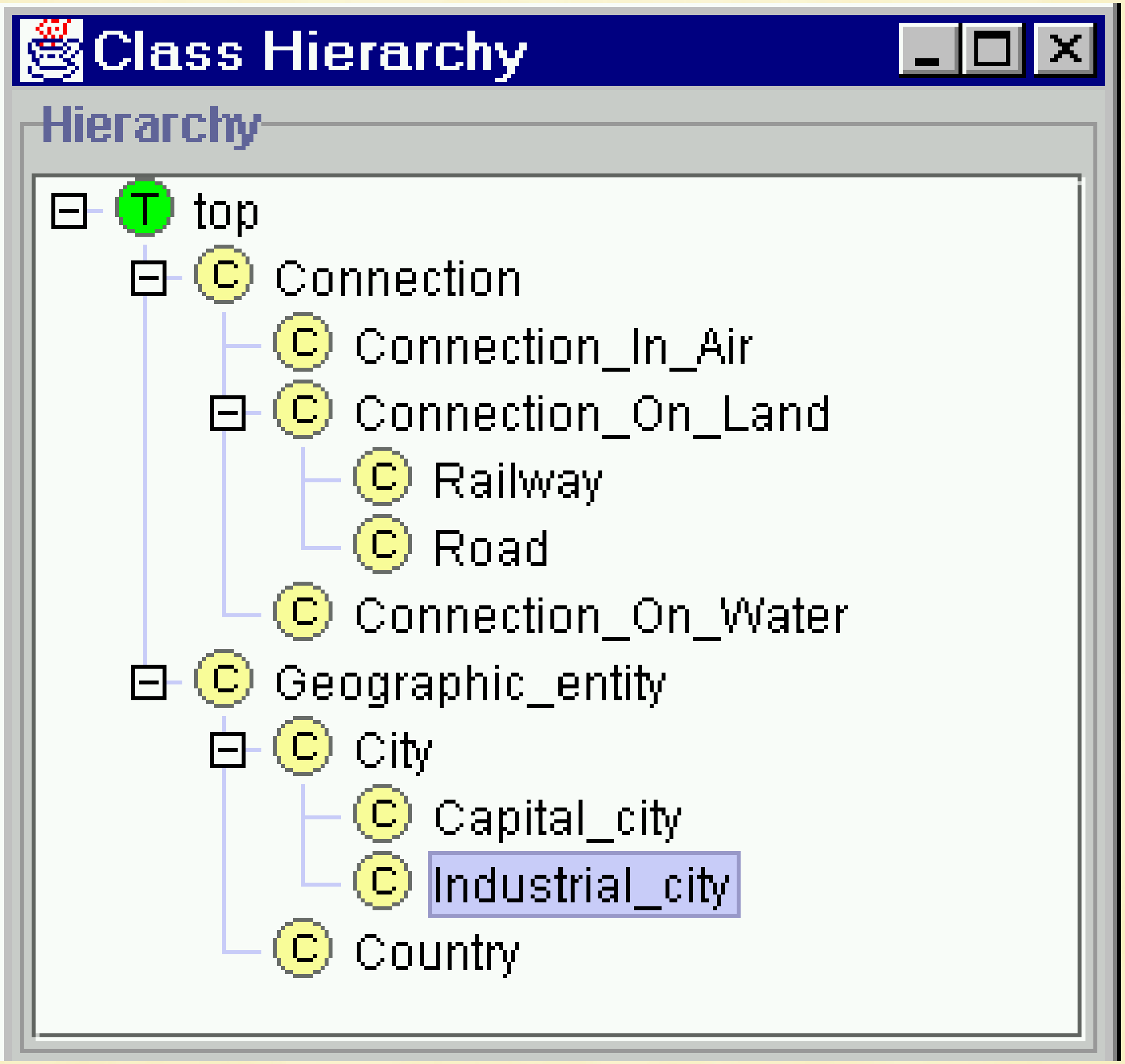
Connection_on_water

currency

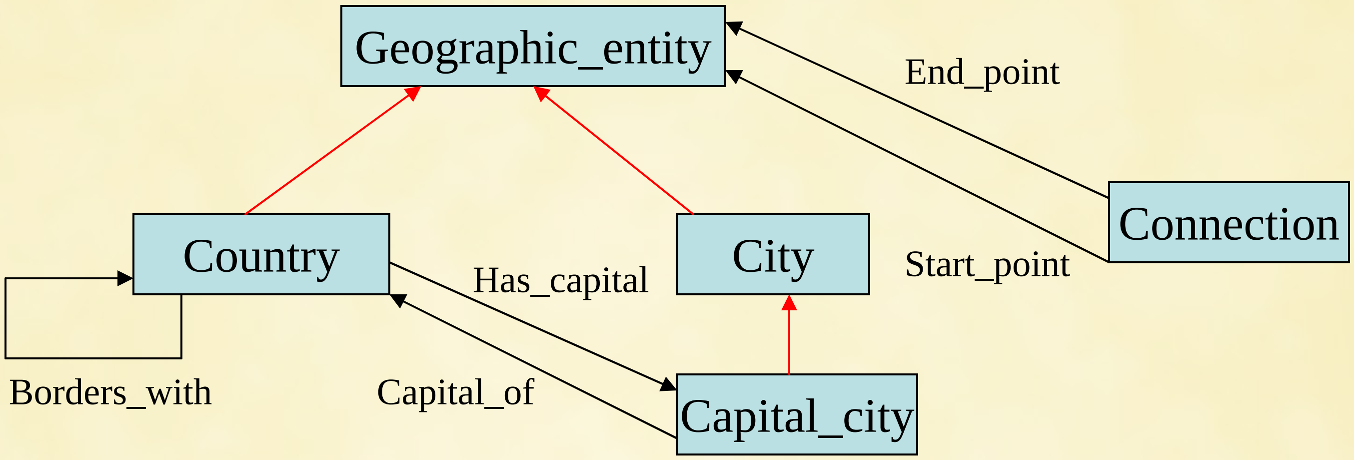
Connection_in_air

connection

Step 3: Define Classes and Class Hierarchy



Step 4: Define Slots of Classes



Step 5: Define slot constraints

- Slot-cardinality

  Ex: Borders_with multiple, *Start_point* single

- Slot-value type

  Ex: Borders_with- Country

RDF and OWL became standard

- 10 February 2004 the World Wide Web Consortium announced final approval of two key Semantic Web technologies, the revised Resource Description Framework (**RDF**) and the Web Ontology Language (**OWL**).
- Read more in: http://www.w3.org/2004/OWL/ and http://www.w3.org/RDF/
- 29 October 2009 the **OWL 2** (with some qualitative updates to OWL) has become a W3C recommendation.
- Read more in: http://www.w3.org/TR/owl2-primer/

OWL Example

- There are two types of animals, Male and Female.

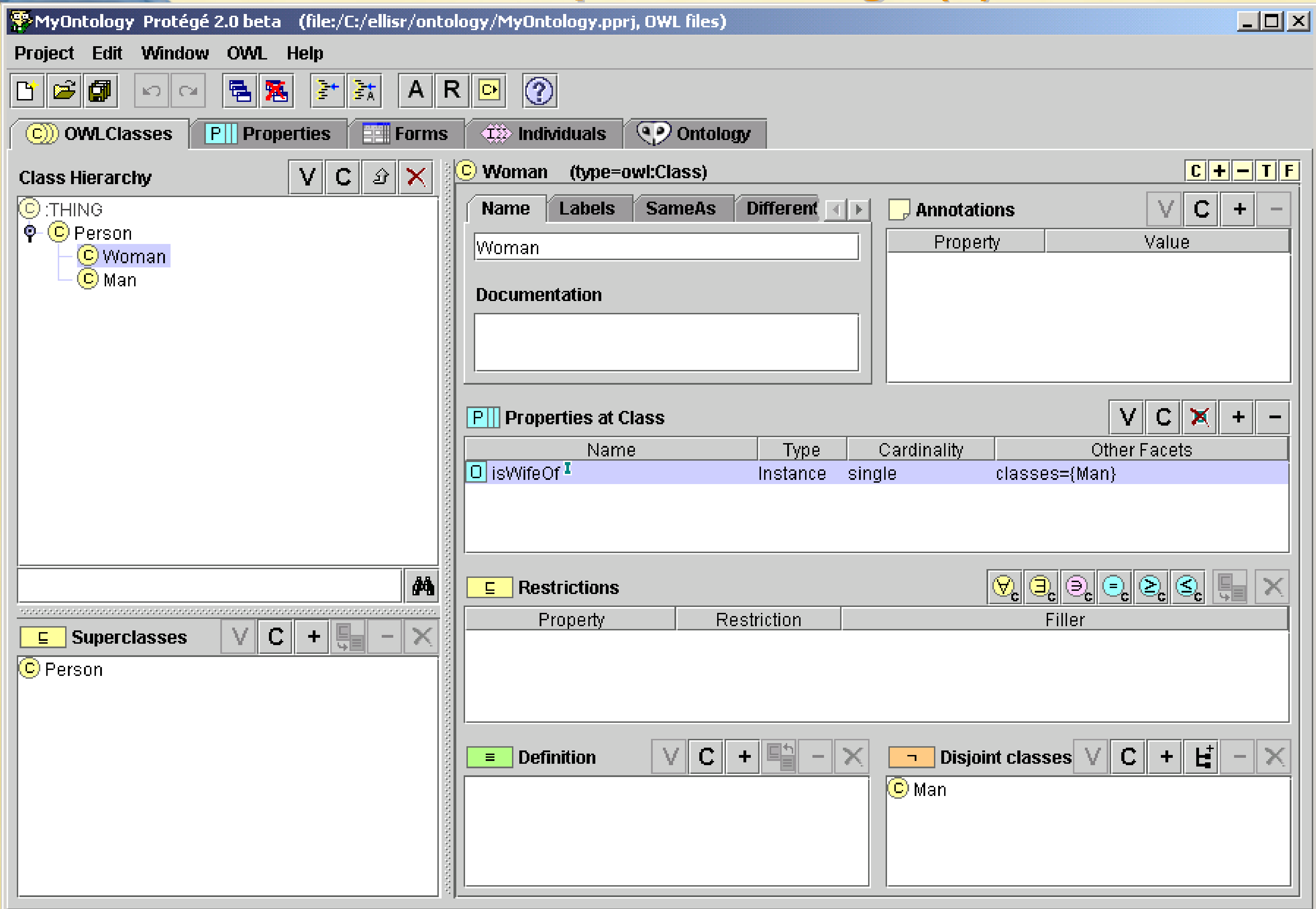
```
<rdfs:Class rdf:ID="Male">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</rdfs:Class>
```

- The subClassOf element asserts that its subject - Male - is a subclass of its object -- the resource identified by #Animal.

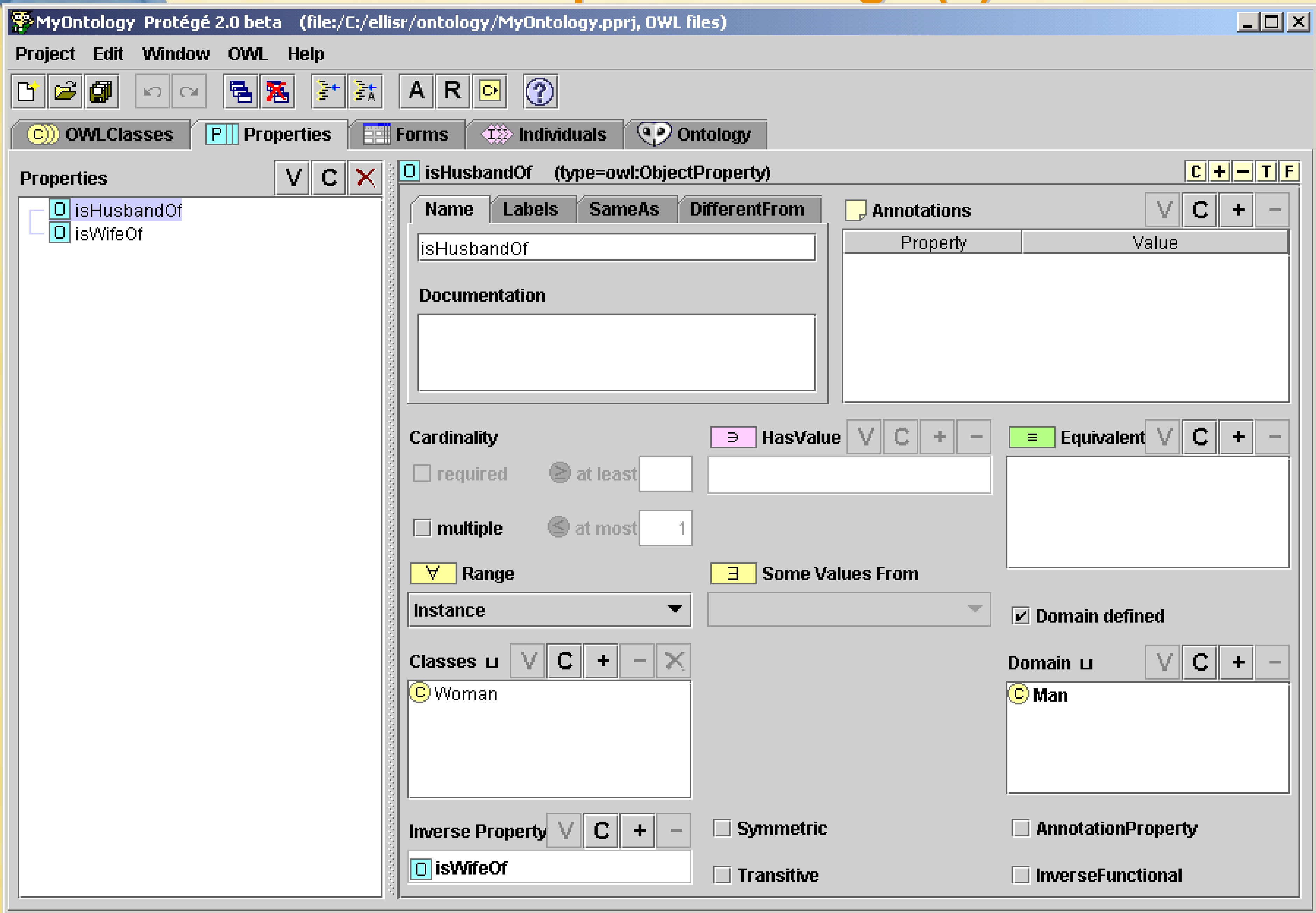
```
<rdfs:Class rdf:ID="Female">
  <rdfs:subClassOf rdf:resource="#Animal"/>
  <owl:disjointWith rdf:resource="#Male"/>
</rdfs:Class>
```

- Some animals are Female, too, but nothing can be both Male and Female (in this ontology) because these two classes are disjoint (using the disjointWith tag).

OWL Example in Protégé (1)



OWL Example in Protégé (2)



OWL on one Slide

- **Symmetric**: if P(x, y) then P(y, x)
- **Transitive**: if P(x,y) and P(y,z) then P(x, z)
- **Functional**: if P(x,y) and P(x,z) then y=z
- **InverseOf**: if P1(x,y) then P2(y,x)
- **InverseFunctional**: if P(y,x) and P(z,x) then y=z
- **allValuesFrom**: P(x,y) and y=allValuesFrom(C)
- **someValuesFrom**: P(x,y) and y=someValuesFrom(C)
- **hasValue**: P(x,y) and y=hasValue(v)
- **cardinality**: cardinality(P) = N
- **minCardinality**: minCardinality(P) = N
- **maxCardinality**: maxCardinality(P) = N
- **equivalentProperty**: P1 = P2
- **intersectionOf**: C = intersectionOf(C1, C2, …)
- **unionOf**: C = unionOf(C1, C2, …)
- **complementOf**: C = complementOf(C1)
- **oneOf**: C = one of(v1, v2, …)
- **equivalentClass**: C1 = C2
- **disjointWith**: C1 != C2
- **sameIndividualAs**: I1 = I2
- **differentFrom**: I1 != I2
- **AllDifferent**: I1 != I2, I1 != I3, I2 != I3, …
- **Thing**: I1, I2, …

Legend:
Properties are indicated by: P, P1, P2, etc
Specific classes are indicated by: x, y, z
Generic classes are indicated by: C, C1, C2
Values are indicated by: v, v1, v2
Instance documents are indicated by: I1, I2, I3, etc.
A number is indicated by: N
P(x,y) is read as: “property P relates x to y”

An Example

- Woman ≡ Person ⊓ Female
- Man ≡ Person ⊓ ¬Woman
- Mother ≡ Woman ⊓ ∃hasChild.Person
- Father ≡ Man ⊓ ∃hasChild.Person
- Parent ≡ Father ⊔ Mother
- Grandmother ≡ Mother ⊓ ∃hasChild.Parent

We can further infer (though not explicitly stated):

→ Grandmother ⊑ Person

Grandmother ⊑ Man ⊔ Woman

etc.

Resources

- W3C Documents
  - Guide: http://www.w3.org/TR/owl-guide/
  - Reference: http://www.w3.org/TR/owl-ref/
  - Semantics and Abstract Syntax: http://www.w3.org/TR/owl-semantics/
- OWL Tutorial
  - Ian Horrocks, Sean Bechhofer: http://www.cs.man.ac.uk/~horrocks/Slides/Innsbruck-tutorial/
- Example Ontologies, e.g. here:

  http://www.daml.org/ontologies/

  http://protegewiki.stanford.edu/index.php/Protege_Ontology_Library

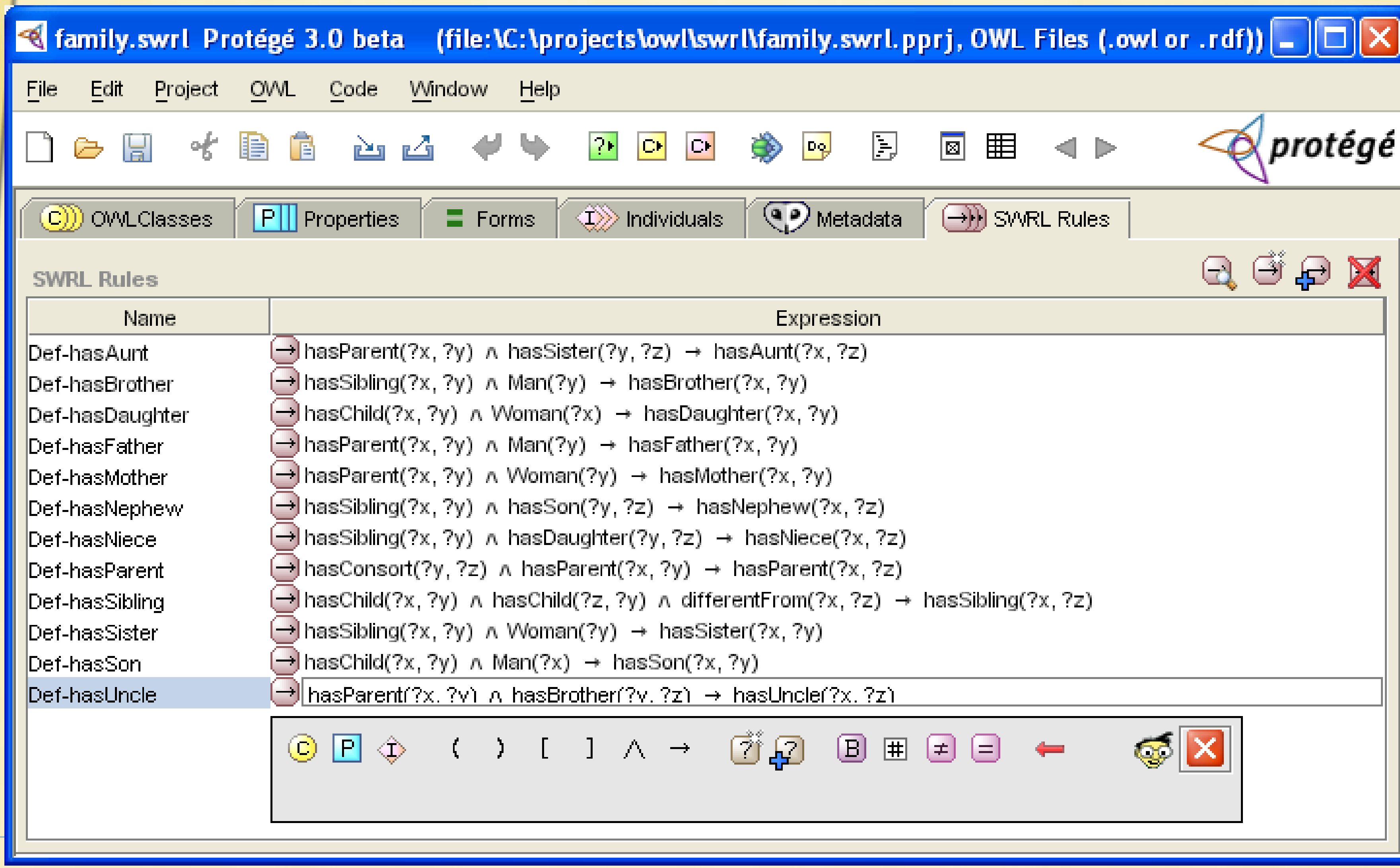
Tutorial: Designing Ontologies with Protégé

- Protégé is an ontology editor and a knowledge-base editor (download from http://protege.stanford.edu).
- Protégé is also an open-source, Java tool that provides an extensible architecture for the creation of customized knowledge-based applications.
- Protégé's OWL Plug-in now provides support for editing Semantic Web ontologies.

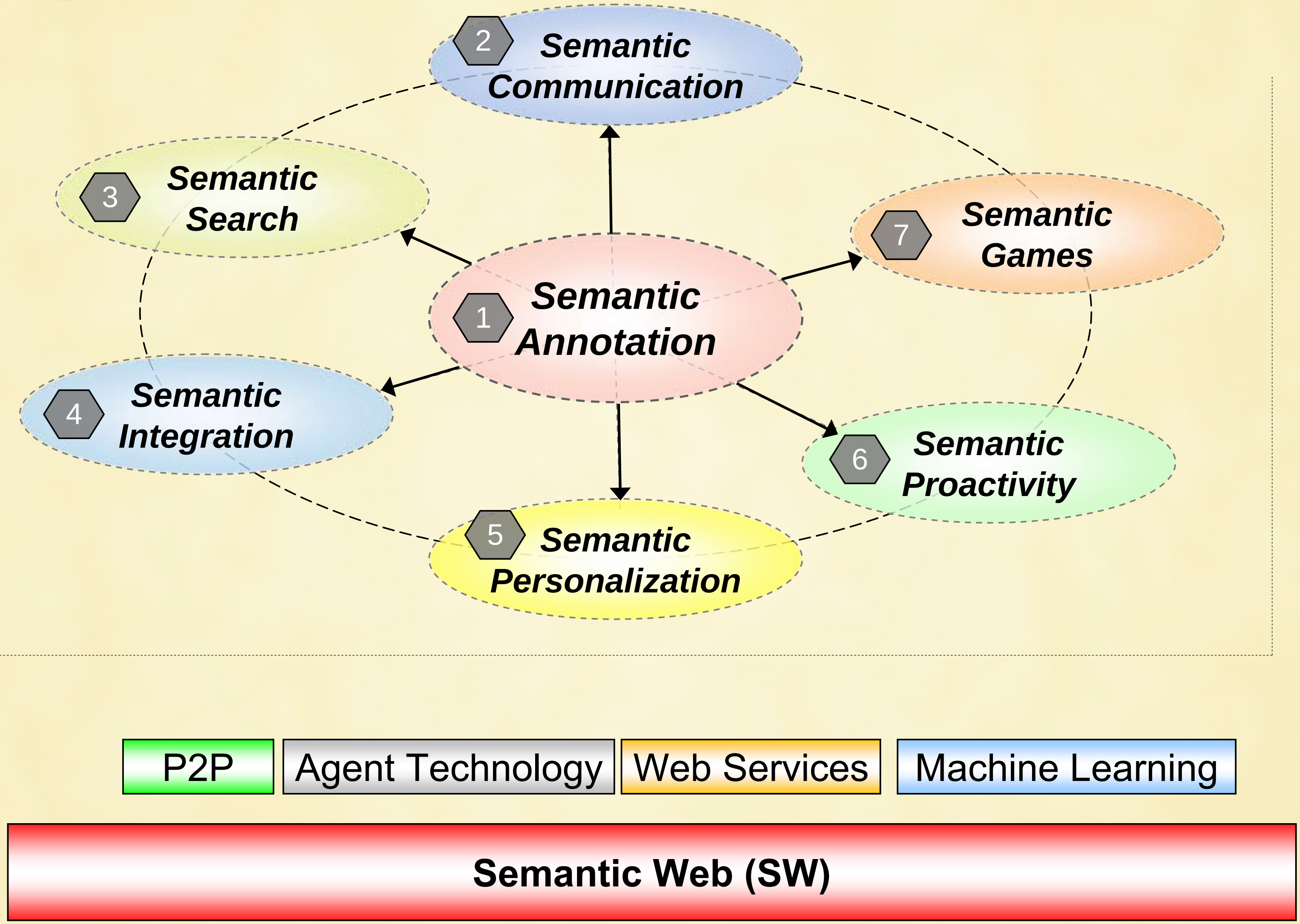
http://www.cs.man.ac.uk/~horrocks/Teaching/cs646/

http://www.co-ode.org/resources/tutorials/ProtegeOWLTutorial.pdf

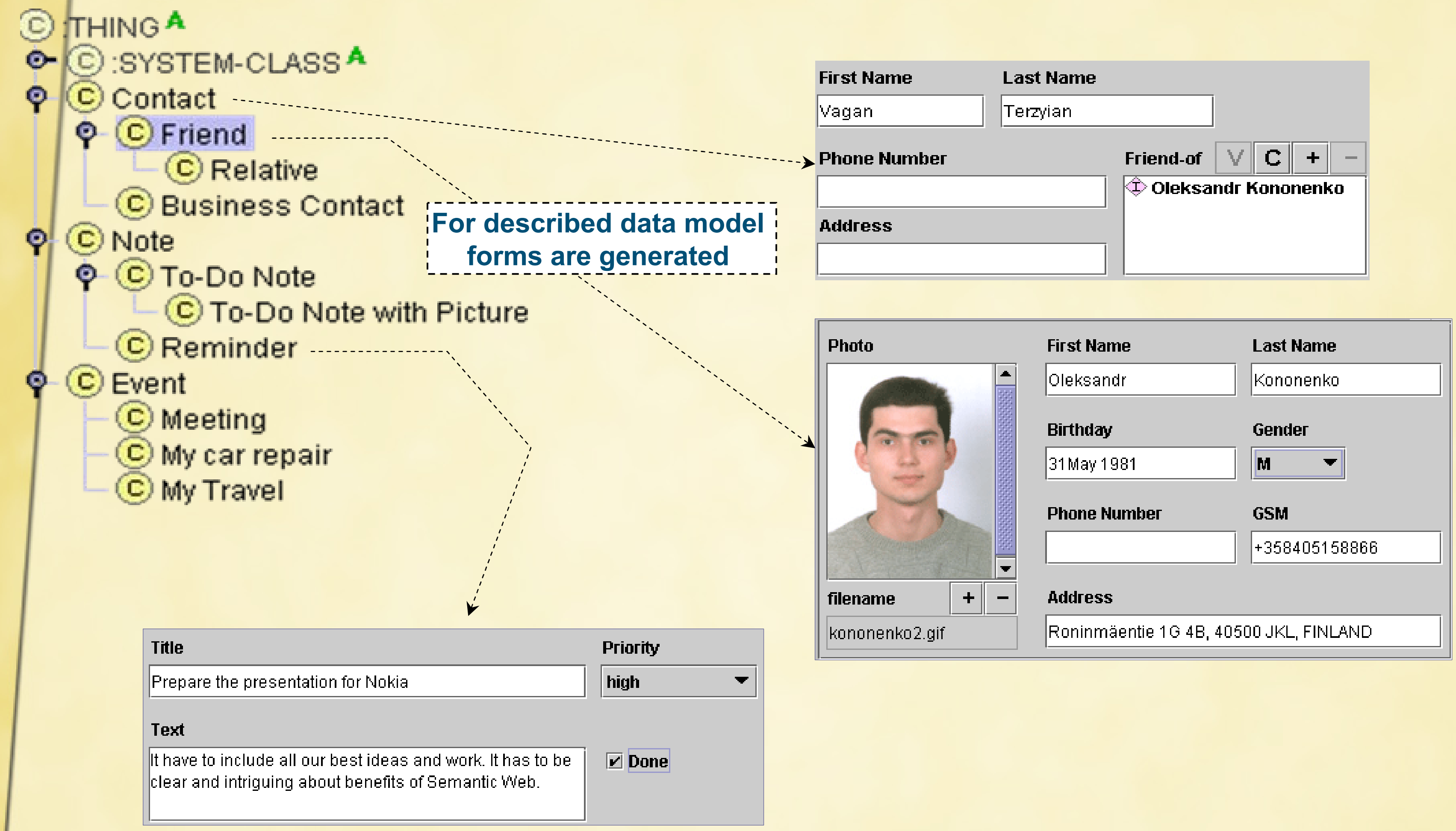
Semantic Rules in SWRL



Technology Roadmap for Applications



Generated interface from Ontology



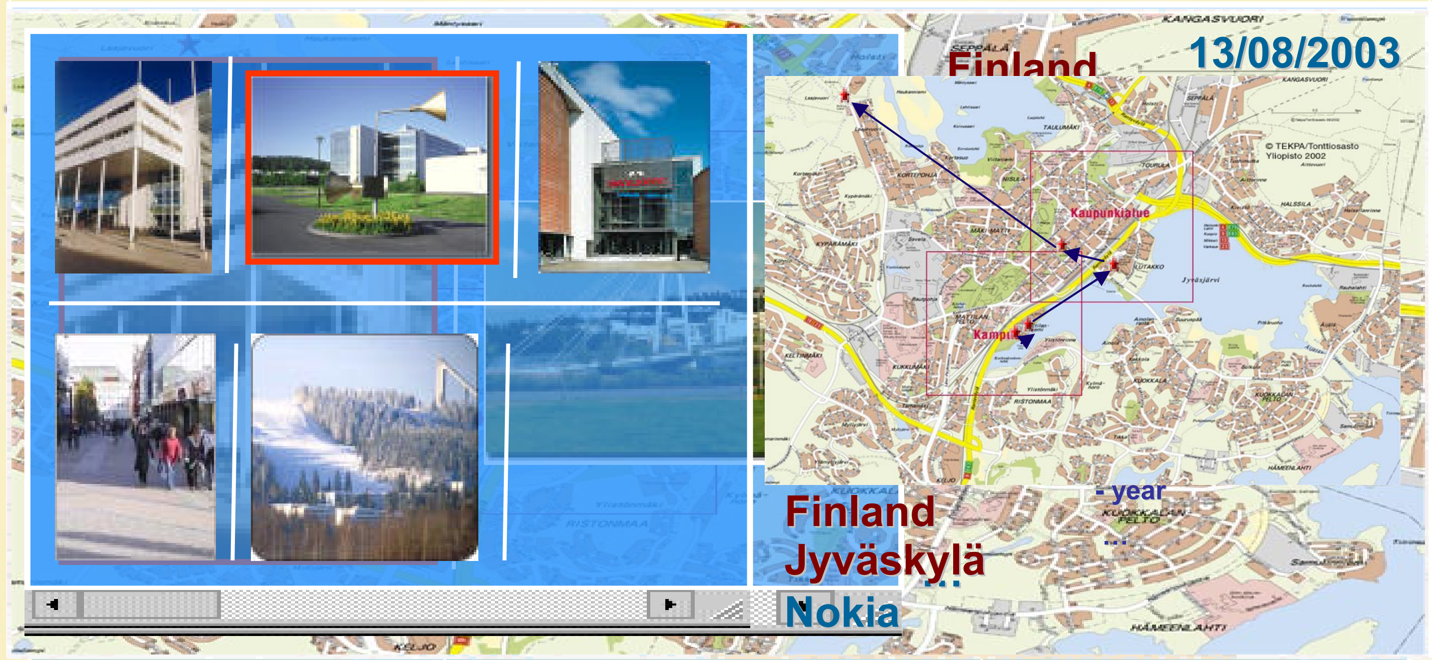
Data view is described as an ontology which contains all needed information about data structure.
User interface is built dynamically from ontology:

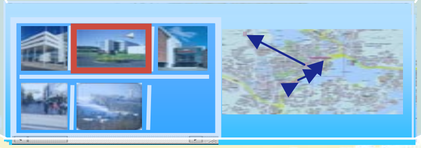
- Fields for data
- Form layout, types of controls (e.g. picture, checkboxes etc.)
- Rules for data that can check some constraints, invoke actions, perform calculations – whatever!

Using image metadata for browsing and linking to other data

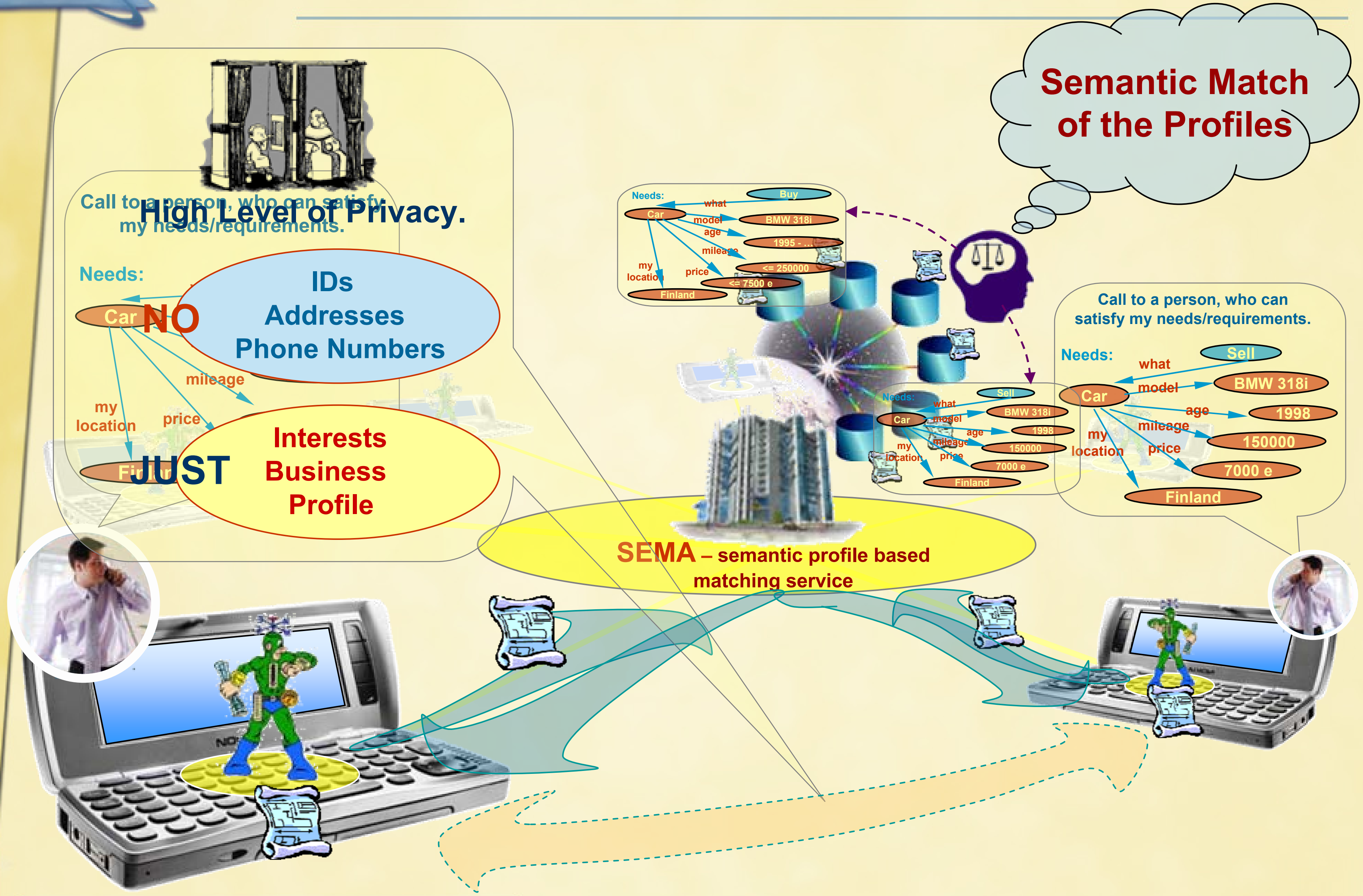
Location based image annotation

Location based Photo Album-Map





Semantic Call

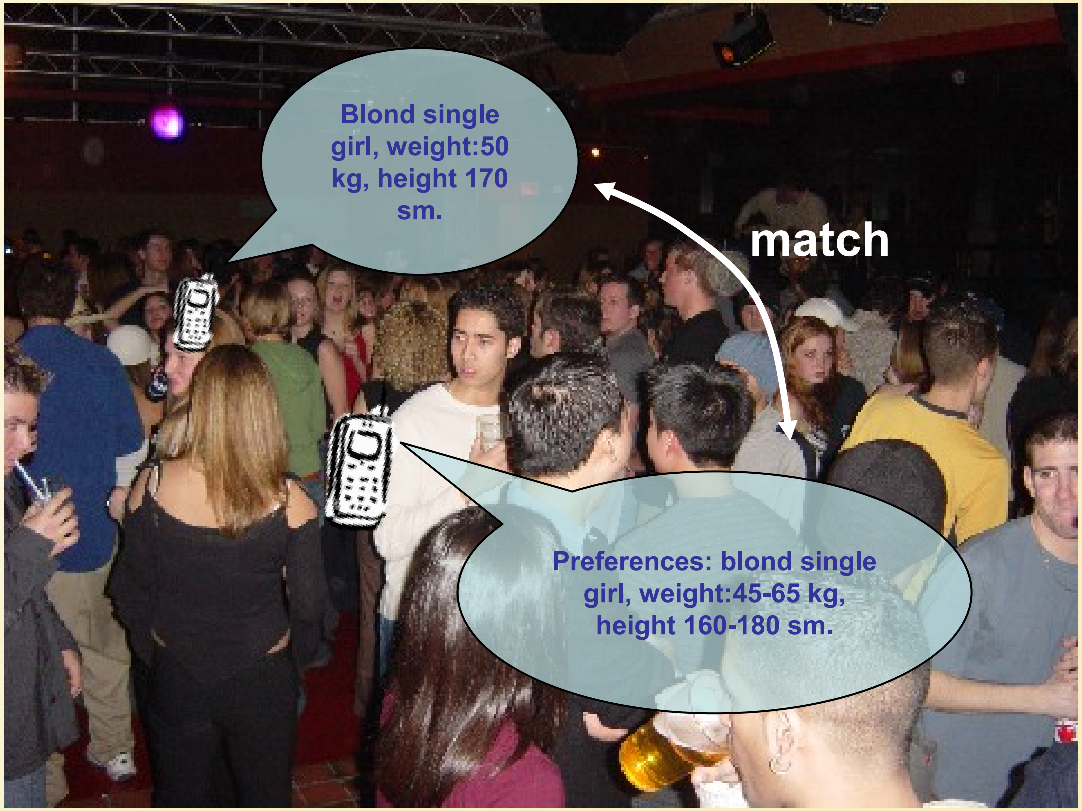


Semantic Call

- Examples:
  - *“Connect me with someone who can sell me cheep (< 500) rowing boat in Jyväskylä”*
  - *“Connect me with a blond girl (21-25) who wants to meet a guy (26) tonight to go to dancing club in Jyväskylä”, etc.*

Semantic Search of People

Searching persons in a P2P environment



- **Every data object/fragment has associated semantic annotation, which makes possible data filtering**
- **Data sharing in big crowds can be performed in the ad-hoc manner (chain messages).**

People gathered for a meeting can browse shared data of each other

Semantic distance calculation (example)

$$D(X,Y)=\sqrt{\sum_{\forall i, x_i \in X, y_i \in Y} \omega_i \cdot d(x_i,y_i)^2}$$

where:

$$d(x_i,y_i)=\begin{cases} \text{if } i-\text{th attribute is nominal - } \begin{cases} 0, \text{ if } x_i = y_i \\ 1, \text{ otherwise} \end{cases} \\ \text{else}: \dfrac{|x_i - y_i|}{range_i} \end{cases}$$

Wine Preference 1:

I prefer white wine served at 15° C

Wine Preference 2:

I prefer red wine served at 25° C

Importance:

Wine color: $\omega_1 = 0.7$

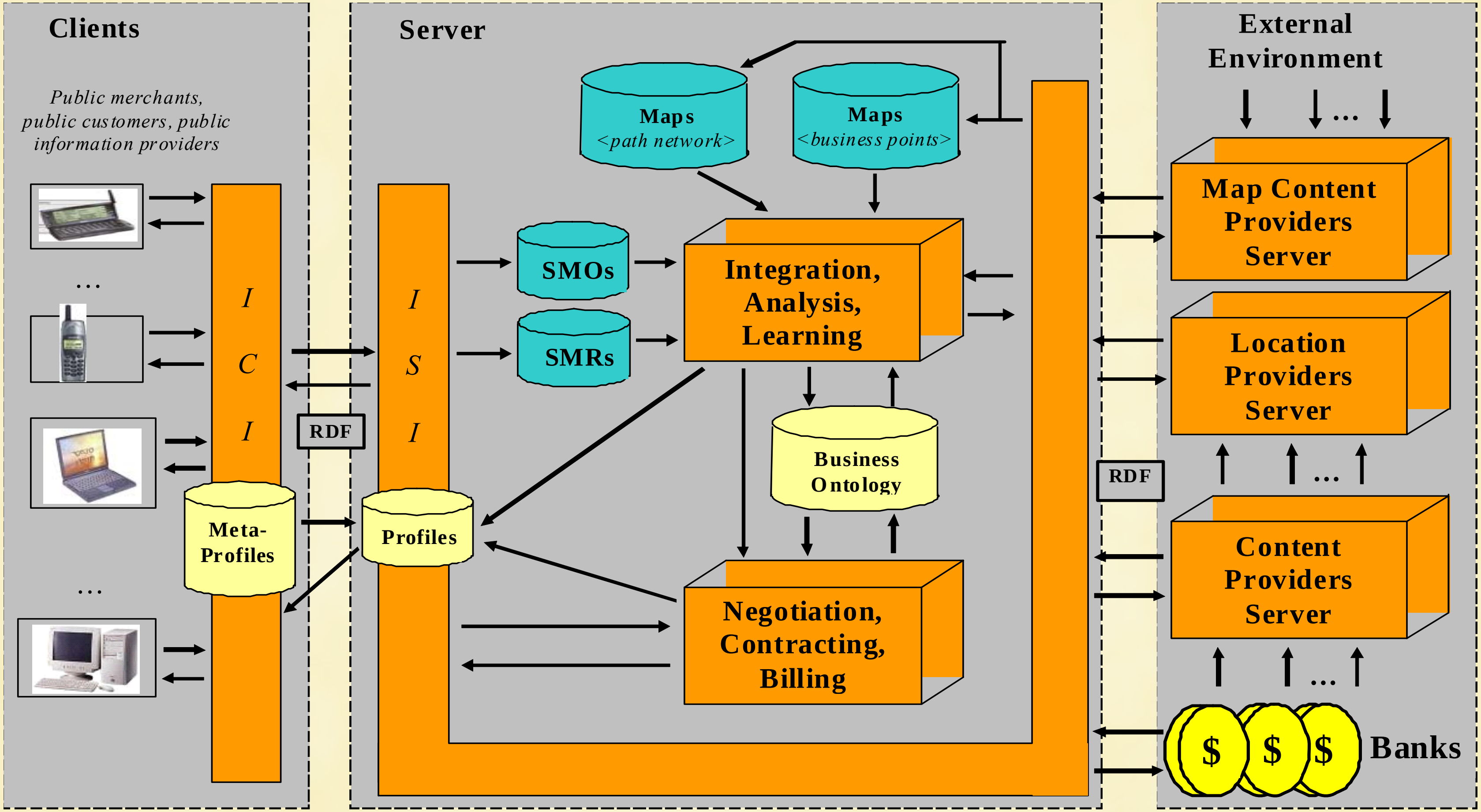
Wine temperature: $\omega_2 = 0.3$

d (“white”, “red”) = 1

d (15°, 25°) = 10°/((+30°)-(+10°)) = 0.5

D (Wine_preference_1, Wine_preference_2) = √ (0.7• 1 + 0.3 • 0.5) ≈ 0.922

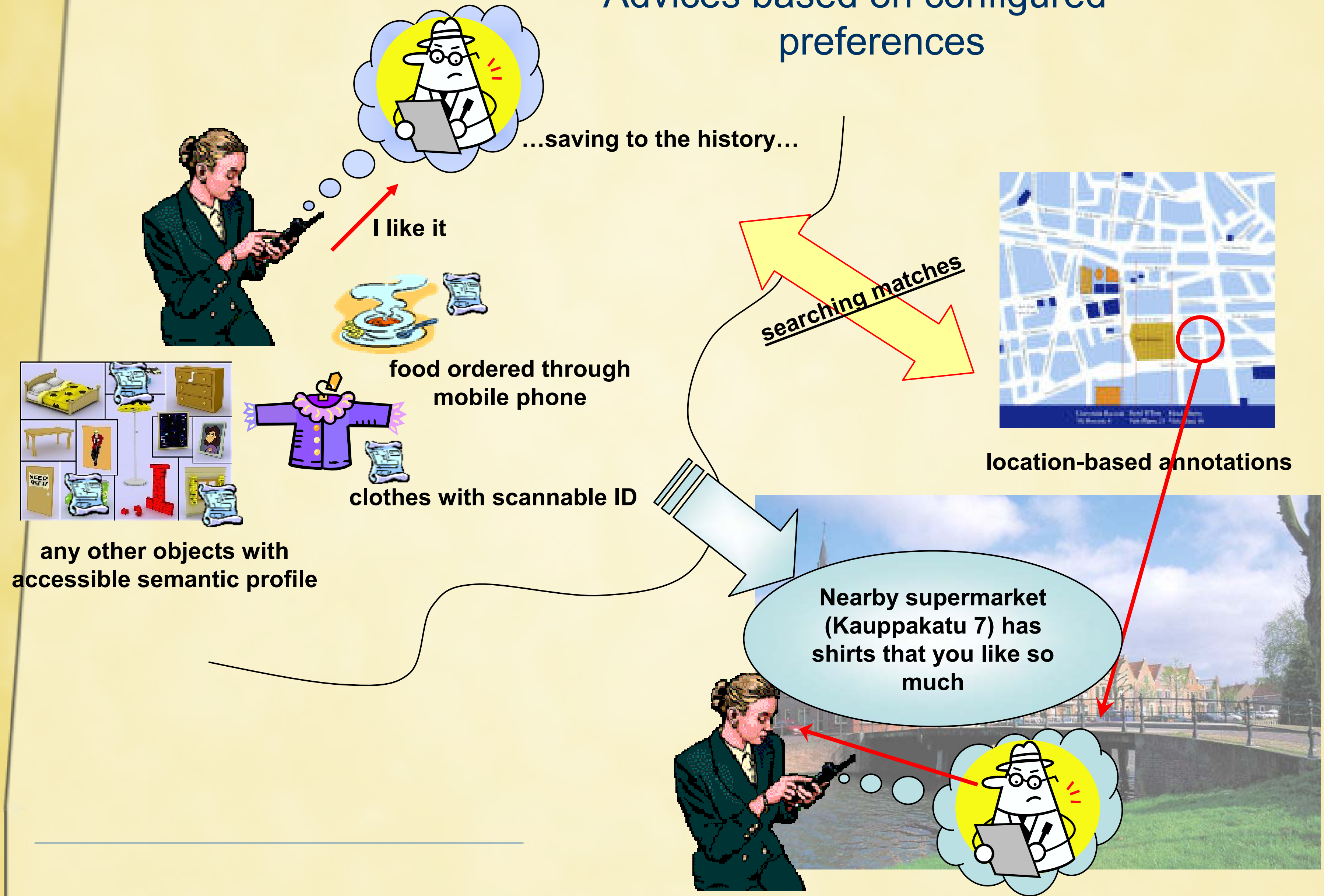
Architecture for a P-Commerce (Public Commerce) Service



Terziyan V., **Architecture for Mobile P-Commerce: Multilevel Profiling Framework**, *IJCAI-2001 International Workshop on "E-Business and the Intelligent Web"*, Seattle, USA, 5 August 2001, 12 pp.

Smart assistant

Advices based on configured preferences



Semantic Enhancement of Games

What is and why Semantic Web Services (SWS) ?

SWS: "Self-contained, self-described, semantically marked-up software resources that can be published, discovered, composed and executed across the Web in a task-driven way ".

S. Arroyo, R. Lara, J. Gomez, D. Berka, Y. Ding and D. Fensel, ***Semantic Aspects of Web Services: Practical Handbook of Internet Computing***, Chapman & Hall and CRC Press, 2004

Proactive SWS: "Self-contained, self-described, semantically marked-up ***proactive*** software resources (***components***) that can be published, discovered, composed and executed across the Web in a task-driven way, ***which behave to increase their utility and are the subject of negotiation and trade***".

Ermolayev V., Keberle N., Plaksin S., Kononenko O., Terziyan V., ***Towards a Framework for Agent-Enabled Semantic Web Service Composition***, *International Journal of Web Service Research*, Idea Group, Vol. 1, No. 3 , 2004, pp. 63-87.

Semantic Web Tools

Jena semantic web framework

protégé

Jess

Cerebra

Racer

Semantic Middleware for Industrial Projects Based on RDF/OWL, a W3C Standard

Joseki

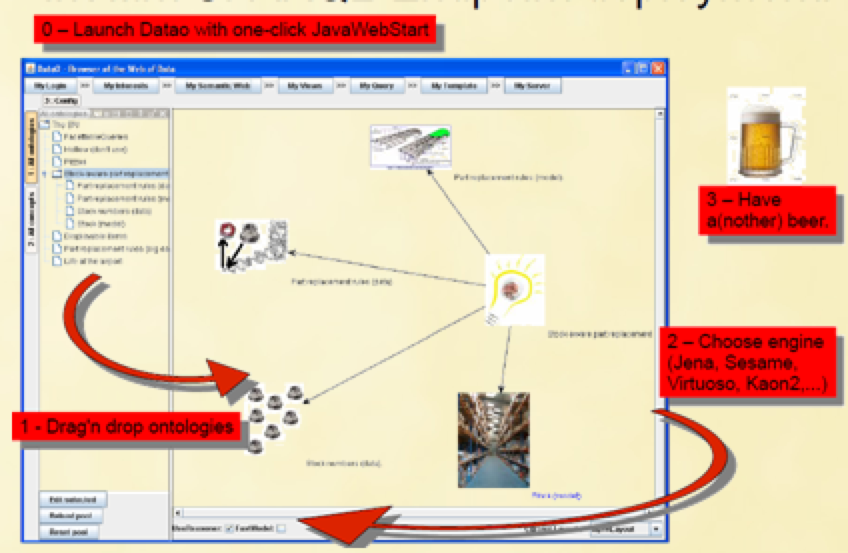
Jena RDF Server

FaCT++

openRDF.org

...home of Sesame

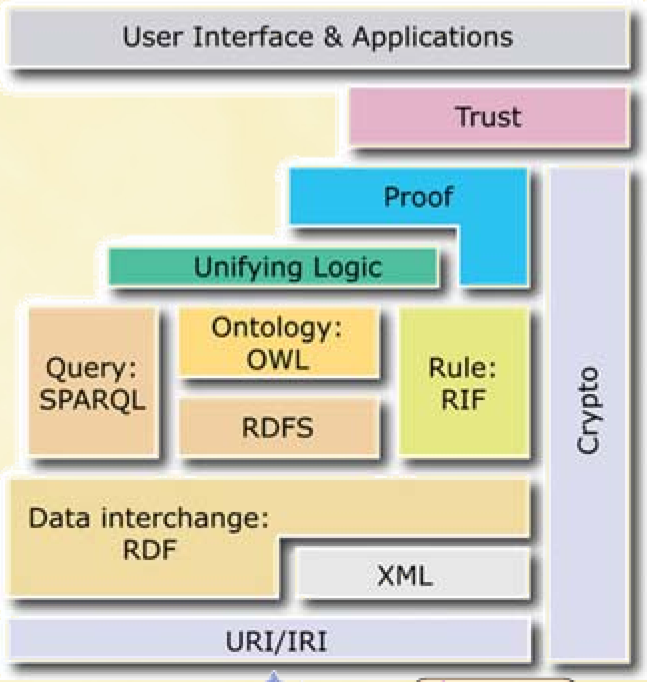
Instant SPARQL-Endpoint deployment.



- Check in:
  - http://www.cs.jyu.fi/ai/vagan/SW_Tools.ppt

Summary: What is Semantic Web ?

- The **Semantic Web** is an evolving development of the World Wide Web in which the meaning (semantics) of information and services published on the Web and their inter-relationships are explicitly defined, making it possible for the Web-based software tools, agents, applications and systems to discover, extract and “understand” Web information resources and capabilities and automatically utilize it.
- **Semantic Technologies** are designed to standardize and support interoperability and integration of information content and capabilities (services) of Web-based systems and components at local and global scale.
- As a ***software technology***, semantic technology encodes meanings separately from data and from application code to enable machines to understand, share and reason with them at execution time.



Why Semantic Web? (Ora Lassila)

A DIFFICULT MESSAGE

- Any <u>specific</u> problem (typically) has a <u>specific</u> solution that does not require Semantic Web technologies
- Q: Why then is the Semantic Web so attractive?
  A: For <u>future-proofing</u>

Semantic Web can be a solution to those problems and situations that we are <u>yet to define</u>

(seriously, I am not kidding...)

NOKIA
Connecting People

- *<u>"Semantic Web is about to reach its full potential and it would be too costly for companies not to invest to it…"</u>*

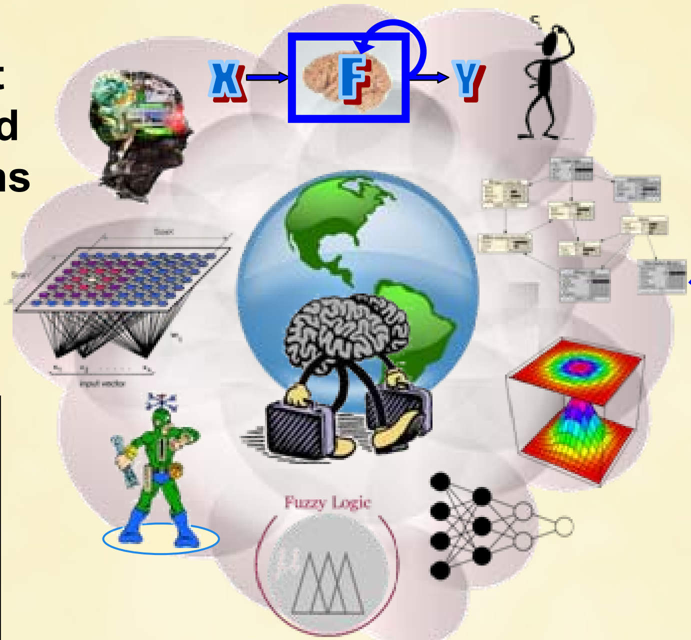
(Ora Lassila, Nokia Research Center (Boston), IASW-2005, Jyvaskyla)

1.7. Web 4.0

Web of Intelligence

Web of Intelligence (Distributed AI, Web 4.0)

Intelligent Agents and Applications

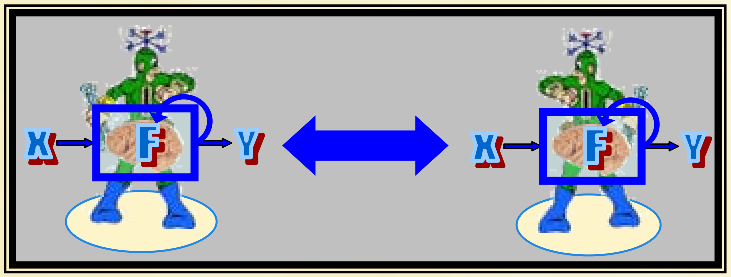


Facilitates Intelligence-to-Intelligence interaction

Web of Intelligence

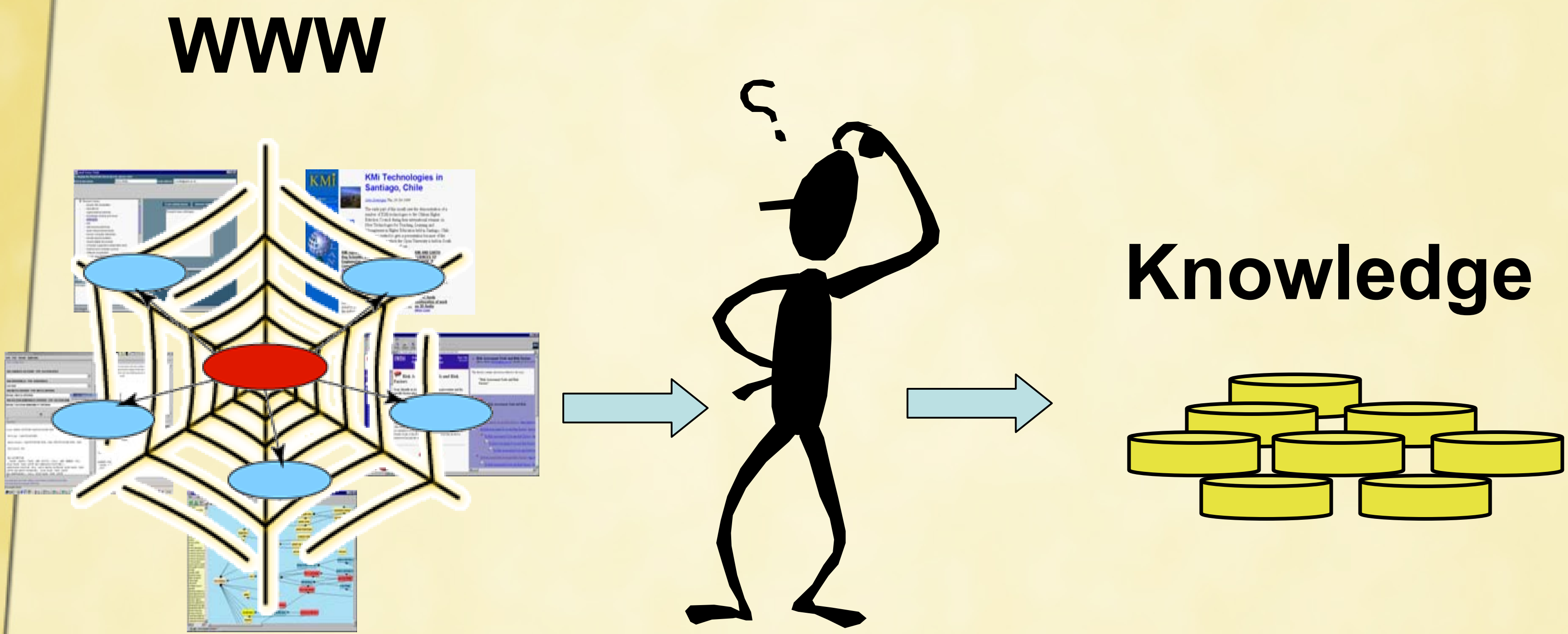
Agents and MAS
Data and Web Mining
Machine Learning
Self-Management
Context-Awareness

Web of intelligent entities (*intelligence* services), browseable, searchable, composable, self-managed, dynamic, mobile …



INTaaS: Intelligence-as-a-Service
INTaaU: Intelligence-as-a-User

Discovering Knowledge from and about WWW - is one of the basic abilities of an intelligent Web-service



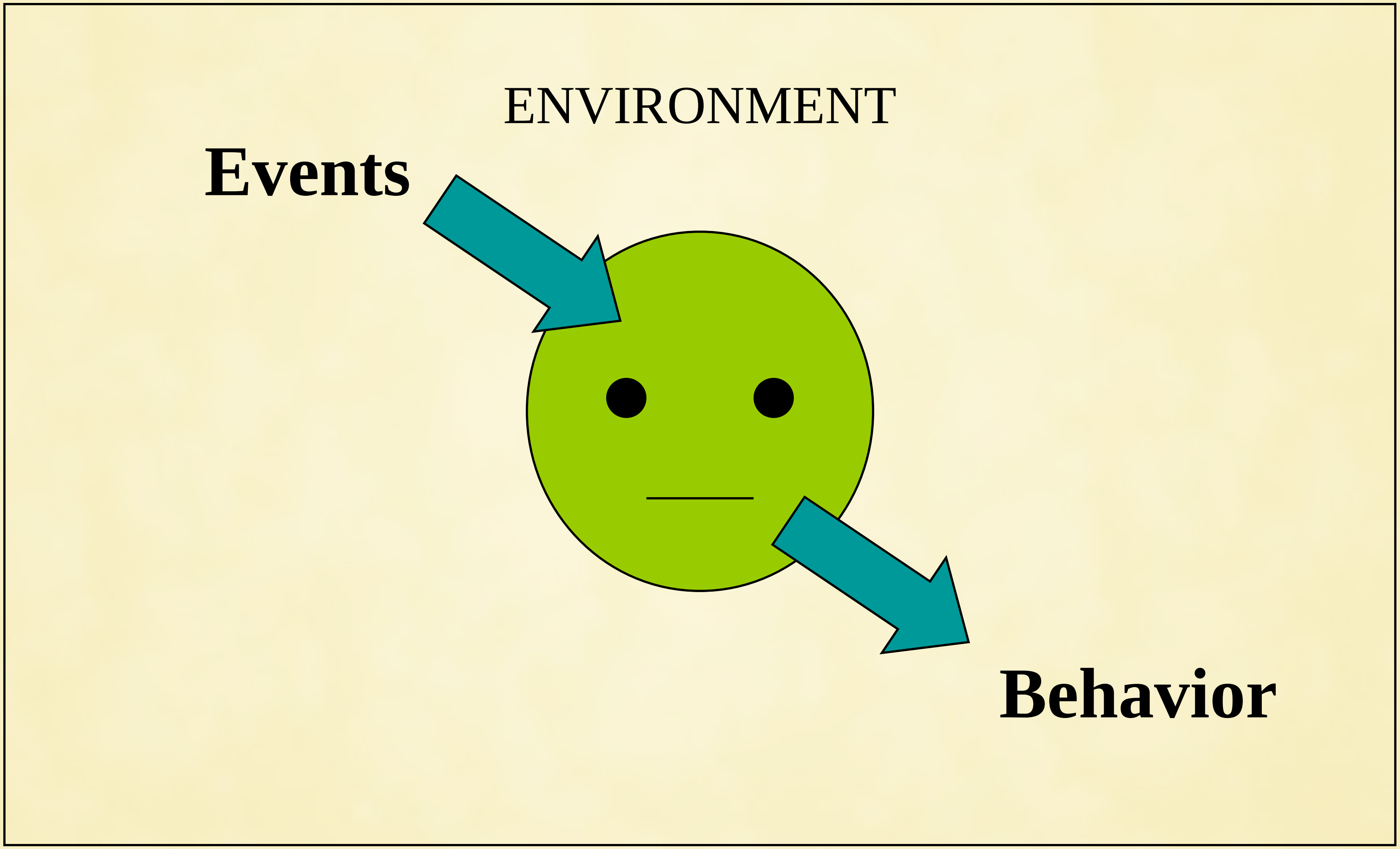
Agent Technology Basics

Agent Definition

- An agent is an entity which is:
  - *Situated* in some *environment*.
  - *Autonomous*, in the sense that it can act without direct intervention from humans or other software processes, and controls over its own actions and internal state.
  - *Flexible* which means:
    - *Responsive (reactive)*: agents should perceive their environment and respond to changes that occur in it;
    - *Proactive*: agents should not simply act in response to their environment, they should be able to exhibit opportunistic, goal-directed behavior and take the initiative when appropriate;
    - *Social*: agents should be able to interact with humans or other artificial agents

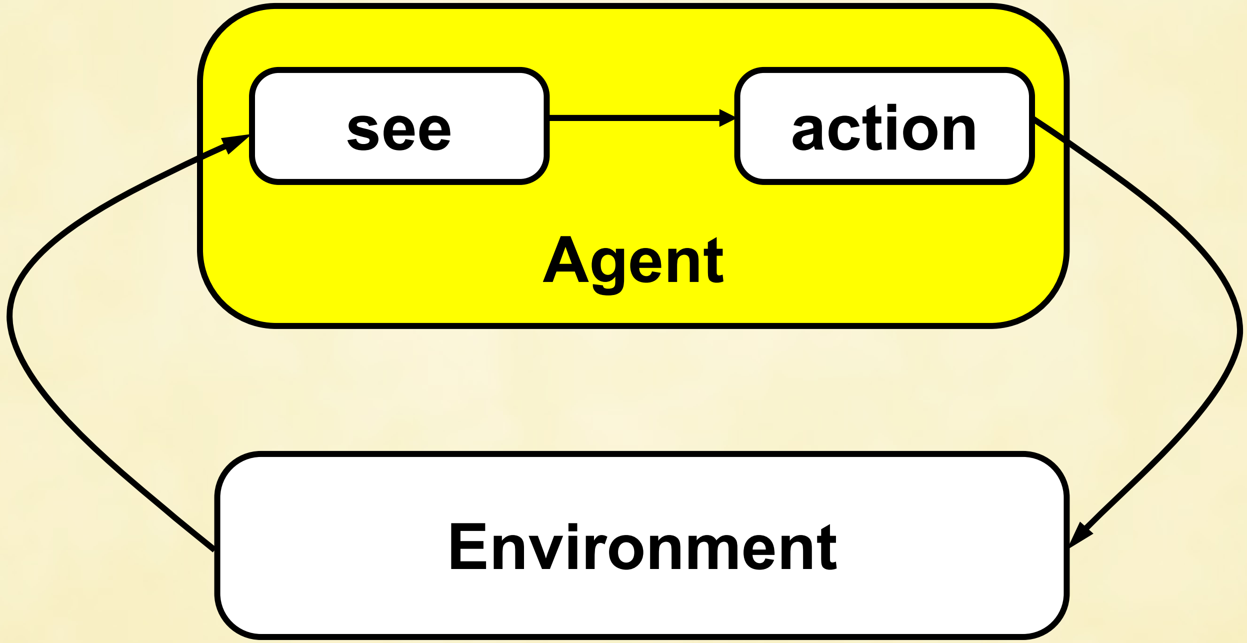
"A Roadmap of agent research and development",
N. Jennings, K. Sycara, M. Wooldridge (1998)

What is an Agent?



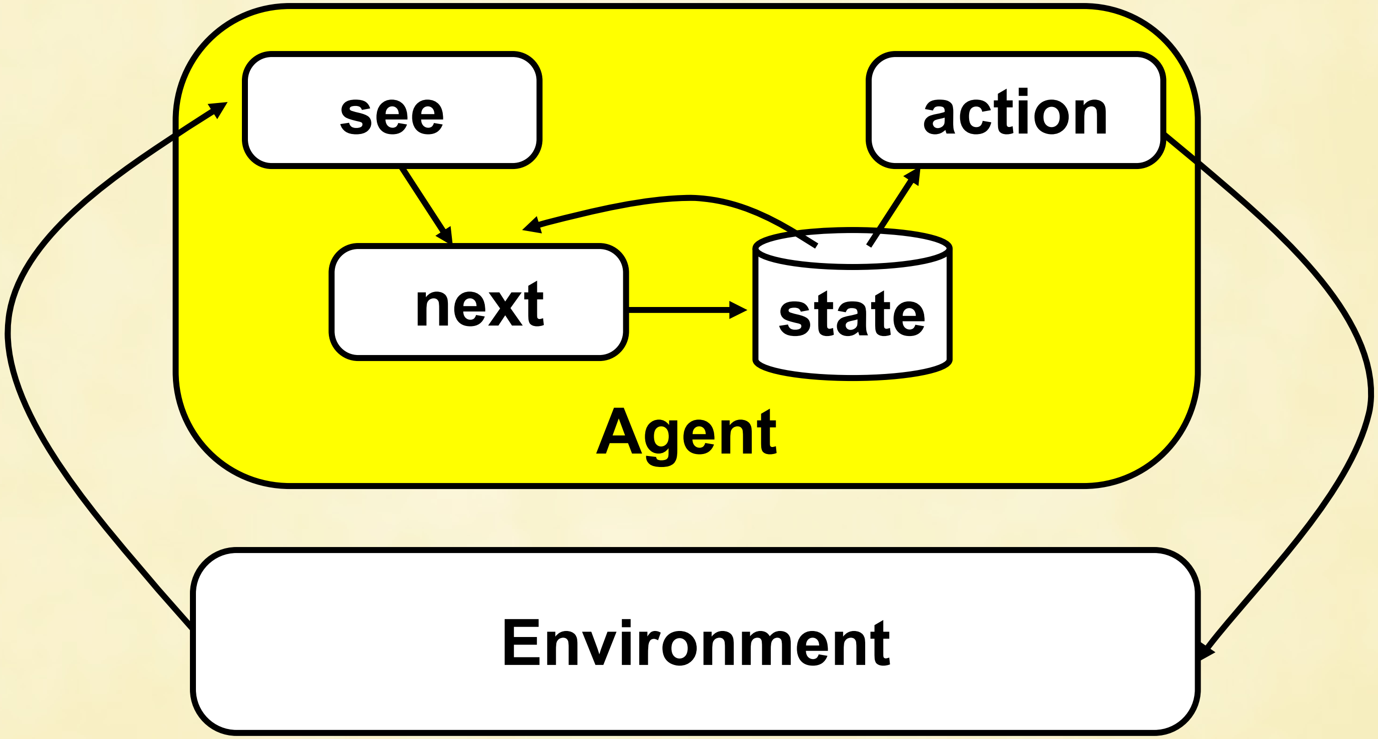
Purely Reactive Agent

- ***see*** and ***action*** functions:



Agent with state

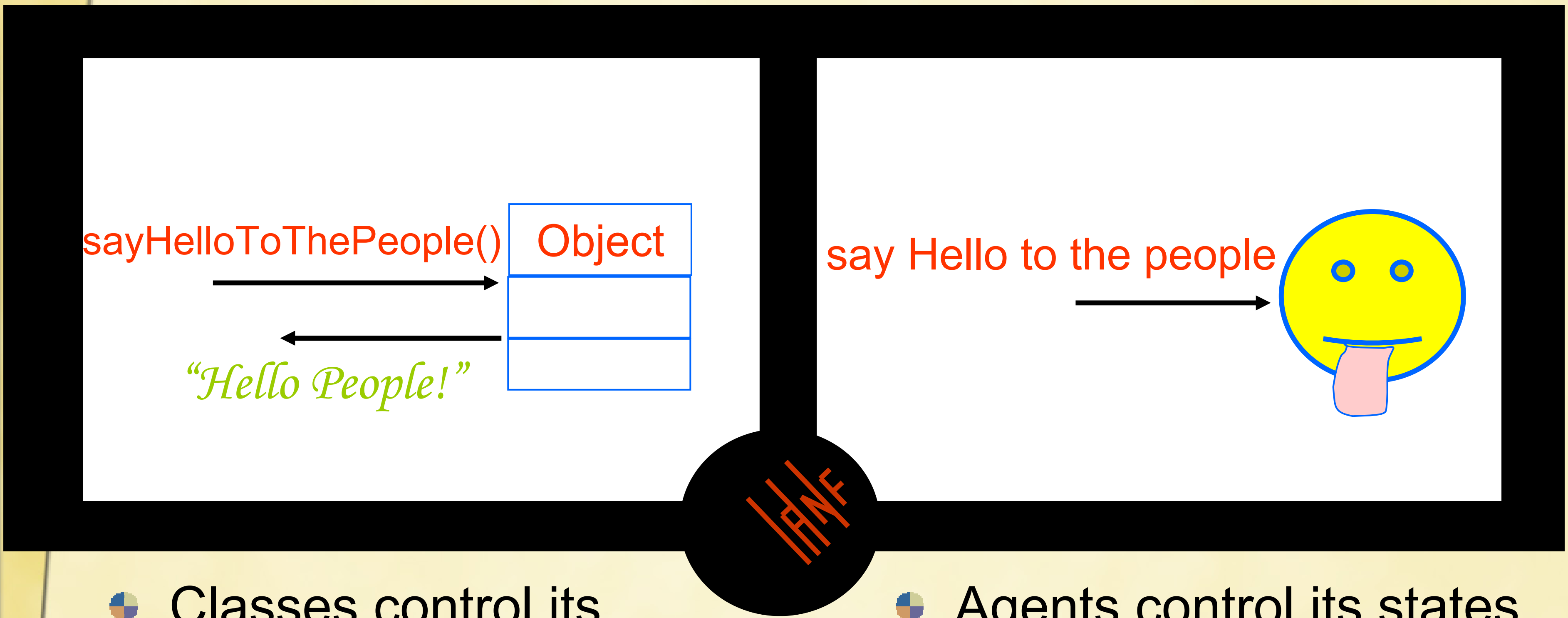
- *see*, *next* and *action* functions



Agents with state

- Behavior:
  - The agent starts in some **internal initial state** i_0
  - Then **observes** its environment state s
  - The **internal state** of the agent is **updated** with $next(i_0, see(s))$
  - The action selected by the agent becomes $action(next(i_0, see(s)))$, and it is performed
  - The agent repeats the cycle observing the environment

Objects & Agents

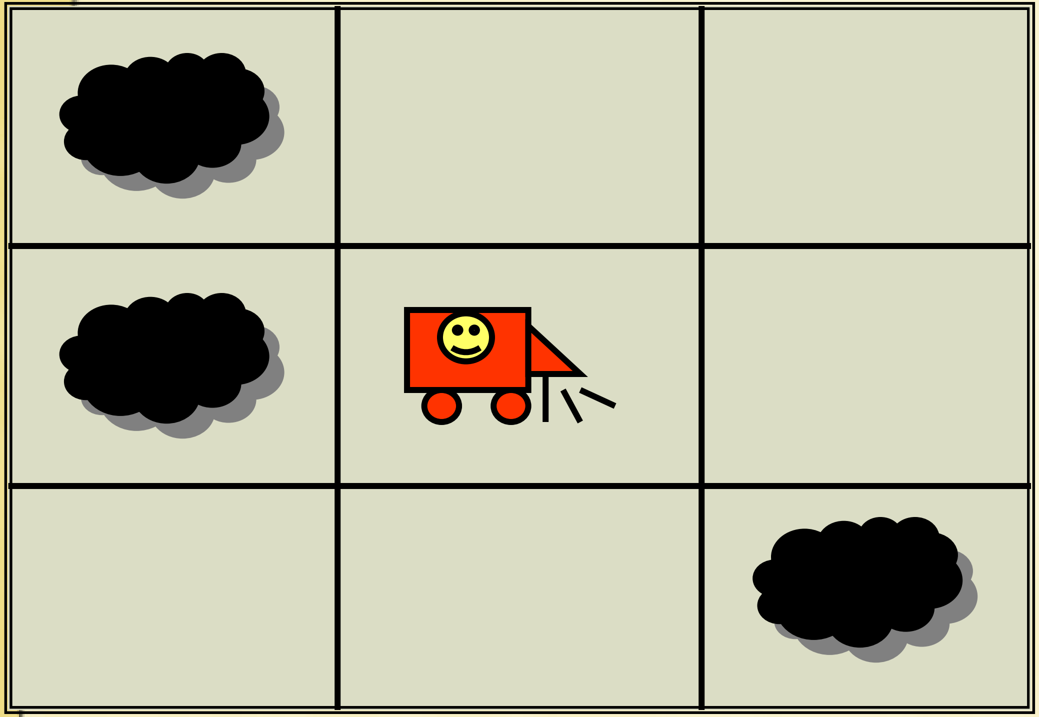


- Classes control its states
- Agents control its states and **behaviors**

"Objects do it for free; agents do it for money"

Logic-based architectures: example

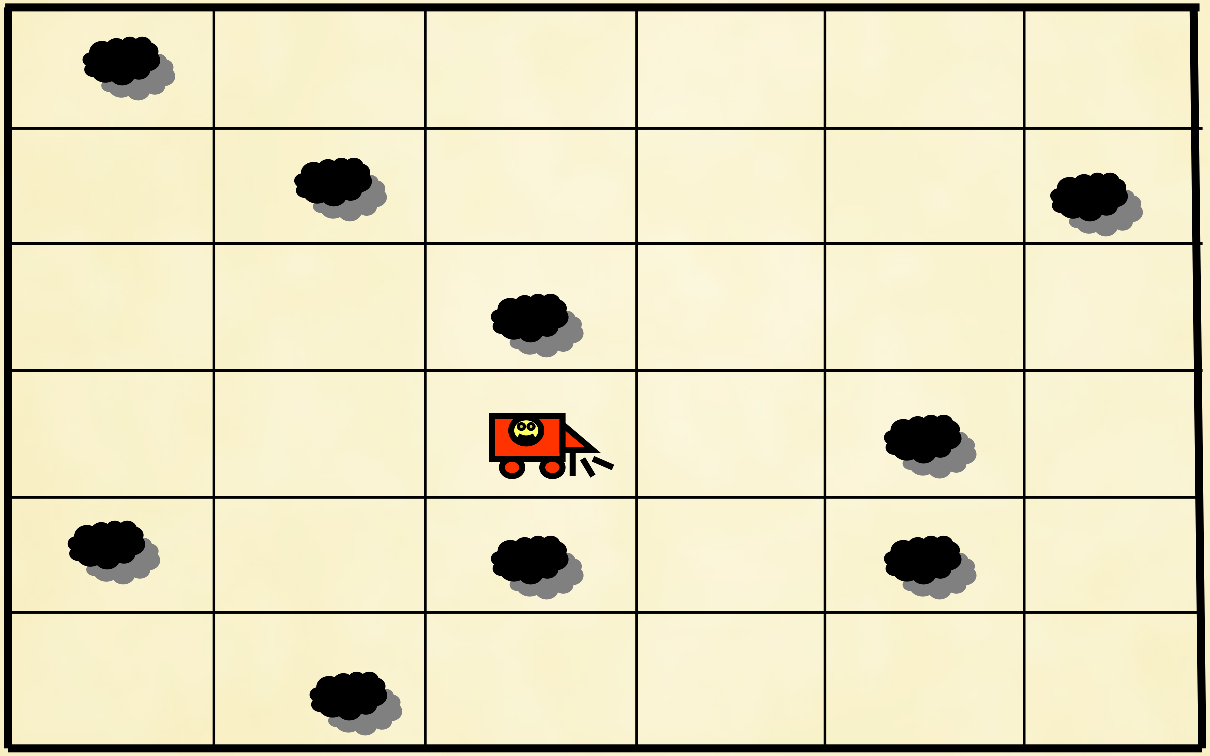
- A cleaning robot



- ***In(x,y)*** *agent is at (x,y)*
- ***Dirt(x,y)*** *there is a dirt at (x,y)*
- ***Facing(d)*** *the agent is facing direction d*
- ***∀x,y (¬ Dirt(x,y)) – goal***
- ***Actions:***
  - ***change_direction***
  - ***move_one_step***
  - ***suck***

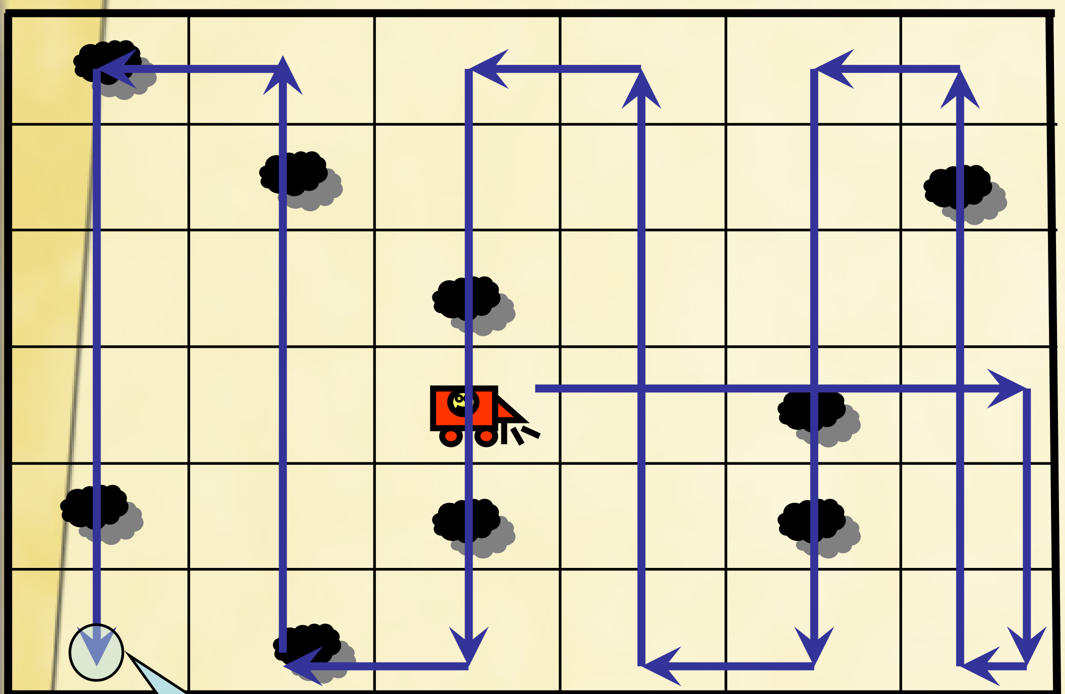
Logic-based architectures: example

- What to do ?



Logic-based architecture: example

- Solution

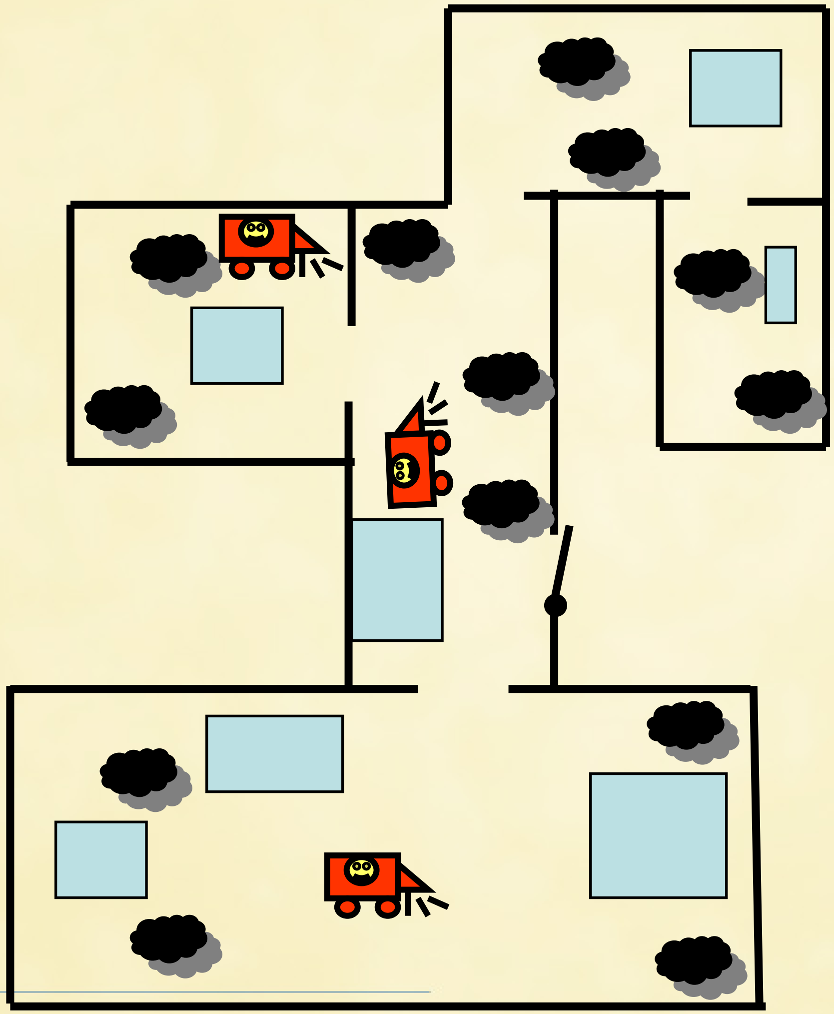


What is stopping criterion ?!

start
// **finding corner**
continue while *fail* *{* **do** *move_one_step}*
do change_direction
continue while *fail* *{***do** *move_one_step}*
do change_direction
finding corner //
// cleaning
continue {
remember *In(x,y)* **to** **Mem**
do change_direction
continue while *fail* *{*
if *Dirt(In(x,y))* **then** *suck*
do *move_one_step* *}*
do change_direction
do change_direction
do change_direction
continue while *fail* *{*
if *Dirt(In(x,y))* **then** *suck*
do *move_one_step* *}*
if *In(x,y)* **equal** **Mem** **then** **stop**
}
cleaning //

Logic-based architecture: example

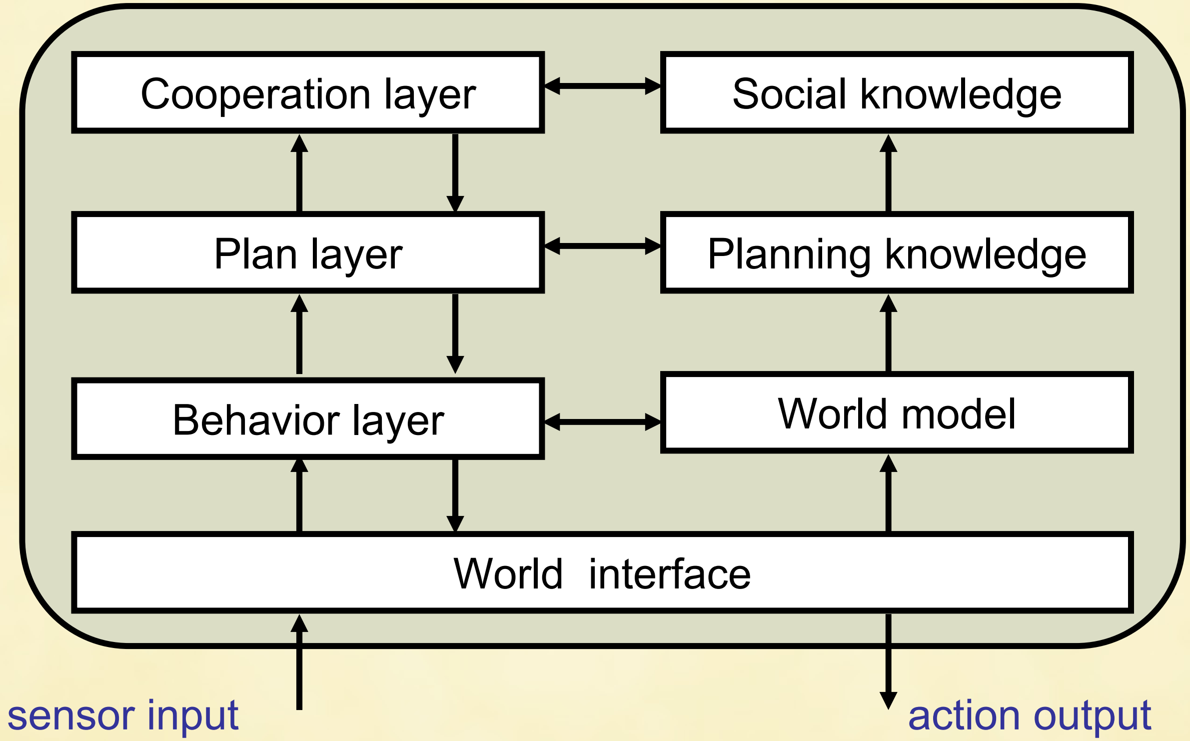
- Now … ??!



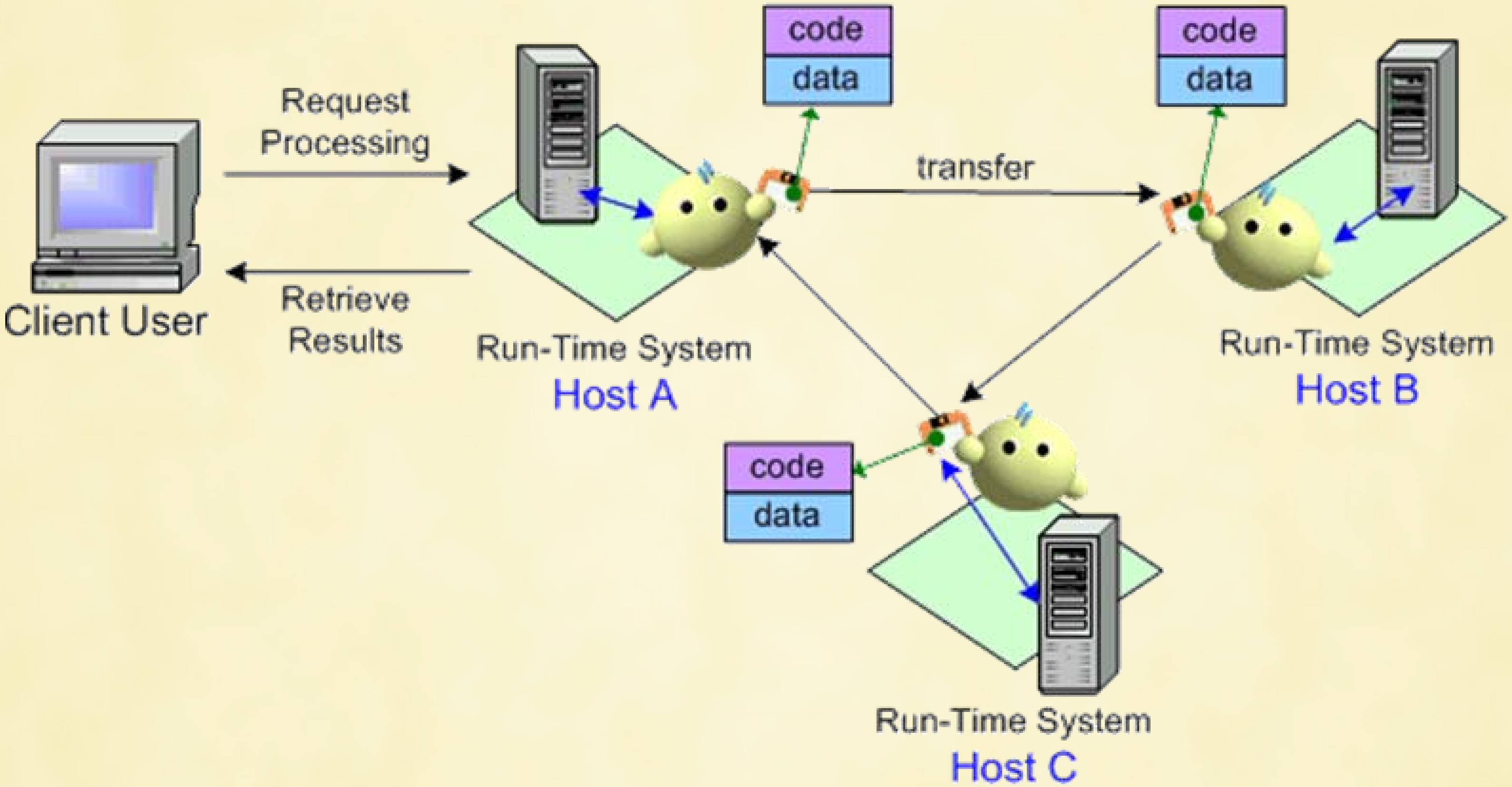
- When you are able to design such a system, this means that you have learned everything you need from the course “Design of Agent-Based Systems”

Layered architectures: INTERRAP

- Proposed by Jörg Müller



Mobile Agent : Conceptual Diagram



What is FIPA?

- The Foundation for Intelligent Physical Agents (FIPA) is a non-profit associations

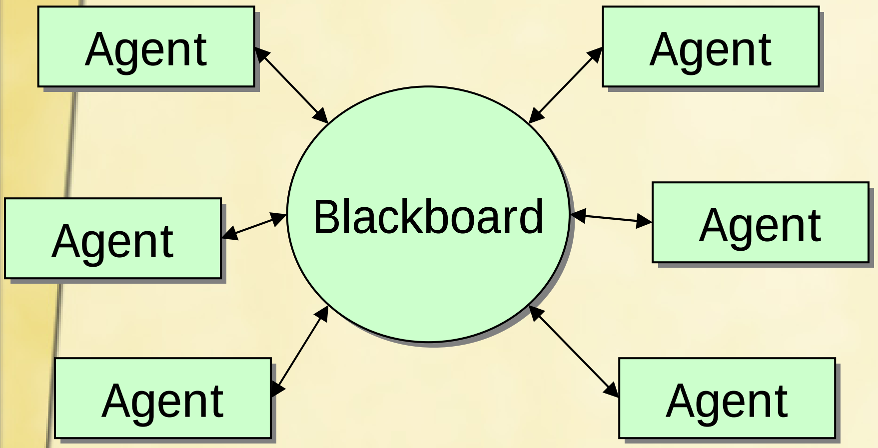
The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications.

- Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found at http://www.fipa.org/.

Agent-to-agent communication

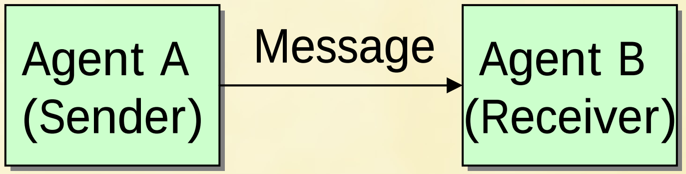
Indirect communication (shared memory)

- information available for all
- no direct communication
- simple architecture



Message passing

- direct exchange
- common language
- conversation - sequences of messages



FIPA ACL

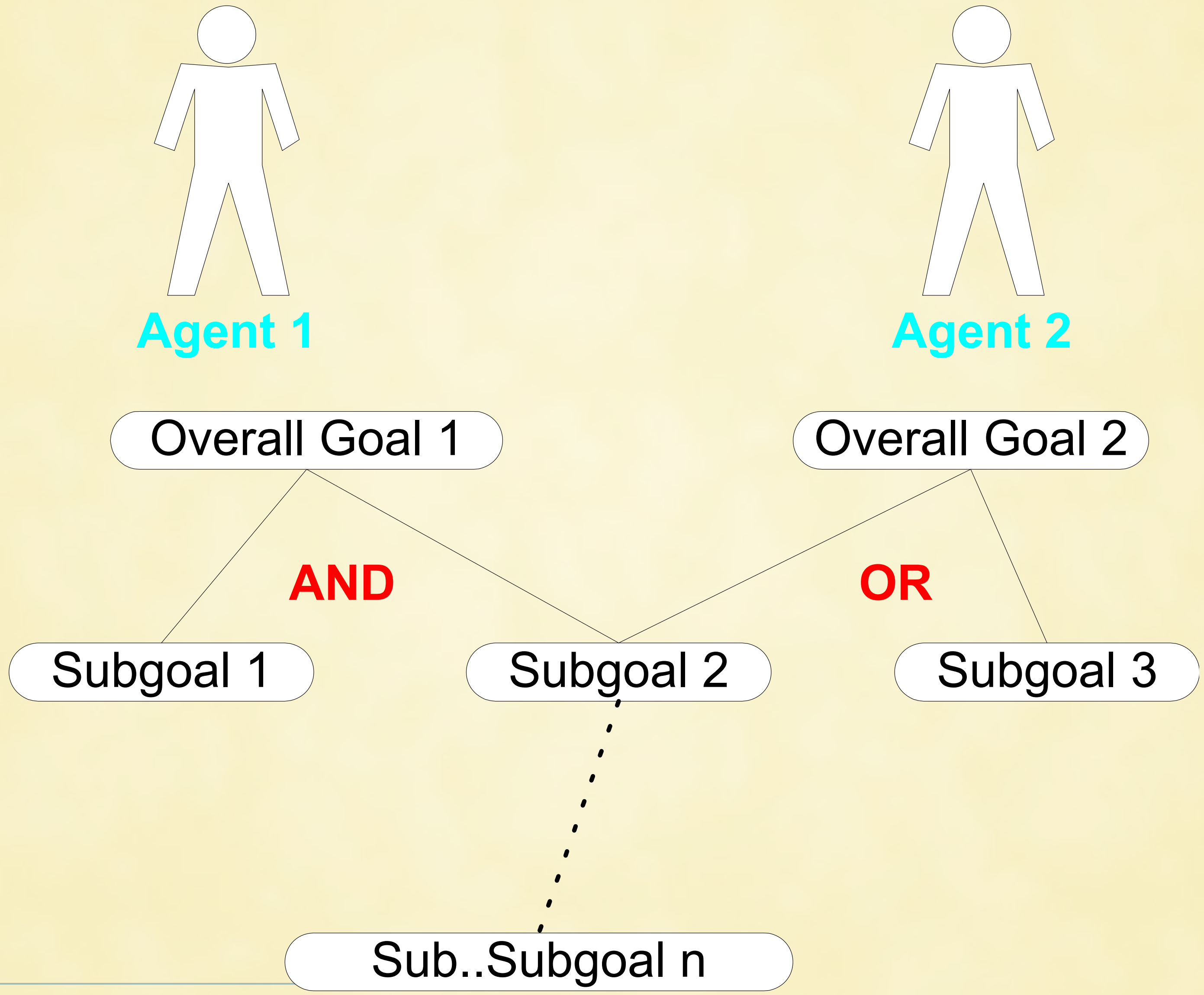
- The standard FIPA Agent Communication Language is FIPA ACL
- FIPA ACL is based on speech acts
- To send and process messages corresponds to perform actions, i.e., communicative acts (**CAs**)
- CAs are described a formal semantics based on modal logic

FIPA CAs Examples

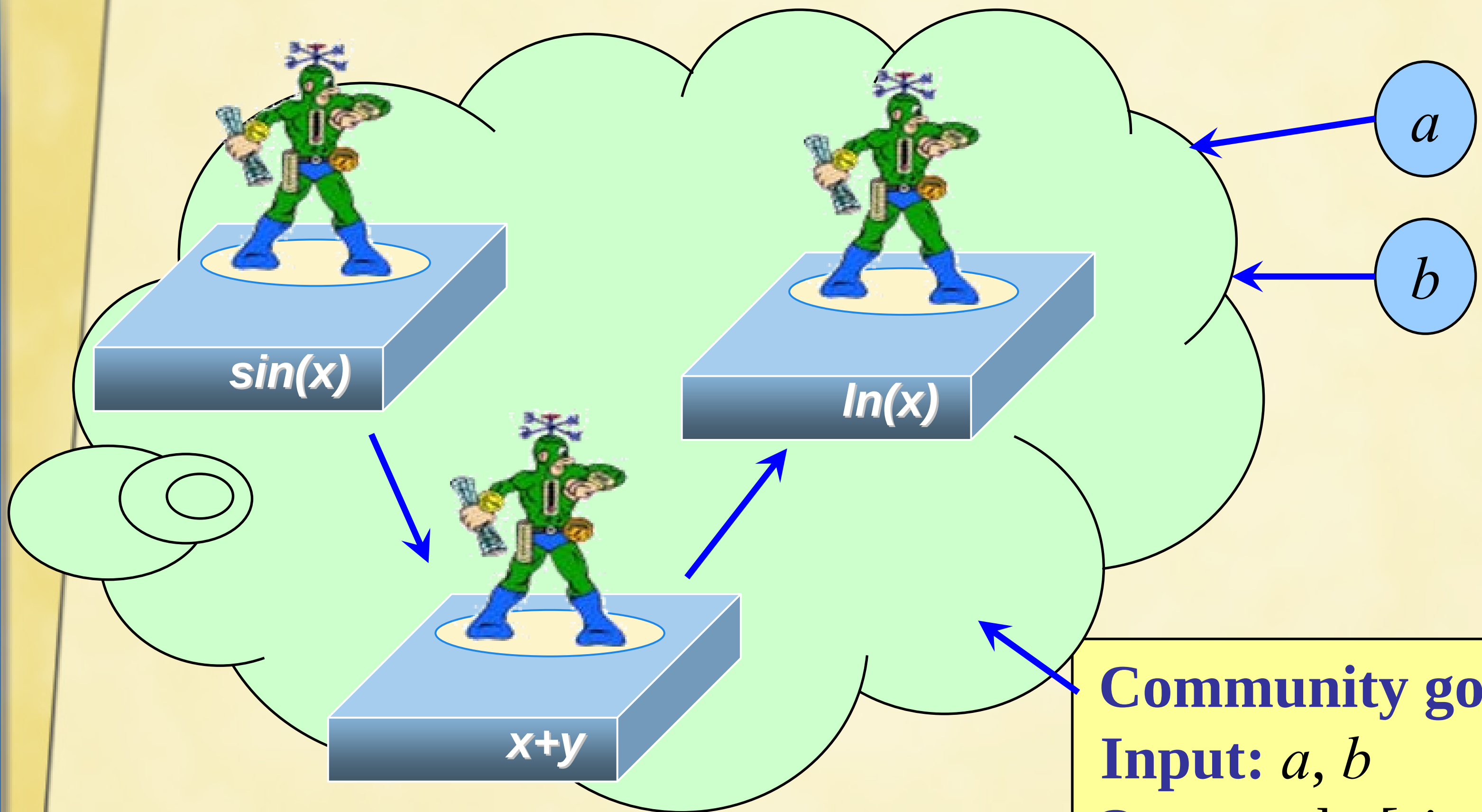
- The door is open?
- Open the door (for me)
- OK! I’ll open the door
- The door is open
- I am unable to open the door
- I will not open the door
- Say when the door becomes open
- Anyone want to open the door?
- I can open the door for you…at a price
- Door? What’s that? Don’t understand...

- query
- request
- agree
- inform
- failure
- refuse
- subscribe
- cfp
- propose
- not-understood

Agent Coordination



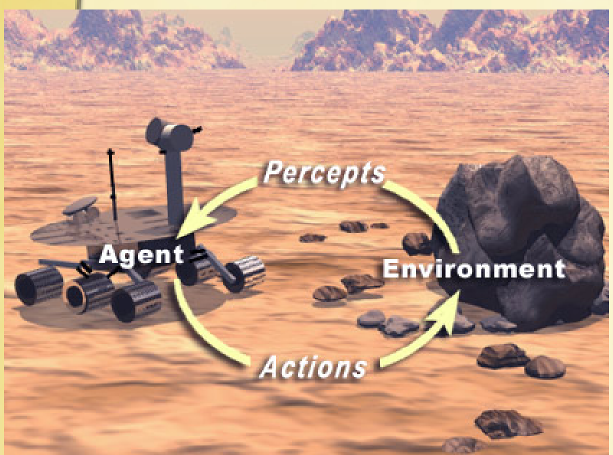
Coordination for reaching a community goal

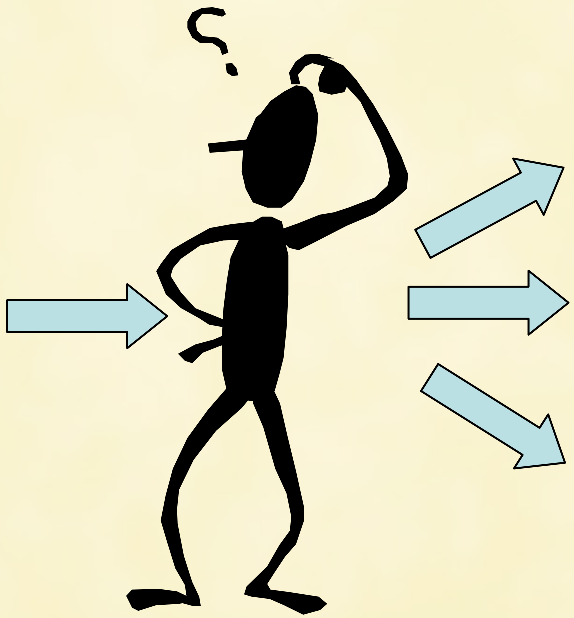


Community goal:
Input: a, b
Output: $\ln[\sin(a) + \sin(b)]$

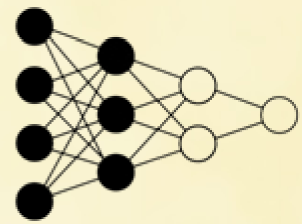
Intelligent perception of the external environment, mining data and discovering knowledge about it, reasoning new facts about it, planning own behavior within it and acting based on plans - are among the basic abilities of an intelligent agent

Agent
Environment





Knowledge and facts



Plans

Behavior

Java Agent Development Environment (JADE)

- Jade is a platform for running agents; it supports:
  - An asynchronous agent programming model
  - Communication between agents either on the same or different platforms
  - Mobility, security, and other utilities

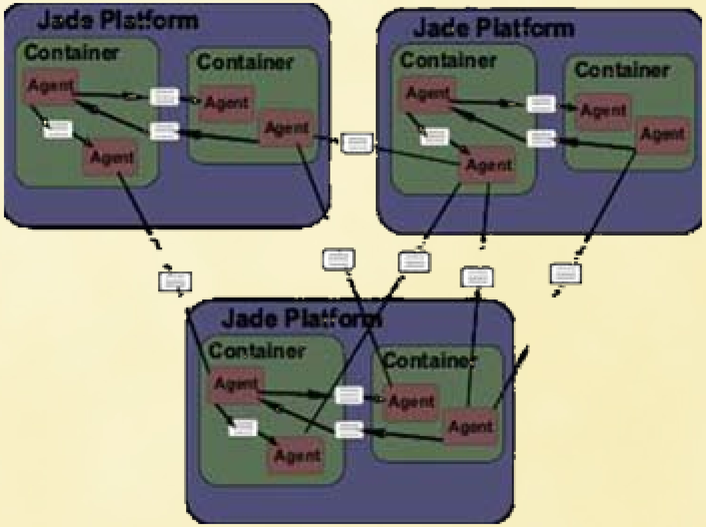
http://jade.tilab.com/

- Container: a running instance of the JADE running environment containing several agents
  - A single Main Container must always be active in a platform and all other containers register with it as soon as they start
  - You do not have to know how the JADE runtime environment works, but just need to start it before executing your agents

The latest version of JADE is JADE 3.7 released on 2nd July 2009.

Platform Services

- Implemented as agents
- AMS: Agent Management Service
  - "White Pages"
  - Maintains set of agents on a platform
- DF: Directory Facilitator
  - "Yellow Pages"
  - Provides a service directory
  - Maps service descriptions to Agent Identifiers
  - Agents can add/modify/delete entries for themselves



Agent Programming Languages

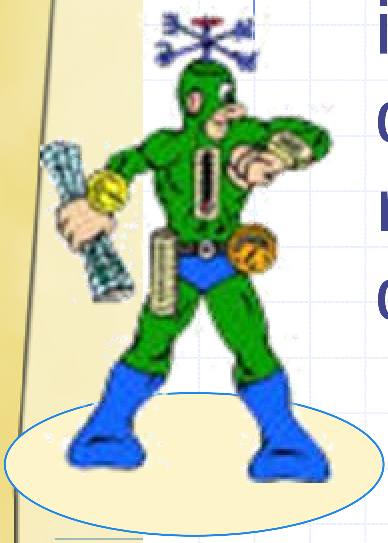
- Examples:
  - AgentSpeak(L)
  - 3APL
  - AFAPL from AgentFactory framework
  - Semantic Agent Programming Language (S-APL)

- All of those are declarative languages and based on the first-order logic of n-ary predicates (Prolog-like).

- For example, in AFAPL (similarly in S-APL):
  - An AFAPL agent program consists of declarations of the *beliefs* and *goals* of that agent and declaration of a set of *rules*, including belief rules (generating new beliefs based on existing ones), reactive rules (invoking some actions immediately), and commitment rules (adopting a commitment to invoke an action).
  - Perceptors (perceiving environment and generating new beliefs) and actuators (implementing the actions to be invoked) are then pieces of external code, in Java.

Summary: What is Agent ?

Intelligent Agents

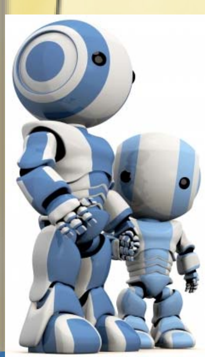


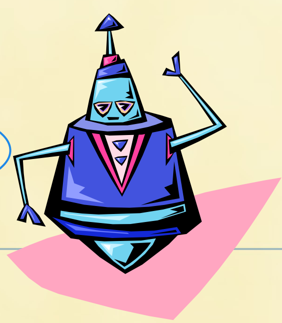
- Software entities that carry out some set of operations on behalf of a user or another program with some degree of independence or autonomy, and in so doing employ some knowledge or representation of a user's goals or desires.

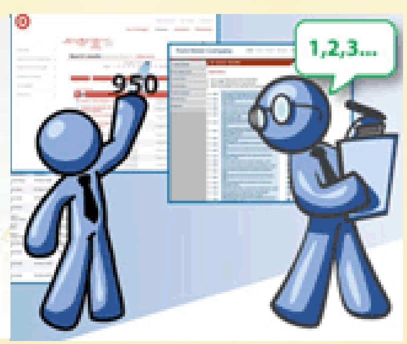
IBM, Intelligent Agent Definition

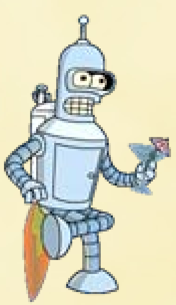
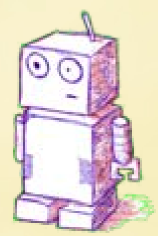
Summary: Why Agents ?

- Growing complexity of computer systems and networks
- Distributed nature of systems (data, software, users, etc.)
- Ubiquitous computing, “Internet of Things” scalability challenges
- Need for self-manageability of a complex system
- Need for new software development paradigms in designing distributed systems
- Agent-based approach meets the above challenges








References

Basic Literature:

Software Agents, Edited by Jeff M. Bradshaw. AAAI Press/The MIT Press.

Agent Technology, Edited by N. Jennings and M. Wooldridge, Springer.

The Design of Intelligent Agents, Jorg P. Muller, Springer.

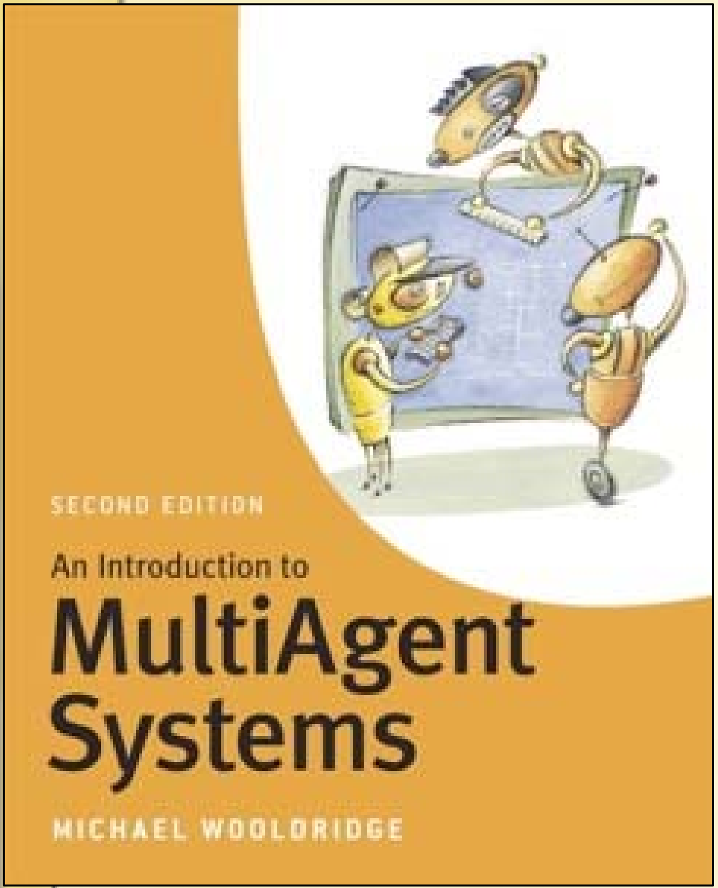
Heterogeneous Agent Systems, V.S. Subrahmanian, P. Bonatti et al., MIT Press.

Papers‘ collections: ICMAS, Autonomous Agents (AA), AAAI, IJCAI.

Links:

- www.fipa.org
- www.agentlink.org
- www.umbc.edu
- www.agentcities.org

Fresh Recommended Literature



An Introduction to MultiAgent Systems - *Second Edition*

by Michael Wooldridge

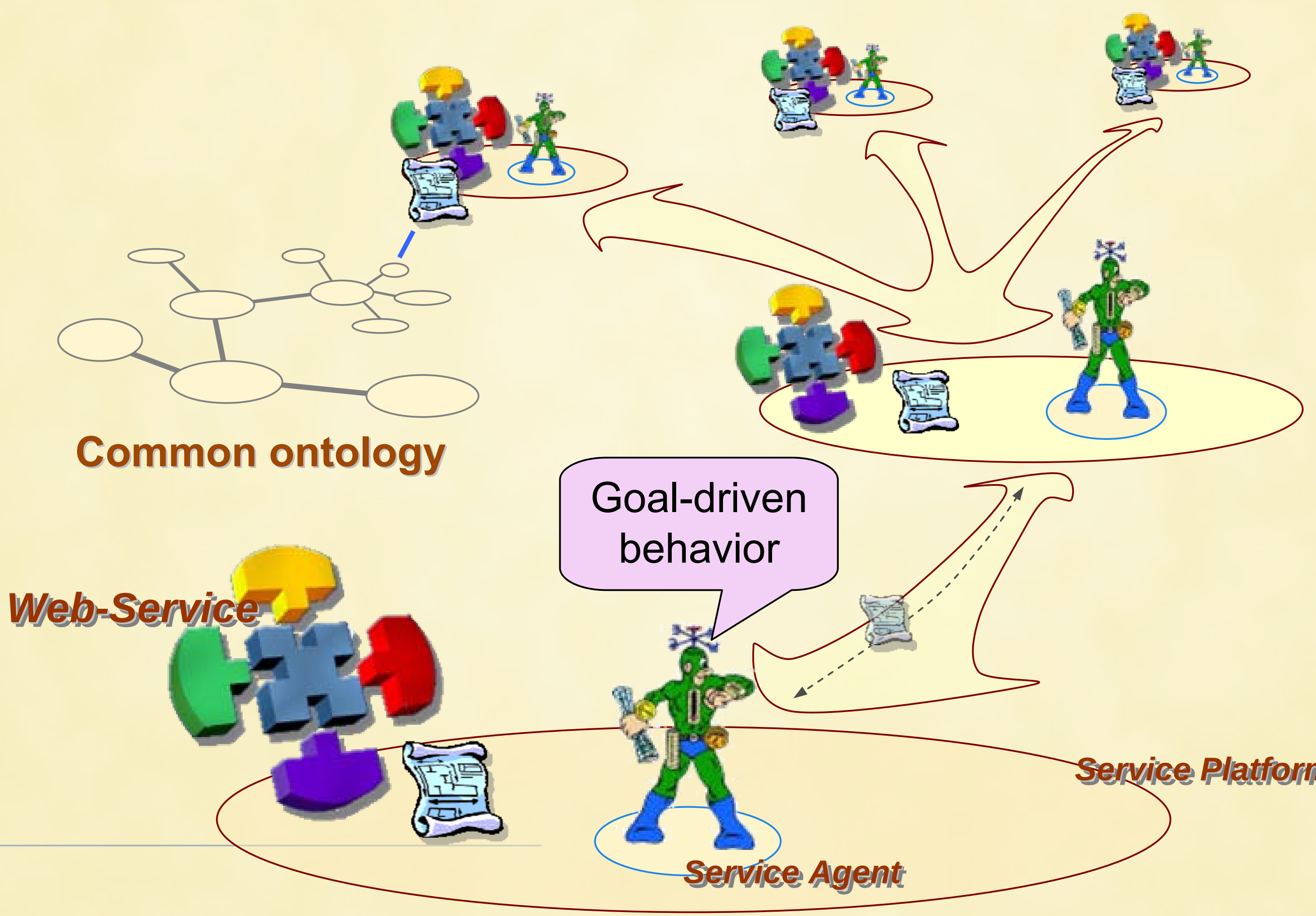
Published May 2009

by John Wiley & Sons

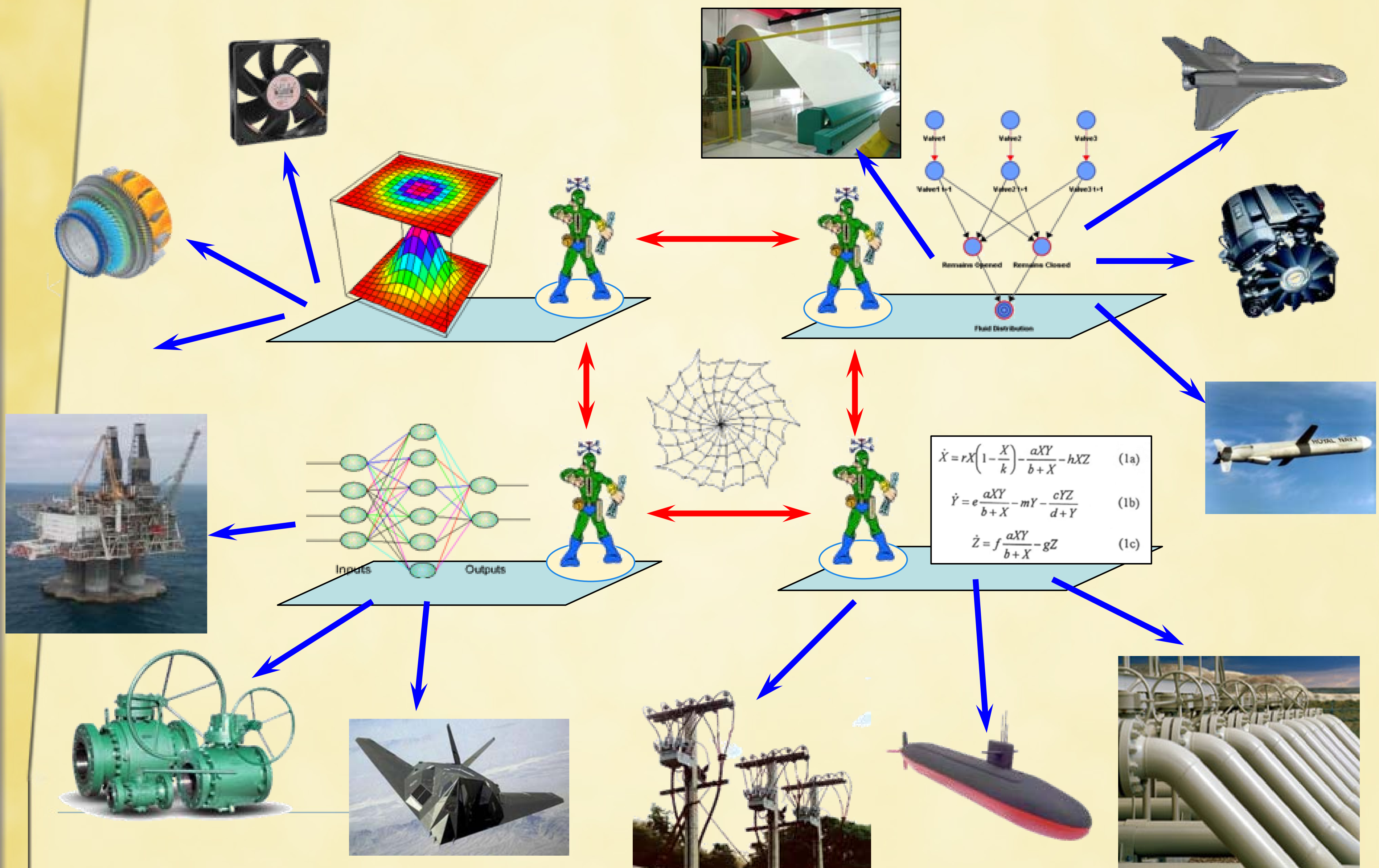
ISBN-10: 0470519460
ISBN-13: 978-0470519462

Handouts available in: http://www.csc.liv.ac.uk/~mjw/pubs/imas/agents.tar.gz

Proactive Web-Services: adding an agent to service platform – allows agent-based S2S communication

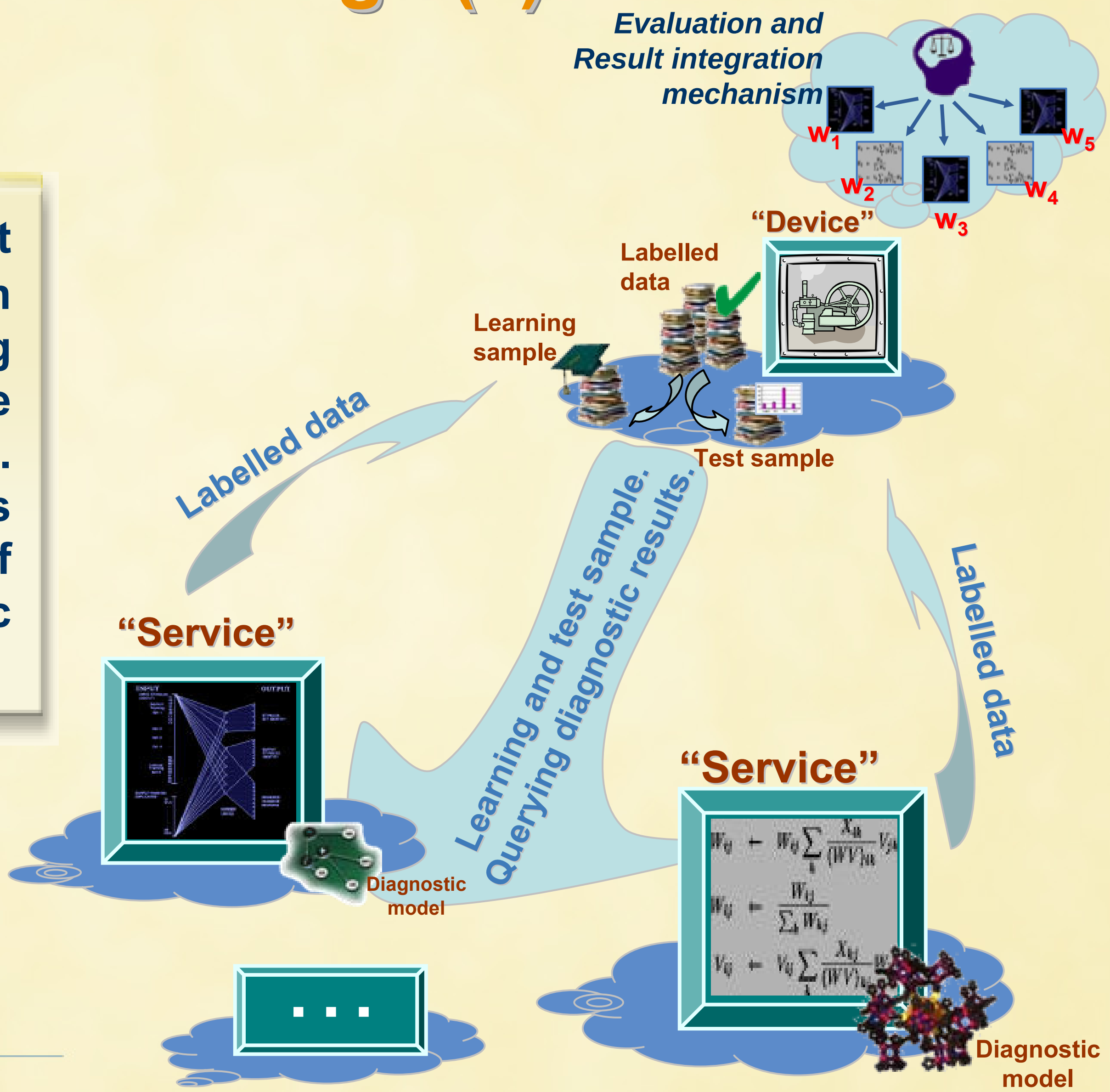


UBIMATH: Network of Proactive Mathematical Models in the Web



Intelligent Services for the Web of Things (1)

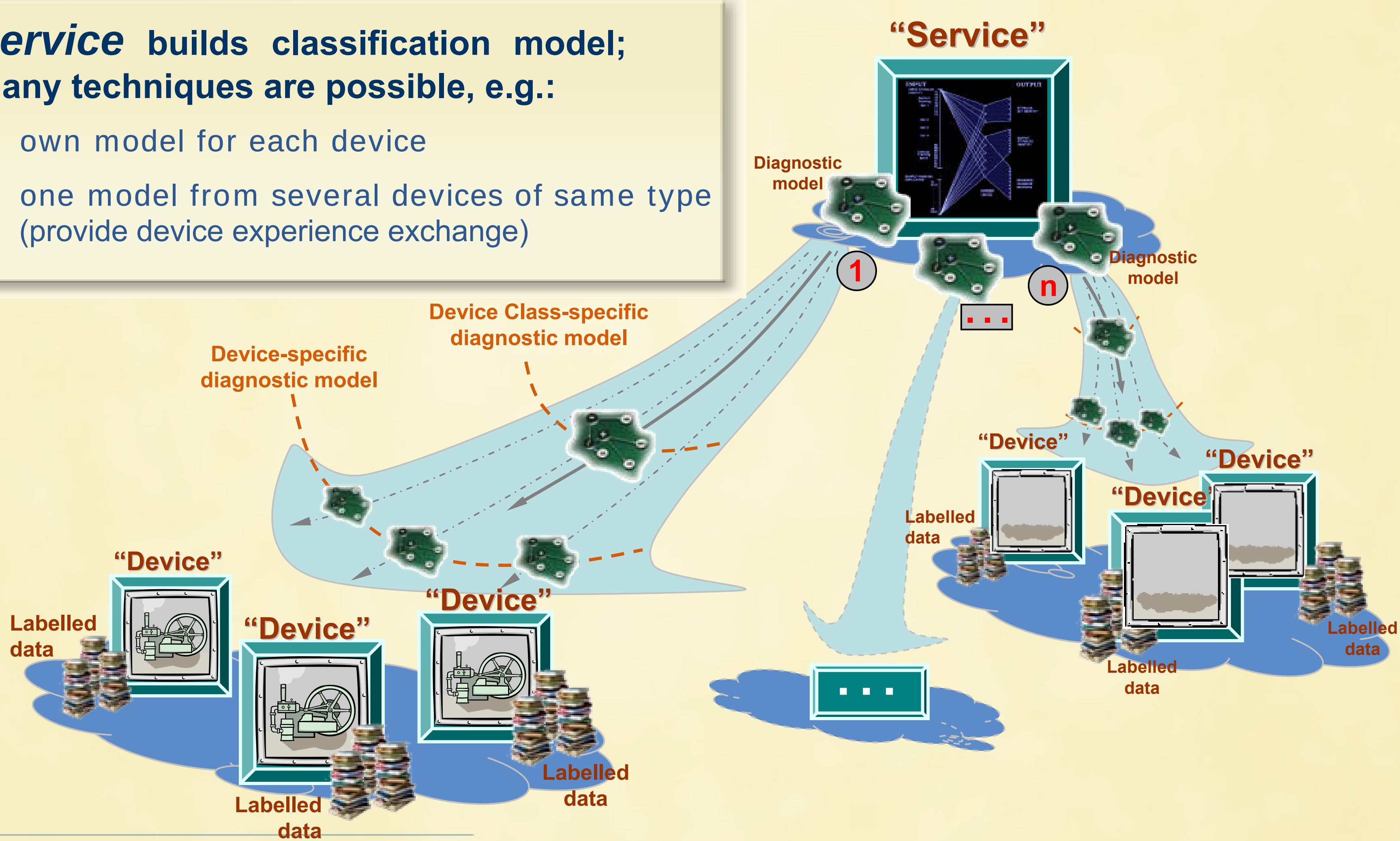
Device **will support service composition in form of ensembles using own models of service quality estimation. Service composition is made with goal of increasing diagnostic performance.**



Intelligent Services for the Web of Things (2)

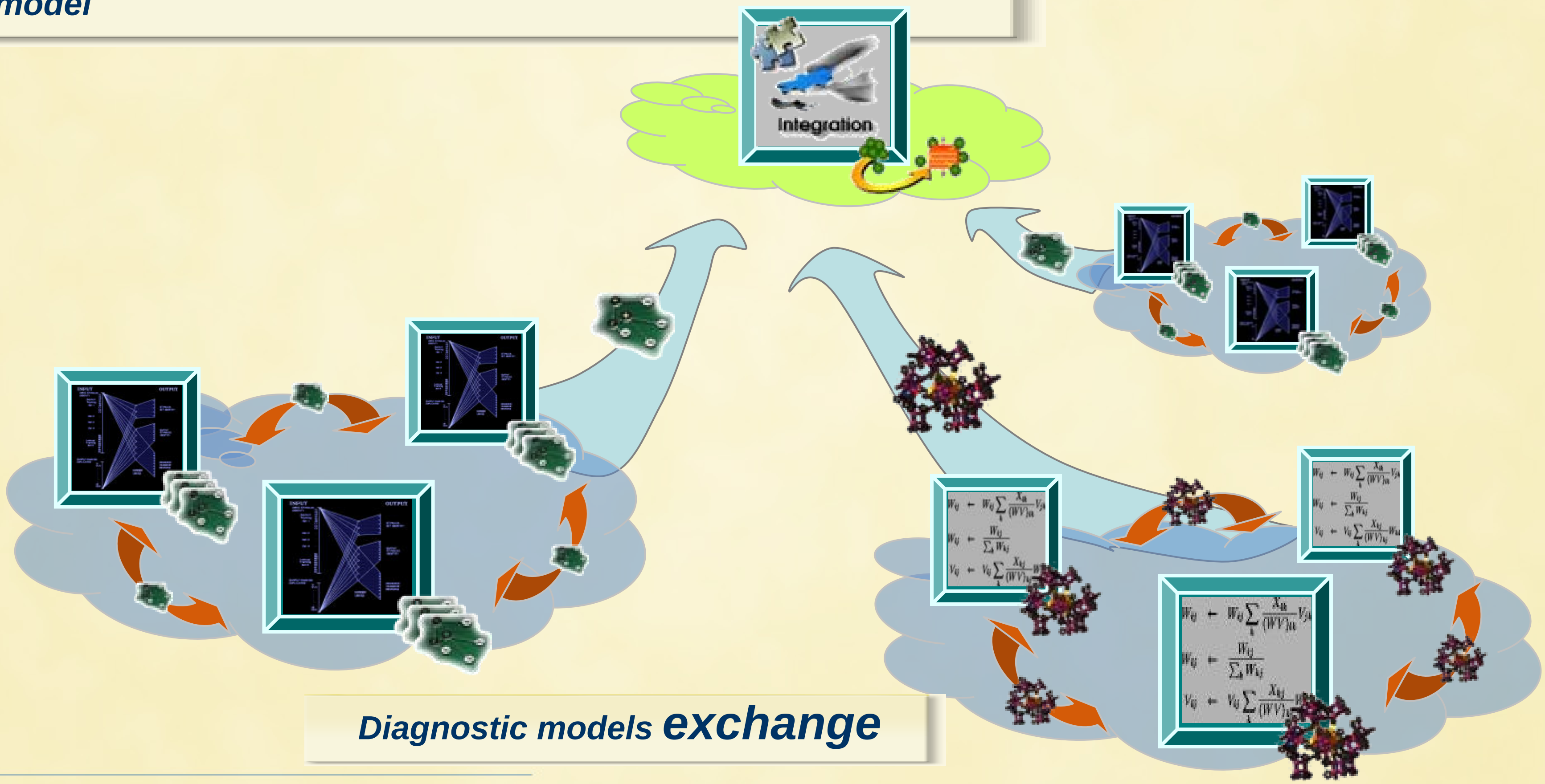
Service **builds classification model; many techniques are possible, e.g.:**

- own model for each device
- one model from several devices of same type (provide device experience exchange)



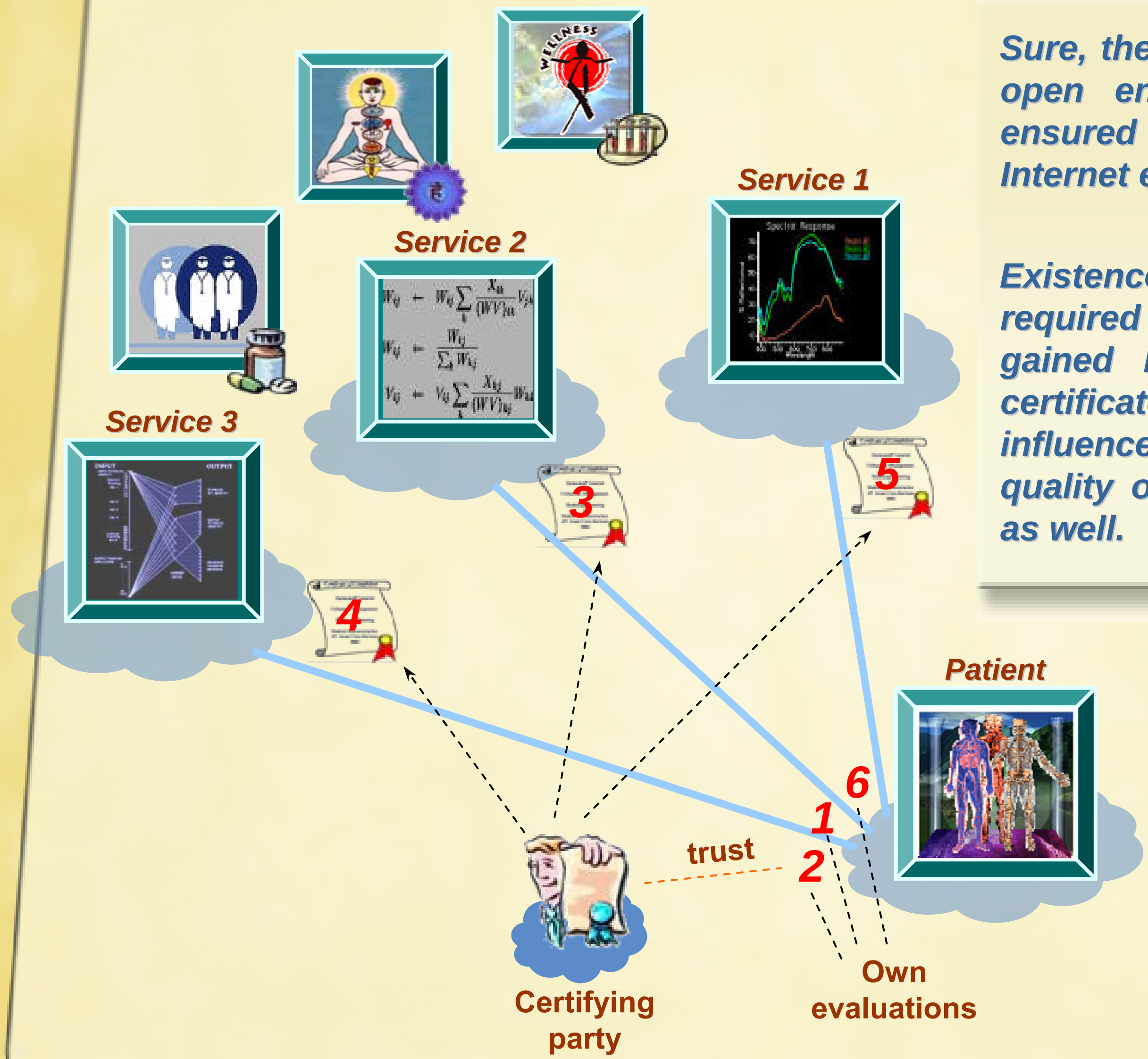
Intelligent Service-to-Service "model" exchange and integration

Diagnostic models* integration *entails creation of a more complex model extension or a service with new diagnostic model



***Diagnostic models* exchange**

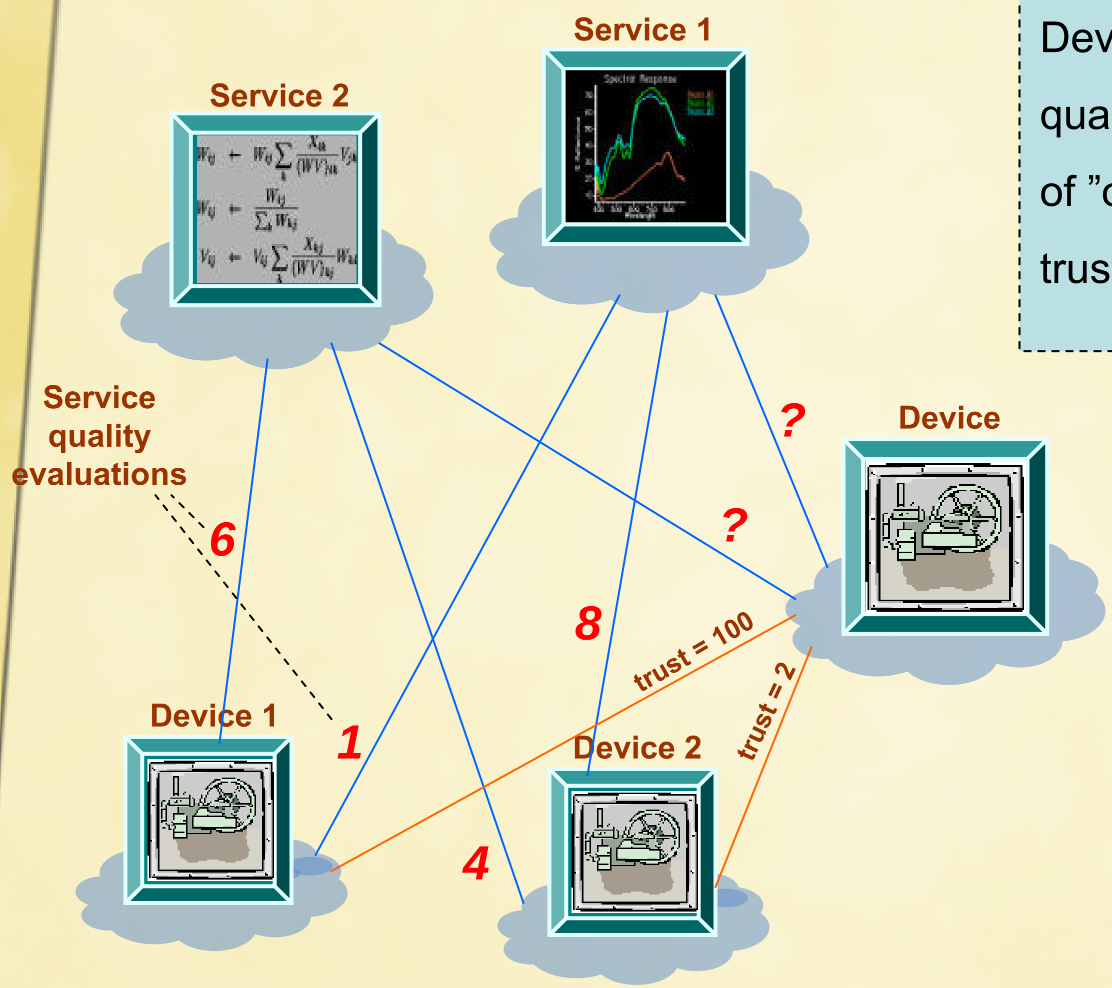
Intelligent Service Certification



Sure, there are security threats as in any open environment. Security is to be ensured using existing solutions for Internet environment.

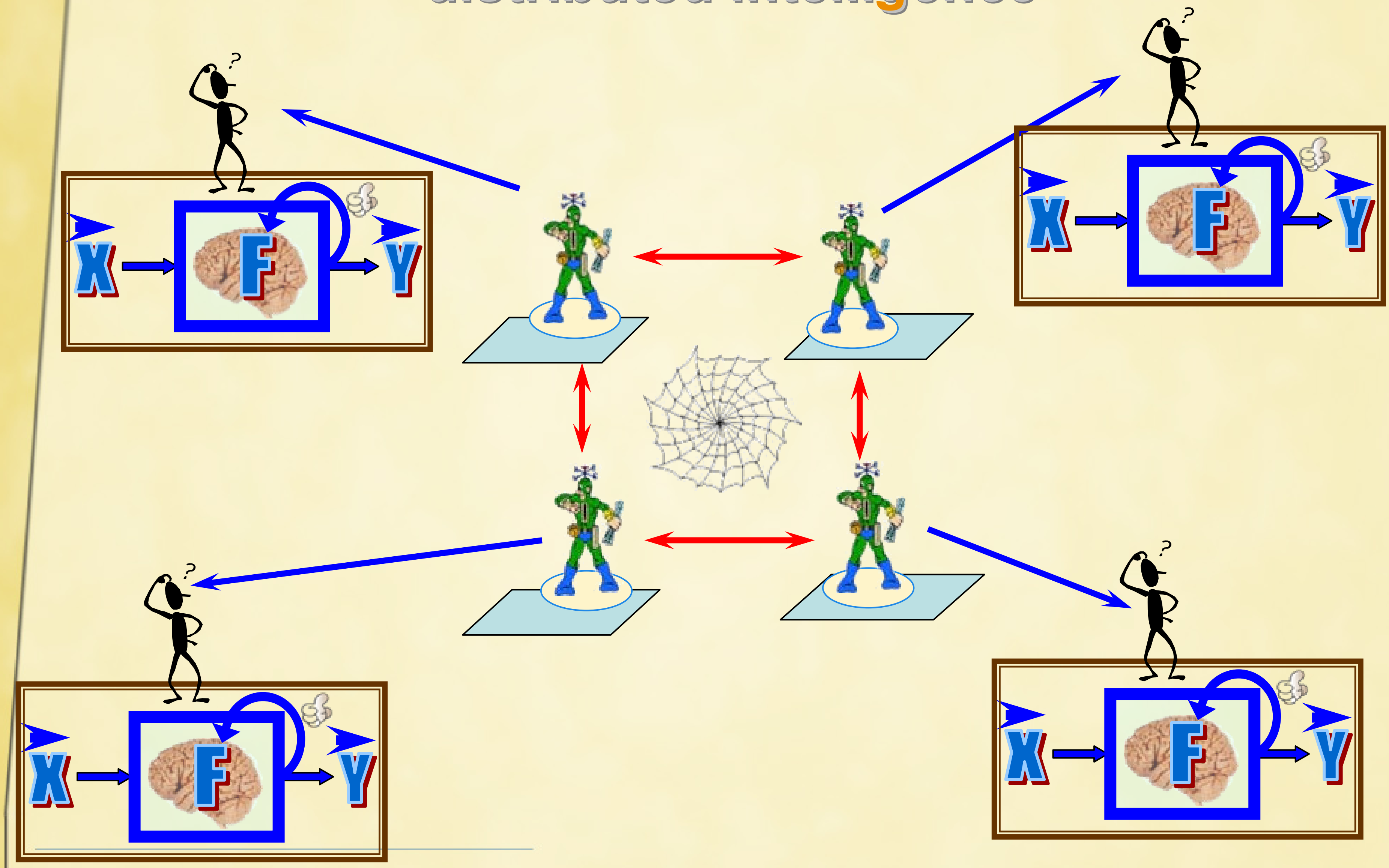
Existence of certification authorities is required in the network. Certificates gained by services and trust to the certificate issuer are factors that influence optimal service selection. The quality of service is evaluated by users as well.

Device-to-Device "opinion" exchange

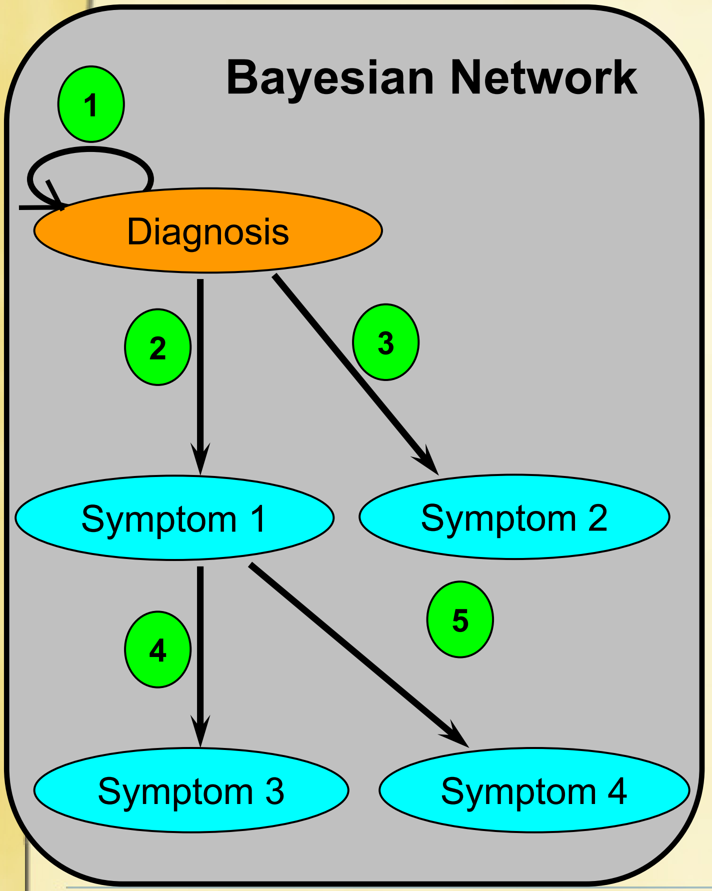


Device will be able to derive service quality estimates basing on analysis of "opinions" of other devices and trust to them.

DistributedBrains: Middleware for distributed intelligence



Semantic Annotation of Models (1)



BN Internal Representation

1
- P(D = OK) = 0.85
- P(D = d1) = 0.13
- P(D = d2) = 0.02

2
- P(S1 = N | D = OK) = 0.68
- P(S1 = N | D = d1) = 0.07
- P(S1 = N | D = d2) = 0.11
- P(S1 = Y | D = OK) = 0.32
- P(S1 = Y | D = d1) = 0.93
- P(S1 = Y | D = d2) = 0.89

3
- P(S2 = N | D = OK) = 0.75
- P(S2 = N | D = d1) = 0.02
- P(S2 = N | D = d2) = 0.14
- P(S2 = Y | D = OK) = 0.25
- P(S2 = Y | D = d1) = 0.98
- P(S2 = Y | D = d2) = 0.86

4
- P(S3 = N | S1 = N) = 0.55
- P(S3 = N | S1 = Y) = 0.60
- P(S3 = Y | S1 = N) = 0.45
- P(S3 = Y | S1 = Y) = 0.40

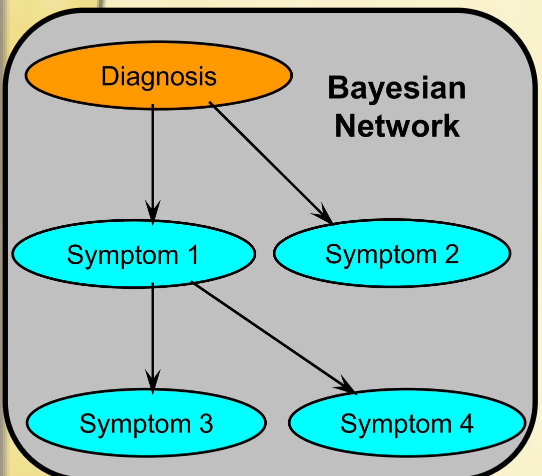
5
- P(S4 = N | S1 = N) = 0.33
- P(S4 = N | S1 = Y) = 0.80
- P(S4 = Y | S1 = N) = 0.67
- P(S4 = Y | S1 = Y) = 0.20

BN Semantic Representation

θ


```xml
<?xml version="1.0" ?>
<owl:Ontology rdf:about="" />
<owl:Class rdf:ID="Diagnoses" />
<owl:Class rdf:ID="Symptoms" />
- <owl:ObjectProperty rdf:ID="depends_on">
<rdfs:domain rdf:resource="#Symptoms" />
- <rdfs:range>
- <owl:Class>
  </owl:Class>
  </rdfs:range>
  </owl:ObjectProperty>
- <owl:DatatypeProperty rdf:ID="conditional_probability">
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float" />
<rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AnnotationProperty" />
<rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty" />
<rdfs:domain rdf:resource="#Symptoms" />
  </owl:DatatypeProperty>
- <owl:FunctionalProperty rdf:ID="probability">
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float" />
<rdfs:domain rdf:resource="#Diagnoses" />
<rdf:type rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty" />
  </owl:FunctionalProperty>
- <Symptoms rdf:ID="Symptom_1">
- <depends_on>
- <Diagnoses rdf:ID="Diagnosis">
<probability rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.85</probability>
  </Diagnoses>
  </depends_on>
  </Symptoms>
- <Symptoms rdf:ID="Symptom_4">
<depends_on rdf:resource="#Symptom_1" />
  </Symptoms>
- <Symptoms rdf:ID="Symptom_3">
<depends_on rdf:resource="#Symptom_1" />
  </Symptoms>
- <Symptoms rdf:ID="Symptom_2">
<depends_on rdf:resource="#Diagnosis" />
  </Symptoms>
  </rdf:RDF>
```

Semantic Annotation of Engines for Models



BN Engine Internal Representation

Conditional Independence

$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i \mid Parents(X_i))$$

Joint Probability

$$P(Y = y_j, X = x_i) = P(X = x_i) \cdot P(Y = y_j \mid X = x_i)$$

Marginalization

$$P(Y = y_j) = \sum_i P(X = x_i) \cdot P(Y = y_j \mid X = x_i)$$

Bayes Theorem

$$P(X = x_i \mid Y = y_j) = \frac{P(X = x_i) \cdot P(Y = y_j \mid X = x_i)}{P(Y = y_j)}$$

P (Diagnosis, S1, S2, S3, S4) =

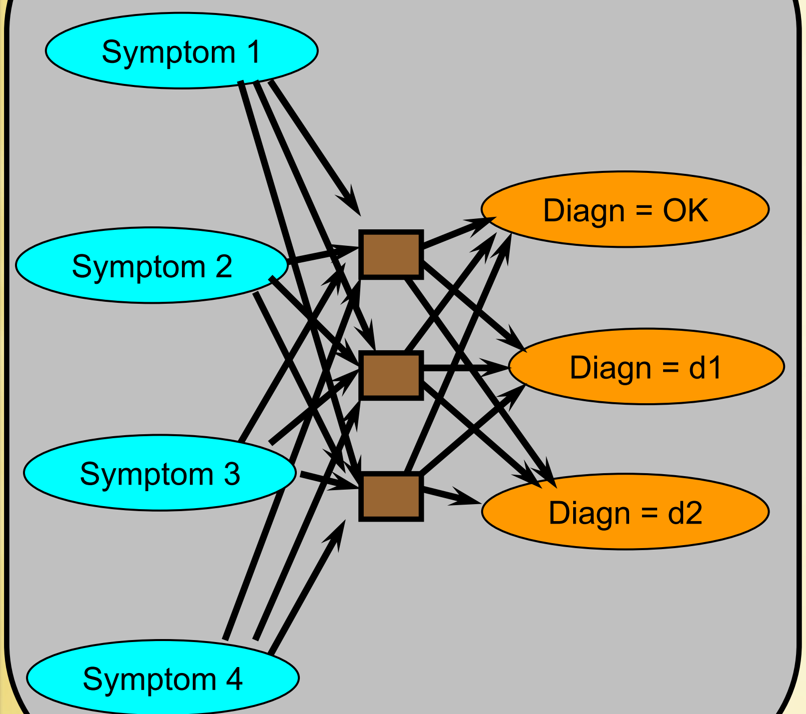
= P(D | S1,S2) * P(S1 | S3,S4) * P(S3) * P(S4)

$$P(Diagnosis) = \sum_{\forall S_1, S_2, S_3, S_4,} P(D, S_1, S_2, S_3, S_4)$$

BN Engine Semantic Representation

Semantic Annotation of Models (2)

Neural Network



NN Internal Representation

In	1	2	3
S1	0.33	0.14	0.25
S2	0.53	0.32	0.12
S3	0.17	0.39	0.44
S4	0.12	0.67	0.21

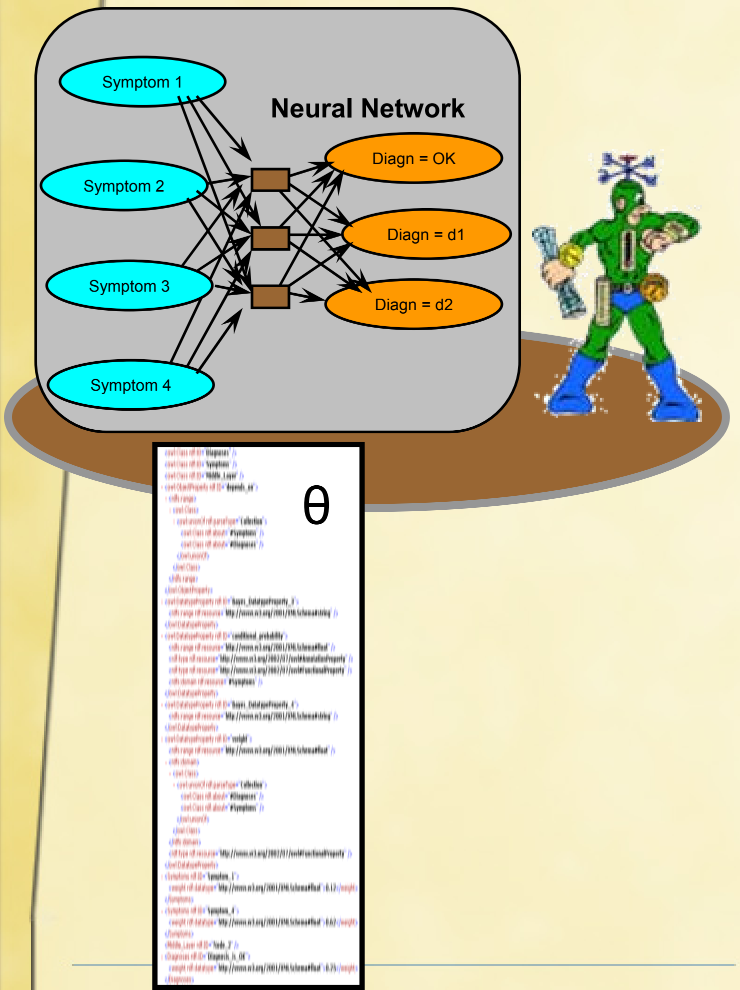
Out	OK	d1	d2
1	0.17	0.14	0.22
2	0.51	0.12	0.18
3	0.11	0.34	0.22

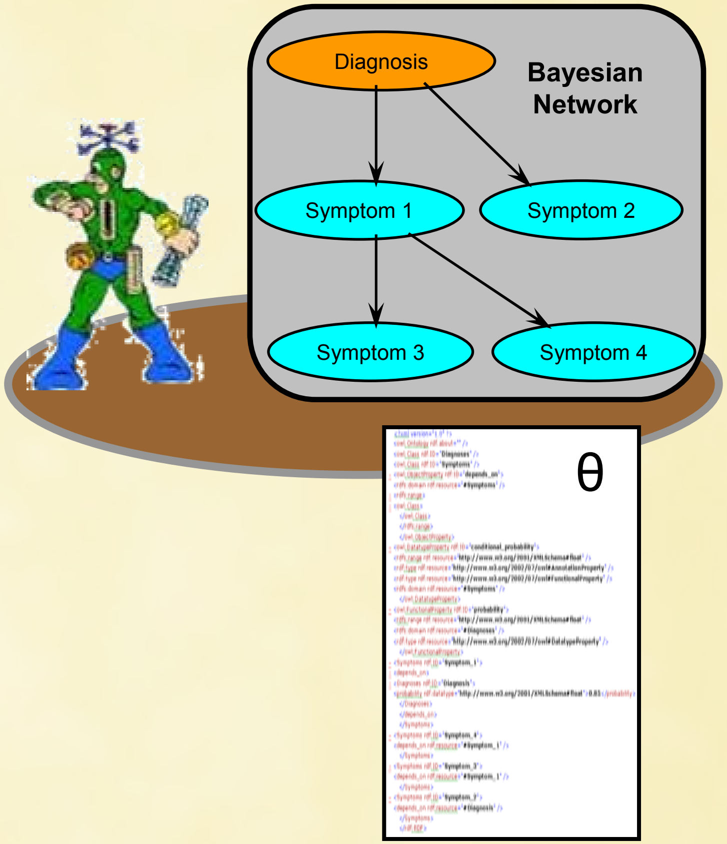
NN Semantic Representation

θ

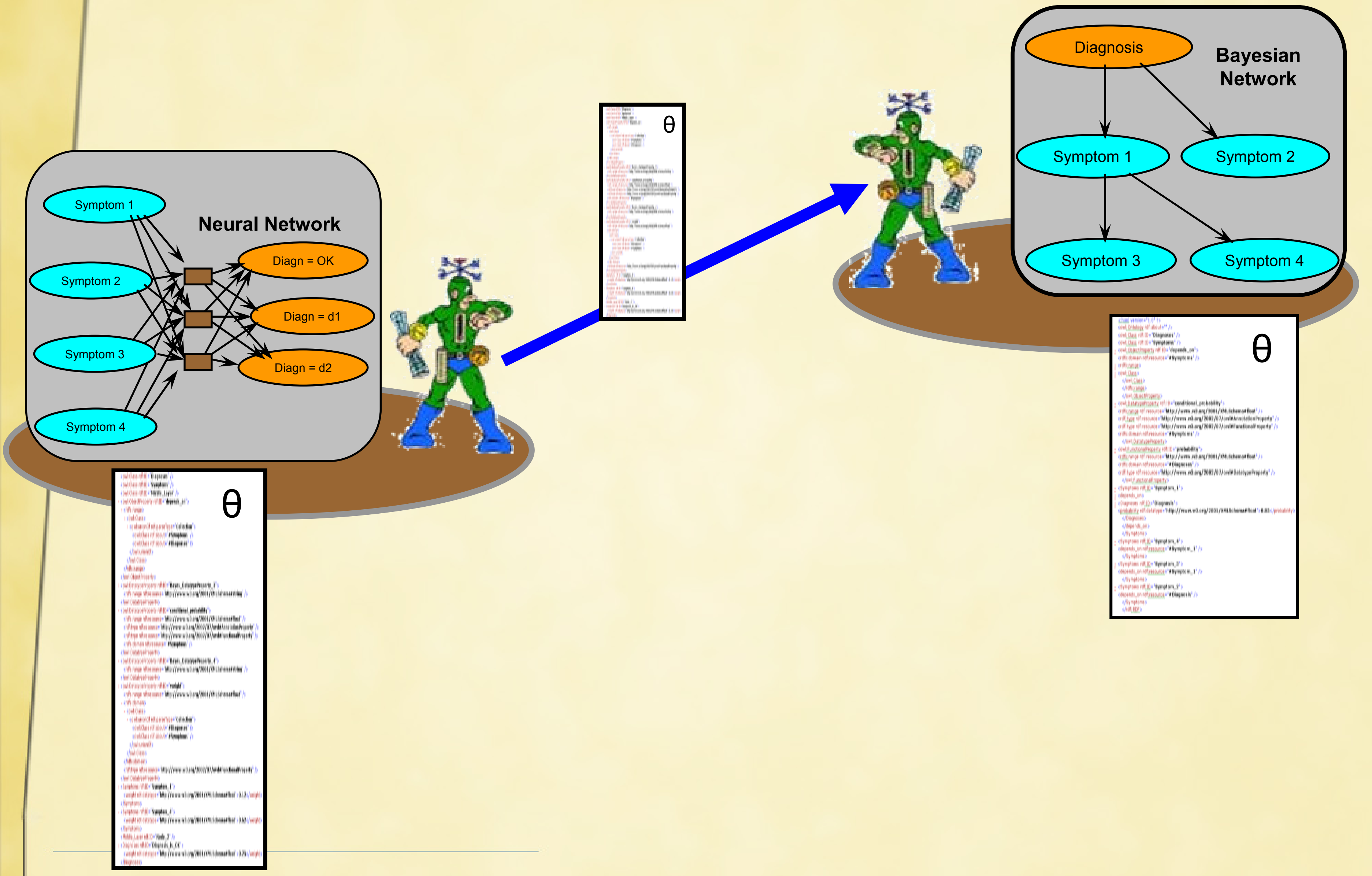

```xml
<owl:Class rdf:ID="Diagnoses" />
<owl:Class rdf:ID="Symptoms" />
<owl:Class rdf:ID="Middle_Layer" />
- <owl:ObjectProperty rdf:ID="depends_on">
  - <rdfs:range>
    - <owl:Class>
      - <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Symptoms" />
          <owl:Class rdf:about="#Diagnoses" />
        </owl:unionOf>
      </owl:Class>
    </rdfs:range>
  </owl:ObjectProperty>
- <owl:DatatypeProperty rdf:ID="Bayes_DatatypeProperty_3">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string" />
  </owl:DatatypeProperty>
- <owl:DatatypeProperty rdf:ID="conditional_probability">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float" />
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AnnotationProperty" />
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty" />
    <rdfs:domain rdf:resource="#Symptoms" />
  </owl:DatatypeProperty>
- <owl:DatatypeProperty rdf:ID="Bayes_DatatypeProperty_4">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string" />
  </owl:DatatypeProperty>
- <owl:DatatypeProperty rdf:ID="weight">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float" />
  - <rdfs:domain>
    - <owl:Class>
      - <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Diagnoses" />
          <owl:Class rdf:about="#Symptoms" />
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty" />
  </owl:DatatypeProperty>
- <Symptoms rdf:ID="Symptom_1">
    <weight rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.12</weight>
  </Symptoms>
- <Symptoms rdf:ID="Symptom_4">
    <weight rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.62</weight>
  </Symptoms>
  <Middle_Layer rdf:ID="Node_2" />
- <Diagnoses rdf:ID="Diagnosis_is_OK">
    <weight rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.25</weight>
  </Diagnoses>
```

Models Exchange and Composition (1)

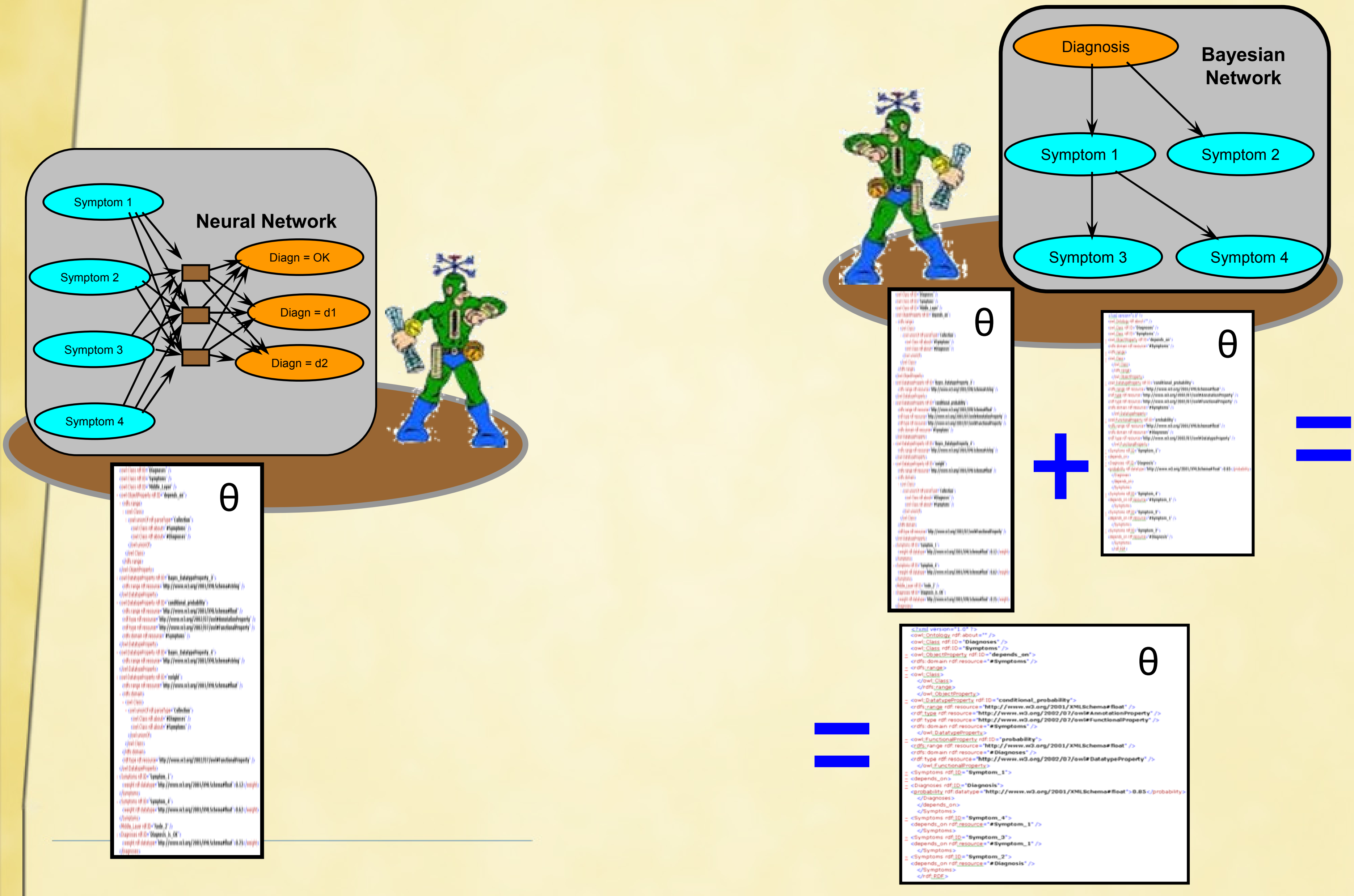




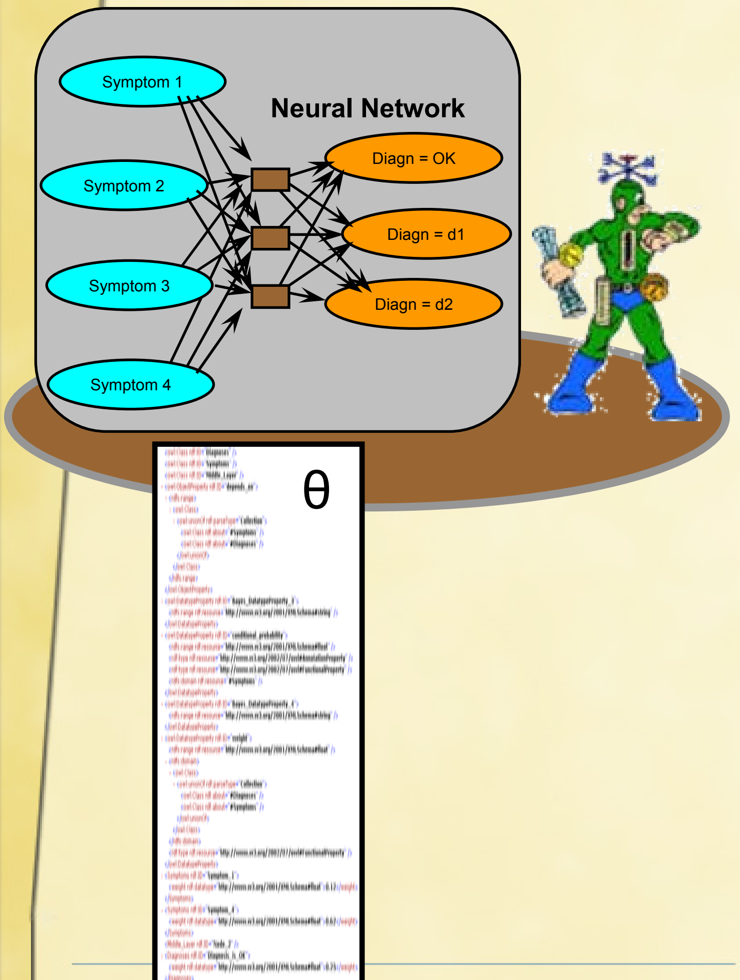
Models Exchange and Composition (2)

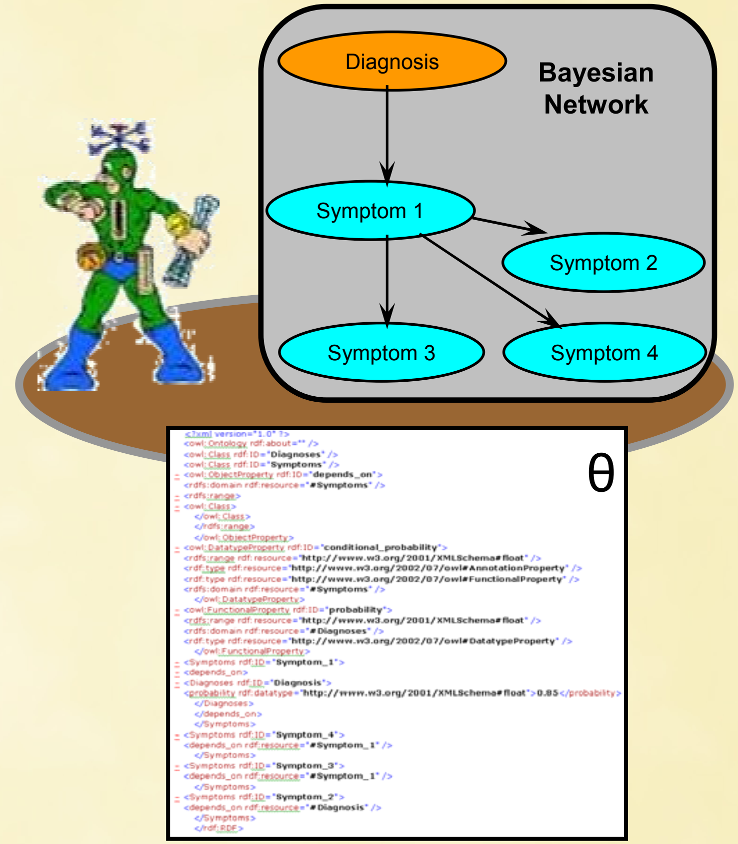


Models Exchange and Composition (3)

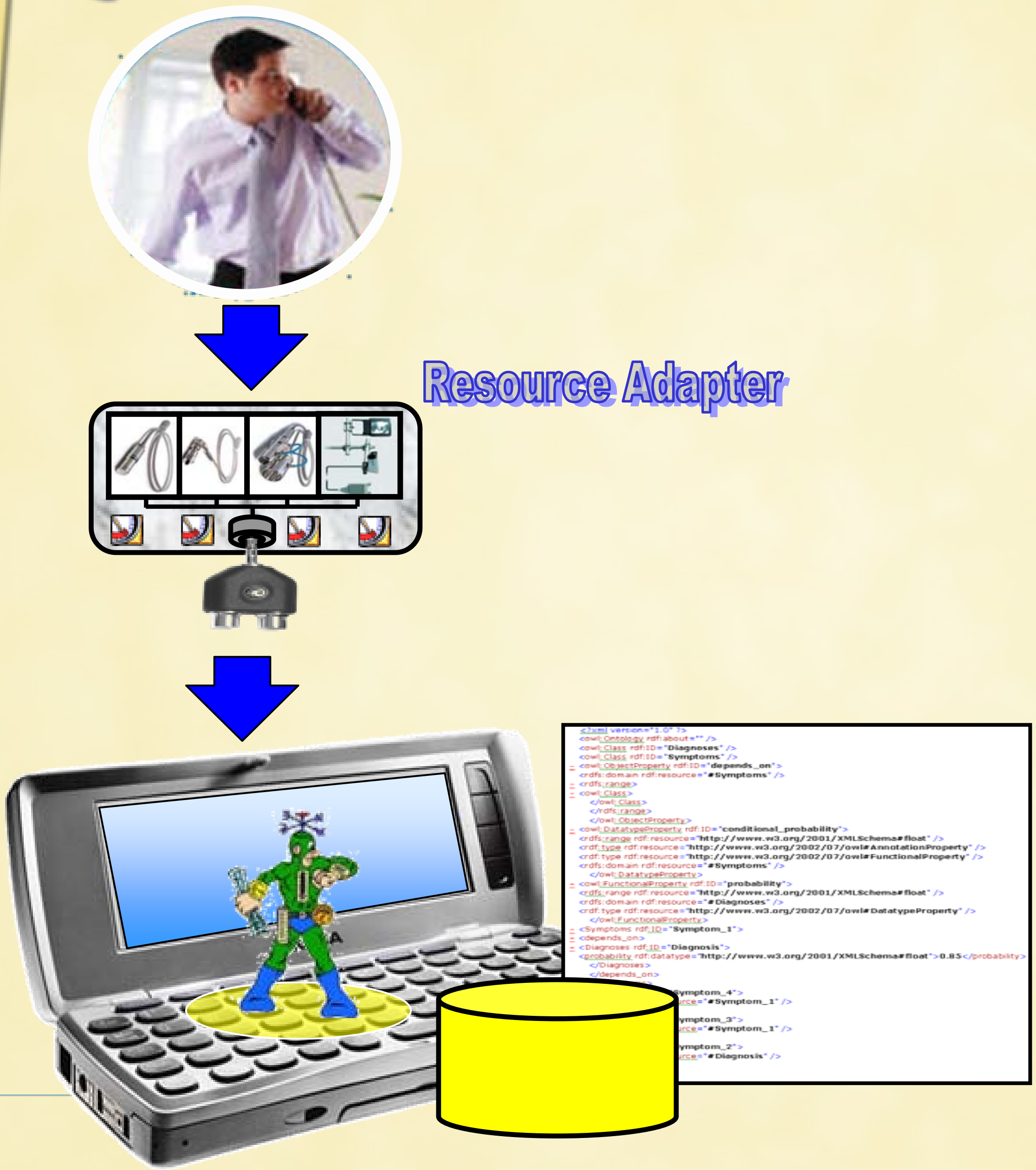


Models Exchange and Composition (4)

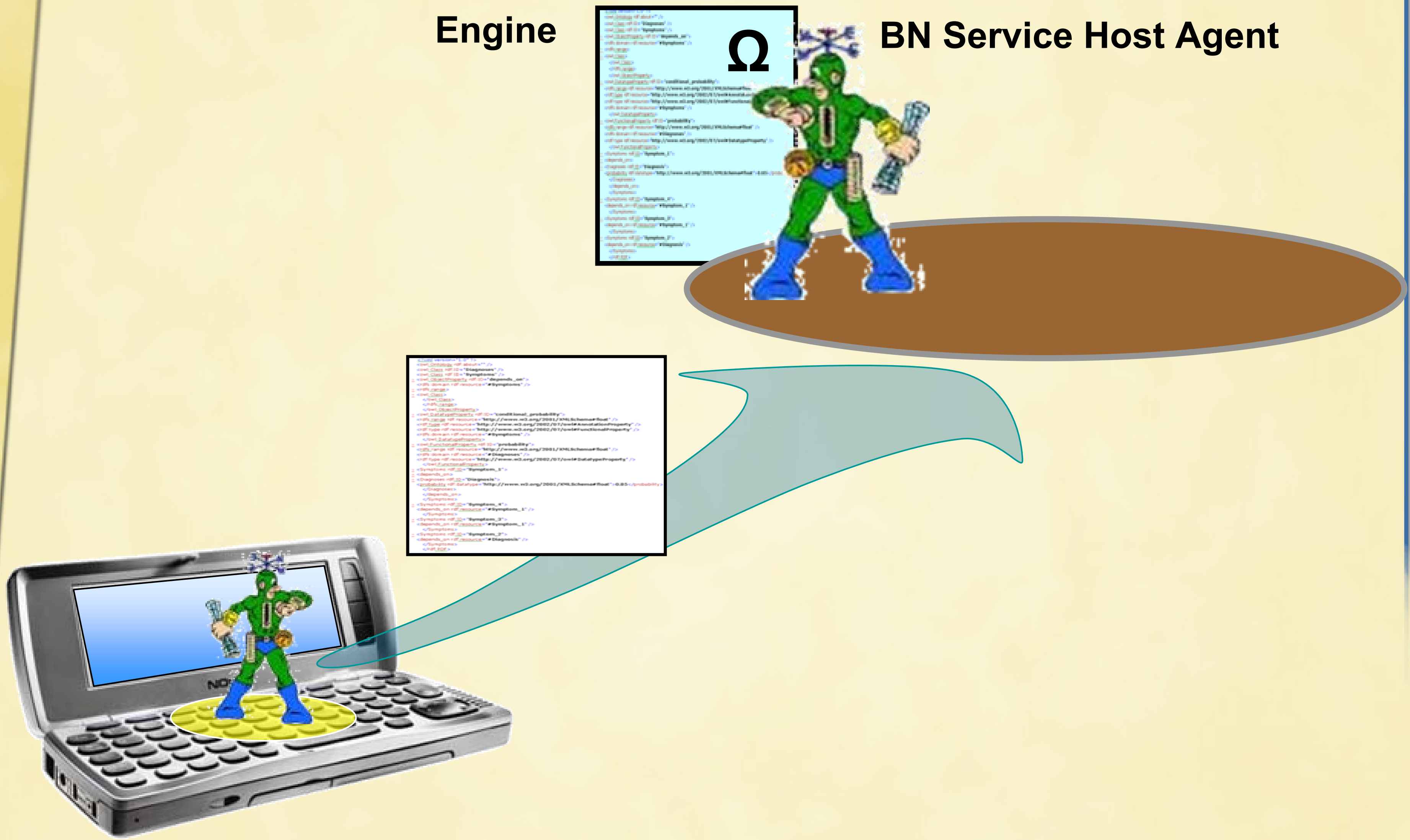




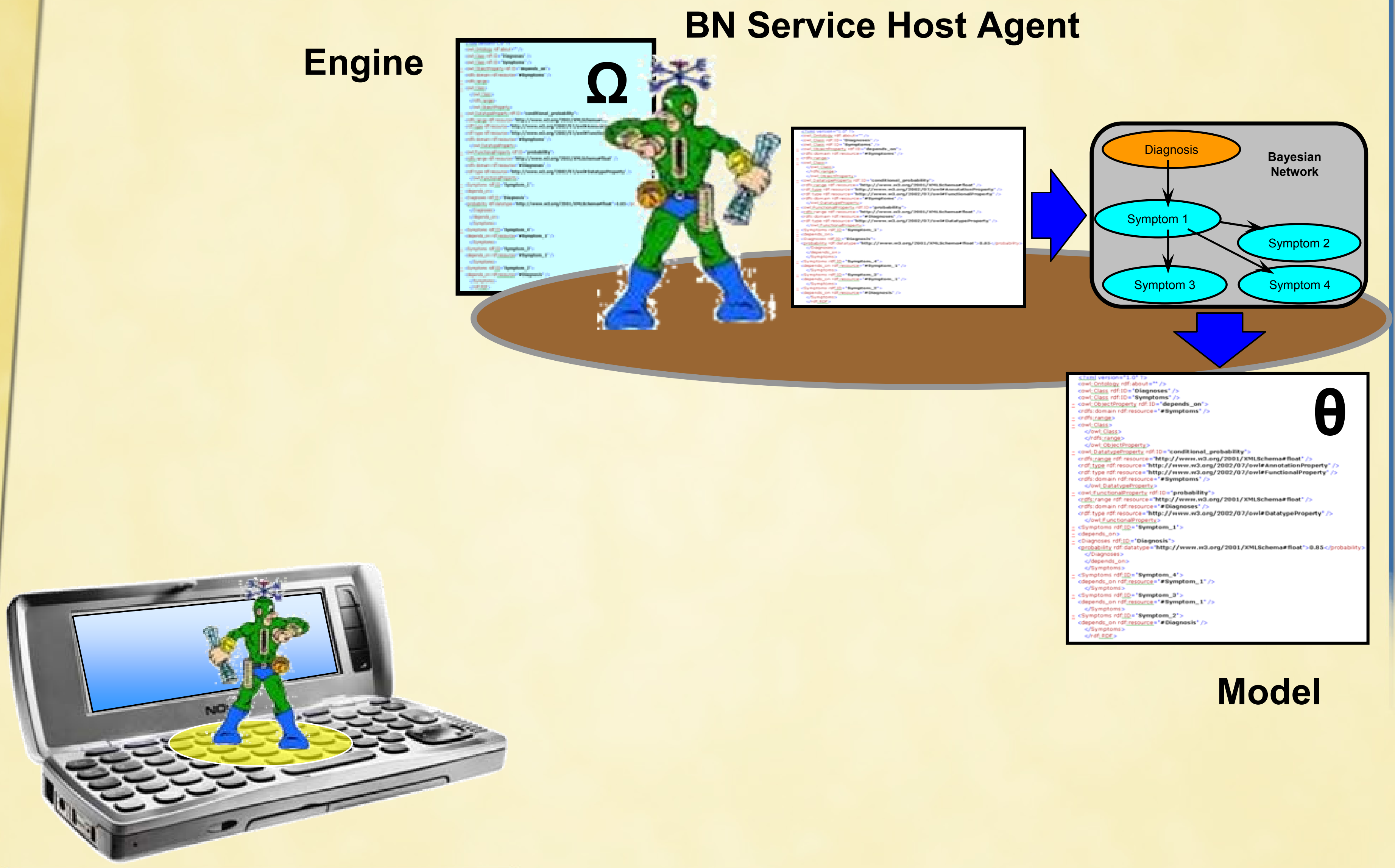
Local Collection of a Resource Data



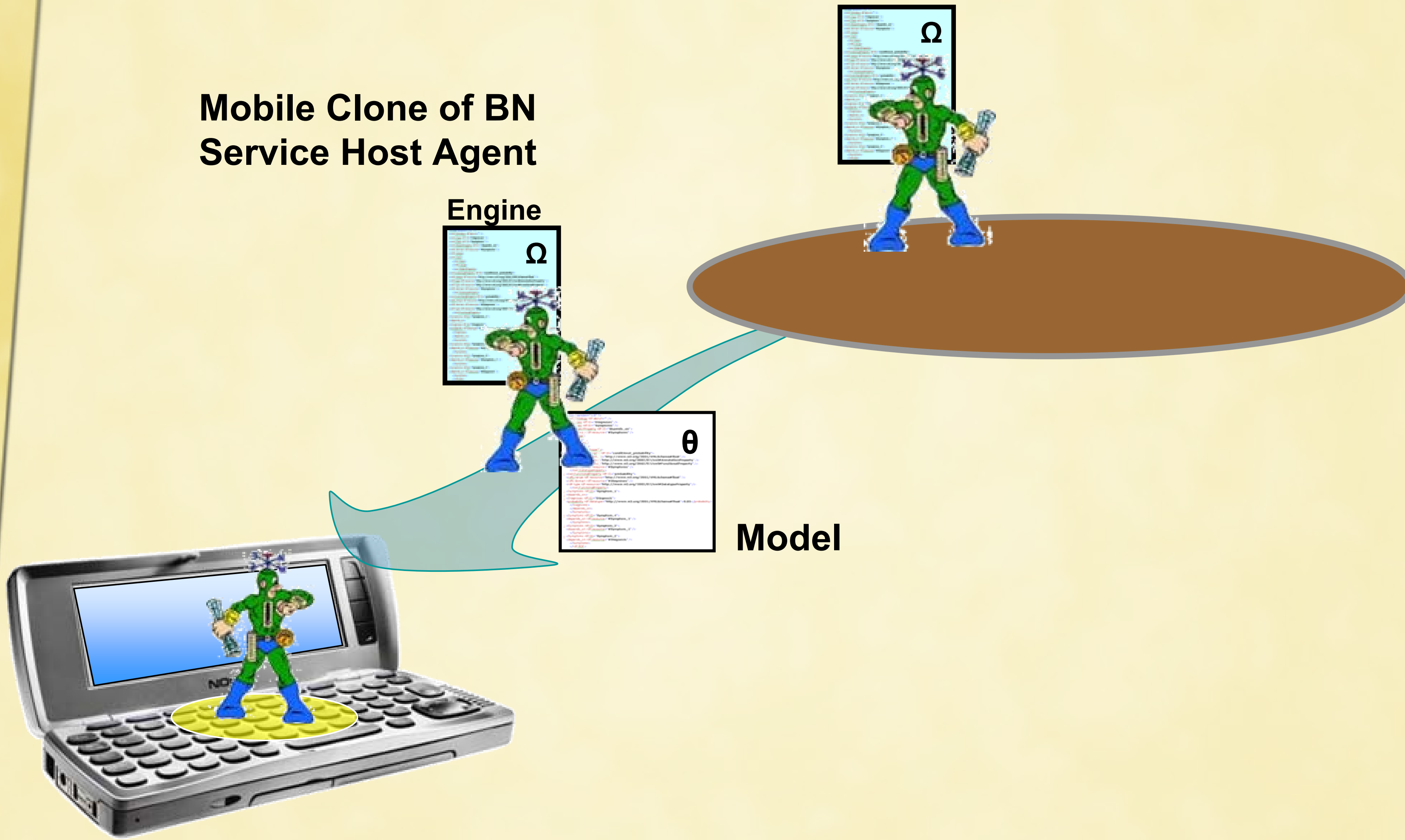
Resource Data Sent to Service



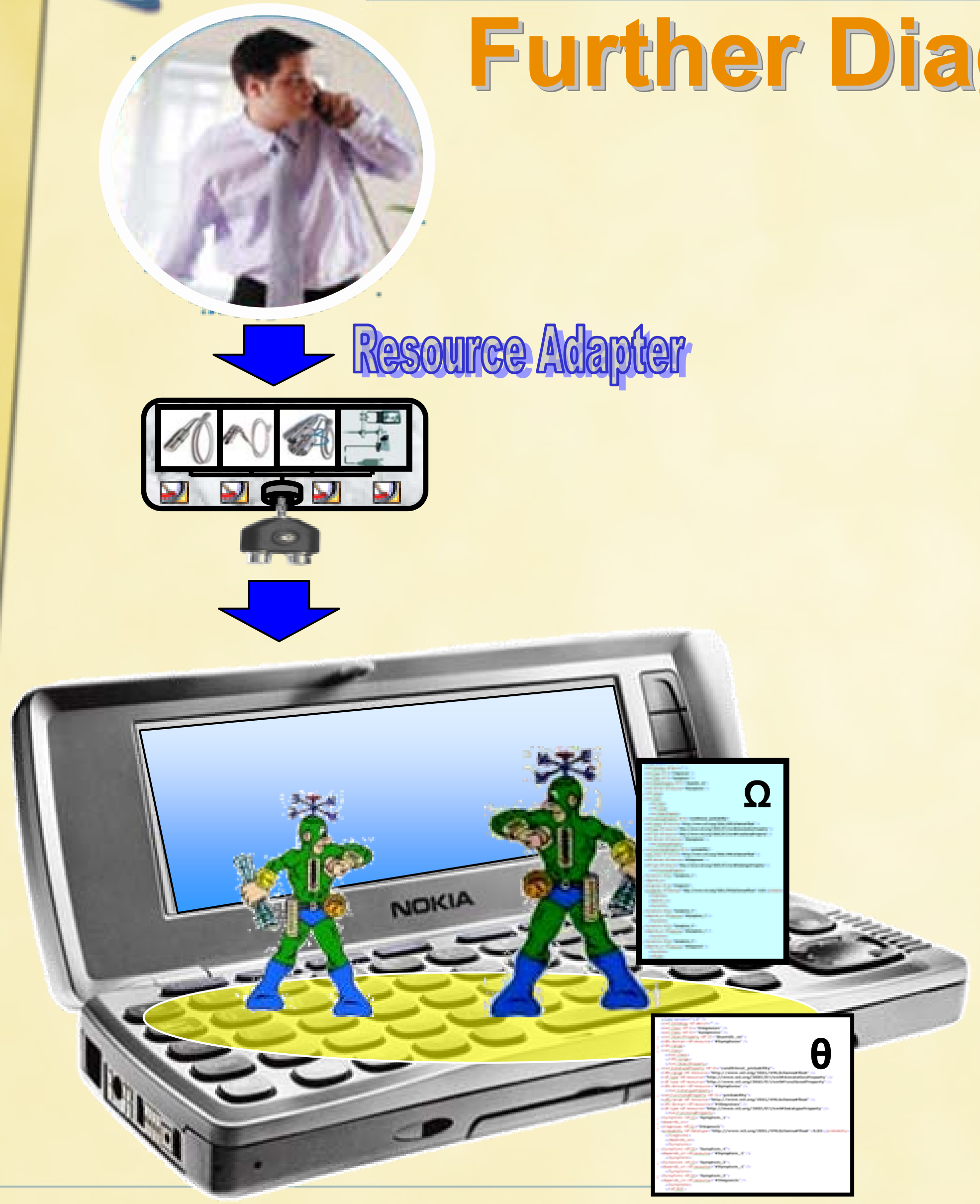
Service Creates and Annotates the Model



Agent with Learned Model Moves to the Client Platform



“Guest” Agent performs Further Diagnostics Locally



1.8. Web 4.1

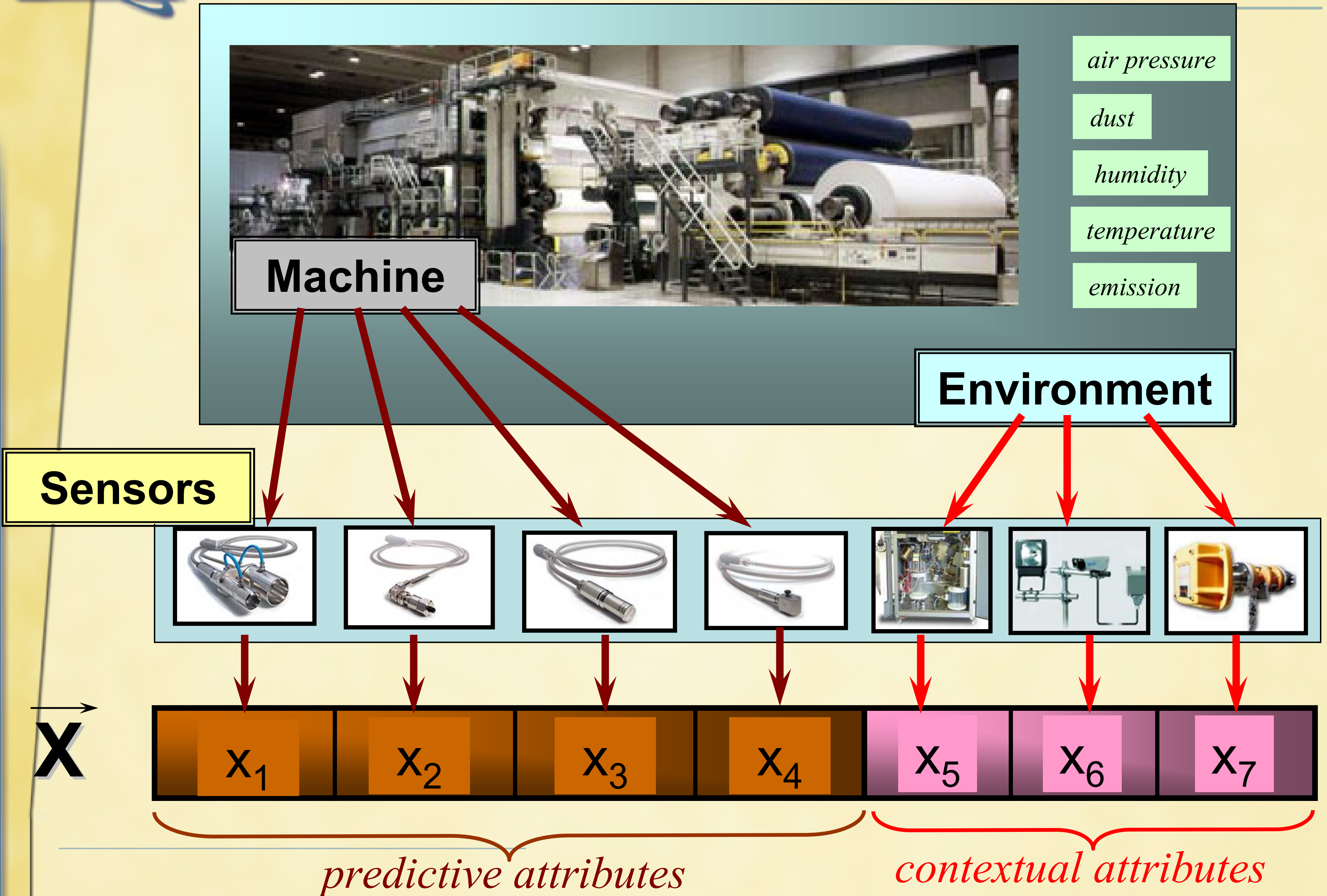
Web of Context

Web of Context

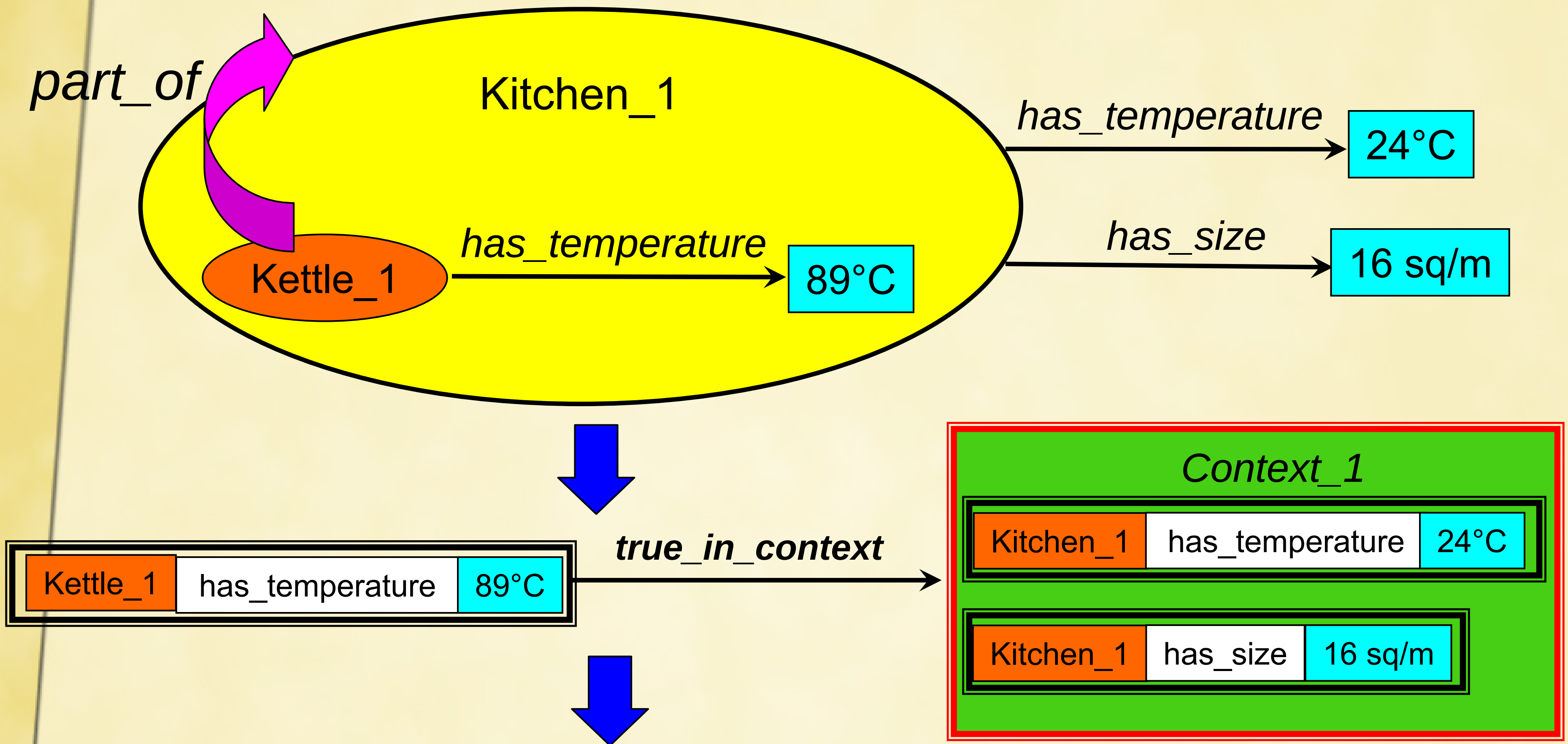
Consider 3 types of a context:

- *Part_of* context
- *Role-based* context
- *Interface-based* context

The nature of *part_of* context

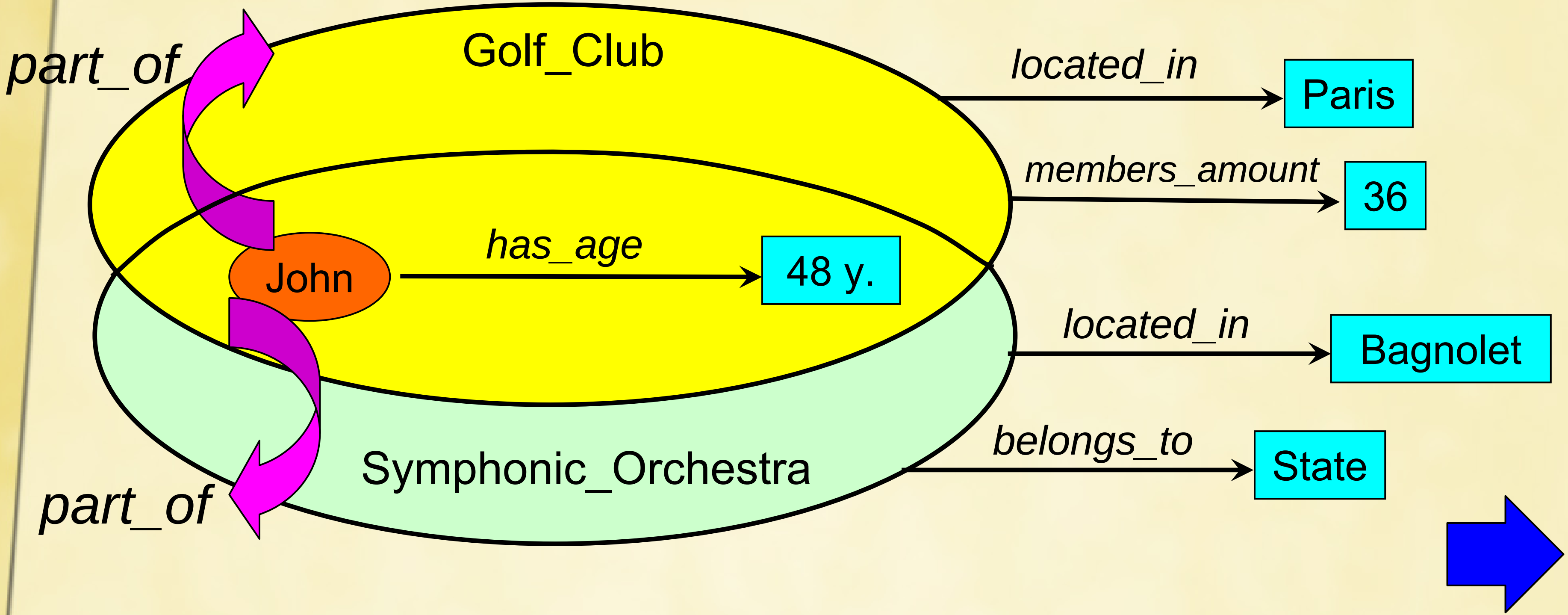


Part-of Context example



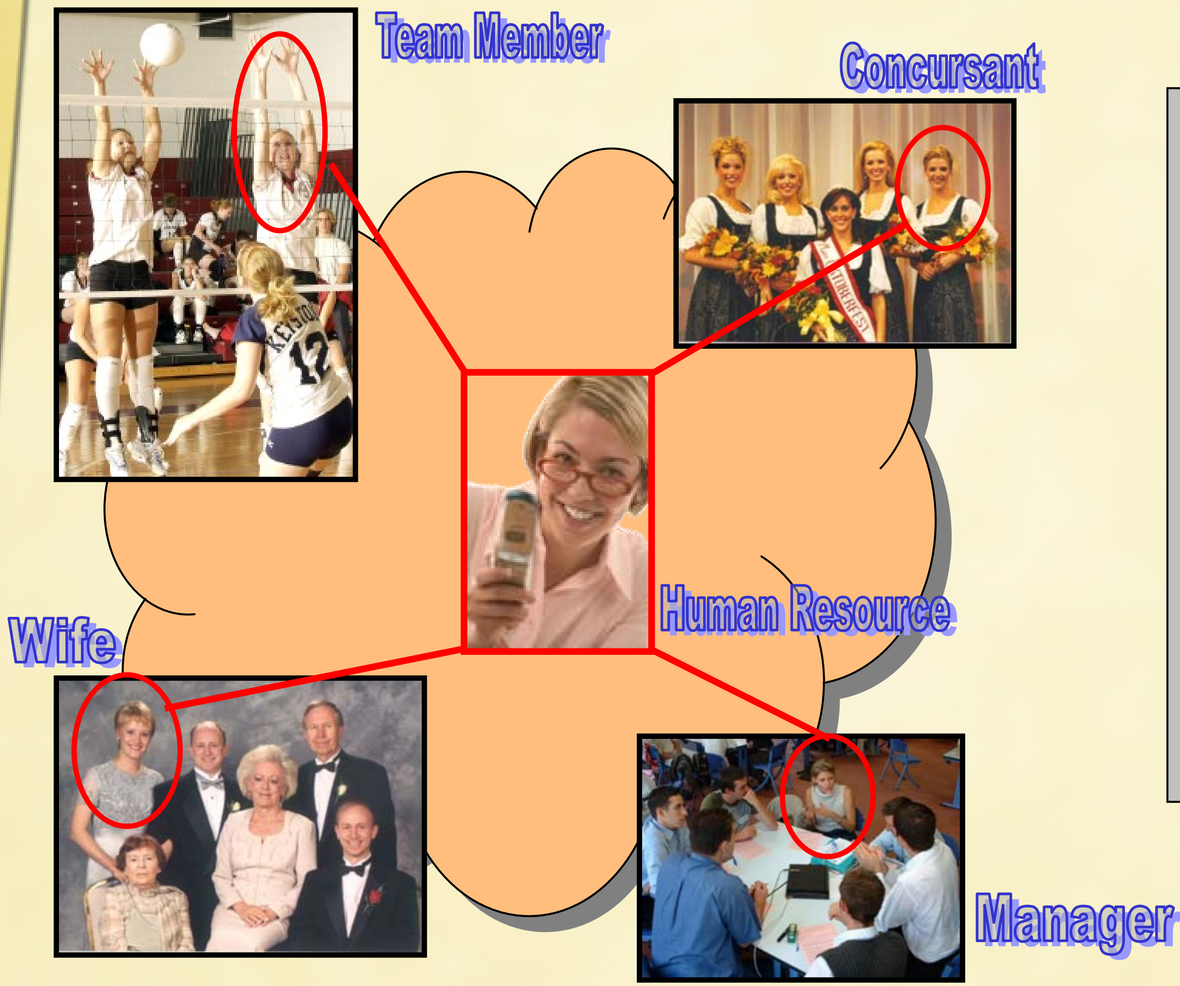
Resource	Predictive features	Contextual Features	
Kettle_1	temperature	*environment*_temperature	*environment*_size

Multiple Context Inheritance …



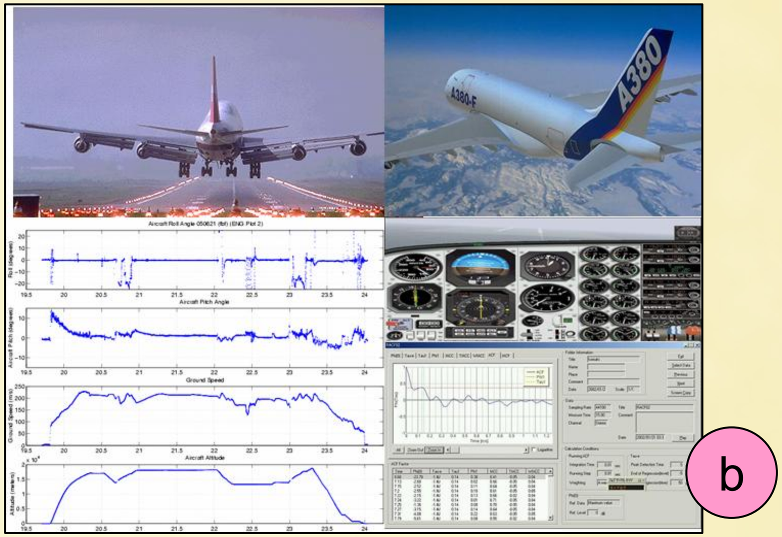
Resource	Predictive features	Contextual Features (inherited from both parents)	
John	age	***environment_1***_location	***environment_1***_members amount
		environment_2_location	***environment_2***_belongs_to

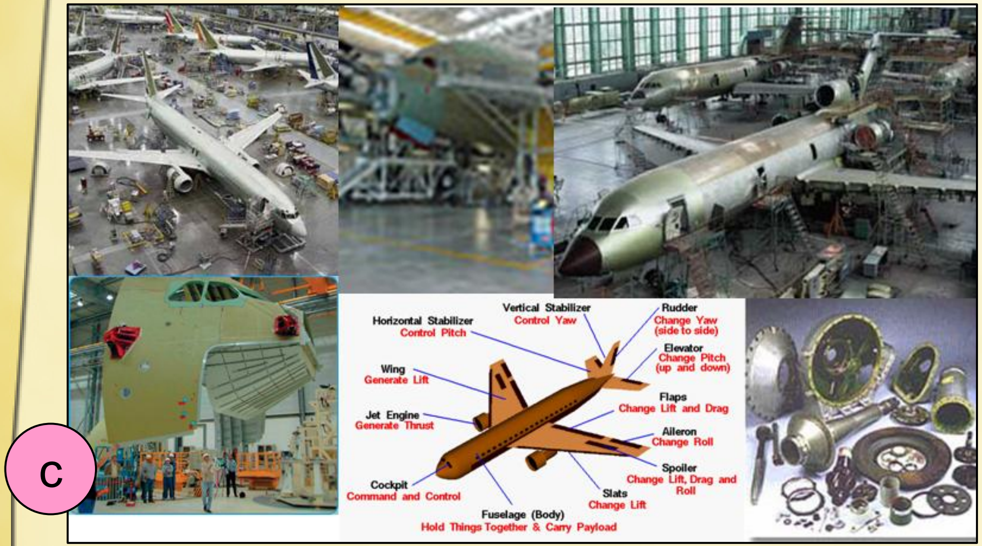
Role-based context



The example of the proactive object (human resource), which is member of several organization and which is playing different roles in each of them. The context of this object should include the description of these roles (duties, commitments, responsibilities, etc).

Interface-based context





The example of the domain object (aircraft) is shown in different interfaces: (a) Google Maps; (b) pilots' control panel; (c) manufacturing design e-manual. Each interface is considered as a context, which affect on which parameters of the aircraft are to be shown

Business potential based on the Web of Context

Context Web Browser

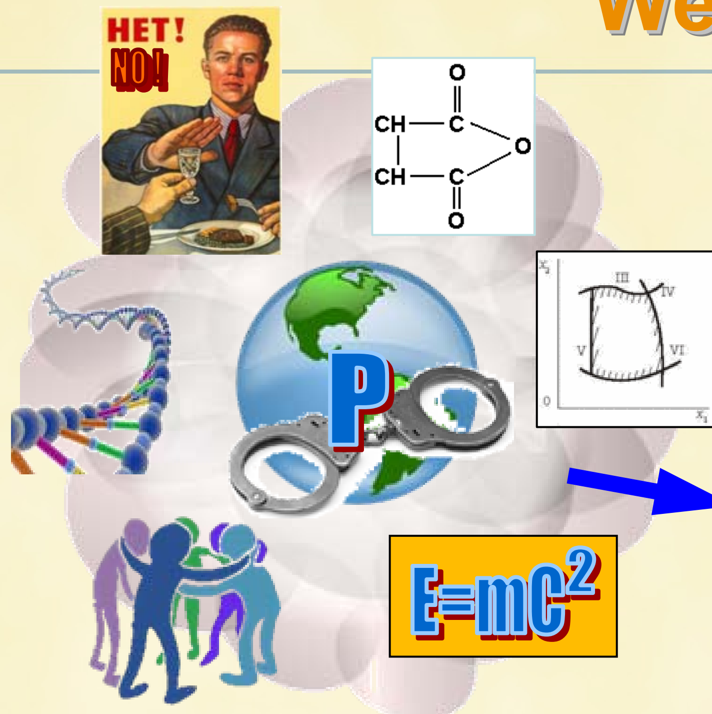
- **Cowboy: Co**ntext **W**eb **B**rowser **Oy** :
  - ❑ Context URIs;
  - ❑ Context creation and publishing tool;
  - ❑ Context annotation tool;
  - ❑ Context search (retrieval, extraction) engine;
  - ❑ Context-based filtering engine;
  - ❑ Decontextualization and context lifting tool;
  - ❑ Context visualization tool;
  - ❑ Content-in-Context presentation engine;
  - ❑ Context mash-ups and context integration.

1.9. Web 4.2

Web of Policies

Web of Policies

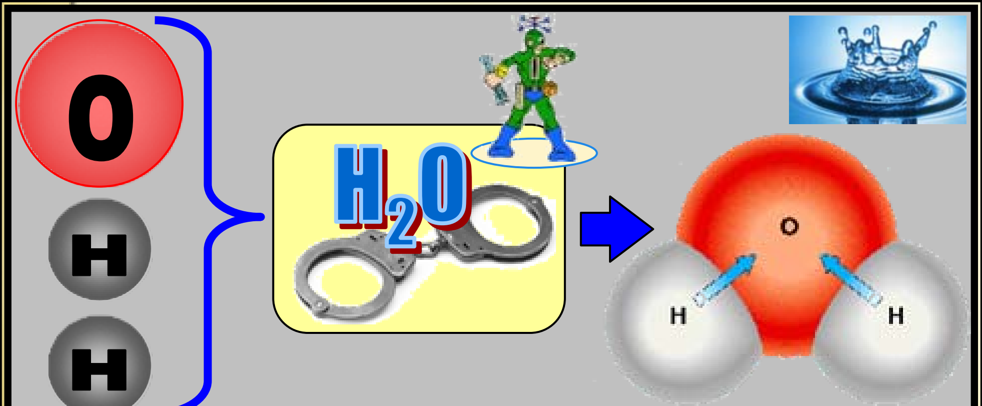
Formal relationships, constraints, limitations, laws, mathematical models, defined business logic, communication protocols, goals/tasks definitions, permissions, prohibitions, commitments, conventions and other policies



Facilitates Policy-to-Policy interaction

Web of Policies

- Policy-Based Control
- Policy-Based Reasoning
- Policy-Based Coordination
- Policy negotiations
- Policy integration

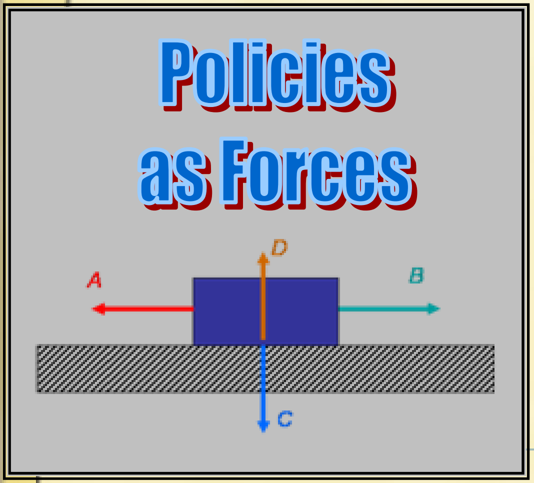


PaaS: Policy-as-a-Service

PaaU: Policy-as-a-User

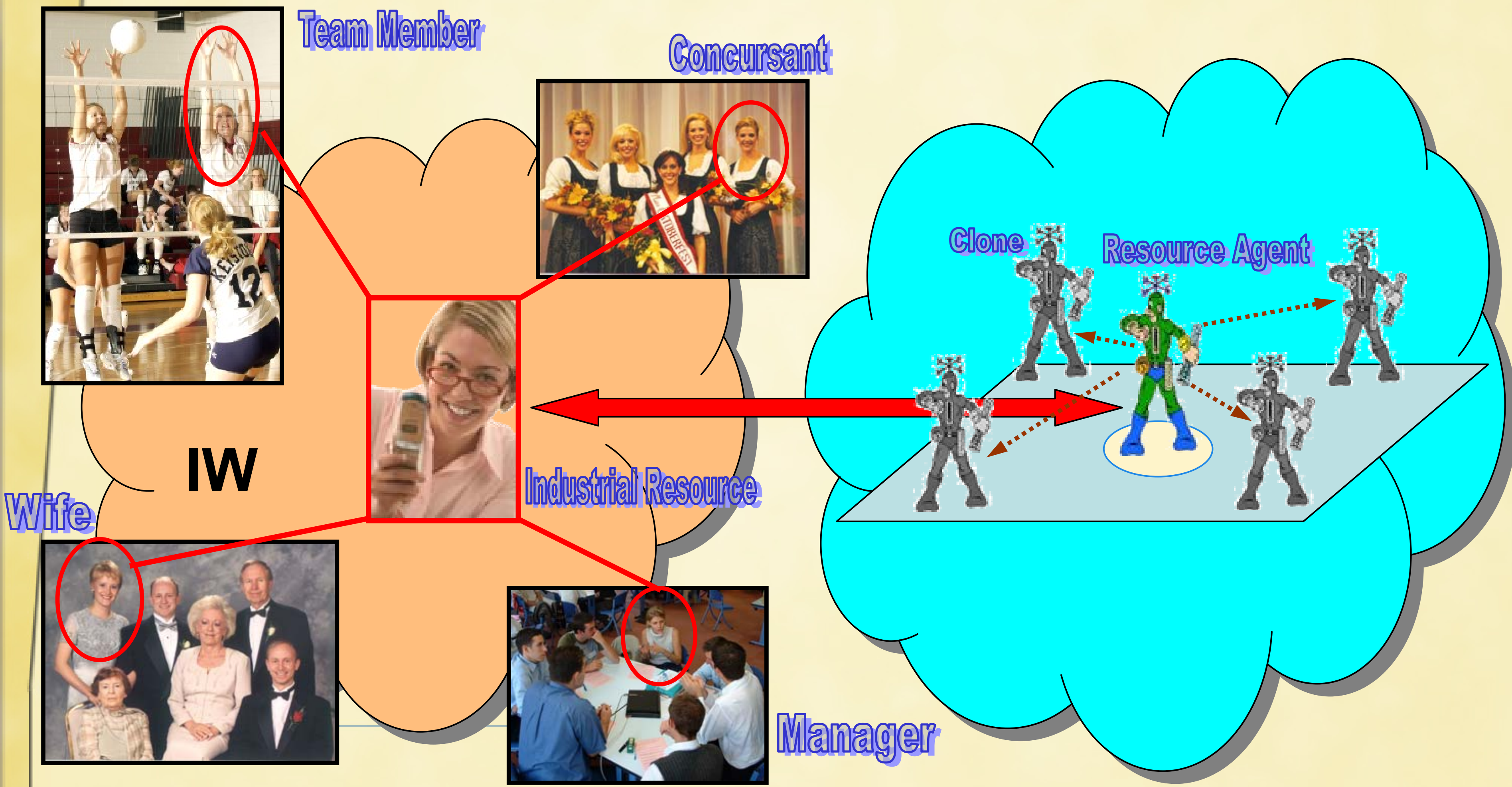
Policy Web Browser

- **WebPolice:** Policy Web Browser:
  - ❑ Policy URIs;
  - ❑ Policy creation and publishing tool;
  - ❑ Policy annotation tool;
  - ❑ Policy search (retrieval, extraction) engine;
  - ❑ Policy enforcement engine;
  - ❑ Policy reconfiguration tool;
  - ❑ Policy visualization tool;
  - ❑ Content-in-Policy presentation engine;
  - ❑ Policy-in-Context presentation tool;
  - ❑ Context-in-Policy presentation tool;
  - ❑ Policy mash-ups and policy integration.



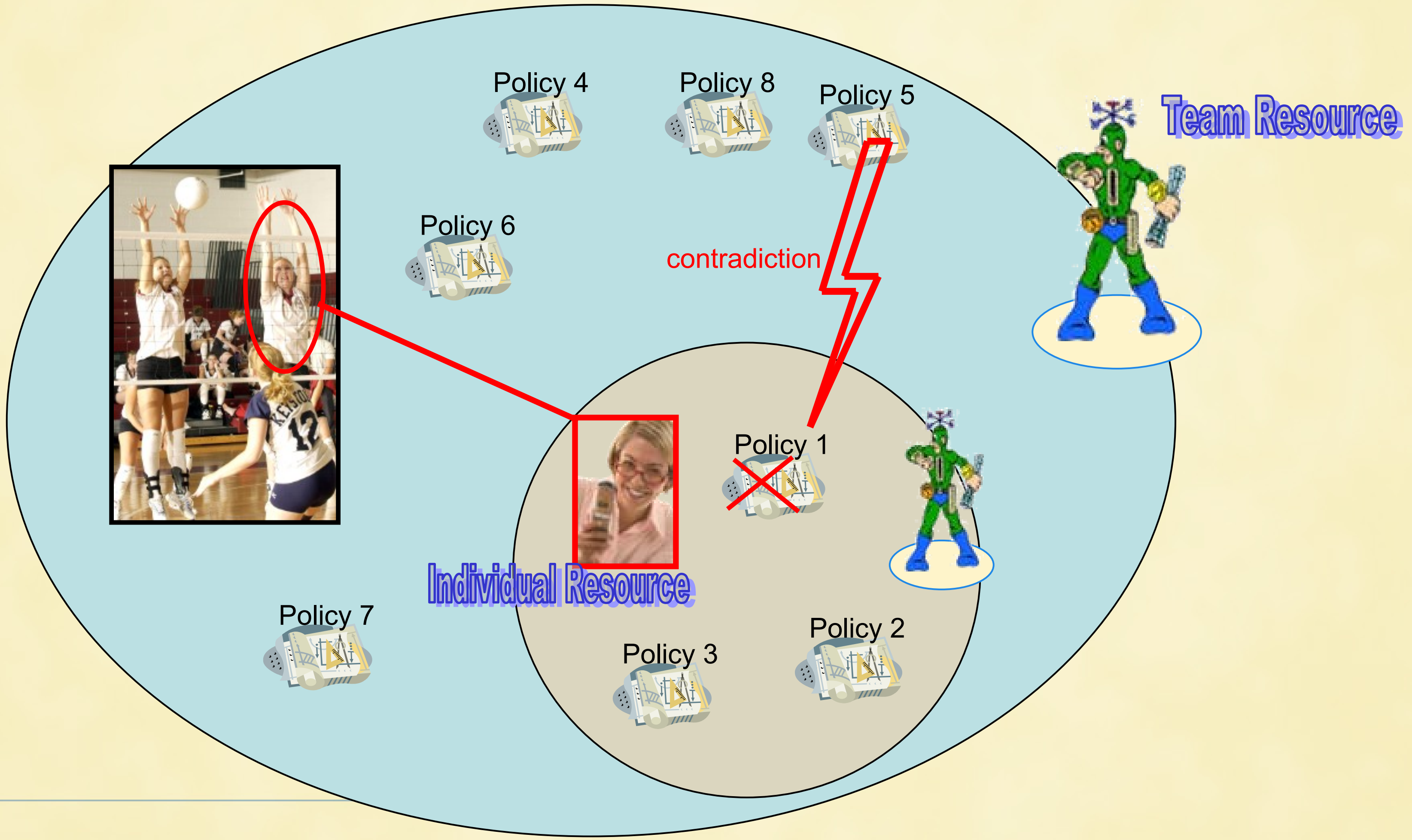
Multiple Policies

- Each industrial resource can theoretically be involved to several processes (organizations, relationships), and appropriate commitments (policies) of each process are applied to it, which can be either supplementary or contradictory. This means that the resource is part of several more complex resources and its role within each of the resource might be different. Modeling such resources can be provided by appropriate resource agent, which can make clones of itself and distribute all necessary roles among them.

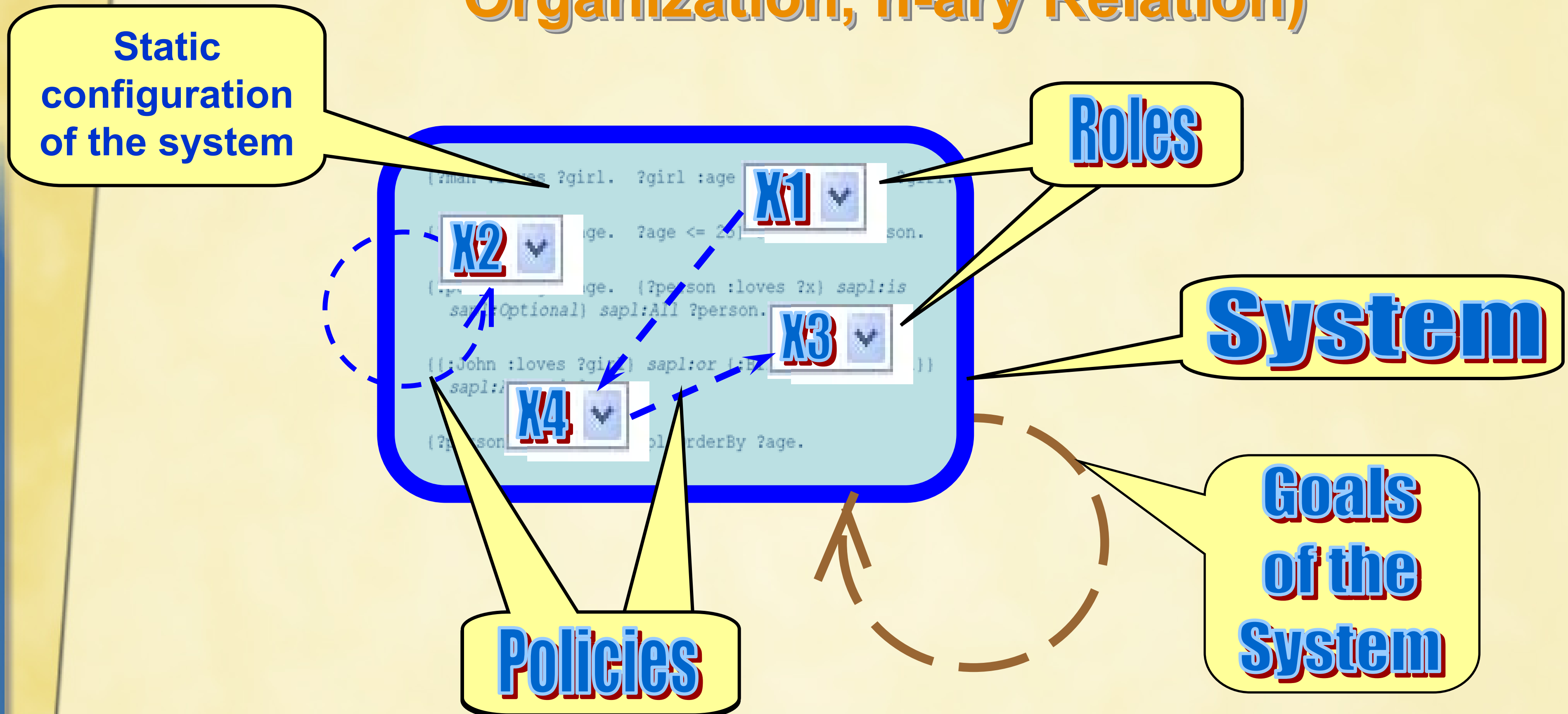


Locally Valid Policies

- Each industrial resource, which joins some commitment, will behave according to the policies, which that commitment requires. The more commitments individual resource takes, the more restrictions will be put on its behavior.

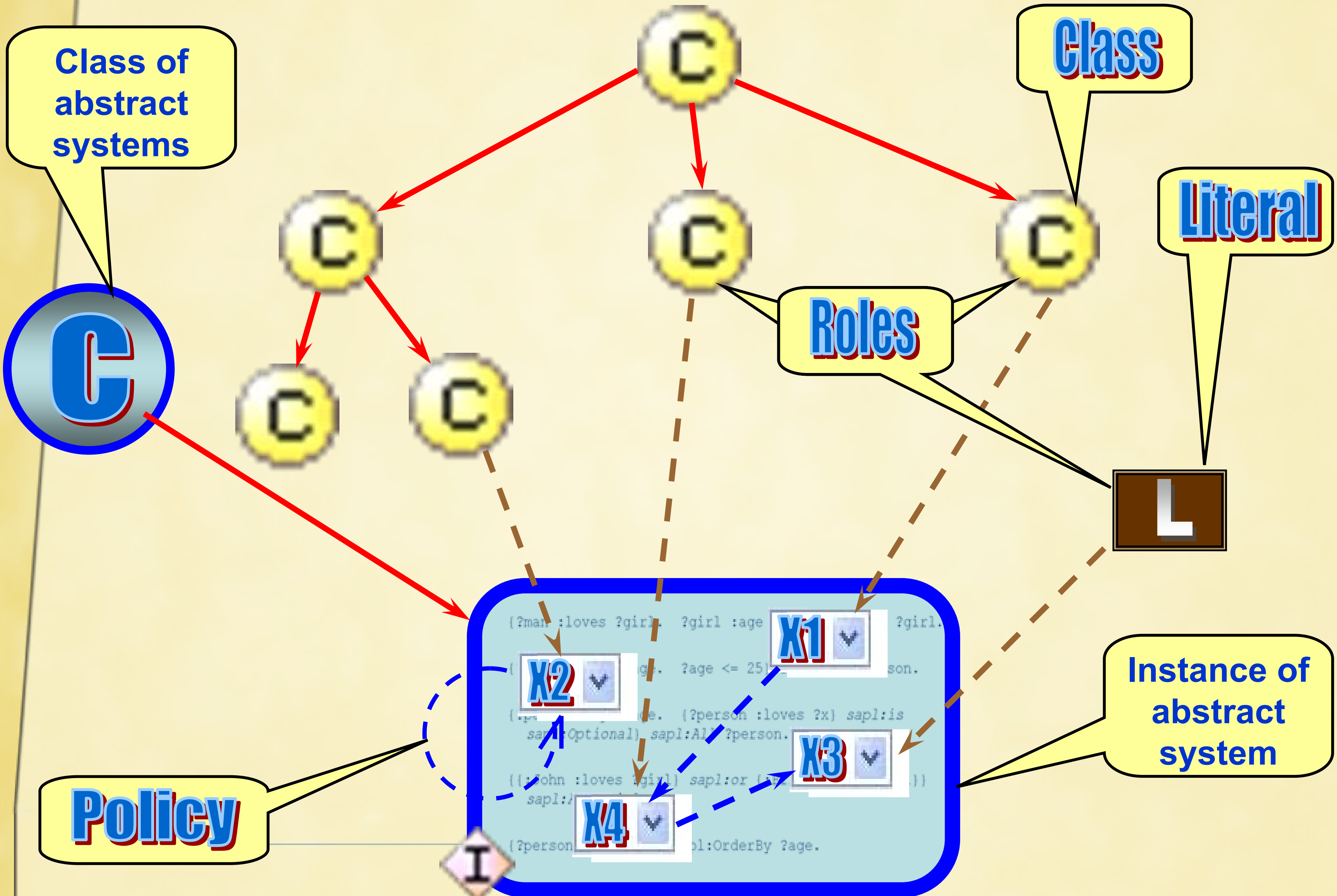


Abstract System (Complex Resource, Organization, n-ary Relation)

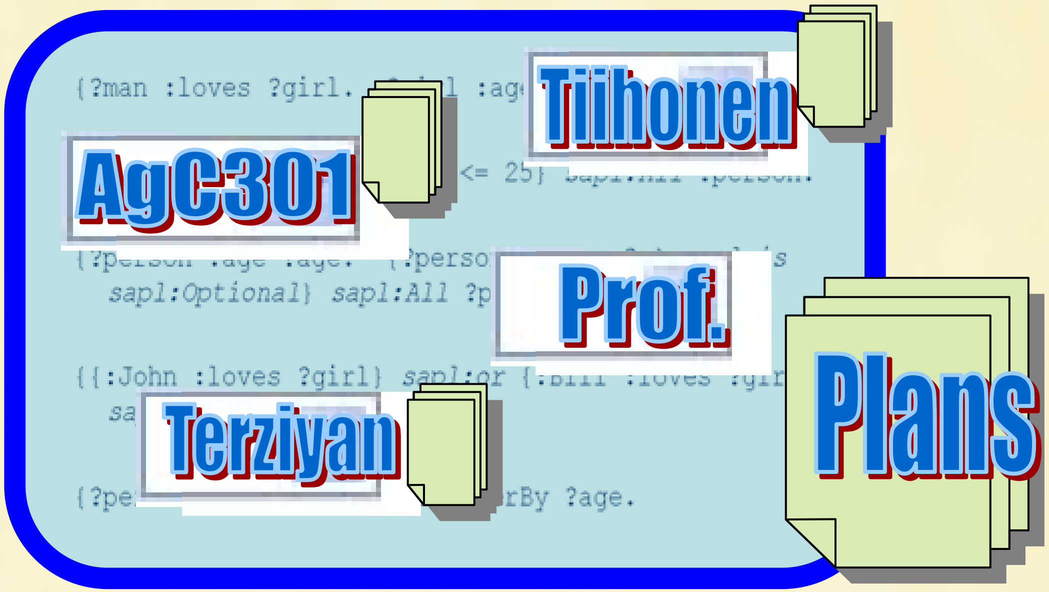


An *abstract system* (or “organization”) is such complex resource, which configuration contains (dynamic) *goals*, variables (“*roles*”) either for resources (system components, subsystems, etc.) or for property values, and also constraints and relationships on/between variables (“*policies*”)

Abstract System within Domain Ontology



Concrete or Executable System



Executable System is the result of transformation from an abstract system, in which all the roles are taken by concrete resources and the goals and policies are replaces by concrete plans (on how to reach the goal with respect to policies).

All subsystems (components) of an executable system are executable systems

1.10. Web 4.3

Web of Configurations

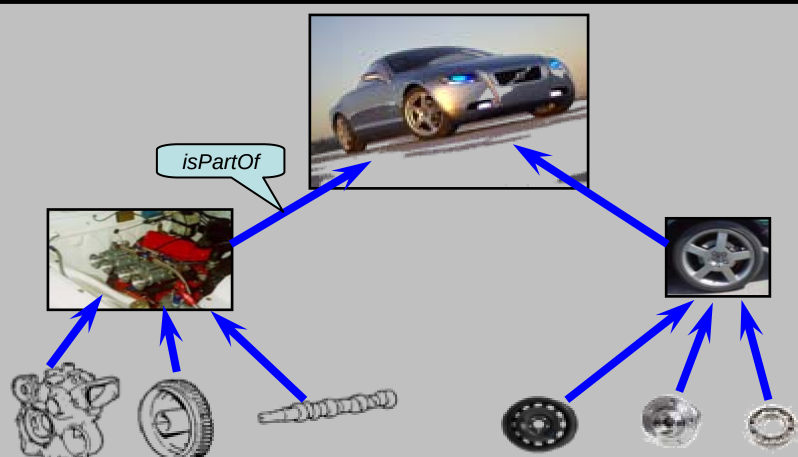
Web of Configurations

… is the Web of "**partonomy**" (a classification based on part-of relation; not the same as taxonomy, which is a classification based on similarities). Configuration of an object (parts and their relationships) together with all policies applied to these parts fully describes the object from inside.

Facilitates Configuration-to-Configuration interaction

Web of Configurations

- Self-Configuration
- Configuration-Based Reasoning
- Mobile Configuration
- Configuration Web Browser
- Proactive Configuration

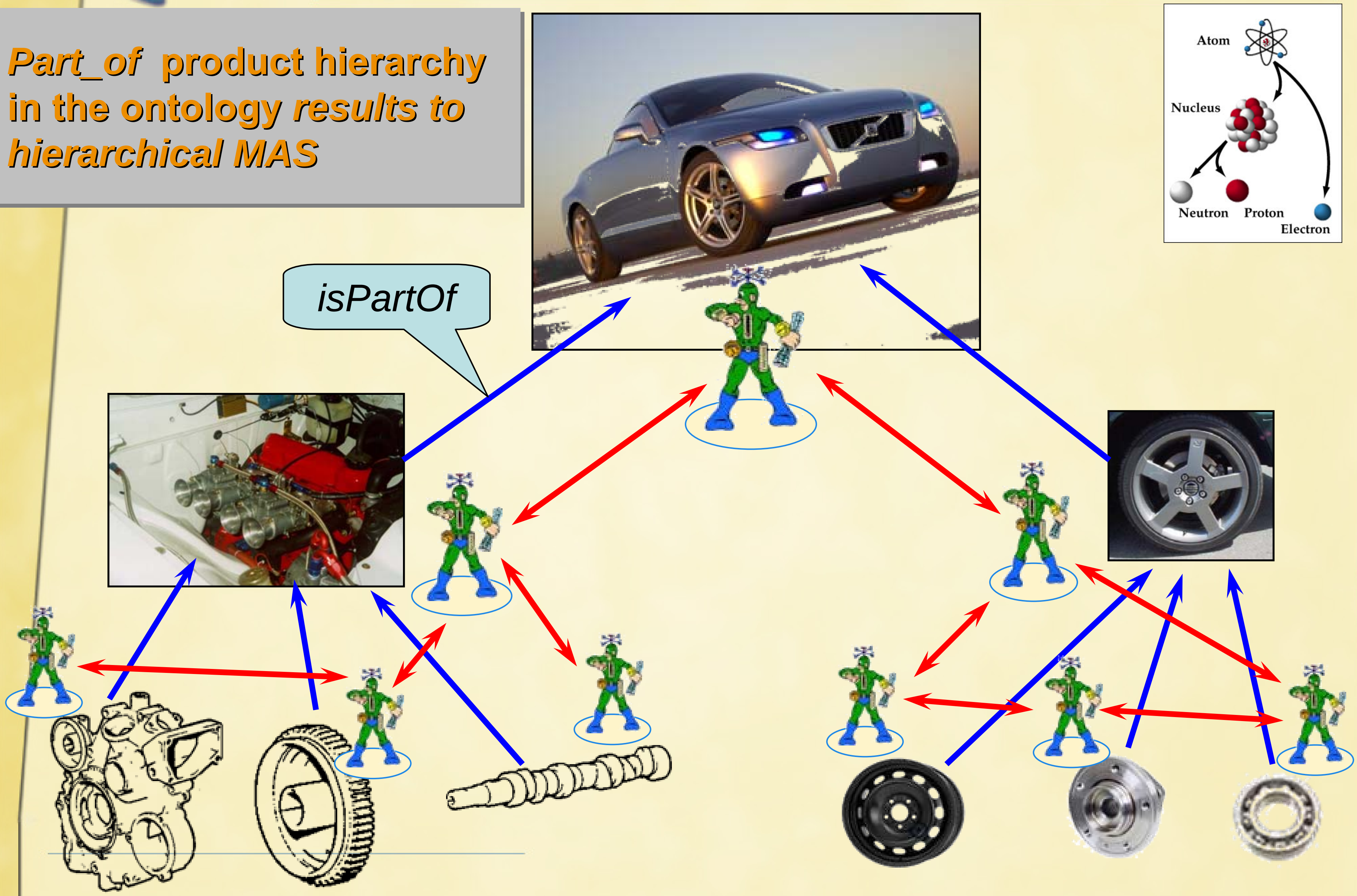


CONFaaS: Configuration-as-a-Service

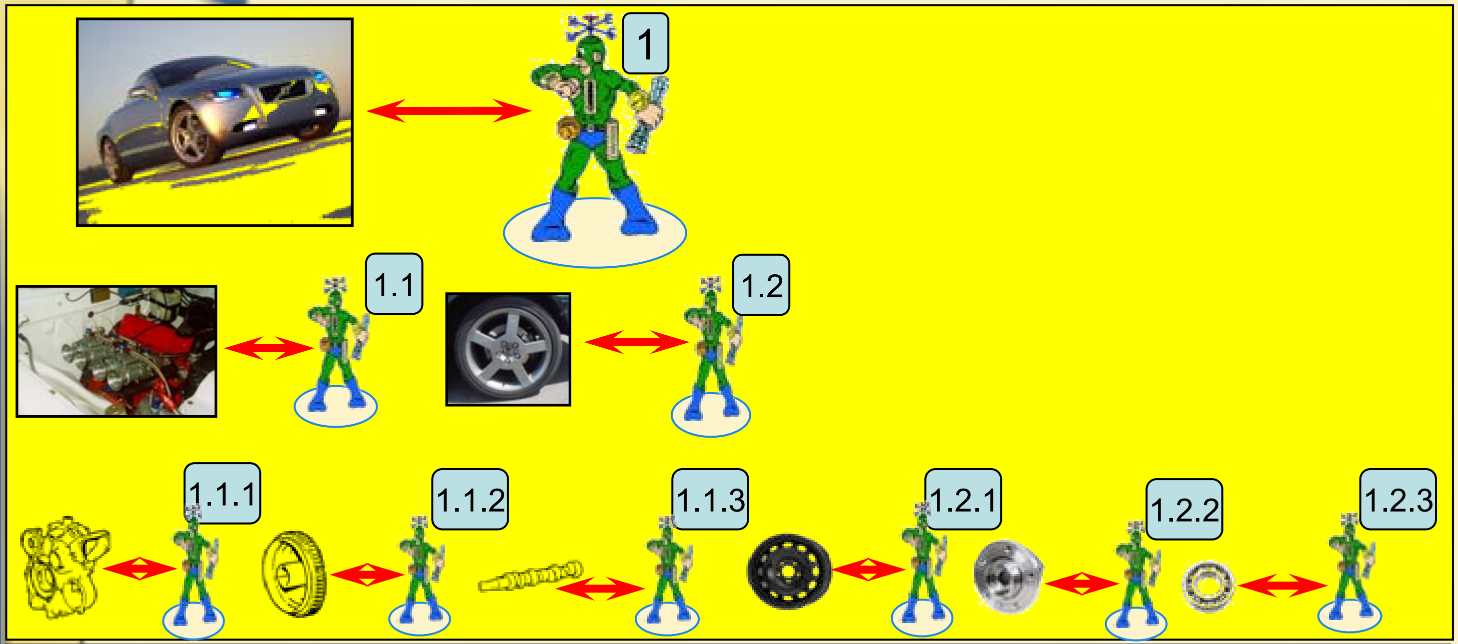
CONFaaU: Configuration-as-a-User

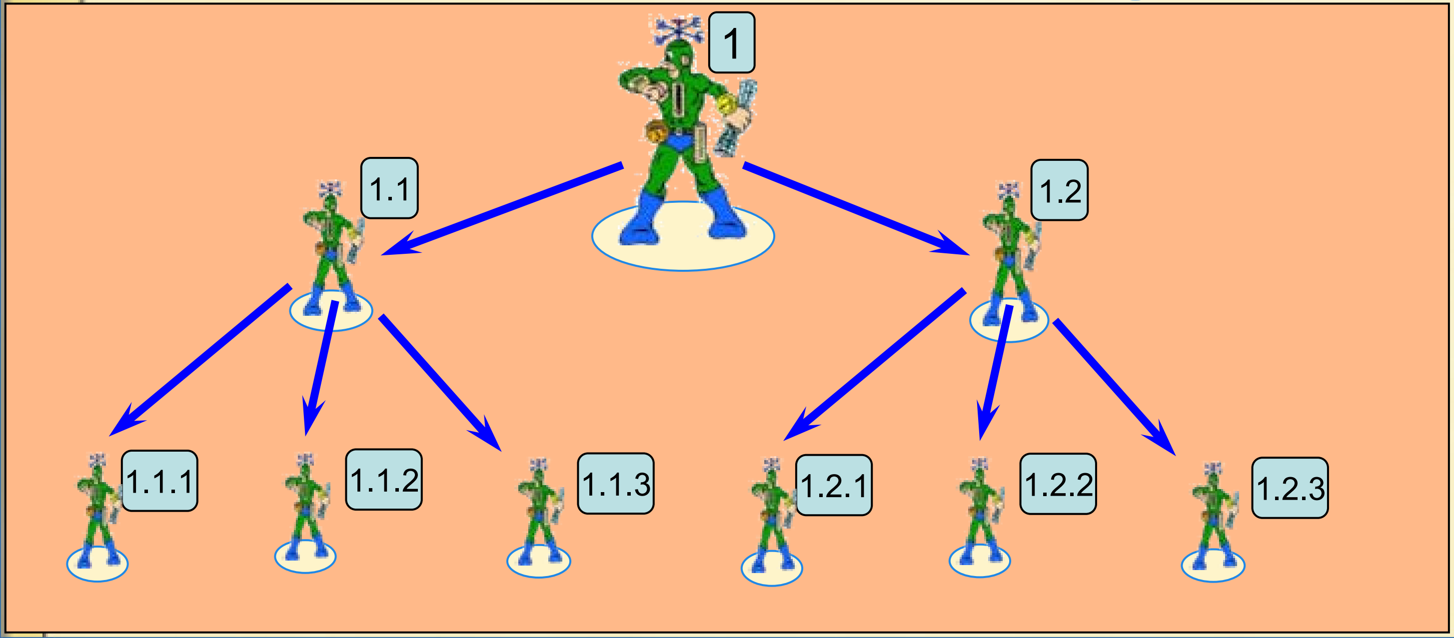
Proactive Configuration

Part_of **product hierarchy in the ontology *results to hierarchical MAS***



Configuration of objects vs processes





Axiom 1: *Each resource in dynamic Industrial World is a process and each process in this world is a resource.*

Axiom 2: *Hierarchy of subordination among resource agents in GUN corresponds to the "part-of" hierarchy of the Industrial World resources.*

Resource Configuration Example

ID1

hasConfiguration (ID1,ID2)

Train (ID1)

hasPart (ID1,ID3)

hasPart (ID1, ID4)

hasPart (ID1, ID5)

hasDestinationTo (ID1, "Paris")

hasDestinationFrom (ID1, "Amsterdam")

hasConfiguration (ID3,ID6)

hasConfiguration (ID4, ID7)

hasConfiguratioin (ID5, ID8)

ID2

ID3

Locomotive (ID3)

hasColor (ID3, "Muticolor")

hasBehind (ID3, ID4)

ID6

ID4

Car (ID4)

hasColor (ID4, "Beige")

hasBehind (ID4, ID5)

hasAhead (ID4, ID3)

ID7

ID5

Car (ID5)

hasColor (ID5, "Red")

hasAhead (ID5, ID4)

ID8

Configuration Components

Object of configuration

ID1

hasConfiguration (ID1,ID2)

Content of configuration

ID2:

Train (ID1)
hasPart (ID1,ID3)
hasPart (ID1, ID4)
hasPart (ID1, ID5)
hasDestinationTo (ID1, "Paris")
hasDestinationFrom (ID1, "Amsterdam")
hasConfiguration (ID3,ID6)
hasConfiguration (ID4, ID7)
hasConfiguratioin (ID5, ID8)

Class of the resource

Structure of the resource

Parameters' values of the resource

Configuration of structural components

Reconfiguration

ID1

hasConfiguration (ID1,ID2)

ID2:

Train	(ID1)
hasPart	(ID1,ID3)
hasPart	(ID1, ID4)
hasPart	(ID1, ID5)
hasDestinationTo	(ID1, “Paris”)
hasDestinationFrom	(ID1, “Amsterdam”)
hasConfiguration	(ID3,ID6)
hasConfiguration	(ID4, ID7)
hasConfiguratioin	(ID5, ID8)

ID1

hasConfiguration (ID1,ID8)

ID8:

Train	(ID1)
hasPart	(ID1,ID3)
hasPart	(ID1, ID4)
hasPart	(ID1, ID5)
hasDestinationTo	(ID1, “Paris”)
hasDestinationFrom	(ID1, “Amsterdam”)
hasConfiguration	(ID3,ID9)
hasConfiguration	(ID4, ID10)
hasConfiguratioin	(ID5, ID11)

Reconfiguration behavior (option 1: reordering)

ID1

ID1

hasConfiguration (ID1,ID8)

ID8:

Train (ID1)

hasPart (ID1,ID3)

hasPart (ID1, ID4)

hasPart (ID1, ID5)

hasDestinationTo (ID1, “Paris”)

hasDestinationFrom (ID1, “Amsterdam”)

hasConfiguration (ID3,ID9)

hasConfiguration (ID4, ID10)

hasConfiguratioin (ID5, ID11)

ID3

Locomotive (ID3)

hasColor (ID3, “Muticolor”)

hasBehind (ID3, ID5)

ID9

ID4

Car (ID4)

hasColor (ID4, “Beige”)

hasAhead (ID4, ID5)

ID10

ID5

Car (ID5)

hasColor (ID5, “Red”)

hasAhead (ID5, ID3)

hasBehind (ID5, ID4)

ID11

Reconfiguration behavior (option 2: recolor)

ID1

ID1

hasConfiguration (ID1,ID12)

ID12:

Train	(ID1)
hasPart (ID1,ID3)	
hasPart (ID1, ID4)	
hasPart (ID1, ID5)	
hasDestinationTo (ID1, "Paris")	
hasDestinationFrom (ID1, "Amsterdam")	
hasConfiguration (ID3,ID6)	
hasConfiguration (ID4, ID13)	
hasConfiguratioin (ID5, ID14)	

ID3

ID6:

Locomotive	(ID3)
hasColor (ID3, "Muticolor")	
hasBehind (ID3, ID4)	

ID4

ID13:

Car	(ID4)
hasColor (ID4, "Red")	
hasAhead (ID4, ID3)	
hasBehind (ID4, ID5)	

ID5

ID14:

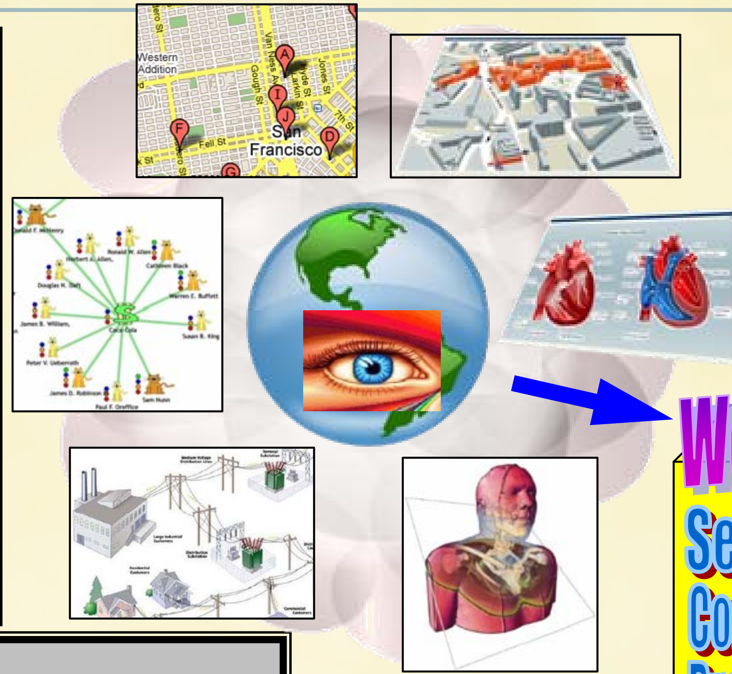
Car	(ID5)
hasColor (ID5, "Beige")	
hasAhead (ID5, ID4)	

1.11. Web 4.4

Web of Presentations

Web of Presentations

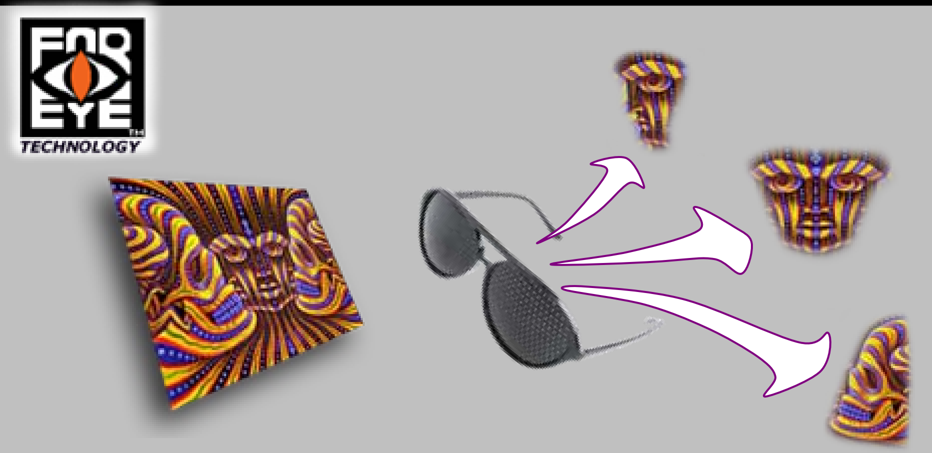
… is the Web of visualization providers (or "metaproviders" according to 4i ("for-eye") technology). The same content (either static or dynamic, homogeneous or heterogeneous) will be presented (and if needed also filtered and mashed) by different ways by different visualization providers .



Facilitates Presentation-to-Presentation interaction

Web of Presentations

Semantic Mash-Ups
Context-based presentation
Presentations Web Browser
Cloud of Visualizators
Proactive Visualization

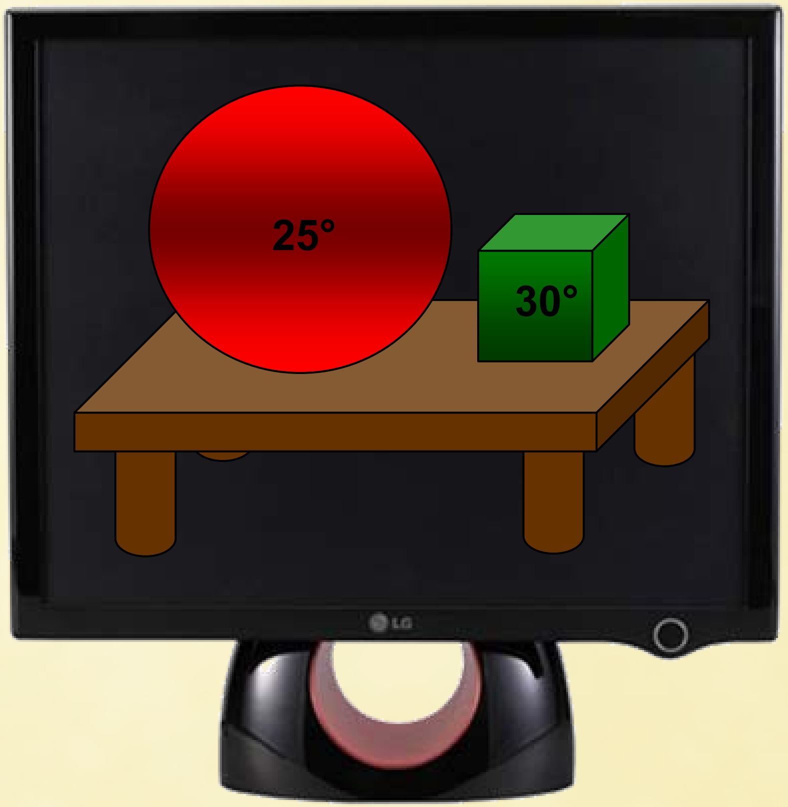


VISaaS: Visualization-as-a-Service
VISaaU: Visualization-as-a-User

This is not simple

RDF/OWL

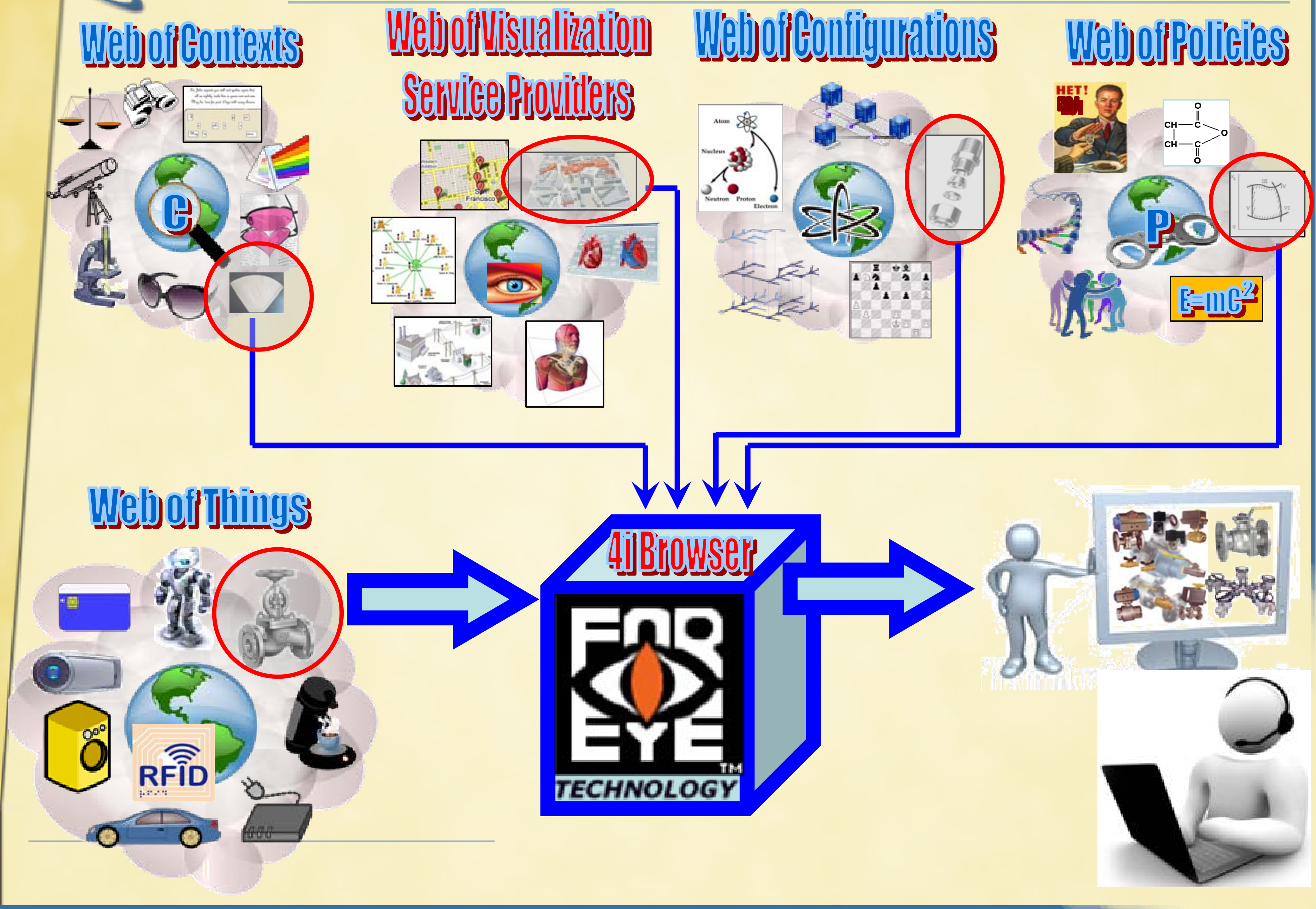
Cube (ID1)
Ball (ID2)
Table (ID3)
hasColor (ID1, “Green”)
hasColor (ID2, “Red”)
hasColor (ID3, “Brown”)
isOnTheLeftSideOf (ID2, ID1)
hasTemperatureC (ID1, 30)
hasTemperatureC (ID2, 25)
isOn (ID1, ID3)
isOn (ID2, ID3)
isLarger (ID2, ID1)



What is 4i ("for-eye") technology?

- **4i** – is smart ontology-based visualization technology able to automatically discover and utilize external visualization service providers and dynamically create and visualize mashups from external data sources in a context-driven way

4i Philosophy: Visualization-as-a-Service



Context - *is a filter of resource representation (visualization)… Depending on a context it might happen that only some of resource properties (some properties of other relevant resources) are relevant and should be visualized to the user to avoid avalanche of irrelevant information and make a stress (highlight) only on important stuff.*

Semantically enhanced context-based multidimensional
Resource Visualization

The visualization of a "human heart" resource in a context of its internal condition can be introduced in a form of internal structure of the heart and its functional parts.

"Human heart" resource in a context of its condition in relation to other human body systems can be visualized as a part of an internal structure of a human body.

The visualization of a "person" resource in a context of healthcare & condition of one's organs can be performed in a way of human body diagram (with a view of the organs).

At the same time, "person" resource in a context of healthcare & location of a healthcare organization can be visualized in a form of a map.

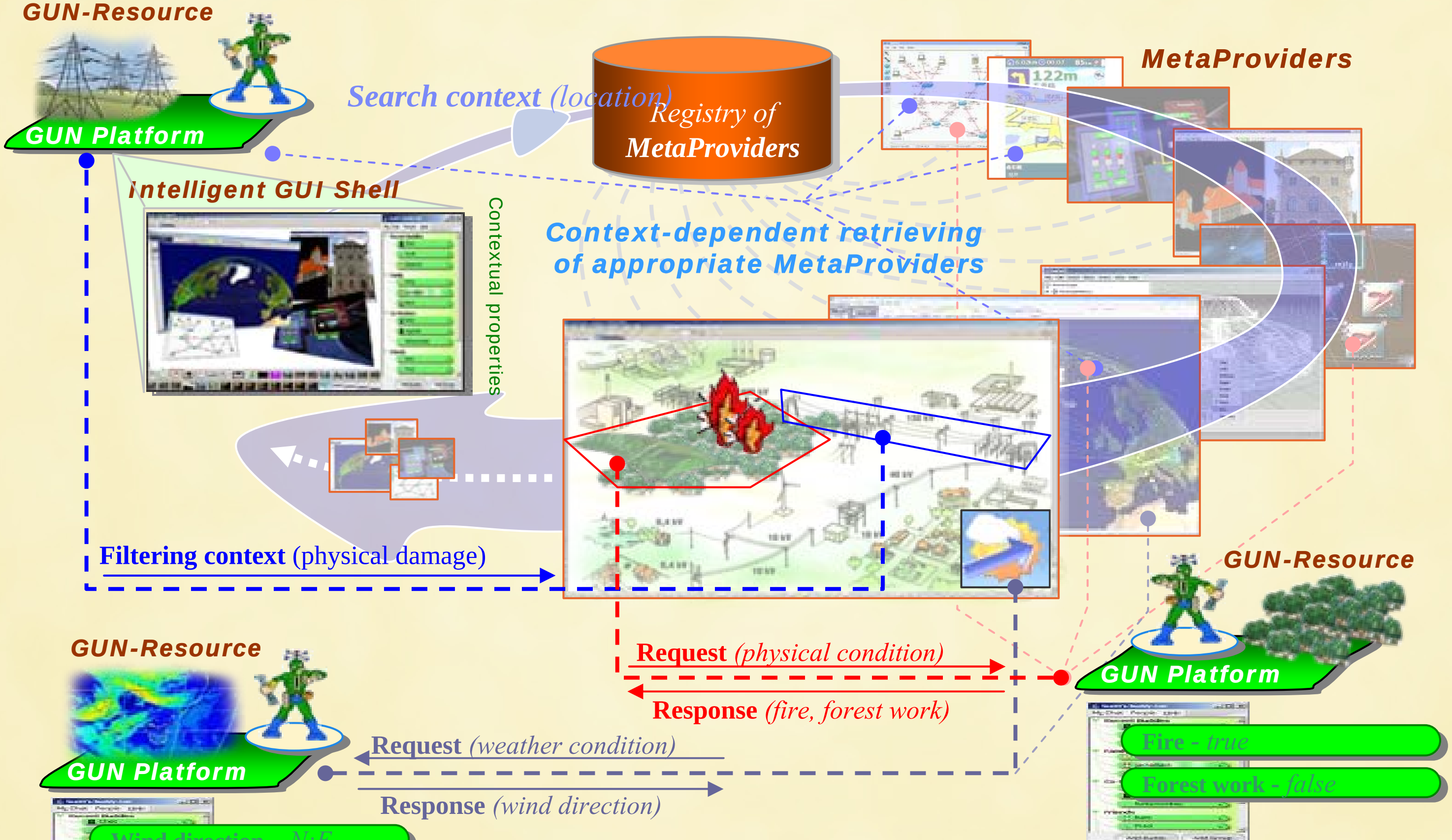
The occupation/profession-based visualization of a "person" resource. Visualization of a working area with the relevant work-related links: duties, area of interests, professional related resources, contacts, etc.

The visualization of a "person" resource in a context of family relations can be displayed in a form of family tree visualization.

Smart Interface

INTELLIGENT INTERFACE for INTEGRATED INFORMATION: 4i (FOR EYE) Technology™

4i (FOR EYE) *is an ensemble of Intelligent GUI Shell (smart middleware for context dependent use and combination of a variety of different MetaProviders depending on user needs) and MetaProviders, visualization modules (remote services) that provide context-dependent filtered representation of information (resource data).*

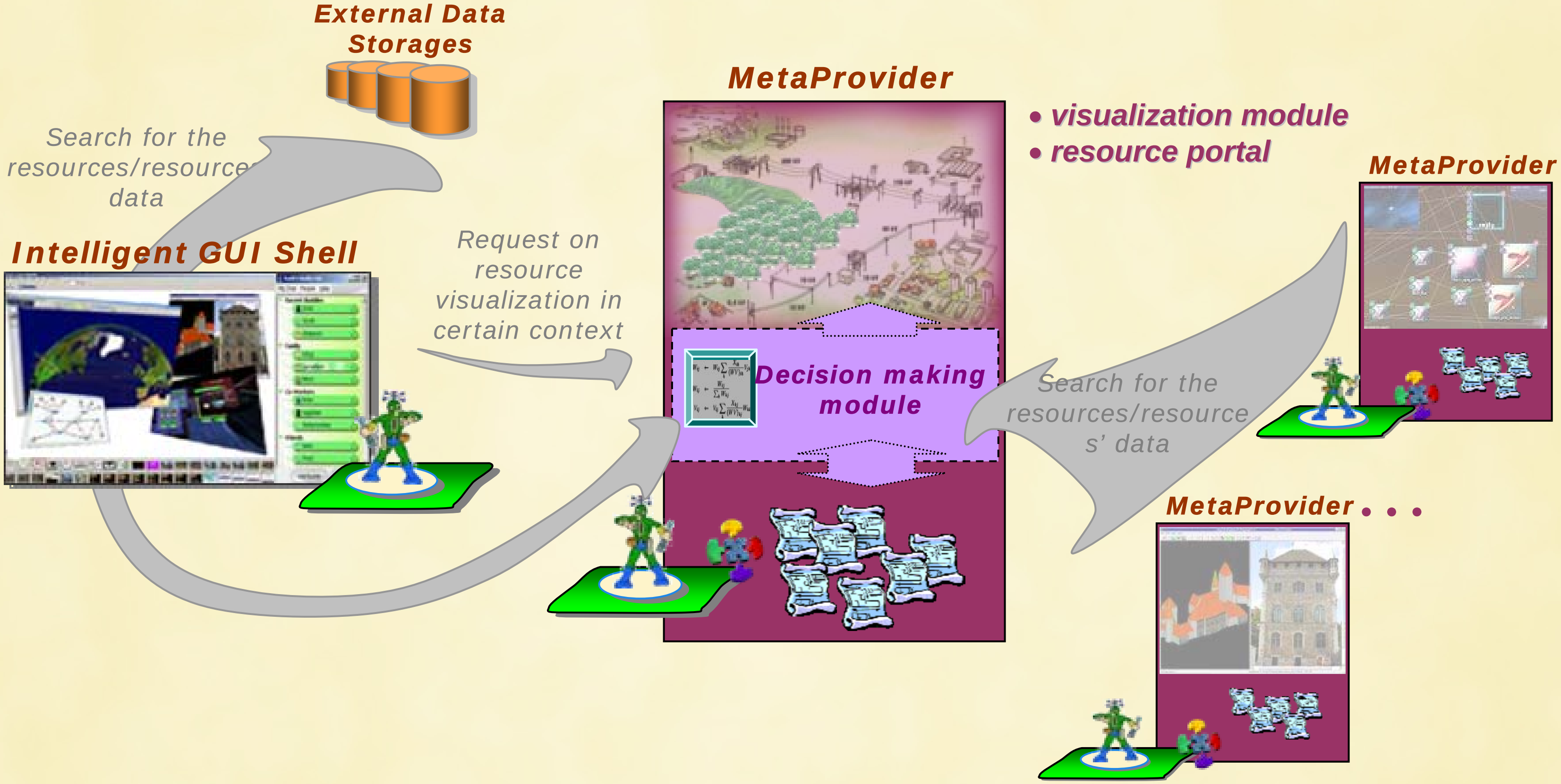


Following the new technological trends, it is time to start a new stage in user visual interface development – a stage of semantic-based context-dependent multidimensional resource visualization.

FOR EYE TECHNOLOGY™

4i "MetaProvider" architecture

Context-driven data filtering and visualization



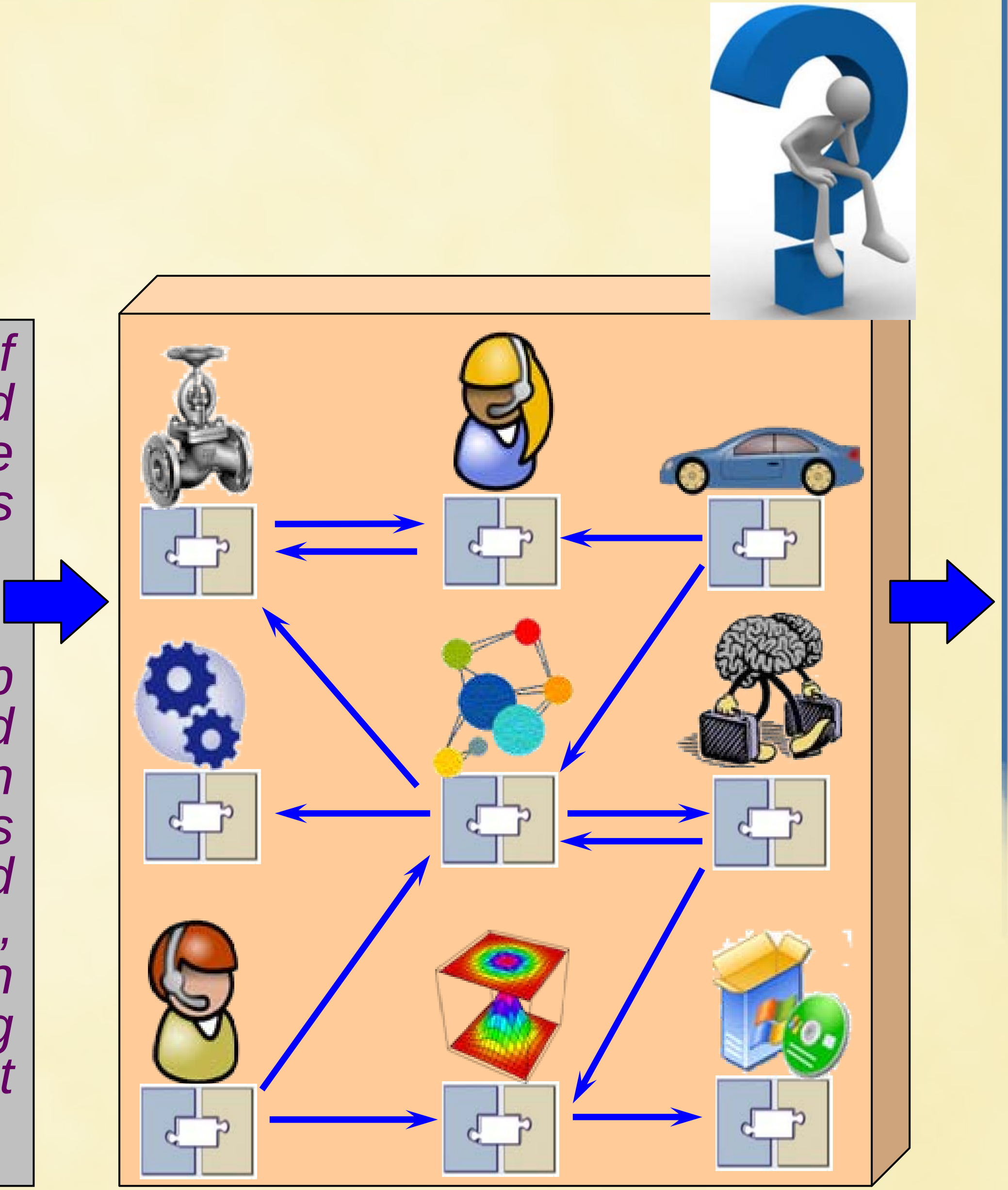
1.12. Web 5.0

Global Understanding Environment

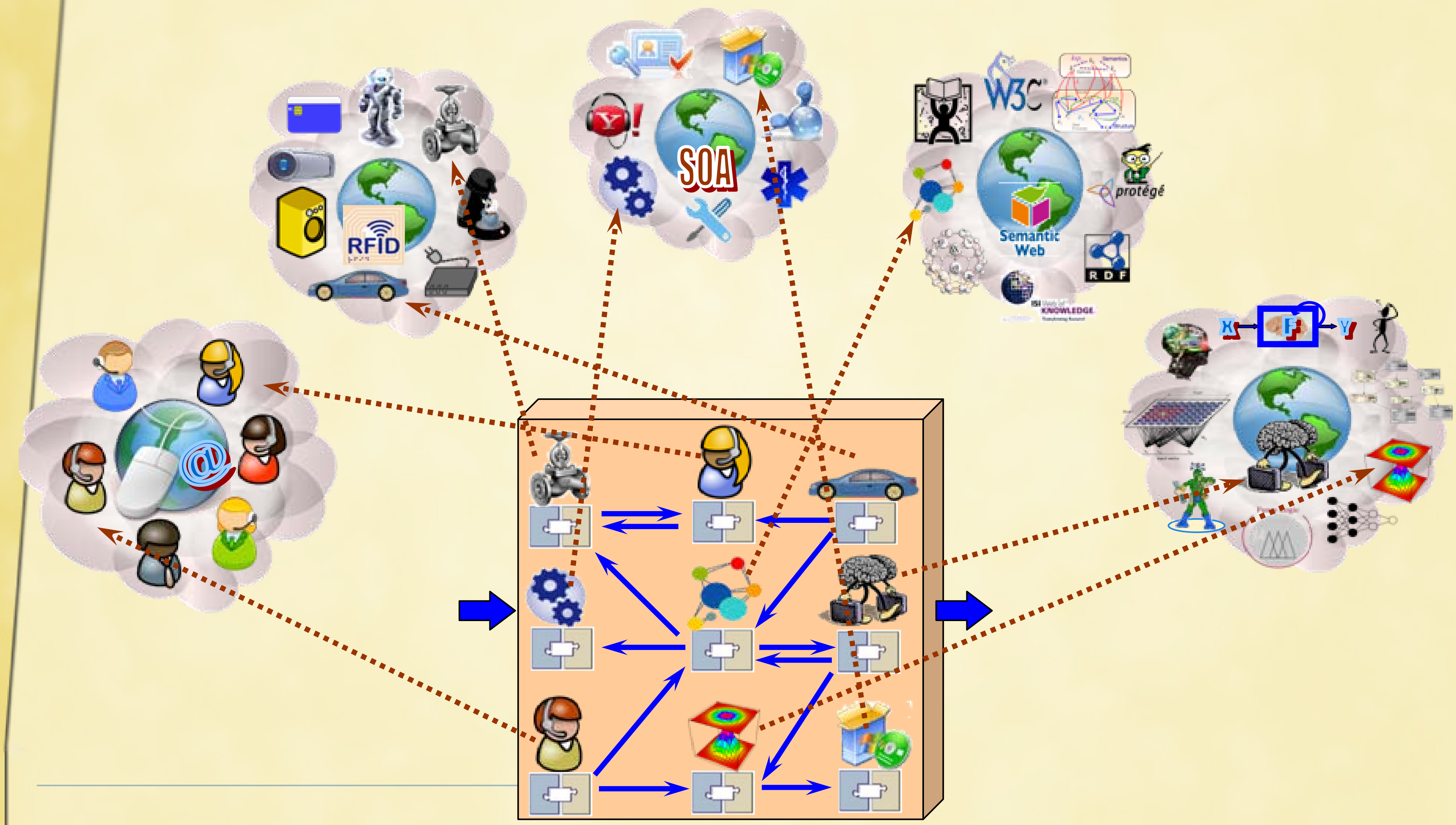
Still not enough?

- According to these visions of future Web, interoperability and collaboration will be possible only within mentioned groups of resources.

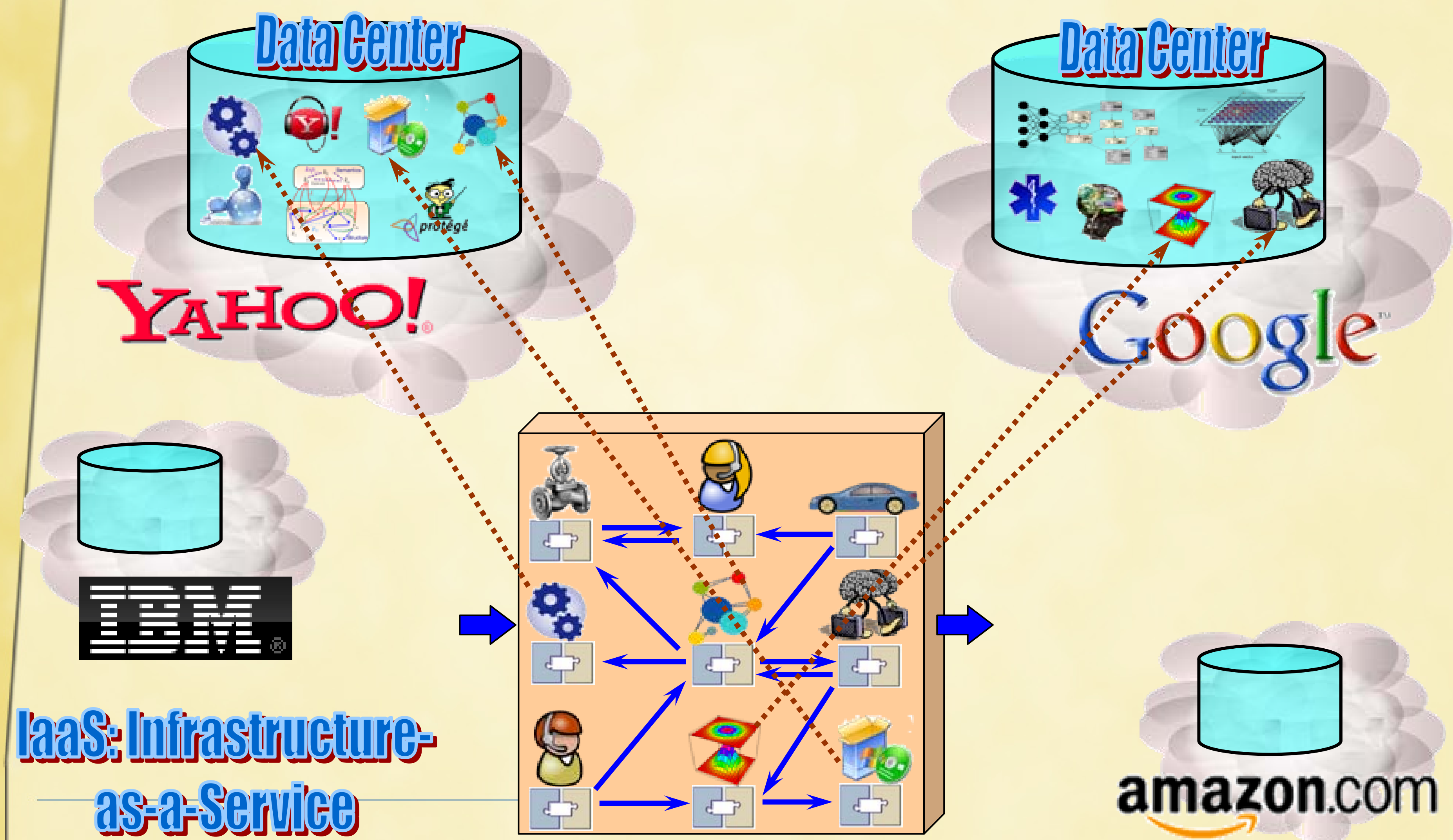
- However future Web applications and Web-based systems will contain heterogeneous components and therefore will demand support for integration, interoperability, collaboration and mutual service provisioning between resources of different types.



Components of a modern system are not only highly heterogeneous but also globally distributed (SOA) …

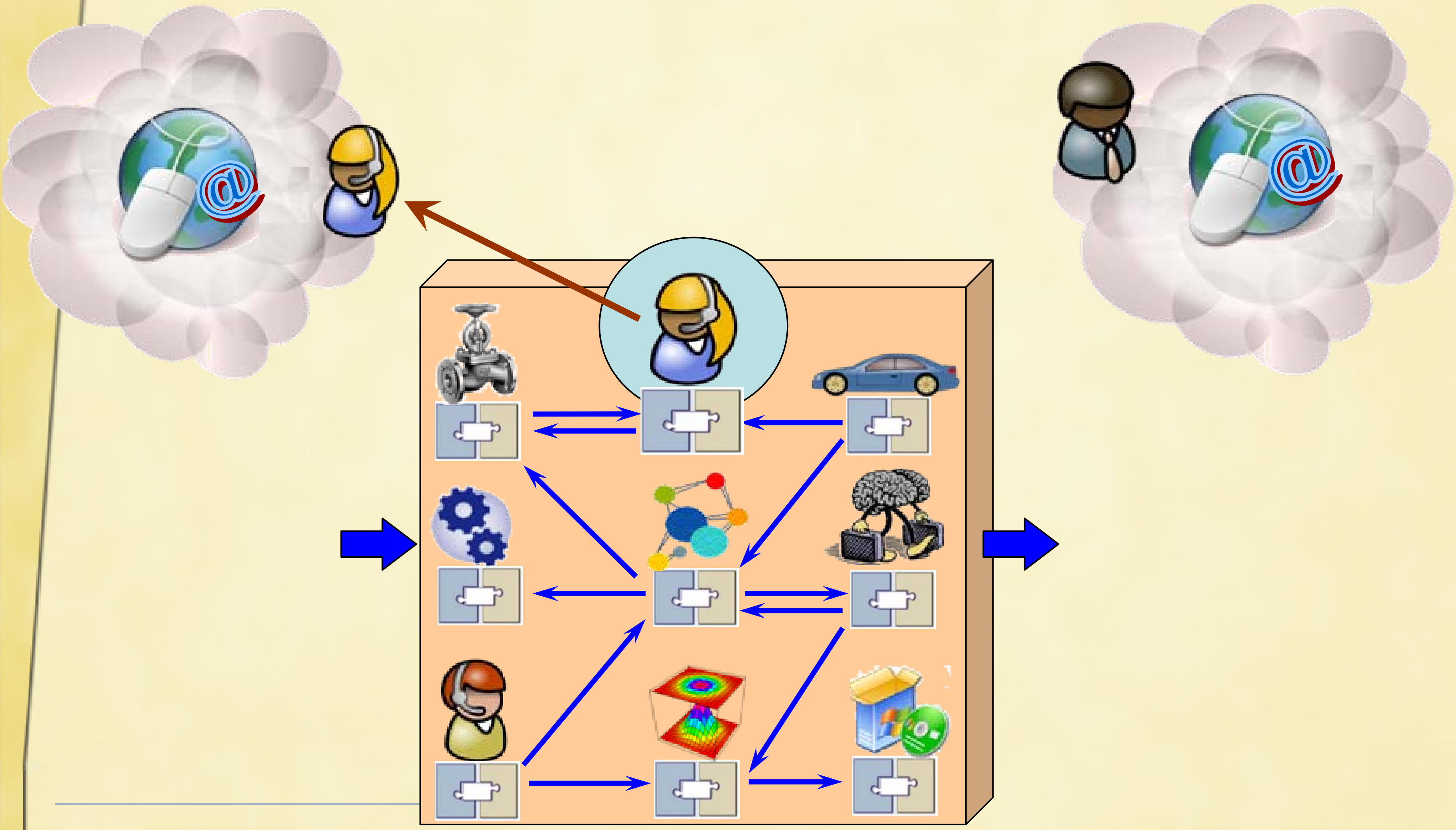


… or some of the components may be concentrated in huge data centers (Cloud Computing)

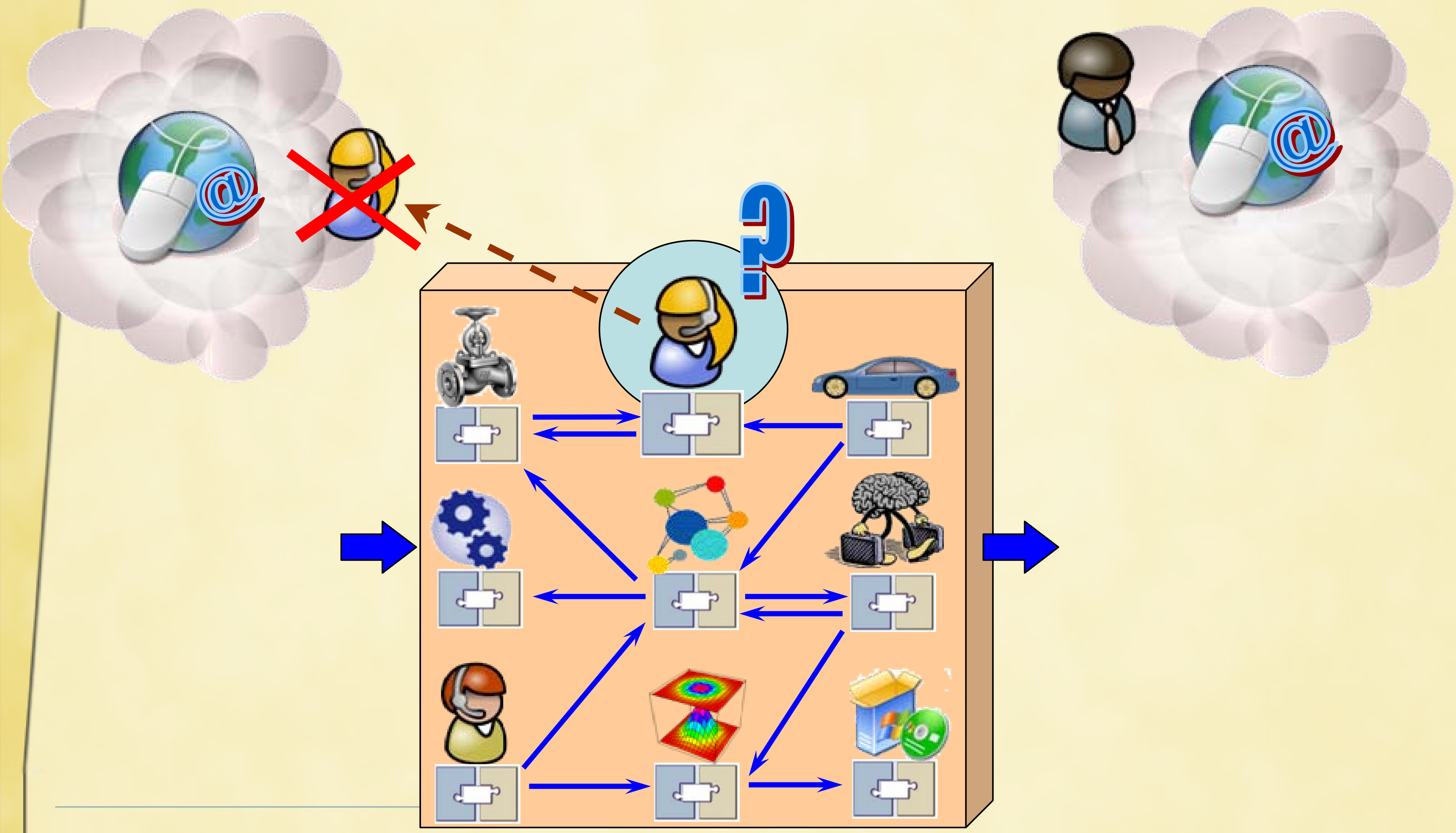


IaaS: Infrastructure-as-a-Service

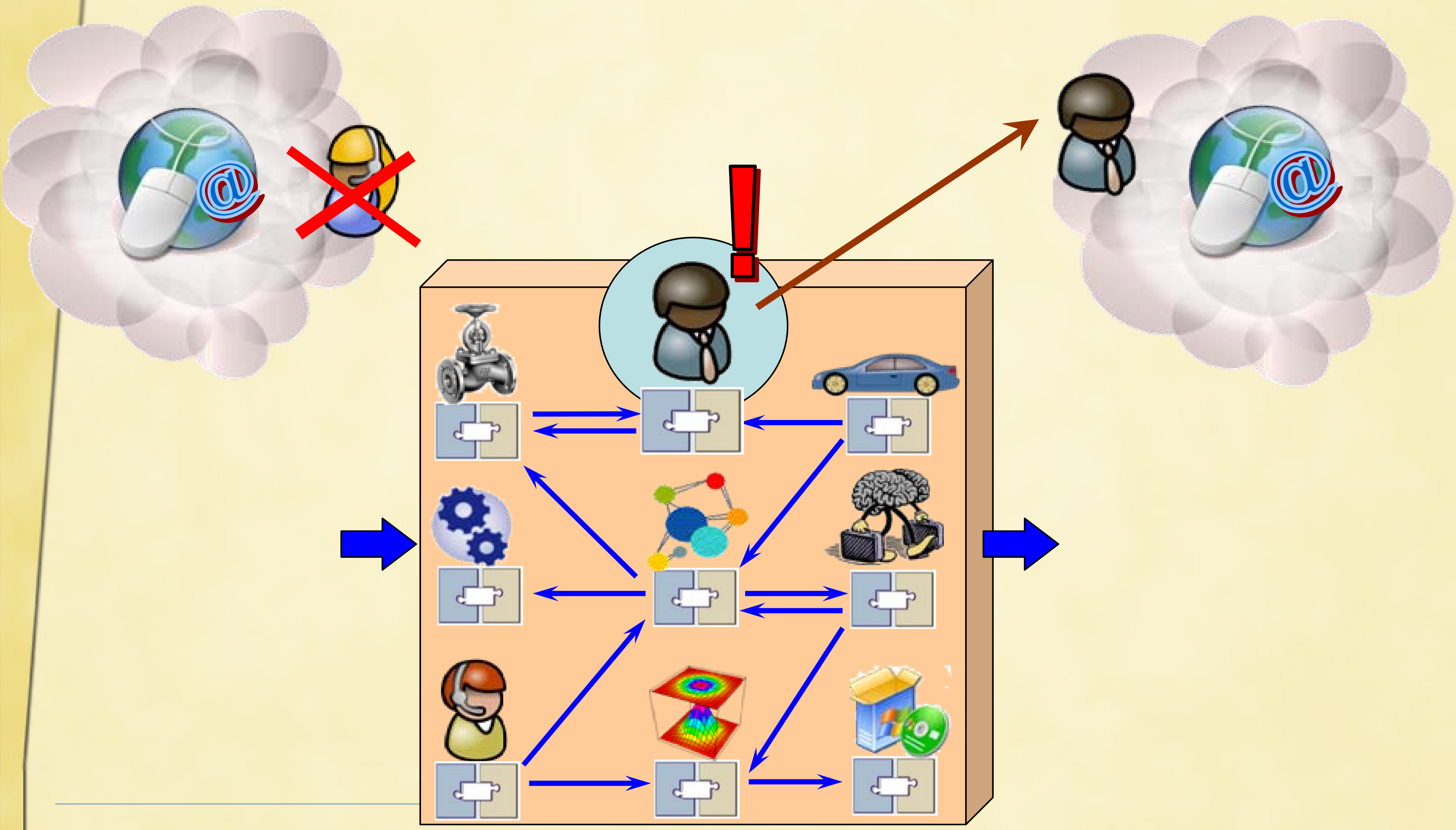
A system should be open and ready to reconfigure itself when needed (1)



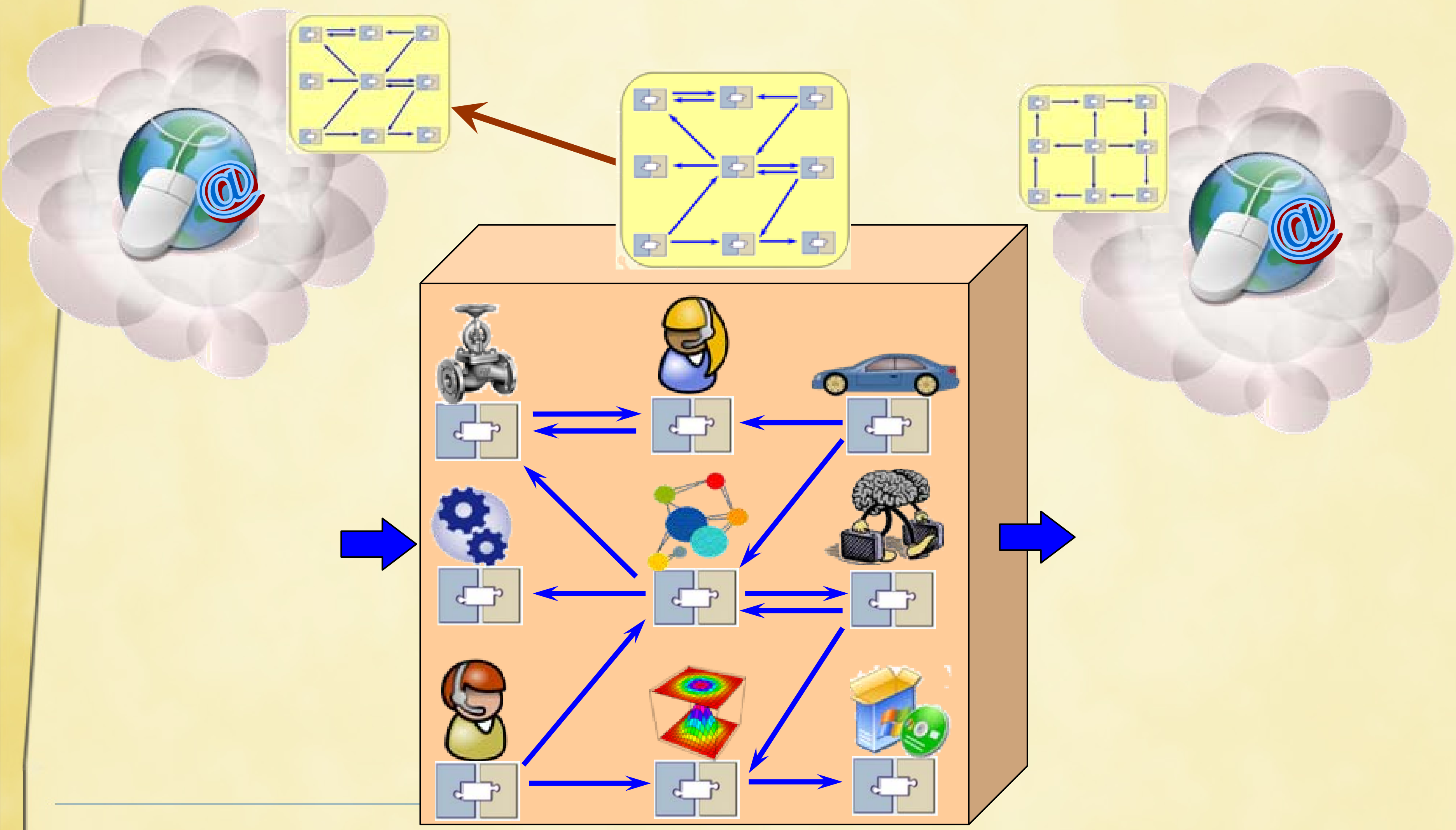
A system should be open and ready to reconfigure itself when needed (2)



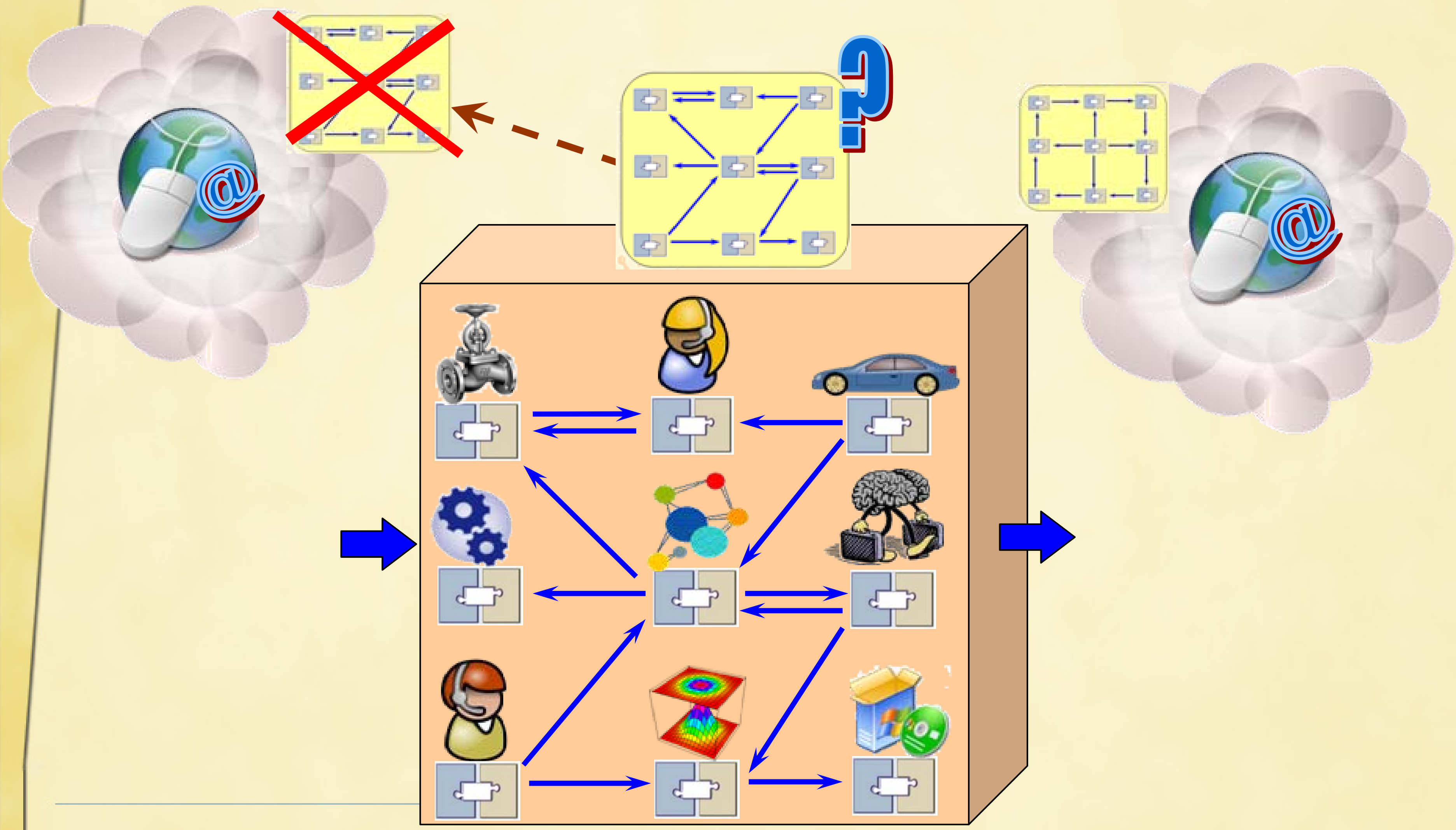
A system should be open and ready to reconfigure itself when needed (3)



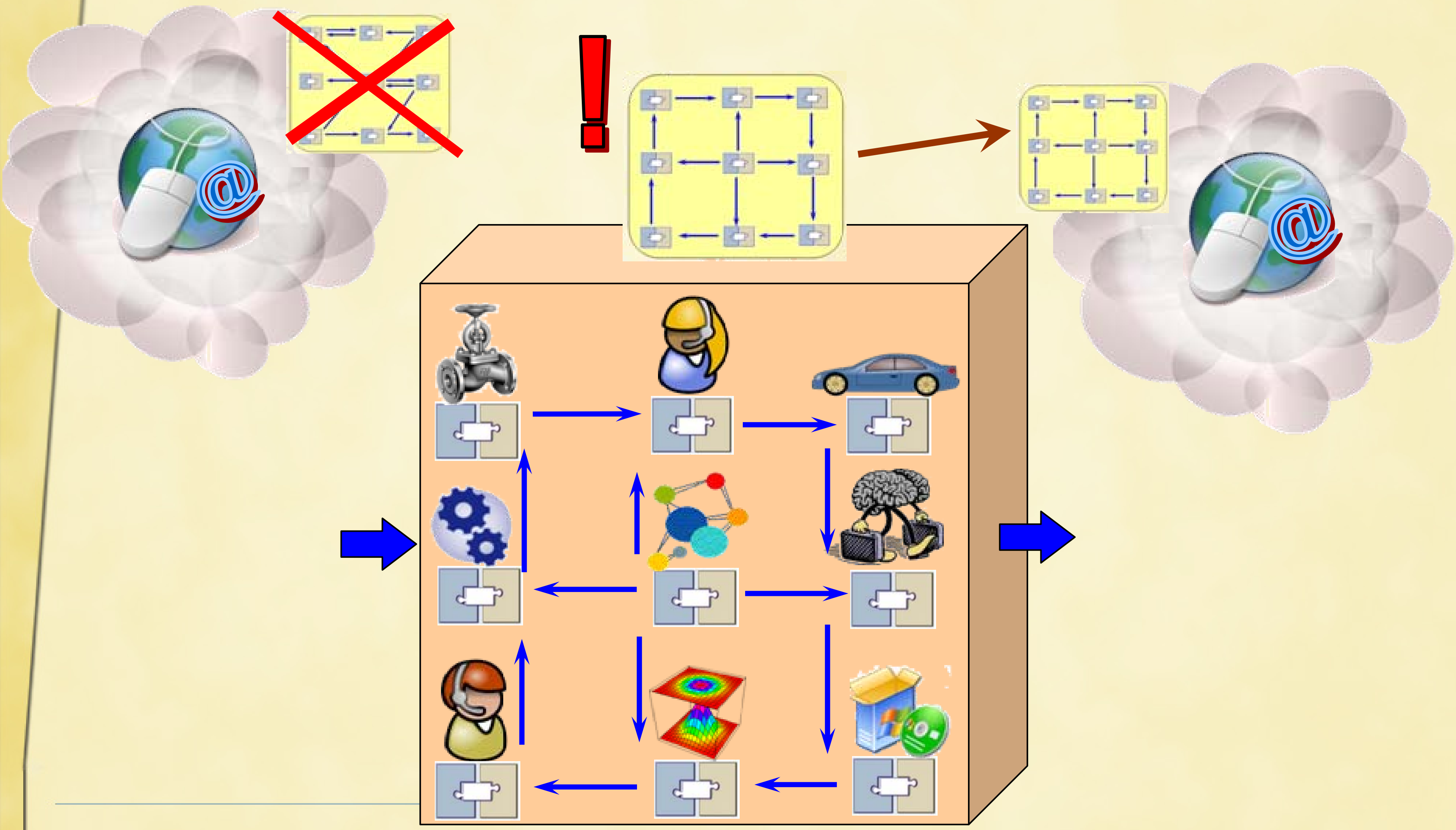
Even a business logic of a system can be imported and reconfigured on-the-fly (1)



Even a business logic of a system can be imported and reconfigured on-the-fly (2)

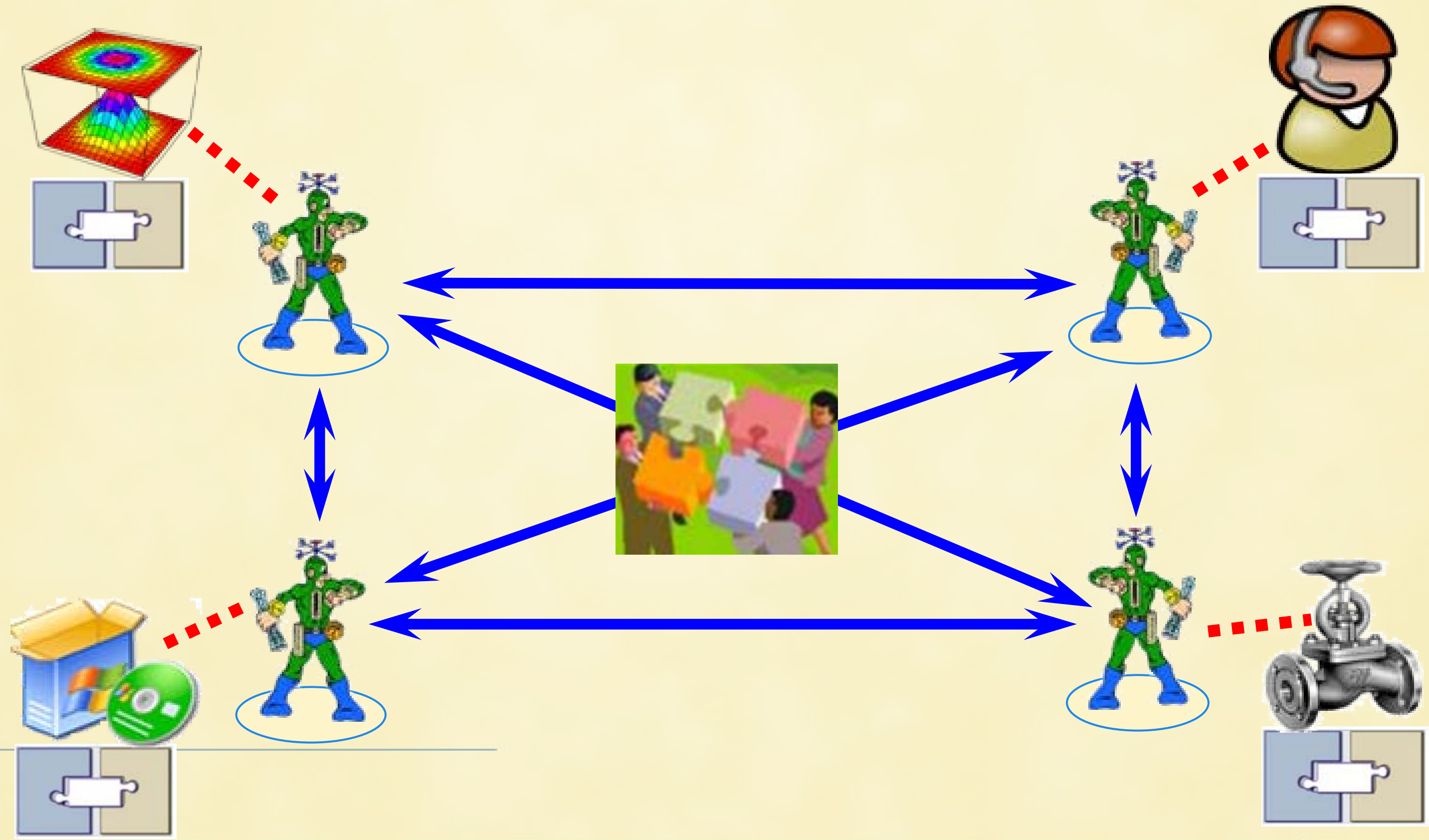


Even a business logic of a system can be imported and reconfigured on-the-fly (3)



Agents are needed !

Adding a ***"virtual representative"*** to every resource solves the global interoperability problem. Intelligent **agent** (a kind of "software robot") will act, communicate and collaborate on behalf of each Web resource



GUN Concept (Industrial Ontologies Group)

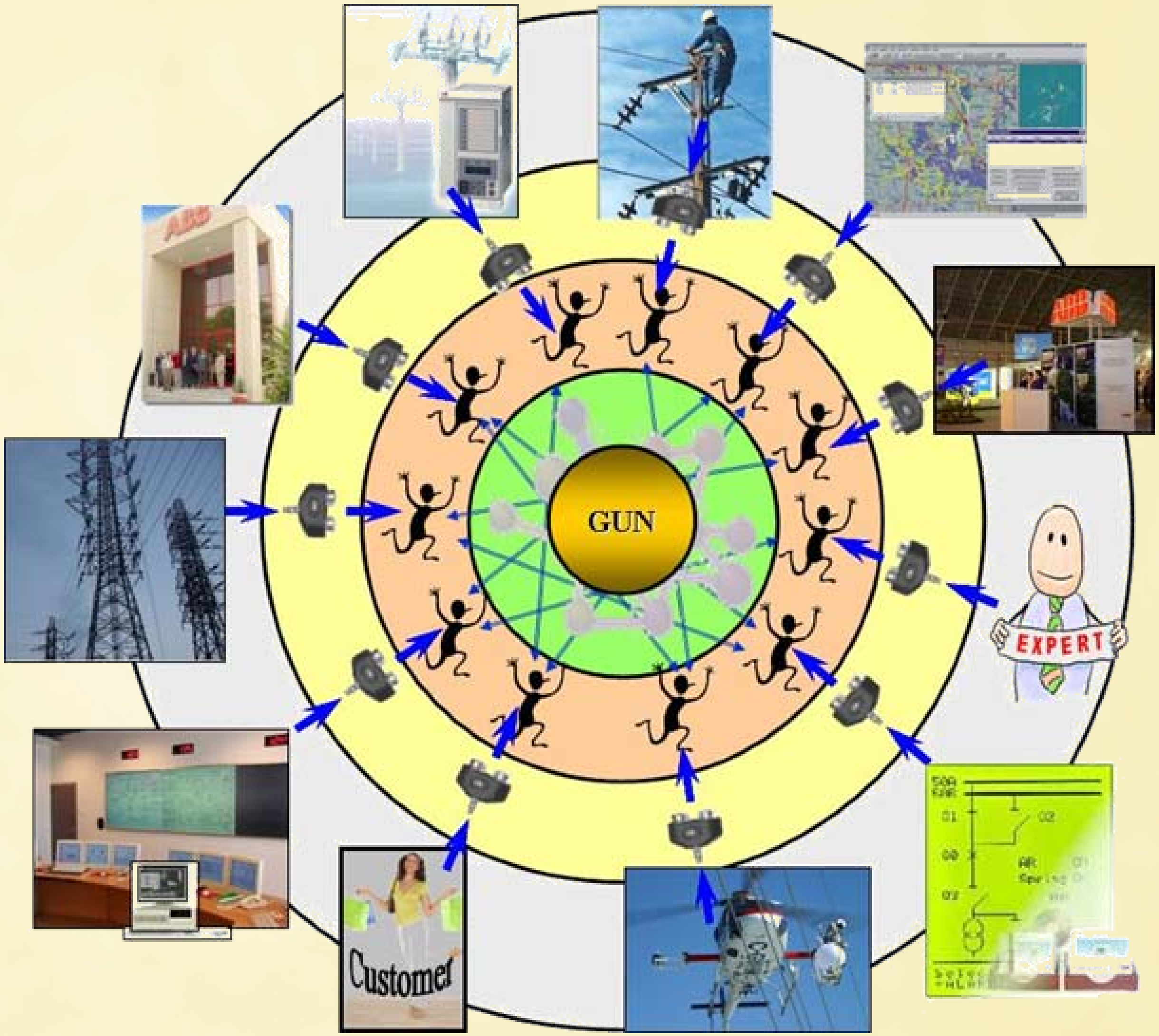
GUN – **G**lobal **U**nderstanding e**N**vironment

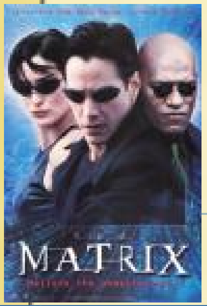
GUN

=

Global Environment
+
Global Understanding
=

Proactive Self-Managed Semantic Web of Everything





http://www.mit.jyu.fi/ai/OntoGroup/projects.htm
http://www.mit.jyu.fi/ai/Industrial_Ontologies_Group_booklet_print.doc

Global Understanding Environment (GUN)

GUN can be considered as a kind of Ubiquitous Eco-System for Ubiquitous Society, which will be such proactive, self-managed evolutionary Semantic Web of Things, People and Abstractions where all kinds of entities can understand, interact, serve, develop and learn from each other.

- Human-to-Human
- Human-to-Machine
- Machine-to-Human
- Machine-to-Machine
- Software-to-Human
- Software-to-Machine
- Software-to-Software
- Human-to-Software
- …

} Agent-to-Agent

“Semantic Wave” (Web X.0)

We may add here:

***Web 5.0** will come finally and it is about **connecting models** in a “**Global Understanding Environment**” **(GUN),** which will be such proactive, self-managed evolutionary Semantic Web of Things, People and Abstractions where all kinds of entities can understand, interact, serve, develop and learn from each other.*

[Vagan Terziyan]

*“The **semantic wave** embraces four stages of internet growth:*

***Web 1.0**, was about **connecting information** …*

***Web 2.0** is about **connecting people**.*

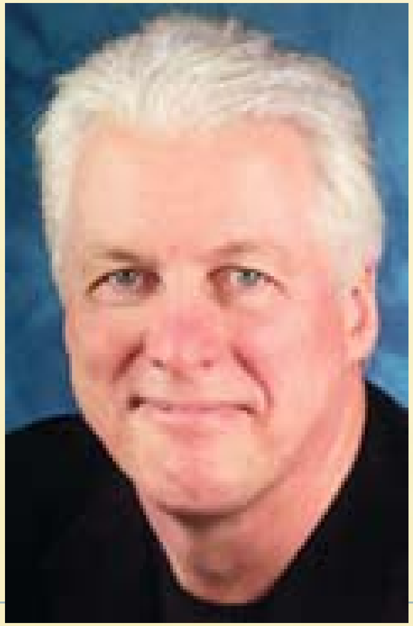
***Web 3.0**, is starting now… and it is about … **connecting knowledge**…*

***Web 4.0** will come later … and it is about **connecting intelligences** in a ubiquitous web where both people and things can reason and communicate together.”*

[**“Semantic Wave 2008”** , **Mills Davis**]

Positive feedback on GUN from "Semantic Wave" father

- From: Mills Davis <project10x@gmail.com>
  To: Vagan Terziyan <vagan@cc.jyu.fi>
  Subject: Design of Agent-Based Systems
  Date: Sat, 15 Nov 2008 12:50:06 -0500
- *"Vagan,*
  Just came across your course presentation on design of agent-based systems. I very much enjoyed your presentation of GUN concepts."
- *Mills Davis*



Mills Davis is Founder and Managing Director of Project10X — a research consultancy specializing in next wave semantic technologies, solutions, and business models. The firm's clients include technology manufacturers, global 2000 corporations, government agencies, and next-generation web start-ups. Mills serves as principal investigator for the Semantic Wave 2008 research program. A noted consultant and industry analyst, he has authored more than 100 reports, white papers, articles, and industry studies. Mills is active in both government and industry-wide technology initiatives that are advancing semantic technologies. He co-chairs SemanticCommunity.net, which carries on the mission the Federal Semantic Interoperability Community of Practice (SICoP) in supporting Communities of Interest in both government and private industry. Mills is a founding member of the AIIM interoperable enterprise content management (iECM) working group, and a founding member of the National Center for Ontology Research (NCOR). Also, he serves on the advisory board of several new ventures in the semantic space.

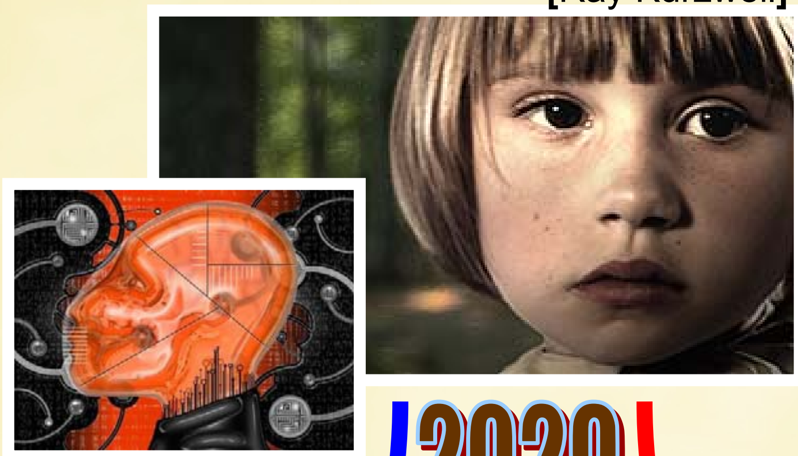
1.13. Beyond Web 5.0

Human 2.0 ?

Beyond Web 5.0 ?

Human v2.0 ?!

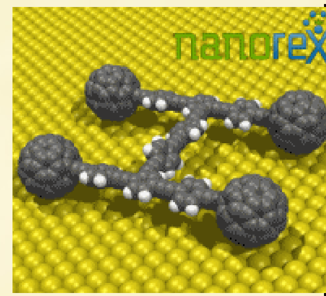
[Ray Kurzweil]



Wireless Brain-Computer Interface

Brain-to-Brain (B2B) Communication

Nanobots



2029

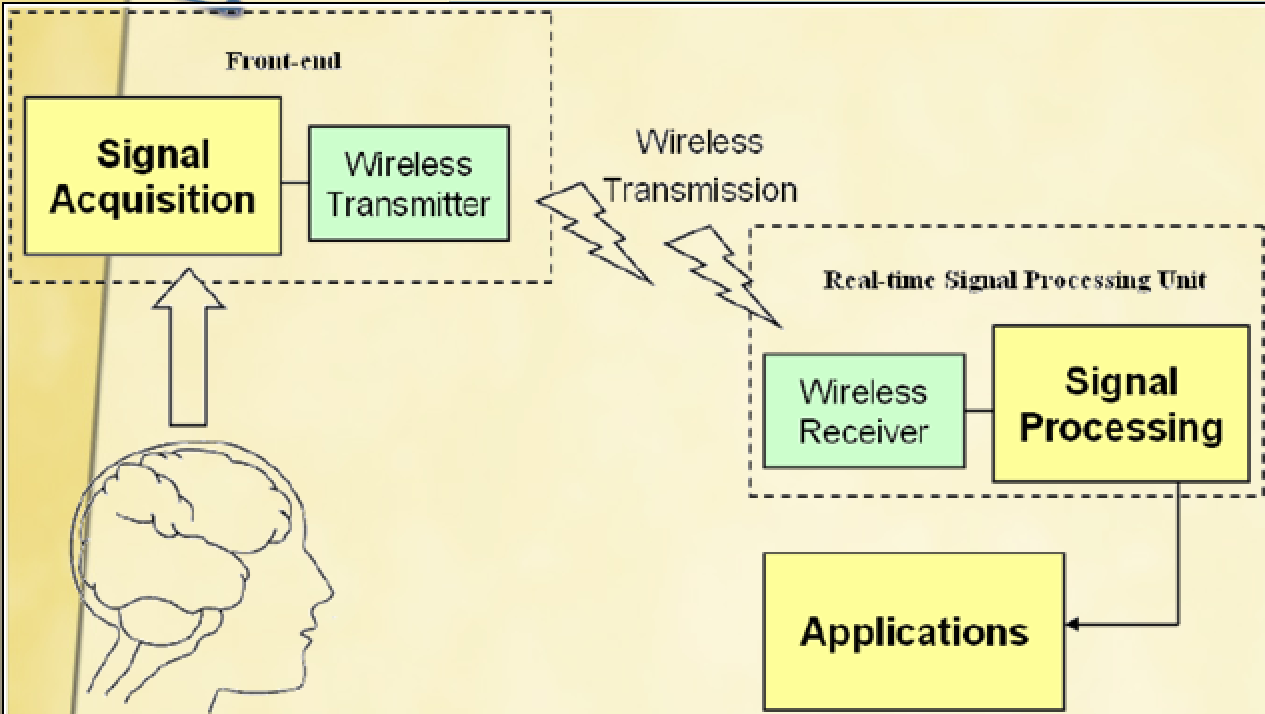
Singularity

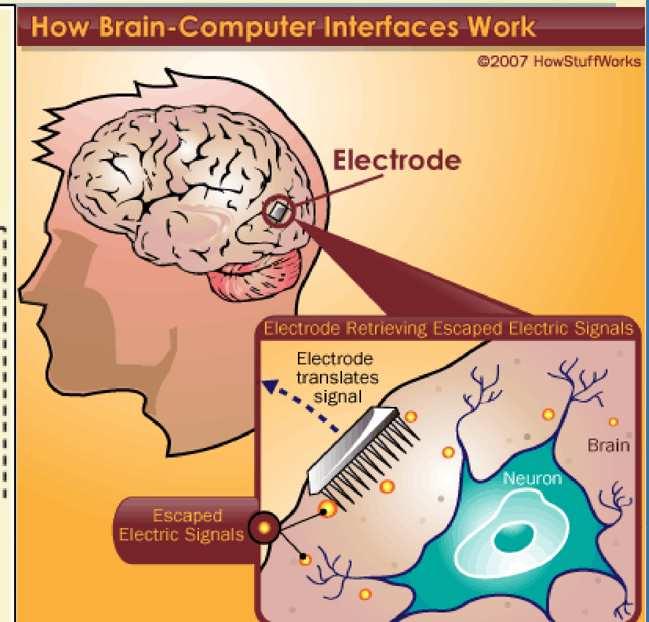
Semantic Wave

Nanotech

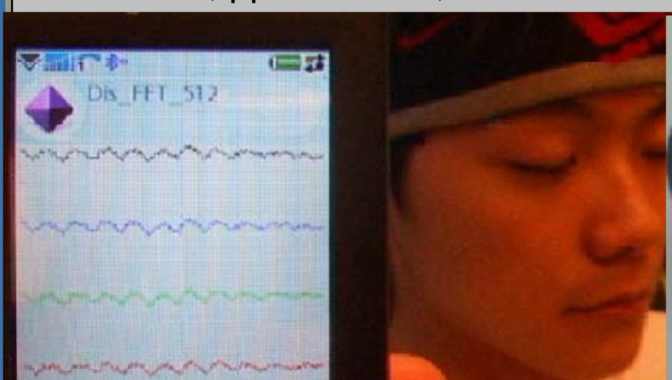
http://www.youtube.com/watch?v=BywCMkbG-Jg

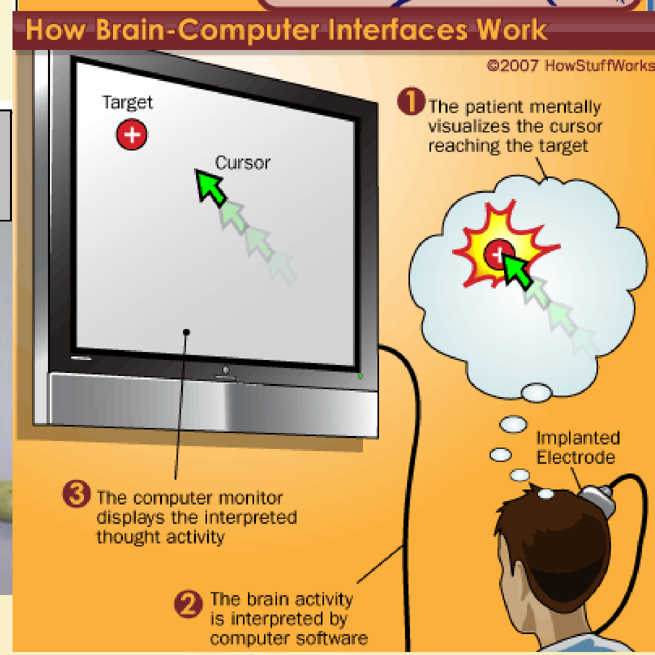
Wireless Brain-Computer Interface





D.D. Schmorrow et al. (Eds.): Augmented Cognition, HCII 2009, Springer, LNAI 5638, pp. 741–748, 2009.





Brain-to-Brain (B2B) Communication

http://www.youtube.com/watch?v=93p7oDkA5WA

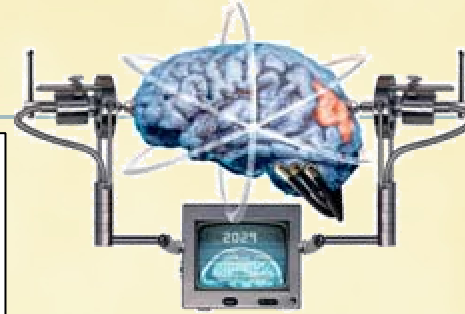
«The **Global Brain** is a metaphor for the intelligent network formed by humans together with the knowledge and communication technologies that connect them.»
– *Wikipedia - Global Brain*

«**Collective intelligence** is a form of intelligence that emerges from the collaboration and competition of many individuals. Collective intelligence appears in a wide variety of forms of consensus decision making in bacteria, animals, humans, and computers.»
– *Wikipedia - Collective Intelligence*

«**Swarm intelligence** is a collective behavior of decentralized, self-organized systems. SI systems are typically made up of a population of simple agents interacting locally with one another and with their environment. The agents follow very simple rules, and although there is no centralized control structure dictating how individual agents should behave, local interactions between such agents lead to the emergence of complex global behavior. Examples of SI: ant colonies, bird flocking, animal herding, bacterial growth, ...»
– *Wikipedia - Swarm Intelligence*

Who is Human 2.0 ? Wizard? Terminator? Surrogate? Agent Smith? Gamer?

Human v. 2.0



h2o

new minds, new bodies, new identities

Ushering in a New Era for Human Capability

The story of civilization is the story of humans and their tools. Use of tools has changed the human mind, altered the human body, and fundamentally reshaped human identity. Now at the dawn of the 21st century, a new category of tools and machines is poised to radically change humanity at a velocity well beyond the pace of Darwinian evolution.

A science is emerging that combines a new understanding of how humans work to usher in a new generation of machines that mimic or aid human physical and mental capabilities. Some 150 million of us are over the age of 80, while 200 million of us suffer from severe cognitive, emotional, sensory, or physical disabilities. Giving all or even most of this population a quality of life beyond mere survival is both the scientific challenge of the epoch and the basis for a coming revolution over what it means to be human. To unleash this next stage in human development, our bodies will change, our minds will change, and our identities will change. The age of Human 2.0 is here. http://h20.media.mit.edu/about.html

“By the 2020s, nanotechnology will enable us to create almost any physical product we want from inexpensive materials, using information processes. We will be able to go beyond the limits of biology, and replace your current "human body version 1.0" with a dramatically upgraded version 2.0, providing radical life extension. The "killer app" of nanotechnology is "nanobots", blood-cell-sized robots that can travel in the bloodstream destroying pathogens, removing debris, correcting errors in DNA and reversing ageing processes. …

As we reach the 2030s, the non-biological portion of our intelligence will predominate. By the mid 2040s, the non-biological portion of our intelligence will be billions of times more capable than the biological portion. Non-biological intelligence will have access to its own design and will be able to improve itself in an increasingly rapid redesign cycle.”

Ray Kurzweil (October 2005)

http://www.smh.com.au/news/next/human-20/2005/10/24/1130006035858.html

Believe we or not, but must be ready

2. Towards Global Understanding Environment

past, present and future projects

2. Towards Global Understanding Environment

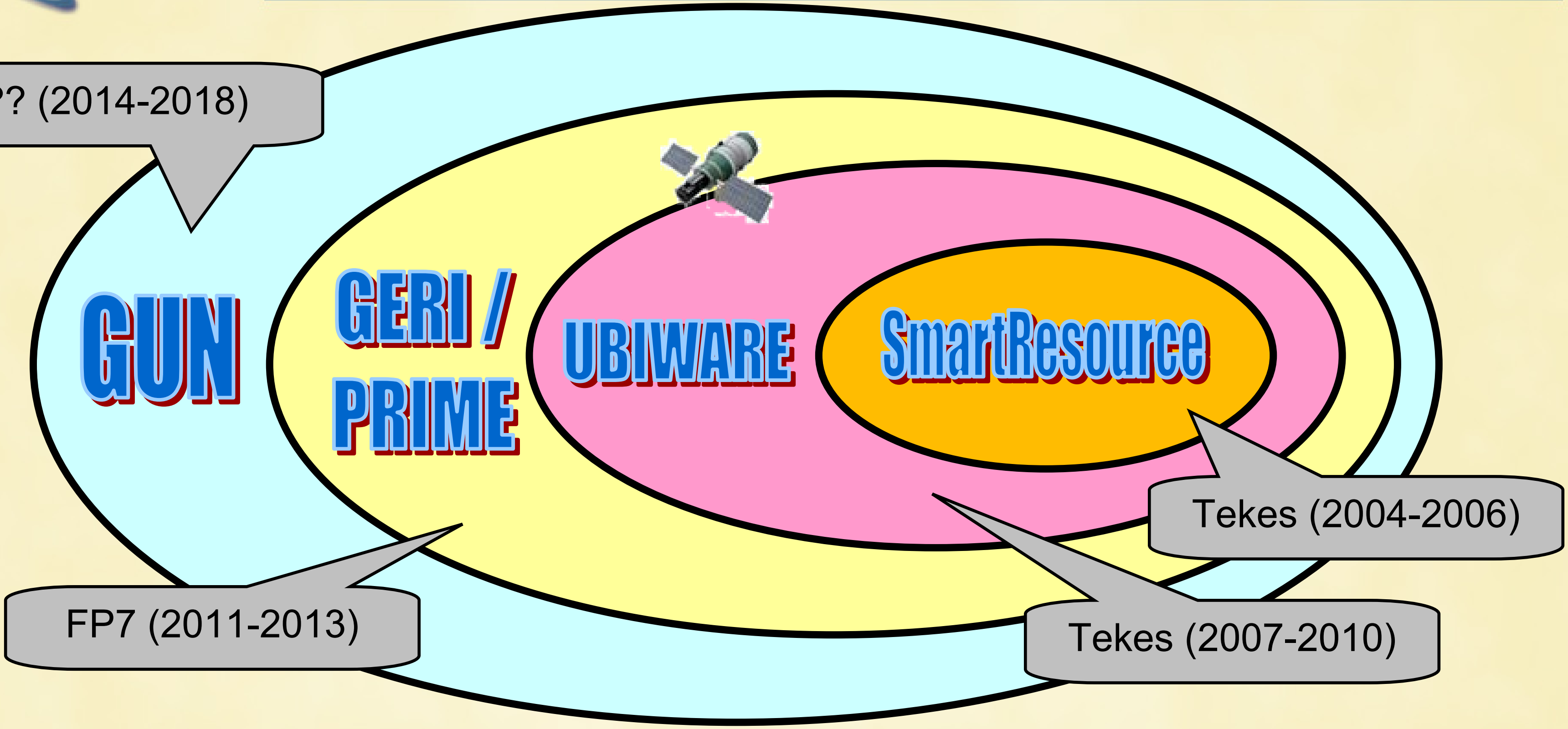
- 2.1. SmartResource Project
- 2.2. UBIWARE Project
- 2.3. PRIME Project
- 2.4. Other Projects

ψ-Projection of GUN-Related Research

PSSI - projection:

- **P**roactivity (agent technologies, Distributed AI, MAS, …)
- **S**emantics (Semantic Web, Semantic Technologies, …)
- **S**ervices (SaaS, SOA, SWS, Cloud Computing, …)
- **I**ntelligence (machine learning, data mining, knowledge discovery, pattern recognition, NLP, …)

Our Roadmap: GUN-GERI-UBIWARE-SmartResource



GUN (Global Understanding Environment) – Proactive Self-Managed Semantic Web of Things – Web 5.0 candidate - general ecosystem and final destination

GERI (Global Enterprise Resource Integration) – GUN subset related to industrial domains - is based on **PRIME** (Proactive Inter-Middleware) as UBIWARE extension

UBIWARE – middleware for GERI

SmartResource – semantic technology, pilot tools and standards for UBIWARE

2.1. SmartResource Project

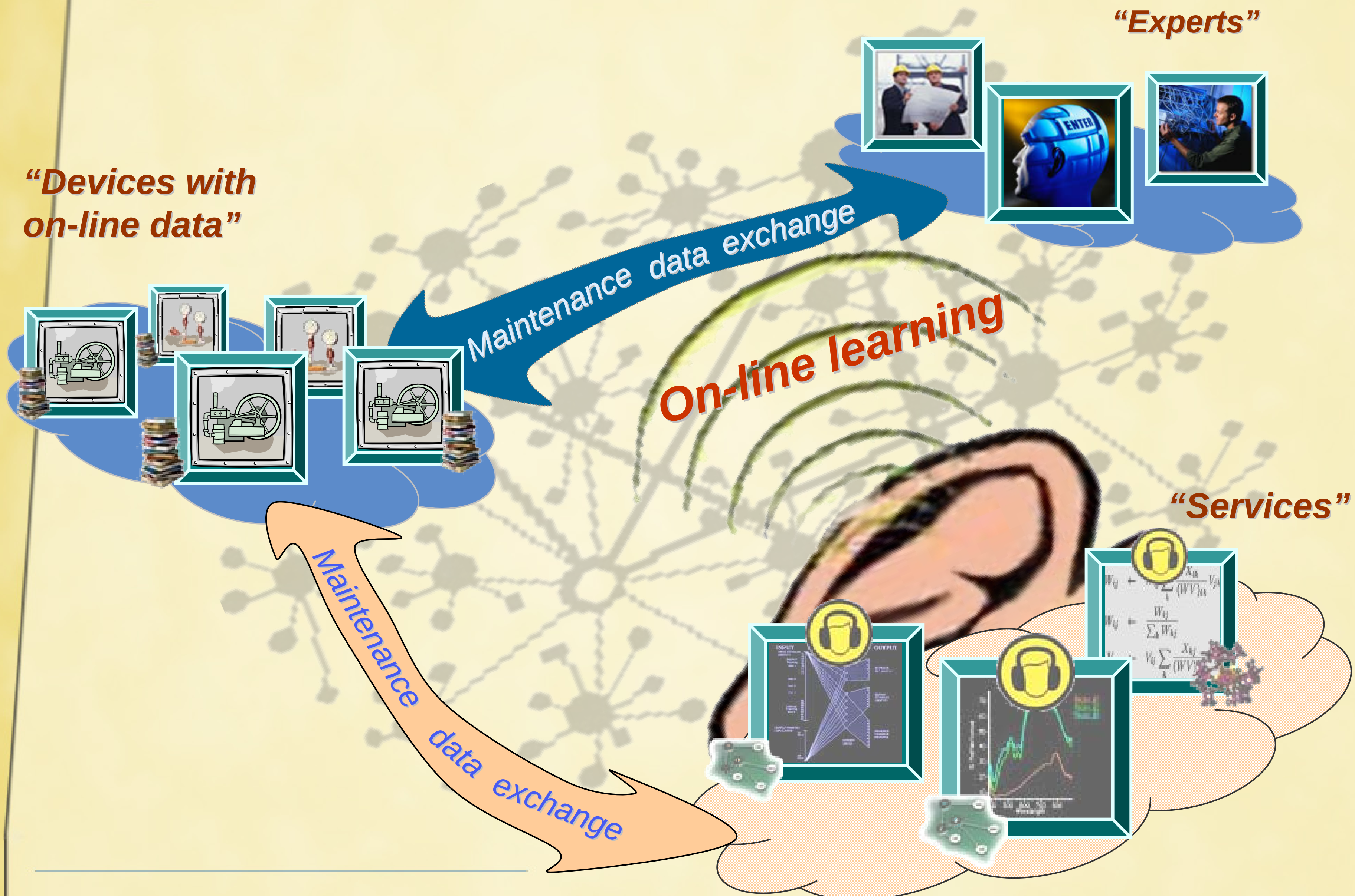
"Proactive Self-Managed Resources in Semantic Web"

SmartResource

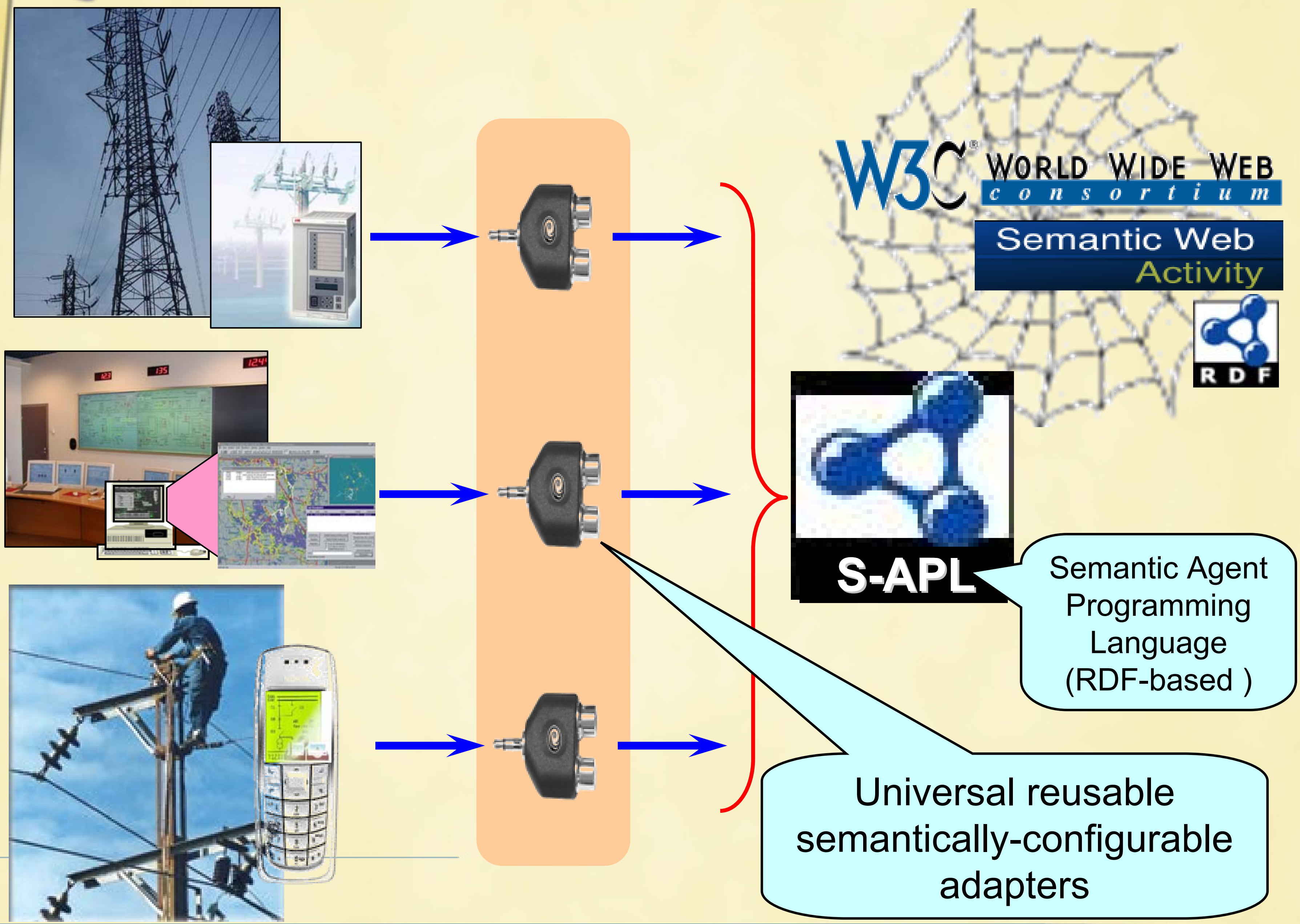
Tekes Project (2004-2006)

- SmartResource: "Proactive Self-Maintained Resources in Semantic Web" Tekes project (2004-2006) performed by Industrial Ontologies Group: http://www.mit.jyu.fi/ai/OntoGroup/SmartResource.htm .
- One of the most essential results of the SmartResource project was creation of the "Smart Resource Technology" for designing complex software systems. The technology allows considering each traditional system component as a "smart resource", i.e. proactive, agent-driven, self-managing. Such approach has shown certain advantages comparably to other software technologies, e.g. OOSE, SOA, Component-Based SE, Agent-Driven SE, Semantic SE, etc.

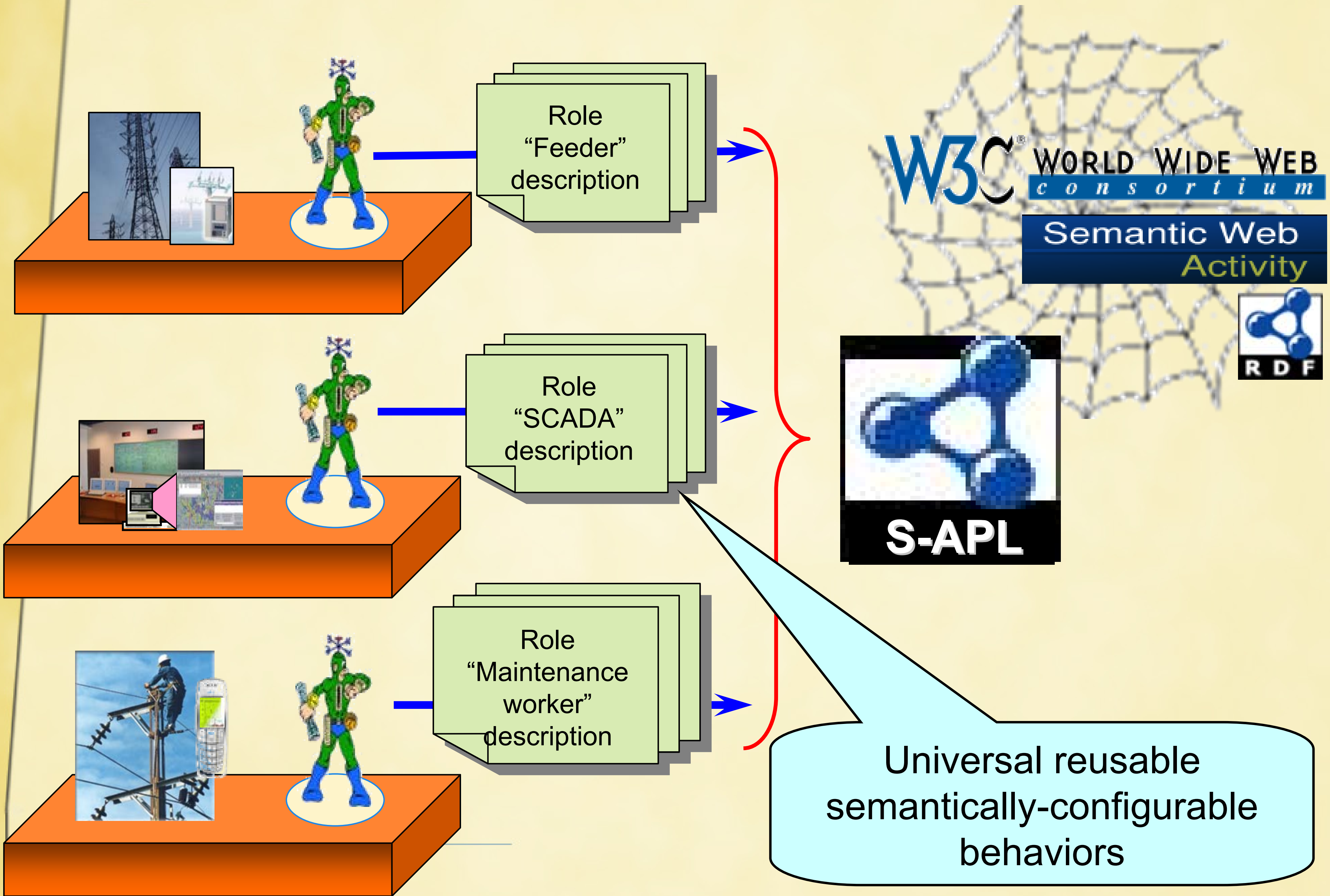
Smart Maintenance Environment



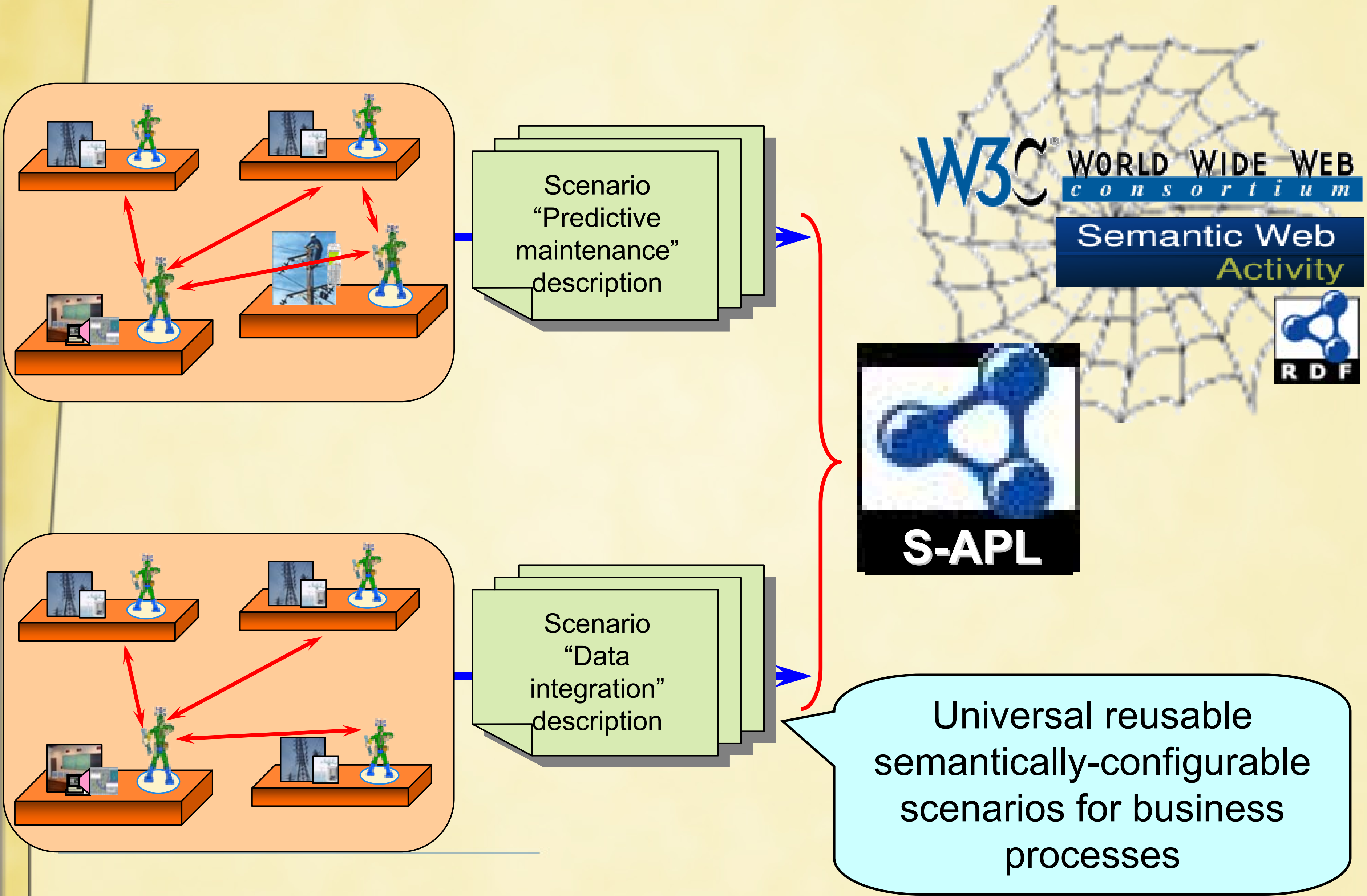
Challenge 1: General Adaptation Framework



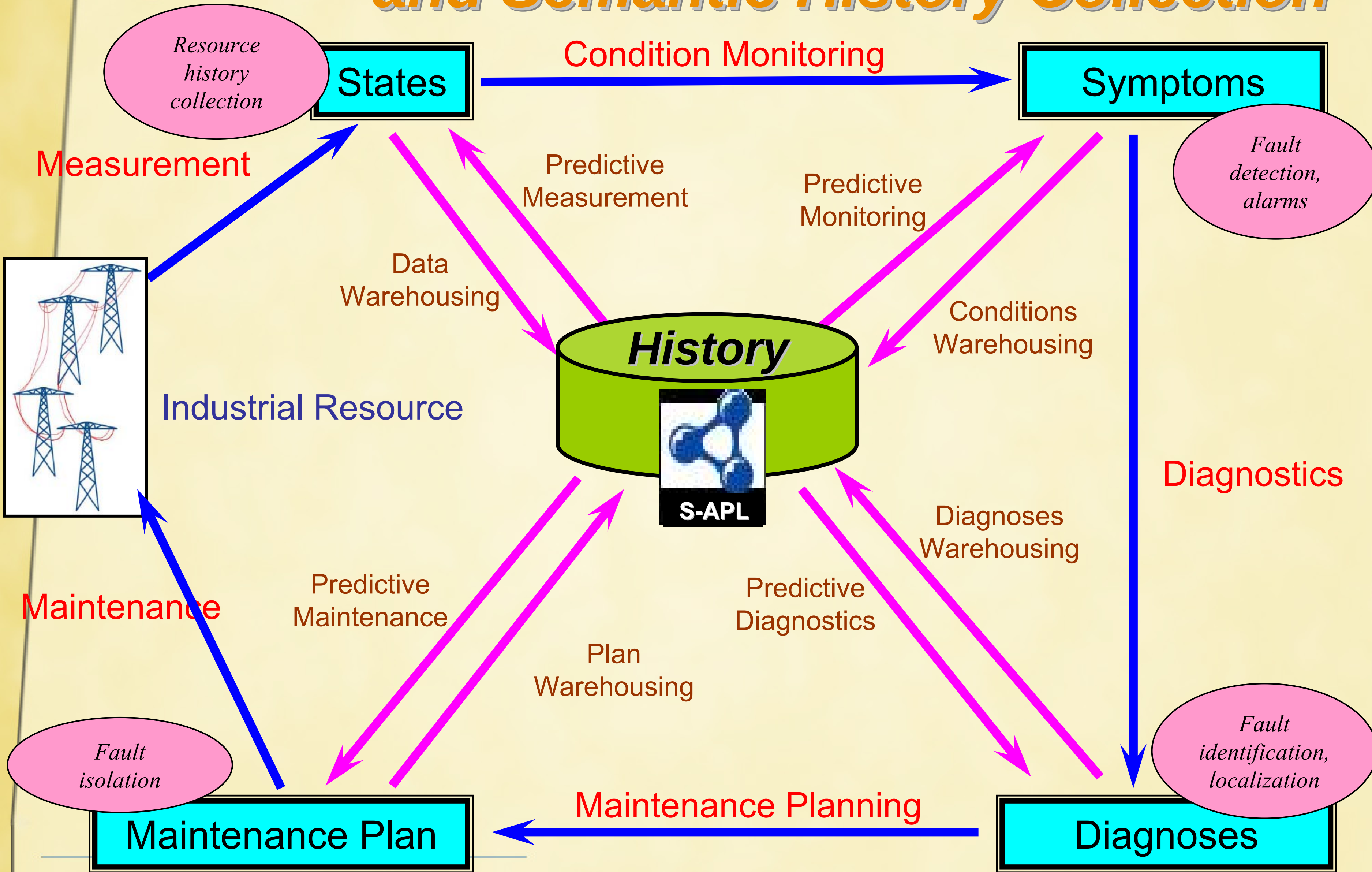
Challenge 2: General Proactivity Framework



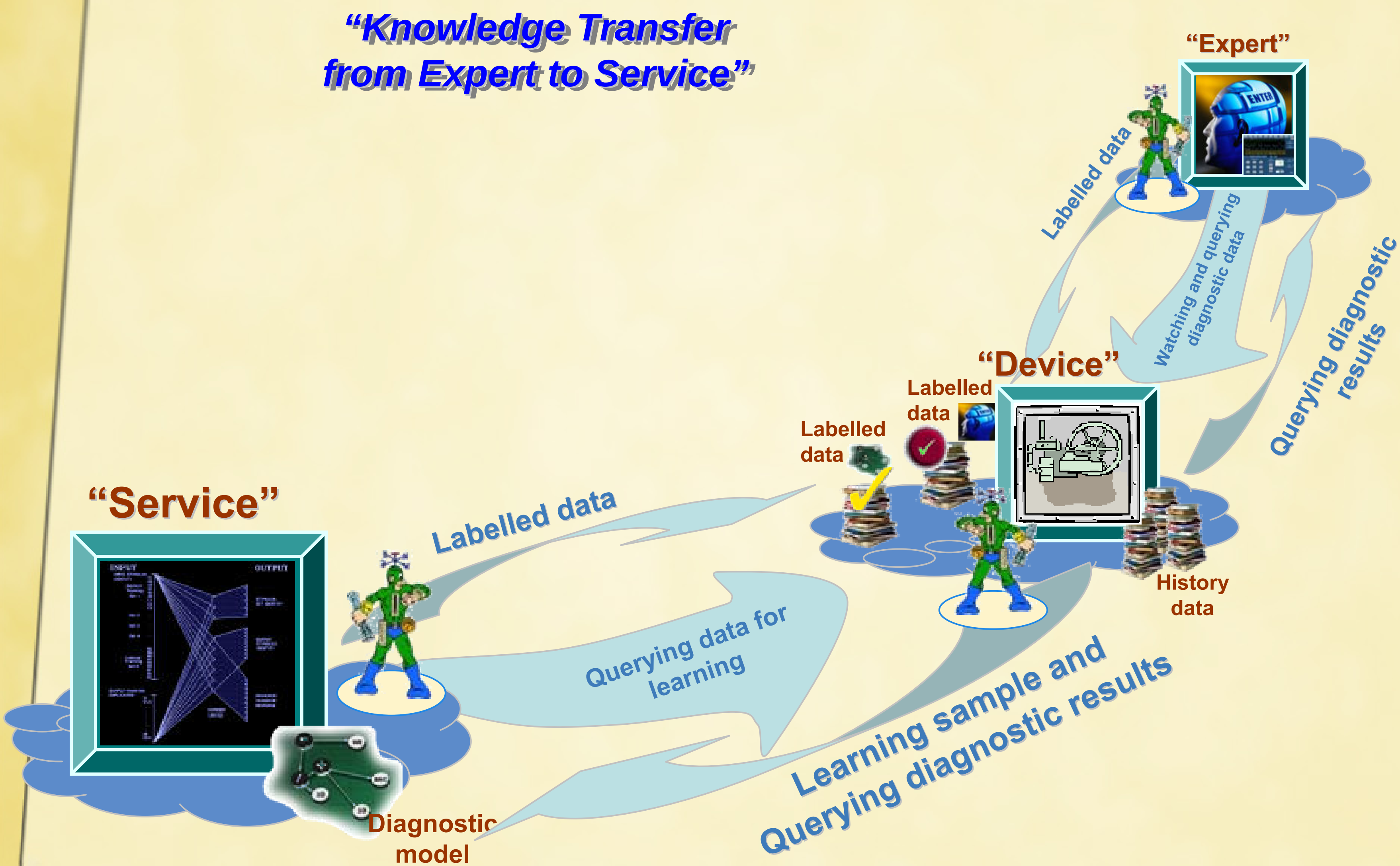
Challenge 3: General Networking Framework



Resource Maintenance Lifecycle and Semantic History Collection



One of Smart Resource Scenarios



User Interface generated from the ontology

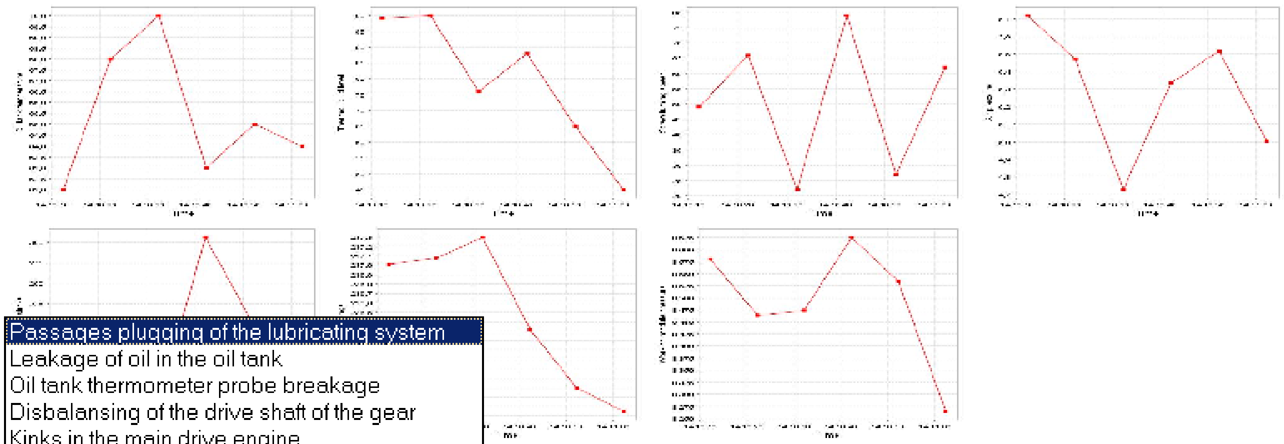
C:\MyTemp\Zharko\Work\HTML_to_expert.html - Microsoft Internet Explorer

File Edit View Favorites Tools Help

ANNOTATION

Diagnostics for 123456XZ24
The manufacture of machine is KUN FONG Machinery Co., LTD
The Contact person is Mr. Chan Tong
Device KF-330 blow molding machine
Mail address of manufacture is 14, LANE 108, YU-MEN ROAD, TAICUNG CITY, TAIWAN
The manufacture's phone and fax 886-4-24610589, 886-4-24631205

Official WEB page is http://www.kunfong.ru
Device WEB page is http://www.kunfong.ru/eng/prod4_1.htm
E-mail of manufacture is kunfong9@ms49.hinet.net



Passages plugging of the lubricating system
Leakage of oil in the oil tank
Oil tank thermometer probe breakage
Disbalansing of the drive shaft of the gear
Kinks in the main drive engine
DevicePhysicalDiagnosis

Passages plugging of the lubricating system

MakeDiagnosis

Possible SmartResource Utilization

2.2. UBIWARE Project

"Smart Semantic Middleware for Ubiquitous Computing"

"Smart Semantic Middleware for Ubiquitous Computing"

- *Due to heterogeneity of provided services and supported components, UBIWARE is based on integration of several technologies: Semantic Web, Distributed Artificial Intelligence and Agent Technologies, Ubiquitous Computing, SOA (Service-Oriented Architecture), Web X.0, P2P and related concepts.*
- *The research and design on UBIWARE is started by* Industrial Ontologies Group *within UBIWARE project: "Smart Semantic Middleware for Ubiquitous Computing" (June 2007 – May 2010) funded by Tekes and industrial companies.*
- *Project web page:*

http://www.mit.jyu.fi/ai/OntoGroup/UBIWARE_details.htm

What is UBIWARE (in short)

- UBIWARE is a new software technology and a tool to support:
  - design and installation of…,
  - autonomic operation of… and
  - Interoperability among…
- … complex, heterogeneous, open, dynamic and self-configurable distributed industrial systems;…
- … and to provide following services for system components:
  - adaptation;
  - automation;
  - centralized or P2P organization;
  - coordination, collaboration, interoperability and negotiation;
  - self-awareness, communication and observation;
  - data and process integration;
  - (semantic) discovery, sharing and reuse.

URL: http://www.cs.jyu.fi/ai/OntoGroup/UBIWARE_details.htm

Challenges and Solutions

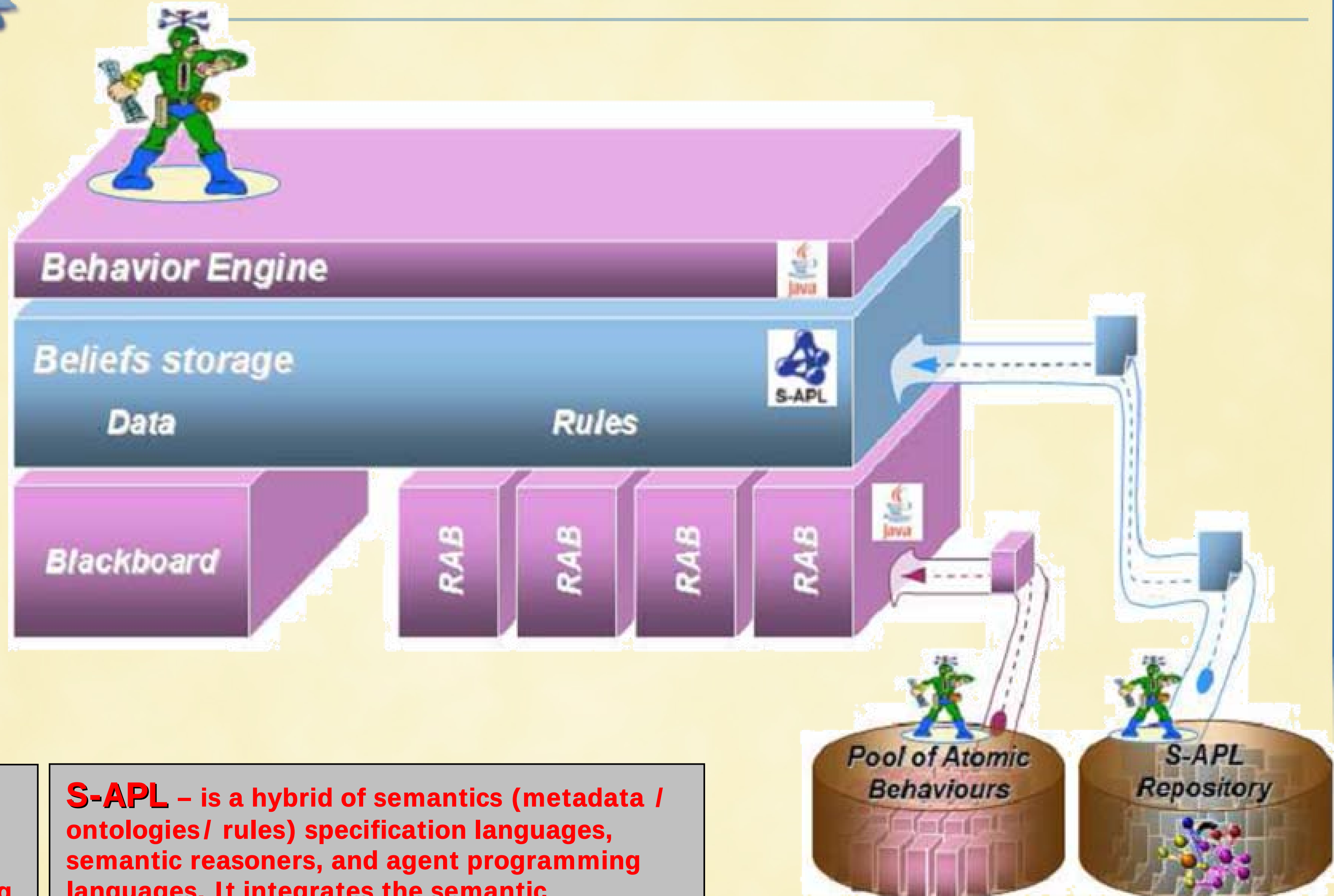
- Very heterogeneous resources
  - Different nature (devices, Web services, humans).
  - Different organizations.
  - Not always the exact same domain.
- Data-level heterogeneity
  - Calls for the ***Semantic Technology***.
- Protocol-level heterogeneity
  - GUN approach through the ***Agent Technology***.
  - Each resource has a representative – software agent (not necessarily intelligent or even fully autonomous, but at least able to act as a programmable proxy).
  - Interactions among resources go through their agents.

Challenges and Solutions (2)

- Coordination
  - When considering physical devices, in contrast to purely digital world of Web services, coordination is critical.
  - Coordination is about resources planning their activities while attempting to avoid negative interactions (e.g. collision over a non-shareable resource) as well as exploit positive interactions (re-using each other results).
  - Enabling coordination among heterogeneous resources is even harder problem than data-level or protocol-level heterogeneity – *communication about actions.*

- GUN approaches through ***semantic programming***
  - Agents are programmed in RDF-based Semantic Agent Programming Language (S-APL).
  - Agents communicate their action plans in S-APL as well.

Current UBIWARE Agent Architecture



S-APL

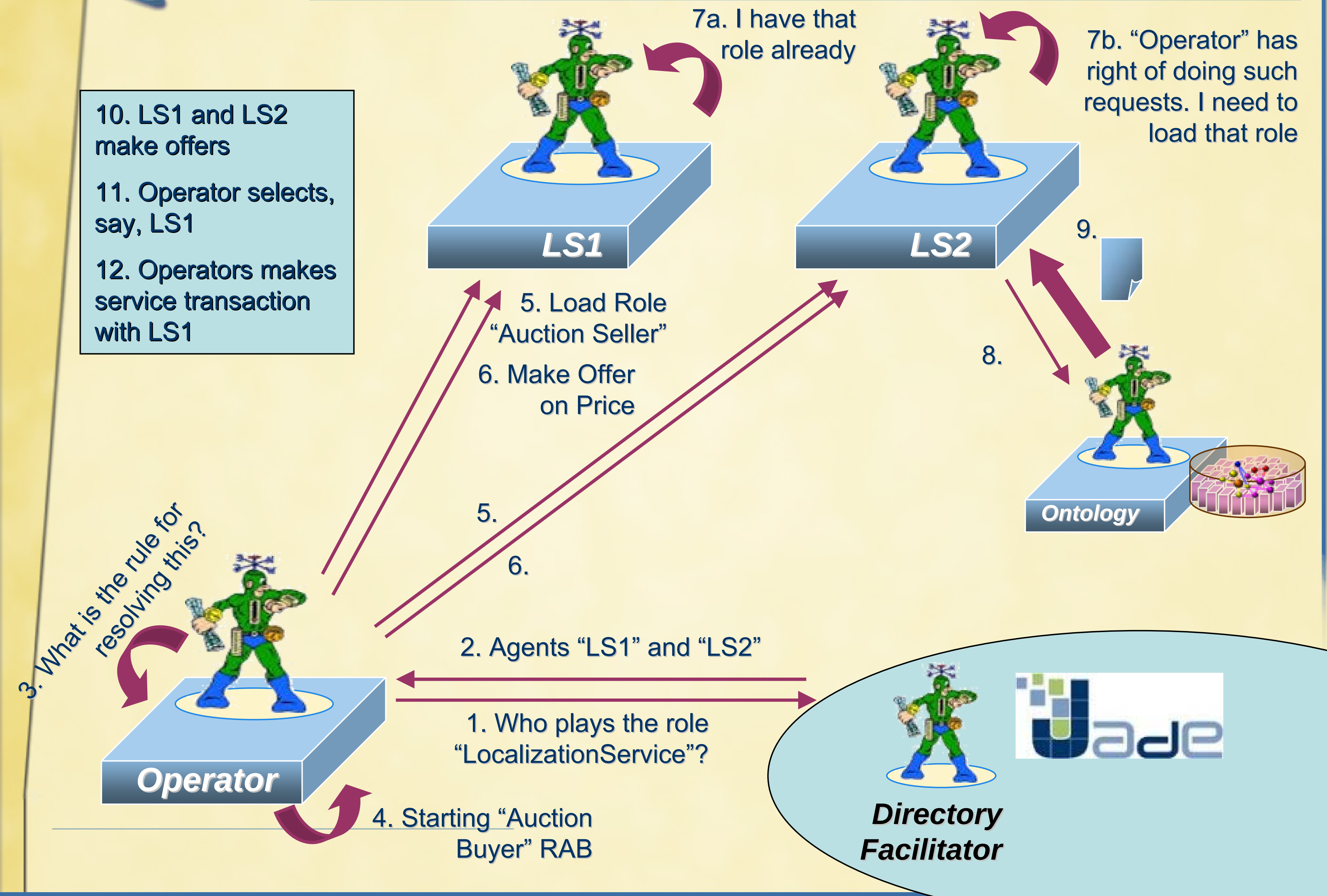
S-APL – Semantic Agent Programming Language (RDF-based)

S-APL – is a hybrid of semantics (metadata / ontologies/ rules) specification languages, semantic reasoners, and agent programming languages. It integrates the semantic description of domain resources with the semantic prescription of the agents' behaviors

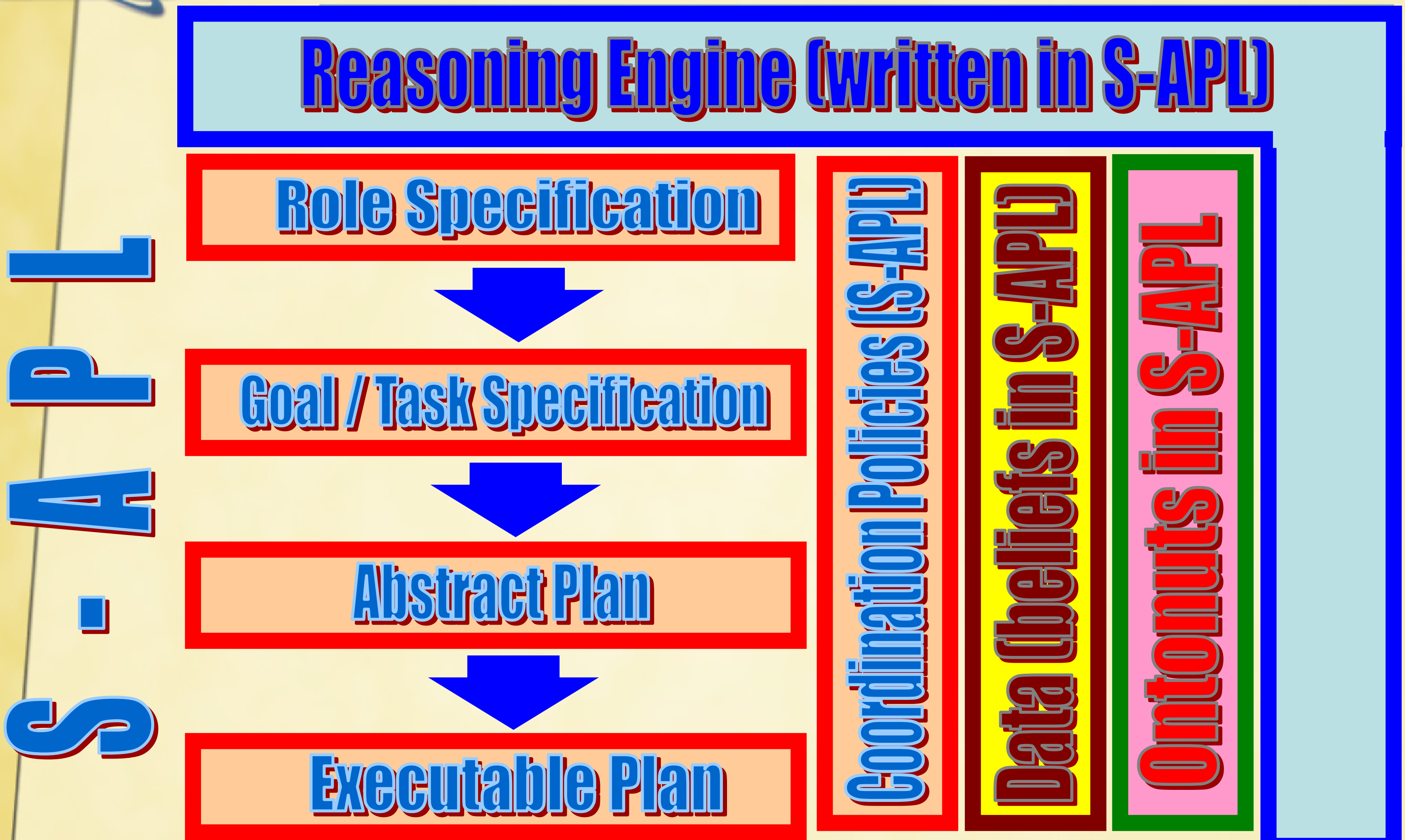
Ontology

http://users.jyu.fi/~akataso/sapl.html

Scenario: Auction for service selection

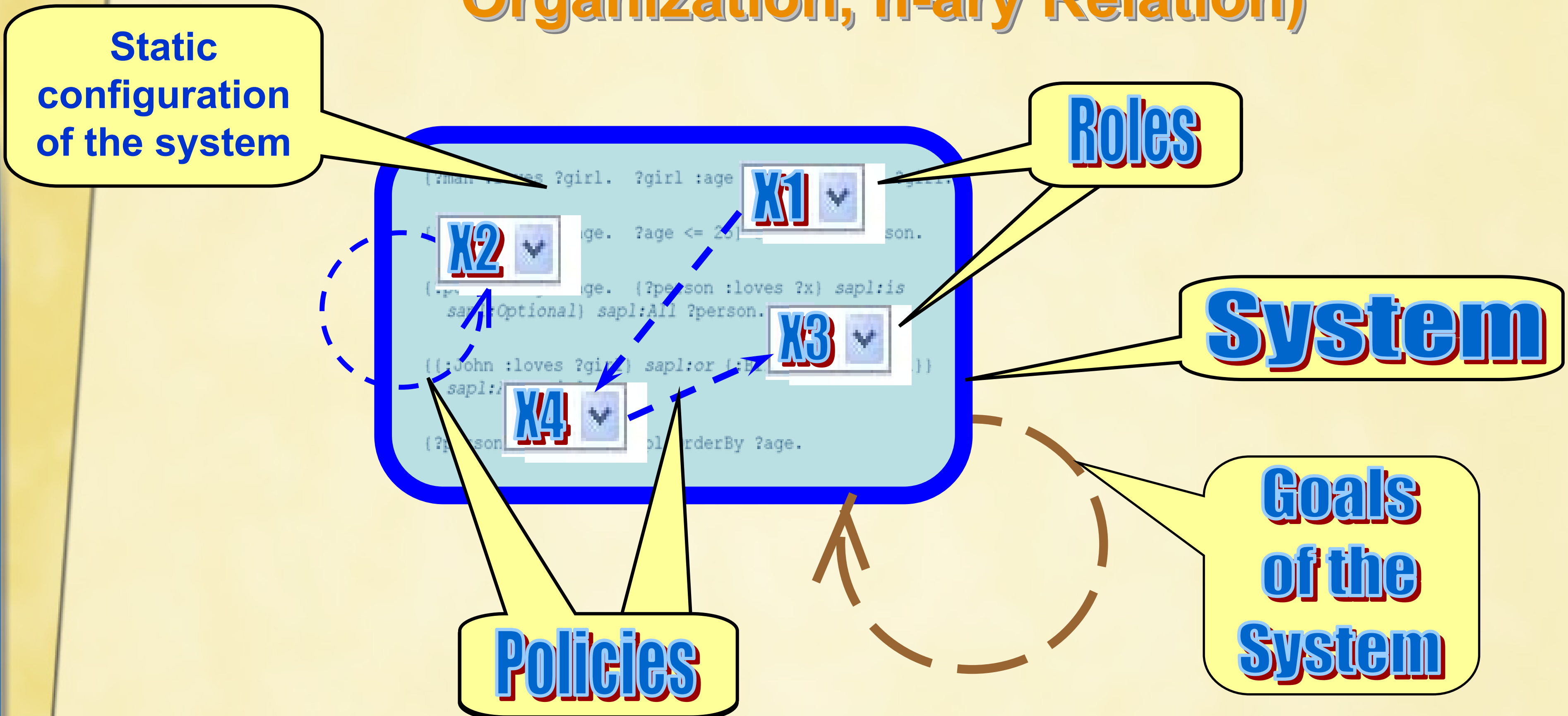


S-APL stack



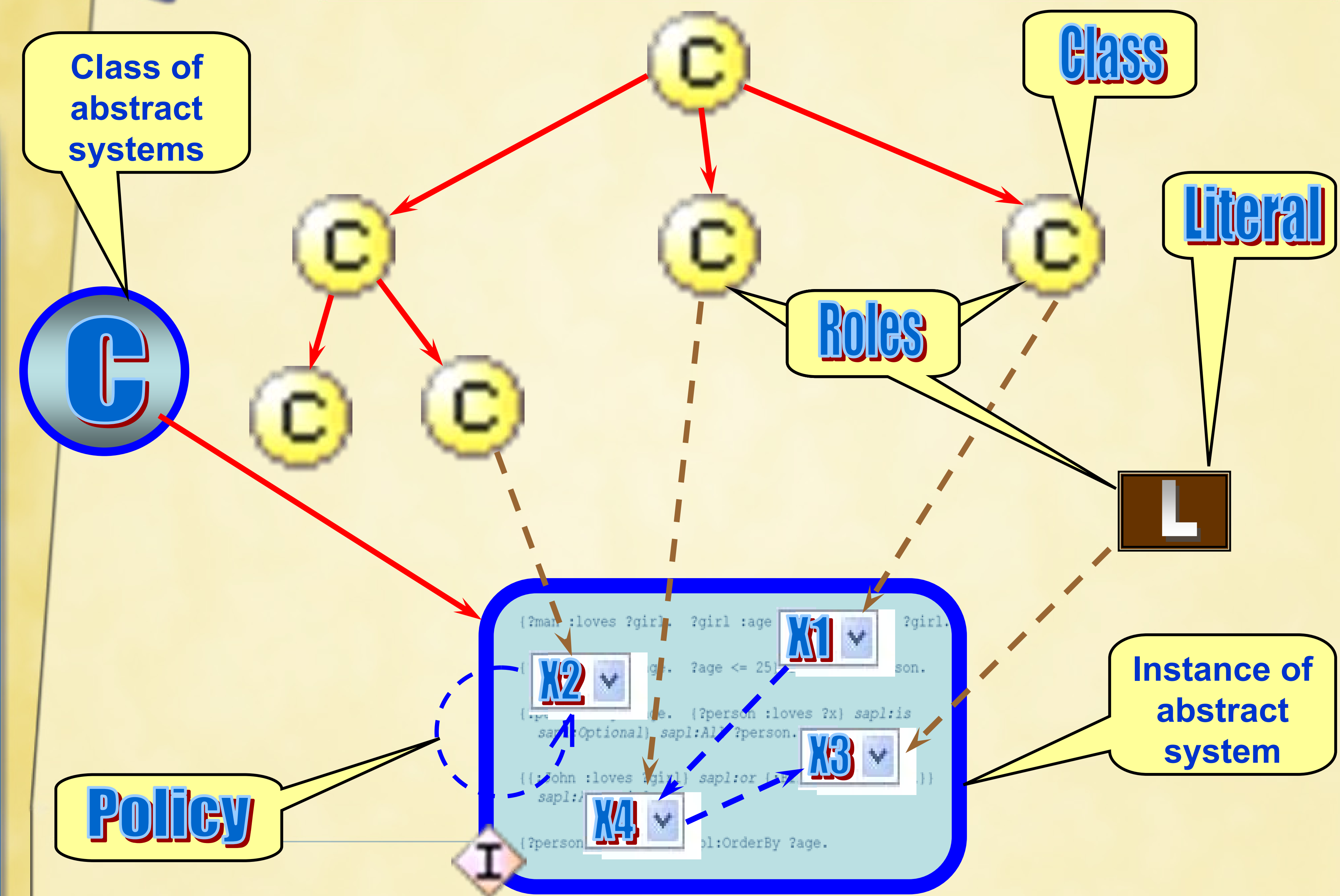
Automatic (reasoning) from roles/policies specifications and perceived data up to executing actions

Abstract System (Complex Resource, Organization, n-ary Relation)

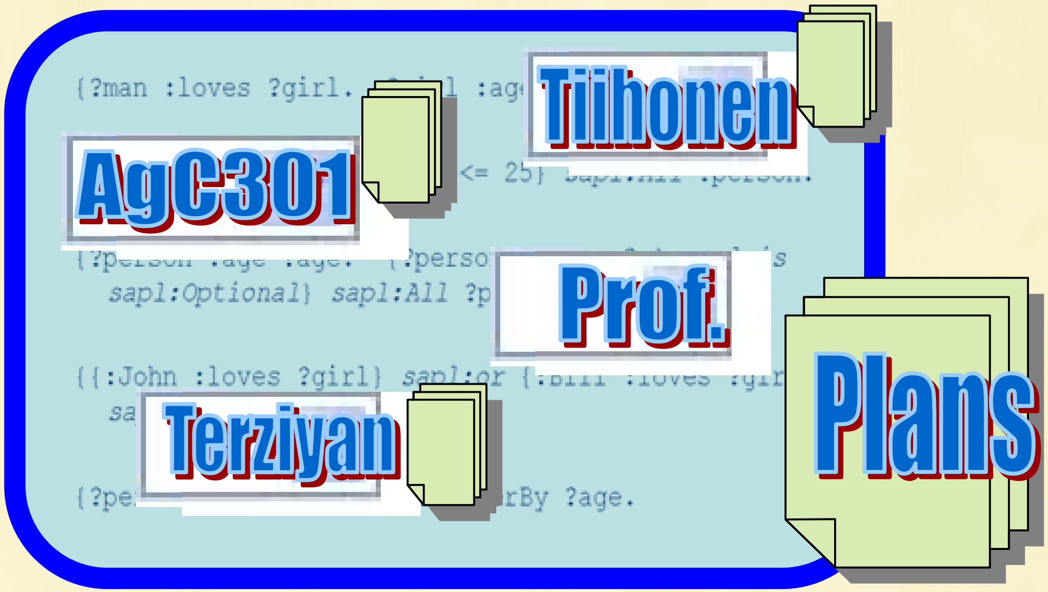


An *abstract system* (or “organization”) is such complex resource, which configuration contains (dynamic) *goals*, variables (“*roles*”) either for resources (system components, subsystems, etc.) or for property values, and also constraints and relationships on/between variables (“*policies*”)

Abstract System within Domain Ontology



Concrete or Executable System



Executable System is the result of transformation from an abstract system, in which all the roles are taken by concrete resources and the goals and policies are replaces by concrete plans (on how to reach the goal with respect to policies).

All subsystems (components) of an executable system are executable systems

Latest Innovations Invented by Industrial Ontologies Group in UBIWARE

- OntoNuts

OntoNuts – is the ontology-based instrument to reconfigure and enhance complex distributed systems by automated discovery and linking external sources of heterogeneous and dynamic data and capabilities during system runtime

- 4i (“for eye”) technology

4i – is smart ontology-based visualization technology able to automatically discover and utilize external visualization service providers and dynamically create and visualize mashups from external data sources in a context-driven way

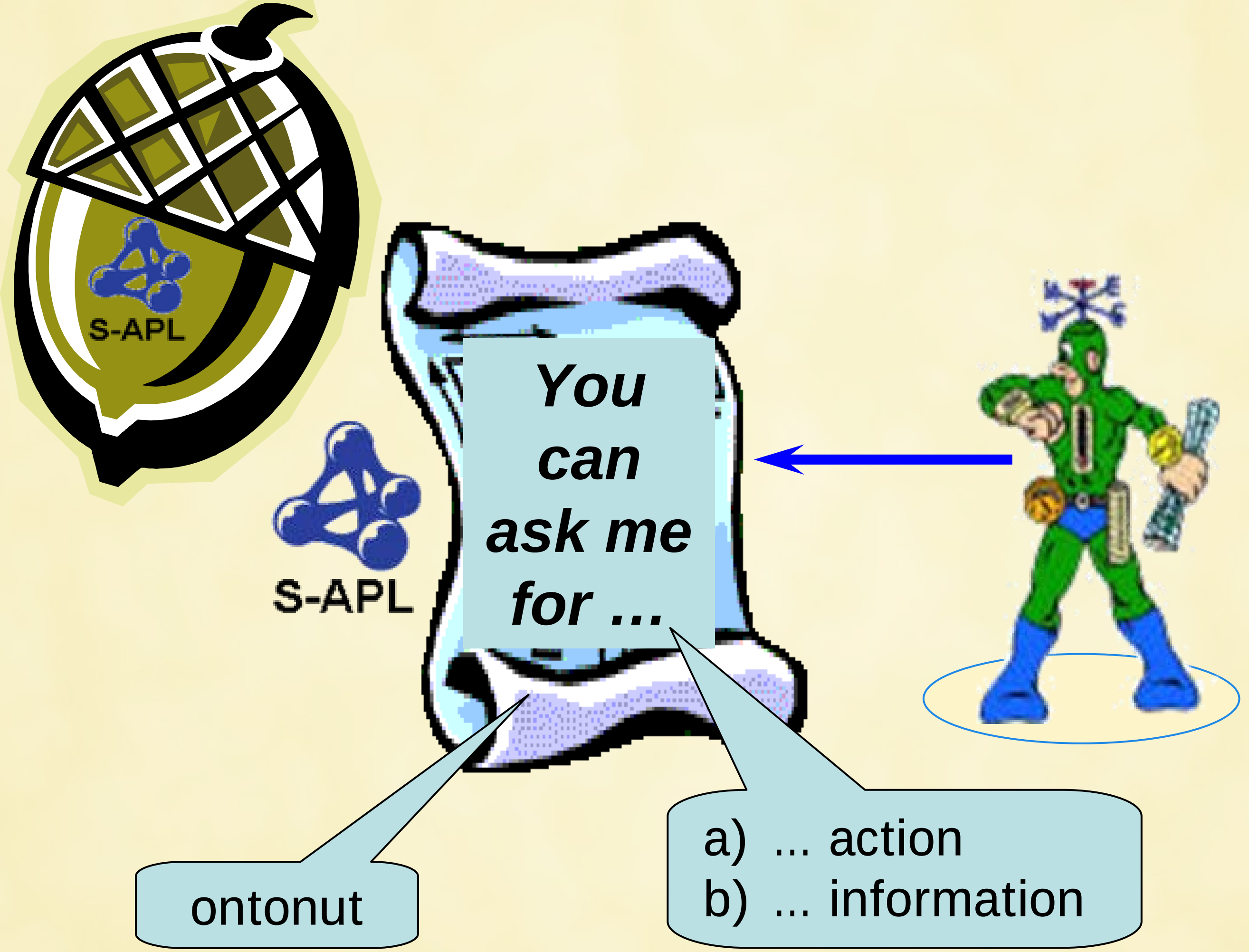
- Smart Comments

Smart Comments – is smart ontology-based technology for end-user-driven control and configuration management of the application in runtime based on smart mapping of appropriate tags from natural language comments provided by a SW engineer and the source code.

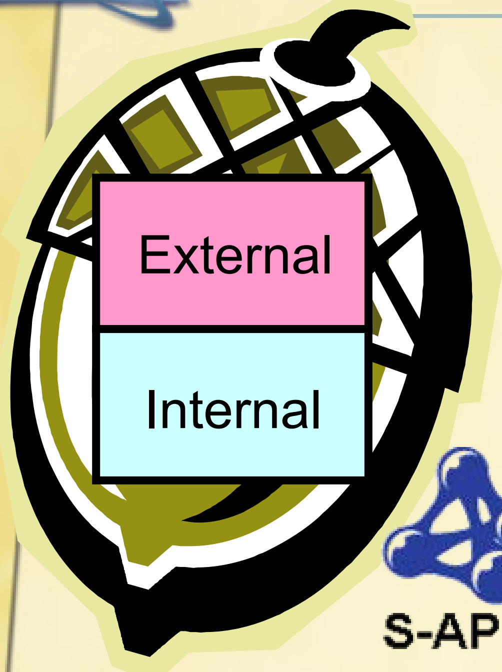
What are OntoNuts ?

- **OntoNuts** – is the ontology-based instrument to reconfigure and enhance complex distributed systems by automated discovery and linking external sources of heterogeneous and dynamic data and capabilities during system runtime

Ontonuts: Competence Profile of an Agent as a service provider (“what can I do” and “what can I answer”) and service plan (“how I do or answer”)



External view to ontonuts: Shared Competence Specification



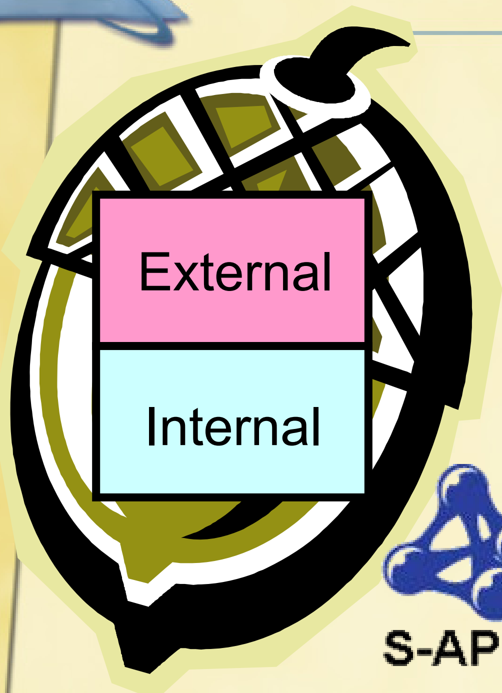
S-APL

a) I know everything about Mary
b) I know everything about cats
c) I know what time it is now
d) I know all lovers of John
e) I know grades on chemistry of all pupils from 4-B

a) I can open the door #456
b) I can fly
c) I can use knifes
d) I can build house from wood
e) I can visualize maps
f) I can grant access to folder “444”

We consider **ONTONUTS** to be shared S-APL specifications of these competences

Internal view to ontonuts: Action or Query Plans



a) I know everything about Mary

S-APL plan of querying either own beliefs or external database about Mary

a) I can open the door #456

S-APL plan of opening the door #456

We consider **ONTONUTS** to be also an internal plans to execute competences

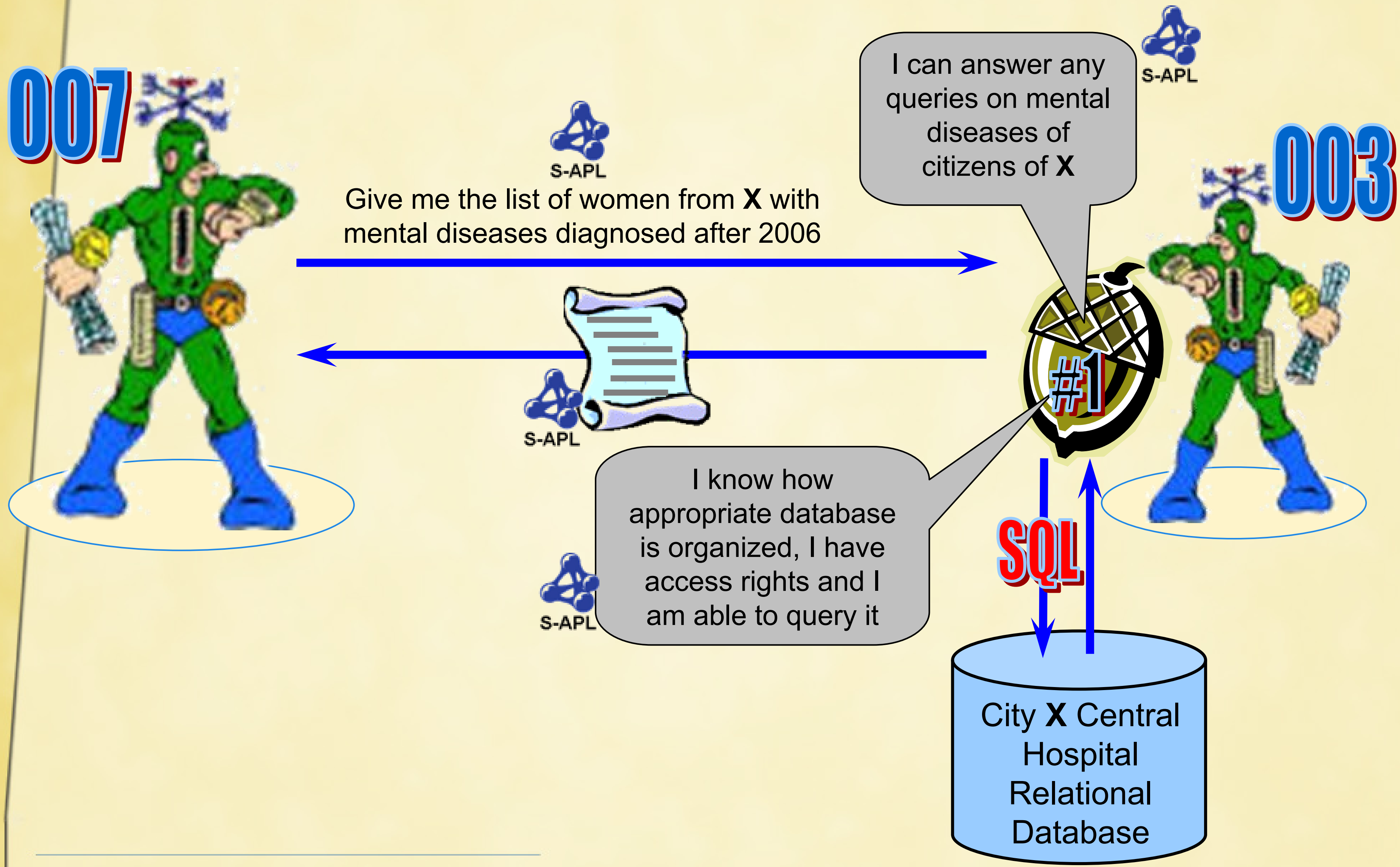
Possible general rule of ontonut appearance

IF I have the plan how to perform certain complex or simple action or the plan how to answer complex or simple query

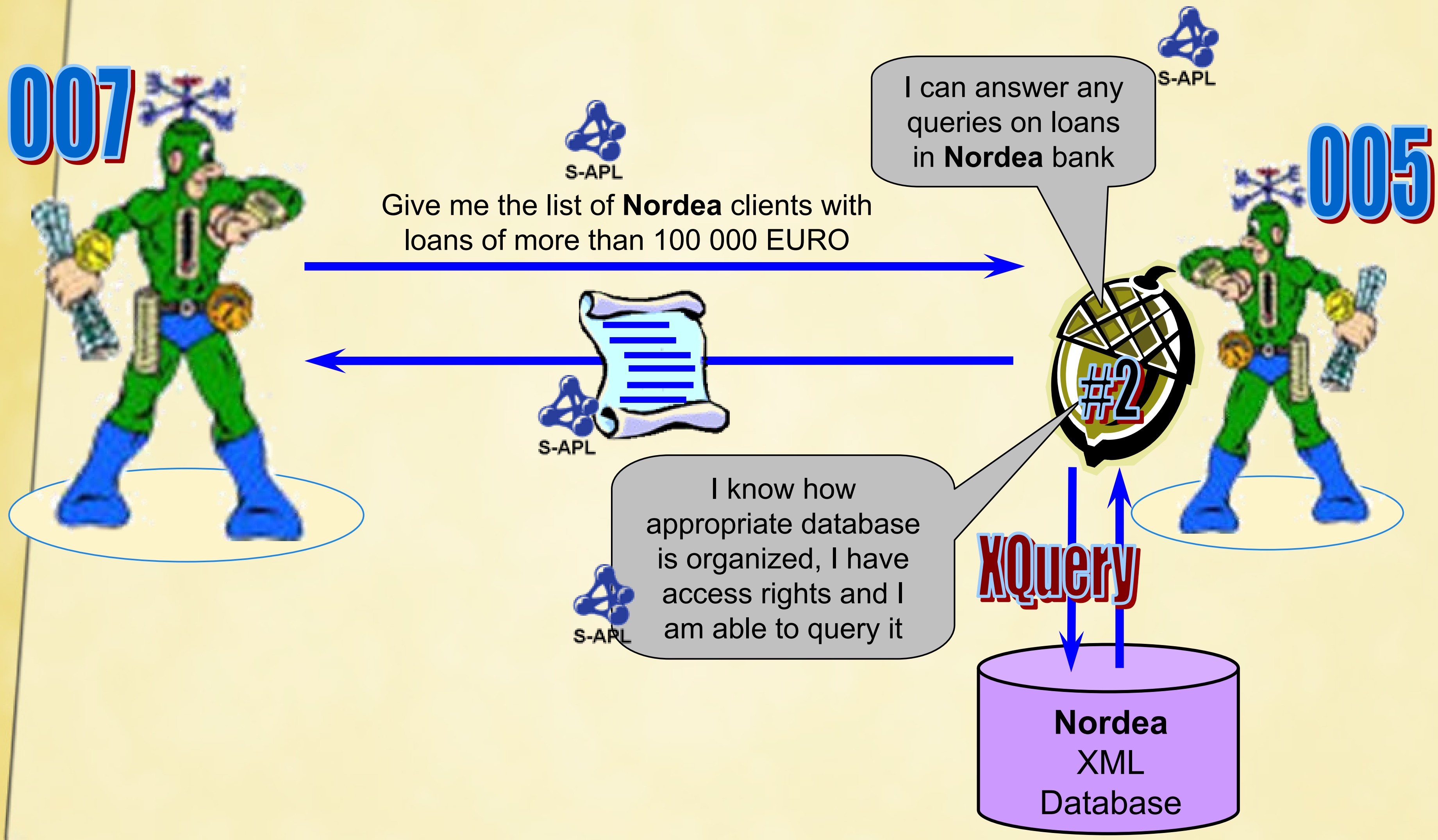
AND {time-to-time execution of the plan is part of my duty according to my role (commitment) **OR** I am often asked by others to execute action or query according to this plan}

THEN I will create ONTONUT which will make my competence on this plan explicit and visible to others

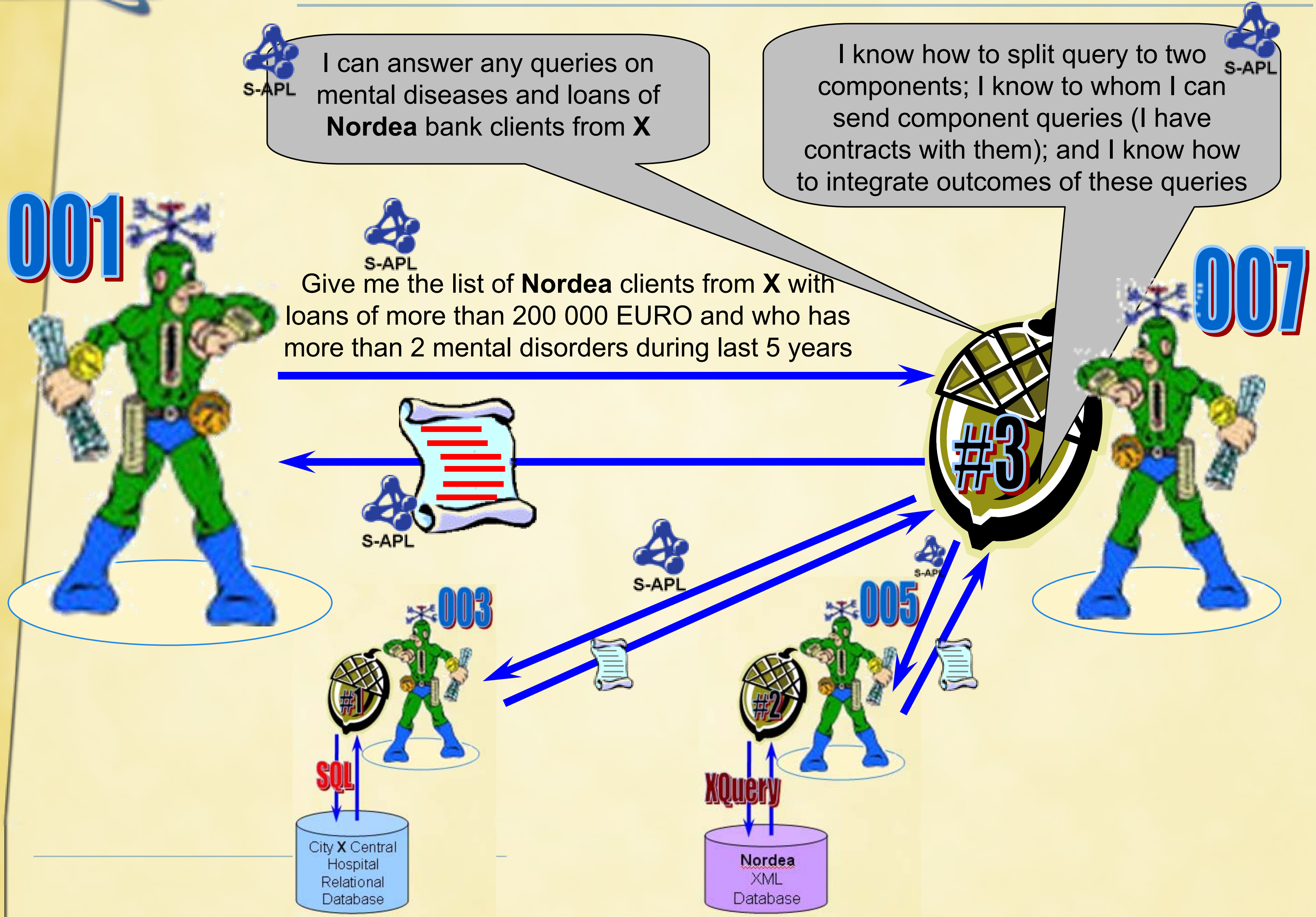
Example (1): Atomic Ontonut #1



Example (2): Atomic Ontonut #2



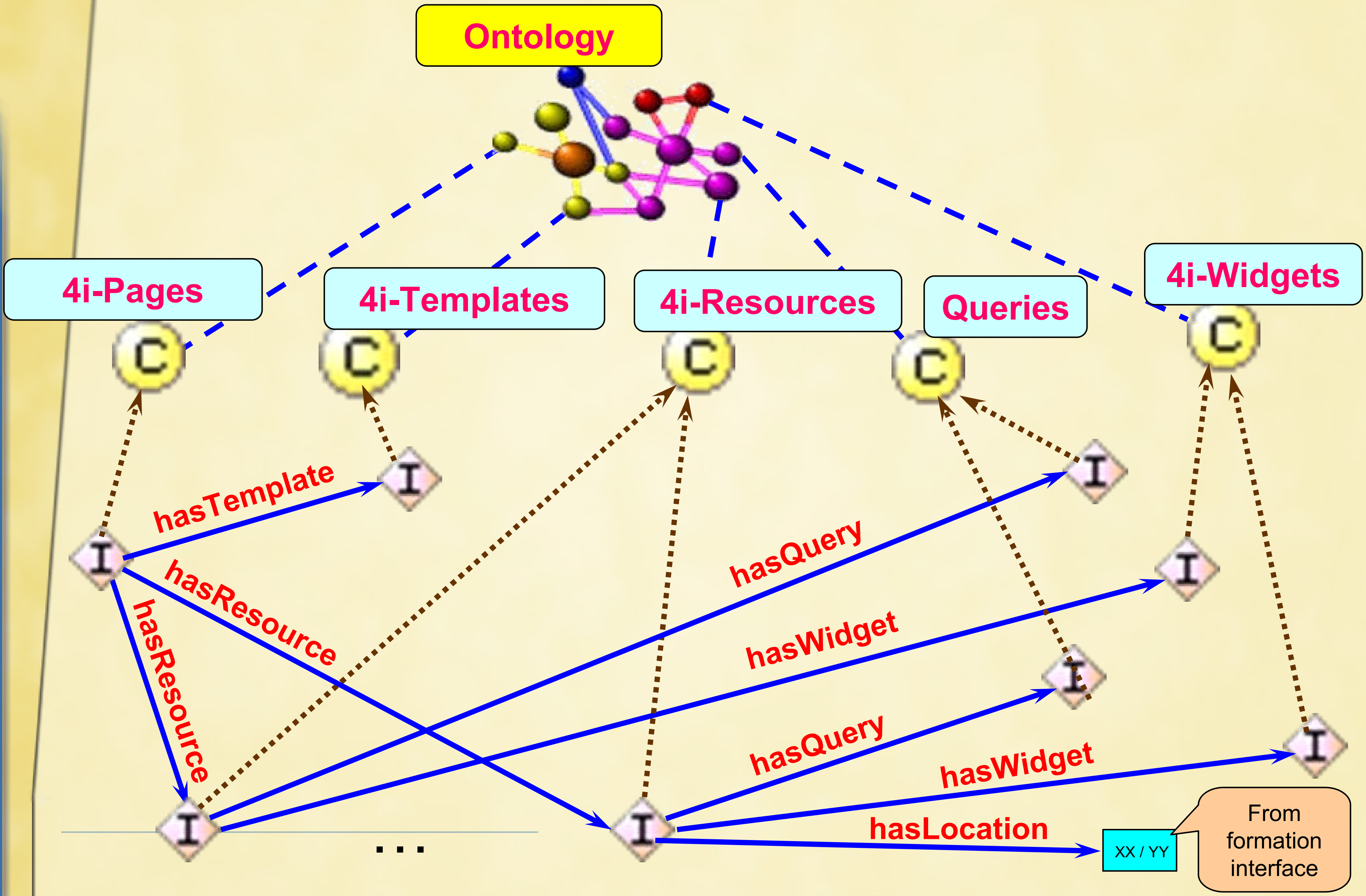
Example (3): Complex Ontonut #3



What is 4i ("for-eye") technology?

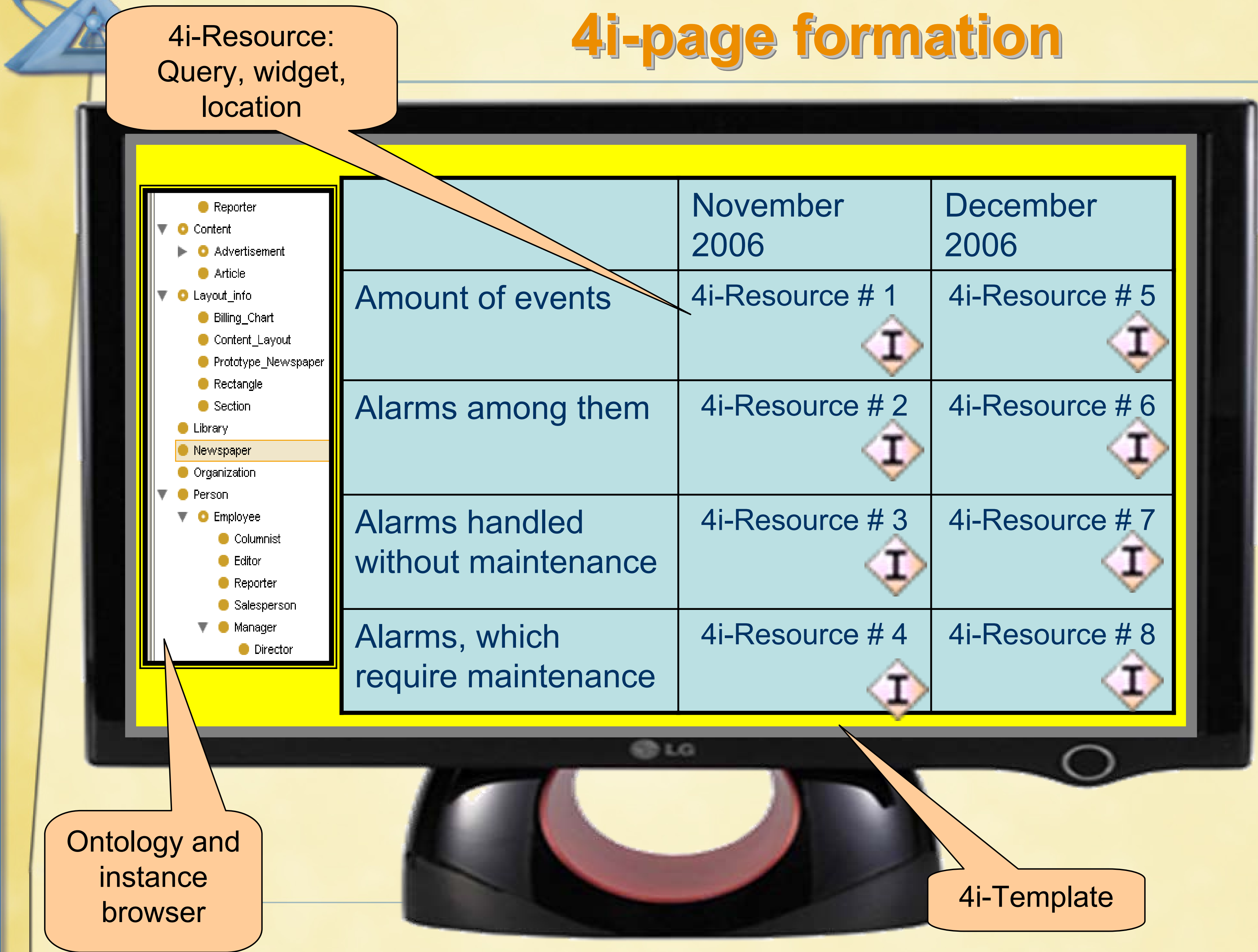
- **4i** – is smart ontology-based visualization technology able to automatically discover and utilize external visualization service providers and dynamically create and visualize mashups from external data sources in a context-driven way

4i-ontology



Ontology
4i-Pages
4i-Templates
4i-Resources
Queries
4i-Widgets
hasTemplate
hasResource
hasResource
hasQuery
hasWidget
hasQuery
hasWidget
hasLocation
XX / YY
From formation interface
...

4i-page formation



	November 2006	December 2006
Amount of events	4i-Resource # 1	4i-Resource # 5
Alarms among them	4i-Resource # 2	4i-Resource # 6
Alarms handled without maintenance	4i-Resource # 3	4i-Resource # 7
Alarms, which require maintenance	4i-Resource # 4	4i-Resource # 8

4i-page calculation

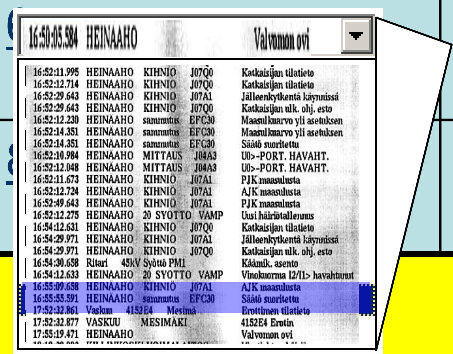
- 4-i engine will resolve all (SPARQL) queries in 4i-page and prepare html page as a result for further visualization

4i-page in html for visualization

	November 2006	December 2006
Amount of events	427	314
Alarms among them	146	102
Alarms handled without maintenance	65	50
Alarms, which require maintenance	81	52

4i-page in html: clickable numbers

	November 2006	December 2006
Amount of events	427	314
Alarms among them	146	102
Alarms handled without maintenance	[illegible]	50
Alarms, which require maintenance	[illegible]	52



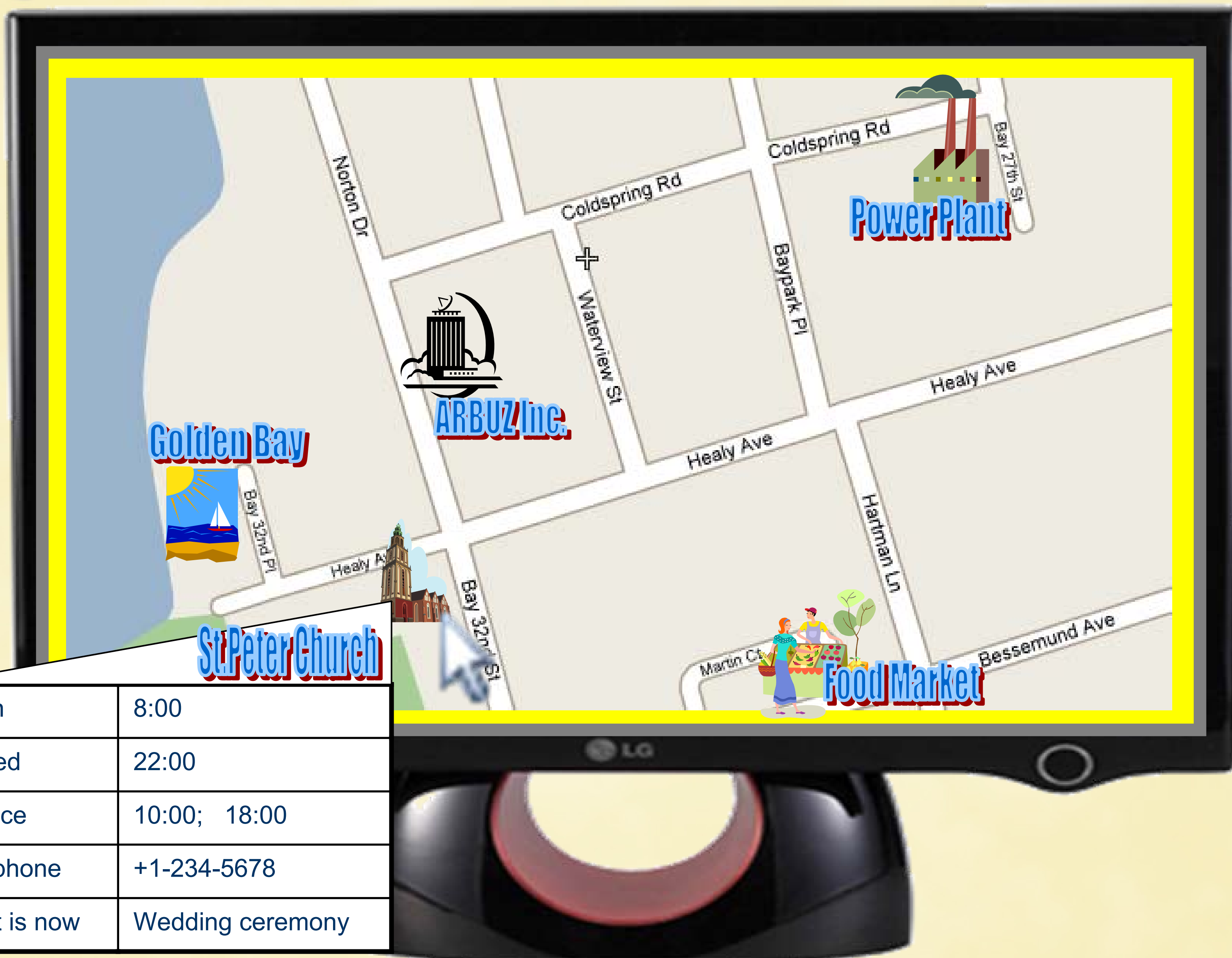
4i-page formation at other template

Ontology and instance browser

4i-Resource: Query, widget, location

4i-Template

4i-page in html



Open	8:00
Closed	22:00
Service	10:00; 18:00
Telephone	+1-234-5678
What is now	Wedding ceremony

What are Smart Comments ?

Smart Comments – is smart ontology-based technology for end-user-driven control and configuration management of the application in runtime based on smart mapping of appropriate tags from natural language comments provided by a SW engineer and the source code.

Smart Comment Example

S-APL source code sample (rule)

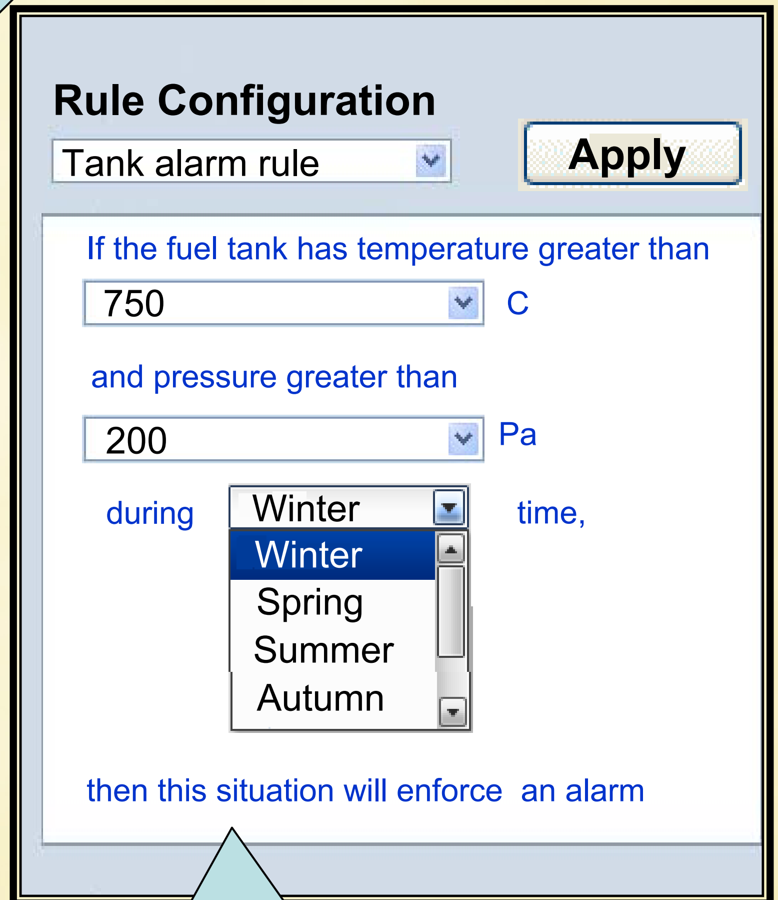
S-APL

S-APL comment

```
/* If the fuel tank has temperature
greater than 750 C and pressure
greater than 200 Pa during
Winter time, then this situation
will enforce  an alarm */
```

```
{
Tnk1 has_temperature ?T .
Tnk1 has_pressure ?P .
?T > 750 .
?P > 200 .
Season is "Winter"
}
 → {I do alarm.rab}
```

S-APL rule



Generated user interface for editing the rule in runtime

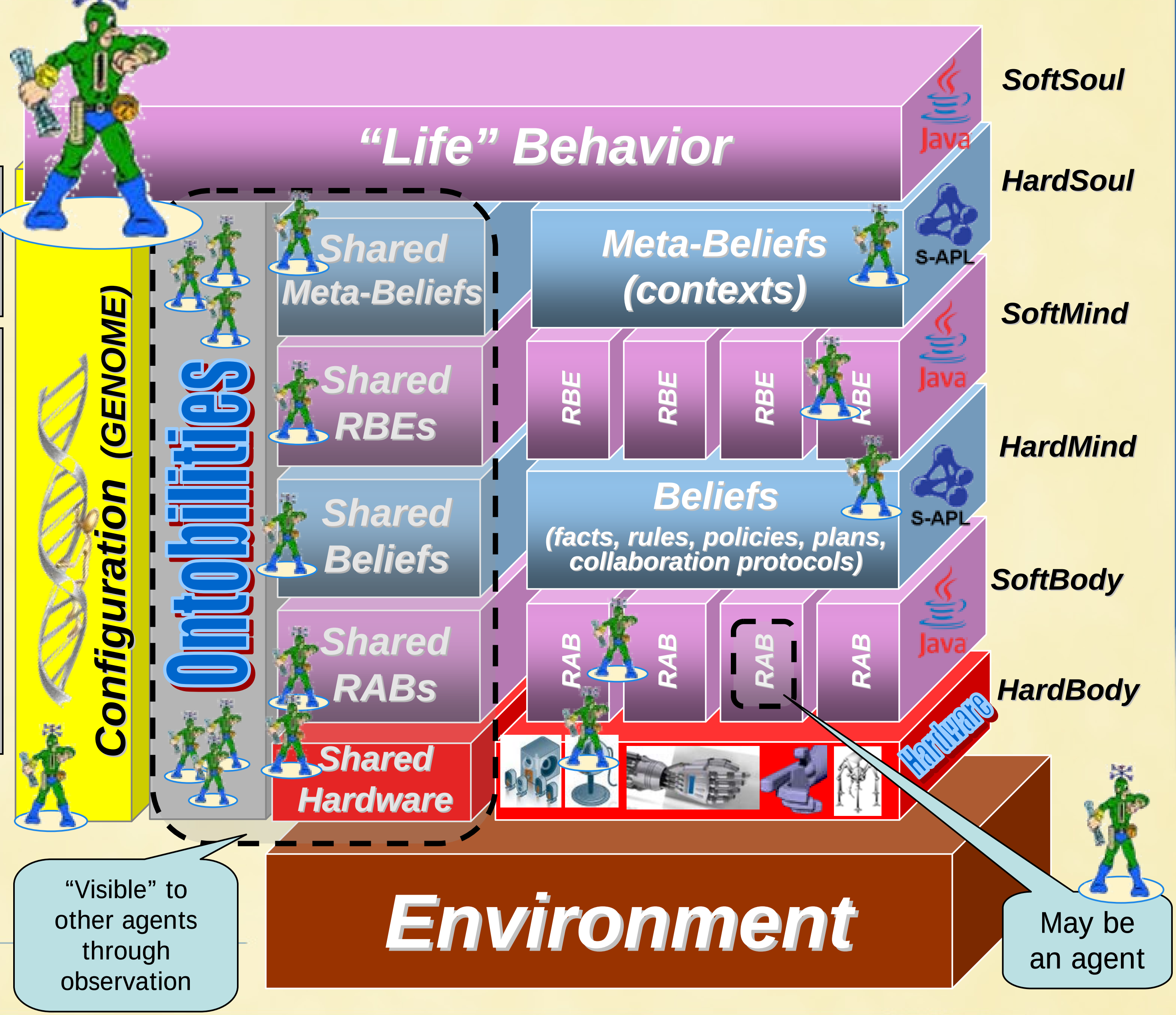
UbiDubi: UBIWARE-driven-UBIWARE (i.e. UBIWARE architectural components are also agent-driven)

RAB – Reusable Atomic Behavior

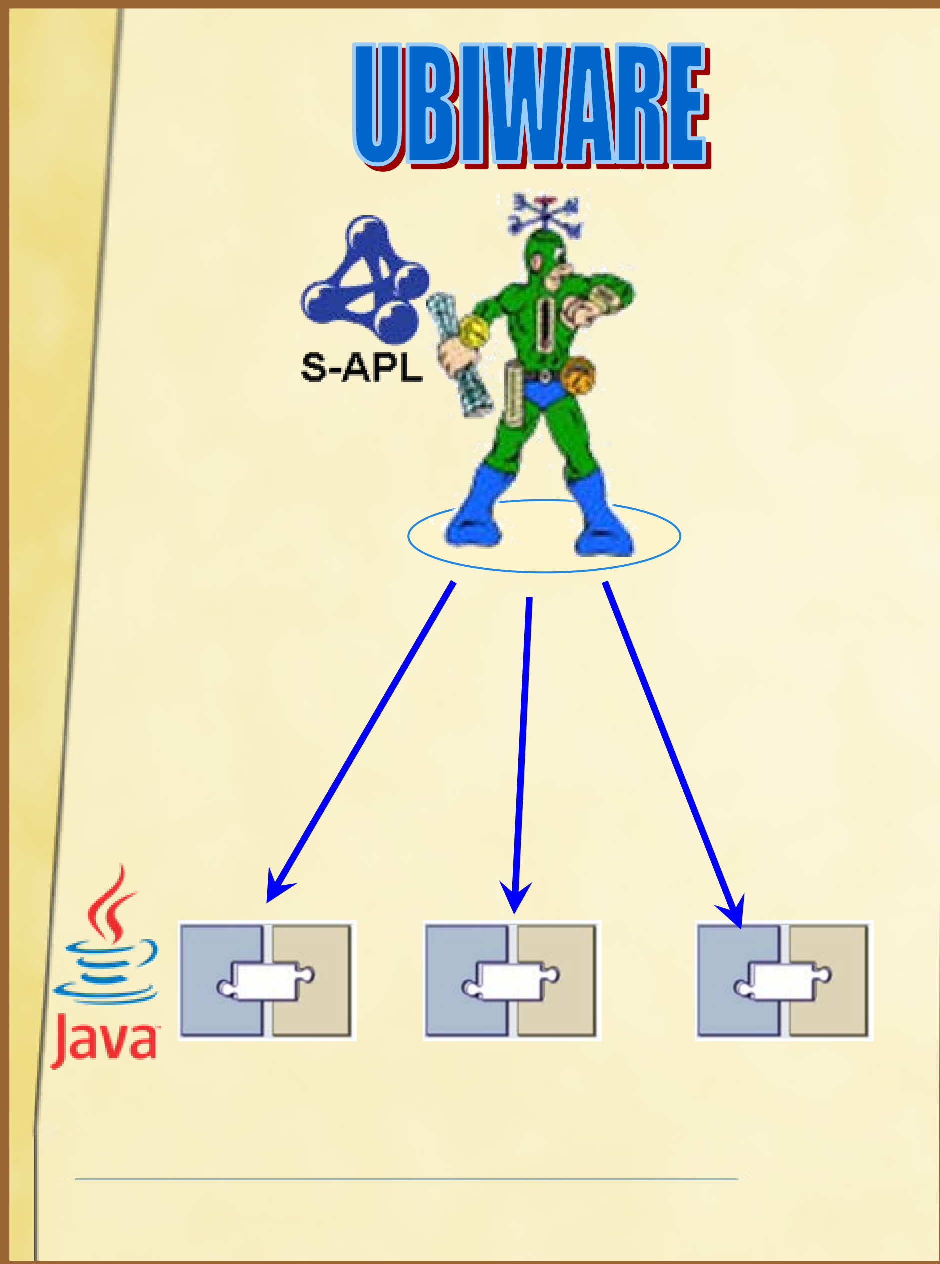
RBE – Reusable Behavior Engine

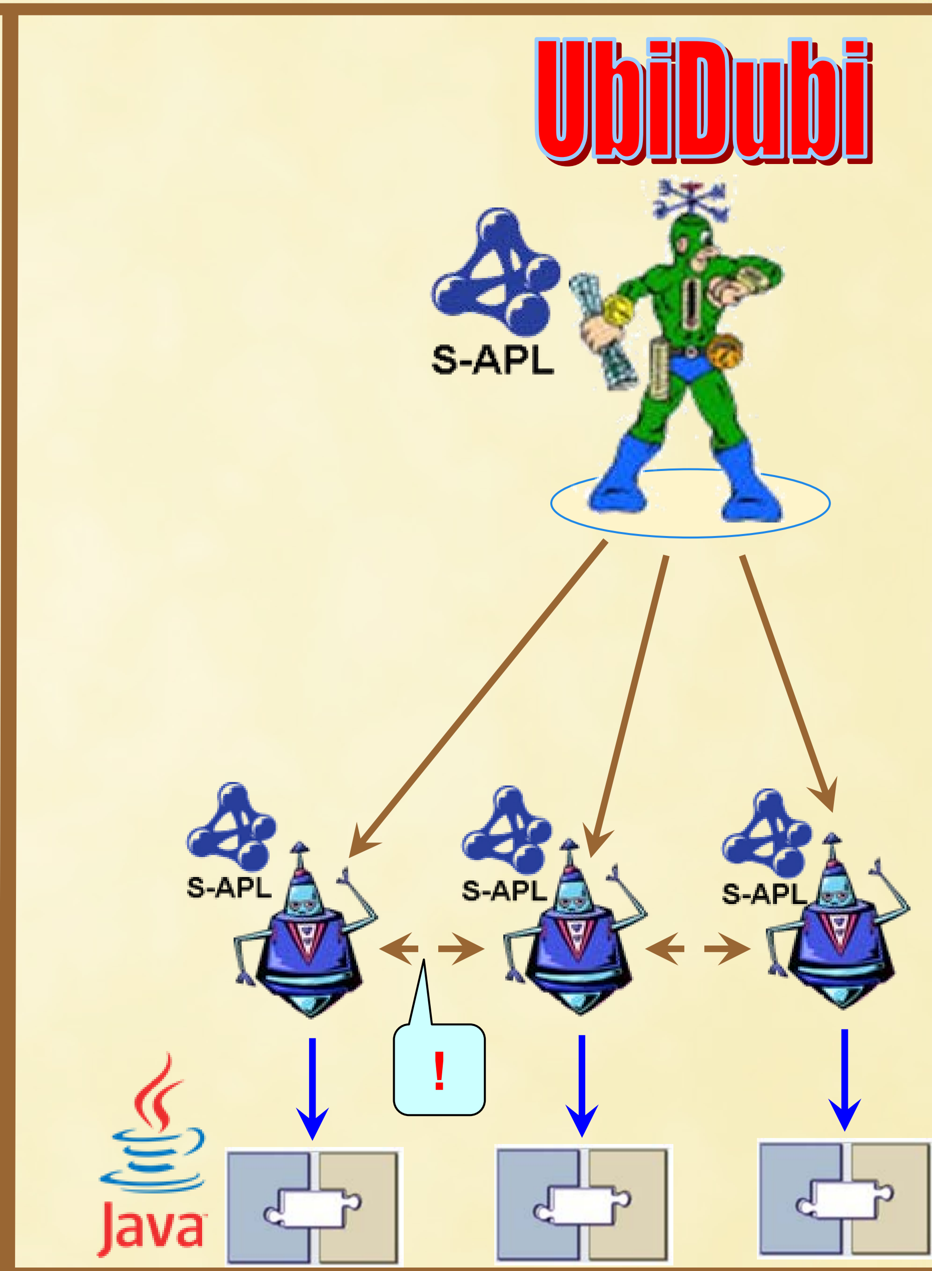
Ontobility is self-contained, self-described, semantically marked-up **proactive** agent capability (agent-driven ***ontonut***), which can be “seen”, discovered, exchanged, composed and “executed” (internally or remotely) across the agent platform in a task-driven way and which can perform social utility-based behavior

Genome is part of semantically marked-up agent configuration settings, which can serve as a tool for agent evolution: inheritance crossover and mutation



UBIWARE vs. UbiDubi: which and when is better?

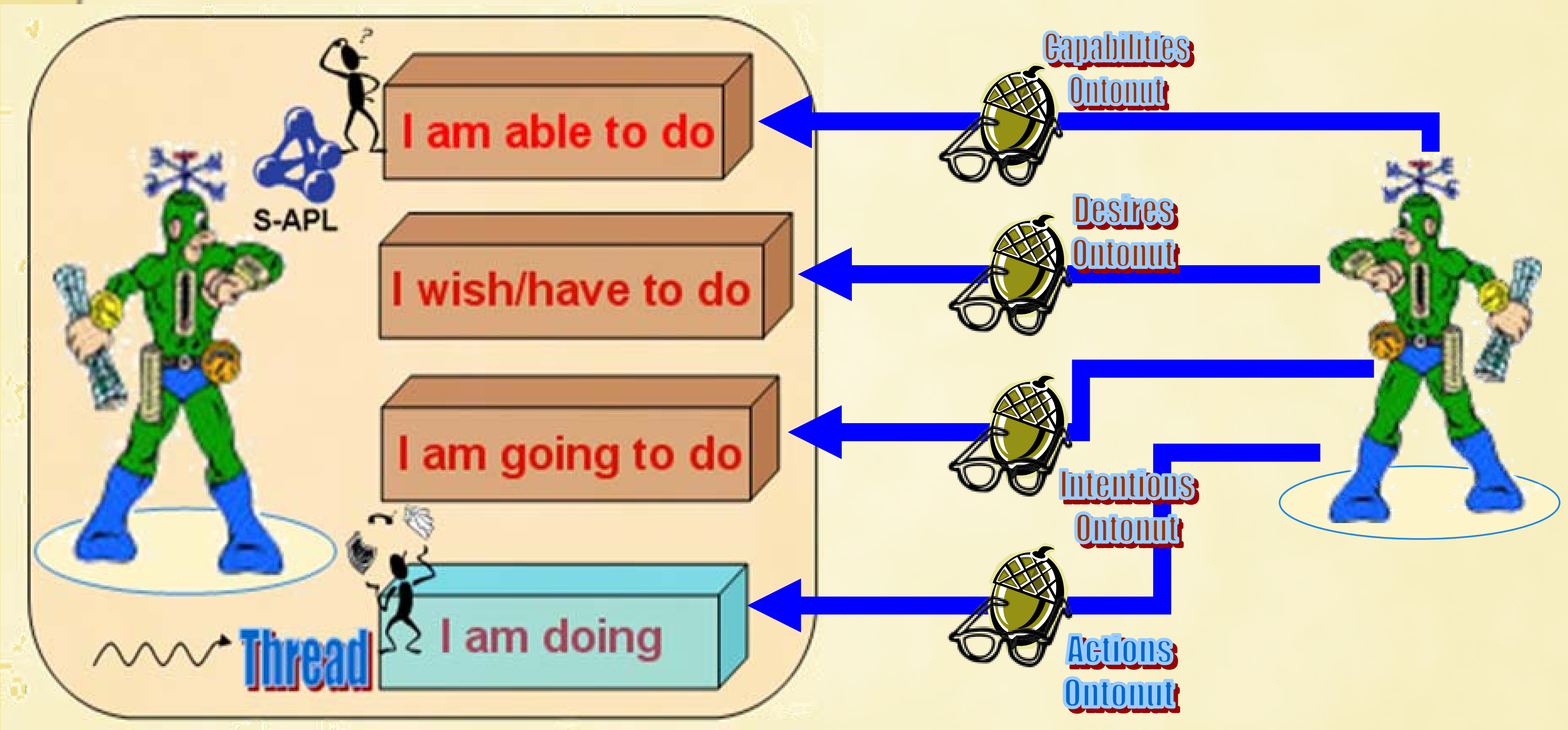




Observation capability in UBIWARE

- Middleware for distributed systems should have capability for information sharing between distributed entities;
- Information sharing is possible either through communication (direct (message passing) or indirect (shared memory)) or through observation;
- Communication capability is already supported by JADE platform, which is basis for UBIWARE;
- Observation capability is missing in JADE although such a capability has several specific advantages over communication especially when talking about inter-agent coordination (when agents should be aware in real time about each other intentions and actions).

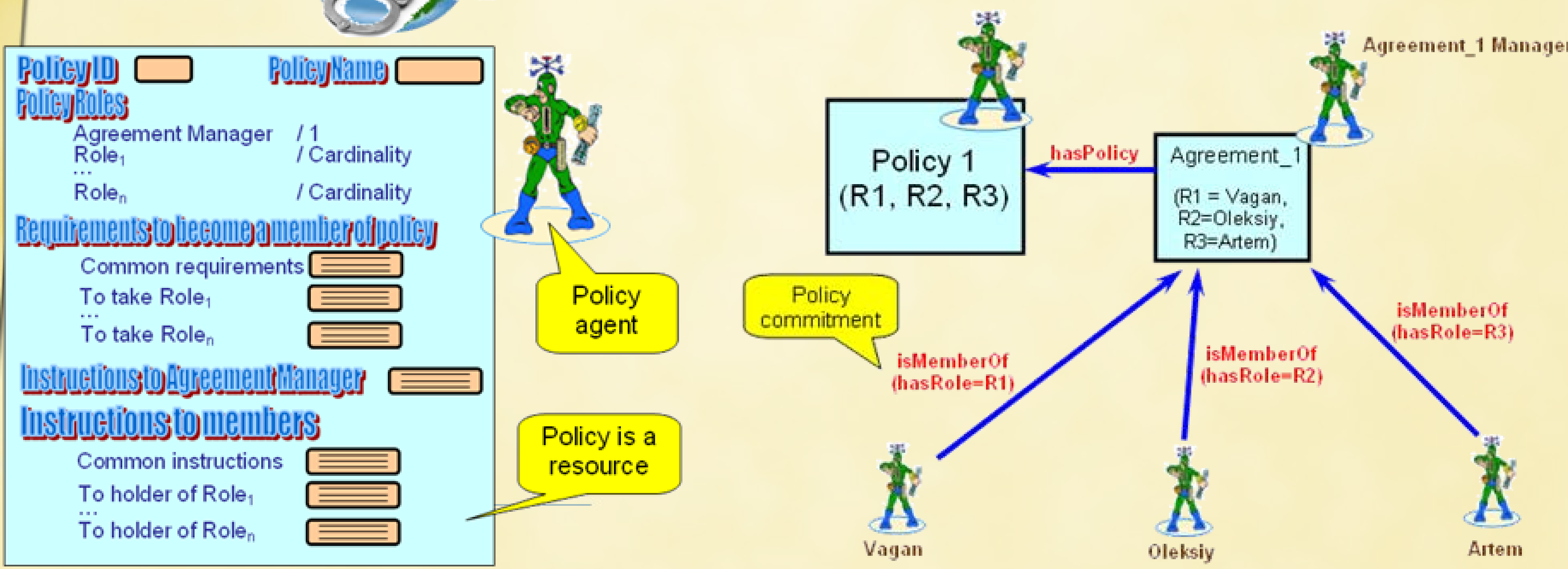
Observation Ontonuts for "Silent Communication"



Coordination through observation will be possible

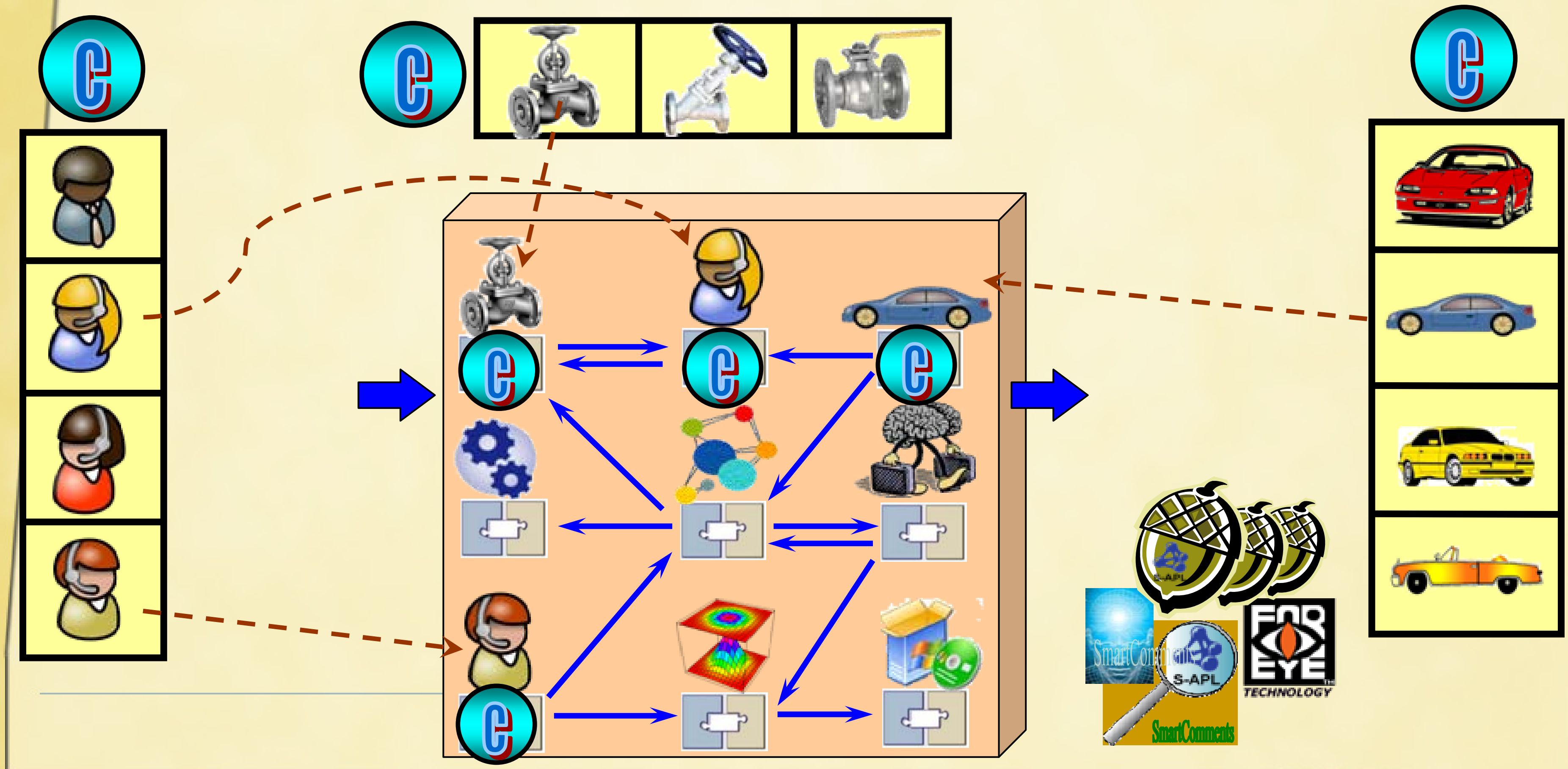
Policy-Based Control in UBIWARE

- Policy creation and annotation in S-APL;
- Policy enforcement engine;
- Policy-driven MAS scenario management;
- Policy-driven (re)configuration;



UBIWARE 3.0 (2009-2010) vision of future platform (3-rd project year plan)

UBIWARE 3.0 supposed to be a platform for creating and executing configurable distributed systems based on generalized and reusable business scenarios, which heterogeneous components (actors) are not predefined but can be selected, replaced and configured in runtime.

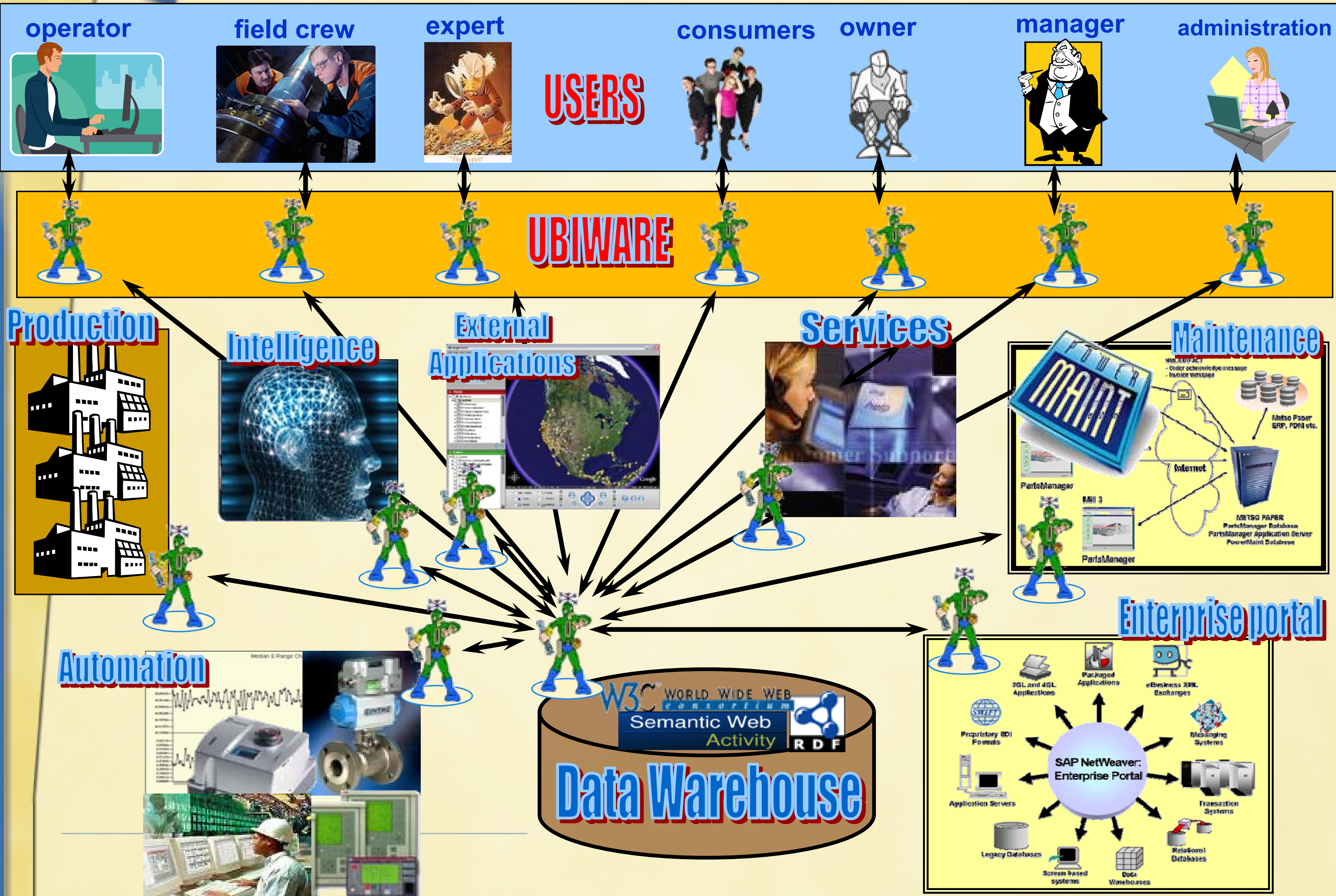


Implementing Industrial Cases

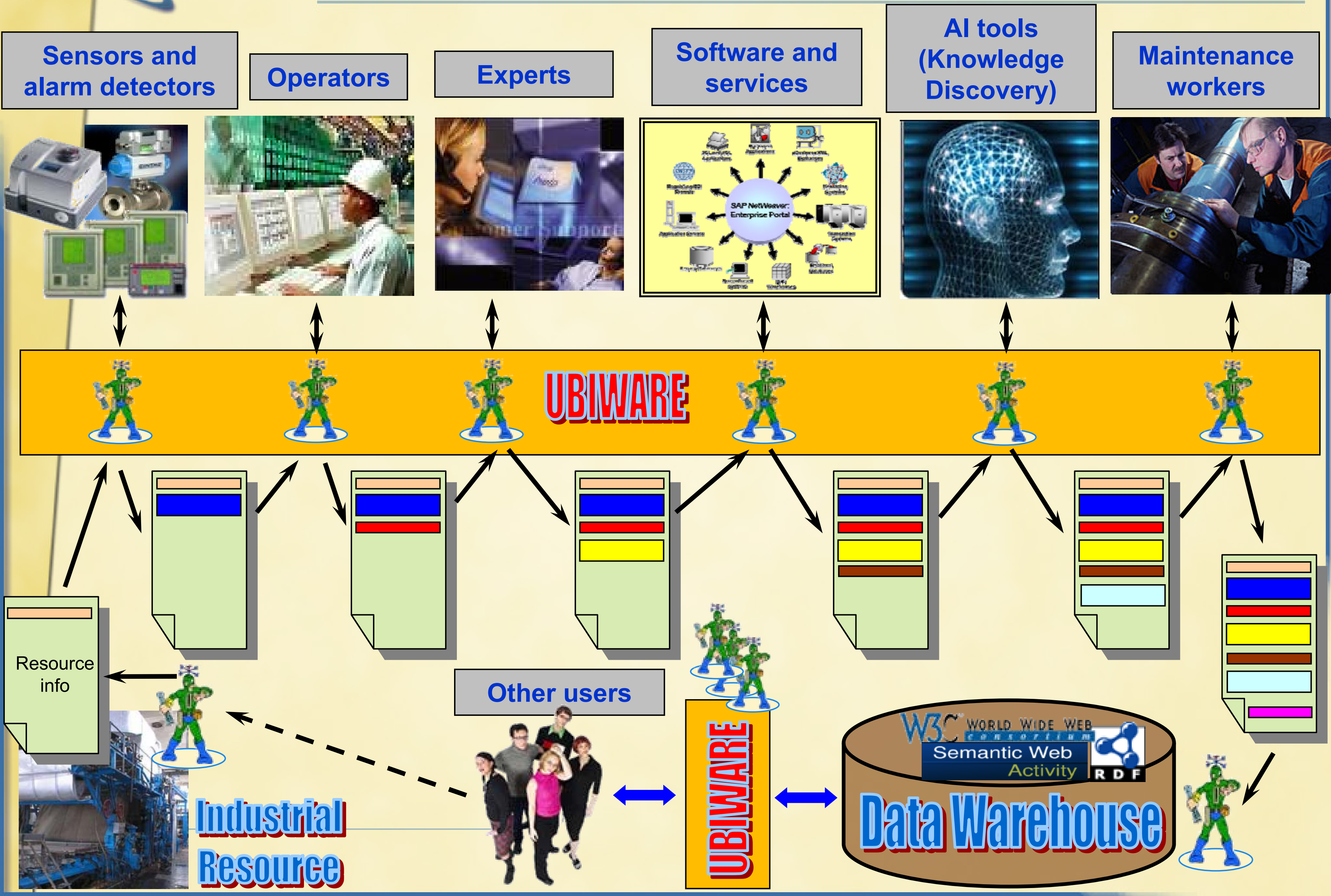
on top of UBIWARE platform

(cases shown as they were in 2008)

Agent-driven EAI



Agent-driven resource life-cycle management



UBIWARE2All: Human-Centric Architecture

Example: UBIWARE2ABB Case

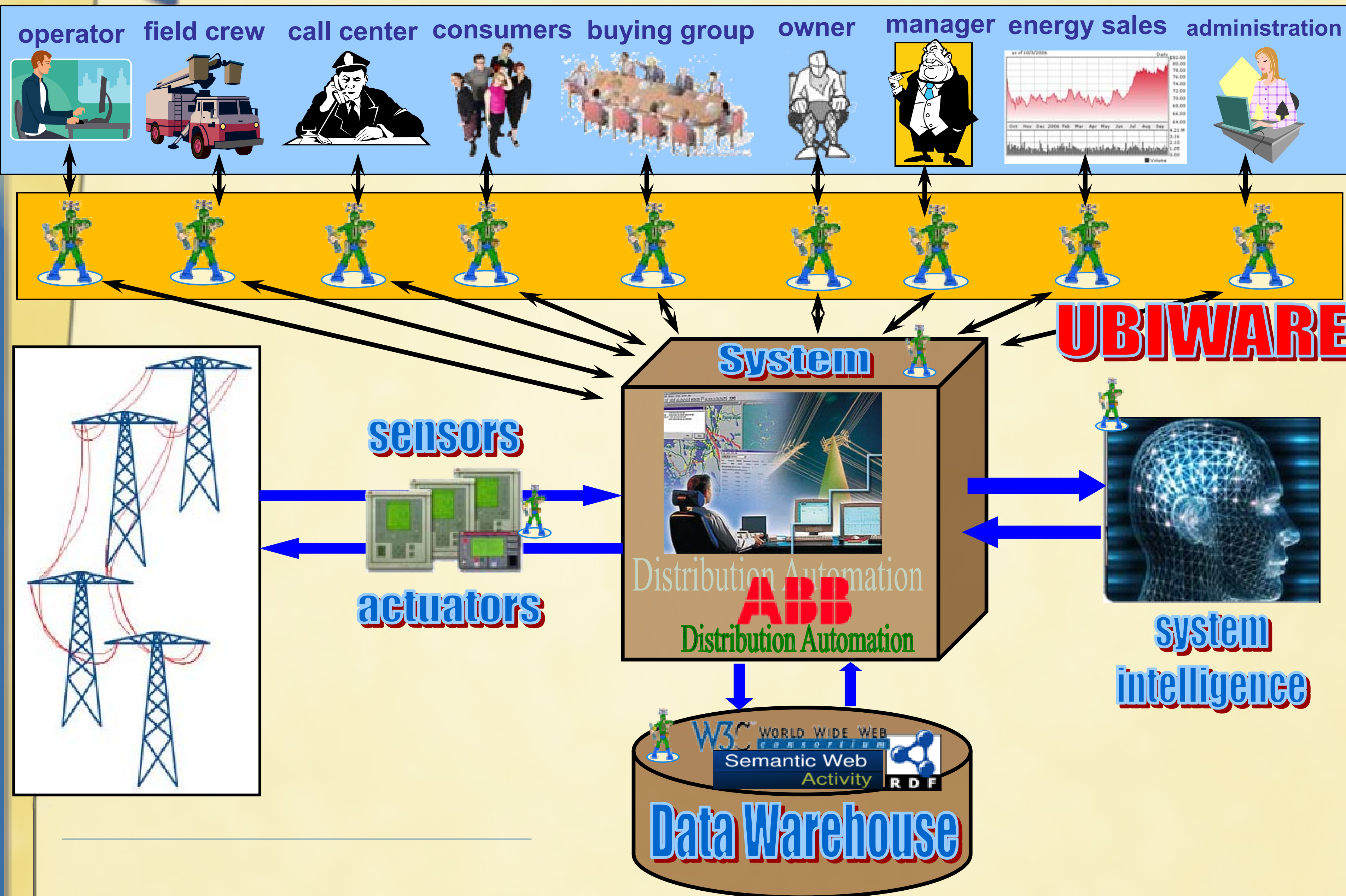
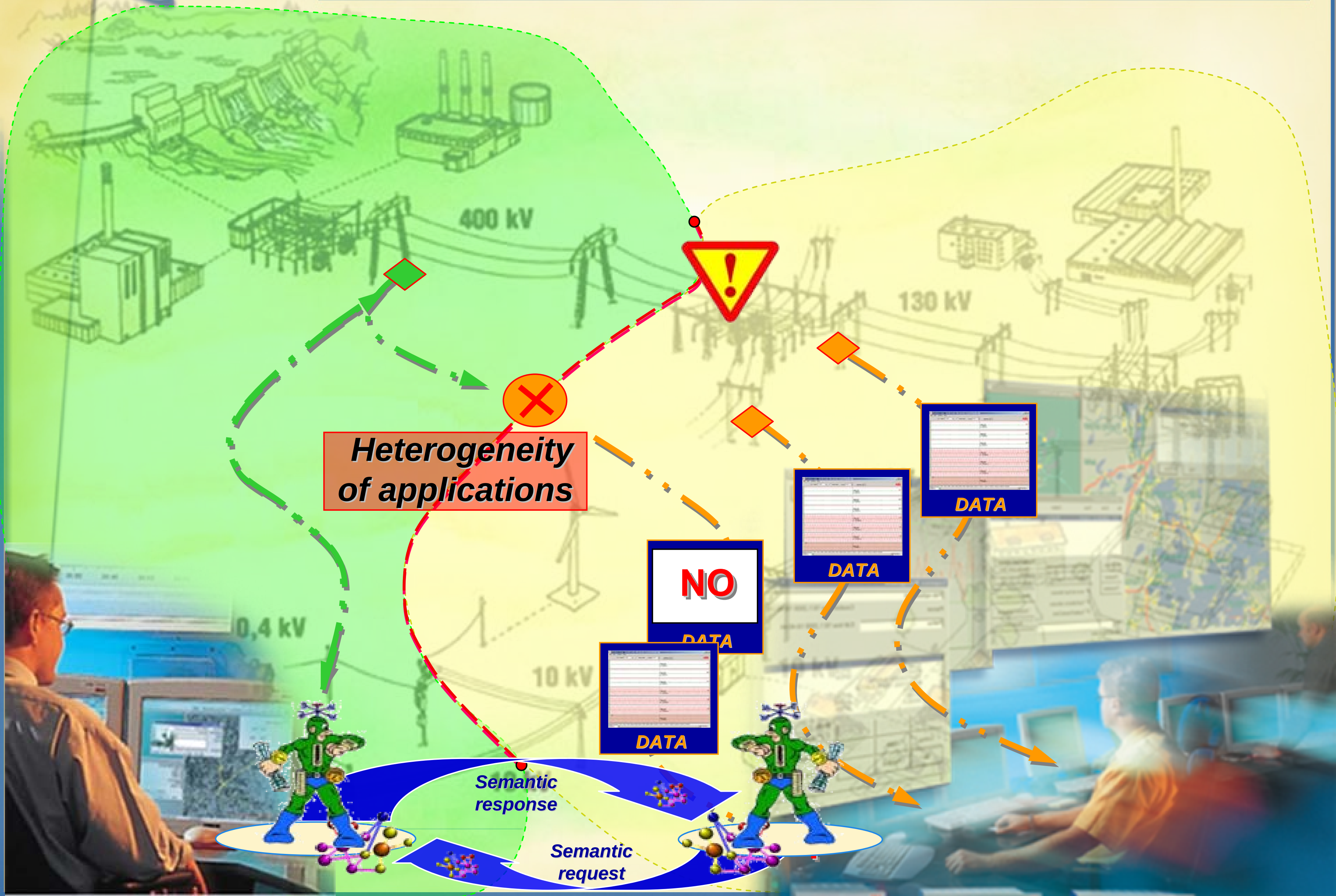


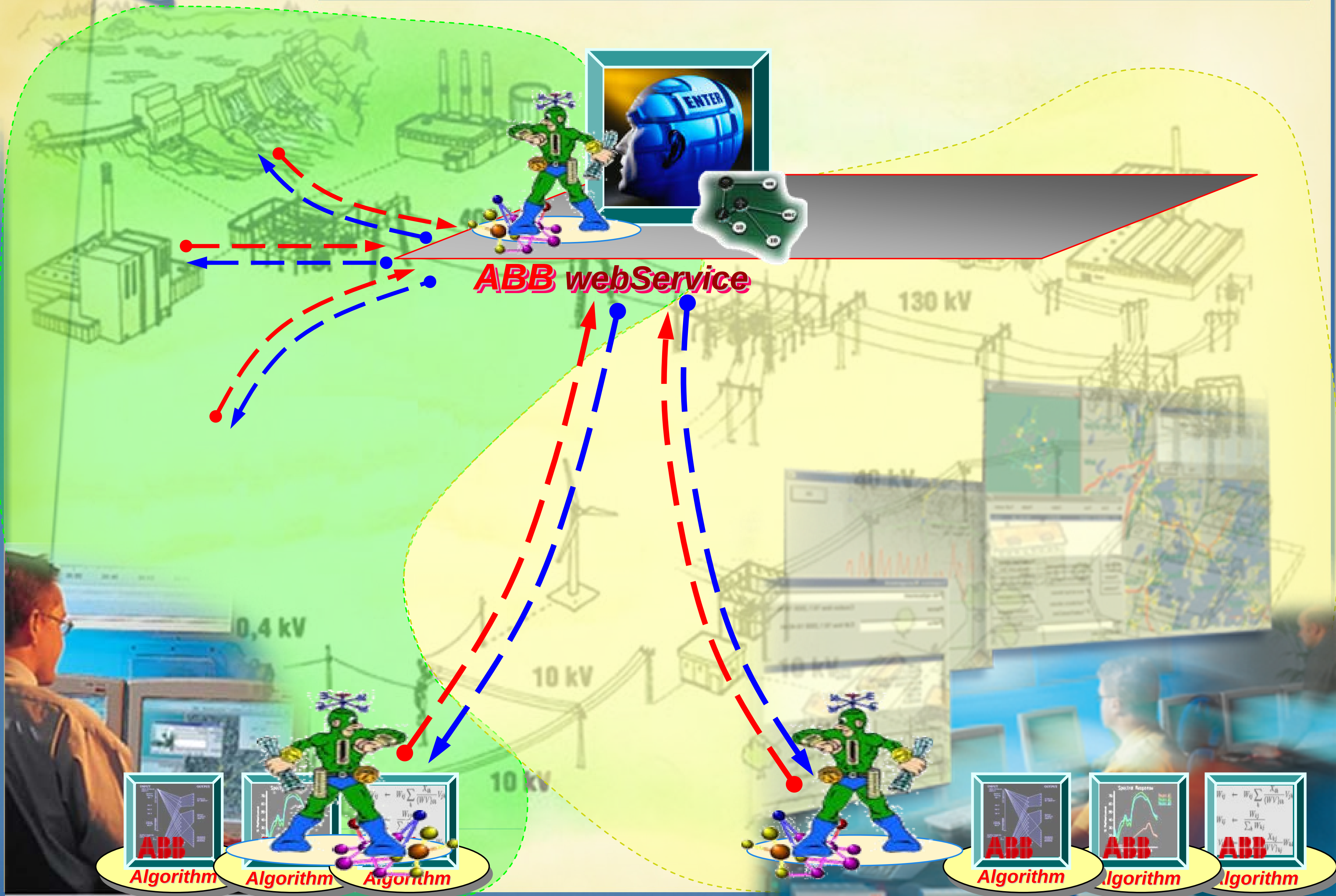
ABB Case: Potential add-value

- ***Networks interoperability:*** sharing information between sub-networks of the integral power network.
- ***New business models:*** possibility of providing some of the ABB knowledge in the form of web-services.
- ***Data integration:*** integration of data that is utilized currently with various contextual information - for risk analysis, facilitation of fault localization, and other.
- ***Interface enhancement:*** including dynamic, e.g., geographic information.
- ***Knowledge transferring:*** possibility of implementing web-services learning from the human experts (e.g. to speed-up the decision making process).
- ***Flexibility of configuration:*** flexibly defining the interaction protocols of all the participating entities (devices, services, humans)

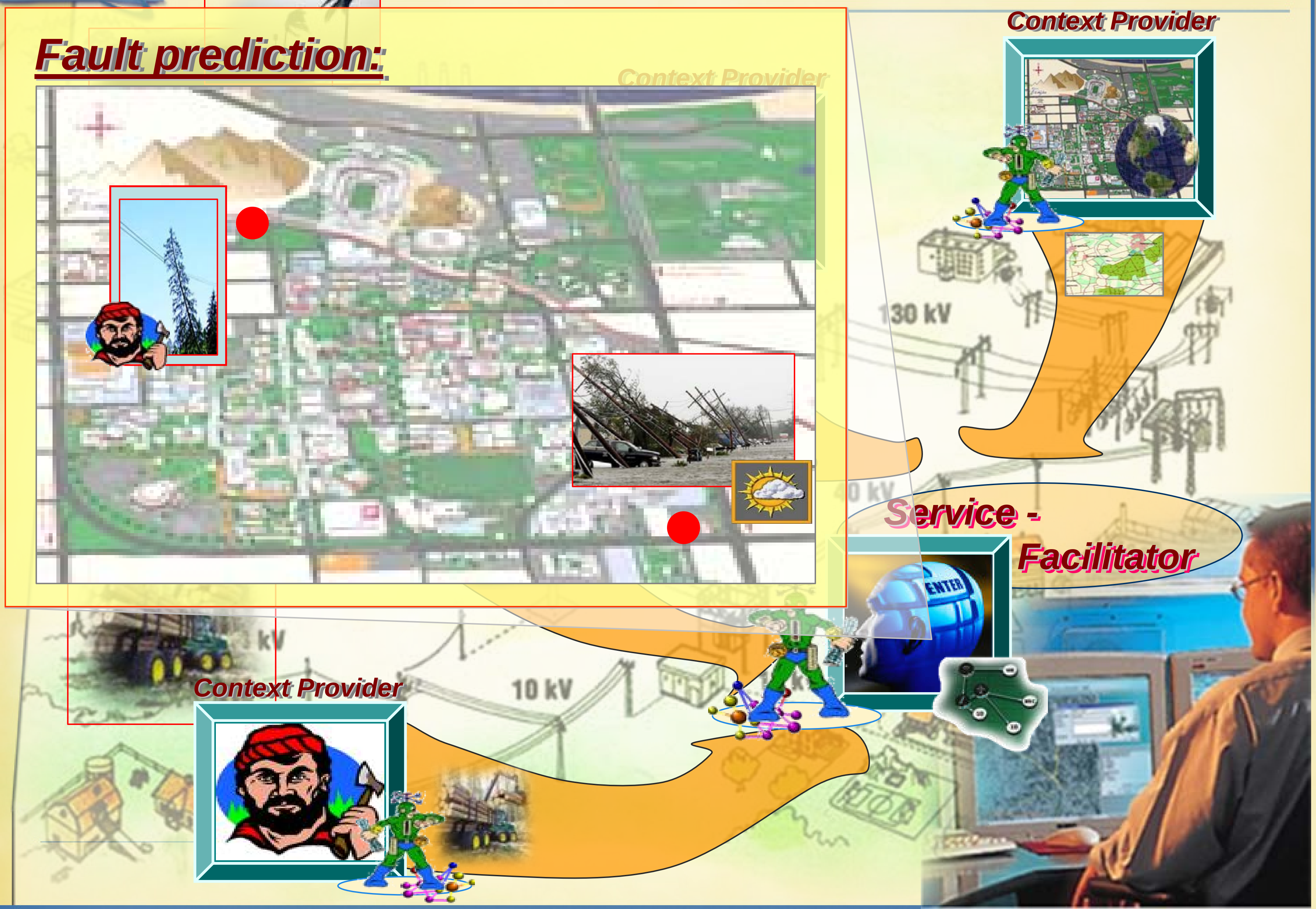
Networks interoperability



New business models



Data integration for risk analysis

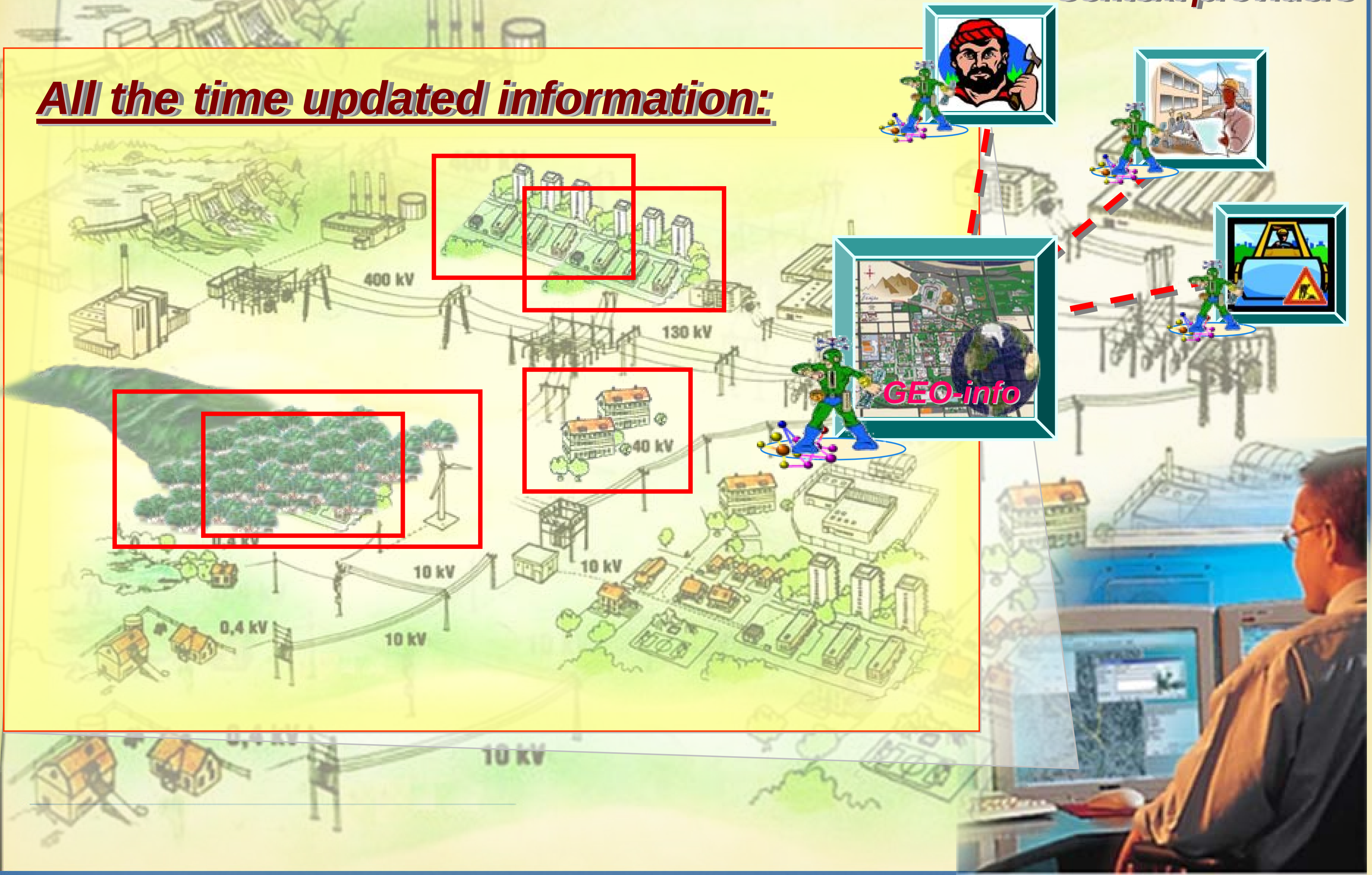


Data integration for fault localization

Interface enhancement

Context providers

All the time updated information:



Knowledge transferring

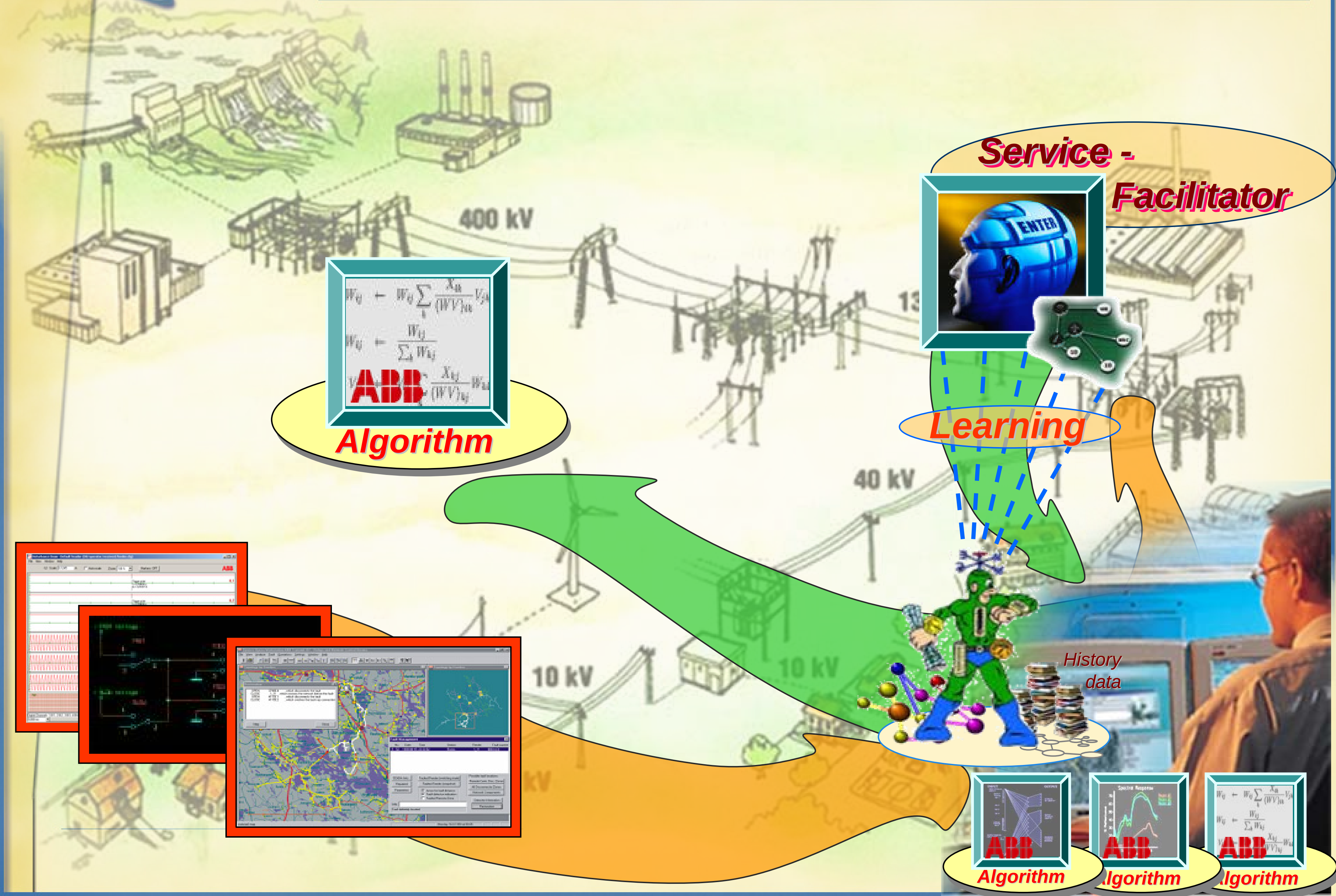


ABB case architecture

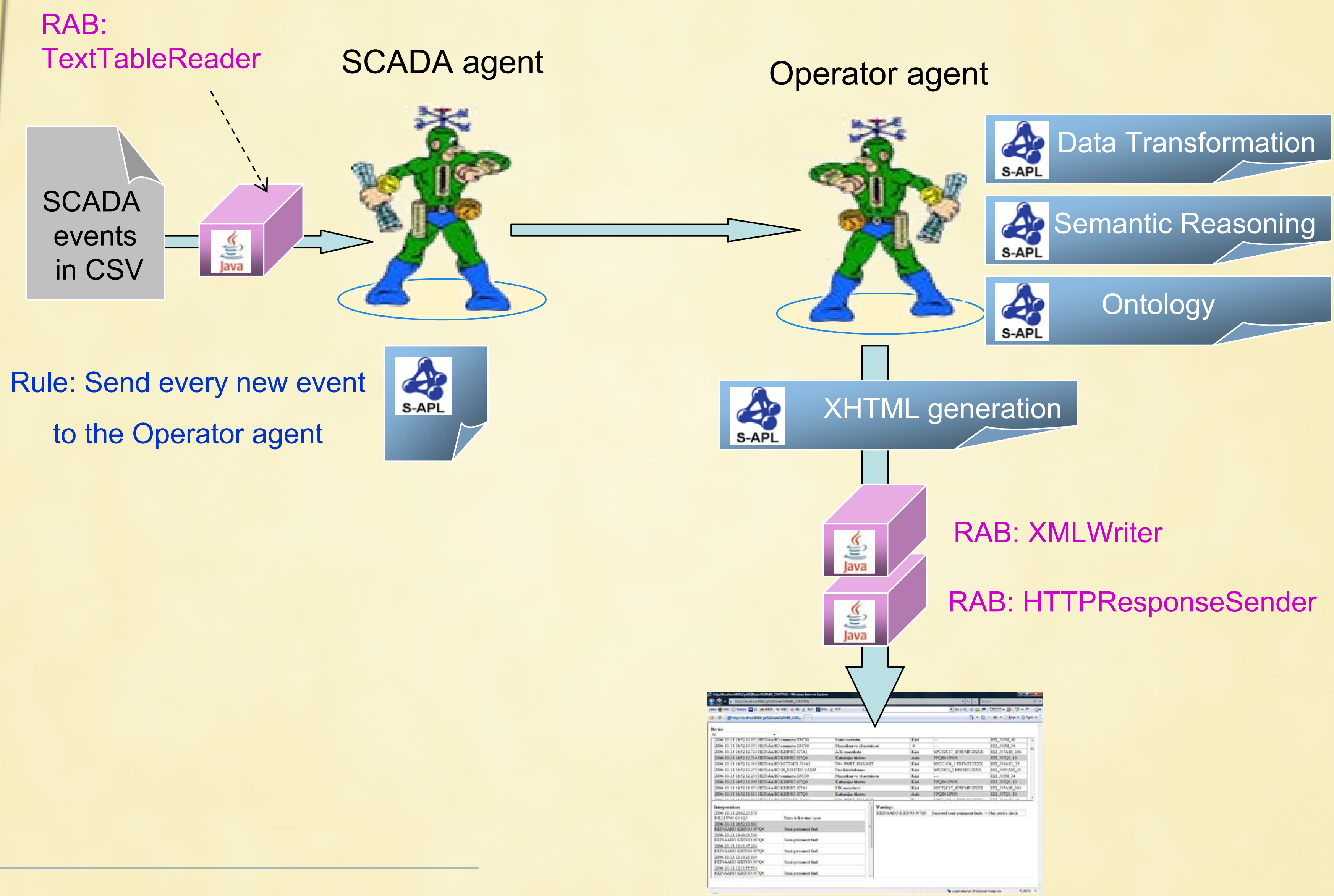


ABB case challenges

- Already provides a demonstration of how S-APL integrates *programming* with *semantic reasoning*:
  - Normal programming tasks like data access, data transmission between agents, HTML interface production.
  - Semantic reasoning – interpretation of data.
- Is a good case for demonstrating the *configurability* (WP4) of UBIWARE:
  - Many “constants” used now in reasoning are customer-specific and can be changed – so must be easily (re-) configurable

ABB case: *interface 1*

http://localhost:8081/get%20state%20ABB_CONTROL - Windows Internet Explorer

http://localhost:8081/get%20state%20ABB_CONTROL

Links FMI Foreca 10 IMDB WIKI GR TUT UTA VTT

Device

All

2006-10-13 16:52:14.351	HEINAAHO sammutus EFC30	Säätö suoritettu	Kiini	---	HEI_J00M_40
2006-10-13 16:52:14.351	HEINÄAHO sammutus EFC30	Maasulkuarvo yli asetuksen	0	---	HEI_J00M_34
2006-10-13 16:52:12.724	HEINÄAHO KIHNIÖ J07A1	AJK maasulusta	Kiini	SPCT2C17_1FRFMFUIXXX	HEI_J07A1R_166
2006-10-13 16:52:12.714	HEINAAHO KIHNIO J07Q0	Katkaisijan tilatieto	Auki	FPQB01IPOS	HEI_J07Q0_10
2006-10-13 16:52:12.480	HEINÄAHO MITTAUS J04A3	U0>-PORT. HAVAHT.	Kiini	SPCU1C6_1 FRFMFUIXXX	HEI_J04A1U_15
2006-10-13 16:52:12.275	HEINAAHO 20_SYOTTO VAMP	Uusi häiriötallennus	Kiini	SPCJ3C3_1 FRFMFUIXXX	HEI_J06VAM_20
2006-10-13 16:52:12.230	HEINÄAHO sammutus EFC30	Maasulkuarvo yli asetuksen	Kiini	---	HEI_J00M_34
2006-10-13 16:52:11.995	HEINAAHO KIHNIO J07Q0	Katkaisijan tilatieto	Kiini	FPQB01IPOS	HEI_J07Q0_10
2006-10-13 16:52:11.673	HEINÄAHO KIHNIÖ J07A1	PJK maasulusta	Kiini	SPCT2C17_1FRFMFUIXXX	HEI_J07A1R_163
2006-10-13 16:52:11.663	HEINAAHO KIHNIO J07Q0	Katkaisijan tilatieto	Auki	FPQB01IPOS	HEI_J07Q0_10

Interpretations

2006-10-13 20:02:23.576 KILLI PM1 G01Q0	Relay is first-time open
2006-10-13 16:52:11.663 HEINAAHO KIHNIO J07Q0	Semi-permanent fault
2006-10-13 14:46:09.506 HEINAAHO KIHNIO J07Q0	Semi-permanent fault
2006-10-13 13:41:45.294 HEINAAHO KIHNIO J07Q0	Semi-permanent fault
2006-10-13 13:10:26.680 HEINAAHO KIHNIO J07Q0	Semi-permanent fault
2006-10-13 12:11:55.554 HEINAAHO KIHNIO J07Q0	Semi-permanent fault

Warnings

HEINAAHO KIHNIO J07Q0	Repeated semi-permanent faults -> May need a check

Local intranet | Protected Mode: On 100%

ABB case: *interface 2*

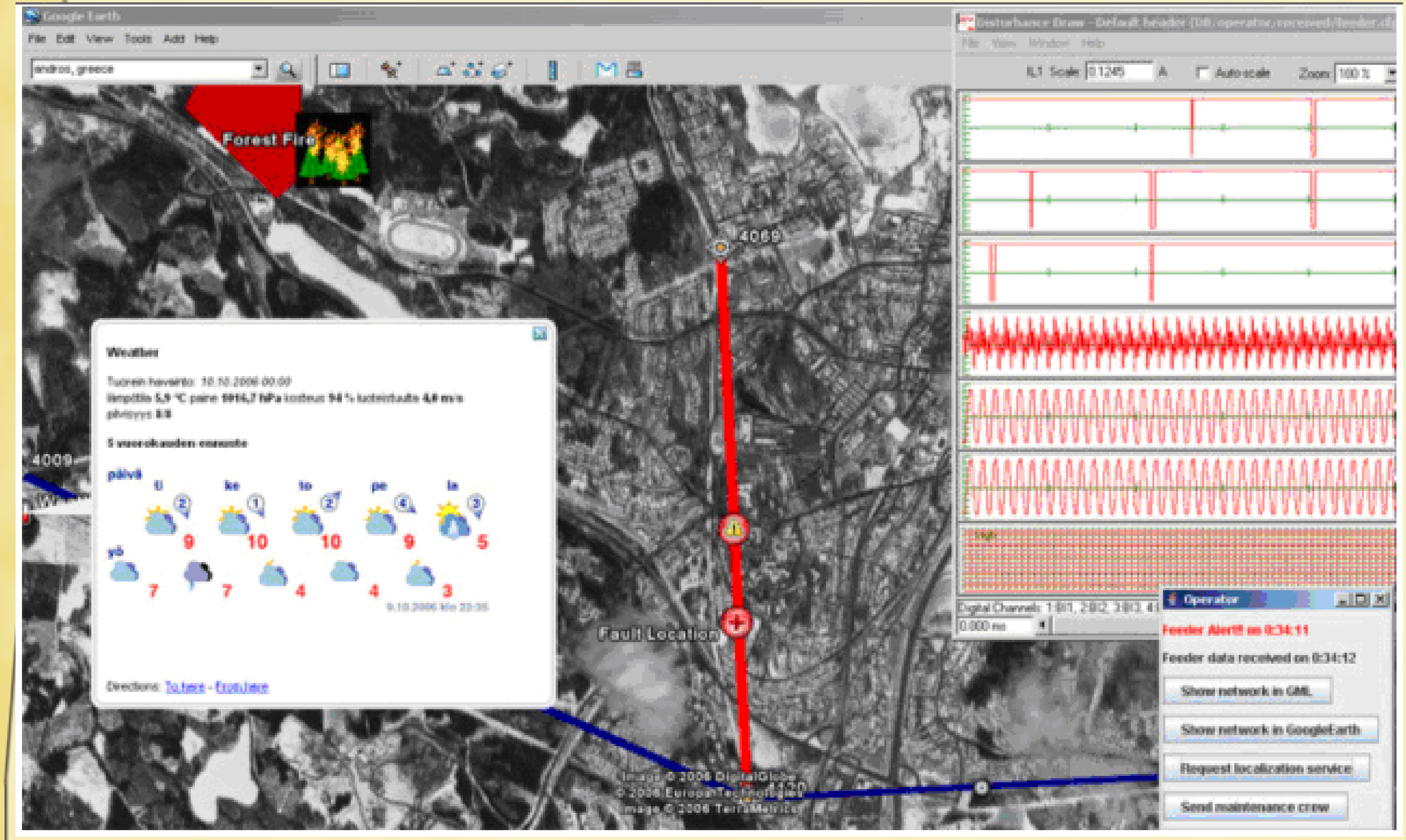
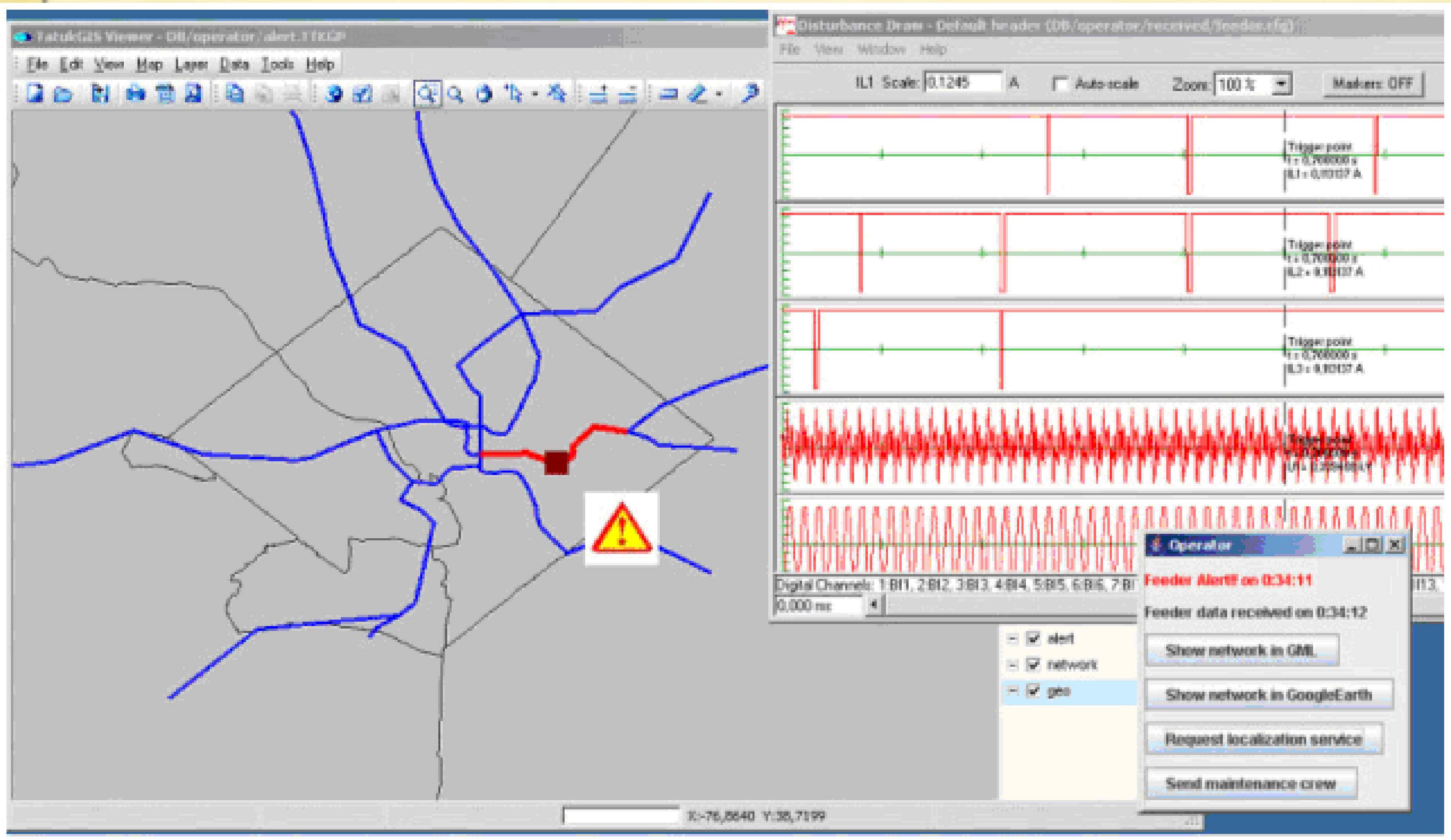
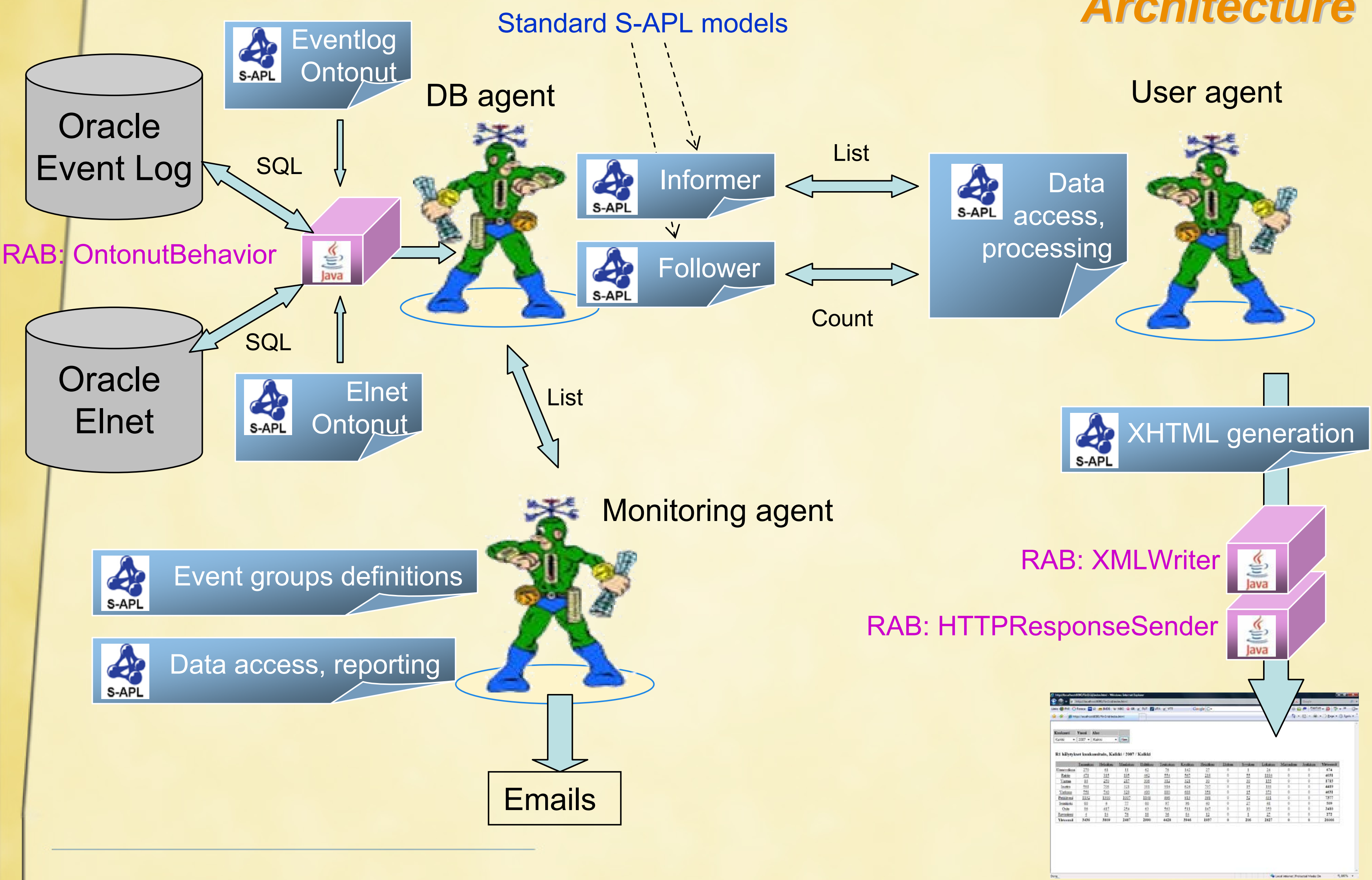


ABB case: *interface 3*



Fingrid case

Architecture



Agents in Fingrid case

- ***DB agent*** is responsible for interfacing with the databases. Implementation of this agent is based on UBIWARE’s **ontonuts** approach. With this approach:
  - DB agent receives from other agents queries that are formulated semantically and encoded using S-APL. The data is sent back to the requestors also in a semantic S-APL form.
  - The databases (relational, non-semantic) are provided each with an *ontonut*, which is a description of the database schema that is sufficient for translating between S-APL semantic queries and SQL as well as between database responses and a needed semantic form.
  - Reusable java component OntonutBehavior takes care of generating SQL and translating responses. If S-APL query concerns both databases, OntonutBehavior two generates SQL sub-queries and cross-joins the results.
- It is notable that given that the interface of DB agent towards other agents is provided by standard S-APL models *Follower* and *Informer*, DB agent does not have any single line of code (either Java or S-APL) that would be written just for it. The only tailored elements are ontonuts, which are declarations, not behavioral code.
- ***User agent*** is responsible for providing XHTML interface to a human user. User agent receives user queries, interacts with DB agent, and presents the data to the user.
- ***Monitoring agent*** is an autonomously operating agent which is responsible for continuously checking the new events appearing in the Eventlog database and sending email notifications.

Fingrid Case: Screenshots

FINGRID

Event Groups | Operation Counts

Vuosi	Kuukausi	Työaika	Laji	Alue	
2007	Kaikki	Työaikana	Vain viat	Kaikki	Hae

R1 hälytykset kuukausittain, 2007 / Kaikki / Työaikana / Vain viat / Kaikki

	Tammikuu	Helmikuu	Maaliskuu	Huhtikuu	Toukokuu	Kesäkuu	Heinäkuu	Elokuu	Syyskuu	Lokakuu	Marraskuu	Joulukuu	Yhteensä
HÄME Laitteet	0	0	0	0	0	0	0	0	0	19	8	31	**58**
LOUNAIS-SUOMI Laitteet	0	0	0	0	0	0	0	0	0	605	183	105	**893**
UUSIMAA Laitteet	0	0	0	0	0	0	0	0	0	53	58	30	**141**
KAAKKOIS-SUOMI Laitteet	0	0	0	0	0	0	0	0	0	154	38	32	**224**
ITÄ-SUOMI Laitteet	0	0	0	0	0	0	0	0	0	68	56	149	**273**
LÄNSI-SUOMI Laitteet	0	0	0	0	0	0	0	0	0	25	59	88	**172**
LÄNSI-SUOMI SJ Laitteet	0	0	0	0	0	0	0	0	0	39	37	43	**119**
POHJOIS-POHJANMAA Laitteet	0	0	0	0	0	0	0	0	0	50	36	24	**110**
LAPPI Laitteet	0	0	0	0	0	0	0	0	0	15	2	8	**25**
Yhteensä	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1028**	**477**	**510**	**2015**

Equipment alarms: Table view

FINGRID

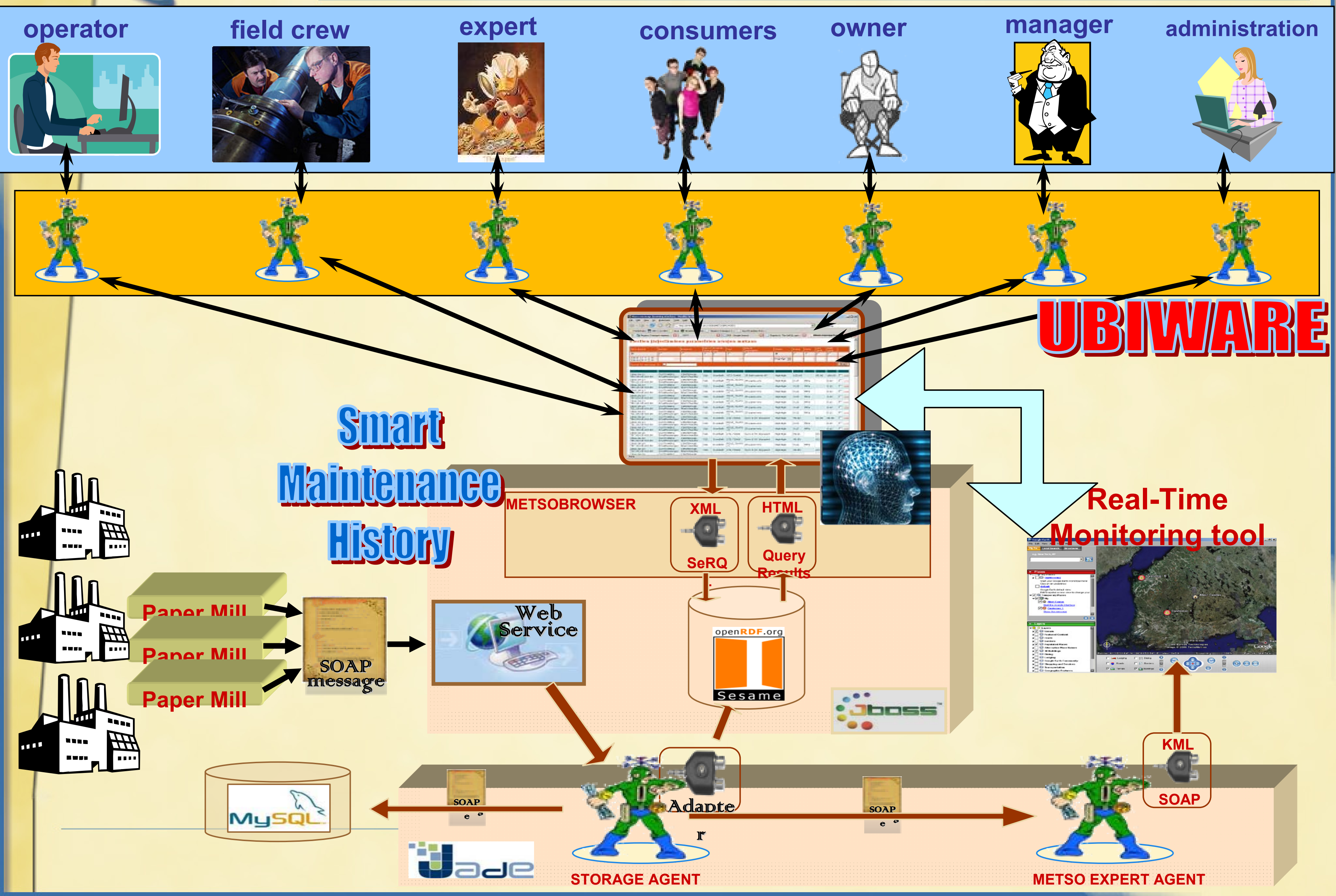
Event Groups | Operation Counts

Vuosi	Kuukausi	Työaika	Laji	Alue	
2007	Kaikki	Työaikana	Vain viat	Kaikki	Hae

R1 hälytykset alueessa HÄME Laitteet: 2007 / Marraskuu / Työaikana / Vain viat, Yhteensä: 8
01.11.07 10:35:09.988 T R HY TAK1 MAASULKU ..E Hälyttää
20.11.07 08:03:31.128 T R FO T1 KAASUPIT. MITTAUSVIKA ..E Vika
20.11.07 08:03:32.051 T R FO 11KQO KATKAISIJA ..E Lauennut
20.11.07 08:11:46.726 T R FO 12KQO KATKAISIJA ..E Lauennut
20.11.07 08:11:47.229 T R FO T1 KAASUPIT. MITTAUSVIKA ..E Vika
23.11.07 08:13:12.745 T R MEL DC/AC INVERTTERI ..E Vika
26.11.07 10:53:25.785 T R HML LVI-HÄLYTYS EI KIIRE ..E Hälyttää
27.11.07 14:50:27.261 T R HML LVI-HÄLYTYS EI KIIRE ..E Hälyttää
Yhteensä: 8

Equipment alarms: List view

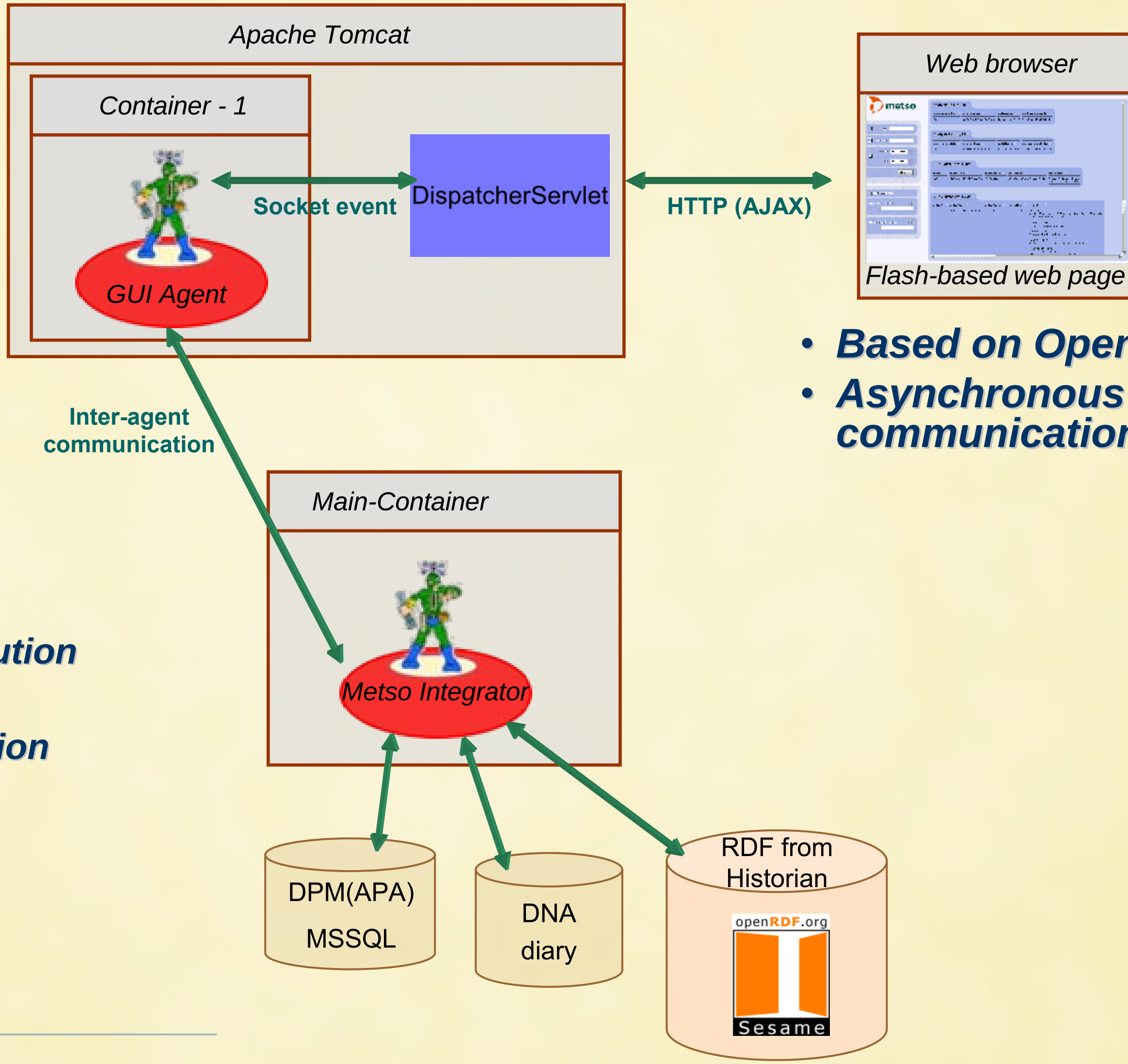
Example: UBIWARE2Metso Case



Metso Automation case

Architecture

- ***Mediating user requests***

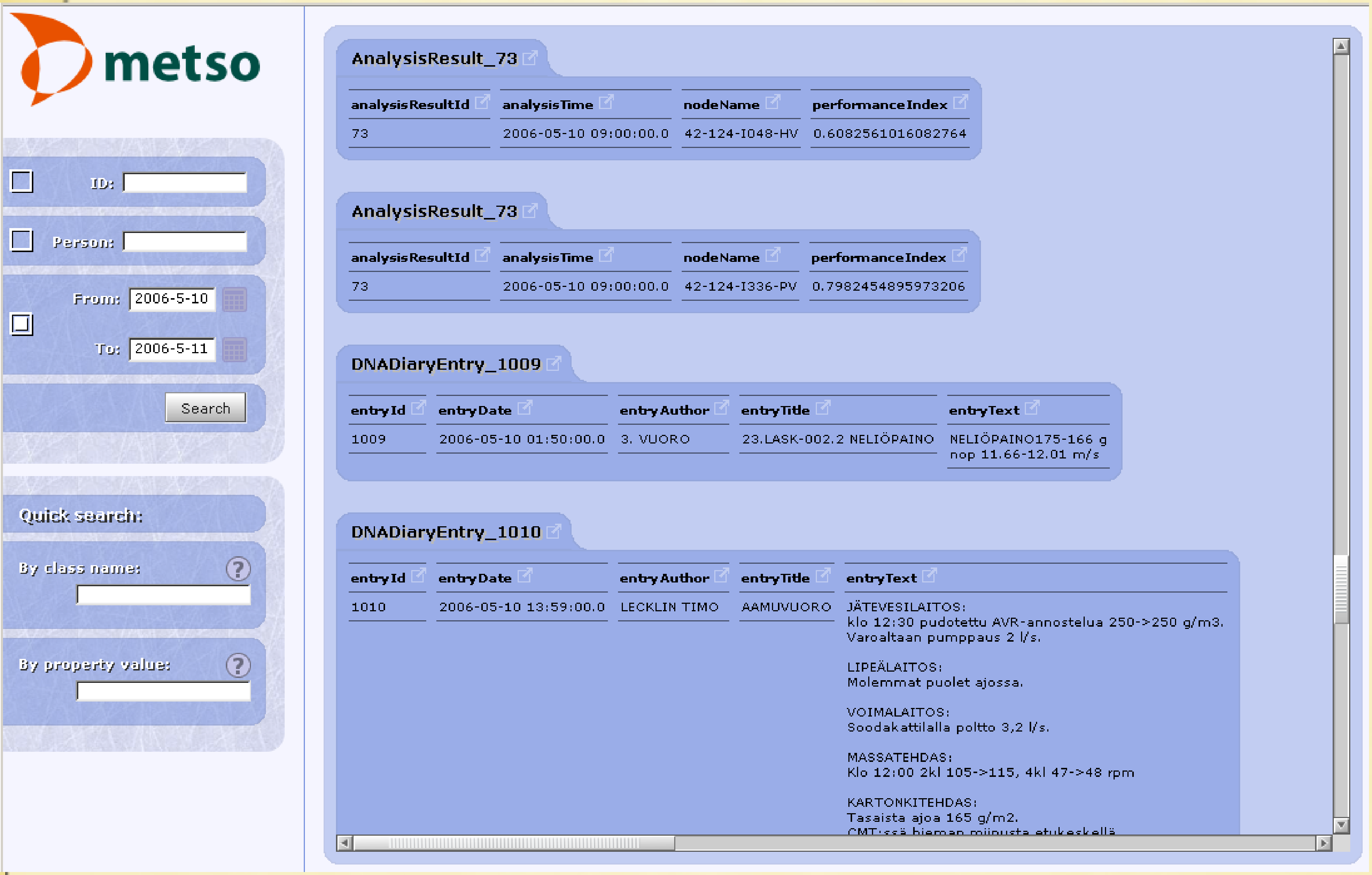


- ***Based on OpenLaszlo***
- ***Asynchronous communication model***

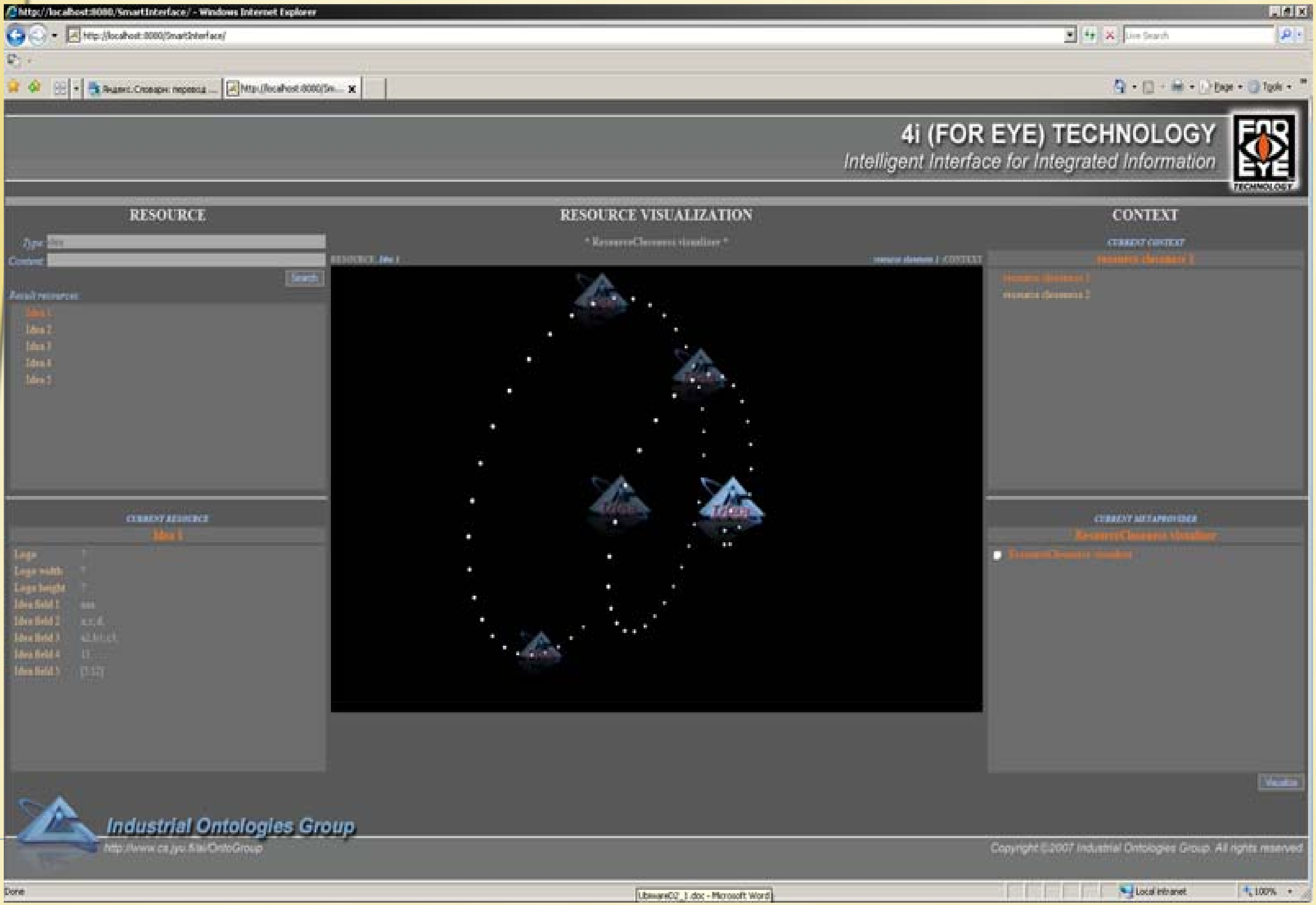
- ***Ontonuts engine***
  - ***Planning***
  - ***Query execution***
  - ***Data transformation***

Metso Automation case

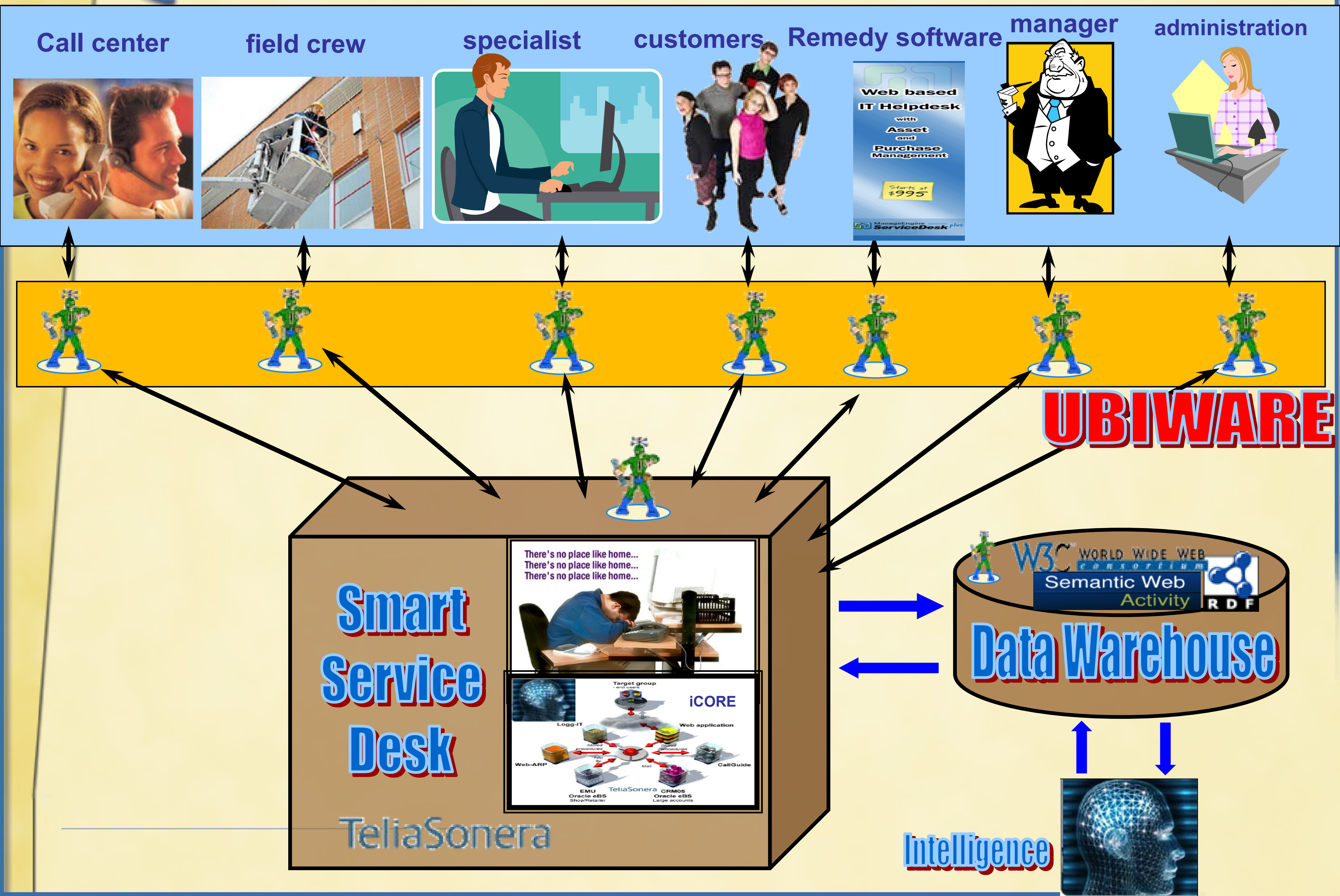
a screenshot



Inno-W Case (Semantic Visualization Browser)



Example: UBIWARE2TeliaSonera Case



UBIWARE2Nokia Case ("Smart Connectivity")

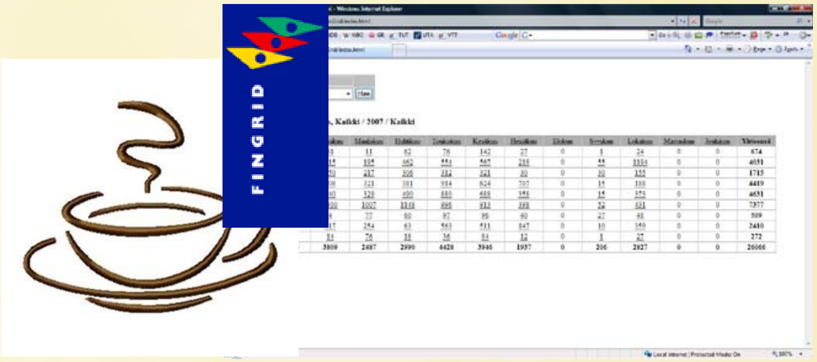
Why do you need UBIWARE ?

What the companies usually want to get from us?

They want to see technology ***applied*** *to their problems, allegorically they want a ready-to-use product, let us say, coffee,*

but…

UBIWARE allegoric view

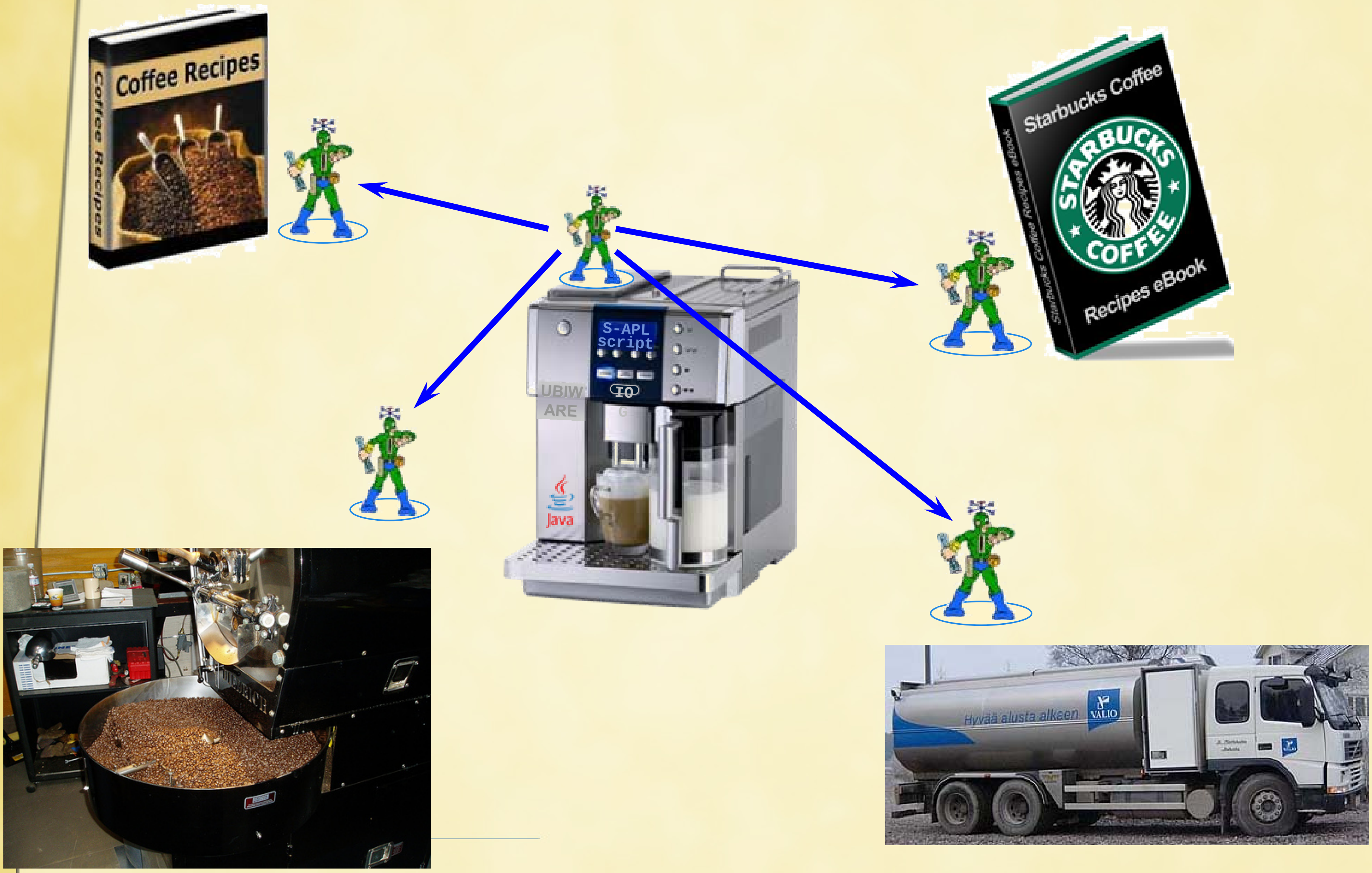


*But we are providing much more than just a coffee, we are providing a **coffee maker** !*

UBIWARE allegoric view (2)

UBIWARE is also about orchestrating external resources to achieve a goal!

Outsourced external resources can be both: information and service providers



External resources can be smart already …

Message to the UBIWARE partners

- Do not loose the opportunity, save your resources – select UBIWARE ("make coffee yourself easily and cheaper whenever you need it and not buy it every time");
- Help us to develop the basis of UBIWARE first of all, then you will be able to manage you future (even more sophisticated) cases by yourself

UBIWARE present status

- The UBIWARE project is a major step in a longer path that aims to build the so called global understanding environment. That is, a platform or middleware that supports flexible integration of all kinds of resources that have not been a priori designed to be interoperable into new processes that have not been specified when designing the platform. The basic approach in development has been that of agile development – creation of a succession of prototypes with improving functionalities on every release combined with concrete use cases with companies.

2.3. PRIME Project

"Proactive Inter-Middleware for Integrating Enterprise Systems into the Internet of Things"

PRIME (GERI) project – FP7 (Call 5)

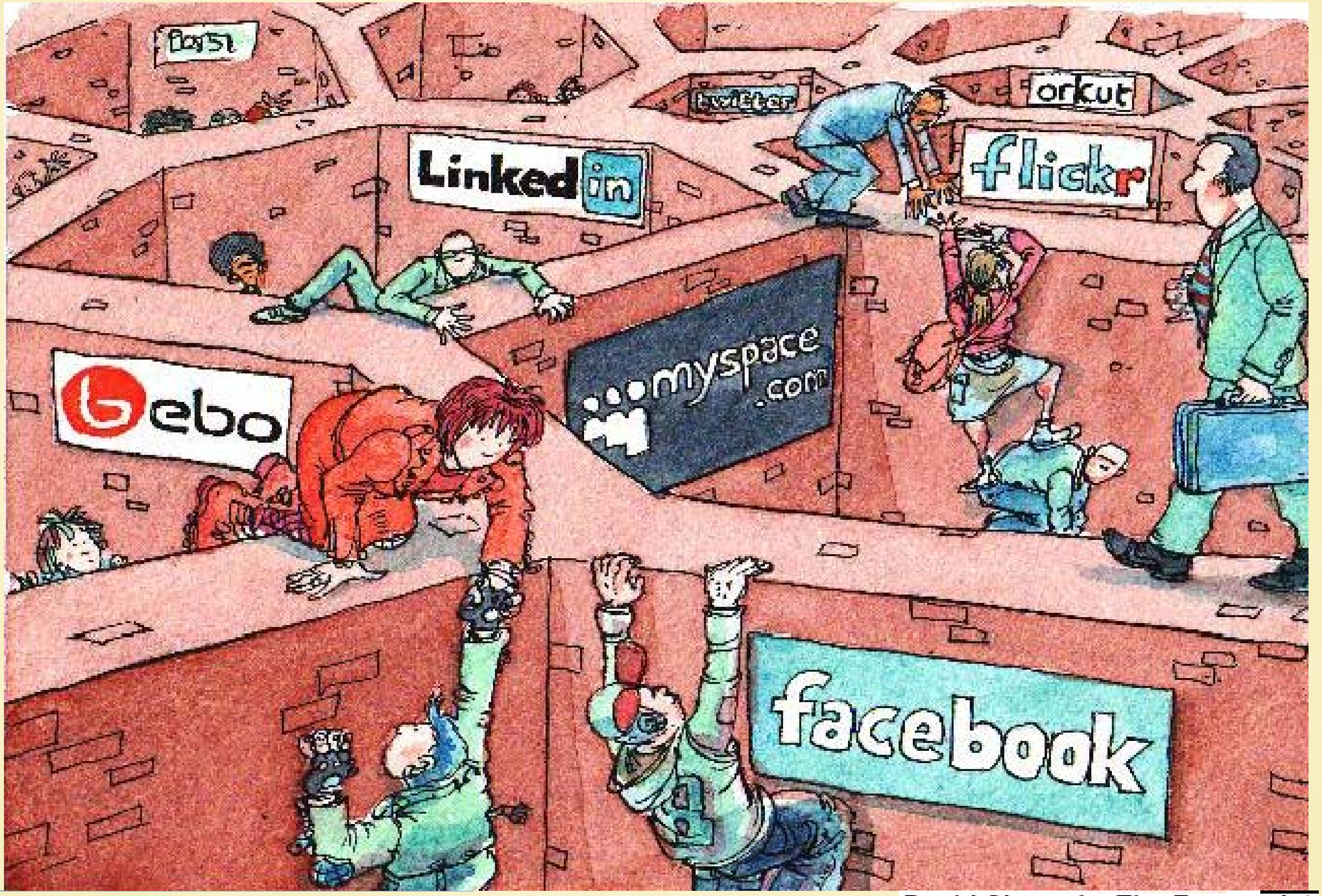
PRIME: "Proactive Inter-Middleware for Integrating Enterprise Systems into the Internet of Things"

ICT Challenge 1: *Pervasive and Trustworthy Network and Service Infrastructures*
Objective ICT-2009.1.3: *Internet of Things and Enterprise environments*

- The technological goal of the project is a PRIME *inter-middleware* which will connect heterogeneous, both industrial and non-industrial, resources belonging to different layers of the Internet of Things (we consider the three layers of physical devices, software, and humans) through the middleware platforms that are normally used for connecting relatively homogeneous resources at the respective individual layers. PRIME will be capable of handling complex interoperability scenarios where information exchange and control is needed between resources (e.g. enterprise resources) of three distinct natures: hardware devices and machinery (including tags, sensors, actuators, and other edge network equipment), software-based systems (including both enterprise information systems and Internet services and applications), and humans along with their user interfaces. With a *declarative programming* approach, the PRIME architecture will favour easy dynamic re-configuration and will provide the necessary paradigms for improving re-usability and composability.

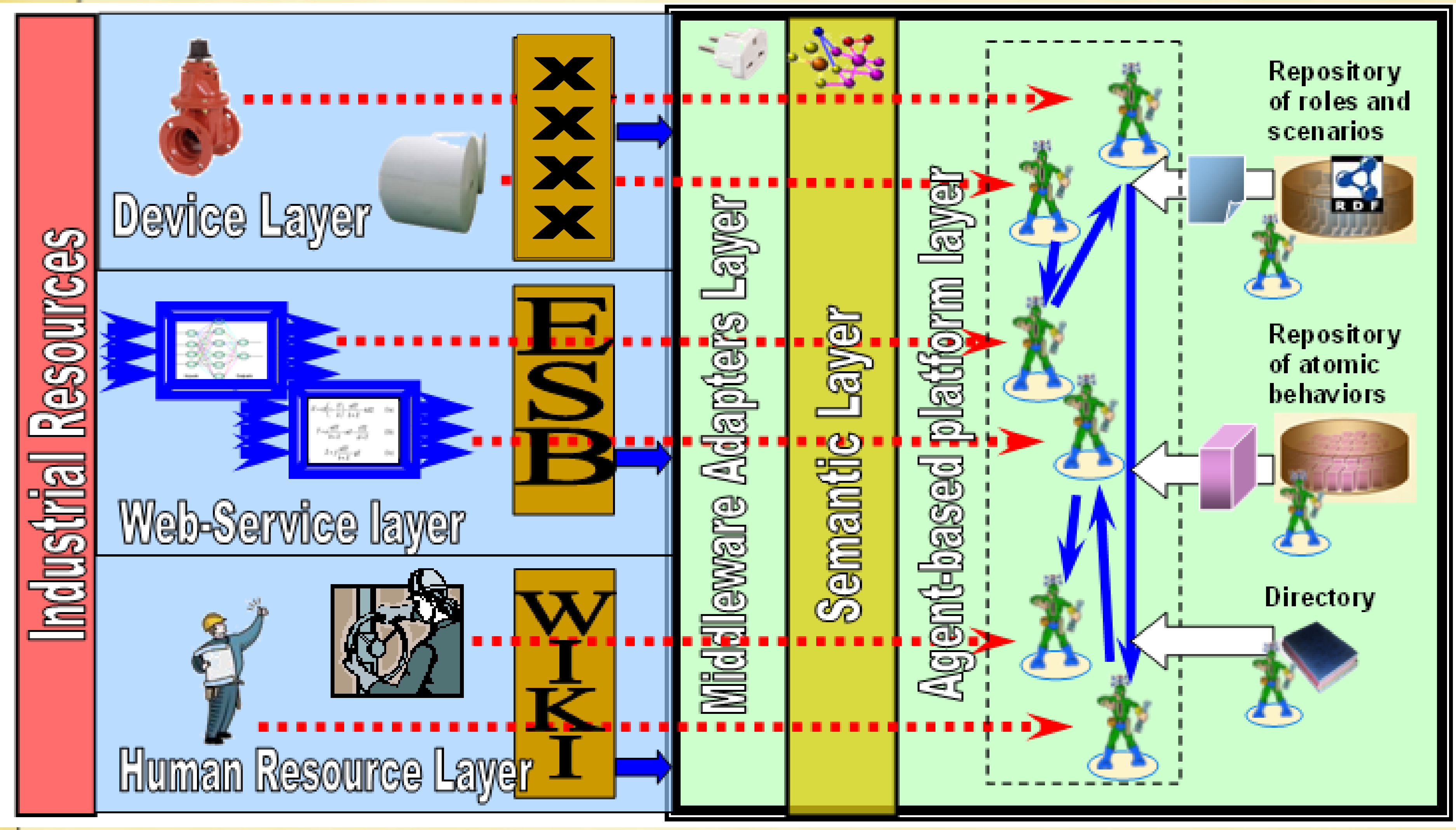
Participant organisation name	Country
University of Jyväskylä (IOG) (Coordinator)	Finland
University of Coimbra	Portugal
National University of Ireland, Galway	Ireland
VTT Technical Research Centre	Finland
SAP AG	Germany
Menta Networks Ltd	Israel
Sapienza SL	Spain
Inno-W Oy	Finland
Endress+Hauser	Germany

PRIME motivation: The "*Walled Gardens*" problem



David Simonds, The Economist

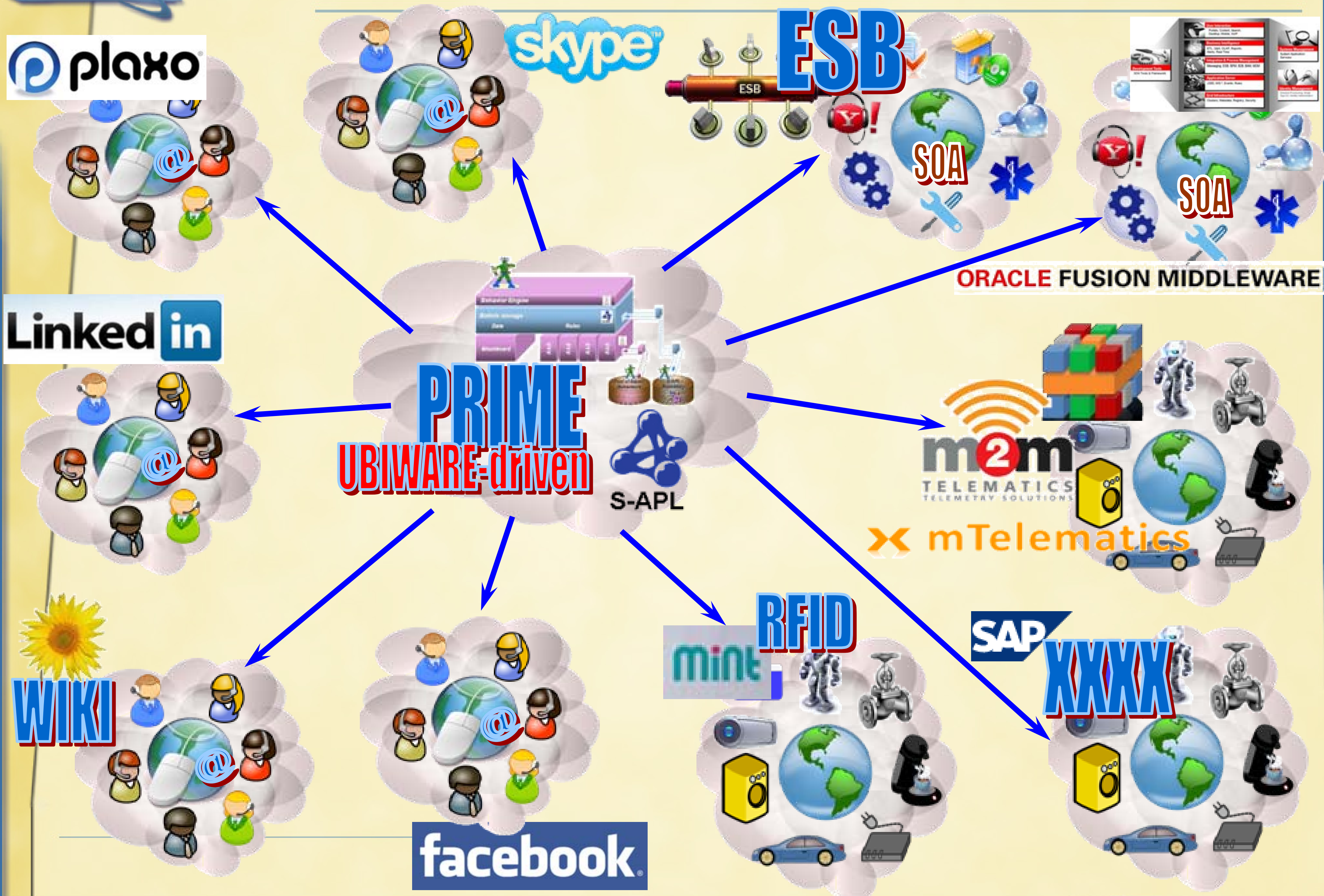
PRIME Inter-Middleware Concept



PRIME Objectives

- Development of the **generic inter-middleware architecture** to enable interoperability and integration of heterogeneous enterprise resources through the middleware platforms that exist for connecting resources of the involved types of resources.
- Development of an **ontological model for interoperability**, covering real-world entities, software systems, and humans along with their user interfaces, from both the technical and the business perspectives.
- Development of a **multi-agent architecture**, in which the interaction scenarios of heterogeneous resources are defined and configured declaratively (semantically) rather than programmatically.
  - Adoption and further elaboration and extension of the ***Semantic Agent Programming Language (S-APL)*** for representation of agent's role behaviour models (behavioural semantics) and the integration scenarios.
  - Enabling flexible yet predictable operation through incorporating commitments imposed by the ***organizational roles and policies***.
  - Design of the core semantic mechanisms for ***inter-agent coordination***.
- Development of a set of solutions enabling **homogeneous interfacing with resources of different nature**.
  - ***Linking to Real-world Entities*** (physical objects with embedded electronics or RFID).
  - ***Linking to the Web of Services***.
  - ***Linking to Human Resources***.
- Facilitating development of new advanced solutions for monitoring and management of **energy-efficient manufacturing plants** and for **remote device management**. Creating a set of tools (methodological, technological and organizational) for the deployment of PRIME in the industry.

PRIME as 2-nd order middleware (UBIWARE-based)

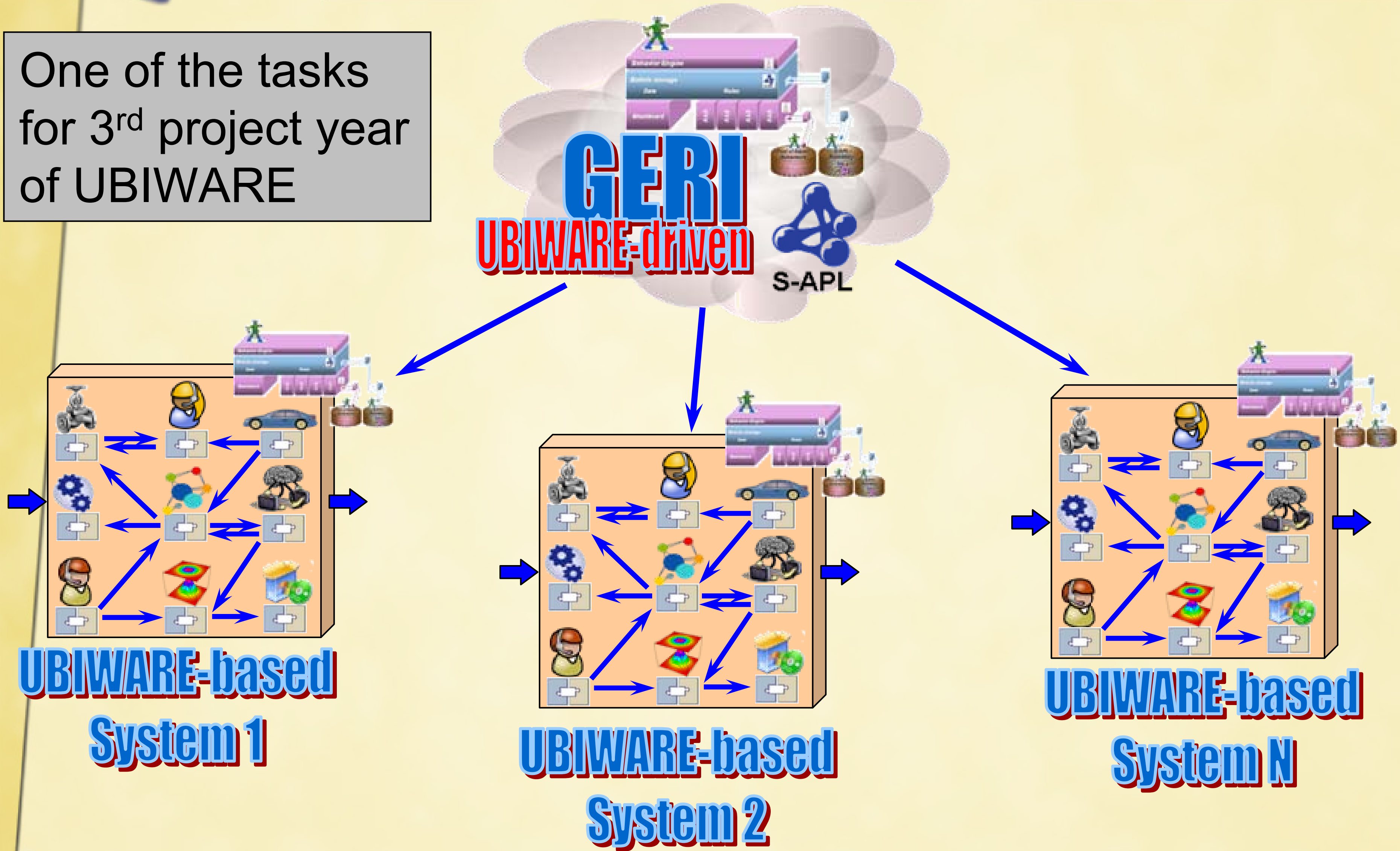


Innovative concepts of PRIME Vision

- The “**Inter-middleware**" approach opens a new challenging concept of **MaaS (Middleware-as-a-Service)** in addition to **SaaS (Software-as-a-Service)** and **DaaS (Device-as-a-Service)**. Through MaaS every resource will be able to automatically get service available in certain ecosystem and even integrate heterogeneous services from different ecosystems. Also a human is considered in various possible roles including **HaaS (Human-as-a-Service)**. The **Knowledge-as-a-service (KaaS)** driven by proactive ontologies is also a new concept. Finally we invented **IaaS (Intelligence-as-a-Service)**, meaning data-mining/knowledge discovery/OLAP/ algorithms (which produce new knowledge to the system), as services of the system. Summarising, the “inter-middleware” vision allows enhancing the Internet of Things functionality with existing and future capabilities provided by the Web of Services, Web of Humans (Web.2.0), Web of Knowledge (Web 3.0) and Web of Intelligence (Web 4.0).

GERI* as 2-nd order UBIWARE !

One of the tasks for 3rd project year of UBIWARE



* GERI - Global Enterprise Resource Integration

2.4. Other Projects

InTIME project – FP7 (Call 5)

InTIME: "Intelligent Information Management Environment Driven by Context"

ICT Challenge 1: *Pervasive and Trustworthy Network and Service Infrastructures*
Objective ICT-2009.4.3: *Intelligent Information Management*

InTIME will develop theory, models and software tools that index and detect knowledge within industrial repositories (e.g. corporative portals) through its context: a creation-context extractor, and a complementary context-based search engine. InTIME defines context as the information required for making stored knowledge reusable. In certain settings knowledge requires substantial context information for instance in cases where products with long life cycles carry large knowledge repositories with them, and the context, in which this knowledge has been created, is difficult to understand at the end of the product life cycle. InTIME offers support for knowledge reuse by helping to find relevant knowledge through its context representations.

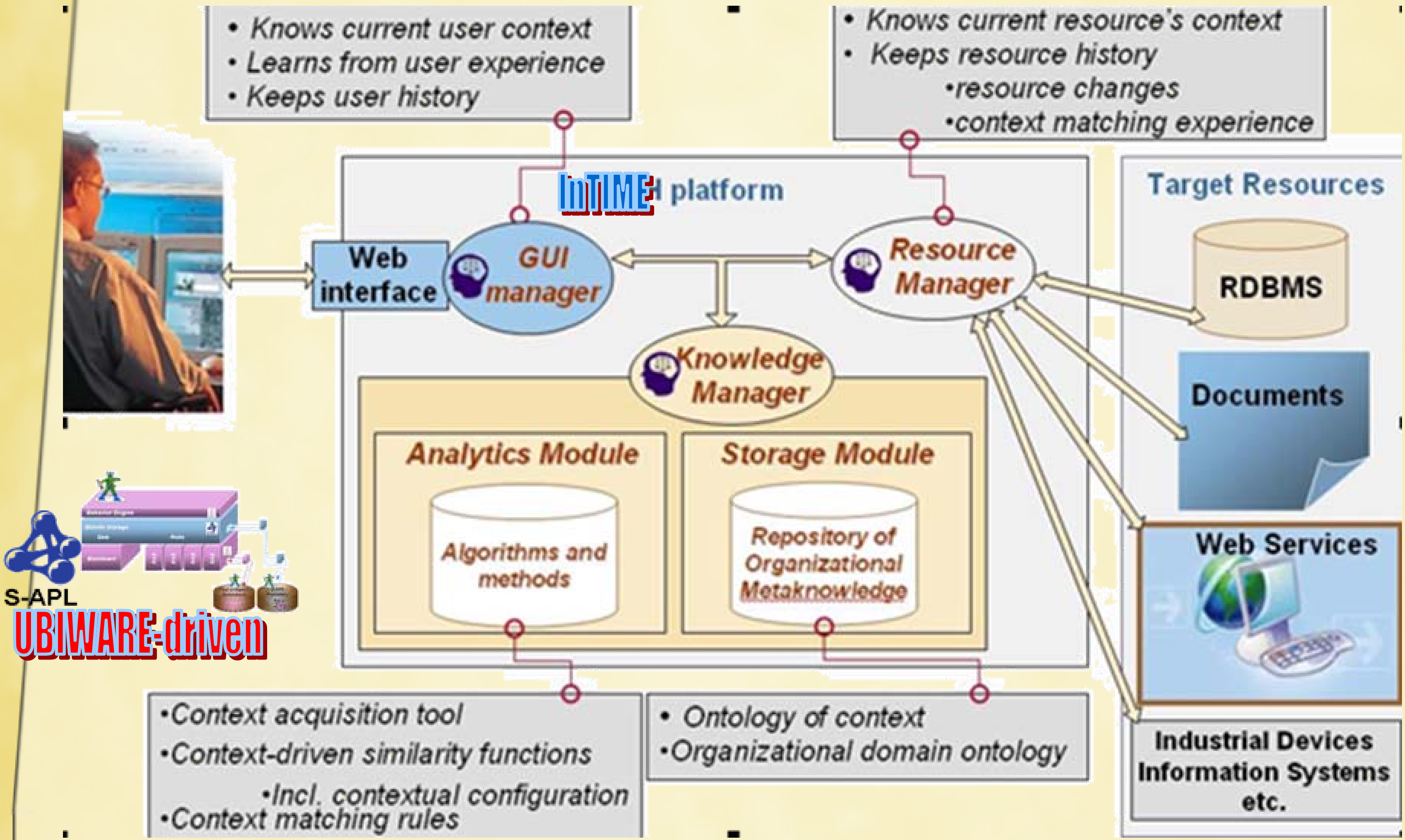
Participant	Country
Interdisciplinary Center for Technology Analysis and Forecasting at Tel-Aviv University (Coord.)	Israel
University of Twente	Netherlands
RWTH Aachen	Germany
University of Jyväskylä (IOG) (WP leader)	Finland
Ort Braude College	Israel
Ecole Centrale de Nantes	France
Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V.	Germany
Inno-W Oy	Finland
Festo	Germany

InTIME: Role of Industrial Ontologies Group

As a leader of WP6 “Software Applications Development”, IOG has to address the following tasks: Task 6.1 Resource Agent and Context Detector (that performs the analysis of the unstructured and/or changing data and builds contextual annotations); Task 6.2 Ontology Agent (persistent storage of metadata, i.e. domain ontology, context handling rules, organization-specific knowledge, policies, etc.); Task 6.3 Distributed search engine (a main feature of the User Agent, that employs planning of distributed queries, inter-agent communication protocols and integration of query results); Task 6.4 User Agent GUI (a human user interface to the InTIME platform mediated by User Agent); Task 6.5 Integrated InTIME platform (assembling and deployment of the platform, solving the scalability issues, preparing the installation).

Another IOG task is to test InTIME toolson the Inno-W case (Forest Cluster Portal): i.e. to adapt the Semantic Distance Measuring Function based on idea/proposal description attributes as an engine of Context-Driven Similarity Search System; to define the context and elaborate a model that configures the distance measuring function based on contextual information (there are several approaches that can be used to define a model of influence of contextual information on search function: supervised and unsupervised machine learning algorithms, theory based constructions, etc.) and finally - to develop infrastructure for visual representation of the results. Contextual information play role of a filter and help to configure search function to present more relevant (in current context) results.

InTIME: Software Platform Architecture



UbiCloud activity of IOG within *TIVIT* ("Cloud Software") ICT-SHOK

UbiCloud: "Heterogeneous and Semantic Cloud Services"

2010 - ...

- Within WP1: "***Service Engineering in the cloud*** ":
  - ❑ Engineering, integration and composition of ***heterogeneous cloud services*** (Software as a Service, Human as a Service, Device as a Service, Intelligence as a Service);
  - ❑ Enabling infrastructure for ***semantic cloud services***.

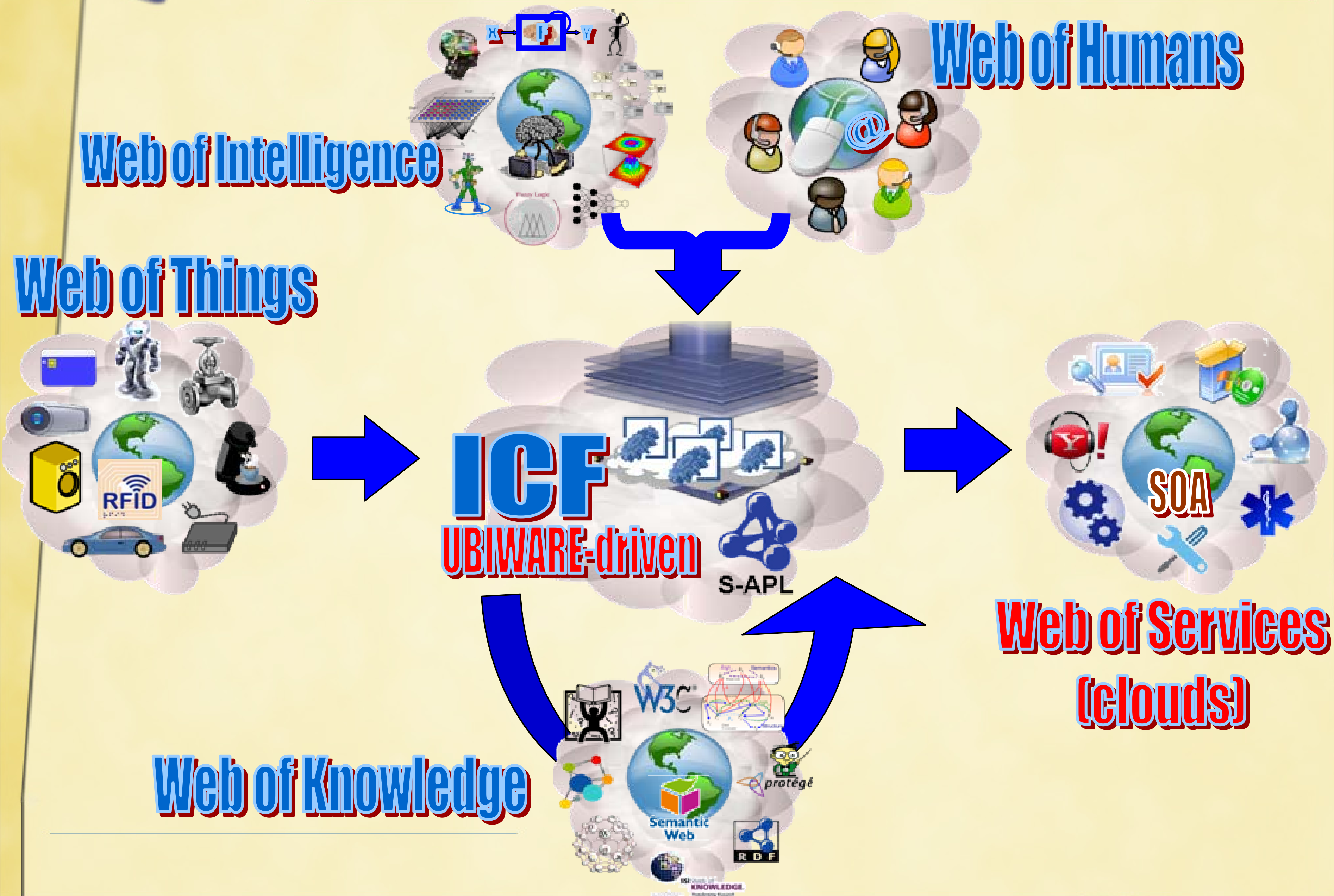
UBIWARE-driven

iCloud: "Intelligence Cloud Factory for Intelligent Product Manufacturers"

Applied to FIMECC ("Intelligent Solutions").

The main iCloud project objective is to design a Web-based portal with high-level intelligent services for FIMECC industrial partners according to cloud computing architecture. Such portal will operate as an "Intelligence Cloud Factory" (ICF), i.e. ICF will be able to automatically access online and offline data through embedded systems and sensor networks from various products of an industrial manufacturer, then based on this data automatically build models (neural networks, bayesian predictors, etc.) for diagnostics, prediction, etc., then automatically wrap these models to the form of Web-services and finally automatically create infrastructure of a "cloud" from these services for further intelligent support of the target products.

Intelligence Cloud Factory concept

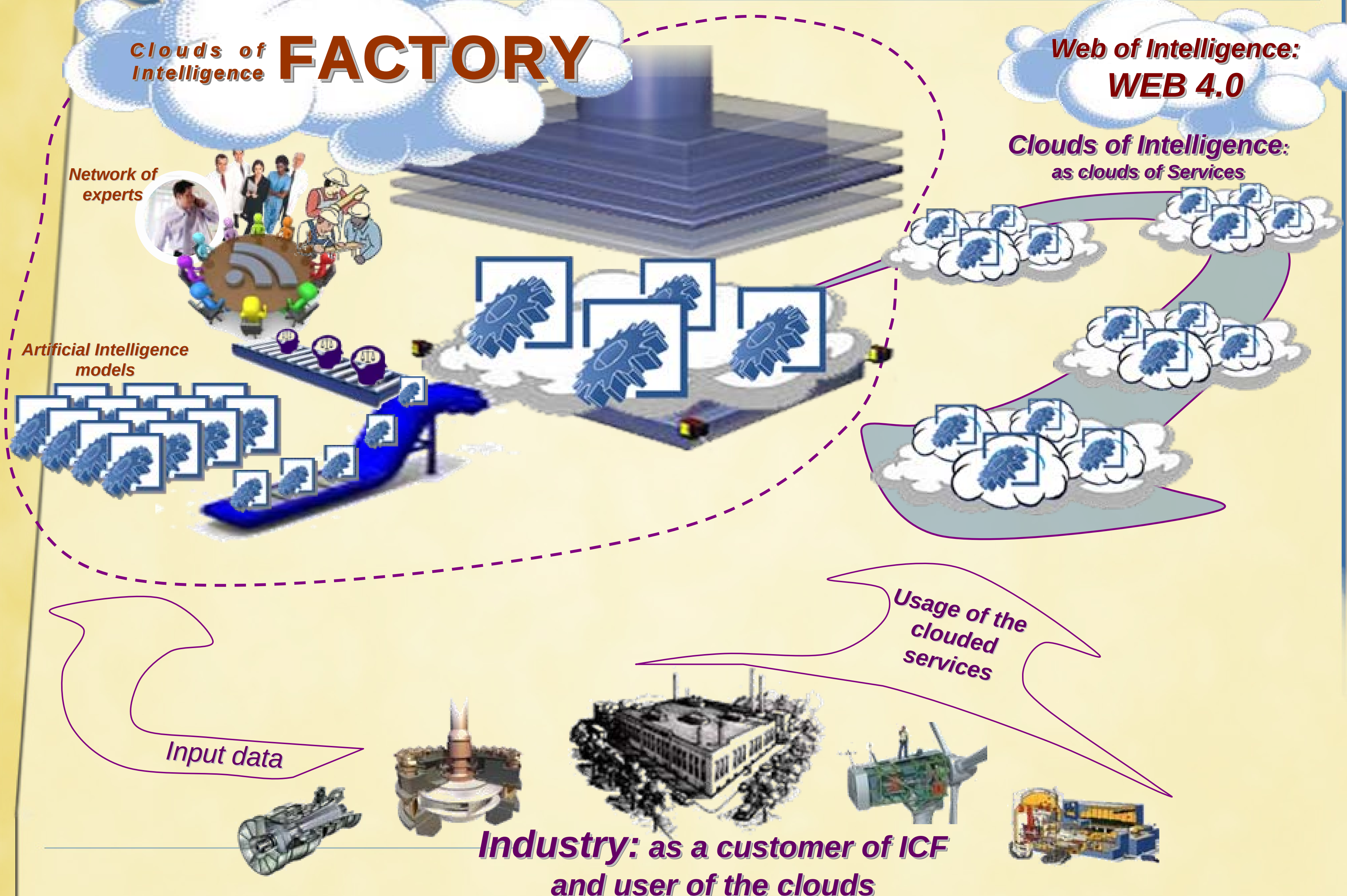


"Intelligence Cloud Factory" as a "Meta-Cloud"

The target is an "Intelligence Cloud Factory" (ICF), which will utilize "Web of intelligence" (machine learning, data mining, etc services) to automatically generate models from data, wrap them as Web services and create "clouds" of such services for various product-centric applications.

ICF is supposed to be a cloud (of a higher order) itself !

Intelligence Cloud Factory (ICF)



hCloud: Cloud Factory for eHealth

Ongoing negotiations of **IOG** with

NATIONAL INSTITUTE
FOR HEALTH AND WELFARE

STAKES Unit for eHealth and eWelfare

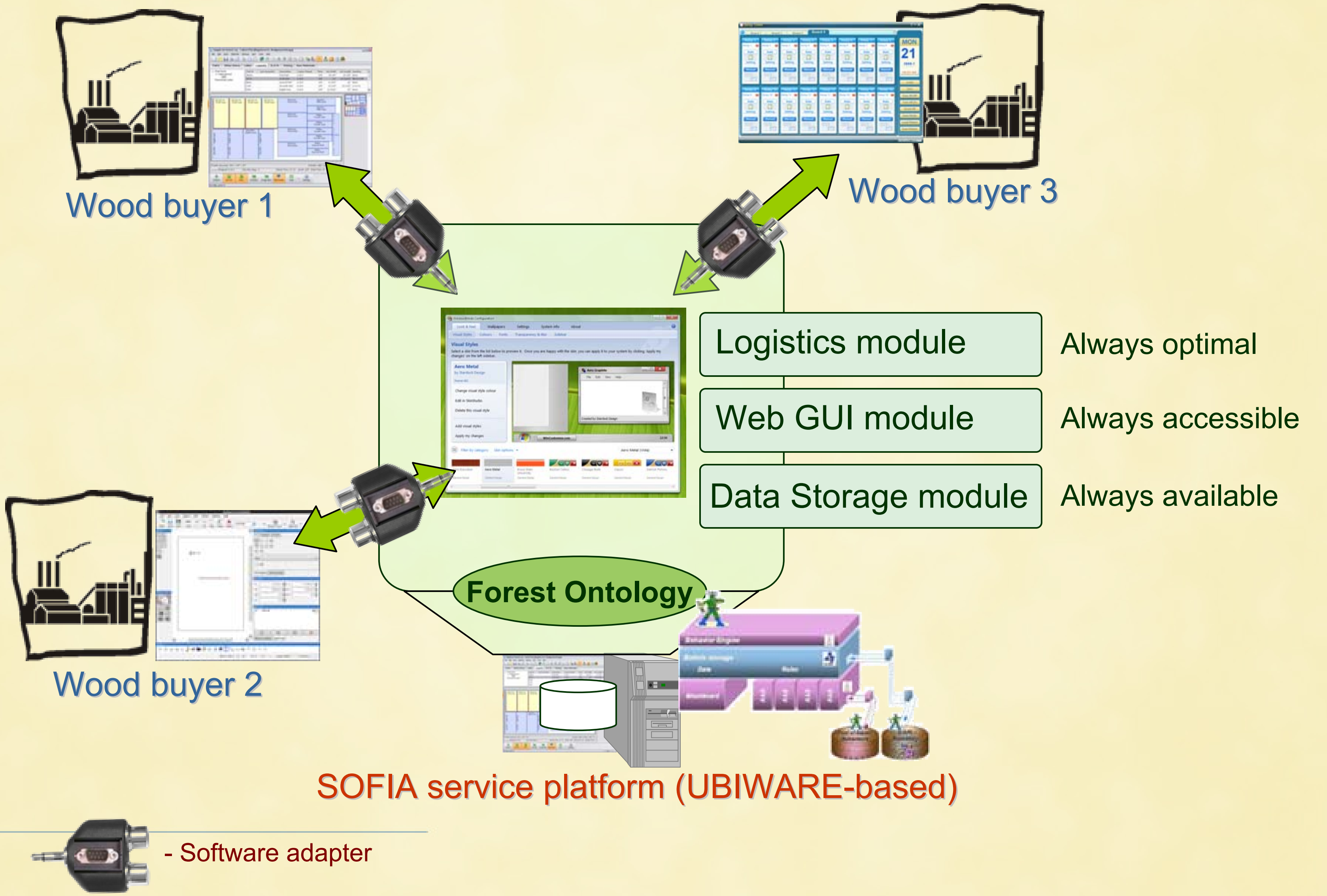
http://sty.stakes.fi/FI/index.htm

Topic for cooperation:
"Agent-driven policies for configurable security and privacy settings in eHealth systems"

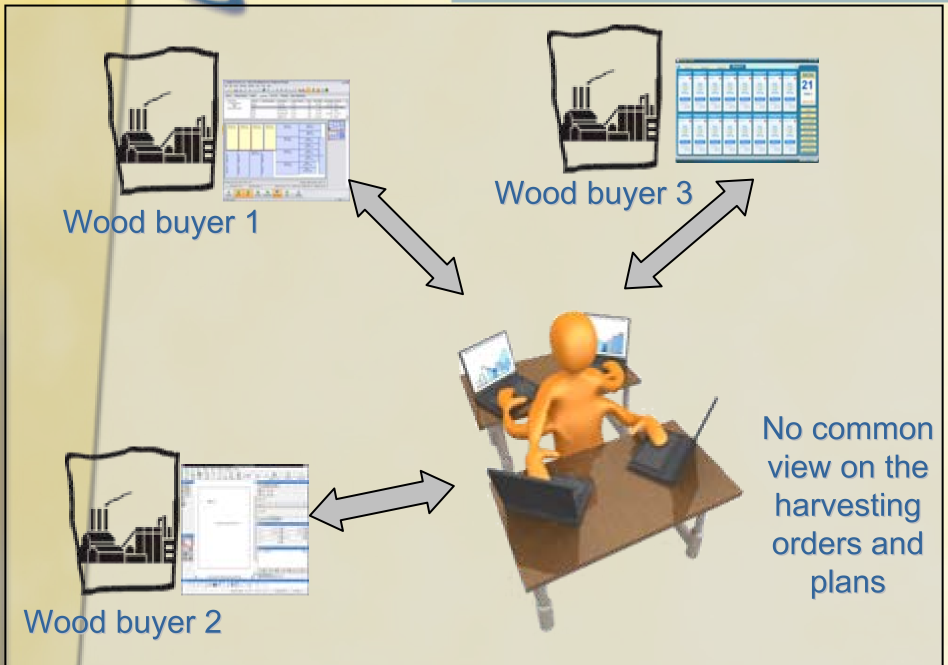
UBIWARE-driven

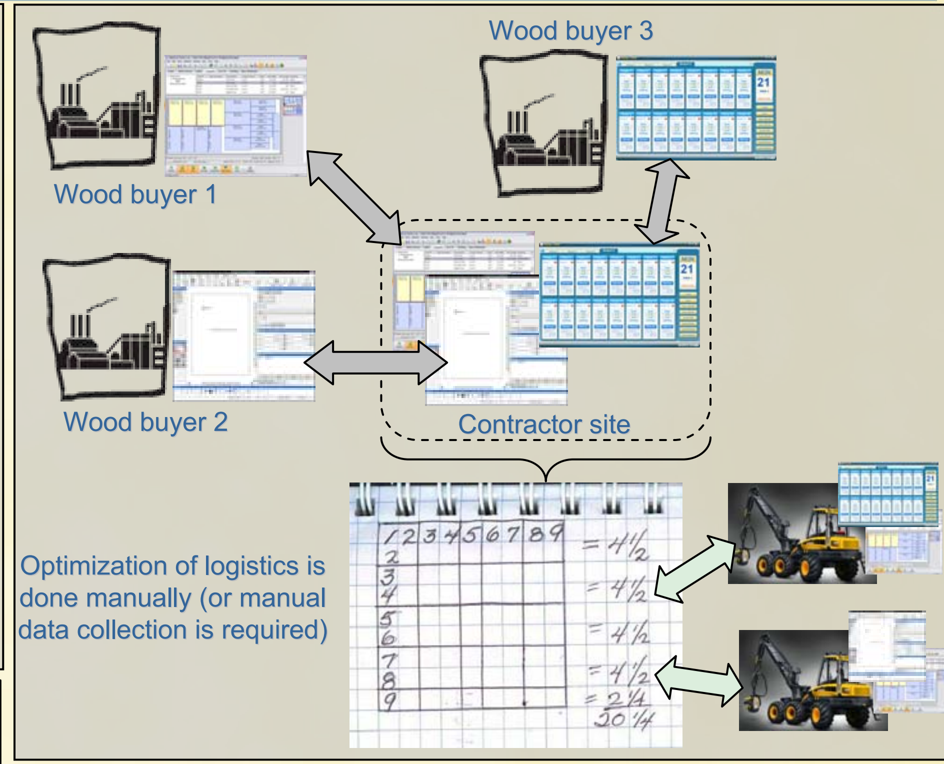
SOFIA - project proposal to Forest Cluster

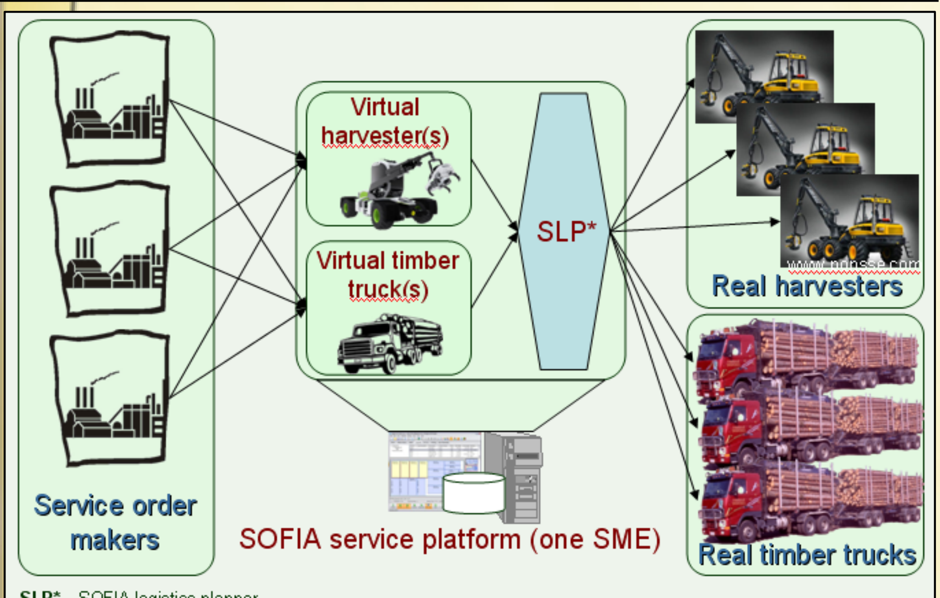
SOFIA – Seamless Operation of Forest Industry Applications



SOFIA motivation and solution







Virtual Service Provider:

From the consumer point of view, logging and transportation services look the same, however, the real equipment of the contractor is hidden. A wood buyer makes orders seamlessly, but the assembly chosen for the execution is virtual. The service provider then plans the operations and assigns tasks to the real units.

Our Main Partners in International Cooperation

- SAP, Germany (Internet of things, product-centric applications);
- Massachusetts Institute of Technology, CS and AI laboratory, USA (semantic language for MAS, policy-based reasoning);
- Massachusetts Institute of Technology, Data Center, USA (semantics in RFID-based systems);
- University of California, Berkeley, USA (declarative networking, user modeling);
- University of Southern California, USA (multi-agent systems, distributed constraints optimization, robots coordination in P2P environments);
- Lulea Technical University, Sweden (smart services, embedded systems, telecommunications);
- VU Amsterdam, Netherlands (agents and Semantic Web);
- University of Athens, Greece (Service-Oriented Architectures);
- DERI, National University of Ireland, Galway (sensor networks middleware, Internet of things);
- University of Coimbra, Portugal (Semantic Web processes and services);
- Ostrava Technical University, Czech Republic (logic in MAS);
- ITIN, Cergy-Pontoise, Paris, France (educational system reforms);
- Kharkov National University of Radioelectronics, Ukraine (machine learning, semantic portals, quality assurance in education, university management).

UBIWARE Benchmarking

- The **ADiWa ("Alliance Digital flow of goods")** (duration: **01.01.2009 - 31.12.2011**, funding: **17.7 M**, program: ICT 2020/Research and Innovation, URL: http://www.adiwa.net/) coordinated by SAP (other partners: German Research Center for Artificial Intelligence (DFKI), Fraunhofer Society, IDS Scheer AG, Software AG, Technical University of Darmstadt and the Institute for Applied computer science from the Technical University of Dresden) aims to provide technologies for enterprise applications to explore the complex and dynamic business processes over the "Internet of Things" plan, optimize, control, and execute it.

!

- In the **SemProM: "Semantic Product Memory"** project (duration: **01.02.2008 - 31.01.2011**, funding: **16.46 M**, program: ICT 2020, same partners, URL: http://www.semprom.org/) the ambition is to provide semantic web communication between objects. Products suppose to keep a diary (log) due to smart labels that give products a memory. Project is based on semantic technologies, (machine-to-machine communication, sensor networks, smart environments, RFID technology, etc. By the use of integrated sensors, relations in the production process become transparent and supply chains as well as environmental influences retraceable. The producer gets supported and the consumer better informed about the product.

UBIWARE Benchmarking

- The **SOCRADES** project (2006-2009) is a European research and advanced development project. Its primary objective is to develop a design, execution and management platform for next-generation industrial automation systems, exploiting the Service Oriented Architecture paradigm both at the device and at the application level. SOCRADES is a part of the Information Society Technologies (IST) initiative of the European Union's 6th Framework Programme. The focus is: communication between and integration of heterogeneous embedded systems and devices, with particular emphasis on platform independence, real-time requirements, robustness and security. A key goal of SOCRADES is to specify a service-oriented framework for device-level infrastructures, where system intelligence is achieved by intelligent physical agents embedded in smart devices. The consortium is made up of 15 partners from 6 European countries. Budget **13.75 M**.
- **SOFIA:** “Smart Objects For Intelligent Applications” (an ARTEMIS JTI project, 2009-2011). Goal: to make information from the physical world available to various applications in the digital world, and in so to enable information-level interoperability between multi-vendor devices and enable development of software applications involving those devices as components. Budget **15 M**.

Conclusion (EaaS4E – beyond Cloud Computing)

- While the academic and business communities are exited with the new **Cloud Computing** and SOA slogan: "**EaaS: Everything-as-a-Service !**" spending to it huge resources and still having no clue about what "*everything*" actually means, our group since 2003 with extremely modest resources is actively working on **GUN Computing and Web 5.0**, which much more challenging slogan (based on "ψ-projection" technological vision) is: "**EaaS4E: EaaS for Everything**", meaning "**<u>Really</u> Everything-as-a-<u>Proactive, Semantic and Intelligent</u> Web Service <u>Provider and Consumer!</u>**".

October 2009
Vagan Terziyan
Head of Industrial Ontologies Group

Obtain More Information about UBIWARE from:

Head of UBIWARE Industrial Consortium (Steering Committee Head) **Dr. Jouni Pyötsiä**, Metso Automation Oy.

Jouni.Pyotsia@metso.com , Tel.: 040-548-3544

UBIWARE Contact Person **Prof. Timo Tiihonen**, Vice-Rector, University of Jyväskylä

timo.tiihonen@jyu.fi , Tel.: 014-260-2741

UBIWARE Project Leader **Prof. Vagan Terziyan**, Agora Center, University of Jyväskylä

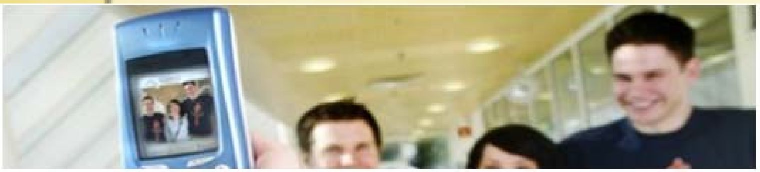
vagan.terziyan@jyu.fi , Tel.: 014-260-4618

Project URL: http://www.cs.jyu.fi/ai/OntoGroup/UBIWARE_details.htm

Welcome to our **MOTEBU** International Master Program

https://www.jyu.fi/it/en/motebu/

University of Jyvaskyla, Finland



MASTER'S DEGREE PROGRAMME IN MOBILE TECHNOLOGY AND BUSINESS

The application period for studies commencing in the autumn 2010 has now closed. The next application round will open in mid-November 2010.

Award: Master of Science in Economics and Business Administration or in Natural Sciences
Credits: 120 ECTS
Full-time duration: 2 years
Language of instruction: English
Major subject: Information Systems or Information Technology
Annual student intake: 20-25 students

Ms Niina Ormshaw
International Coordinator
Faculty of Information Technology
P.O.Box 35 (Agora)
FI-40014 University of Jyväskylä
Finland

Tel: +358 14 260 4602 / +358 50 4432 360 (mobile)
Fax: +358 14 260 2209
E-mail: international-info@it.jyu.fi

Main Focus:

"**Intelligent Web Applications, Systems and Services**: *Technology and Business*"

Students will graduate with a Master of Science either in *Economics and Business Administration* or in *Natural Sciences* depending on their study profile.