AI Meets Web 2.0:

Building the Web of Tomorrow, Today

Jay M. Tenenbaum
with illustrations by Kevin Hughes

Abstract

Imagine an Internet-scale Knowledge System where people and intelligent agents can collaborate on solving complex problems in business, engineering, science, medicine, and other endeavors. Its resources include semantically tagged Web sites, wikis, and blogs, as well as social networks, vertical search engines and a vast array of Web services from business processes to AI planners and domain models. Research prototypes of decentralized knowledge systems have been demonstrated for years, but now, thanks to the Web and Moore's Law, they appear ready for prime time. Architectural concepts for incrementally growing an Internet-scale knowledge system are introduced, with descriptions of early commercial deployments in manufacturing and healthcare.

CommerceNet Labs Technical Report 05-07
December 2005

1. Introduction

I want to share a vision of how to build, or more precisely, grow Internet-scale knowledge systems. Such systems enable large numbers of human and computer agents to collaborate on solving complex problems in engineering, science, and business, or simply managing the complexities of life (say planning a trip or an event). It's a vision that's been evolving over 20 years since my days as an AI researcher, and more recently as an Internet entrepreneur. Thanks to the explosive growth of the Web, it's a vision whose time has come. I also have a larger goal: to bridge the AI and Web communities, which have so much to give to and learn from each other.

25 years ago, at the birth of AAAI, Allan Newell articulated a set of criteria that a system had to exhibit to be considered intelligent (See Table 1). Newell was very explicit that an intelligent system had to exhibit all of these criteria. This requirement reflected the then prevailing view that intelligent systems were monolithic, and developed centrally by an individual or small group.

- Exhibit adaptive goal-oriented behavior
- Learn from experience
- Use vast amounts of knowledge
- Exhibit self-awareness
- Interact with humans using language and speech
- Tolerate error and ambiguity in communication
- Respond in real time

Table 1: Newell's Criteria for Intelligent Systems

The Web has shown us a different path to intelligence – millions of simple *knowledge services*, developed collaboratively in a decentralized way by many individuals and groups, all building on each other, and demonstrating a collective form of intelligence.

It's time to reconcile these two views and create a new breed of hybrid knowledge systems that combine the best elements of AI and the Web and of humans and machines. Such hybrid systems can solve heretofore intractable real world problems, and potentially demonstrate heretofore unattainable levels of machine intelligence.

1.1 Scenarios

To give you a hint of where I'm headed, let's revisit some classic AI problem solving tasks, which are as timely now as they were in Newell's day (Table 2). I've worked on many of them, and I'm sure you have too. How would you, as a person, approach any of these tasks today? Undoubtedly, you'd start by searching the Web, where you'd find thousands of relevant information sources and services.

Travel	What's the best way to get to Pittsburgh?
Meetings	Who should I have dinner with tonight? Where?
Supply chain	How can I get 100 PCs delivered by tomorrow?
Medicine	What drugs might be effective against this new bug?
Ebusiness	Where can I get the best deal on a car and a loan?

Table 2: Classic AI Problems

If I were building an AI system to solve these problems, I too would start with the Web, gradually adding a little AI here and there to automate time consuming functions and integrate processes. Take travel, for example. I might start by creating a simple agent to check many travel sites for available flights and fares. A second agent might use these results to select an optimal itinerary based on my personal frequent flyer accounts and calendar schedule.

Now imagine giving millions of people this ability to create agents that incrementally automate tasks, as well as the ability to publish those agents as services that others can use and build upon. Before long, interesting forms of collective intelligence will begin to emerge.

Approaching AI applications in this way has manifest benefits – the ability to leverage billions of dollars of Web infrastructure, millions of Web services, and millions of knowledge engineers all building on each other's work. Moreover, such systems are useful from day one, because they start with the Web. Automation proceeds incrementally – one person, and one task at a time. Each step provides immediate, incremental benefit. And because people remain in the loop, such systems can fail gracefully, by handing off hard or unusual cases for manual processing.

1.2 Plan

We'll start by reviewing the evolution of the vision from its roots in AI research through the early days of the Web and more recent developments such as Web Services and Web 2.0 (referring to the spate of recent Web innovations that includes blogs, wikis, social networks and the like). These recent developments make the Web more programmable and more participatory, important attributes for Internet-scale knowledge systems. Because many of you may not be familiar Web 2.0, I'm going to take you on a brief tour. We'll look both at the potential role of Web 2.0 technologies and methodologies in building AI systems, as well as how AI can contribute to enhancing Web 2.0. We'll then demonstrate through case studies and scenarios, a path toward Internet-scale knowledge systems that synthesizes Web 2.0 and classic AI approaches. Finally, we'll conclude with a call for action for realizing the vision on an Internet-wide scale.

2. Evolution

20 years ago, I was running a research lab for Schlumberger. At the time, Schlumberger was trying to break into the CAD/CAM business through its acquisition of Applicon.

I was fascinated by the possibility of using the Internet to support large-scale engineering projects, such as the design of a new airplane, involving thousands of people at hundreds of companies (Figure 1).

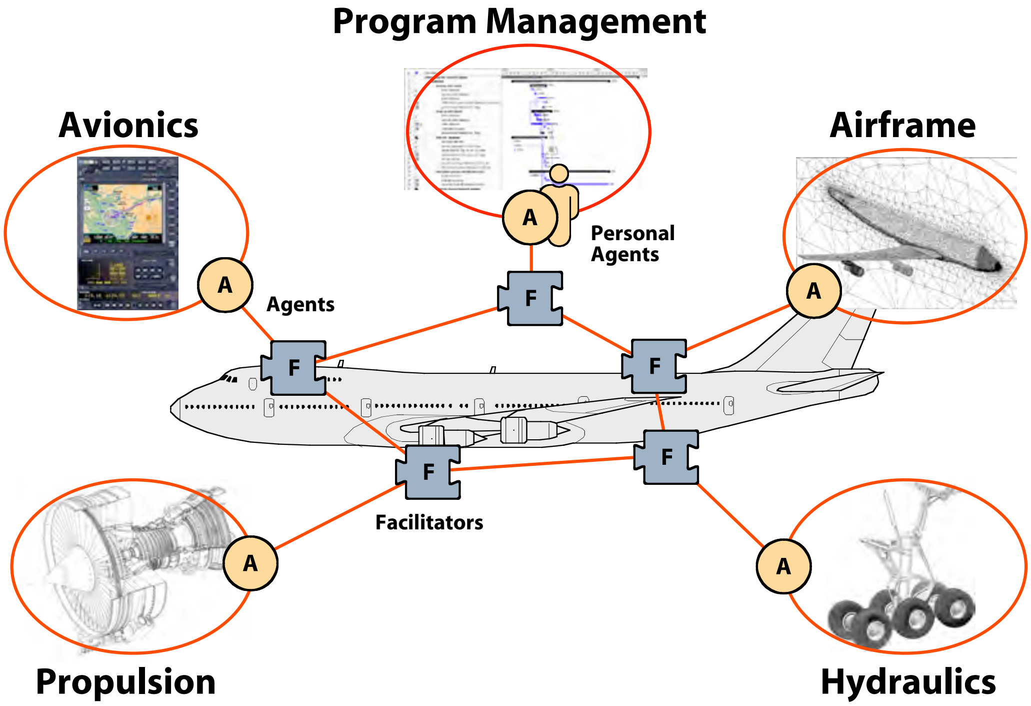


Figure 1: Concurrent Engineering and the PACT Architecture

Inspired by distributed blackboard systems such as Hearsay, we modeled the process as human and computer agents collaborating through a shared knowledge base, representing a model of the artifact. When an agent modified the design, affected agents were notified so they could critique the change, or respond with changes of their own. For example, a hydraulics engineer installing some fluid lines might alert an airframe agent to check whether they interfered with the movement of any control surface. Although a centralized shared knowledge base is depicted, the model can be distributed in practice to facilitate scaling. Each agent maintains aspects of the model most relevant to it in local CAD systems, and provides that information to other agents that need it.

2.1 The Palo Alto Collaborative Testbed (PACT)

We built several research prototypes of such agent-based collaborative engineering systems during the late 1980s. The most impressive was PACT (Palo Alto Collaborative Testbed), which was documented in the January 1992 issue of *IEEE Computer*. PACT used the Net to link four independently developed knowledge-based design systems (covering requirements management, kinematics, dynamics, and electronics) at three different organizations – EIT, Lockheed and Stanford. The systems then collaborated on the design of a small robotic manipulator

The PACT architecture consisted of three components, as depicted in Figure 1:

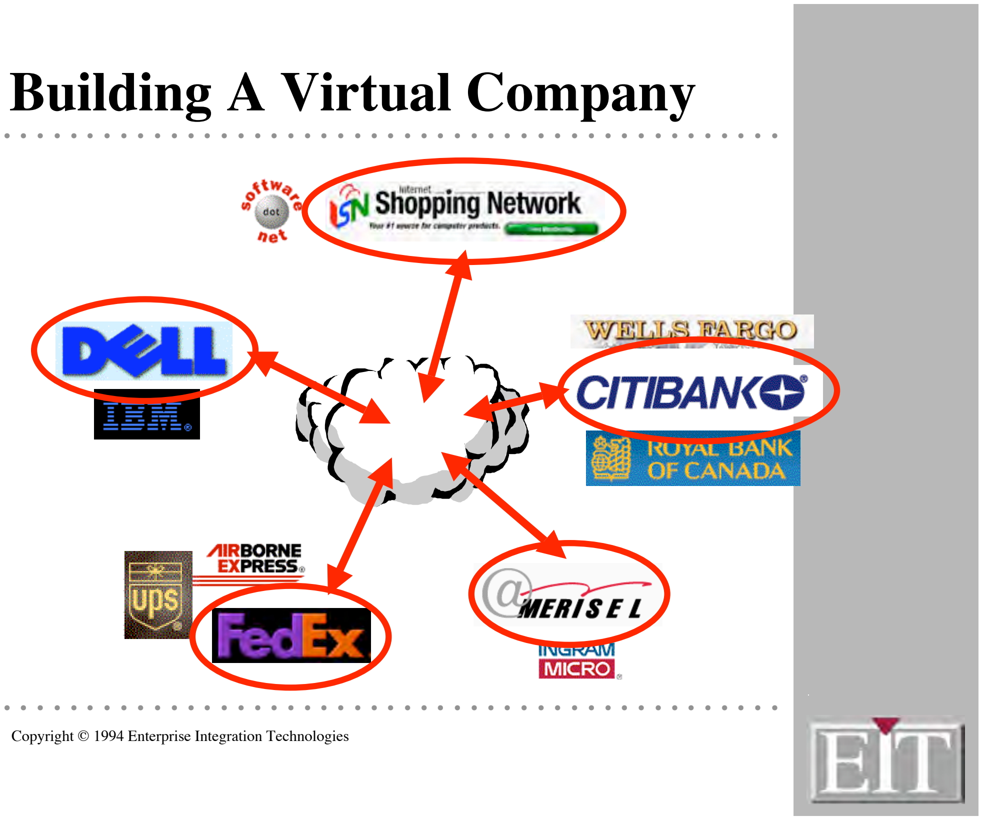
- Agents encapsulated existing design systems, transforming them into Knowledge Services.
- Facilitators handled inter-agent communication, routing messages on the basis of content to potentially interested agents. Facilitators also filtered, prioritized and semantically mapped messages so that agents could focus on design-related tasks.
- Personal agents enabled people to participate in this agent ecology. Users could respond directly to messages from other agents using a GUI; they could also specify pattern-action rules that generated automated responses to routine messages.

In retrospect, PACT was a seminal paper, articulating three key principles for building large distributed knowledge systems. Paraphrasing from the conclusion, the first principle states that "Instead of literally integrating code, modules should be encapsulated by agents, and then invoked when needed as network services". This is the core idea behind today's service-oriented architectures. The second principle holds that agents should communicate on a knowledge level, using the semantics of the problem domain, rather than the local standards employed by encapsulated systems. Finally, messages should be routed intelligently at run time based on their content, since it's often impossible to know a priori which agents might be affected by a design action.

2.2 Electronic Commerce

By 1990, I had shifted my focus to Internet commerce. The company I founded, Enterprise Integration Technologies (EIT), developed some of the earliest shopping sites including Internet Shopping Network, launched in April 2004. ISN pioneered many innovations including online catalogs, shopping carts, and cookies, and served as a role model for Amazon and other shopping sites. PACT's notion of integrating services also figured prominently. As depicted in this vintage slide (Figure 2), we envisioned ISN as a virtual company, outsourcing fulfillment services to Merisel, shipping services to

FedEx, and payment services to Citibank. We even envisioned integrating a build-to-order supplier like Dell and creating a virtual computer company.



Building A Virtual Company

Copyright © 1994 Enterprise Integration Technologies

Figure 2: The Virtual Company circa 1994

From an AI perspective, we thought of these services as special purpose problem solvers. Fed Ex, for example, with over 100,000 employees and thousands of trucks and planes was a special purpose problem solver for logistics, invoked through a simple message-based API – “Get this package to New York City by tomorrow morning”. It struck us as foolish to write a logistics planner, rather than leveraging such a powerful off-the-shelf resource.

3. Web Services

Fast forward to today’s world of Web Services. Companies in competitive industries like electronics are beginning to publish business services (e.g., place an order, make a payment, track a shipment) that others can discover in registries and integrate into their own processes, then publish new services that other's can build on. With critical mass, business services are expected to grow explosively like the Web itself.

Prior to Web Services, companies that wanted to integrate their business processes had to resort to expensive custom integration or EDI solutions. Such projects typically cost hundreds of thousands of dollars and took months to implement. They made sense for large trading partners like an IBM and Ingram, who do a billion dollars a year of business together. However, they did nothing for the tens of thousands of small and medium size companies that comprise the bulk of the electronics industry value chain.

With Web services, any company can publish on its Web server product descriptions, prices and availabilities, shipping and production schedules, etc., in a form that both people and machines can understand. With such information, it suddenly becomes feasible to optimize forecasting, production planning, inventory management and other ERP-like functions across an entire industry, rather than just within a single enterprise.

At least that's the vision. Unfortunately, in practice, there are three problems with Web Services that stand in the way of doing business.

a. **Complex standards**: Web services promised lightweight integration, using simple, open protocols inspired by the Web. Things started out simply enough with SOAP, WSDL and UDDI. However, they rapidly got out of hand as people began adding features that pushed Web services toward a full-fledged CORBA-like distributed operating system. Today there are literally dozens of standards supporting features such as business process orchestration, security, and service provisioning and management. Additionally, there are hundreds of specialized XML languages that define the semantics for doing business in particular industries. Each such language typically entails a painful multi-year standards process whereby companies attempt to reconcile their local data and business process standards. The electronics industry has been attempting to standardize on RosettaNet, which is sponsored by leading manufacturers and distributors. However, after more than five years, only a few RosettaNet processes have moved past initial pilots.

b. **Static processes**: Business demands agility, but BPEL (Business Process Execution Language), the emerging Web services standard for business process orchestration, was designed for static processes. At the least, a rules engine is needed to select the right BPEL script at run time. Even better, rules should be embedded within BPEL so that actions can respond dynamically to events.

c. **Point-to point communications:** Standard Web services rely on point to point communications. However, as we've seen, it's often not clear until run time who needs to know about a design change, or which supplier has the best prices and inventory on hand to fulfill a request.

We can overcome these problems by combining the best features of PACT and Web services. Agents wrap Web services, transforming them into rule-based "Knowledge Services." Facilitators enable agents to communicate on a knowledge level; they route messages on the basis of their content, and translate between local data standards and messaging protocols. And Personal Agents enable humans to participate in automated processes.

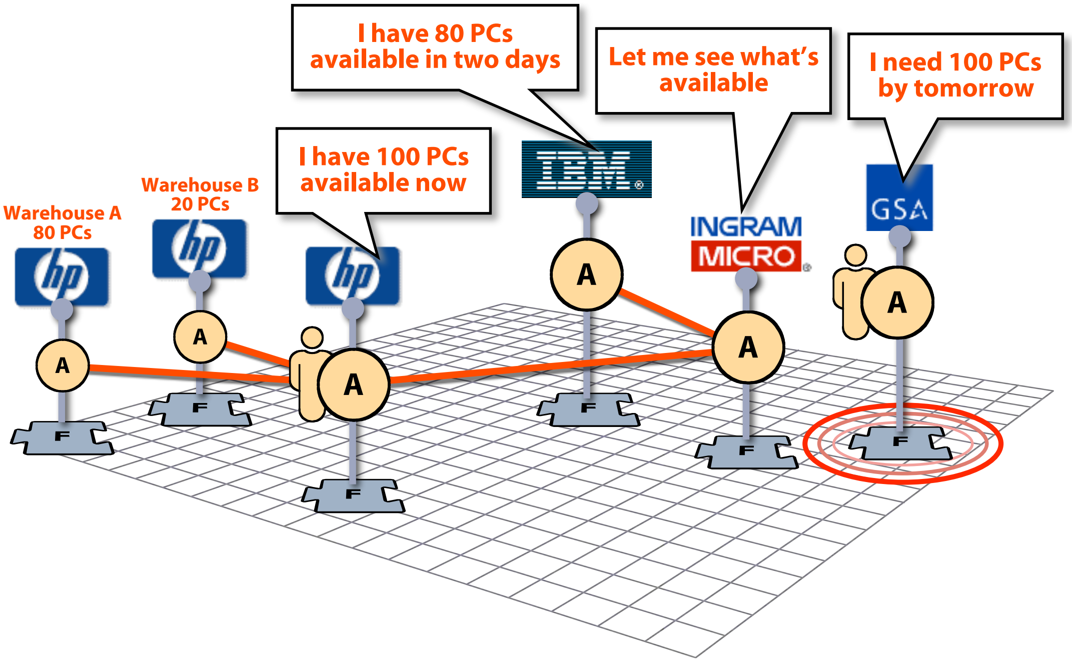

I have 80 PCs available in two days
Let me see what's available
I need 100 PCs by tomorrow
I have 100 PCs available now
Warehouse B 20 PCs
Warehouse A 80 PCs
IBM
INGRAM MICRO
GSA
hp
A
F

Figure 3: Knowledge Services at Work

Figure 3 illustrates how such a solution might work in practice. A purchasing agent at the General Services Administration broadcasts a solicitation for a hundred PCs needed by tomorrow. Ingram, a leading distributor, has an agent that subscribes to such solicitations. The agent picks up the RFQ and forwards it to interested suppliers, whose agents respond with current inventory levels. In the case of one supplier (HP), that involves polling and aggregating inventories from several warehouses. Ingram's agent informs the GSA agent that it can fulfill the request. The order is consummated through Jane, HP's account rep for GSA; her agent automatically approves the order because it falls within GSA's contractual guidelines and credit limit.

3.1 Aggressive Interoperability

I'd like to point out two important features of this knowledge services architecture that we'll be relying on throughout the rest of this presentation. The first feature is Aggressive Interoperation (or "A.I." for short). Aggressive Interoperation means never letting standards, or the lack thereof, get in the way of doing business. If Ingram and HP use different data formats or messaging protocols, then just use a translation service to map them; if there's one thing the AI community is good at, it's mapping between two structured representations. The translation can be performed by the message sender, the recipient, or some third party broker – whoever has the economic incentive. Thus, if HP is eager to attract new customers, it could accept orders in any major format and translate them into its internal standard. One enduring lesson of the Web is that if a service is economically viable, someone will step forward to provide it.

3.2 Incremental Automation

The second feature is Incremental Automation. Let's assume that every step of this cross-organizational purchasing process is initially performed manually, using email, Web browsers and phones. One day, Joe in GSA's stock room, decides he's got better things to do than manually checking a Web site to determine whether there are sufficient PCs on hand. He scripts his Personal Agent to check the inventory level of PCs every morning, and if there are less than 100 on hand, to email a purchase request (or fill out a web-based order form) for approval by his supervisor Bill. Joe's still there to handle the rest of his job as well as unexpected events concerning stocking PCs, but now he's got one less rote task to do.

Joe assumes his agent's request will continue to be processed manually by Bill. However, at some point, Bill gets smart and has his Personal Agent automatically approve routine requests that are within budget and pass them along to John in purchasing. In similar fashion, John might have his Personal Agent pass along routine purchase orders to approved suppliers like Ingram. At the supplier, an order entry clerk might task her agent to check if the order can be fulfilled from stock on hand, and if not, to pass it along to the manufacturer (HP) for drop shipping to the customer (GSA).

> Imagine tens of thousands of individuals at thousands of organizations in a supply chain, each independently and for their own reasons, deciding to automate some routine parts of their jobs. The result is a form of emergent intelligence, where complex processes are automated not through an explicit design process, but rather through the independent actions of many, one person and one task at a time. This instance of mass collaboration is an appropriate segue to our tour of Web 2.0.

4. Web 2.0 Tour

The term "Web 2.0" refers to a collection of emerging Web technologies and methodologies (Table 3) that make the Web more Participatory (i.e., 2-way versus read only), more Semantic, and more Real Time (i.e., event driven). Perhaps most importantly, Web 2.0 is a cultural phenomenon. Developers start with a simple but useful idea, and get it out quickly so others can refine and embellish it. The process has come to be known as mass collaboration – thousands of individuals, building incrementally upon each other's work.

Participatory (P2P)	Blogs, wikis, social networking, RSS feeds, read/write
Semantic	Tags, microformats, semantic web, vertical search
Real time	Instant messaging, events (publish/subscribe)
Pervasive	Billions of edge devices with substantial computing and broadband access – phones, cars, RFID readers…
Community	Simplicity, rapidity, mass collaboration, empowerment

Table 3: Web 2.0 Technologies and Methodologies

I've highlighted the Web 2.0 concepts that are most important to realizing our vision. Because some of them may be unfamiliar, I'm going to take you on a brief tour of these concepts, from an AI perspective. I'll cover both their potential as building blocks for Internet-scale knowledge systems, as well as AI research opportunities for enhancing their functionality. We'll organize the tour around the four highlighted Web 2.0 themes, beginning with making the Web more participatory. I encourage you to visit the referenced Web sites and online figures for more details.

4.1 More Participatory

Our first Web 2.0 theme deals with transforming the Web into a more participatory and personalized medium through blogs, syndication, wikis, and the like.

Blogs, the first major Web 2.0 phenomena, are shared on-line journals where people can post diary entries about their personal experiences and hobbies. Thanks to user-friendly editing tools like *Movable Type* (movabletype.org) and services like *Blogger (*blogger.com*)*, there are now over 12 million blogs, with nine new ones created every minute (according to a recent estimate by *Wired Magazine*). While many blogs are admittedly full of drivel, some contain real time information and insights that would be difficult to find anywhere else. If you're in business or a professional, you ignore Blogs at your peril. If you're in AI and looking for a quick way to get rich, figure out how to mine this trove of information and use the results to predict markets.

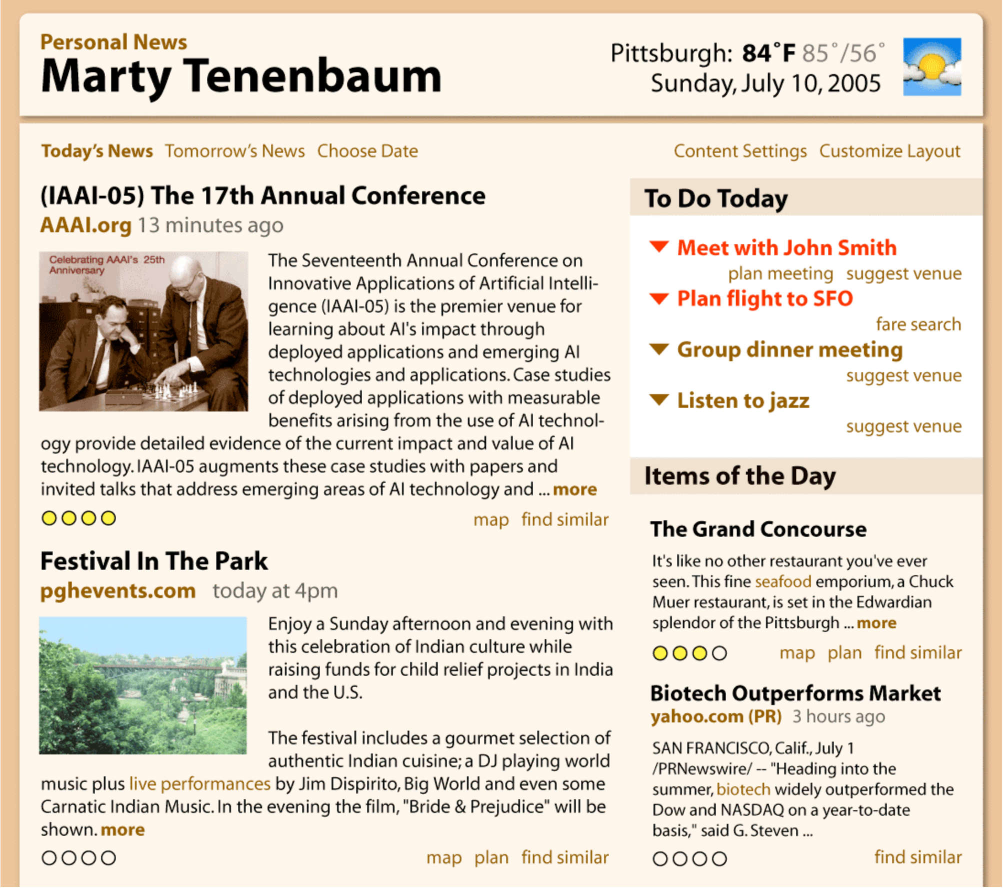
In a world of 12M+ blogs, there are two big challenges: for bloggers, it's getting noticed; for blog readers, it's avoiding information overload. An individual with a browser can at best regularly track a few of dozen sites for updates.

Syndication Feeds, such as RSS and Atom, address both problems by defining an XML format for metadata about new content added to a Web site – information such as the title of a posting, its URL, and a short description. Using a feed reader such as NetNewsWire, an individual can periodically ping the feeds of favorite sites to determine quickly if anything has changed since their last visit. Site owners can also submit their updates to search engines and aggregators. Aggregators subscribe to numerous feeds, and sort postings into subject-specific channels such as "Science News" or "Travel".

By subscribing to syndicated feeds, an individual can effectively track hundreds or even thousands of sites, qualitatively transforming the Web surfing experience.

RSS and Atom may disappoint AI folks with their limited semantic structure and content. In the example above, an aggregator might provide only the most basic information elements, such as the name of the channel ("Yahoo! News: Science News"), a headline ("NASA Sets July 13 Shuttle Launch Date") and a brief natural language summary ("NASA plans to blast into space on July 13 after more than two years on the ground…”). (commerce.net/wiki/semweb2slide29, commerce.net/wiki/semweb2slide30) The AI community can significantly improve the ability to find and filter Web content by enriching the semantics of syndication feeds, and adding inference rules to "feed-reader" software. We’ll identify some specific opportunities later in this paper.

Wikis are Web sites where content is developed and edited collaboratively. The most impressive example is *Wikipedia* (wikipedia.org), a wiki-based encyclopedia with over six hundred thousand articles on every conceivable topic, contributed and collaboratively edited by over 300,000 people. Wikipedia provides interesting lessons for the AI community on developing and maintaining collaborative ontologies. For example, as of last July, the top level subject headings for Geography included Antarctica, Landforms, and Villages. You may not agree that these categories deserve equal billing. The good news is that with a wiki, if you don’t like how things are organized, edit it. If you disagree with someone’s edit, open a discussion.



Personal News

Marty Tenenbaum

Pittsburgh: **84°F** 85°/56°
Sunday, July 10, 2005

Today’s News Tomorrow’s News Choose Date

Content Settings Customize Layout

(IAAI-05) The 17th Annual Conference

AAAI.org 13 minutes ago

The Seventeenth Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-05) is the premier venue for learning about AI's impact through deployed applications and emerging AI technologies and applications. Case studies of deployed applications with measurable benefits arising from the use of AI technology provide detailed evidence of the current impact and value of AI technology. IAAI-05 augments these case studies with papers and invited talks that address emerging areas of AI technology and ... **more**

●●●● map find similar

Festival In The Park

pghevents.com today at 4pm

Enjoy a Sunday afternoon and evening with this celebration of Indian culture while raising funds for child relief projects in India and the U.S.

The festival includes a gourmet selection of authentic Indian cuisine; a DJ playing world music plus live performances by Jim Dispirito, Big World and even some Carnatic Indian Music. In the evening the film, "Bride & Prejudice" will be shown. **more**

○○○○ map plan find similar

To Do Today

▼ **Meet with John Smith**
plan meeting suggest venue

▼ **Plan flight to SFO**
fare search

▼ **Group dinner meeting**
suggest venue

▼ **Listen to jazz**
suggest venue

Items of the Day

The Grand Concourse

It's like no other restaurant you've ever seen. This fine seafood emporium, a Chuck Muer restaurant, is set in the Edwardian splendor of the Pittsburgh ... **more**

●●●○ map plan find similar

Biotech Outperforms Market

yahoo.com (PR) 3 hours ago

SAN FRANCISCO, Calif., July 1 /PRNewswire/ -- "Heading into the summer, biotech widely outperformed the Dow and NASDAQ on a year-to-date basis," said G. Steven ...

○○○○ find similar

Figure 4: The Learning Browser

Imagine the power of combining these participatory technologies with the power of AI. Figure 4 is a mockup of a "Learning Browser" being developed with support from CommerceNet. Unlike traditional aggregators that summarize new feeds by listing headlines or subject lines, it presents the user with a personalized "'*zine*." In the spirit of the Media Lab's "Daily Me", the 'zine is a digest of the latest feeds relevant to the task at hand — whether it's researching a report, following a sport, or in the example shown, planning my trip to Pittsburgh for AAAI-05. Over time, the browser can refine its model of my interests and modify what's presented, based on implicit feedback such as what I click on and what I don't, how long I spend on an article, what I recommend to my friends, what they recommend to me, as well as what's happening in the world. Many sites can benefit from this kind of implicit feedback, from search engines to dating services.

4.2 Semantics

Our second Web 2.0 theme – **Semantics** – includes Tags, Microformats, and Vertical Search.

4.2.1 Tagging & Folksonomies

Tags are personalized labels for describing Web content – web pages, blogs, news stories, photos, and the like. One person might use the terms *Beijing* and *Buildings* to describe a photo of the downtown Beijing skyline (commerce.net/wiki/semweb2slide34). Another might prefer the tags *China* and *Vacation* to describe the same photo. Collectively, the set of tags adopted by a community to facilitate the sharing of content is known as a folksonomy.

Folksonomies organize knowledge differently than traditional classification hierarchies. While tags often correspond to traditional subject headings or categories, they can also represent more subjective descriptors: "Cool", "Recommended", "Reading List", and so forth. Moreover, folksonomies are flat: categories can overlap and items can be classified in multiple ways. Instead of force fitting an item into a traditional classification hierarchy like the Dewey Decimal system, just affix all the tags that apply. Most people find this easier and more intuitive than agonizing over whether a picture shown should be classified under *China*, *buildings*, or *vacation*.

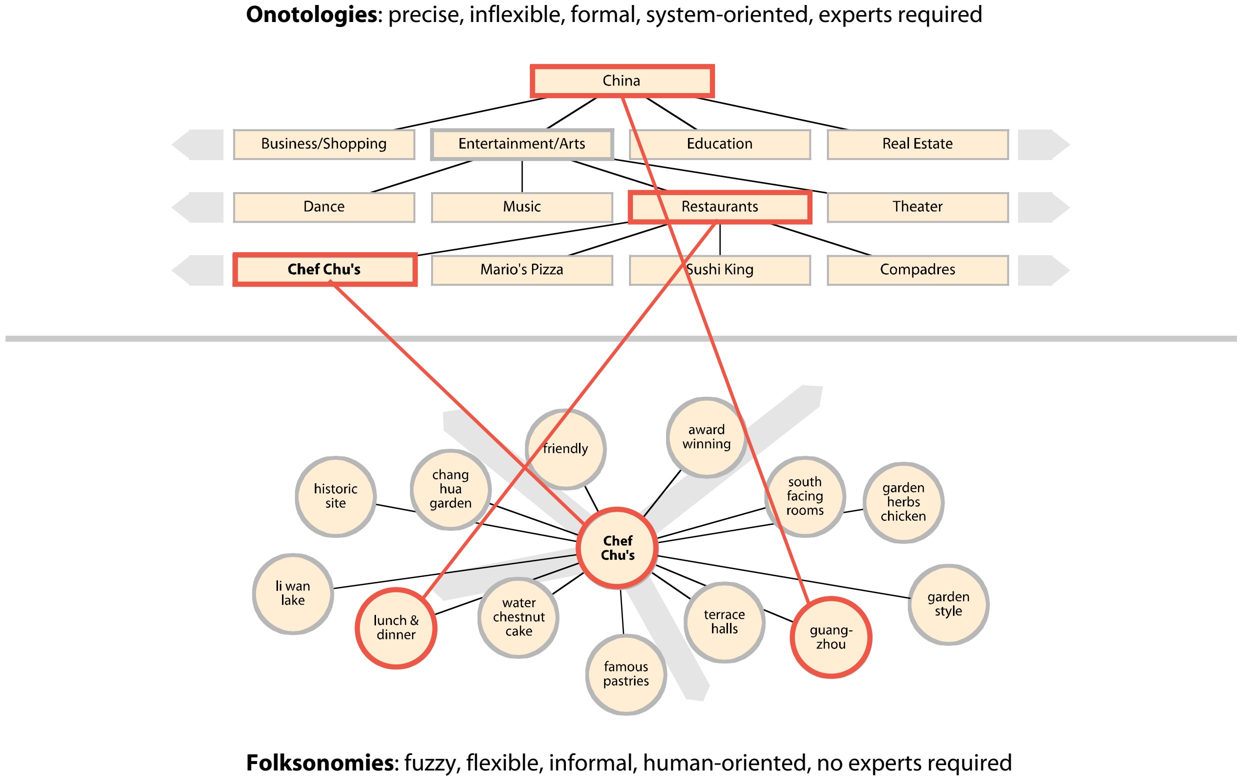
The most defining characteristic of folksonomies, however, is that they evolve through mass collaboration within a community, rather than through the dedicated efforts of one (or a few) expert designers. Unlike most ontologies developed by AI researchers, folksonomies are always widely used – at least within the community that evolved them.

Some of the best examples of collaborative tagging can be found at *Flickr* (flickr.com), a photo sharing site recently purchased by Yahoo!. Most of the photos on Flickr have been tagged by multiple community members. Useful tags stick. Popular tags get reused, and thus reinforced, because people want their content to be found. Frequently co-occurring tags, such as *China* and *Beijing*, provide an opportunity to infer relationships (causal, structural, geographical, etc.).

I'd also like to call your attention to *Del.icio.us* (del.icio.us), the Web site that pioneered collaborative tagging (commerce.net/wiki/semweb2slide36, commerce.net/wiki/semweb2slide37, commerce.net/wiki/semweb2slide38). Del.icio.us lets users assign arbitrary tags to a web page to help them remember it (e.g., pages about China, cool places, hot night spots). However, that's just the beginning. Del.icio.us also lets you see how others tagged the page, which may influence your own choice of tags. You can click on an existing tag, and see all other pages sharing that tag. Additionally, you can drill down and see who affixed a particular tag; other pages they labeled with that tag; as well as any frequently co-occurring related tags, (e.g., *San Francisco* and *California*), and the pages associated with them. I can look generically for *cool* or *recommended* pages, or for pages a specific individual thinks are cool or worth recommending.

The wealth of folksonomy data available from collaborative tagging sites like Flickr and Del.icio.us creates some tantalizing AI research opportunities. One possibility: deriving formal ontologies automatically from raw folksonomies. Imagine taking the collected tags assigned to thousands of restaurant sites in China (Figure 5) and inferring, for example, that Chef Chu's is a restaurant, that restaurants serve lunch and dinner, that Guang-Zhou is a city in China, and so forth. If deriving ontologies automatically proves too hard, a valuable fallback would be to map folksonomies onto existing ontologies. Such a mapping would impose a hierarchical structure on tags, enabling helpful inferences. Knowing that Guang-Zhou is in China, for instance, one could infer that Chef

Chu's was in China, even though "China" did not appear explicitly as a tag for Chef Chu's Web page. A little knowledge can go a long way in increasing the usefulness of tags.



Onotologies: precise, inflexible, formal, system-oriented, experts required

Folksonomies: fuzzy, flexible, informal, human-oriented, no experts required

Figure 5: Ontologies and Folksonomies

4.2.2 Microformats

Microformats are simple data schemas for describing people, places, events, reviews, lists, and so forth – the things people frequently blog about. People use them today to embed structured metadata in Web pages and blogs, so that their content can be more easily discovered, indexed, aggregated, and summarized. Soon, microformats will also be used by knowledge services. Microformats are an important development for the AI community because they promise to significantly increase the availability of structured Web content. I'm therefore going to spend some time on them.

Microformats adhere to five basic design principles, articulated by Tantek Çelik of Technorati, which appear to be driving their popularity and rapid adoption (See Table 4):

- Solve a specific problem
- Start as simply as possible, and evolve
- Humans first and machines second
- Reuse existing widely adopted standards
- Modular and embeddable

Table 4: Tantek Çelik's Microformat Design Principles

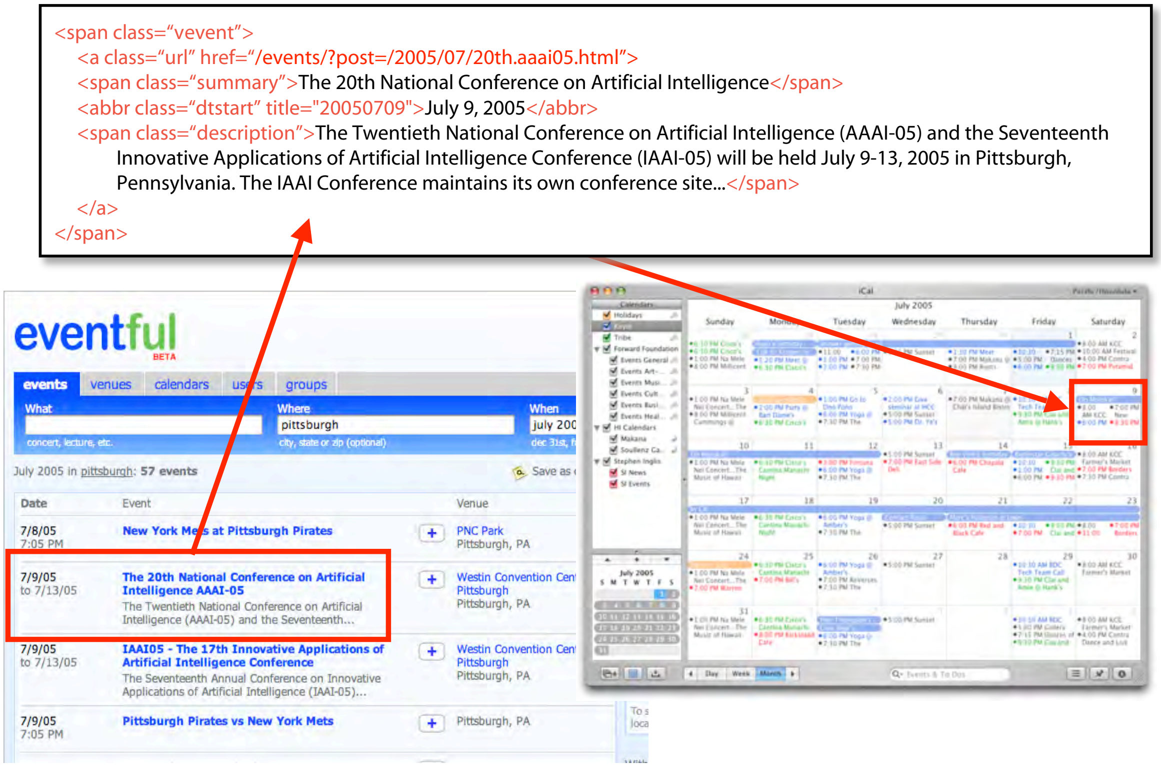
a. Design microformats to **solve a specific problem**, such as keeping track of your friends, or sharing calendar events. Develop them in tandem with an initial Web service or application that uses them. As John Seeley Brown observed, Microformats win because they include pragmatics as well as semantics.

b. Start with the **simplest possible representation** for the problem at hand, deploy it quickly, and add more features only when forced by a real application. By simple, think vCard – where a person might be represented initially by just their name and contact information.

c. Design microformats for **humans first and machines second**. Microformats consist of a little bit of metadata embedded in ordinary HTML – just enough so that a computer can understand the meaning of what is being displayed. In Tantek's words, information on the Web should be "Presentable ***and*** Parseable". Microformats do that, as simply as possible.

d. Whenever possible, **reuse existing well-known standards** rather than wasting time and money inventing and marketing new ones. HTML, hCard (based on the vCard format) and hCalendar (based on the iCalendar format for events) – few standards are more well-known.

e. Microformats should be **modular and embeddable**. A reservation booking service, for example, might use as input, a composite microformat that contained the previously mentioned microformats for personal information and event descriptions.



```
<span class="vevent">
    <a class="url" href="/events/?post=/2005/07/20th.aaai05.html">
    <span class="summary">The 20th National Conference on Artificial Intelligence</span>
    <abbr class="dtstart" title="20050709">July 9, 2005</abbr>
    <span class="description">The Twentieth National Conference on Artificial Intelligence (AAAI-05) and the Seventeenth
          Innovative Applications of Artificial Intelligence Conference (IAAI-05) will be held July 9-13, 2005 in Pittsburgh,
          Pennsylvania. The IAAI Conference maintains its own conference site...</span>
    </a>
</span>
```

July 2005 in pittsburgh: **57 events**

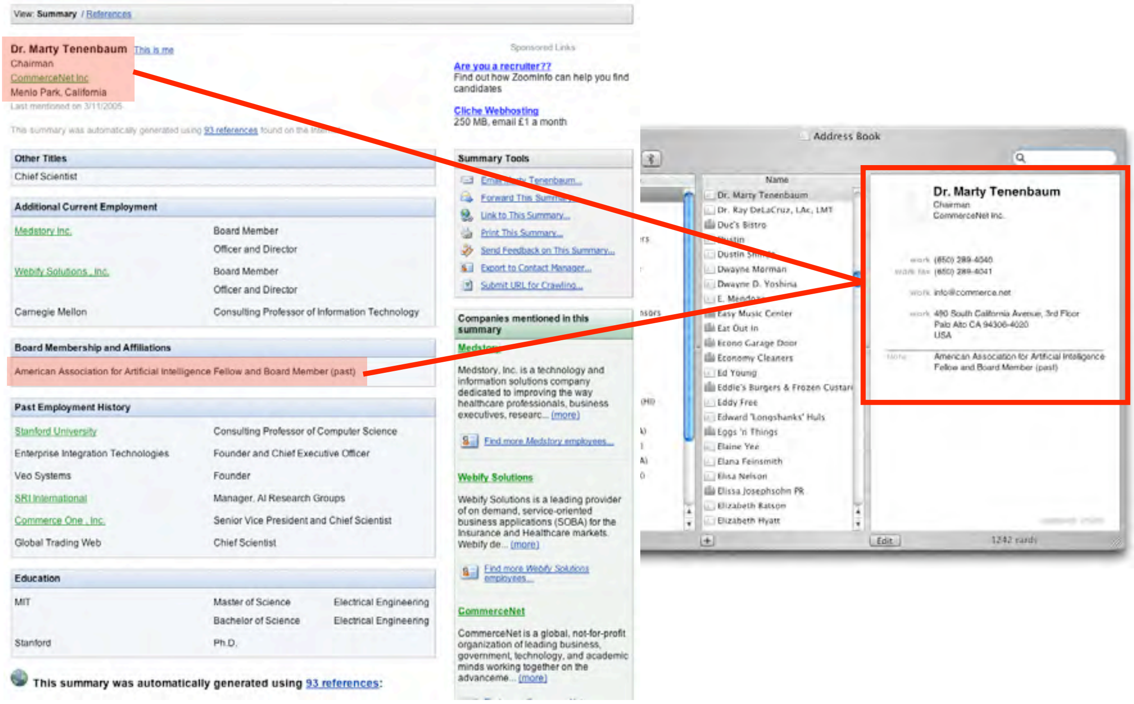
Date	Event	Venue
7/8/05 7:05 PM	**New York Mets at Pittsburgh Pirates**	PNC Park Pittsburgh, PA
7/9/05 to 7/13/05	**The 20th National Conference on Artificial Intelligence AAAI-05** The Twentieth National Conference on Artificial Intelligence (AAAI-05) and the Seventeenth...	Westin Convention Cen Pittsburgh Pittsburgh, PA
7/9/05 to 7/13/05	**IAAI05 - The 17th Innovative Applications of Artificial Intelligence Conference** The Seventeenth Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-05)...	Westin Convention Cen Pittsburgh Pittsburgh, PA
7/9/05 7:05 PM	**Pittsburgh Pirates vs New York Mets**	Pittsburgh, PA

FIgure 6: The hCalendar Microformat

Figure 6 illustrates how adding tiny bits of markup through microformats can elegantly transform any Web site into a structured knowledge source. On the left of the screen is a page from *Eventful* (eventful.com), a Web site that aggregates event listings. On the top is the vanilla HTML source for that page. Look carefully, however, and you'll see a few highlighted insertions. These are hCalendar tags, and they tell a computer the meaning of each block of displayed text.

Note that the hCalendar tags piggyback on existing HTML syntax extensions for Cascading Style Sheets, such as <span> and <class>.

Thanks to the hCalendar tags, Eventful users can highlight events of interest, and import them directly into their personal calendars, in this example Apple's iCal program. A free Web service, available at suda.co.uk/projects/X2V, translates the hCalendar-formatted event listings used by Eventful into the iCalendar standard. Soon, a Personal Agent might use these listings to alert you to upcoming events of interest. You'll never again have to kick yourself for missing an event because you didn't hear about it in time.



View: Summary / References

Dr. Marty Tenenbaum This is me
Chairman
CommerceNet Inc
Menlo Park, California
Last mentioned on 3/11/2005

This summary was automatically generated using 93 references found on the Inter

Other Titles

Chief Scientist

Additional Current Employment

Medstory Inc.	Board Member Officer and Director
Webify Solutions , Inc.	Board Member Officer and Director
Carnegie Mellon	Consulting Professor of Information Technology

Board Membership and Affiliations

American Association for Artificial Intelligence Fellow and Board Member (past)

Past Employment History

Stanford University	Consulting Professor of Computer Science
Enterprise Integration Technologies	Founder and Chief Executive Officer
Veo Systems	Founder
SRI International	Manager, AI Research Groups
Commerce One , Inc.	Senior Vice President and Chief Scientist
Global Trading Web	Chief Scientist

Education

MIT	Master of Science	Electrical Engineering
	Bachelor of Science	Electrical Engineering
Stanford	Ph.D.	

This summary was automatically generated using 93 references:

Sponsored Links

Are you a recruiter??
Find out how ZoomInfo can help you find candidates

Cliche Webhosting
250 MB, email £1 a month

Summary Tools

Emai Tenenbaum...
Forward This Summ
Link to This Summary...
Print This Summary...
Send Feedback on This Summary...
Export to Contact Manager...
Submit URL for Crawling...

Companies mentioned in this summary

Medsto

Medstory, Inc. is a technology and information solutions company dedicated to improving the way healthcare professionals, business executives, researc... (more)

Find more Medstory employees...

Webify Solutions

Webify Solutions is a leading provider of on demand, service-oriented business applications (SOBA) for the Insurance and Healthcare markets. Webify de... (more)

Find more Webify Solutions employees...

CommerceNet

CommerceNet is a global, not-for-profit organization of leading business, government, technology, and academic minds working together on the advanceme... (more)

Address Book

Name
Dr. Marty Tenenbaum
Dr. Ray DeLaCruz, LAc, LMT
Duc's Bistro
ustin
Dustin Sh
Dwayne Morman
Dwayne D. Yoshina
E. Mendo
Easy Music Center
Eat Out In
Econo Garage Door
Economy Cleaners
Ed Young
Eddie's Burgers & Frozen Custar
Eddy Free
Edward 'Longshanks' Huls
Eggs 'n Things
Elaine Yee
Elana Feinsmith
Elisa Nelson
Elissa Josephsohn PR
Elizabeth Batson
Elizabeth Hyatt

Dr. Marty Tenenbaum
Chairman
CommerceNet Inc.

work (650) 289-4040
work fax (650) 289-4041
work info@commerce.net
work 490 South California Avenue, 3rd Floor
Palo Alto CA 94306-4020
USA
Note American Association for Artificial Intelligence Fellow and Board Member (past)

Edit
1242 cards

FIgure 7: The Semantic Highlighter

Ideally, one should be able to extract micro-formatted content from any Web page for use by knowledge services and applications. The Semantic Highlighter project at CommerceNet Labs is a step toward that goal. In the screen mockup shown in Figure 7, the user has highlighted selected items of interest on a Web page. The program then creates a structured representation of that information by selecting and populating relevant microformat(s). In this example, the software is filling out a vCard describing me. Note that it has to follow the URL to

CommerceNet's Web site to obtain my contact information. In general, given some highlighted data on a Web page, an arbitrary amount of interference may be required to pick relevant microformats, and fill in missing fields.

To learn more, check out microformats.org, which CommerceNet helps sponsor.

4.2.3 Vertical Search

The final concept for making the Web more Semantic is **Vertical Search**. Horizontal search engines such as Google work well for finding popular Web sites. However, when the site you want is ranked 13,000 out of 147,000, or when the answer to your question must be compiled from information spread over many sites, current search engines are found wanting.

Vertical search engines overcome these limitations by using models of the user, the domain, and the available information sources to determine where and how to look. Vertical search often exploits deep web sources that aren't accessible to Google's spiders, either because they are proprietary or require specialized access protocols (e.g., SQL queries, filling out forms). The raw information retrieved from these sources is initially organized into data- or knowledge-bases. Domain and user models are then applied to connect the dots and return useful answers. Let's look at a three examples:

i. **Zoominfo** (commerce.net/wiki/semweb2slide49) is a vertical search engine for people in the business and professional worlds. Its spiders scour the Web for information about individuals, which is then aggregated into personal CVs. AI-based techniques are used to recognize redundant CVs that describe the same person, and merge them into a composite summary of career accomplishments. Typing in my name, for instance, produces a CV compiled from over 100 distinct Internet sources. It's quite comprehensive and reasonably accurate.

ii. **Dulance** (commerce.net/wiki/semweb2slide50) is a shopping search engine that scours the Web for information about products and prices. Dulance is not the first shopping bot, but it's the first one I know of that offers syndicated feeds to inform users when the memory card they want becomes available at their price.

iii. **Medstory** (commerce.net/wiki/semweb2slide51) is a healthcare search engine. Its spiders scour the deep Web, focusing on high quality information sources that are inaccessible to Google's Web crawlers, such as the National Library of Medicine's PubMed database and subscription-based medical journals. It organizes the information it finds in a relational database and then applies domain models of medicine and related fields to infer deeper connections. For example, it can use models of molecular pathways that are common to several diseases to suggest novel uses of drugs and infer potential side-effects.

Looking ahead, I see vertical search engines becoming a major source of structured information for knowledge services. Agents will be able to submit microformatted query forms containing some blank fields, and receive microformatted responses with those fields instantiated. Search engines themselves will increasingly rely on collaboratively tagged and micro-formatted content to improve search results, especially within professional communities like healthcare and pharmaceutical research.

4.3 The Real Time Web

The third Web 2.0 theme is making the Web more **real time** and **event driven**. A new generation of real-time search engines, such as Technorati and PubSub, are indexing the fast-changing subset of the Web that some call the "World Live Web." Technorati provides the latest blog, Flickr, and Del.icio.us posts, usually within minutes of their posting. PubSub tracks over 12 million sources, primarily blogs and news feeds. Users subscribe to topics of interest, and receive IM alerts as soon as items of interest are published.

How do these real time search and alerting services work? Several major blogging tools ping a trackback cloud when new items are posted. Technorati and PubSub.com monitor this cloud and use the postings to update their databases and alert users. While perhaps adequate for tracking blogs, which change relatively slowly, such centralized notification service are not likely to scale.

A truly Internet-scale event bus must accommodate potentially billions of agents publishing and subscribing to each other's information and events in near-real-time (100 milliseconds to a few seconds, as fast as the underlying networks permit). And it must route information intelligently, not blindly copy spam. A supply chain manager tracking events at airports is likely to be interested only in those events that could interrupt his supply chain – a fire or strike for example, but not the appointment of a new landscaping contractor. And once notified, he doesn't need 20 redundant messages from other news sources on the same event. Such a notification service requires a federated peer-to-peer architecture and intelligent agents that can filter and route content using domain knowledge and context, not just keywords.

At CommerceNet Labs, we're prototyping such a federated, Internet-scale event bus (commerce.net/wiki/semweb2slide54). Agents aggregate events in their areas of interest and also act as facilitators, routing selected items to other agents. An agent aggregating auto industry news, for example, might pass along items that affect the price of cars to an agent aggregating car rental industry news. That agent, in turn, might forward news affecting the price or availability of rental cars to an agent aggregating travel industry news. CommerceNet's notification service is being built on TPd, an experimental pub-sub network infrastructure for semi-structured syndication feeds. Unlike other middleware systems, it is designed from the ground up to deal with the potentially-hostile world of decentralized information, where multiple competing sites and users are sharing their own opinions of the world. In AI terms, it provides a "blackboard" for triggering agents when the world changes.

Internet-scale pub-sub is a potentially disruptive technology because it reduces the need for big aggregators like eBay and Google. Once millions of players can directly subscribe to offers to buy and sell, they'll invent many new business models and ways to add value. When the information you need can find you – that your order is ready for pickup, that a new drug is available for your disease — without your having to search for it, life will be a little easier.

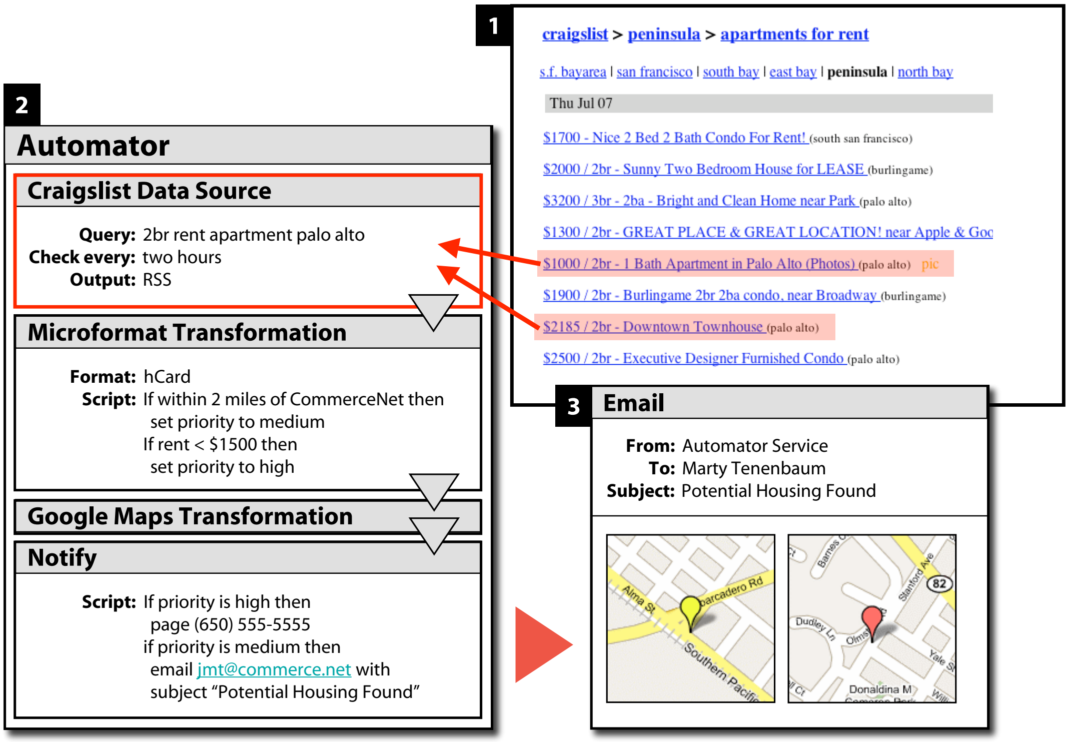
4.4 Community Empowerment

The fourth major thrust of Web 2.0 is about empowering end-users. A great example is **Greasemonkey**. Greasemonkey is a Firefox extension that automatically invokes Javascripts when selected URLs are loaded. The scripts can transform the appearance of pages, perform computations, and even invoke external Web resources. A popular

Greasemonkey script is Book Burro (commerce.net/wiki/semweb2slide55), written by Jessie Andrews at the University of Kentucky. Activated by any Amazon page offering a book for sale, Book Burro fetches and displays prices from competing booksellers. Another good example is Greasemap (commerce.net/wiki/semweb2slide56). Greasemap scans every page for street addresses, sends them to Google's mapping service, and displays the results at the top of the page.

By inserting computation between the viewer and content source, Greasemonkey scripts let users take charge of their screen real estate. Imagine the possibilities if one could insert logic, decision making and learning. What if ordinary users could create scripts using visual programming techniques, without having to become JavaScript wizards? Our goal at CommerceNet Labs is precisely that: to give everyone a Personal Agent tool that can turn any Web site into a knowledge service. Here's a prospective scenario, inspired by the Semantic Highlighter, and by Apple Automator, a popular visual scripting tool for Macs (apple.com/macosx/features/automator/).

The task is to find a reasonably priced apartment near work. I start by going to *Craigslist* (craiglist.org) and highlighting a few apartments of interest (Figure 8.1). Generalizing from those examples, my agent infers that I'm interested in two bedroom apartments in the Palo Alto area. Next, I create the Automator-like script shown in Figure 8.2: "Check every two hours for new listings. If they're within two miles of work then notify me by email (see Figure 8.3). However if the rent is less than $1500, page me." Imagine millions of people composing scripts like this and publishing them as services that others can use and build on. Such a vision would indeed transform AI and the Web as we know them.



craigslist > peninsula > apartments for rent

s.f. bayarea | san francisco | south bay | east bay | **peninsula** | north bay

Thu Jul 07

$1700 - Nice 2 Bed 2 Bath Condo For Rent! (south san francisco)

$2000 / 2br - Sunny Two Bedroom House for LEASE (burlingame)

$3200 / 3br - 2ba - Bright and Clean Home near Park (palo alto)

$1300 / 2br - GREAT PLACE & GREAT LOCATION! near Apple & Goo

$1000 / 2br - 1 Bath Apartment in Palo Alto (Photos) (palo alto) pic

$1900 / 2br - Burlingame 2br 2ba condo, near Broadway (burlingame)

$2185 / 2br - Downtown Townhouse (palo alto)

$2500 / 2br - Executive Designer Furnished Condo (palo alto)

Automator

Craigslist Data Source

Query: 2br rent apartment palo alto
Check every: two hours
Output: RSS

Microformat Transformation

Format: hCard
Script: If within 2 miles of CommerceNet then
set priority to medium
If rent < $1500 then
set priority to high

Google Maps Transformation

Notify

Script: If priority is high then
page (650) 555-5555
if priority is medium then
email jmt@commerce.net with
subject "Potential Housing Found"

Email

From: Automator Service
To: Marty Tenenbaum
Subject: Potential Housing Found

FIgure 8: Transforming Web Sites Into Knowledge Services via Personal Agents

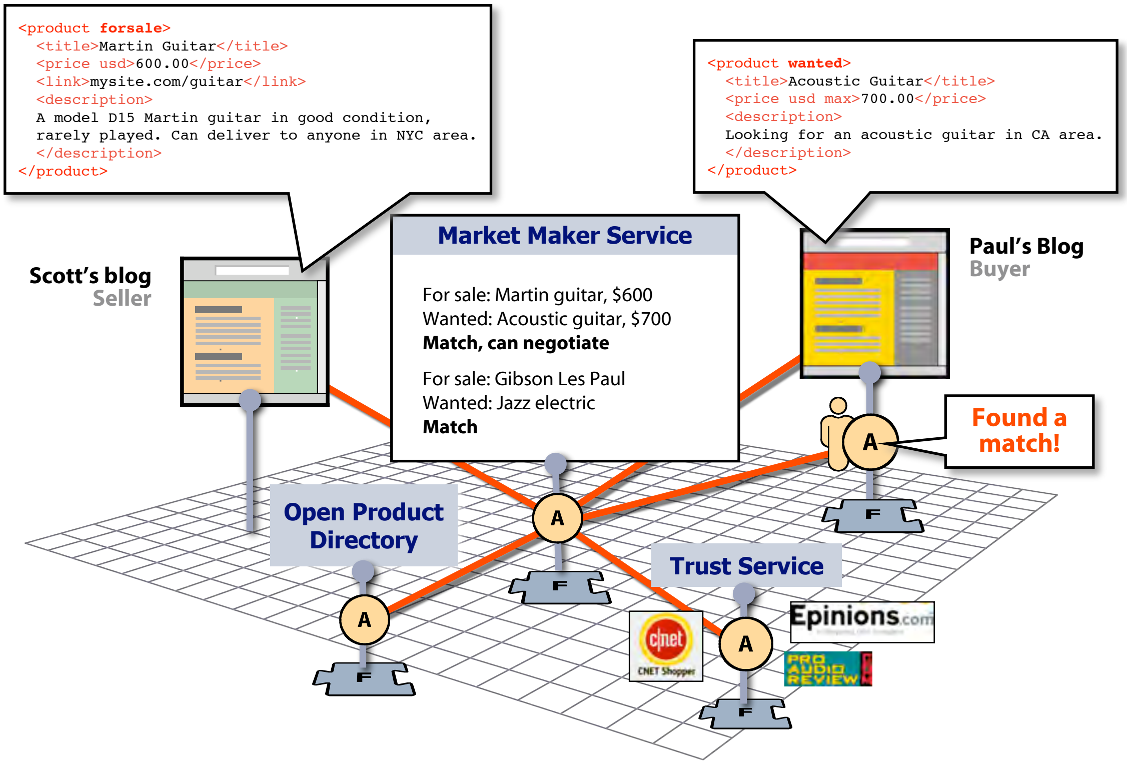
5. Case Studies & Scenarios

Having concluded our tour of Web 2.0, we're now ready to talk about *growing* Internet-scale knowledge systems that combine Web 2.0 concepts with intelligent agents and Web Services. We'll illustrate such systems using both visionary scenarios and case studies of actual deployed systems.

5.1 zBay

zBay (as in decentraliZed) is an ecommerce scenario we're exploring at CommerceNet, that involves literally blowing up today's centralized ecommerce ecosystems. Think eBay without an eBay in the center! zBay was inspired by a prescient 2002 blog posting by writer and consultant Paul Ford (ftrain.com/google_takes_all.html). Ford wanted to illustrate the potential of the Semantic Web. However, his scenario is a lot more credible with Web 2.0 technologies like microformats and event buses.

In zBay, buyers and sellers post their desires directly on their blogs, using a microformat for classified ads (see Figure 9). In this example, Scott on the left wants to sell a Martin Guitar for $600, while Paul on the right is seeking an Acoustic guitar under $700. The problem is how to connect them? Step one is aggregating the raw listings. Posting tools might submit listings automatically to large aggregators such as Google and Craigslist. Smaller, specialty aggregators could spider for listings. These specialty market makers can now apply deep domain knowledge, including open wiki-based product ontologies, to match up compatible buyers and sellers. A market maker specializing in used guitars would know, for example, that most Martin guitars are acoustic, while most Gibsons are electric. The big advantage of decentralized markets is that many specialists can add value by bringing their domain knowledge to bear.



```
<product forsale>
  <title>Martin Guitar</title>
  <price usd>600.00</price>
  <link>mysite.com/guitar</link>
  <description>
  A model D15 Martin guitar in good condition,
  rarely played. Can deliver to anyone in NYC area.
  </description>
</product>
```

```
<product wanted>
  <title>Acoustic Guitar</title>
  <price usd max>700.00</price>
  <description>
  Looking for an acoustic guitar in CA area.
  </description>
</product>
```

Market Maker Service

For sale: Martin guitar, $600
Wanted: Acoustic guitar, $700
Match, can negotiate

For sale: Gibson Les Paul
Wanted: Jazz electric
Match

FIgure 9: zBay

Blowing up the center unleashes other opportunities for innovation. Paul himself might decide to be become proactive, and task his personal agent with monitoring favorite dealer sites for new guitar listings. Knowledge-based Trust services can lubricate the market, using social networking and blogs to track the reputations of buyers and sellers, and compile product reviews. Other service providers can facilitate fulfillment, shipping and payment. These ideas extend directly to industrial-strength B2B applications.

5.2 Electronics Industry Supply Chain

The electronics industry is fiercely competitive. Short product cycles make forecasting supply and demand difficult. Error rates of 50% or more are not uncommon. Firms must closely monitor changes in supply and demand, and respond instantly to keep them balanced. The traditional sources of supply and demand signals are channel partners, such as distributors and resellers. Increasingly, however, the most up to date information comes from Web 2.0 sources including news feeds, blogs and real time sensors.

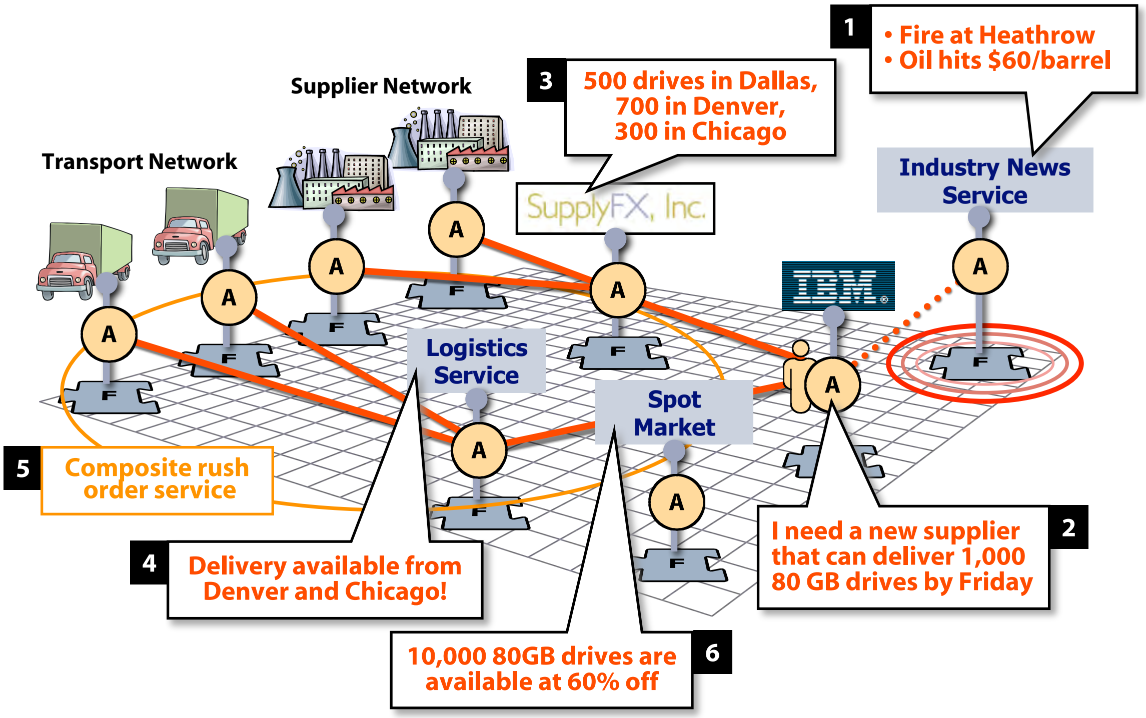

1
• Fire at Heathrow
• Oil hits $60/barrel
3
500 drives in Dallas, 700 in Denver, 300 in Chicago
Supplier Network
Transport Network
SupplyFX, Inc.
Industry News Service
IBM
A
F
Logistics Service
Spot Market
5
Composite rush order service
4
Delivery available from Denver and Chicago!
2
I need a new supplier that can deliver 1,000 80 GB drives by Friday
6
10,000 80GB drives are available at 60% off

FIgure 10: Electronics Industry Knowledge Services

Let's return to our earlier electronics industry scenario (Figure 3) and extend it to accommodate these Web 2.0 resources. In the spirit of incremental automation, a supply chain manager might initially just have his personal agent monitor relevant feeds and notify him of events potentially warranting attention (Figure 10.1). If you had a critical shipment of disk drives that was scheduled to be routed through Heathrow around the time of last summer's London bombings, you would have been wise to explore alternative sources of supply. The manager or his personal agent prepares an RFQ (Figure 10.2) and submits it to *SupplyFX* (supplyfx.com), a live, commercial Knowledge Service for sourcing electronic parts (Figure 10.3). SupplyFX expedites sourcing by selectively forwarding RFQs to interested suppliers in its network, and aggregating their responses.

Fortunately, drives are available from several suppliers. But can they arrive by Friday? Our supply chain manager submits that requirement to a third-party logistics service, which queries its network of carriers and determines that Friday delivery is possible from Denver and Chicago (Figure 10.4). Problems at Heathrow will not disrupt production! SupplyFX, sensing an opportunity, assembles and publishes a one-stop composite service for rush orders, that takes into account both product availability and shipping schedules (Figure 10.5).

But why stop here? There are potentially thousands of value-added knowledge services.

How about a knowledge service that tracks suppliers and learns which ones provide the most reliable on-time delivery? Or a demand forecasting service that monitors blogs for subtle demand signals, such as an emerging interest in 80GB iPods. Or a spot market service that notifies subscribers of buying opportunities that arise when another manufacturer loses a large order and is forced to dump inventory (Figure 10.6). No ERP system can deal effectively with such opportunistic, real time information. Humans aren't very good at it either, due to information overload and limits on time-bounded decision making. Don't you think a little AI could help?

5.3 Case Study: Webify's Insurance Ecosystem

Webify (webifysolutions.com) is commercially deploying simple agent-based knowledge systems today in the insurance industry. The insurance industry is a complex distributed ecosystem. It includes the primary carriers, their agents, underwriters, and claims adjusters, re-insurers, fraud investigators, and many other service providers (See Figure 11). Webify uses agents, operating within a service-oriented architecture, to route documents among service providers based on content. Routine claims, for example, can be routed automatically to $5-per-hour adjusters in Bangalore, while major claims with the potential for significant payouts are routed to $80-per-hour in-house specialists in Hartford. This ability to selectively outsource claims is especially helpful following disasters, enabling carriers to handle peak loads without unduly inconveniencing other customers.

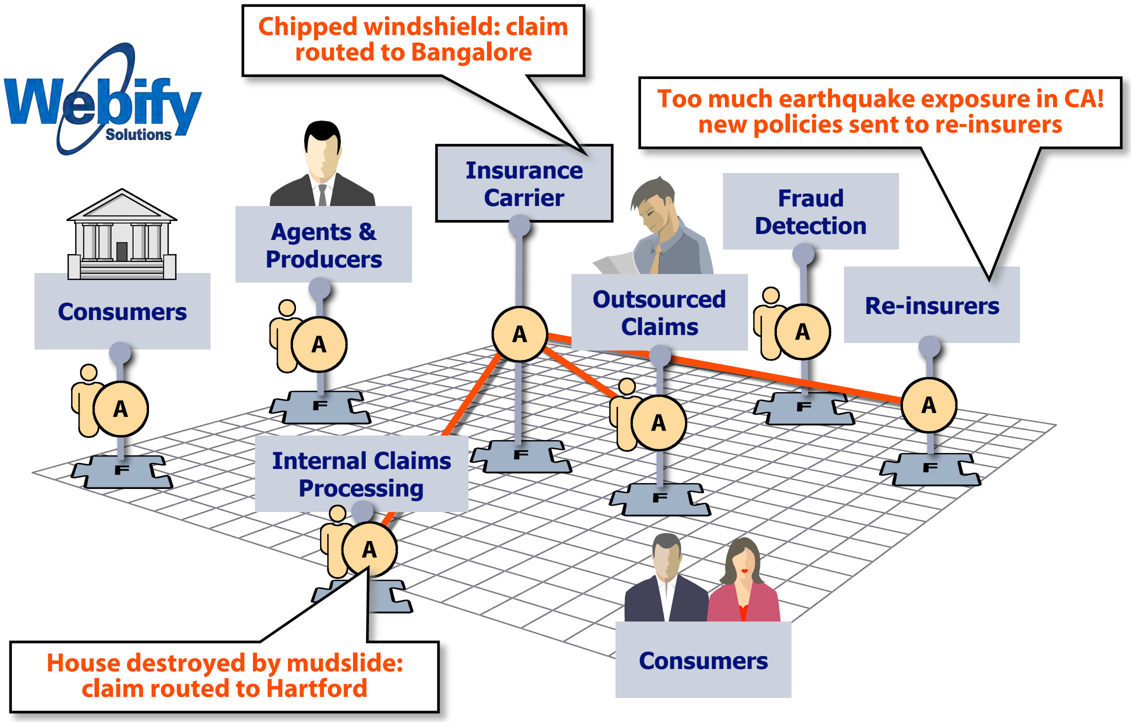

Webify Solutions
Chipped windshield: claim routed to Bangalore
Too much earthquake exposure in CA! new policies sent to re-insurers
Insurance Carrier
Agents & Producers
Fraud Detection
Consumers
Outsourced Claims
Re-insurers
A
F
Internal Claims Processing
House destroyed by mudslide: claim routed to Hartford
Consumers

FIgure 11: Insurance Industry Knowledge Services

There are many other opportunities for knowledge services to incrementally improve the performance of distributed insurance ecosystems. For example, new policy applications for coverage can be automatically off-loaded to re-insurers when the exposure to a given risk exceeds predetermined levels. An outright moratorium on new hurricane coverage can be instantly imposed at the first indication of an approaching storm. Suspect claims can be forwarded to fraud investigators, and so forth.

Insurance was a prime domain for expert systems in the 1980's. Several companies, such as Syntelligence, were formed specifically to address this lucrative market. None are still around. Their technology consisted of large, monolithic mainframe-based systems. It was ill-suited to decentralized insurance ecosystems, where integration and automation are best approached incrementally, one organization and one task at a time. Web services overcome this problem in insurance and other industries. They enable companies to blow up existing value chains and reassemble the pieces into radically new cross-company processes and business models.

5.4 E-science

Scientific problems are increasingly complex and multidisciplinary, and thus beyond the reach of a single individual or group. In response, many scientific communities are now relying on Internet-based grids for sharing information, computing, and other services. A recent issue of *Science* contained three papers on Service-oriented Science. Most of the services discussed, however, involve only the sharing of data and computing resources. Knowledge services offer a significant extension, enabling groups to share their specialized knowledge and expertise.

Alain Rappaport of *Medstory* (medstory.com) is pioneering the application of knowledge services in biomedical research. In collaboration with colleagues at MIT, CMU, and the Institute for Human and Machine Cognition, he's developing a knowledge services grid for mobilizing the resources of the global scientific community in response to biological threats (see Figure 12). Suppose a virologist at CDC is confronting a suspicious new virus? Urgent questions spring to mind. Is it natural or engineered? How is it molecularly related to other known infectious agents? Does that relationship present opportunities for intervention?

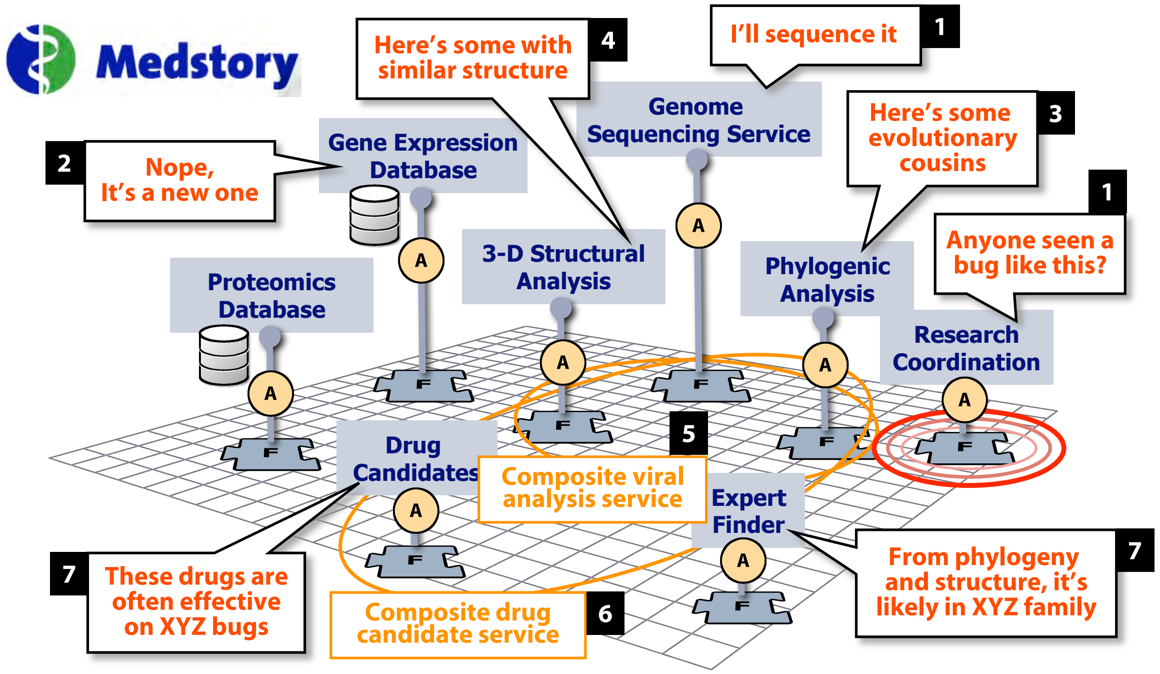

Medstory
1
I'll sequence it
4
Here's some with similar structure
Genome Sequencing Service
3
Here's some evolutionary cousins
2
Nope, It's a new one
Gene Expression Database
1
Anyone seen a bug like this?
A
3-D Structural Analysis
Phylogenic Analysis
Proteomics Database
Research Coordination
F
5
Drug Candidates
Composite viral analysis service
Expert Finder
7
These drugs are often effective on XYZ bugs
Composite drug candidate service
6
From phylogeny and structure, it's likely in XYZ family
7

FIgure 12: Service-Oriented Science

The E-science grid springs into action. First, a sequencing center with cutting edge viral RNA chips analyzes the new virus's genome and publishes it to the grid (Figure 12.1). Based on that data, a knowledge service associated with the leading gene expression database concludes that it is indeed a new bug (12.2). Other knowledge services look for close relatives to the new bug based on its phylogenetic class (evolutionary neighbors) and 3-D molecular structure (12.3 and 12.4). Experts on these related viruses are alerted. One of those experts creates a knowledge service that synthesizes the phylogenetic and structural analyses to determine the precise family of known viruses the new bug most closely resembles (12.5). A pharmacology expert can now combine this composite virus classifier with a service that provides access to a database of anti-viral drugs, creating a knowledge service that can take a viruses DNA and suggest the best drugs to defeat it. (12.6).

The circles depicted by 12.5 and 12.6, and thousands more like them, represent composite knowledge services. Most will be created by scientists for their own use, and then made available for others to build on. Collectively they will transform how science is done. Knowledge services are already impacting how drugs are developed (and not just for bio-emergencies). Increasingly, drugs are developed collaboratively by academic researchers, biotech scientists and big pharma marketers, supported by a vast ecosystem of third party services – combinatorial chemistry, toxicity assays, proteomic analyses, clinical trials management, and so forth. Knowledge services help facilitate the sharing of information and expertise, the allocation of resources and the coordination of workflows across organizations.

5.5 E-life

Our final example is an e-life scenario: planning a dinner for tonight in Pittsburgh.

The task breaks down into a series of sub tasks (See Figure 13): determining who is in town that I might like to spend some time with? Are they available and interested in meeting for dinner? Finding a restaurant; booking reservations; making arrangements and coordinating any last minute changes in plans. These are precisely the types of tasks the Web excels at.

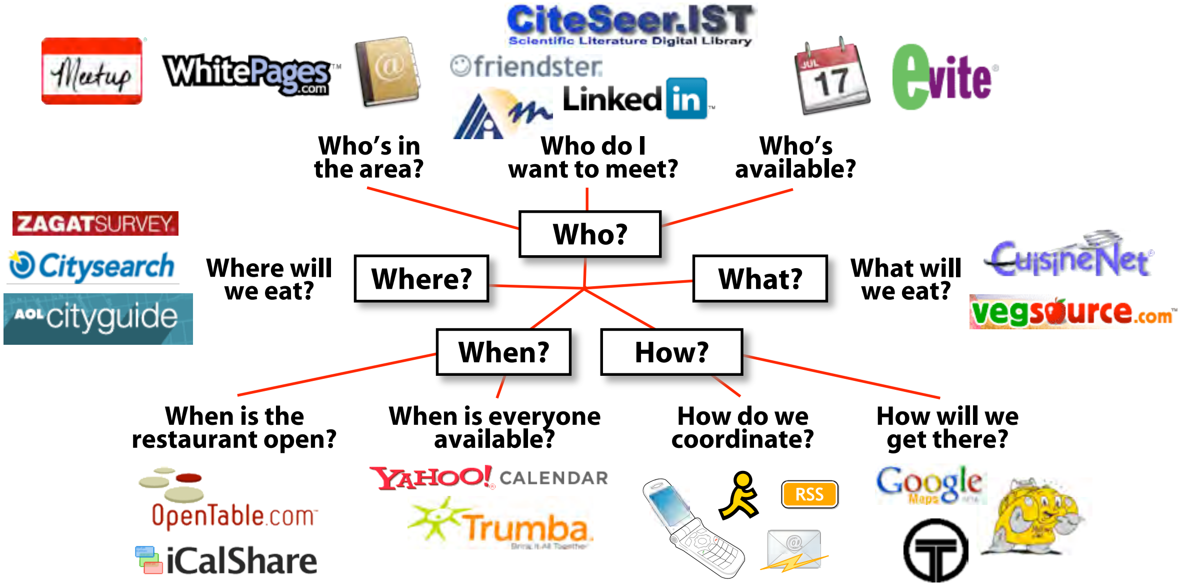


FIgure 13: E-life

Let's start with who's in town that I might like to meet. That list would include people I know who live in Pittsburgh, as well as conference attendees with whom I have something in common – perhaps authors of papers I've liked, or colleagues who liked the same papers I did, or those who fund research in my area. I might start by asking my Personal Agent to compile a list of the people I know who live in Pittsburgh, using contacts from my address book and social networking sites. To compile a list of conference attendees, someone would have to create a service that scraped the conference site for speakers, session chairs and the like, and published the results as a micro-formatted list; in the future, conference organizers might routinely publish structured lists of registered attendees. Someone else might build a collaborative filtering service for CiteSeer, where users could share ratings of papers. Given such services, a simple agent could correlate the list of conference attendees with the authors of papers I liked, to suggest possible dinner guests.

Given a list of potential dinner guests, an agent could use email or a service like Evite to see who's available. An incremental service, built on Evite, might offer responders the opportunity to fill out a micro-form, indicating preferences regarding cuisine, location, etc. Next up is selecting a place to eat. There are numerous sites on the Web that rate and review restaurants. A helpful knowledge service could use these existing sites, together with guests' indicated preferences, to recommend restaurants. Another knowledge service could then check availability at Open Table, book the reservation, and notify the guests.

Each of these knowledge services is helpful in its own right, so it's a good bet that someone will create services like them. Once they exist, it won't take long before someone drops in a constraint-based planner and offers a completely automated dinner planning service.

6. Conclusions

I'm convinced, and I hope I've convinced you that the time has come to rethink how we do AI applications in the context of the Web. This is not an academic exercise. It's not a question of if or even when this will happen. It's already happening in the Web community. We just need to insert a little AI! And if the AI community doesn't step up to this opportunity, others will reinvent AI for us.

Stripped to its essence, we're talking about:

- Using microformats and agents to transform ordinary Web resources into Knowledge Services
- Enabling these Knowledge Services to communicate directly and in a content-directed fashion, via events to create new compound Knowledge Services, and
- Providing millions of end users with personal agent tools to do this on an unprecedented scale, all building upon each other to scale exponentially like the Web.

The result would be millions of simple, interacting, Internet-scale knowledge systems, built through mass collaboration, which support how people actually work and live. And because they're man-machine systems, they could utilize any Web resource. Such systems would stand in stark contrast to contemporary AI and Semantic Web applications, which are typically built by a handful of researchers to perform deep reasoning on a tiny sub-Web of formally structured information. Internet-scale knowledge systems, by contrast, are developed pragmatically: Start with the most basic functionality that does something useful. Get it out quickly so others can build on it. Automate complex tasks incrementally. And never let a lack of standards get in the way of connecting things.

The resulting systems are not incompatible with the Semantic Web. Feel free to use RDF if you like – it's just another microformat, one particularly good for deeper reasoning. Indeed, microformats are a great way to bootstrap the Semantic Web and get traction on problems that people actually care about. Some in the Web 2.0 community have begun referring to their microformat-based applications as the *lower case semantic web.* I prefer *Semantic Web 2.0,* which appropriately acknowledges the contributions from both the Semantic Web and Web 2.0 communities.

6.1 But is it AI?

Internet-scale knowledge systems are clearly useful. But are they intelligent? Recall Newell's criteria for intelligent systems (Table 1). Most Web sites exhibit at least a few of these characteristics. Moreover, each of them can be found somewhere on the Web. Newell, however, was adamant that all characteristics must simultaneously be exhibited.

A few mega-sites, such as Amazon, come close to meeting Newell's condition. Amazon has adaptive goal-oriented behavior – it tries to sell you books; It certainly learns from experience – what to recommend. Someone has built a speech interface using their published APIs. And Amazon even has a modicum of self-awareness, – the ability to recognize when servers go down and take corrective action. So is Amazon intelligent?

Begging that question, the real issue is whether a Web of distributed knowledge services, each of which may satisfy at most a few of the criteria, can collectively achieve true intelligence? The jury is still out. However, I'm sanguine the answer will be yes. And because they at least give people the information they need to solve problems and make decisions, such systems will surely boost the collective intelligence of man and machine.

6.2 Call to Action

I'd like to close with a call for action. Building Internet-scale Knowledge Systems takes mass collaboration. I can't do it alone, and neither can you. I invite those of you interested in helping realize the Semantic Web 2.0 vision to visit our wiki at commerce.net/semweb2. There you'll find the abstract and original slides from my IAAI-05 presentation, together with links to the people, projects and companies mentioned. More importantly, there's a discussion forum where you can share with the community your thoughts on the vision, as well as relevant work you've done or know about (software, tools, ontologies…) for realizing it. I've also suggested a few open source, collaborative research projects to create the tools, infrastructure and initial content necessary for jumpstarting mass collaboration. They include:

- Semantic scraping tools for populating microformats from existing Web sites
- Tools for bridging folksonomies and ontologies, and inference tools that exploit the ontologies
- An Internet-scale event bus
- Tools for writing Web-aware "Automator" scripts that assemble existing knowledge services into new ones
- A knowledge-services registry and microformat for describing Knowledge Services in terms of their input, output, and services provided.
- A few useful knowledge services, to serve as building blocks and exemplars, created by "agentifying" existing Web sites.

7. Acknowledgements

I'd like to thank the many colleagues who helped with this presentation – Alain Rappaport, Tantek Çelik, Randy Davis, Dan Gould, Chris Hibbert, Rohit Khare, Greg Olsen, Barney Pell, Raj Reddy, Adam Rifkin, Manoj Saxena, Allan Schiffman, Kragen Sitaker, Jay Weber, and especially Neil Jacobstein who persuaded me that these ideas were worth talking about, and Kevin Hughes whose illustrations and collaboration helped crystallize them.