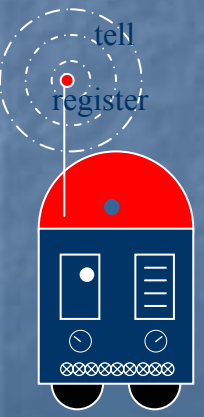
Agents for Handheld and Embedded Devices

Tim Finin
University of Maryland
Baltimore County



Presentation given at the Workshop on Cooperative Information Agents, Modena, Italy, September 6, 2001

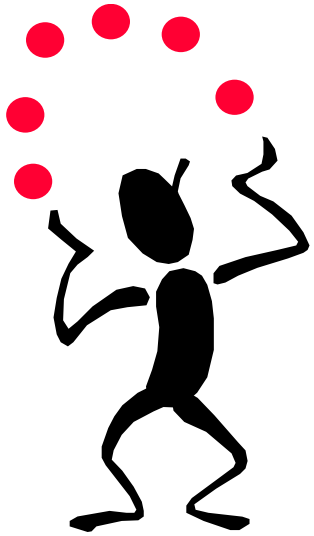
Joint work with Anupam Joshi, Yun Peng, Scott Cost, Yelena Yesha and many students.

This work was partially supported by DARPA contract F30602-97-1-0215 and NSF grants CCR007080 and IIS9875433.

Overview

- Big picture
- Semantic web
- Some current research at UMBC
  - Systems: Centaurus, DReggie, ESDP
  - Infrastructure: Distributed trust
  - Application: Agents2Go
- Comments
- Conclusion

Today: Life is Good.



Tomorrow: We Got Problems!

The Big Picture

- Mobile/pervasive computing and software agents are a good match
- The combination offers new challenges for each
- Attempts are being made to bridge the gap to connect the two
- **Pervasive computing** is the real target and will require an integrated model to support both wired and wireless computing

Mobile and Agents are a Good Match

- The agents community has relatively advanced approaches to many of the problems faced by mobile computing, since we have assumed a very dynamic, ad hoc environment, open environment. Some common issues:
  - Service description, discovery, composition.
  - Negotiation for services and information
  - Authentication, authorization, and trust
  - Delegation and degrees of autonomy
  - Coordination and teamwork models
- Mobile/pervasive computing will provide good justification for an agent oriented approach.

Special challenges for agents

- Today's mobile computing environment offers special challenges for us. Mobile systems have:
  - Low/variable bandwidth, limited CPU, memory, disk, power etc.
    - Resource poor systems connected over thin pipes.
    - "Resource gap" is (mostly) indifferent to absolute values.
  - (Elective) disconnections, dynamically changing network topology …

Special challenges for mobile computing

- Current technologies being used for mobile computing (e.g., Bluetooth) or likely to be adopted (e.g., Jini, UDDI) have problems.
  - The languages for describing and matching services are much too simple.
  - No or poor support for shared ontologies beyond those selected for us by business consortia.
  - No or poor support for evolution and maintenance in such an open environment.
- Envisioned pervasive computing environments must be "context aware".

Current activities and needs

- There are ongoing efforts to bridge the gap
- From the agents side
  - Porting of FIPA platforms (e.g., Jade, FIPA-OS) to handheld devices
  - Enhanced versions of Jini and Bluetooth SDP using RDF and DAML
- From the other side?

Rest of Talk

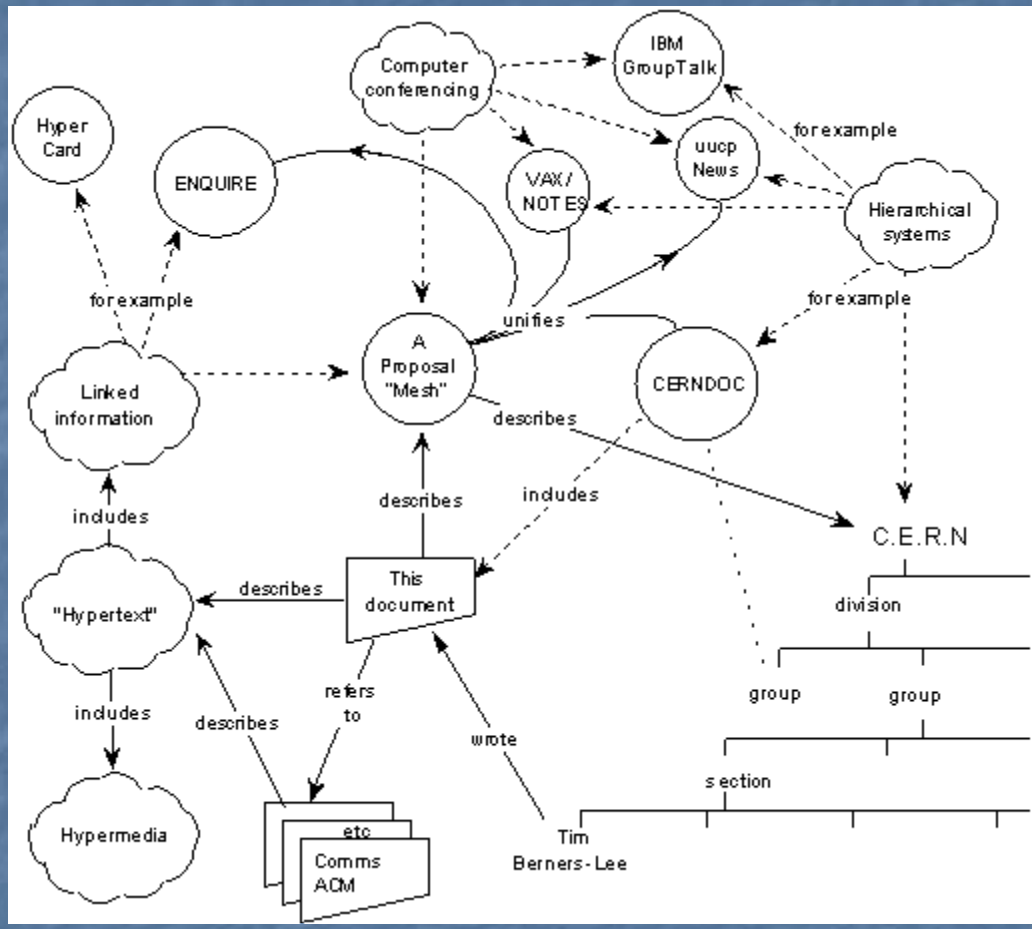
- The semantic web as a potential common model for both wired and mobile information sharing
- Several UMBC ongoing projects addressing issues at different levels: systems, infrastructure and application:
  - Centaurus communication protocol
  - Using semantic web languages for service discovery in Jini and Bluetooth
  - A model for distributed authorization and trust
  - Agents2Go mobile application
- Comments and conclusion

Semantic Web?

- I'll argue that the semantic web provides a good approach, language and tools to support mobile and pervasive computing.
- This isn't obvious, since the SW seems grounded in the "traditional" wired web.
- But, I think the principles which drive it are the right ones for our wireless/mobile/pervasive computing environment as well.
- Next: overview of Semantic Web

Origins of the Semantic Web

- Tim Berners-Lee's original 1989 WWW proposal described a Web of relationships among named objects that unified many information management tasks.
- Guha designed MCF at Apple (~1994)
- XML+MCF=>RDF
- RDF+OO=>RDFS
- RDFS+KR=>DAML+OIL (2000)
- W3C's SW activity (2001)



http://www.w3.org/History/1989/proposal.html

W3C's Semantic Web Goals

- Realizing the full potential of the Web
- Making it cost-effective for people to effectively record their knowledge
- Focus on machine consumption.
  - "The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation." -- Berners-Lee, Hendler and Lassila, The Semantic Web
  - "The bane of my existence is doing things that I know the computer could do for me." -- Dan Connolly, The XML Revolution
- Ultimate goal - effective and efficient global knowledge exchange

Semantic Web Principles

- Everything is on the web
- Partial information
- Web of trust
- Support information evolution
- Minimalist design
- Common data model

Adapted from Eric Miller, W3C

Principle: Everything is on the Web

People, places, and things in the physical world will have online representations identified by Uniform Resource Identifiers (URIs) which will facilitate effective integration, active participation and be contextualized in the Semantic Web (SW).

Adapted from Eric Miller, W3C

Principle: Partial Information

The Web is unbounded.

Its design differed from traditional hypertext systems in sacrificing link integrity for scalability.

In the SW there should be no constraint on what is said, what it is said about, and where it is said.

Anyone can say anything about anything, and there will always me more to learn.

Adapted from Eric Miller, W3C

Principle: Web of Trust

All statements found on the SW occur in some context and applications need this context in order to determine the trustworthiness of the statements

The SW does not assert that all statements found on the Web are "true".

Truth - or more pragmatically, trustworthiness - is evaluated by, and in the context of, each application that processes the information found on the Web.

Adapted from Eric Miller, W3C

Principle: Evolution

The SW must permit distributed communities to work independently, adding new information without insisting that the old be modified.

This supports the resolution of ambiguities and inconsistencies while taking advantage of the wealth of backgrounds and abilities.

The SW must expand as our understanding expands and be able to capture information linking independent representations of overlapping areas of knowledge.

Adapted from Eric Miller, W3C

Principle: Minimalist Design

Make the simple things simple, and the complex things possible.

Standardize no more than is necessary.

Adapted from Eric Miller, W3C

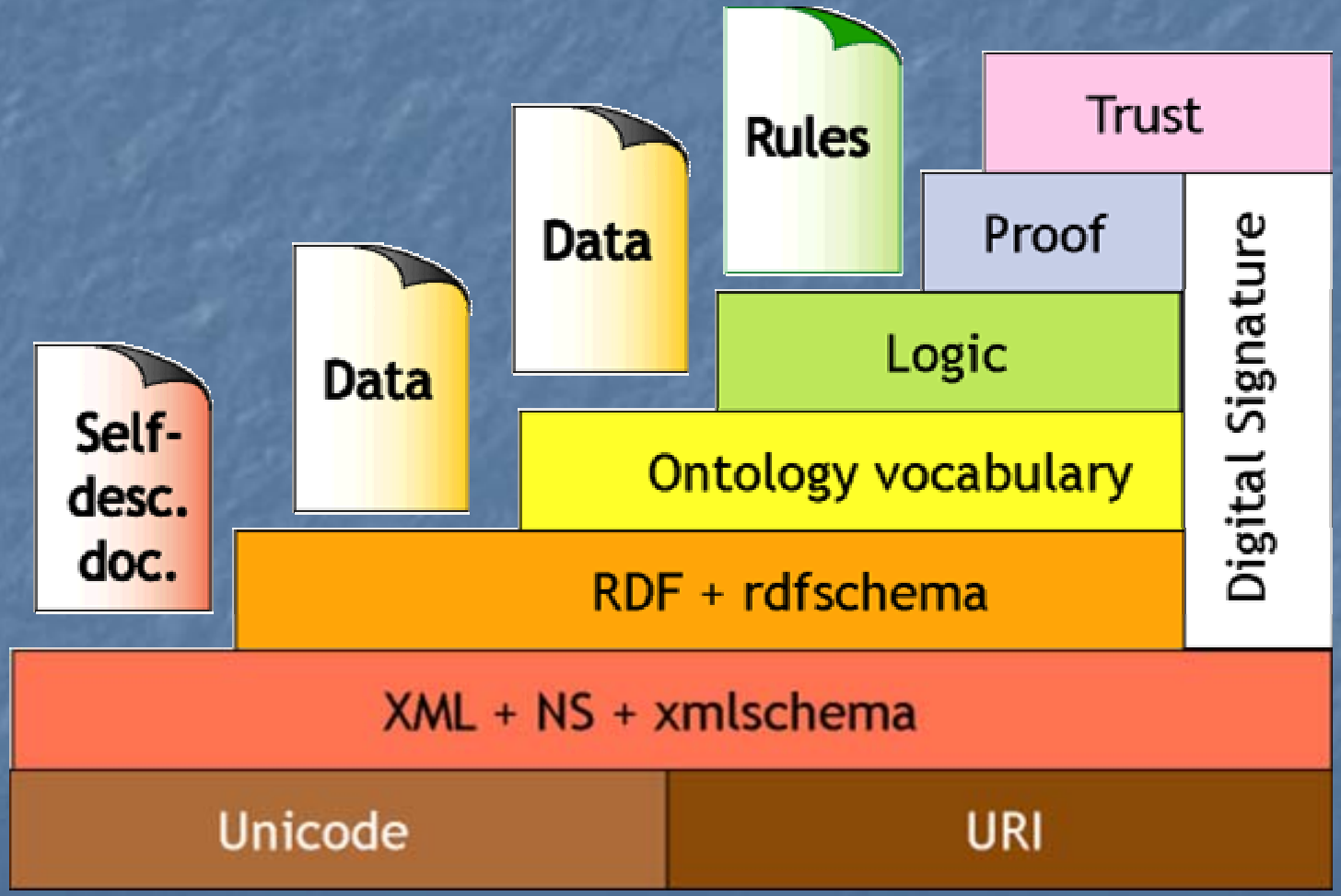
Principle: Common Models

To encompass the universe of network-accessible information, the SW must provide a way of exposing information from different systems.

We need a very general data model which is fairly low level to allow individual application and communities to map their own representations to it.

Adapted from Eric Miller, W3C

Tbl's semantic web vision



DAML+OIL

- DAML = Darpa Agent Markup Language
  - DARPA program with 17 project teams and an integrator developing language spec, tools, applications for semantic web.
- OIL = Ontology Inference Layer
  - An EU effort aimed at developing a layered approach to representing knowledge on the web.
- Process
  - Joint Committee: US DAML and EU Semantic Web Technologies participants
  - DAML+OIL spec released 01/01 and revised in 03/01
  - Technical discussions take place on the www-rdf-logic@w3.org email list

DAML in One Slide

DAML is built on top of XML and RDF

It allows the definition, sharing, composition and use of ontologies

DAML is ~= a frame based knowledge representation language

It can be used to add metadata about anything which has a URI.

URIs are a W3C standard generalizing URLs

everything has URI

```
<rdf:RDF xmlns:rdf ="http://w3.org/22-rdf-syntax-ns#"
   xmlns:rdfs="http://w3.org/rdf-schema#"
   xmlns:daml="http://daml.org/daml+oil#“>
<daml:Ontology rdf:about="">
    <daml:imports rdf:resource="http://daml.org/daml+oil"/>
</daml:Ontology>
<rdfs:Class rdf:ID="Person">
 <rdfs:subClassOf rdf:resource="#Animal"/>
 <rdfs:subClassOf>
  <daml:Restriction>
   <daml:onProperty rdf:resource="#hasParent"/>
   <daml:toClass rdf:resource="#Person"/>
  </daml:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <daml:Restriction daml:cardinality="1">
   <daml:onProperty rdf:resource="#hasFather"/>
  </daml:Restriction>  </rdfs:subClassOf> </rdfs:Class>
<Person rdf:about=“http://umbc.edu/~finin/">
<rdfs:comment>Finin is a person.</rdfs:comment>
</Person>
```

Why RDF Is Not Enough

- Expressive inadequacy
  - Only range/domain constraints (on properties)
  - No properties of properties (unique, transitive, inverse etc.)
  - No equivalence, disjointness, coverings etc.
  - No necessary and sufficient conditions (for class membership)
- Poorly (un) defined semantics

A Simple DAML Example

```
<rdfs:Class about="#Animal"/>
<rdfs:Class about="#Plant">
        <daml:disjointFrom
                resource="#Animal"/>
</rdfs:Class>
```

We're going down a familiar road

KR trends

- 55-65: arbitrary data structures
- 65-75: semantic networks
- 75-85: simple frame systems
- 85-95: description logics
- 95-??: logic

Web trends

- 95-97: XML as arbitrary structures
- 97-98: RDF
- 98-99: RDF schema as a frame-like system
- 00-01: DAML+OIL
- 02-??: DAML-L

Only much faster!

Some UMBC Work

I'll briefly describe several ongoing projects involving mobile/pervasive computing at UMBC.

(1) Centaurus communication infrastructure

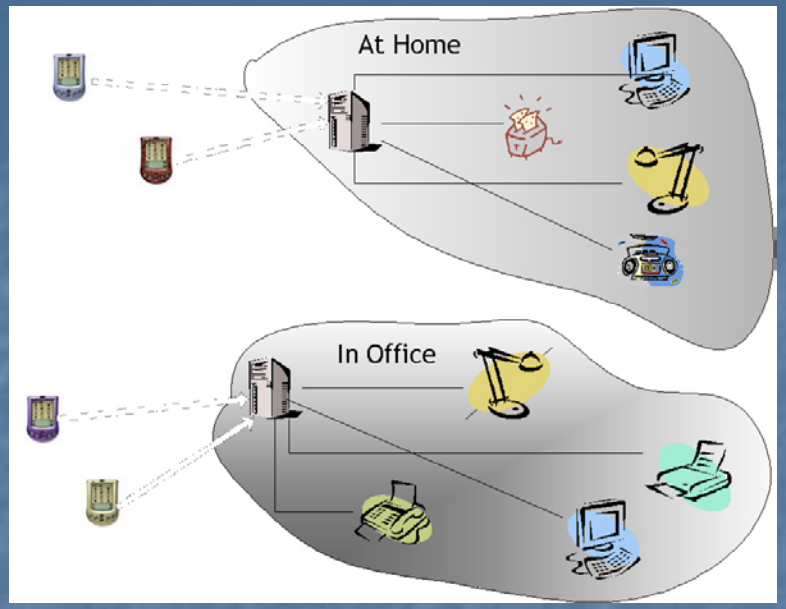
(2) Enhancing Jini with DAML for service description and discovery

(3) Enhancing Bluetooth's SDP with DAML

(4) A model of distributed authorization and trust

(5) A simple mobile application

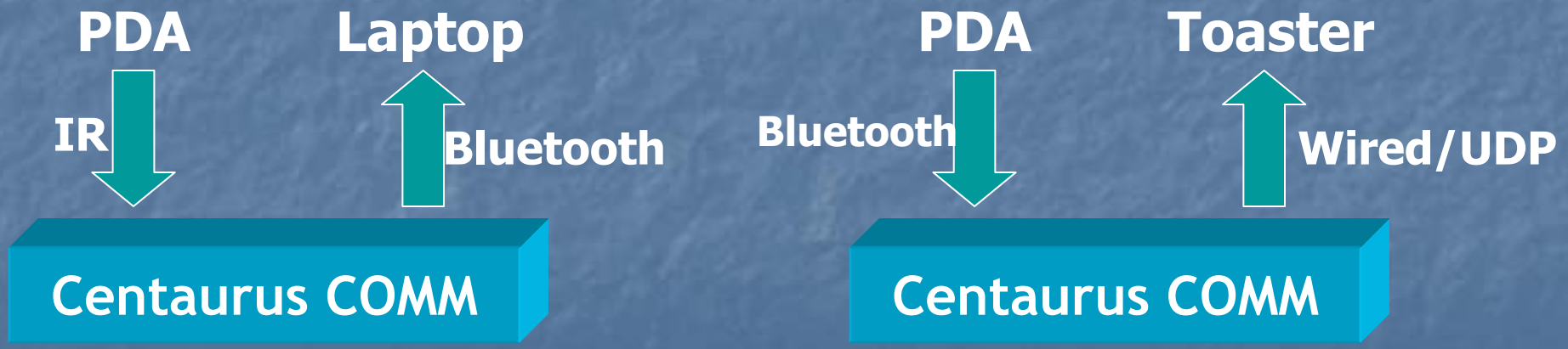
(1) Centaurus



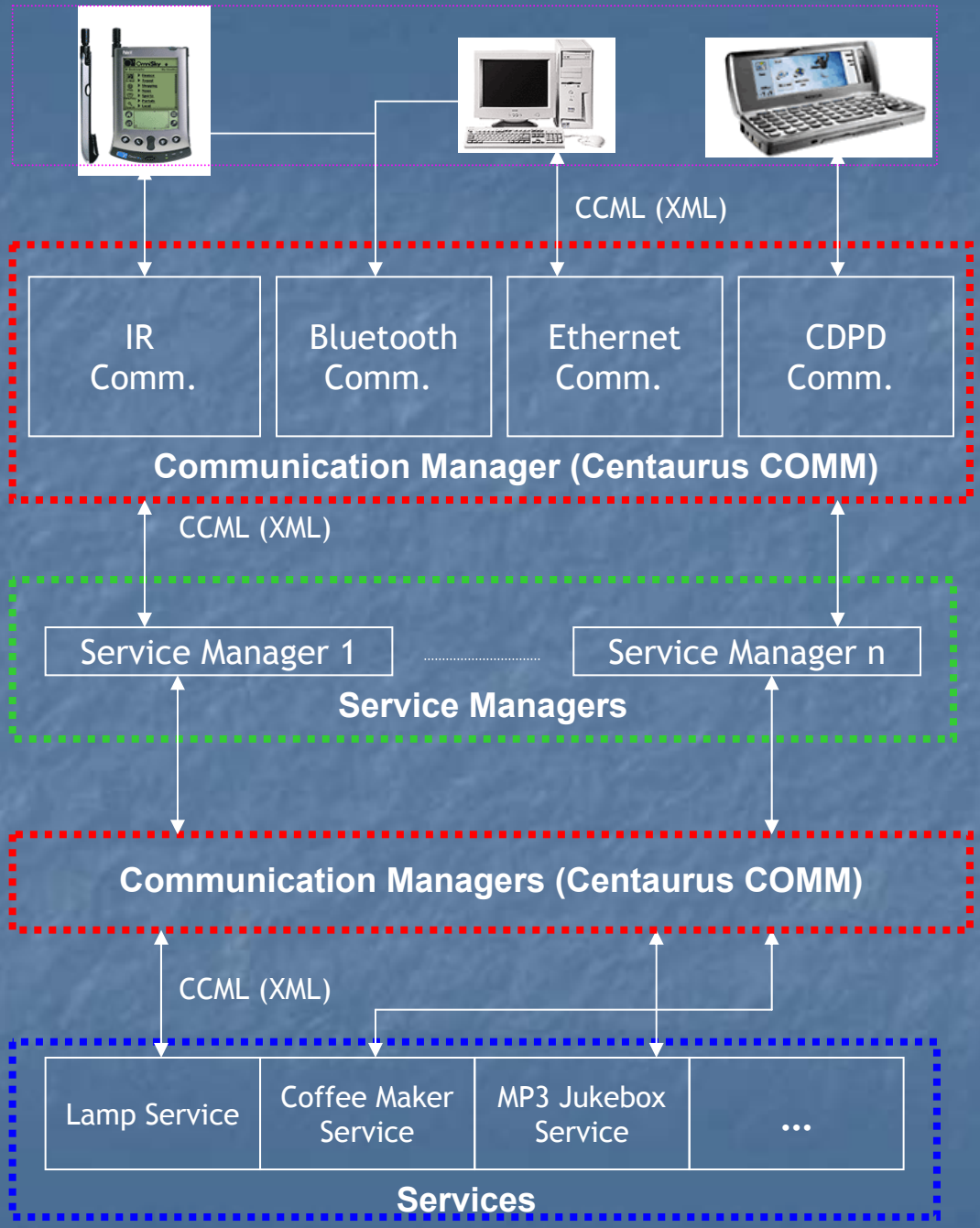
- Centaurus is a framework for developing and delivering heterogeneous services in a mobile environment
- Computers and devices are facing interpretability problems.
  - Devices want to talk to each other; printers, lamps, toasters etc.
- The computing platforms are less likely to be uniform.
  - Palm OS, Windows CE, Cell phones, Linux, Windows, etc.
- The communication mediums between devices are less likely to be uniform.
  - GSM, CDPD, Infrared, Bluetooth, Wired cables, 802.11b

Centaurus Communication

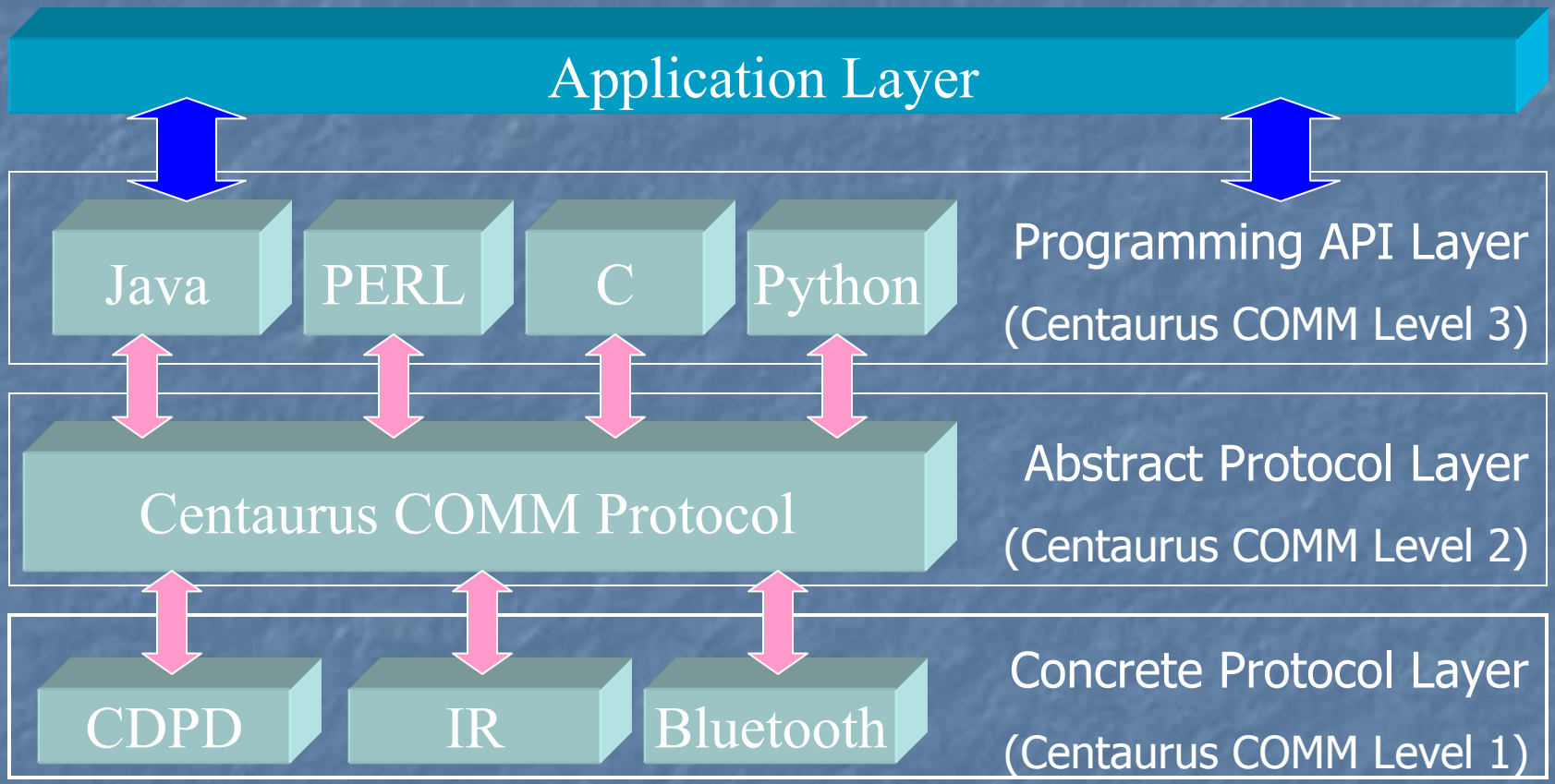
Centaurus Communication (Centaurus COMM) provides a message passing network architecture that allows heterogeneous devices to communicate through varied communication mediums in a uniform fashion



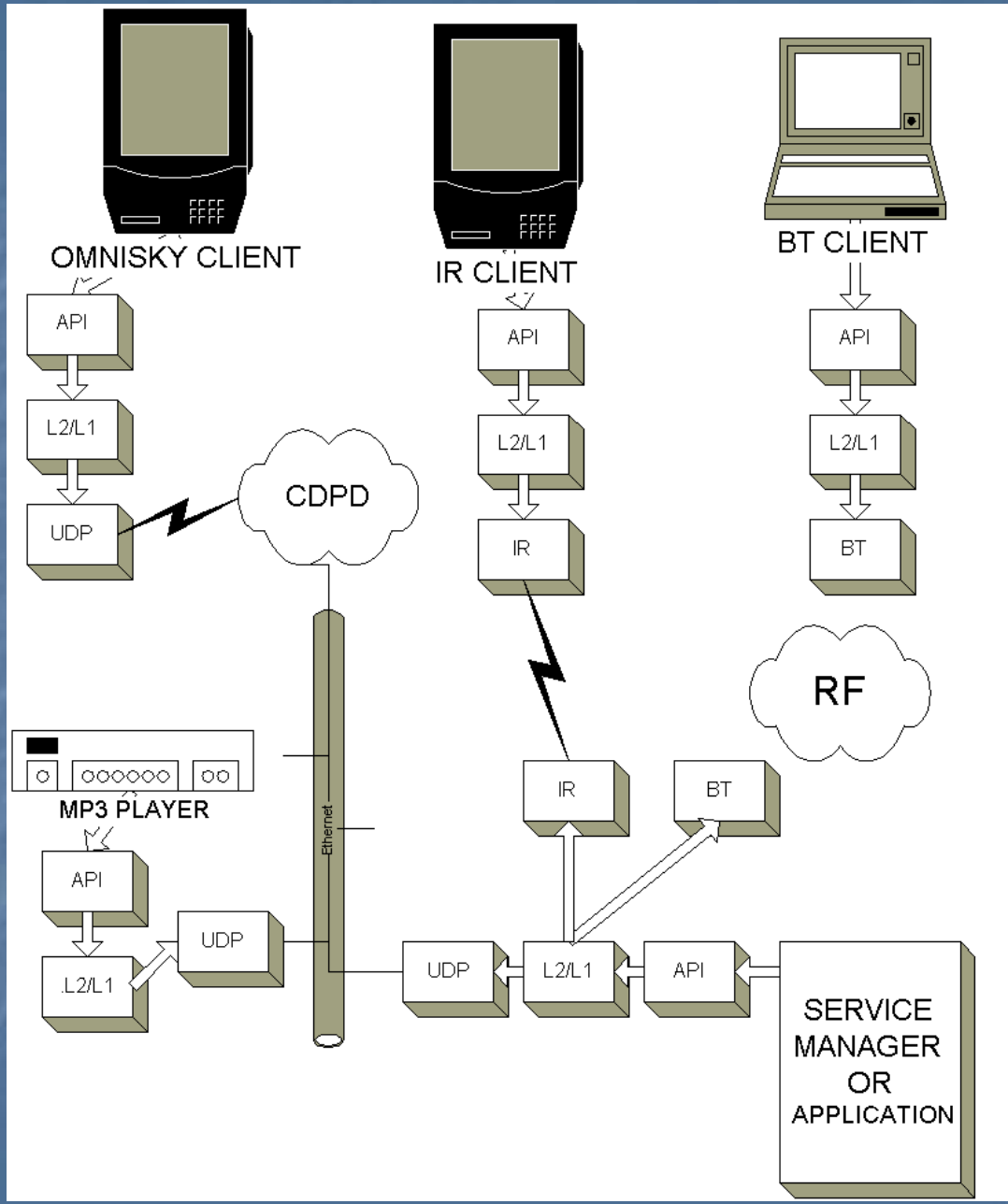
The Centaurus Architecture



Centaurus Communication



An Example


OMNISKY CLIENT
IR CLIENT
BT CLIENT
API
L2/L1
UDP
CDPD
IR
BT
RF
MP3 PLAYER
Ethernet
SERVICE MANAGER OR APPLICATION

Future Work

Integration with advanced features:

- Use of DAML descriptions of services offered and sought
- Distributed delegation and trust model

(2) Enhancing Jini's registration server

- Jini is a very attractive collection of ideas and components.
- One deficiency is the Jini registration server's inexpressive approach to describing services offered and sought.
- We've produced a modified Jini registration server which allows agents to use DAML+OIL to describe services offered or sought

DReggie: A Smart Lookup Service

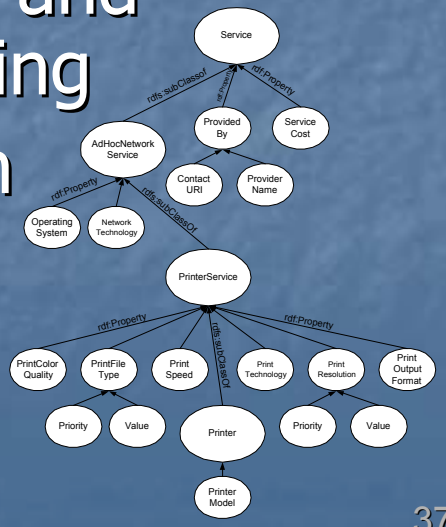
- Modify Jini's registry and lookup mechanisms.
  - Each service registers a DAML description of capabilities and requirements and invocation technique.
  - Each lookup request carries a DAML description.
  - Match can be "fuzzy."
    - Handles constraints.
    - Returns similar matches.

(3) Enhancing Bluetooth's SDP

- Bluetooth is a short-range RF wireless technology that supports ad-hoc networks and uses P2P protocols.
- Bluetooth Service Discovery Protocol:
  - Simple service discovery mechanism
  - Services and attributes represented by UUIDs
  - UUID-based matching
  - No registration, aggregation, multicasting, event notification
- Not very expressive!

Prototyped Solution

- Assume Bluetooth ad-hoc networks with at least one resource rich device (e.g., each room has a facilitator).
- Enhanced SDP
  - Services and attributes described in DAML using a "standard" ontology
  - All available information from service and attribute descriptions used for matching
  - Tries to obtain *closest* possible match
  - Support service registration facility



(4) Delegation Based Model for Distributed Trust

- We are developing a delegation based model for distributed authorization and trust for use in both wired and wireless scenarios.
- Focus on trust from a "security perspective"
- Building on concepts like authentication, authorization, role-based access control, public key infrastructure, digital signatures, authoritative sources of information, etc.
- Agents make speech acts about and reason over these properties and relations.
- Grounded in an ontology represented in DAML

What is Distributed Trust

- Issues
  - No central authority
  - *logging in* is not possible
  - Access control for entities never encountered before

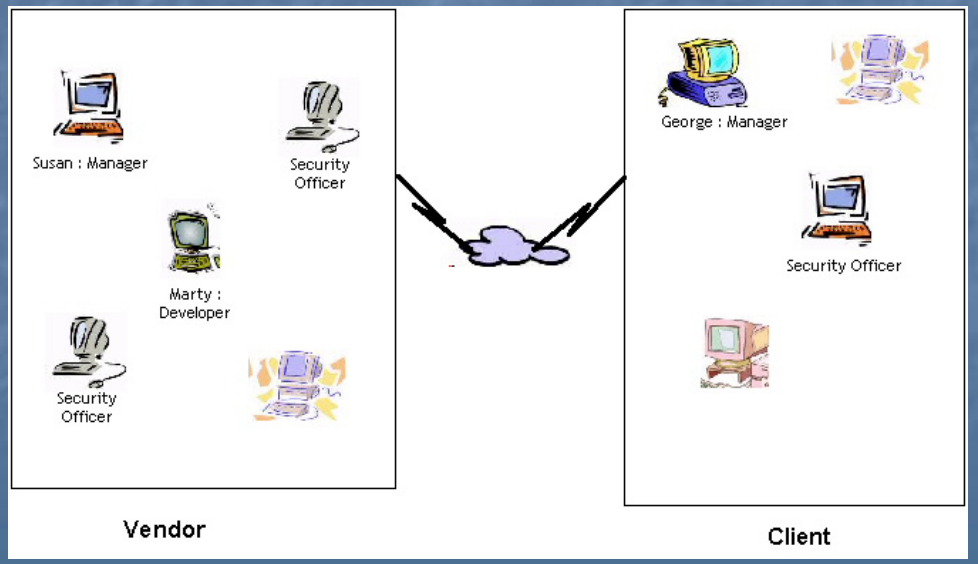
- We use *Distributed Trust* to solve these issues
- trust = policies + credentials + delegation actions + proofs of deontic properties

Three Scenarios

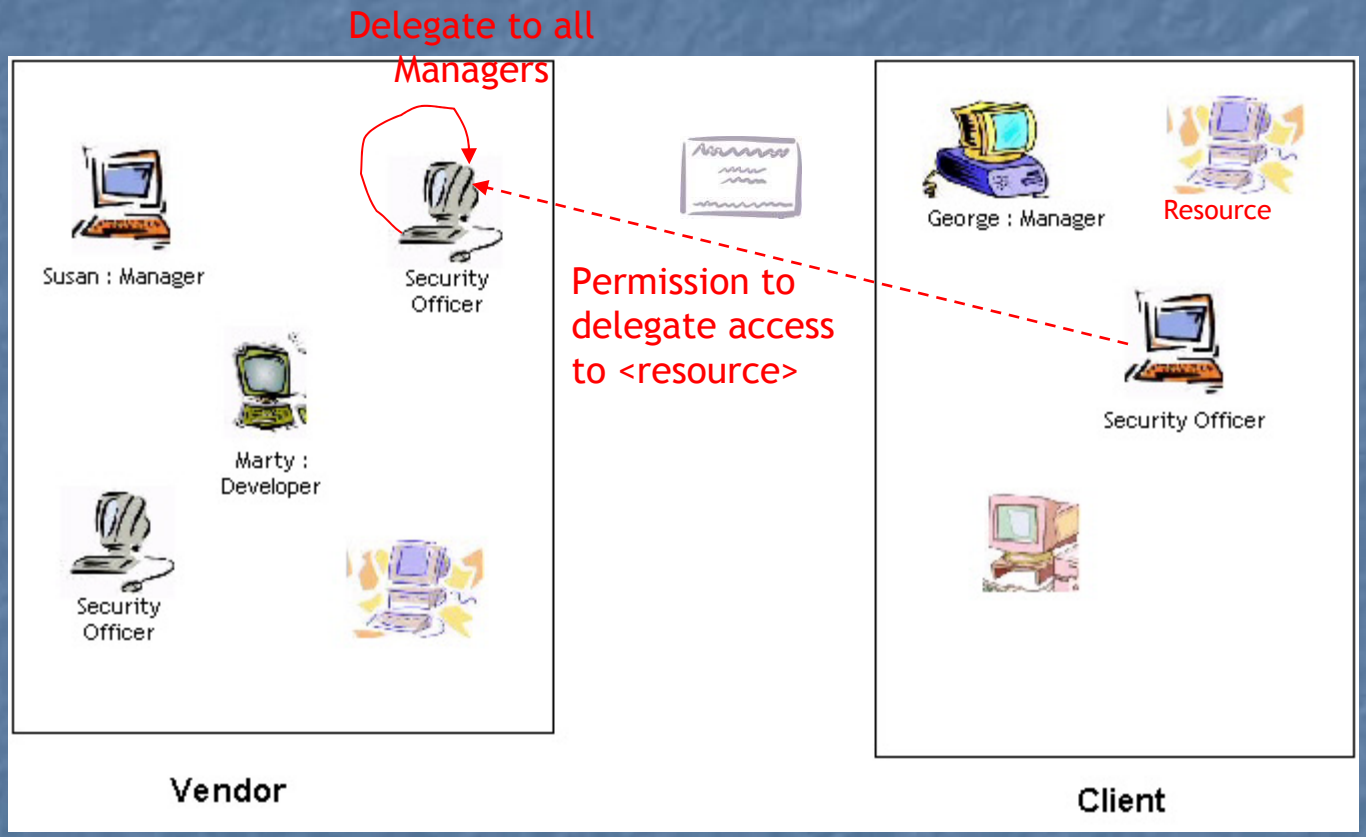
- Supply Chain Management System
  - ✓ Already implemented
- Dynamic Wireless Environment
  - ✓ Ongoing work
- Distributed Trust for Web Services
  - ✓ Future work
  - ✓ To be applied to ITTALKS (http://www.ittalks.org/)

Design for SCM

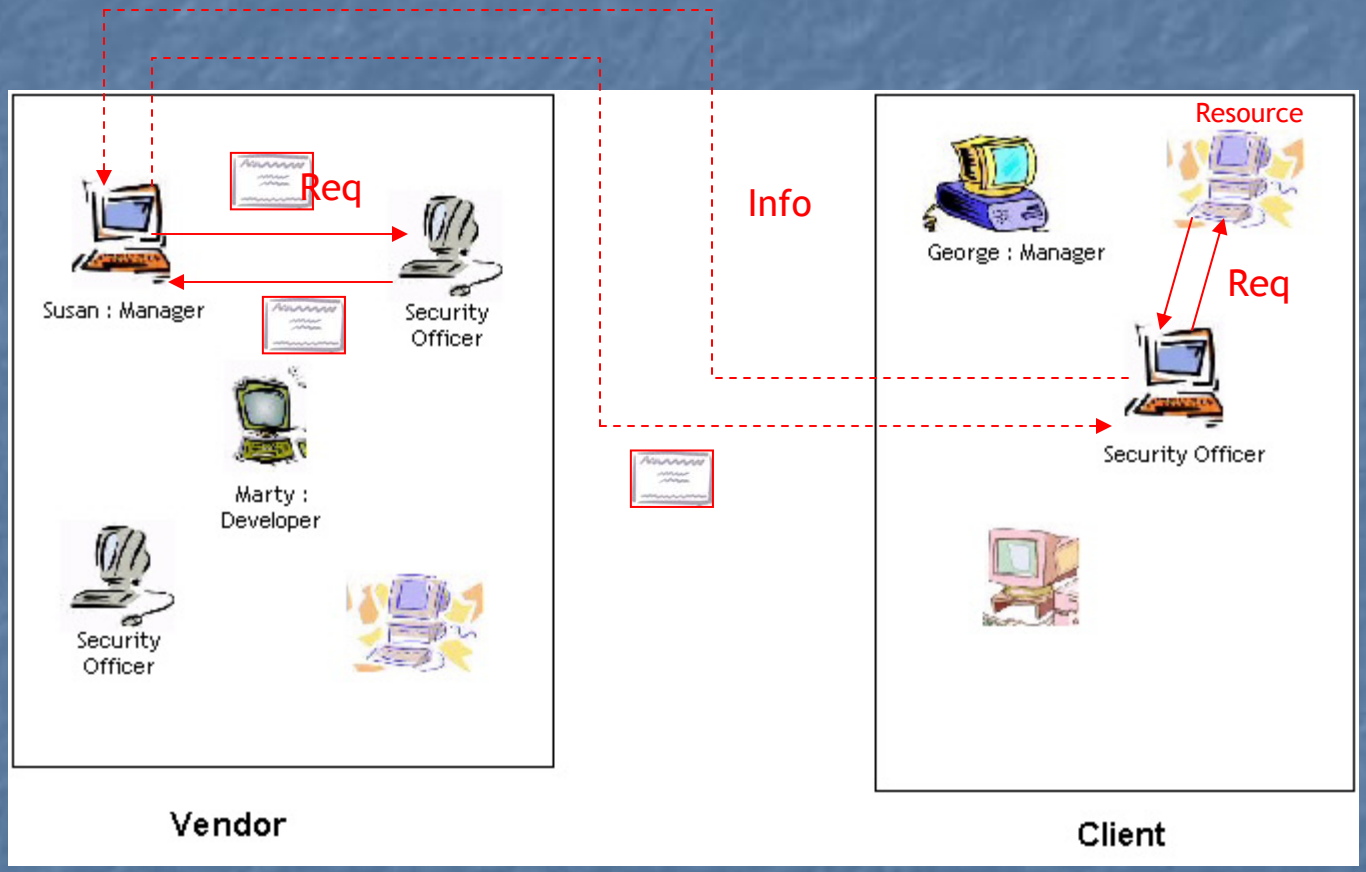
- Companies have security policies
- Policy enforced by a number of 'security officers'
- Each agent in the system has an ID certificate, X.509
- All communication via signed messages
- Trust and policy info encoded as horn clauses



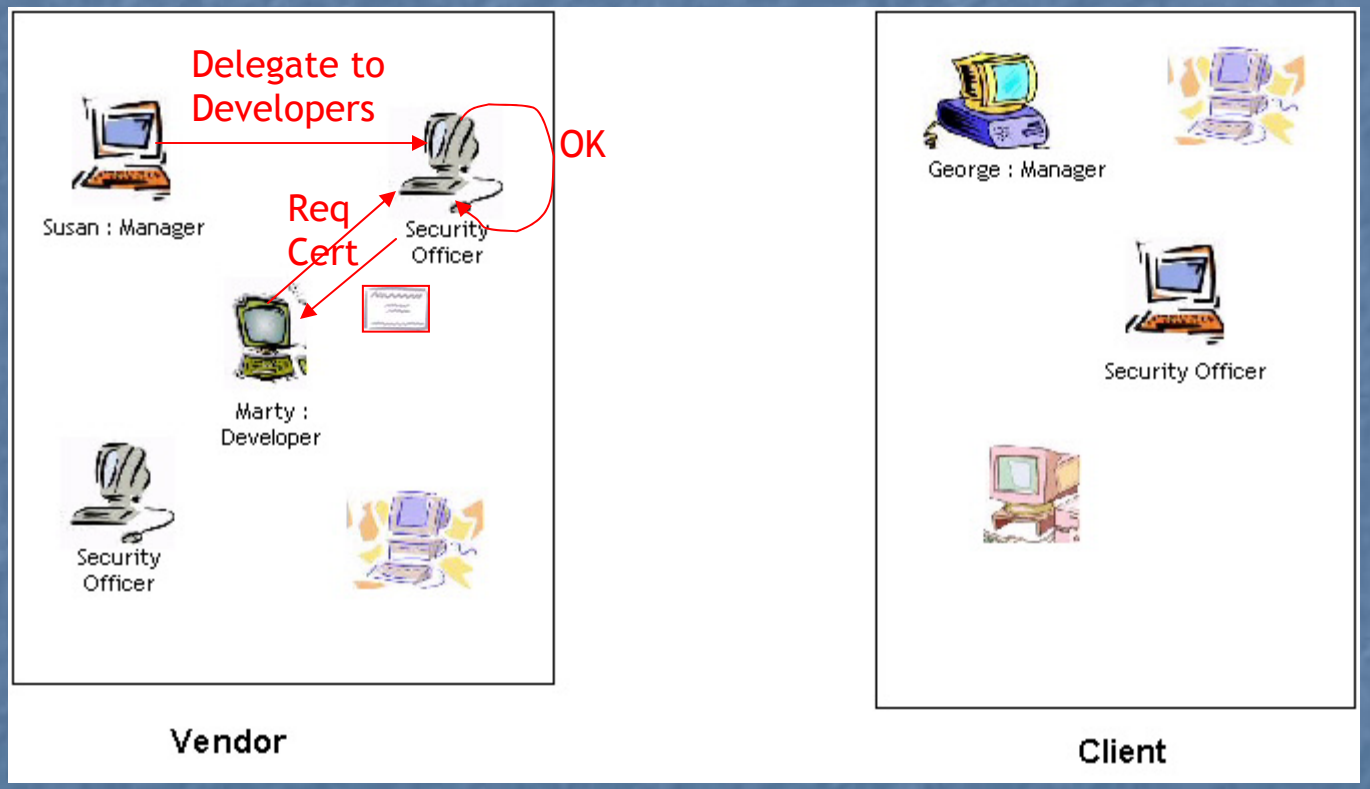
How it works : Initialization



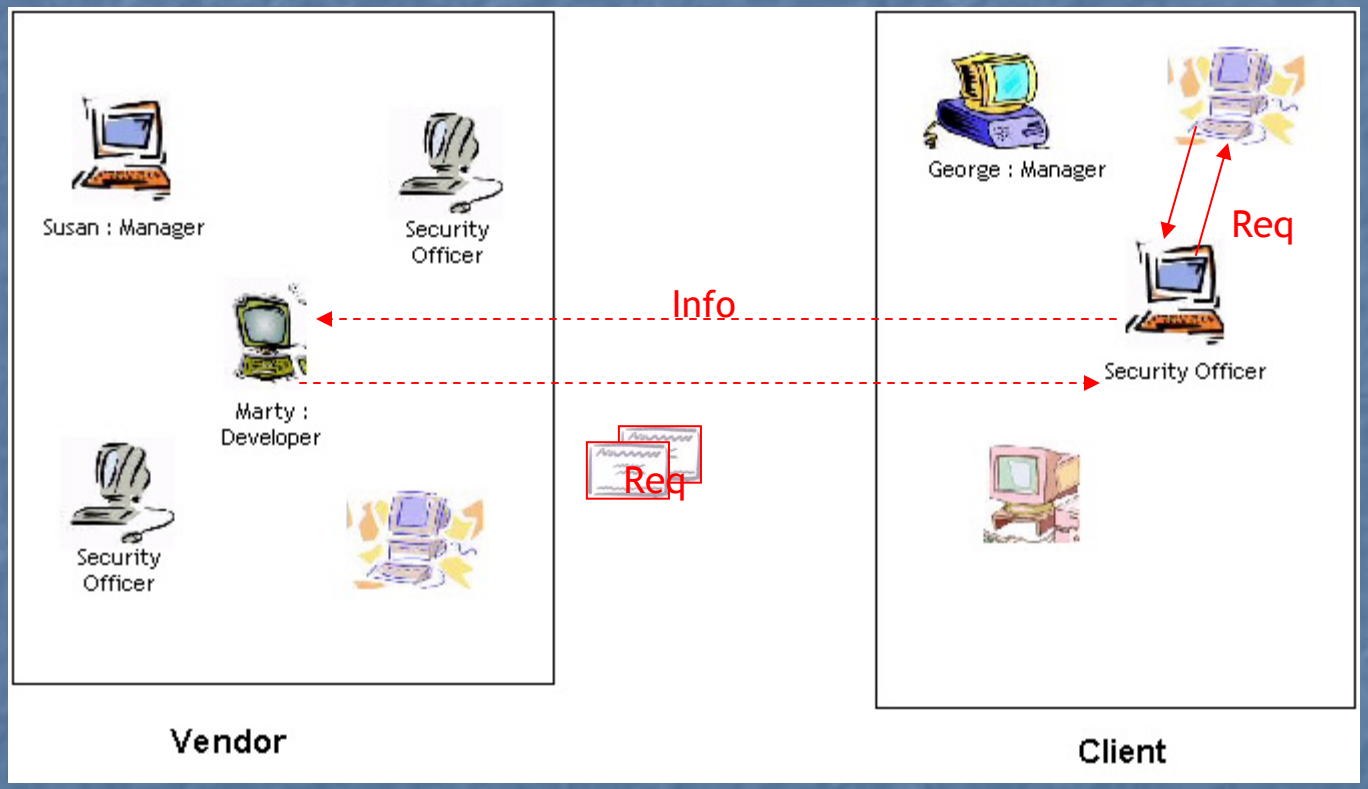
How it works : Request



How it works : Delegation



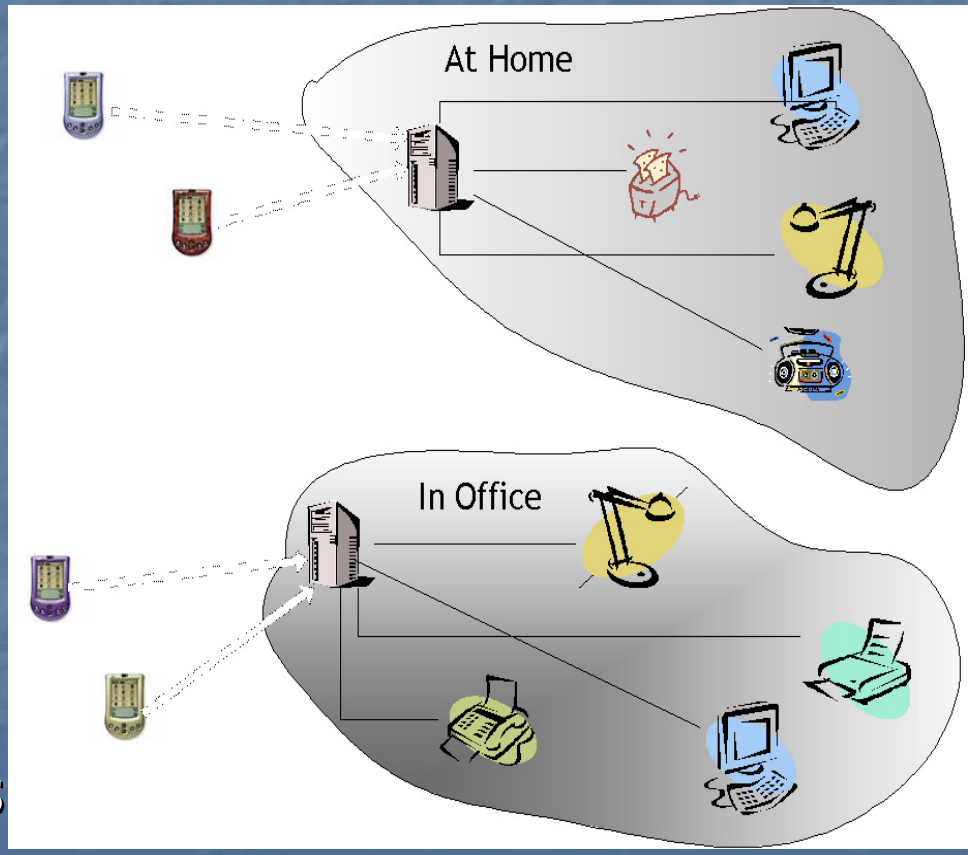
How it works : Request



Scenario 2 : Dynamic Wireless Environment

Working with dynamic, ad hoc wireless environments like Bluetooth

- Unknown entities are involved
- Wireless devices are resource poor
- Authenticate other wireless devices
- Need to communicate and sometimes use other devices



Ongoing Work

- Specifying ontology for permissions, obligations, entitlements, prohibitions in DAML/RDF
- Also model distributed belief
- Encoded in DAML and/or RDF
- Delegating of permissions, obligations, entitlements, prohibitions and belief
- To avoid the permission revocation problem we use "short lived propositions", e.g.

  "My proof that agent xyzzy has permission to do action X is good until time t."

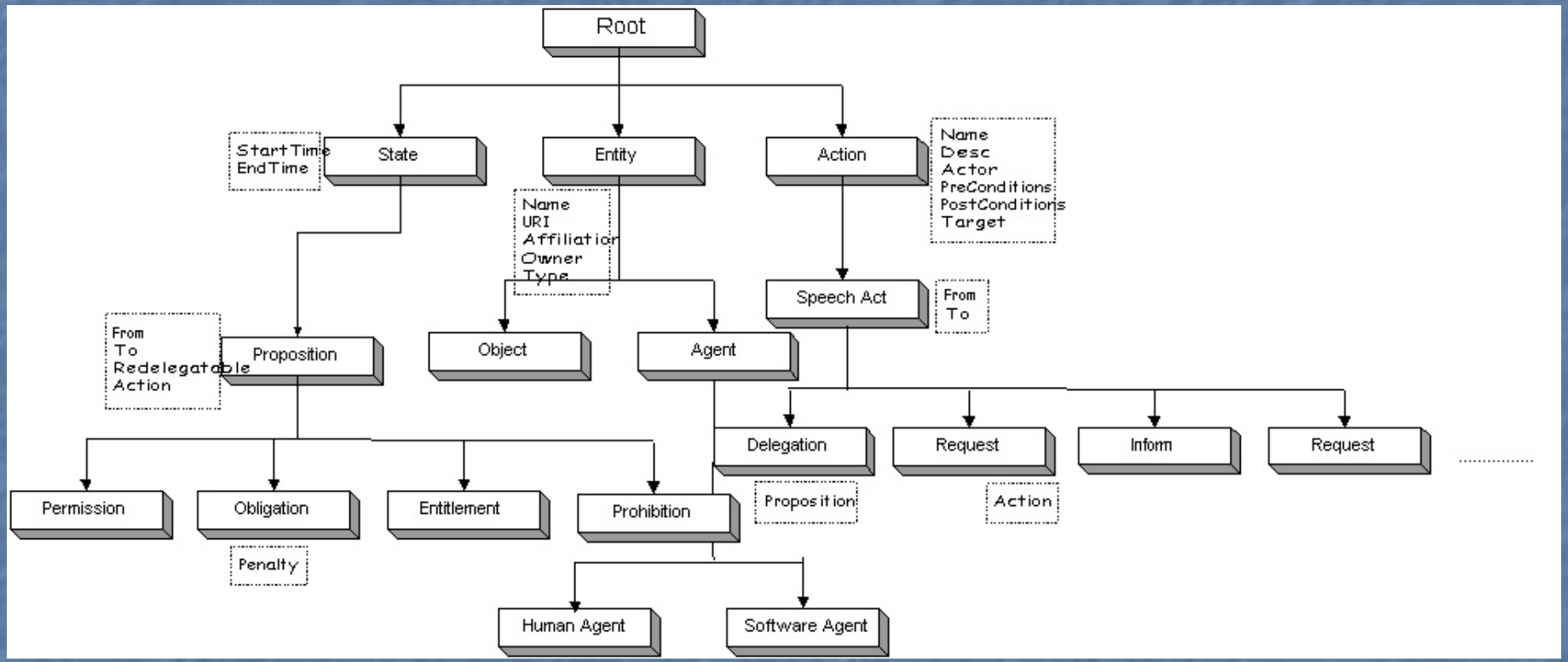
Distributed Belief

- A policy specified that "UMBC CSEE faculty are allowed to do X", but how do we determine who they are?
- Our dtrust language allows us to say

  "We accept http://www.csee.umbc.edu/faculty.html as a trusted source of information about membership in the class http://umbc.edu/ontologies/people#faculty"
- faculty.html has a human-readable faculty list (in HTML) and (possibly signed) statements (in DAML) asserting who the faculty are.
- Beliefs can be delegated as well

  "I delegate belief of phdAdvisee(X,Y) to X if X is a CSEE faculty member"

Dtrust Ontology



A DAML ontology for describing authorization and trust actions, states and policies.

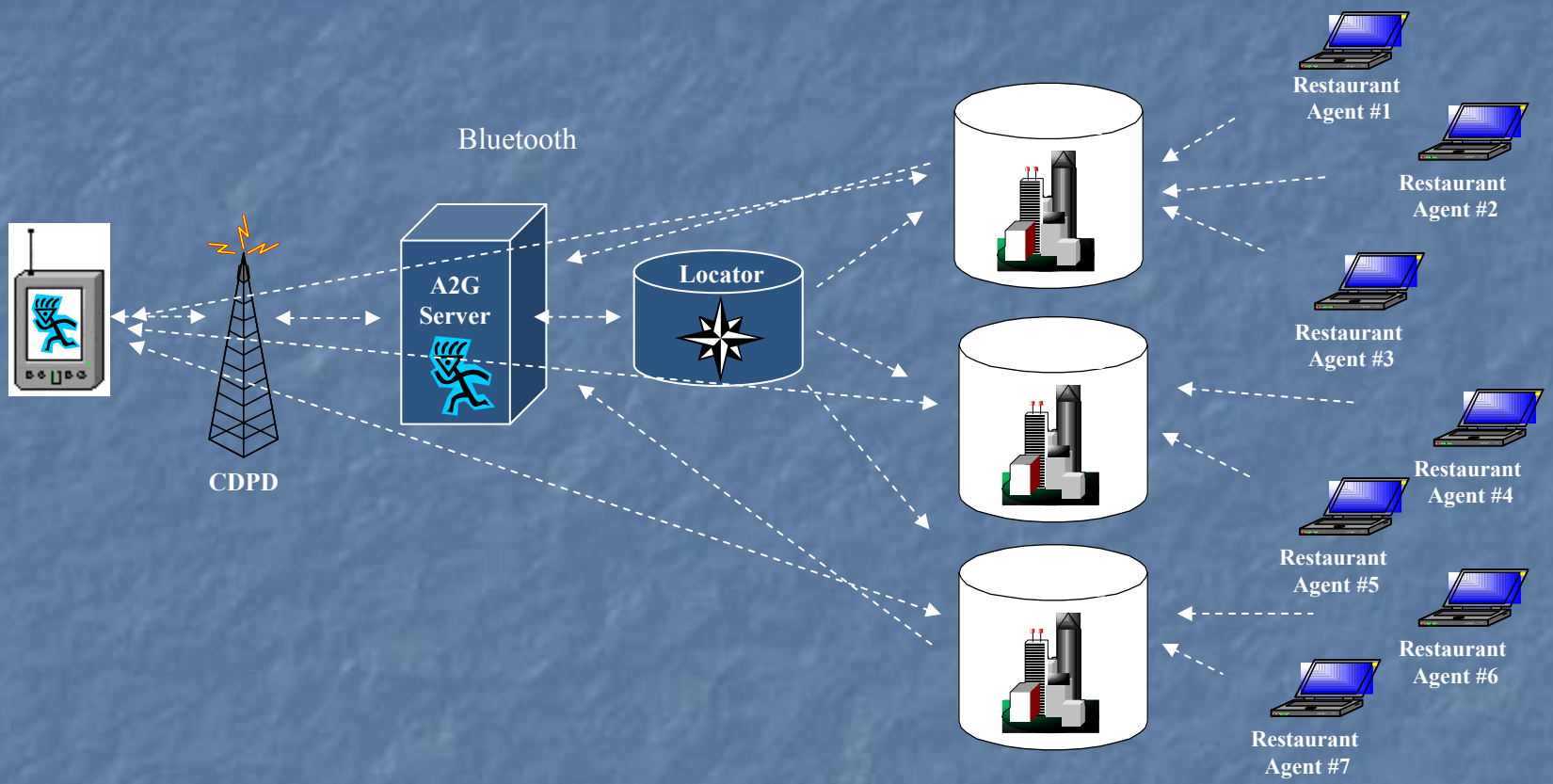
Future Work

- Use XML Signature to sign DAML statements
- Incorporate a reputation mechanism to provide sanctions for failing to follow obligations
- Detect conflicting policies
- Develop a *dtrust* language for web services

(5) The Agents2Go Platform

- Location dependent services discovery
  - Location dependent information retrieval
  - The search results contain information about restaurants that are local to the requesting user.
- Distributed services
  - Distributed Information
  - Service information is distributed and grouped by regions.
  - Information about the restaurant is stored locally.
- Automatic location detection
  - Cell tower ids are mapped to the geographical region name.
- Service provider representation
  - Service Agents reside at the service provider locations.
  - Restaurant Agents reside at the restaurant locations.

The Agents2Go Infrastructure



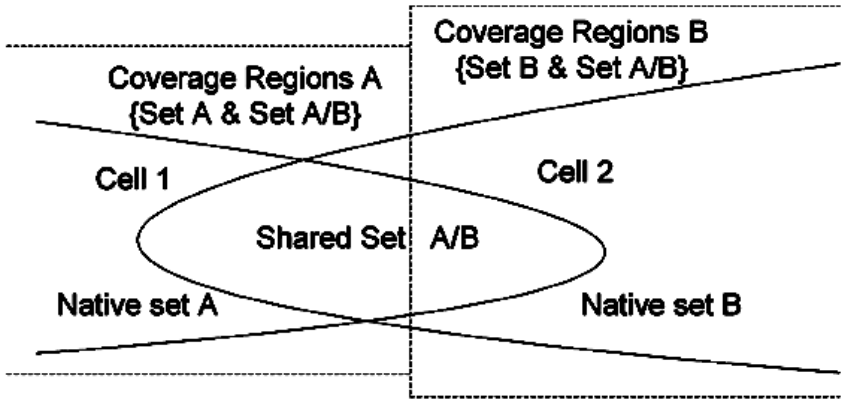
The PalmApp

- The *PalmApp* is a generic form visualizer, independent of the system functionality.
- XML dynamically rendered by the *PalmApp*
- The *PalmApp* provides GUI for submitting requests and seeing responses.

Some Screen Shots

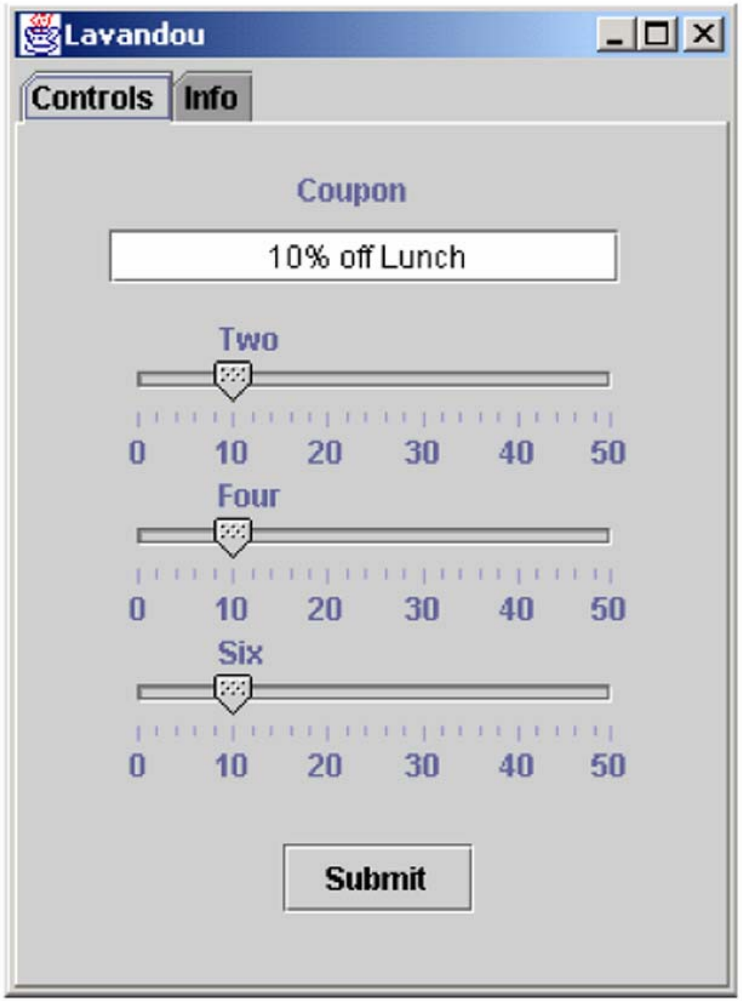
The A2G Server and Locator

- The *A2G Server* receives user requests from a *PalmApp*, maps its cell tower id to the geographical region and forwards this request including region name to the *Locator*.
- The *Locator* maps *Brokers* to regions. It delegates requests from the *A2G Server* to the designated *Brokers*.
- The *A2G Server* and the *Locator* could be replaced when using Bluetooth technology
- Collection of cell tower ids.
- Cell Overlaps



The Restaurant Agent

- The *Restaurant Agent* provides graphical interface to a restaurant host.
- The *Restaurant Agent* is used to send dynamic information lik updates and promotion to the *Broker* that manages that restaurant's geographical regio



Agents2Go Future Work

- The Agents2Go System dynamic, location aware, distributed system.
- The *Broker* holds auctions to find the best deals for the user.
- The *Broker* maintains reputation information about service providers.
- The *Broker* forwards a request that yields no matches to its neighboring *Brokers*.
- The Agents2Go System anticipates the future geographical location of the user.
- Disconnected operations or operations in regions with poor wireless connectivity.

Today's Conclusions

- Different mobile environments
- Rethinking agent communication
- Final thoughts

Mobile computing environments

- Today there are several kinds of wireless technologies of interest:
  - Cellular telephone systems
  - Local peer-to-peer RF networking technologies like Bluetooth
  - Wireless LANs like 802.11
  - Local sensing technologies like RFID
- Each of these offers advantages and challenges.
- The pervasive computing environments of the future will be a combination of (some of) these.

Current coordination infrastructure

- There are many current systems for service registration and coordination
  - UDDI at the internet level
  - Jini at a more local level
  - Bluetooth SDP
- All are characterized by their relatively inexpressive languages for describing services offered and sought.
- This is where the agents/AI/KR community has something to offer.

Rethinking the agent communication paradigm

- Much multi-agent systems work is grounded in Agent Communication Languages (e.g., KQML, FIPA) and associated software infrastructure.
- This paradigm was articulated ~1990, about the same time as the WWW was developed.
- Our MAS approach has not yet left the laboratory yet the Web has changed the world.
- Maybe we should try something different?

Rethinking the agent communication paradigm

- The communication MAS paradigm has been peer-to-peer message oriented communication mediated by brokers and facilitators.
- This approach was, I think, inherited from the dominant software paradigms at the time: client-server and OO systems.
- The semantic web invites different paradigms which will require some changes in ACLs and their associates software systems.

Rethinking the agent communication paradigm

- New paradigm?
  - Agents "publish" beliefs, requests, and other "speech acts" on web pages.
  - Brokers "search" for and "index" published content
  - Agents "discover" what peers have published on the web and browse for more details
  - Agents "speak for" content on web pages by
    - Answering queries about them
    - Accepting comments and assertions about them

Waiting for Moore?

- Agent platforms like JADE and FIPA-OS have been ported to run on PDAs and phones
- The hardware platforms are just barely up to it.
- … and if we want to make the handheld agents intelligent (e.g., adding a KB and reasoner) then things get ugly fast.
- We may have to wait for Moore's law to do a few more iterations
- …or explore alternative architectures which distribute the intelligence over proxies on larger devices.

Context aware computing

- An exciting general view of the new mobile/pervasive computing environment goes under the name of "context aware computing".
- This inherits from work in intelligent HCI
- The computing devices in our environment are aware of each other and also of the people and things in their vicinity.
- Awareness of people entails inferring their internal states and individual and joint activities

Context aware computing

- This is a promising area which can draw on lots of the things we know:
  - Interpreting sensor inputs
  - Sensor and data fusion
  - Abductive reasoning and belief revision
  - Machine learning
  - Plan recognition
  - User modeling
  - Using shared ontologies
  - Models of coordination and teamwork

Final thoughts

- Agents and mobile computing may be a good marriage.
- As usual, only time will tell and all will be obvious in hindsight.
- See http://research.ebiquity.org/ for more information and papers on this work.