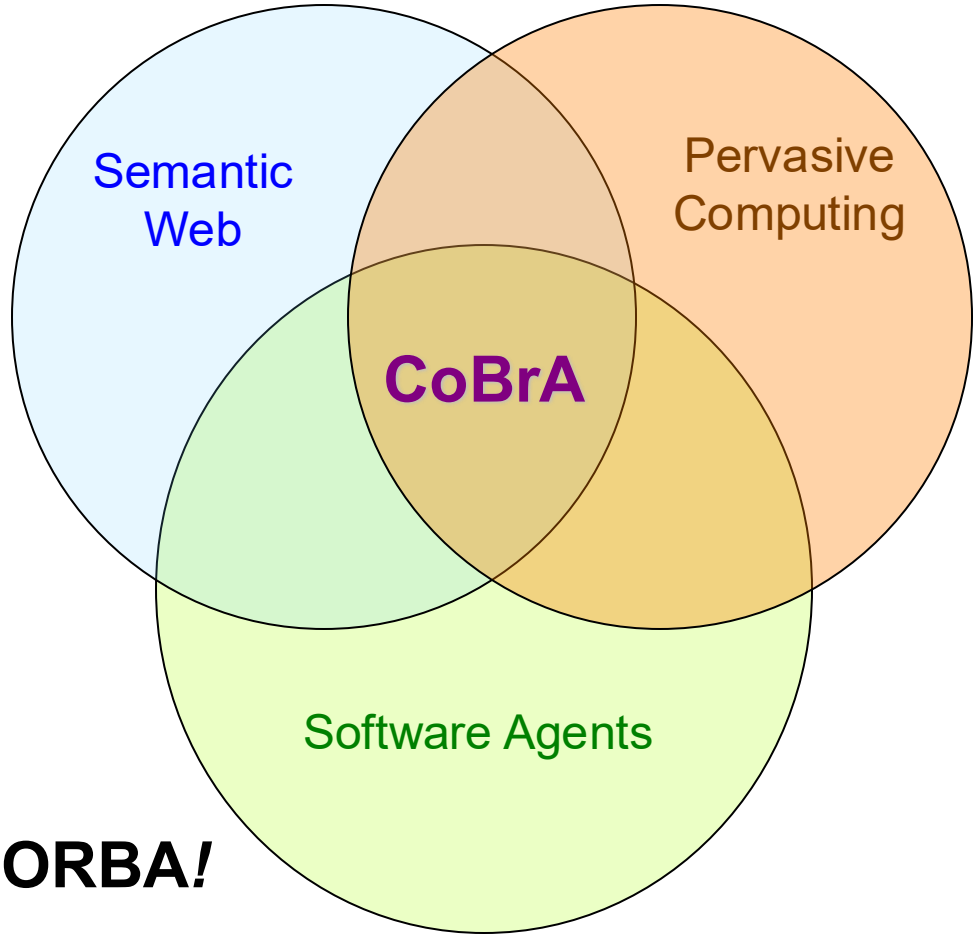
Semantic Web in a Pervasive Context-Aware Architecture

Harry Chen

U of Maryland Baltimore County

Context Broker Architecture

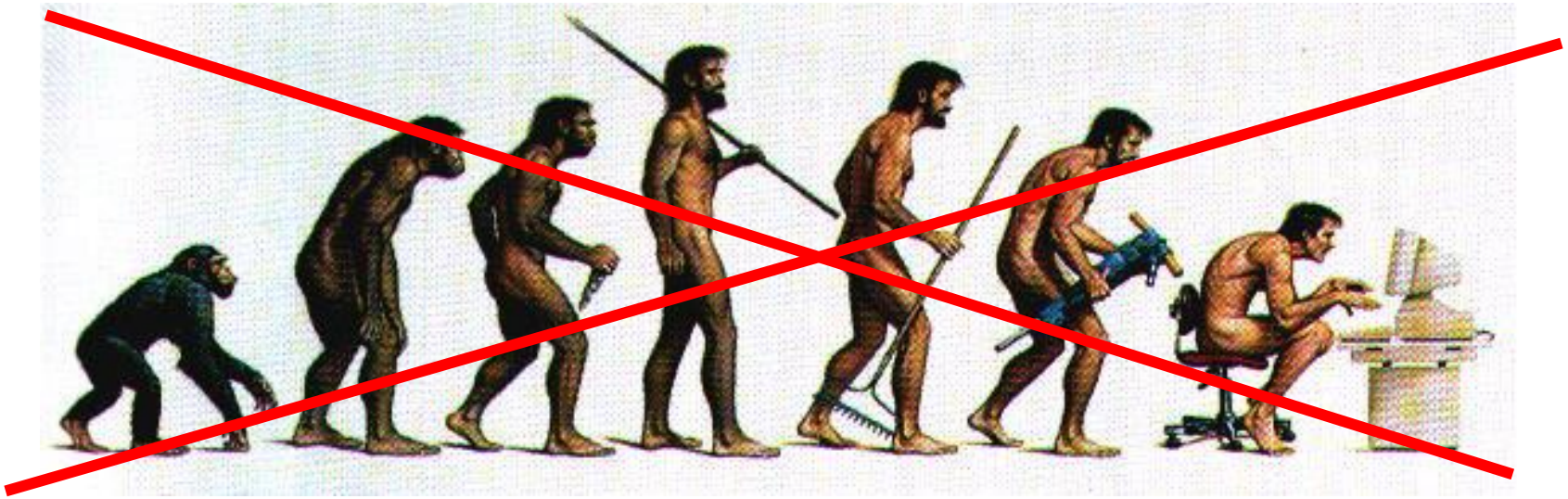


CoBrA not CORBA!

Outline

- **Introduction**
  - Issues in building context-aware systems
  - Context Broker Architecture (CoBrA)
- **Background**
  - Previous work in context-aware systems
- **Approach & Plans**
  - CoBrA prototype
- **Conclusions**

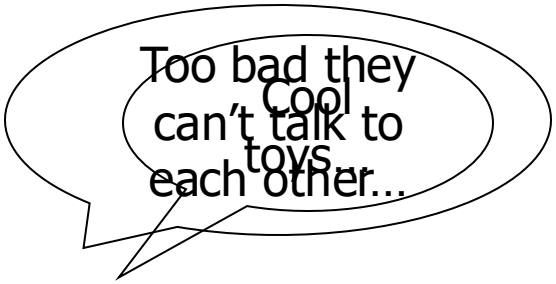
Computing Evolution …



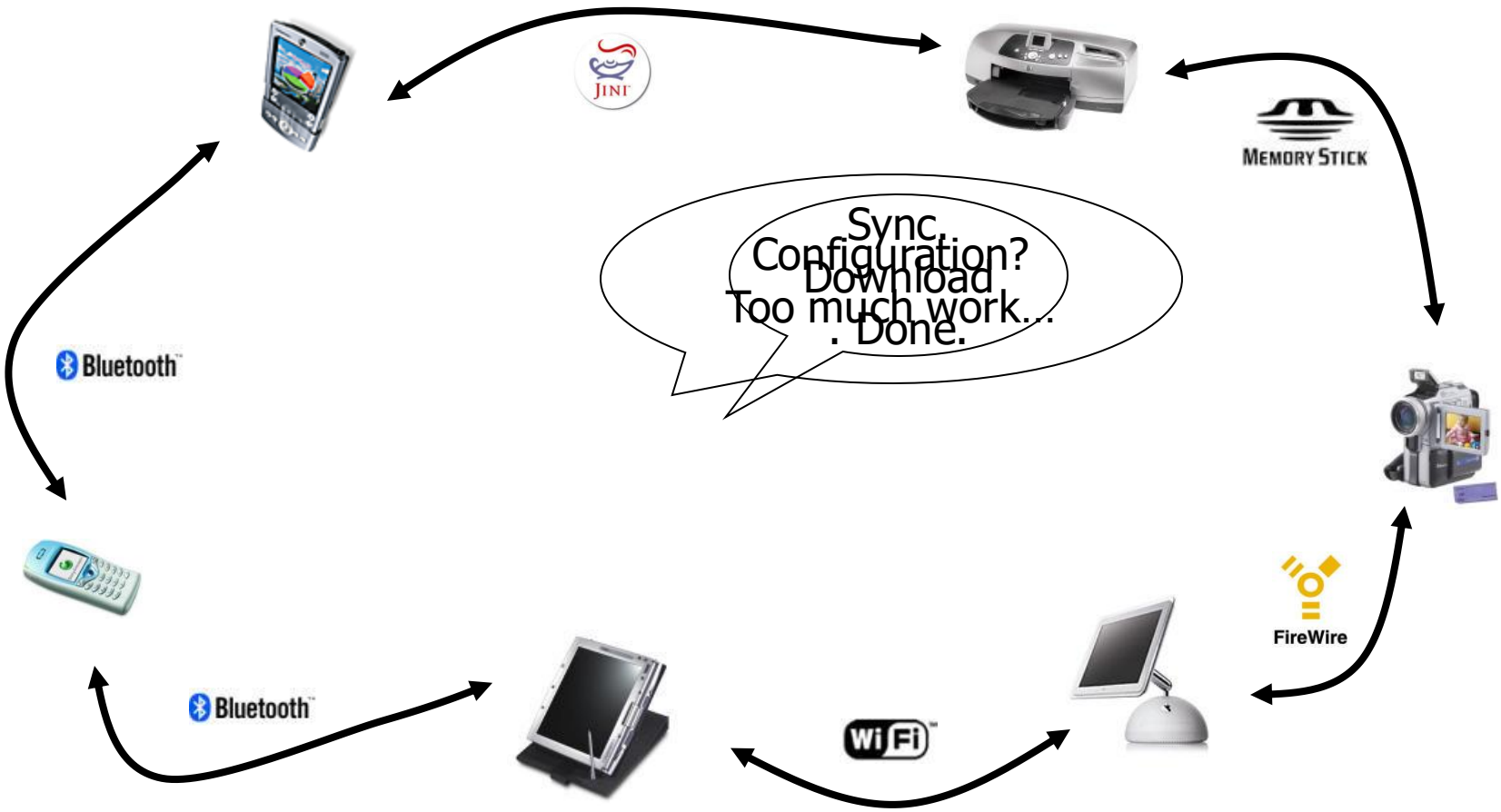
The Vision

- **Pervasive Computing**: a natural extension of the present human computing life style
  - Using computing technologies will be as natural as using other non-computing technologies (e.g., pen, paper, and cups)
  - Computing services will be something that is available anytime and anywhere.

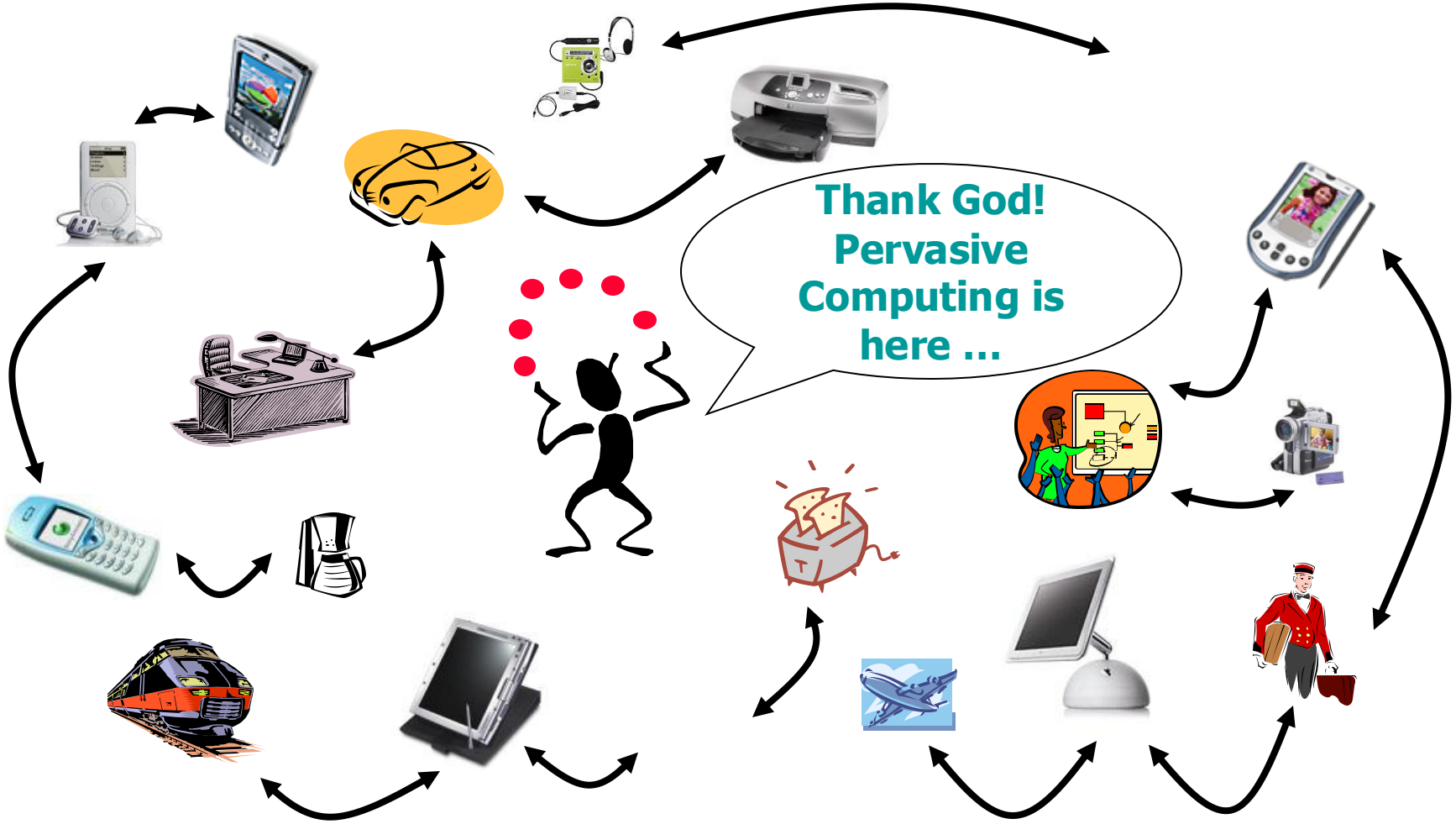
Yesterday: Gadget Rules



Today: Communication Rules



Tomorrow: Services Will Rule



One Step Towards the Vision

- **Context-aware systems**: computer systems that can anticipate the needs of users and act in advance by "understanding" their context
  - Systems know I am the speaker
  - Systems know you are the audiences
  - Systems know we are in a meeting
  - …

Contexts

- By context, we mean the situational conditions that are associated with a user
  - Location, room temperature, lighting conditions, noise level, social activities, user intentions, user beliefs, user roles, personal information, etc.

Research Issues

- **Context Modeling & Reasoning**
  - How to build representations of context that can be processed and reasoned about by the computers
- **Knowledge Maintenance & Sharing**
  - How to maintain consistent knowledge about the context and share that information with other systems
- **User Privacy Protection**
  - How to give users the control of their situational information that is acquired from the hidden sensors

Research Contributions

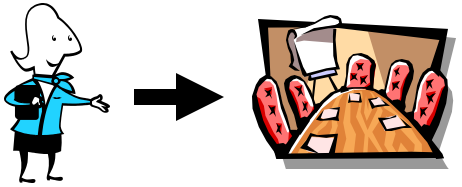
- Developing a broker-centric agent architecture to support pervasive context-aware systems
  - Defines ontologies for context modeling and reasoning
  - Includes a logic inference engine to reason with contextual information and to detect and resolve inconsistent context knowledge
  - Defines a policy language that users can use to control the usage and the sharing of their context information

Other Contributions

- Prototype an intelligent meeting room system that exploits CoBrA
  - Providing relevant services and information to meeting participants based on their situational needs
  - Allowing users to control the use and the sharing their location and social context.

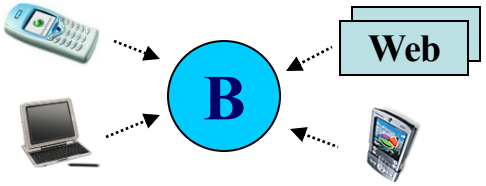
An EasyMeeting Scenario

Alice enters a conference room

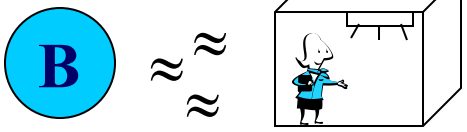


Alice "beams" her policy to the broker

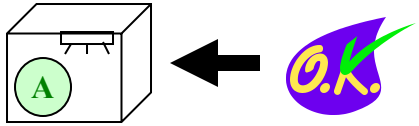
The broker builds the context model



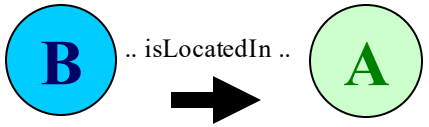
The broker detects Alice's presence



Policy says, "can share with any agents in the room"

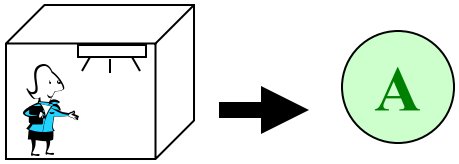


Policy says, "inform my personal agent of my location"

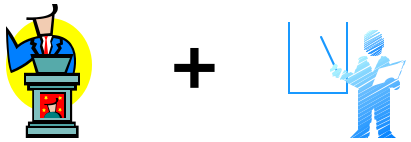


An EasyMeeting Scenario

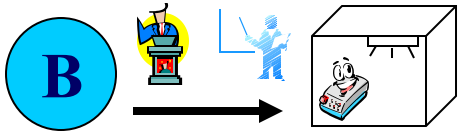
The broker tells her location to her agent



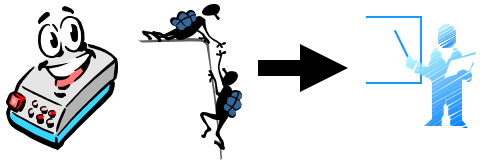
Her agent informs the broker of her role and intentions



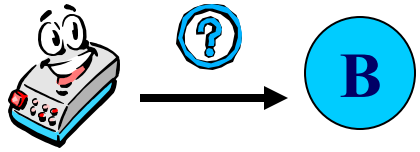
The broker informs the subscribed agents



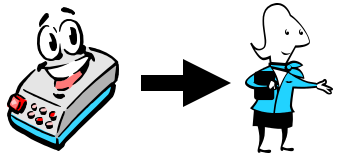
The projector agent wants to help Alice



The projector agent asks slide show info.



The projector agent sets up the slides



Background

Different Types of Context-Aware Systems

	Enhancing User Interface	Guiding Behavior Adaptation	Building Pervasive Computing Env.
Microsoft Cassiopeia E-105E	X		
Video Streaming App. (Odyssey)		X	
MIT Intelligent Room			X
XeroxPARC Active Badge Apps		X	
Cooltown Museum			X
Context Broker Architecture			X

Different Designs of Context-Aware Architectures

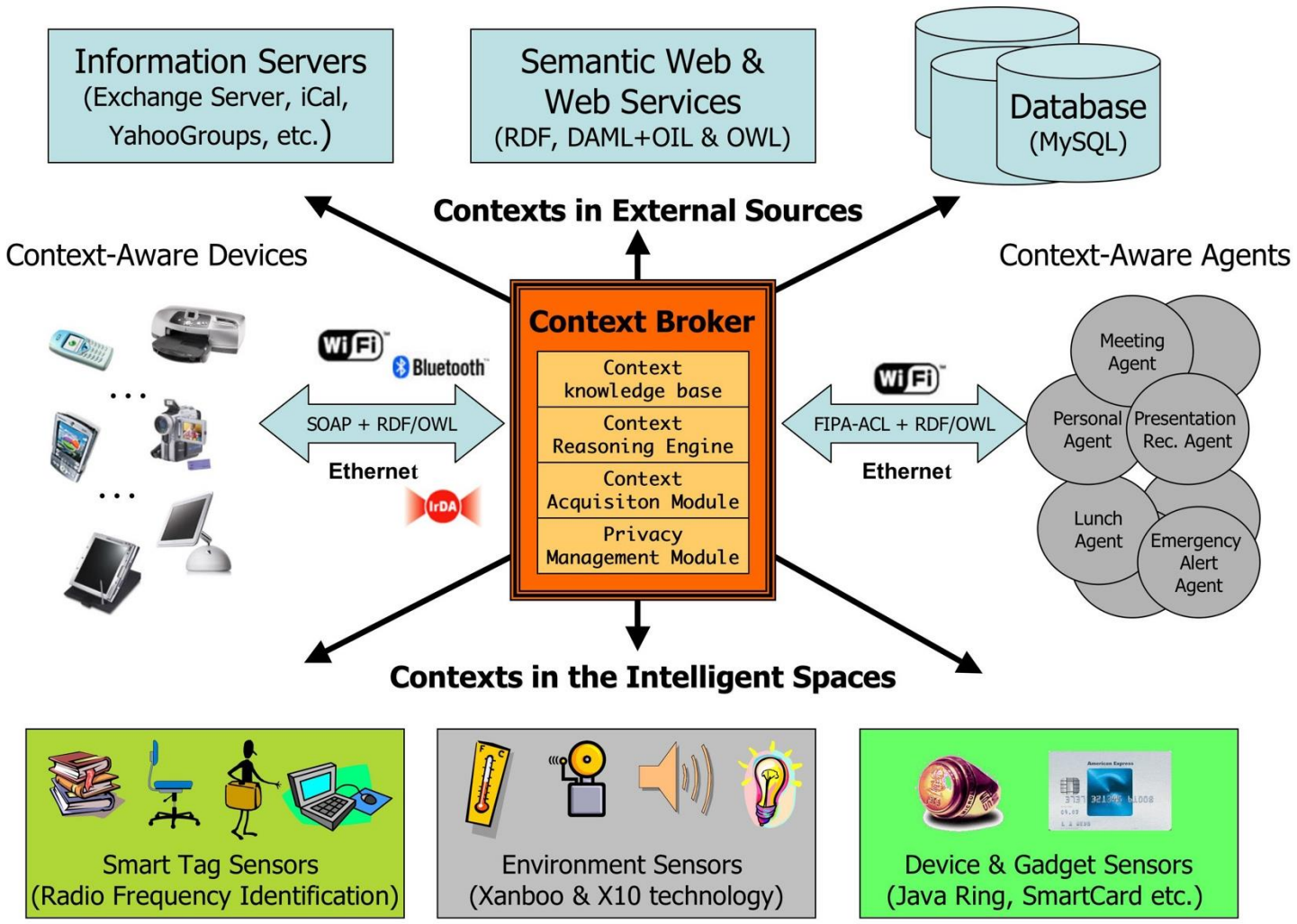
	Direct Sensors Access	Facilitated by Middle-wares	Server-oriented Approach
Microsoft Cassiopeia E-105E	X		
XeroxPARC Active Badge Apps	X		
Video Streaming App. (Odyssey)		X	
Context Toolkit		X	
MIT Intelligent Room	X	X	
Context Broker Architecture			X

The Shortcomings of the Previous Systems

- Lacking an adequate representation for modeling context
- Individual agents are responsible for managing their own context knowledge
- Users do not have full control over how their context information is shared and used

Context Broker Architecture (CoBrA)

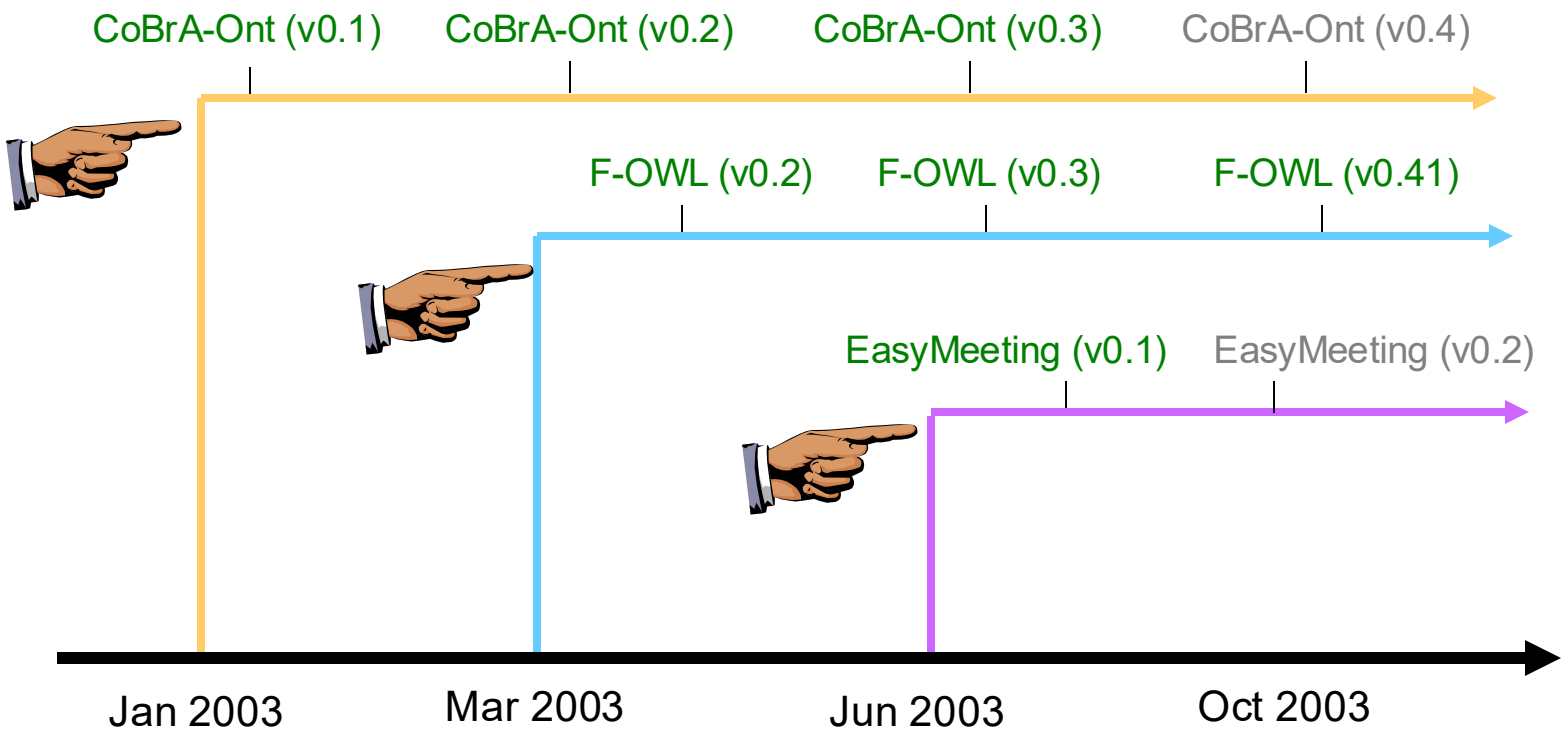
A Bird's Eye View of CoBrA



Key Features of CoBrA

- Using OWL to define ontologies to enable agents to process and reason about context
- Taking a rule base approach to build an inference engine for reasoning with context
- Using a policy-based approach to control how context knowledge are shared

CoBrA Research Roadmap



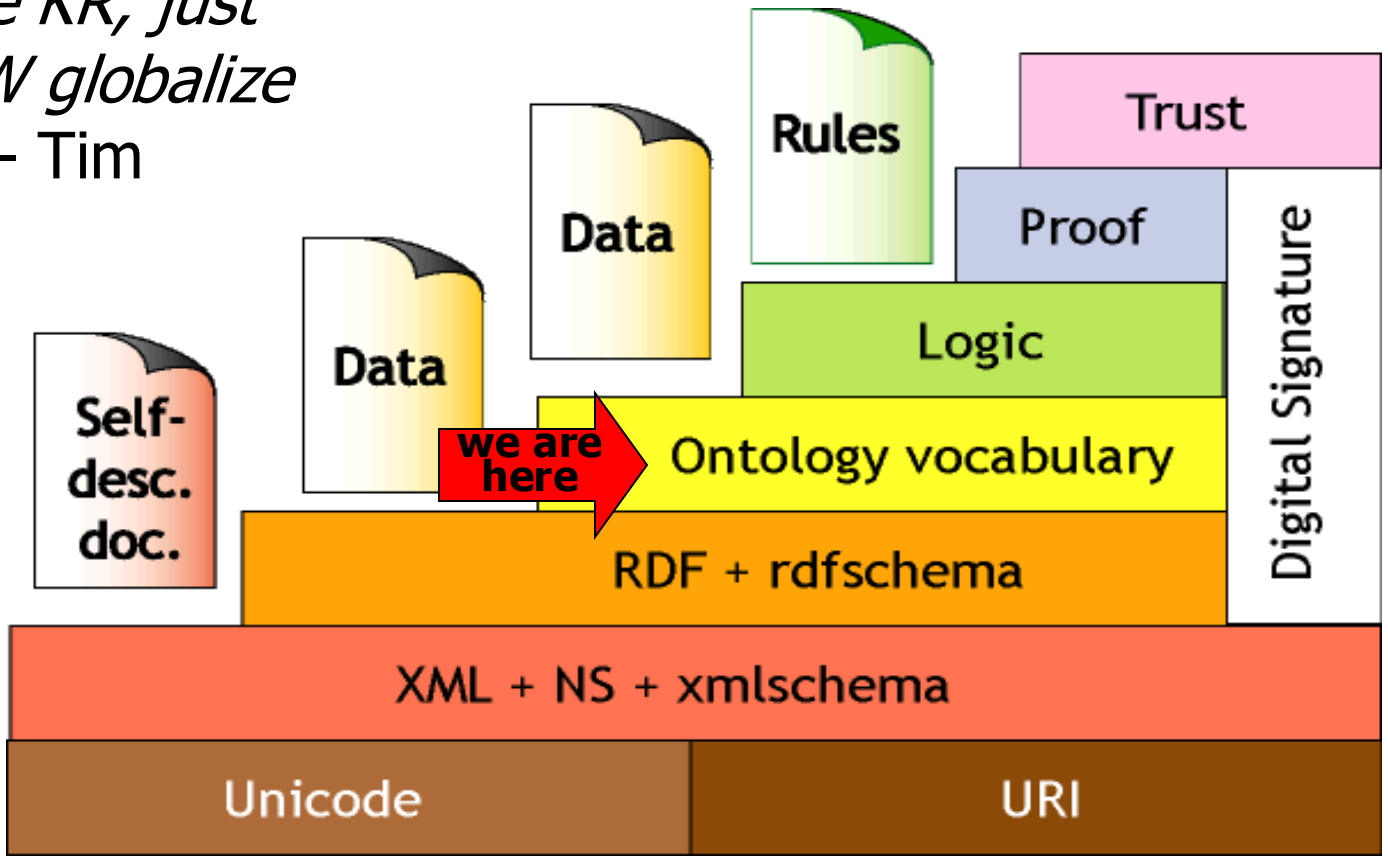
Ontologies (in OWL) for supporting context-aware systems

About Semantic Web

- Semantic Web envisioned by Tim Berners-Lee is an extension to the present World Wide Web.
- The focus is on enabling computers to be able *to reason* about web information in addition *to displaying* web information.

Semantic Web 101

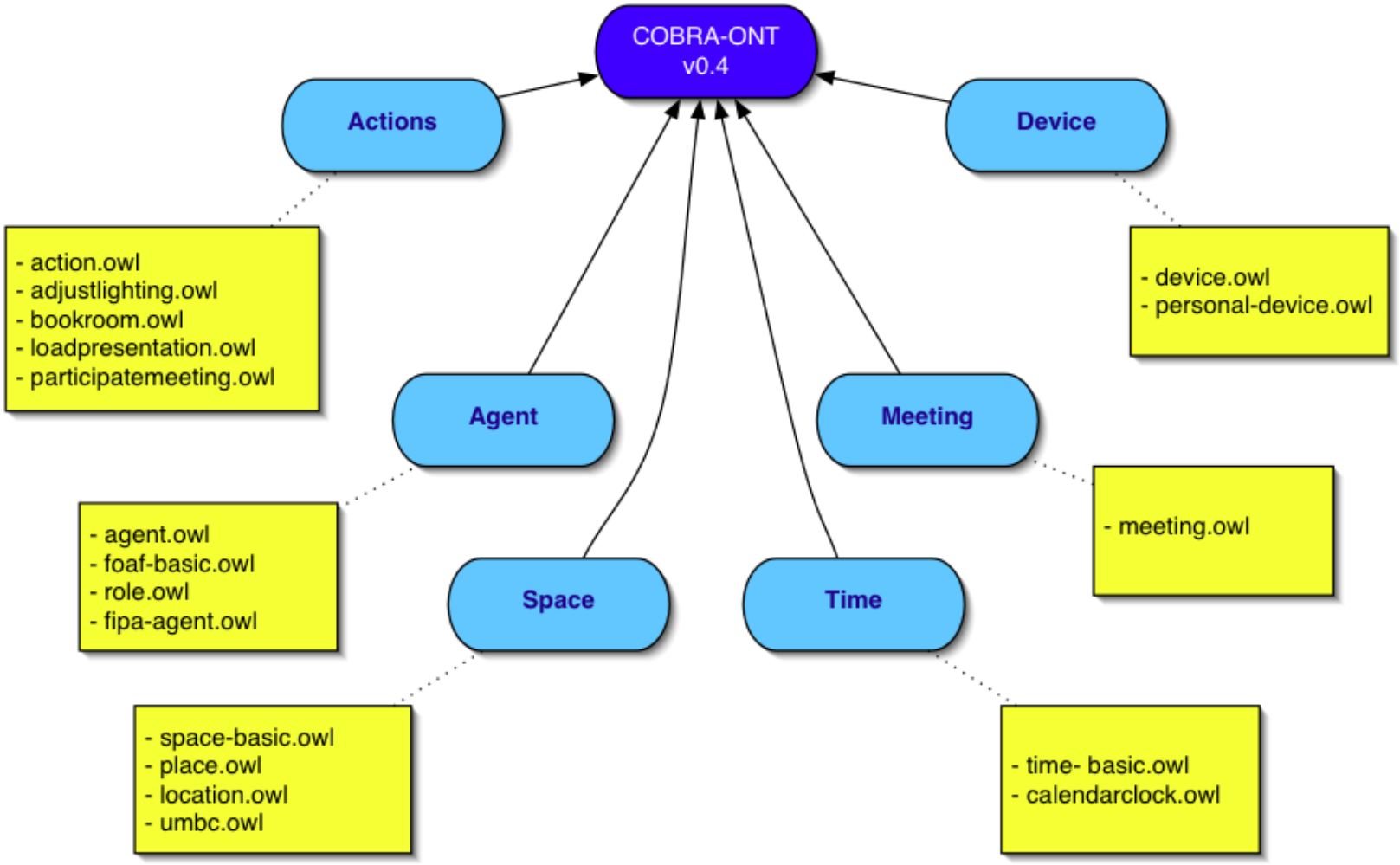
"The Semantic Web will globalize KR, just as the WWW globalize hypertext" -- Tim Berners-Lee



Semantic Web Languages

- RDF/RDFS (supported by W3C)
  - Defines basic N-Triple modeling
  - Every piece of web information is represented as a “resource”
- DAML+OIL (supported by DRAPA)
  - Adds Description Logic extension to the existing RDF/RDFS
- **OWL** (supported by W3C)
  - DAML+OIL “v2.0”
  - Better defined ontology vocabularies

The CoBrA Ontology (v0.4)



COBRA-ONT Design

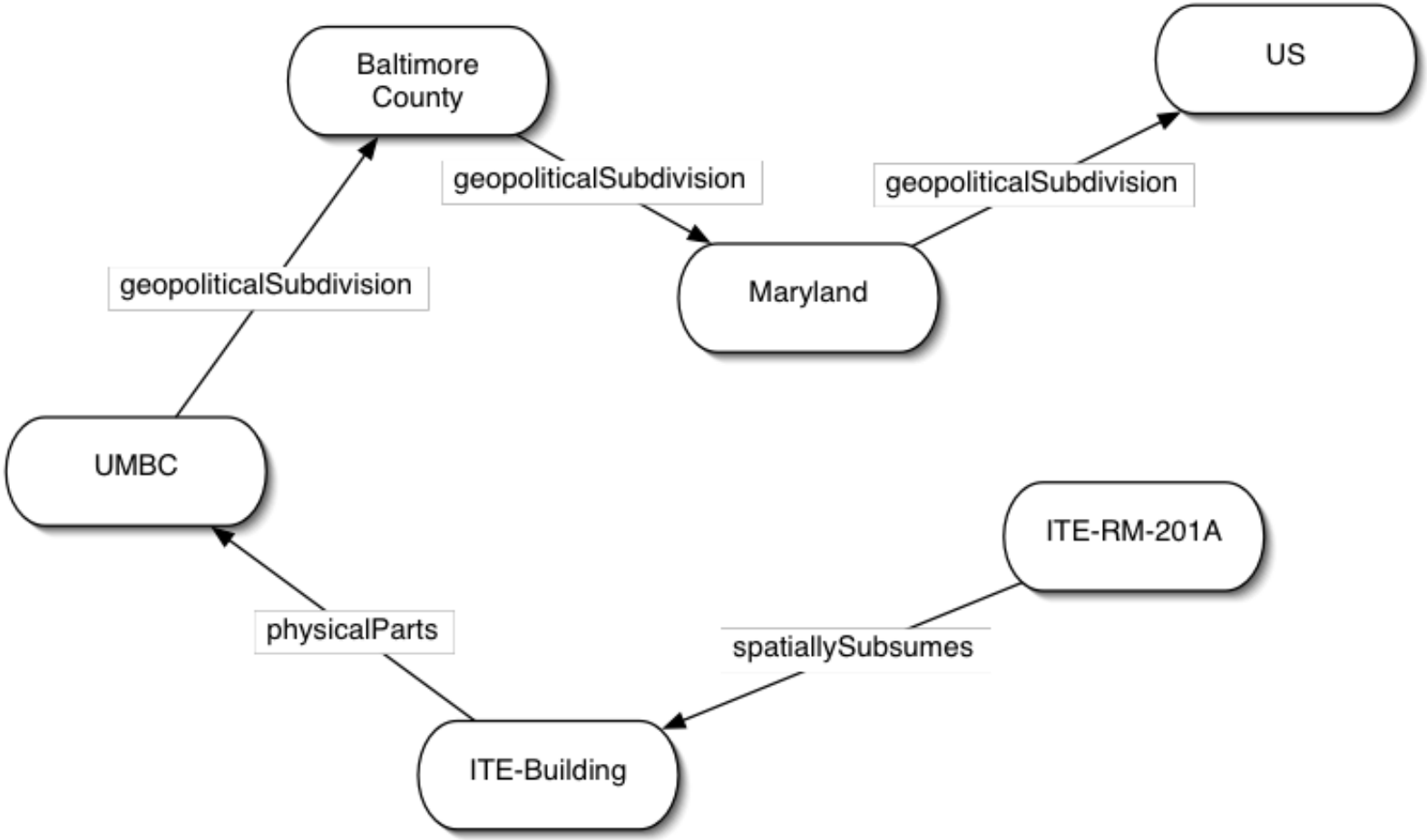
- A set of ontologies for supporting knowledge sharing and context reasoning
- Ontologies of different subjects are grouped with distinctive “namespaces”.
- Always use “owl:import” if possible
- Adopts and maps to other consensus ontologies (e.g., DAML Time, OpenCyc spatial, FIPA Device, FOAF, ITTalks)

Example 1: Location Inference

- **Goal**: Develop a context broker that can reason about a person's location using available sensing info.

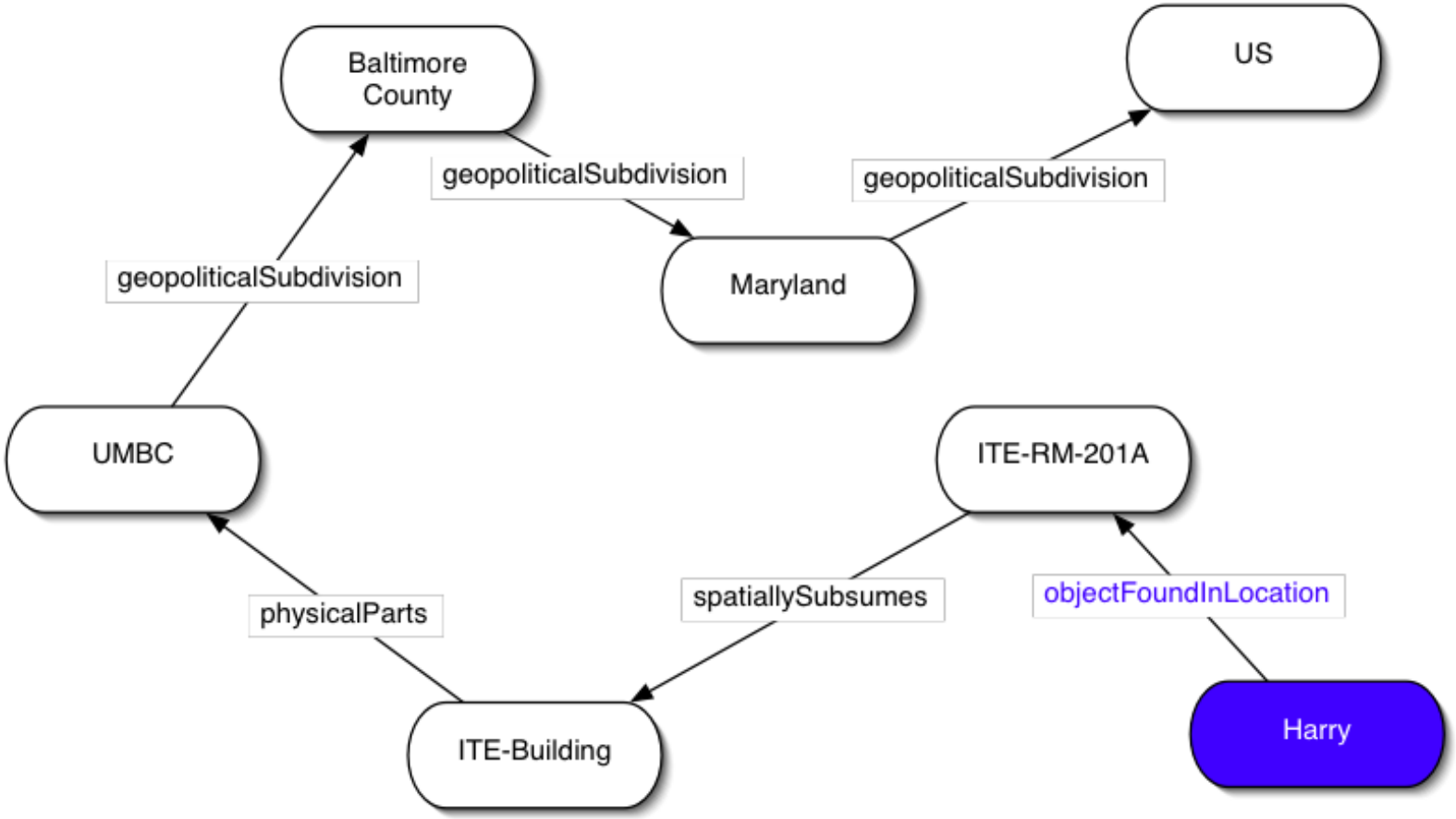
  => Step 1: Define a spatial ontology of the domain

A Simple UMBC Ontology



Location Inference

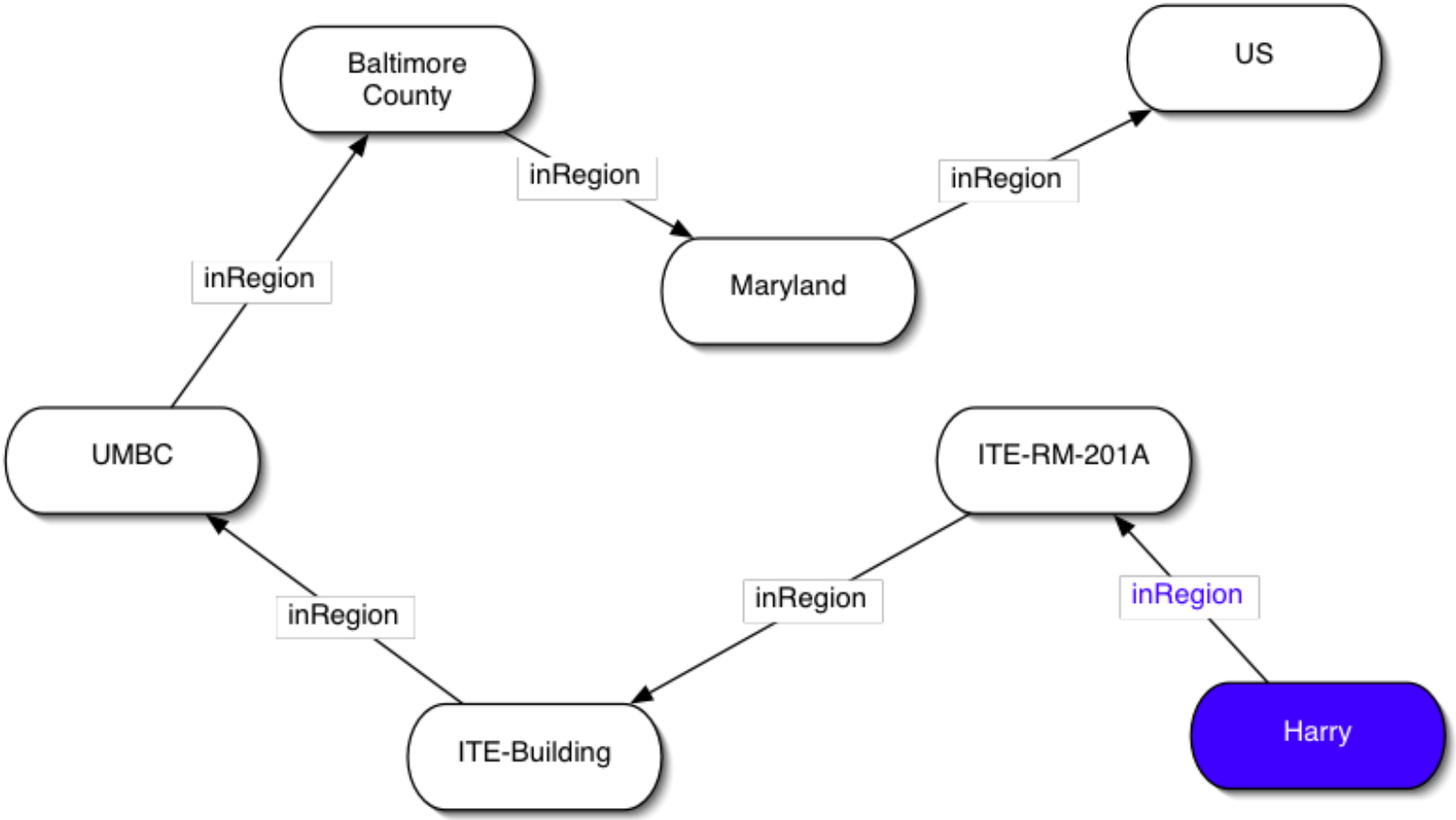
- Assume the broker is told that Harry is located in RM-201A



Location Inference

- A: the used spatial relations are “rdfs:subProeprtyOf” the “inRegion” proeprty
- B: “inRegion” is a type of “Transitive Property”
  - If p(x,y) & p(y,z) => p(x,z).
- Based on A & B => …

Location Inference



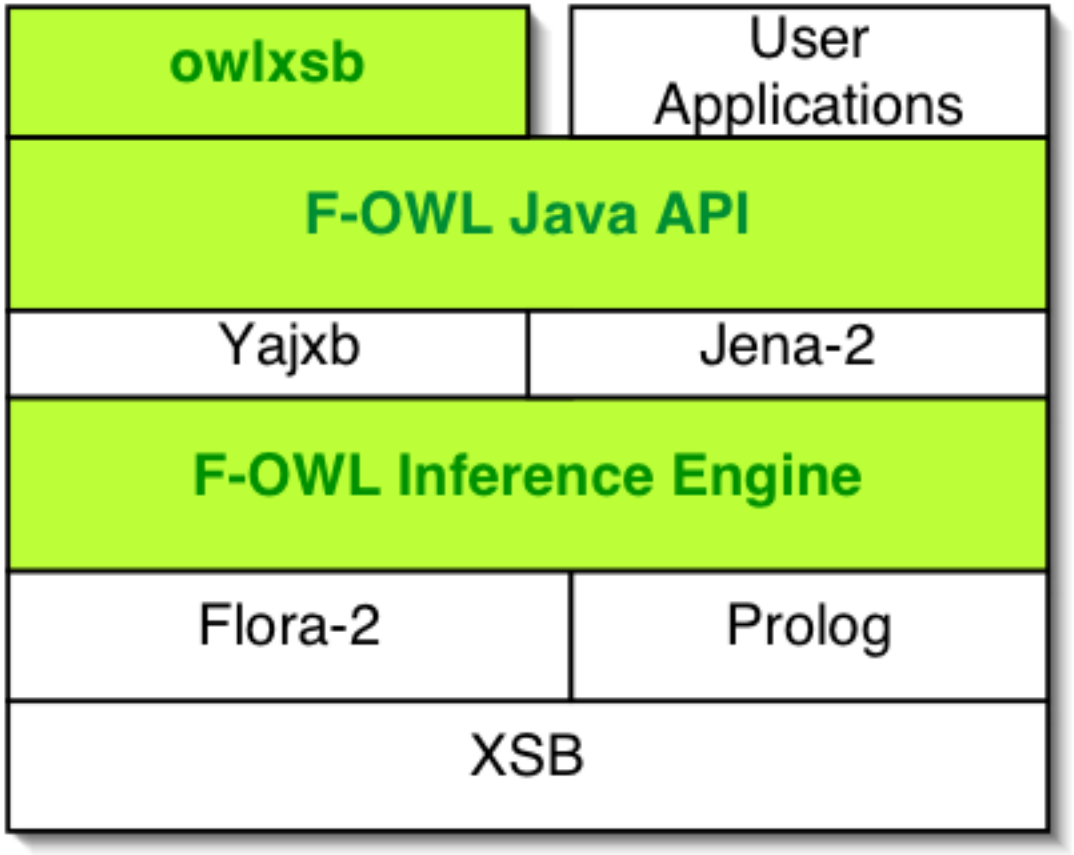
Example 2: Spotting Sensor Errors

- Premise (static knowledge):
  - R210 rdf:type AtomicPlace.
  - ParkingLot-B rdf:type AtomicPlace.
- Premise (dynamic knowledge):
  - Harry isLocatedIn R210.
  - Harry isLocatedIn ParkingLot-B.
- Premise (domain knowledge):
  - No person can be located in two different AtomicPlace at the same time.
- Conclusion:
  - There is an error in the knowledge base.

F-OWL

- **F-OWL** is an implementation of the OWL inference rules in Flora-2.
  - Flora-2 is an F-Logic (Frame Logic) based language in XSB (Prolog).
  - F-Logic is an object-oriented knowledge representation language.
- Similar to TRIPLE, F-OWL defines the ontology models in rules.

F-OWL Design


owlxsb
User Applications
F-OWL Java API
Yajxb
Jena-2
F-OWL Inference Engine
Flora-2
Prolog
XSB

An Example of F-OWL

Premises

animals:John a animals:Person.
animals:Mark a animals:Person ; animals:hasFather animals:John.
animals:hasFather rdfs:subPropertyOf animals:hasParent.
animals:hasChild owl:inverseOf animals:hasParent.

Query

Who is John's child? What classes does John belong to?
Who are the parents of Mark?

F-OWL Query

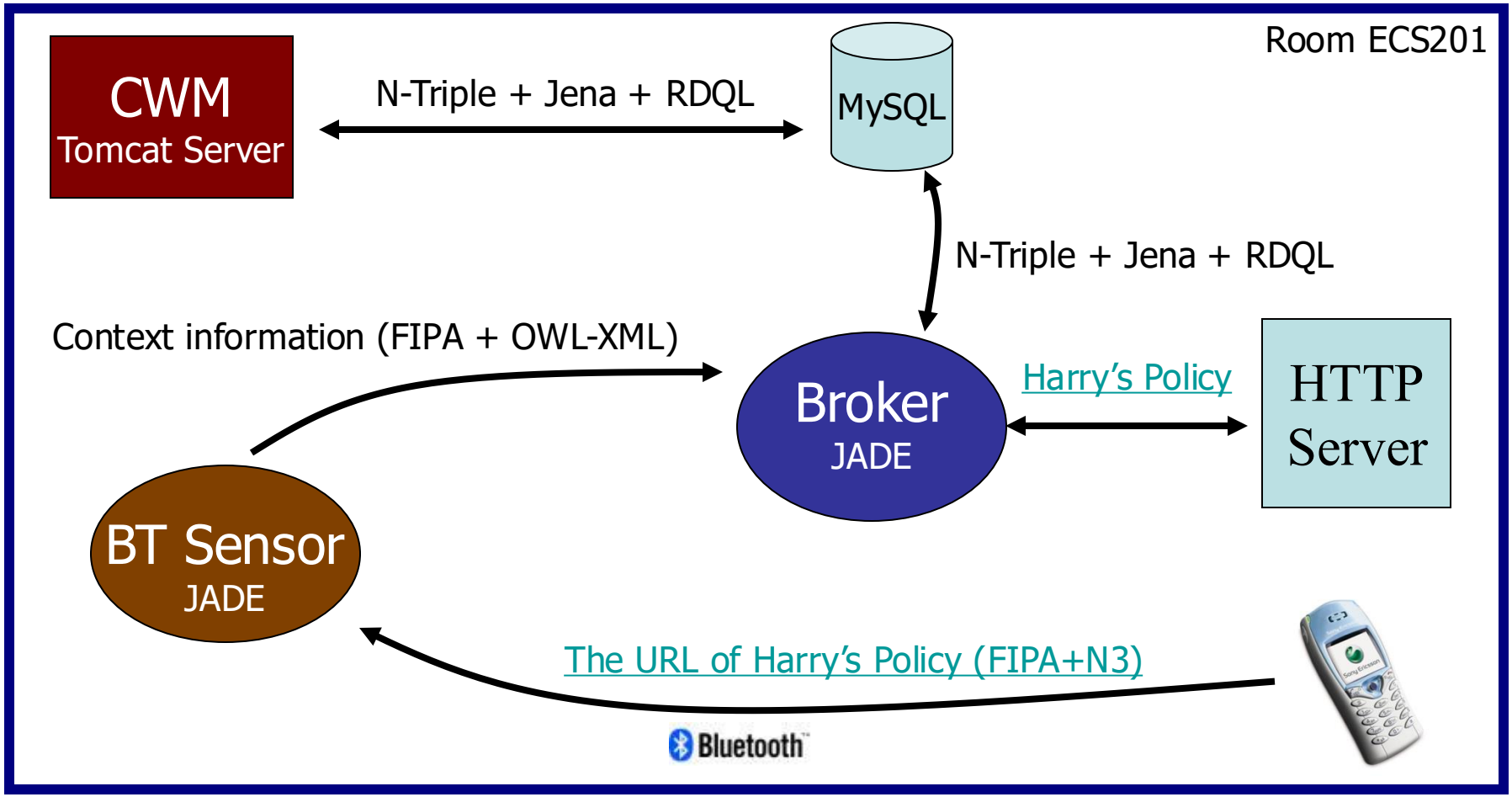
animals_John:Class [animals_hasChild -> X].
animals_Mark [animals_hasParent -> X].

More about F-OWL

- **F-OWL** (aleph release)
  - F-OWL v0.41 (as of today) supports a full RDF-S inference and limited OWL inference (OWL-Lite and some OWL Full).

http://fowl.sourceforge.net

EasyMeeting Prototype



Work In Progress

- Implementing a rule based inference engine to reason about the temporal and spatial relations that are associated context events
  - Allen's temporal interval calculus
  - Region Connection Calculus (RCC8)
  - Abductive Reasoning
- Using REI, a security policy language based on deontic concepts, to develop a policy-based systems to protect user privacy

Privacy Policy Use Case (1)

- The speaker doesn't want others to know the specific room that he is in, but does want others to know that he is present on the school campus
- He defines the following policies:
  - Can share my location with a granularity > ~1 km radius
- The broker:
  - isLocated(US) => Yes!
  - isLocated(Maryland) => Yes!
  - isLocated(BaltimoreCounty) => Yes!
  - isLocated(UMBC) => Yes!
  - isLocated(ITE-RM-201A) => I don't know…

Privacy Policy Use Case (2)

- The problem of inference!
  - Knowing your phone + white pages => I know where you live
  - Knowing your email address (.mil, .gov) => I know you works for the government
- The broker models the inference capability of other agents
  - mayKnow(X, homeAdd(Y)) :- know(X,phoneNum(Y))

Conclusions

Conclusions

- By providing a broker to manage and reason about context, we can greatly reduce the difficulty and cost in building context-aware systems
  - A repository of context knowledge can help resource-limited devices to become context aware
  - Ontologies can help agents to share context knowledge, reducing the redundancy in sensing
  - Policies can give users the control of their context information, protecting their privacy in an open environment

Questions?

- Harry Chen
  - http://umbc.edu/~hchen4/
  - Email: harry.chen@umbc.edu
- CoBrA
  - http://cobra.umbc.edu/
- eBiquity.ORG
  - Pervasive computing news and development
  - Since 2000