Ontology-based Context Modeling

CMSC 818G – Student Presentation
03/12/2013
Samet Ayhan

Disclaimer: All tables and figures presented herein made available from the original articles at:
http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.61.977
http://dl.acm.org/citation.cfm?id=991806
Overview

• An ontology for Context-aware Pervasive Computing Environments

• SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications
  • Int’l Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004

• Harry Chen, Filip Perich, Tim Finin, Anupam Joshi
• Department of Computer Science and Electrical Engineering
• University of Maryland, Baltimore County
SOUPA – Problem Statement

• Pervasive Computing: Computer systems to
  • Seamlessly integrate into the life of everyday users
  • Provide them with services and information in “anywhere, anytime” fashion

• Many prototyping systems demonstrated pervasive computing

• They offer only weak support for knowledge sharing and reasoning
  • Not built on a foundation of common ontologies w/explicit semantic representation
  • Use programming language objects to represent knowledge about situational environment
  • So, can’t facilitate knowledge sharing in open and dynamic environment
SOUPA – Introduction

• To address these issues shared ontology must be developed
  • Knowledge sharing
  • Context reasoning
  • Interoperability in ubiquitous and pervasive computing systems

• SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications
  • Uses Web Ontology Language (OWL)
  • Includes modular component vocabularies
  • Representing intelligent agents with beliefs, intensions, space, events, user profiles, actions, and policies for security and privacy
SOUPA – Introduction

• Goal: define ontologies for supporting pervasive computing applications
• Design: driven by a set of use cases
• Merits: provide developers a shared ontology that combines useful vocabularies from various consensus ontologies

• SOUPA is a part of ongoing effort of the Semantic Web in Ubi-Comp Special Interest Group
The Web Ontology Language, OWL

• Semantic Web language for use by computer applications
• Knowledge representation language for defining and instantiating ontologies
• For processing the content
• Developed by Web initiatives sponsored by W3C

• The current human-centered web is largely HTML
  • Focuses rendering of text and images for human viewing
• XML is an alternative encoding
  • Primarily for machine processing
  • Does not represent and reason about the knowledge essential to semantic Web vision
The Web Ontology Language, OWL

• Ontology is a formal explicit description of concepts in domain of discourse
• Normative OWL exchange syntax is RDF/XML
• Ontologies expressed in OWL
  • Placed on web servers as web documents,
  • Referenced by other ontologies
  • Downloaded by applications that use ontologies
Related Ontologies

• Part of the SOUPA vocabularies are adopted from other ontologies (borrowed)
• Borrowed terms in SOUPA are mapped to the foreign ontology terms using the mapping constructs
  • For better interoperability

• Ontologies referenced by SOUPA includes
  • Friend-Of-A-Friend ontology (FOAF)
  • DAML-Time and the entry sub-ontology of time
  • Spatial ontologies in OpenCyc, Regional Connection Calculus
  • COBRA-ONT, MoGATU BDI ontology and
  • Rei policy ontology
Related Ontologies

• FOAF
  • Allows expression of personal information and relationships
  • Supports creating info systems for online communities
• DAML-Time & the Entry Sub-ontology of Time
  • Vocabularies designed for expressing temporal concepts
  • Properties common to any formalization of time
• OpenCyc Spatial Ontologies & RCC
  • Defines a comprehensive set of vocabularies for symbolic representation of space
  • Supports qualitative spatial reasoning
Related Ontologies

• COBRA-ONT & MoGATU BDI Ontology
  • Supports knowledge representation and ontology reasoning
  • COBRA-ONT focuses on modeling concepts in smart meeting rooms
  • MoGATU BDI ontology focuses on modeling the belief, and intention of human users and software agents

• Rei Policy Ontology
  • Defines a set of deontic concepts (rights, prohibitions, etc.)
  • For specifying and reasoning about security access control rules
SOUPA Ontologies

- Consists of two distinctive but related set of ontologies

  - SOUPA Core and
    - Attempts to define generic vocabularies that are universal

  - SOUPA Extension
    - Extended from the core
    - Defines additional vocabularies
      - For supporting specific types of applications
      - Provides examples for future ontology extensions
SOUPA Core & SOUPA Extension

SOUPA Ontology (2004-06)
http://pervasive.semanticweb.org/ont/2004/06/
SOUPA Core

- Person
  - Defines vocabularies describing contact information and profiles of a person
  - \texttt{per:Person} defined to represent a set of all people in the SOUPA domain

```xml
<per:Person>
  <per:firstName>
    \texttt{rdf:datatype="\texttt{xsd:string}">Harry</per:firstName>
  <per:lastName>
    \texttt{rdf:datatype="\texttt{xsd:string}">Chen</per:lastName>
  <per:gender rdf:resource="\texttt{&per;Male}"/>
  <per:birthDate>
    \texttt{rdf:datatype="\texttt{xsd:date}">1976-12-26</per:birthDate>
  <per:homepage>
    \texttt{rdf:resource="http://umbc.edu/people/hchen4"}/>
  <foaf:weblog>
    \texttt{rdf:resource="http://umbc.edu/people/hchen4"}>
  <per:hasSchoolContact rdf:resource="#SchoolContact"/>
  <per:hasHomeContact rdf:resource="#HomeContact"/>
  <foaf:workplaceHomepage>
    \texttt{rdf:resource="http://ebiquity.umbc.edu"}/>
  <foaf:workplaceHomepage>
    \texttt{rdf:resource="http://www.umbc.edu"}/>
  <foaf:workplaceHomepage>
    \texttt{rdf:resource="http://www.cs.umbc.edu"}>
</per:Person>

<per:ContactProfile rdf:ID="SchoolContact">
  <per:address rdf:datatype="\texttt{&xsd:string}">
    Dept. of CSEE, UMBC, 1000 Hilltop Circle, Baltimore, MD 21250, USA
  </per:address>
  <per:phone>
    \texttt{rdf:datatype="\texttt{&xsd:string}>
    +1-410-455-8648
  </per:phone>
  <per:email>
    \texttt{mailto:harry.chen@umbc.edu"}>
  <per:im>
    \texttt{rdf:resource="aim:goim?screenname=hc1379"}>
</per:ContactProfile>
...
SOUPA Core

- Policy & Action
  - Defines vocabularies representing security and privacy policies
  - Description logic based mechanism for reasoning defined policies
  - `act:Action` represents a set of all actions
  - `pol:Policy` represents a set of all policies

```xml
<owl:Class rdf:ID="ShareHarryLocInfoWithEBMembers">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#act:Action"/>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#act:actor"/>
      <owl:hasValue>
        <agt:Agent rdf:about="ctb@cobra1.cs.umbc.edu"/>
      </owl:hasValue>
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>

<owl:Class rdf:ID="LocationContextOfHarry">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#loc:LocationContext"/>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#loc:contextOf"/>
      <owl:hasValue>
        <loc:LocationContext rdf:resource="&eb:EBiquityMembers"/>
      </owl:hasValue>
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>
```

...
SOUPA Core

• Agent & BDI
  • Defined with a strong notion of agency, by a set of mentalistic notions such as knowledge, belief, intention and obligation
  • Both computational entities and human users can be modeled as agents
  • \textit{agt:Agent} represents a set of all agents
  • Associated with three properties
    • \textit{agt:believes}
    • \textit{agt:desires}
    • \textit{agt:intends}
SOUPA Core

• Time
  • Defines a set of ontologies for expressing time and temporal relations
  • Used to describe temporal properties of different events that occur in physical world
  • tme:TimeInstant
  • tme:TimeInterval
SOUPA Core

• Space
  • Designed to support reasoning about the spatial relations between various types of geographical regions, mapping from geospatial coordinates to symbolic representation of space and vice versa
  • Partly adopted from OpenCyc and OpenGIS
  • spc:SpatialThing represents a set of all things with spatial extensions
  • To support spatial containment reasoning spc:GeographicalSpace class can relate to each other through spc:spatiallySubsumes and spc:spatiallySubsumedBy properties
SOUPA Core

• Event
  • Event activities that have both spatial and temporal extensions
  • Used to describe occurrence of various activities, schedules and sensing events
  • `eve:Event` represents a set of all events
  • `eve:SpatialTemporalThing` represents a set of things with both spatial and temporal extensions
SOUPA Extension

• Meeting & Schedule
  • For describing typical information associated with meetings, event schedules, and event participants

• Document & Digital Document
  • For describing meta information about documents and digital documents

• Image Capture
  • When a camera phone takes a picture, this event type of image capturing event
SOUPA Extension

- Region Connection Calculus
  - A spatial ontology supplementing the core space ontology
  - Used for expressing spatial relations for qualitative spatial reasoning

- Location
  - For describing sensed location context of a person or an object
  - Used for describing whereabouts of a person or an object
SOUPA Applications

• CoBrA
  • Broker-centric architecture for supporting context-aware systems in small spaces

• MoGATU
  • Framework for handling pro-active peer-to-peer semantic data management in a pervasive computing environment
COBRA-ONT – Problem Statement

- In previous systems, user location information is widely used for guiding adaptive behavior of the systems.

- However, none have taken advantage of the semantics of spatial relations in reasoning about context.

- Previous systems often implemented context as simple programming language objects.
  - These representations require establishment of a prior low-level implementation agreement between the programs that wish to share information.
  - They cannot facilitate sharing of contextual knowledge.
COBRA-ONT – Introduction

• CoBrA is a broker-centric agent architecture for supporting context-aware systems in smart spaces

• Central to the architecture is the presence of an intelligent agent called the context broker
  • Specialized server entity that runs on a resource-rich stationary computer in the space
  • Its role is to maintain a shared model of context on behalf of a community of agents and devices in the space
  • Protect the privacy of users by enforcing the user-defined policies when sharing information with agents in the space
COBRA-ONT – Introduction

• All computing entities in a smart space are presumed to have prior knowledge about the presence of a context broker

• High level agents are presumed to communicate with the broker using standard FIPA Agent Communication Language

• Design of context broker comprises:
  • Context Knowledge Base: persistent storage of context knowledge
  • Context Reasoning Engine: reactive inference engine reasoning over the stored context knowledge
  • Context Acquisition Module: a library of procedures forming a middleware abstraction for context acquisition
  • Policy Management Module: a set of inference rules deducing instructions for deciding right permissions for various entities to share a particular piece of contextual information
COBRA-ONT – Introduction

- Centralized design of broker could create a bottleneck
- To address the problem, fault-tolerant approach proposed
  - Multiple brokers grouped together to form a broker federation
COBRA-ONT – Use Cases

• A sensor agent detects the presence of Bluetooth-enabled cell phone in Room 210
• It composes a description of this sensed event using COBRA-ONT, which then is sent to the context broker in the associated space
• The broker asserts that the owner of the device is also in present in Room 210
• Based on a physical location ontology predefined in COBRA-ONT, knowing Room 210 is a part of the CS Building which is a part of the UMBC campus, the context broker concludes the device owner is in school today
COBRA-ONT – Use Cases

• After a speaker enters the meeting room, her mobile device sends the context broker her predefined user policy, describing the privacy rules, broker should enforce during the meeting

• Knowing the user does not want to reveal her home address to services at the meeting, based on a privacy protection ontology predefined in COBRA-ONT, the broker reasons that it should keep secret her home phone number also since it is relatively easy to determine an address given a telephone number
COBRA-ONT – Classes and Properties v0.2

• Consists of 41 classes and 36 properties

• Categorized into 4 distinctive but related themes
  • Ontologies about physical places
  • Ontologies about agents
  • Ontologies about the location context
  • Ontologies about the activity context of the agents
# COBRA-ONT – Classes and Properties v0.2

<table>
<thead>
<tr>
<th>CoBrA Ontology Classes</th>
<th>CoBrA Ontology Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>“Place” Related</strong></td>
<td><strong>“Place” Related</strong></td>
</tr>
<tr>
<td>Place</td>
<td>latitude</td>
</tr>
<tr>
<td>AtomicPlace</td>
<td>longitude</td>
</tr>
<tr>
<td>CompoundPlace</td>
<td>hasPrettyName</td>
</tr>
<tr>
<td>Campus</td>
<td>spatiallySubsumedBy</td>
</tr>
<tr>
<td>Building</td>
<td>spatiallySubsumes</td>
</tr>
<tr>
<td>AtomicPlaceInBuilding</td>
<td>accessRestricted-ToGender</td>
</tr>
<tr>
<td>Room</td>
<td>lotNumber</td>
</tr>
<tr>
<td>Hallway</td>
<td></td>
</tr>
<tr>
<td>Stairway</td>
<td></td>
</tr>
<tr>
<td>OtherPlaceInBuilding</td>
<td></td>
</tr>
<tr>
<td>Restroom</td>
<td></td>
</tr>
<tr>
<td>Gender</td>
<td></td>
</tr>
<tr>
<td>LadiesRoom</td>
<td></td>
</tr>
<tr>
<td>MensRoom</td>
<td></td>
</tr>
<tr>
<td>ParkingLot</td>
<td></td>
</tr>
<tr>
<td><strong>“Agent” Related</strong></td>
<td><strong>Agent’s Activity Context</strong></td>
</tr>
<tr>
<td>Agent</td>
<td>PresentationSchedule</td>
</tr>
<tr>
<td>Person</td>
<td>Event</td>
</tr>
<tr>
<td>SoftwareAgent</td>
<td>EventHappeningNow</td>
</tr>
<tr>
<td>Role</td>
<td>PresentationHappeningNow</td>
</tr>
<tr>
<td>SpeakerRole</td>
<td>RoomHasPresentationHappeningNow</td>
</tr>
<tr>
<td>AudienceRole</td>
<td>ParticipantOfPresentation-HappeningNow</td>
</tr>
<tr>
<td>IntentionalAction</td>
<td>SpeakerOfPresentationHappeningNow</td>
</tr>
<tr>
<td>ActionFoundInPresentation</td>
<td>AudienceOfPresentationHappeningNow</td>
</tr>
<tr>
<td>PersonFillsRoleInPresentation</td>
<td></td>
</tr>
<tr>
<td>PersonFillsSpeakerRole</td>
<td>PersonFillsAudienceRole</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Agent’s Activity Context</th>
<th>Agent’s Activity Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>hasContactInformation</td>
<td>participatesIn</td>
</tr>
<tr>
<td>hasFullName</td>
<td>startTime</td>
</tr>
<tr>
<td>hasEmail</td>
<td>endTime</td>
</tr>
<tr>
<td>hasHomePage</td>
<td>Location</td>
</tr>
<tr>
<td>hasAgentAddress</td>
<td>hasEvent</td>
</tr>
<tr>
<td>fillsRole</td>
<td>hasEventHappeningNow</td>
</tr>
<tr>
<td>isFilledBy</td>
<td>invitedSpeaker</td>
</tr>
<tr>
<td>intendsToPerform</td>
<td>expectedAudience</td>
</tr>
<tr>
<td>desiresSomeoneToAchieve</td>
<td>presentation</td>
</tr>
<tr>
<td></td>
<td>presentationAbstract</td>
</tr>
<tr>
<td></td>
<td>eventDescription</td>
</tr>
<tr>
<td></td>
<td>eventSchedule</td>
</tr>
</tbody>
</table>
COBRA-ONT – Places

• Top-level class is Place
  • Represents the abstraction of a physical location
  • Has set of properties used to describe a location (lat., long.)
• COBRA-ONT defines two special subclasses
  • AtomicPlace (Room, Hallway, Restroom, etc.)
  • CompoundPlace (Campus, Building, etc.)
• Containment property is represented by 2 properties
  • SpatiallySubsumes
  • SpatiallySubsumedBy
COBRA-ONT – Agents

- Top-level agent class is `Agent`
  - Has two predefined subclasses
    - `Person` class of all human agents
    - `SoftwareAgent` class of all software agents
  - Each agent can have associated roles in an event
    - `Role`
      - `SpeakerRole`
      - `AudienceRole`

- To describe user’s intended action, `intendsToPerform` property is defined
COBRA-ONT – Agent’s location

• Dynamic knowledge describing location of an agent
• Location property is represented by locatedIn

• Following context reasoning is defined
  • No agent can be physically present in two different atomic places during the same time interval
  • An agent can be physically present in two different compound places during the same time interval, just in case one spatially subsumes the other

• This reasoning helps the broker detect inconsistent knowledge about the current location of an agent
COBRA-ONT – Agent’s activity context

- Dynamic knowledge describing events, agent participates

- `PresentationSchedule` represents the schedule

- Associated properties describe start time, end time, presentation title, presentation abstract, and the location of presentation event

- To describe a presentation that has a speaker or an audience, one can use
  - `invitedSpeaker`
  - `expectedAudience`
Flora-2 – An OWL Inference Engine

• F-OWL, OWL inference engine
  • Supports ontology reasoning in CoBrA
  • Supports consistency checking using axiomatic rules defined in Flora-2
• Object-oriented knowledge base language
• Open API for Java applications integration
• Application development platform translating a unified language of F-logic, HiLog, and transaction logic into XSB deductive engine
Flora-2 – An OWL Inference Engine

• F-OWL consists of
  • Assertions for triple representation of the RDF and RDFS data models
  • Assertions for triple representations of OWL data model
  • Rules for reasoning with RDF and RDFS data model
  • Rules for reasoning with OWL data model

• The latest version (v0.3) supports ontology reasoning over RDFS and OWL-Lite sub-language constructs
Summary

• SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications
  • Uses Web Ontology Language (OWL)
  • Includes modular component vocabularies
  • Representing intelligent agents with beliefs, intensions, space, events, user profiles, actions, and policies for security and privacy

• An ontology for Context-aware Pervasive Computing Environments
  • CoBrA is a broker-centric agent architecture for supporting context-aware systems in smart spaces