



Aplicație Android bazată pe agenți pentru participanții la conferințe



Implementarea agentului software folosind platforma tATAmI

Emma Sevastian

Coordonator: Ș.I. Dr. Ing. Andrei Olaru

Facultatea de Automatică și Calculatoare
Universitatea POLITEHNICĂ din București

Septembrie 2014

Cuprins

1. Aplicatii existente
2. Structura aplicatiei
3. Componenta cognitiva
4. Smart Conference: Modelarea agentului
5. Concluzii

“If you look at history, innovation doesn’t come just from giving people incentives; it comes from creating environments where their ideas can connect.”

Steven Johnson

Septembrie 2014

Smart Conference

Coordonatori:

Andrei Olaru

Alexandru Sorici

in cadrul acestei aplicatii a lucrat si Alexandru Tomescu
(proiectul de diploma)

Contributia mea in cadrul acestui proiect:

integrarea Grafurilor de Context in Componenta Cognitiva

modificarea corespunzatoare a primitivelor de limbaj si a executiei

comportamentelor agentului

modelarea agentilor folosind S-CLAIM

Aplicatii existente:

- Conference2Go
- MySmartEvent

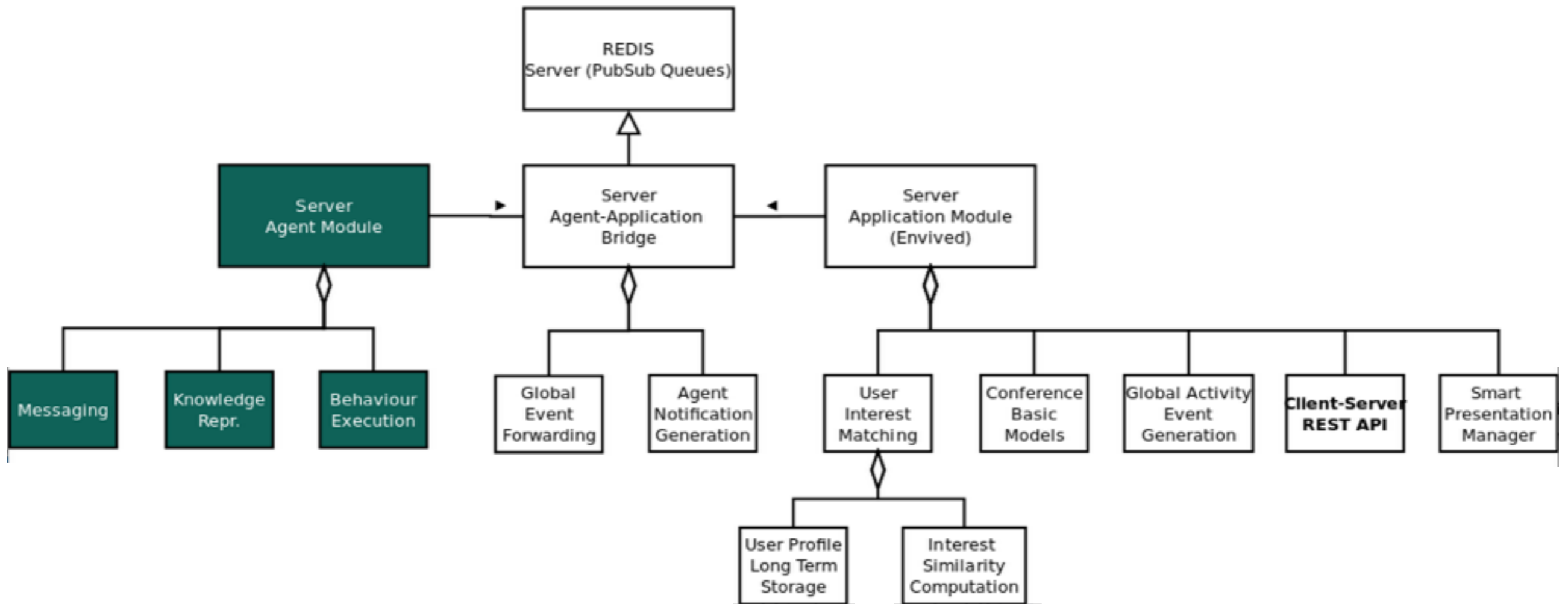
Elemente caracteristice ale aplicatiilor existente:

- vizualizarea profilelor celorlalti participanti
- comunicarea intre participanti
- calendar personalizat al evenimentului

Imbunatatire adusa de aplicatie:

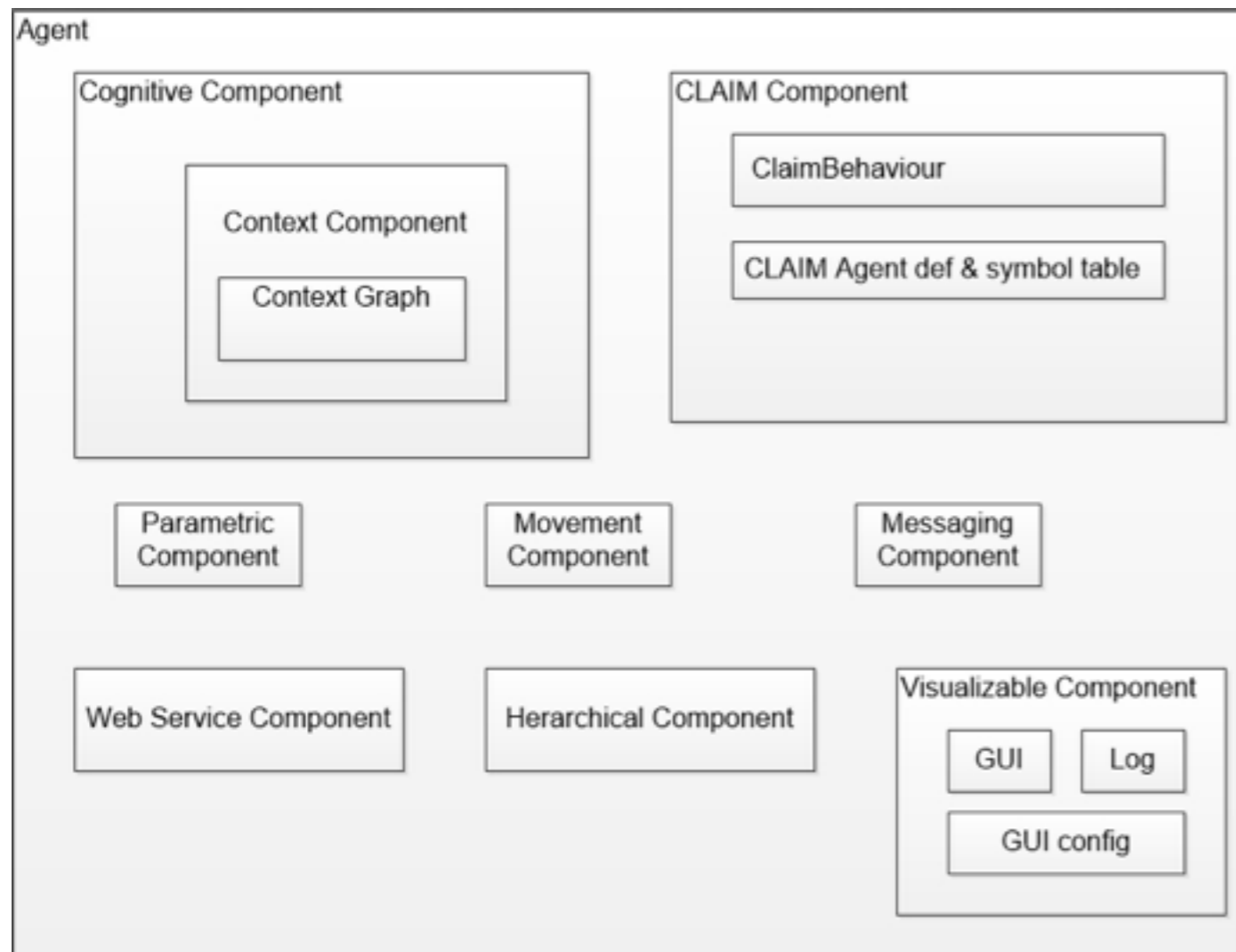
- integrarea contextului utilizatorului in aplicatie determinat de:
 - interactiunea intre agenti
 - pozitia utilizatorului (determinata cu ajutorul senzorilor)
 - actiunile agentilor / utilizatorului

Structura aplicatiei



- Reprezentarea bazei de cunostinte
 - implementata folosind Grafuri de Context (Context Graphs)
- Executia comportamentelor agentilor
 - executia codului S-CLAIM
- Comunicarea (intre agenti)

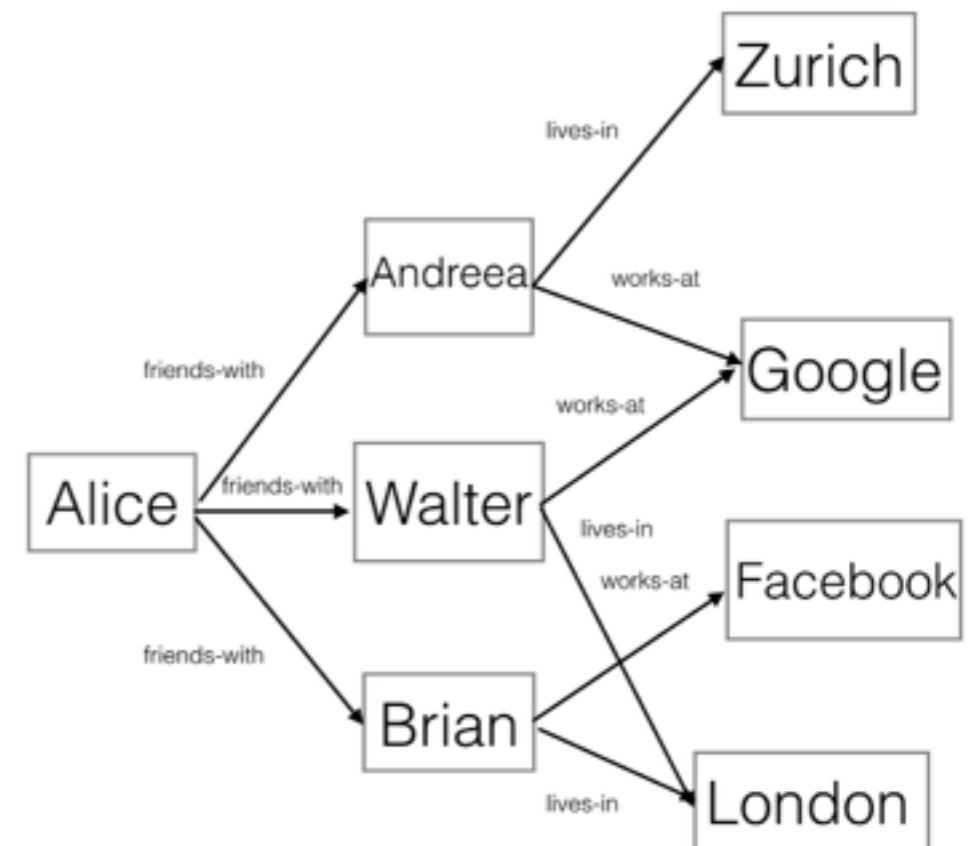
Implementarea agentului: Componenta cognitiva



- implementarea agentului folosind platforma tATAmI
- baza de cunostinte a agentului este implementata utilizand Context Graphs

CG:

- se poate realiza o buna modelare a mediului
- relationarea intre concepte prin muchii etichetate
- usurinta in efectuarea operatiilor



Septembrie 2014

Implementarea agentului: Comportamentul agentilor

- descris prin intermediul definițiilor agentilor (cod S-CLAIM)
- permite 4 tipuri de comportamente (initiale, reactive, proactive si ciclice)

```
question = Alice -friends-with-> ?Who -works-at->Google
(if (readK (struct knowledge pattern question))
  then
    (readK (struct knowledge ?Who -lives-> ?Where))
    (addK (struct knowledge Alice -goes-> ?Where))
  else
    (addK (struct knowledge Alice -goes-> Home))
)
```


Modelarea agentului:

- modelarea elementelor conferintei intr-o reprezentare specifica CG
- Elemente: spatiu, utilizator, prezentari, sesiuni, notificari

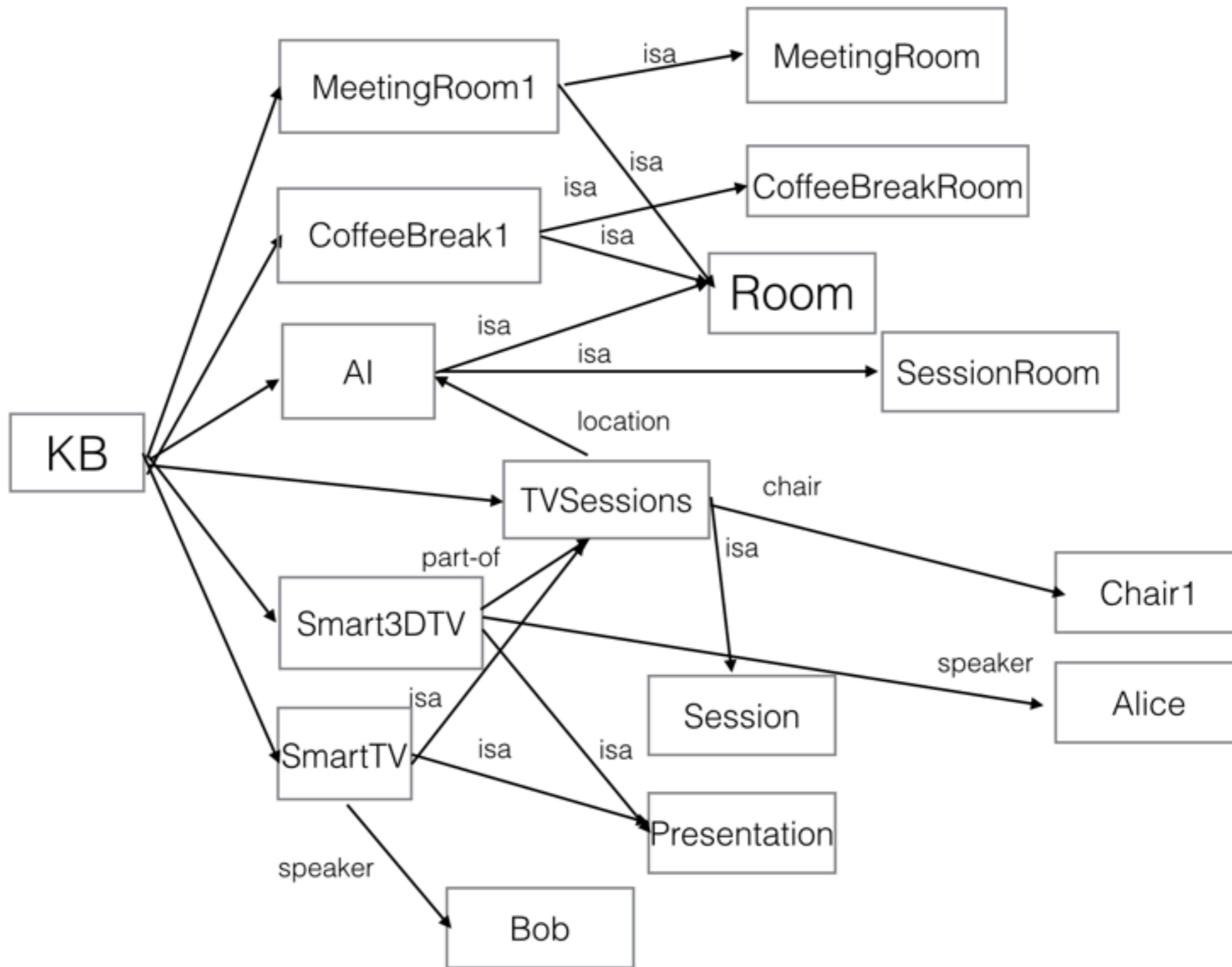
Exemplu proprietati utilizator:

- interest-> ExampleOfDomain
- affiliation-> ExampleOfDomain
- research-field -> ExampleOfDomain / Field
- attends-> PresentationName
- presents-> PresentationName
- chair-of-> SessionName
- is-in-> RoomName / Area (Speaker or Public Area)

CG permite descrierea relatiilor intre concepte

ex: SmartHousePresentation

- (-part-of-> SmartEnvironmentsSession -isa-> Session)
- isa-> Presentation



Exemple de situatii detectate

- inceperea unei prezentari
- prezenta participantilor
- prezenta prezentatorului
- prezenta participantilor intr-o intalnire ad-hoc
- prelungirea unei prezentari

folosind modelul descris anterior, putem modela o gama larga de conferinte

exemple de viitoare imbunatatiri ale aplicatiei:

- detectarea similaritatilor / intereselor participantilor

- integrarea senzorilor fizici

Va multumesc!

