A Rule-based Contextual Reasoning Platform for Ambient Intelligence Environments

Assaad Moawad[1], Antonis Bikakis[2], Patrice Caire[1], Gregory Nain[1], Yves Le Traon[1]

[1]University of Luxembourg, SnT

[2]Department of Information Studies
University College London

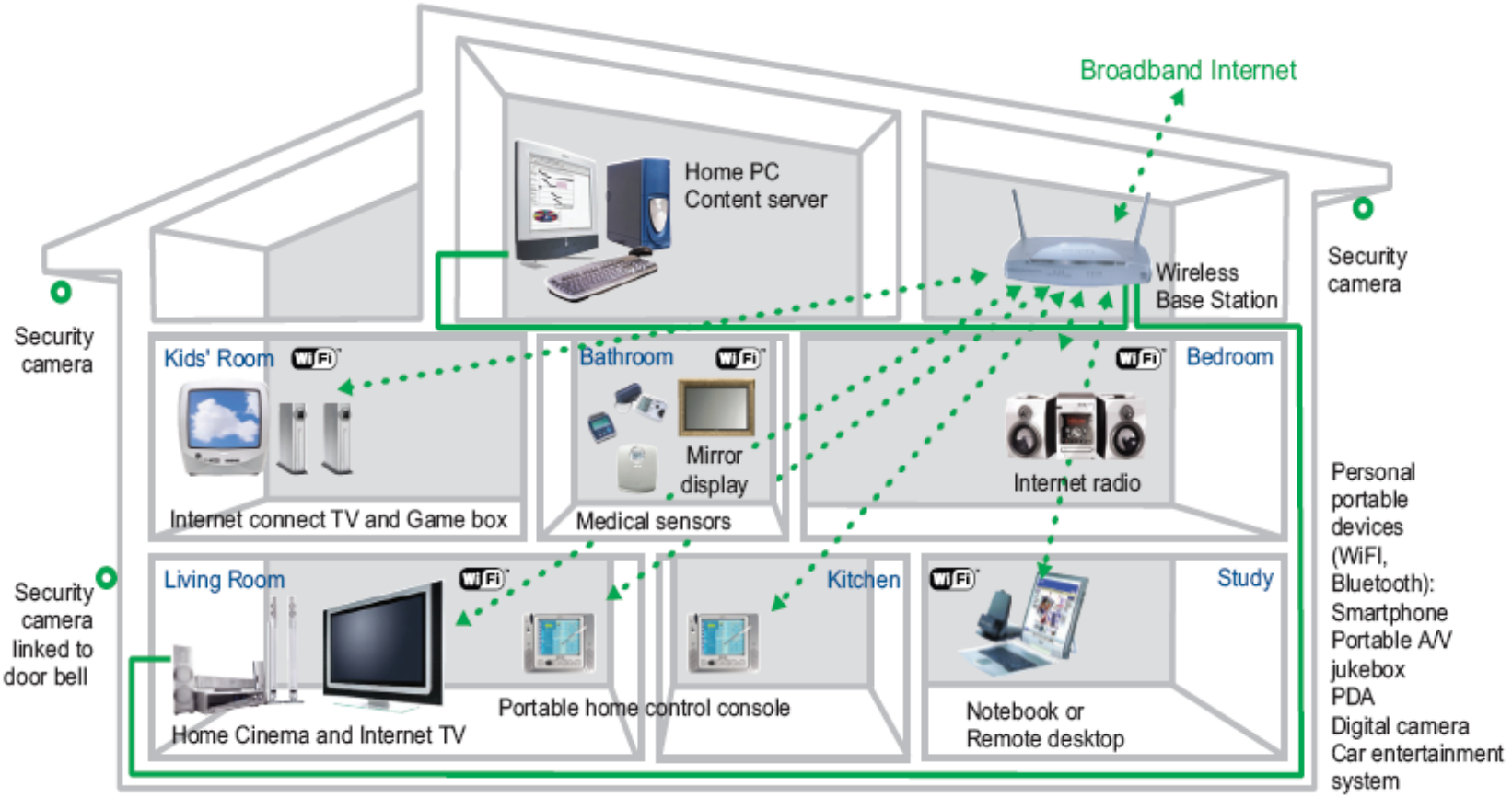
RuleML-2013, Seattle, USA, July 2013

Outline

- **Ambient Intelligence**
  - Goals & Requirements
  - Context & Contextual Reasoning
  - Example Scenario
- **R-CoRe**
  - Main Features
  - Reasoning model: CDL
  - Software Platform: Kevoree
  - Architecture
  - R-CoRe in action
  - Limitations & Ongoing Work

Ambient Intelligence

- **Goal**: Transform our living and working environments into smart spaces
- **Requirement**: Augment environments with sensing, computing, communication and reasoning capabilities



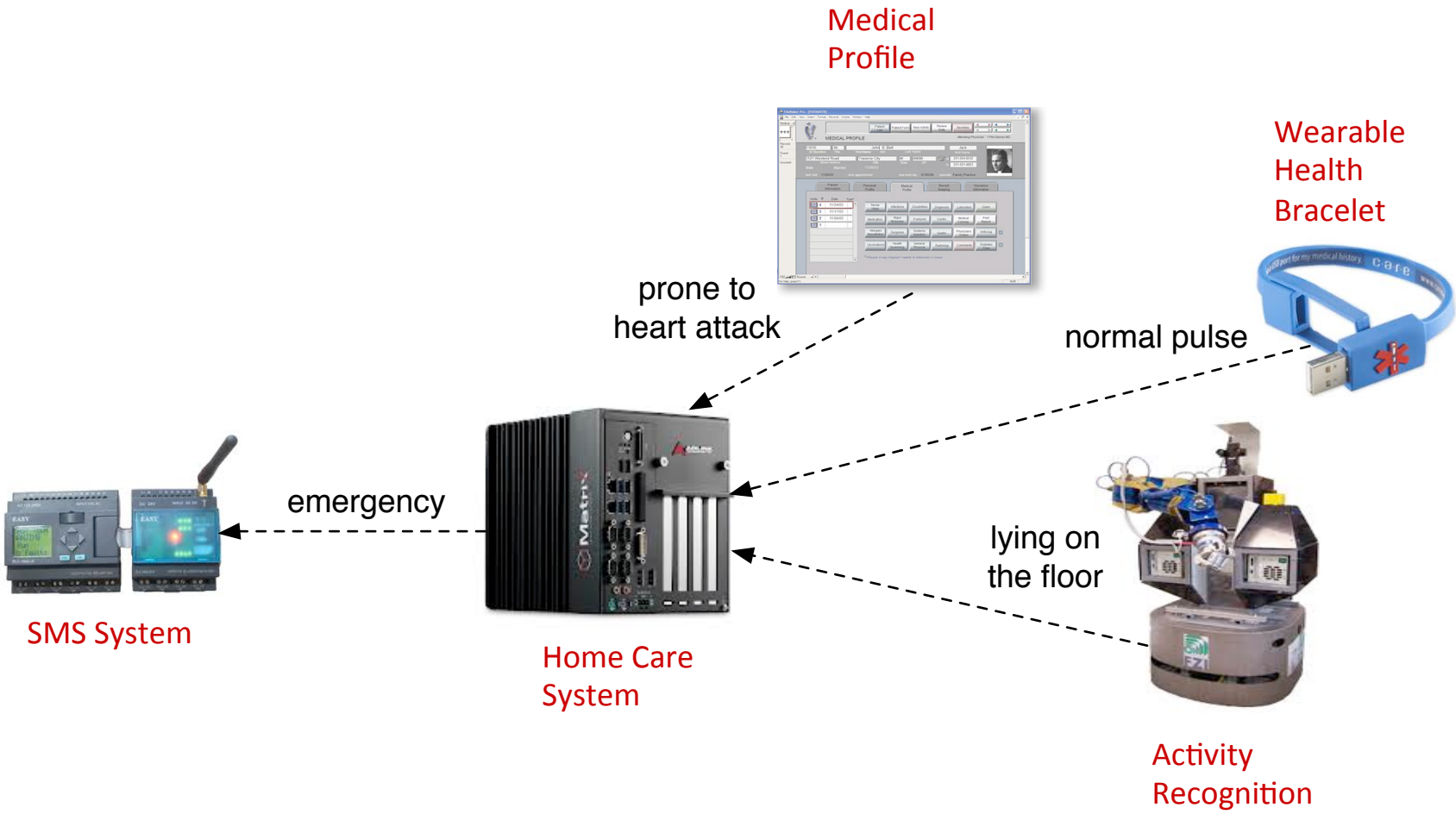
Context & Contextual Reasoning

Context is any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and application, including the user and applications themselves

[Dey and Abowd, 1999]

- Challenges of Contextual Reasoning
  - Imperfect context information
  - Heterogeneous entities
  - Highly dynamic and open environments
  - Distributed context information
  - Unreliable wireless communications
  - …restricted by the range of transmitters

Example Scenario



R-CoRe

- Main Features
  - **Distributed**
  - **Rule-based**
  - **Non-monotonic**
  - **Preference-based conflict resolution**
  - **Dynamic & Adaptive**

- Underlying technologies
  - **Contextual Defeasible Logic (CDL)**
    - a distributed version of Defeasible Logic
  - **Kevoree**
    - a s/w framework for Distributed Dynamically Adaptive Systems

Contextual Defeasible Logic (CDL)

- Combines elements of
  - **Defeasible Logic**
    - rule-based skeptical logic, which uses priorities among rules to resolve conflicts
  - **Multi-Context Systems**
    - logical formalizations of distributed contexts connected through a set of bridge rules, which enable information flow between contexts
    - Context: logical theory that models local knowledge of an agent

- Results
  - Argumentation Semantics (TKDE, 2010)
  - Proof Theory (TSMC-A, 2011)
  - Algorithms for distributed query evaluation (KAIS, 2011)

CDL – Representation Model

A Defeasible MCS C is a collection of contexts C_i

Each context C_i is a tuple (V_i, R_i, T_i)

- V_i : vocabulary used by C_i
- R_i : set of rules
- T_i : preference ordering on C

V_i : a set of literals of the form $a, \neg a$

CDL – Representation Model (cont'd)

Three types of rules in R_i

- Strict local rules

  $r_i^l : (c_i : a^1),\ldots, (c_i : a^{n-1}) \rightarrow (c_i : a^n)$

- Defeasible local rules

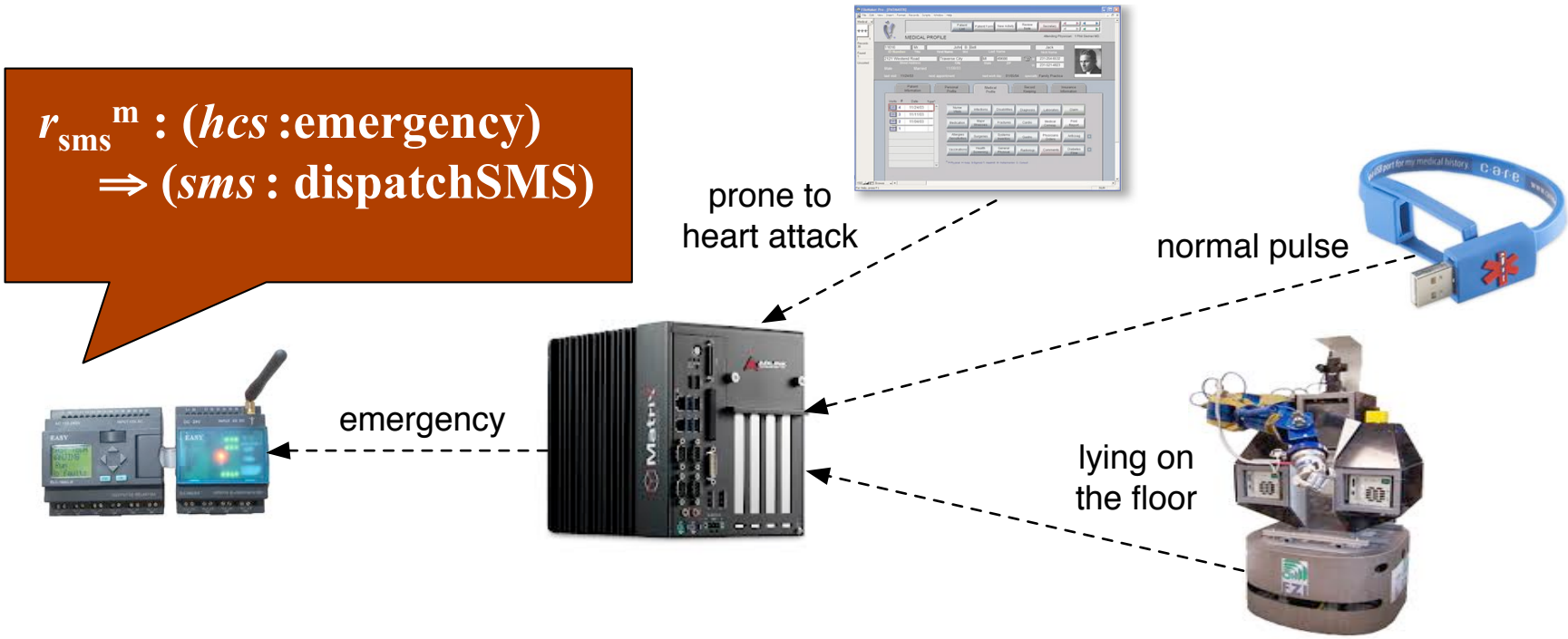
  $r_i^d : (c_i : a^1),\ldots, (c_i : a^{n-1}) \Rightarrow (c_i : a^n)$

- Mapping rules

  $r_i^m : (c_j : a^1),\ldots, (c_k : a^{n-1}) \Rightarrow (c_i : a^n)$

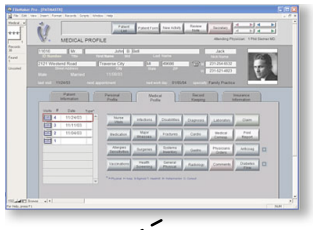
T_i is a partial preference ordering on C
modeled as a Directed Acyclic Graph

Example Scenario – in CDL terms



Example Scenario – in CDL terms

r_{sms}^{m} : (*hcs* :emergency)
⇒ (*sms* : dispatchSMS)



prone to heart attack

normal pulse

emergency

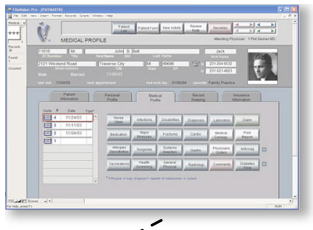
lying on the floor

r_{hcs}^{m1} : (*br* :normalPulse)
⇒ (*hcs* : ¬ emergency)
r_{hcs}^{m2} : (*arm* :lyingOnFloor), (*med* :proneToHA)
⇒ (*hcs* : emergency)

Example Scenario – in CDL terms

$r_{med}^{1} : \rightarrow (med : proneToHA)$

$r_{sms}^{m} : (hcs : emergency) \Rightarrow (sms : dispatchSMS)$



prone to heart attack

normal pulse

emergency

lying on the floor

$r_{hcs}^{m1} : (br : normalPulse) \Rightarrow (hcs : \neg emergency)$

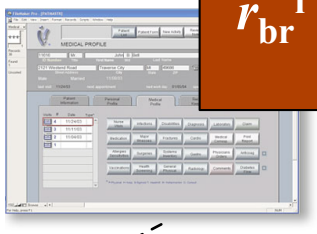
$r_{hcs}^{m2} : (arm : lyingOnFloor), (med : proneToHA) \Rightarrow (hcs : emergency)$

Example Scenario – in CDL terms

$r_{med}^{1} : \rightarrow (med : \text{proneToHA})$

$r_{br}^{1} : \rightarrow (br : \text{normalPulse})$

$r_{sms}^{m} : (hcs : \text{emergency}) \Rightarrow (sms : \text{dispatchSMS})$



prone to heart attack

normal pulse

emergency

lying on the floor

$r_{hcs}^{m1} : (br : \text{normalPulse}) \Rightarrow (hcs : \neg \text{emergency})$

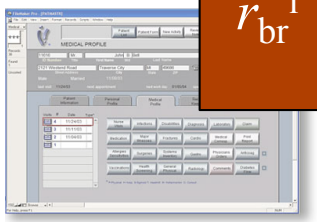
$r_{hcs}^{m2} : (arm : \text{lyingOnFloor}), (med : \text{proneToHA}) \Rightarrow (hcs : \text{emergency})$

Example Scenario – in CDL terms

$r_{med}^{1} : \rightarrow (med : \text{proneToHA})$

$r_{br}^{1} : \rightarrow (br : \text{normalPulse})$

$r_{sms}^{m} : (hcs : \text{emergency}) \Rightarrow (sms : \text{dispatchSMS})$



prone to heart attack

normal pulse

emergency

lying on the floor

$r_{hcs}^{m1} : (br : \text{normalPulse}) \Rightarrow (hcs : \neg \text{emergency})$

$r_{hcs}^{m2} : (arm : \text{lyingOnFloor}), (med : \text{proneToHA}) \Rightarrow (hcs : \text{emergency})$

$\mathbf{r_{arm}^{1} : \rightarrow (arm : lyingOnFloor)}$

Distributed Query Evaluation

- When a context receives a query for one of its local literals q
  - Evaluates answer based on local knowledge

    If not possible
  - Collects relevant information from other contexts through mappings
  - Checks applicability of rules *for* and *against* q
  - Evaluates answer based on

    - Applicable rules

    - Preferences

- Given two rules

  $r_{i1}^{m} : (c_j : a^1),\ldots, (c_k : a^{n-1}) \Rightarrow (c_i : a^n)$

  $r_{i2}^{m} : (c_{k+1} : a^1),\ldots, (c_l : a^{n-1}) \Rightarrow (c_i : \sim a^n)$

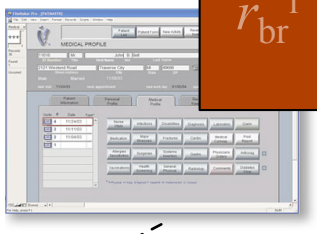
  - r_{i1}^{m} is "stronger" than r_{i2}^{m} if there is a context c_y in $c_{k+1},\ldots,c_l$ s.t. for all contexts c_x in $c_j,\ldots,c_k$, c_x is preferred to c_y according to $\mathbf{T}_i$ (there is a path from c_y to c_x in $\mathbf{T}_i$)

Example Scenario – query evaluation

$r_{med}^{1} : \rightarrow (med : \text{proneToHA})$

$r_{br}^{1} : \rightarrow (br : \text{normalPulse})$

$r_{sms}^{m} : (hcs : \text{emergency}) \Rightarrow (sms : \text{dispatchSMS})$



prone to heart attack

normal pulse

emergency

lying on the floor

$r_{hcs}^{m1} : (br : \text{normalPulse}) \Rightarrow (hcs : \neg \text{emergency})$

$r_{hcs}^{m2} : (arm : \text{lyingOnFloor}), (med : \text{proneToHA}) \Rightarrow (hcs : \text{emergency})$

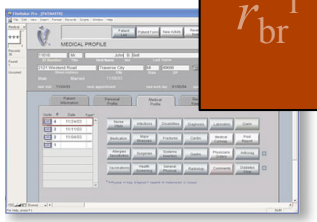
$r_{arm}^{1} : \rightarrow (arm : \text{lyingOnFloor})$

Example Scenario – query evaluation

$r_{med}^{1} : \rightarrow (med : \text{proneToHA})$

$r_{br}^{1} : \rightarrow (br : \text{normalPulse})$

$r_{sms}^{m} : (hcs : \textbf{emergency}) \Rightarrow (sms : \text{dispatchSMS})$



prone to heart attack

normal pulse

emergency

lying on the floor

$r_{hcs}^{m1} : (br : \text{normalPulse}) \Rightarrow (hcs : \neg \textbf{emergency})$

$r_{hcs}^{m2} : (arm : \text{lyingOnFloor}), (med : \text{proneToHA}) \Rightarrow (hcs : \textbf{emergency})$

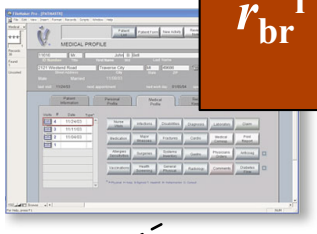
$r_{arm}^{1} : \rightarrow (arm : \text{lyingOnFloor})$

Example Scenario – query evaluation

$r_{med}^{1} : \rightarrow (med : \text{proneToHA})$

$r_{br}^{1} : \rightarrow (br : \text{normalPulse})$

$r_{sms}^{m} : (hcs : \text{emergency}) \Rightarrow (sms : \text{dispatchSMS})$



prone to heart attack

normal pulse

emergency

lying on the floor

$r_{hcs}^{m1} : (br : \text{normalPulse}) \Rightarrow (hcs : \neg \text{emergency})$

$r_{hcs}^{m2} : (arm : \text{lyingOnFloor}), (med : \text{proneToHA}) \Rightarrow (hcs : \text{emergency})$

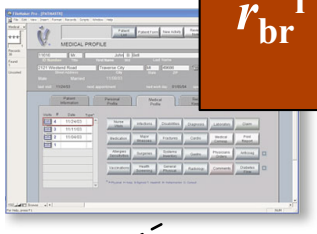
$r_{arm}^{1} : \rightarrow (arm : \text{lyingOnFloor})$

Example Scenario – query evaluation

$r_{med}^{1} : \rightarrow (med : \text{proneToHA})$

$r_{br}^{1} : \rightarrow (br : \text{normalPulse})$

$r_{sms}^{m} : (hcs : \text{emergency}) \Rightarrow (sms : \text{dispatchSMS})$



prone to heart attack

normal pulse

emergency

lying on the floor

$r_{hcs}^{m1} : (br : \text{normalPulse}) \Rightarrow (hcs : \neg \text{emergency})$

$r_{hcs}^{m2} : (arm : \text{lyingOnFloor}), (med : \text{proneToHA}) \Rightarrow (hcs : \text{emergency})$

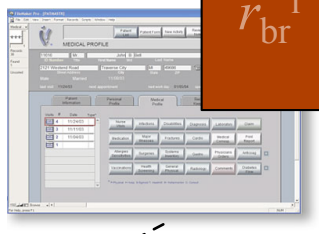
$r_{arm}^{1} : \rightarrow (arm : \text{lyingOnFloor})$

Example Scenario – query evaluation

$r_{med}^{1} : \rightarrow (med : proneToHA)$

$r_{br}^{1} : \rightarrow (br : normalPulse)$

$r_{sms}^{m} : (hcs : emergency) \Rightarrow (sms : dispatchSMS)$



prone to heart attack

normal pulse

emergency

lying on the floor

$r_{hcs}^{m1} : (br : normalPulse) \Rightarrow (hcs : \neg emergency)$

$r_{hcs}^{m2} : (arm : lyingOnFloor), (med : proneToHA) \Rightarrow (hcs : emergency)$

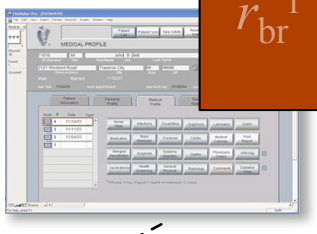
$r_{arm}^{1} : \rightarrow (arm : lyingOnFloor)$

Example Scenario – query evaluation

$r_{med}^{1} : \rightarrow$ **(*med* : proneToHA)**

$r_{br}^{1} : \rightarrow$ (*br* : normalPulse)

r_{sms}^{m} : (*hcs* :emergency)
$\Rightarrow$ (*sms* : dispatchSMS)



prone to heart attack

normal pulse

emergency

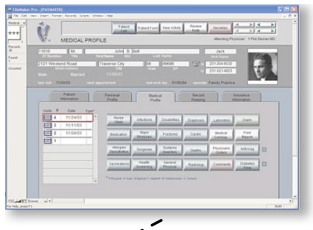
lying on the floor

r_{hcs}^{m1} : (*br* :normalPulse)
$\Rightarrow$ (*hcs* : $\neg$ emergency)
r_{hcs}^{m2} : **(*arm* :lyingOnFloor), (*med* :proneToHA)**
$\Rightarrow$ (*hcs* : emergency)

$r_{arm}^{1} : \rightarrow$ **(*arm* : lyingOnFloor)**

Example Scenario – query evaluation

r_{sms}^{m} : (*hcs* :emergency)
⇒ (*sms* : dispatchSMS)



prone to heart attack

normal pulse

emergency

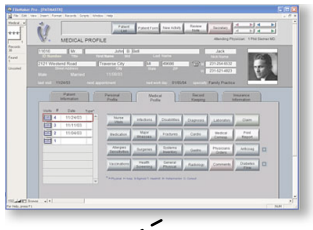
lying on the floor

r_{hcs}^{m1} : (*br* :normalPulse)
⇒ (*hcs* : ¬ emergency)
r_{hcs}^{m2} : (*arm* :lyingOnFloor), (*med* :proneToHA)
⇒ (*hcs* : emergency)

T_{hcs}=[*med*,*arm*,*br*]

Example Scenario – query evaluation

r_{sms}^{m} : (*hcs* :emergency)
⇒ (*sms* : dispatchSMS)



prone to heart attack

normal pulse

emergency

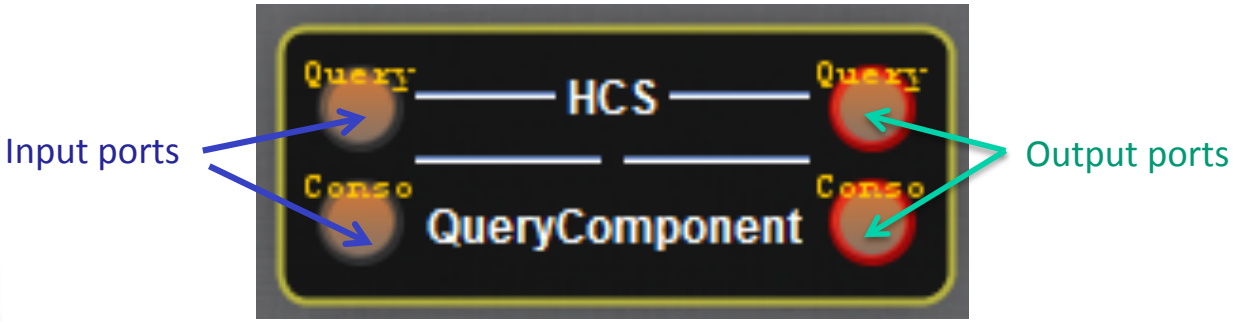
lying on the floor

r_{hcs}^{m1} : (*br* :normalPulse)
⇒ (*hcs* : ¬ emergency)

r_{hcs}^{m2} : (*arm* :lyingOnFloor), (*med* :proneToHA) T_{hcs}=[*med*,*arm*,*br*]
⇒ (*hcs* : emergency)

Kevoree

- **Open source project available at: www.kevoree.org**
  - Enables distributed reconfigurable software development
  - Any sensor, software application, web service can be represented as a **component** (with I/O) in Kevoree
  - The set of services/applications offered by a single entity (e.g. device) is represented as a Kevoree **node**
  - **Channels** represent different types of communication among components (TCP/IP, email, SMS, etc.)

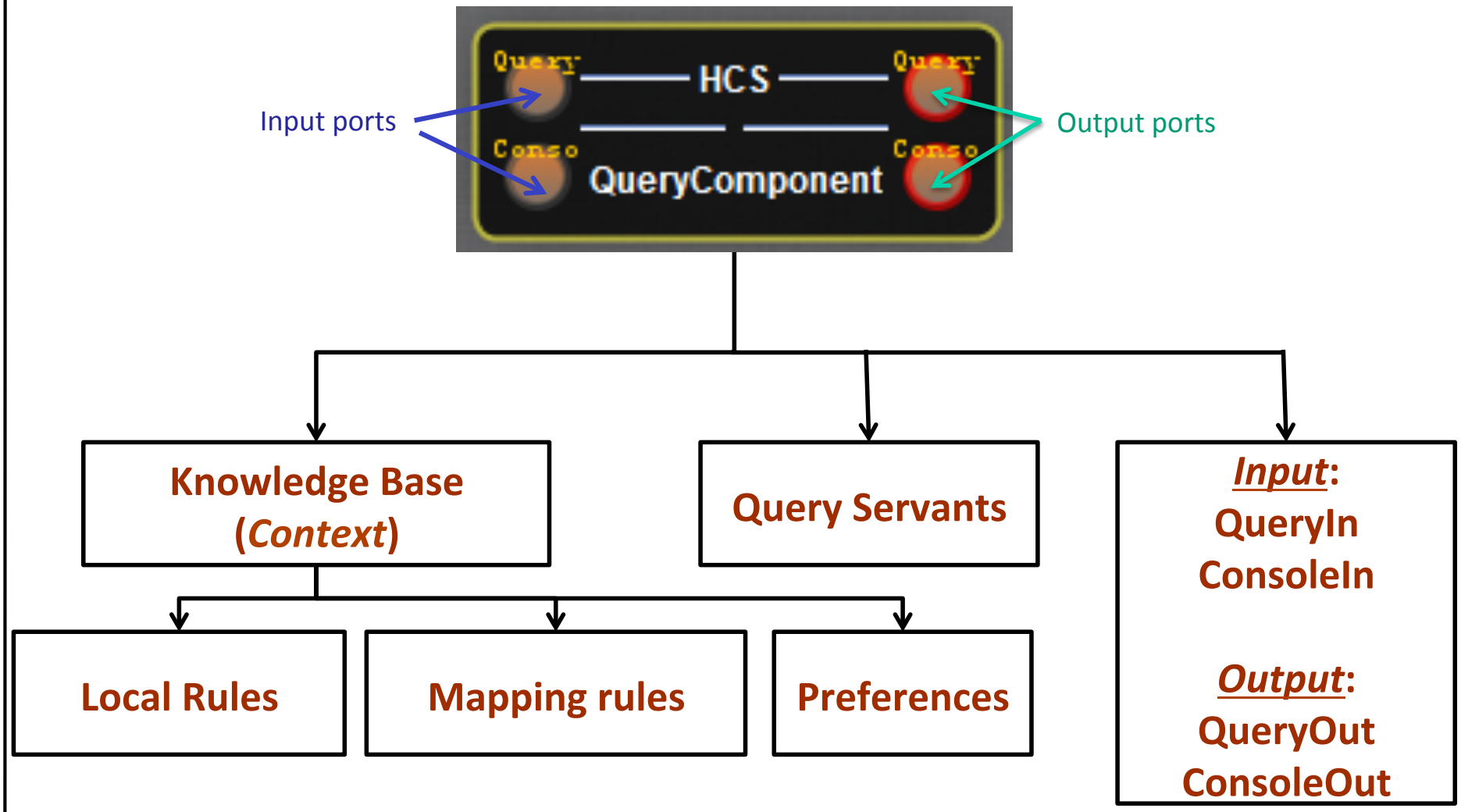


A Kevoree component

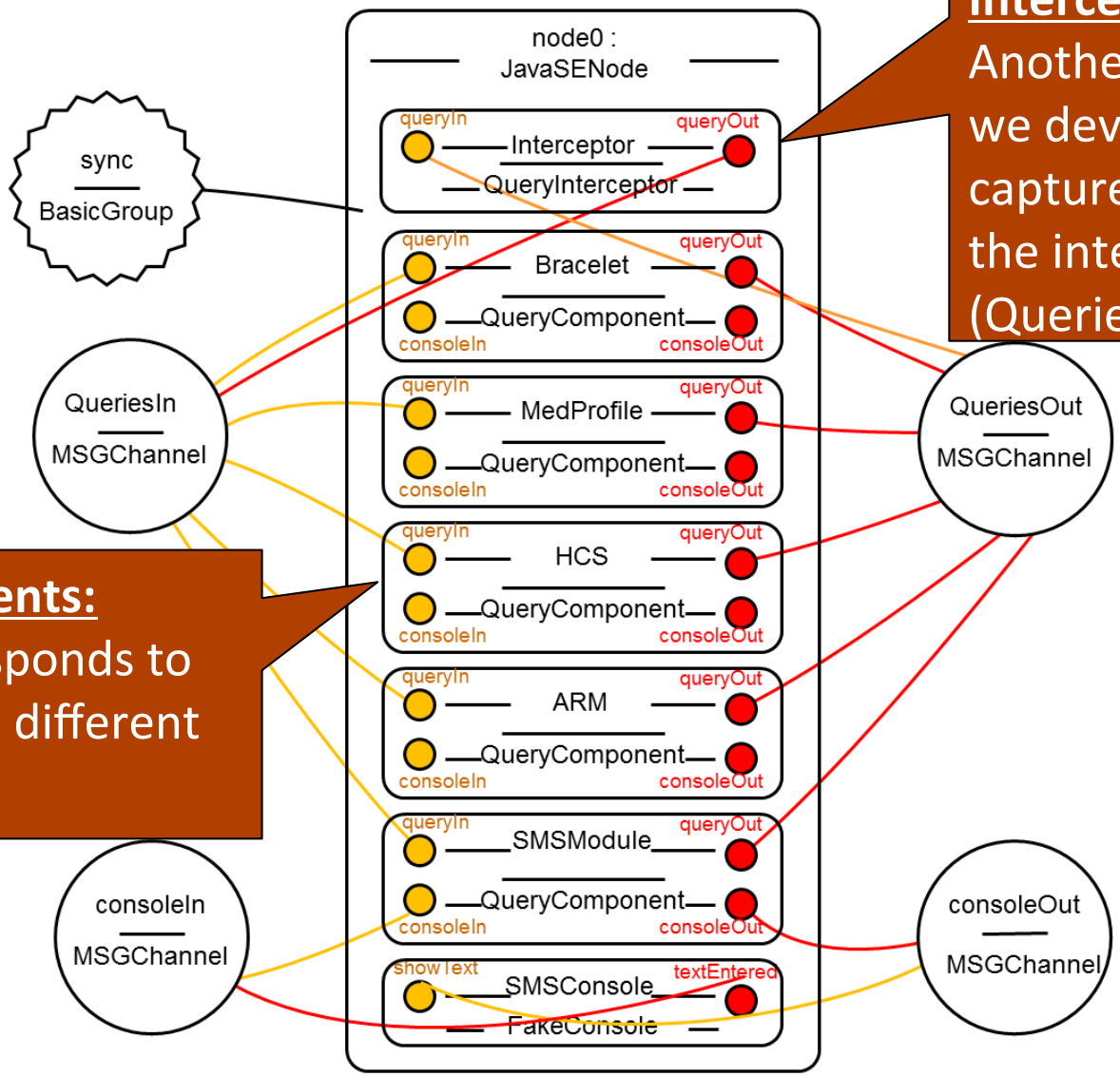
Kevoree in R-CoRe

- Each entity (mobile computing device) is implemented as a Kevoree node.
- Each context is implemented as a Kevoree component.
- Kevoree channels enable exchange of information (messages) between different components.
- Kevoree's adaptive and auto-discovery capabilities enable detecting new nodes and adapting to any context changes.

R-CoRe Architecture



Example Scenario - in R-CoRe terms



Interceptor:
Another component we developed to capture and display all the interactions (Queries/responses)

Query components:
Each one corresponds to the context of a different entity

Example Scenario - in R-CoRe terms

File Name	File contents
smsModuleKB.txt	M1: (hcs:emergency) $\rightarrow$ (sms:dispatchSMS)
BraceletKB.txt	L1: $\rightarrow$ (br:normalPulse)
MedProfileKB.txt	L1: $\rightarrow$ (med:proneToHA)
ArmKB.txt	L1: $\rightarrow$ (arm:lyingOnFloor)
HCSKB.txt	M1: (br:normalPulse) $\Rightarrow \neg$(hcs:emergency) M2: (arm:lyingOnFloor), (med:proneToHA) $\Rightarrow$ (has:emergency)
HCSPref.txt	med, arm, br

Rule bases and preferences in the example scenario

R-CoRe Limitations & Ongoing Work

- Components have limited memory, computation and power resources.
  - Limited Knowledgebase to 500 literals and rules.
  - Time-out: 10 seconds.
  - **Not really a limitation, just trying to be realistic!**
  - **Working on configurable components**
- Non-overlapping vocabularies, no common knowledge
  - **CDL extension to enable different contexts use common terms**
  - **R-CoRe extension using the *groups* feature of Kevoree**
- Top-down algorithms do not fit well with the needs of AmI
  - **Developing bottom-up reactive algorithms for CDL**
- Works only with high-level context predicates
  - **Integrating CEP methodology to reason with low-level sensor data**
- Not yet tested in real environments
  - **Planning to do tests at the IoT lab of SnT**

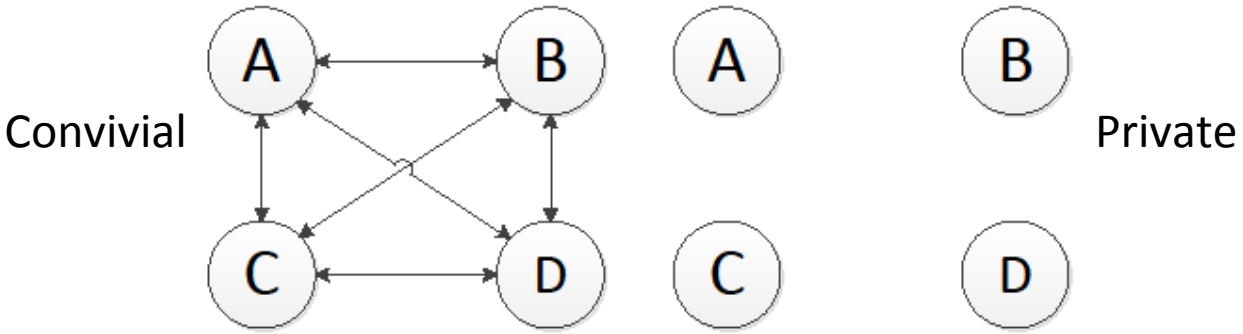
Summing up

- **R-CoRe** (today)
  - Rule-based Contextual Reasoning Platform for AmI
  - Developed on top of Kevoree
  - Implements the nonmonotonic reasoning model of CDL
- **R-CoRe demo** (Thursday)
  - Check also the demo R-CoRe paper and presentation for more details about the demonstration.
  - You can download the demo and test it yourself from https://github.com/securityandtrust/ruleml13

We would really appreciate your feedback!!!

The CoPAInS Project

- Conviviality & Privacy in Ambient Intelligence Systems*
  - Tradeoff between conviviality and privacy
  - Conviviality: sharing information with all
  - Privacy: keep local knowledge private



*Supported by the National Research Fund, Luxembourg (I2R-SER-PFN-11COPA)

Fonds National de la Recherche Luxembourg

A Rule-based Contextual Reasoning Platform for Ambient Intelligence Environments