



2011 IEEE INTERNATIONAL CONFERENCE  
ON INTELLIGENT COMPUTER COMMUNICATION  
AND PROCESSING

# CAPIM: A Context-Aware Platform using Integrated Mobile Services

Ciprian Dobre, Flavius Manea, Valentin Cristea

University POLITEHNICA of Bucharest, Romania

Emails: [flavius.manea@cti.pub.ro](mailto:flavius.manea@cti.pub.ro), [{ciprian.dobre, valentin.cristea}@cs.pub.ro](mailto:{ciprian.dobre, valentin.cristea}@cs.pub.ro)

# OUTLINE

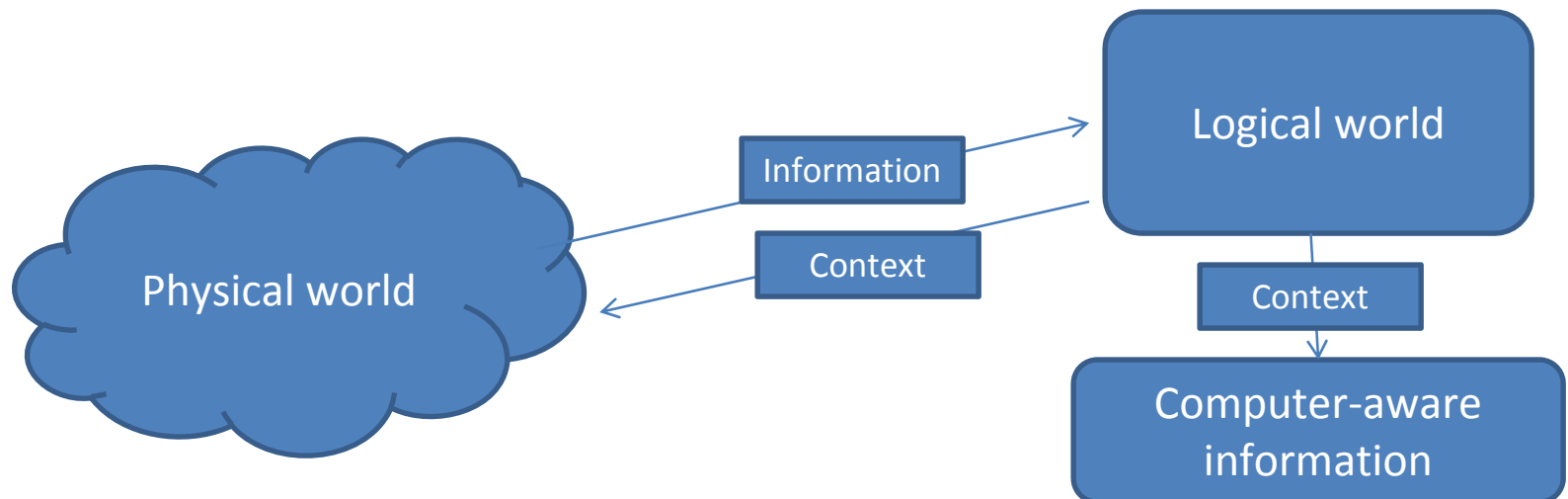
---

- Scope and motivation
  - Context – awareness
- CAPIM platform
  - Architecture
  - Pilot implementation
- Conclusions and future work



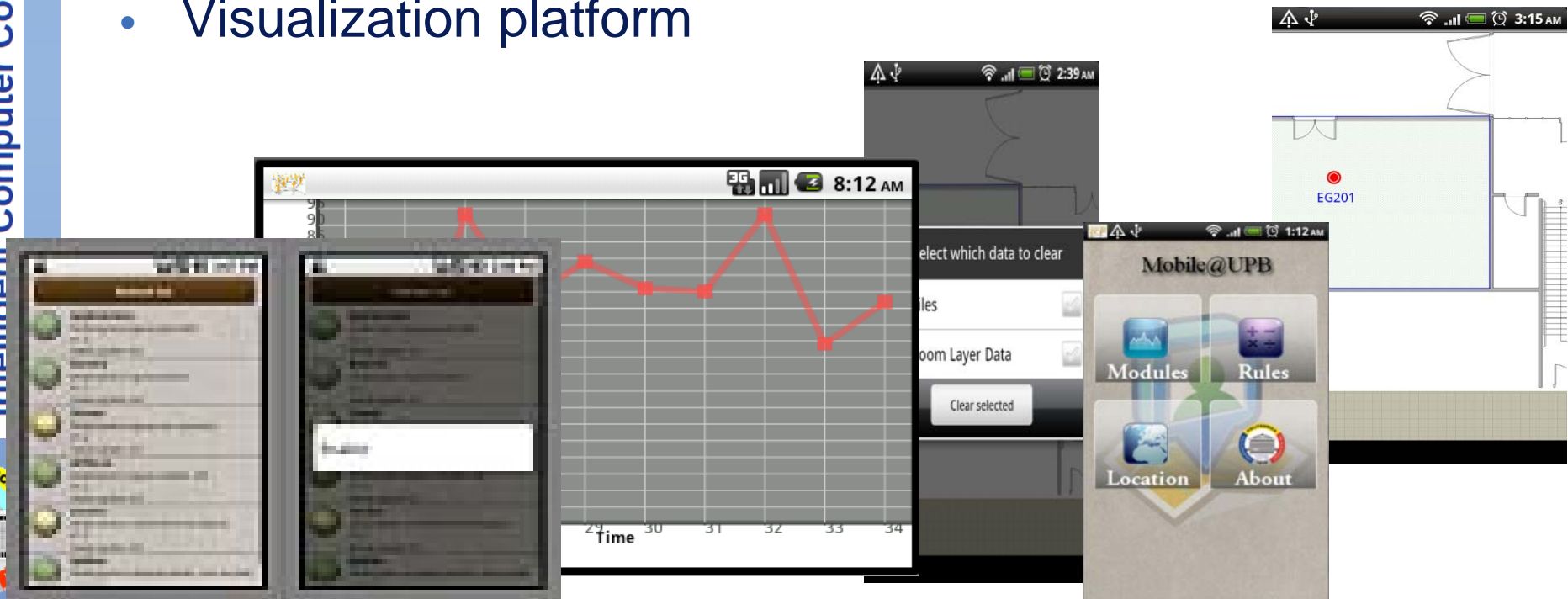
# Context - awareness

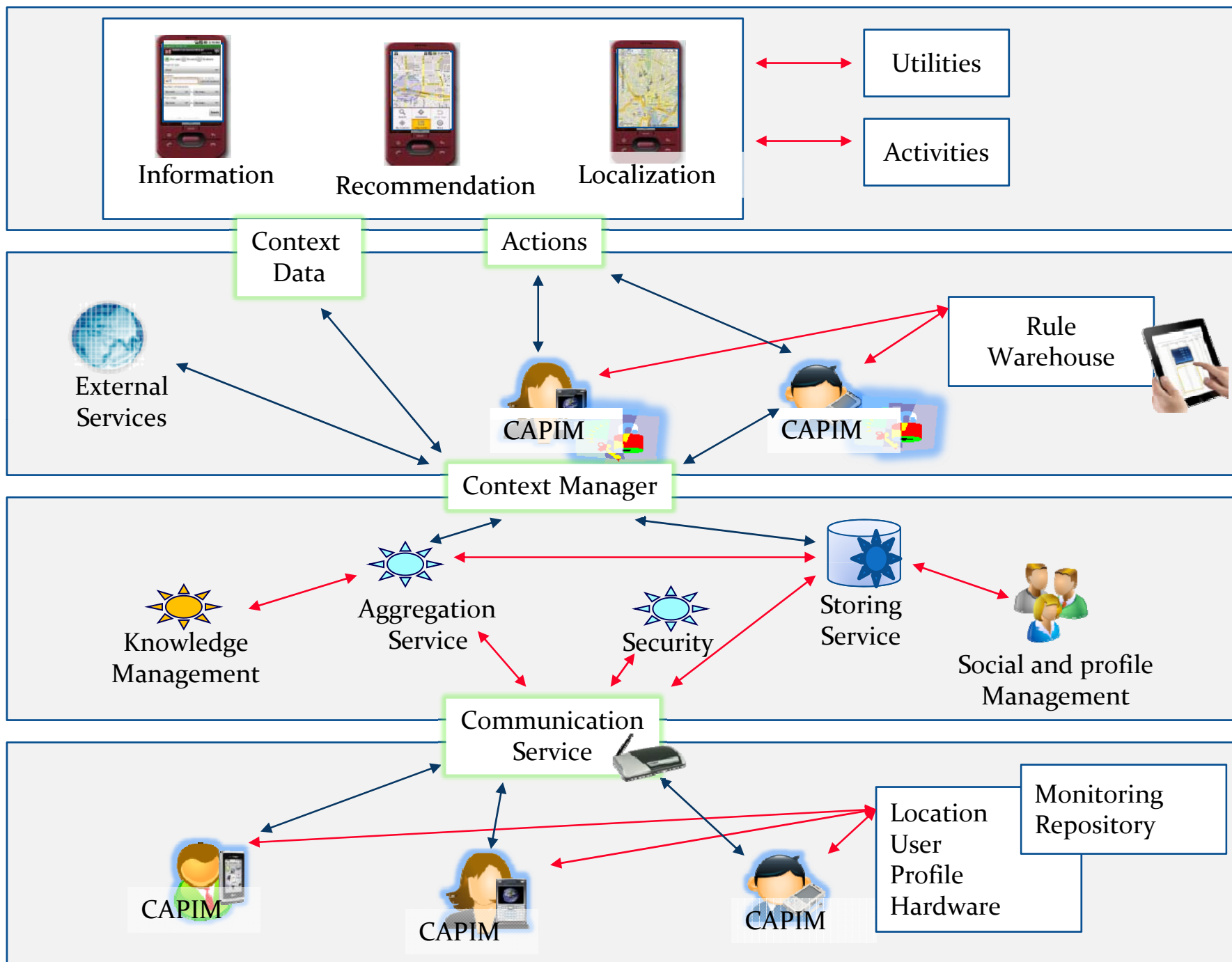
- *Context* : Information that can be used to characterize the situation of an entity
- *Context – awareness*: A property of a system that uses context to provide relevant information and/or service to the user



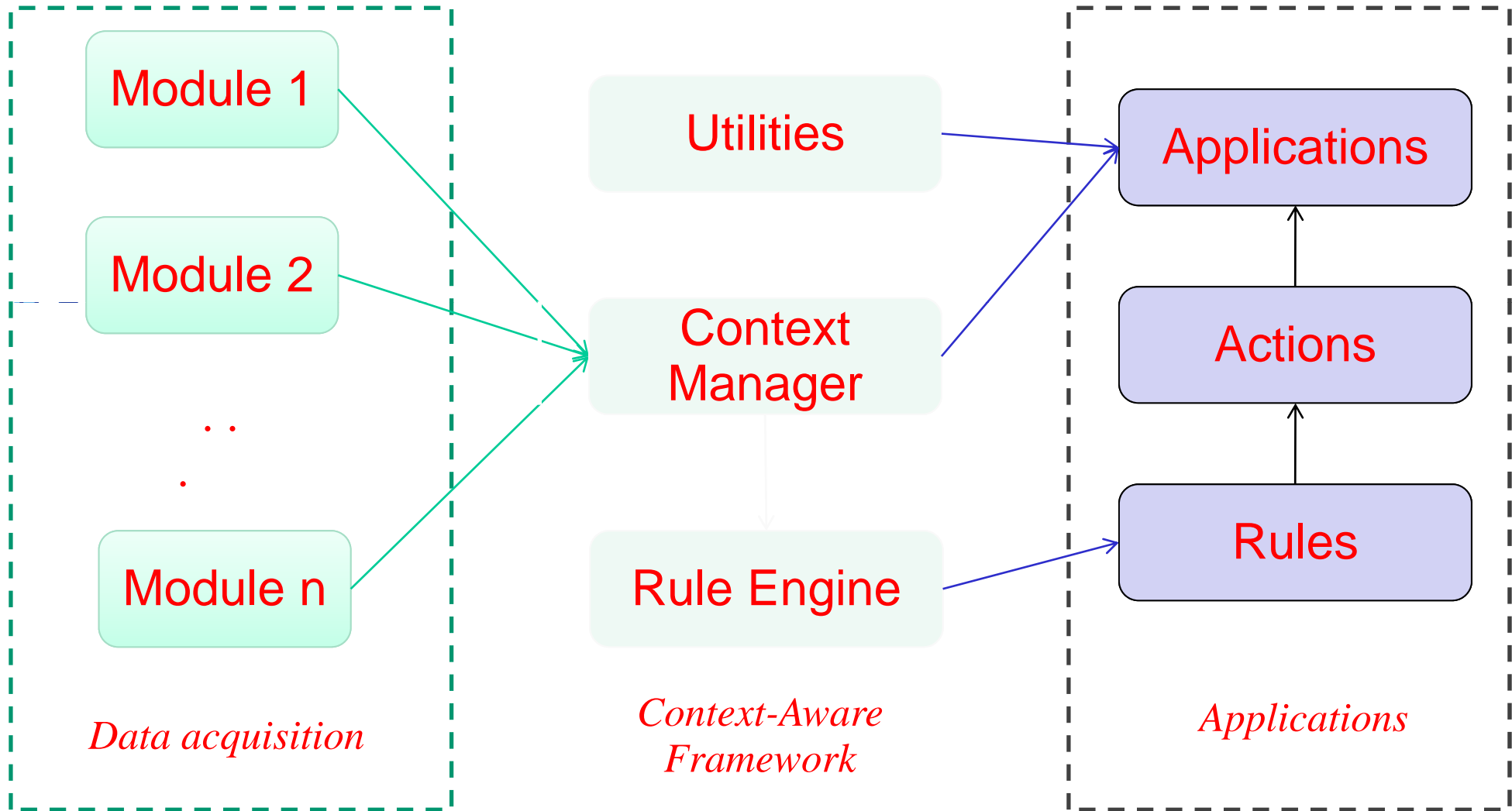
# CAPIM: Context-Aware Platform using Integrated Mobile services

- Platform to support construction of context-aware applications
- Services for collecting, storing, aggregation of context
- Context-enabled rule execution engine
- Visualization platform

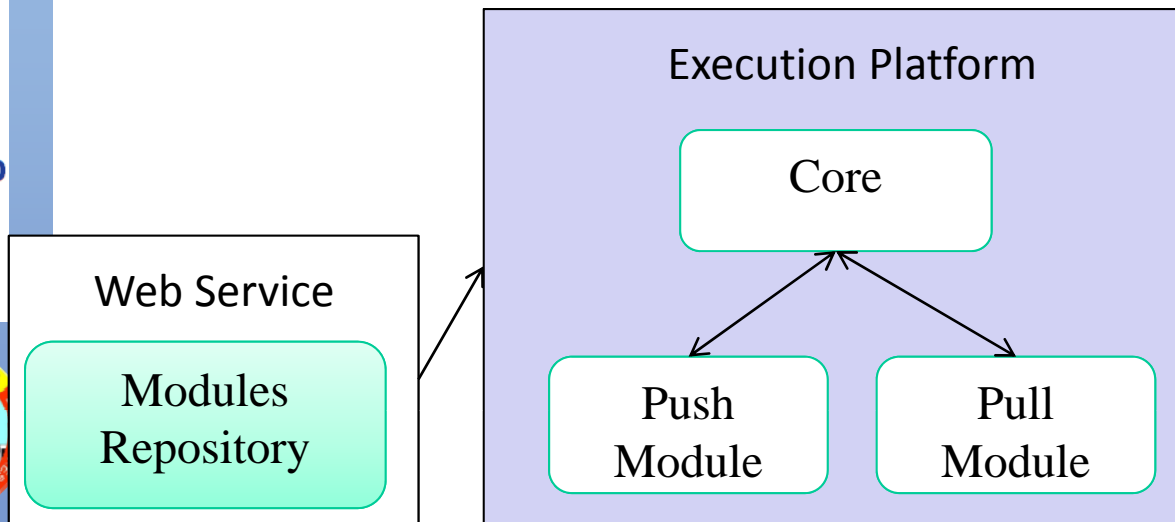
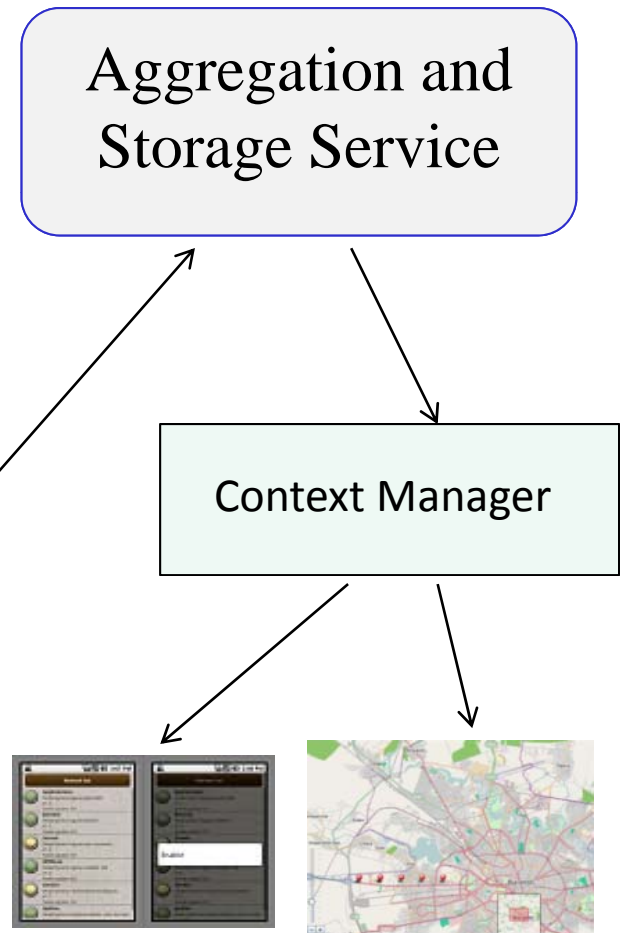
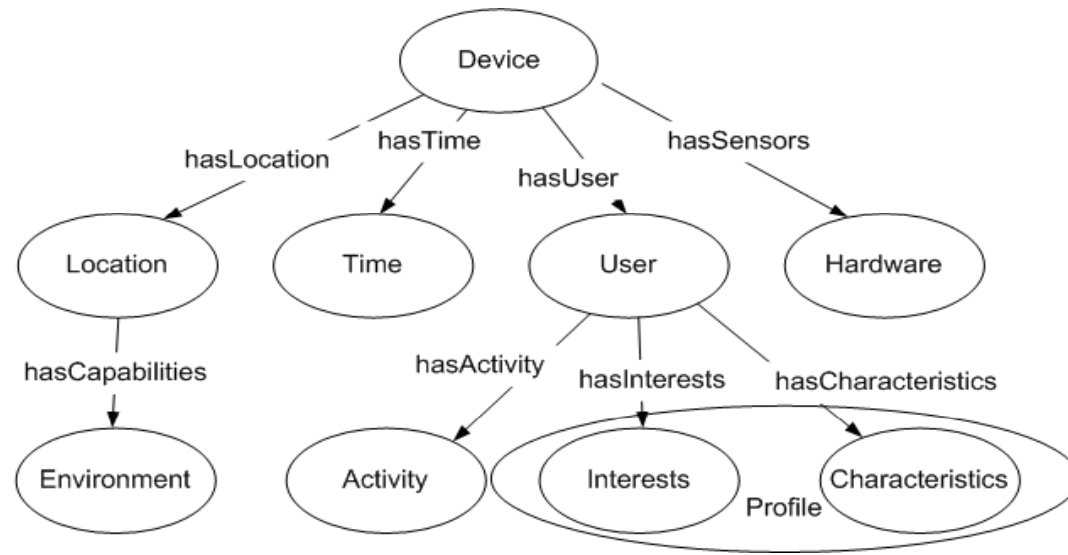




# Context framework



# Context Monitoring



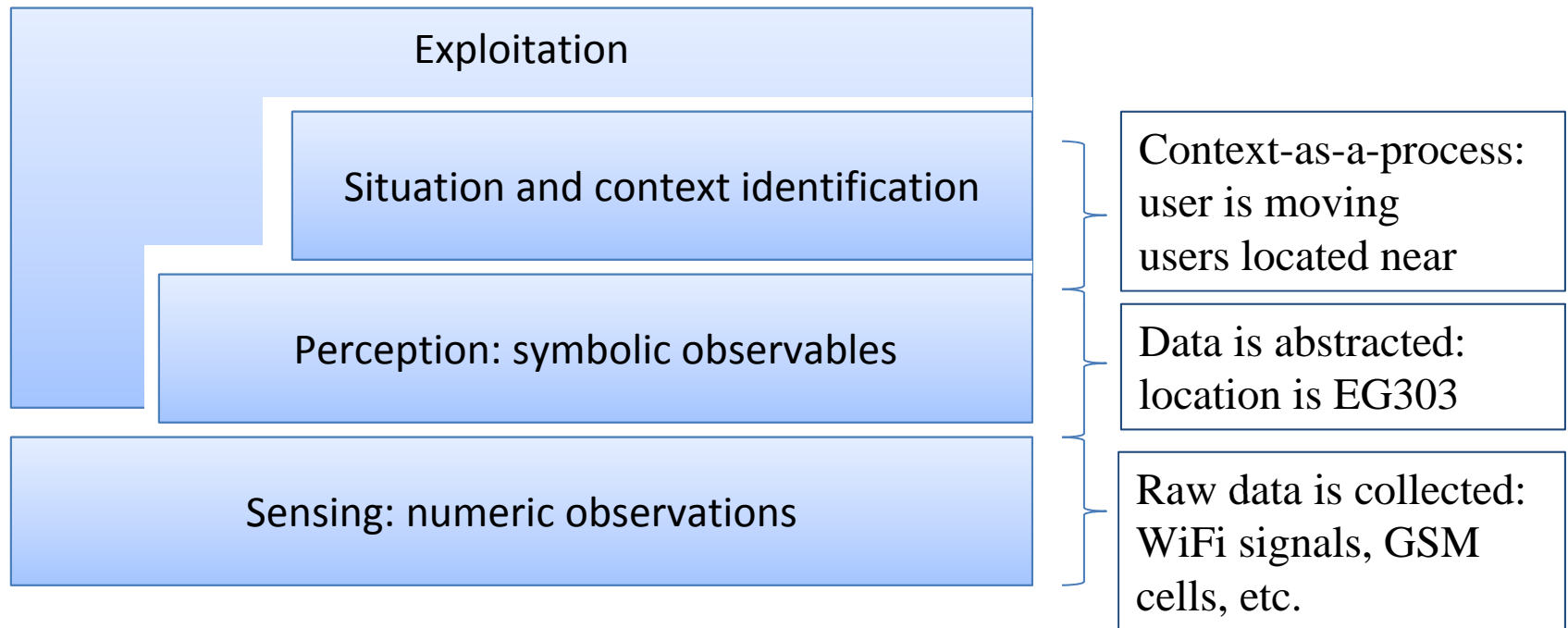
# Context acquisition modules

- **Push**
  - The Context Manager is notified by the module when the context changes and the corresponding values are updated
- **Pull**
  - Information from a pull module has a limited life span. When this time period is exceeded the Context Manager will try to update the context data.





# The levels of abstraction

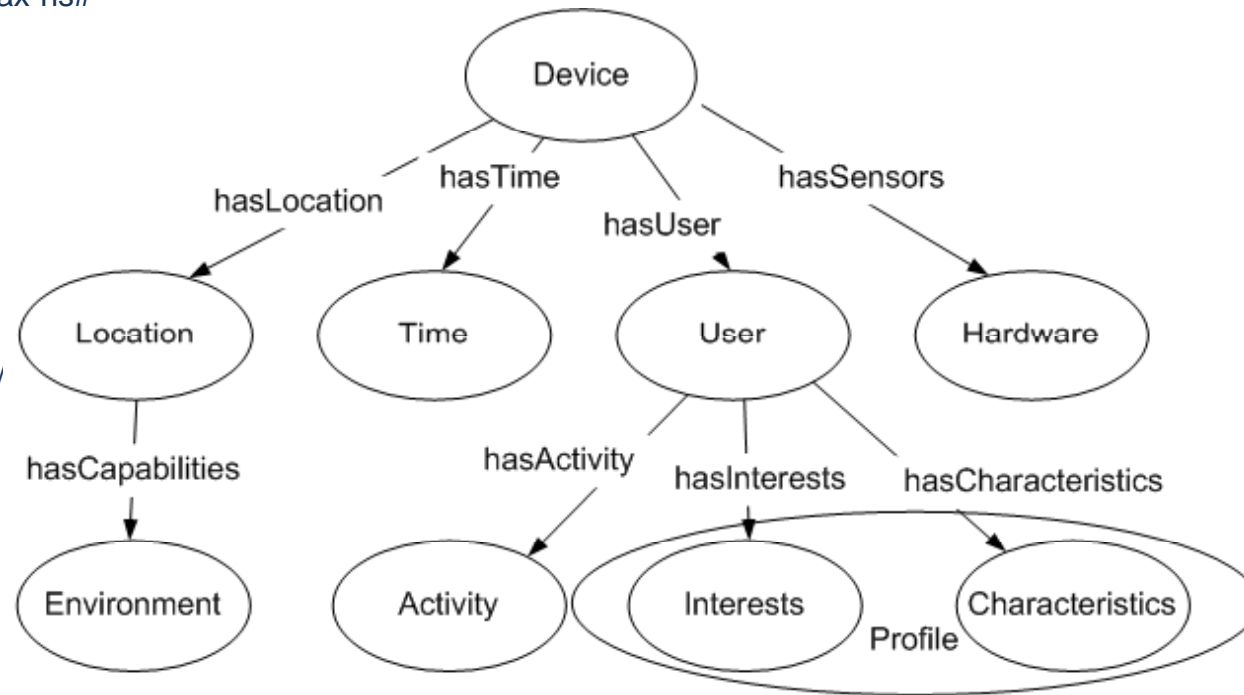


Taking the output “the user is running” and “the user has a high pulse” for example, determine a context such as “the user is jogging”.

Considering output “user under the car”, determine if the user was hit by the car or just inspecting its engine.

# Data model...

```
<?xml version="1.0" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:org="http://www.w3.org/ns/org#">
  <foaf:Person rdf:about="andreea.starparu">
    <org:memberOf rdf:resource="Gr341C3"/>
    <foaf:name>Andreea Starparu</foaf:name>
    <foaf:nick>andreea.starparu</foaf:nick>
    <foaf:interest>Semantic_Web</foaf:interest>
    <foaf:interest>Distributed Systems</foaf:interest>
    <rdfs:subClassOf rdf:resource="prezentare_licenta"/>
    <wail:location>
      <geo:Point>
        <geo:lat>47.235</geo:lat>
        <geo:long>25.581</geo:long>
      </geo:Point>
    </wail:location>
  </foaf:Person>
  <ical:vevent rdf:about="prezentare_licenta"/>
    <ical:summary>thesis presentation</ical:summary>
    <ical:dtstart rdf:datatype="xsd:data">2011-07-11</ical:dtstart>
    <ical:dtend rdf:datatype="xsd:data">2011-07-15</ical:dtend>
    <ical:location>
      <geo:Point>
        <geo:lat>47.235</geo:lat>
        <geo:long>25.581</geo:long>
      </geo:Point>
    </ical:location>
  </ical:vevent>
</rdf:RDF>
```



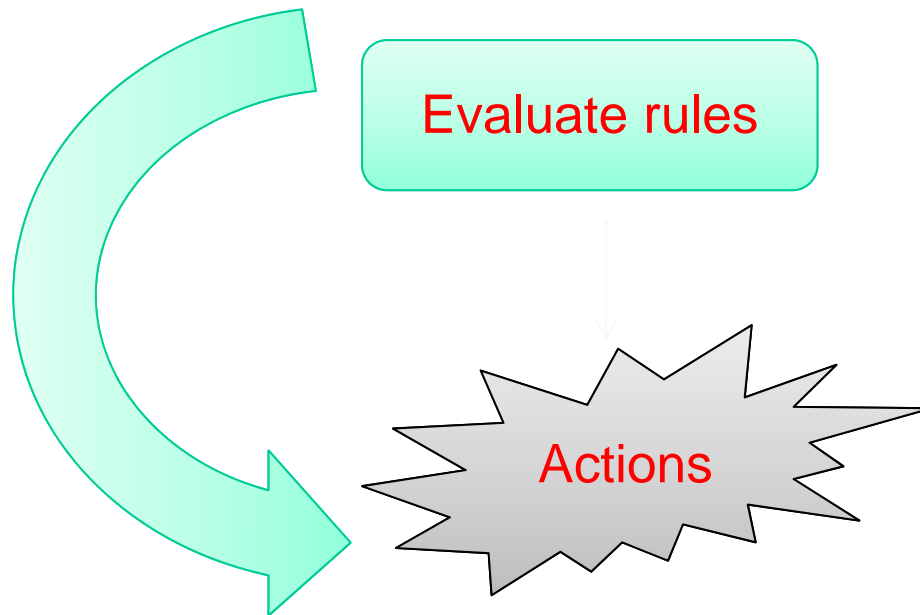
**Semantic world** ←

↑ **Logical world**

**Physical world**

# The rule engine

## *Context Service Routine*



Context updates are triggered by notifications or validity expirations

Only the rules which are affected by context changes are evaluated

An action is executed when a rule passes the evaluation

# Rules

Context  
Conditions



Action

Conditions are defined using context parameters (an enumeration of keys available as environment variables) and some predefined operators which will help us to build expressions

Each action is defined as a autonomous application

*Rule = Rule OR Rule | Rule AND Rule | RuleImplementation*

Context  
Parameter

Values



Operator



True  
False

# A rule example

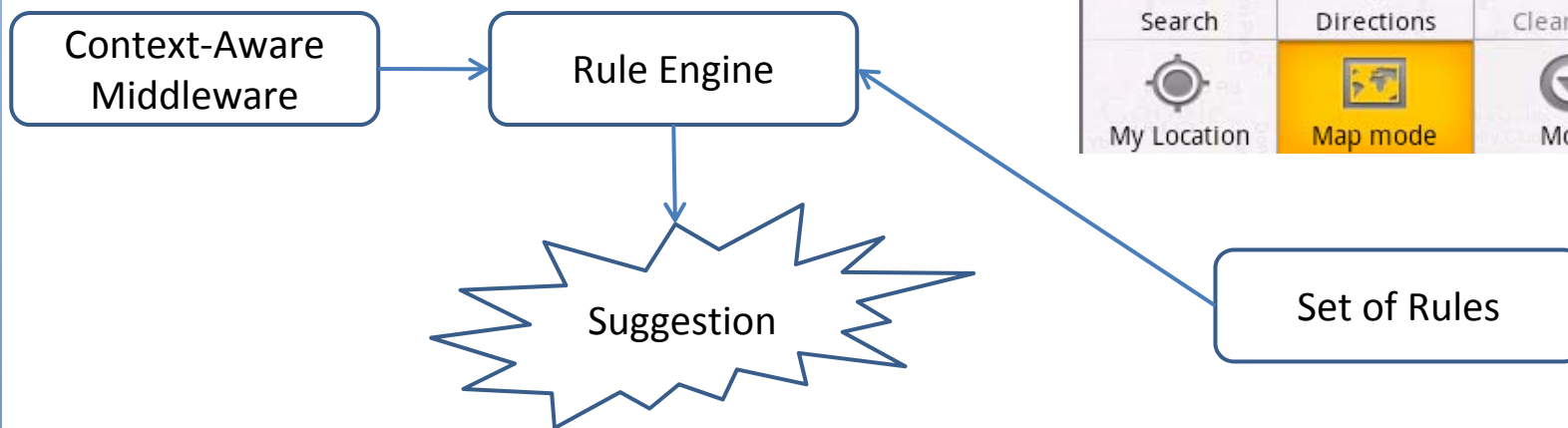
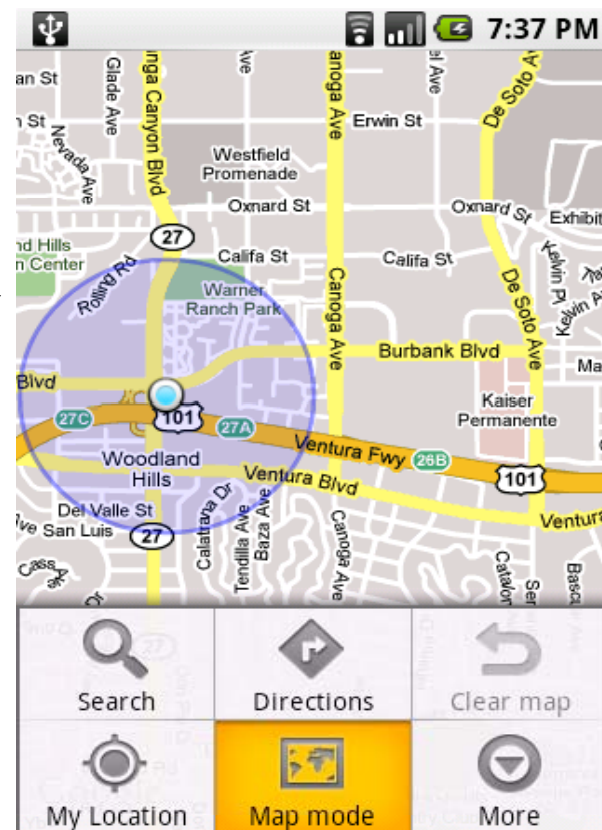
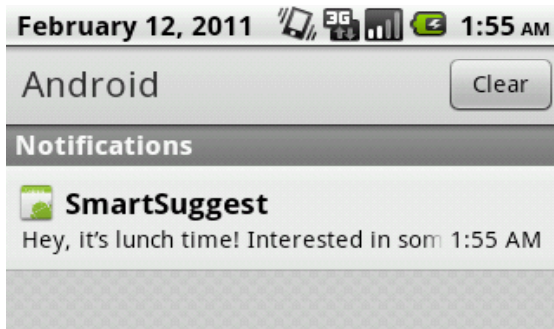
```
<rules-config>
  <rule-definitions>
    <rule-def name="DistributedSystemEventNotification"
action="category.EVENT_NOTIFICATION">
      <rule name="userIsFree" />
      <operator name="AND" />
      <rule name="userHasInterest" />
    </rule-def>
  </rule-definitions>
  <rule-implementations>
    <rule-impl name="userIsFree" class="rules.StringFieldEquals">
      <property name="argField" value="CURRENT_ACTIVITY"/>
      <property name="target" value="free"/>
    </rule-impl>
  </rule-implementations>
  <rule-implementations>
    <rule-impl name="userHasInterest"
class="rules.StringFieldContainedInList">
      <property name="argField" value="INTERESTS"/>
      <property name="target" value="Distributed Systems"/>
    </rule-impl>
  </rule-implementations>
</rules-config>
```

# An experimental application – SmartSuggests

Do you happen to know a restaurant nearby?



# An experimental application – SmartSuggests



# The rule

```
<?xml version="1.0" encoding="UTF-8"?>
<rules-config>
  <rule-definitions>
    <rule-def name="showRestaurantSuggestion"
      action="category.PLACE_SUGGESTION"
      parameter="restaurants">
      <rule name="isLunchTime" />
    </rule-def>
  </rule-definitions>
  <rule-implementations>
    <rule-impl name="isLunchTime" class="rules.IntFieldBetween">
      <property name="argField" value="TIME"/>
      <property name="targetStart" value="13"/>
      <property name="targetEnd" value="14"/>
    </rule-impl>
  </rule-implementations>
</rules-config>
```



# Conclusions and future work

---

- CAPIM – context-aware platform for context-aware applications
  - adapt to context conditions,
  - understand more easily the user needs,
- A pilot implementation of the contextualization platform has proven the great advantages it provides in terms of simplicity and flexibility.
- Future improvements of CAPIM involve the increase of energy efficiency, which is a critical property for software on portable devices.

# Q&A

---

Thank you! 😊

Ciprian Dobre

[ciprian.dobre@cs.pub.ro](mailto:ciprian.dobre@cs.pub.ro)