



# Orchestrating Context-Aware Systems

Alfredo Cádiz, Sebastian González, Kim Mens

**Presented by Antoine Beugnard**

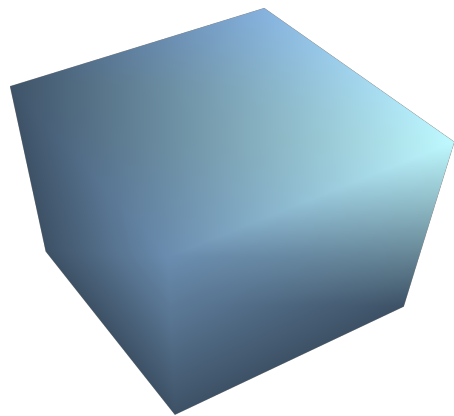
Warning: What follows is my interpretation...

24/08/2009

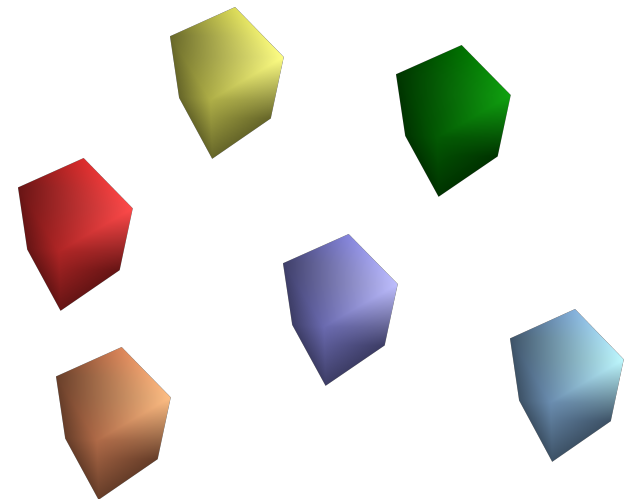
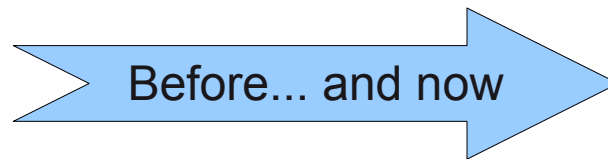


# Motivation

- Real-time availability of information
- Context of use changes hence applications must change !



Single (big) application  
Simple context



Many (small) applications  
Evolving context

## How to design ?

Avoiding anticipating all behaviours

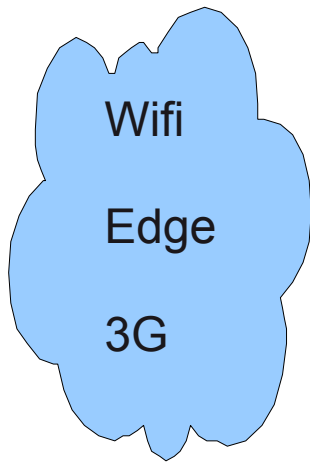
Setting causal connections between the environnement and the application



# Outline

- **Motivation**
- **Example**
- **General Architecture/Current solutions**
  - Context Discovery
  - Context Management
  - Active Context
  - Behaviour Adaptation
- **Questions...**

# Example : a model of the system

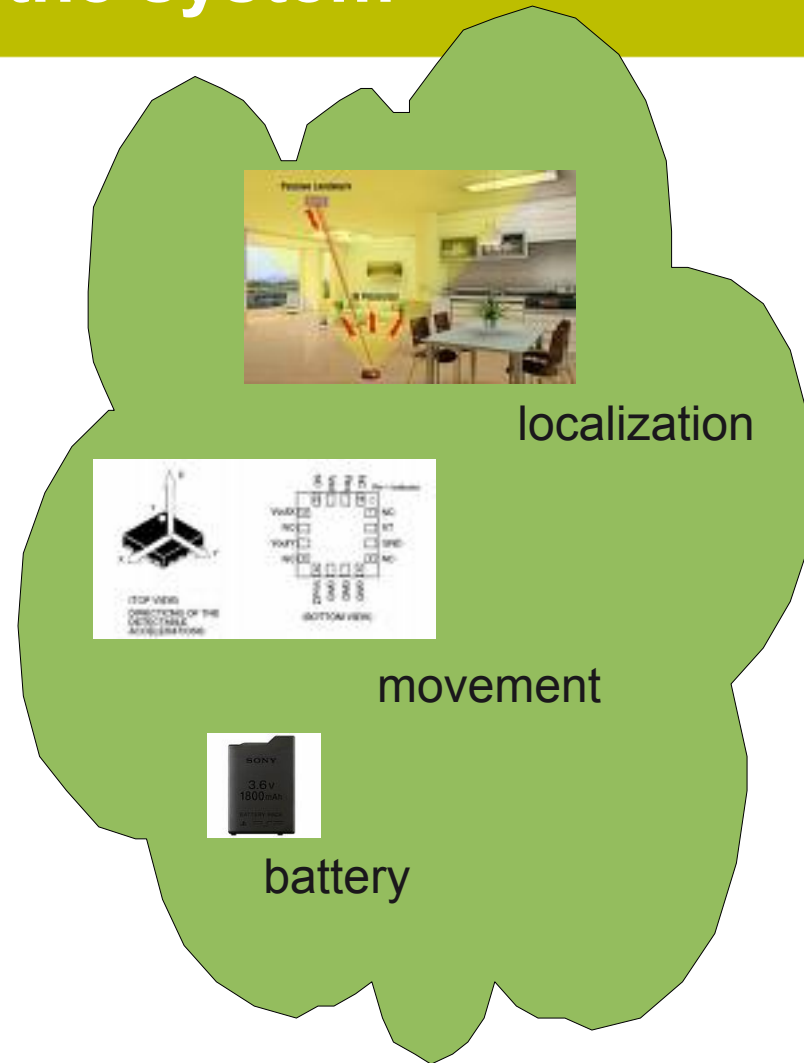


Technical options

The System



Services



Model of the environment



## Example : scenarios...

### ■ Low battery

- Send a warning
  - Simple
  - If in a known location add a map with a plug location
  - If in movement - so no known close plug
    - Switch to protocol with lower energy needs (edge)
    - But if a connection is running

How to (specify and) design ?



# Outline

- **Motivation**
- **Example**
- **General Architecture/Current solutions**
  - Context Discovery
  - Context Management
  - Active Context
  - Behaviour Adaptation
- **Questions...**

# General Architecture

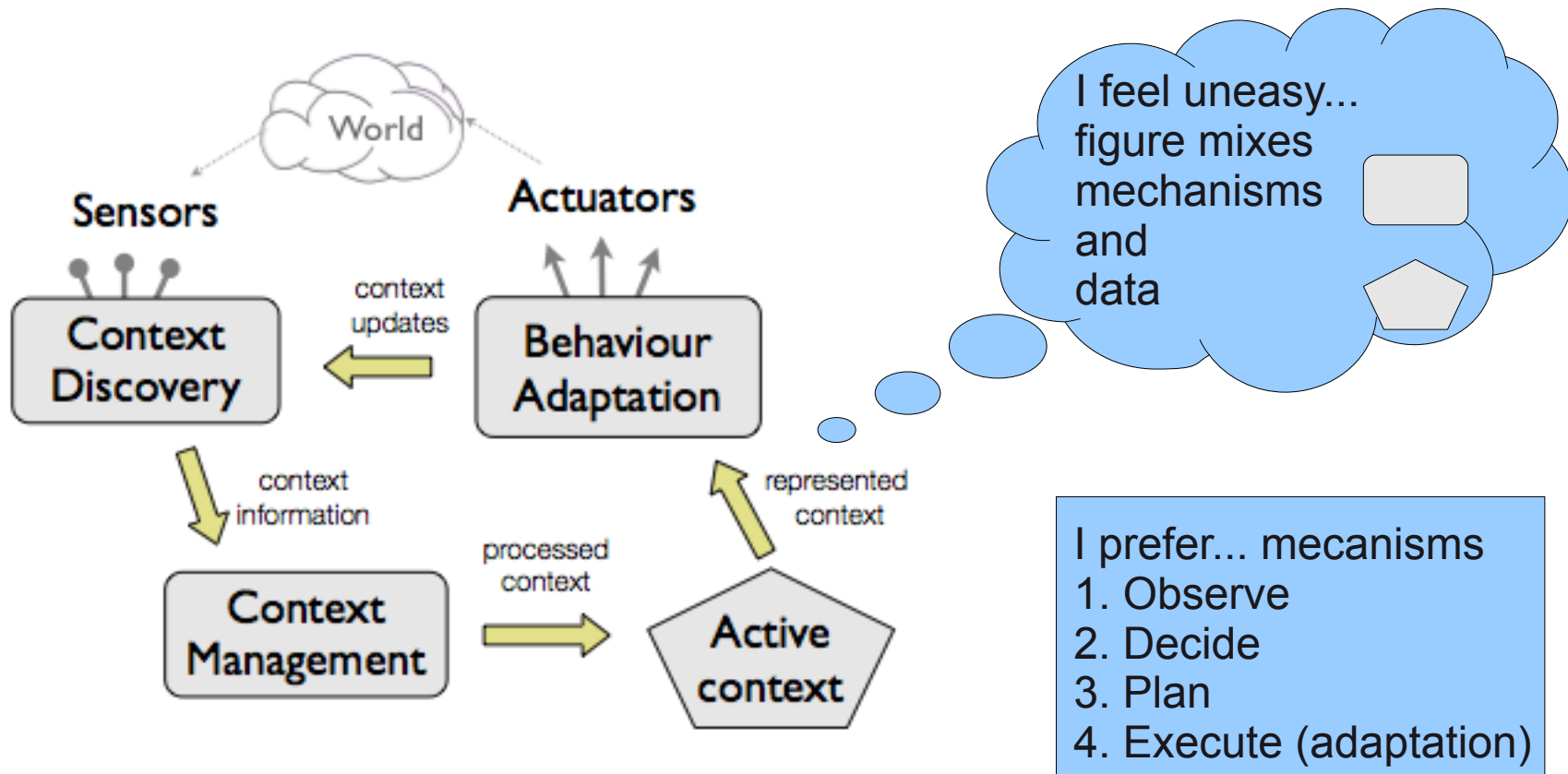


Figure 1: Context-aware system architecture

# Context Discovery [Observe #1]

## ■ Goal

- Extract, aggregate, deduce environment state

## ■ Input

- The environment and the system

## ■ Output

- Contexts (potential conflicts inconsistencies)

## ■ Comments

- push/pull modes, locus of data abstraction ?
  - Speed (still, walking, running, moving with a vehicle)
  - System status (idle, calling) [enough ? Audio, video ?]

## ■ Tools

- Context Toolkit, Wildcat, [COSMOS (sensors), SensorML 1







# Context Management [Observe #2]

## ■ Goal

- Remove contradictions, build a consistent active context

## ■ Input

- Contexts from Context Discovery

## ■ Output

- Active Context

## ■ Comments

- Rule and knowledge based processing

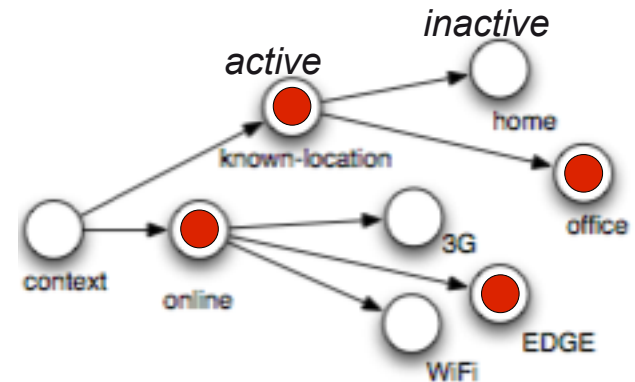
## ■ Tools

- SOCAM, CRIME, [COSMOS (processors)]

# Active Context [Decide #1 (recognize)]

## ■ Goal

- Capture the actual context [and map it to a model context]
- Glue between the context information and the application's logic



## ■ Comments

- Model of contexts (what and relationships) [a model is an anticipation of what can be treated?]

## ■ Tools

- Ambience

# Behaviour Adaptation [Decide#2, Plan, Execute]

## ■ Goal

- Do the job

## ■ Input

- Active Context

## ■ Output

- The right behaviour

## ■ Comments

- Strategy design pattern, reify context
- ONE signature, many contexts

## ■ Tools

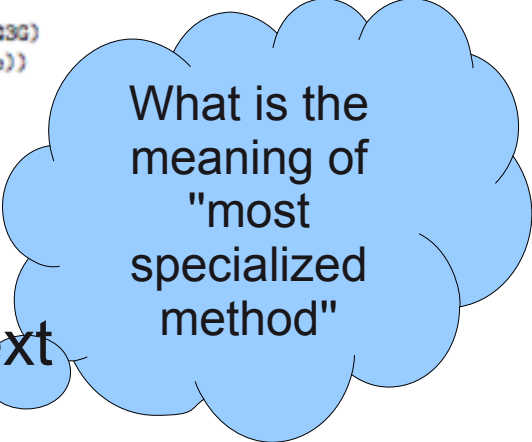
- Context-Oriented Programming Languages, ContextL, Ambience

```
(defmethod call ((from @phone) (to @phone))
  ... Make a call ...)

(with-context (@low-batt)
  (defmethod call ((from @phone) (to @phone))
    ... Show warning message
    (resend)))

(with-context (@low-batt @online)
  (defmethod call ((from @phone) (to @phone))
    ... Search for electricity and show map
    (resend)))

(with-context (@low-batt @unknown-location @3G)
  (defmethod call ((from @phone) (to @phone))
    ... Switch to EDGE...
    (resend)))
```



What is the meaning of "most specialized method"



# Outline

- **Motivation**
- **Example**
- **General Architecture/Current solutions**
  - Context Discovery
  - Context Management
  - Active Context
  - Behaviour Adaptation
- **Questions...**



## Remarks, questions, ...

### ■ **Control Command systems**

- Hysteresis, oscillations
- Real-time data? Latency, precision?
  - Need of: Quality of Context / metadata

### ■ **Application requirements**

- **If I go in a long run conversation, I know I need thrifty protocols**
  - Need for information status: battery longevity per option

### ■ **Viability space**

- Are all cases covered?



## Remarks, questions, ...

- **Application dependency?**
  - Fully independent? Observe? Specific application status or abstractions...
  - The model of contexts? Universal model?
  - What kind of coupling?
- **Many responsibilities for the Behaviour Adapter**
  - Assumption: the adaptation is a method selection
  - Are all adaptations reductive to a method selection?



# Thanks