Laboratoire d'Informatique pour l'Entreprise et les Systèmes de Production

ORCHESTRA: Formalism to Express Static and Dynamic Model of Mobile Collaborative Activities and Associated Patterns

Bertrand DAVID, René CHALON,
Olivier DELOTTE, Guillaume MASSEREY
LIESP Laboratory
Ecole Centrale de Lyon
France

Presented by Ahmed SEFFAH (Concordia University)

Agenda

- ➔ Introduction
- ➔ Cooperative applications and capillary CW
- ➔ CoCSys Methodology
- ➔ Motivations
- ➔ ORCHESTRA
  - ➲ Principles
  - ➲ 2 case studies
  - ➲ Transformation process
- ➔ Conclusion and perspectives

Introduction

➔We are working on design, development and adaptation of cooperative systems

- ➲CSCW
- ➲Capillary Cooperative Systems
- ➲Context aware behavior
  - ☒MOCOCO:
    - Mobility
    - Cooperation
    - Contextualization

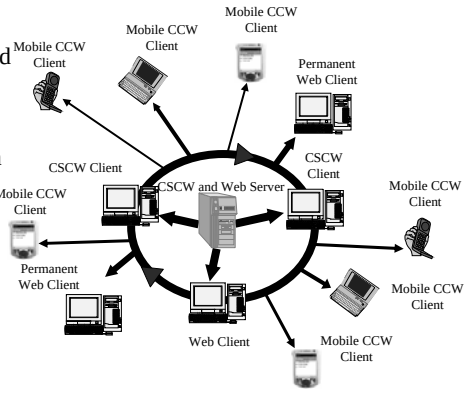
CCW - Capillary Cooperative Work

In relation with new IT technologies (wireless networking and handheld devices: laptops, tablePC, PDA, mobile phone), it is possible to extend CSCW to **Capillary Cooperative Work** (CCW).

We use this term not un relation with hair but by **analogy with the network of blood vessels**.

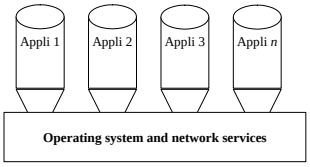
The purpose of the Capillary CW is:

"to extend the capacities provided by co-operative working tools in increasingly fine ramifications, hence their can use fixed workstations and handheld mobile devices"

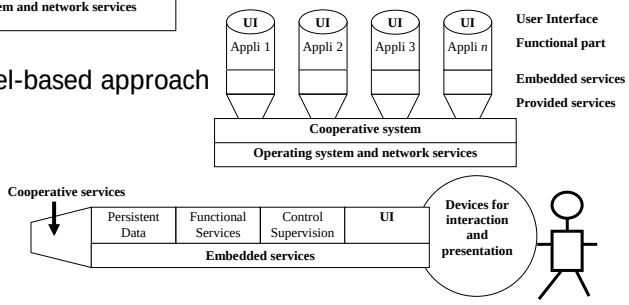


Evolution of software development

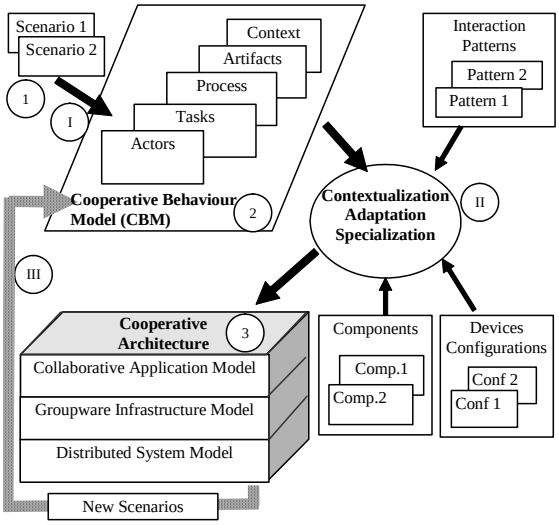
➔ From craftsman work



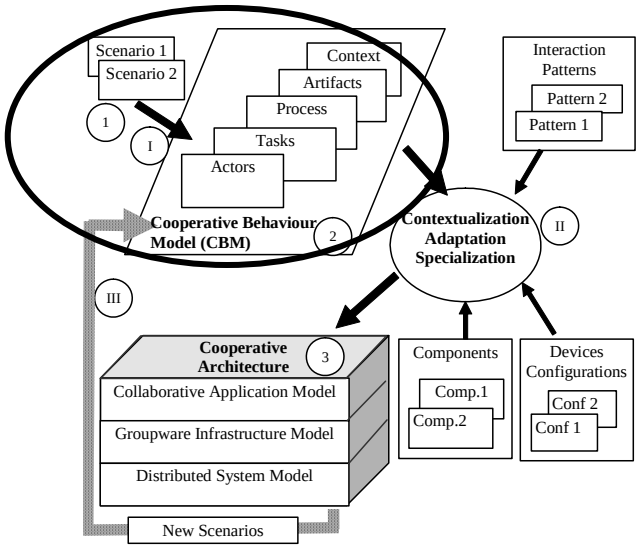
➔ To model-based approach



CoCSys Methodology



CoCSys Methodology



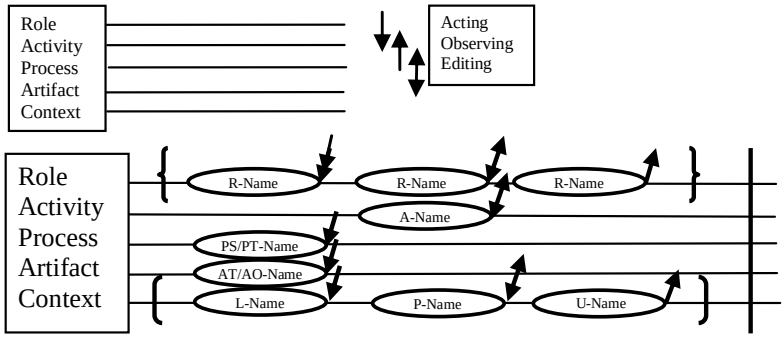
Reasons of ORCHESTRA (requirements)

- The objective of Orchestra is to propose a more comprehensive formalism which is able to express together all main aspects of the CBM (Cooperative Behaviour Model).
- In the literature we can find partial solutions with Paterno's CTT (**ConcurTaskTrees**), **UML** activity diagram or C. Ferraris **LDL** devoted to CSCL.
- Main requirement for ORCHESTRA is to propose a graphical formalism easy to read, to learn, to understand and to write expressing together:
  - Actors and Roles
  - Tasks and activities
  - Process (states and transitions)
  - Artefacts (objects and tools)
  - Context (location, platform and user's characteristics)
  - Cooperation styles and modes
  - Coordination modes
  - Awareness

  ORCHESTRA adapts **musical score notation** to CBM (Cooperative Behaviour Model) description.

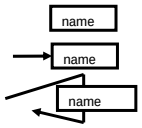
ORCHESTRA main concepts (1/3)

- For us, the **5 lines of a staff** are expressing 5 main aspects of the CBM which are: **user's role, activity concerned, process state or transition, artefacts involving in the activity and the context.**
- On each line, we can situate one or several "**notes**" expressing names of corresponding items i.e. roles, activities, process states or transitions, artefacts and contexts. Each note can receive a **stem** which indicates the participation of the element (acting, observing or editing done by main or secondary actors).

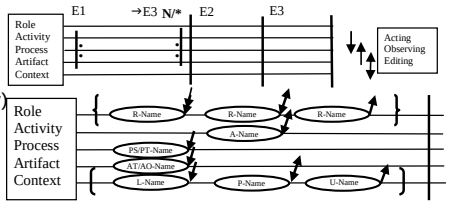


ORCHESTRA main concepts (2/3)

- A **bar line** indicates the separation between independent **cooperation episodes**. Each cooperation episode expresses a state or a transition in the cooperation process description network (workflow).
- For each cooperation episode temporal organization is expressed either sequentially from the left to the right, or by explicit change of episode.
- Repetitions can be unlimited, explicitly limited (numerically or contextually)
- Explicit change of episode uses named episodes
  - Rerouting to an episode
  - Call an episode
    - ☒ These two can be unconditional or on condition



- Parentheses are used to express different situations of "chords" participation:
  - (...) alternatives
  - {...} mandatory participation
  - [....] potential participation
- We distinguish main actor (double arrow) and secondary actor (simple arrow) as well as active role and passive role:
  - ✡ Active role
  - ◎ Passive role



ORCHESTRA main concepts (3/3)

Coordination modes:

- Computational coordination
- Social coordination

Cooperation modes:

- ● Implicit cooperation (instantaneous, short term)
- ▬ Explicit cooperation (long term)

Cooperation styles:

- @ Asynchronous with infinite answer delay
- @@ Asynchronous with limited answer delay (on call)
- & Synchronous "in-meeting" cooperation
- && Synchronous "in-depth" cooperation

Cooperation awareness:

- No awareness
- Partial awareness (for specific actors)
- Overall awareness (for all actors)

Patterns

➔ **Pattern = Problem + Context + (potential) Solution(s)**.

➔ **ORCHESTRA pattern** is a schema with one or several chords constituting cooperation episode(s) organized temporally and associated to a particular configuration of complementary annotations expressing

- ➲ **nature of cooperation** (Synchronous or Asynchronous),
- ➲ **level of cooperation** (asynchronous with infinite delay, on call, in meeting or in-depth cooperation),
- ➲ **coordination style** (social or computational),
- ➲ **nature of coordination** (implicit or explicit)
- ➲ **awareness** (overall, partial or no awareness).

Pattern	S/As	Coop	Coord	Exp/Imp	Aware	Coop. configurations
Intervention appointment	S/As	&/@@	☺	●		

ORCHESTRA Cooperation Patterns

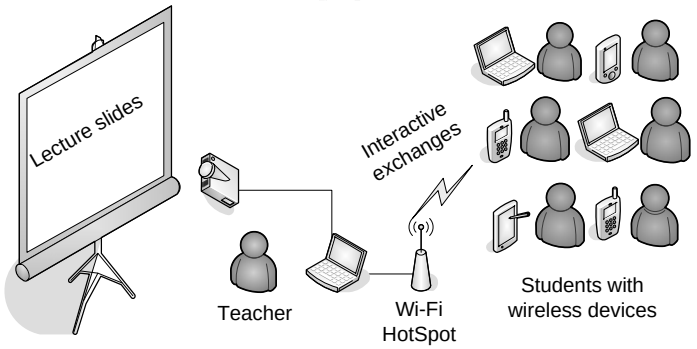
To exemplify this approach we identified six important cooperation patterns:

- **Intervention appointment:** Synchronous or asynchronous, on-call or in-meeting cooperation with computational implicit coordination and no awareness.
- **Consultation – vote:** Synchronous, in-meeting cooperation with computational and implicit coordination and either overall awareness or no awareness.
- **Presentation:** Synchronous and in-meeting cooperation with social and explicit coordination with overall awareness.
- **In-depth work:** Synchronous, in-depth cooperation with computational explicit coordination and partial awareness.
- **Questions / Answers:** Synchronous, in-meeting activity with social or computational explicit coordination.
- **Validation:** Asynchronous, on-call cooperation with implicit coordination and no awareness.

Patterns

Pattern	S/As	Coop	Coord	Exp/Imp	Aware	Coop. configurations
Intervention appointment	S/As	&/@@	😐	●	🚗	↓◎ ✲↑
Consultation Vote	S	&	💻	●	👥 / 🚗	✲ ✲ ✲ ◎ ✲ ✲
Presentation	S	&	😐	▬	👥	✲ ◎ ◎ ◎ ◎
In-depth work	S	&&	💻	▬	🚗	✲ ✲ ✲
Questions / Answers	S	&	😐 / 💻	▬	👥 / 🚗	↓✲ ✲↑
Validation	A	@@		●	🚗	→ ✲ →

Case Study: Cooperation in teaching in large class



List of scenarios:

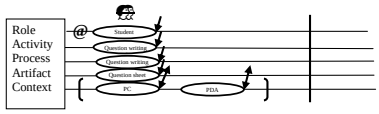
Students can give feedback about lecture speed (too slow, too fast), write and submit questions or remarks to the teacher, take notes, answer to full assessments or quick polls, ...

Teacher receives and reads all students messages on his laptop. He can answer to submitted questions or get the general feeling about the running lecture, …

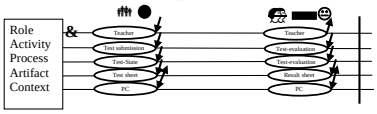
Screenshots of teacher interface on different devices

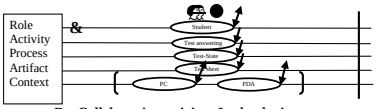
Screenshots of student interface on different devices

Cooperation in teaching in large class Different levels of description

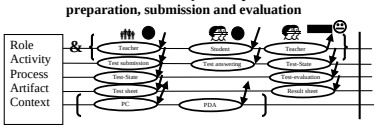


A – Individual activity: question to teacher writing





B – Collaborative activity - In-depth view: test preparation, submission and evaluation



C - Collaborative activity - Synthetic view

Cooperation styles:
- @ Asynchronous with infinite answer delay
- @@ Asynchronous with limited answer delay (on call)
- & Synchronous "in-meeting" cooperation
- && Synchronous "in-depth" cooperation

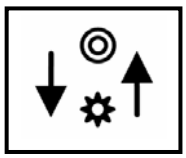
Cooperation modes:
- ● Implicit cooperation (instantaneous, short term)
- ▬ Explicit cooperation (long term)

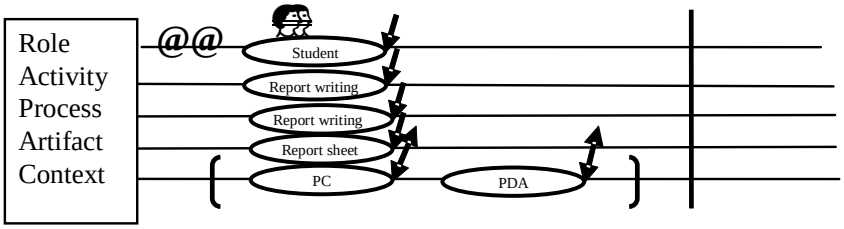
Coordination modes:
- Computational coordination
- Social coordination

Cooperation awareness:
- No awareness
- Partial awareness (for specific actors)
- Overall awareness (for all actors)

Pattern Intervention appointment instantiated as « Report writing »

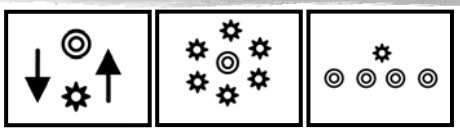
➔ **Intervention appointment:** Synchronous in-meeting cooperation with computational implicit coordination and no awareness.



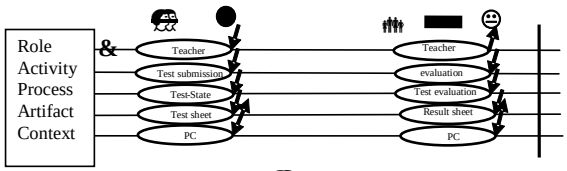


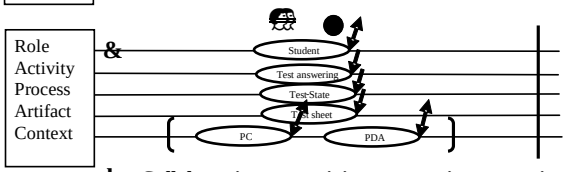
a - Individual activity "Report writing"

Patterns used for test activity expression



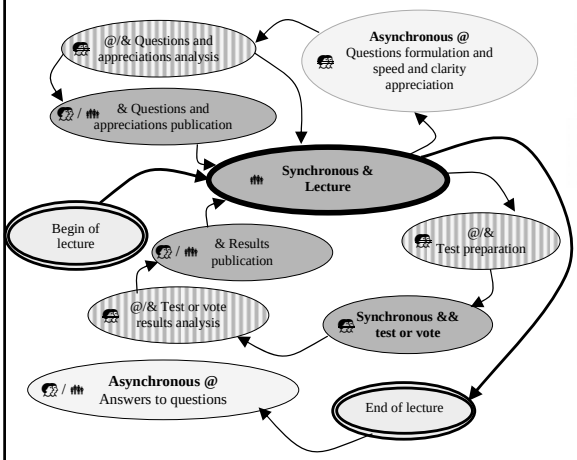
Intervention, Vote and Presentation Patterns





b - Collaborative test activity preparation, execution and treatment

Cooperation in teaching in large class: Collaboration process (workflow)



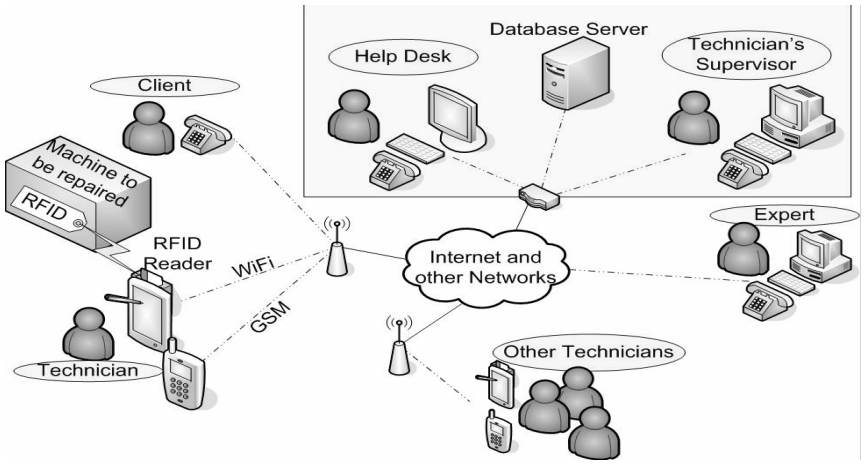
Cooperation styles:
@ Asynchronous with infinite answer delay
@@ Asynchronous with limited answer delay (on call
& Synchronous "in-meeting" cooperation
&& Synchronous "in-depth" cooperation

Cooperation modes:
Implicit cooperation (instantaneous, short term)
Explicit cooperation (long term)

Coordination modes:
Computational coordination
Social coordination

Cooperation awareness:
No awareness
Partial awareness (for specific actors)
Overall awareness (for all actors)

Case study: Heating Equipment maintenance activities



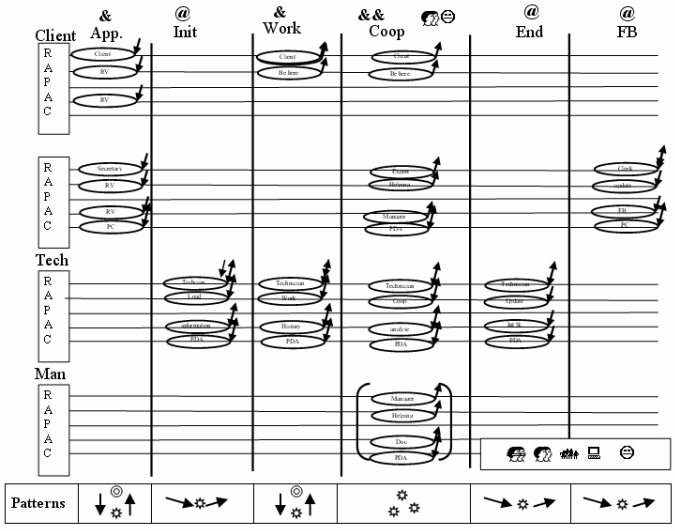
Case study: Heating Equipment maintenance activities

Six actors: client, secretary, technician, supervisor, expert and clerk.

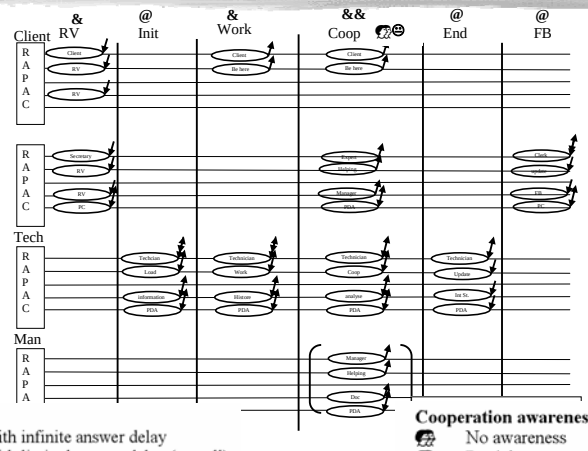
Main scenarios to put together are the following:

1. A client (secondary actor), observing a problem with his heating equipment, phones to the repair company to ask intervention. The secretary (secondary actor) asks him his profile (address, equipment...) and finds him in the database. He organizes an appointment with a technician. **State: RV (RendezVous), Actors: Client, Secretary, &**
2. In the morning, before leaving the company, the technician (main actor) loads on his PDA necessary information for his round with appropriate information (clients and their addresses, nature of intervention ...). **State: Init, Actor: technician, @**
3. At client house, the technician works on maintenance process, he can study history file of the supplies, precise blueprints, elaborates a diagnosis using appropriate tools, and repair, or ask for spare parts. **State: Work, Actors: Client, Technician, &**
4. In a situation of impossibility to diagnose alone, he can contact his manager (secondary actor) to ask him some helps and to exchange some information. He can also contact, in a synchronous manner the heating manufacturer expert (secondary actor) to study the situation with him. **State: Coop, Actors: Technician, Manager, Expert, &&**
5. At the end of his round technician, back at the company, updates history file of visited equipments and gives his intervention statement. **State: End, Actor: Technician, @**
6. Next day the clerk (secondary actor) produces the financial balance and statement of accounts and either sends the bill to the client or he integrates it in the client record. **State: FB (Financial Balance), Actor: clerk, @**

ORCHESTRA expression of Case study: Heating Equipment maintenance activities



ORCHESTRA expression of Case study: Heating Equipment maintenance activities



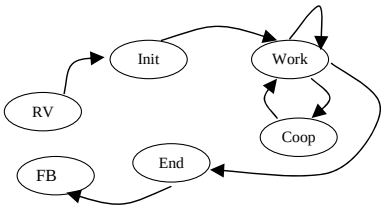
Cooperation styles:
@ Asynchronous with infinite answer delay
@@ Asynchronous with limited answer delay (on call)
& Synchronous "in-meeting" cooperation
&& Synchronous "in-depth" cooperation

Cooperation modes:
● Implicit cooperation (instantaneous, short term)
▬ Explicit cooperation (long term)

Cooperation awareness:
No awareness
Partial awareness (for specific actors)
Overall awareness (for all actors)

Coordination modes:
Computational coordination
Social coordination

Heating Equipment maintenance activities Cooperation process description



- ➔ **State: RV (RendezVous), Actors: Client, Secretary, &**
- ➔ **State: Init, Actor: technician, @**
- ➲ **State: Work, Actors: Client, Technician, &**
- ➲ **State: Coop, Actors: Technician, Manager, Expert, &&**
- ➲ **State: End, Actor: Technician, @**
- ➲ **State: FB (Financial Balance), Actor: clerk, @**

Cooperation styles:
@ Asynchronous with infinite answer delay
@@ Asynchronous with limited answer delay (on call)
& Synchronous "in-meeting" cooperation
&& Synchronous "in-depth" cooperation

Cooperation modes:
● Implicit cooperation (instantaneous, short term)
▬ Explicit cooperation (long term)

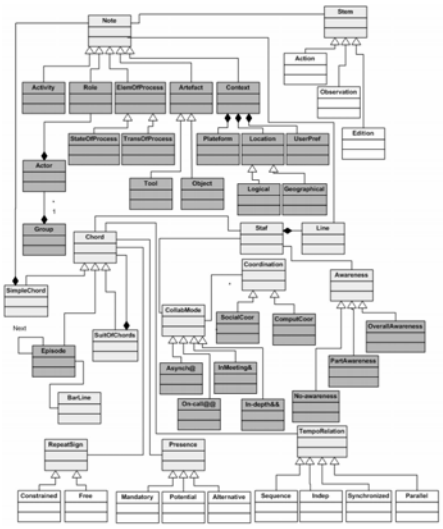
Cooperation awareness:
No awareness
Partial awareness (for specific actors)
Overall awareness (for all actors)

Coordination modes:
Computational coordination
Social coordination

ORCHESTRA and in-depth description

- ➔ Of course graphical description gives only global overview
- ➔ More comprehensive description is textual: identification of precise characteristics, participating artefacts (objects and tools with associated methods) and so on.
- ➔ On this textual description is possible to do different validations of CBM:
  - ➲ Completeness
  - ➲ Consistency
  - ➲ Coherence

ORCHESTRA meta-model

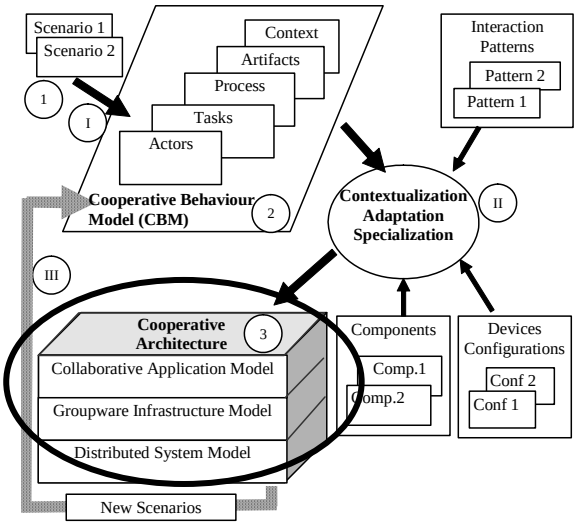


Main concepts

Derived concepts

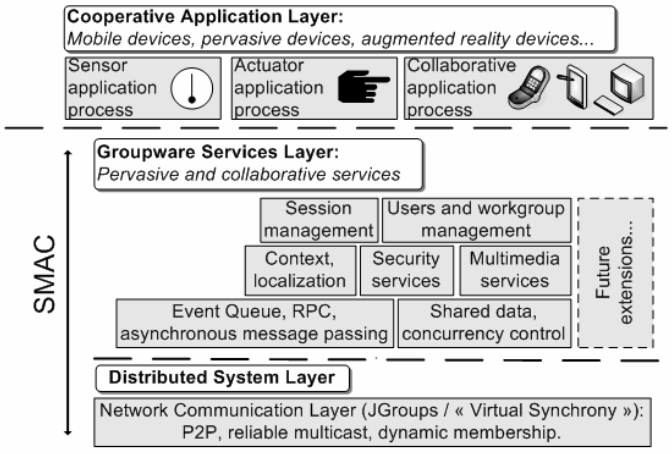
Secondary concepts

CoCSys Methodology

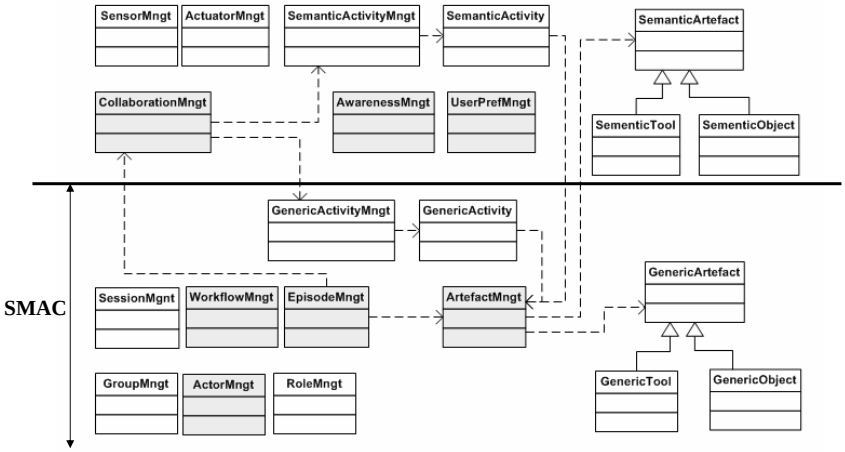


Three layer collaborative architecture model

Cooperative middleware called SMAC (Services for Mobile Applications and Collaborations) implements the two lower layers (groupware services and distributed system) of this conceptual cooperative architecture.



SMAC and Cooperative Application Layer core classes



Conclusion and perspectives

- We proposed a model-based approach to mobile (capillary) cooperative system design with
  - CoCSys methodology
  - ORCHESTRA as global overview graphical formalism of CBM (Cooperative Behavior Model)
  - Transformation process from CBM to Collaborative architecture model
  - We tried to show you on two case studies that ORCHESTRA is easy to read and to understand.
  - We also presented identified reusable patterns allowing to speed-up ORCHESTRA description process.
  - Integration of Mix Reality (Augmented Reality) with new I/O devices (see-throw goggles) and communicating objects (RF-ID based).
- Main perspectives are
  - We are working on Visio based Editor allowing to collect not only graphical expression, but also in-depth textual characteristics.
  - XML representation is also in progress as well as the projection process to the Collaborative architecture.
  - Study of Evolutivity of CBM in order to take into account users' evolution demands and the role which can be played by ORCHESTRA in this process.

Thank you .....

Questions ?

In-depth questions can be send to
Bertrand.David@ec-lyon.fr