CAR'2007 - Paris, France

Towards an Adaptive Robot Control Architecture

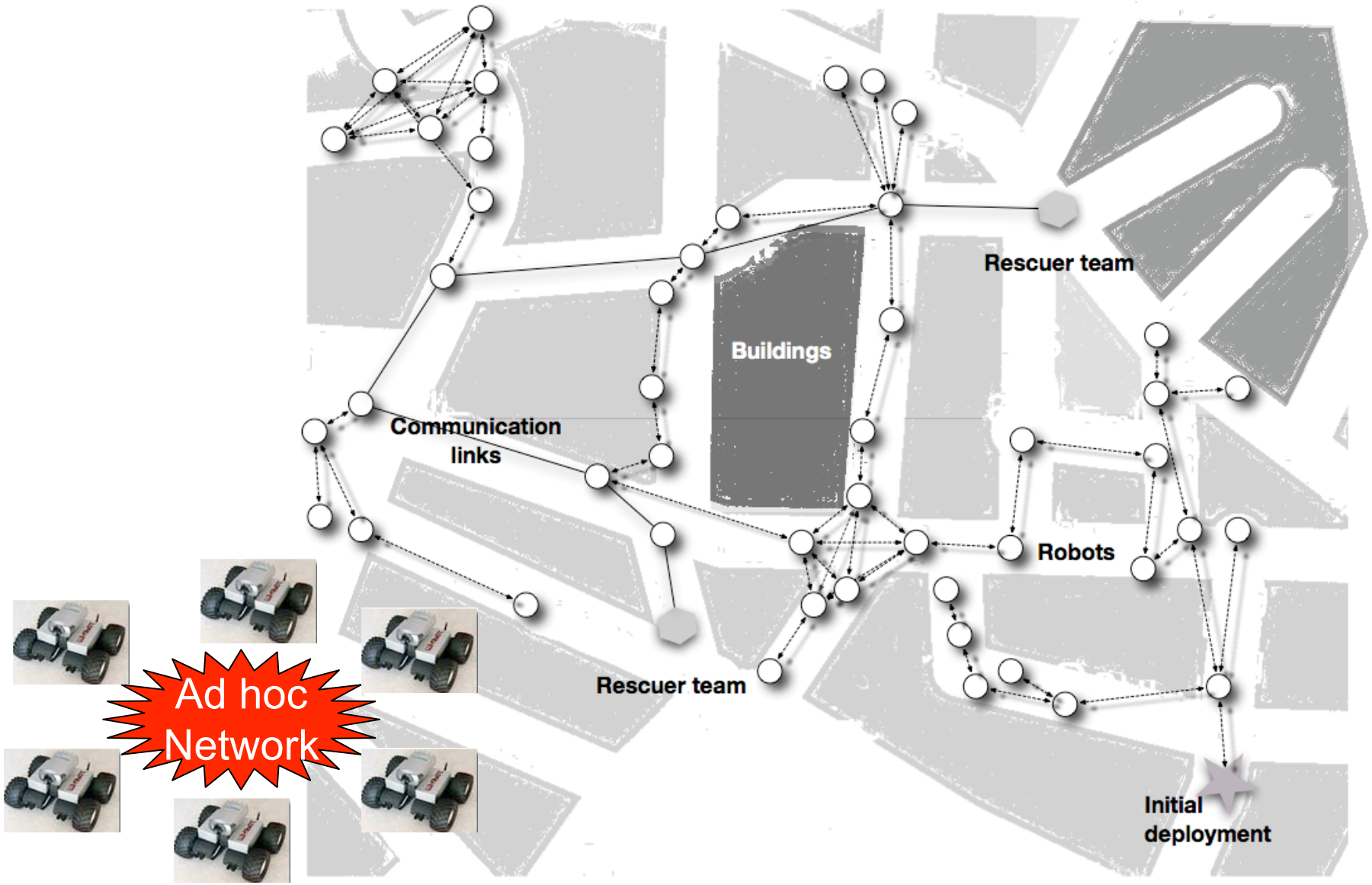
Noury Bouraqadi
École des Mines de Douai
France
http://csl.ensm-douai.fr/noury

Serge Stinckwich
GREYC - CNRS / Univ. de Caen
France
http://www.iutc3.unicaen.fr/serge

Robotic Urban Search & Rescue

- Robotic USAR involves
  - the location, extrication, and initial medical stabilization of victims trapped in confined spaces using mobile robots.
- AROUND project
  - Autonomous Robots for Observation of Urban Networks after Disaster
  - Automated observation system for disaster zone in developing countries (Vietnam).
- Robots tasks = Reconnaissance + Covering

AROUND Scenario

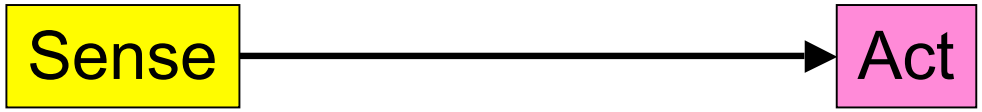


Agents and MAS

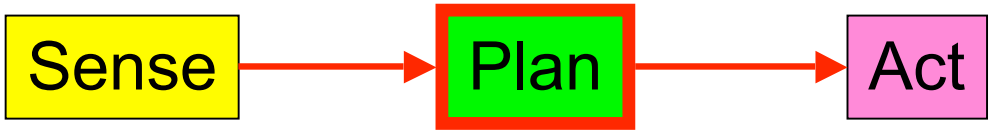
- MAS = Multi-Agents System
  - study of the collective behavior of a group
  - possibly heterogeneous agents with conflicting goals.
- Autonomy principle
  - Agents are objects that say go and no [Parunak]
- Self-organization principle
  - Application developers simply identify the agents suitable to solve a specific problem
  - Agent organize themselves to perform the required functionality.

Hybrid Agents are Suitable for Robotics

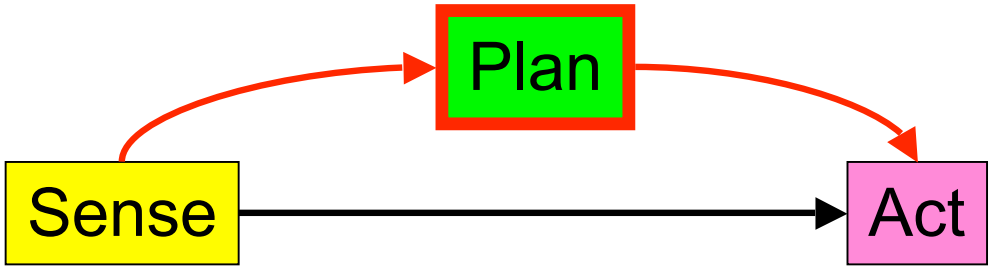
- Reactive agents are fast but "dumb"



- Cognitive agents are smart but slow



- Hybrid agents mix cognition and reflexes



Motivation for Adaptation

"Adaptation is the process of conforming a software to new or different conditions" [Ketfi et al.]

- Resource Awareness
- Fault tolerance
- Changing environment

Adaptation Dimensions

- **What** is adapted ?
- **Who** performs adaptations ?
- **When** does the adaptation occur ?
- **How** is the adaptation performed ?

What is adapted ?

- Parameters
  - Simplest form of adaptation
  - Scalability problem
- Reorganization (Logical or physical)
  - Reorganize connections
  - Distribution
- Addition/Suppression of building blocks
  - Open-ended system

Who performs adaptations ?

- Adaptation Autonomy ?
  - Human or Machine or both ?
  - Multiple autonomy "degrees"
    - Choose triggering conditions
    - Trigger adaptation
    - Compute vs. select appropriate adaptations
    - Plan adaptations (when and how)
    - Accomplish adaptation operations

When does the adaptation occur ?

- Statically = before starting the software
  - i.e. compile-time or deploy-time

- Dynamically = at run-time
  - More relevant decisions
  - More complex
    - System coherence
    - Resource consumption

How is the adaptation performed ?

- Mechanisms and strategies for adaptation
- Comparison criteria:
  - Ease of use = development effort
  - Transparency = impact on execution (e.g. freezing)
  - Efficiency = amount of required resources for adaptation
  - Control = drive & coordinate (distributed) adaptations
  - Separation of concerns = Adaptation vs. Adapted code

Related Work on Adaptation in Agent Models

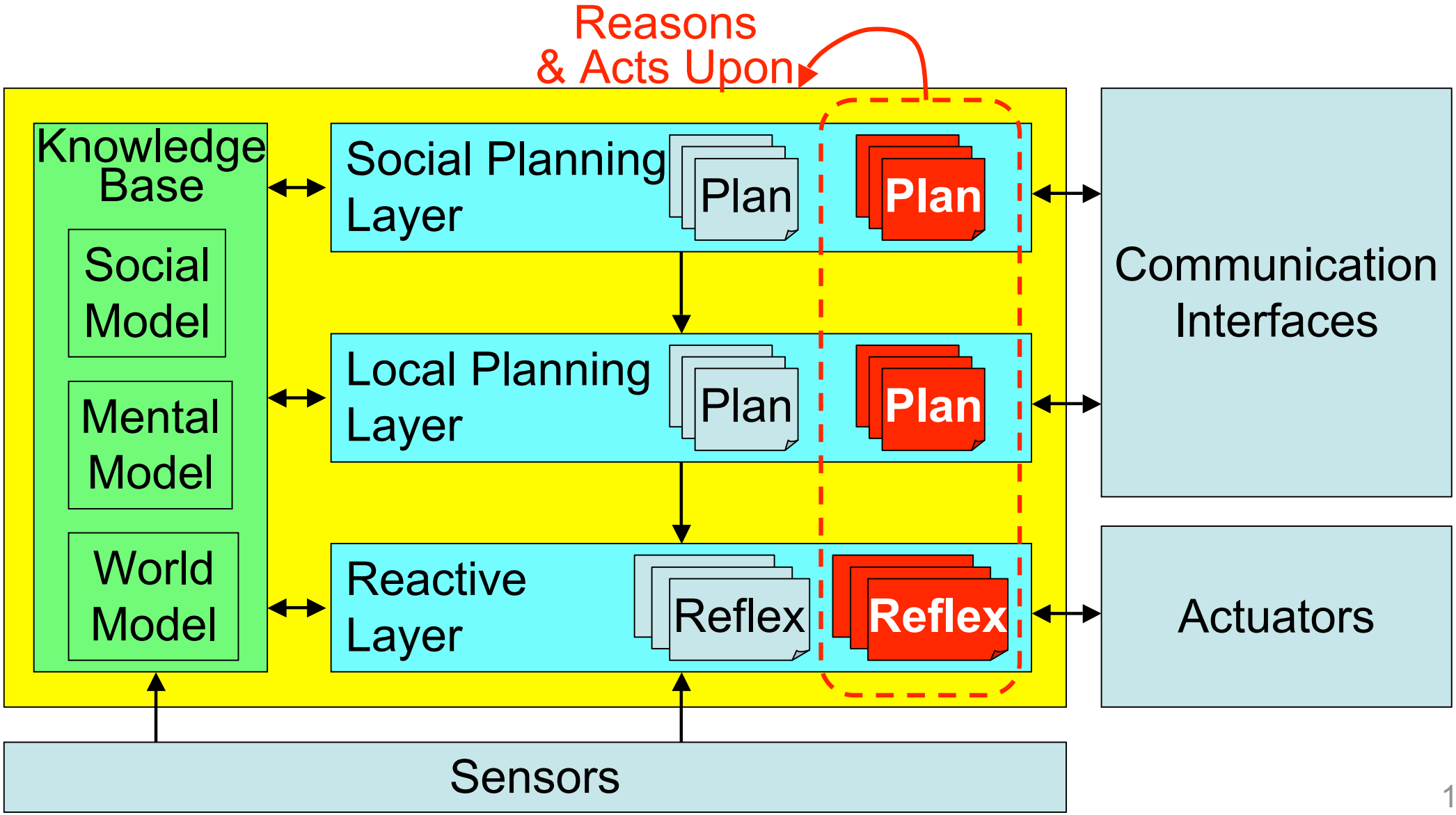
- Touring Machine
- InteRRaP
- Meta-Control Agents

- => Most existing models commit in early design stages to some particular software agent architecture
  - Need for "Hybridity Tunning"

Our Approach

- **Hybrid agents**
  - Adaptations include "Tuning Hybridity"
- **Software Components**
  - Clear Architecture
  - Simple Mapping of Design to Implementation
  - Reuse
- **Reflection**
  - "Ability of a system to reason and act upon itself" [Smith 84]
  - Adaptation and Autonomy

Our Architecture = "InteRRaP"+Components+Reflection



Adaptation Dimensions for our Architecture - 1

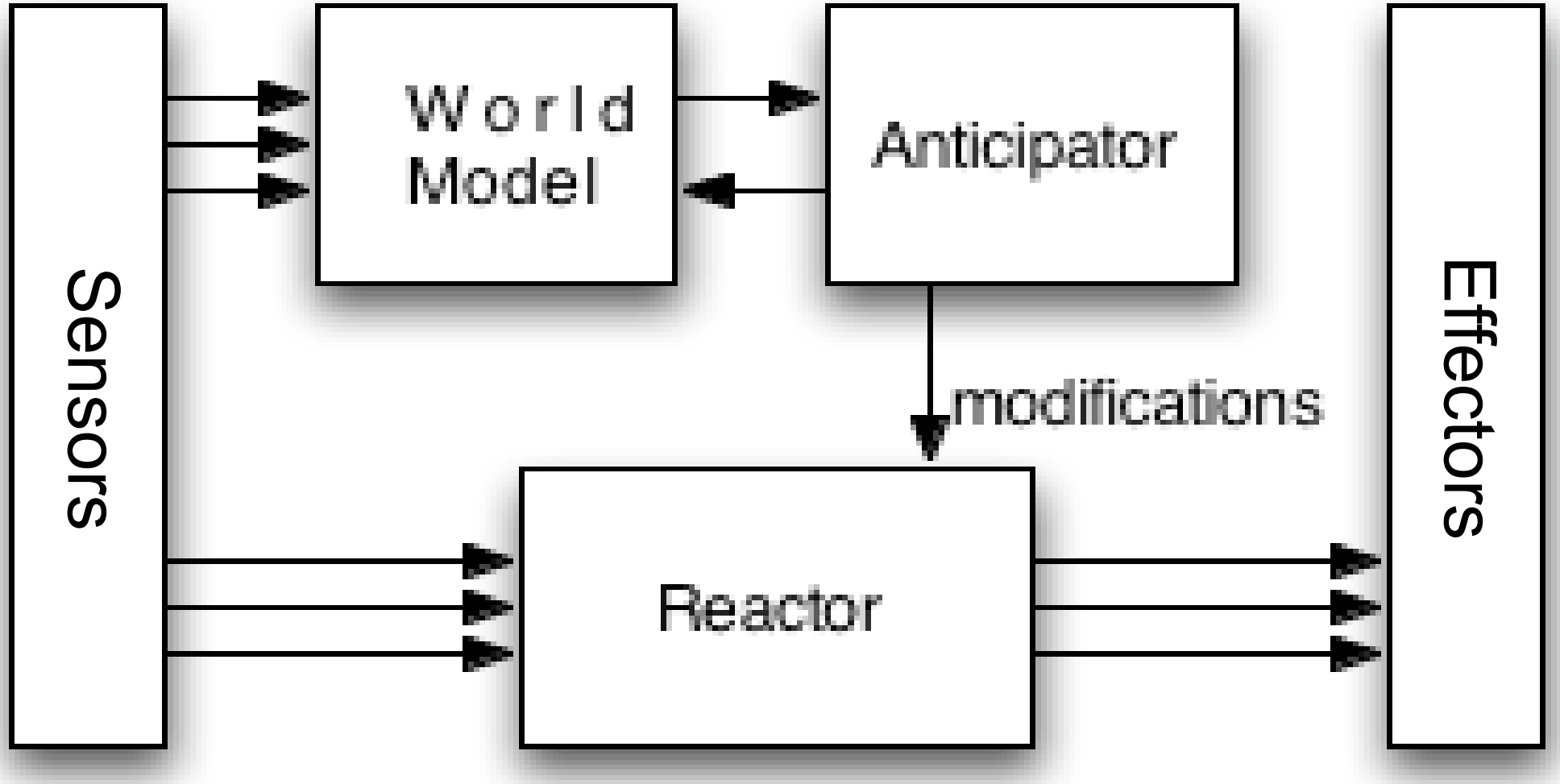
- **What** is adapted?
  - Every "part" = components and their connections
  - Enable "hybridity tuning"
    - Replace a reflex with a plan or vice versa
  - Allow adapting adaptation behavior
- **Who** performs adaptations?
  - Adaptation designers (during development)
    - E.g. Remove the social layer
  - Administrators (while the system is running)
    - E.g. Install new versions of some components
  - Agents themselves **autonomously**
    - Any layer including the social layer => coordinated adaptations

Adaptation Dimensions for our Architecture - 2

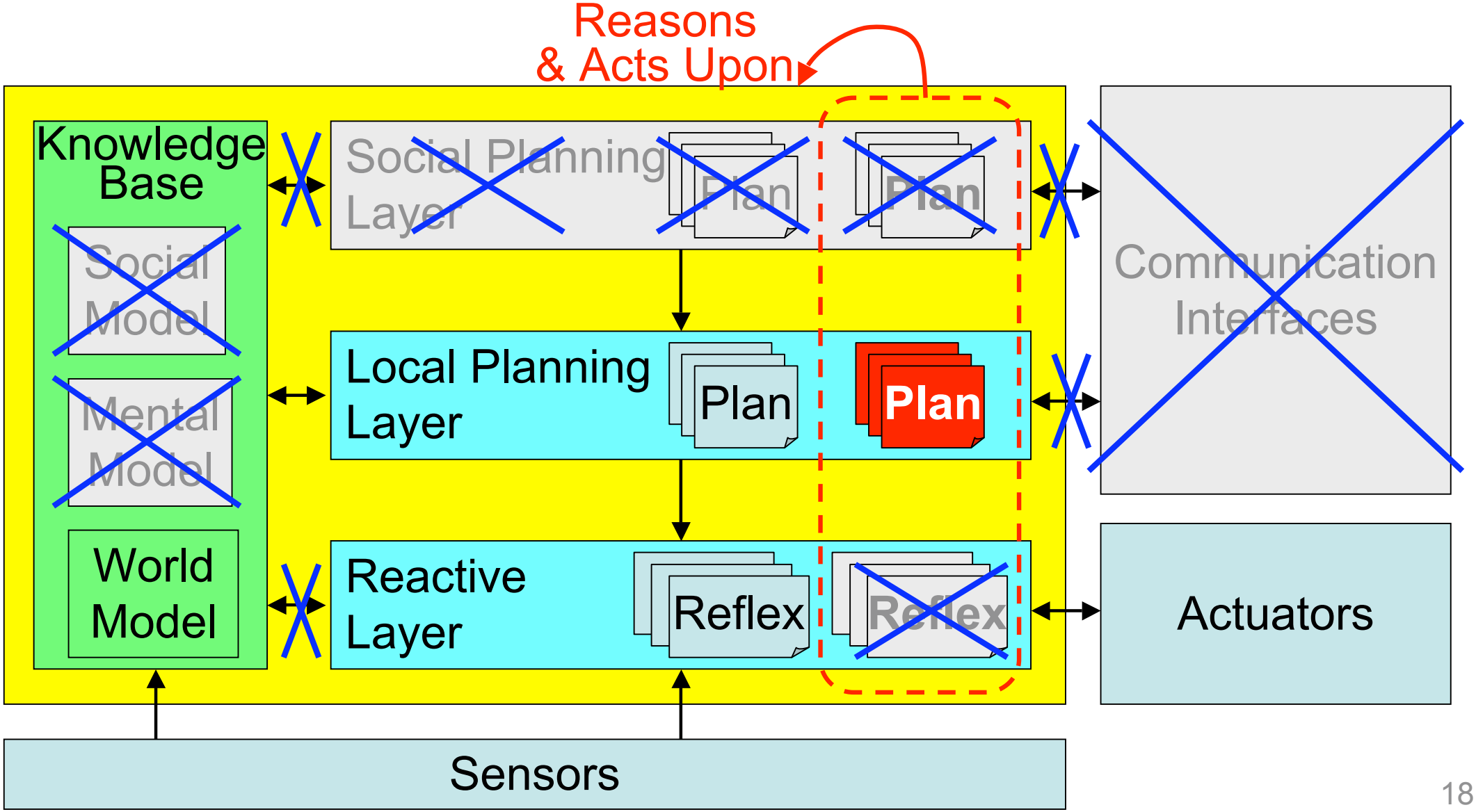
- **When** does the adaptation occur?
  - Statically by designers
  - Dynamically by agents or administrators
    - e.g. switch from path planning to obstacle avoidance reflexes
- **How** is the adaptation performed?
  - Ease of use : Adapt = reassemble/reconfigure components
  - Transparency based on adaptation planning
    - Ability to choose the appropriate moment, or adapt by separated steps
  - Efficiency: Adaptation cost can be tuned
    - Adapting adaptation strategies
  - Control: local (reflex + local plans) & distributed (social plans)
  - Separation of concerns is achieved
    - Adaptations = identified plans/reflexes

Anticipatory Agents
A "Projection" of our Architecture

[Davidsson 1996]



Anticipatory Agents
A "Projection" of our Architecture



Adaptations in Anticipatory Agents

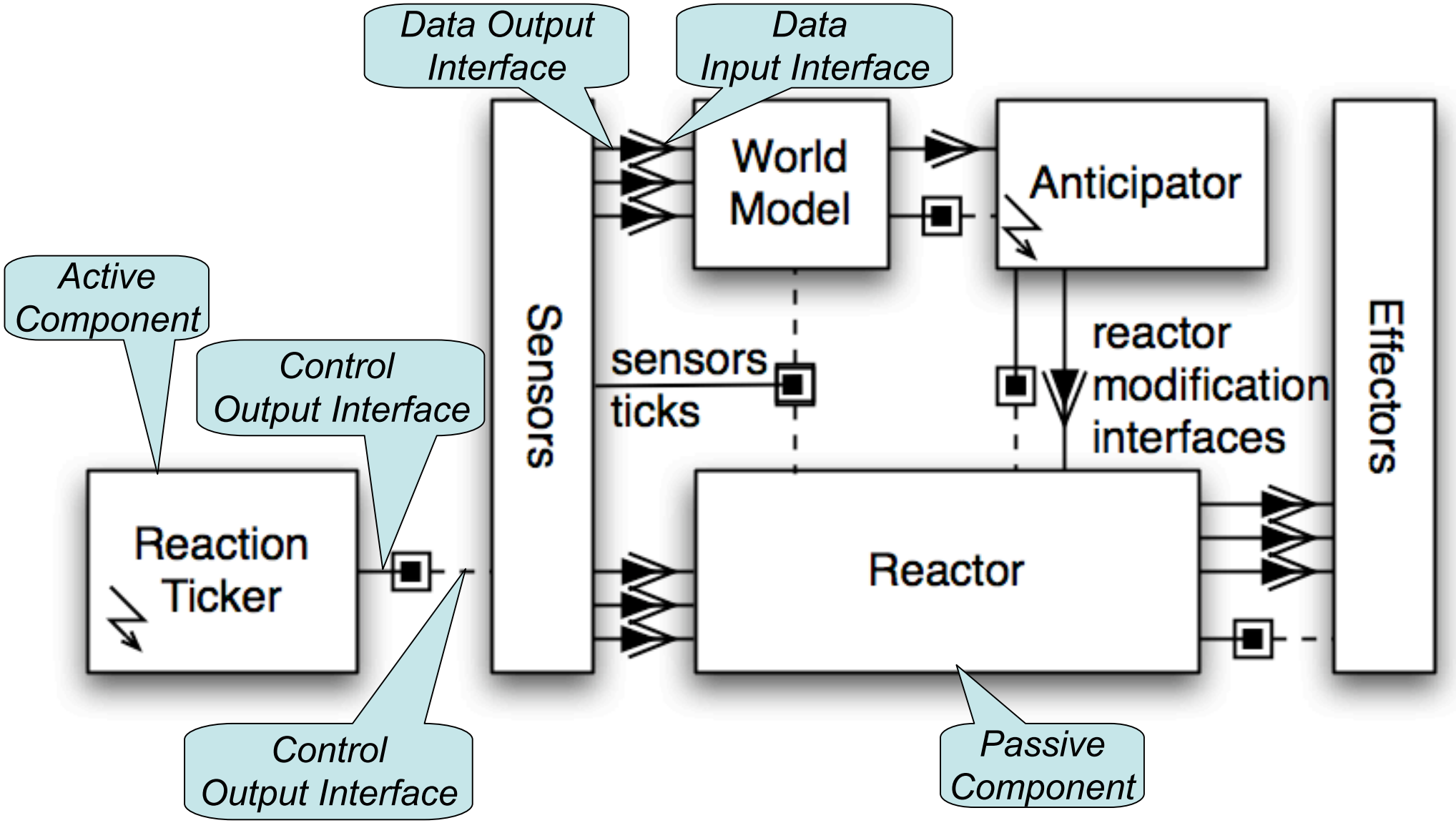
- Statically
  - No Social Layer
  - Knowledge base = World Model Only
  - Adaptation only in the local planning layer

- Dynamically
  - Adapting the Reactor
  - Updating the simulation with Reactor modifications

Implementation based on the Maleva Component Model

- Maleva components [Briot et al. 2006]
  - are run-time entities
  - with explicit **data** flows
    - A component can have multiple data inputs and outputs
    - Received data is just stored
  - with explicit **control** flows
    - A component can have multiple control inputs and outputs
    - Control reception triggers some processing
  - Can be composites
    - Built out of other components
- A component can be active or passive
  - An active component has a thread
    - Does some processing without being triggered

Maleva Based Anticipatory Agent Architecture



Conclusion

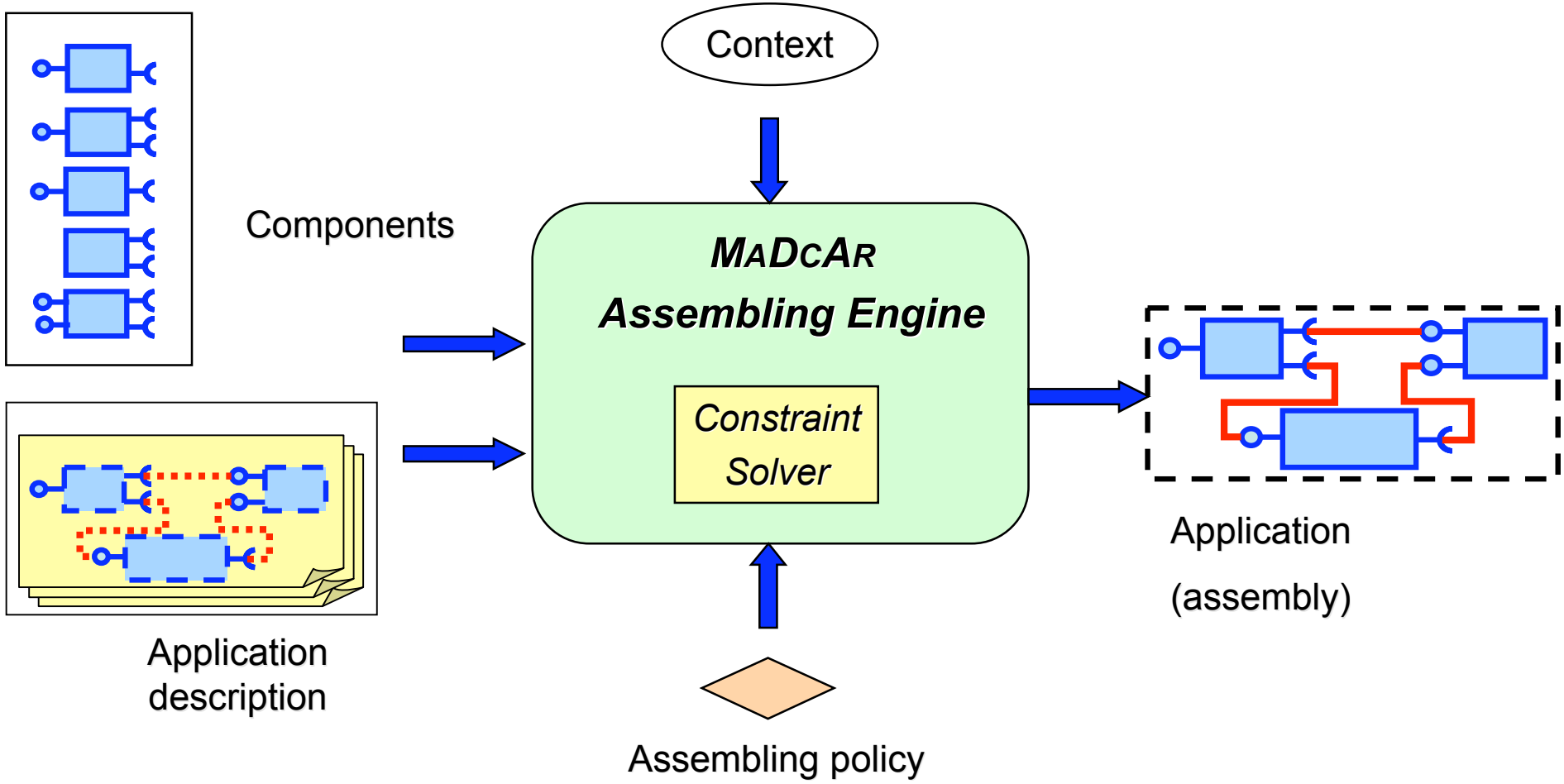
- Context = Robotic Urban Search And Rescue (USAR)
- Proposal for an Abstract Architecture Based on
  - **Hybrid agents (InteRRap)**
    - Suitable for robot control
  - **Software Components**
    - Adaptation operations =
      - ✓ Adding/Removing Components
      - ✓ Connecting/Disconnecting Components
      - ✓ Changing Components Attribute Values
  - **Reflection**
    - Autonomy
    - Adaptation to "Tune Hybridity"
    - Adaptation of adaptation strategies

Future Work

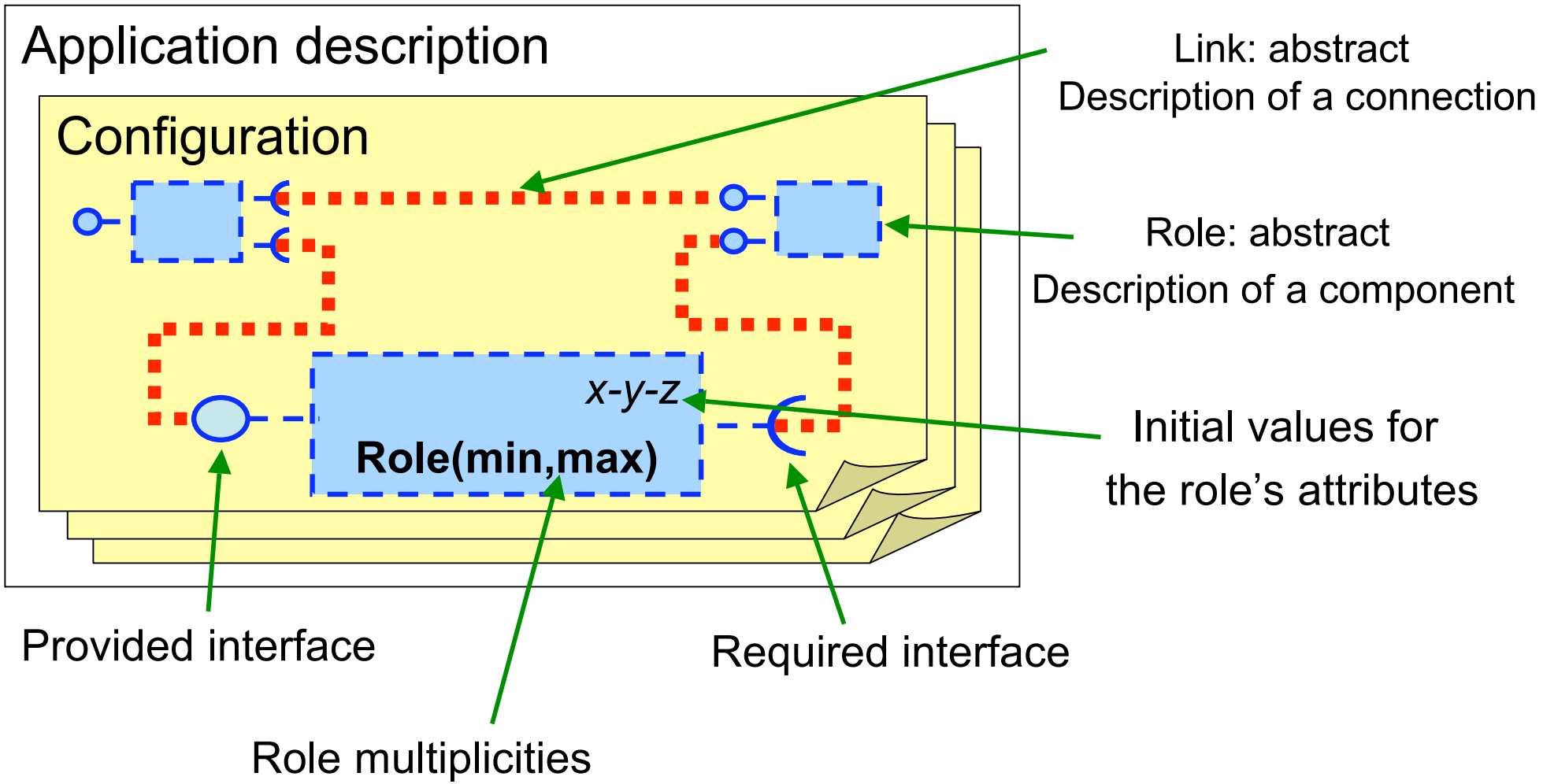
- Validation and Further Experiments
  - In the Context of Robotic USAR
  - Fleet of Robots Exploring a Damaged Area

- Link to Work on Adaptation and Reflection
  - MaDcAr [Grondin et al. 2006]
    - An Engine for Dynamic and Automatic (Re-)Assembling of Component-Based Applications
  - ARM [Malenfant 2006]
    - Asynchronous Reflection Model

MaDcAr's Assembling Engine



Application Description



Announcement

- JMAC 2007
  - Journée Multi-Agents & Composants
  - En marge des JFSMA 2007
  - 3ème édition
    - présidée par L. Vercouter (E. Mines St Etienne)
- Dates
  - Soumissions : 15 juin
  - Journée : Carcassonne 16 octobre
- Web
  - http://www.emse.fr/~vercouter/jmac07