First National Workshop on Control Architectures of Robots - April 6,7 2006 - Montpellier

CAR'06

Pleading for open modular architectures

Interests and issues about modularity and standardisation

Aurélien Godin
aurelien.godin@dga.defense.gouv.fr

Department of Defence / Angers Technical Centre

DGA

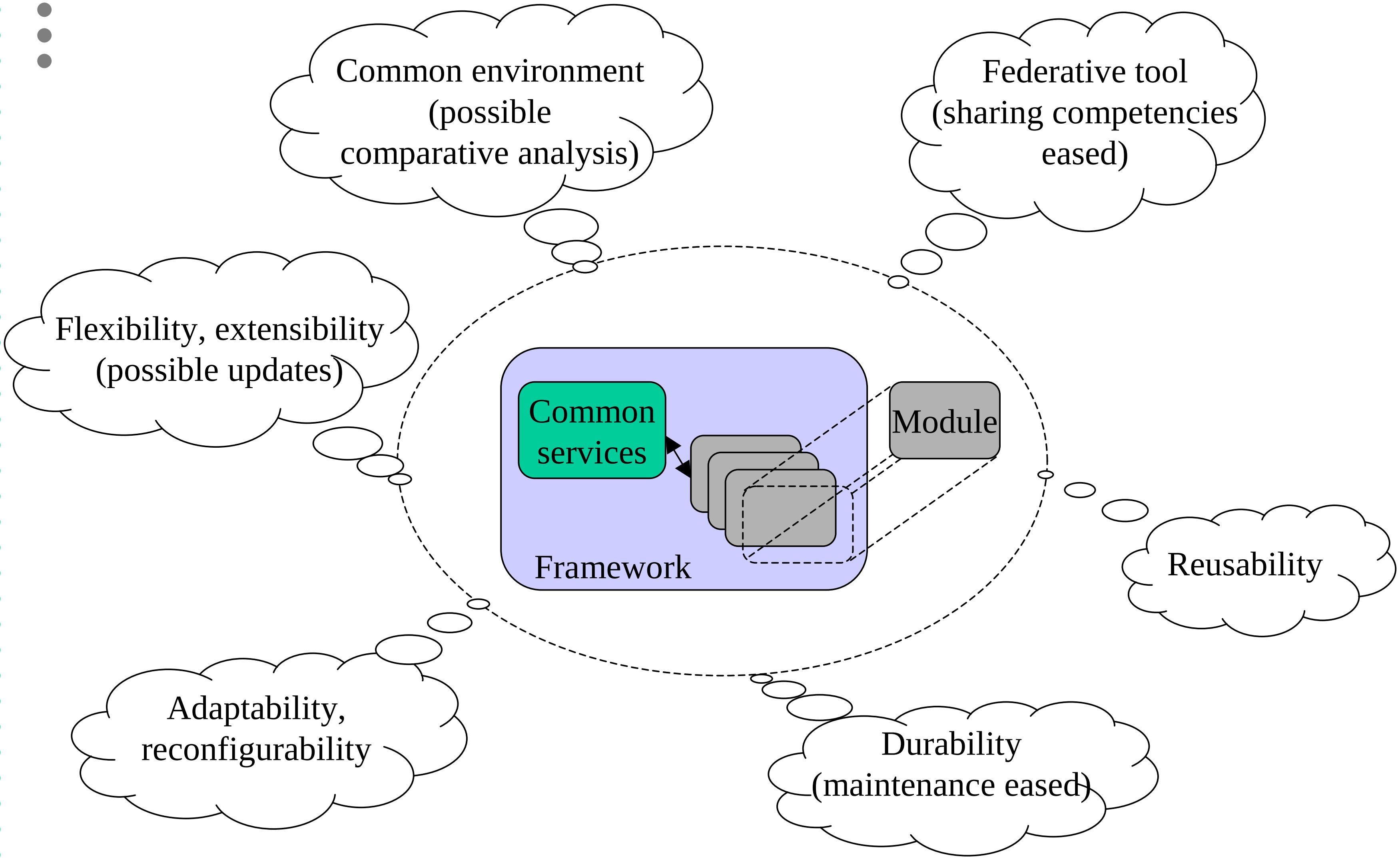
Definitions

- Architecture
  structured organisation of components that enables their simultaneous and correct execution, by offering the basic services needed for all of them

- Modularity
  ability to receive new components that were not included in the original release (definition to be completed by further discussions)

Table of contents

Interests of modularity



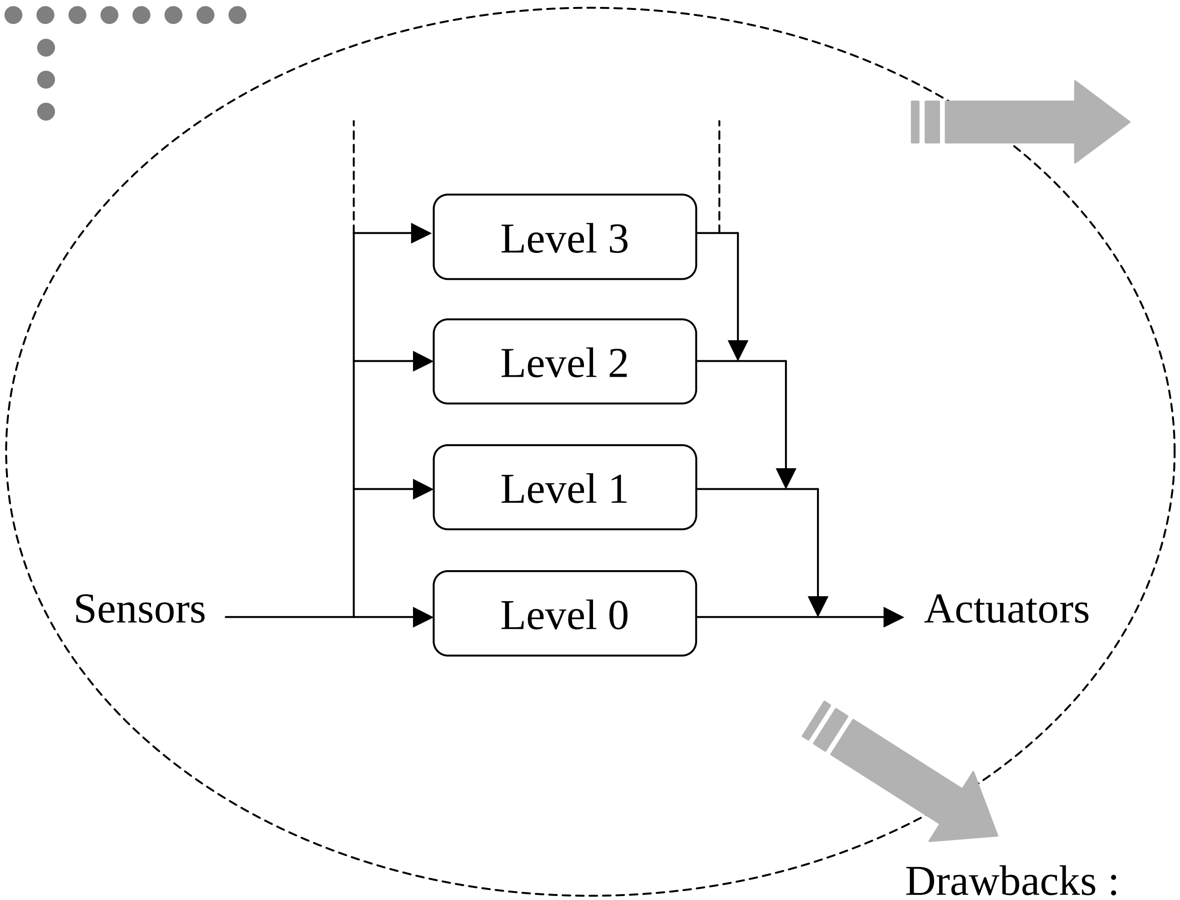
Synthesis: conditions for modularity

- Normalized data exchanges
  - public interfaces
  - common communication mechanisms
- Extensibility, with no assumptions on underlying hardware
- Flexibility, with no assumptions on missions

? Do proposed architectures meet these requirements ?

Brooks, *A robust layered control system for a mobile robot*, 1986

Review: subsumption architecture

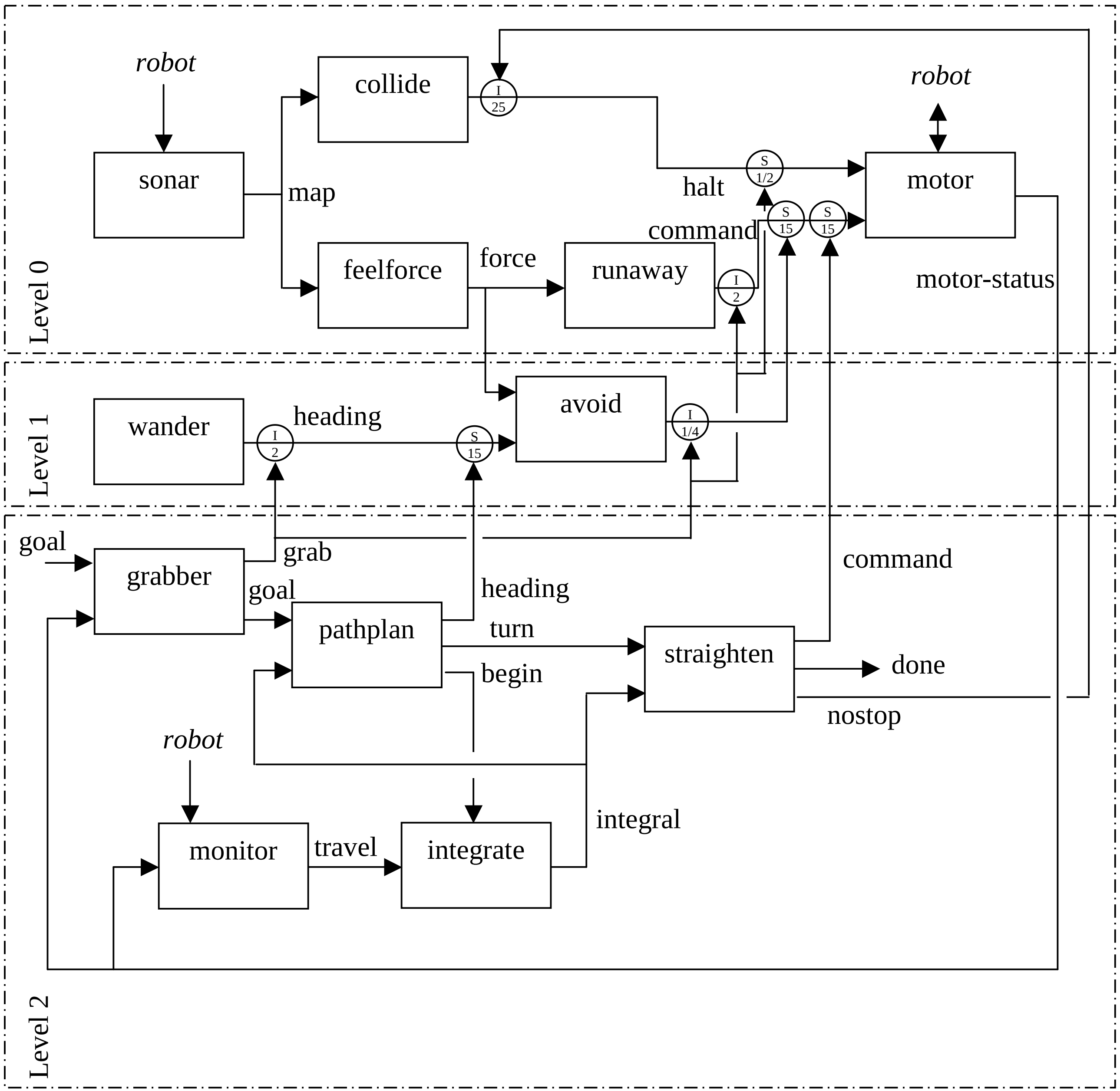


Advantages :

. First modular design ?
. Parallelism in execution

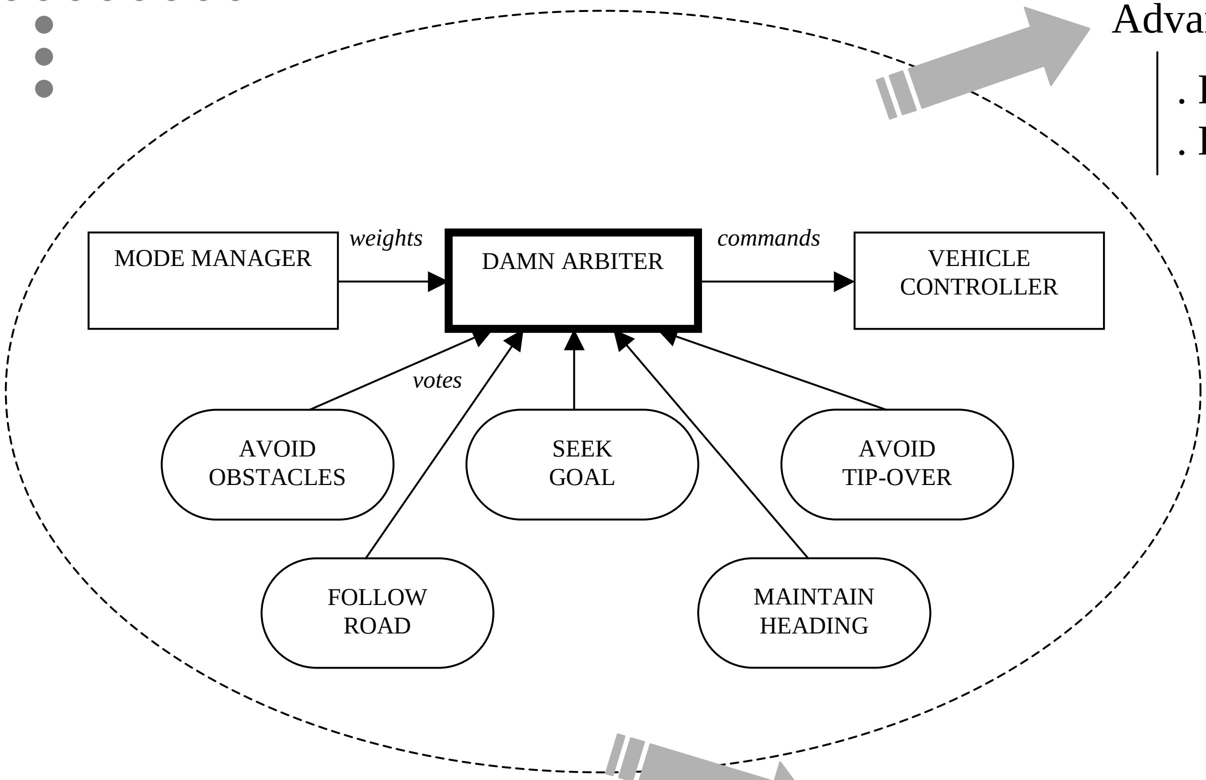
Drawbacks :

. Rigidity due to the inherent hierarchy
. Complexity of interactions between layers
. Linkage between layers "on-demand"


robot
collide
sonar
map
feelforce
force
runaway
halt
command
motor
robot
motor-status
Level 0
wander
heading
avoid
Level 1
goal
grabber
grab
goal
pathplan
heading
turn
begin
straighten
done
nostop
command
robot
monitor
travel
integrate
integral
Level 2

Rosenblatt, *DAMN: A distributed architecture for mobile navigation*, 1995

Review: DAMN



Advantages :

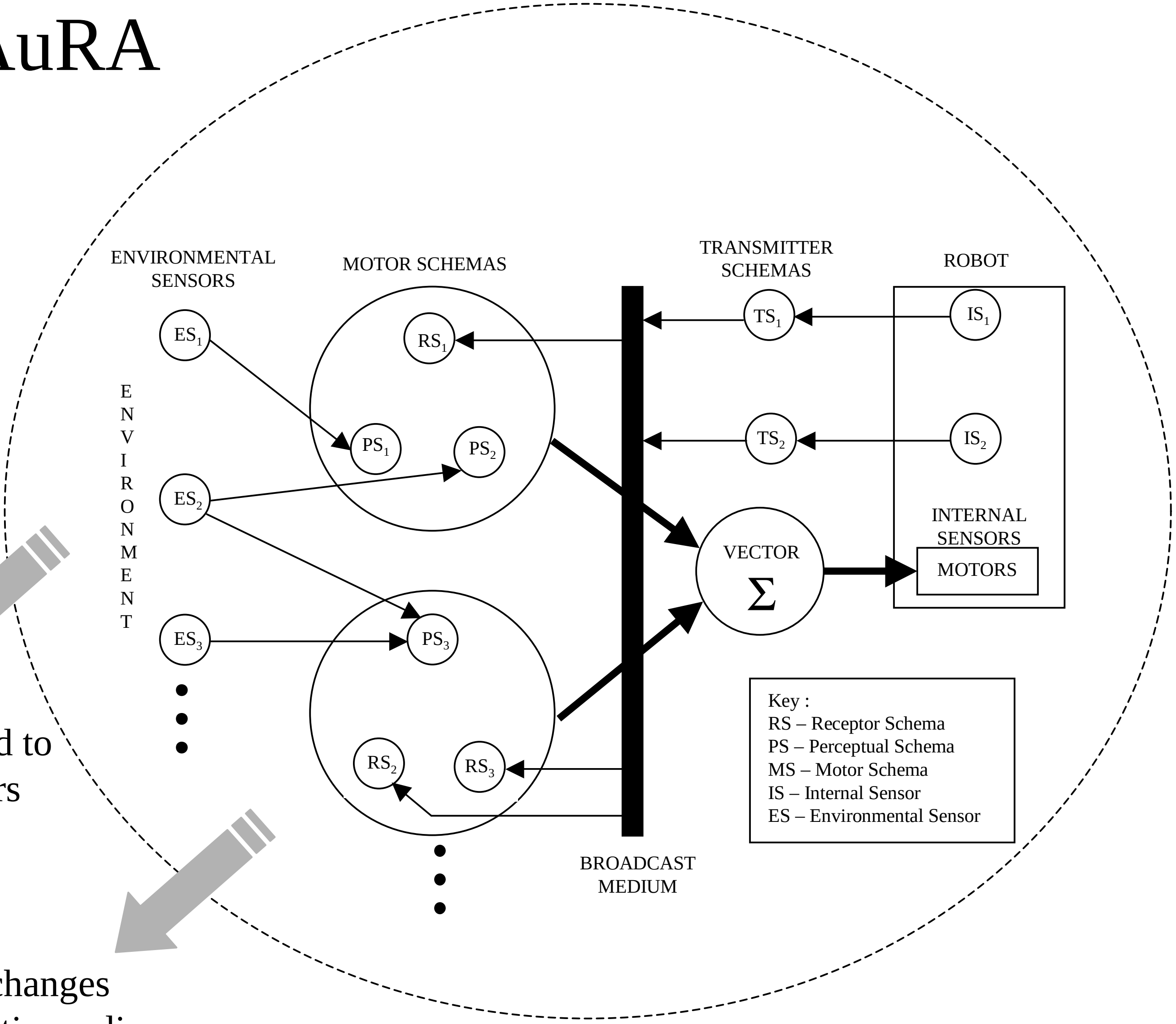
- . Flexibility
- . Homogeneous data exchanges

Drawbacks :

- . Arbiter = congestion point ?
- . Mobility oriented
- . Votes are not real standardized data (not generic enough)

Arkin, Balch, *AuRA: principles and practice in review*, 1997

Review: AuRA



Drawbacks :

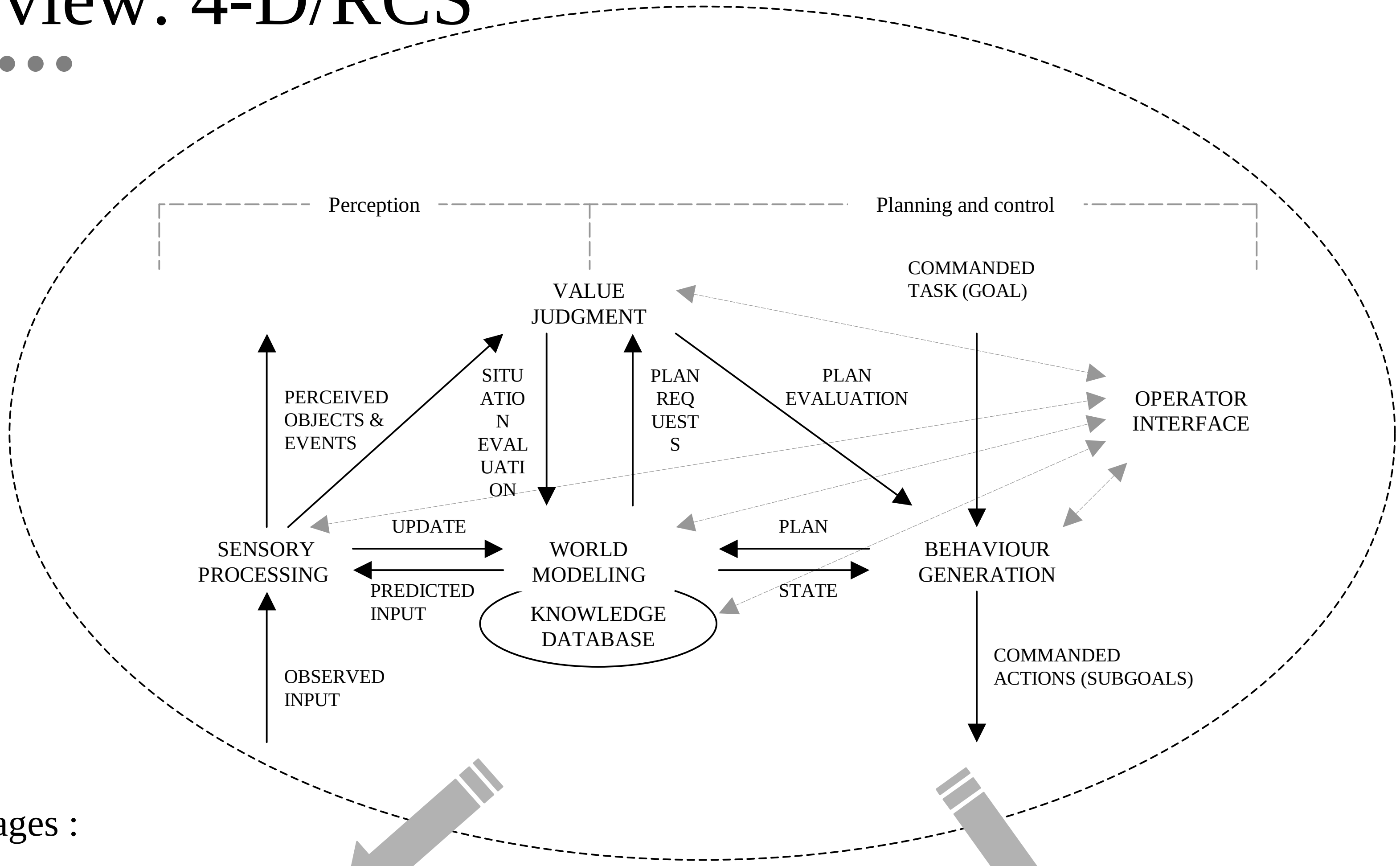
. Modularity limited to reactive behaviours

Advantages :

. Flexibility
. Common data exchanges through homeostatic medium

Albus, *4-D/RCS: a reference model architecture for Demo III*, 1997

Review: 4-D/RCS



Advantages :

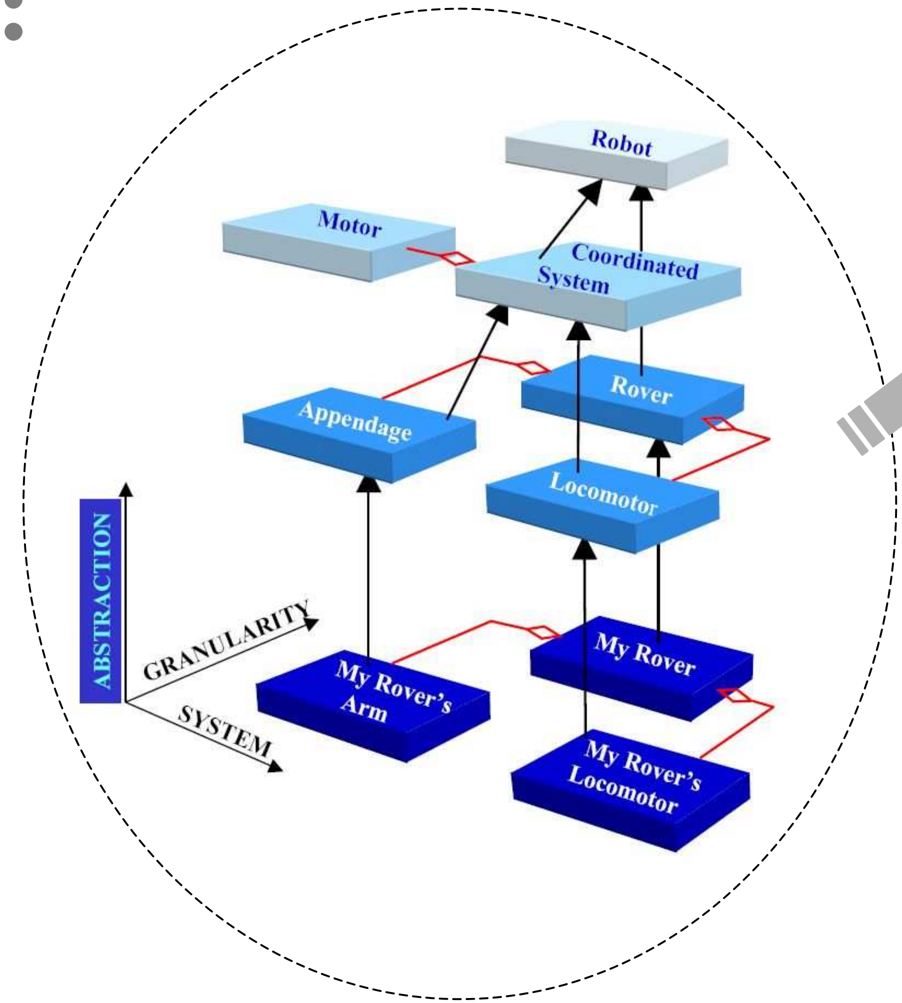
. Clear I/O interfaces
. Explicit human interface
. Modularity at all levels

Drawback

. Complexity for implementation ??

Volpe et al., *CLARAty: coupled layered architecture for robotic autonomy*, 2000

Review: CLARAty



Advantages :

. Clear hierarchy of modules
. Takes benefit from object-oriented design power (inheritance, encapsulation…)
. No assumptions on the nature of behaviours internal implementation

Performances of existing frameworks

- Technology Readiness Levels (TRLs)

Low maturity	
1	Basic principles of technology observed & reported
2	Technology concept and/or application formulated
3	Analytical and laboratory studies to validate analytical predictions
Medium maturity	
4	Component and/or basic sub-system technology valid in lab environment
5	Component and/or basic sub-system technology valid in relevant environment
6	System/sub-system technology model or prototype demo in relevant environment
High maturity	
7	System technology prototype demo in an operational environment
8	System technology qualified through test & demonstration
9	System technology ‘qualified’ through successful mission operations

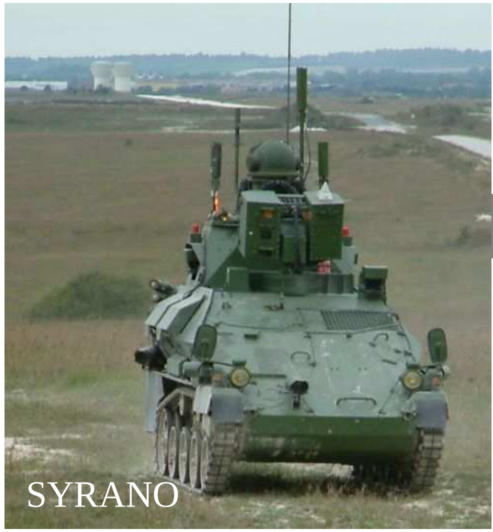
Performances of existing framework

- System Readiness Levels (SRLs)

1	User requirements defined
2	System requirements defined
3	Architectural design refined
4	Detailed design is nominally complete
5	Sub-systems verification in laboratory environment
6	Sub-system verification in representative integration environment
7	System prototype demonstration in a representative integration environment
8	Pre-production system completed and demonstrated in a representative operational environment
9	System proven through successful representative mission profile

Performances of existing frameworks

TRL-5/6



TRL-6

TRL-6/7

RACKHAM

SRL-6/7 ----→

Current programmes:
SRL-2 ????

Limitations

- Most approaches were developed concurrently: no exchanges possible between them
- This issue is even more central when tackling the multi-robot problem (scalability and interchangeability needs)
- Current challenge: allowing **interoperability**

⇒ need for a **standard.**

American example: JAUS

- Joint Architecture for Unmanned Systems
- Adopted in 1998 by American DoD
- Five targets:
  - support for all classes of unmanned systems
  - rapid technology insertion
  - interoperable operating control units
  - interchangeable/interoperable payloads
  - interoperable unmanned systems

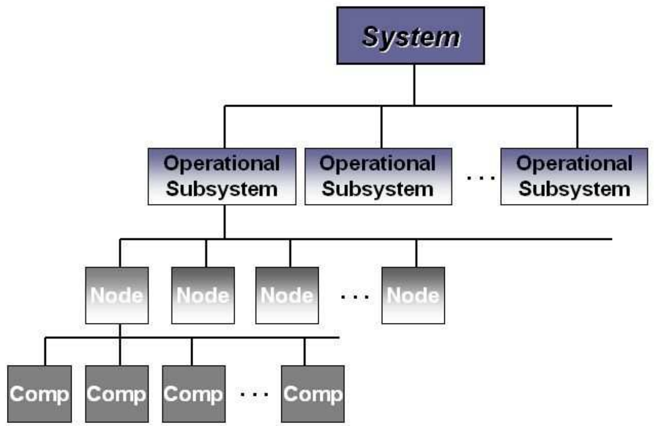
⇒ Component based, message-passing framework

JAUS specifications

- Domain model
  - Defines constraints on messages (independency)
  - Provides a roadmap to define operational requirements (functional capabilities, FC)
  - Defines categories for informational capabilities (IC)
  - Device groups may interface with FC and IC

JAUS specifications

- Reference Architecture Specification
  - Details all functions and messages employed by new components
  - Defines rules that govern messaging
  - Depicts the structure of a JAUS compatible system



JAUS additional references

- Document control plan
  - Process to update the DM and the RA
- Standard operating procedures
  - Organisation of the JAUS working group
- Compliance plan

⇒ A whole framework defining the architecture characteristics as well as the rules to be respected to develop a JAUS-compliant system

A multi-domain issue

- Same preoccupations in UAVs fields: "defining open, standardised, modular and evolutionary architecture for a generic and interoperable UAV system"
- Main technical axes:
  - Configurable and generic architecture
  - Interchangeability of payloads (plug and play)
  - Secure, certifiable and everlasting systems

French efforts towards standardisation

- BOA context
  - Kind of network centric warfare
  - Aims at proposing new organisations for ground forces
  - Requires a high degree of interoperability
- Challenge
  - Aero-terrestrial system: UAVs + UGVs
  - High communication needs
  - Sharing information between UXVs and human units
  - Reconfigurable systems
  - Obligation for an open framework (integration of modules provided by other programmes)

A federative programme

- OISAU: open and interoperable autonomous systems
- A coherent programme that aims at enabling:
  - Platform/hardware independency
  - Cost reduction (by standardisation)
  - Easy integration and replacement of functionalities
  - Incremental integration/replacement
- No assumption on target domain: could potentially apply to UGVs, UAVs (UUVs ?)

Conclusion

- Many interests in modular architectures (technical, commercial, practical)
- Technologically achievable as shown by existing systems

But …

- Real interoperable systems **require a higher level of specification**: standardisation
- The **role of humans** in the systems will also have consequences on the definition of the architecture: **urgent need** for researches on this issue

Thank you to all of you !

Any questions ?