

An Architectural Strategy for Self-Adapting Systems

Danny Weyns and Tom Holvoet
Katholieke Universiteit Leuven, Belgium

SEAMS
Minneapolis, May 26, 2007

Outline

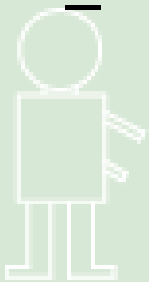
- Setting
- Synthesizing Architectural Knowledge
- Architectural Approaches and Architectural Strategy
- Conclusions and Future Work

Setting

- Multiagent systems to study and engineer distributed systems characterized by
 - Dynamism and change
 - Important quality goals: flexibility and openness
 - Inherent distribution of resources & locality of activity
 - Central control hard to achieve
- **Self-adaptive systems**
 - Perspective: the ability of a software system to manage dynamic and changing operating conditions autonomously

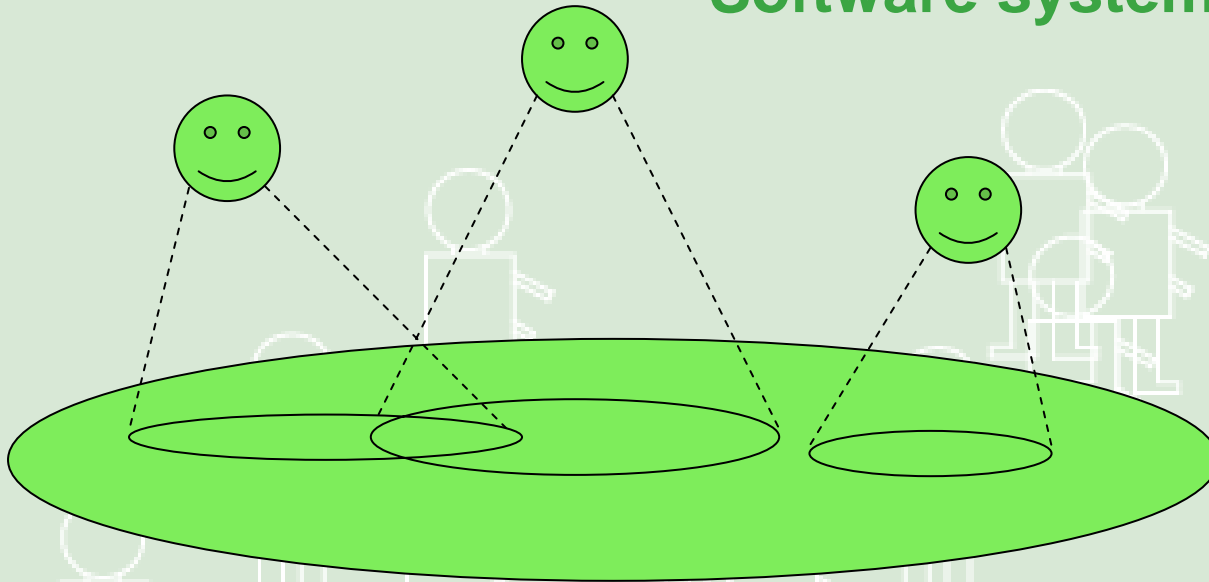
Situated Multiagent Systems

- **Approach to structure the software**
 - Set of autonomous entities (agents) that cooperate to provide system functionality
 - Agents can flexibly adapt to dynamics and changes
- **Decentralized control**
 - Deal with inherent distribution of resources and locality of activity



Situated Multiagent Systems

Software system



Agents

Agent Environment

- Share information
- Mediate interaction

External World

System Environment

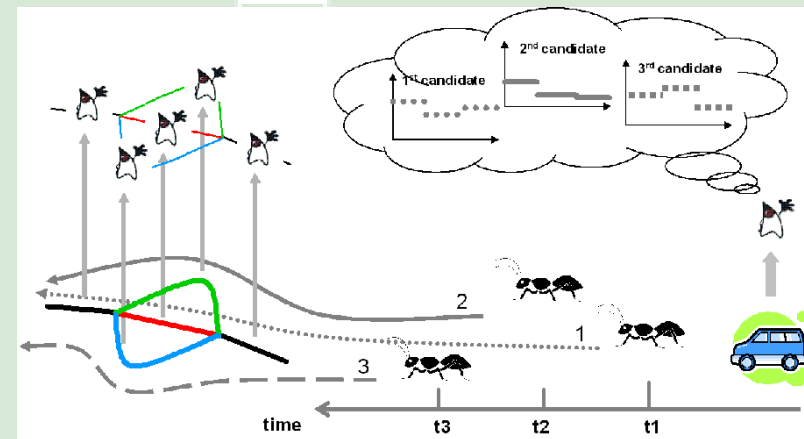
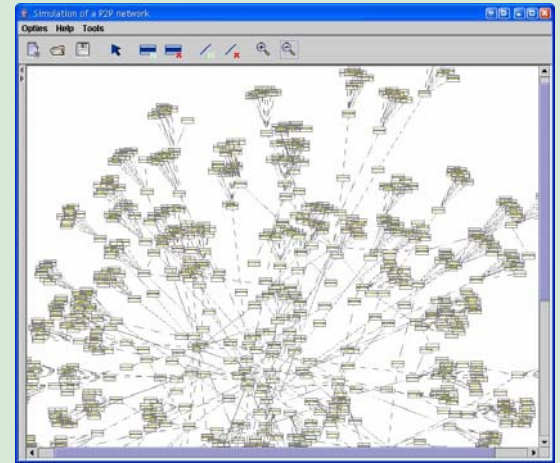
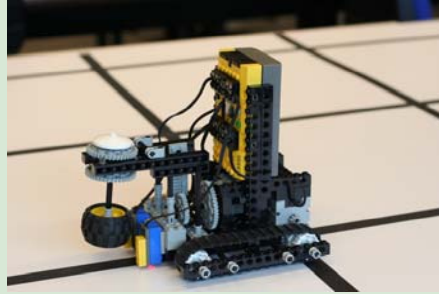
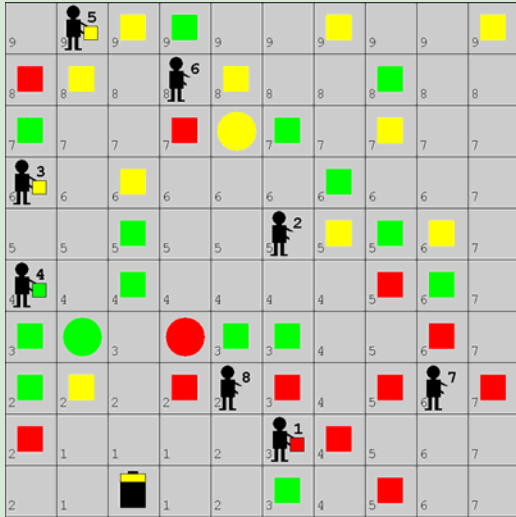
Outline

- Setting
- Synthesizing Architectural Knowledge
- Architectural Approaches and Architectural Strategy
- Conclusions and Future Work

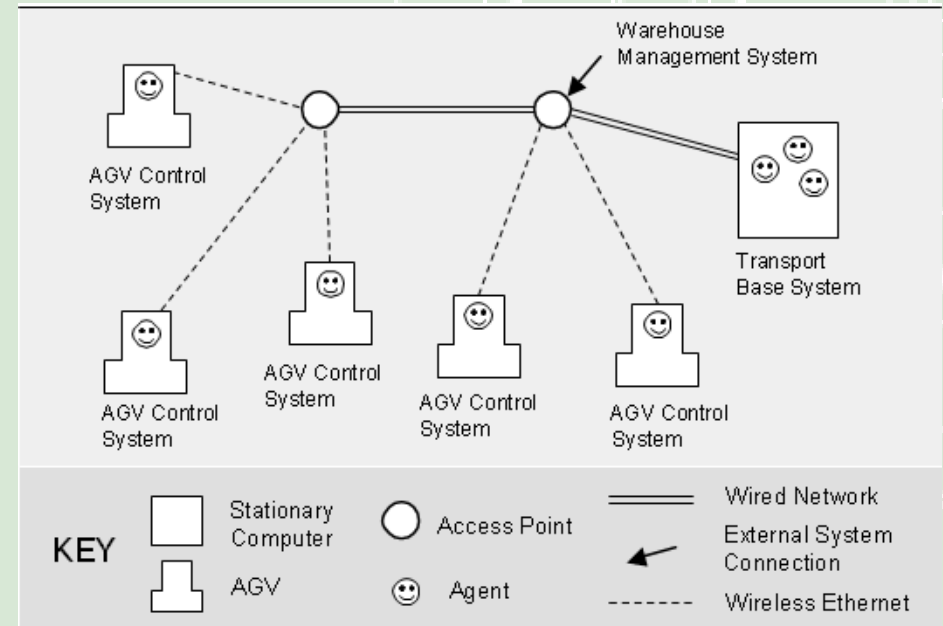
Synthesizing Architectural Knowledge

- From studying and building various applications
 - We derived a set of architectural patterns
 - These patterns are integrated with one another
- We call this integrated set of patterns an **architectural strategy**
 - Vehicle for study and communication
 - Blueprint for developing new systems with similar properties and characteristics

Example applications



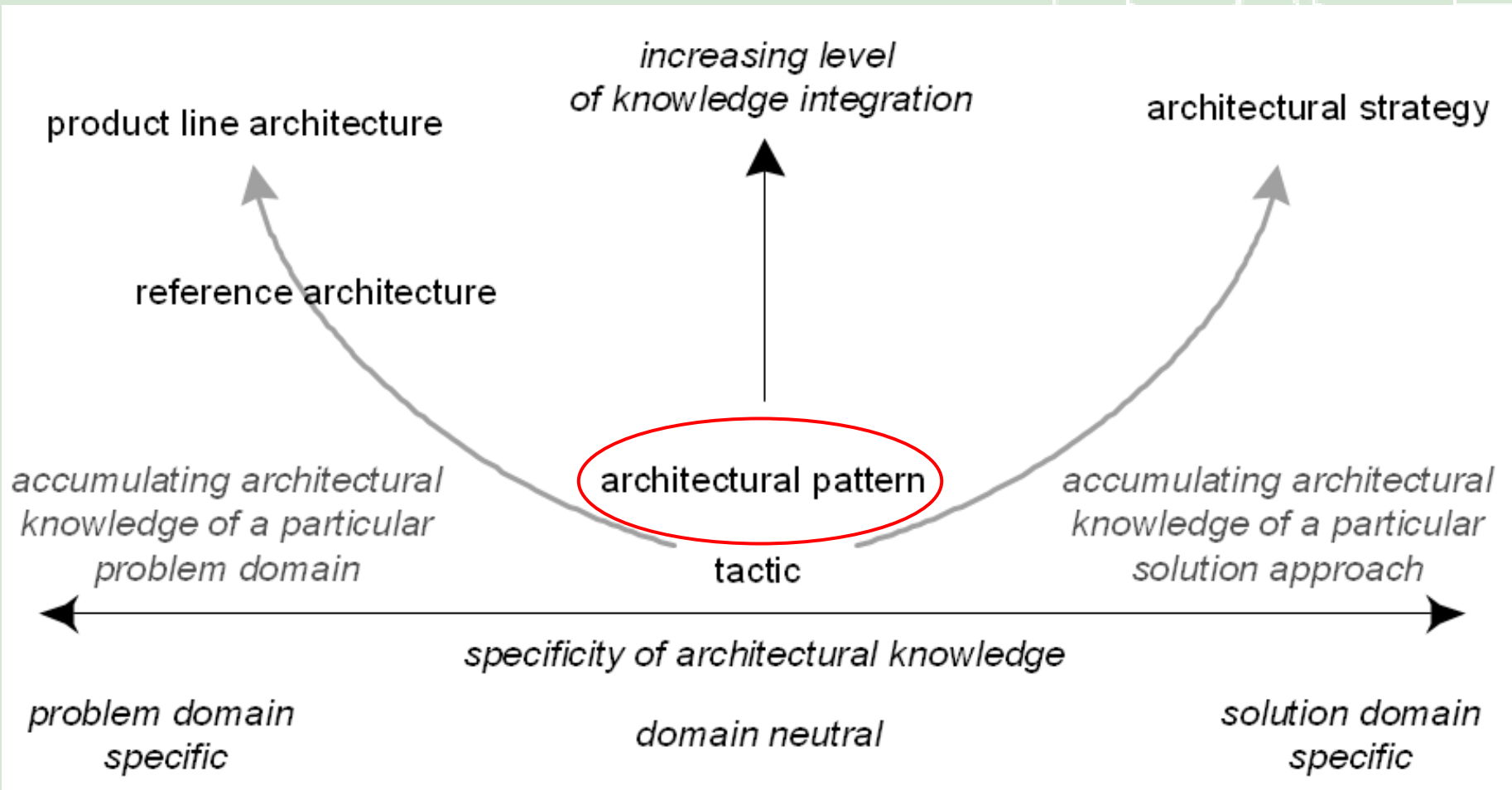
High Level Model AGV System



Outline

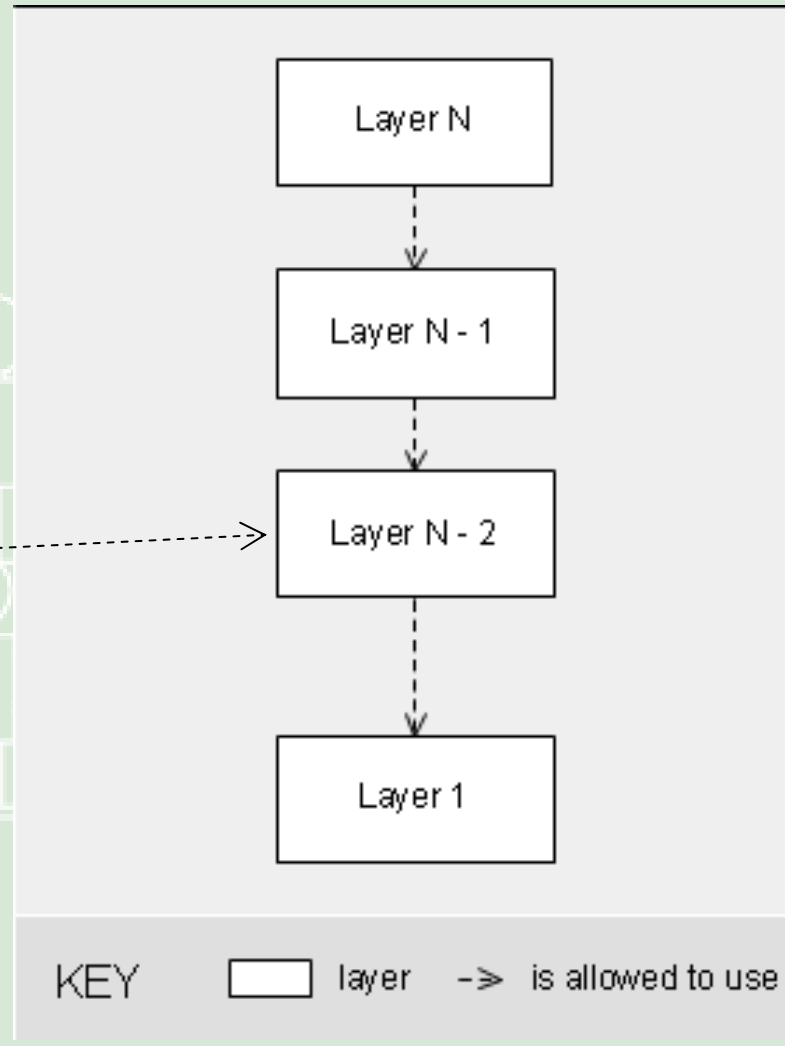
- Setting
- Synthesizing Architectural Knowledge
- Architectural Approaches and Architectural Strategy
- Conclusions and Future Work

Types of Architectural Approaches

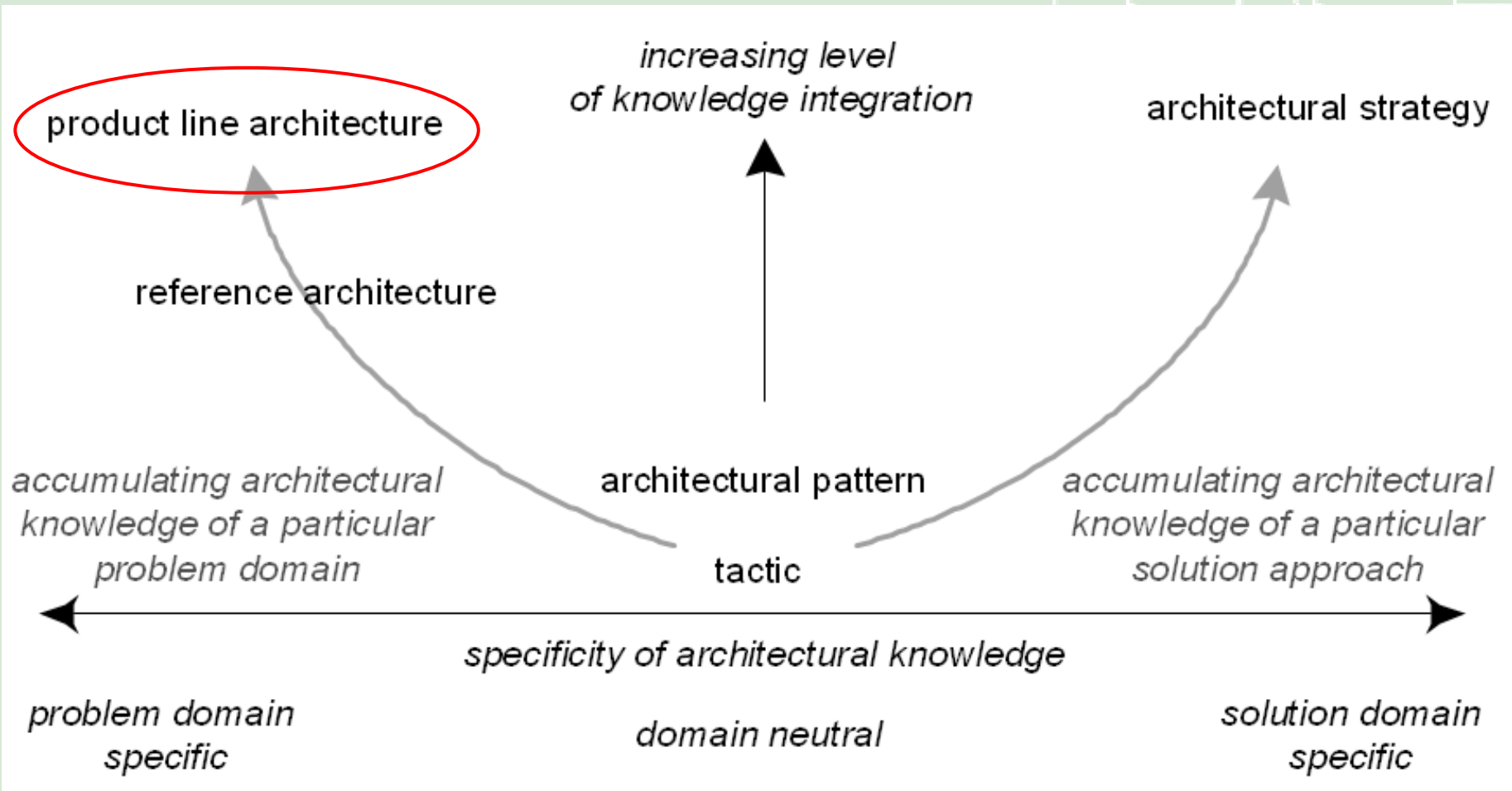


Layers Pattern

Architectural elements are neutral w.r.t. problem and solution domain

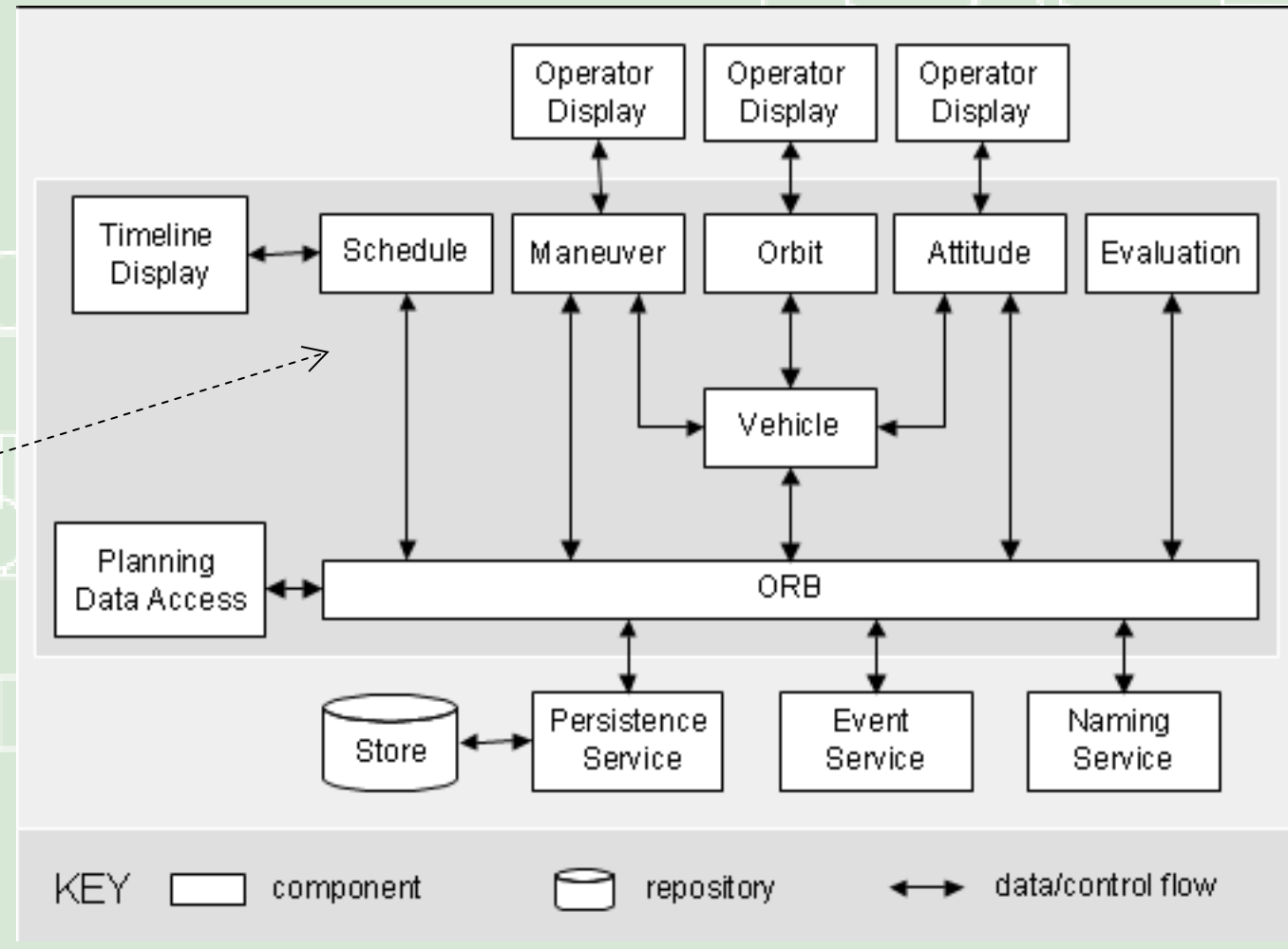


Types of Architectural Approaches

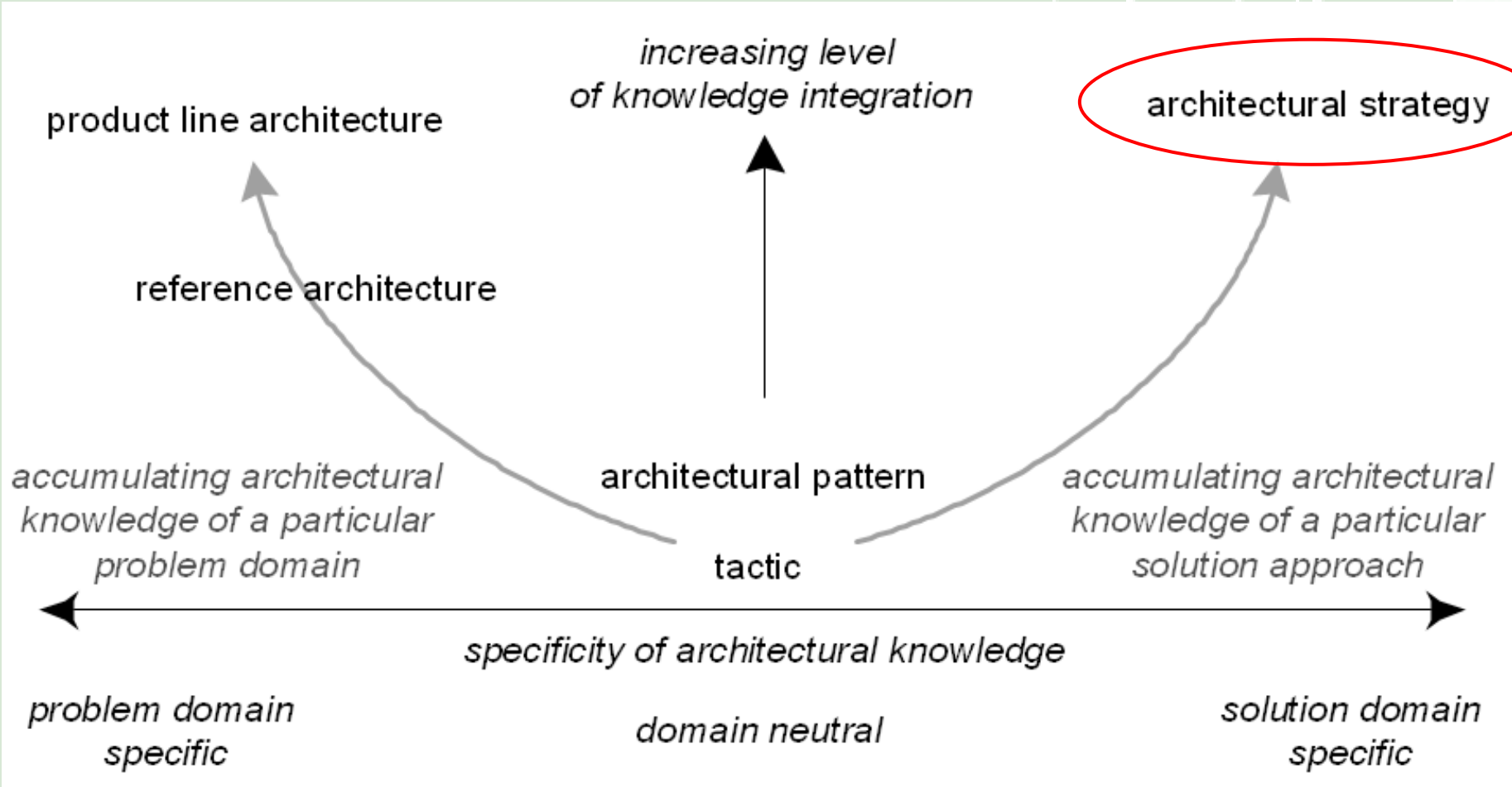


Excerpt of Product Line Architecture for Satellite Control

Architectural elements are specific to particular problem domain

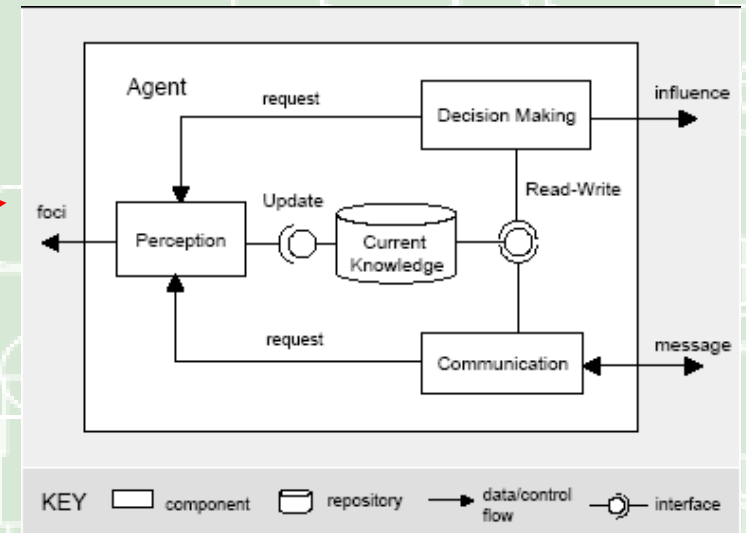
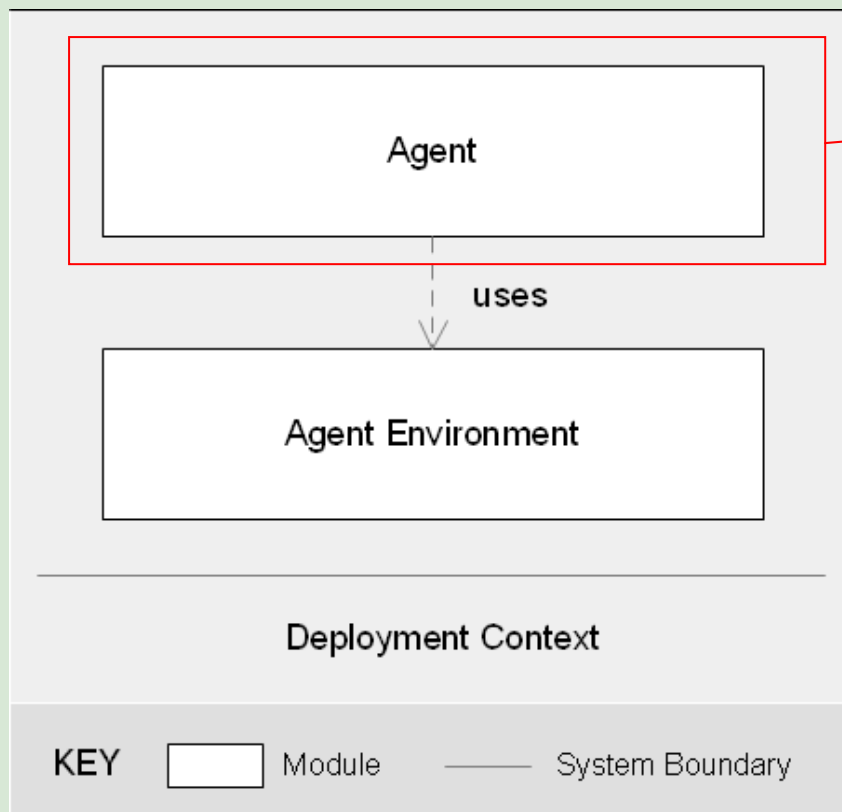


Types of Architectural Approaches

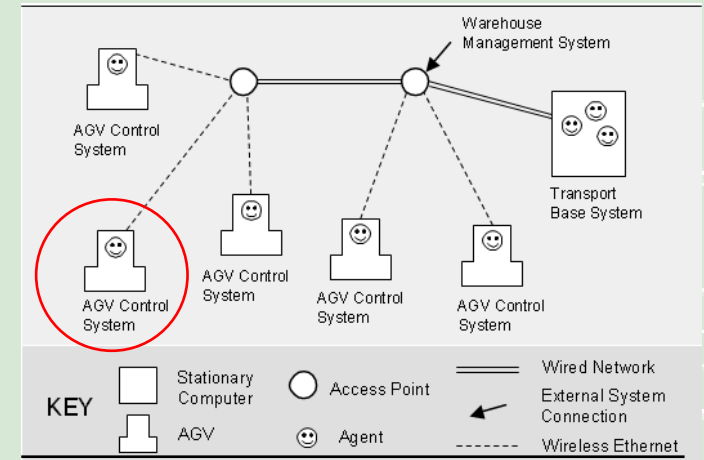
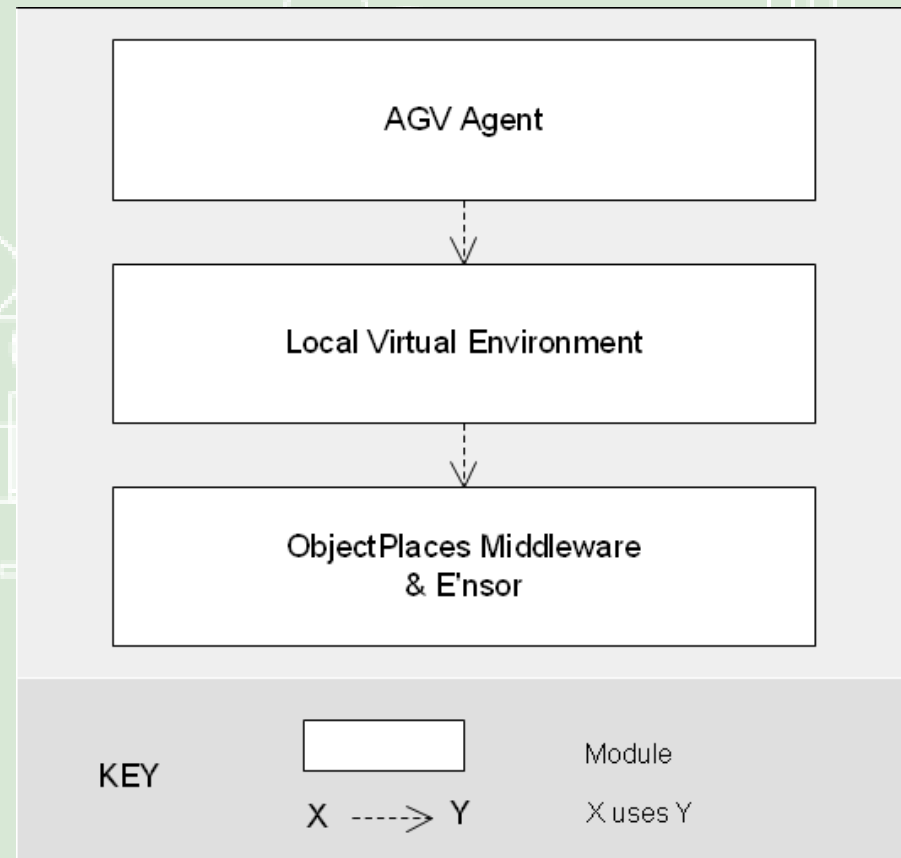
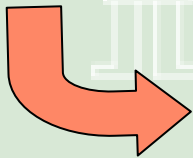
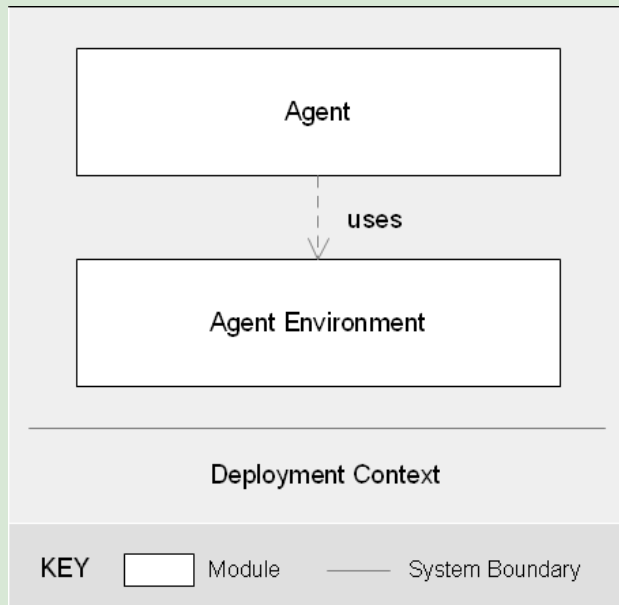


Excerpt Architectural Strategy

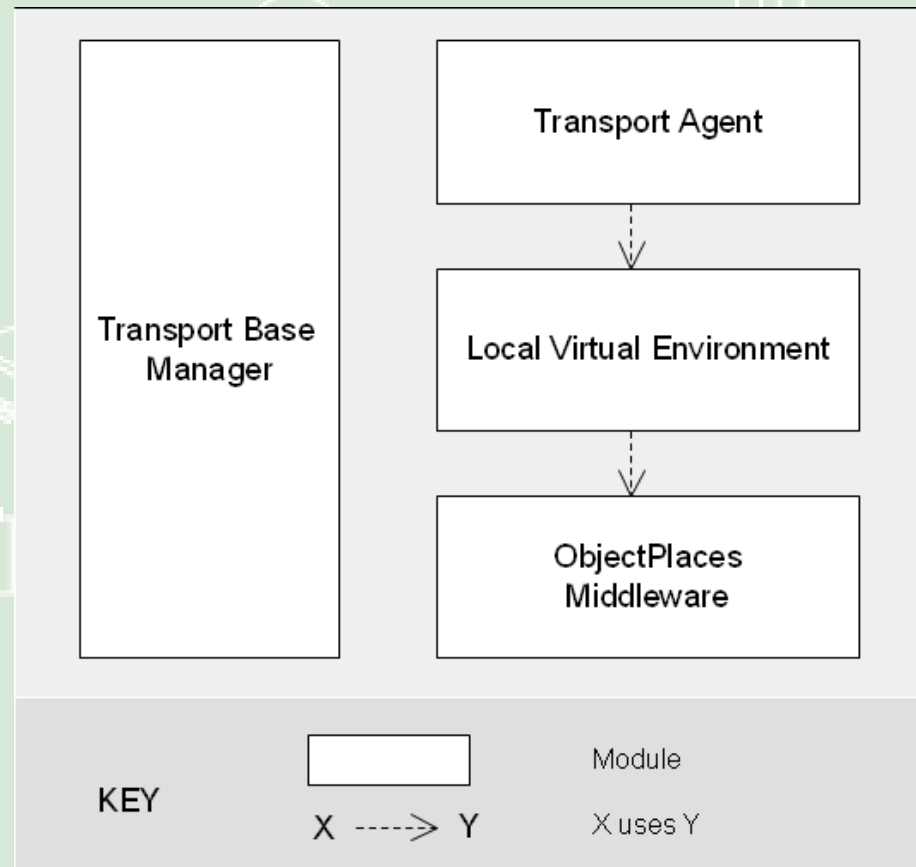
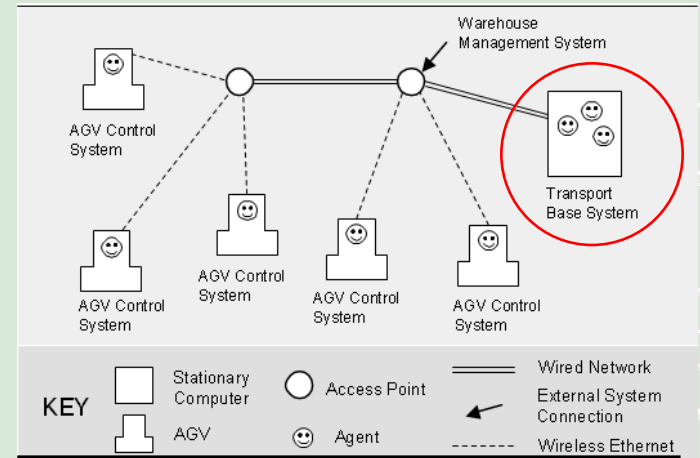
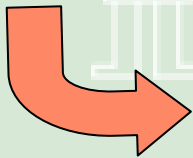
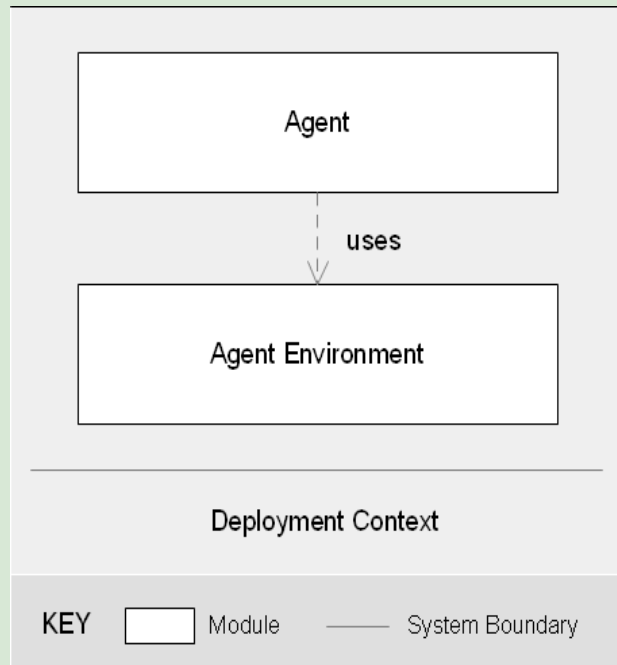
Top Level Module View



Top Level Module View AGV Control System

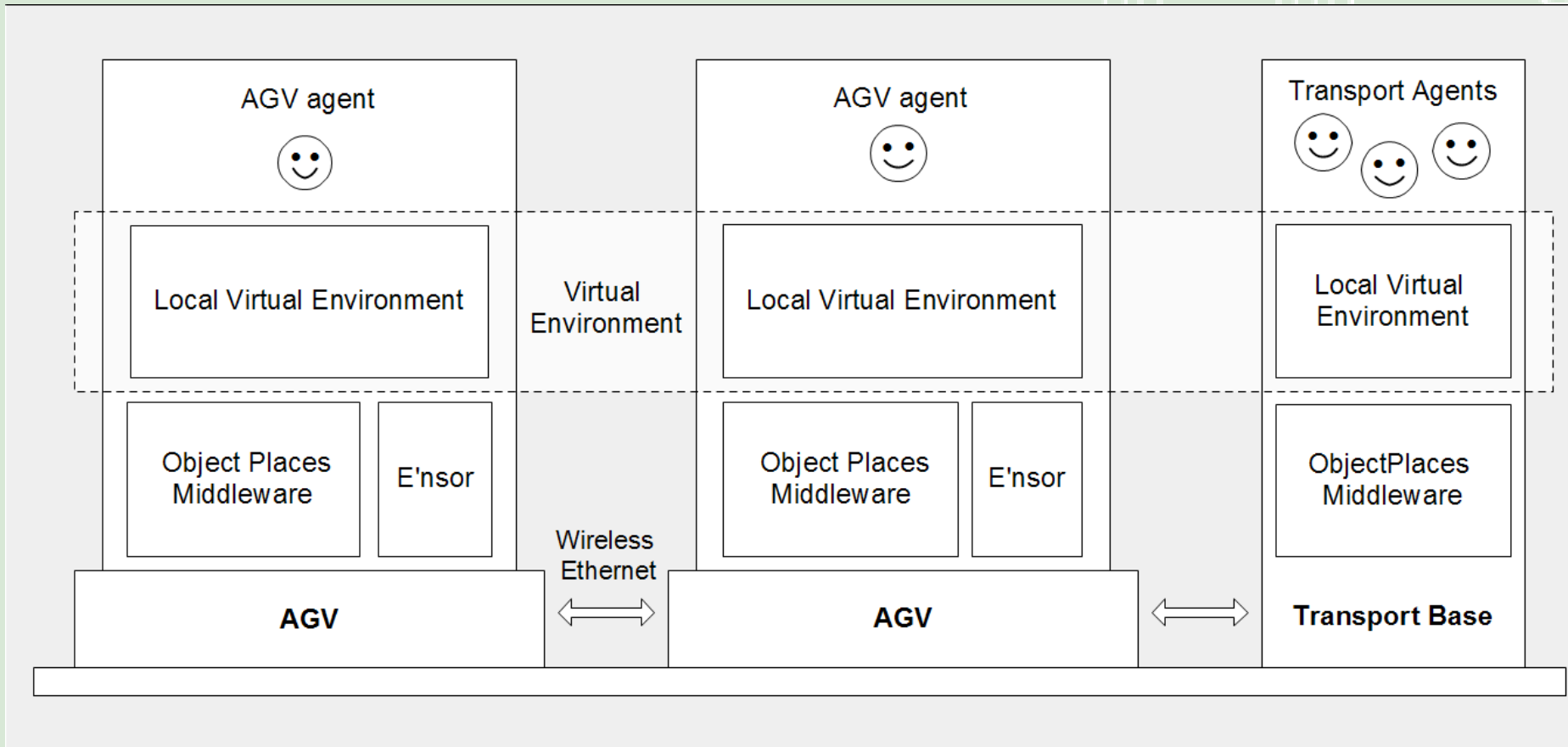
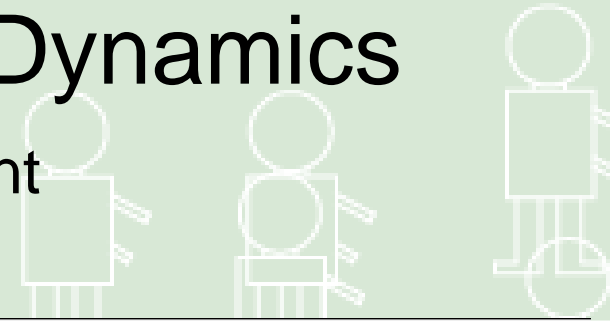


Top Level Module View Transport Base System

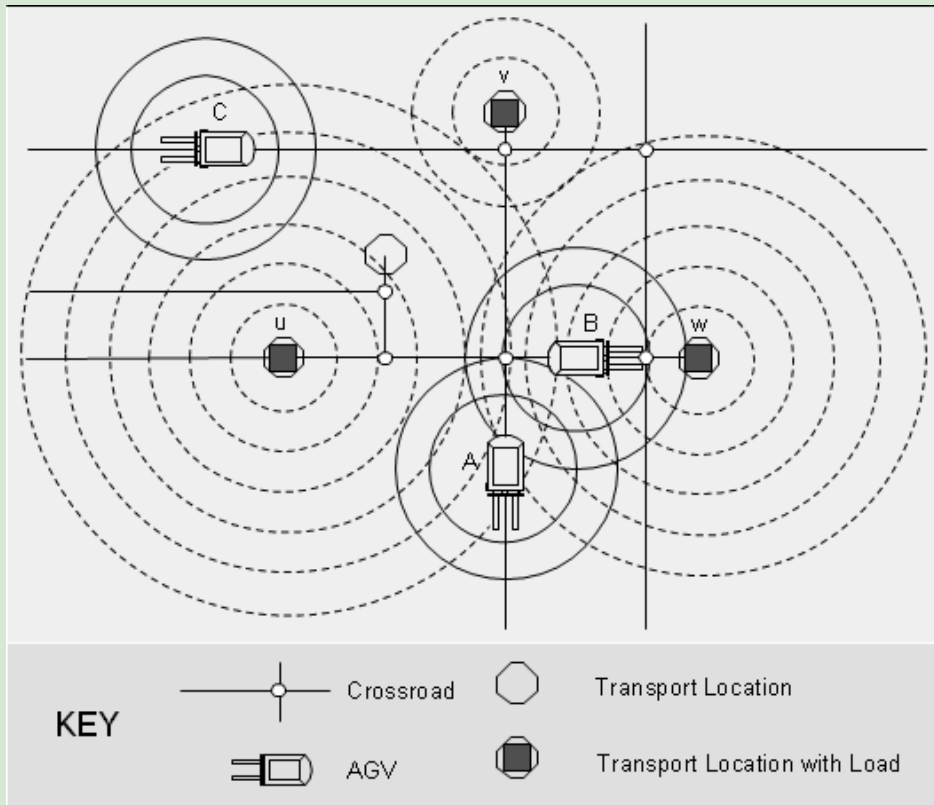


Agents Flexibly Adapt to Dynamics

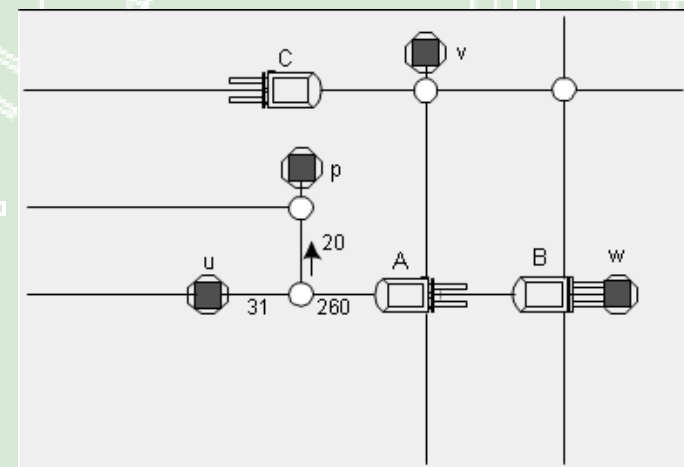
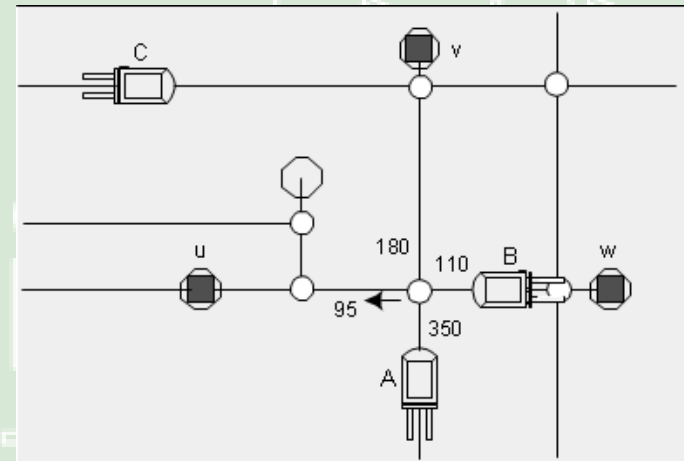
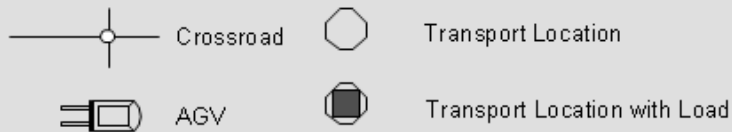
Example: Task Assignment



Field-Based Task Assignment



KEY



Outline

- Setting
- Synthesizing Architectural Knowledge
- Architectural Approaches and Architectural Strategy
- **Conclusions and Future Work**

Conclusions

- Architectural strategy for situated multiagent systems
 - Synthesizes architectural knowledge about solution domain
 - Blueprint for designing self-adaptive systems with similar characteristics and requirements
- Future work: define a formally founded ADL for decentralized systems
 - Core issues: locality & dynamics

Thanks for your attention!
Questions?