



Laboratório de Engenharia de Software

Fine-grained Variability in the Development of Families of Software Agents

Ingrid Nunes, Elder Cirilo,
Donald Cowan and Carlos Lucena

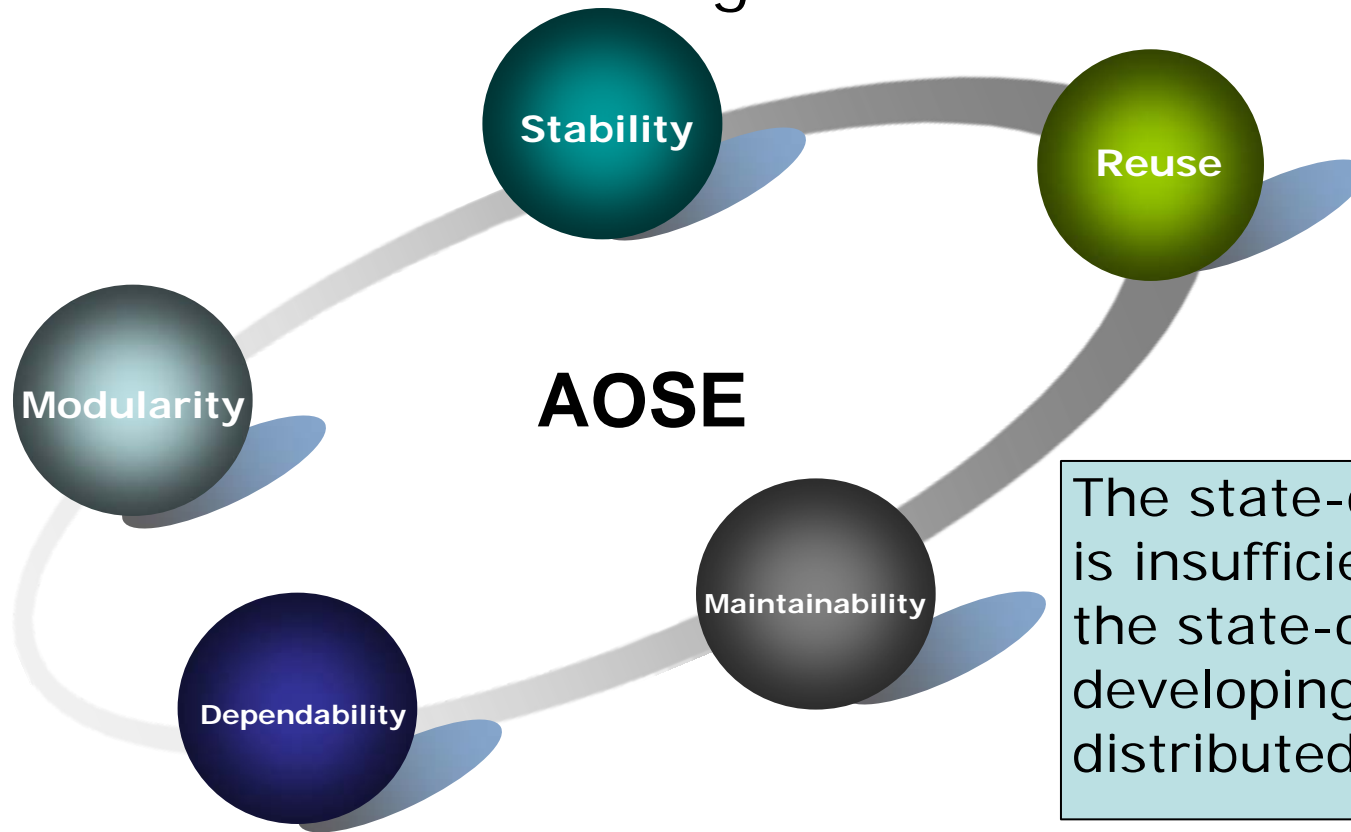
Agenda



- 1 Introduction
- 2 Buyer Agent Family Case Study
- 3 Lessons Learned
- 4 Conclusion

Introduction

- ❖ MAS in Software Engineering → Paradigm
 - Decomposition of a complex problem into autonomous agents



The state-of-the-art of AOSE is insufficiently reflected in the state-of-practice in developing complex distributed systems.

This work

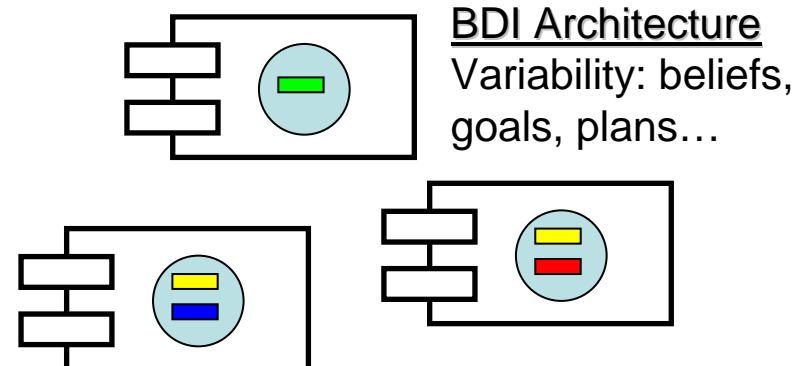
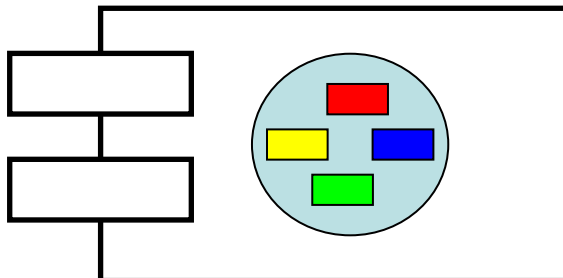
Exploratory Study

of the development of a family of agents, focused on fine-grained variability. The Buyer Agent SPL.

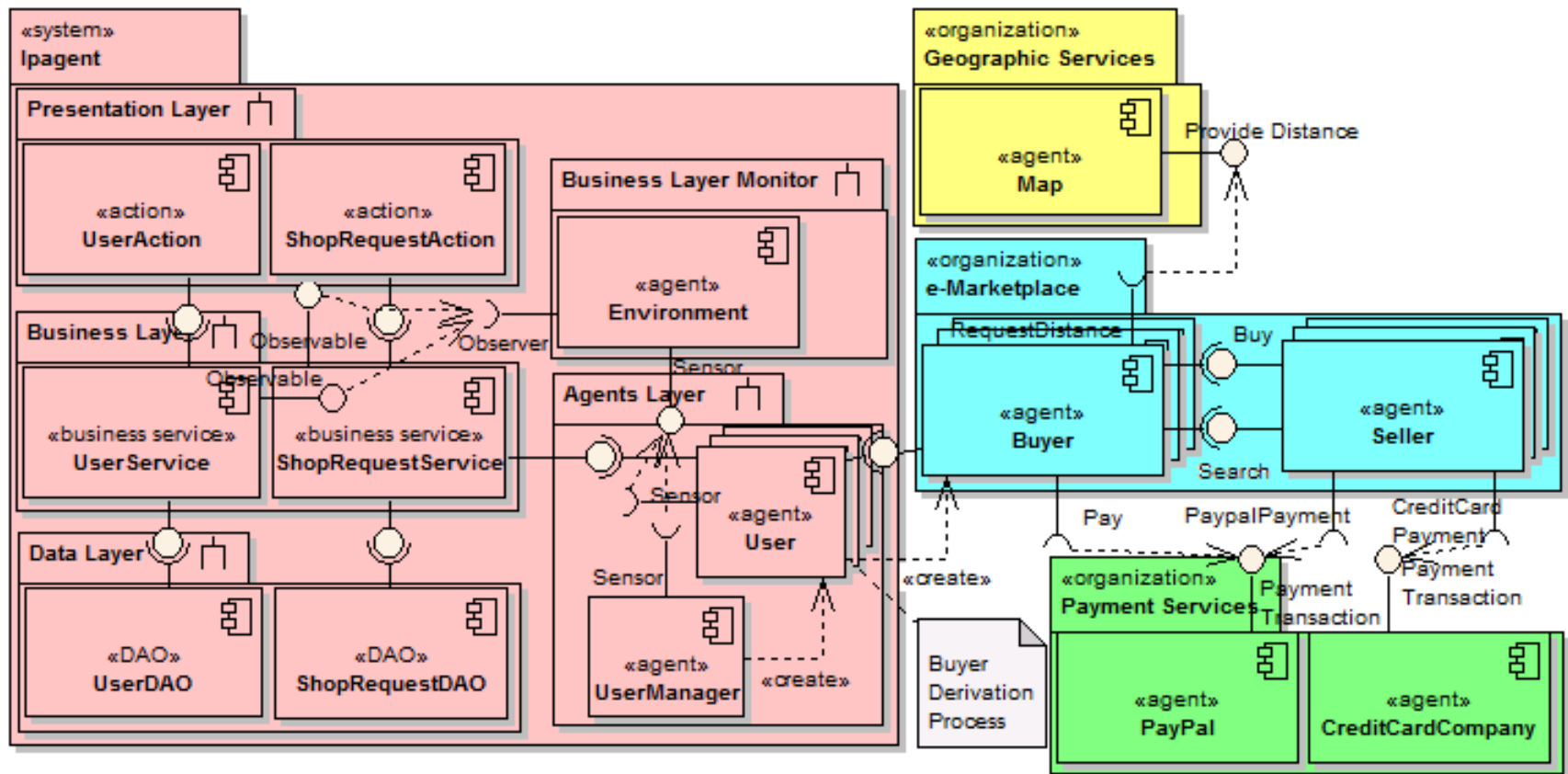
Lessons Learned

mainly related to techniques to build agent architectures that take into account SE principles.

Why a Software Product Line?



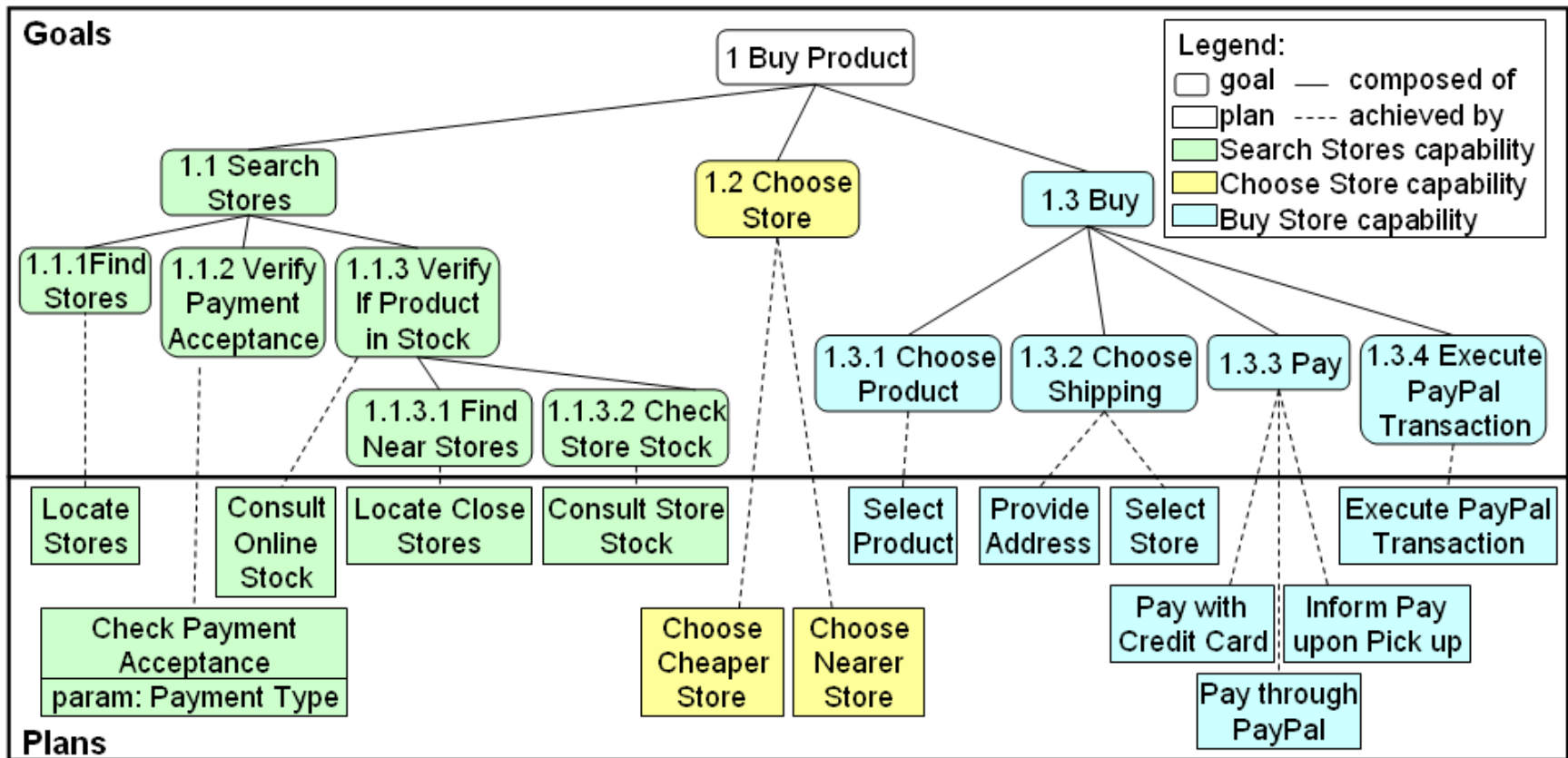
The Buyer Agent Family Case Study



The Buyer Agent Family Case Study **LES**



❖ Variability

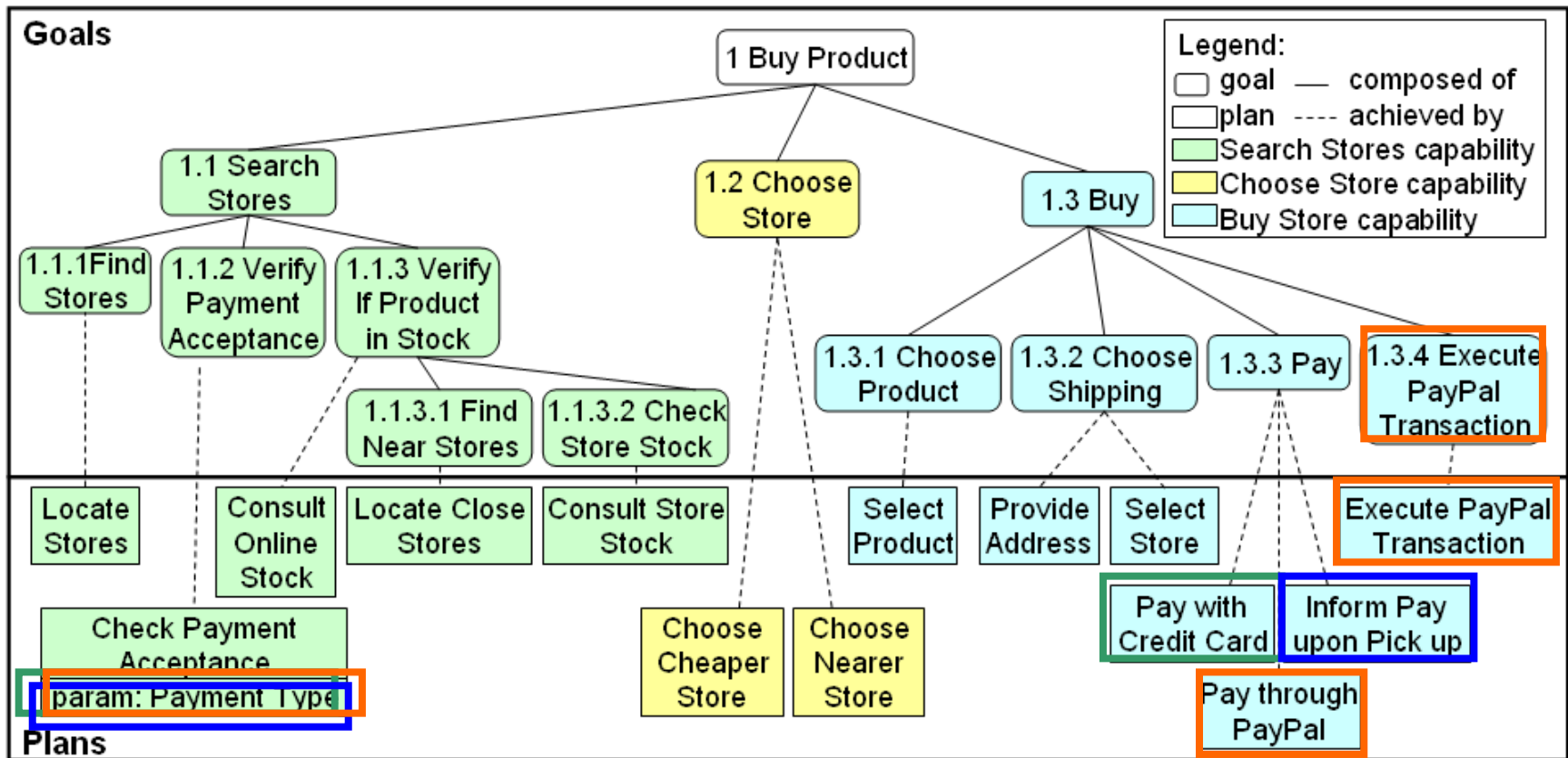


The Buyer Agent Family Case Study **LES**



❖ Variability

- Payment Type

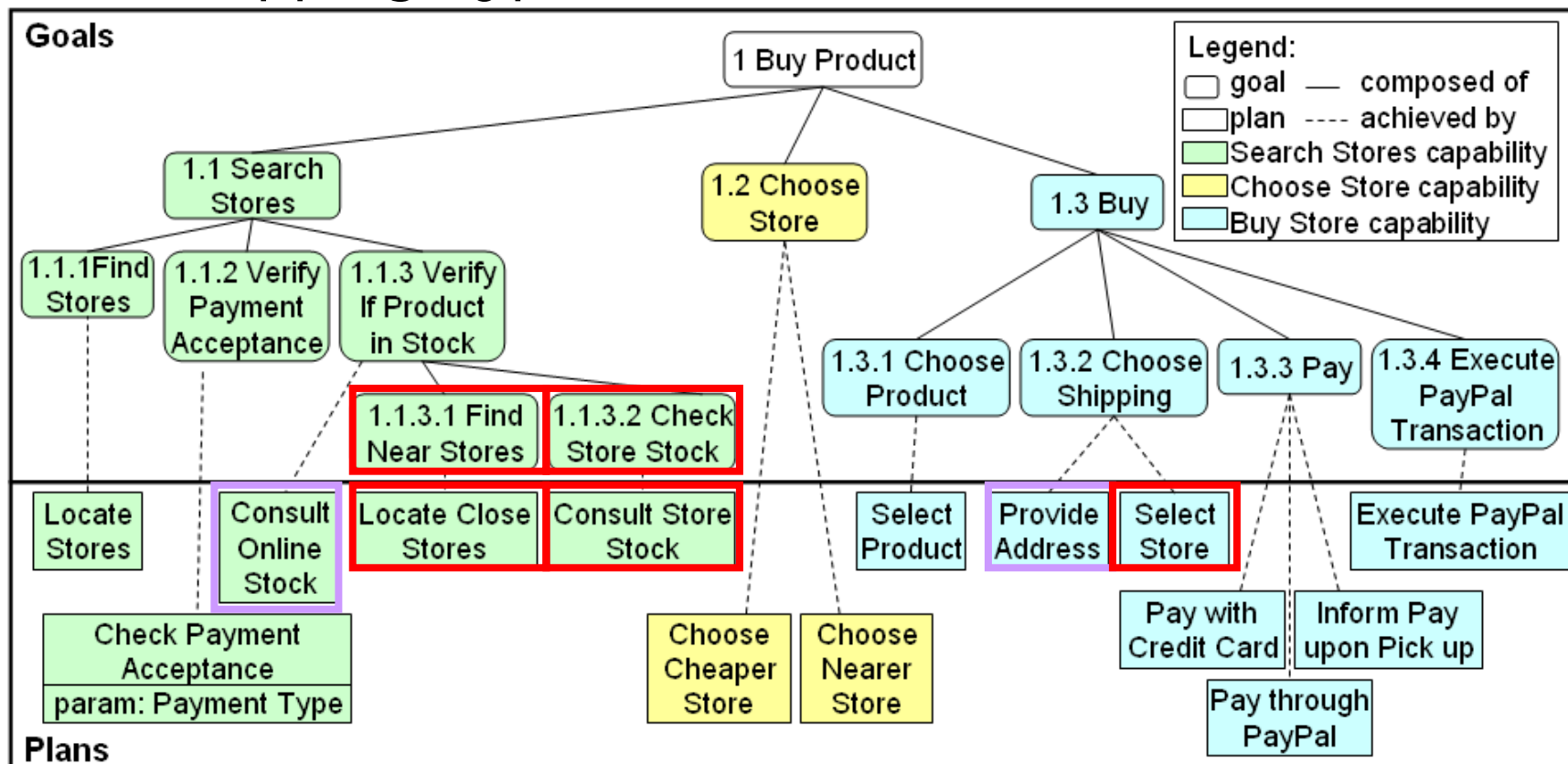


The Buyer Agent Family Case Study **LES**



❖ Variability

- Shipping Type

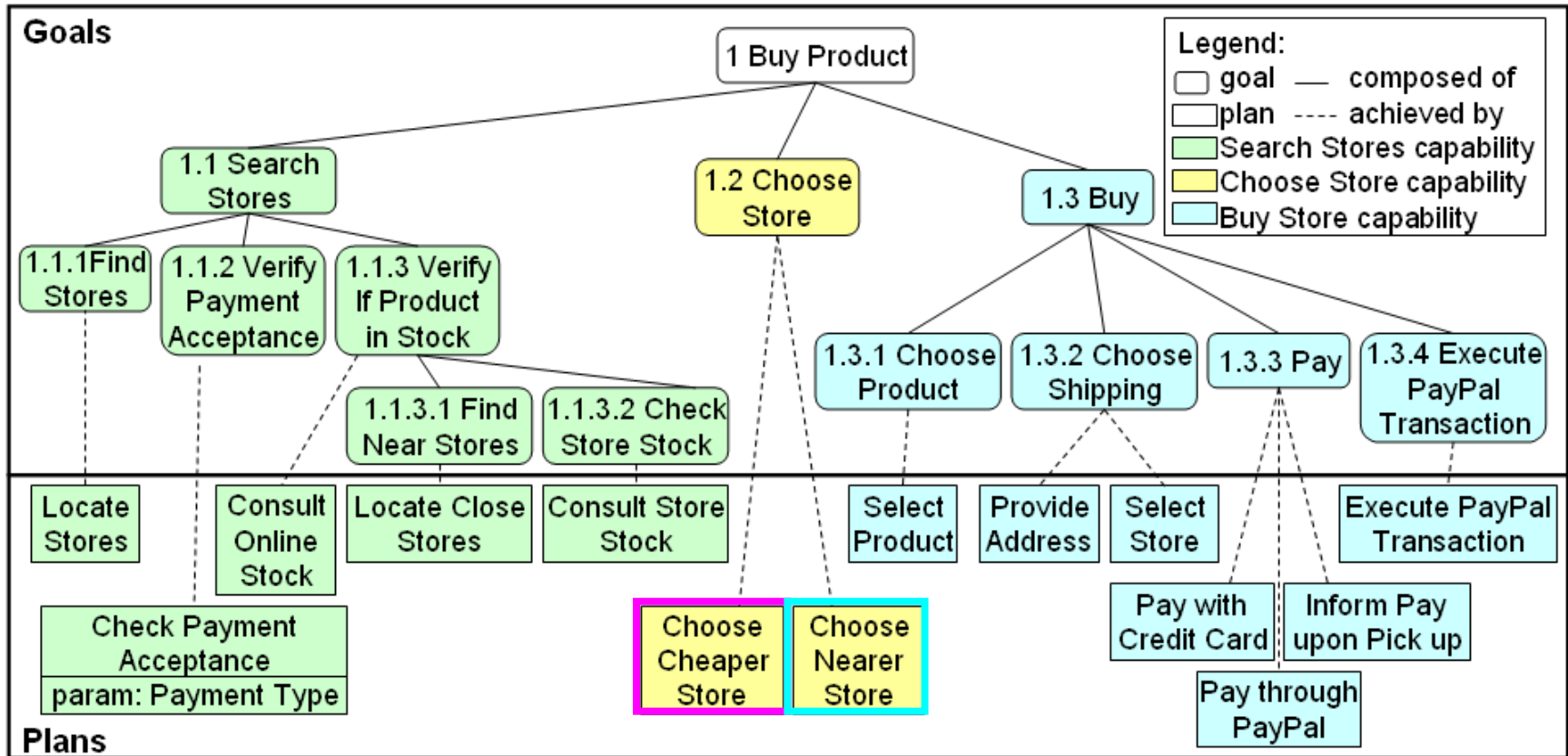


The Buyer Agent Family Case Study **LES**

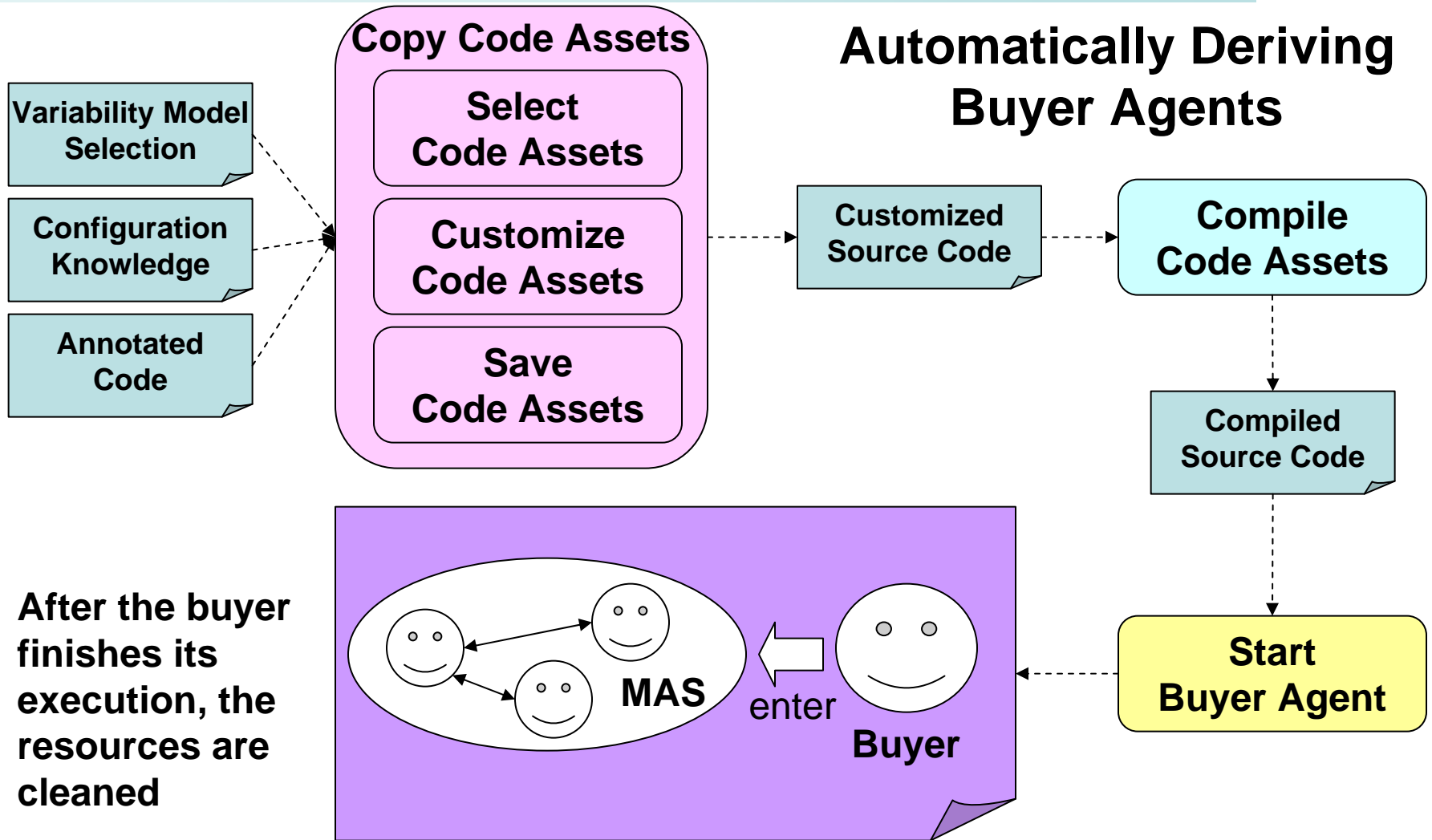


❖ Variability

- Store Selection Strategy



The Buyer Agent Family Case Study



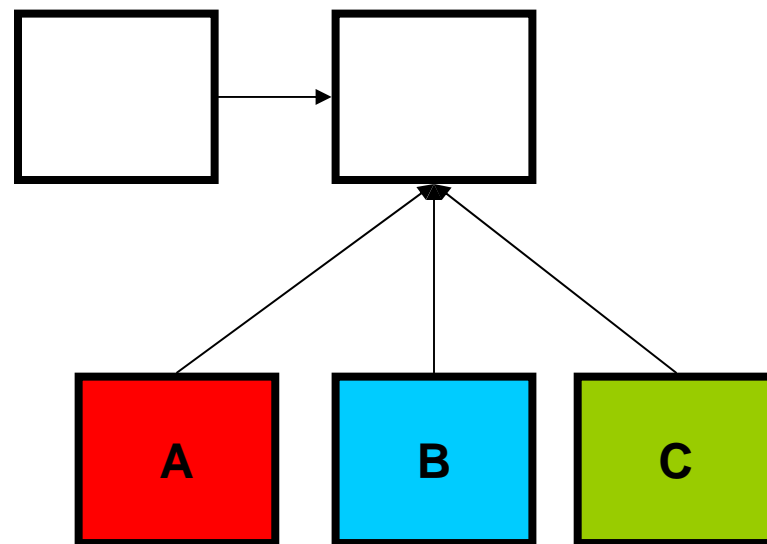
The Buyer Agent Family Case Study



Derivation of Specific Agents **X** Parameterization

- Control variables
- The need of modularization techniques

```
methodX
  if (A) {
  } else if (B) {
  } else if (C) {
  }
methodY
  if (A) {
  } else if (B) {
  } else if (C) {
  }
methodZ
  if (A) {
  } else if (B) {
  } else if (C) {
  }
```

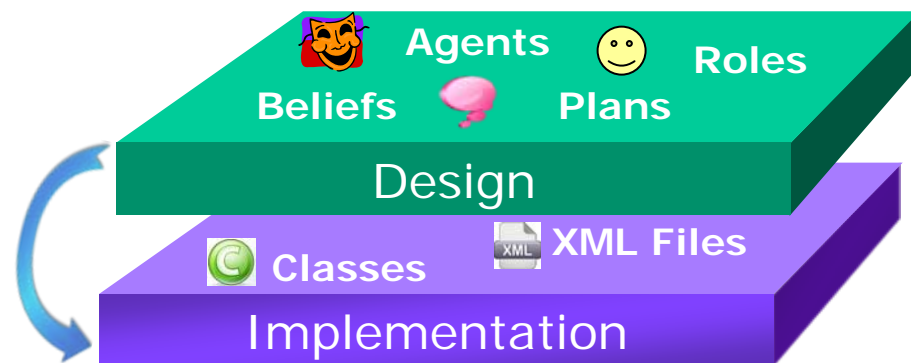


Modeling Variability

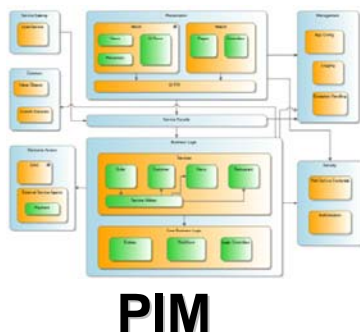


❖ MAS Approaches

- Design abstractions may not exist in the implementation platform



❖ MDA helps to bridge the gap between design and implementation



Jadex
BDI Agent System



Modeling Variability



❖ Goals Reuse

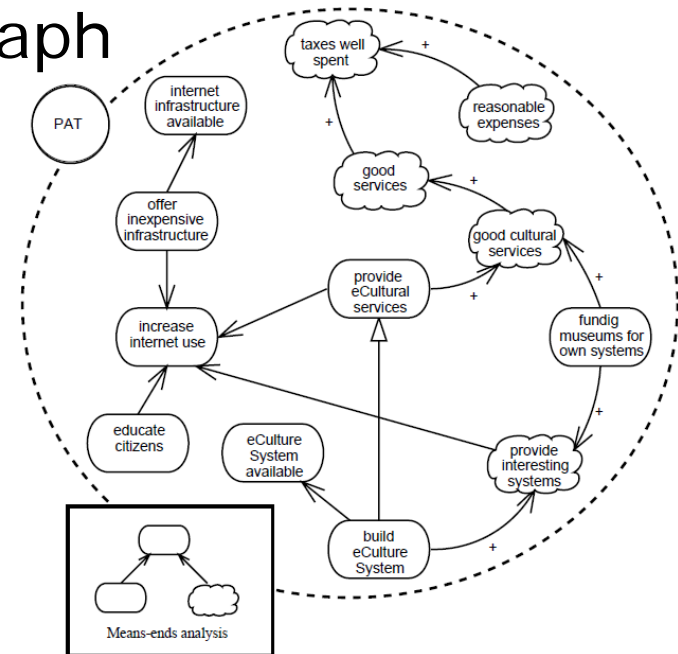
- Goal Diagram: Tree or Graph?
- Tropos models goals as a graph
 - Fine-grained elements
 - Complex design models

❖ Capabilities

- Modularization and reuse
- Design first-class elements?
- Alternative: two agents?

❖ Plan Parameterization

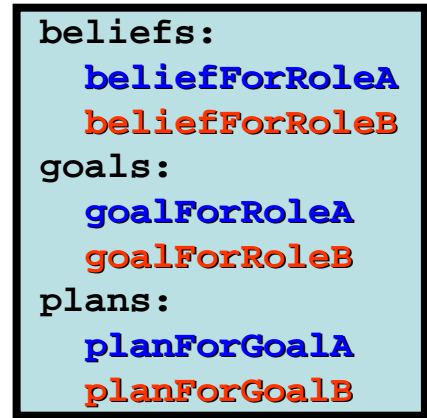
- Where are they in design models?



Should we strictly follow human models?

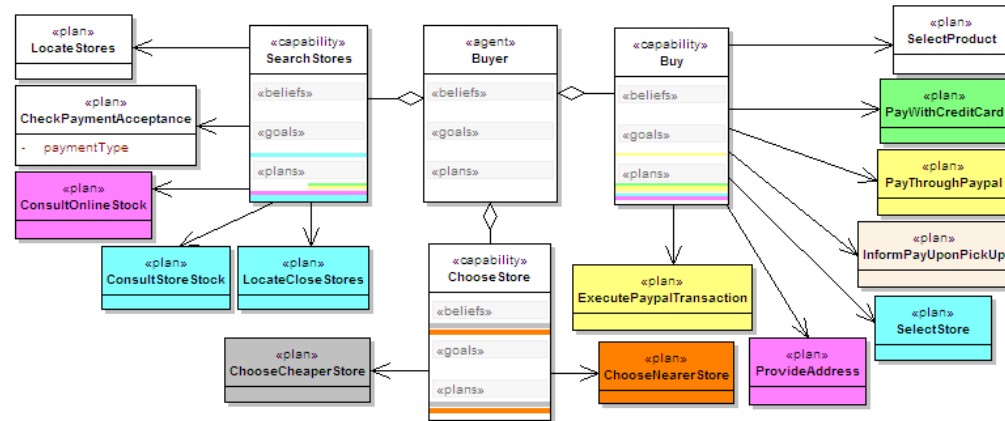
Variability Modularization

- ❖ Agents encapsulate both state and behavior
 - Lower coupling → inter-agent
- ❖ And low coupling and high cohesion between intra-agent components?
 - Capability concept

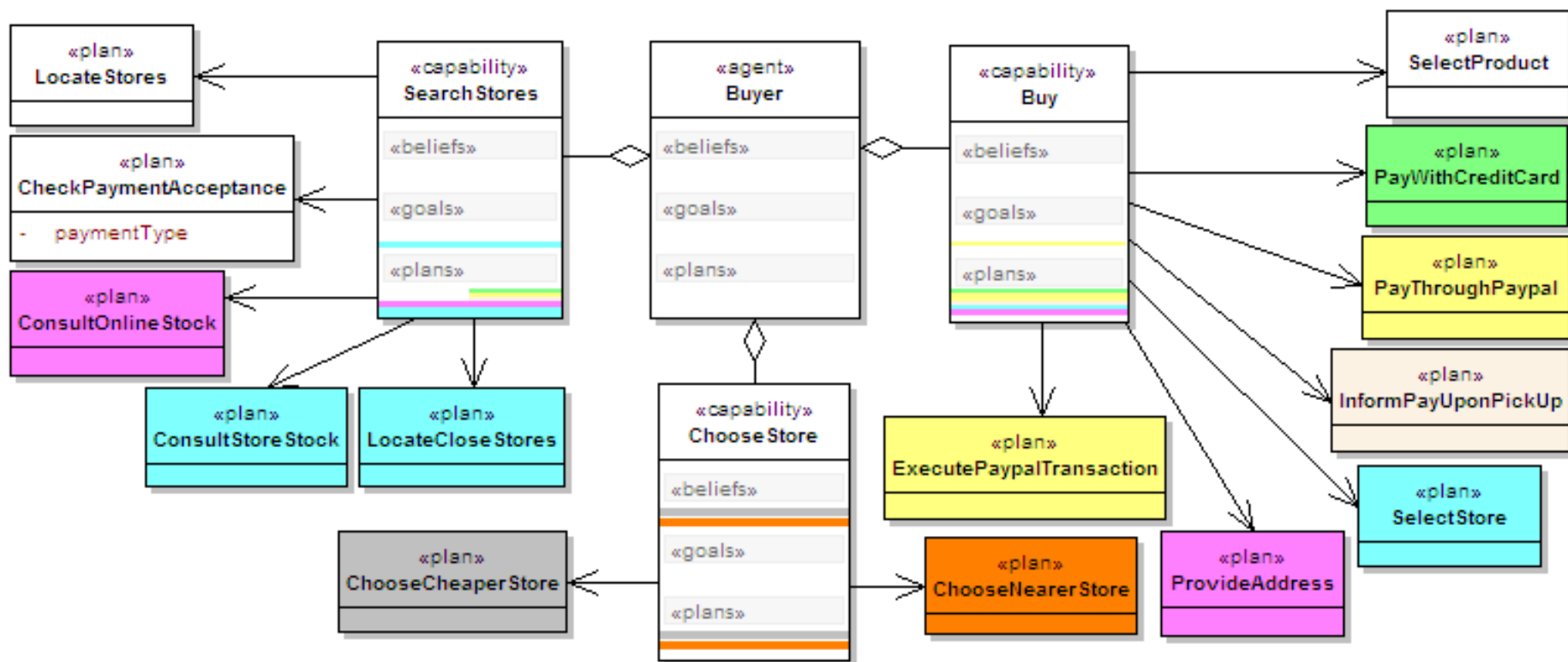


❖ Goal decomposition and Plan Modularization

- Variants modularized in single plans
- Additionally conditional compilation
 - Capabilities
 - increase number of components
 - AOP



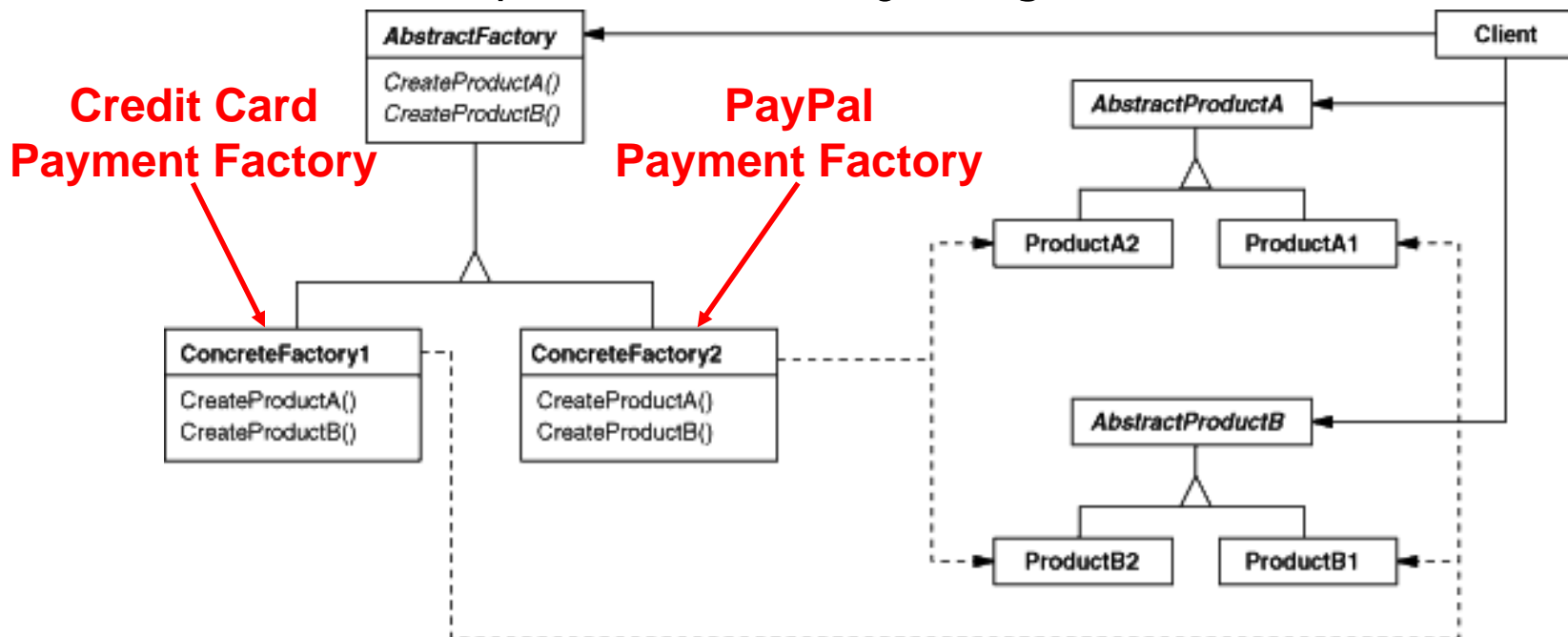
Modeling Variability



Variability Modularization



- ❖ SE approaches to improve software architectures
 - Reusability, maintainability, modularity, stability, dependability
 - E.g. Use of the *AbstractFactory* pattern to instantiate families of plans in the Buyer Agent SPL



Variability Modularization



AOSE could learn from research work that has been done in state-of-the-art SE to design and implement better software architectures.

- ❖ The need of empirical studies to evaluate approaches
 - Show the real effectiveness of proposed approaches
 - Several MAS methodologies and processes
 - How can we ensure that their proposed models indeed improves software development?

- ❖ Exploratory study of the development of a family of agents
 - Fine-grained variability
- ❖ Lessons learned
 - Variability types, design and implementation
- ❖ MASs aim at developing complex, distributed systems in terms of high level abstractions



However, these models will not likely be adopted in the industry if they do not promote reduced time-to-market, lower costs and higher quality.

❖ Fine-grained Variability in the Development of Families of Software Agents

- Ingrid Nunes
 - ionunes@inf.puc-rio.br
- Elder Cirilo
 - ecirilo@inf.puc-rio.br
- Donald Cowan
 - dcowan@cs.uwaterloo.ca
- Carlos J.P. de Lucena
 - lucena@inf.puc-rio.br

