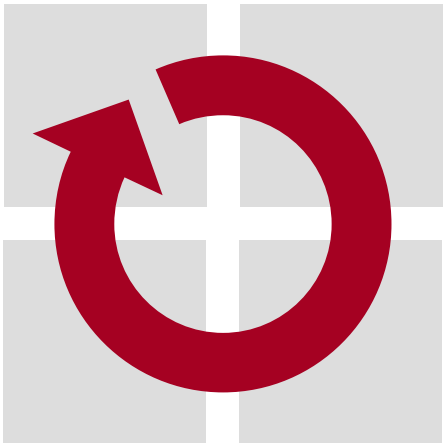
Architecture-Neutral Operating System Components

Daniel Lohmann

Department of Computer Science IV
Distributed Systems and Operating Systems
Friedrich-Alexander University Erlangen-Nuremberg

http://www4.informatik.uni-erlangen.de/~lohmann
daniel.lohmann@informatik.uni-erlangen.de



The Tyranny of OS Architectures

- Architecture defines fundamental properties and policies
  - Module structure and interaction schemes
  - Scheduling and locking strategies
  - Interrupt handling and synchronization
  - Supported hardware platforms
  - ...
- Architecture is usually considered as something static
  - Decisions are made at early stages of OS development
  - Extremely costly to adapt later

But why...

shouldn't it be possible to *configure* architectural properties?

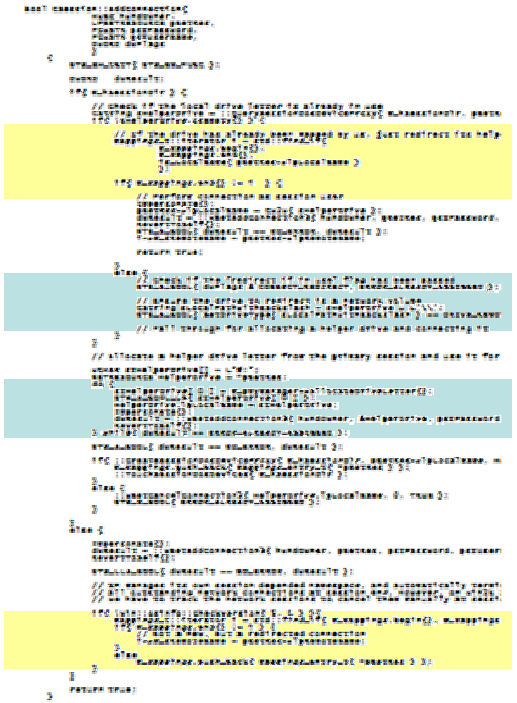
- Optimize a server OS for a dedicated services demands
  - Database / network / internet service
- Taylor down an embedded systems OS
  - Often very specific application demands and extreme resource constraints
- Cope better with architectural evolution

The Problem

Architectural properties are inherent *crosscutting*

Synchronisation

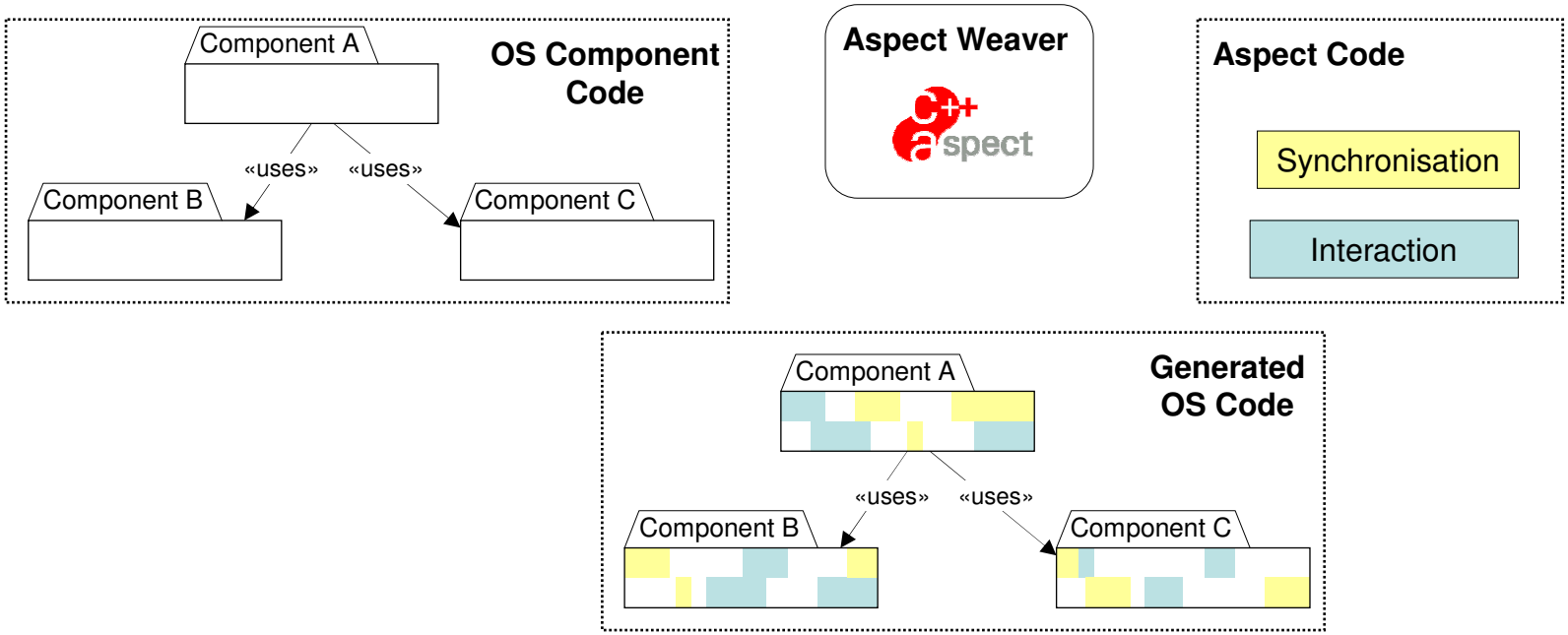
Interaction



Aspect-Oriented Programming

- AOP has proven to be successful in dealing with crosscutting concerns
- With *AspectC++* it is now possible to investigate the benefits for system software development

http://www.aspectc.org

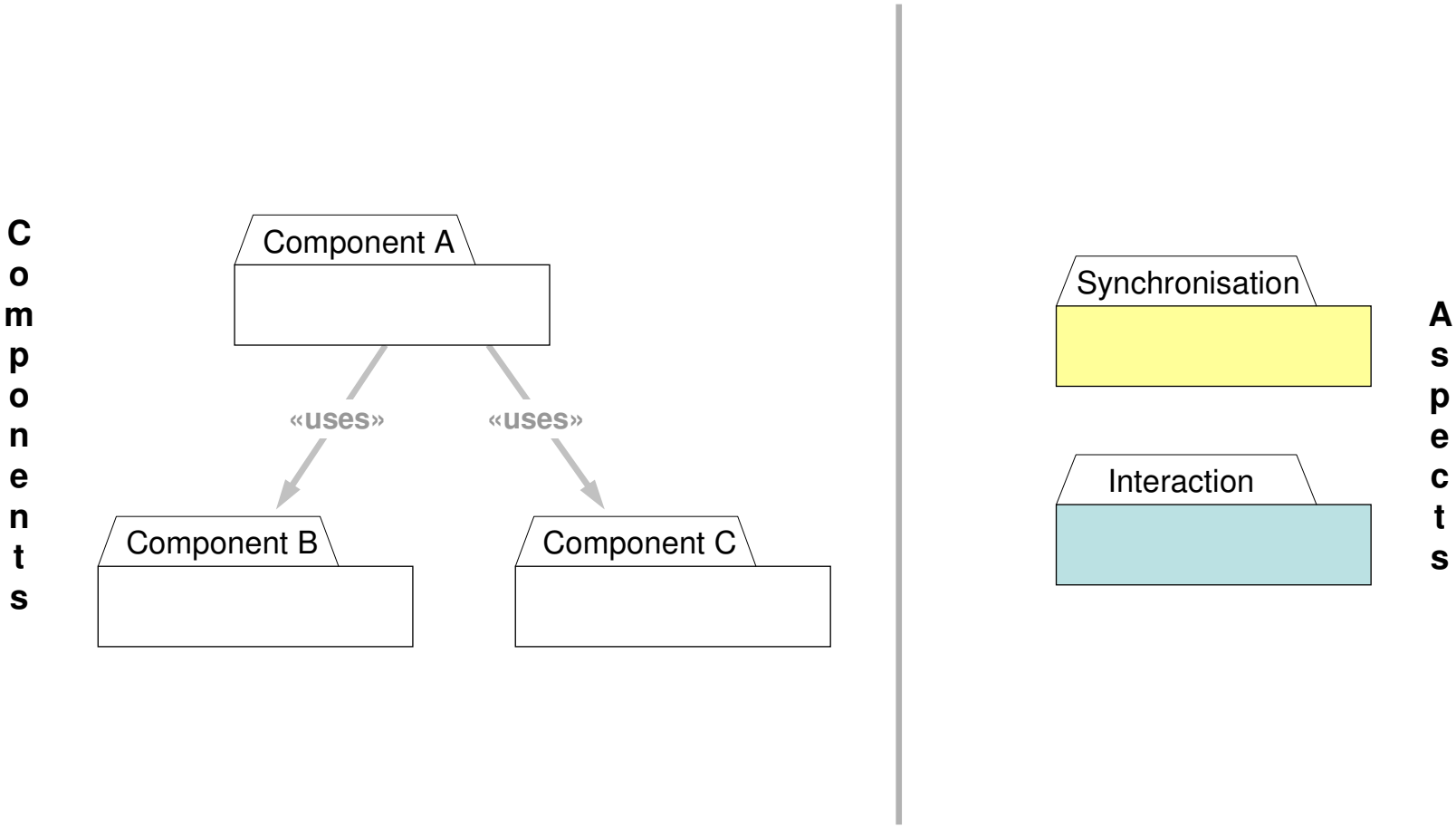


The CiAO Project (CiAO is Aspect Oriented)

- Aspect-oriented family of operating systems
  - Designed in an aspect-oriented manner
  - Full encapsulation of policies and architectural properties
- Target: Embedded Systems
  - From deeply-embedded devices up to embedded UNIX systems
  - High level of adaptability and configurability

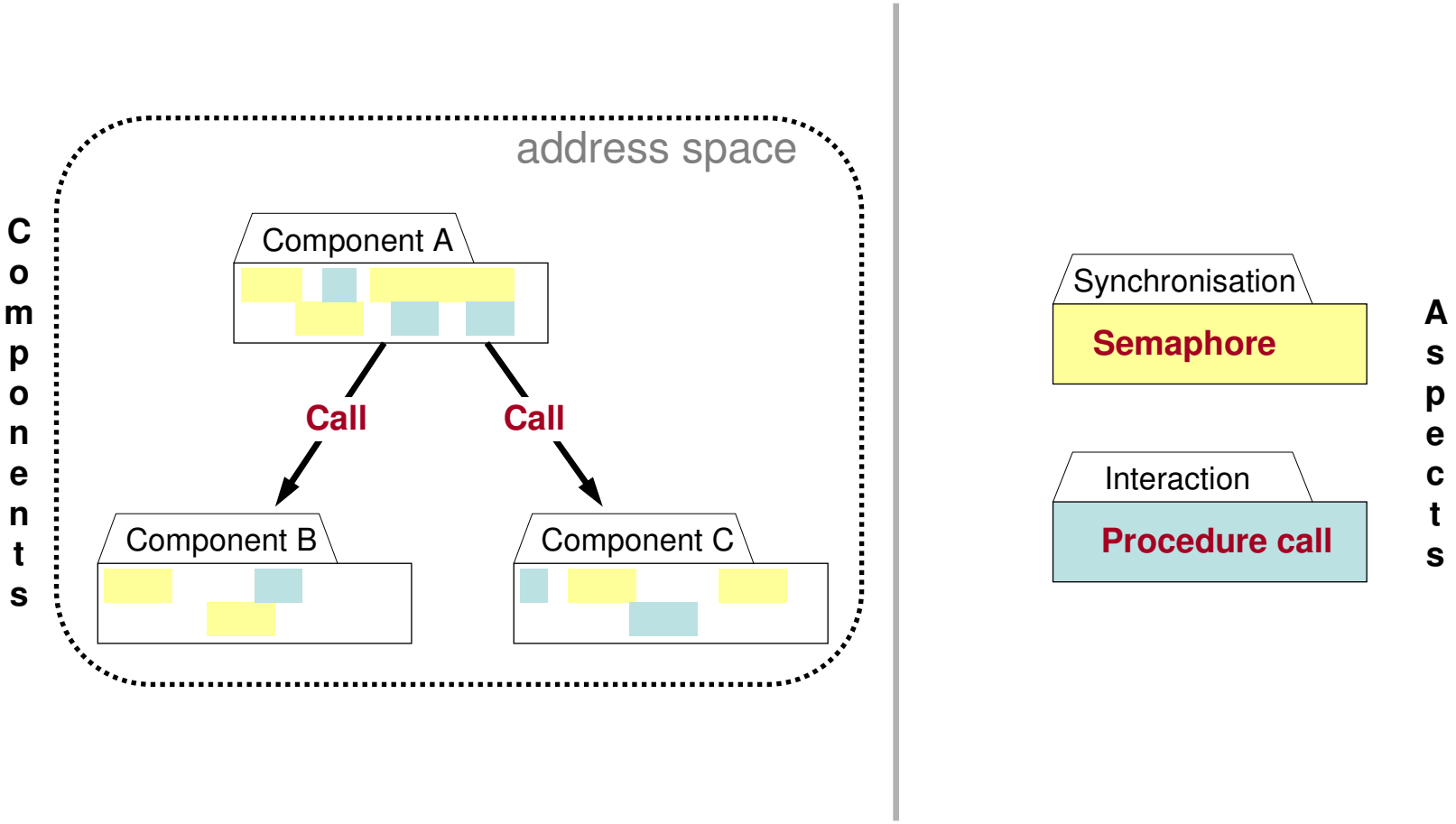
The CiAO Vision: Architectural Transparency

Abstract Model



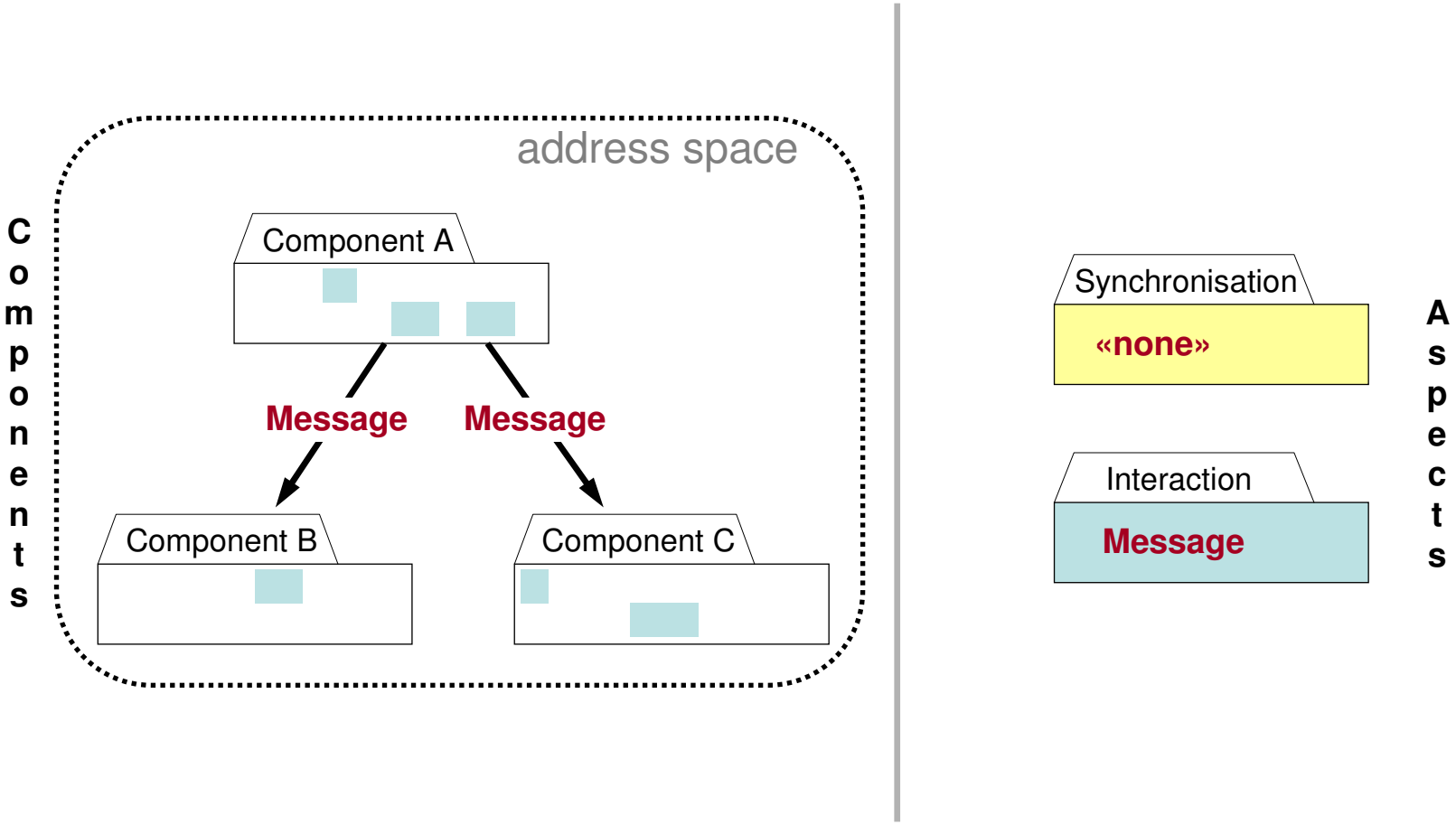
The CiAO Vision: Architectural Transparency

Configuration 1: Procedure based (monolithic)



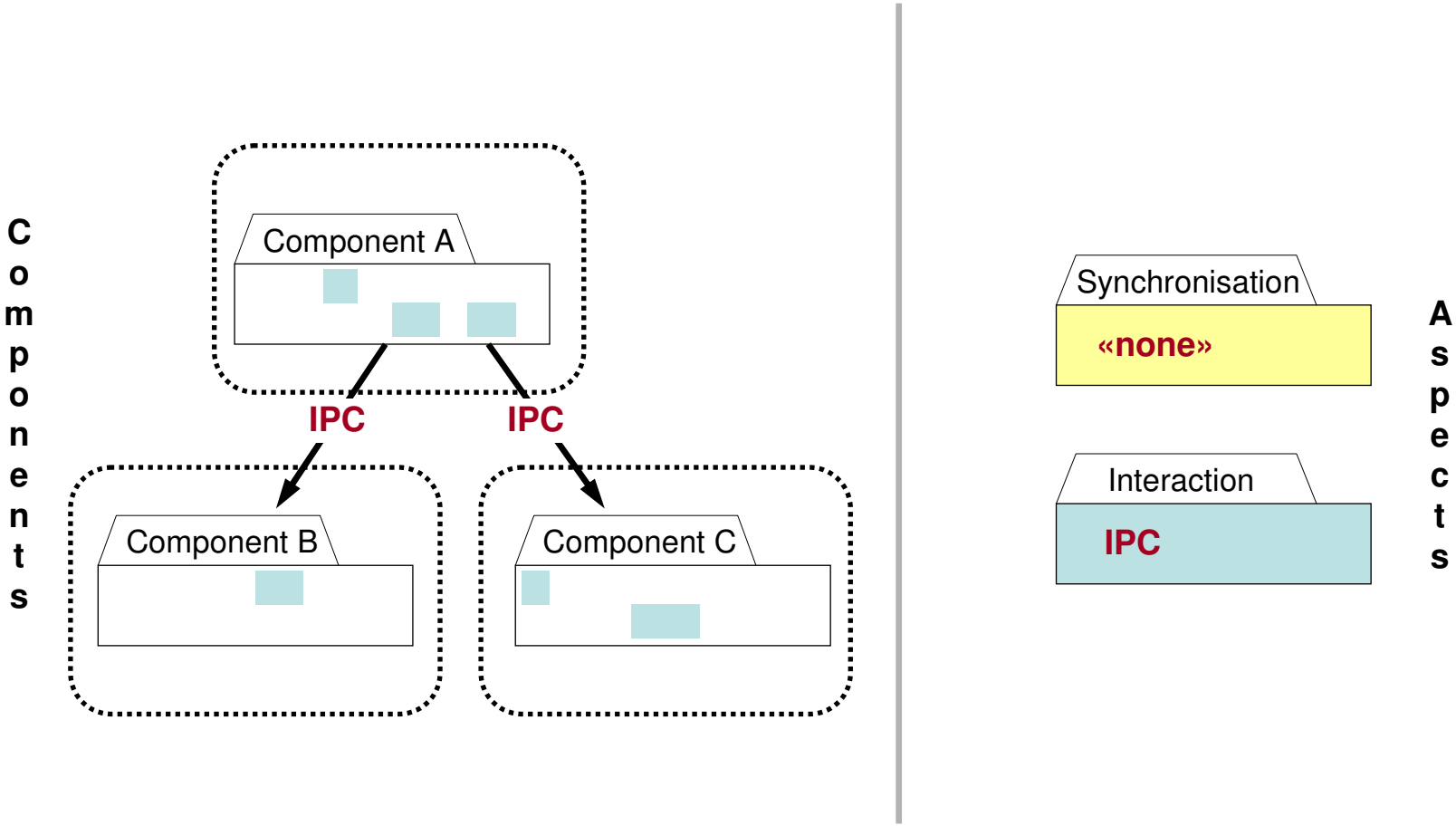
The CiAO Vision: Architectural Transparency

Configuration 2: Message oriented (threaded)



The CiAO Vision: Architectural Transparency

Configuration 3: Process oriented (µ-kernel)



Conclusion

- Architectural properties are inherent crosscutting
  - Low configurability
  - Hard to cope with architectural evolution
- Aspect-Orientd Software Development can help here
  - AspectC++ brings AOP to the C/C++ world
  - Evaluation of AOP concepts in system software development
  - Encapsulation without sacrificing efficiency
- Real chance to reach architectural transparency

Thanks for your attention!

Daniel Lohmann

Department of Computer Science IV
Distributed Systems and Operating Systems
Friedrich-Alexander University Erlangen-Nuremberg

http://www4.informatik.uni-erlangen.de/~lohmann
daniel.lohmann@informatik.uni-erlangen.de