

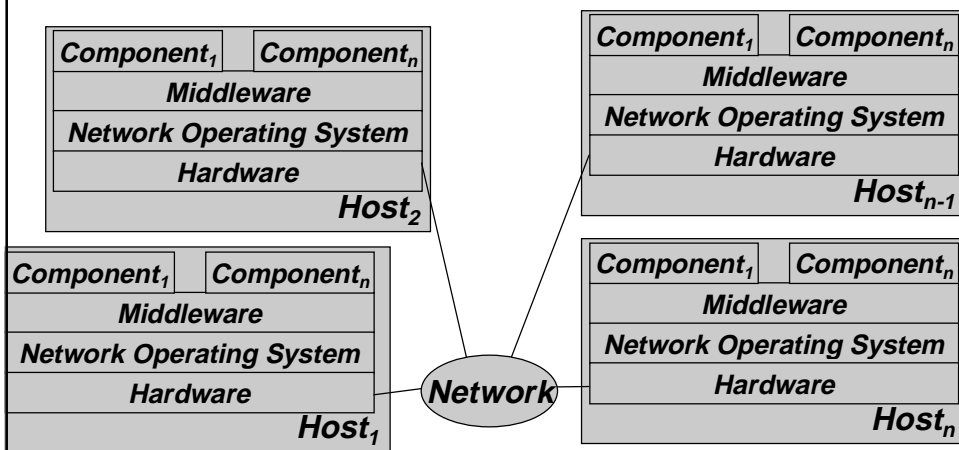


Software Engineering and Middleware: A Roadmap

Wolfgang Emmerich
Dept. of Computer Science
University College London
Gower Street, London WC1E 6BT, UK
<http://www.cs.ucl.ac.uk/staff/w.emmerich>



What is Middleware?





Requirements for Middleware

- **Help implementing non-functional requirements (“ilities”)**
 - **Network communication**
 - **Coordination of distributed components**
 - **Reliable execution of components**
 - **Scalability to future loads**
 - **Securing components against intruders**
 - **Resolving heterogeneity of**
 - **hardware**
 - **operating systems**
 - **network protocols**
 - **component implementations**
 - **middleware**

© Wolfgang Emmerich, 2000

3

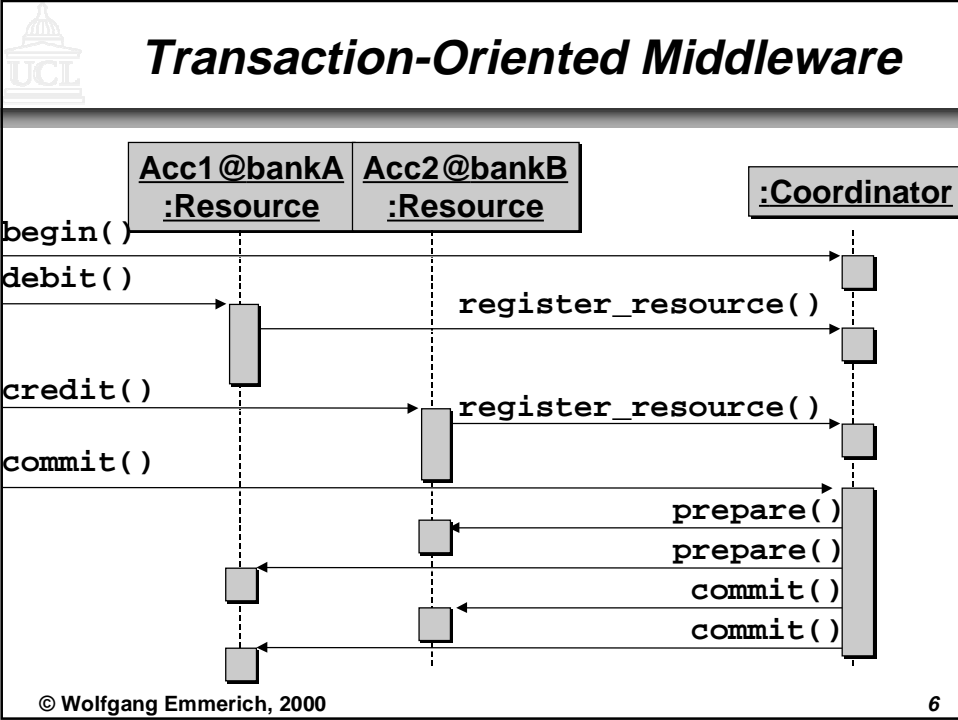
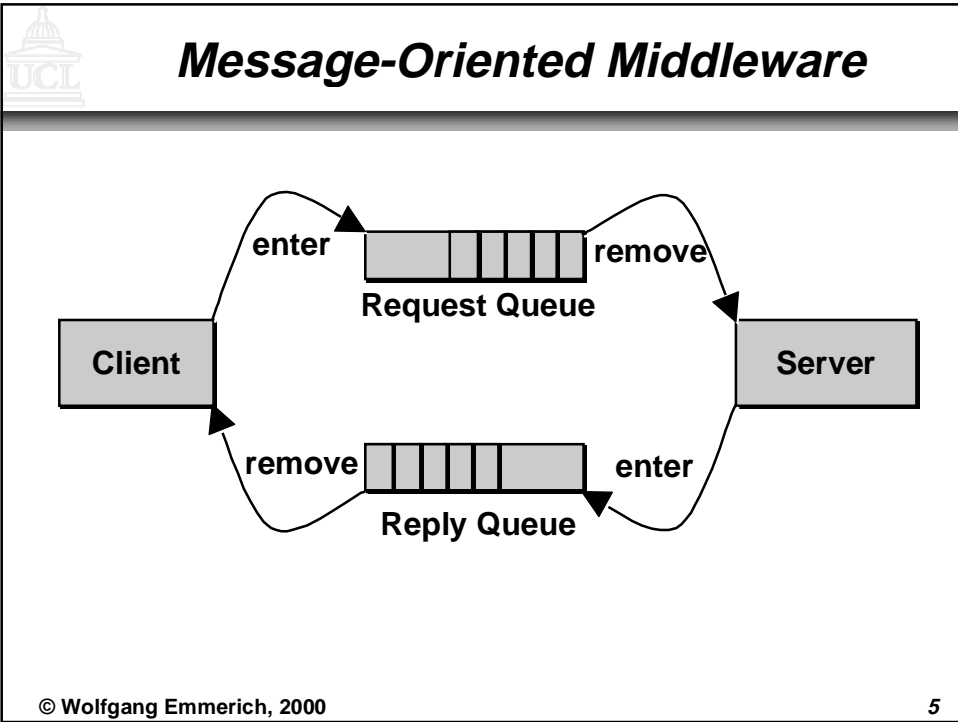


Middleware Solutions

- | | |
|--|--|
| ■ Transaction-Oriented <ul style="list-style-type: none">• IBM CICS• BEA Tuxedo• Encina | ■ Object-Oriented <ul style="list-style-type: none">• OMG/CORBA• COM• Java/RMI |
| ■ Message-Oriented <ul style="list-style-type: none">• IBM MQSeries• DEC Message Queue• NCR TopEnd | ■ Component-Oriented <ul style="list-style-type: none">• Enterprise Java Beans• COM+• CORBA Components |
| ■ RPC Systems <ul style="list-style-type: none">• ANSA• Sun ONC• OSF/DCE | |

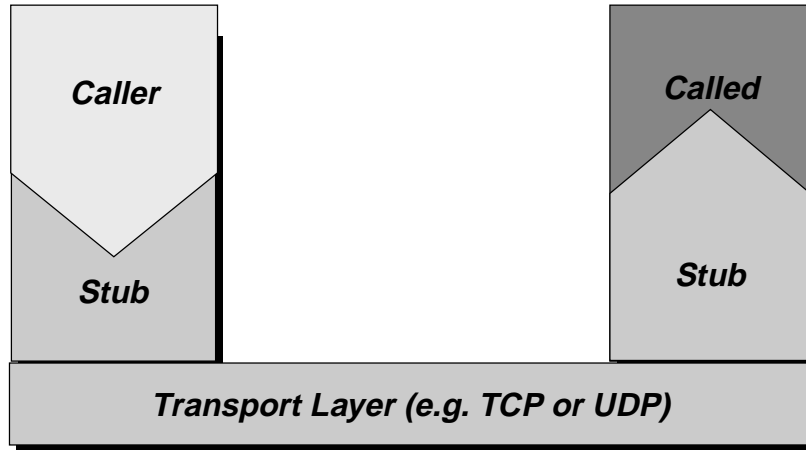
© Wolfgang Emmerich, 2000

4





Procedural Middleware

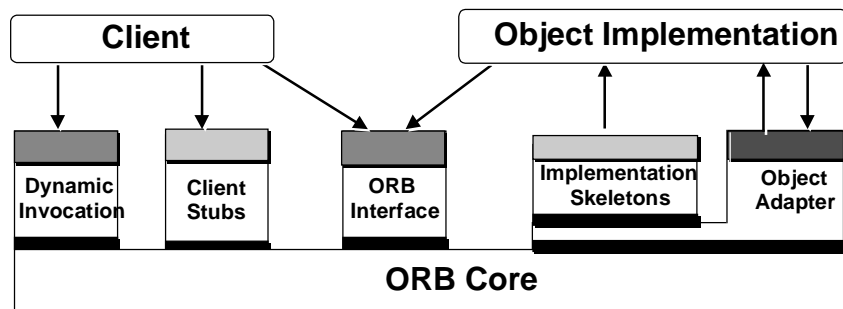


© Wolfgang Emmerich, 2000

7



Object-Oriented Middleware

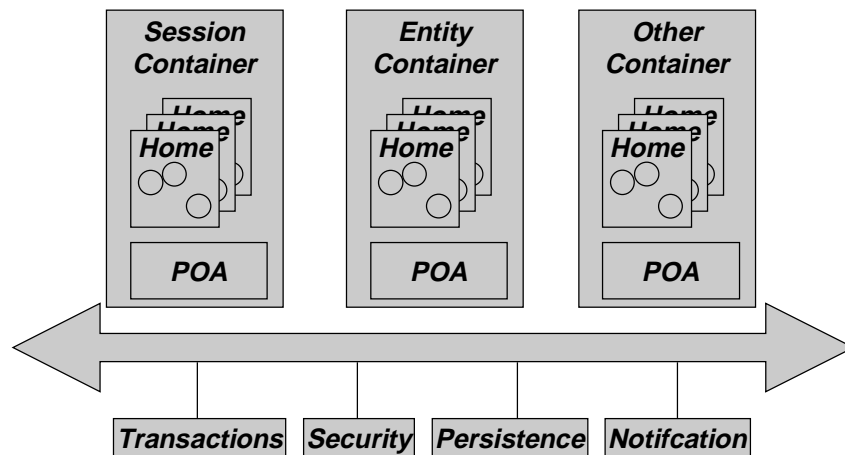


© Wolfgang Emmerich, 2000

8



Component-Oriented Middleware



© Wolfgang Emmerich, 2000

9



Middleware Research

- **Recent trends in ICDCS, Middleware and EDOC Conferences include**
 - **Component location by trading (Raymond)**
 - **Reflexive Middleware (Blair)**
 - **Non-transparent replication (Tanenbaum)**
 - **Middleware for real-time systems (Schmid)**
 - **Middleware for mobile and pervasive computing (Sun, Picco)**
- **Will influence middleware solutions 3-5 years down the road**

© Wolfgang Emmerich, 2000

10



Trends

- ***Middleware incredibly successful in industry because they provide programming and run-time environment for building reliable, secure, scalable, open and distributed software systems***
- ***Middleware industry has proven track record for efficient transfer of research results into products (e.g. ODP Trading)***
- ***If software engineering research ignores this our results will be ignored!***

© Wolfgang Emmerich, 2000

11



Basic Research Question

- ***Middleware provides the programming support, but what about analysis, architecture, design and testing of middleware-based distributed systems?***
- ***How do we build on middleware to systematically engineer software systems?***

© Wolfgang Emmerich, 2000

12



Requirements Engineering

- **Existing requirements engineering techniques have a strong focus on functional requirements**
 - *Use case based approaches*
 - *Formal methods*
- **Middleware selection needs to be driven by non-functional requirements**
- **How do we specify (and quantify!) non-functional requirements?**
- **How do we elicit dimensions in which they may change?**

© Wolfgang Emmerich, 2000

13



Software Architecture

- **ADLs will be superseded by the IDLs, CDLs and CIDLs that are provided by middleware**
- **Middleware provides connectors**
- **How do we select an appropriate middleware as the basis of a distributed software architecture?**
- **Difficulties:**
 - *Complexity of middleware solutions*
 - *Rapid development of the industry*

© Wolfgang Emmerich, 2000

14



Software Architecture

- **Research challenge: Specify / reason on non-functional characteristics of SW architecture**
- **Speculative Examples:**
 - **Performance and Scalability: Queing Theory and Markov Models?**
 - **Security: Deontic Logic?**
 - **Coordination: Process algebra?**



Design

- **Reflection makes various aspects of distribution visible to distributed systems engineers**
- **We need design notations, methods and tools that can express concerns such as**
 - **application-level transport protocols**
 - **replication strategies**
 - **scheduling policies**
 - **location-awareness**



Conclusions

- *Middleware solutions assist in building complex distributed systems right now*
- *Middleware research results will extend that support in the future*
- *Potential synergies between software engineering middleware research: SE community needs to build on middleware and support requirements, architecture, design and test stages of the distributed systems life cycle*

© Wolfgang Emmerich, 2000

17



Further Information

- *<http://www.distributed-objects.com>*
- *<http://www.omg.org>*
- *<http://www.msn.com>*
- *<http://www.cetus-links.org>*

© Wolfgang Emmerich, 2000

18