

Trường Đại học Khoa Học Tự Nhiên  
Khoa Công Nghệ Thông Tin  
Bộ môn Công Nghệ Phần Mềm

# CTT526 - Kiến trúc phần mềm

## Tài liệu

## kiến trúc phần mềm

PGS.TS. Trần Minh Triết  
[tmtriet@fit.hcmus.edu.vn](mailto:tmtriet@fit.hcmus.edu.vn)



KHOA CÔNG NGHỆ THÔNG TIN  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

- Nội dung của bài giảng sử dụng:

Session 6:

Documenting a Software Architecture

trong bộ slide [Software Architecture Essential](#)  
của GS. Ian Gorton

Software Engineering Institute  
Carnegie Mellon University

# Architecture Documentation

- ☐ Architecture documentation is a thorny issue
- ☐ Commonly there is no documentation covering the architecture.
  - ☐ If it is, it's out-of-date, inappropriate and basically not very useful.
- ☐ Also projects that have masses of architecture related information
  - ☐ Sometimes invaluable, but often it's out-of-date, inappropriate and not very useful!

# Documenting an Architecture is good!

- ☐ Others can understand/evaluate the design.
- ☐ We can understand the design after a period of time.
- ☐ Others in the project team and development organization can learn from the architecture.
- ☐ We can do analysis on the design, perhaps to assess its likely performance, or to generate standard metrics.

# But it's difficult ...

- No universally accepted architecture documentation standard.
- An architecture can be complex, and documenting it in a comprehensible manner is time consuming and non-trivial.
- An architecture has many possible views. Documenting all the potentially useful ones is time consuming and expensive.
- An architecture design often evolves. Keeping the architecture documents current is often forgotten, especially with time and schedule pressures in a project.

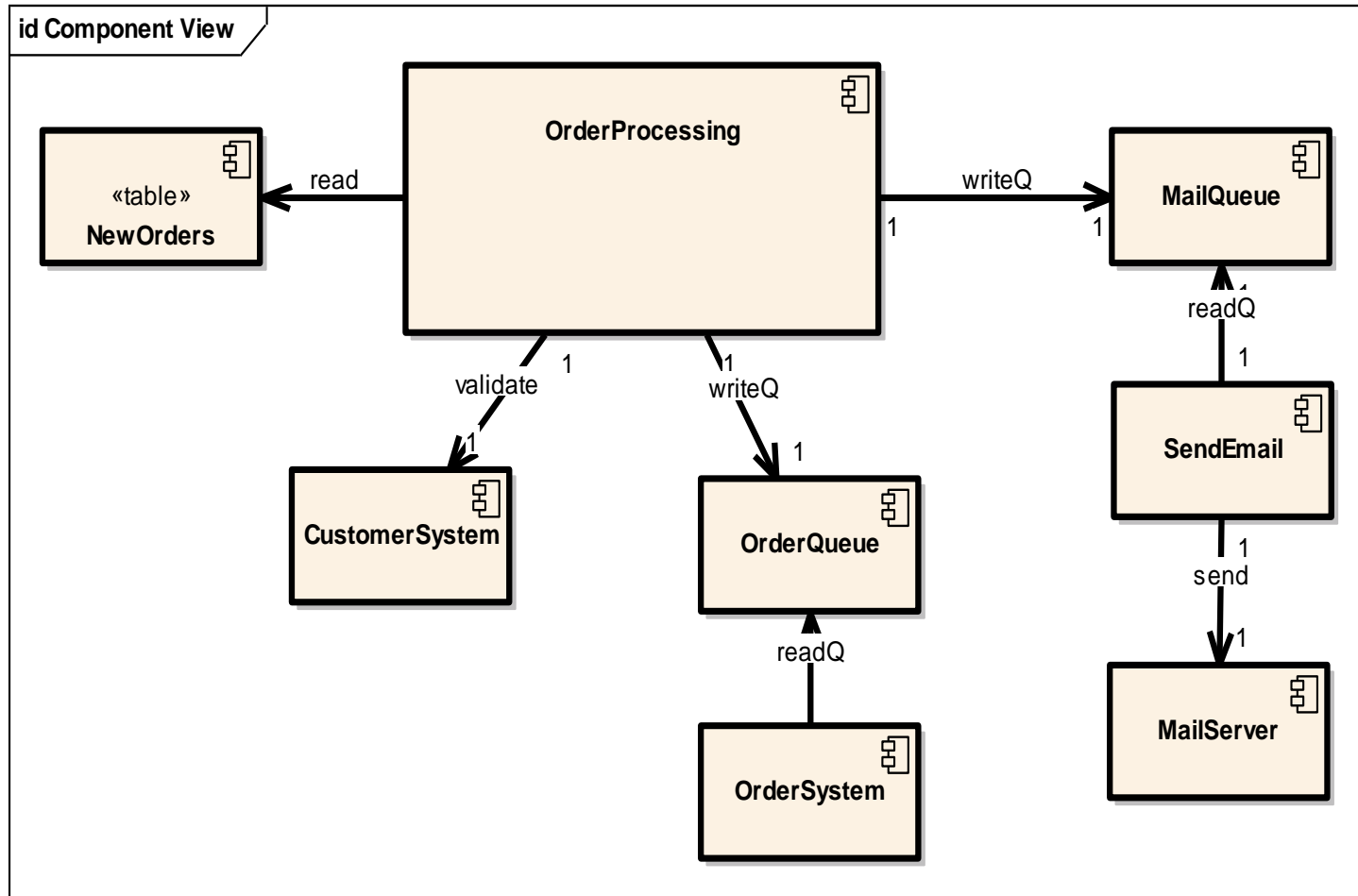
# Think carefully about what to document

- ☐ Project complexity
  - ☐ A small project may only need a 'marketecture'
- ☐ Project longevity
  - ☐ One-off stop gap software?
  - ☐ Strategic, long-term, will evolve?
- ☐ Needs of stakeholders
  - ☐ Small team, a whiteboard might be ok
  - ☐ Large, dislocated team needs more
  - ☐ Integrators? Testers? Programmers?
- ☐ Need to spend documentation dollars/euros wisely on high value products

# UML 2.0

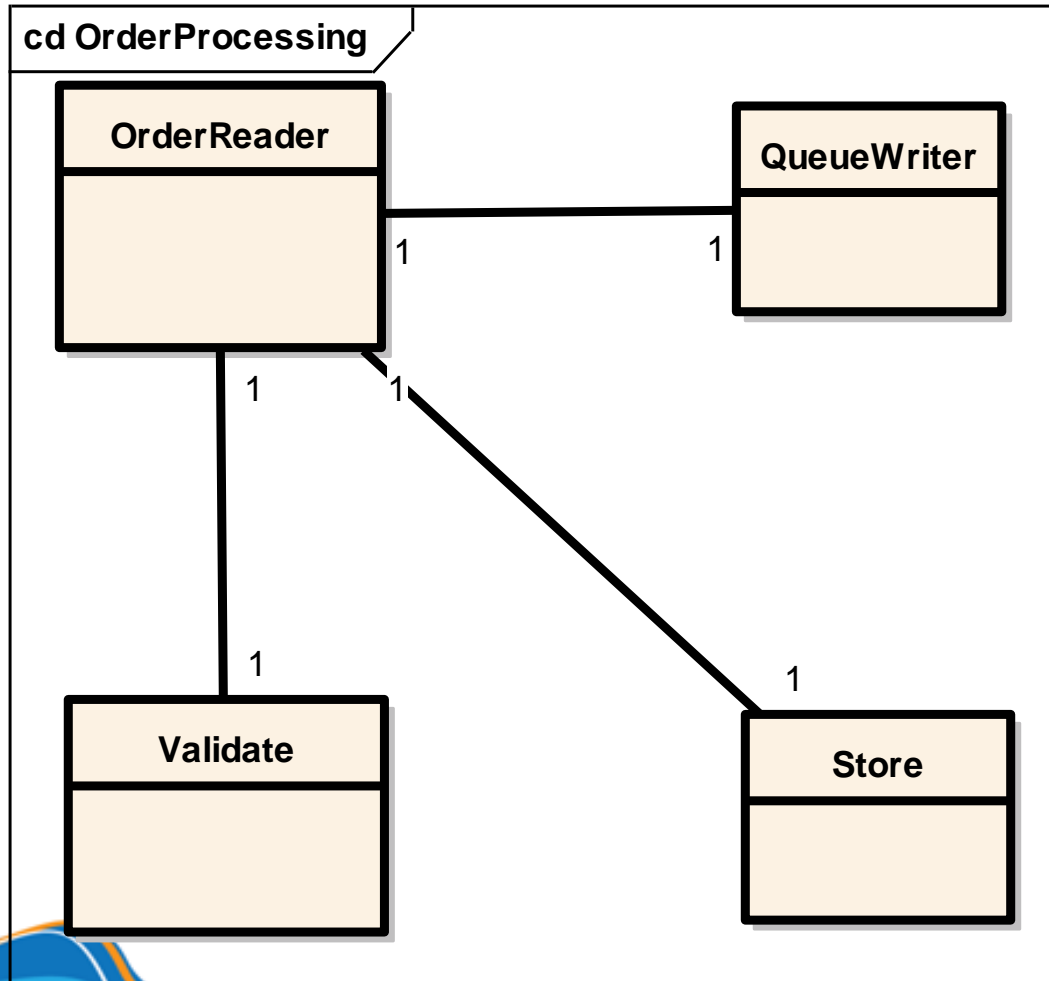
- ☐ UML is a powerful way to document an architecture
- ☐ Provides a relatively formal, unambiguous description
- ☐ New features in UML 2.0 appropriate for architectures
- ☐ Good tools available, some free
- ☐ Can be used to depict various structural/behavioral architecture views

# Component Diagram

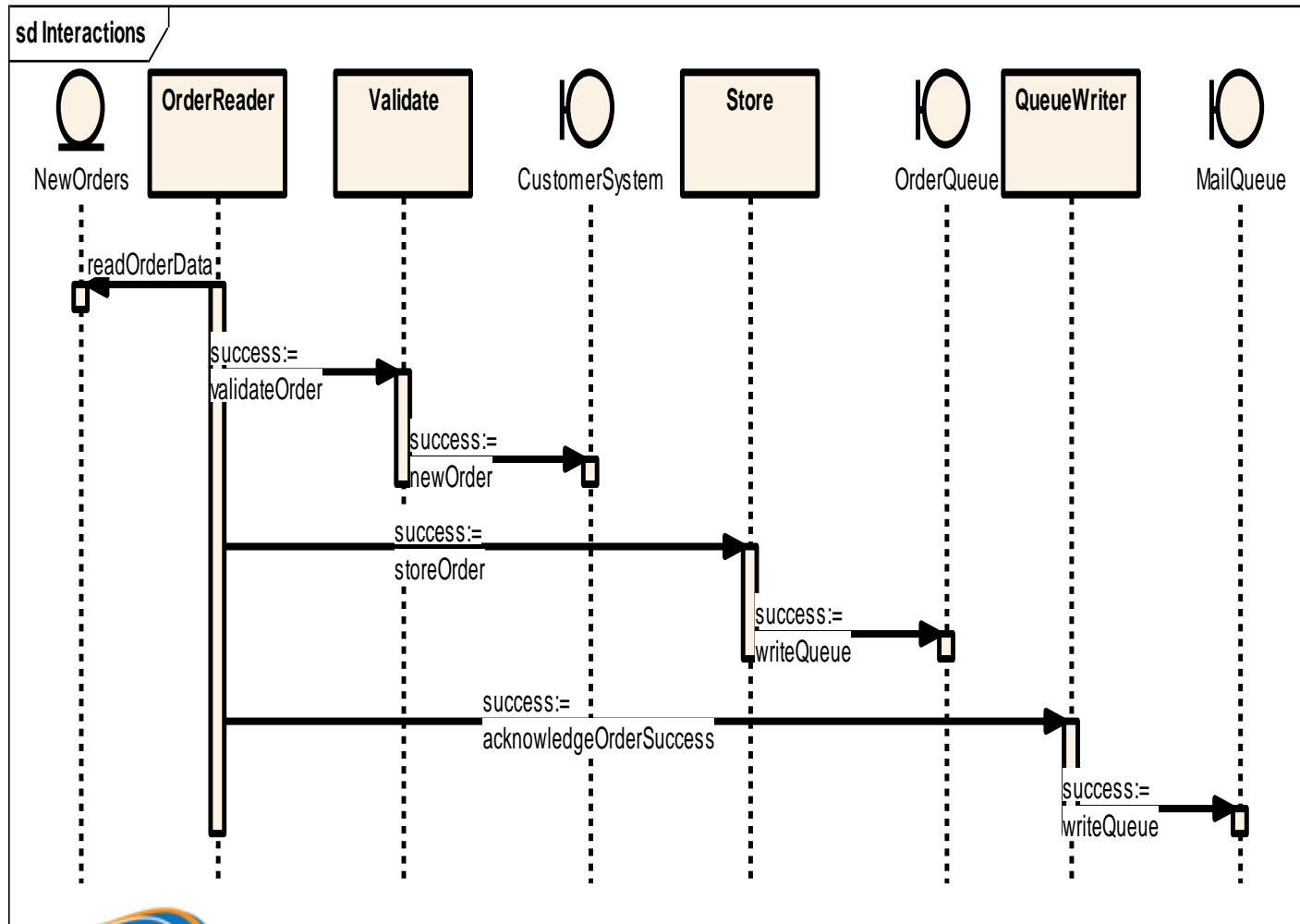




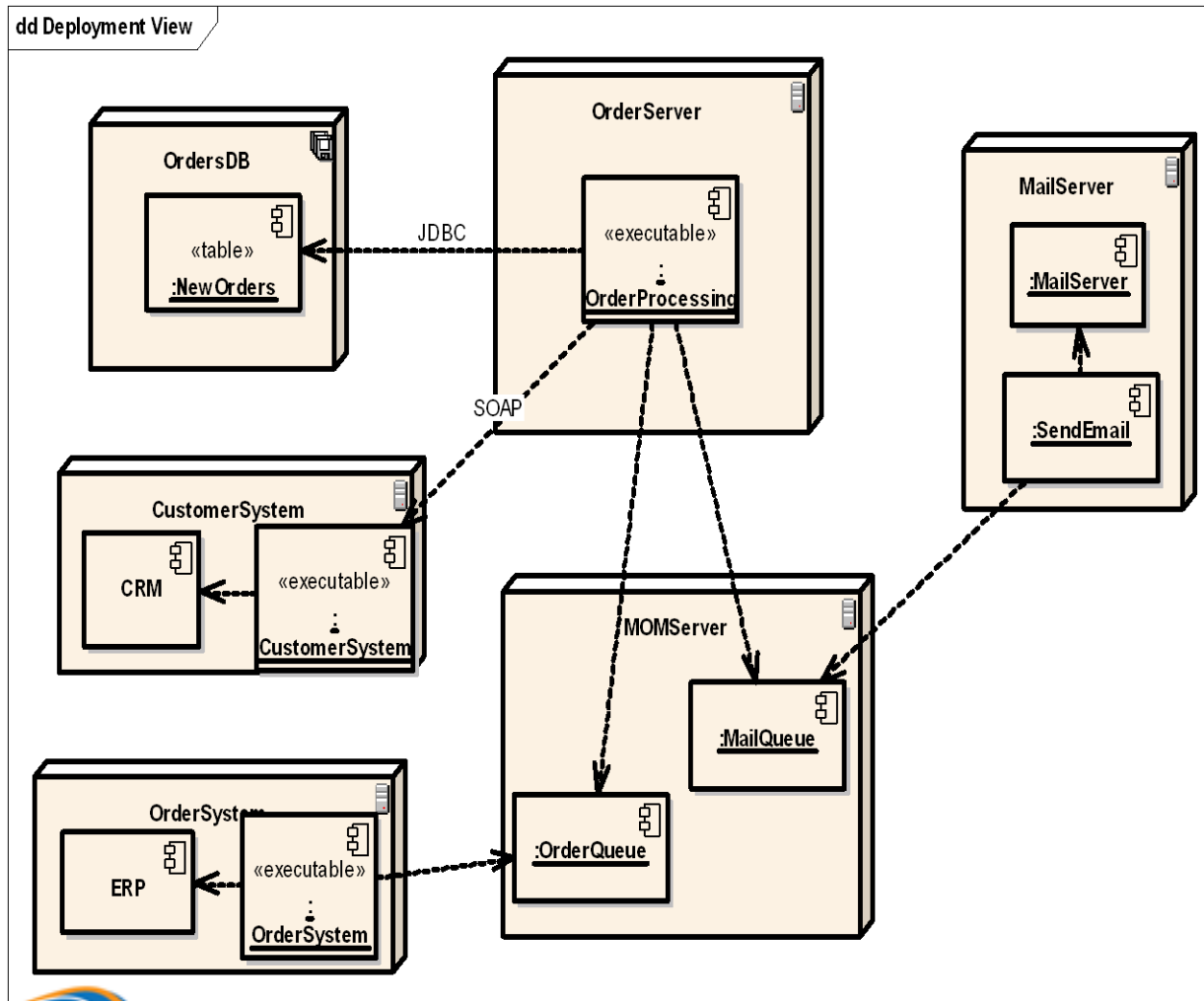
# Class Diagram



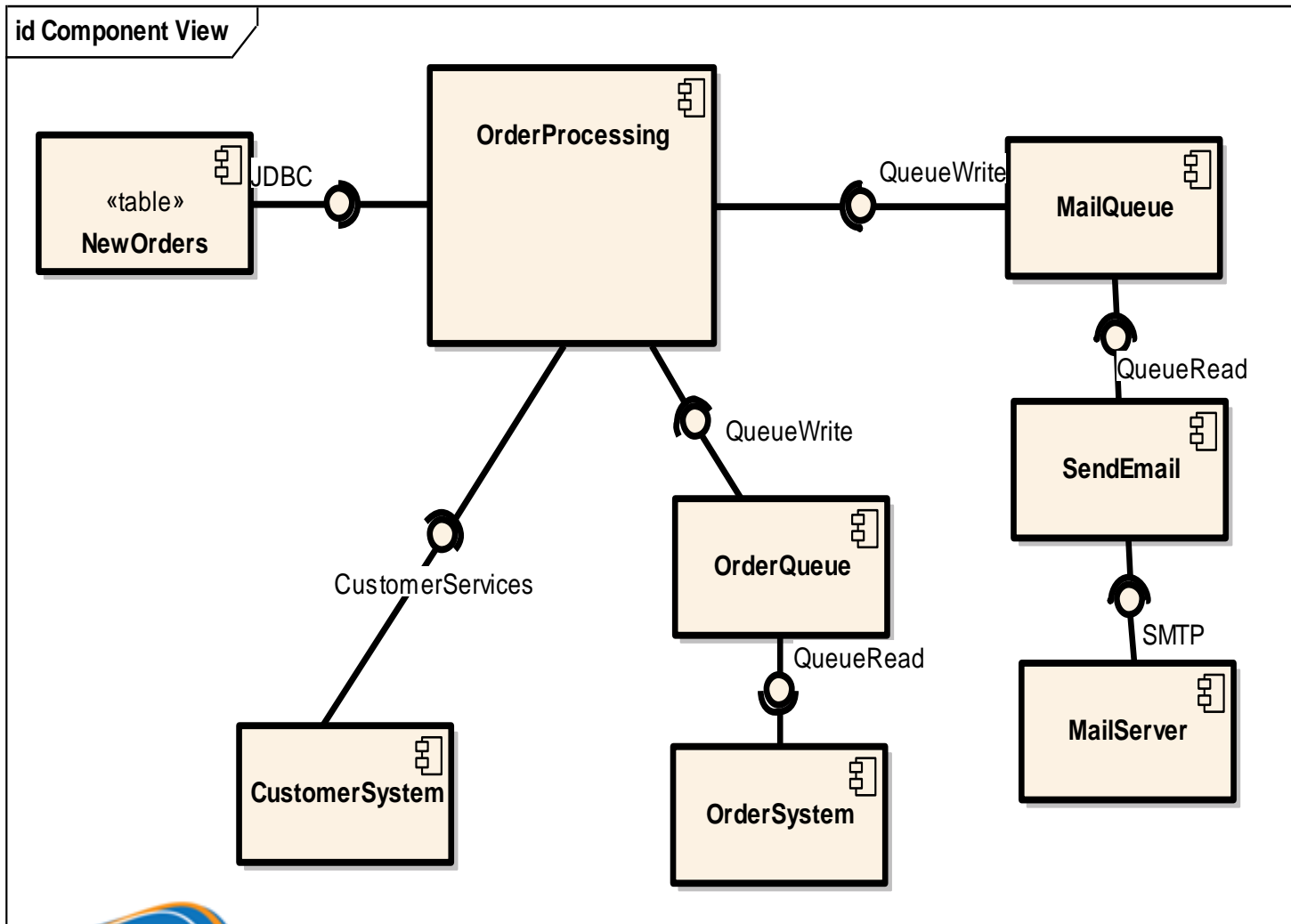
# Sequence Diagram



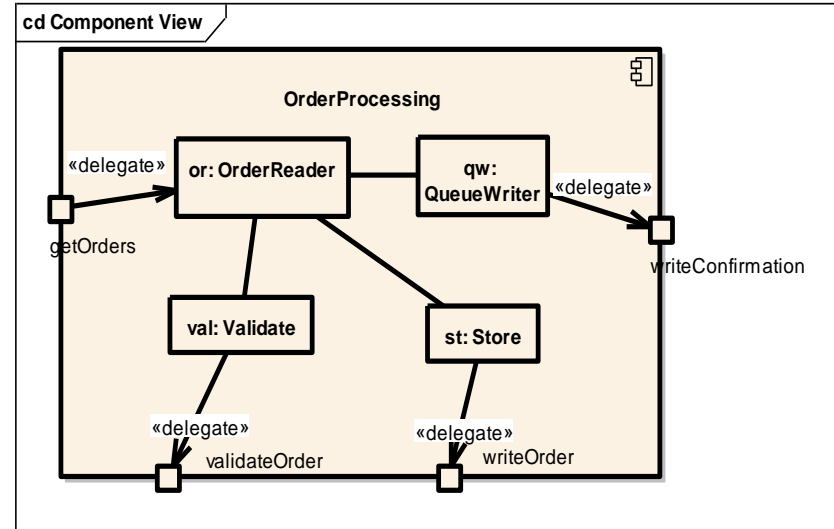
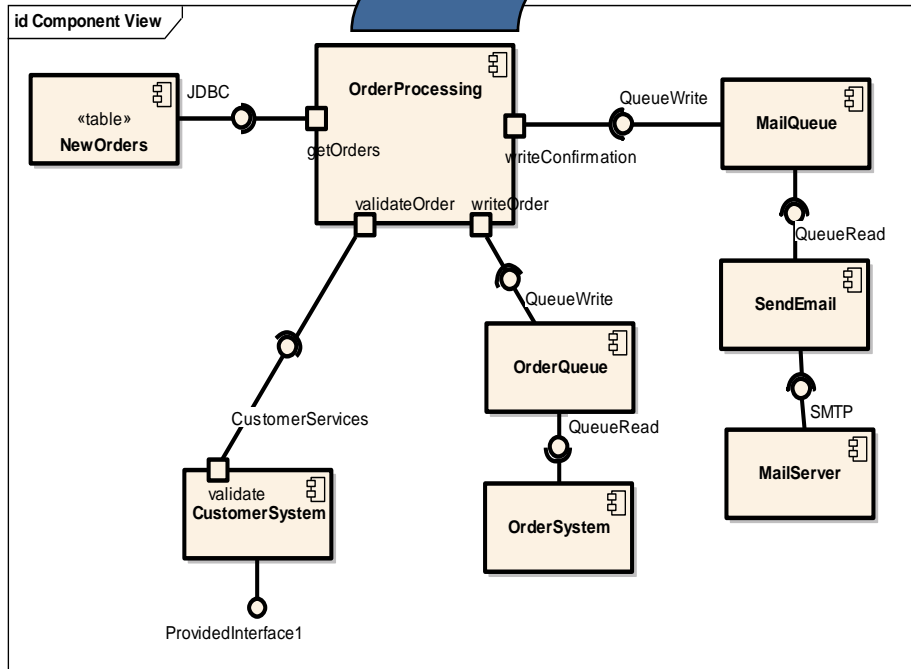
# Deployment Diagram



# Component Interfaces



# Component Decomposition



# Document Template

- Documentation is easier if there's a template to use
  - ▣ Reduces start-up time for projects by providing ready-made document structures
  - ▣ familiarity gained with the document structure aids in the efficient capture of project design details.
  - ▣ help with the training of new staff

# Template Headings

## Architecture Documentation Template

Project Name: XXX

- 1 Project Context
- 2 Architecture Requirements
  - 2.1 Overview of Key Objectives
  - 2.2 Architecture Use Cases
  - 2.3 Stakeholder Architectural Requirements
  - 2.4 Constraints
  - 2.5 Non-functional Requirements
  - 2.6 Risks
- 3 Solution
  - 3.1 Relevant Architectural Patterns
  - 3.2 Architecture Overview
  - 3.3 Structural Views
  - 3.4 Behavioral Views
  - 3.5 Implementation Issues
- 4 Architecture Analysis
  - 4.1 Scenario analysis
  - 4.2 Risks

# Summary

- ☐ Some documentation is nearly always a good idea
- ☐ Trick is to produce 'just enough' and no more
  - ☐ requires upfront planning and thinking
  - ☐ Commitment to keeps docs current
- ☐ UML 2.0 makes architecture documentation easier
- ☐ Some good UML 2.0 tools, try 'em out.