

Component-based Software Development

Lecture No. 1

Introduction to the Course

Outline of the Lecture

- Purpose of the course
 - Introductions and fundamentals of CBD
 - Design for CBD & Practical experience
- Motivation for CBD
- Prerequisites for the participants
 - Software engineering, OO design and development
 - Internet technologies (WWW, HTML, Java servlets)
- What will the participants learn
 - CBD state-of-the-art
 - Development of 3rd party software / COTS
- Practical arrangements

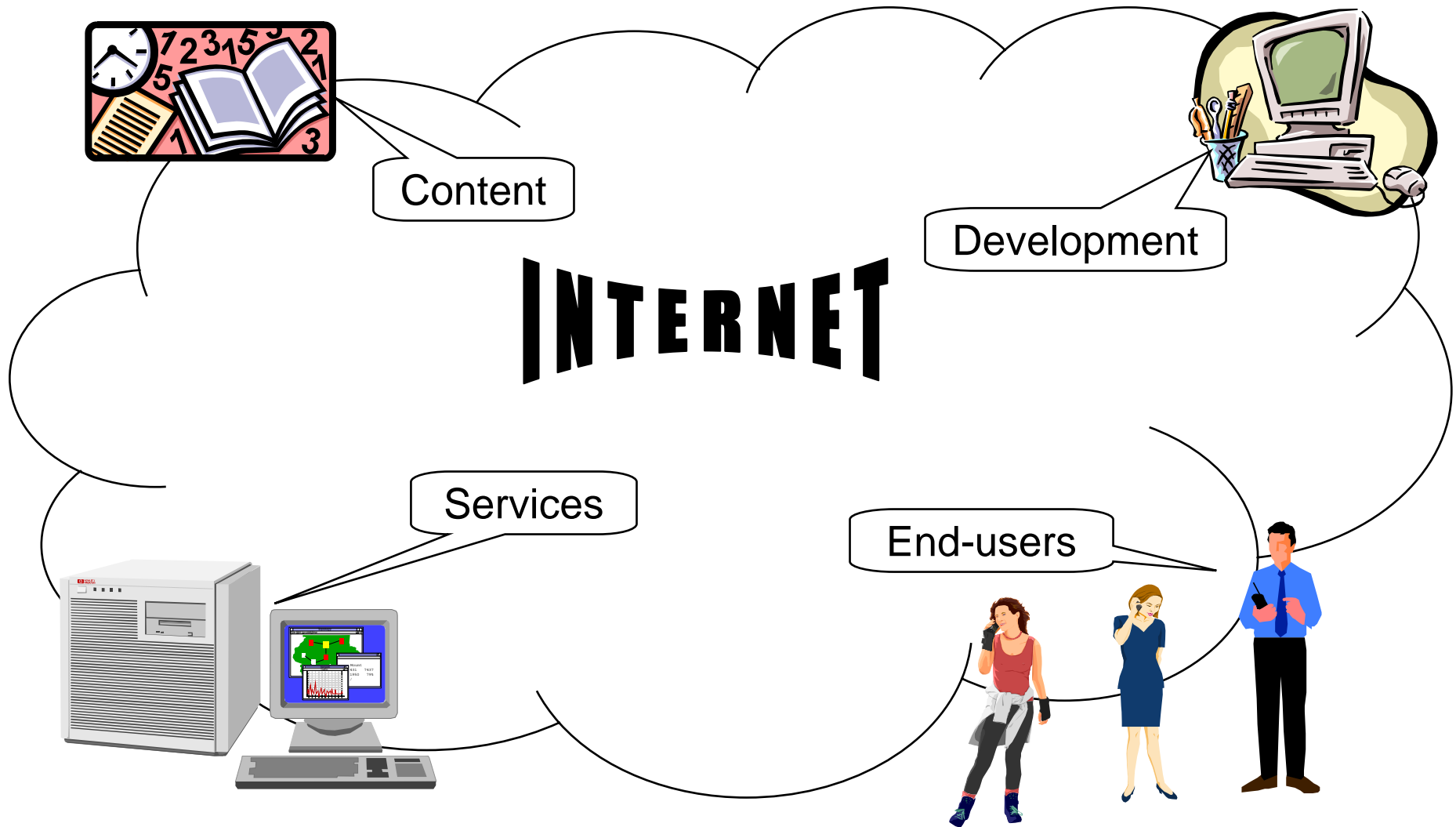
Purpose of the course

- Introduction and fundamentals of CBD
 - Evolution of software development
 - Need for software reuse
 - '60s: functions/procedures
 - '70s: modules
 - '80s: objects
 - '90s: components
 - What is a component
 - What are the characteristics of a component
 - What are the differences of components from modules and from objects

Purpose of the course (*cont.*)

- Introduction and fundamentals of CBD
 - How to design and to develop components
 - How to design and to develop component systems
 - How to integrate legacy software in component systems
 - Design methods, models and tools for CBD support
 - Populating the web with 3rd-tier services

Purpose of the course (cont.)



Purpose of the course (*cont.*)

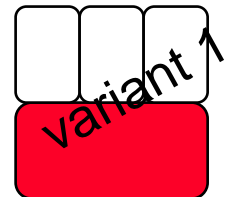
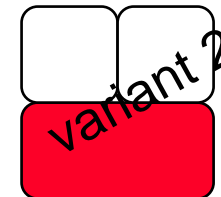
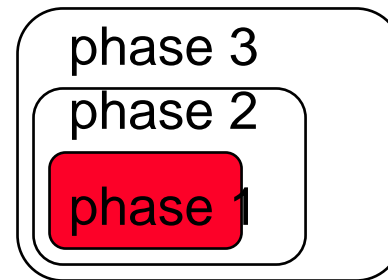
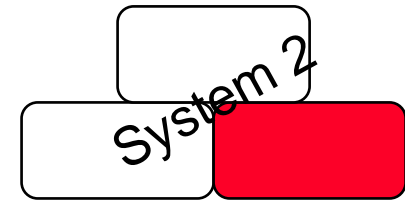
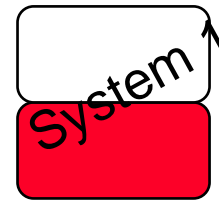
- Design for CBD
 - Reusable software
 - Composable software
 - Configurable systems
 - Rapid Application Development
 - Future proof systems
 - Simplicity is a virtue (KISS)
 - Many terms and concepts to chose from
 - Small subset of appropriate constructs
 - Consistent use of the chosen constructs

Purpose of the course (*cont.*)

- Practical experience
 - Homework to enable the participants to comprehend the fundamentals of CBD
 - Development exercises to get 1st hand experience on the taught material (UIoLI)
 - CCM, EJB, Web Services (, DCOM)

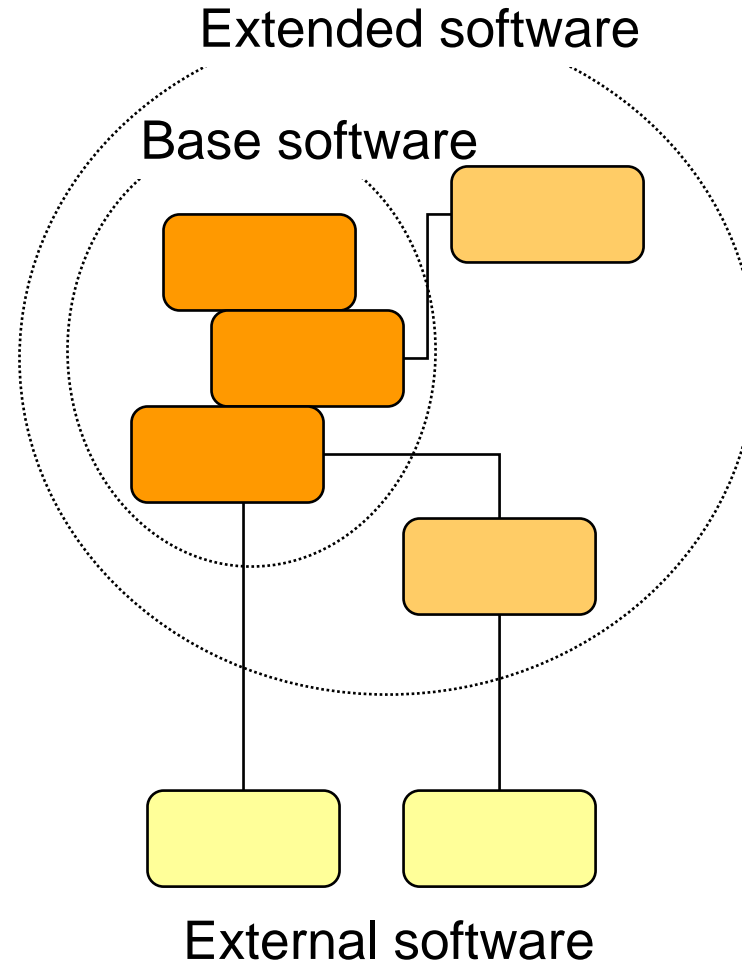
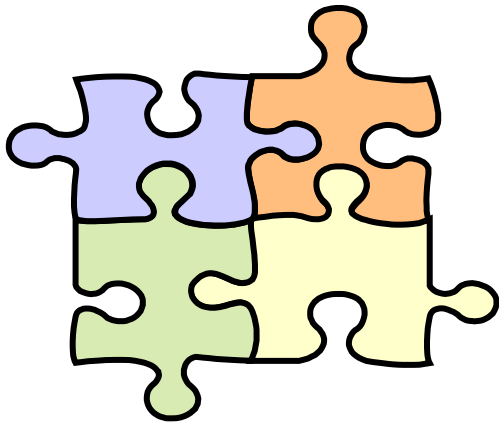
Motivations

- Software Reuse
 - Same code, many applications
 - Incremental development
 - Multiple variants



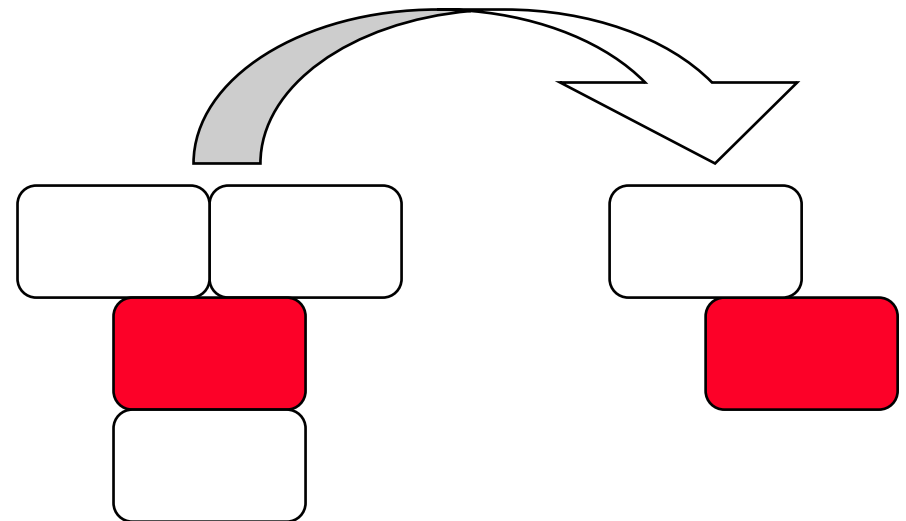
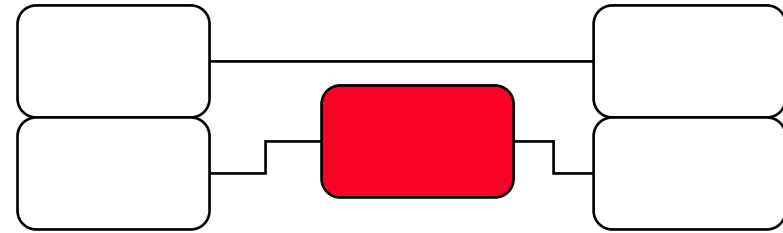
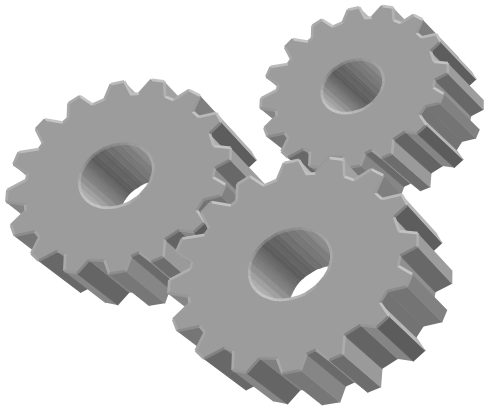
Motivations (*cont.*)

- Composable Software
 - 3rd party software
 - Integrate with existing software
 - Extend on request



Motivations (*cont.*)

- Configurable Systems
 - Interoperable software
 - Adjustable software



Motivations (*cont.*)

- Future proof systems
 - Integration of new functionality
 - Access of new service
 - Extensible software

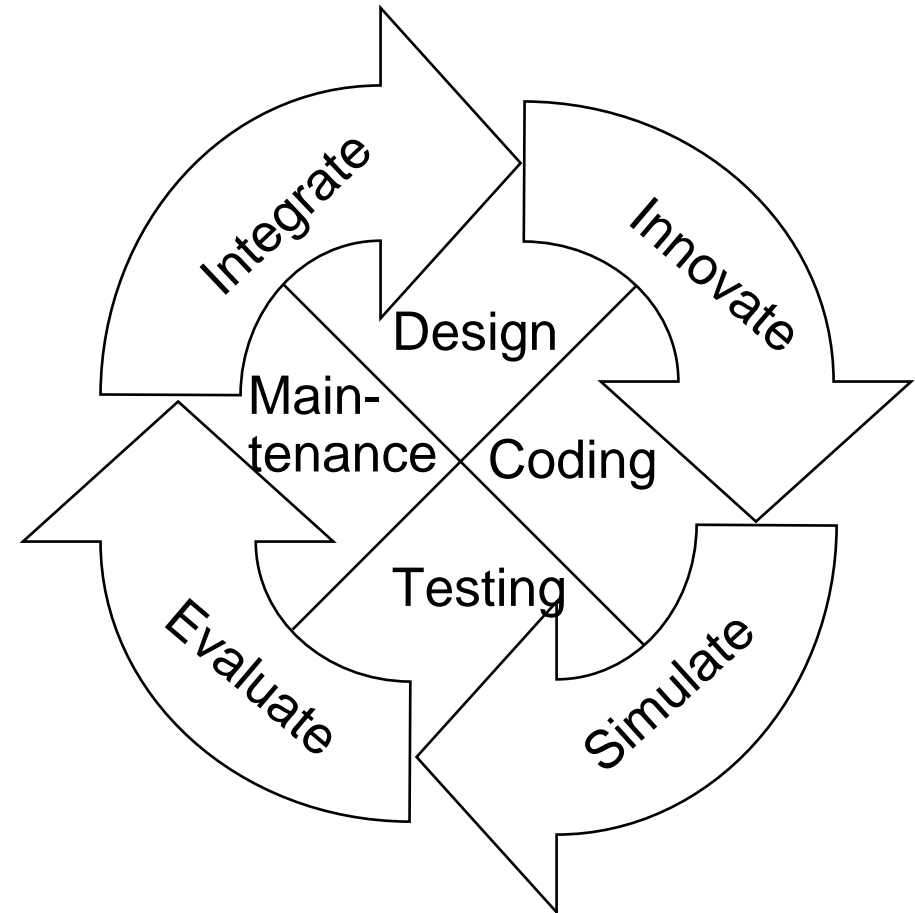
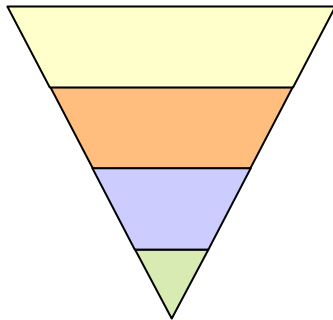


Composable
+ **Configurable**

Future Proof

Motivations (cont.)

- Rapid Application Development
 - Iterative development
 - Step-wise development



Prerequisites for the participants

- S.E., OO development, Internet technologies
 - Basics of software engineering
 - Basics of OO analysis and development
 - Basics of objects, classes, inheritance, interfaces
 - Java and C++ (2nd part of the course)
 - Basics of UML
 - Basics of Internet
 - HTML
 - CGI
 - Servlets

What will the participants learn

- CBD state-of-the-art
 - Analysis and design of components and component systems
 - Component models: similarities and discrimination factors
 - Support (tools and frameworks) for CBD
- Development of 3rd party software
 - Developing components that will be assembled by others
 - Assembling components developed by others
 - COTS based software

Structure of the course

- 2 parts
 - Theory of CBD – (4cr)
 - 10 lectures
 - 2 homework assignments (preparation of participants presentations)
 - CBD in practice – (4cr)
 - 4 lectures
 - 4 homework assignments (development of EJB, CORBA and Web Services components)
- Last lecture on April 27, 2006
- *Skip to the CBD web-pages*
 - <http://www.cs.helsinki.fi/u/przybils/courses/CBD06/>

Course Schedule

Thursday 19.01.2006

Thursday 26.01.2006

Thursday 02.02.2006



Thursday 16.02.2006

Thursday 23.02.2006

Thursday 02.03.2006

Thursday 09.03.2006

Thursday 16.03.2006

Thursday 23.03.2006

Thursday 30.03.2006

Thursday 06.04.2006

Thursday 13.04.2006

Thursday 20.04.2006

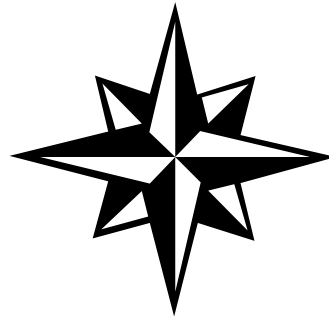
Thursday 27.04.2006

14:15 – 16:00

Exactum, C 221

Friday 05.05.2006

Questions?



- *Groups*
- *Practical homework deadlines*
- *Sign up: email to me (Name, student-ID)*
- *Exam and grading*

Next Time

- 26.01.2006
 - Evolution of programming
 - from structured programming to CBD
 - Evolution of software systems
 - from modules to components
 - Evolution of design methodologies
 - from functional analysis to n-tier architecture modeling
 - Evolution of software development support
 - from monolithic systems to service-oriented solutions
 - What is a Component
 - Basic elements of a component
 - Components and Objects
 - Partners or Competitors?
 - Components and Distributed Systems
 - Evolution, succession or coexistence?