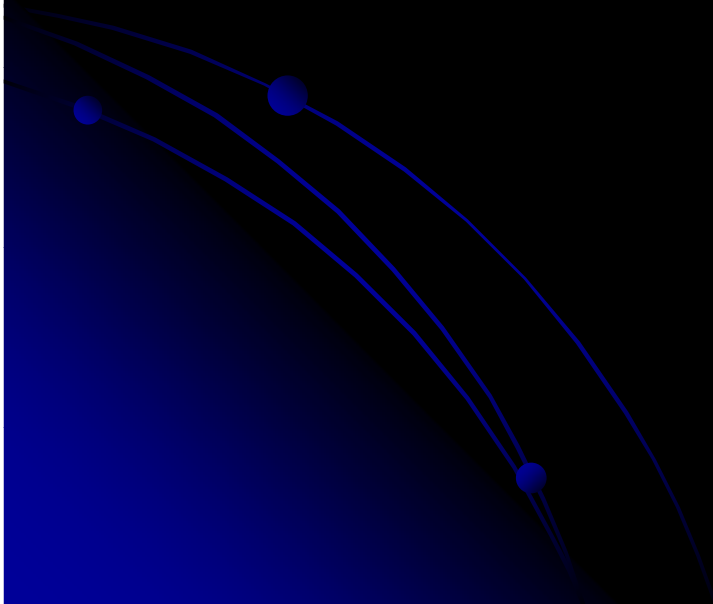


Object-Oriented Development

by Linda M. Northdrop

200614167

양한석



Historical Perspective

- Object-Oriented Design
- Object & Object Attributes
 - Simula
 - by Kristen Nygaard & Ole-Johan Dahl
 - SmallTalk
 - by PARC (Palo Alto Research Center)
 - ➔ SmallTalk-80
- 1980 – Concepts of OOD
- 1985 – OO based Database System

Motivation

- Object Orientation is Silver Bullet.
- Watching the world as Object is closer to Human Thinking.
- OOD supports Information Hiding, Data Abstraction, Encapsulation.
- Object Approach
 - naturally offers Prototype.
 - Reusability is for SW, Design, and Analysis Model.
 - Reduce Risk of Complex Systems

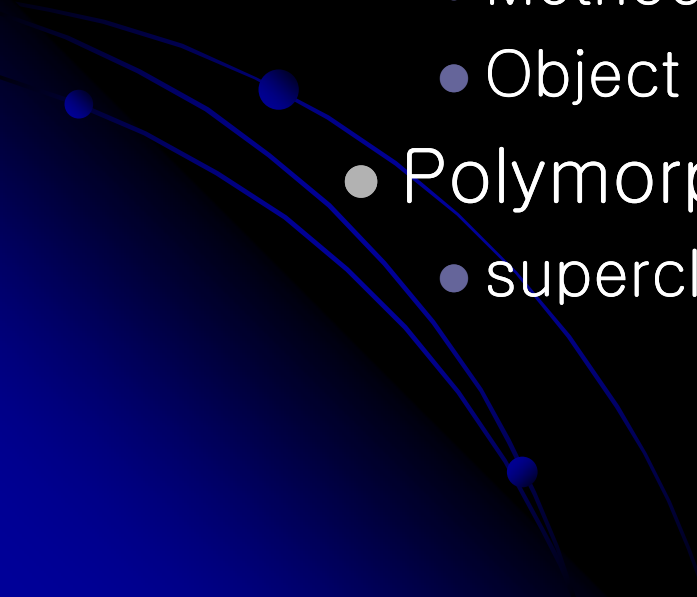
Object-Oriented Model(1/2)

- Object-Oriented Model??
 - Structure and Processing
- Framework
 - Abstraction (추상화)
 - Encapsulation (캡슐화)
 - Modularity (모듈화)
 - Hierarchy (계층화)
 - Typing (정형성)
 - Concurrency (동시성)
 - Persistence (영속성)
 - Reusability (재사용성)
 - Extensibility (확장성)

Object-Oriented Model(2/2)

- Approach
 - Object Orientation is the integration of Procedural & Data-Driven Approach
 - New Software Development Approach
- Progress
 - Major factor is the advance in Programming Methodology
 - Abstractions From Address, to name (Assembly), to expression (FORTRAN), to control (COBOL), to procedure and function (Pascal), to modules and data (Modula 2), and to object (OO based & OO Languages)
- Object-Oriented Techniques
 - Mainstream of Industrial-Strength software development

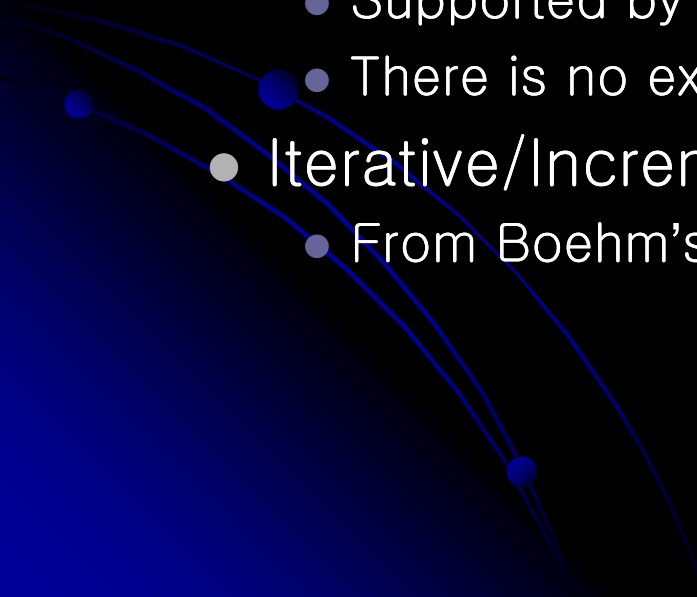
Object-Oriented Programming (1/2)

- Concepts
 - Object
 - Data Structure and Operations
 - Message
 - Method or Routine
 - Object Chain
 - Polymorphism
 - superclass with subclass
- 

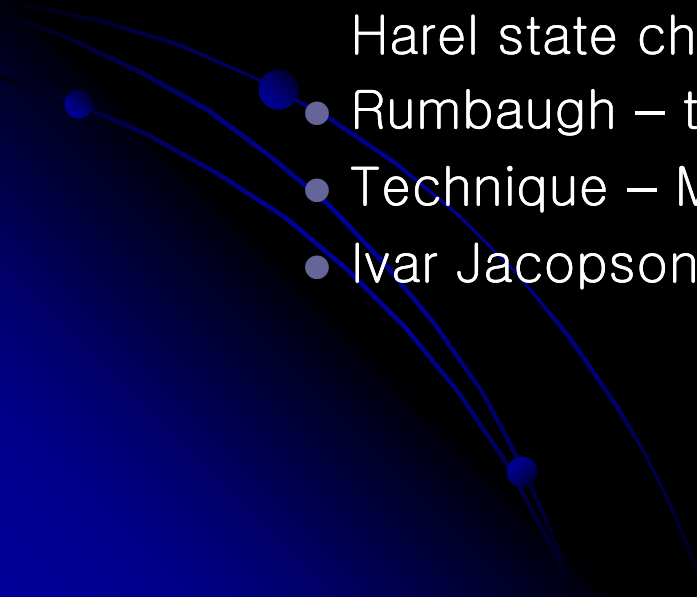
Object-Oriented Programming (2/2)

- Languages
 - Simula is the common ancestor of all of these languages
 - SmallTalk Based
 - SmallTalk-80 (the truest OO Language)
 - C Based
 - Objective-C, C++ (Encapsulation by STRUCT, Polymorphism, Multiple Inheritance), Java
 - LISP Based
 - LOOPS, Flavors, Common LOOPS, CLOS (Common LISP Object System)
 - Pascal Based
 - Object-Pascal, Turbo Pascal, Eiffel, ADA95 (Not the first ADA)

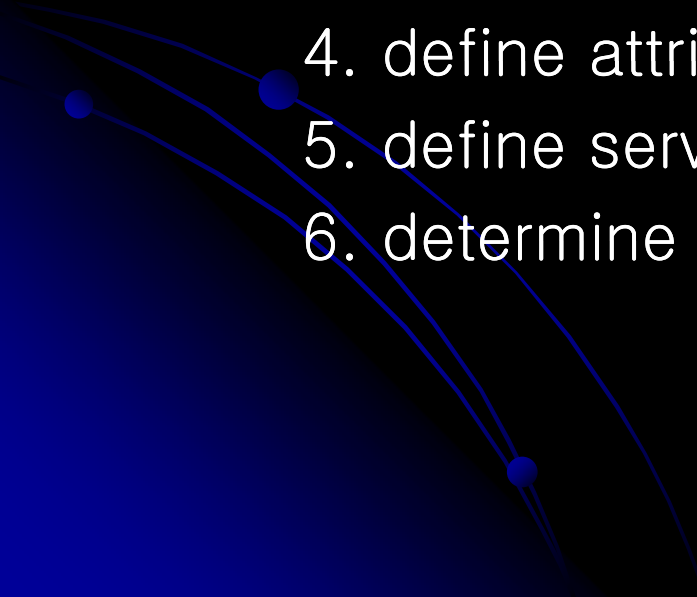
Object-Oriented Software Engineering (1/3)

- Life Cycle
 - Coding is not the first consideration
 - Waterfall / Structured Life Cycle
 - The software crisis by sequential processing
 - Water Fountain Life Cycle
 - Supported by Prototyping and Feedback Loop
 - There is no exact Boundary
 - Iterative/Incremental Life Cycle
 - From Boehm's Spiral Model
- 

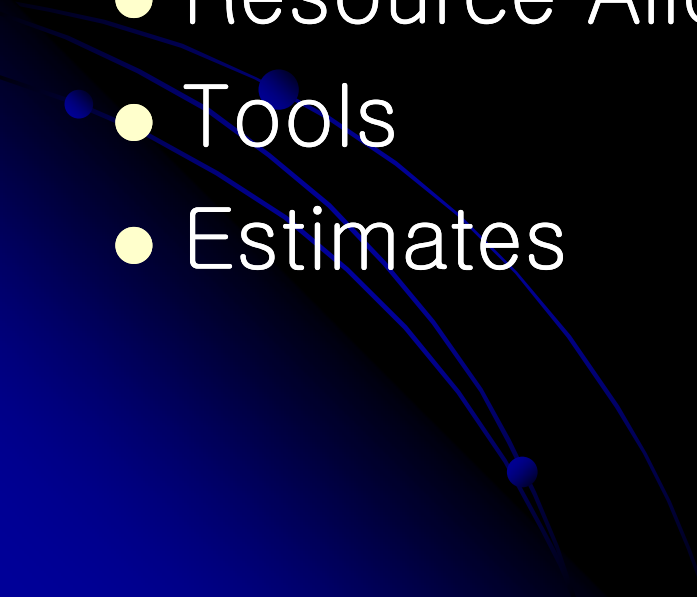
Object-Oriented Software Engineering (2/3)

- Object-Oriented Analysis and Design
 - Framework
 - Object-Oriented Design
 - Reusability
 - Meyer – Class Diagrams, Class Category diagrams, Class Templates, and Object Diagrams to record Design by Harel state chart
 - Rumbaugh – the Object, Dynamic, Functional Model
 - Technique – MultiLayer & MultiComponent
 - Ivar Jacopson's Method – Objectory with “use case”
- 

Object-Oriented Software Engineering (3/3)

- Oriented-Object Analysis
 - Techniques
 1. find classes and objects
 2. identify structures and relationships
 3. determine subjects
 4. define attributes
 5. define services
 6. determine a multilayer object-oriented model
- 

Management Issues

- Milestones
 - Object-Oriented Development is the new milestone.
 - No LOC (Lines of Code)
 - Resource Allocation
 - Tools
 - Estimates
- 

- Object-Oriented Transition
 - Object-Oriented Approach is the successful way for any Project

- THE FUTURE

- Objects are the most exciting innovation in software since the 70s
- Don't know about the future.