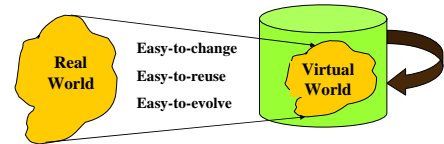


Object Oriented Software Development Goal and Scope

Koichiro Ochimizu
Japan Advanced Institute of
Science and Technologies
School of Information Science

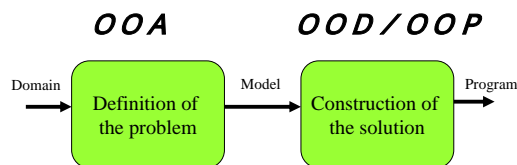
Three major merits of OOSD

- Project the real world into the computer as you recognize and understand it.
- Maintain the virtual world constantly corresponding to mismatches between the real world and the virtual world and evolution of the real world.



Object Oriented Analysis/Design/Programming

Iterative and Incremental



2005/2/26

Scope and Goal 1/2

- **Goal** Understanding of basic principles and concepts and their application to practical use.
- **Content**
 - Basic principles and concepts
 - Modeling Language(UML) and Programming Languages(Java)
 - Object-oriented Software Development Method (Unified Process)
 - Software Patterns (Analysis Patterns, Architectural Patterns, Design Patterns, Idioms, Process Patterns)

Scope and Goal 2/2

- **History and Views**
 - Technical merits of object oriented approach
 - Achievement (information hiding, abstract data type, inheritance)
 - Current topics (Design patterns, Application framework, software architecture, middleware)
- **Case Studies**
 - Matched Parenthesis
 - Dining Philosophers
 - House Alarm System
 - Elevator Control System

Important Concepts to be studied

- Class and Instance (O.J. Dahl SIMULA67,1967)
 - Removal of redundant description
- Information hiding (D.L.Parnas)
 - Easiness of modifying a data structure
- Abstract Datatype
 - Both
- Inheritance
 - Reuse of classes by subclassing
 - Easiness of extension of functions by subtyping
- Polymorphism
 - Dynamic binding
- Use of the same concepts through analysis, design and programming
 - Simple correspondence among software artifacts

Object-Oriented Programming

- 1967: Simula by O.J. Dahl **Class and Instance**
- 1972: Parnas Module by D.Parnas **Information hiding**
- 1972: Smalltalk72(Xerox PARC)
- 1977: CLU by B. Liskov **abstract data type**
- 1981: Smalltalk80 by Xerox **class library**
- 1986: Objective-C by Cox, C++ by Strusrup

B. Liskov and S. Zilles: Programming with abstract data types , SIGPLAN Notices , 9(4), (1974) , pp.50--60 .

Object-Oriented Programming

- 1988: Eiffel by B. Meyer
- 1989: CLOS by Moon
- Now: Java

Object-Oriented Technologies (Object Oriented Analysis and Design)

- 1986: OOD by G. Booch
- 1988: Shlare/Mellor,
- 1991: Coad/Yordon,
- 1991: OMT by J.Rumbaugh
- 1995: OOSE by Ivar Jacobson
- now: UML

References

- Hassan Gomaa, Designing Concurrent, Distributed And Real-Time Application with UML, Addison Wesley, (2000).
- Hassan Gomaa, Designing Softwar Product Line with, Addison Wesley, (2004).
- Grady Booch: Object Oriented Design with Applications, The Benjamin/Cummings Publishing Company, (1991).
- James Rumbaugh et. al. : Object-Oriented Modeling and Design), Prentice-Hall Englewood, N.J., (1991).
- Ivar Jacobson et. al. : Object-Oriented Software Engineering -- A Use Case Driven Approach, ACM Press, (1992).
- Ivar Jacobson, James Rumbaugh, Grady Booch: The Unified Software Development Process, Addison-Wesley, (1999).
- James Rumbaugh, Ivar Jacobson, Grady Booch: The Unified Modeling Language Reference Manual, Addison-Wesley, (1999).
- Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modeling Language User Guide, Addison-Wesley, (1999).

References

- Martin Fowler, " Analysis Patterns: Reusable Object Models" Addison-Wesley Publishing Company, 1997.
- Frank Bushmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, "Pattern-Oriented Software Architecture --- A System of Patterns" John Wiley & Sons, 1996
- Douglas Schmidt, Michael Stal, Hans Rohnert, Frank Bushmann, "Pattern-Oriented Software Architecture Vol.2 --- Patterns for Concurrent and Networked Objects" John Wiley & Sons, Ltd, 2000.
- Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design Patterns", Addison-Wesley Publishing , 1995.
- James O. Coplien, "A Generative Development Process Pattern Language", Pattern Languages of Program Design1 (PLoP), 1995.