

Software Development Methodologies

Lecturer: Raman Ramsin

Lecture 1: Basics

Department of Computer Engineering



Software Development Methodology (SDM)

- A framework for applying software engineering practices with the specific aim of providing the necessary means for developing software-intensive systems
- Consisting of two main parts:
 - A set of modeling conventions comprising a *Modeling Language* (syntax and semantics)
 - □ A *Process*, which
 - provides guidance as to the order of the activities,
 - specifies what artifacts should be developed using the *Modeling* Language,
 - directs the tasks of individual developers and the team as a whole, and
 - offers criteria for monitoring and measuring a project's products and activities.

Department of Computer Engineering



Object-Oriented Software Development Methodology (OOSDM)

- Specifically aimed at viewing, modeling and implementing the system as a collection of interacting objects
- First appeared in late 1980s
- Categorized as
 Seminal (First and Second Generations) Integrated (Third Generation) Agile
- UML was the result of the 'war' among seminal methodologies
- Process has now replaced modeling language as the main contentious issue



Genealogy: Seminal and Integrated Methodologies (until 1996)



4

Department of Computer Engineering

Sharif University of Technology

[Webster 1996]



UML



[Graham 2001]

Department of Computer Engineering



Genealogy: Agile Methodologies



[Abrahamsson et al. 2003]

Department of Computer Engineering



Topics: Overall vs. Selected

- Seminal Methodologies
 - 1. Coad-Yourdon (1989, 1991)
 - 2. RDD (1990)
 - 3. Booch (1991, 1994)
 - 4. OMT (1991)
 - 5. OOSE (1992)
 - 6. BON (1992, 1995)
 - 7. Syntropy (1994)
 - 8. Fusion (1994)
- Integrated Methodologies
 - 1. OPM (1995, 2002)
 - 2. OPEN (1996, 2010)
 - 3. <u>RUP</u> (1998, 2000, 2003, ..., MEC-2023)
 - 4. <u>USDP/UP</u> (1999, ..., MEC-2023)
 - 5. TSP/PSP (1999, 2010)
 - 6. <u>EUP</u> (2000, 2005)
 - 7. FOOM (2001, 2007)

- Agile Methodologies/Frameworks
 - 1. <u>DSDM</u> (1994..2014)
 - 2. <u>Scrum</u> (1995..2020)
 - 3. <u>XP</u> (1996, 1999, 2004, 2013)
 - 4. ASD (1997, 2000)
 - 5. <u>Crystal</u> (1998, 2004, 2006)
 - 6. <u>FDD</u> (1999, 2002)
 - 7. AUP (2006)
 - 8. <u>DAD</u> (2012, 2020)
- <u>SME Approaches</u> (2014)
- Process Patterns/Antipatterns
 - 1. <u>Ambler</u> (1998, 1999)
 - 2. <u>Neill et al.</u> (2012)
- Process Metamodels
 - 1. OPF (2001, 2009)
 - 2. <u>SPEM 2.0</u> (2008)



Problems

- Requirements engineering is still the weak link.
- Model inconsistency is a dire problem.
- Integrated methodologies are too complex to be effectively mastered, configured, and enacted.
- Some prominent agile methods are not capable enough:
 - □ Unrealistic assumptions (e.g. Scrum)
 - □ Lack of scalability (All, more or less)
 - □ Lack of a specific, unambiguous process (e.g. XP, Crystal)
- Seamless development, pioneered by seminal methodologies, is not adequately appreciated and supported in modern-day methodologies.



Methodology Development

- Methodologies can be categorized according to the circumstances leading to their development, including the approach and method applied:
 - □ *Revolutionary*: novel ideas and approaches
 - □ *Evolutionary*: based on existing methodologies
 - *Extension*: adding new features to an existing methodology
 - *Integration*: consolidating ideas from two or more methodologies
 - Merger: typically carried out through a design-by-committee procedure.
 - Ad hoc: features are scavenged from prominent methodologies in order to fill the needs of the methodologist.
 - Engineered: based on analysis of the problem domain and requirements thereby identified, and pre-implementation design.
- Modern Solution: <u>Situational Method Engineering</u>
 Software processes are software too.'



References

- Webster, S., "On the evolution of OO methods", Bournemouth University, 1996.
- Graham, I., Object-oriented Methods: Principles and Practice (3rd Edition), Addison-Wesley, 2001.
- Abrahamsson, P., Warsta, J., Siponen, M. T., Ronkainen, J., "New directions on agile methods: A comparative analysis", Proceedings of the International Conference on Software Engineering – ACM/ICSE 2003, 2003, pp. 244-254.
- Ramsin, R., Paige, R. F., "Process-centered review of object-oriented software development methodologies", ACM Computing Surveys, vol. 40, no. 1 (February), 2008, Article 3, pp. 1-89.