MACIASZEK, L.A. (2007):
Requirements Analysis and System Design, 3rd ed.
Addison Wesley, Harlow England
ISBN 978-0-321-44036-5

Chapter 1
Software Process

Topics

- The nature of software development
- System planning
- Systems for three management levels
- The software development lifecycle
- Development models and methods
- Problem statements for case studies (separate set of slides)

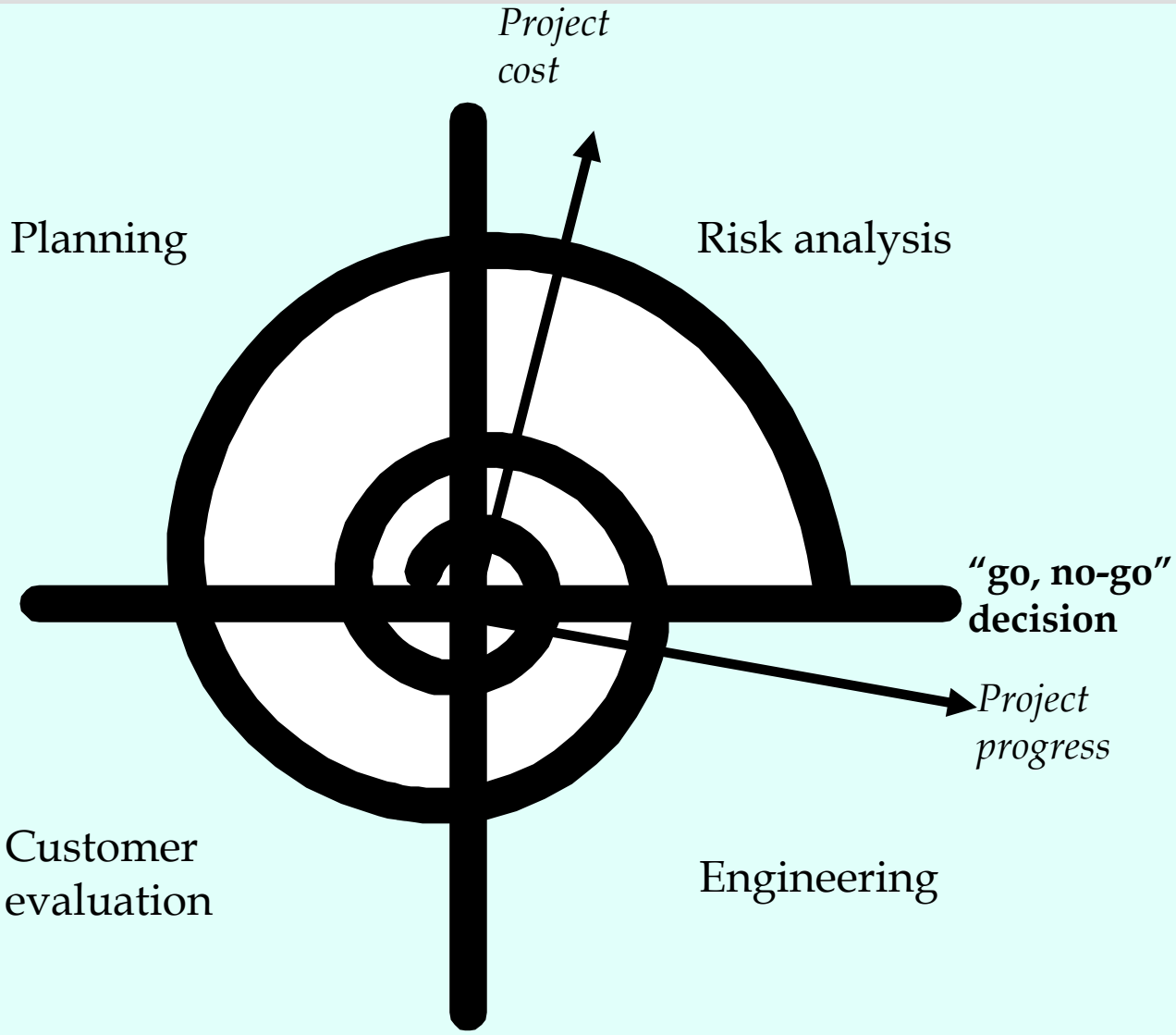
5. Development models and methods

- “how” of software production
- called also lifecycle models

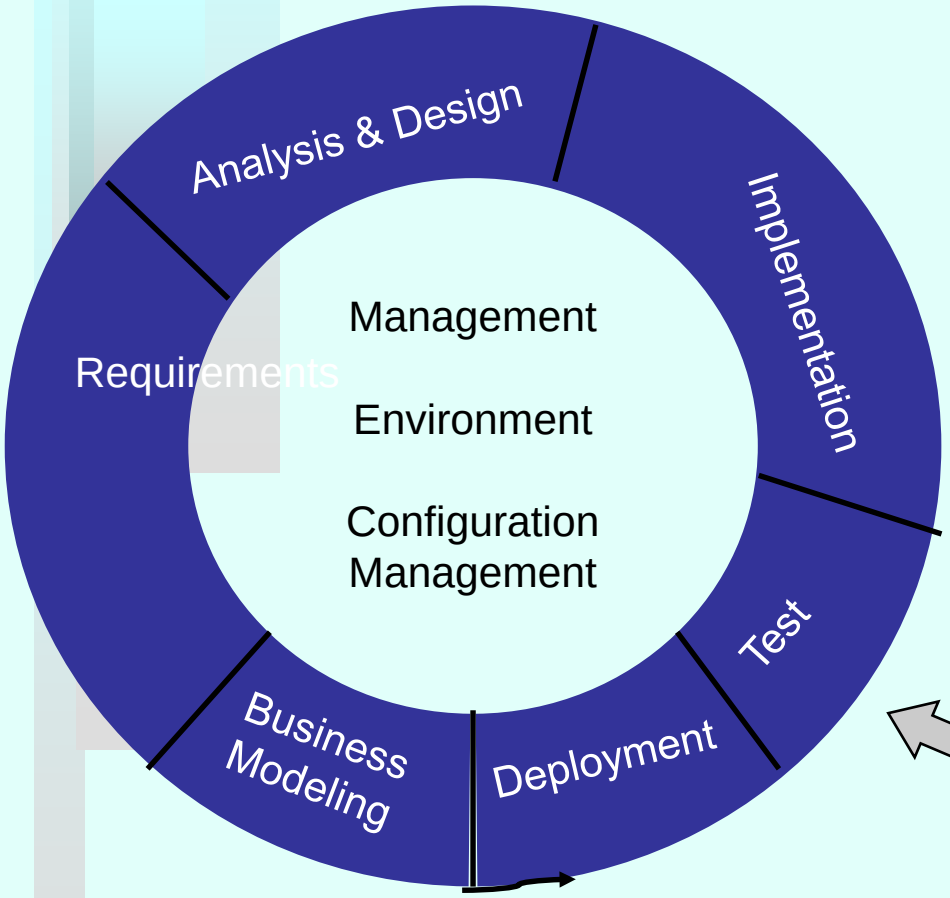
Development models

- Model defines an approach to software production
- Has an associated process – iterative and incremental
- Process is unique (different) for each organization
- Representative models for iterative and incremental development:
  - the spiral model
  - the IBM Rational Unified Process (RUP)
  - Model Driven Architecture (MDA)
  - agile software development
  - aspect-oriented development

Spiral model



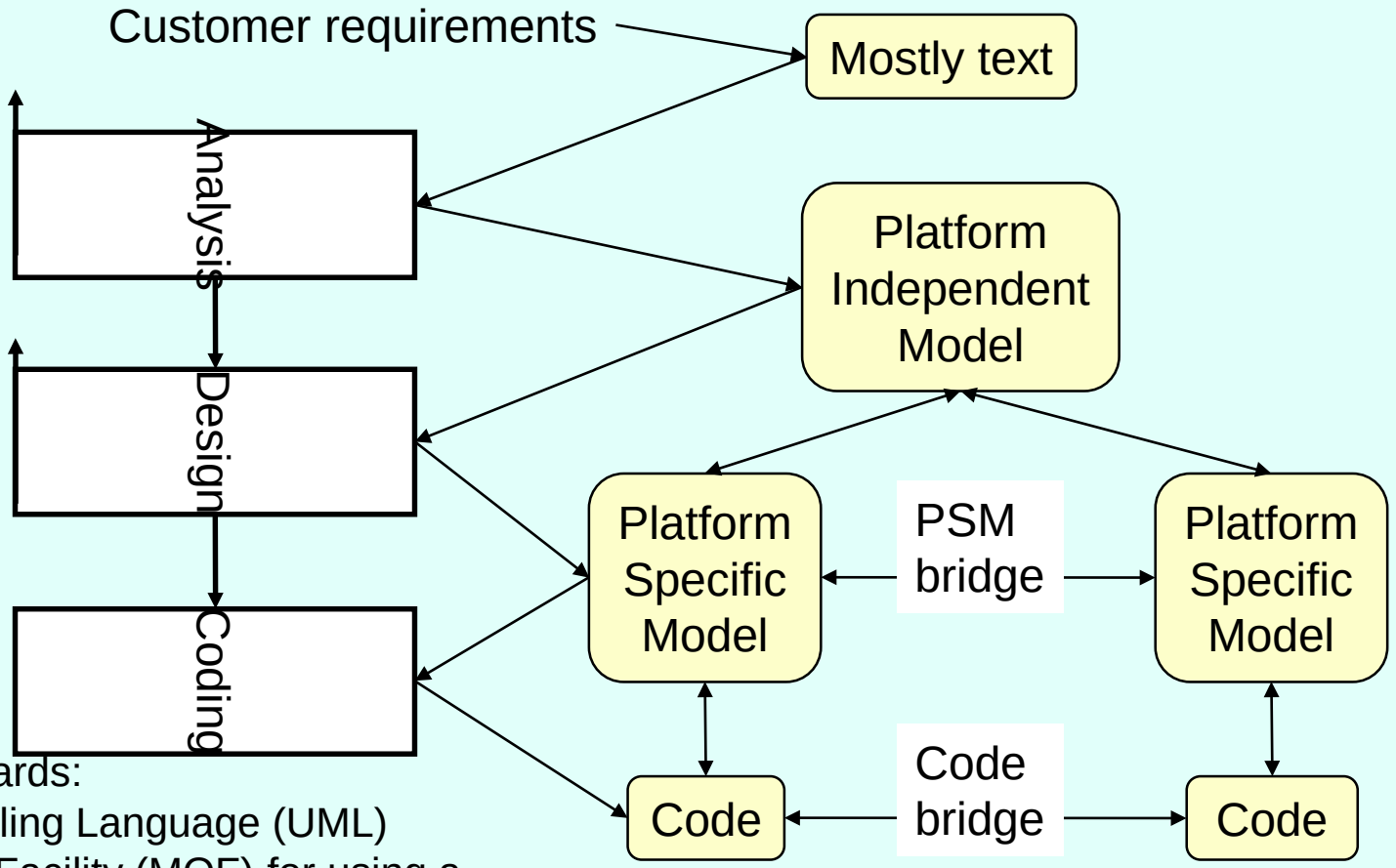
IBM Rational Unified Process (RUP)



Start of the process

- RUP organizes projects in two-dimensional terms
- The **horizontal dimension** represents the successive phases of each project iteration:
  - inception,
  - elaboration,
  - construction, and
  - transition.
- The **vertical dimension** represents seven software development disciplines and supporting activities of configuration and change management, project management, and environment.

Model Driven Architecture (MDA)



Applicable standards:

- Unified Modeling Language (UML)
- Meta-Object Facility (MOF) for using a standard meta-model repository so that derived specifications can work together
- XML Meta-Data Interchange (XMI) for mapping UML to XML for interchange purposes,
- Common Warehouse Meta-model (CWM) for mapping of MDA to database schemas and permitting flexible data mining.

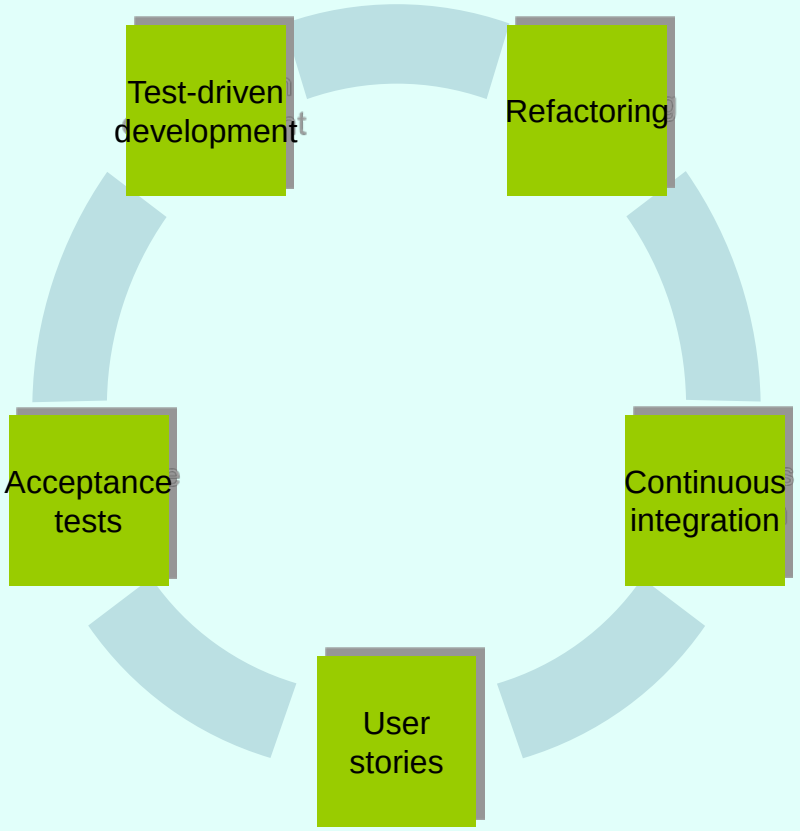
Agile software development

Key points of agility in software production:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

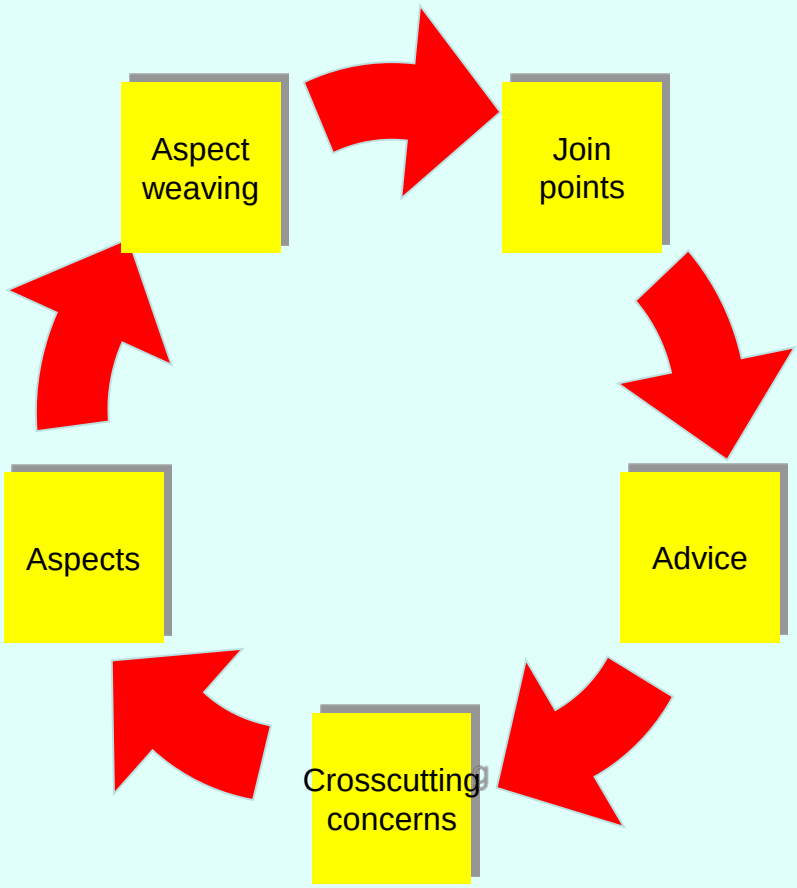
The best known representatives:

- eXtreme Programming (XP)
- Aspect Oriented Software Development
- Feature-Driven Development
- Lean Development



Aspect-oriented software development

- Produces more modular systems by identifying so called *crosscutting concerns*
  - Typically, nonfunctional requirements/qualities
- The modules are called *aspects*
  - Base code vs aspect code
- The aspects are integrated through the process called *aspect weaving*
  - Static vs dynamic weaving
  - Applies to *join points* in a program's execution
- A particular action that needs to be taken for a join point is called an *advice*



Review Quiz 1.5

1. Is adding a new functionality to a software project the responsibility of a software iteration or integration?
2. Which of the development models/methods is most explicit about risk analysis?
3. Which development model/method is directly linked to the traditional concept of formal specifications?
4. Which development model/method is directly linked to the concept of intentional programming?
5. Which development model/method is directly linked to the concept of crosscutting concerns?