



Method Engineering

Introduction

Session 1

7 February 2019

Prof. dr. Sjaak Brinkkemper

Dr. Sietse Overbeek



Universiteit Utrecht

Suppose...

- You just graduated from MBI and got a nice job as Junior Consultant at Deloitte Consulting
- Your first project: Implement Google Analytics and integrate this with the Web Content Management system for both intranet and all corporate websites in a large organization
- Where do you start?
- What activities are you planning to perform?
- What should be delivered by the team to the customer?

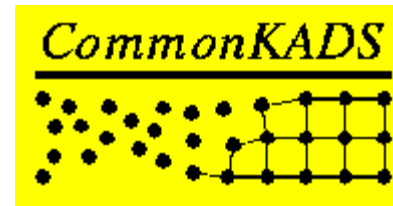




Task analysis



DSCM[®]



Which one to choose?

UML Object-oriented
Systems
Development

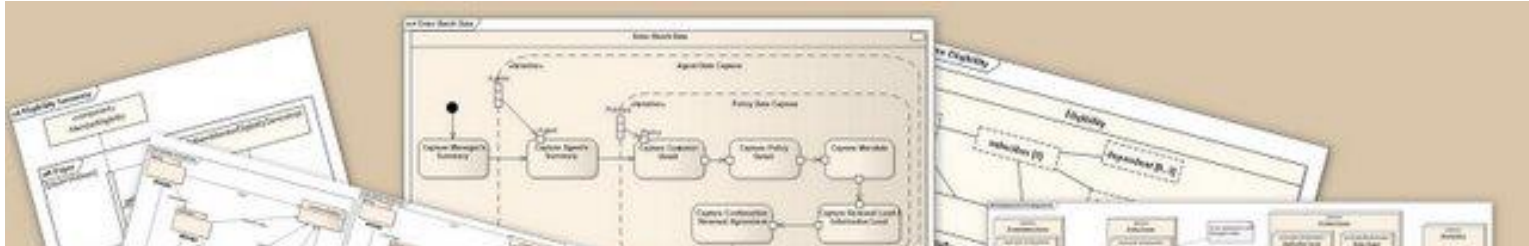
Data Model
Use Case



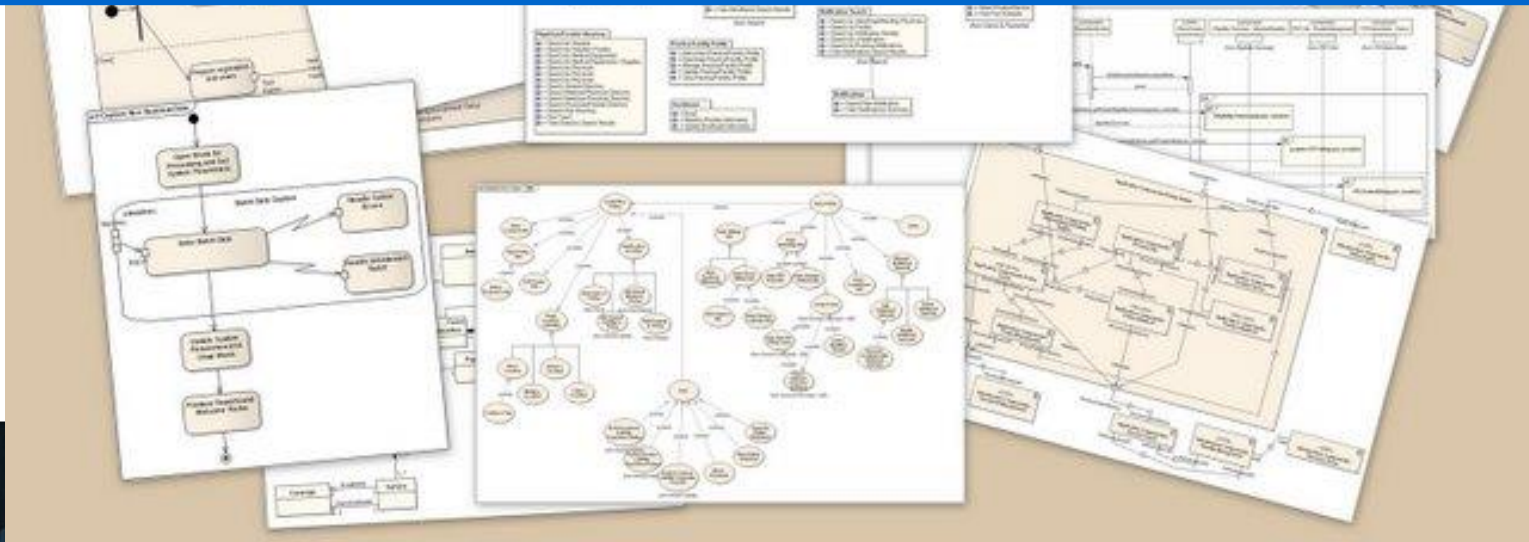
Class Diagram
Class Diagram
Structured
Analysis and
Design



A collage of UML diagrams



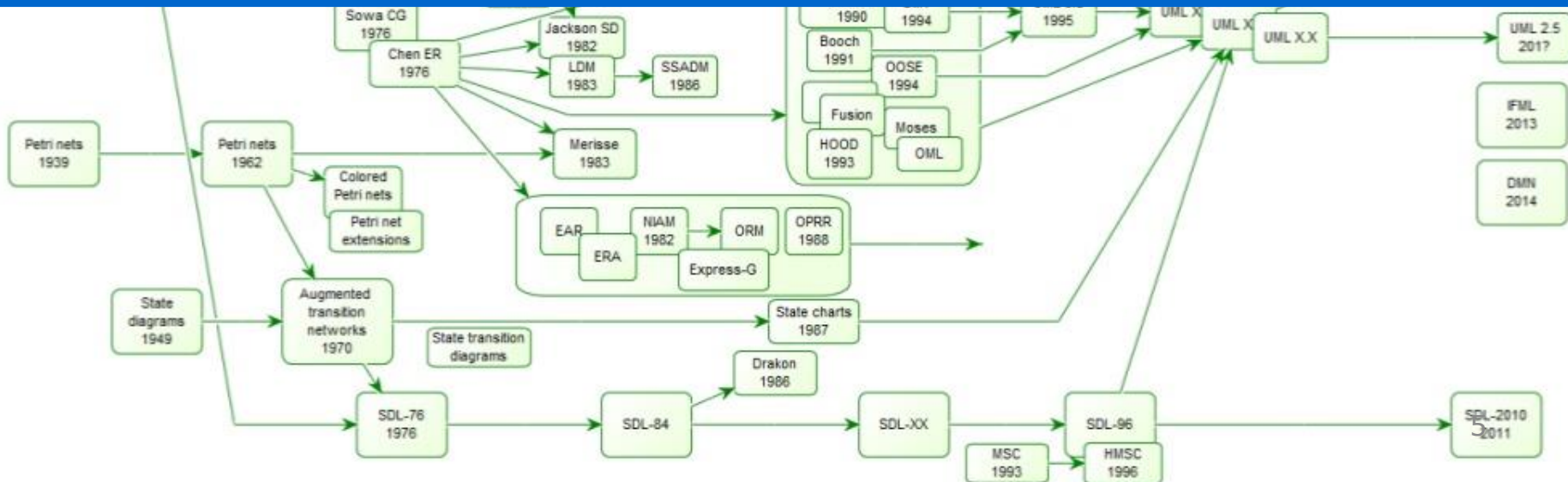
When to apply what diagram?



And most methods, techniques and models are related...

Source: www.modeling-languages.com

But what are the differences?



So again, suppose...

- You just graduated from MBI and got a nice job as Junior Consultant at Deloitte Consulting
- Your first project: Implement Google Analytics and integrate this with the Web Content Management system for both intranet and all corporate websites in a large organization
- Where do you start?
- What activities are you planning to perform?
- What should be delivered by the team to the customer?



This course will help you in...

Learning objectives:

- Providing of insight and skills into the **systematic description, explanation and evaluation** of all aspects of the methodology of ICT systems development
- Being able to **use and adapt** methods, techniques and tools for the **situation** of the project or system
- Contributing to the **body of knowledge** on method engineering
- Being able to review **scientific literature**, analyzing a paper in depth, and create a **videoclip** of the results



Lecturers

- Sjaak Brinkkemper
- Sietse Overbeek



Student assistants

- Kleopatra Chasioti
- Melchior Keijdener
- Philippe van der Voorn



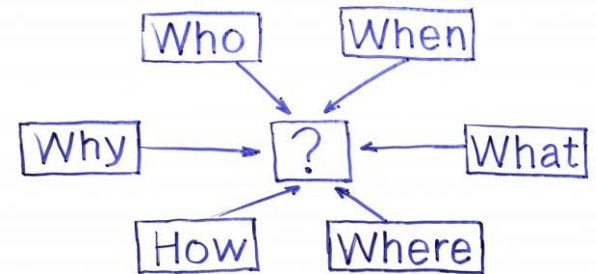
Agenda

- Introduction to method engineering
- Practical issues
- Assignment week 7 & 8



Typical issues in the Information Systems domain

- How should this **project** be executed?
- Which **method** is the best?
- What **model** should be created now?
- What is a good **tool** for that method?
- How can we get to the next level of **process maturity**?



Response to these issues

- Many **methods and techniques** have been proposed
- **No generic answer** can be given.
- It all depends on the **situation**:
 - project, company, people, budget, timing, technology, platforms

What is needed:

- Formal study of methods and tools
- Adaptation of methods

→ **Scientific domain of Method Engineering**



Method engineering

Scientific studies in Method Engineering:

- Formal **description** of methods
- Identification of method **fragments**
- **Situational configuration** of project specific methods
- Method **comparison**
- **Incremental** method evolution
- **Formalization** of method **properties**
- Derivation of **tool** support

Conferences:

Conf. Advanced Information Systems Engineering (CAiSE)
IFIP WG8.1 Working Conference on Method Engineering
Exploring Modeling Methods for Systems Analysis and Design (EMMSAD)

Journals:

Requirements Engineering Journal
Information and Software Technology
Int. Journal Information Systems Modeling and Design



Main concepts in Method Engineering

- Method
- Activity
- Methodology
- Model
- Meta-model
- Diagram
- Technique
- Tool
- Notation
- MetaCASE
- Procedure

Read:

Brinkkemper, S. (1996). Method engineering: engineering of information systems development methods and tools. *Information and Software Technology* 38(4), 275-280.



Method

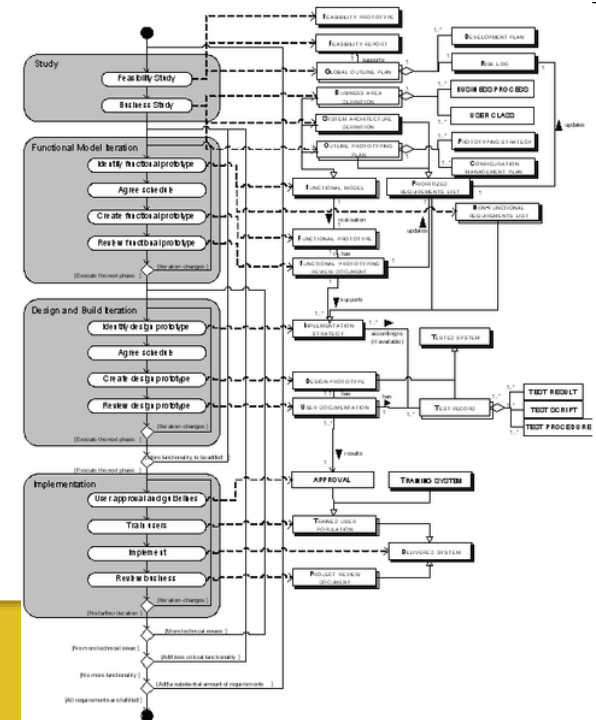
Definition

A method is an **approach** to perform a systems development project, based on a **specific way of thinking**, consisting of **directions and rules**, structured in a systematic way in development **activities** with corresponding development **products**.

Origins from Greek "methodos":
way of investigation

Examples

RUP, UM, SSADM, SDM, ITIL, T-map, DSDM, DYA, Scrum, and many, many others.



Examples of online methods

- DSDM
<http://www.dsdm.org/>
- Open Process Framework
<http://wiki.c2.com/?OpenProcessFramework>
- ISPMA-Compliant Software Product Management
<https://doi.org/10.1007/978-3-642-55140-6>
- SCRUM
<http://www.scrumalliance.org/>



Phasing of methods



A **method** consists of **phases**

- Preparation phase
- Several execution phases
- Completion phase

Phase end: **Major deliverable**; **Go / No Go decision**



Phases

Dynamic Systems Development Method

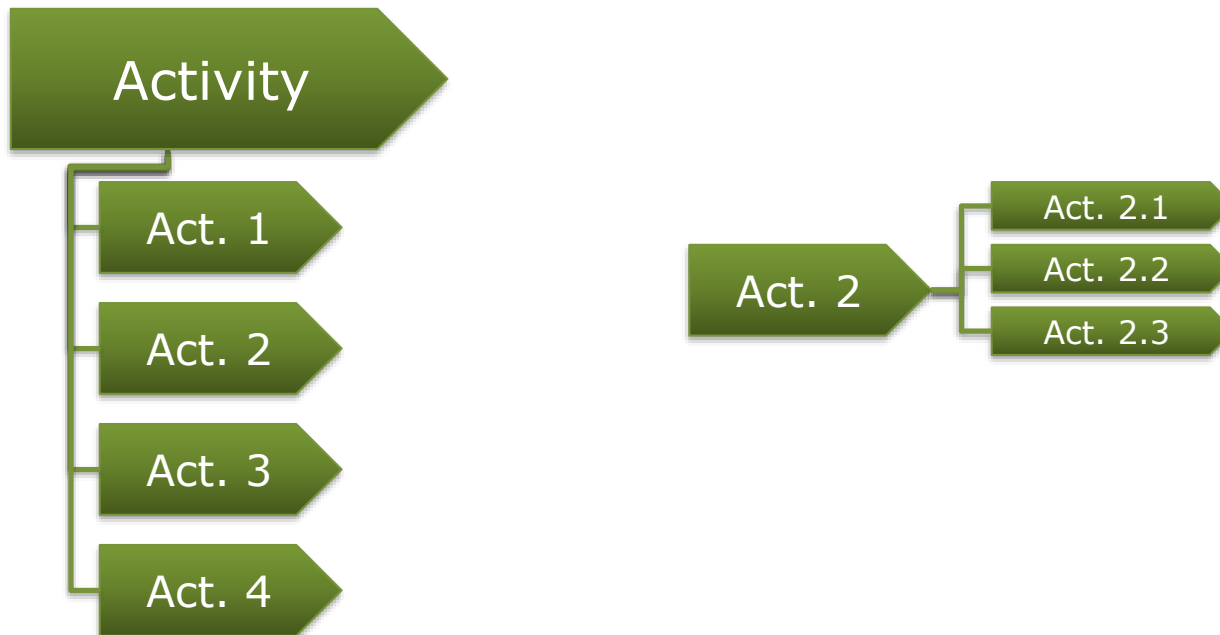


Scrum

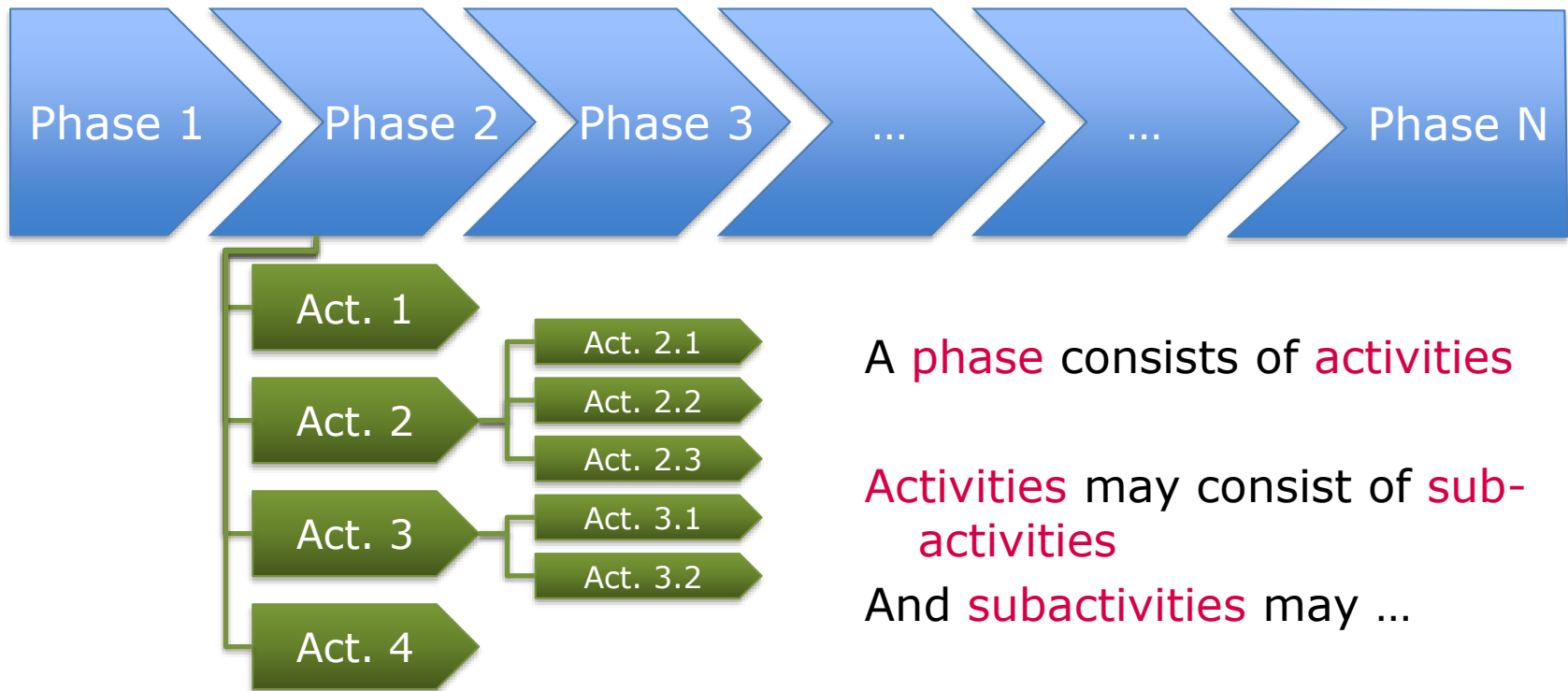


Activity

- An **activity** consists of a set of basic activities to be executed possibly **conditionally** in some **order**.



Activities of methods



A **phase** consists of **activities**

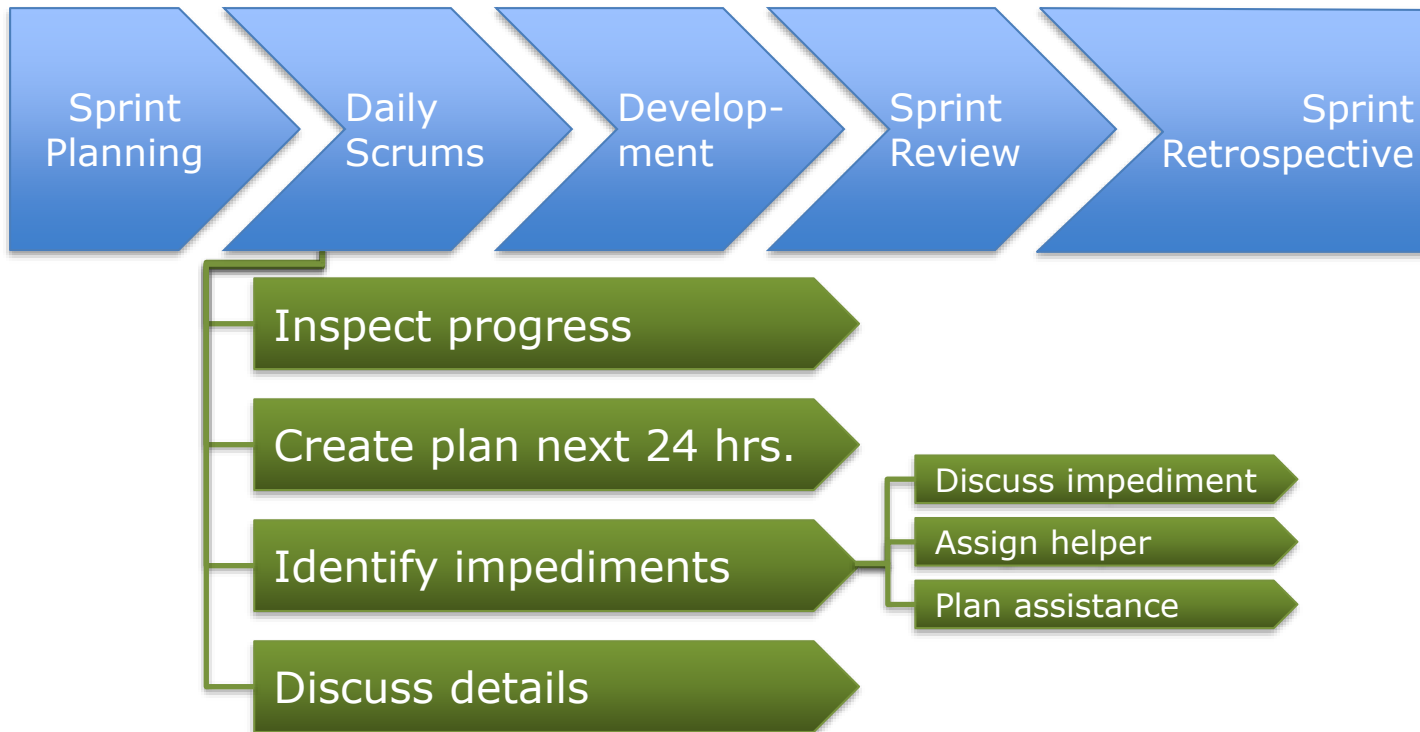
Activities may consist of **sub-activities**

And **subactivities** may ...

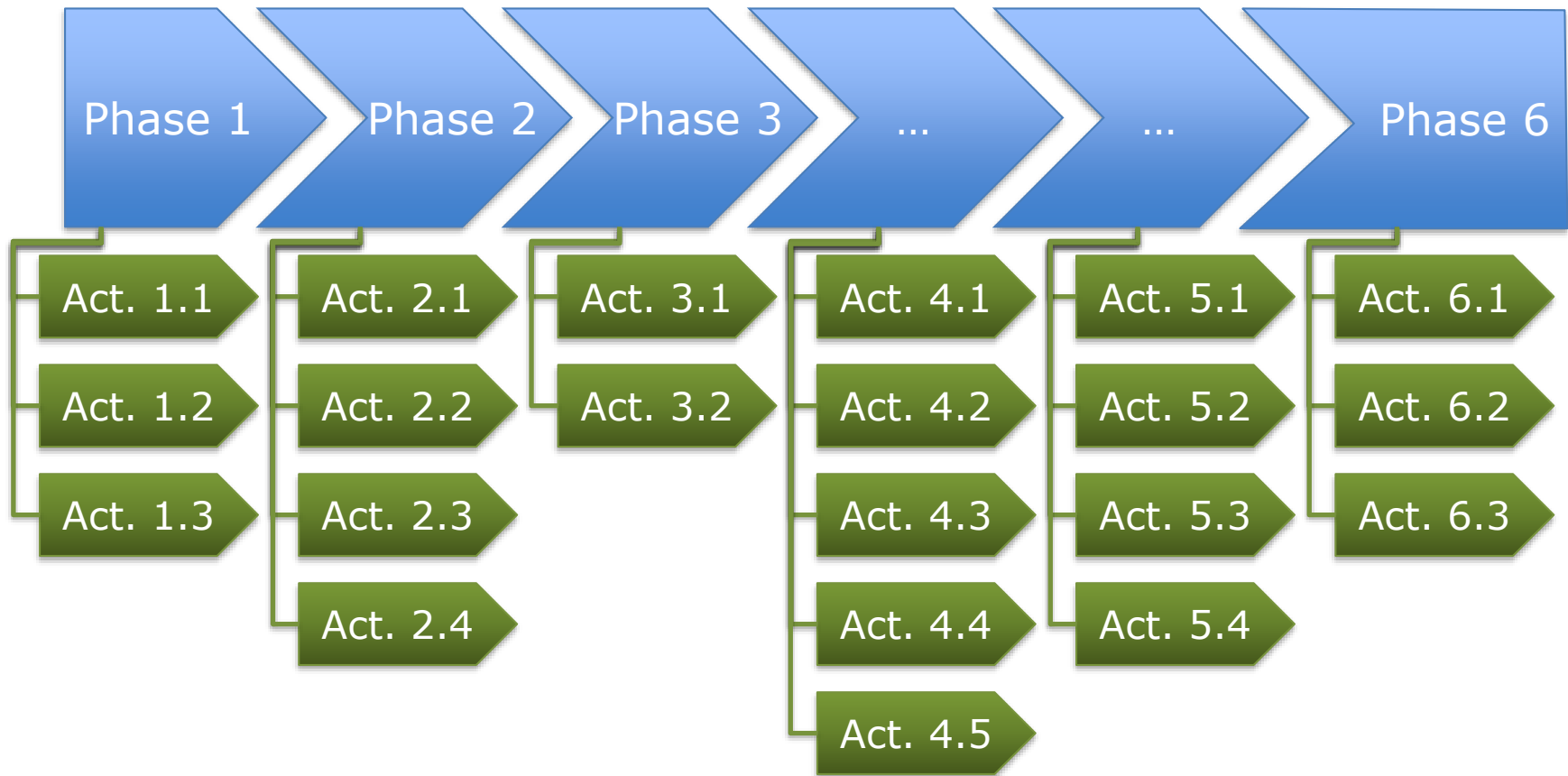
3 or 4 levels is sufficient



Activities of Scrum



Activities of methods



Main concepts in Method Engineering

- Method
- Activity
- Methodology
- Technique
- Notation
- Procedure
- Model
- Meta-model
- Diagram
- Tool
- MetaCASE



Technique

Definition

A technique is a **procedure**, possibly with a prescribed **notation**, to perform a development **activity**

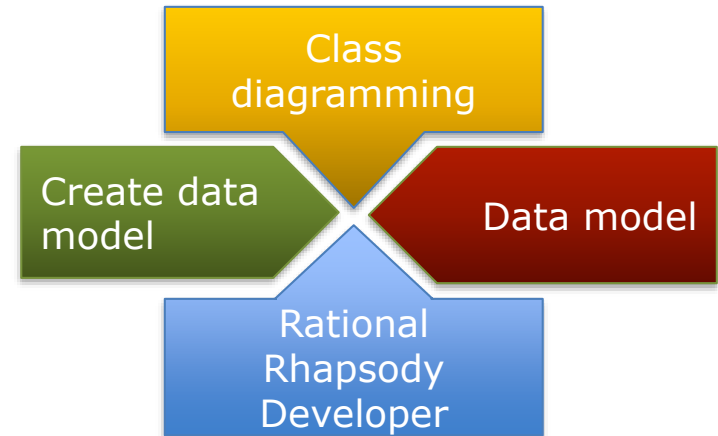
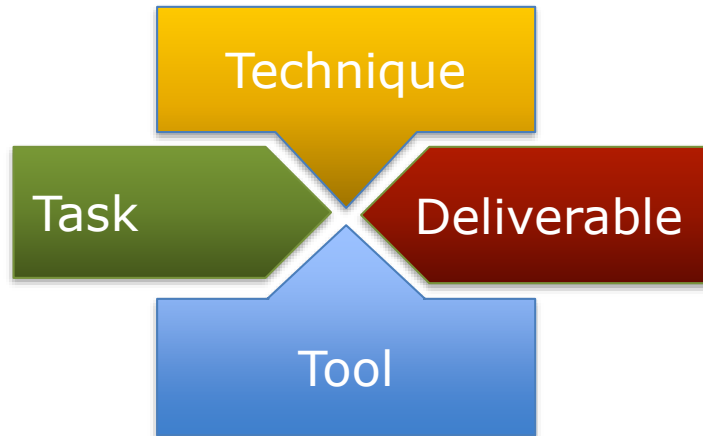
Examples

- data modelling with class diagrams
- interviewing with plain natural language
- requirements modelling with use-case diagrams
- software architecting with component diagrams



Basic activity

- A **basic activity** is a prescribed **task** resulting in a **deliverable** utilizing some **technique** and supported by a **tool**.



Notation and procedure

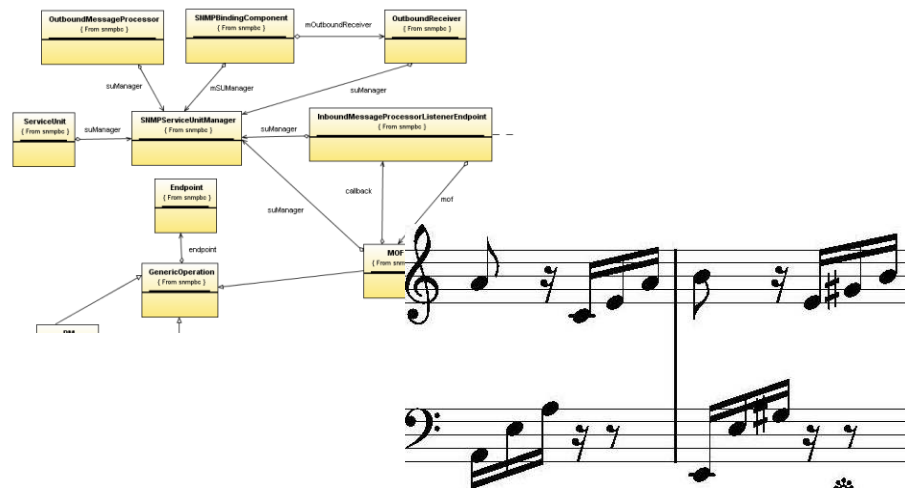
Definitions:

A **notation** is a system of symbols with a corresponding set of rules to construct artefacts used in communication.

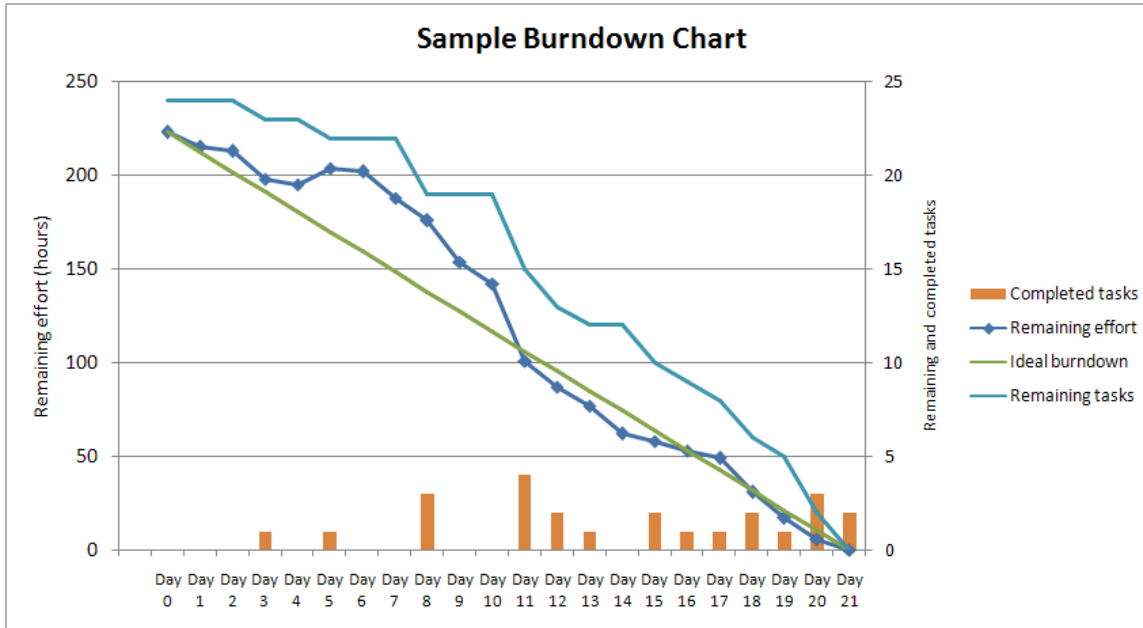
A **procedure** is a stepwise description of a process

Examples of notation:

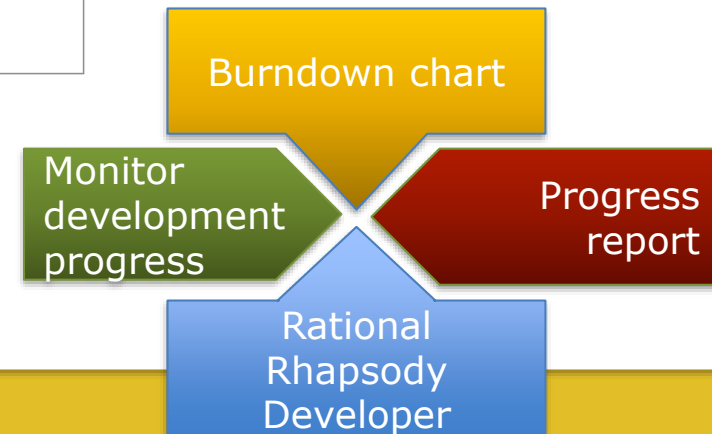
- Use case symbols
- English language
- Music score
- Predicate logic



Sprint monitoring



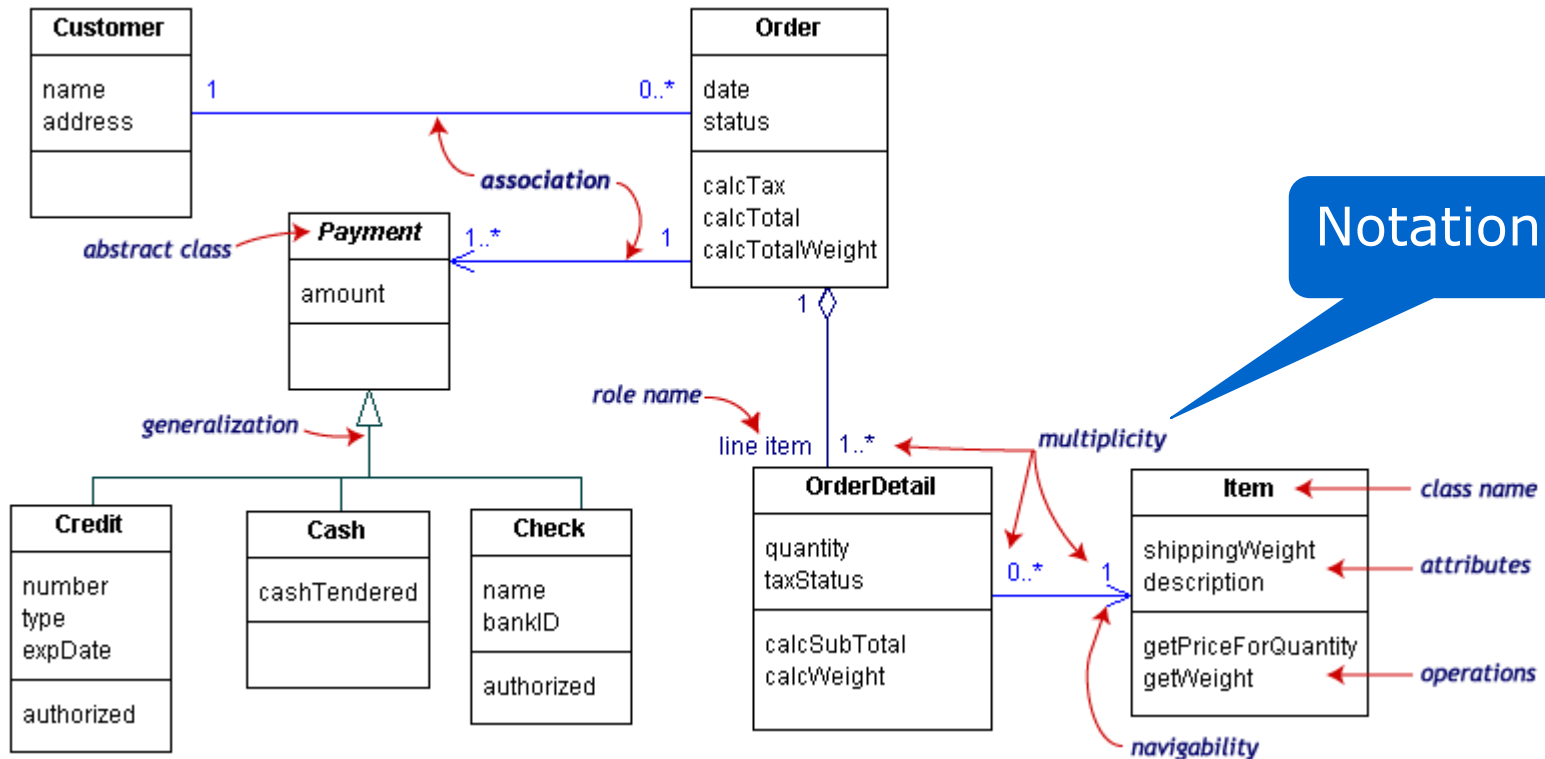
- Burndown chart is a technique for development progress monitoring



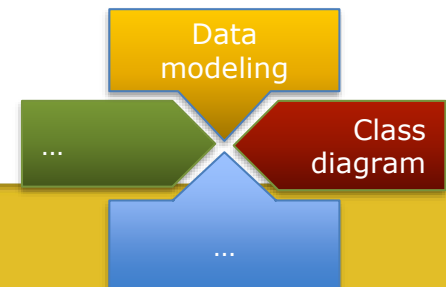
Data modeling

Technique

Notation



(Used for creating object-oriented software programs)

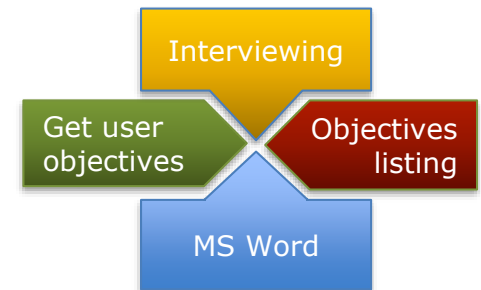


Interviewing

Interview process

1. Establish information needs.
2. Prepare for the interview.
3. Plan your questions.
4. Be careful about taking notes.
5. Know your subject.
6. Be specific.
7. Be accurate.
8. Look for color.
9. Do not talk too much.
10. Remember your sense of humor.

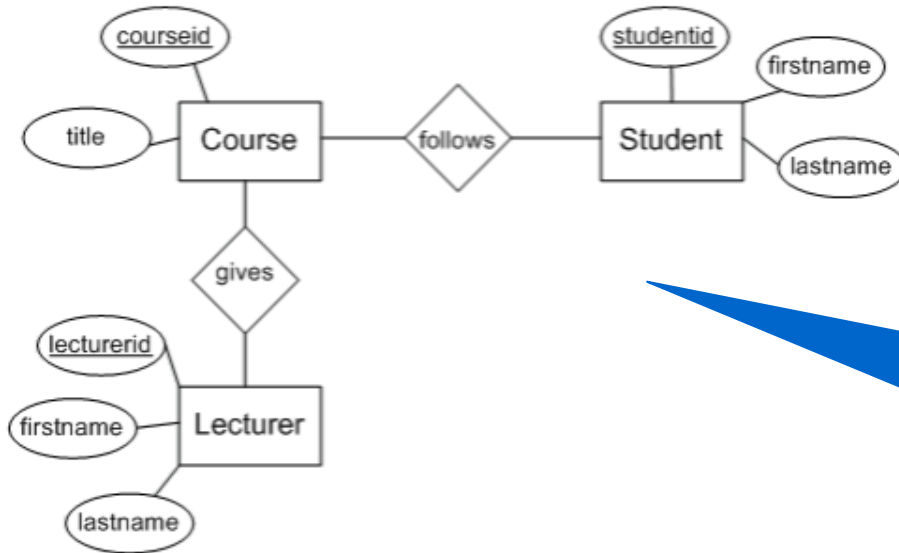
Procedure



Integrated Publishing (www.tpub.com)

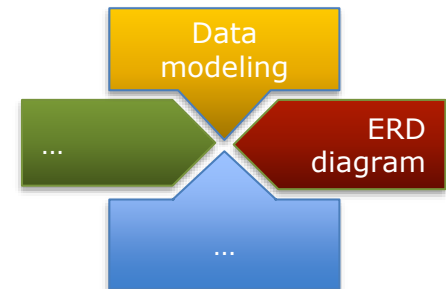


Entity-relationship diagram

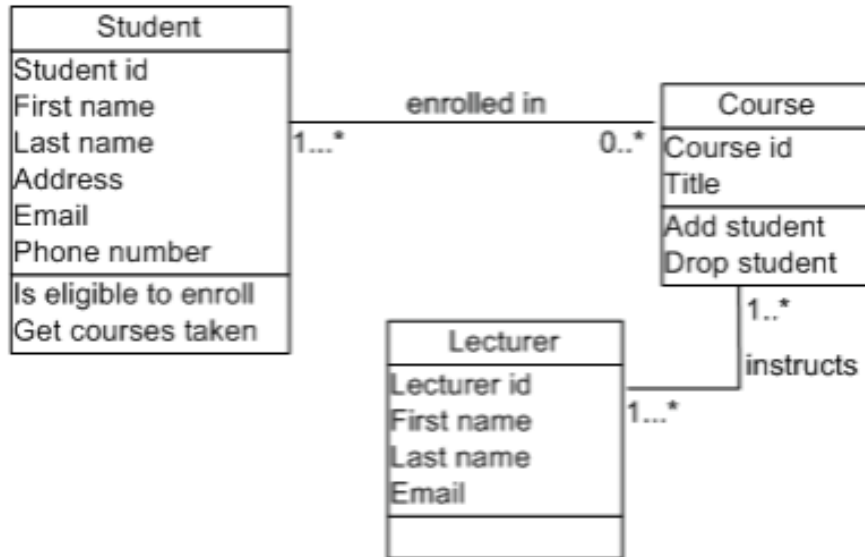


Model of the structure of a database, expressed in an ERD

- Used for modeling data for information systems
- Symbols:
 - entity
 - relationship
 - attribute

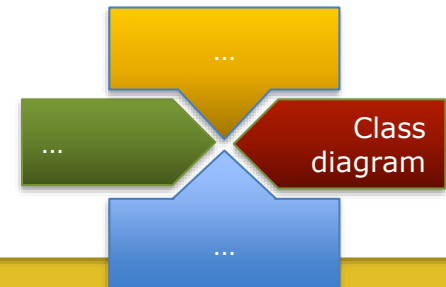


Class diagram

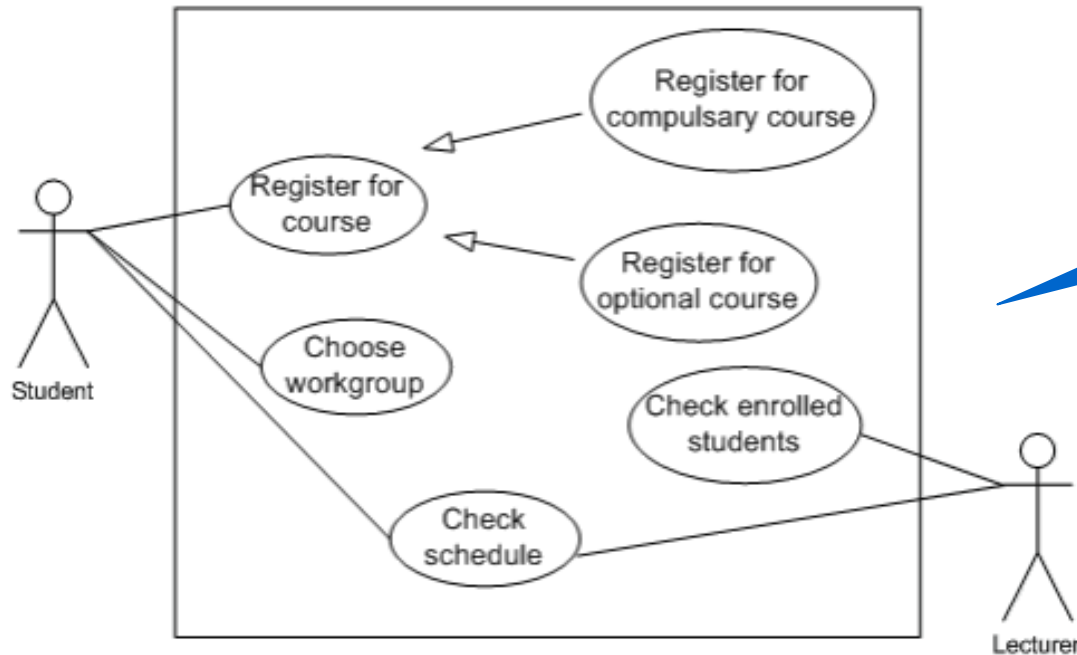


Model of the structure and interaction of an application, expressed in a class diagram

- Used for data modeling and for functions of classes

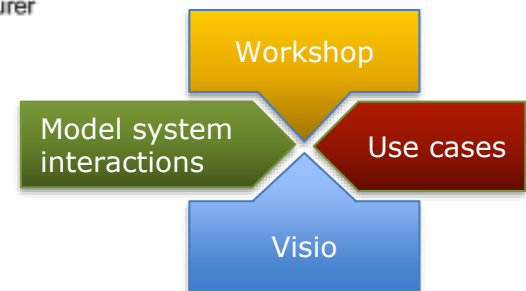


Use case diagram



Model of the functionality of an application, expressed in a use case diagram

- Used to represent functionality provided by a system in terms of actors and dependencies.



Classification of techniques

- Formality of the notation
 - natural language
 - structured graphics
 - mathematical
- Type of development activity, e.g.:
 - data modelling
 - process modelling
 - interaction design



Main concepts in Method Engineering

- Method
- Activity
- Methodology
- Model
- Meta-model
- Diagram
- Technique
- Tool
- Notation
- MetaCASE
- Procedure



Model

Definition:

A **model** is a representation that contains statements about the properties of an artifact (object) in a real or imagined world (universe of discourse).

Models can be expressed in **diagrams**.

Examples

- Data model (a description of database structure)
- Business model (a framework of the business logic of a firm)
- Maquette (scale model)



```

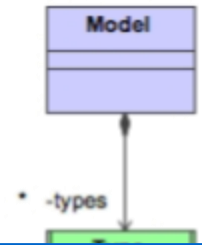
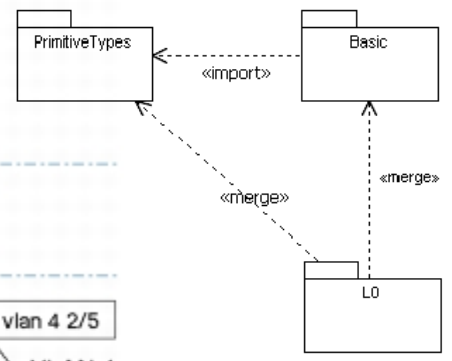
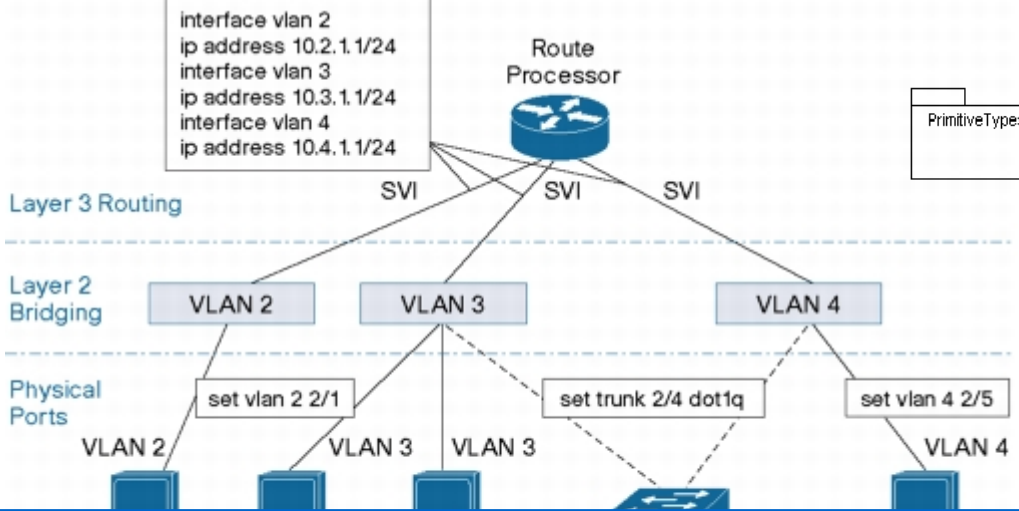
interface vlan 2
ip address 10.2.1.1/24
interface vlan 3
ip address 10.3.1.1/24
interface vlan 4
ip address 10.4.1.1/24

```

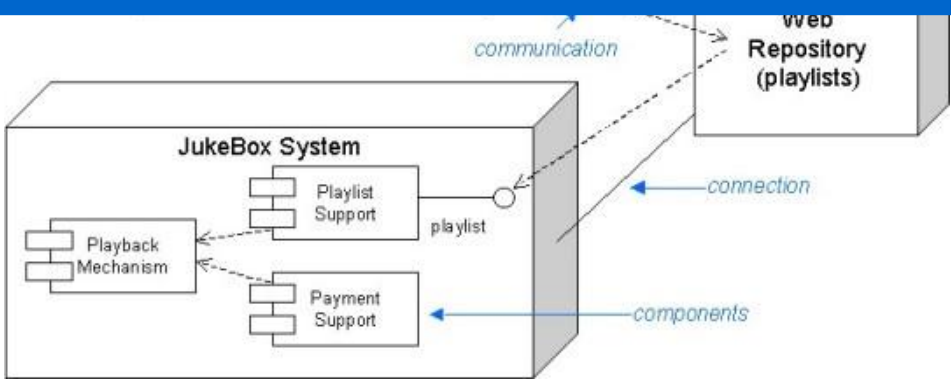
Layer 3 Routing

Layer 2 Bridging

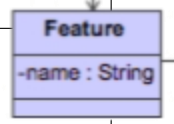
Physical Ports



How do we make science out of this?



ROW	FIRST NAME	LAST NAME	AGE
#1	Bob	McBob	42
#2	John	Johnson	24
#3	Steve	Smith	38



Meta-model

Definition

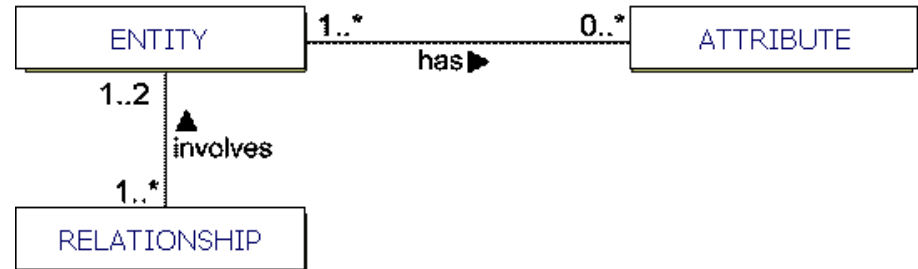
A **meta-model** is a model that consists of formal statements about models.

Strictly speaking, a meta-model is also a model but its universe of discourse is the set of models that are described by the meta-model.

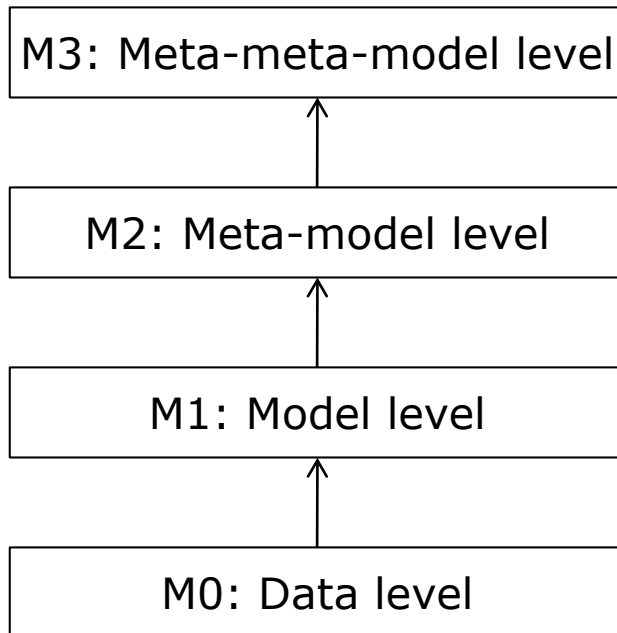
Meta-models can be expressed in **diagrams**.

Examples

- Meta-model of an entity-relationship diagram
- Meta-model of a class diagram



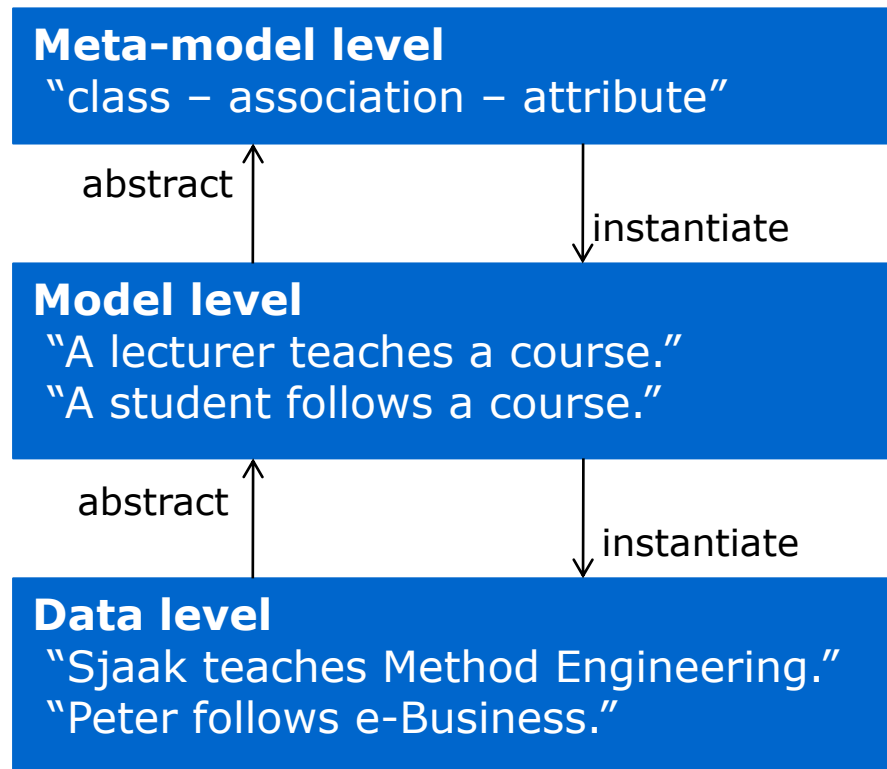
Four levels of reasoning



See also: Object Management Group OMG/MOF Meta Object Facility (MOF) Specification



Abstractions in statements about artifacts



Method Engineering

IS Analysis and Design



BREAK



Main concepts in Method Engineering

- Method
- Activity
- Methodology
- Technique
- Notation
- Procedure
- Model
- Meta-model
- Diagram
- Tool
- MetaCASE



Tool

Definition

A tool is a possibly **automated means** to support (a part of) the development process

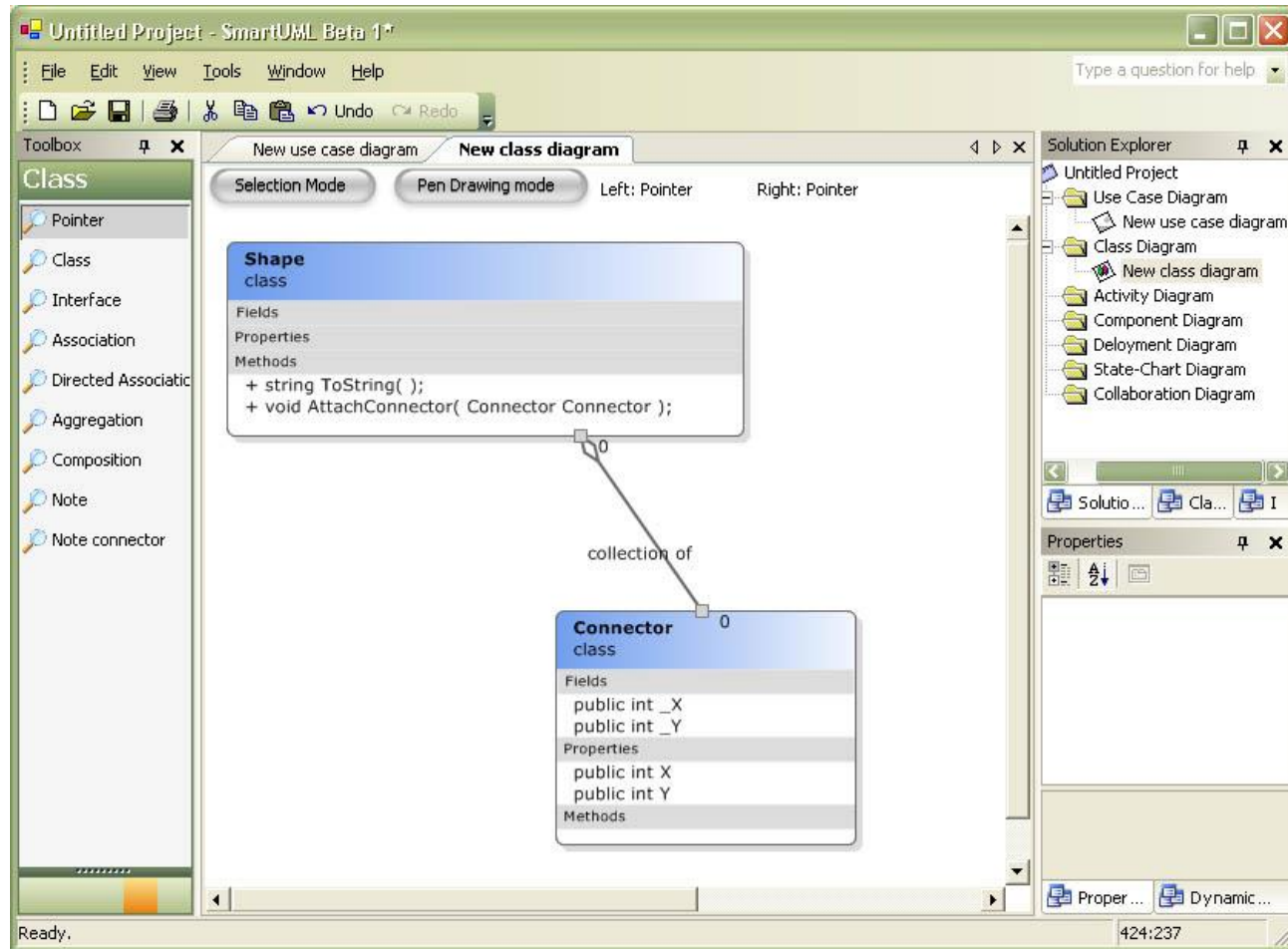
- Support a collection of different notations
- Assistance to whole development life-cycle
 - Computer Aided Software Engineering tool (CASE)
 - Integrated Development Environment (IDE)
 - Software Development Kit (SDK)

Examples

Rational Rose, Eclipse, Visio, BeWise, Flower, and many more

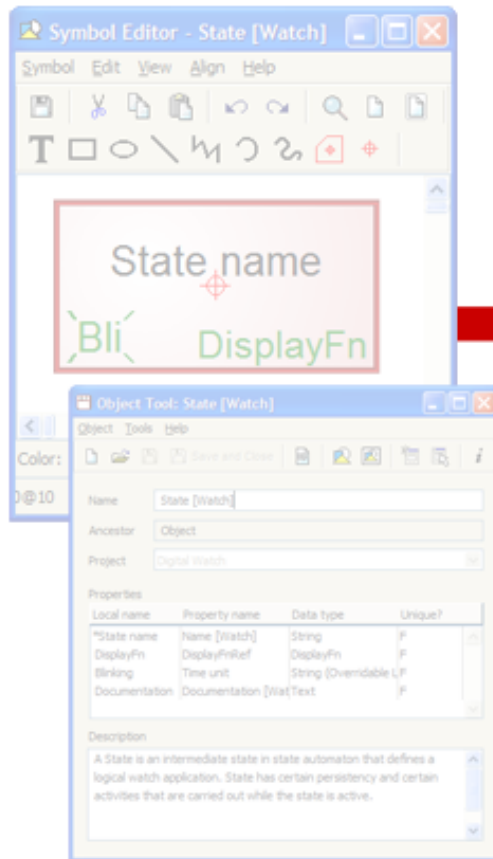


SmartUML

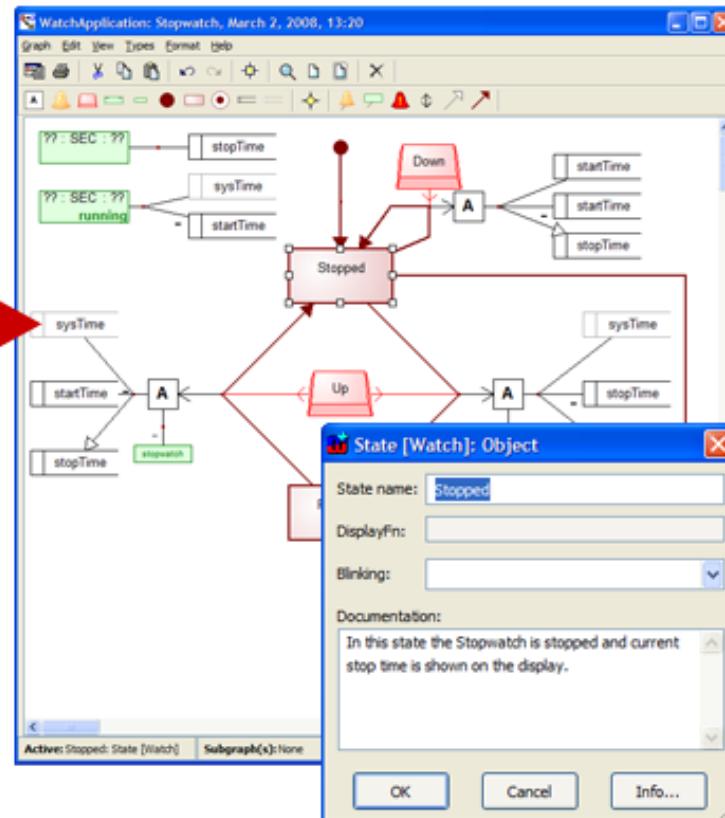


MetaCASE

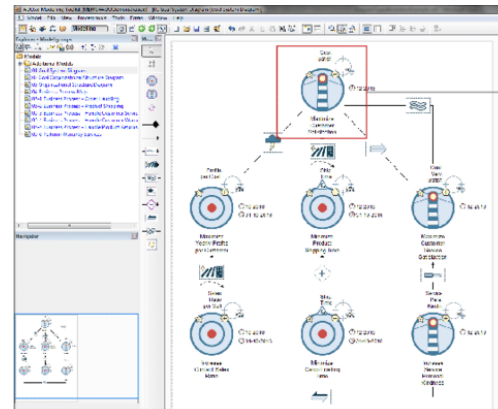
Design your modeling language



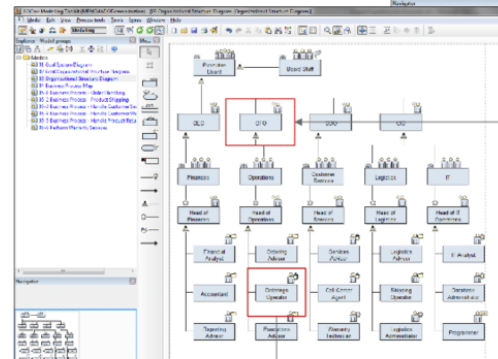
Use your modeling language



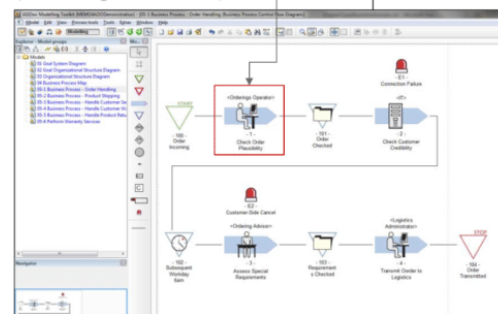
ADOxx (MEMO4ADO example)



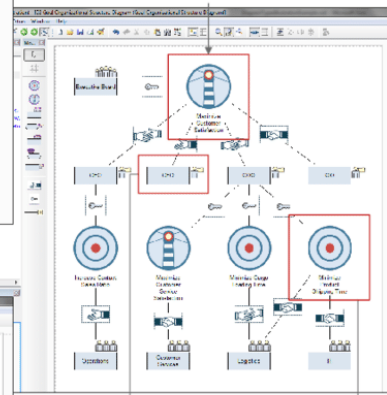
Organisational Structure Diagram
(MEMO OrgML: Structures)



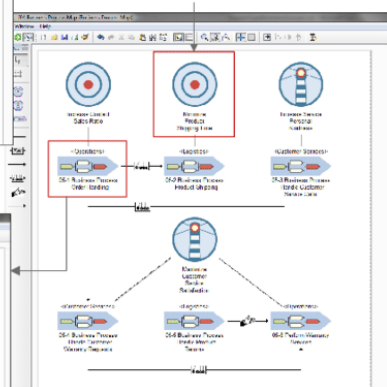
Business Process Control Flow Diagram
(MEMO OrgML: Processes)



Goal-Organisational Structure Diagram
(MEMO GoalML, OrgML: Structures)



Business Process Map
(MEMO OrgML: Processes, GoalML)



Main concepts in Method Engineering

- Method
- Activity
- Methodology
- Technique
- Notation
- Procedure
- Model
- Meta-model
- Diagram
- Tool
- MetaCASE



Methodology

Definition

The methodology of systems development is the systematic description, explanation and evaluation of all aspects of methodical systems development.

or: *The Science of Methods*

Often methodology is erroneously used where method is meant.

Methodological schools

- Object-oriented (OO)
- Socio-technical approaches
- Agile movement

So: Scrum and XP are methods in the methodological school of the Agile movement, UML and RUP in the OOP school.



Discussion

- What are your experiences with applying
 - methods,
 - techniques, or
 - tools?
- Do they work in all situations?
 - When do they?
 - When don't they?



Observations

- Different IT systems
 - Information systems
 - Web-applications, mobile apps
 - Workflow, business intelligence
 - Product software, commercial software
 - Real-time systems, embedded software
- Different platforms
 - Relational DBMS
 - Java, J2EE
 - MS Access, .NET,
 - LAMP (Linux, Apache, MySQL, PHP)
- Different domains
 - Financial, insurance, banking
 - Educational
 - Government, public sector
 - Transport, logistics
 - IT services
 - Manufacturing, production
 - Service industry
 - Energy, oil, utilities



Motivation

- There is not one method that fits all
 - Tolvanen (2000): 65% of organizations which use methods develop them in-house and 85% of organizations adapt existing methods in-house
- Understand how to extend or adapt methods and techniques to
 - the current project
 - future project
 - the changes in the organisation
 - the platform or application



Method Engineering

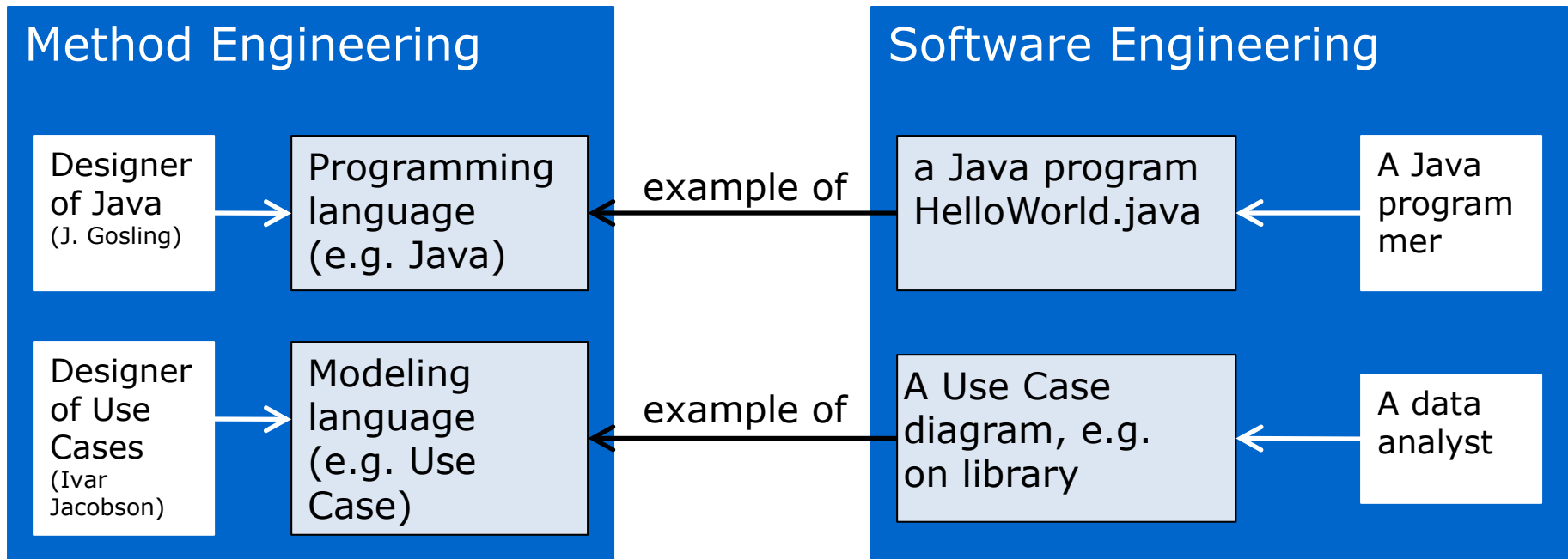
Definition:

Method engineering is the engineering discipline to design, construct, and adapt methods, techniques and tools for systems development.

c.f. Software Engineering is the engineering discipline for software



Comparison Method Engineering and Software Engineering



Situational method

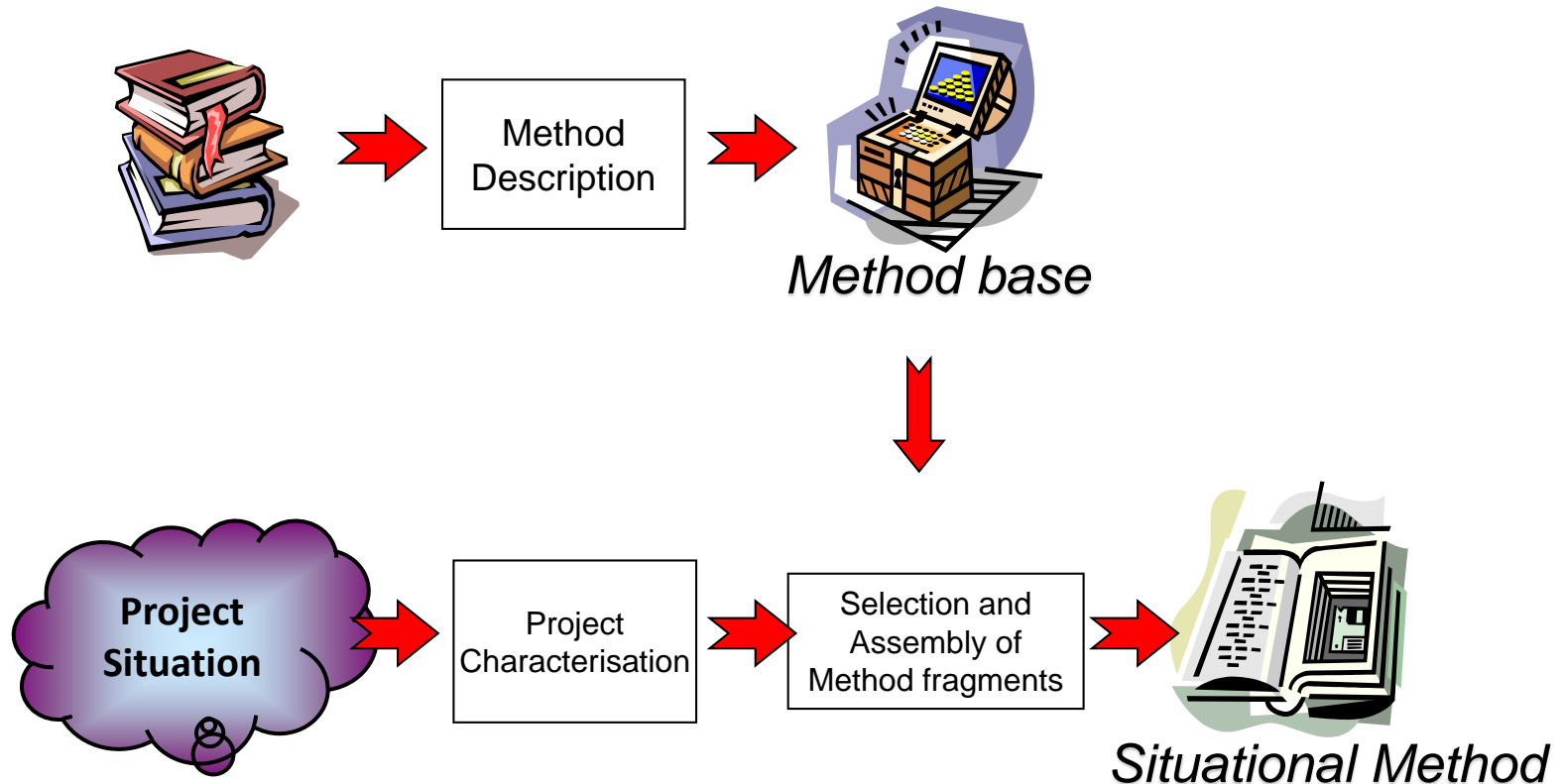
Definition

A **situational method** is a method tuned to a specific situation.

Situational Method Engineering is the area of method engineering focusing on situational methods.



Situational method engineering



Meta-modeling

Problem

How to **describe** and **store methods** in the method base, in order to easily reuse the **method fragments** in the **method engineering** process?

Solution

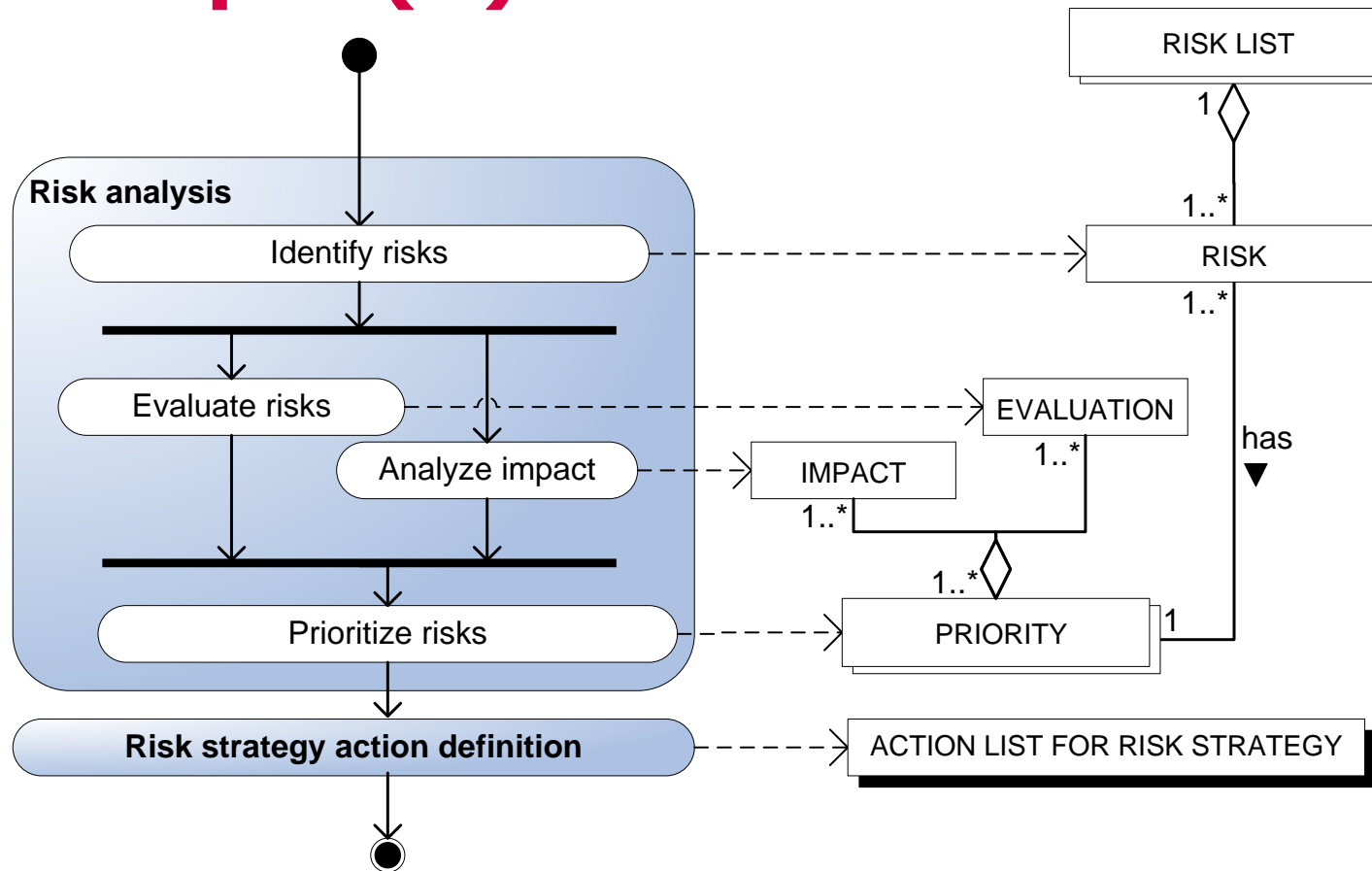
Meta-modeling technique to describe both **process** and **deliverable** part of the method
→ Process-Deliverable Diagram (PDD)

Advantages

Clear, compact, consistent with UML standards.



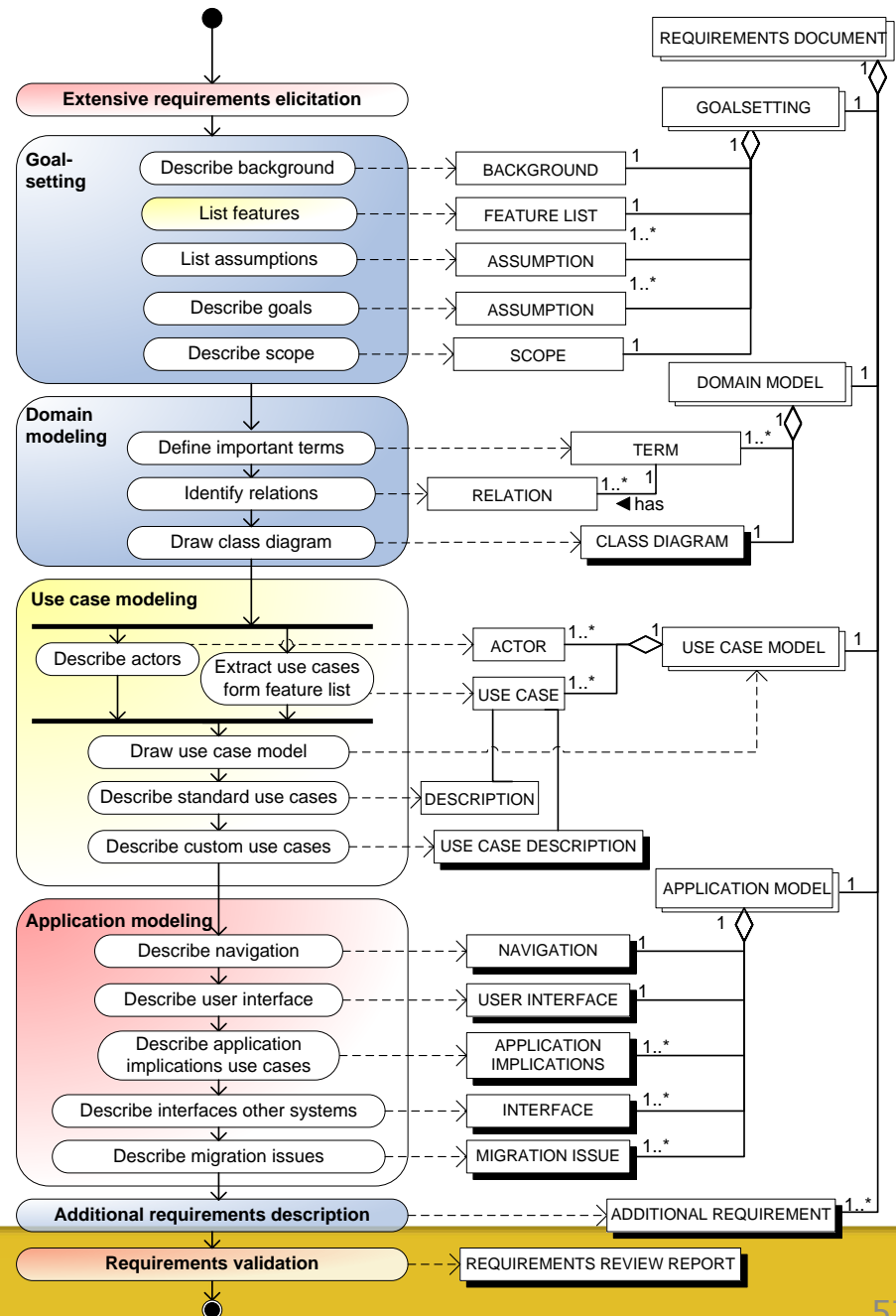
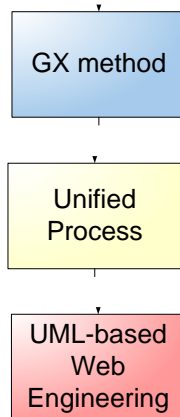
Example (1)



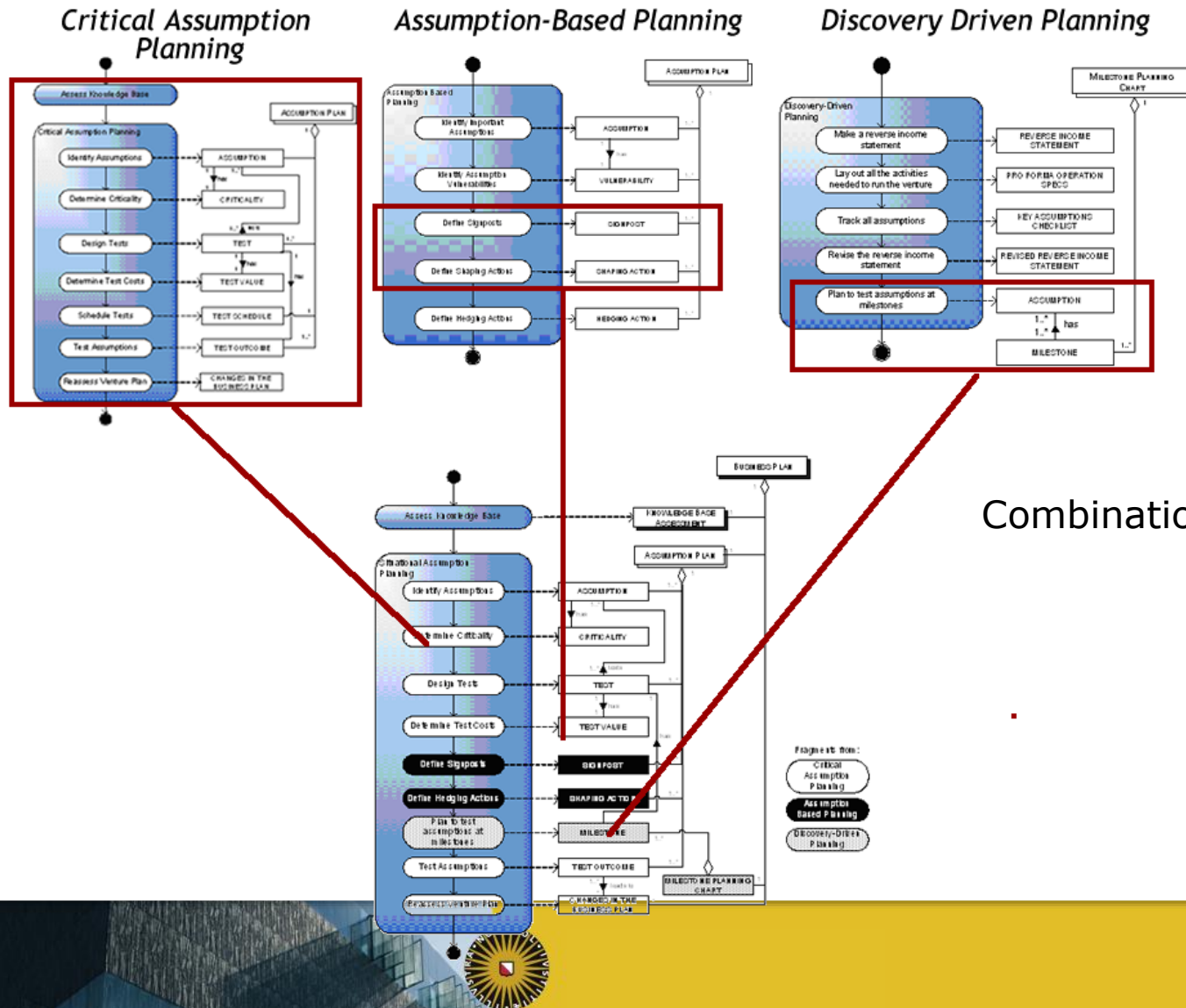
Process-deliverable diagram example:
Risk workflow in UML-Based Web Engineering (Koch, 2000)

Example (2)

Combination of methods



Example (3)



Combination of methods

... and what do you have to ask?

Learning objectives:

- Providing of insight and skills into the **systematic description, explanation and evaluation** of all aspects of the methodology of ICT systems development
- Being able to **use and adapt** methods, techniques and tools for the **situation** of the project or system
- Contributing to the **body of knowledge** on method engineering
- Being able to review **scientific literature**, analyzing a paper in depth, and create a **videoclip** of the results



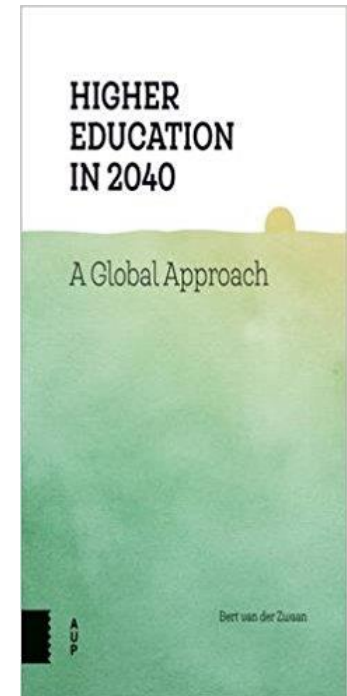
Agenda

- Introduction to method engineering
- Practical issues
- Assignment week 7 & 8



Blended learning

- Digital learning will strongly increase in the coming 25 years
- Massive Open Online Courses (MOOCs), Small Private Online Courses (SPOCs), and *blended learning*
- Modularization (customized learning) vs. completing an entire curriculum
- Foreseeable changes on university campi: more *informal* learning and less buildings for formal education
- Zwaan, B. van der (2017). Higher Education in 2040: A Global Approach. Amsterdam: Amsterdam University Press.



Educational tools

- **LectureNet**: high-quality recording of regular lectures
- **Scalable Learning**: adding interactive e-lectures to the course
- **Revisely**: online submission of practical exercises, peer-reviewing, and peer-grading
- **Remindo**: digital exams
- **Mediasite**: recording of final presentations
- **Blackboard**: hosting course content



Course website: Blackboard

- 2018-2019 3-GSNS
- Method engineering (INFOME)
- Course Information
- Staff Information
- Announcements
- Schedule
- Assignments
- Literature
- Practice Exams
- Grading Scheme
- Discussion Board
- My Grades
- Course Management
 - Control Panel
 - Content Collection

Course Information

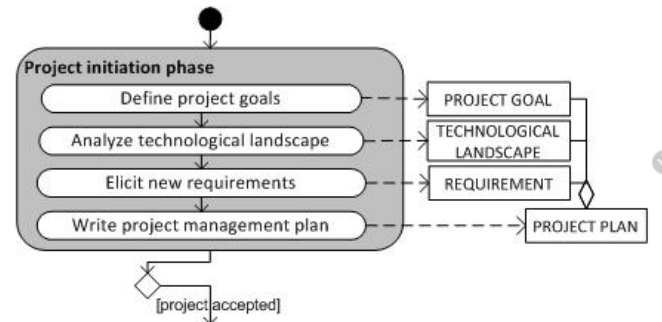
Build Content ▾ Assessments ▾ Tools ▾



Welcome!

This is the Blackboard site of the course Method Engineering, as being given in the Master Program of Business Informatics at Utrecht University. Check out the schedule, literature, and assignments in the menu at the left. We hope you like the course! In Method Engineering, regular lectures are blended with online lectures. By means of this approach, you will be able to learn the theories through a series of videos complemented with discussion sessions during the regular lecture hours.

Method Engineering is defined as the engineering discipline to design, construct, and adapt methods, techniques and tools for systems development. The primary focus of the course is to apply meta-modeling for the assembly and evaluation of methods; the adaption of concepts and rules for techniques; and the derivation of support tools for information systems development.



Example of a meta-model of a method depicted by means of a Process-Deliverable diagram (PDD).



Please remember

- Please check Blackboard regularly for news regarding the course. Announcements regarding the course will also be sent by e-mail.
Note: your UU e-mail address will be used!
- Slight changes in the course schedule might occur, make sure to regularly double-check the schedule in Blackboard and in the Google Calendar of the course



LectureNet ME library

- Most of the regular lectures of this year will be recorded using LectureNet.
- The link to the LectureNet Method Engineering Library can be found on Blackboard.
- Rewatch lectures at home; convenient for exam preparation and while working on assignments.



Flipping the classroom (1/2)

- We are using the concept of a *flipped classroom*:
 - Watch recorded e-lectures (a collection of knowledge clips) at home.
 - Answer all quiz questions while watching the e-lectures.
 - In-class discussion sessions, where the experience of watching the e-lectures and the quiz results are discussed.
- Aims: increased student activation and learning effect.
- The lectures [Meta-Data Modeling](#), [Meta-Process Modeling](#), [Formalization of Methods](#), and [Process-Deliverable Diagrams & Modeling Patterns](#) have been transformed into e-lectures.



Flipping the classroom (2/2)

- Note that your quiz results will **not** be graded.
- However: watching and preparing the e-lectures is **mandatory** for the sake of useful in-class discussion sessions and to prepare yourself for the exams.



How to watch the e-lectures?

- Go to [Scalable Learning](#). Select 'Use your School/University Account', select '**Utrecht University**' and login with your **SolisID**.
- Go to 'My Courses' → 'Join a new course (student)'.
- Use **MWXAZ-10117** as the enrollment key.
- **Prepare the Meta-Data Modeling** lecture first. The other e-lectures will be opened later.
- **Answer all questions.** Optionally, add notes or even pose online questions while watching.
- Deadlines for watching the e-lectures and making the quiz questions can be found in the schedule on Blackboard.
- Read the student manuals in the Help Centre.



Course outline: Assignments

- Reading of scientific literature
 - Can be downloaded from [Blackboard](#)
- Writing a paper concerning the ME **topic** you have selected, i.e., a **technique** or a **method**
 - Your selected topic should be within one of the themes provided under assignment A on Blackboard
- Recording of a videoclip in which the results of your final paper are presented and conducting peer reviews



Attendance

- Attendance to lectures and workshops is strongly advised, but **not compulsory**
- **Compulsary:**
 - Watching the e-lectures before the discussion sessions
 - Answering the quiz questions in the e-lectures
 - Conducting the peer reviews
- **Compulsary attendance:**
 - Excursions to PWC and Deloitte
 - Final videoclip presentation sessions



Workshop groups

Group 1

Kleopatra Chasioti and Melchior Keijdens

Group 2

Philippe van der Voorn and Sietse Overbeek

Students with *last names* starting with letters **A up to and including M** are in group **1**.

Students with *last names* starting with letters **N up to and including Z** are in group **2**.



Agenda

- Introduction to method engineering
- Practical issues
- Assignment week 7 & 8



Assignment A week 7 & 8:

Topic choice and description (1/2)

- Enter the information on your selected technique or method in [this](#) Google Sheet and provide your student information.
- You need to select a topic (and a paper) that is **unique** in the aforementioned Google Sheet (i.e., it should not have been selected by a fellow student already).
- Write the description of your chosen technique or method
 - Introduction (global overview, creators technique or method)
 - Example (your own example)
 - Related literature (origins, positioning, application)
 - References (use APA style)



Assignment A week 7 & 8:

Topic choice and description (2/2)

- Summarize, cite, recap, mention, and review other work.
- But. Always. Make. Proper. References.



How to choose a paper concerning your technique or method? (1/2)

- Under 'Assignments' on Blackboard, links can be found to theme-related websites as a source of inspiration.
- Find search engines for conference papers and journals at the [Information and Computing Sciences page](#) of the UU Library.
 - Please also look at the [online access page](#) of the UU Library to discover possible ways to get easy access to the digital resources offered by the UU Library.
 - Good search engines for literature are [SpringerLink](#), [ScienceDirect](#), [ACM Digital Library](#), and the [DBLP Library](#) of Computing sciences.



How to choose a paper concerning your technique or method? (2/2)

- By using [Citeseer](#), [Google Scholar](#) and [Microsoft Academic Search](#) you can search for more literature and also do a citation analysis to see how the different articles of your subject relate to each other.
- If you cannot find a suitable paper, choose another established conference or workshop - as long as the paper is peer-reviewed!
- Choose a paper which is about your technique or method.
- **Notify a student assistant in your group to approve your paper**



Requirements for the selected paper

1. Published between ~2005 and 2019
2. Clearly contains the steps of the technique or method that you selected (or these should be derivable)
3. Contains at least one **example** of the application of the technique or method



Submitting your assignment A

- Deadline is on Blackboard
- Submit your **pdf** file to [Revisely](#)
- Plagiarism check
- Failing to submit on time causes a 1.0 pt deduction per day (starting at 22 Feb. 2019 after 18h00)



Peer reviewing and peer grading (1/2)

- Assignments **B** and **D**: peer reviewing and peer grading
- Assignment **F**: peer reviewing
- Rationale: learn from each other's work and gain experience in providing feedback and grading.
- After submission of a deliverable in Revisely, *three* reviewers will be assigned to each submission.
- You need to provide textual feedback (i.e., remarks and suggestions for improvement) and provide scores for the individual grading criteria.



Peer reviewing and peer grading (2/2)

- Read the grading criteria document that is attached to the assignment in Revisely carefully to understand the specifics of where to pay attention to when reviewing and grading.
- Failing to finalize your peer reviews on time or failing to deliver a serious peer review will result in the deduction of one point per day from your own final paper grade.
- The student assistants will perform a meta-review of the results
- The lecturers will perform a meta-meta-review



How to start?

- Read this week's paper (Brinkkemper, 1996).
- Next week Thursday 11h00 the workshop starts.
- Monday: **Lecture**: Further explanation on the assignment
 - How to find literature, how to make proper references, how to create examples, etc.
- Tuesday: **In-class discussion** of the Meta-Data Modeling e-lecture.
- Thursday: **Workshop**: working on method engineering assignment
 - » Read the paper, find related literature, start writing the topic description.



QUESTIONS?



References

- Brinkkemper, S. (1996). Method engineering: engineering of information systems development methods and tools. *Information and Software Technology* 38(4), 275-280.
- Inge van de Weerd, Sjaak Brinkkemper, Johan Versendaal (2010) Incremental method evolution in global software product management: A retrospective case study. *Information & Software Technology* 52(7): 720-732
- Jolita Ralyté, Sjaak Brinkkemper, Brian Henderson-Sellers (2007) Situational Method Engineering: Fundamentals and Experiences, Proceedings of the IFIP WG 8.1 Working Conference, Geneva, Switzerland. IFIP 244, Springer 2007, ISBN 978-0-387-73946-5
- Jeusfeld, M.A., M. Jarke, & Mylopoulos, J. (2009). *Metamodeling for Method Engineering*. Cambridge, USA: The MIT Press.
- Ralyté, J., Deneckère, R., & Rolland, C. (2003). Towards a generic model for situational method engineering. *Lecture Notes in Computer Science* 2681, 95-110.

