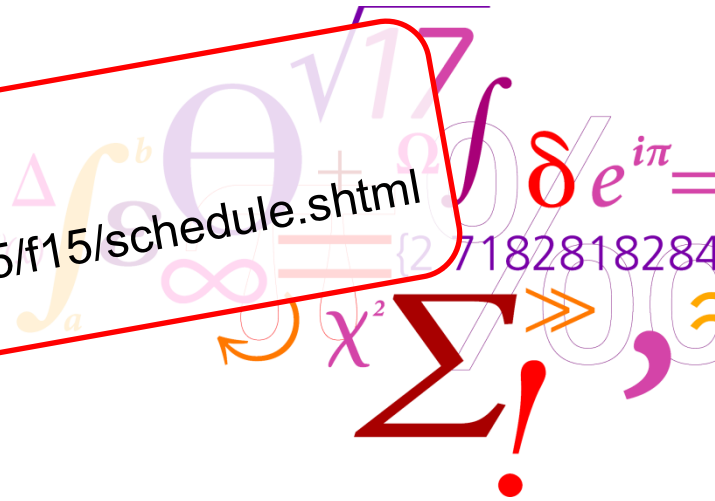
Advanced Topics in Software Engineering (02265)

Ekkart Kindler

DTU Informatics
Department of Informatics and Mathematical Modeling

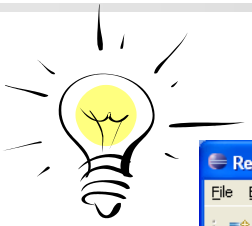
All presentations available at:
http://www2.compute.dtu.dk/courses/02265/f15/schedule.shtml

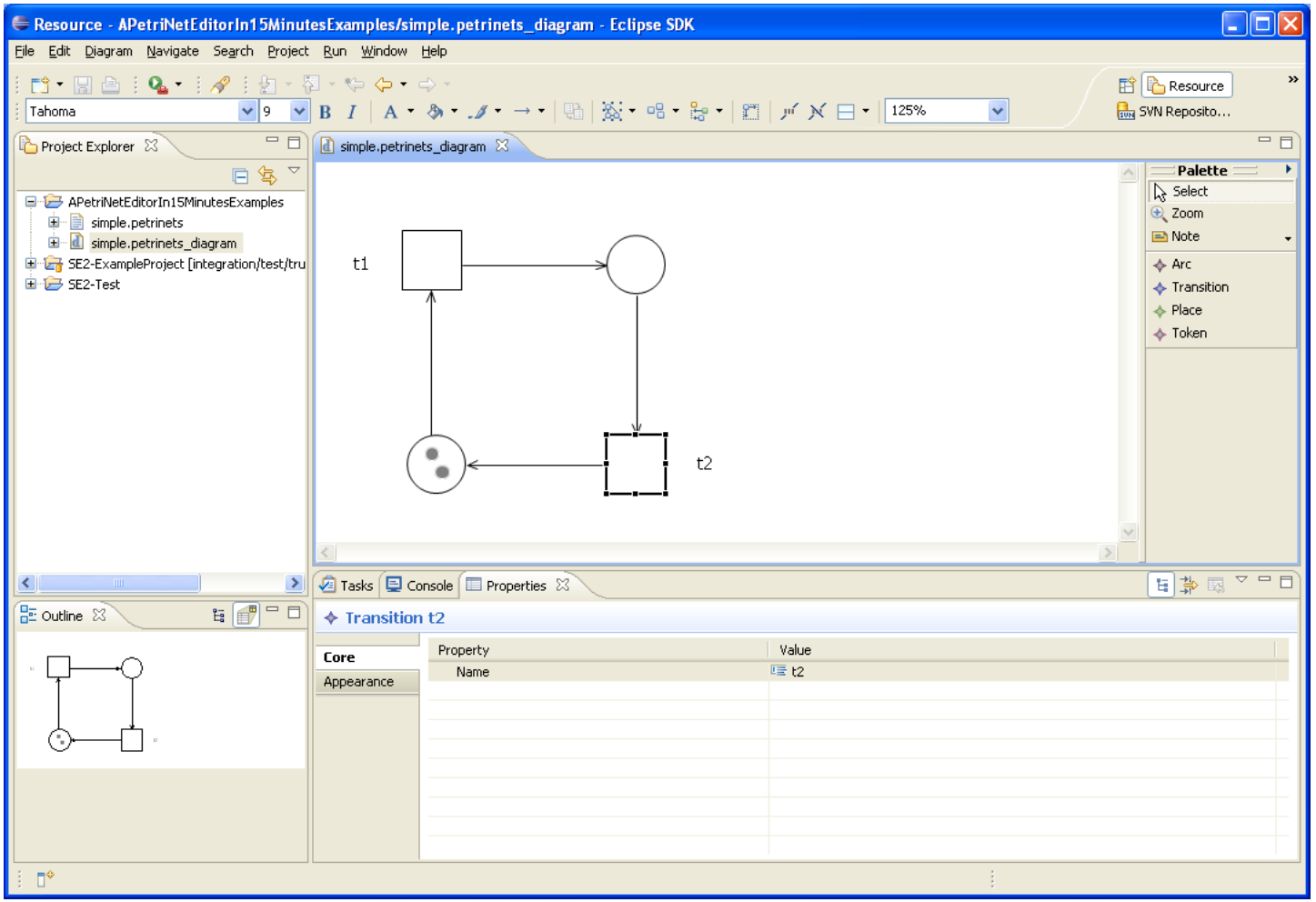


I. Introduction and Motivation

1. Vision

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

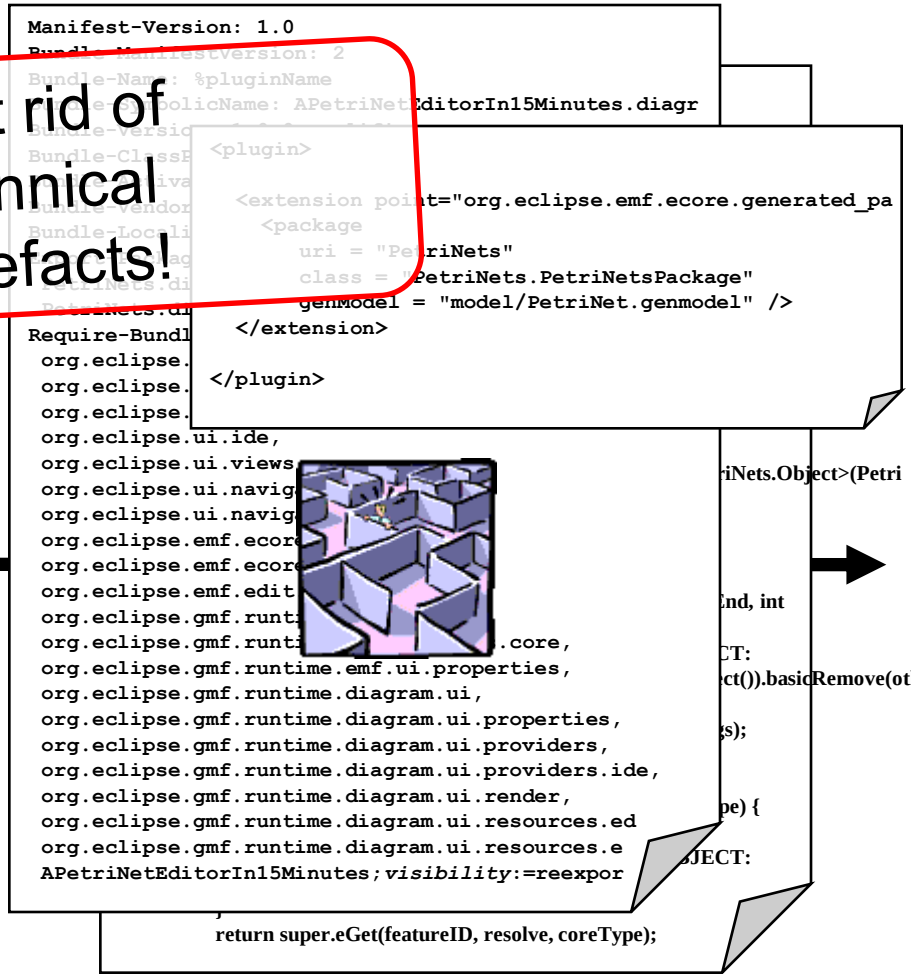


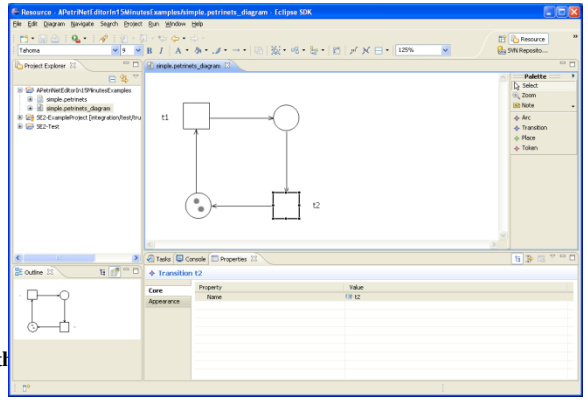


Today

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

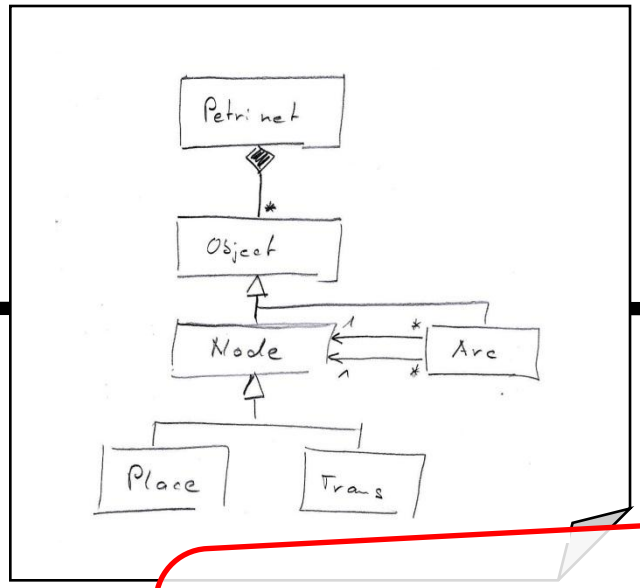
Get rid of technical artefacts!

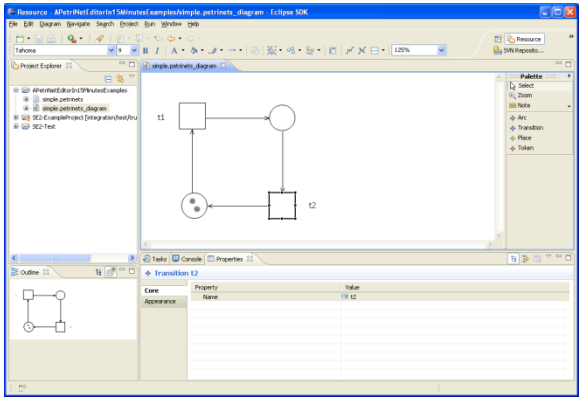




Vision

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler



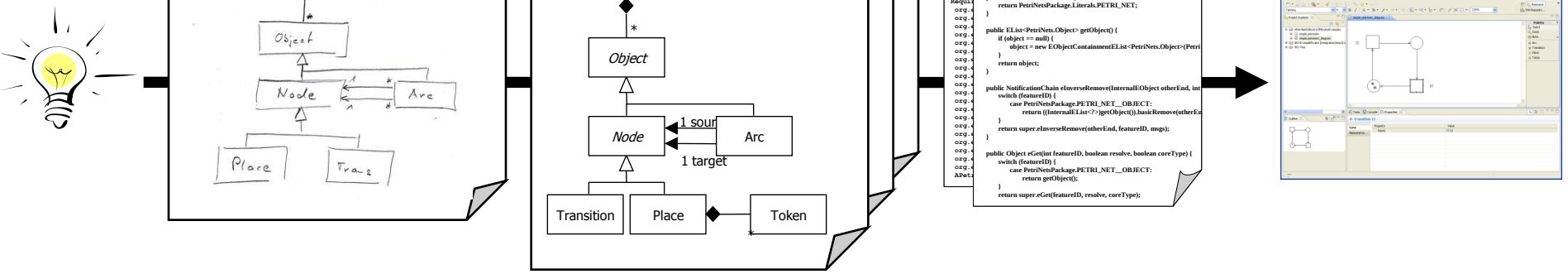


Exploit conceptual artefacts instead (and generate software from them).

Vision

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

There are tools that partially support this idea already today (e.g. Eclipse and EMF as used in earlier editions of SE2 (2013 or earlier).



Analysis

Design

Implementation

~~Coding~~

Code is generated

”Buzzwords”

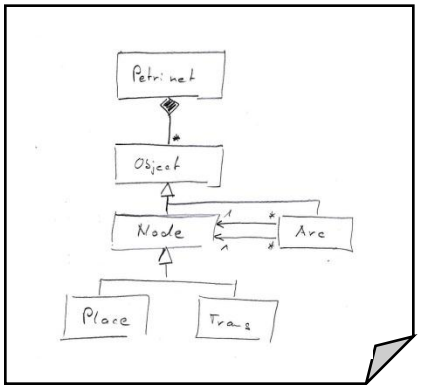
- Model Driven Architecture® (MDA®)
  OMG™ software development approach for separating business logic from platform specific details
  - using models
  - automatic generators (for code and other models)

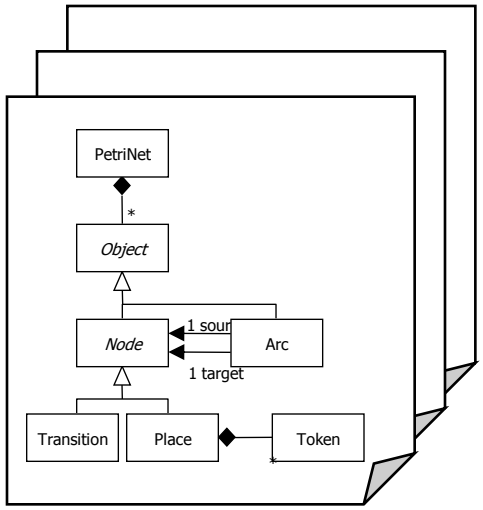
- Model-based Software Engineering (MBSE)
  General term for making “better” use of models for easing the software development

Ultimately: Getting rid of programming resp. technical artefacts.

So!? What's new?

- But, that's UML! We know UML already!
- Didn't we do all that already?
  (e.g. in SE1 and SE2, etc.)
- Why do we need another course on that?





2. Goals of this course

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

- Acquiring a bit more experience!

- Learn what is behind the scenes!
  - Understand concepts and technology
  - Apply (some of) them
  - Experiment and evaluate technology

- Contribute to MBSE?
  - Extend (and develop new) technology
  - Combine them in a new way
  - Formalize and analyze them

Understand and work on the **meta-level**

See http://en.wikipedia.org/wiki/Meta for the meaning and co-notations of the prefix “meta”. We will come back to that later.

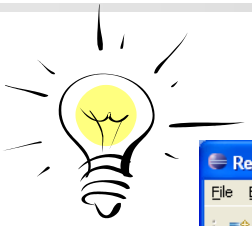
Overview

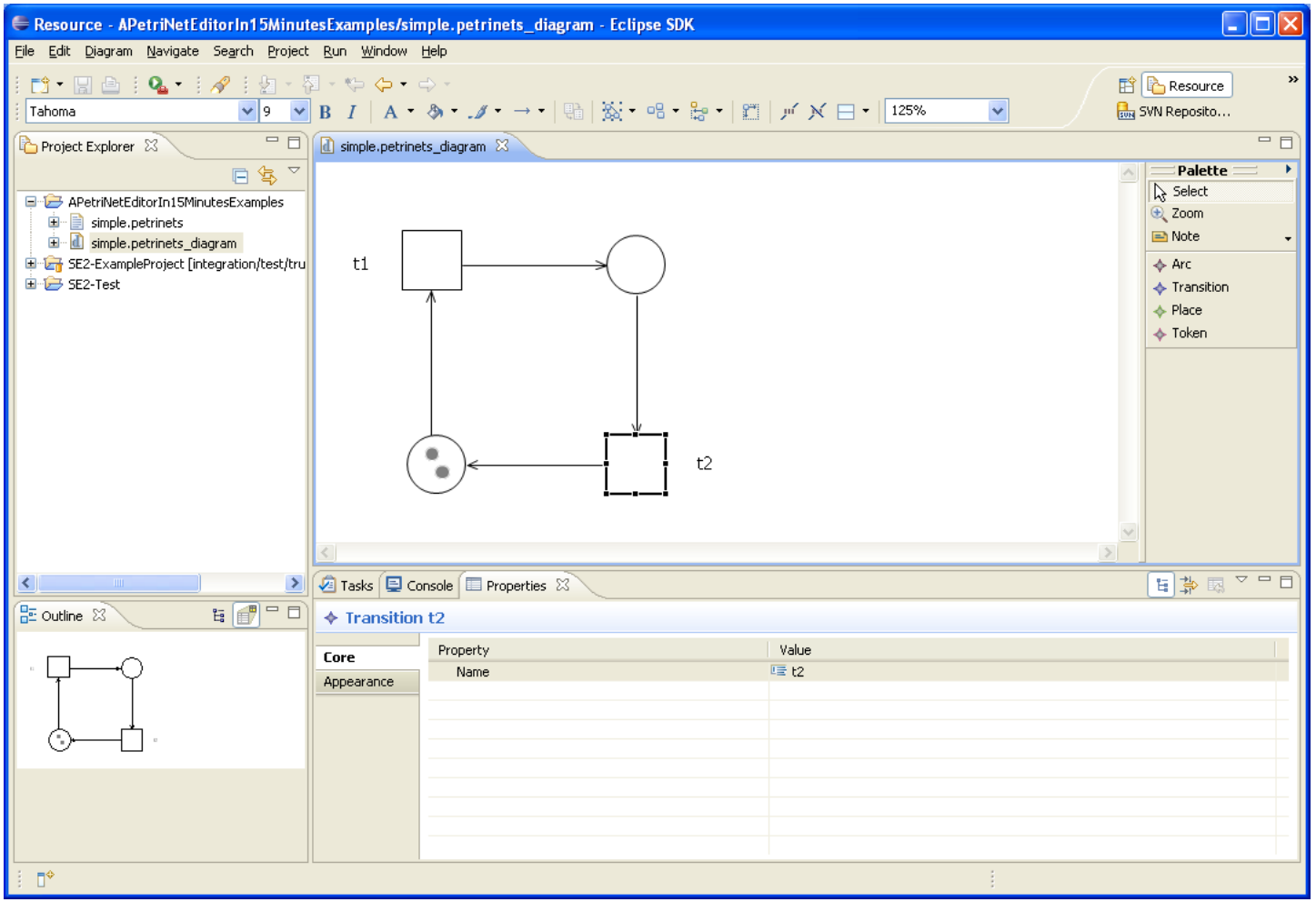
- Modelling notations
  - formalizing, implementing, exchanging them
  - integrating new ones
  - relating different models
  - model notations vs. / w. languages
- Transforming models
  - into other models
  - into text (generate code)
- Synchronizing and merging models
- Analysing and verifying models

- Working with models
- Behaviour models

3. Example (a Petri net editor)

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

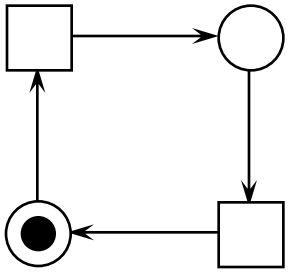




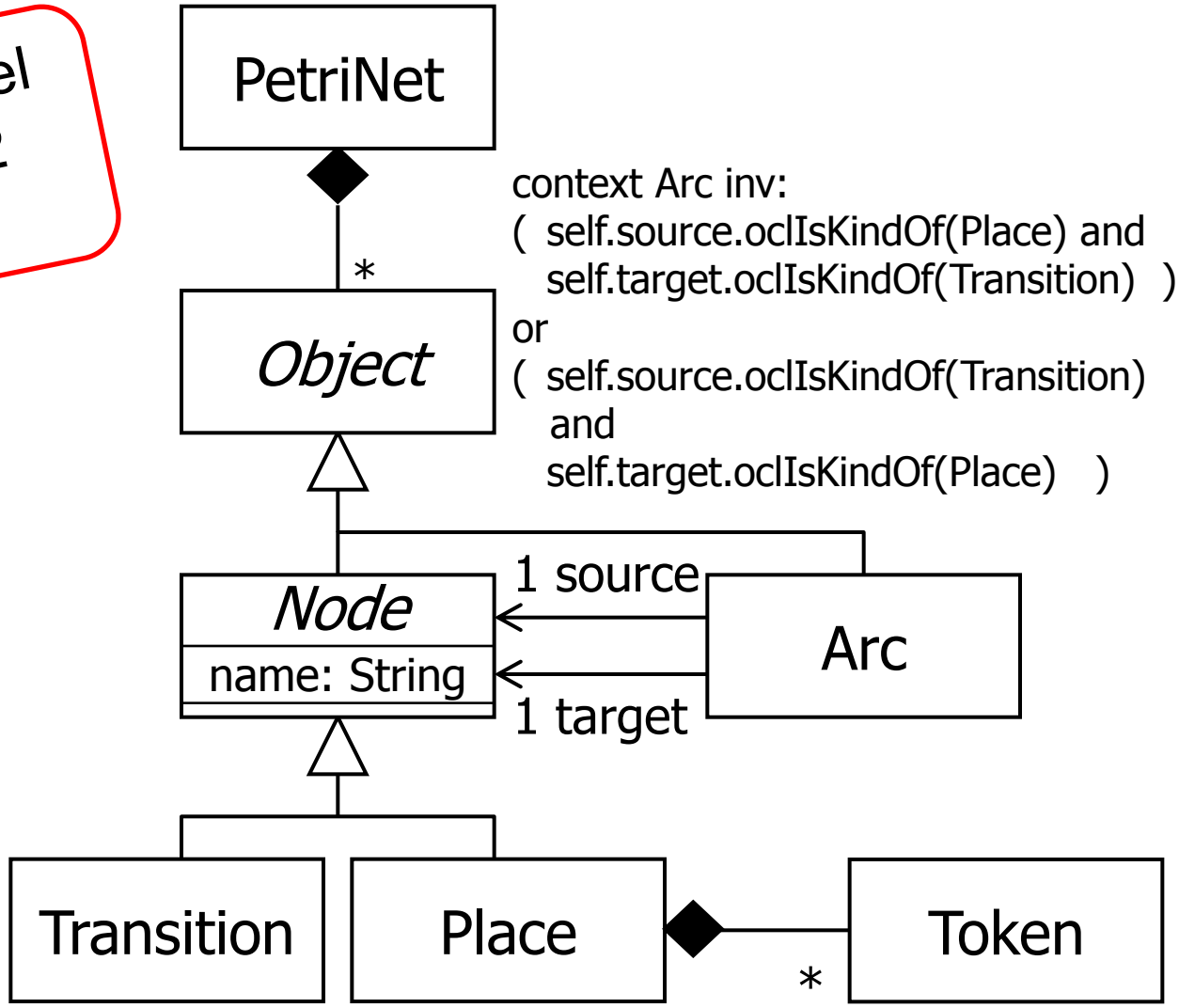
Models and meta-models

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

→ PNML meta model
ISO/IEC 15909-2
→ ePNK



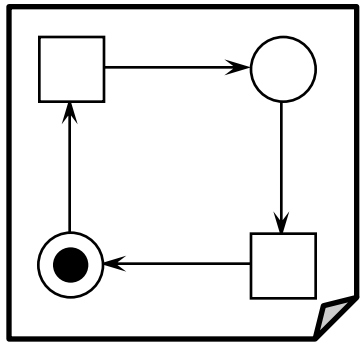
Petri net model



Meta-model for Petri nets

Syntax (abstract and concrete)

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler



graphical /
concrete syntax

Overview

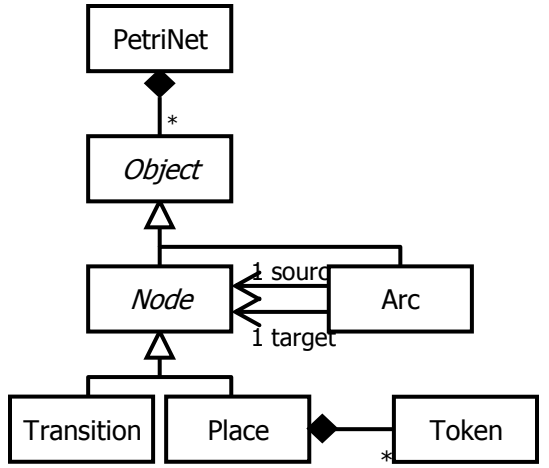
meta model

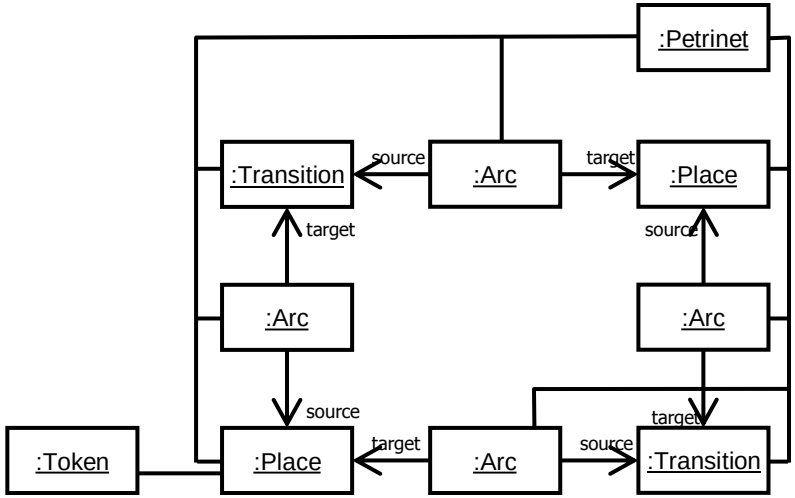
build-time

is an instance of

model

runtime



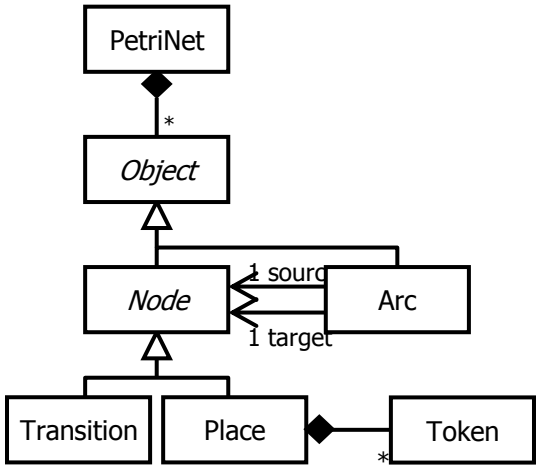


Benefits of modelling

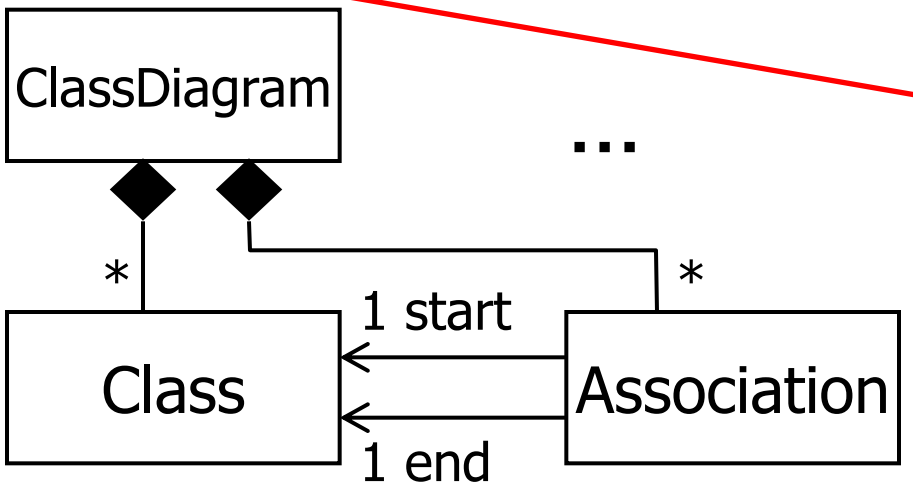
- Better understanding
- Mapping of instances to XML syntax (XMI)
- Automatic code generation
  - API for creating, deleting and modifying model
  - Methods for loading and saving models (in XMI)
  - Standard mechanisms for keeping track of changes (observers)

Class Diagrams are models too

Note: *We will see later that this is* ***much more*** *involved!*



UML model



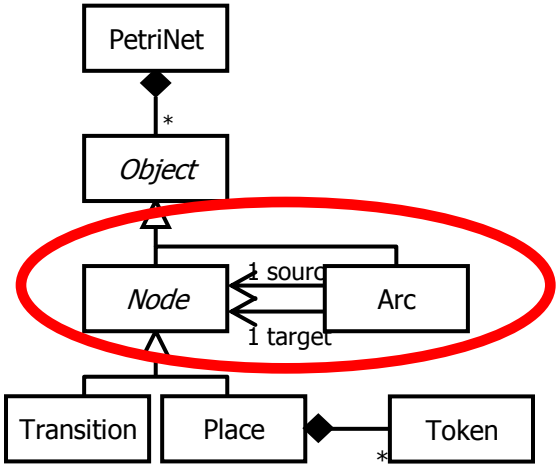
Meta-model for UML (class diagrams)

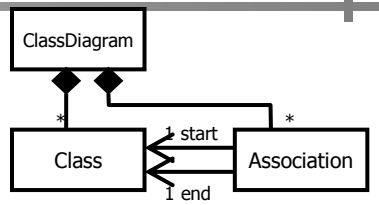
Now, the term “meta” model makes sense!

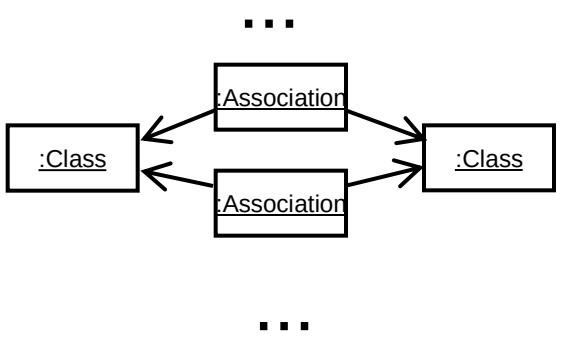
Meta-levels

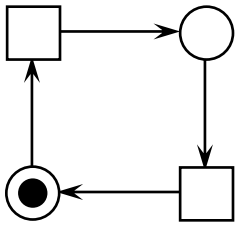
DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

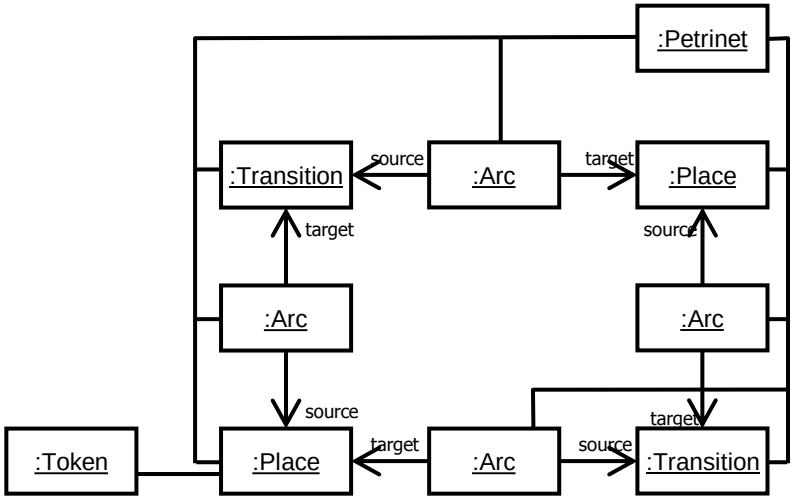
We come back to that, when we learn more about the Meta Object Facility (MOF)





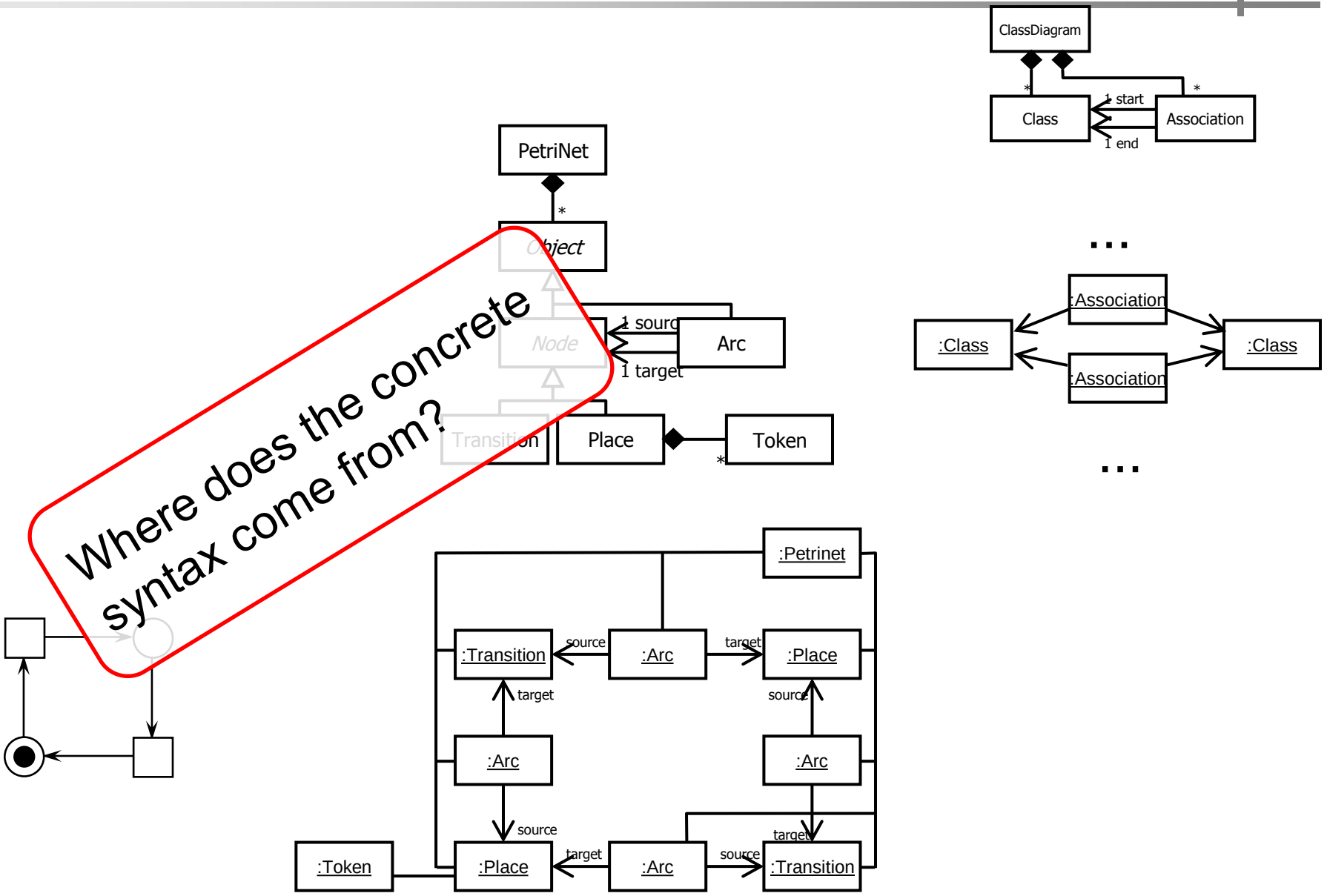




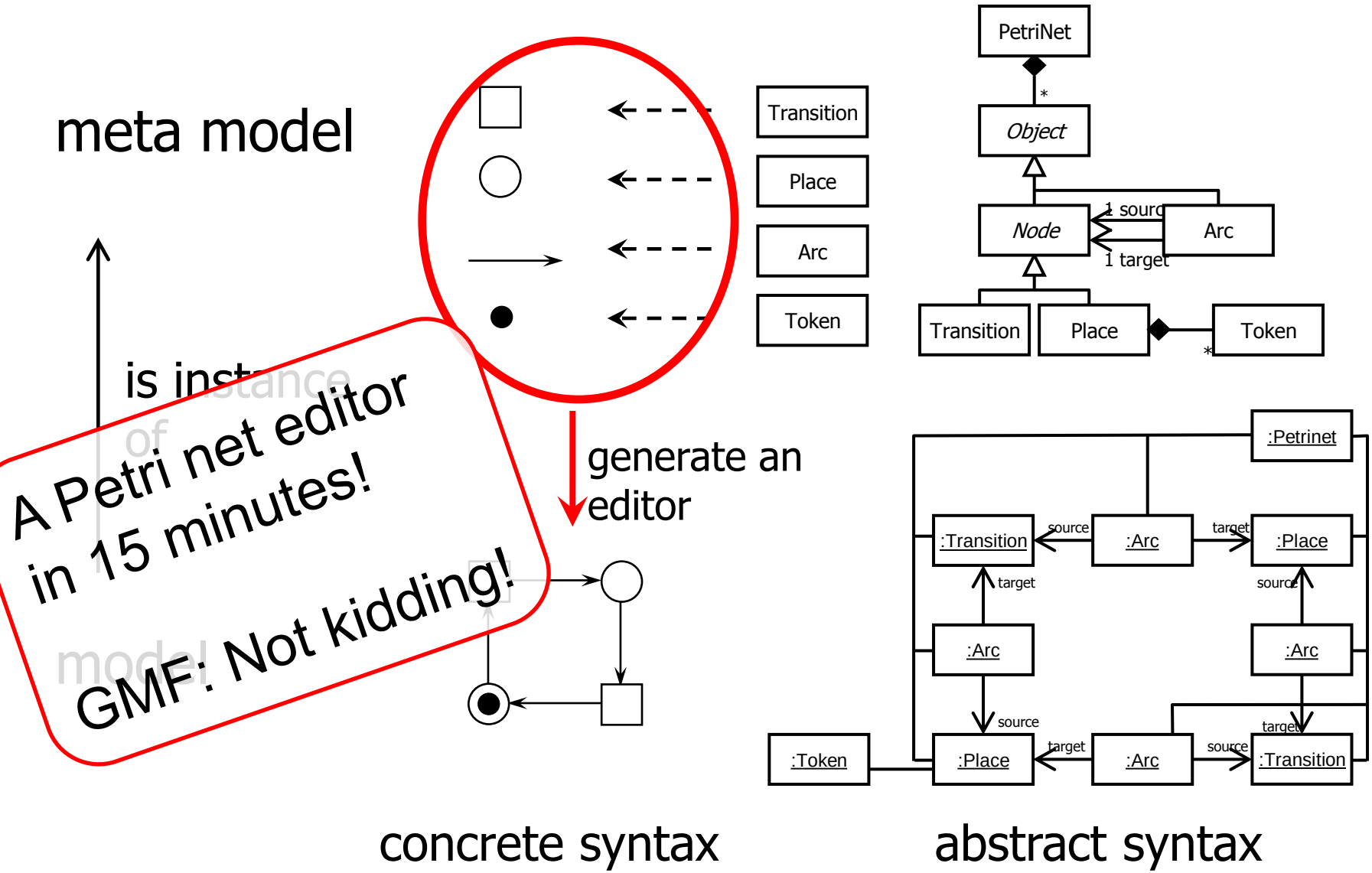


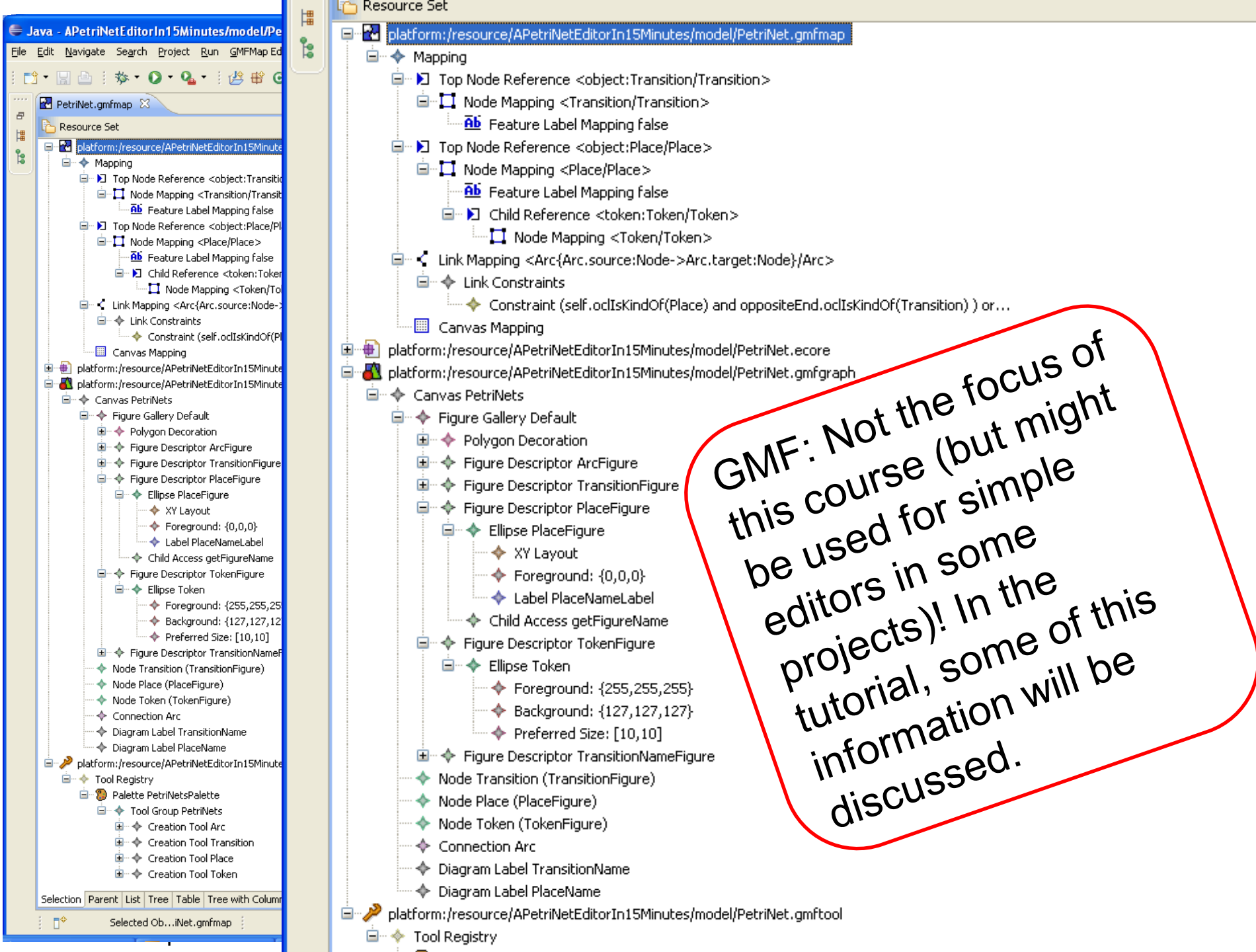
Defining concrete syntax

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler



EMF/GMF Technology




Java - APetriNetEditorIn15Minutes/model/Pe
File Edit Navigate Search Project Run GMFMap Ed
PetriNet.gmfmap
Resource Set
platform:/resource/APetriNetEditorIn15Minute
Mapping
Top Node Reference <object:Transiti
Node Mapping <Transition/Transit
Feature Label Mapping false
Top Node Reference <object:Place/Pl
Node Mapping <Place/Place>
Feature Label Mapping false
Child Reference <token:Token
Node Mapping <Token/To
Link Mapping <Arc{Arc.source:Node->
Link Constraints
Constraint (self.oclIsKindOf(Pl
Canvas Mapping
platform:/resource/APetriNetEditorIn15Minute
platform:/resource/APetriNetEditorIn15Minute
Canvas PetriNets
Figure Gallery Default
Polygon Decoration
Figure Descriptor ArcFigure
Figure Descriptor TransitionFigure
Figure Descriptor PlaceFigure
Ellipse PlaceFigure
XY Layout
Foreground: {0,0,0}
Label PlaceNameLabel
Child Access getFigureName
Figure Descriptor TokenFigure
Ellipse Token
Foreground: {255,255,25
Background: {127,127,12
Preferred Size: [10,10]
Figure Descriptor TransitionNameF
Node Transition (TransitionFigure)
Node Place (PlaceFigure)
Node Token (TokenFigure)
Connection Arc
Diagram Label TransitionName
Diagram Label PlaceName
platform:/resource/APetriNetEditorIn15Minute
Tool Registry
Palette PetriNetsPalette
Tool Group PetriNets
Creation Tool Arc
Creation Tool Transition
Creation Tool Place
Creation Tool Token
Selection Parent List Tree Table Tree with Colum
Selected Ob...iNet.gmfmap
Resource Set
platform:/resource/APetriNetEditorIn15Minutes/model/PetriNet.gmfmap
Mapping
Top Node Reference <object:Transition/Transition>
Node Mapping <Transition/Transition>
Feature Label Mapping false
Top Node Reference <object:Place/Place>
Node Mapping <Place/Place>
Feature Label Mapping false
Child Reference <token:Token/Token>
Node Mapping <Token/Token>
Link Mapping <Arc{Arc.source:Node->Arc.target:Node}/Arc>
Link Constraints
Constraint (self.oclIsKindOf(Place) and oppositeEnd.oclIsKindOf(Transition)) or...
Canvas Mapping
platform:/resource/APetriNetEditorIn15Minutes/model/PetriNet.ecore
platform:/resource/APetriNetEditorIn15Minutes/model/PetriNet.gmfgraph
Canvas PetriNets
Figure Gallery Default
Polygon Decoration
Figure Descriptor ArcFigure
Figure Descriptor TransitionFigure
Figure Descriptor PlaceFigure
Ellipse PlaceFigure
XY Layout
Foreground: {0,0,0}
Label PlaceNameLabel
Child Access getFigureName
Figure Descriptor TokenFigure
Ellipse Token
Foreground: {255,255,255}
Background: {127,127,127}
Preferred Size: [10,10]
Figure Descriptor TransitionNameFigure
Node Transition (TransitionFigure)
Node Place (PlaceFigure)
Node Token (TokenFigure)
Connection Arc
Diagram Label TransitionName
Diagram Label PlaceName
platform:/resource/APetriNetEditorIn15Minutes/model/PetriNet.gmftool
Tool Registry
GMF: Not the focus of this course (but might be used for simple editors in some projects)! In the tutorial, some of this information will be discussed.

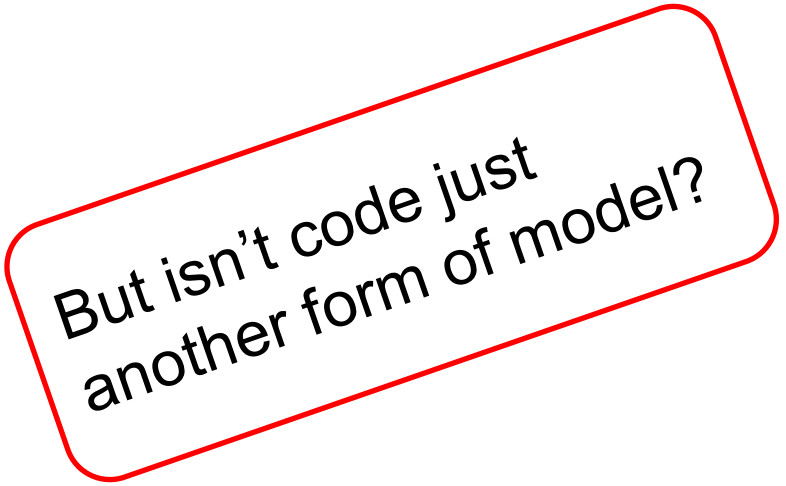
Benefits of Modelling (cntd.)

- Better Understanding
- Mapping of instances to XML syntax (XMI)
- Automatic Code Generation
  - API for creating, deleting and modifying model
  - Methods for loading and saving models (in XMI)
  - Standard mechanisms for keeping track of changes (observers)
  - Editors and GUIs

How about “real” functionality / behaviour?
→ later in this course!

Purpose of (good) models

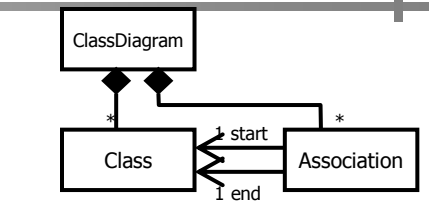
- **Abstraction**
- Focus
- Simplification
- Separation

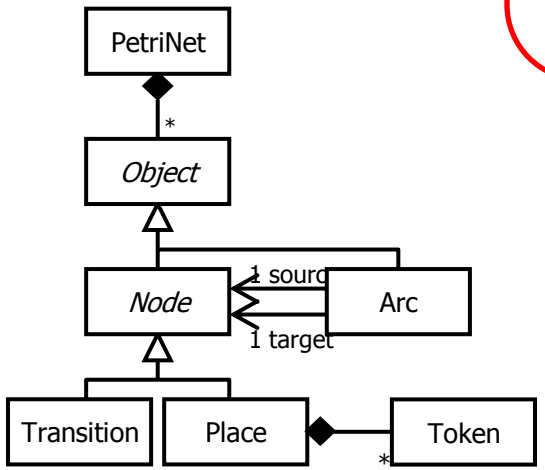


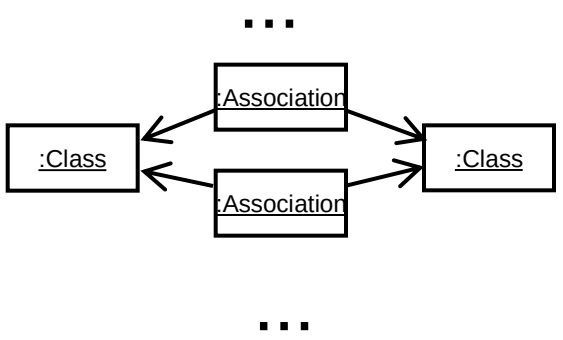
- Understanding
- Communication
- Analysis
- Execution (interpretation / code generation)

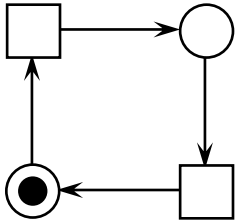
Our focus in ATSE:

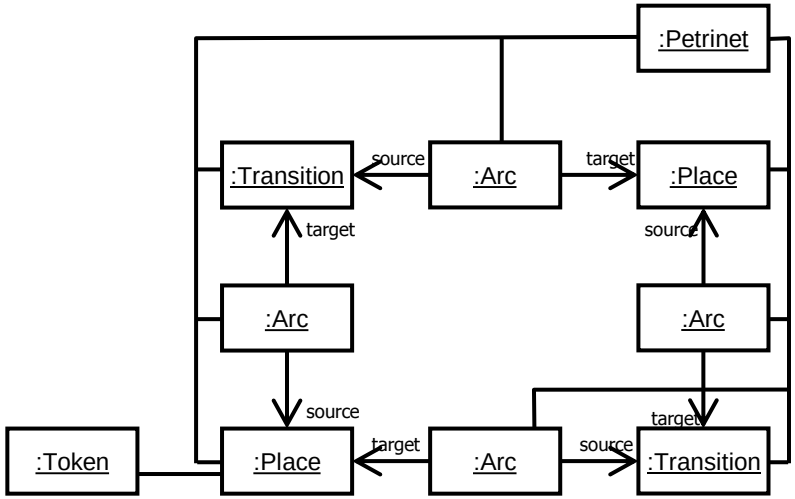
DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler











4. Basic terminology

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

- Concept
- Formalism
- Method / methodology
- Model / meta-model
- Notation
- Principle
- Technique
- Technology
- Tool

- Software engineering

- Taxonomy
- Ontology

- Framework
- Approach

Modell

Modell [*lat.-vulgärlat.-it.*] *das; -s, -e*:

…

7. die vereinfachte Darstellung der Funktion eines Gegenstands od. des Ablaufs eines Sachverhalts, die eine Untersuchung od. Erforschung erleichtert od. erst möglich macht.

…

[nach Duden: Das Fremdwörterbuch, 1990].

Model

Model [*lat.-vulgärlat.-it.*]

…

7. the simplified description of the function, purpose, or process of something; it enables us investigating and analysing this thing.

…

[translated from Duden: Das Fremdwörterbuch, 1990].

Technology

Technology Technology is a term with origins in the Greek "*technologia*", "*τεχνολογία*" — "*techne*", "*τέχνη*" ("craft") and "*logia*", "*λογία*" ("saying"). However, a strict definition is elusive; "technology" can refer to material objects of use to humanity, such as machines, hardware or utensils, but can also encompass broader themes, including systems, methods of organization, and techniques. The term can either be applied generally or to specific areas: examples include "construction technology", "medical technology", or "state-of-the-art technology".

From: http://en.wikipedia.org/wiki/Technology

Technique

A technique is a procedure used to accomplish a specific activity or task:

- Technology, the study of or a collection of techniques
- Skill, the ability to perform a task
- Scientific technique, any systematic method to obtain information of a scientific nature

From: http://en.wikipedia.org/wiki/Technique

Principle

Principle

Etymology: Middle English, from Middle French *principe, principle,* from Old French, from Latin *principium* beginning, from *princip-, princeps* initiator — more at prince

Date: 14th century

1 a**:** a comprehensive and fundamental law, doctrine, or assumption
b (1)**:** a rule or code of conduct (2)**:** habitual devotion to right principles <a man of *principle*> c**:** the laws or facts of nature underlying the working of an artificial device

2 …

From: http://www.merriam-webster.com/dictionary/Principle

Concept

A **concept** (abstract term: **"conception"**) is an cognitive unit of *meaning*— an abstract idea or a mental symbol sometimes defined as a "unit of knowledge," built from other units which act as a concept's characteristics. A concept is typically associated with a corresponding representation in a language or symbology such as a word.

From: http://en.wikipedia.org/wiki/Concept

Method

method

Etymology: Middle English, prescribed treatment, from Latin *methodus,* from Greek *methodos,* from *meta-* + *hodos* way

Date: 15th century

1**:** a procedure or process for attaining an object: as a
(1)**:** a systematic procedure, technique, or mode of inquiry employed by or proper to a particular discipline or art
(2)**:** a systematic plan followed in presenting material for instruction
b (1)**:** a way, technique, or process of or for doing something (2)**:** a body of skills or techniques

2**:** a discipline that deals with the principles and techniques of scientific inquiry

3 a**:** orderly arrangement, development, or classification **:** plan b**:** the habitual practice of orderliness and regularity

From: http://www.merriam-webster.com/dictionary/Method

Methodology

methodology

- a set of methods for achieving something (typically along with some instructions on when and how to use them, and the rational behind them)

Generally “X-ologies” comprise a body of knowledge and science on “X” (comes from Greek “logos”).

Often abused to make things to “sound more important”!

Notation

notation

Etymology: Latin *notation-, notatio,* from *notare* to note

Date: 1584

1**:** annotation , note

2 a**:** the act, process, method, or an instance of representing by a system or set of marks, signs, figures, or characters
b**:** a system of characters, symbols, or abbreviated expressions used in an art or science or in mathematics or logic to express technical facts or quantities

We call a notation a formalism, if it has a more or less precise meaning!

From: http://www.merriam-webster.com/dictionary/notation

Basic terminology

- Concept
- Formalism
- Method / methodology
- Model / meta-model
- Notation
- Principle
- Technique
- Technology
- Tool

- Software engineering

- Taxonomy
- Ontology

- Framework
- Approach

More specific terminology explained throughout this course, e.g.: MBSE, MDA, DSL, Meta-modelling, M2M, M2T, ...

What is UML?

A method?

A methodology?

A principle?

A notation?

A formalism?

A concept?

A technique?

A technology?

Software engineering

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

is the sum of all means, facilities, procedures, processes, notations, methods, concepts and principles for developing, operating and maintaining a software system.

In this course, the focus is on the software development and technology for that purpose.

What is software engineering?

A method?

A methodology?

A principle?

A notation?

A formalism?

A concept?

A technique?

A technology?

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

ATSE in a nutshell

- Advanced SE technologies (principles, concepts, methods, notations, tools, ...)
- and how to implement tool support for software developent (and concepts and technologies for that purpose)

Projects implement a part of such a tool!

DSL:
Domain Specific Language

Content

- DSLs and ”its” flavours
- Meta-modelling
  - MOF (eMOF)
  - XMI
- Notations
  - OCL
  - …
- Using models
  - Frameworks (ex. EMF+)
- Formalisation
  - Mathematical models
  - Model checking
  - Graph grammars
- MDA / MBSE
  - Idea and principles
  - Code generation (JET, …)
  - Transformations (QVT, TGGs)
  - Behaviour modelling
  - “Textmodels” (ASTs,Xtext?)
- Further ideas
  - Merging and ”diffing” models
  - Aspect oriented modelling
- IDE-Integration
- Behaviour modelling
- ...

6. Course organisation

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

- Lecture part:
  - Concepts and underlying theory of Model-based Software Engineering (with focus on the meta-level)
  - Relation between the concepts and rationale behind them
- Tutorial part:
  - Use of basic technology (Eclipse/EMF/ePNK/ECNO Tool)
  - Practical application of (some of the) concepts and techniques for small examples
- Project:
  - A simple tool for some aspect of software development (which requires to use a combination of some concepts of this course)
  - In groups of 2-**4** students
  - There are different predefined topics from which the groups may chose (see other presentation);

Weekly Schedule

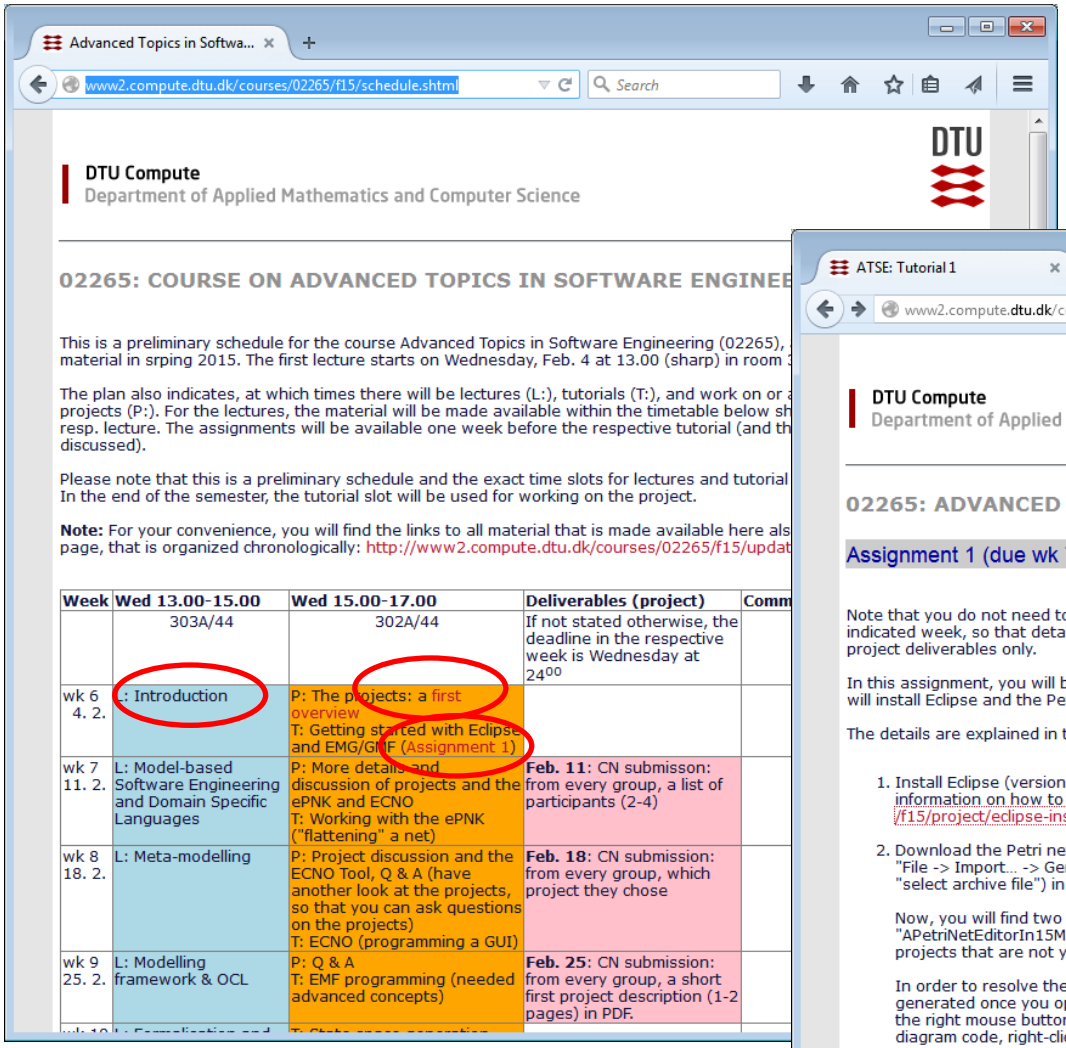
DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

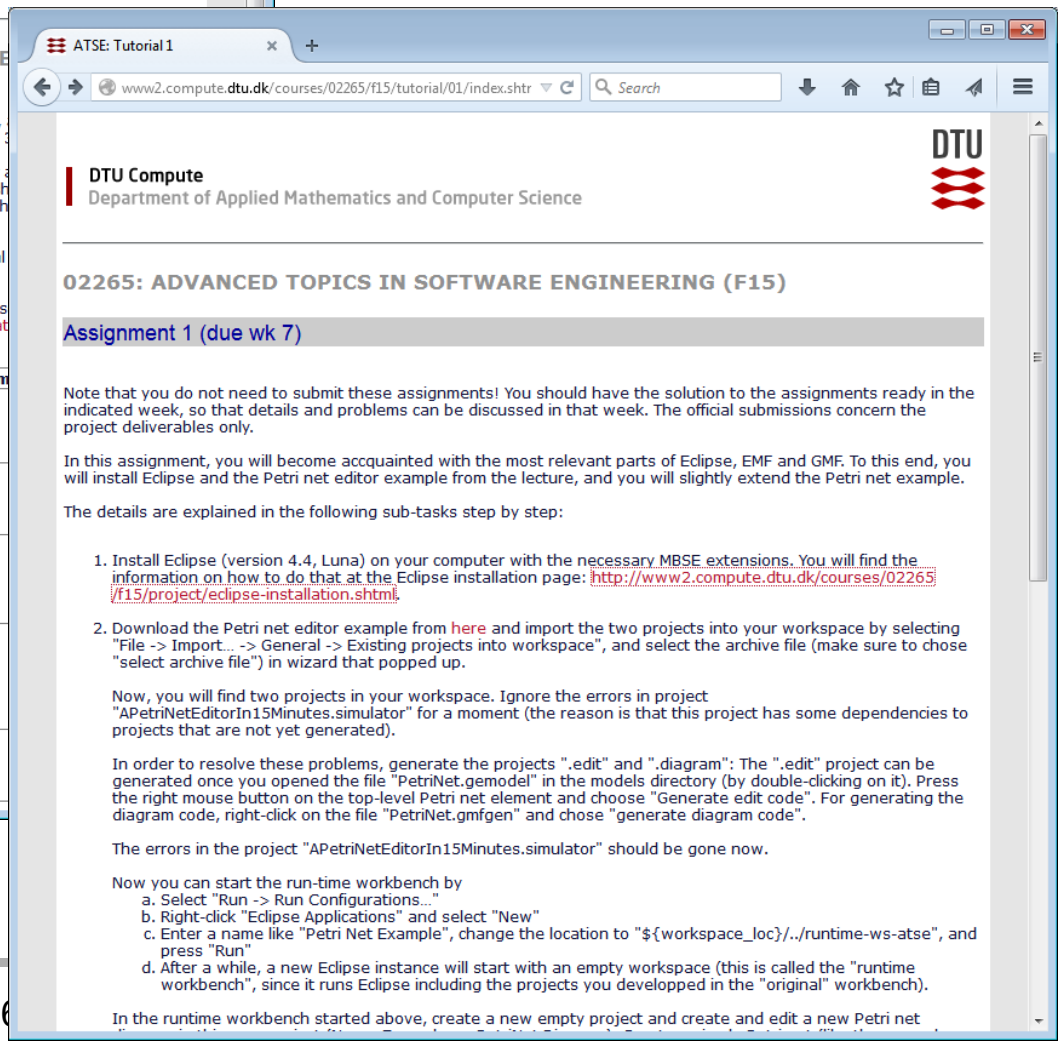
	Mon	Tue	Wed	Thu	Fri
8-10					
10-12					
13-15			lecture		
15-17			tutorials / project		

Plus actual work on tutorials an the project!!

There will some exceptions (check web pages regularly) :
http://www2.compute.dtu.dk/courses/02265/f15/schedule.shtml

Bookmark this page!!!





Contact

DTU Informatics
Department of Informatics and Mathematical Modelling
Ekkart Kindler

- Teacher: Ekkart Kindler
  - Email: ekki@dtu.dk
  - Office: 303B.060
  - Consultation hours: whenever
    (after the lecture/tutorial, just drop in, call, or email me)

- Material:
  - http://www2.compute.dtu.dk/courses/02265/f15/
    (in particular check for schedule, and assignments on the web)