

From Domains via Requirements to Software Designs

TU Graz Lectures, October/November/December 2008

Dines Bjørner
Fredsvej 11, DK-2840 Holte, Danmark
E-Mail: bjorner@gmail.com, URL: www.imm.dtu.dk/~db

August 18, 2008

1 Abstract

A plan is suggested for DB's

- 17 day,
- one 90 minute am lecture session
- + one 2 hour pm tutoring session per day,
- lectures at TU Graz, October/November/December 2008.

Contents

1	Abstract	1
2	Course Aims and Objectives	2
2.1	Aims	2
2.2	Objectives	2
3	Calendar	3
4	Lecture Plan	3
5	An Explication	5
6	Course Project	6

2 Course Aims and Objectives

2.1 Aims

- The course lecturer shall cover
 - Main aspects of **Domain Engineering**,
 - main aspects of **Requirements Engineering**,
 - and otherwise overview aspects of **Software Design**.

2.2 Objectives

- The course participants are expected to become
 - well-versed in principles, techniques and tools for
 - the informal and formal description of domains and
 - the informal and formal prescription of requirements;
 - and will get to understand crucial relations between
 - * requirements and
 - * software design.
 - The course participants are further expected to become
 - * intimately aware of the how
 - * requirements prescriptions
 - * can be systematically derived
 - * from domain descriptions.
 - Finally course participants are expected to
 - * better appreciate the need for formalisations
 - * whether in an abstract specification language
 - * or in a reasonably high level programming language;

and

 - * identify important
 - software engineering
 - cum programming methodology
 - research topics.

3 Calendar

October 2008

Mo	Tu	We	Th	Fr	Sa	Su			
				30	31				DB's arrival at TUG

November 2008

Mo	Tu	We	Th	Fr	Sa	Su			
					1	2			
3	4	5	6	7	8	9	lects.	1 - 5	
10	11	12	13	14	15	16	lects.	6 - 10	
17	18	19	20	21	22		lects.	11 - 15	
					29	30	DB in Japan	(Nov. 23 - 28)	

December 2008

Mo	Tu	We	Th	Fr	Sa	Su			
1	2	3	4	5	6	7	lects.	16 - 17 (Dec. 8 - 9)	
8	9	10	11	12			exam	8 Dec., grades 12 Dec.	

4 Lecture Plan

- **Introduction:**

Lecture 1 : The Triptych and Informative Documents (I): **Mon. 3 Nov.**

1. The Triptych Paradigm
2. Phases, Stages and Steps
3. Informative Documents (I)

- | | |
|------------------------------------|----------------------------------|
| (a) Project Name and Date | (e) Scope and Span |
| (b) Project Partners and Addresses | (f) Assumptions and Dependencies |
| (c) Current Situation | (g) Implicit/Derivative Goals |
| (d) Needs and Ideas | (h) Synopsis |

Lecture 2 : Inform.Docs. (II) & Method.: **Tue. 4 Nov.**

1. Informative Docs. (II)

- | | |
|---------------------------------|---------------------------------|
| (i) Software Development Graphs | (l) Standards Compliance |
| (j) Resource Allocation | (m) Contracts and Design Briefs |
| (k) Budget Estimate | (n) Logbook |

2. Methodology

Lecture 3: Conceptual Framework (I): **Wed. 5 Nov.**

1. Modelling and Analysis
2. Descrs., Prescrs., Specs.
3. Informal and Formal Development
4. Software

Lecture 4: Conceptual Framework (II): **Thu. 6 Nov.**

5. Entities, Functions, Events, Behaviours
6. Domain Modelling versus Operational Research

• **Domain Engineering:**

Lecture 5 : Prelude Stages: **Fri. 7 Nov.**

1. The Domain Concept
2. Stages of Domain Engineering
3. Domain Stakeholders
4. Domain Acquisition
5. Domain Analysis and Concept Formation
6. Business Processes
7. Terminology

Lectures 6–8 : Domain Modelling:

1. **Lecture 6:** **Mon. 10 Nov.**

- (a) Intrinsic
- (b) Support Technologies

2. **Lecture 7:** **Tue. 11 Nov.**

- (a) Management and Organisation
- (b) Rules and Regulations

3. **Lecture 8:** **Wed. 12 Nov.**

- (a) Scripts
- (b) Human Behaviour

Lecture 9 : Postlude Stages: **Thu. 13 Nov.**

1. Verification
2. Validation
3. Theory Formation
4. Domain Engineering Process Graph
5. Domain Engineering Documents

• **Requirements Engineering:**

Lecture 10 : Prelude Stages: The Requirements Engineering Stages **Fri. 14 Nov.**

Lecture 11 : Domain Reqs. Modelling: **Mon. 17 Nov.**

1. Projection,
2. Instantiation,
3. Determination
4. Extension
5. Fitting
6. Composition

Lecture 12 : Interface Reqs. Modelling: **Tue. 18 Nov.**

- | | |
|---------------------------------|----------------------------------|
| 1. Shared Phenomena | 4. Shared Event Requirements |
| 2. Shared Entity Requirements | 5. Shared Behaviour Requirements |
| 3. Shared Function Requirements | |

Lecture 13 : Machine Reqs.:

Wed. 19 Nov.

- | | |
|--------------------|--------------------|
| 1. Performance | 4. Platform |
| 2. Dependability | 5. Documentability |
| 3. Maintainability | 6. Etcetera |

Lecture 14 : Postlude Stages:

Thu. 20 Nov.

- | | |
|-------------------------------------|---------------------------------------|
| 1. Verific., Valid. | Graph |
| 2. Feasibility, Satisfiability | 4. Requirements Engineering Documents |
| 3. Requirements Engineering Process | |

• **Software Design:**

Lecture 15 : Architectural Design

Fri. 21 Nov.

Lecture 16 : Component Design &c.

Mon. 1 Dec.

- | | |
|----------------------------------|------------------------------|
| 1. Component Design | 3. Software Design Documents |
| 2. Software Design Process Graph | |

• **Summary:**

Lecture 17 : Review of Phases, Stages and Steps:

Tue. 2 Dec.

- | | |
|---|---|
| 1. Domains, Requirements, and Software Design | 3. Documents |
| 2. Process Graphs | 4. Process Assessment and Process Improvement |

5 An Explication

- I assume students to be more-or-less full time occupied by this course, **at least the first 3 weeks of the course.**
- Each lecture session is two times 45 minutes.
- By a ‘formal’ session we mean
 - a possibly tiered auditorium session in which the lecturer
 - lectures cover Vol. 1 material
 - while showing some Vol. 2 examples
 - on two overhead projectors simultaneously —

- * one for Vol. 1 slides,
- * the other, occasionally “blinded”, for Vol. 2 slides.
- By a ‘tutoring’ session we mean a
 - a, usually flat classroom, usually 2 hour session
 - in which the lecturer
 - only shows Vol. 2 slides
 - while walking around the room, discussing the examples
 - and their work on the course project with students.

6 Course Project

- The course project is
 - to take a subset of the formalisations shown in Vol. 2
 - and re-express them in the Alloy specification language:
 - * Daniel Jackson:
 - * *Software Abstractions Logic, Language, and Analysis*.
 - * The MIT Press, Cambridge, Mass., USA, April 2006.
 - * ISBN 0-262-10114-9.
- The lecturer will provide full L^AT_EX files:
 - .cls, .sty., .tex, .fig, and .eps
- for the students, as a whole, to produce an alternative Vol. 2
 - for a subset of what is narrated and formalised in the Vol. 2 provided by the lecturer;
 - that “student” volume is proposed to become public property.