Analyzing Students' Perception of Modeling in Software Engineering

Omar Badreddin, Arnon Sturm, Abdelwahab Hamou-Lhadj, Timothy Lethbridge, Waylon Dixon, Ryan Simmons

Survey, Empirical SE, Modeling, Education

Outline

1. Introduction
2. Study Design & Execution
   - Research Questions
   - The Survey
   - Educational Background
3. Results & Analysis
   - Threats to Validity
4. Conclusions and Future Work

Model Driven Engineering Usage

- ☐ Models can
  - ◻ be easier to understand
  - ◻ improve communications amongst stakeholders
  - ◻ help generate executable artifacts
  - ◻ improve system portability
- ☐ UML is the standard modeling language?
  - ◻ What does it mean?
- ☐ The adoption rate of modeling

Our goal in this study is to investigate students' perception of modeling

Study Goals and Objectives

Research Questions

- Do students perceive models to be useful? And in what context? What are the reasons for that?

- How does students' perception of modeling evolve over the years?

- Do students think or wish to have a more substantial modeling education?

Study Plan

☐ Distribute questionnaires among CS, SE, ISE students in three institutes:

Northern Arizona University (U.S), Ben-Gurion University (Israel), and Concordia University (Canada).

☐ The questionnaire include two parts:

- Demographic data: the study program, the academic year, age, work experience, and the average grade
- Reflection over modeling
  - Applicability of Models (APP)
  - Modeling Characteristics (CHR)
  - Implementation (IMP)
  - Modeling Education (EDU)
- The students were requested to rank the various statements in a Likert scale: Strongly Agree, Agree, Neutral, Disagree, Strongly Disagree, and NA.

Questionnaire - Applicability of Models

1. Models are very useful
2. Models are useful for documentation
3. Models are useful for communication
4. Models are useful for representing requirements
5. Models are useful for specification
6. Models are useful for implementation and/or code generation
7. Models are useful for testing
8. Models are useful for maintenance

Questionnaire - Modeling Characteristics

1. Models are normally used just as drawings
2. Code is just a type of model
3. Models are precise (i.e., unambiguous)
4. Models can be easily checked to find opportunities for improvement
5. Models are more comprehensible than code
6. In general, models are easy to understand
7. Models facilitate abstractions and comprehension
8. Textual models are easier to understand than graphical models
9. Textual models are easier to construct than graphical models
10. Models are implementation independent
11. Models help provide flexibility during the development process
12. Modeling is counterproductive since the models need to be changed all the time
13. Models are usually abandoned after the code is written

Questionnaire - Implementation

1. Modeling tools are not mature enough
2. Modeling tools are too complex and are difficult to learn
3. It is not easy for developers to obtain modeling tools that meet their needs

Questionnaire - Modeling Education

1. Modeling should be taught before programming
2. Modeling and programming should be taught at the same time
3. Modeling is not being taught sufficiently
4. Modeling should be integrated in most software engineering and computer science courses

Educational Background

- Concordia University – two programs CS and SE
- Ben-Gurion University – two programs SE and ISE, the former emphasizes SE principles, the latter emphasizes data processing
- Northern Arizona University – two programs CS and ACS, the former emphasizes theoretical foundation, the latter is more applied.
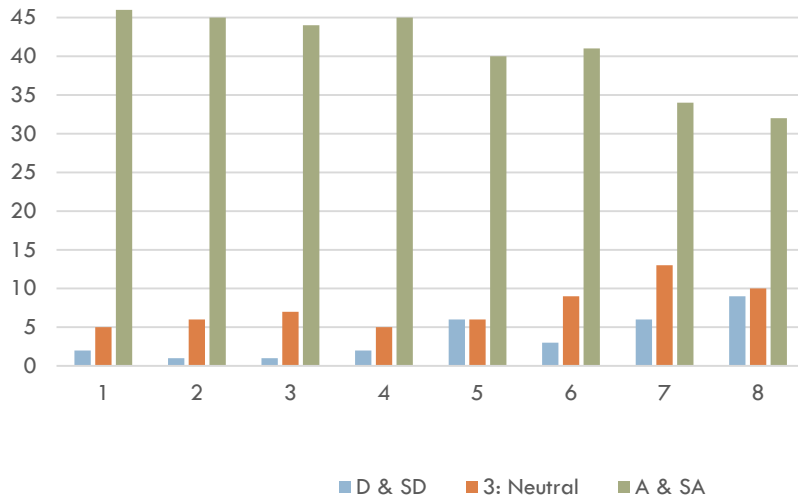
Results

☐ All in all we got 195 responses

Institute	Number of Responses				
	Y1	**Y2**	**Y3**	**Y4**	**Grad**
NAU	5	10	26	8	2
BGU-SE	8	12	17	22	25
BGU-ISE	3	3	23	12	
CU	0	0	0	0	19

☐ Most students had a minimal work experience

☐ The students grades spanned over a wide range

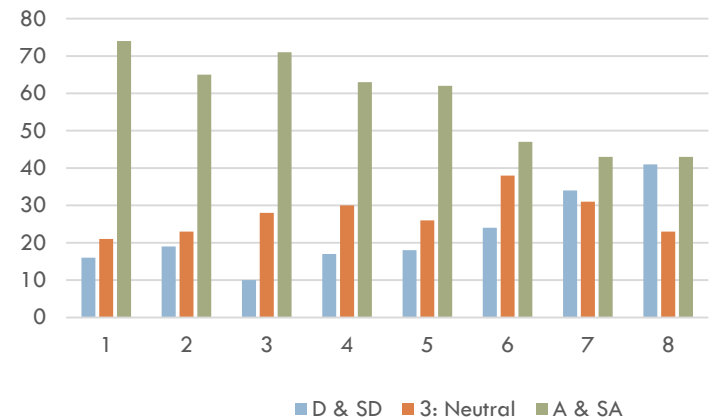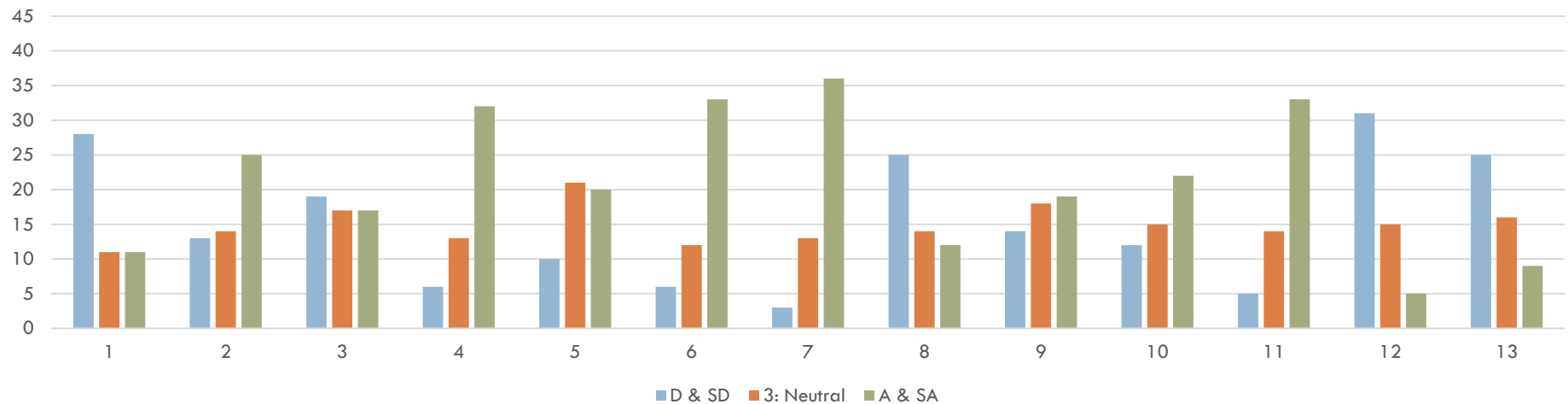# Results - Applicability of Models
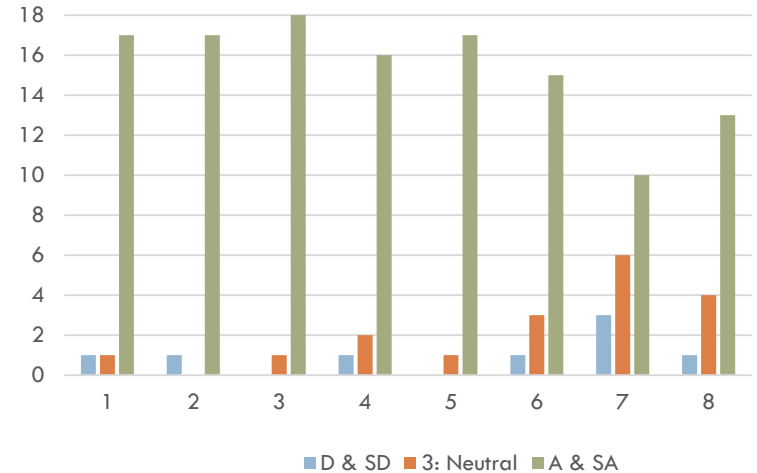
NAU

CU

BGU

1. Models are very useful
2. Models are useful for documentation
3. Models are useful for communication
4. Models are useful for representing requirements
5. Models are useful for specification
6. Models are useful for implementation and/or code generation
7. Models are useful for testing
8. Models are useful for maintenance

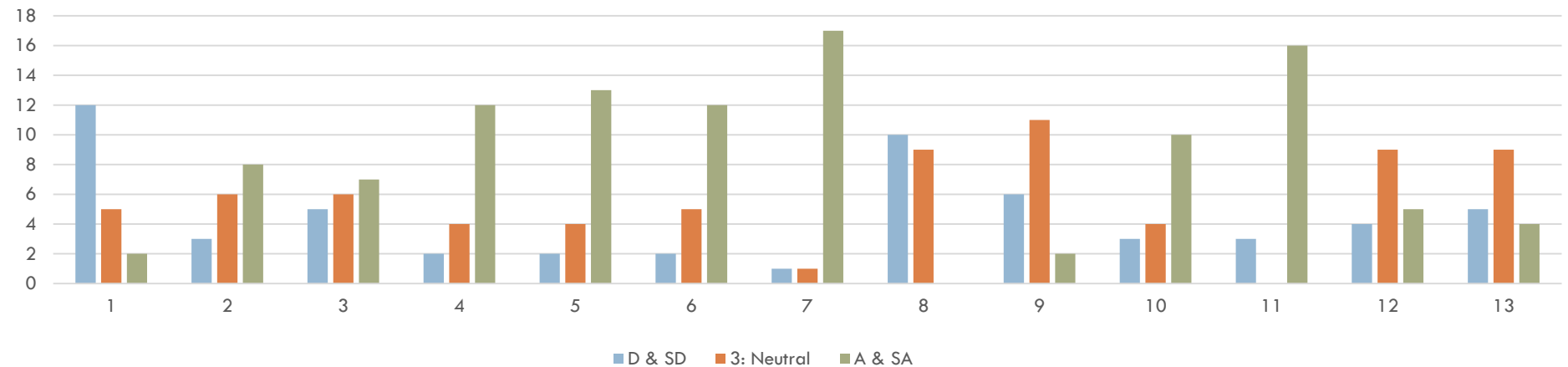# Results - Modeling Characteristics - NAU

Legend: D & SD | 3: Neutral | A & SA

1. Models are normally used just as drawings
2. Code is just a type of model
3. Models are precise (i.e., unambiguous)
4. Models can be easily checked to find opportunities for improvement
5. Models are more comprehensible than code
6. In general, models are easy to understand
7. Models facilitate abstractions and comprehension
8. Textual models are easier to understand than graphical models
9. Textual models are easier to construct than graphical models
10. Models are implementation independent
11. Models help provide flexibility during the development process
12. Modeling is counterproductive since the models need to be changed all the time
13. Models are usually abandoned after the code is written
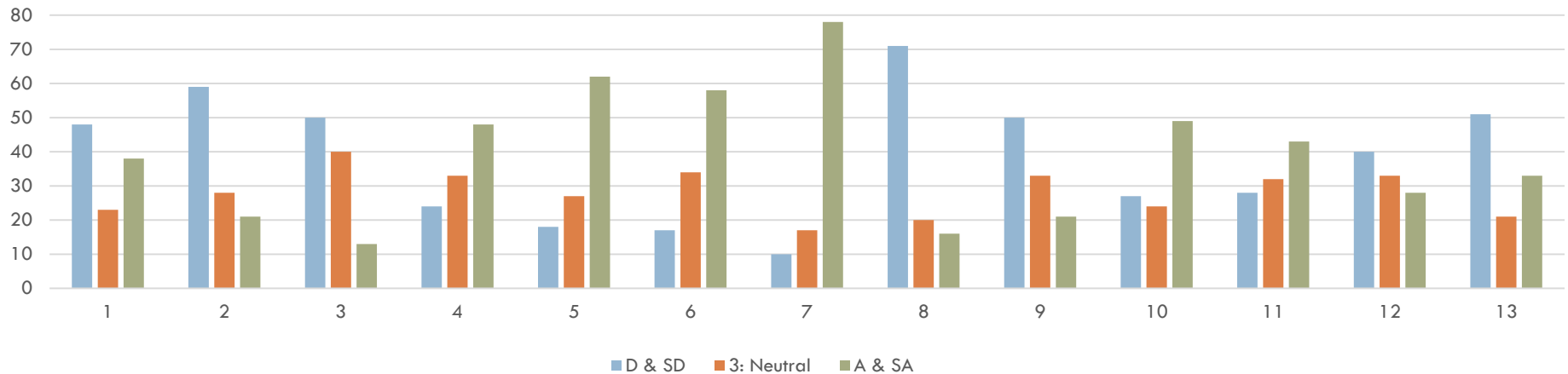
Results - Modeling Characteristics - CU

1. Models are normally used just as drawings
2. Code is just a type of model
3. Models are precise (i.e., unambiguous)
4. Models can be easily checked to find opportunities for improvement
5. Models are more comprehensible than code
6. In general, models are easy to understand
7. Models facilitate abstractions and comprehension
8. Textual models are easier to understand than graphical models
9. Textual models are easier to construct than graphical models
10. Models are implementation independent
11. Models help provide flexibility during the development process
12. Modeling is counterproductive since the models need to be changed all the time
13. Models are usually abandoned after the code is written

Results - Modeling Characteristics - BGU

Legend: ■ D & SD   ■ 3: Neutral   ■ A & SA

1. Models are normally used just as drawings
2. Code is just a type of model
3. Models are precise (i.e., unambiguous)
4. Models can be easily checked to find opportunities for improvement
5. Models are more comprehensible than code
6. In general, models are easy to understand
7. Models facilitate abstractions and comprehension
8. Textual models are easier to understand than graphical models
9. Textual models are easier to construct than graphical models
10. Models are implementation independent
11. Models help provide flexibility during the development process
12. Modeling is counterproductive since the models need to be changed all the time
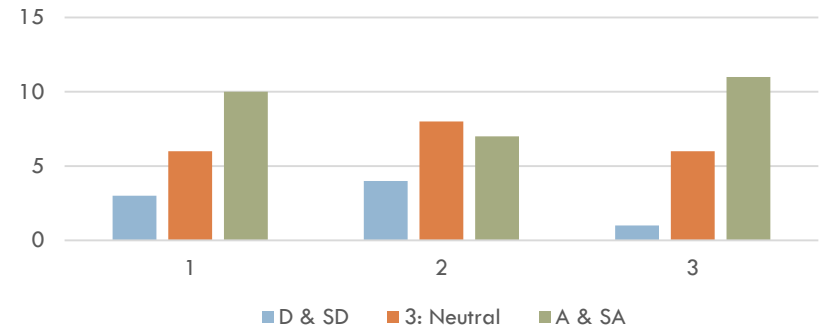13. Models are usually abandoned after the code is written
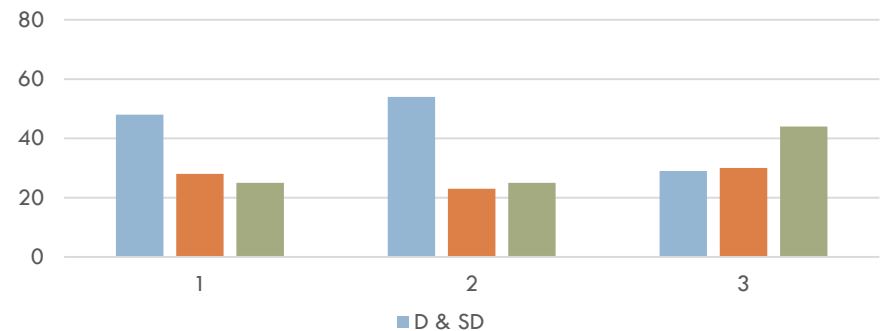
Results - Implementation
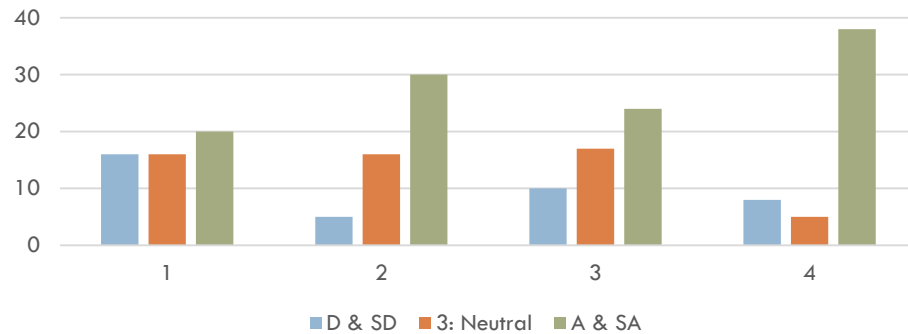
NAU
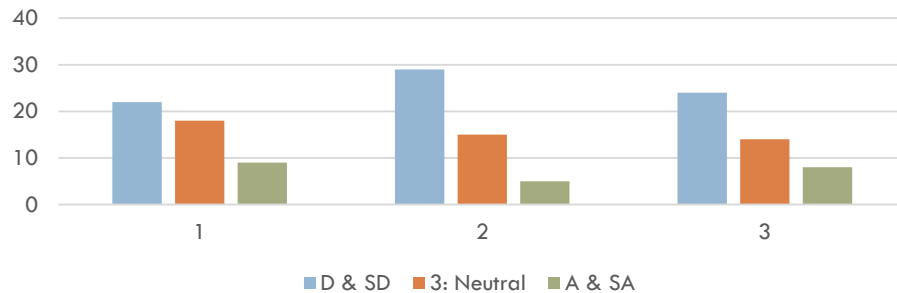
CU

BGU

1. Modeling tools are not mature enough
2. Modeling tools are too complex and are difficult to learn
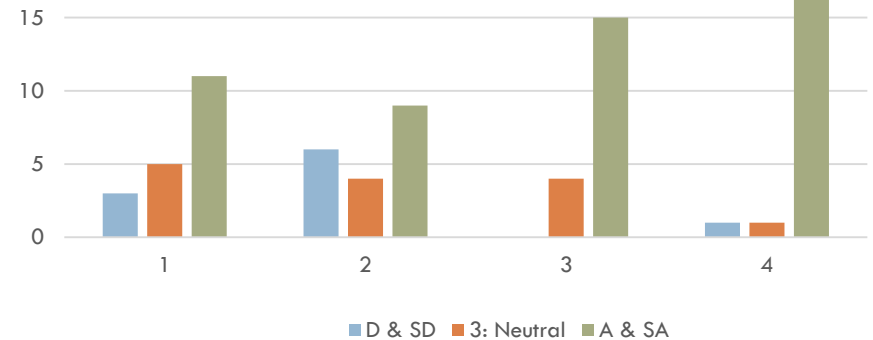3. It is not easy for developers to obtain modeling tools that meet their needs
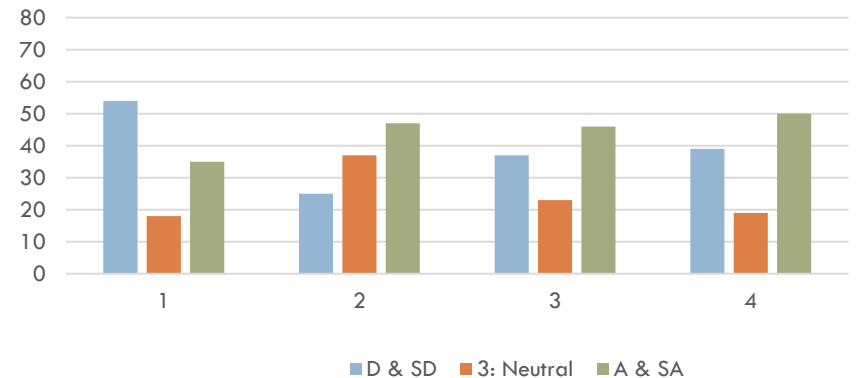
Results – Modeling Education

NAU

CU

BGU

1. Modeling should be taught before programming
2. Modeling and programming should be taught at the same time
3. Modeling is not being taught sufficiently
4. Modeling should be integrated in most software engineering and computer science courses

A Finer Grained Analysis

- □ Sign analysis among the years

	Usefulness		Education	
	+	-	+	-
NAU UG	0	14	0	5
NAU G	5	13	3	2
BGU UG	6	10	0	4
BGU G	3	12	1	3

Summary of the Results

- It is evident that students tend to find models more useful for documentation and communication, and early development tasks.

- It is evident that students think that more modeling education is required.

- We found a consistent pattern of declining perceptions emerging in both NAU and BGU undergrad and grad students. In general, the perception declination was more prominent in the case of undergraduate than graduate students.

Results Interpretation

- The curriculum fails to highlight the value of modeling in software engineering
- Students come to the program assuming unrealistically high value of modeling, however, during their education, the curriculum does not improve on that initial perception

Threats to Validity

□ ***Question Bias***

□ ***Profile of the Respondents***

□ ***Different Modeling Teaching Approaches***

Usage of Modeling by Students in Practice (Modeling Educators' Symposium – Mira Balaban)

□ To what extent do students that were taught a detailed modeling course continue using models in their studies and in particular in their capstone projects?

□ BGU, Babes-Bolyai University (Hungary) and Florida International University

- Most students are not happy with the modeling aspect
- Modeling is mostly enforced
- Students do not recognize the important role of models in project design
- The same holds for Babes-Bolyai University (Hungary) and Florida International University

□ At the University of Ottawa, and McGill University it seems the modeling is put in use more frequently.

□ At Michigan State University the focus is on early stages

Usage of Modeling by Students in Practice

□ Three teaching modes:

- Teaching modeling using an integrated environment that smoothly combines code and models, which clarifies the role of models in software construction

- Teaching modeling in a carefully designed process of requirement analysis and problem solving

- Teaching modeling using explicit formulation of models – syntax plus semantics, followed by teaching of their usage and design process

Conclusions

- The results suggest that students' perception of the value of modeling although is high declines as they progress in their education
  - In practice, students do not model….
- Further investigation and repetition of such a survey along with students' interviews are required
- Also there is a need to revisited the modeling curriculum

Questions???

Omar Badreddin, Northern Arizona University, Flagstaff, U.S.A

Omar.Badreddin@nau.edu; www.Badreddin.com

Arnon Sturm, Ben-Gurion University of the Negev, Israel

sturm@bgu.ac.il

Abdelwahab Hamou-Lhadj, Concordia University, Montreal QC, Canada

abdelw@ece.concordia.ca

Timothy Lethbridge, University of Ottawa, Ottawa ON, Canada

tcl@eecs.uottawa.ca