# Threshold Concepts in Object-Oriented Modelling

**Ven Yu Sien**

Department of Information Technology,
HELP University College,
Kuala Lumpur, Malaysia

**sienvy@help.edu.my**

# Overview

1. Threshold Concepts
2. Object-Oriented Modelling
3. Threshold Concepts in OO Modelling
4. Evaluation: Changes in Level of Understanding
5. Conclusion

# Threshold Concepts

- A threshold concept is defined to have 5 characteristics:
  - **Transformative**: A threshold concept or skill is one which when grasped, changes the way in which the learner views the discipline or approaches a task.
  - **Irreversible** or difficult to unlearn.
  - **Troublesome:** The journey across a conceptual threshold is likely to be difficult or *troublesome*, and may involve traversing the conceptual space (recursiveness) until 'the penny drops'.
  - **Integrative:** bringing together different aspects of a subject that previously appeared unrelated.
  - **Boundary:** threshold concepts help to define the boundaries of the discipline to which they belong.

# Threshold Concepts in Computing Education Research

- The following candidate threshold concepts within computing education research have been proposed:
  - OO programming [ET05]
  - Pointers [BEM+07]
  - Abstraction– modularity, data abstraction, OO [EMM+06]
  - OO at its most basic – including classes, objects and encapsulation [EMM+06]
  - Recursion [RR09]

# Overview

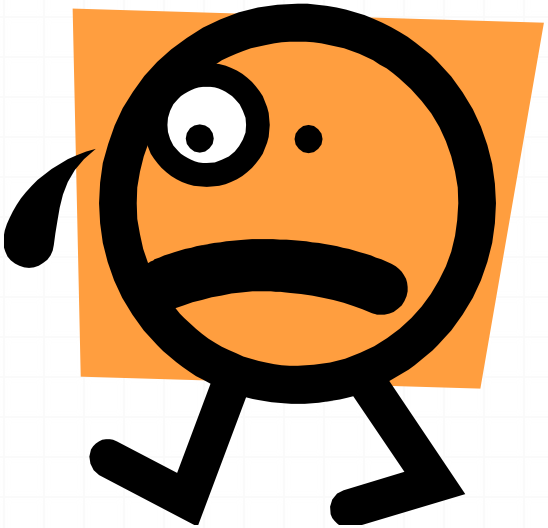
1. Threshold Concepts
2. Object-Oriented Modelling
3. Threshold Concepts in OO Modelling
4. Evaluation: Changes in Level of Understanding
5. Conclusion

# Object-Oriented Modelling

- During the OO analysis and design (OOAD) phases, models are produced to show the type of information processing that is required of the new system.
- A model of an OO system is an abstract representation of the system.
- It represents the problem domain and emphasises some characteristics of the real-world.
- Although OO is representative of real-world problems, it does not always reflect the way in which people think.

# Object-Oriented Modelling

- OOAD is not easy to learn.
- Students in general find difficulty understanding OO concepts.
- It has been observed that students frequently produce class and sequence diagrams that are incomplete, with many concepts at inconsistent abstraction levels [BL06], [SBG06].



# Difficulties in OO Modelling

- Some difficulties faced by academics and instructors when teaching OOAD:
  - Teaching students the fundamental concept of identifying objects from the problem domain.
  - Paradigm shift students need to undergo to learn OO (especially if previously exposed to structural thinking and programming).
  - Relatively long period of time that is required to get students accustomed to *OO thinking*.

# Overview

1. Threshold Concepts
2. Object-Oriented Modelling
3. Threshold Concepts in OO Modelling
4. Evaluation: Changes in Level of Understanding
5. Conclusion

# Threshold Concepts in OO Modelling

Several aspects of OO modelling are consistent with the defining criteria for threshold concepts:

- **Troublesome:** OO modelling is difficult both to teach and learn: it represents *troublesome* knowledge.
- **Irreversible:** The conceptual grasp of OO modelling is difficult to unlearn.
- OO modelling **transforms** and **integrates** understanding of analysis and design of information systems.

# Threshold Concepts in OO Modelling

- OO modelling however is a broad area within which other thresholds exist - may represent a family or group of associated thresholds including:
  - classes;
  - generalisation-specialisation hierarchies; and
  - object interactions.

# Study to Identify Difficulties in OO Modelling

Information captured on the background of the participants and their experience with OOAD.

| | Study (N=51) |
|---|---|
| Average age (years) | 22 |
| Age range (years) | 20-27 |
| Gender | |
| • male | 61% |
| • female | 39% |
| OO experience | |
| • < 1 year | 35% |
| • 1- 2 years | 49% |
| • 3-5 years | 16% |
| • > 5 years | 0% |

# Difficulties in OO Modelling

- The class diagram is the key artifact in the analysis phase as its *appropriateness* can have a significant impact on the design of the overall system.
- Appropriate Classes

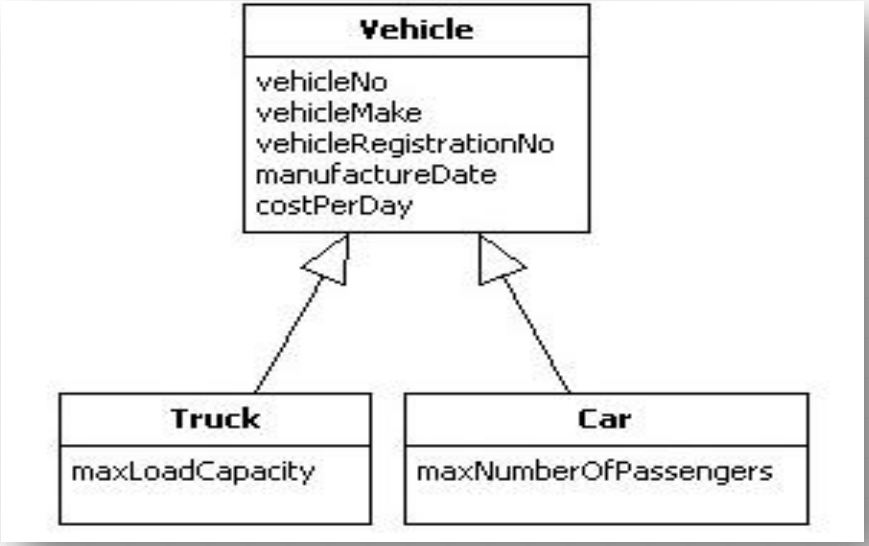
| Classes | | | | | | | |
|---|---|---|---|---|---|---|---|
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| 2% | 2% | 8% | 14% | 10% | 22% | 35% | 8% |

- Assigning attributes to a class in a class diagram depends on an understanding of how classes are defined in the problem domain i.e., the responsibilities of the classes and what information they need to know or remember.

| Misassigned Attributes | One or More Missing Attributes |
|---|---|
| 86% | 20% |

# Difficulties in OO Modelling

- In our study we expected participants to identify a generalisation-specialisation hierarchy.




| Inappropriate Use of Inheritance Hierarchy | Appropriate Use of Inheritance Hierarchy | Inheritance Hierarchies not Defined |
|---|---|---|
| 12% | 45% | 49% |

# Difficulties in OO Modelling

- Students generally have difficulty identifying messages to be sent in UML sequence diagrams. They do not know how to fulfil the responsibilities of the use case by getting objects to pass messages to each other.
- Students also have difficulty understanding that the interaction diagrams are dependent on the analysis class diagram in terms of its classes, associations and multiplicities.
- Only 40% of sequence diagrams displayed some evidence of responsibilities delegated to the appropriate objects.
- None of the student diagrams fulfilled all the responsibilities of the use case.
- 52% of sequence diagrams did not include any parameters in the messages – we do not however consider this a serious design fault as parameters can be optionally defined in UML.

# Difficulties in OO Modelling

- Some of the most common errors committed by students when producing OOAD models:
  - assignment of a large business activity to a single object while it should be fulfilled through the collaboration with other objects;
  - missing responsibilities that should be assigned to objects;
  - missing objects that should participate in the overall responsibilities;
  - under-specified analysis models – the students were unable to identify most of the appropriate concepts; and
  - a large number of software concepts were identified at inconsistent abstraction levels.

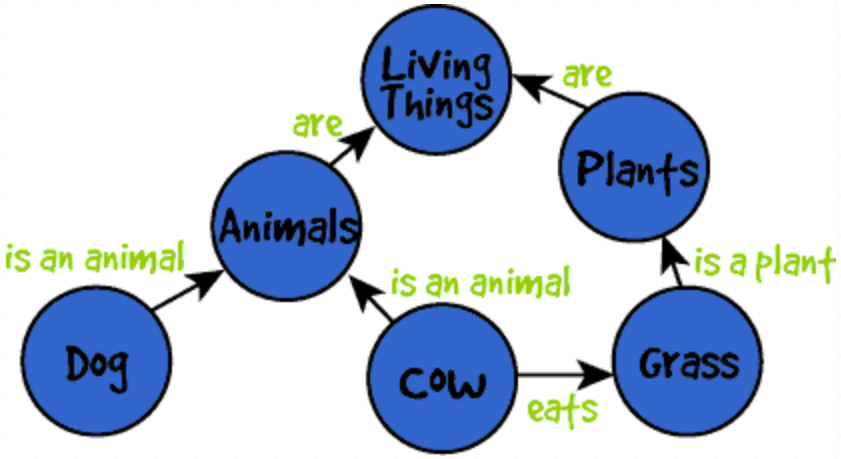
# Overview

1. Threshold Concepts
2. Object-Oriented Modelling
3. Threshold Concepts in OO Modelling
4. Evaluation: Changes in Level of Understanding
5. Conclusion

# What is Concept Mapping?

- In previous research we successfully adopted concept maps as a **stepping-stone** to assist students in developing analysis class and sequence diagrams.
- Concept mapping is a technique for representing knowledge in graphs.
- Concept maps are 2-dimensional, hierarchical diagrams that illustrate the knowledge within a problem domain.
- Developed by Joseph D. Novak in 1972

  at Cornell University.

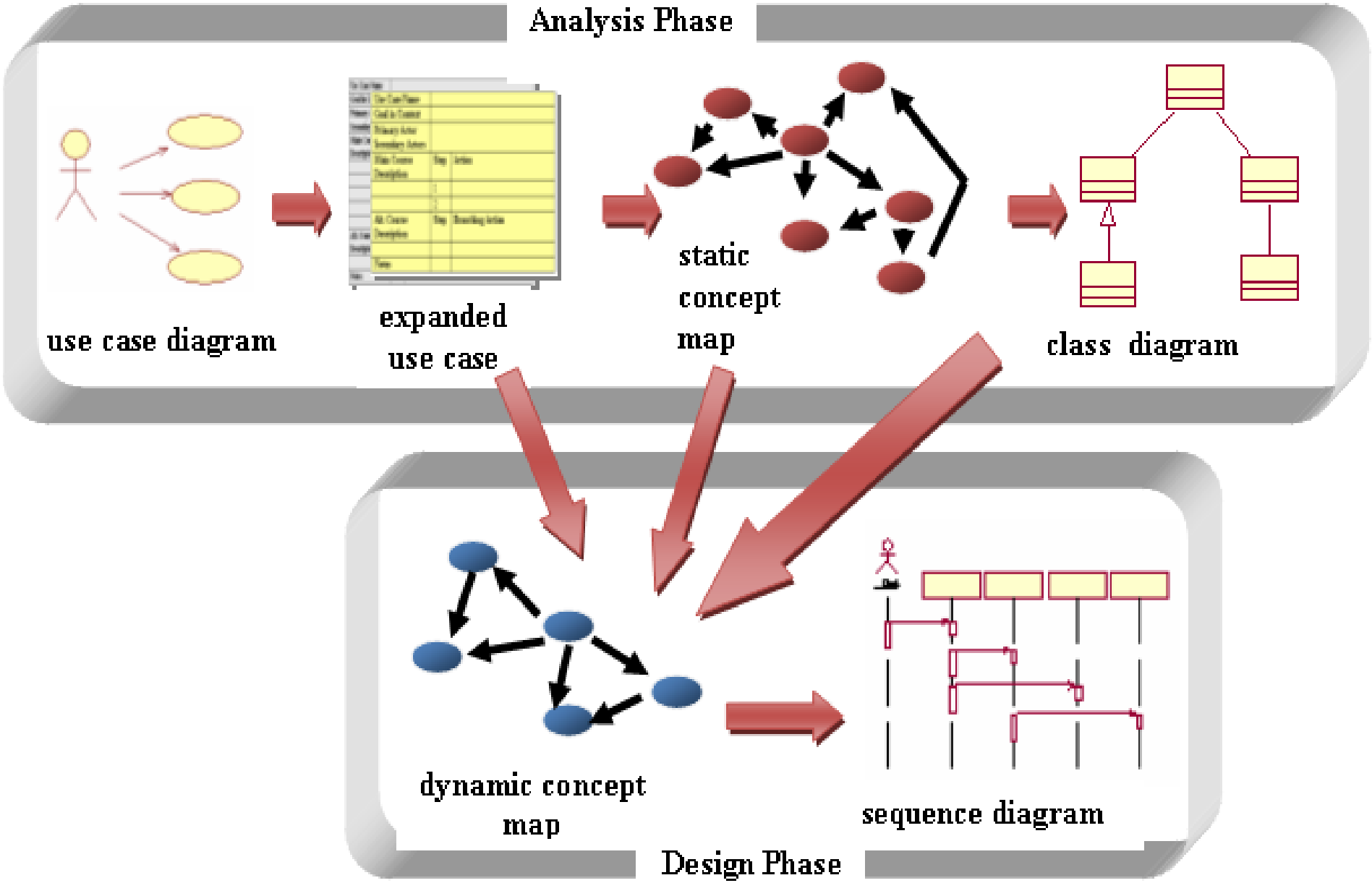


# Why Use Concept Maps?

We consider concept maps to have the following advantages:

- It is easier to distinguish between classes and attributes in concept maps by using specifically defined labelled links.
- It is easier to identify the different types of relationships e.g., associations and generalisation-specialisation hierarchies in concept maps by using specifically defined labelled links.
- Concept maps help clarify the meaning of a concept by means of propositions .
- It is relatively easy to teach concept maps. There are only two types of notations used in a concept map – nodes and links.

# The Concept-Driven Approach



# Evaluation: Changes in Level of Understanding

- We conducted two studies [Sie10] that reported the effects of using concept mapping to assist learners in OOAD produce class and sequence diagrams.
- Participants in Study 1 were not taught any concept mapping techniques.
- Participants in Study 2 were taught the concept mapping techniques and this study found a statistically significant reduction in the number of faults produced in UML class and sequence diagrams, particularly in terms of:
  - identification of expected classes representing the key concepts in the problem domain;
  - assignment of attributes to appropriate classes;
  - identification of appropriate generalisation-specialisation hierarchies; and
  - assignment of responsibilities to fulfil a particular scenario of a use case.

# Evaluation: Changes in Level of Understanding

- The results achieved by Study 2 participants may not be attributed solely to the effect of concept mapping – other contributory factors to quality improvement include:
  - As the students in Study 2 were currently enrolled in an OOAD course, their knowledge of OOAD concepts and experience in OO modelling was likely to be fresh in their minds.
  - The students in Study 2 may have been given a better foundation on OOAD concepts.
  - Students in Study 2 were given more time to produce the concept maps and UML models.

# Overview

1. Threshold Concepts
2. Object-Oriented Modelling
3. Threshold Concepts in OO Modelling
4. Evaluation: Changes in Level of Understanding
5. Conclusion

# Conclusion

- Classes
  - troublesome - the students experience difficulties in identify appropriate classes from the problem domain; and assigning appropriate attributes to classes.
  - irreversible - once the concept is understood, it cannot be easily forgotten.
  - transformative and integrative - knowledge gained can assist in developing the logical design of databases.

# Conclusion

- Generalisation-specialisation hierarchies. Novices usually find defining hierarchies
  - Troublesome - they have problems grouping real-world objects in terms of classification because they are not used to grouping objects in hierarchies. This is not intuitive enough to be mastered without training [RA96].
  - Irreversible - knowledge gained is irreversible (once the concept is understood, students will invariably find it easier to identify generalisation-hierarchies).
  - Transformative and integrative - this hierarchy can be used in use case diagrams and data base models.

# Conclusion

- Object interactions. Students in general have found this topic to be
  - Troublesome to learn - difficult to understand.
  - Irreversible - knowledge gained is irreversible (once the concept is understood, students will invariably find it easier to delegate the necessary responsibilities of the objects to fulfil a particular scenario of a use case).
  - Transformative and integrative – knowledge gained can be used in OOP.

# Conclusion

- Meyer and Land [ML03] suggest that once a student has been introduced to a threshold concept, he/she enters a state of '*liminality*'– a state associated with being 'stuck' and not possessing a mastery of the concept – until the necessary transformation of understanding has taken place and the 'threshold' is crossed.
- Students who have problems producing analysis class diagrams will first need to cross the threshold in identifying appropriate classes (representing real-world objects) before they can assign attributes and relationships (e.g. associations, generalisation-specialisation hierarchies and whole-part hierarchies) to the classes.
- These thresholds have to be crossed before the students can successfully produce appropriate sequence diagrams.

# References


- [BL06] N Bolloju & F Leung. Assisting Novice Analysts in Developing Quality Conceptual Models with UML. Communications of the ACM 49(7), pp. 108-112, 2006.
- [BEM+07] J Boustedt, A Eckerdal, R McCartney, JE Mostrom, M Ratcliffe, K Sanders, C Zander. Threshold concepts in computer science: do they exist and are they useful? In *Proc. 38th SIGCSE Technical Syymposium on Computer Science Education*. 2007.
- [EMM+06] A Eckerdal, R McCartney, JE Moström, M Ratcliffe, K Sanders, C Zander. Putting Threshold Concepts into Context in Computer Science Education. In *Proc. 11th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*. 2006.
- [ET05] A Eckerdal, M Thuné . Novice Java Programmers' Conceptions of "object" and "class", and Variation Theory. In *Proc. 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*. 2005.
- [ML03] JHF Meyer, R Land. Threshold Concepts and Troublesome Knowledge (1): Linkages to Thinking and Practising within the Disciplines. Improving Student Learning: Improving Student Learning Theory and Practice - Ten Years On. Oxford: OCSLD, pp. 412-424, 2003.
- [RA96] C Ryan & G Al-Qaimari. A Cognitive Perspective on Teaching Object Oriented Analysis and Design. http://citeseer.ist.psu.edu/cache/papers/cs/3179/http:zSzzSzgoanna.cs.rmit.edu.auzSz~ghassanzSzrmit.pdf/a-cognitive-perspective-on.pdf
- [RR09] J Rountree, N Rountree. Issues Regarding Threshold Concepts in Computer Science. In *Proc. 11th Australasian Conference on Computing Education.* 2009.
- [SBG06] D Svetinovic, DM Berry, MW Godfrey. Increasing Quality of Conceptual Models: Is Object-Oriented Analysis that Simple? In *Proc. 2006 International Workshop on Role of Abstraction in Software Engineering*. 2006.
- [Sie10] VY Sien, Teaching Object-Oriented Modelling using Concept Maps. In *Proc. 6th Educators' Symposium: Software Modeling in Education at MODELS 2010*.

# THANK YOU