SOFTENG 701:
Advanced Software Engineering Development Methods
Part 2

Lecture 5a: Inheritance and its use

Ewan Tempero
Department of Computer Science

Agenda

- Admin
  - No lecture on Friday (graduation)
  - Assignment 4?
- Part 1
  - Metrics for “measuring” use of inheritance
  - How much inheritance is used in practice (Or at least open-source Java systems)?
- Part 2
  - Metrics for “OO design”, according to Robert Martin

How Much Inheritance?

- Inheritance is good, therefore good designs must have lots of inheritance
  - Three fundamental principles of all object-oriented software: encapsulation, inheritance, and polymorphism (any number of sources)
  - "The [second] step in learning object-oriented programming is organizing classes into a hierarchical structure based on the concept of inheritance" Budd, **Introduction to Object-Oriented Programming**
  - Two of the "Seven steps towards object-based happiness" (Meyer, **Object-oriented software construction**) are *inheritance* and *Multiple and repeated inheritance*

But!

- Gang of Four to exhort us to “Favor object composition over class inheritance”
- ”As a rule of thumb, we tend to build lattices that are balanced and that are generally no deeper than 7 ± 2 classes and no wider than 7 ± 2 classes” Booch, **Object-oriented design and analysis with applications**
- “Most good designers avoid implementation inheritance (the extends relationship) like the plague.” Holub *Why extends is evil* **JavaWorld**

And!

- J. Daly, A. Brooks, J. Miller, M. Roper, and M. Wood. Evaluating inheritance depth on the maintainability of object-oriented software. Empirical Software Engineering, 1(2):109-132, Jan. 1996.

  Three levels of inheritance easier to maintain than zero levels, five levels takes longer than both
- M. Cartwright. An empirical view of inheritance. Information and Software Technology, 40:795-799, 1998.

  Inheritance has a positive effect on maintenance
- R. Harrison, S. Counsell, and R. Nithi. Experimental assessment of the effect of inheritance on the maintainability of object-oriented systems. Journal of Systems and Software, 52:173-179, 2000.

  Zero levels is easier to maintain than three or five levels

Inheritance and Good Design

- Claim: Good designs must have lots of inheritance
  - There must be some good designs, so some designs must have lots of inheritance
  - Some designs must have some inheritance

Inheritance and Good Design

- Claim: Good designs must have lots of inheritance
  - There must be some good designs, so some designs must have lots of inheritance
  - Some designs must have some inheritance

  $\Rightarrow$ How much inheritance does a design have?

Inheritance Measurements

Present slides for the following here (`TemperoECOOP08.pdf`)

E. Tempero, J. Noble, and H. Melton. “How do Java programs use inheritance? an empirical study of inheritance in Java software” In J. Vitek, editor, *22nd European Conference on Object-Oriented Programming (ECOOP)*, pages 667-691, Paphos, Cyprus, July 2008. Springer Berlin / Heidelberg.

Threats to Validity

- Are the metrics measuring what we think they are measuring?
- Is the corpus representative of use of inheritance?
- Is the use of inheritance observed “proper” use of inheritance?

Replication — Python (2015)

- Matteo Orr'u, Ewan Tempero, Michele Marchesi and Roberto Tonelli "How Do Python Programs Use Inheritance? A Replication Study", *Asia-Pacific Software Engineering Conference (APSEC)*. 2015
- 51 open-source Python systems
- DUI: median of 55%
- IF: median of 22%

Key Points

- Definitions of DIT, NOC are incomplete for Java
- While there are some very deep classes, they are rare
- While there are some classes with many children, they are rare
- Measurements relating to inheritance for individual classes (e.g. DIT, NOC) do not tell us about overall use of inheritance
- DUI and IF indicate degree to which developers choose to use inheritance