Material and some slide content from:

- Mehdi Amoui Kalareh
- Derek Rayside
- Steve Easterbrook
- David Budgen

Design Qualities Reid Holmes

Software design

- Designers control risk by limiting unknown factors
 - e.g., to reduce risk they can:
 - Compose existing components in a novel way
 - Compose new components in a known way
- All software quality measures are relative
 - ▶ e.g.,



Software design

- Predominant quality concern: fitness for purpose
 - Does it do what it should?
 - Does it do it as the users require?
 - Is it reliable / safe / fast / secure enough?
 - Is it affordable? Will it be finished on time?
 - Can it be adapted in the future?
- Questions are intermingled between software and its domain



Measuring quality

REID HOLMES - SE2: SOFTWARE DESIGN & ARCHITECTURE



Quality attributes

Simplicity

- There are two ways of constructing a software design. One way is to make it so simple that there are no obvious deficiencies. And the other is to make it so complicated that there are no obvious deficiencies." -- Hoare [1981]
- Meets goals without extraneous embellishment
- Measured by its converse --> complexity



Spotting incoherency

- An operation's description is full of 'and' clauses:
 - ▶ e.g.,

- Results in temporal cohesion, logical cohesion
- An operation's description has many 'if..then..else'
 - ▶ e.g.,

 Results in control coupling, coincidental cohesion, logical cohesion

Coupling



Cohesion



