# Software Quality

Good
Cheap
Fast

Pick any two

# Today's Lecture

1. Intro to Software Engineering
2. Inexact quantities
3. Error propagation
4. Floating-point numbers
5. Design process
6. Teamwork
7. Project planning
8. Decision making
9. Professional Engineering
10. Software quality
11. Software safety
12. Intellectual property

# Agenda

- **Quality attributes**

- **Measurable quality attributes**

- **Conflicting attributes**

- **Quality assurance**

- **SE Orientation**

# Quality Attributes

**Functional requirements** define the *functionality* of the software.

- What services should the software provide?
- What tasks should the software perform?

**Quality attributes** *qualify* acceptable software solutions.

- What characteristics should the software possess?
- What constraints are placed on the software?

**Example:** On an exam, you have 20 minutes to write a 1-page essay on the history of Canadian citizenship.

# Example: TrackerRobot Requirements

**Functional requirements** define the *functionality* of the software.
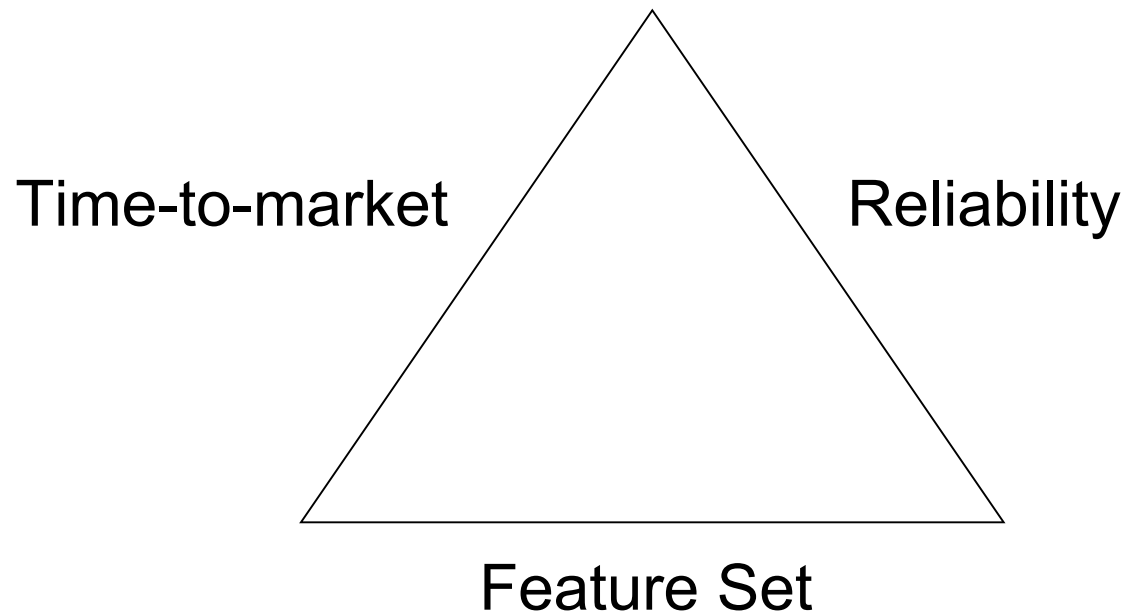
- 
- 
- 

**Quality attributes** *qualify* the acceptable software solutions.

- 
- 
-

# Other Quality Attributes

- 
- 
- 
- 
- 
- 
-

# Example:  Microsoft Project Triangle

Time-to-market                    Reliability

Feature Set

# Exercise

I am the mayor of the Swiss village of Wengen, and I would like a transportation device for tourists.

You are the designer I have hired.

What questions do you ask to tease out the quality attributes of an acceptable solution?

# Specifying Quality Attributes

**Motherhood specifications**
- Fast            *slow?*
- Reliable        *unreliable?*
- User-friendly   *unfriendly?*
- Maintainable    *unmaintainable?*

**Measurable specifications**

- Fast: Software responds to a query within 1 sec.

- Reliable: Mean-time-to-failure is $10^9$ sec.

# Example: TrackerRobot Requirements

**Measurable specifications**

- Time to delivery
- Robust
- Fast
- Memory-efficient
- Adaptable

# Example: Other Quality Attributes

**Measurable specifications**

- Reliable
- Available
- User-friendly
- Easy to learn
- Maintainable
- Fun

# Conflicting Requirements

Many quality requirements **conflict** with one another, which means it is impossible to build a product that maximizes all attributes.

- "Good, cheap, fast - pick any two"
- 
- 
- 

Conflicting requirements must be **prioritized.**

# Conflicting Requirements

- Safety vs. Economy (most prevalent conflict)

*"Anyone can build a bridge.*

*It takes an engineer to build a bridge that barely stands."*

*[unknown source]*

# Summary

- **Quality requirements** play a major role in defining what attributes a software solution must have to be acceptable.

- A software's quality requirements must be specified in a way that they are **measurable.**

- Conflicting requirements must be **prioritized.**

# Announcements

- **Lab this week:** Workplace ethics workshop

- **Scheduling exercise Part II**
      Due Thursday at 4:30 p.m.

- For **next week's lecture and web review**:
      **Safety, Risk** IPE Ch. 19, 20
      **Word Choice** Dupré 14,61,71,75,99,129

- **Ethics memo** due Thursday Nov. 25 at 4:30 p.m.

# Announcements

**No Office Hours on Thursday**