CSC 436, Fall 2017

# Project Planning: Estimation

*Ravi Sethi*

THE UNIVERSITY OF ARIZONA

# Starting Point

## How do you keep a project on track?

- **You have a prioritized list of tasks**
  - Entered into the tracking system
  - Each task has an effort estimate
- **You have selected which tasks go into the release**
  - "Iceberg list," with top *n* items in this release
  - Additional requests from the customer push other items below the waterline
  - Meanwhile, faster expected progress lifts items up above the waterline

# Software Engineering Management

## Builds on traditional Project Management

- **Shared with traditional project management**
  - Constraints: cost, schedule, scope, risk, resources, …
  - Uncertainty and Estimation
- **Challenges unique to software**
  - Clients often don't know what's needed or what's feasible
  - Clients lack appreciation of the impact of changing requirements
  - Often, high degree of novelty and complexity
  - Maintaining a balance between creativity and discipline
  - Rapidly changing underlying technology

THE UNIVERSITY OF ARIZONA®
Computer Science

# Origins of Modern Project Management?

## Crystal Palace built between July 1850 and May 1851


https://commons.wikimedia.org/w/index.php?curid=48718934

# Project Management



- **Project**
  - A set of activities with a start, a finish, and deliverables
  - Subject to schedule and cost constraints
- **Software-related examples of deliverables**
  - A product; e.g., a web browser
  - An event; e.g., land a rover on Mars
  - A service delivered from the cloud
  - An assessment; e.g., from an investigation into a security breach

# Role of Estimation

THE UNIVERSITY OF ARIZONA
Computer Science

## Estimation is a key part of Project Planning

- **Project Management includes**
  - Planning, Organizing, Tracking, Controlling
- **Project Planning involves**
  - Selection of a software development process
  - Work assignment
  - Size, schedule, and cost estimation
  - Risk assessment
  - Quality planning

# Project Planning: Advance and Adaptive

- **Advance Planning**
  - Geared to plan-driven processing
  - Up-front, before design, coding, testing
  - Careful planning, based on best available information
  - Unfortunately, requirements change
- **Adaptive Planning**
  - Geared to iterative and agile processes
  - Spread evenly across iterations
  - Information improves as a project progresses
  - Short planning horizon for each incremental plan

# Constraints

# Project Control

## Initially, a manager needs good estimates

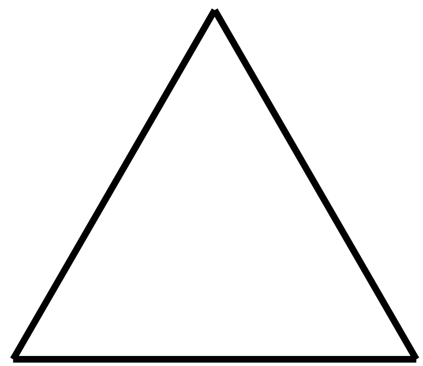
- **A manager can control 4 things**
  - Resources: Can get more dollars to pay for personnel, investment, …
  - Time: Can increase schedule, delay milestones, …
  - Product: Can reduce functionality; e.g., scrub requirements
  - Risk: Can decide which risks are acceptable
- **To do this, a manager needs to keep track of**
  - Effort: How much further effort? How much has been expended?
  - Time: What is the expected schedule? How far off are we?
  - Size: How big is the planned system? How much have we built?
  - Defects: How many errors have we detected? How many fixed?

# The Iron Triangle



## Also known as the Project Management Triangle

**Scope**
**with Quality**



**Time**
**or Schedule**

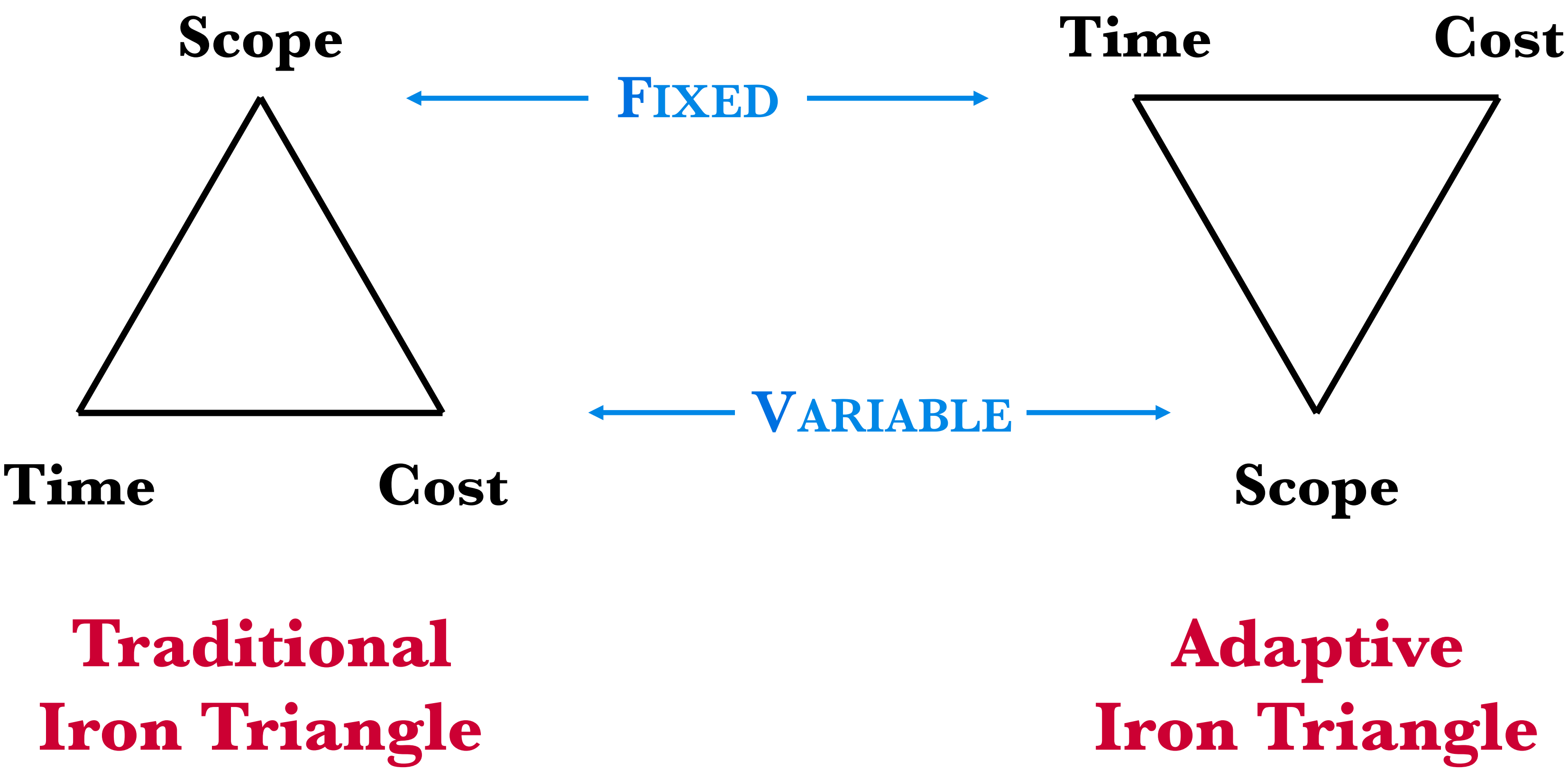
**Cost**
**or Budget**

## Pick Any Two!

# Project Management Triangles

THE UNIVERSITY OF ARIZONA
Computer Science

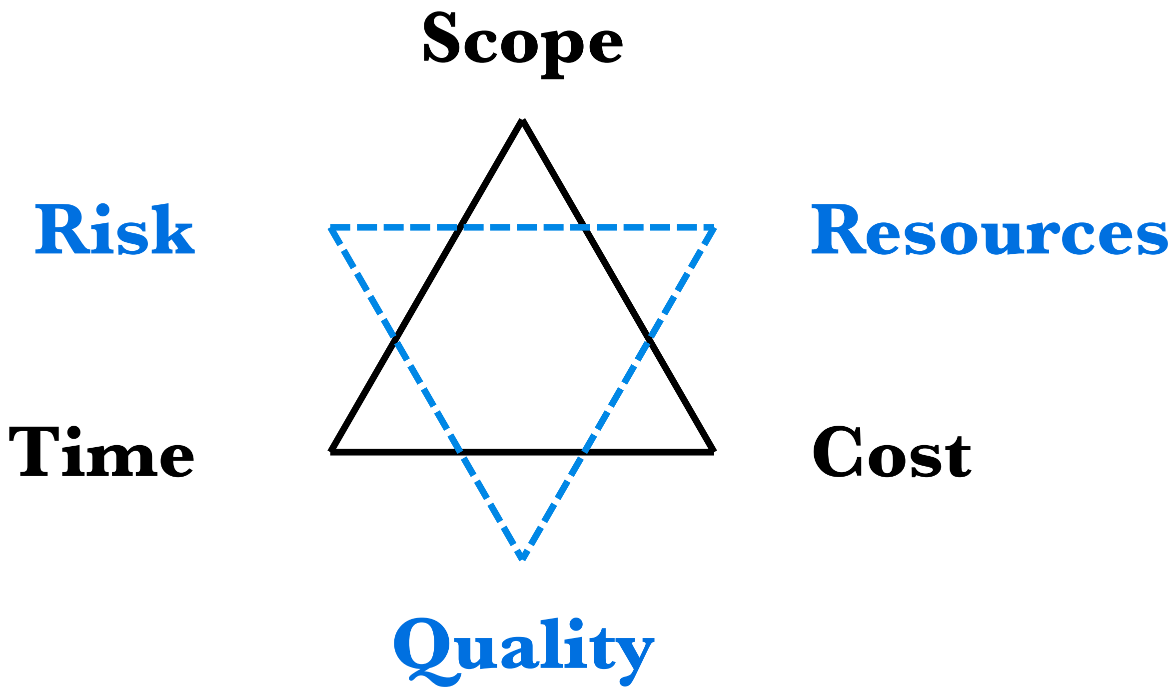
## For plan-driven and iterative development



# Project Management Triangle: Variant

THE UNIVERSITY OF ARIZONA
Computer Science

## Highlighting Quality, Risk, and Resources



# Anchoring and Cognitive Bias

# Definition of Anchoring

## Daniel Kahneman and Amos Tversky

**“In many situations, people make estimates by starting from an initial value that is adjusted to yield the final answer. … different starting points yield different estimates, which are biased toward the initial values.  We call this phenomenon anchoring.”**

# Anchoring

THE UNIVERSITY OF ARIZONA
Computer Science

## Case Study: A Software Estimation Exercise

- **Participants asked to estimate time to deliver an application**
  - Given a 10-page requirements document
  - And a 3-page "project setting" document about
    - The client organization (including quotes from interviews)
    - The developers (experience, previous performance, team dynamics)
- **Participants were in 3 groups**
  - Only difference was a paragraph on the 2nd page of the project setting document

# Anchoring



## Case Study: A Software Estimation Exercise

- **Control group: quote with no mention of time**

  ***"I admit I have no experience estimating. We'll wait for your calculations for an estimate."***

- **Second group: quote mentioned "2 months"**

  ***"I admit I have no experience with software projects, but I guess this will take about 2 months to finish. I may be wrong of course, we'll wait for your calculations for a better estimate."***

- **Third group: quote mentioned "20 months"**

  ***"I admit I have no experience with software projects, but I guess this will take about 20 months to finish. I may be wrong of course, we'll wait for your calculations for a better estimate."***

Aranda and Easterbrook [2005]

# Group Activity

THE UNIVERSITY OF ARIZONA
Computer Science

**Based on the information provided, what is your estimate?**

- **Participants were not told the purpose of the study**
  - 23 participants, all with either industrial or academic experience
- **57% had estimation experience with real projects**
  - 22% with medium to large software projects
  - 35% with small projects
- **Control Group Estimate (experienced participants)**
  - Mean: 9.0 months

Aranda and Easterbrook [2005]

# Anchoring

THE UNIVERSITY OF ARIZONA
Computer Science

## Case Study: Results of the experiment

- **Mean Estimates from Experienced Participants**
  - Control group: 9.0 months
  - 2-month group: 7.8 months
  - 20-month group: 17.8 months
- **Mean Estimates Including All Participants**
  - Control group: 8.3 months
  - 2-month group: 6.8 months
  - 20-month group: 17.4 months
- **An anchor (e.g., 2 months or 20) can lead to bias**

Aranda and Easterbrook [2005]

# Biased Estimates

THE UNIVERSITY OF ARIZONA
Computer Science

## Case Study: Discussion

- **Possible reasons for the small difference between the control group and the 2-month anchor group**
  - Estimators may be optimistic by nature and the control group may have had external anchors
  - The 2-month value may not have been low enough
  - A greater pool of participants may be needed to separate results for the two groups
  - Low anchors may not affect estimation processes as powerfully as high anchors

Aranda and Easterbrook [2005]

# Estimation Uncertainty

# Estimation



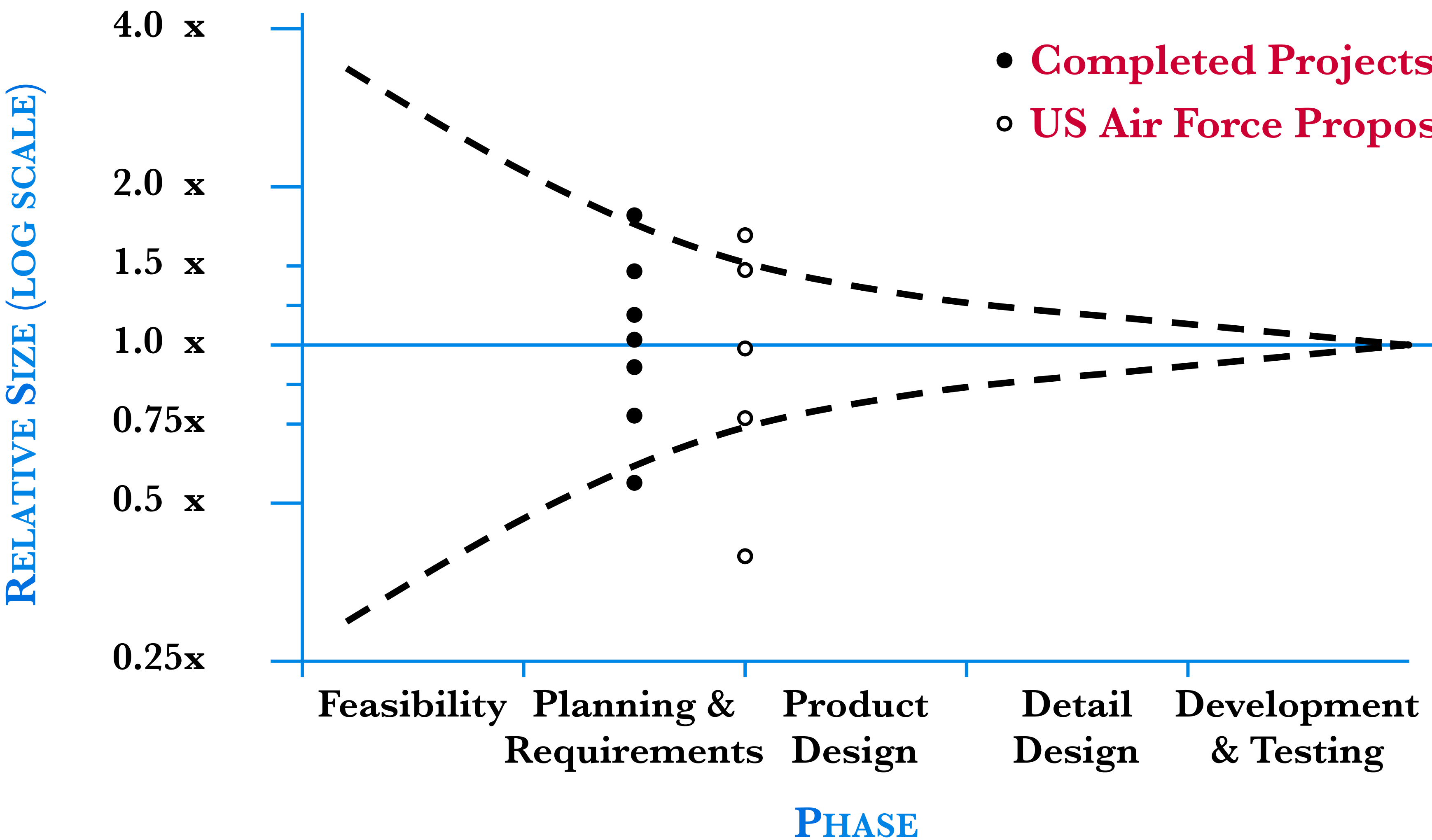
## Some Complications

- **Not all requirements are comparable**
  - e.g., different levels of abstraction
  - e.g., core functionality vs. customer enhancements
- **Requirements may not be independent**
  - No selecting between *x* and *y* if they are mutually dependent
- **Stakeholders might not agree**
  - Different cost/value assessments for different types of stakeholder
  - Even for a given stakeholder, if $x > y$ and $y > z$, then is $x > z$?

# Cone of Uncertainty: Conceptual Diagram



## Inherent uncertainty, by phase

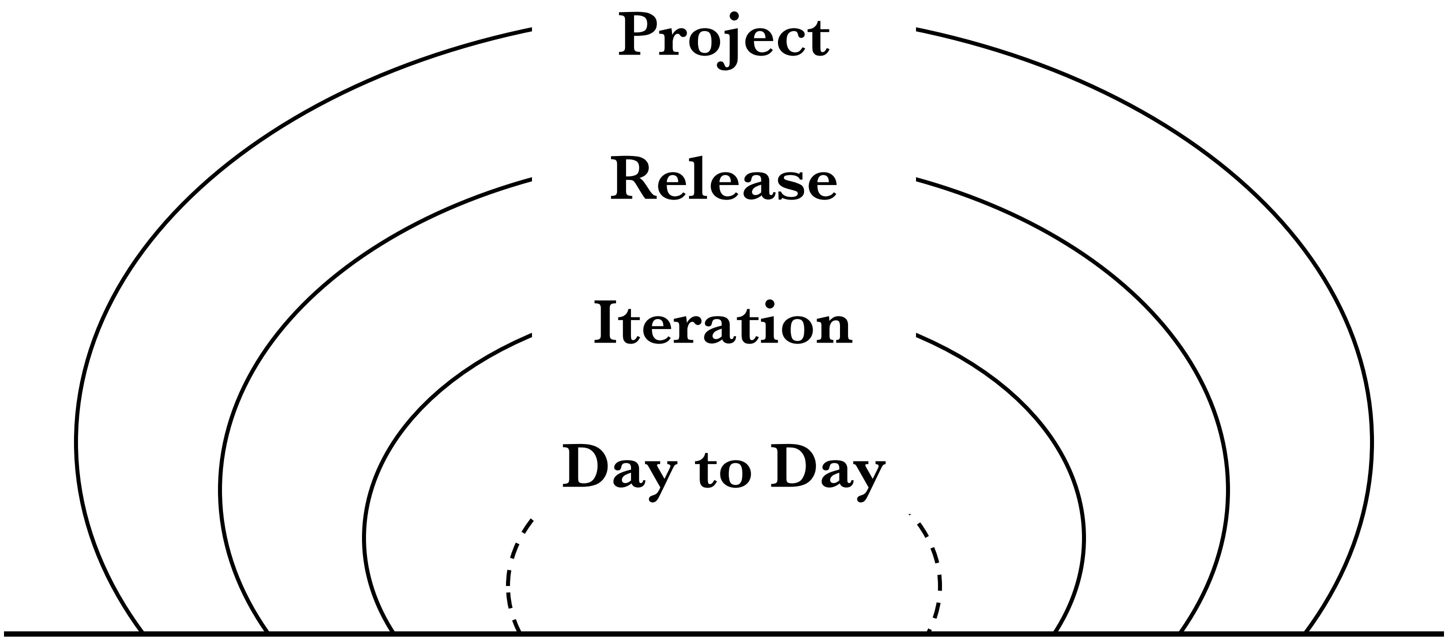



Barry Boehm [1981] (dashed lines); Boehm et al. [1995] (data)

# Levels of Uncertainty: Adaptive Planning



**The smaller the oval, the lower the uncertainty**




**The shorter the planning horizon, the lower the uncertainty**

Cohn [2006]

# Three-Point Estimation

THE UNIVERSITY OF ARIZONA
Computer Science

- **Better estimates than just asking for a range**
  - $w$ = worst possible case
  - $m$ = most likely case
  - $b$ = best possible case

    $estimate = (b + 4m + w)/6$

- **Based on a statistical model**
  - In the Model: $b$ is the 95th percentile case, $w$ the 5th
  - In the Model: Estimates for work items are independent

Moder, Phillips, and Davis [1983]

# Collective Judgment

*Consensus estimate from a group can be more accurate than individual estimates*

*Observation dates back to Aristotle*

# Best Buy

THE UNIVERSITY OF ARIZONA
Computer Science

## How many gift cards will they sell around Christmas?


- **Internal Experts**
  - 95% accurate
- **CEO asked 100 randomly selected employees**
  - 99.9% accurate
- **Both groups**
  - Were told actual sales for the past year

Smith and Sidky [2009]

# Group Consensus Estimates

THE UNIVERSITY OF ARIZONA
Computer Science

## Differ on how they address two issues

- **Avoiding cognitive bias**
  - Groups can be swayed by the "loudest voice" in the room
  - Approach: keep experts apart to get independent estimates
- **Converging on a consensus**
  - How to combine independent estimates?
  - Rand Approach: experts kept separate, moderator summarizes
  - Wideband Approach: group discussion

# Software Estimation: Delphi Method



## Named after Project Delphi at Rand Corp., circa 1960

- Assemble a group with varied experience in development and estimation: 3 to 7 estimators
- Provide group members with requirements or other project description
- Have each member independently estimate size and effort (or cost)
- Gather estimates and compare
- Ask for rationale from outliers
- If there is not close agreement, repeat 2-3 times

# Software Estimation: Delphi Method

## Experts remain anonymous (initials are not shared)

| Initials | Round 1 | | Round 2 | | |
|---|---|---|---|---|---|
| | Interval (weeks) | Size (LOC) | Interval | Size | Productivity |
| KI | 9 | 700 | 3 | 1,000 | 333 |
| KG | 3 | 500 | 4 | 2,000 | 500 |
| JC | 3 | 2,000 | 5 | 2,500 | 500 |
| BC | 3.5 | 1,200 | 4 | 2,500 | 625 |
| SB | 4 | 3,000 | 3.5 | 3,000 | 860 |
| EB | 4 | 1,000 | 4 | 2,000 | 500 |

# Wideband Delphi



## Basis for Planning Poker (agile estimation)

- **Group discussion of anonymous estimates**
  - Discussion can lead to valuable insights for design, coding, testing
  - Record issues during group discussion
  - Can help avoid pitfalls
- **Planning Poker**
  - Group is given cards marked with Fibonacci Story Points
  - Reveal cards simultaneously
  - Get rationale for outliers
  - Repeat until consensus is reached

# Case Study:
# Four Companies, Same Requirements

# Four Companies Develop the Same System

## Study by Simula Research Laboratory, Norway

- **35 companies submitted bids**
  - Same Requirements document
  - Call for tender sent to 81 companies; 35 of them submitted bids
- **4 to independently develop the "same" system**
  - Controlled requirements, team size, developer skill, technology environment so they were as similar as possible across the four projects
  - Kept interaction with the developers as similar as possible
- **Measured, after the systems in use for 2 years**
  - Cost (bids were firm-price), lead time, schedule overrun
  - Reliability, usability, maintainability

Anda, Sjøberg, and Mockus [2009]

# Four Companies Develop the Same System

THE UNIVERSITY OF ARIZONA
Computer Science

**The 4 finalists were chosen to have similar team size, developer skills, programming languages and customer-developer interaction**

**Tender sent to 81 companies**

**Bids submitted by 35 companies**

**Chosen for study: 4 companies**

### Well-Defined Requirements

### Bidding Information

- Resources
- Process

### Project Success Outcomes

- Costs
- Actual Days
- Schedule

### Project Quality Outcomes

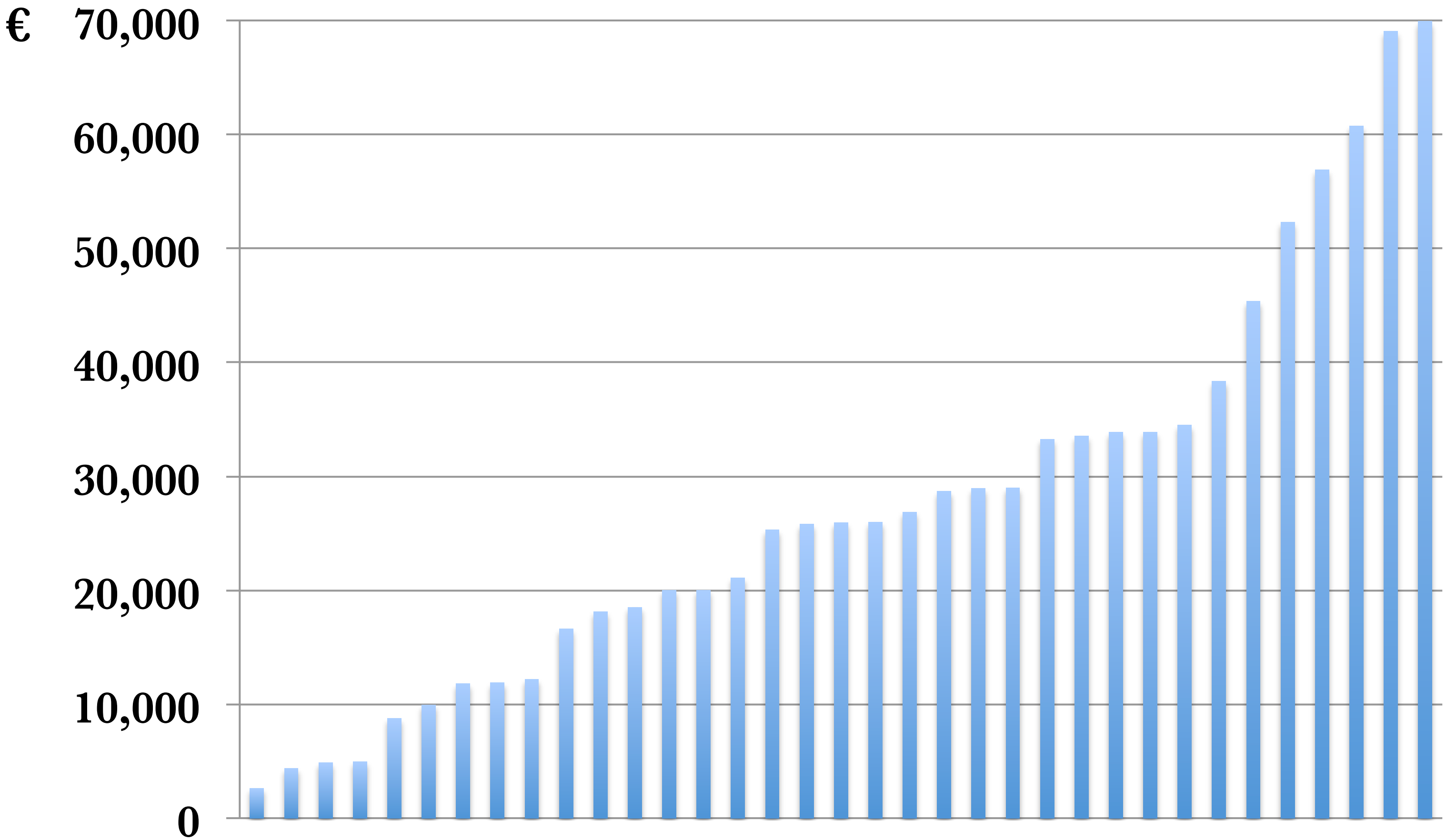
- Reliability
- Usability
- Maintainability

Anda, Sjøberg, and Mockus [2009]

# 35 Bids, based on the same requirements

THE UNIVERSITY OF ARIZONA
Computer Science

## Firm bids in Euros.  Proposed schedule range: 14 to 94 days




**Similar large variations in a study of software development projects, with 30 companies from 11 countries in Eastern Europe and Asia.**

Anda, Sjøberg, and Mockus [2009]

# Results of the Study

## Conducted by Simula Research Laboratory, Norway

| Company | A | B | C | D |
|---|---|---|---|---|
| Size | ~100 | ~25 | ~8 | ~13,000 |
| Firm Price | €20,000 | €45.380 | € 8,750 | €56,000 |
| Customer Cost | €25,370 | €51,860 | €18,020 | €61,070 |
| Agreed Schedule | 55 days | 73 days | 41 days | 62 days |
| Actual Days | 87 days | 90 days | 79 days | 65 days |
| Lines of Code | 7,937 | 14,549 | 7,208 | 8,293 |
| Classes | 63 | 162 | 24 | 96 |
| Reliability | Good | Good | Poor | Fair |
| Usability | Good | Fair | Fair | Good |
| Maintainability | Good | Poor | Poor | Good |

Anda, Sjøberg, and Mockus [2009]

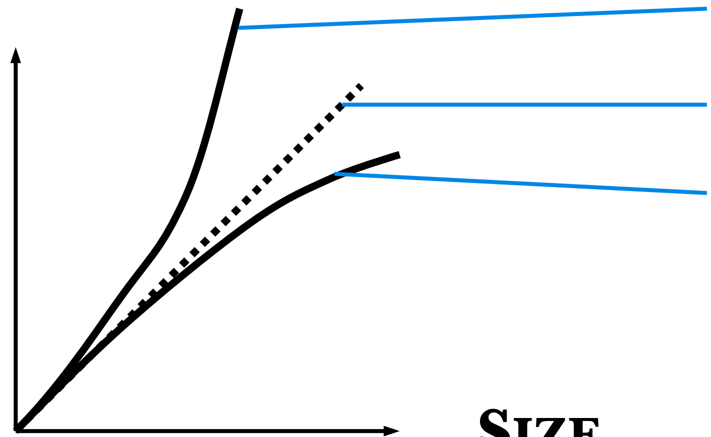
# Formal Models

*Estimating Effort from Size*

# How Does Effort Grow with Size?

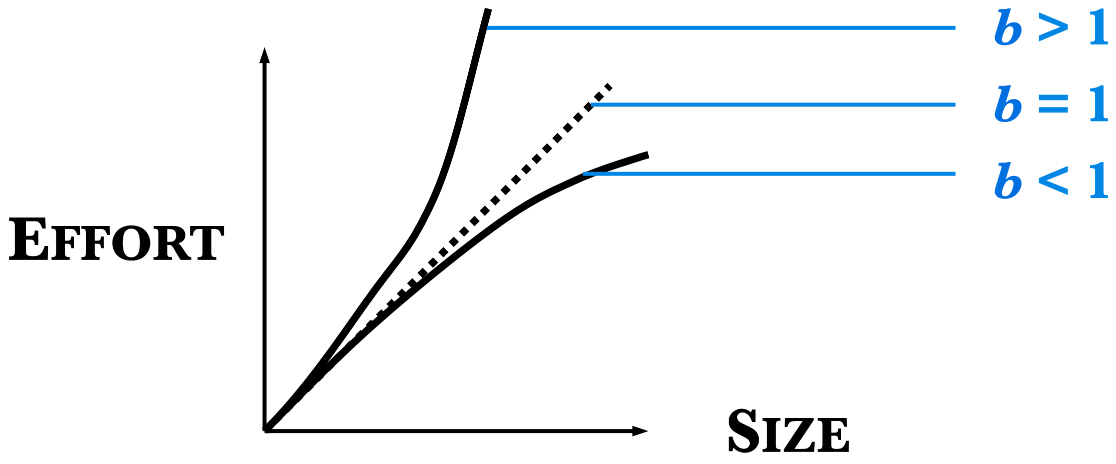


## Effort and size are related, but they are not the "same"



# How Does Effort Grow with Size?

**Model: $E = a\,S^{b}$**



# Empirical Observations

The University of Arizona
Computer Science

## From waterfall projects in the 1970s

- **IBM data**
  - $E = 5.2\ S^{0.91}$
- **TRW data**
  - $E = 2.4\ S^{1.05}$ Basic Systems
  - $E = 3.0\ S^{1.12}$ Intermediate
  - $E = 3.6\ S^{1.22}$ Embedded Systems

# Software Estimation: Cocomo

THE UNIVERSITY OF ARIZONA
Computer Science

## Boehm's Constructive Cost Model for Estimating Effort

- **Basic Cocomo Model:** $E = a\,S^{b}$
  - $E$ = Effort
  - $S$ = Lines of Code
  - $a$ and $b$ are project specific factors based on past experience
- **Rely on data from past projects with this team**
  - Terms like "small", "medium", "good", "mixed", "less", "rigid", … are subject to interpretation

# Software Estimation: Cocomo-81



## Boehm's Constructive Cost Model for Estimating Effort

- **In $E = aS^b$ factors $a$ and $b$ depend on the project**
  - Organic projects: small teams with good experience working with less than rigid requirements
  - Semi-detached projects: medium teams with mixed experience working with a mix of rigid and less than rigid requirements
  - Embedded projects: developed within a set of tight constraints
- **The "81" in Cocomo-81 is from 1981**
  - To distinguish it from later models in the Cocomo family

# Cocomo II: A Major Redesign

THE UNIVERSITY OF ARIZONA
Computer Science

## Late 1990s

- **Cocomo-81 lost its predictive power**
  - It was designed for waterfall projects
  - Does not work well for iterative processes
- **Cocomo II basic form: $E = a\,S^{b} + c$**
  - The Cocomo family differs in values for the parameters $a$, $b$, $c$
  - Parameter values are based on factors such as
    - Desired reliability
    - Use of tools and platforms
    - Plus many other factors

# Best Practices: Estimation

*"Overwhelming evidence documents a tendency toward cost and effort overruns in software projects. On average, this overrun seems to be around 30 percent. Furthermore, comparing the estimation accuracy of the 1980s with that reported in more recent surveys suggests that the estimation accuracy hasn't changed much since then."*

Jørgensen [2014]

# Improving Estimation Accuracy

THE UNIVERSITY OF ARIZONA
Computer Science

## Recommendations

- **Use tailored simple estimation models in combination with expert estimation**
- **Set min-max effort intervals based on historical estimation errors**
- **Avoid anchors**
- **Use customized checklists**
- **Avoid early estimates**