Software Engineering Economics (CS656)

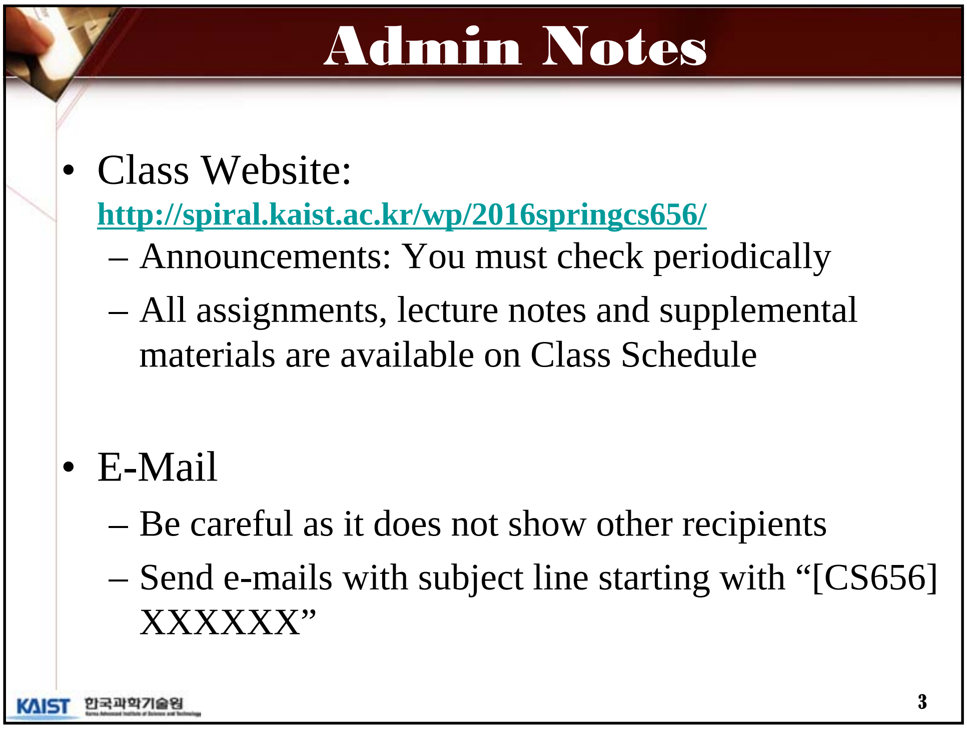
Overview

Jongmoon Baik

KAIST 한국과학기술원 SINCE 1971

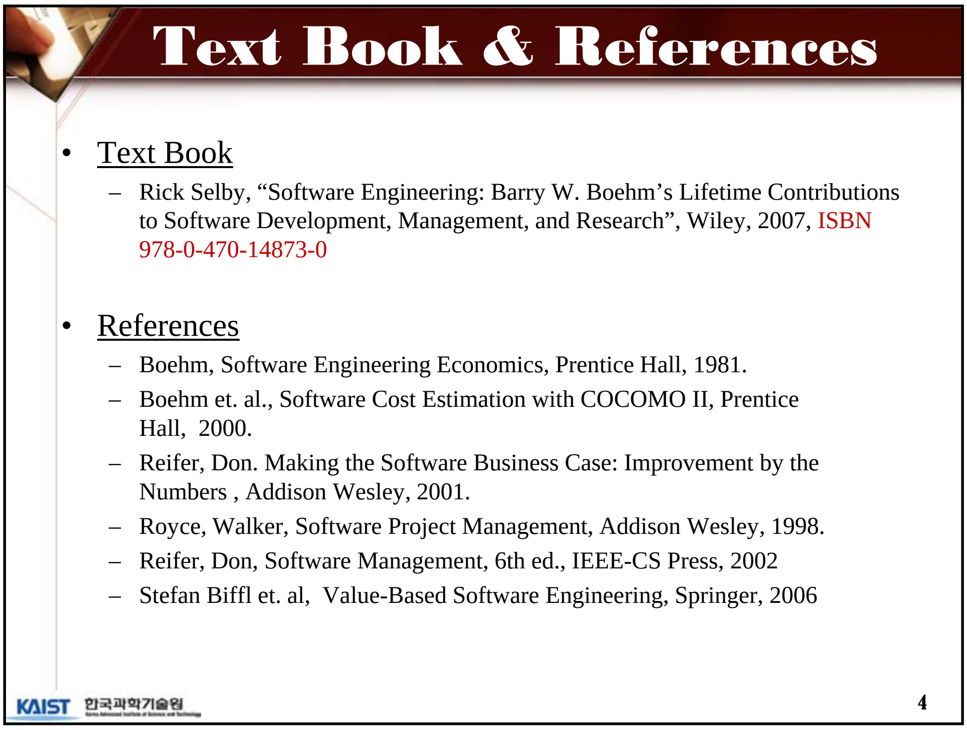
Class Information

- Instructor: Jongmoon Baik
  - Office: N1, Rm. 502
  - Phone: 042-350-3556
  - Email: jbaik@kaist.ac.kr
  - Office Hour: MON & WED: 10:30AM-12:00PM (or By Appointment)
- Class Info.
  - Class Hours: MON & WED 13:00 PM– 14:15 PM
  - Class Room: E3-1 Rm. 3445
  - We'll start on time with any questions and end on time
- Teaching Assistant: Duksan Ryu Email: dsryu@kaist.ac.kr
  - Office: N1, Rm. 525 , Office Hours: TBA
  - Tel: 042-350-7756



Admin Notes

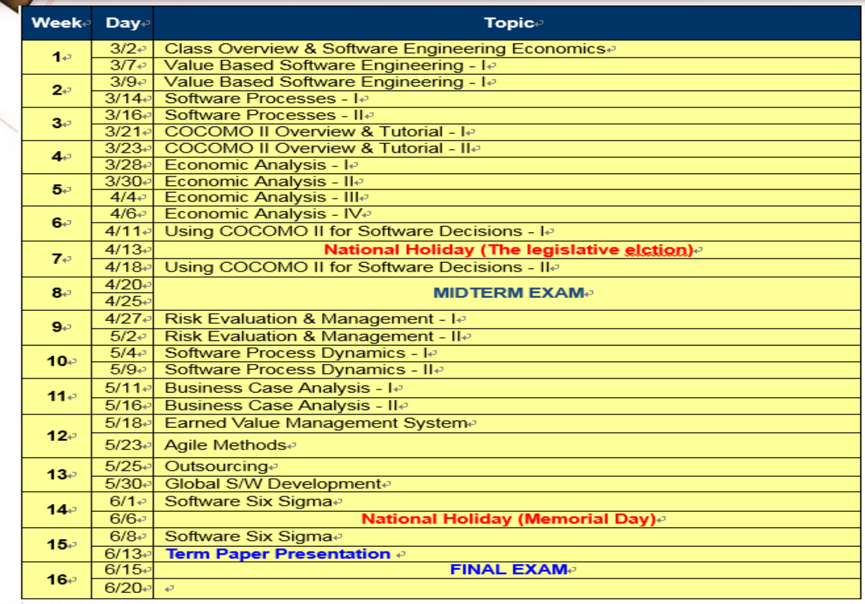
- Class Website: http://spiral.kaist.ac.kr/wp/2016springcs656/
  - Announcements: You must check periodically
  - All assignments, lecture notes and supplemental materials are available on Class Schedule
- E-Mail
  - Be careful as it does not show other recipients
  - Send e-mails with subject line starting with “[CS656] XXXXXX”



Text Book & References

- Text Book
  - Rick Selby, “Software Engineering: Barry W. Boehm’s Lifetime Contributions to Software Development, Management, and Research”, Wiley, 2007, ISBN 978-0-470-14873-0
- References
  - Boehm, Software Engineering Economics, Prentice Hall, 1981.
  - Boehm et. al., Software Cost Estimation with COCOMO II, Prentice Hall, 2000.
  - Reifer, Don. Making the Software Business Case: Improvement by the Numbers , Addison Wesley, 2001.
  - Royce, Walker, Software Project Management, Addison Wesley, 1998.
  - Reifer, Don, Software Management, 6th ed., IEEE-CS Press, 2002
  - Stefan Biffl et. al, Value-Based Software Engineering, Springer, 2006

Class Schedule



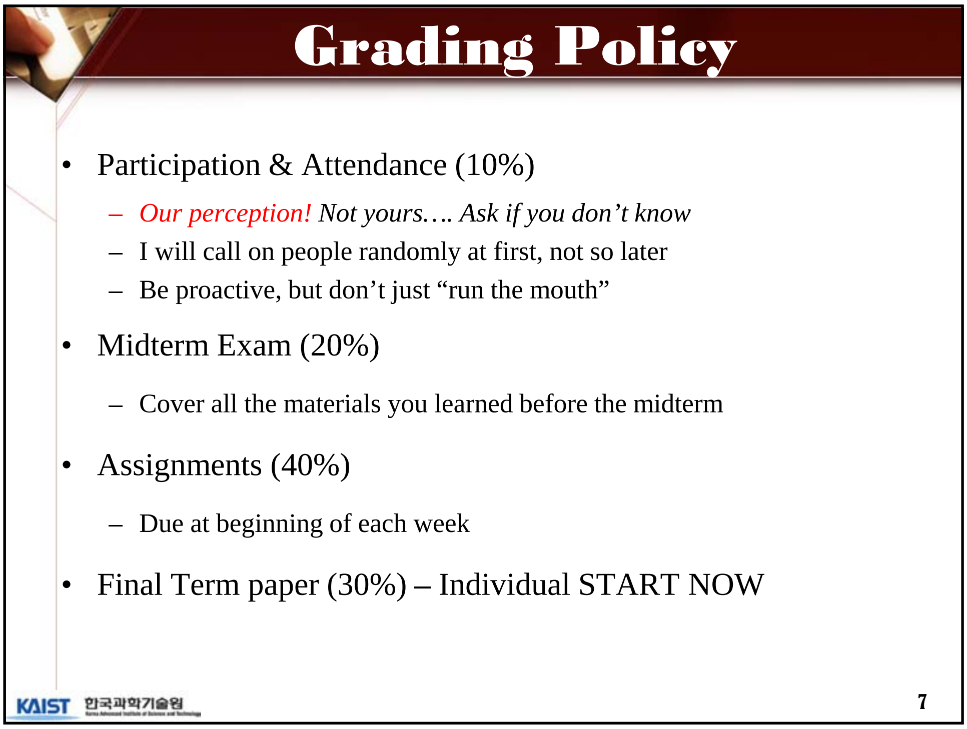
Week	Day	Topic
1	3/2	Class Overview & Software Engineering Economics
	3/7	Value Based Software Engineering - I
2	3/9	Value Based Software Engineering - I
	3/14	Software Processes - I
3	3/16	Software Processes - II
	3/21	COCOMO II Overview & Tutorial - I
4	3/23	COCOMO II Overview & Tutorial - II
	3/28	Economic Analysis - I
5	3/30	Economic Analysis - II
	4/4	Economic Analysis - III
6	4/6	Economic Analysis - IV
	4/11	Using COCOMO II for Software Decisions - I
7	4/13	**National Holiday (The legislative elction)**
	4/18	Using COCOMO II for Software Decisions - II
8	4/20	**MIDTERM EXAM**
	4/25	
9	4/27	Risk Evaluation & Management - I
	5/2	Risk Evaluation & Management - II
10	5/4	Software Process Dynamics - I
	5/9	Software Process Dynamics - II
11	5/11	Business Case Analysis - I
	5/16	Business Case Analysis - II
12	5/18	Earned Value Management System
	5/23	Agile Methods
13	5/25	Outsourcing
	5/30	Global S/W Development
14	6/1	Software Six Sigma
	6/6	**National Holiday (Memorial Day)**
15	6/8	Software Six Sigma
	6/13	**Term Paper Presentation**
16	6/15	**FINAL EXAM**
	6/20	

Course Structure

- Lectures
  - Check for slides DAY OF LECTURE please
  - We try to update all the time
  - Not all the supplemental materials will be on Class Website due to copyright protections

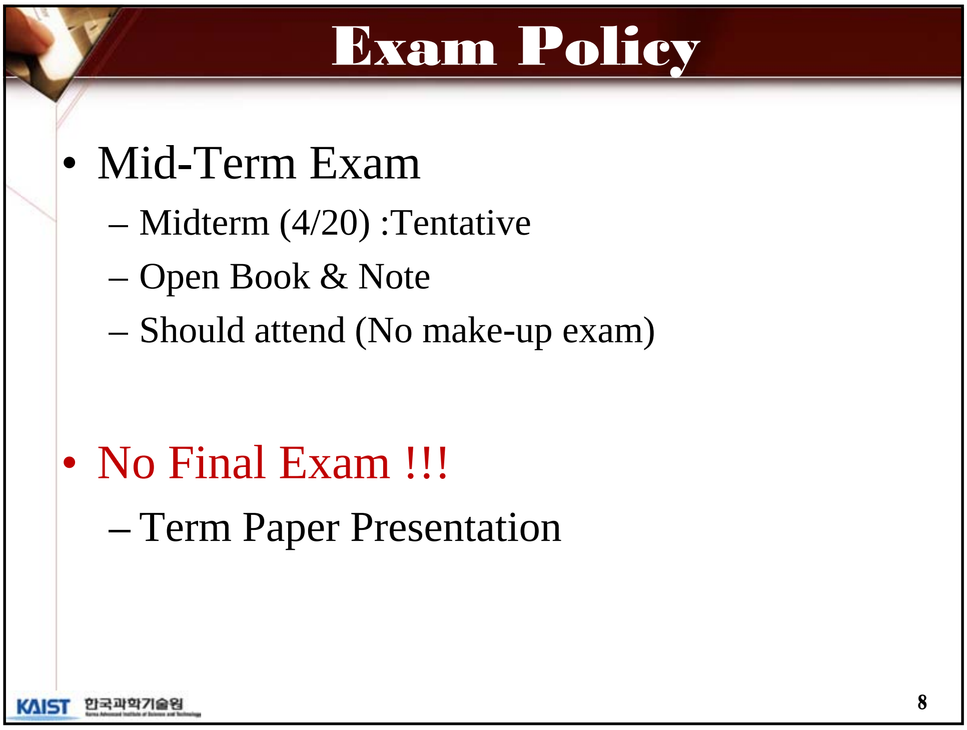
  - Proactive participation is critical
  - Ask questions
  - Prepare questions ahead of class

Challenge the instructors and each other!



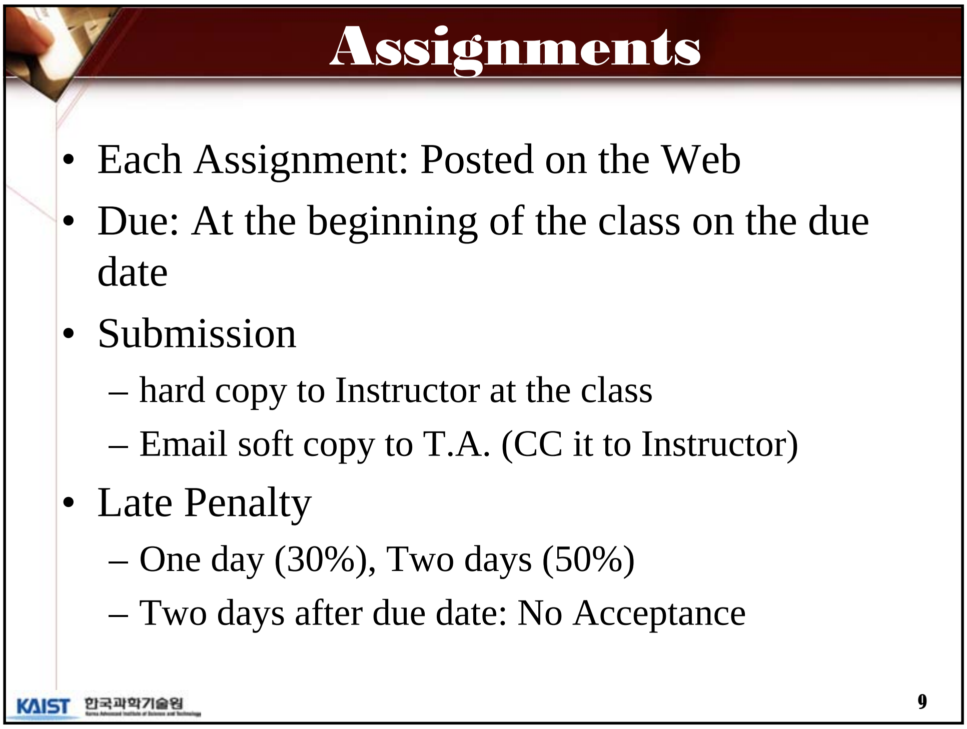
Grading Policy

- Participation & Attendance (10%)
  - *Our perception! Not yours…. Ask if you don't know*
  - I will call on people randomly at first, not so later
  - Be proactive, but don't just "run the mouth"
- Midterm Exam (20%)
  - Cover all the materials you learned before the midterm
- Assignments (40%)
  - Due at beginning of each week
- Final Term paper (30%) – Individual START NOW



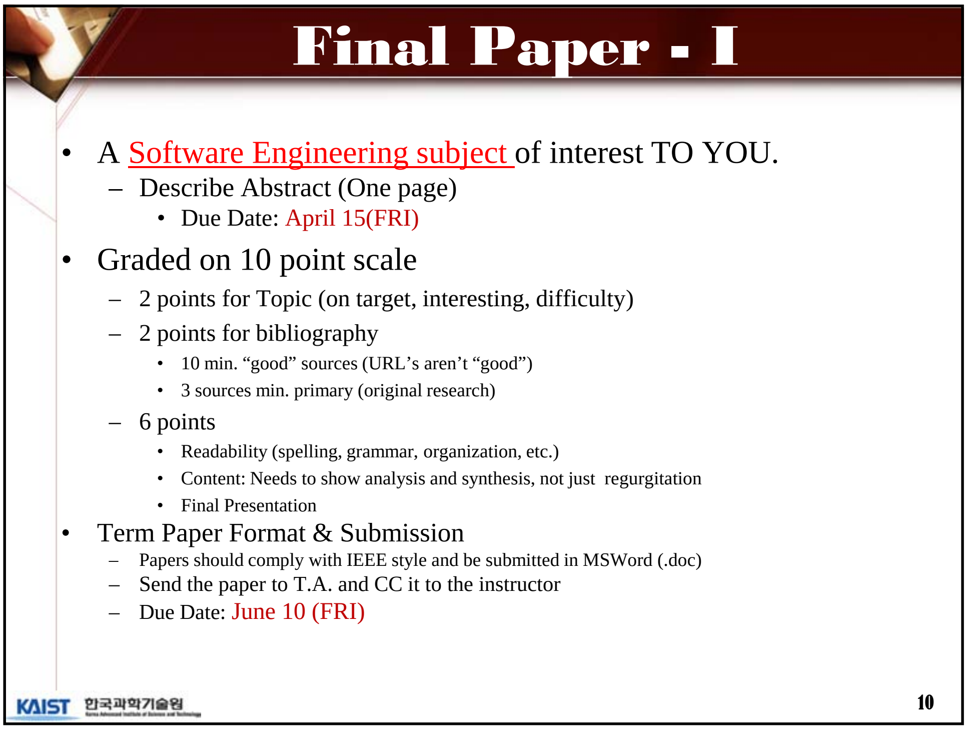
Exam Policy

- Mid-Term Exam
  - Midterm (4/20) :Tentative
  - Open Book & Note
  - Should attend (No make-up exam)
- No Final Exam !!!
  - Term Paper Presentation



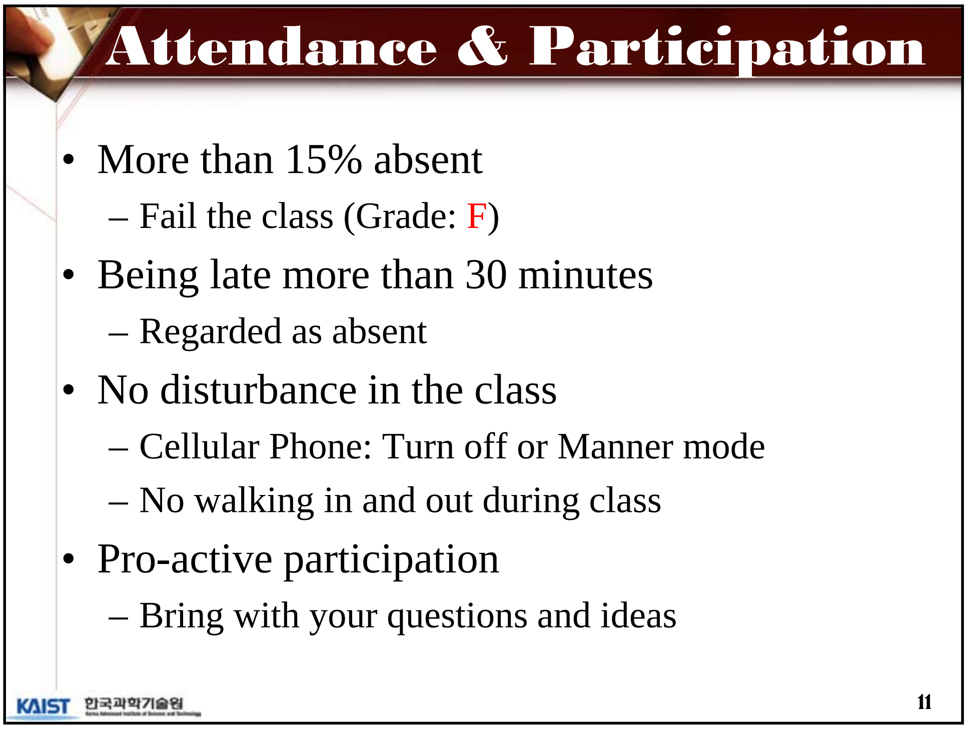
Assignments

- Each Assignment: Posted on the Web
- Due: At the beginning of the class on the due date
- Submission
  - hard copy to Instructor at the class
  - Email soft copy to T.A. (CC it to Instructor)
- Late Penalty
  - One day (30%), Two days (50%)
  - Two days after due date: No Acceptance



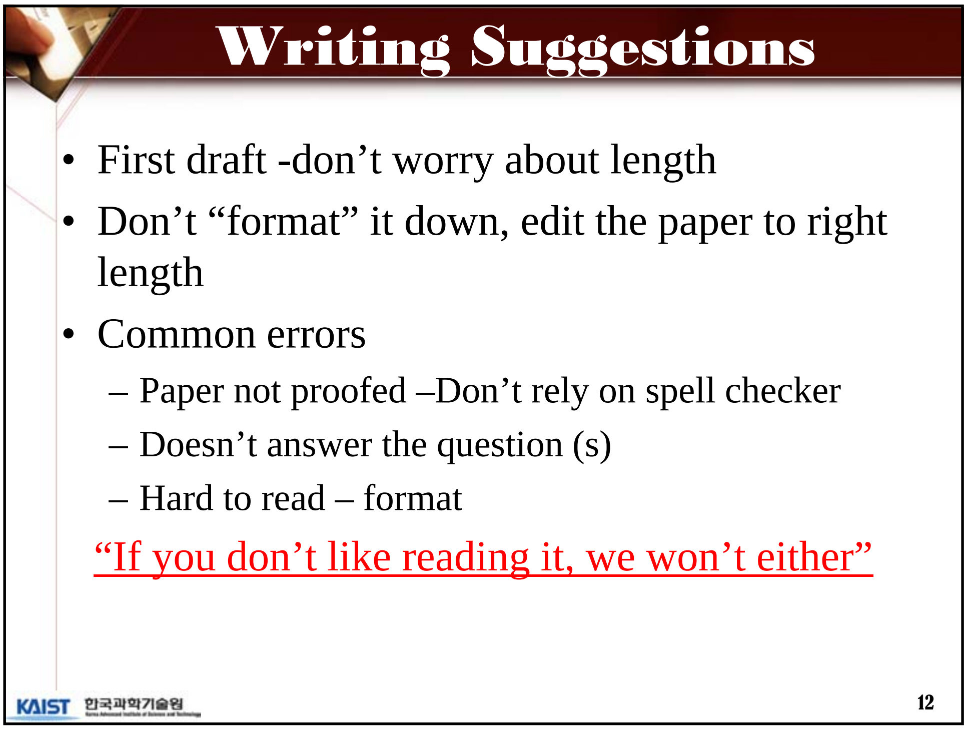
Final Paper - I

- A Software Engineering subject of interest TO YOU.
  - Describe Abstract (One page)
    - Due Date: April 15(FRI)
- Graded on 10 point scale
  - 2 points for Topic (on target, interesting, difficulty)
  - 2 points for bibliography
    - 10 min. “good” sources (URL’s aren’t “good”)
    - 3 sources min. primary (original research)
  - 6 points
    - Readability (spelling, grammar, organization, etc.)
    - Content: Needs to show analysis and synthesis, not just regurgitation
    - Final Presentation
- Term Paper Format & Submission
  - Papers should comply with IEEE style and be submitted in MSWord (.doc)
  - Send the paper to T.A. and CC it to the instructor
  - Due Date: June 10 (FRI)



Attendance & Participation

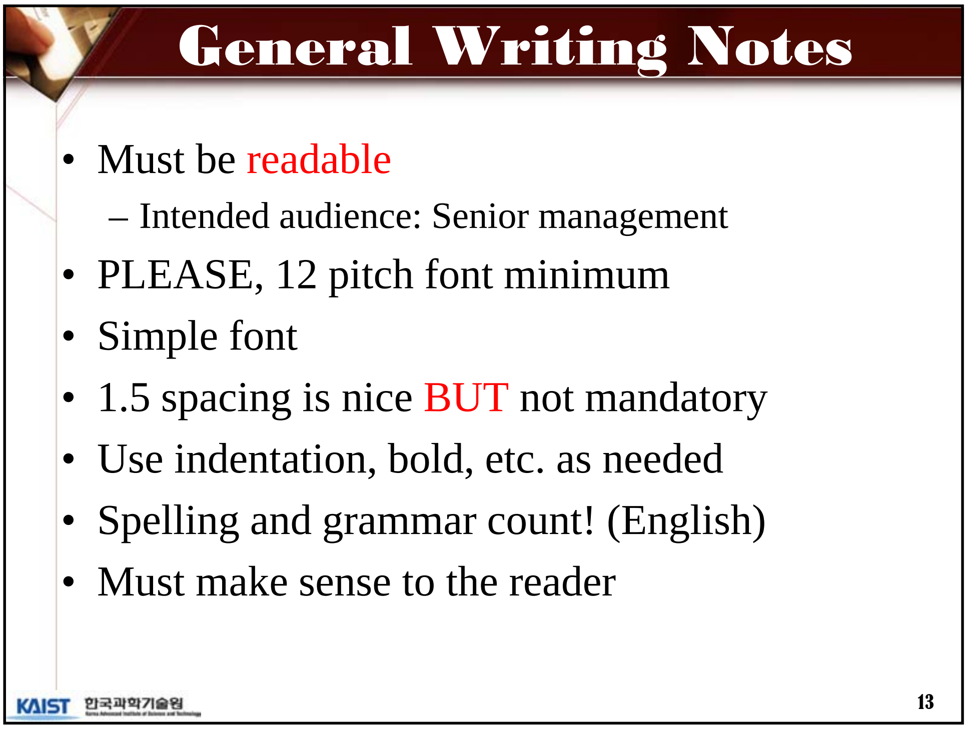
- More than 15% absent
  - Fail the class (Grade: F)
- Being late more than 30 minutes
  - Regarded as absent
- No disturbance in the class
  - Cellular Phone: Turn off or Manner mode
  - No walking in and out during class
- Pro-active participation
  - Bring with your questions and ideas



Writing Suggestions

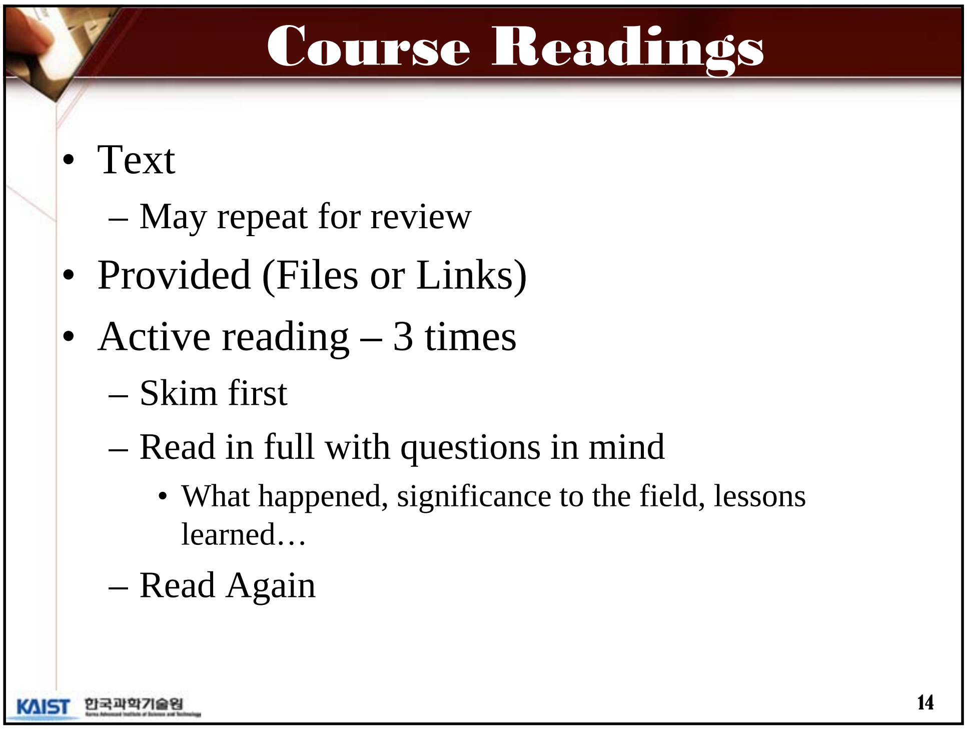
- First draft -don’t worry about length
- Don’t “format” it down, edit the paper to right length
- Common errors
  - Paper not proofed –Don’t rely on spell checker
  - Doesn’t answer the question (s)
  - Hard to read – format

“If you don’t like reading it, we won’t either”



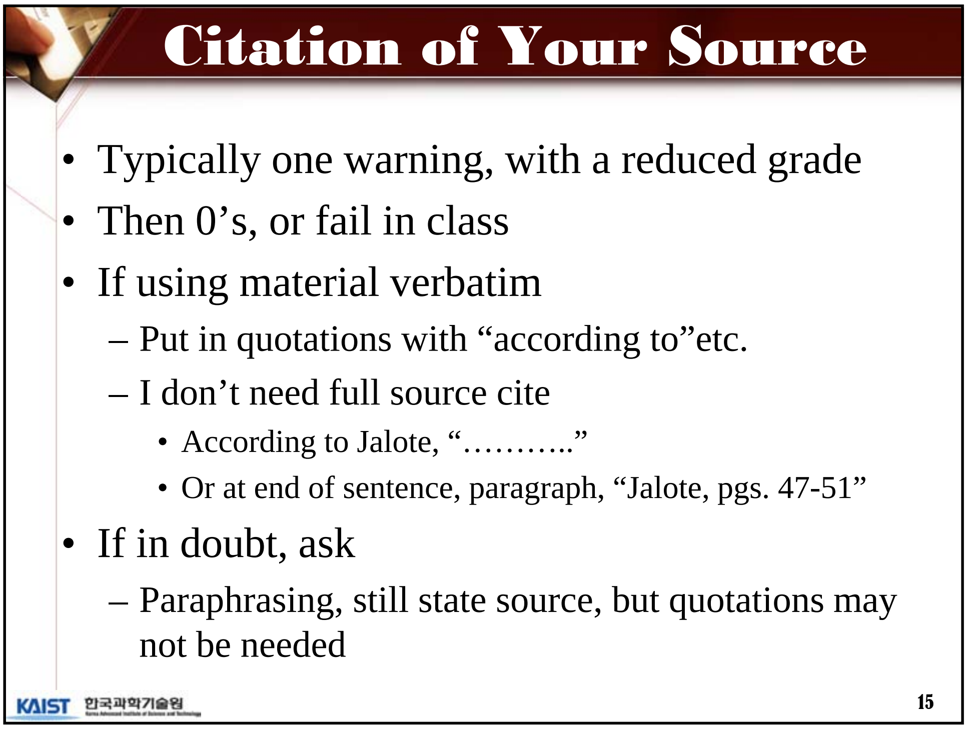
General Writing Notes

- Must be readable
  - – Intended audience: Senior management
- PLEASE, 12 pitch font minimum
- Simple font
- 1.5 spacing is nice BUT not mandatory
- Use indentation, bold, etc. as needed
- Spelling and grammar count! (English)
- Must make sense to the reader



Course Readings

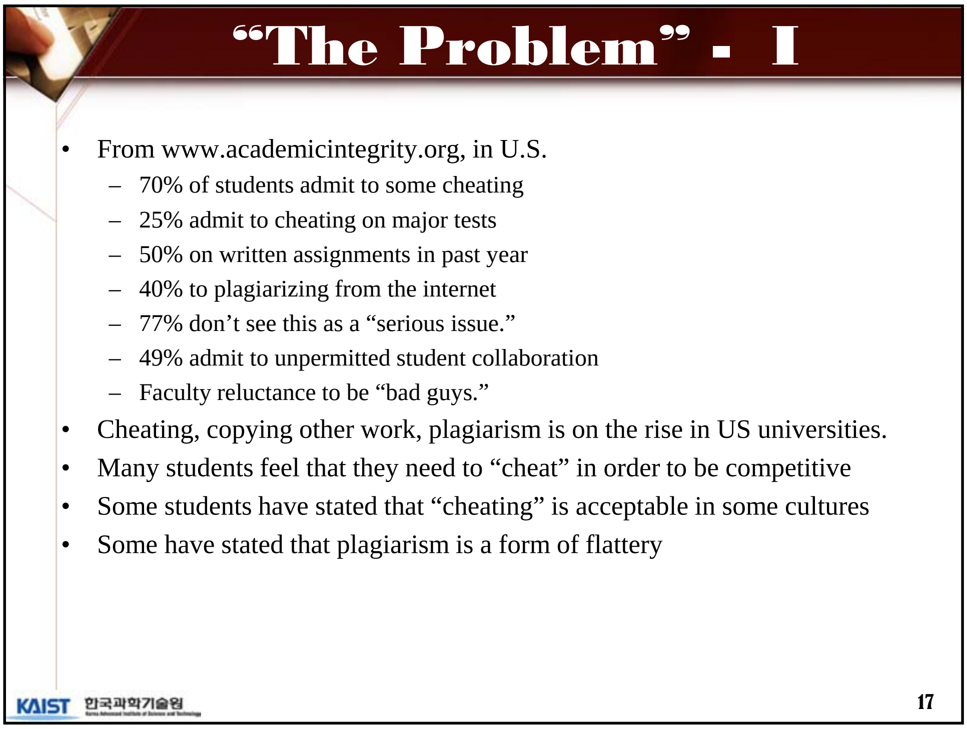
- Text
  - – May repeat for review
- Provided (Files or Links)
- Active reading – 3 times
  - – Skim first
  - – Read in full with questions in mind
    - What happened, significance to the field, lessons learned…
  - – Read Again



Citation of Your Source

- Typically one warning, with a reduced grade
- Then 0’s, or fail in class
- If using material verbatim
  - Put in quotations with “according to”etc.
  - I don’t need full source cite
    - According to Jalote, “………..”
    - Or at end of sentence, paragraph, “Jalote, pgs. 47-51”
- If in doubt, ask
  - Paraphrasing, still state source, but quotations may not be needed

Plagiarism !!!

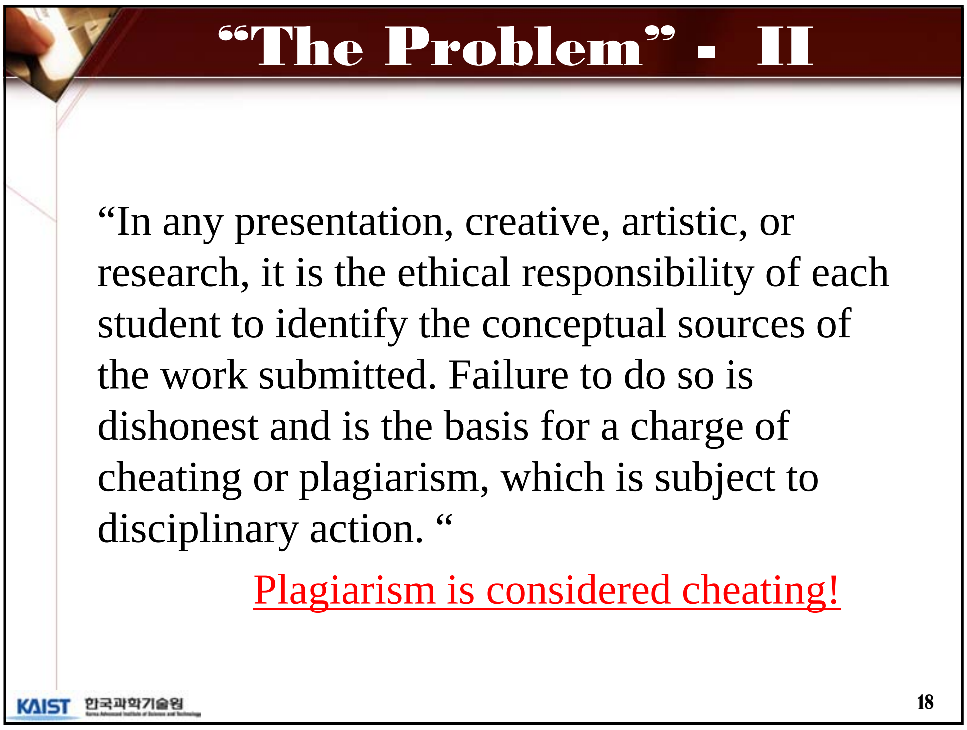


"The Problem" - I

- From www.academicintegrity.org, in U.S.
  - 70% of students admit to some cheating
  - 25% admit to cheating on major tests
  - 50% on written assignments in past year
  - 40% to plagiarizing from the internet
  - 77% don't see this as a "serious issue."
  - 49% admit to unpermitted student collaboration
  - Faculty reluctance to be "bad guys."
- Cheating, copying other work, plagiarism is on the rise in US universities.
- Many students feel that they need to "cheat" in order to be competitive
- Some students have stated that "cheating" is acceptable in some cultures
- Some have stated that plagiarism is a form of flattery

KAIST 한국과학기술원

17



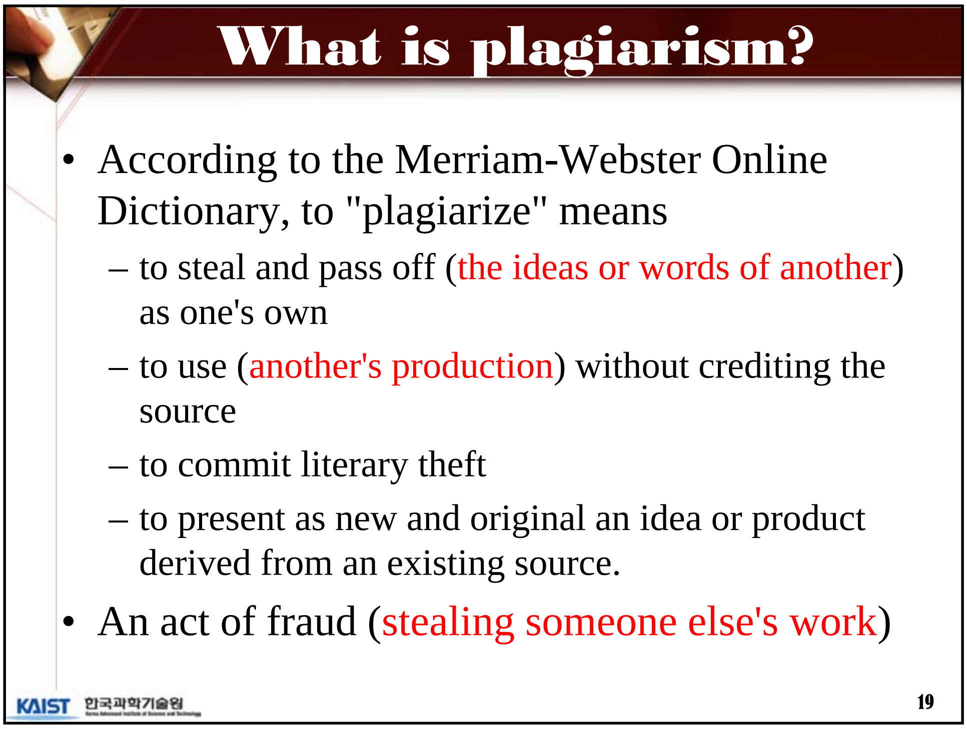
"The Problem" - II

"In any presentation, creative, artistic, or research, it is the ethical responsibility of each student to identify the conceptual sources of the work submitted. Failure to do so is dishonest and is the basis for a charge of cheating or plagiarism, which is subject to disciplinary action. "

Plagiarism is considered cheating!

KAIST 한국과학기술원

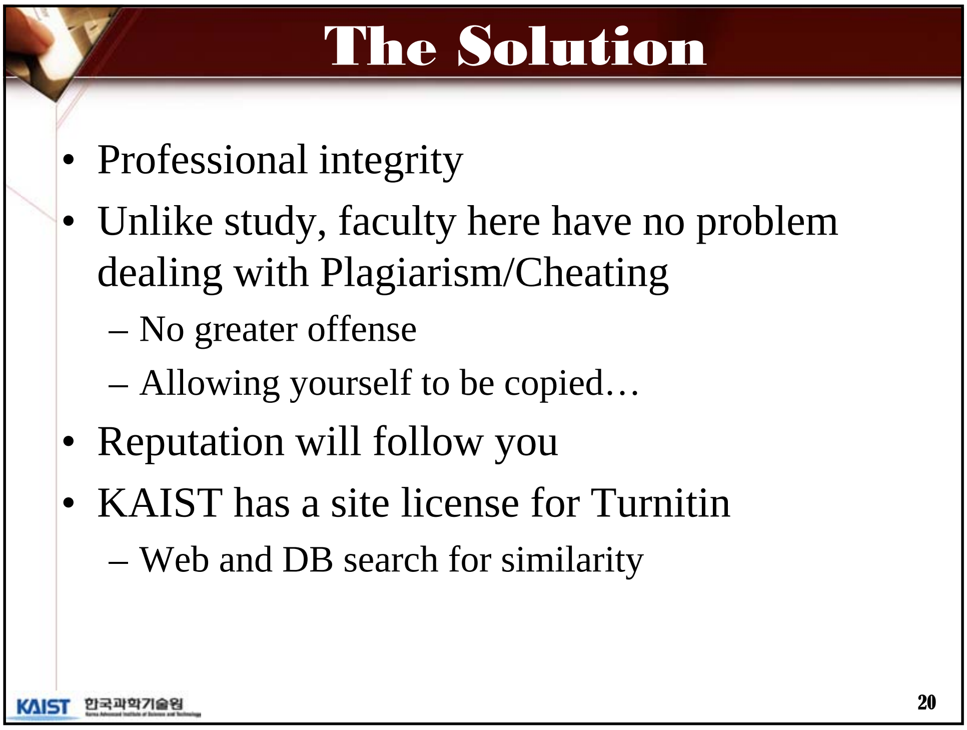
18



What is plagiarism?

- According to the Merriam-Webster Online Dictionary, to "plagiarize" means
  - to steal and pass off (the ideas or words of another) as one's own
  - to use (another's production) without crediting the source
  - to commit literary theft
  - to present as new and original an idea or product derived from an existing source.
- An act of fraud (stealing someone else's work)

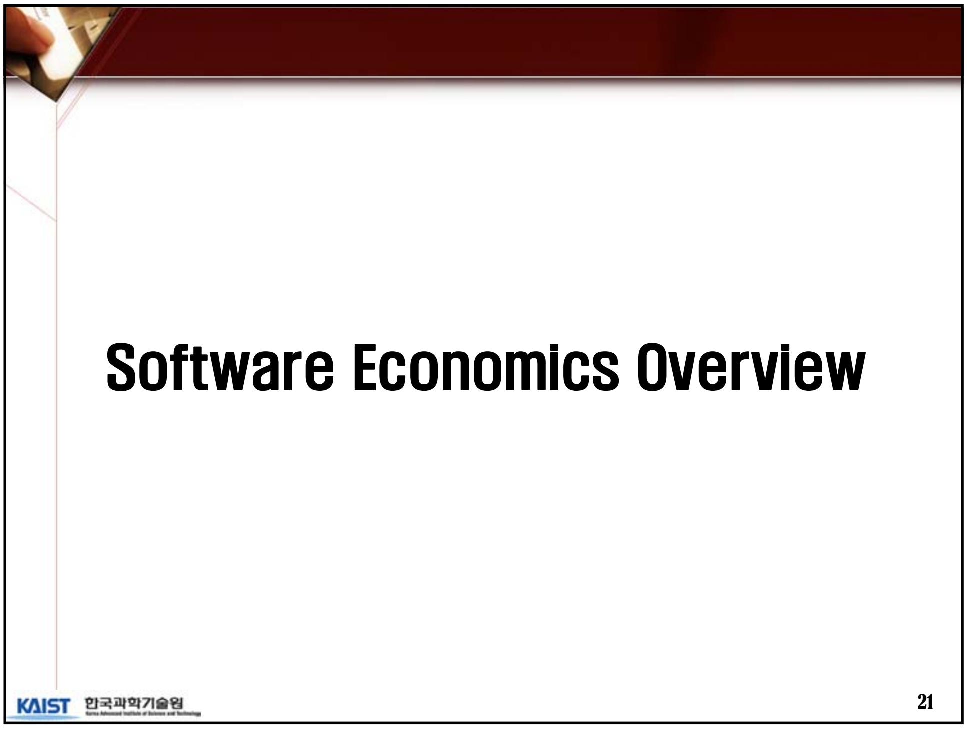
19



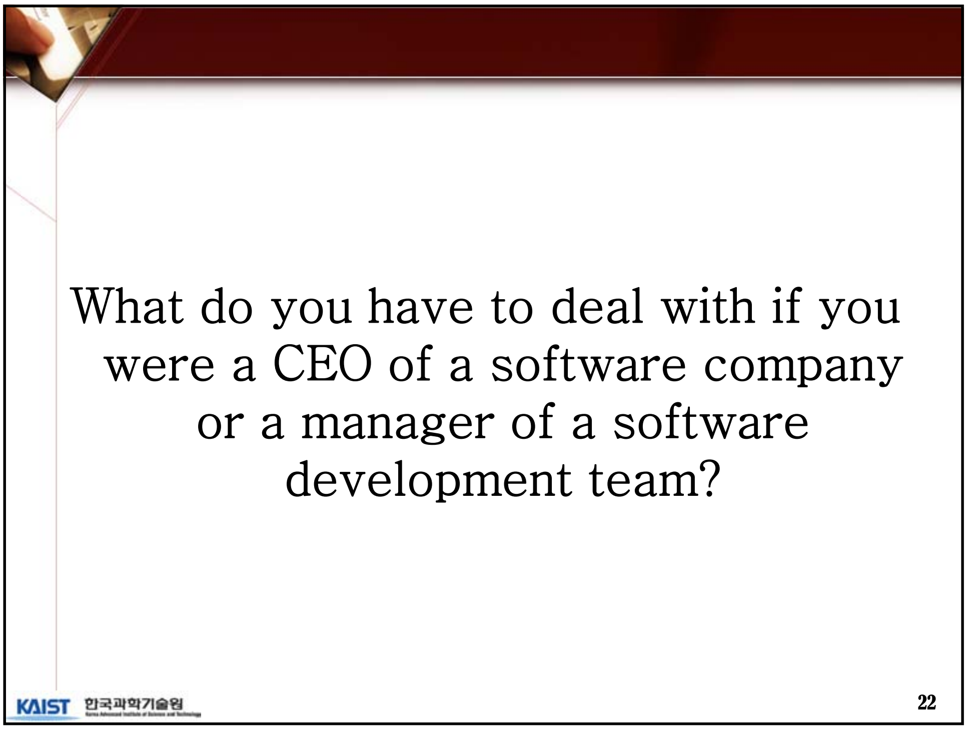
The Solution

- Professional integrity
- Unlike study, faculty here have no problem dealing with Plagiarism/Cheating
  - No greater offense
  - Allowing yourself to be copied…
- Reputation will follow you
- KAIST has a site license for Turnitin
  - Web and DB search for similarity

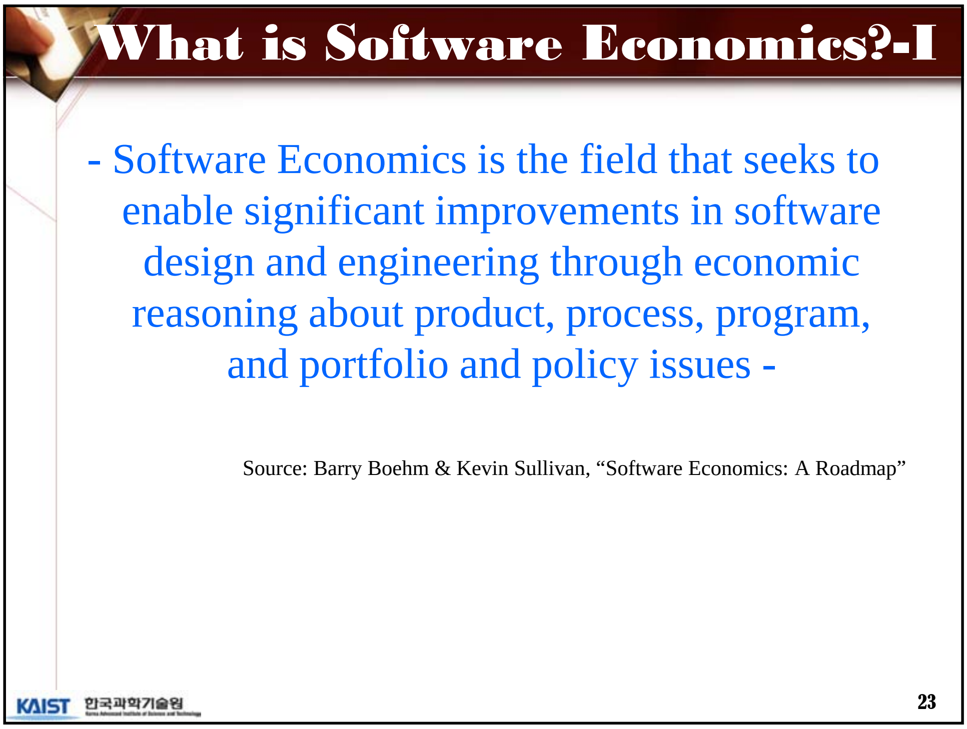
20



Software Economics Overview



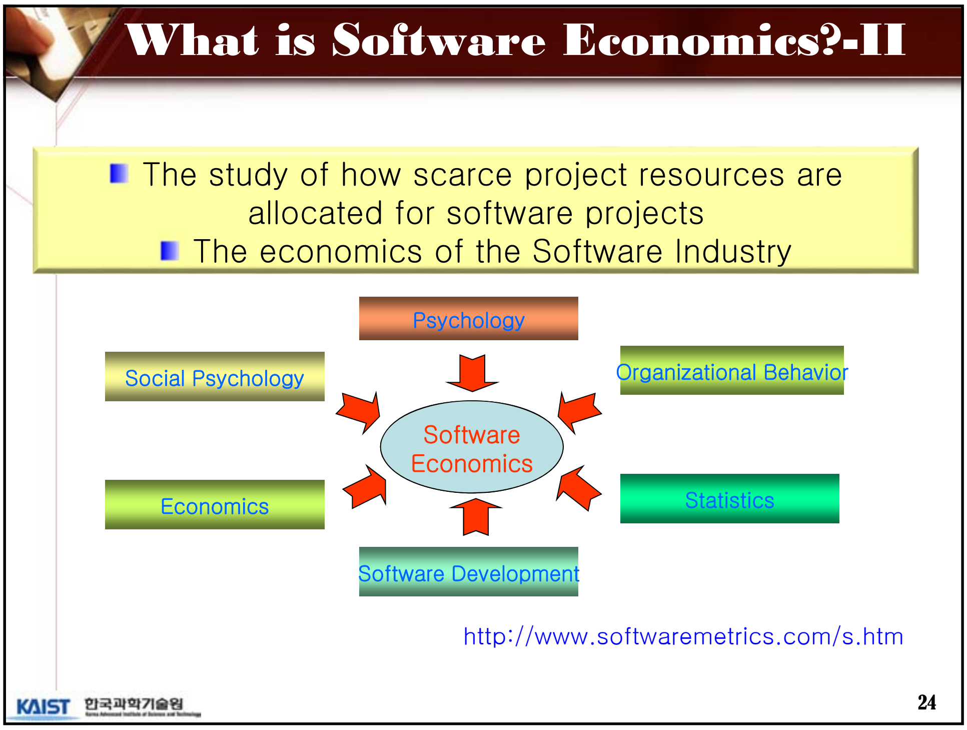
What do you have to deal with if you were a CEO of a software company or a manager of a software development team?



What is Software Economics?-I

- Software Economics is the field that seeks to enable significant improvements in software design and engineering through economic reasoning about product, process, program, and portfolio and policy issues -

Source: Barry Boehm & Kevin Sullivan, “Software Economics: A Roadmap”



What is Software Economics?-II

- The study of how scarce project resources are allocated for software projects
- The economics of the Software Industry

Psychology

Social Psychology

Organizational Behavior

Software Economics

Economics

Statistics

Software Development

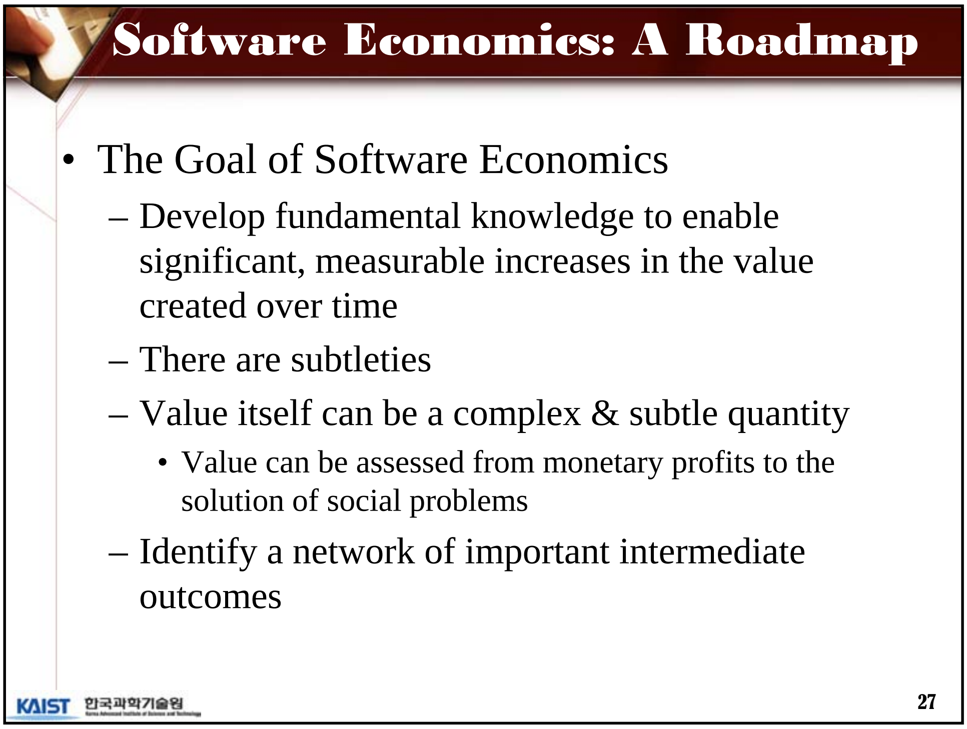
http://www.softwaremetrics.com/s.htm

Objectives of Software Economics

- To provide an empirical view of where the effort and money goes when we build large software systems
- To suggest ways of reducing and controlling software development costs

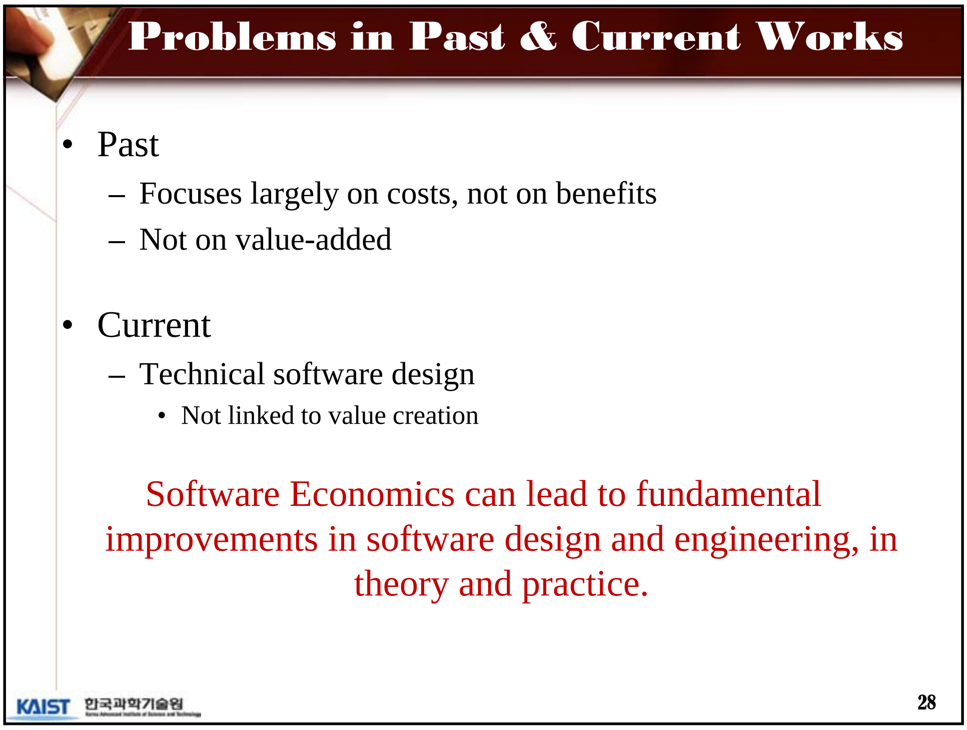
Macro vs. Micro Economics

- Macro economics
  - SE supports a commercial S/W sector that earns $200B to $240B in US
  - SE drove $1T of economic growth in US
- Micro economics
  - About ¼ of software projects are delivered successfully
  - Commercial developers typically write 1.5K lines of code per year
  - Government developers typically write 1.2K lines of code per year



Software Economics: A Roadmap

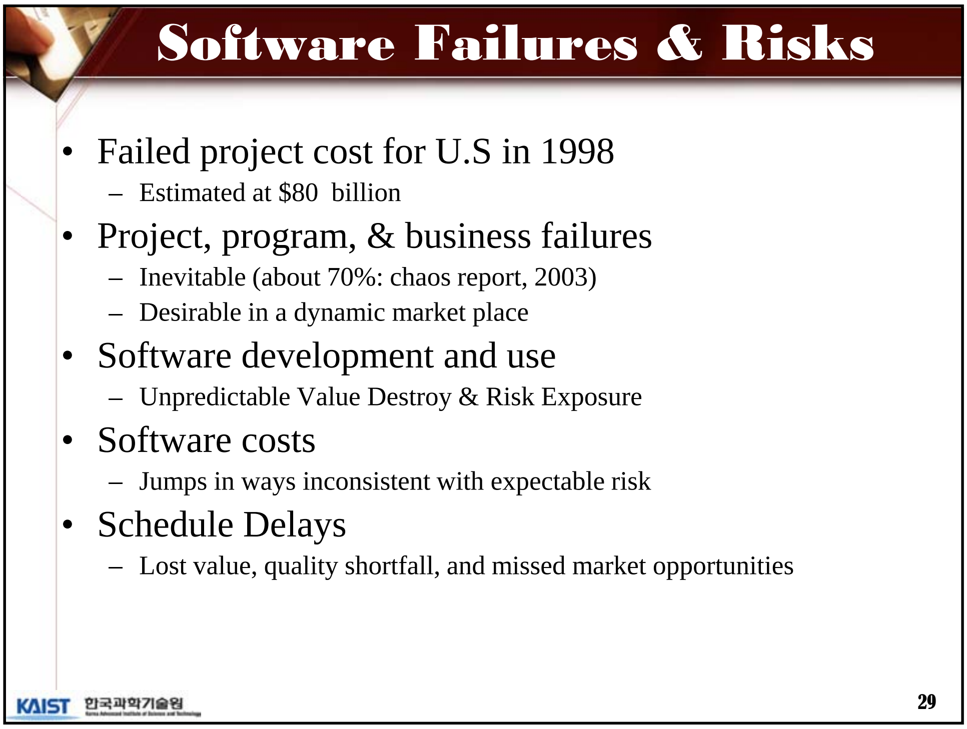
- The Goal of Software Economics
  - Develop fundamental knowledge to enable significant, measurable increases in the value created over time
  - There are subtleties
  - Value itself can be a complex & subtle quantity
    - Value can be assessed from monetary profits to the solution of social problems
  - Identify a network of important intermediate outcomes



Problems in Past & Current Works

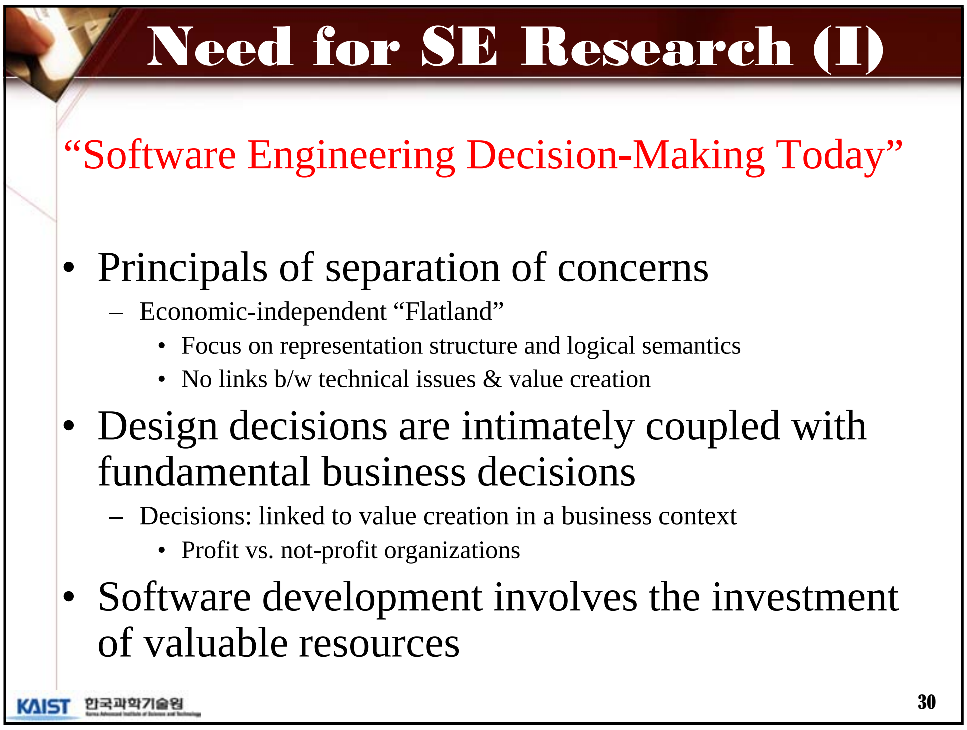
- Past
  - Focuses largely on costs, not on benefits
  - Not on value-added
- Current
  - Technical software design
    - Not linked to value creation

Software Economics can lead to fundamental improvements in software design and engineering, in theory and practice.



Software Failures & Risks

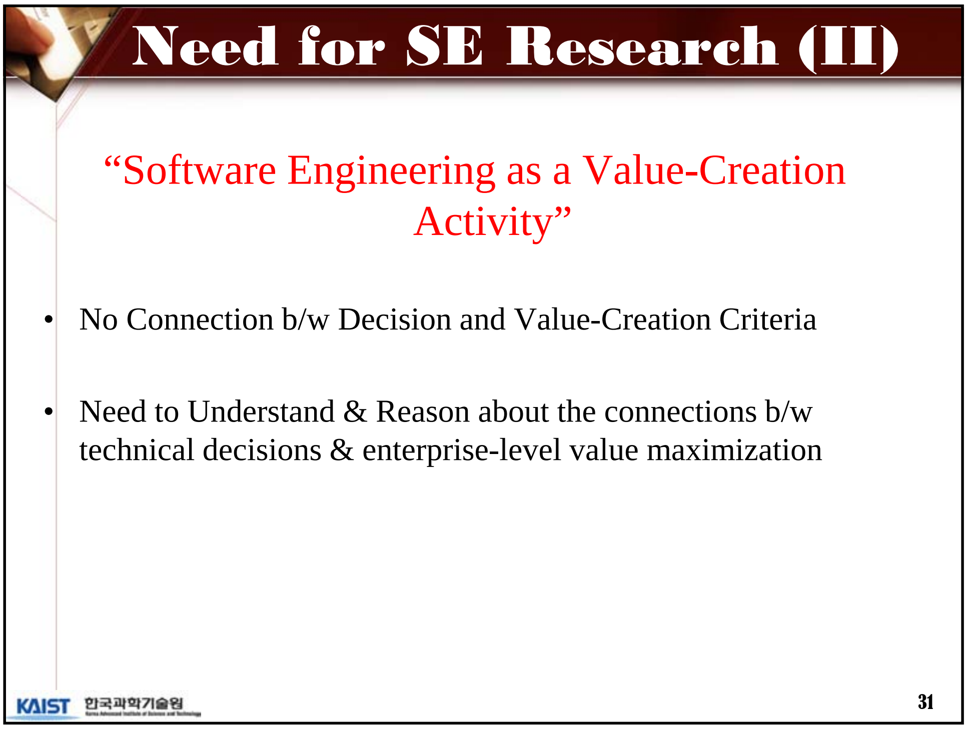
- Failed project cost for U.S in 1998
  - Estimated at $80 billion
- Project, program, & business failures
  - Inevitable (about 70%: chaos report, 2003)
  - Desirable in a dynamic market place
- Software development and use
  - Unpredictable Value Destroy & Risk Exposure
- Software costs
  - Jumps in ways inconsistent with expectable risk
- Schedule Delays
  - Lost value, quality shortfall, and missed market opportunities



Need for SE Research (I)

"Software Engineering Decision-Making Today"

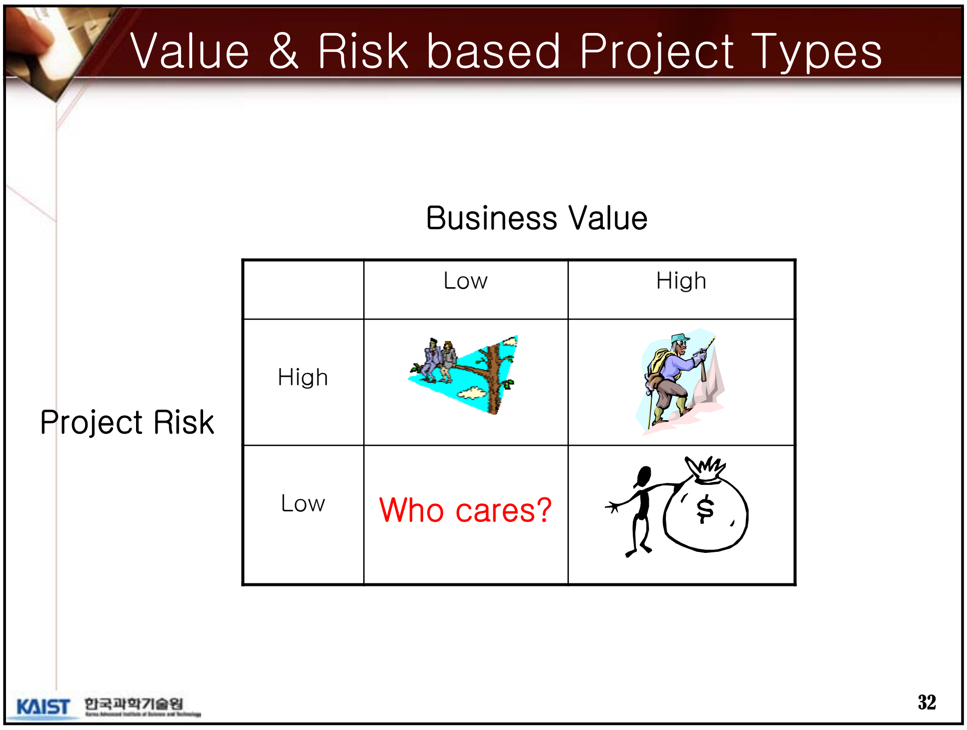
- Principals of separation of concerns
  - Economic-independent "Flatland"
    - Focus on representation structure and logical semantics
    - No links b/w technical issues & value creation
- Design decisions are intimately coupled with fundamental business decisions
  - Decisions: linked to value creation in a business context
    - Profit vs. not-profit organizations
- Software development involves the investment of valuable resources



Need for SE Research (II)

"Software Engineering as a Value-Creation Activity"

- No Connection b/w Decision and Value-Creation Criteria
- Need to Understand & Reason about the connections b/w technical decisions & enterprise-level value maximization

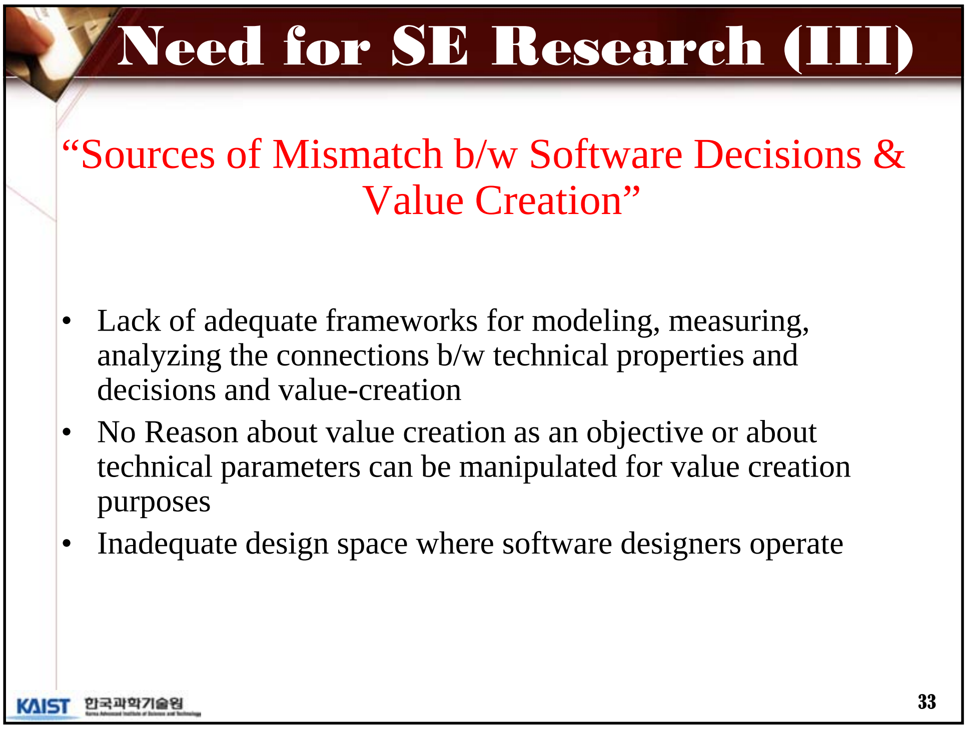


Value & Risk based Project Types

Business Value

Project Risk

	Low	High
High		
Low	Who cares?	

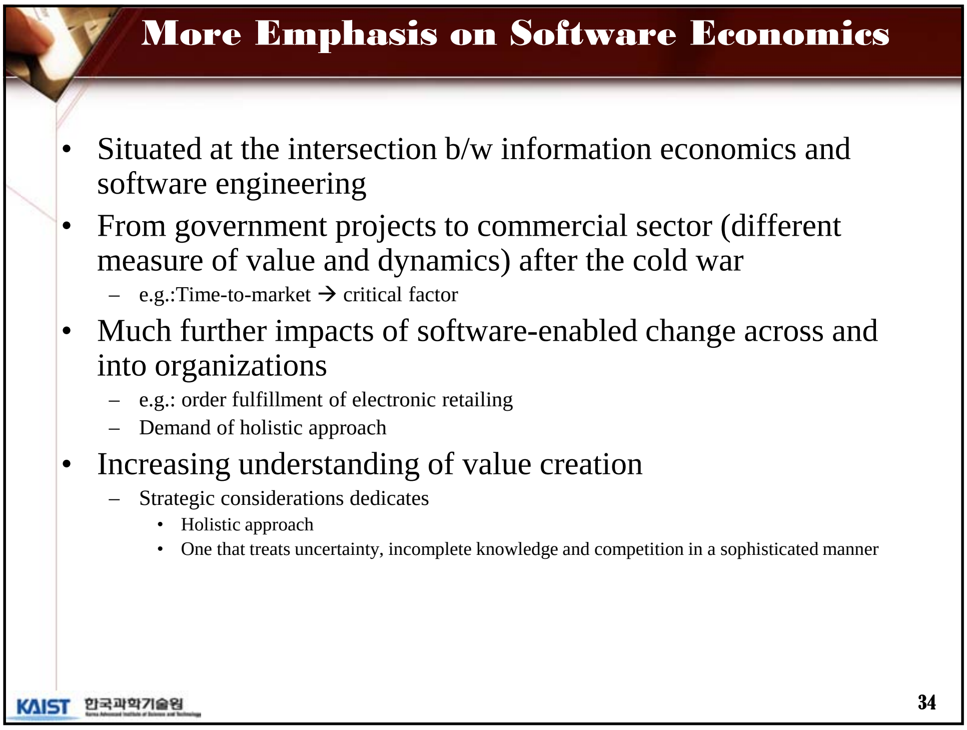


Need for SE Research (III)

"Sources of Mismatch b/w Software Decisions & Value Creation"

- Lack of adequate frameworks for modeling, measuring, analyzing the connections b/w technical properties and decisions and value-creation
- No Reason about value creation as an objective or about technical parameters can be manipulated for value creation purposes
- Inadequate design space where software designers operate

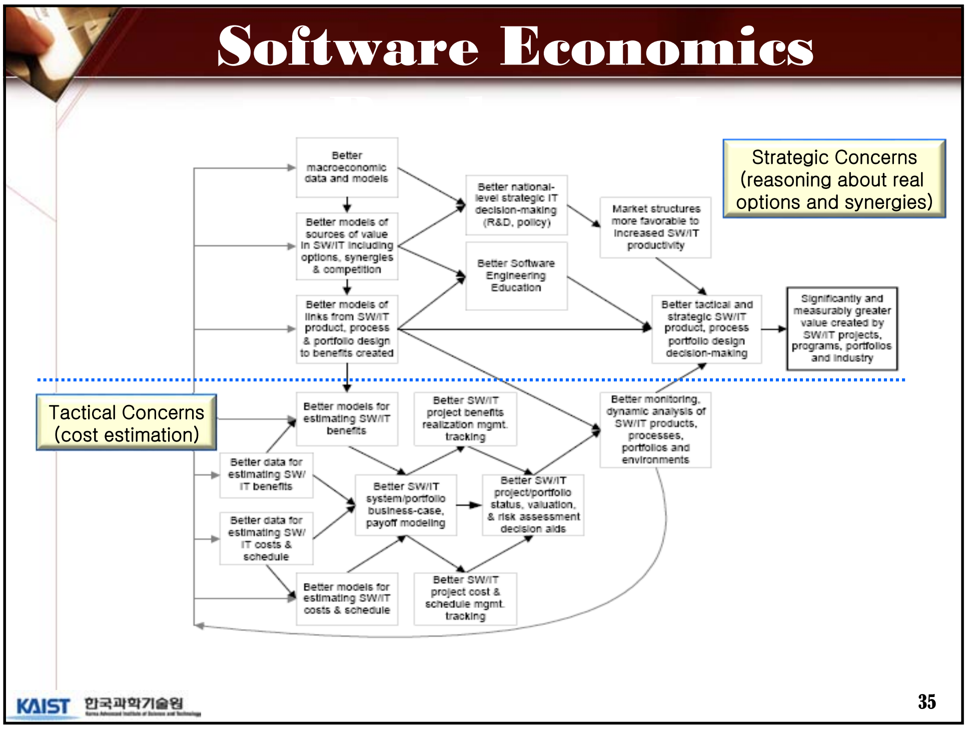
KAIST 한국과학기술원 33



More Emphasis on Software Economics

- Situated at the intersection b/w information economics and software engineering
- From government projects to commercial sector (different measure of value and dynamics) after the cold war
  - e.g.:Time-to-market → critical factor
- Much further impacts of software-enabled change across and into organizations
  - e.g.: order fulfillment of electronic retailing
  - Demand of holistic approach
- Increasing understanding of value creation
  - Strategic considerations dedicates
    - Holistic approach
    - One that treats uncertainty, incomplete knowledge and competition in a sophisticated manner

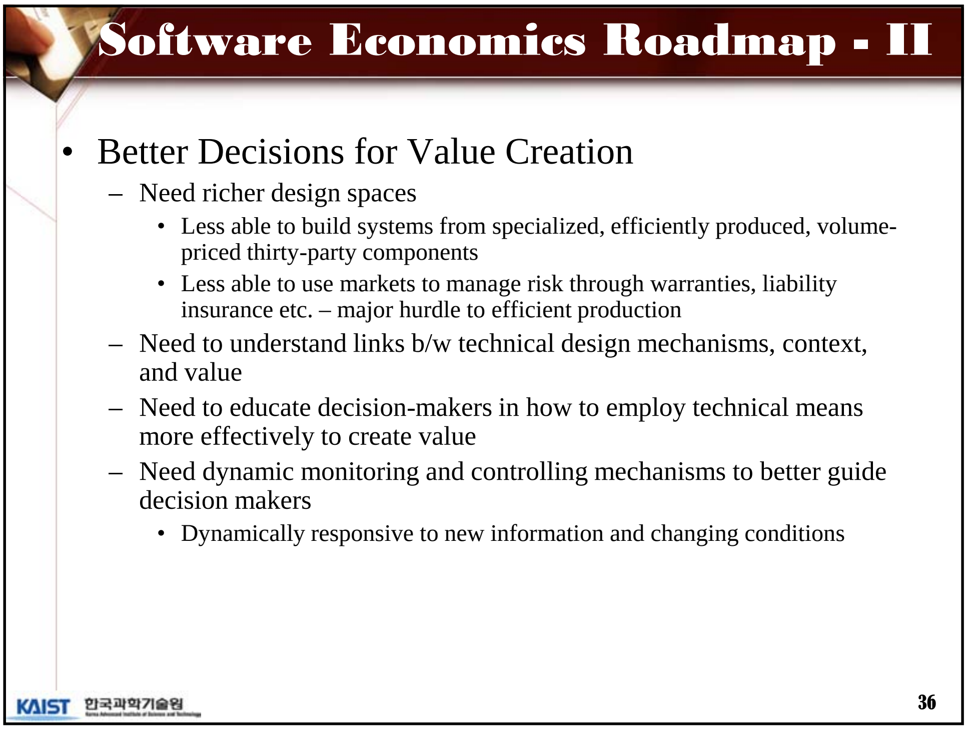
KAIST 한국과학기술원 34



Software Economics

Better macroeconomic data and models
Better models of sources of value in SW/IT including options, synergies & competition
Better models of links from SW/IT product, process & portfolio design to benefits created
Better national-level strategic IT decision-making (R&D, policy)
Better Software Engineering Education
Market structures more favorable to increased SW/IT productivity
Strategic Concerns (reasoning about real options and synergies)
Better tactical and strategic SW/IT product, process portfolio design decision-making
Significantly and measurably greater value created by SW/IT projects, programs, portfolios and industry
Tactical Concerns (cost estimation)
Better models for estimating SW/IT benefits
Better SW/IT project benefits realization mgmt. tracking
Better monitoring, dynamic analysis of SW/IT products, processes, portfolios and environments
Better data for estimating SW/IT benefits
Better SW/IT system/portfolio business-case, payoff modeling
Better SW/IT project/portfolio status, valuation, & risk assessment decision aids
Better data for estimating SW/IT costs & schedule
Better models for estimating SW/IT costs & schedule
Better SW/IT project cost & schedule mgmt. tracking

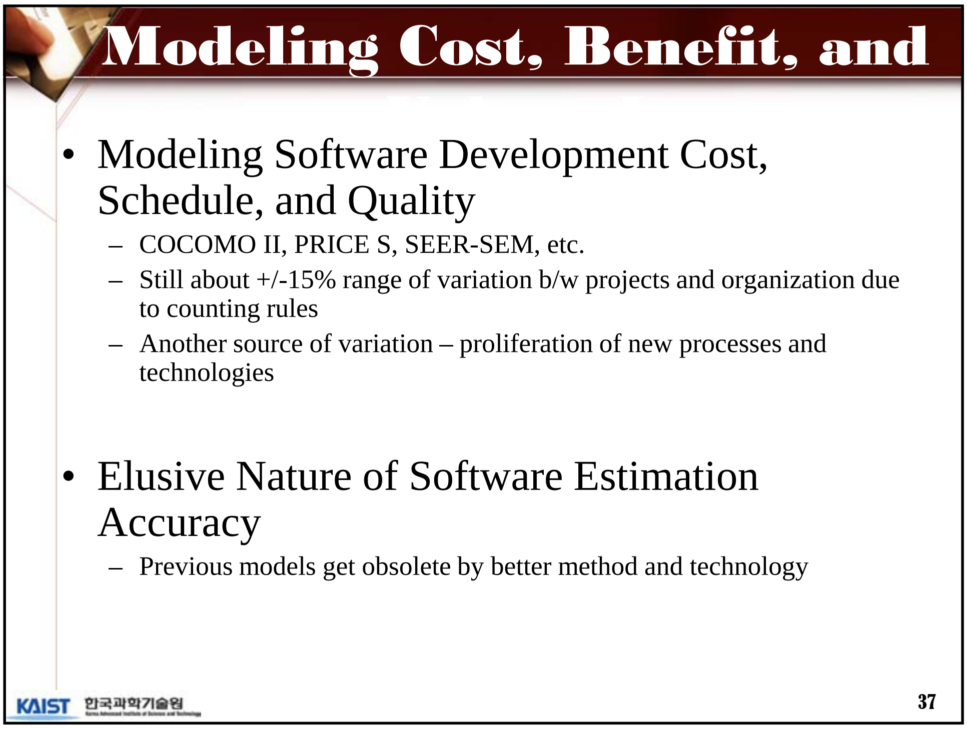
35



Software Economics Roadmap - II

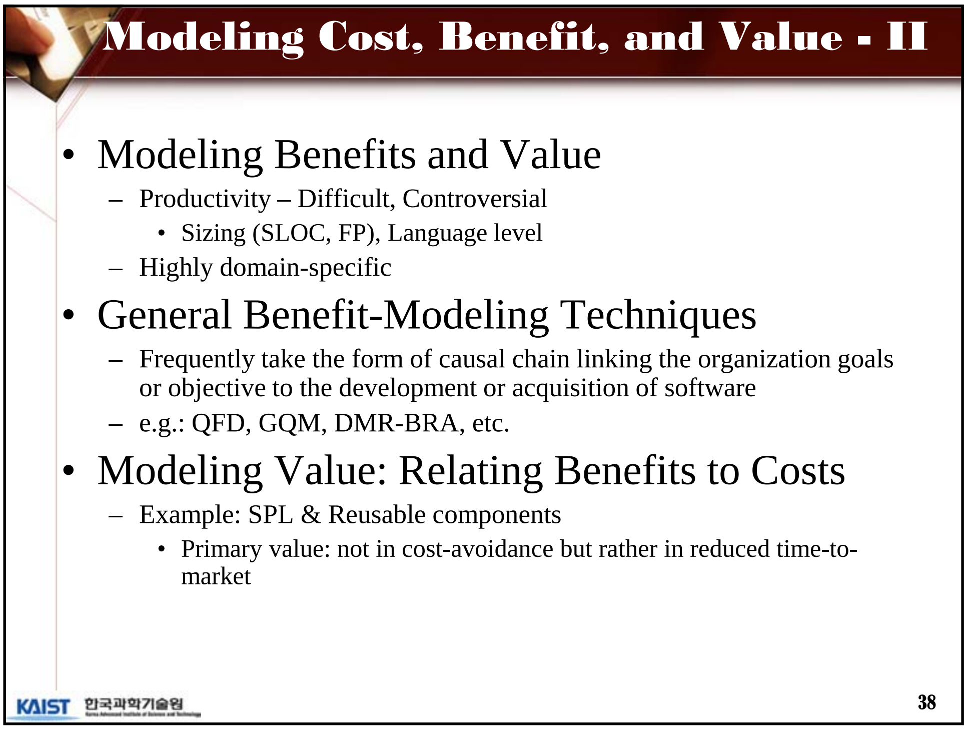
- Better Decisions for Value Creation
  - Need richer design spaces
    - Less able to build systems from specialized, efficiently produced, volume-priced thirty-party components
    - Less able to use markets to manage risk through warranties, liability insurance etc. – major hurdle to efficient production
  - Need to understand links b/w technical design mechanisms, context, and value
  - Need to educate decision-makers in how to employ technical means more effectively to create value
  - Need dynamic monitoring and controlling mechanisms to better guide decision makers
    - Dynamically responsive to new information and changing conditions

36



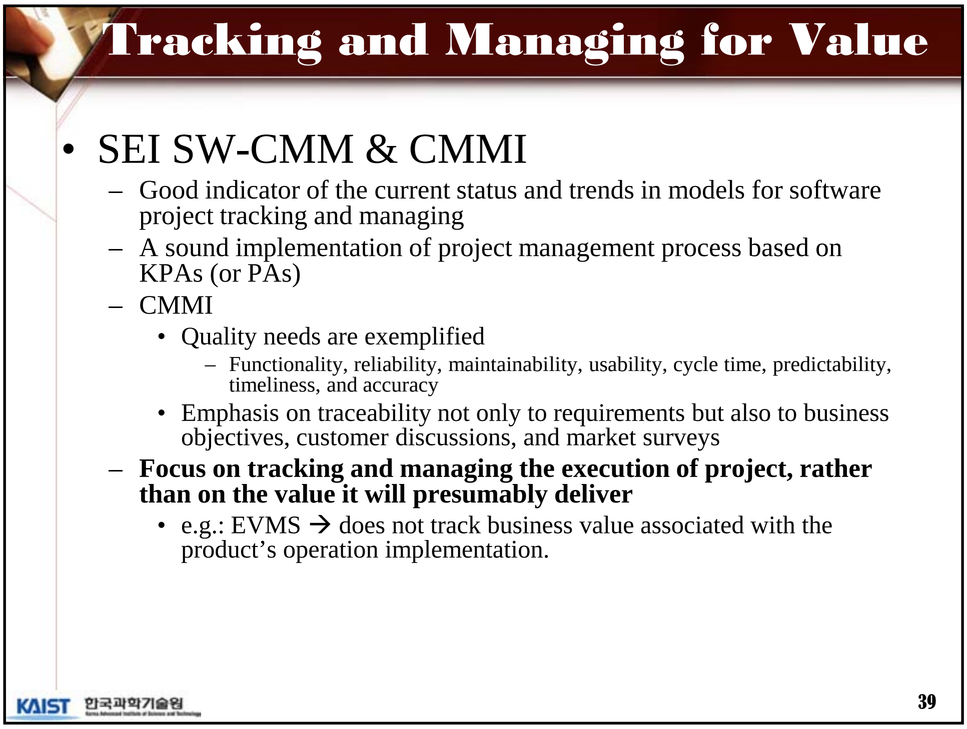
Modeling Cost, Benefit, and

- Modeling Software Development Cost, Schedule, and Quality
  - COCOMO II, PRICE S, SEER-SEM, etc.
  - Still about +/-15% range of variation b/w projects and organization due to counting rules
  - Another source of variation – proliferation of new processes and technologies
- Elusive Nature of Software Estimation Accuracy
  - Previous models get obsolete by better method and technology



Modeling Cost, Benefit, and Value - II

- Modeling Benefits and Value
  - Productivity – Difficult, Controversial
    - Sizing (SLOC, FP), Language level
  - Highly domain-specific
- General Benefit-Modeling Techniques
  - Frequently take the form of causal chain linking the organization goals or objective to the development or acquisition of software
  - e.g.: QFD, GQM, DMR-BRA, etc.
- Modeling Value: Relating Benefits to Costs
  - Example: SPL & Reusable components
    - Primary value: not in cost-avoidance but rather in reduced time-to-market

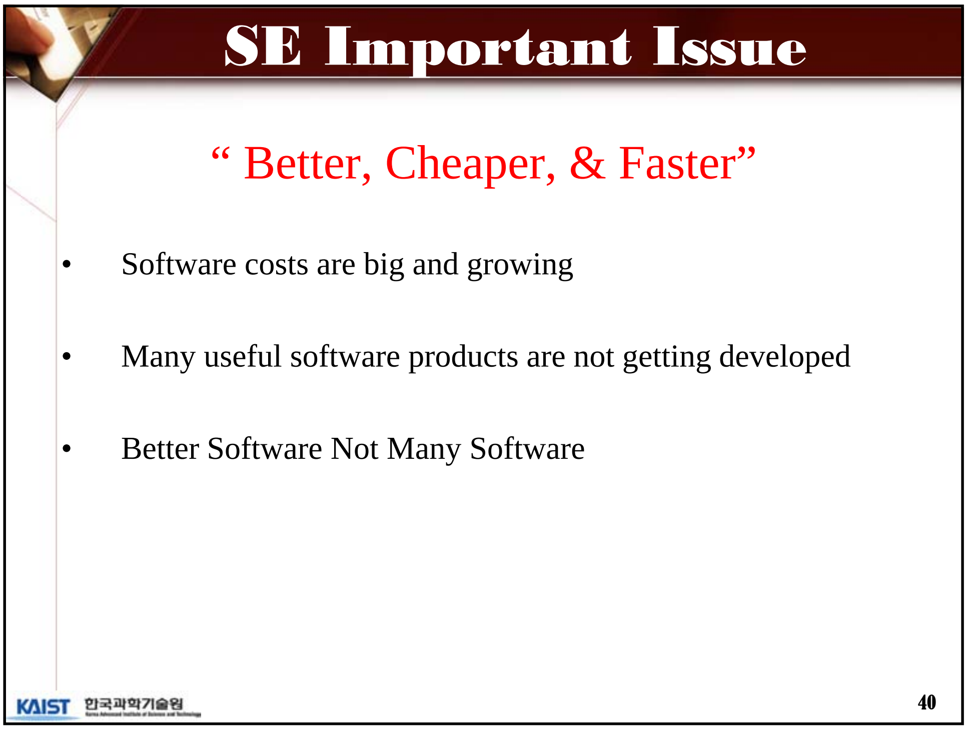


Tracking and Managing for Value

- SEI SW-CMM & CMMI
  - Good indicator of the current status and trends in models for software project tracking and managing
  - A sound implementation of project management process based on KPAs (or PAs)
  - CMMI
    - Quality needs are exemplified
      - Functionality, reliability, maintainability, usability, cycle time, predictability, timeliness, and accuracy
    - Emphasis on traceability not only to requirements but also to business objectives, customer discussions, and market surveys
  - **Focus on tracking and managing the execution of project, rather than on the value it will presumably deliver**
    - e.g.: EVMS → does not track business value associated with the product's operation implementation.

KAIST 한국과학기술원

39



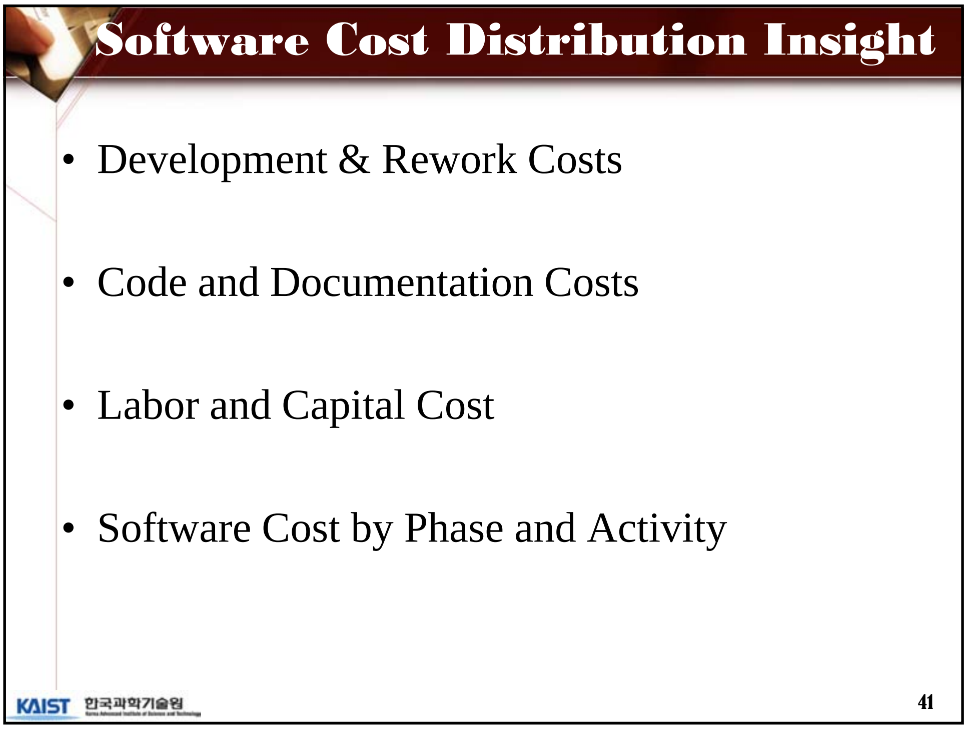
SE Important Issue

" Better, Cheaper, & Faster"

- Software costs are big and growing
- Many useful software products are not getting developed
- Better Software Not Many Software

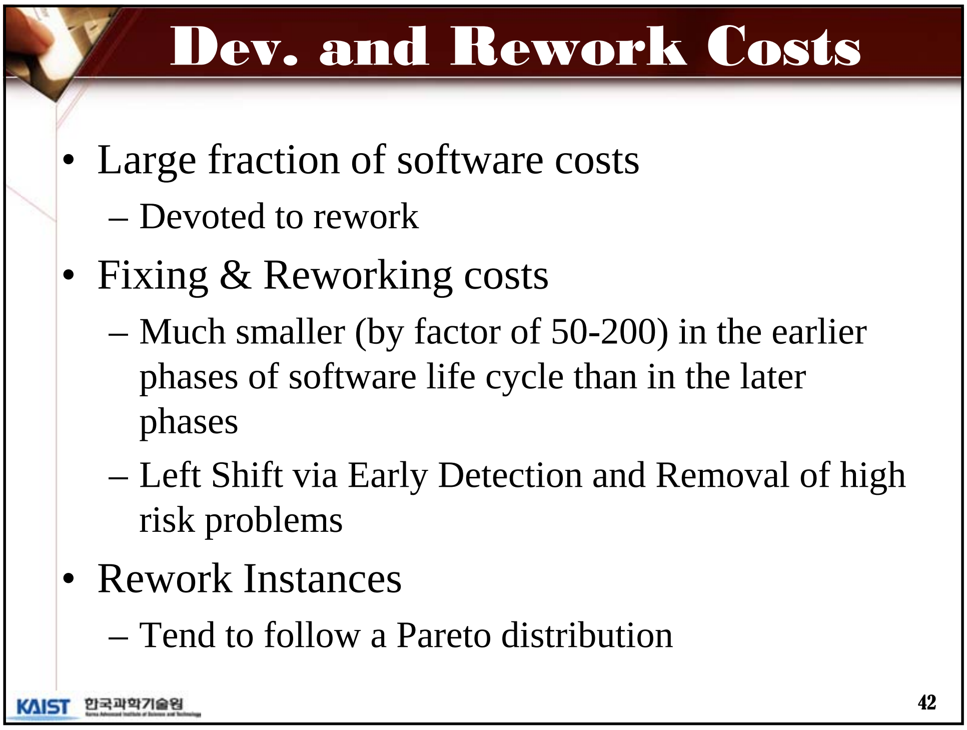
KAIST 한국과학기술원

40



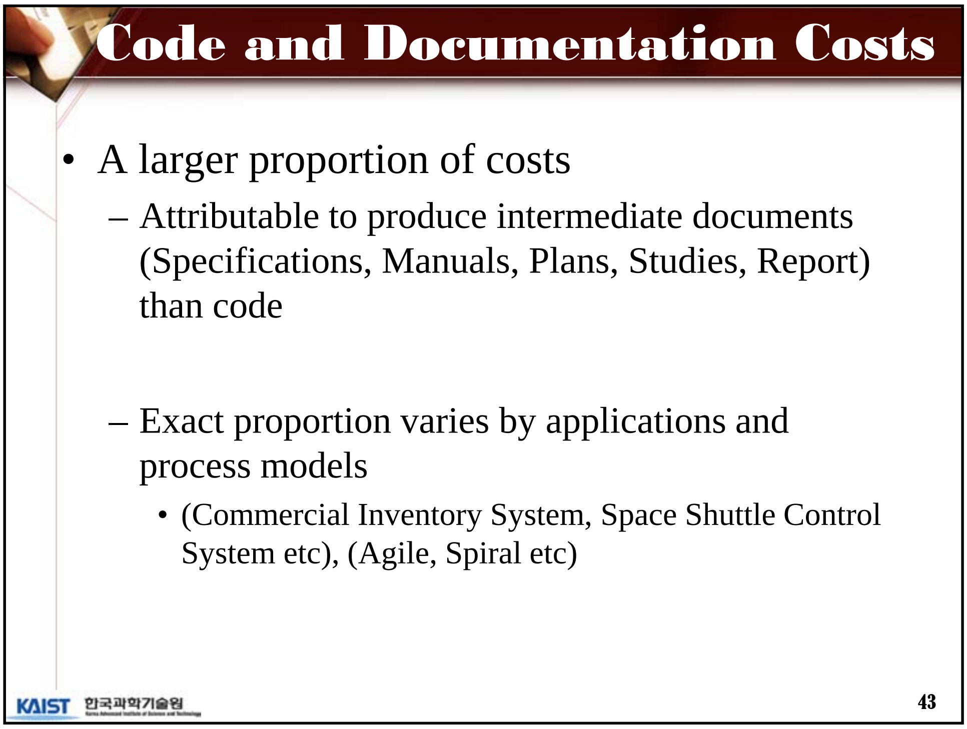
Software Cost Distribution Insight

- Development & Rework Costs
- Code and Documentation Costs
- Labor and Capital Cost
- Software Cost by Phase and Activity



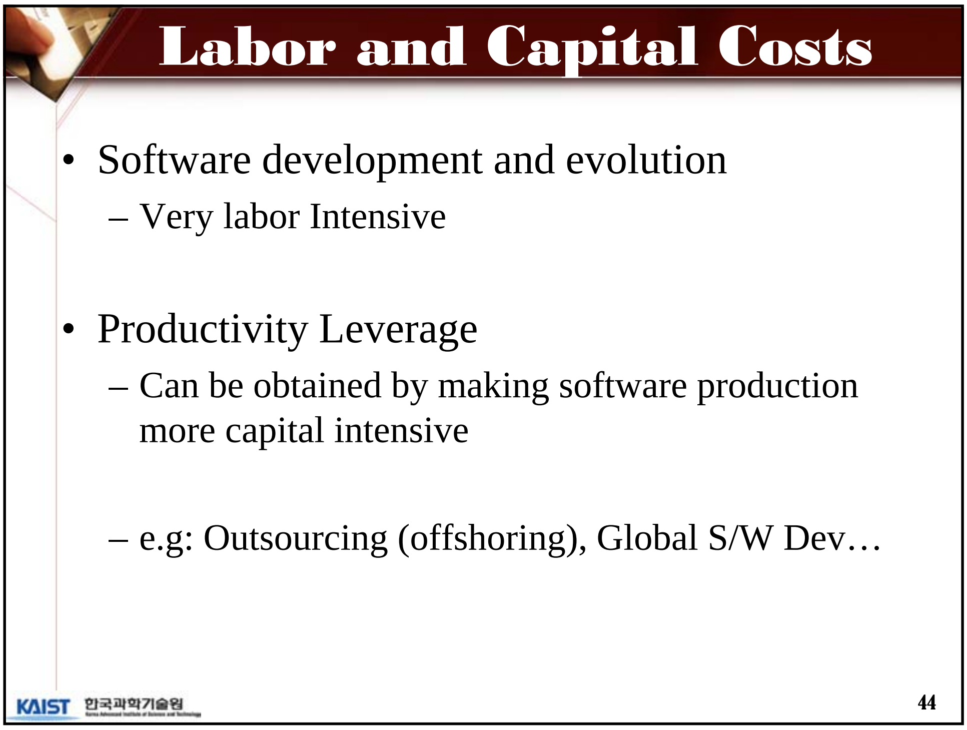
Dev. and Rework Costs

- Large fraction of software costs
  - Devoted to rework
- Fixing & Reworking costs
  - Much smaller (by factor of 50-200) in the earlier phases of software life cycle than in the later phases
  - Left Shift via Early Detection and Removal of high risk problems
- Rework Instances
  - Tend to follow a Pareto distribution



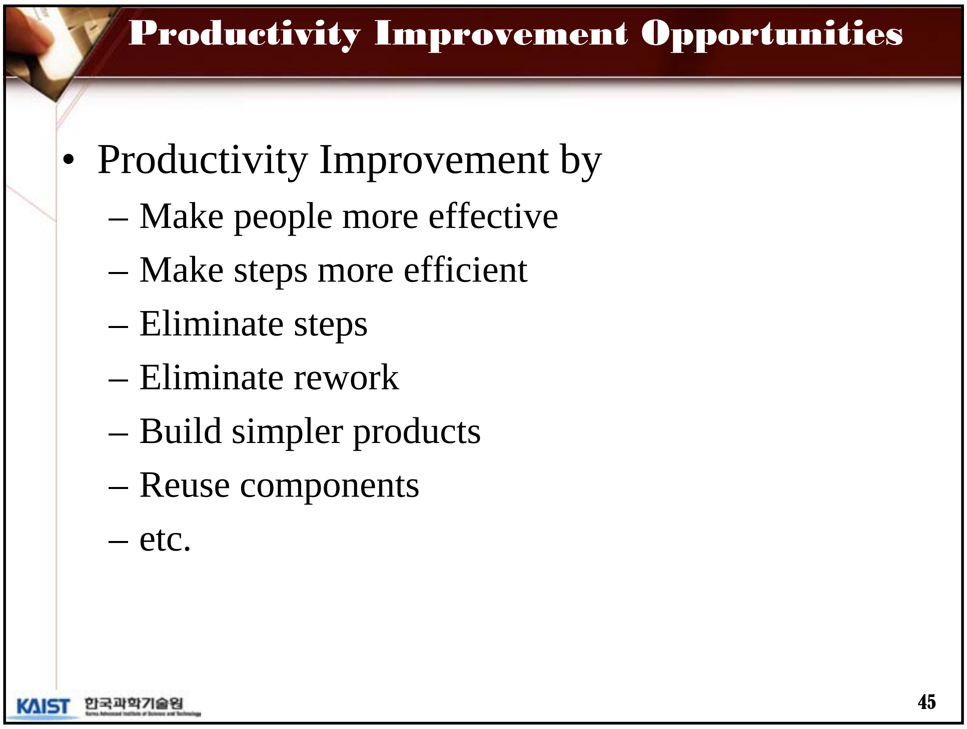
Code and Documentation Costs

- A larger proportion of costs
  - Attributable to produce intermediate documents (Specifications, Manuals, Plans, Studies, Report) than code
  - Exact proportion varies by applications and process models
    - (Commercial Inventory System, Space Shuttle Control System etc), (Agile, Spiral etc)



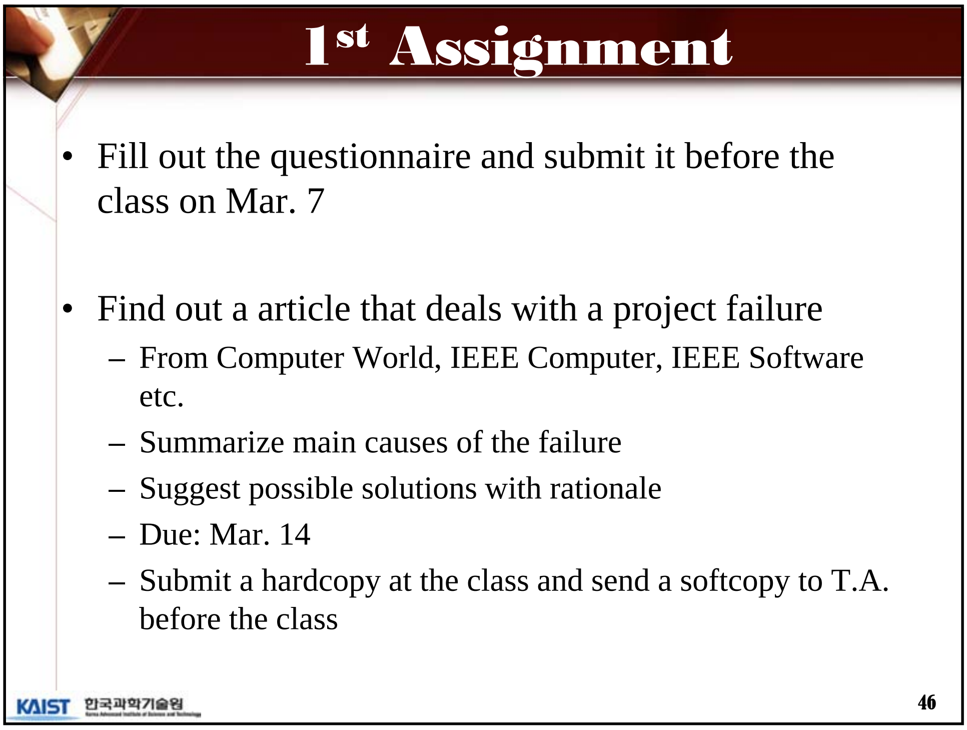
Labor and Capital Costs

- Software development and evolution
  - Very labor Intensive
- Productivity Leverage
  - Can be obtained by making software production more capital intensive
  - e.g: Outsourcing (offshoring), Global S/W Dev…



Productivity Improvement Opportunities

- Productivity Improvement by
  - Make people more effective
  - Make steps more efficient
  - Eliminate steps
  - Eliminate rework
  - Build simpler products
  - Reuse components
  - etc.



1st Assignment

- Fill out the questionnaire and submit it before the class on Mar. 7

- Find out a article that deals with a project failure
  - From Computer World, IEEE Computer, IEEE Software etc.
  - Summarize main causes of the failure
  - Suggest possible solutions with rationale
  - Due: Mar. 14
  - Submit a hardcopy at the class and send a softcopy to T.A. before the class

Q & A