



University of Southern California
Center for Software Engineering

Software Economics: A Roadmap

Barry Boehm, USC
Kevin Sullivan, UVirginia
ICSE 2000
June 8, 2000



University of Southern California
Center for Software Engineering

Software Economics: A Roadmap

- **Where have we been?**
- **Where are we now?**
- **Where do we need to go?**
- **How can we get there?**

Where Have We Been?

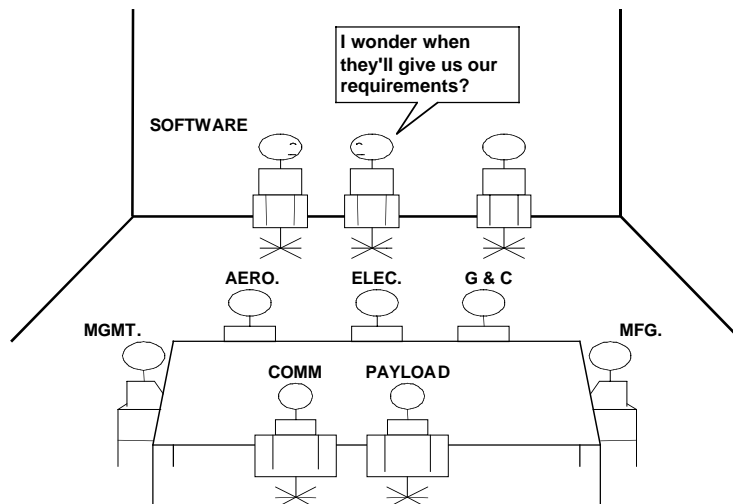
- **Overemphasis on separation of concerns**
 - System engineers determine requirements
 - Enough for software engineers to transform requirements into code, verify correctness
- **Correctness at any cost: a sacred duty**
- **Contract theory of programming, software acquisition**
 - Weak understanding of requirements-cost mapping
- **Some lucky successes**
- **More unlucky failures**
 - Overrun, user-unfriendly, mission-disruptive
- **Trust to luck that requirements are done right**

6/8/00

©USC-CSE

3

Resulting Project Social Structure





Where Are We Now?

- **Increasing understanding of requirements-cost mapping**
 - And its relationship to design
 - Understanding erodes due to pace of technology
- **Learning how to put separation of concerns in context**
 - Domain and evolution-driven architectures
 - Requires both domain and software expertise
- **Slowly changing traditional cultures**
 - Software CMM vs. CMMI, SPICE
- **Learning to use cost and schedule estimates to manage projects**
 - Earned value tracking
- **Fewer overrun, user-unfriendly systems**
- **But still many mission-disruptive and rejected systems**

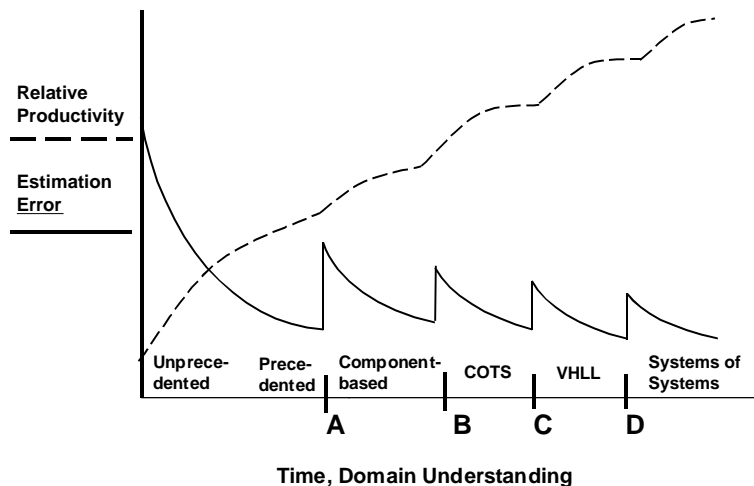
6/8/00

©USC-CSE

5



Productivity and Estimation Accuracy Trends



6/8/00

©USC-CSE

6



The Gospel According to SW CMM v.1.1

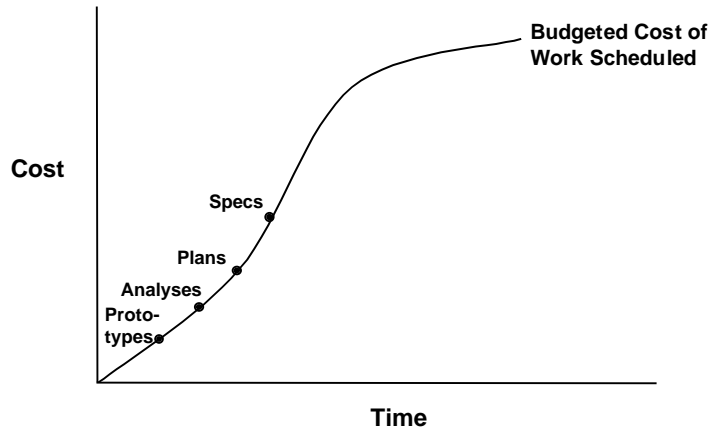
- **Requirements Management, Ability 1:**
“Analysis and allocation of the system requirements
*is not the responsibility of the software engineering group
but is a prerequisite for their work.”*”



CMMI/SPICE Software Paradigm

- **System and software engineering are integrated**
 - Software has a seat at the center table
- **Requirements, architecture, and process are developed concurrently**
 - Along with prototypes and key capabilities
- **Developments done by integrated teams**
 - Collaborative vs. adversarial process
 - Based on shared vision, negotiated stakeholder concurrence

Earned Value System

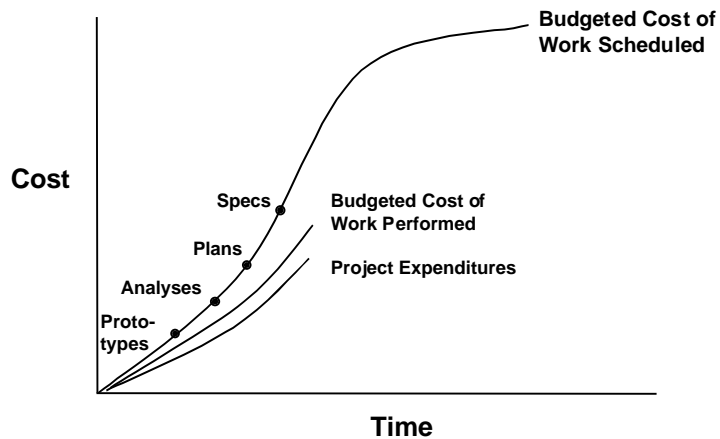


6/8/00

©USC-CSE

9

Earned Value System



6/8/00

©USC-CSE

10



Major Current Problems

- **Legacy of cost-insensitive methods**
 - OO book survey: 2 of 16 books had “cost” in index
 - Reflected in SE education and practice
- **Cost-conscious vs. value-conscious management**
 - “Earned value” tracks cost, not business value
- **“Field of Dreams” approach to SE**
 - Build the software and the profits will come
- **Problems exacerbated by pace of change**
 - In cost structures, technology and market opportunities

6/8/00

©USC-CSE

11



Software Economics: A Roadmap

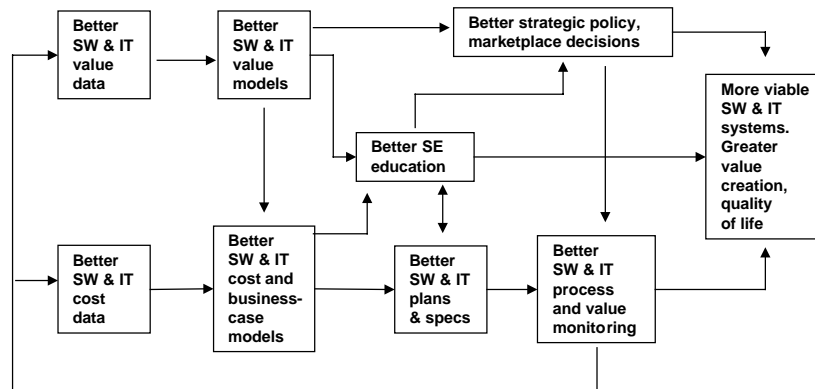
- Where have we been?
- Where are we now?
- • Where do we need to go?
- How can we get there?

6/8/00

©USC-CSE

12

Simplified Software Economics Roadmap



6/8/00

©USC-CSE

13

How Can We Get There? Strategy Elements

- **Research and development of value-based SW & IT design and management methods**
 - Strategic Software Design; real options
 - Benefits Realization Approach
 - Model-Based (System) Architecting and Software Engineering (MBASE)
- **R & D on models of SW & IT value creation**
 - Time value, network effects, stakeholder values, value of options
- **Emphasis on value creation in SE education, maturity models**
 - Shared stakeholder understanding of SW & IT value-creation phenomenology and strategies

6/8/00

©USC-CSE

14



Strategic Software Design

- **Real options theory and practice**
 - Options have value (e.g. Amazon.com)
- **Other value-determination models**
 - Decision tree, scenario-based, domain-based, ...
- **Role of value in design process**
 - Decision theory: buying information to reduce risk
 - via prototyping, models, simulations, market surveys, ...
 - Involving stakeholders in balancing needs and affordables
 - Adapting design to situation changes
 - Economic value of Parnas information-hiding
 - Design-to-schedule: need to design for ease of dropping features to meet schedule
- **Role of software design in value-based life cycle process**

6/8/00

©USC-CSE

15



Real Options Example

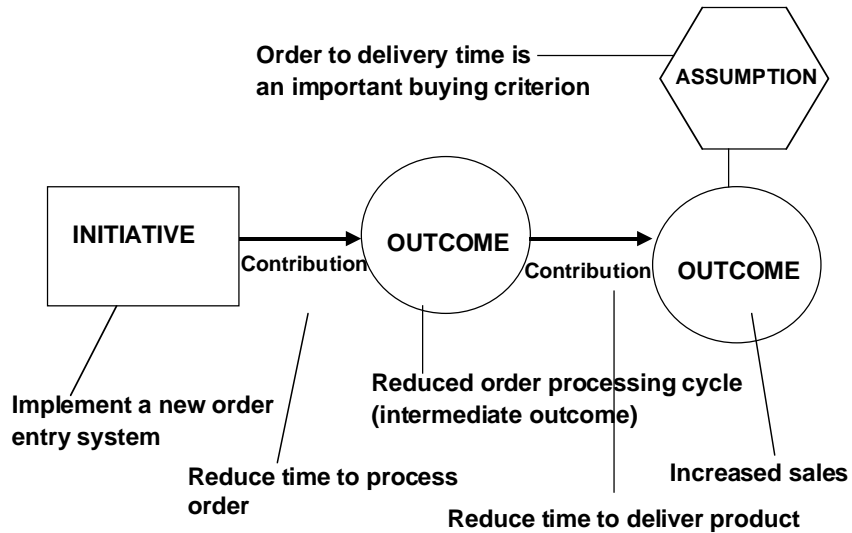
- **Software vendor has option to make product OS-portable**
 - Costs: more effort; slower performance; later time to market
 - Benefits: option to sell product in more OS market sectors
- **What is the net present value NPV of the option?**
$$\text{NPV} = \sum \text{PV} (\text{benefit flows}) - \sum \text{PV} (\text{cost flows})$$
- **Key challenge: estimating $\sum \text{PV} (\text{benefit flows})$ for various OS options**
 - Windows, Mac, Unix, ...
- **Real options approach**
 - Can approximate $\sum \text{PV} (\text{benefit flows})$ by relative present market value of OS supplier firms
 - Used successfully in electronic components industry

6/8/00

©USC-CSE

16

DMR/BRA Results Chain

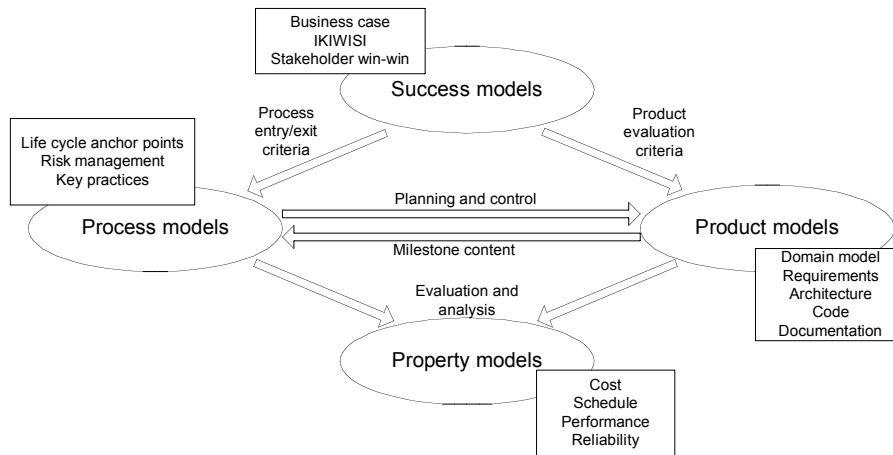


6/8/00

©USC-CSE

17

MBASE Integration Framework

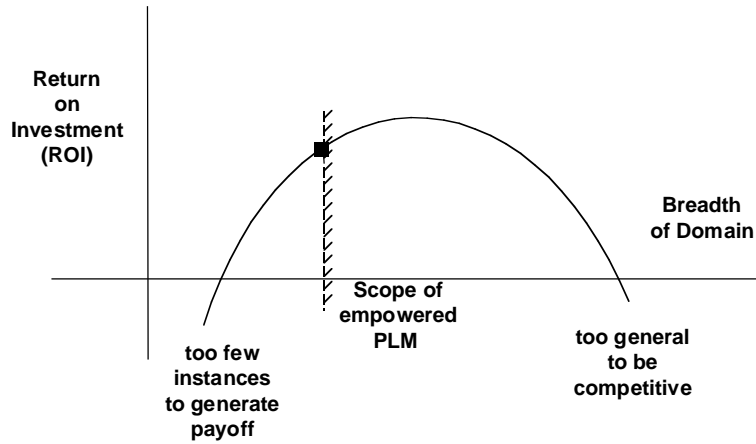


6/8/00

©USC-CSE

18

Product Line Domain Scope a Function of ROI, Scope of Empowered PL Manager

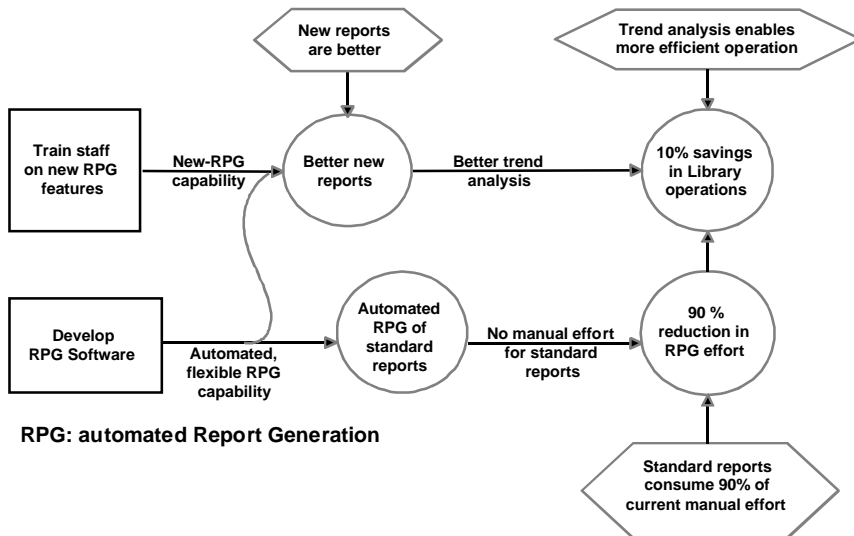


6/8/00

©USC-CSE

19

Results Chain for Digital Library Application



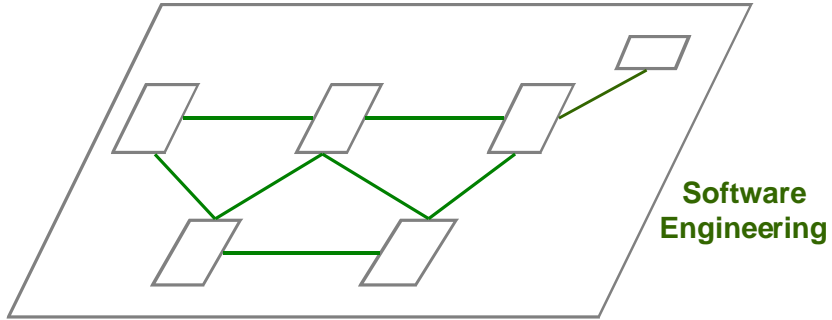
RPG: automated Report Generation

6/8/00

©USC-CSE

20

A Software Engineering Perspective

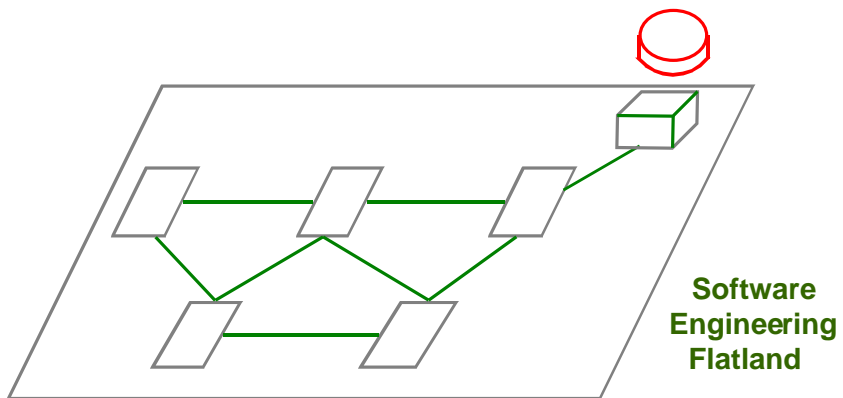


6/8/00

©USC-CSE

21

A Software Engineering Perspective

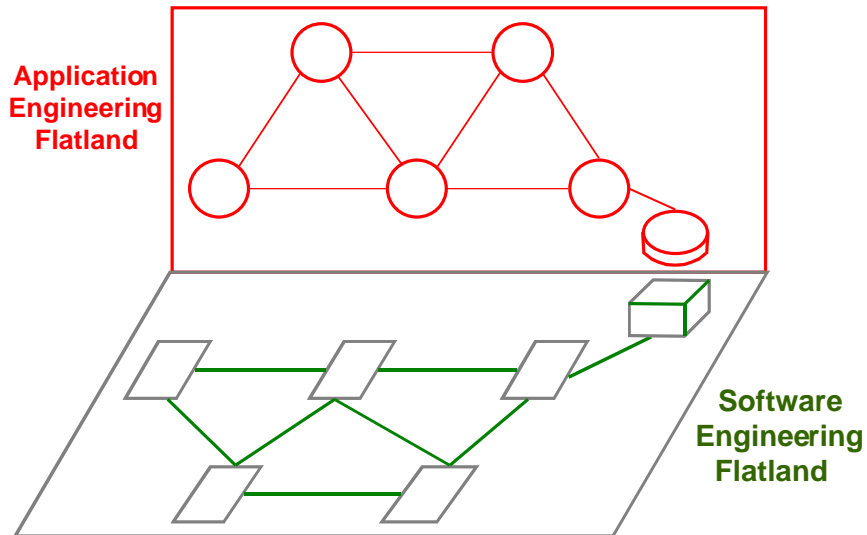


6/8/00

©USC-CSE

22

A Software Engineering Perspective



6/8/00

©USC-CSE

23

Conclusions: I

- **The software field exists because processed information has value**
- **Understanding and working with information-value effects is in our enlightened self-interest**
- **The Roadmap provides a starting point for doing this**
 - Value-oriented SW & IT models, metrics, and methods
 - Rethinking of SW & IT design, process maturity, education, ...

6/8/00

©USC-CSE

24



Conclusions: II

- **The Roadmap is a benefits-realization Results Chain**
 - It links initiatives, contributions, outcomes, and assumptions
- **Key assumption: there are enough software- and economics-knowledgeable people to pursue the initiatives**
 - We need more software people to emerge from an economics-unaware Flatland
 - We need to help more non-software people emerge from their software-unaware Flatlands