MODELING SOFTWARE PROCESSES WITH SYSTEM DYNAMICS: CURRENT DEVELOPMENTS

Raymond J. Madachy
USC Center for Software Engineering
Litton Data Systems

1996 International System Dynamics Conference
July 22-25, 1996

Outline

- **Introduction and brief history**
- **Process improvement initiatives**
- **Software process applications and current work**
- **Research issues and future work**
- **References**

Software Development and Systems Thinking

- Competitive advantage is increasingly dependent on software development in many industrial sectors.
- Software development, a dynamic and complex process, requires systems thinking in order to improve in current environment.
  - Software process: a set of activities, methods, practices and transformations used by people to develop software.
- Models can be used to quantitatively evaluate the software process
  - can experiment with changed processes before committing project resources
  - interactive training for software managers; "process flight simulation"
  - implement process re-engineering and benchmark process improvement

Brief History

1984	Tarek Abdel-Hamid completes Ph.D. dissertation at MIT
late 1980s	NASA JPL and a few others begin research with system dynamics
1991	Tarek Abdel-Hamid and Stuart Madnick publish *Software Project Dynamics*
1991-1996	Many industrial and academic implementations, including the effects of process improvement initiatives (see References).

Software Process Initiatives

- Software Engineering Institute (SEI)
  - Capability Maturity Model (CMM) for process improvement
- ISO certification
- SPICE
- ESPIRIT, others
- Business process re-engineering
- Several software metrics initiatives

Comparison of Modeling Paradigms

- Software engineers already employ a host of models
  - predictive static cost models
    - these are being extended with dynamic modeling
  - discrete event approaches for low-level process descriptions
    - generally lack feedback
    - comparison study underway

Discipline Comparison

- Software engineers are particularly well-suited for the system dynamics modeling process
  - systems view and programming experience
  - comfortable with levels of abstraction
  - similar heuristics and incremental development process used
- Model validation involves problems unique to system dynamics simulation

Software Process Level Instances

- Work artifacts (requirements, tasks, lines of code, function points, documentation pages, others)
- Defect levels
- Personnel levels
- Effort expenditure
- Schedule date
- Others

Basic Flow Processes and Infrastructures

- Software product transformations
- Error co-flows
- Error detection and rework
- Personnel experience pools and effort expenditures
- Cost/quality tradeoffs enabled

Model Validation

- Controversial issue in the software community
- Multi-perspective validation with quantitative and qualitative criteria needs to be "sold" and accepted
- Often confusion between point prediction and "understanding"
- Aided by improvements in metrics collection

Model Implementations

- Industry/government: AT&T, Bellcore, Draper Labs, Litton, Mitre, NASA, Siemens, others
- Academic: ASU, Imperial College, Stanford, MIT, Naval Postgraduate School, USC, others
- Tool vendors/workshops: Bartz Associates, Dynamica, Rubin Systems
- Many other companies are evaluating system dynamics for process improvement
- Several academic research projects in proposal stage

Process Evaluation

- Investigating the dynamic effects of inspections [Madachy 94], [Tvedt 95]
- Incremental development [Tvedt 95]
- Unit testing phase [Collafello et al. 96]
- Requirements phase (several)
- Investigating software reuse from a macro-inventory perspective [Abdel-Hamid 93a]
- Process model tradeoffs

University of Southern California Center for Software Engineering

Process Evaluation (continued)

- Other process improvement investments
  - staffing policies
  - work environment investments
  - computer aided tool investments
  - staff training investments
  - metrics, reuse, risk management and others
- Global software process feedback, stability and product evolution [Lehman et al. 96]

University of Southern California Center for Software Engineering

Flight Simulators

- Personnel training
  - graduate software project management (ASU)
  - vendor tools (Rubin et al.)
- Navigating new skies
  - process maturity initiatives
- Stimulate dialogues for shared mental models
- Virtual reality for court cases

University of Southern California Center for Software Engineering

Other Applications

- Integration with cost estimation models
  - improving on static assumptions [Madachy 95], [Rubin et al. 95]
  - calibrations between [Madachy 95]
  - deriving static parameters with dynamic experiments [Madachy 95]
- Knowledge-based assistance
  - heuristic project risk analysis and input checking [Madachy 94]
  - input evaluation and change recommendation [Lin et al. 92]
  - QA expert simulator
- Examining heuristics
  - Brookes' Law (several)
  - cost estimation correction processes [Abdel-Hamid 93]
  - others

University of Southern California Center for Software Engineering

Sample Insights

- Inspection policy tradeoff analysis - diminishing returns from inspections as a function of error generation rates [Madachy 94]
- QA policy tradeoff analysis - finding the optimal QA effort [Abdel-Hamid/Madnick 91]
- Rework staffing allocation [Tvedt 95]
- Organizational process improvement transition requires temporary productivity setbacks [Rubin, Johnson, Yourdon 95]

University of Southern California Center for Software Engineering

Sample Insights (continued)

- Leverage of experienced staff (several)
- Internal workings of Brookes' Law - training and communication losses [Abdel-Hamid 93]
- Schedule compression not a static decision [Abdel-Hamid 90]
- Anchor-dragging in project control [Abdel-Hamid 93]
- Competing feedback loops in software reuse factory [Abdel-Hamid 93b]
- Many others

University of Southern California Center for Software Engineering

Directions for Future Work

- Model structures
- Common models and component reusability
- Usability
- Process model selection
- Knowledge-based techniques
- Object orientation
- Related simulation research
- Industrial data analysis

References

- BOOKS
  - Abdel-Hamid T, Madnick S, *Software Project Dynamics*, Englewood Cliffs, NJ, Prentice-Hall, 1991
  - Madachy R, *Software Process Modeling with System Dynamics*, IEEE Computer Society Press, Washington, D.C., 1997 (current plan)
- JOURNAL ARTICLES, CONFERENCE PROCEEDINGS AND OTHERS
  - Abdel-Hamid T, *The dynamics of software project staffing: A system dynamics based simulation approach*, IEEE Transactions on Software Engineering, February 1989
  - Abdel-Hamid T, *Lessons learned from modeling the dynamics of software development*, Communications of the ACM, December 1989
  - Abdel-Hamid T, *Investigating the cost/schedule trade-off in software development*, IEEE Software, January 1990
  - Abdel-Hamid T, *Adapting, correcting, and perfecting software estimates: a maintenance metaphor*, IEEE Computer, March 1993
  - Abdel-Hamid T, *Thinking in circles*, American Programmer, May 1993

References (continued)

- JOURNAL ARTICLES, CONFERENCE PROCEEDINGS AND OTHERS (continued)
  - Abdel-Hamid T, *Modeling the dynamics of software reuse: an integrating system dynamics perspective*, Presented at the Sixth Annual Workshop on Software Reuse, Owego, NY, November 1993
  - Aranda R, Fiddaman T, Oliva R, *Quality microworlds: Modeling the impact of quality initiatives over the software product life cycle*, American Programmer, May 1993
  - Chichakly K, *The bifocal vantage point: Managing software projects from a systems thinking perspective*, American Programmer, May 1993
  - Collofello J, Yang Z, Tvedt J, Merrill D, Rus I, *Modeling Software Testing Processes*, Submitted to the International Phoenix Conference on Computers and Communications, 1996
  - Cooper K, Mullen T, *Swords and plowshares: The rework cycles of defense and commercial software development projects*, American Programmer, May 1993
  - Diehl E, *The analytical lens: Strategy-support software to enhance executive dialog and debate*, American Programmer, May 1993

References (continued)

- JOURNAL ARTICLES, CONFERENCE PROCEEDINGS AND OTHERS (continued)
  - Glickman S, *The Bellcore-CSELT collaborative project*, Proceedings of the Ninth International Forum on COCOMO and Software Cost Modeling, USC, Los Angeles, CA, 1994
  - Johnson M: *Dynamic Systems Modeling: The Software Management Process*, Bartz Associates, 1995
  - Lehman M,, *Process Improvement - The way forward*, Proceedings of CAiSE 95, Jyvaskyla, June 1995
  - Lin C, Levary R: *Computer-aided software development process design*, IEEE Transactions on Software Engineering, September 1989
  - Lin C, Abdel-Hamid T, Sherif J: *Software-engineering process simulation model*. TDA Progress Report 42-108, Jet Propulsion Laboratories, February 1992
  - Lin C, *Walking on battlefields: Tools for strategic software management*, American Programmer, May 1993
  - Madachy R, *A software project dynamics model for process cost, schedule and risk assessment*, Ph.D. dissertation, Department of Industrial and Systems Engineering, USC, December 1994

References (continued)

- JOURNAL ARTICLES, CONFERENCE PROCEEDINGS AND OTHERS (continued)
  - Madachy R, *Knowledge-based risk assessment and cost estimation*, Automated Software Engineering, Kluwer Academic Publishers, September 1995
  - Madachy R, *Process improvement analysis of a corporate inspection program*, Proceedings of the Seventh Software Engineering Process Group Conference, May 1995
  - Madachy R, *System Dynamics and COCOMO; Complementary Modeling Paradigms*, Proceedings of the Tenth International Forum on COCOMO and Software Cost Modeling, SEI, Pittsburgh, PA, 1995
  - Madachy R, *System Dynamics Modeling of an Inspection-Based Process*, Proceedings of the Eighteenth International Conference on Software Engineering, IEEE Computer Society Press, Berlin, Germany, March 1996
  - Rubin H, Johnson M, Yourdon E, *Software process flight simulation: dynamic modeling tools and metrics*, Information Systems Management, Summer 1995

References (continued)

- JOURNAL ARTICLES, CONFERENCE PROCEEDINGS AND OTHERS (continued)
  - Smith B, Nguyen N, Vidale R, *Death of a software manager: How to avoid career suicide through dynamic process modeling*, American Programmer, May 1993
  - Tvedt J, Collofello J, *Evaluating the effectiveness of process improvements on software development cycle time via system dynamics modeling*, Proceedings of the Computer Software and Applications Conference, 1995
- WORLD WIDE WEB
  - My system dynamics site, including updates on forthcoming book: *http://www-rcf.usc.edu/~madachy/sd*
  - Arizona State University Process Modeling Group : *http://www.eas.asu.edu/~sdm/*