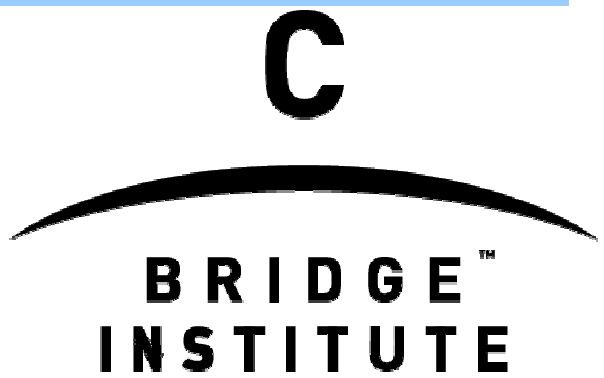
To Main Proceedings Document

Recent Results in Software Process Modeling

Ray Madachy

Eighteenth International Conference of the System Dynamics Society
August 6-10, 2000

University of Southern California
Center for Software Engineering



Introduction

- USC-CSE has added a new course *CS599 Software Process Modeling*
- Course objectives
  - review the field of software process modeling
  - develop systems thinking skills for developing increasingly deep understandings of software process structures
  - show basic building blocks and model infrastructures for software development processes
  - describe the modeling process, including calibration of models to software metrics data
  - provide details of critical implementation issues and future research motivations
  - develop simulation term projects that address critical software issues
- This presentation will overview this year's student projects from the Fall 1999 course

CS599 *Software Process Modeling*
Student Term Projects

- Dynamics of architecture development process in MBASE inception and elaboration phases
- COTS glue code development and integration dynamics
- Reuse and language-level effects in software development
- CMM-based software process improvement strategies
- Application of RAD techniques to pre-IPO Internet companies

Scope / Purpose	Portion of lifecycle	Development project	Long-term organization
Planning and control	MBASE Architecting	Reuse and High Level Languages Internet RAD COTS Glue Code	CMM Software Process Improvement
Process improvement and technology adoption			

MBASE Architecting Modeling Goals

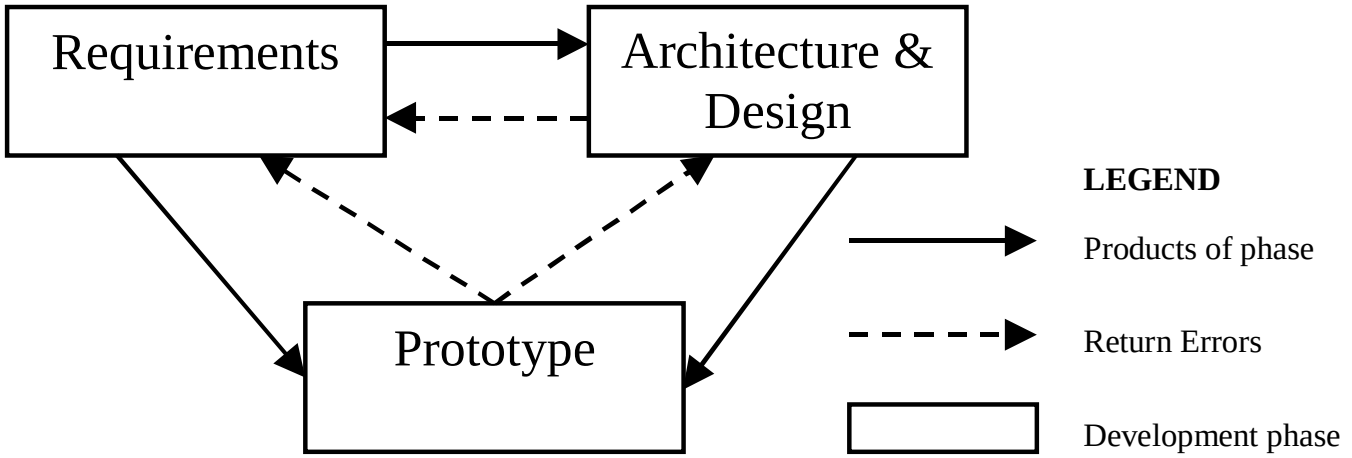
- Investigate the dynamics of architecture development during early MBASE lifecyle phases
- Identify nature of process concurrence in early MBASE phases
- Understand impact of collaboration and prototyping on lifecycle parameters

MBASE Architecting Model Features

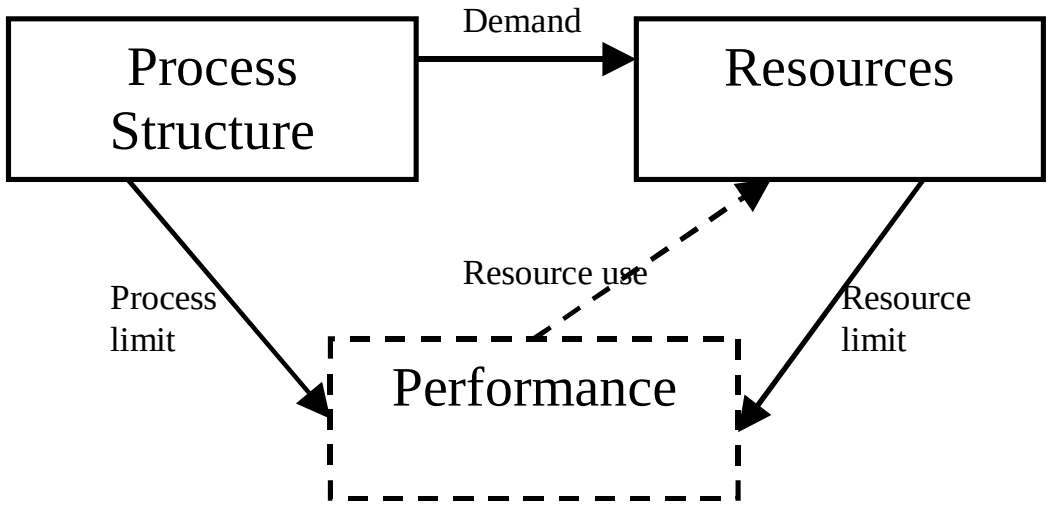
- Schedule as independent variable
- Contains iterative process structures
- Covers sequentiality and concurrency
  - phases: requirements and architecture/design
  - activities: initial completion, coordination, quality assurance, iteration
- Demand-based resource allocation
- External and internal precedence constraints
- Calibrated to CS577A data

MBASE Architecting Model Overview

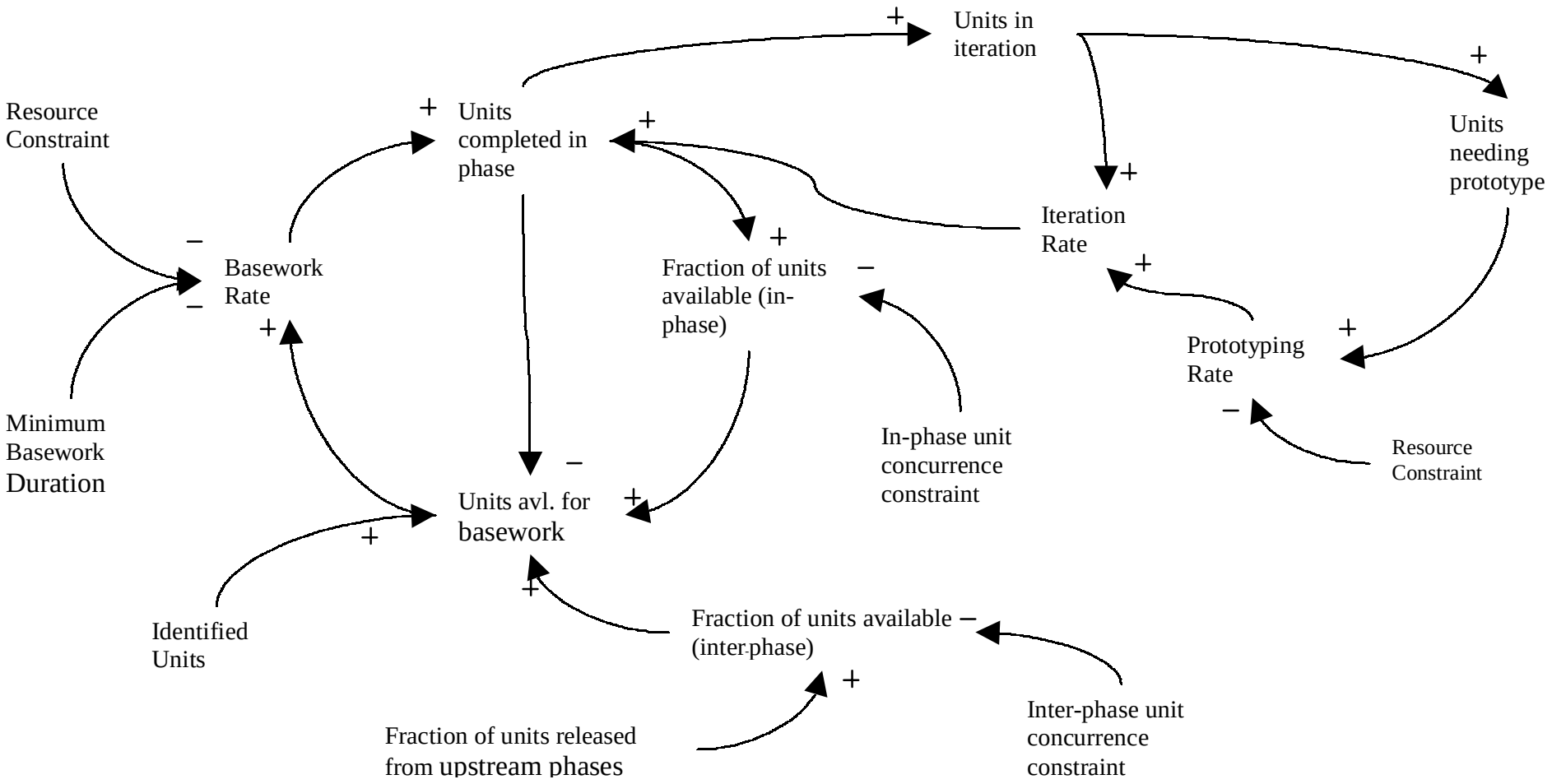
- Lifecycle artifacts



- Model structure



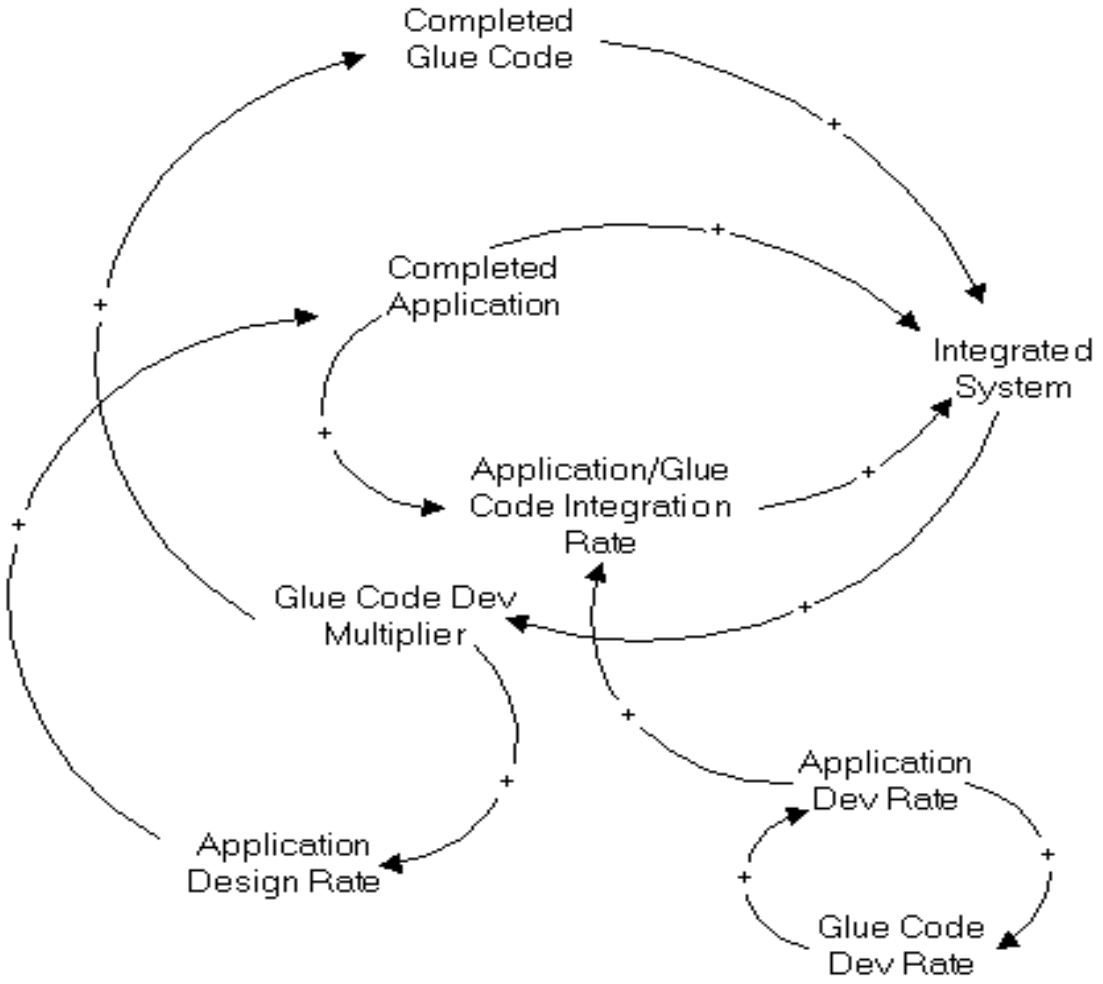
MBASE Architecting Causal Loop

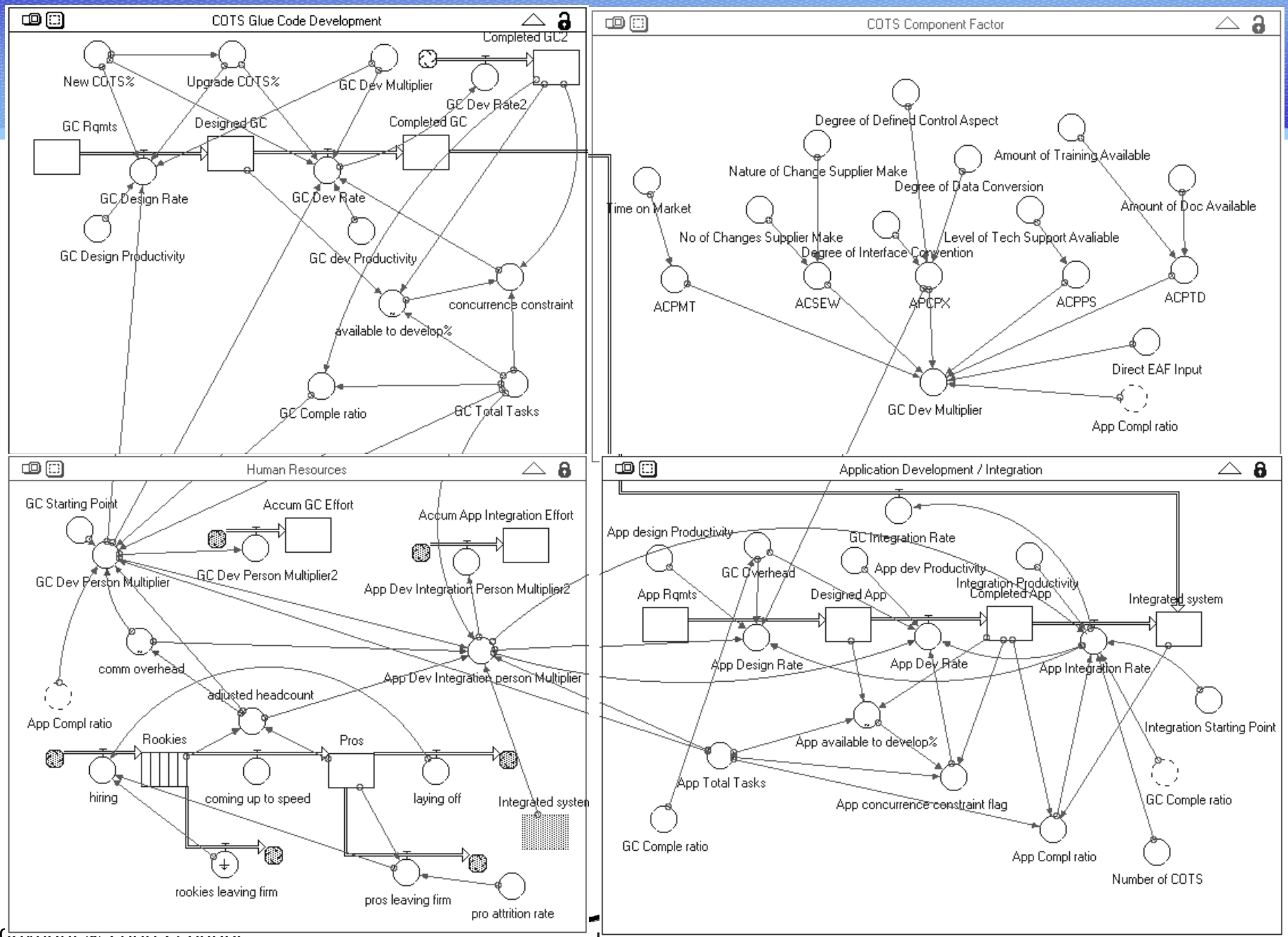


COTS Modeling Goals

- To understand glue code development and COTS integration process and their correlation
- To determine efficient starting points of glue code development and COTS integration
- To calibrate the component parameters from COCOTS
- To analyze the impact of new parameters such as ratio of new and updated COTS component and number of COTS component

COTS Model Causal Loop

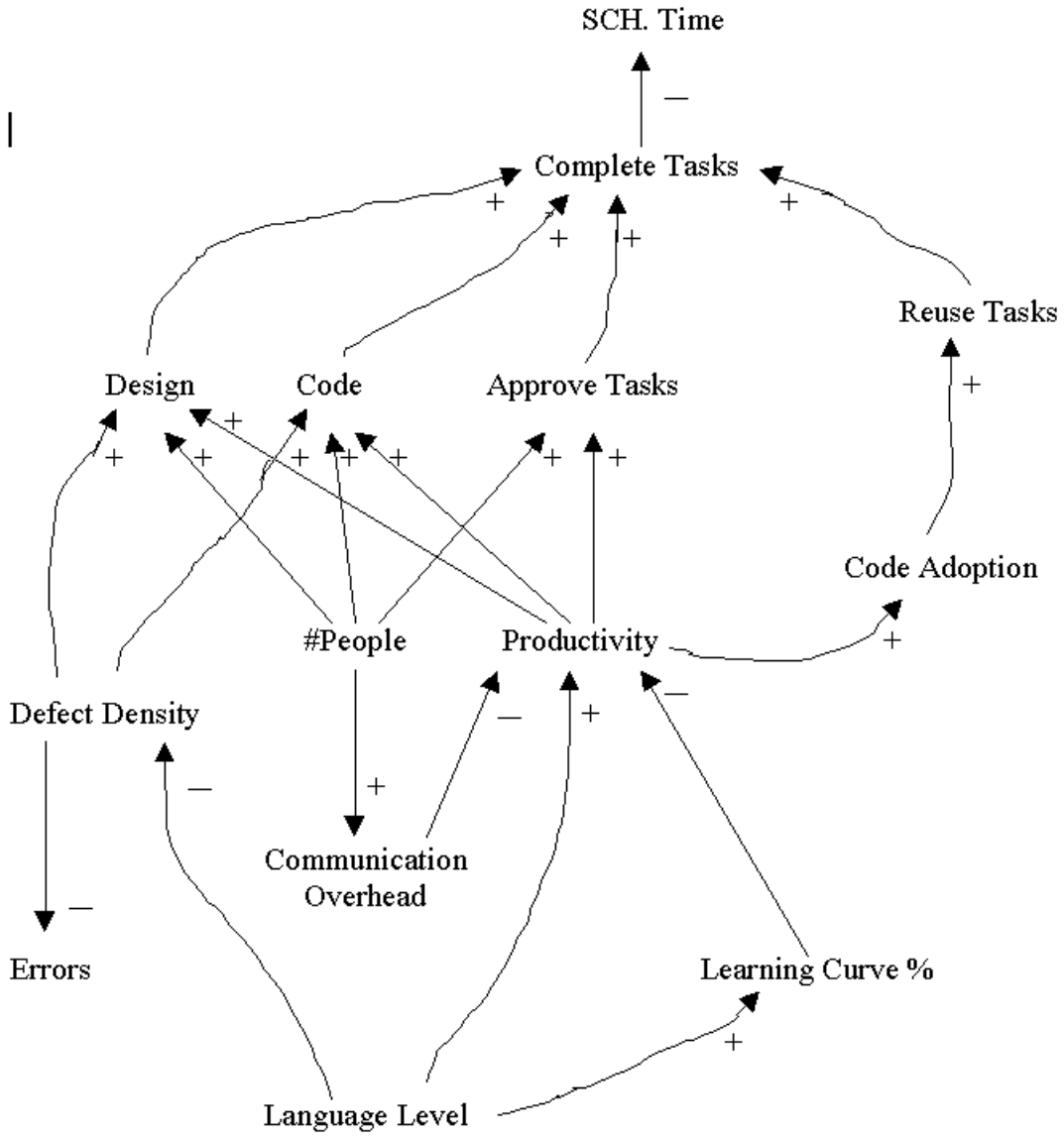




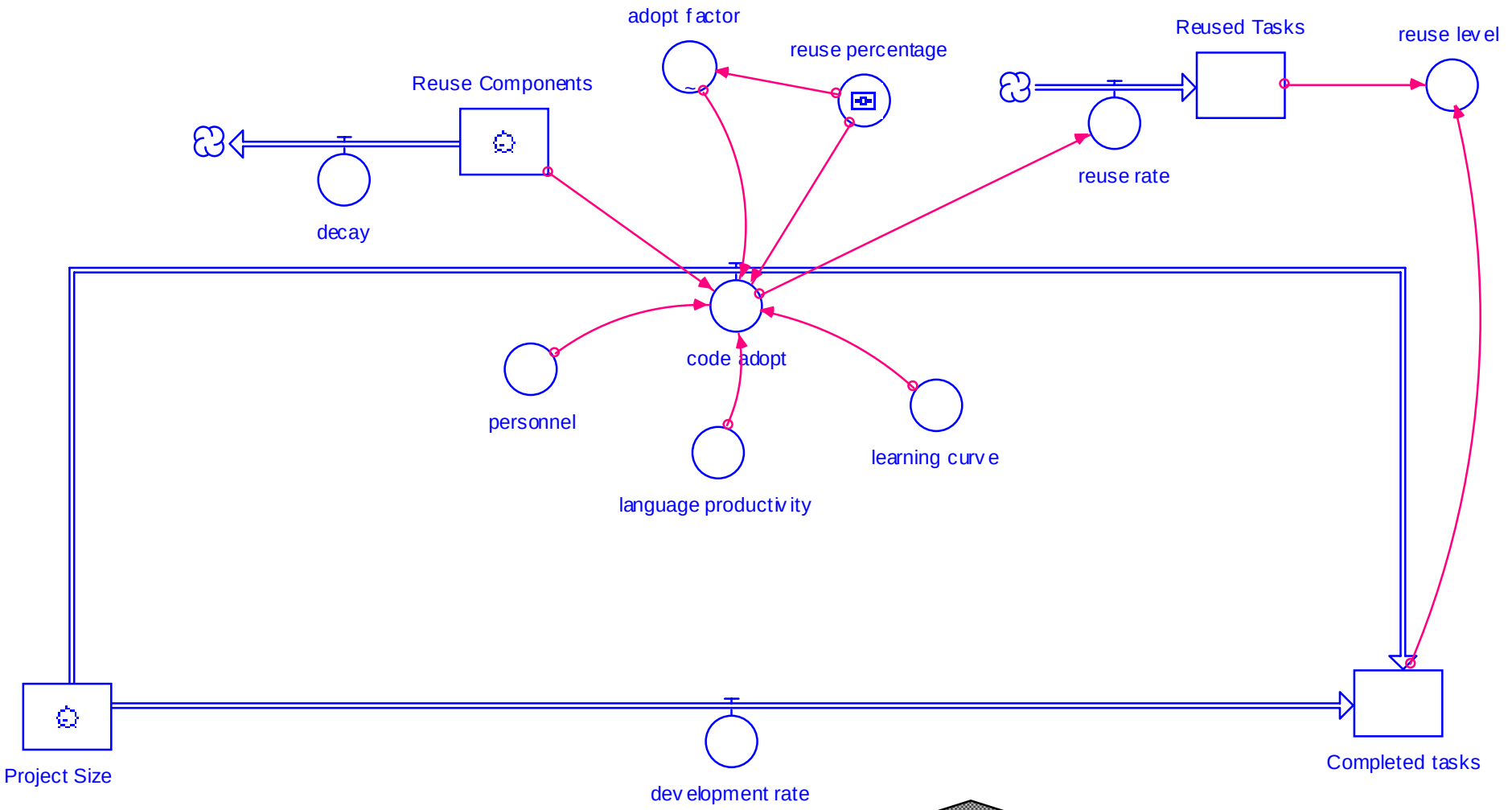
Reuse and High Level Language Modeling

- Goals
  - Investigate project reuse dynamics
    - productivity and effort of individual phases
  - Understand effects of different language levels
- Model features
  - Rework included
  - Learning curve formulations
  - Increased training for higher level languages

Reuse Model Causal Diagram



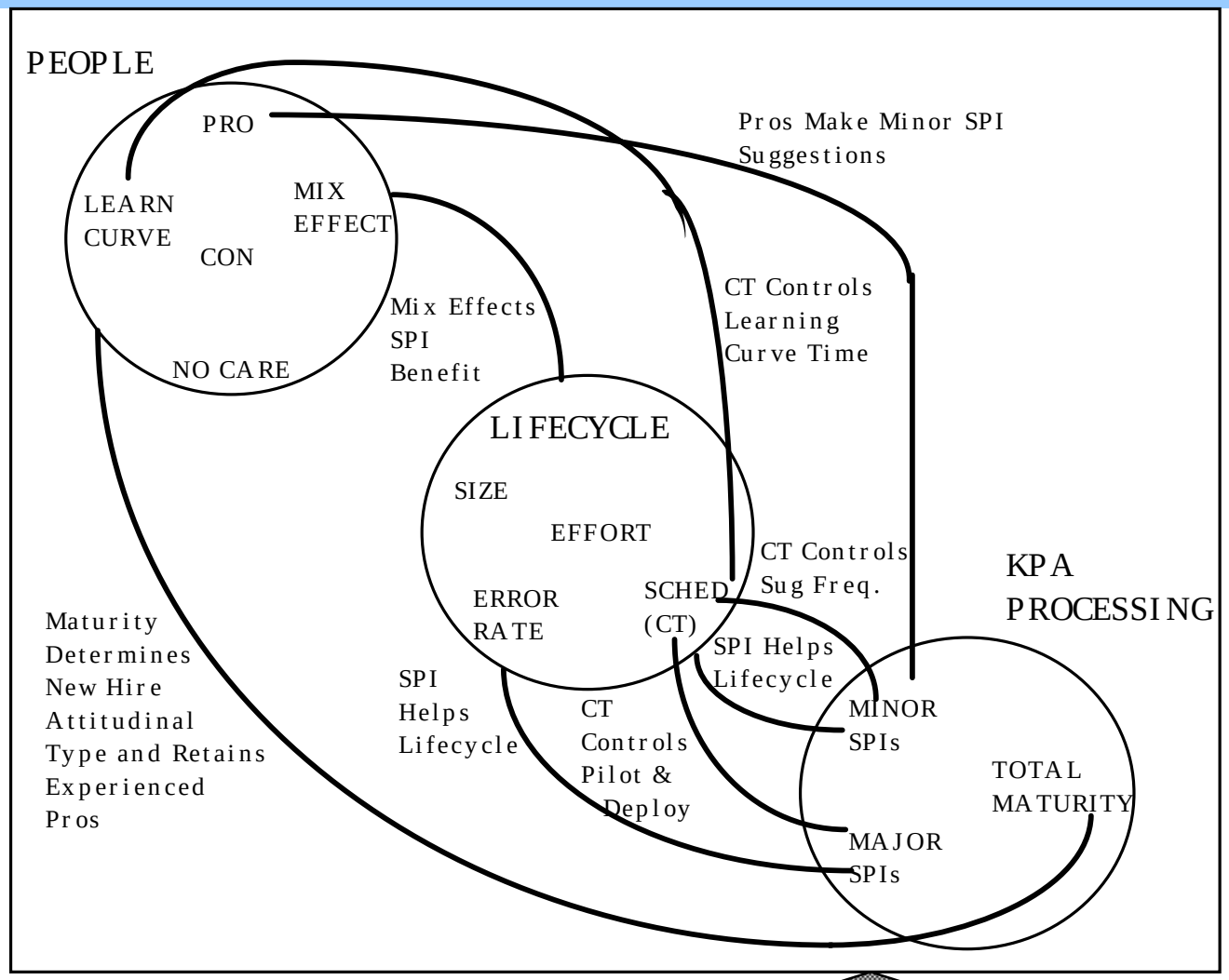
Reuse Structure



CMM-Based Software Process Improvement Study

- Goal: to research and produce a system dynamics model for Software Process Improvement based on the CMM model
- Provides insights into complex process behavior
  - help evaluate different approaches
- Support planning, tracking and prediction
  - reduce costs
  - reduce cycle time
  - reduce defects
- Based on the scenario of a Xerox S/W development group working from just assessed as a Level 2 organization moving towards achieving Level 3.

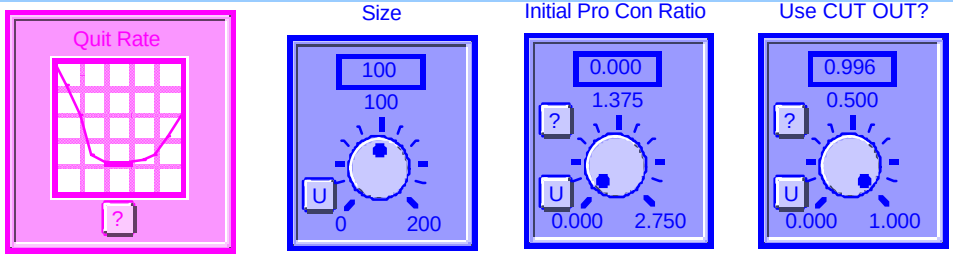
Process Improvement Model Structure

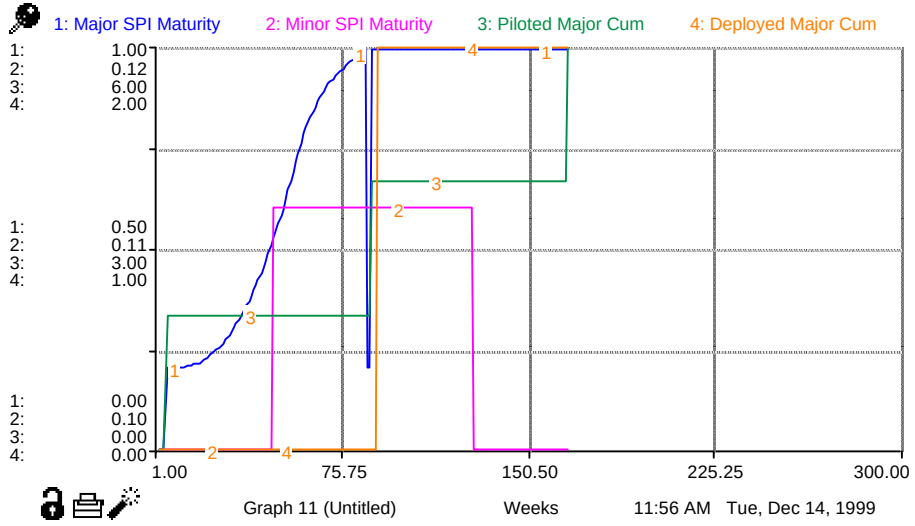


Process Improvement Model Overview

- The ***Lifecycle*** process models how software size, effort, quality, and schedule relate to each other in order to produce a product.
- SPI benefits are modeled as percent reductions in either size, effort, error rate or schedule.
- In ***Staff***, three attitudes of staff that affected the potential benefit of process improvement: pro-SPI people, con-SPI people, and no-care people.
- The attitudinal mix and the pro/con ratio can affect the overall potential benefit realized by a SPI effort.
- ***KPA Activities*** models the timing of the flow of process improvements into the life-cycle and people subsystems.

Process Improvement Control Panel





Effort Via SPI&Size&...	13.06
Errors	0.98
Schedule	79.18
Size	88.97
Pro Exp	9.3
Con Exp	19.0
NC Exp	29.2
Total Maturity	0.1
Major SPI Maturity	1.0
Minor SPI Maturity	0.1
Deployed Major Cum	2.0
Piloted Major Cum	6.0

Return to Map | Run | Pause | Stop | To Control Panel II

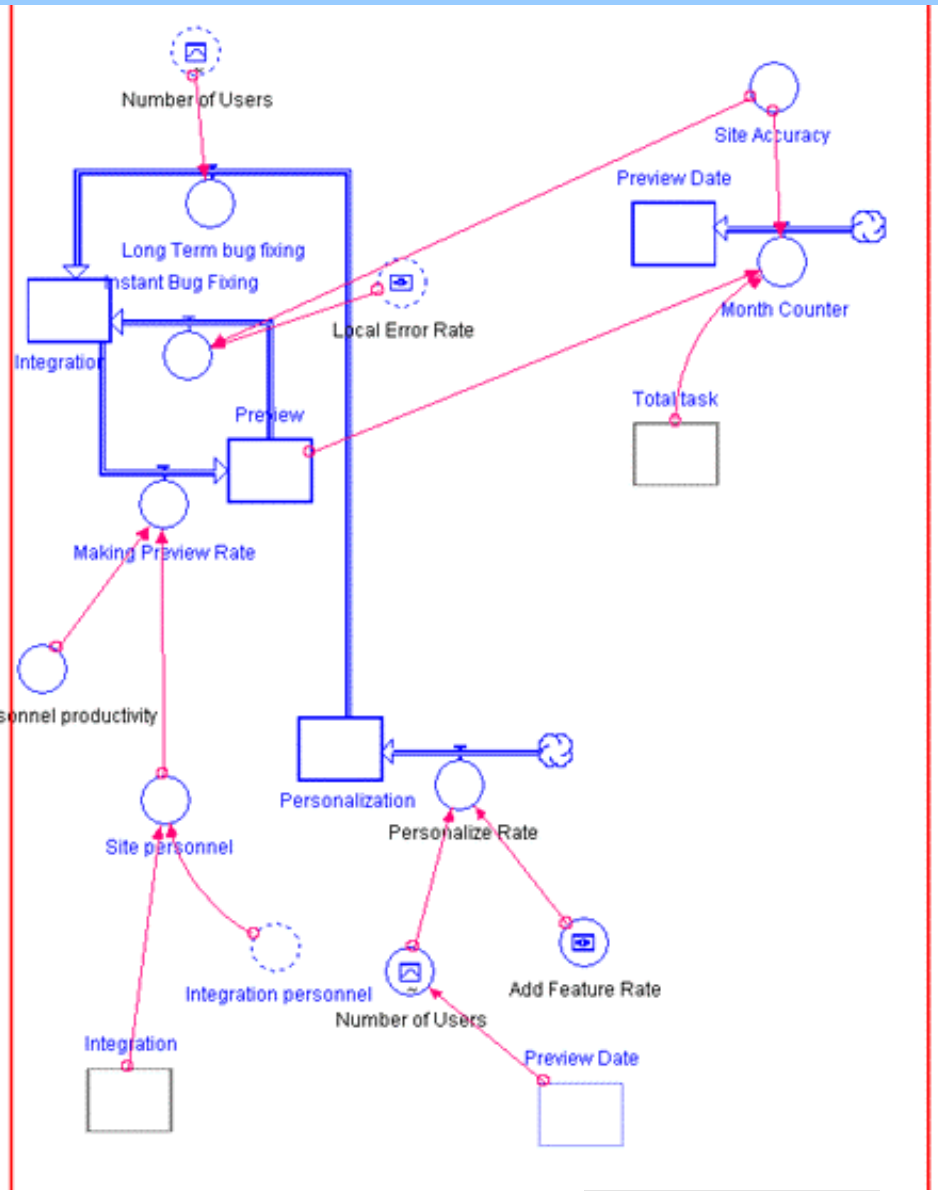
Internet RAD Modeling Goals

- Investigate dynamics of pre-IPO Internet companies
- Contrast to non-Internet software development
- Survey companies and determine major impact factors

Internet RAD Model Features

- Modified evolutionary delivery lifecycle with small teams
  - schedule minimization
- Outsourcing considerations
- Defect detection and elimination
  - short term and long-term feedback
- Includes Internet preview and web-site personalization
- Model sectors: Specification and Design, Outsourcing, Development, Integration and Personalization, Human Resources

Internet RAD Integration and Personalization



References

- Madachy R, *Tutorial: Cost/Schedule/Process Modeling via System Dynamics*, Proceedings of the 14th International Forum on COCOMO and Software Cost Modeling, Los Angeles, CA, October 1999

USC-CSE Web Sites

- **http://sunset.usc.edu/classes/cs599_99**
  - USC-CSE Software Process Modeling Course (includes other system dynamics links)
- **http://sunset.usc.edu/Research_Group/ray/spd**
  - portions of forthcoming book: Madachy R, Boehm B, *Software Process Dynamics*, IEEE Computer Society Press, Washington, D.C., 2001