Software Models

What Model is Right for Me?

July 2002

by

Joe Dean

DELUSIONS

There is no greater joy than soaring high on the wings of your dreams.
Except maybe the joy of watching a dreamer who has no where to land
but in the sea of reality.

S/W Model History

Fedrick	1974	SLIM	1981
Walston-Felix	1977	JS-1	1983
Jones	1977	SEER	1986
Halstead	1977	SASET	1986
Schneider	1978	REVIC	1987
Freburger-Basili	1979	Sage	1995
PRICE	1979	COCOMO II	1995
COCOMO	1980	CostXpert	1997

Software Estimating Applications

- Bid/No Bid
- Development
- Maintenance
- Modifications
- LCC Analysis
- Risk Analysis
- Should Cost Analysis
- Procurement Quote Analysis
- Negotiation Analysis
- Estimate to Complete
- CAIV Tradeoff Analysis

Software Cost Drivers

- Program Size
- Percentage of New Design/Code
- Documentation Level
- Memory/Timing Utilization in Computer
- Stability of Requirements
- Complexity of Concurrent Hardware Design
- Schedule
- End-Use Environment
- Personnel
- Management

Fundamental Effort Equation

$$E=CS^{a}$$

E = Development Effort

C = Environment Calibration Constant

S = Lines of Source Code (includes reuse effects)

a = Entropy Constant

Typical Software Development Activities

- Planning Phase
- System Requirements
- System Allocation
- Software Requirements
- Preliminary Software Design
- Detailed Software Design
- Code & Debug
- Unit Test
- Software Integration
- System Integration
- Acceptance Testing
- Formal Verification
- Maintenance

Golub's Laws of Computerdom

- Fuzzy project objectives are used to avoid the embarrassment of estimating the corresponding costs.
- A carelessly planned project takes three times longer to complete than expected; a carefully planned project will take only twice as long.
- The effort required to correct the course of development increases geometrically with time.
- Project teams detest weekly progress reporting because it so vividly manifests their lack of progress.

Current Widely Used Models

COCOMO

PRICE

SEER

SLIM

REVIC

SASET

Sage

CostXpert

COCOMO Background

- Developed by Dr. Barry Boehm
- It a Project Based Model
- Based on 63 Programs from 1964 - 1979
- Ten were greater than 100KSLOC
  - Six Embedded
  - Four Semidetached
  - Zero Organic
- One was greater than 500KSLOC
- Has become the basis for many “XYZOMO” models

COCOMO
What it Does

- Has three basic modes of capability
  - **Basic** - Project size input only
  - **Intermediate** - EAF and Project/Component size input
  - **Detailed** - EAF and Component/Module size input
- Allows for application groupings
  - Embedded , Semidetached, Organic
- Estimates optimum schedule as a default
- Provides Maintenance estimates with Annual Change Traffic

COCOMO
Feeding and Maintenance

- Free
- Different Modes can be used throughout the lifecycle
- Should use existing data as a “benchmark” for calibration for your own environment

PRICE Background

- Developed by RCA PRICE Systems with Dr. Bob Park as lead for the PRICE - S and Dr. Bill Kuhn for the PRICE - SL Model
- It is a CSCI Based Model
- Based on Expert Opinions and applied to many RCA programs for validation developed in the 70s
- Hosted on a mainframe with dial-in capability until the late 80s, when a PC based system was released

PRICE
What it Does

- Provides guidance for typical inputs of Application, PROFAC, and Complexity values
- Size inputs can be either LOC or Object Points
- Allows the user to calibrate each of fifteen functional areas within a Matrix type of organization
- Provides an optimal cost and schedule solution
- PRICE SL estimates Maintenance estimates including growth and enhancements

PRICE
Feeding and Maintenance

- Requires an annual Fee for use and updates
- If the model is calibrated to an organization and it "reorganizes" then it must be recalibrated
- Should use existing data as a "benchmark" for calibration of PROFAC and APPL values

SEER Background

- Was based on JS-1 Software equations developed by Dr. Randy Jensen and has been modified by Dan and Judy Galorath to its present form
- It is a CSCI Based Model
- Original model was based on Hughes Data and has been adjusted with data received on recent programs collected by Galorath and Associates
- Is a PC based system

SEER-SEM
What it Does

- Variety of pre-determined “Knowledge Bases” to allow for quick “ball-park” estimates
- Sizing inputs can be either LOC or Function Points
- Provides a Minimum and Optimal Schedule Cost Solution
- Allows for Maintenance estimates including growth and enhancements

SEER-SEM
Feeding and Maintenance

- Requires an annual Fee for use and updates
- EAFs need to be adjusted for the developer's environment
- Should use existing data as a "benchmark" for calibration

SLIM Background

- Was based on RADC software data collected in the 60s This data consisted of a cluge of over 1,000 data points.
- Larry Putnam noticed that even though the data was widely disbursed it fit nicely on a Rayleigh-Norden 60/40 distribution curve.
- Is a Project Based Model
- The model is updated regularly with data received on recent programs collected by the Putnams
- Is a PC based system

SLIM
What it Does

- Uses default “Productivity Indexes” to allow for quick “ball-park” System level estimates
- Sizing inputs can be either LOC or Function Points
- Provides a Minimum Schedule Solution
- Allows for Maintenance estimates including growth and enhancements

SLIM
Feeding and Maintenance

- Requires an annual Fee for use and updates
- Should use existing data as a “benchmark” for calibration at the proper Productivity Index

REVIC Background

- This is a derivative of the Intermediate COCOMO Model based on lab developments at Albuquerque New Mexico
- Ray Kyle needed a reliable way to estimate the effort required to develop software in a unique environment where a significant amount of testing was needed
- Is a CSCI Based Model
- This model even had a users' group that met once a year
- It is a PC DOS based application

REVIC
What it Does

- Allows for four application groupings
  - Embedded , Semidetached, Organic, and Ada
- Sizing inputs are only LOC
- Estimates optimum schedule as a default
- Provides Maintenance estimates with Annual Change Traffic

REVIC
Feeding and Maintenance

- Free
- Currently no longer supported
- Should use existing data as a “benchmark” for calibration

SASET Background

- Bill Cheadle & Dr Silver from Martin Marietta in Denver Colorado initiated an SBIR with the Navy and Air Force to develop a DoD Software Estimating Model
- Primary data used was from ESC and Martin Marietta
- Is a CSCI Based Model
- It was developed to be a layered model that would use the level of the development architecture to help determine the level of difficulty for the development
- Is presently maintained by Lockheed Martin

SASET
What it Does

- Uses the Architecture, based on the software type of a system and its size to determine the basic estimate
- Sizing inputs are only LOC
- Basic Architectures include;
  - Ground
  - Avionics
  - Space
  - Commercial

SASET
Feeding and Maintenance

- Free for Government use
- Should use existing data as a “benchmark” for calibration of the Software Type Multiplier

Sage Background

- Developed by Dr. Randy Jensen to account for differences in management philosophies.
- It is a CSCI Based Model
- Based on Data Dr. Jensen has collected over the years
- Is a PC based system

Sage
What it Does

- Uses a set of pre-determined management style "Knowledge Bases" to provide a basic estimate
- Other Effort Adjustment Factors are very similar to SEER
- Sizing inputs LOC only
- Provides a Minimum and Optimal Schedule Cost Solution
- Allows for Maintenance estimates including growth and enhancements

Sage
Feeding and Maintenance

- Requires an annual Fee for use and updates
- EAFs need to be adjusted for the developer's environment
- Should use existing data as a "benchmark" for calibration

CostXpert Background

- Developed by Marotz, Inc. In San Diego Calif.
- This is a combination of a derivative of the COCOMO II Model and a Database driven model
- Is a Project Based Model
- Relativity new model with a high potential

CostXpert
What it Does

- Allows for a wide range of sizing/volume inputs, SLOC, Function Points, Feature Points, GUI Metrics, Object Metrics, Bottom up, Top Down
- Allows the user to define their own effort, schedule and distribution equations
- Provides Development and Sustainment Estimates

CostXpert
Feeding and Maintenance

- Requires an annual Fee for use and updates
- Should use existing data as a “benchmark” for calibration

Which One Should I Choose?

COCOMO

SEER

PRICE

SLIM

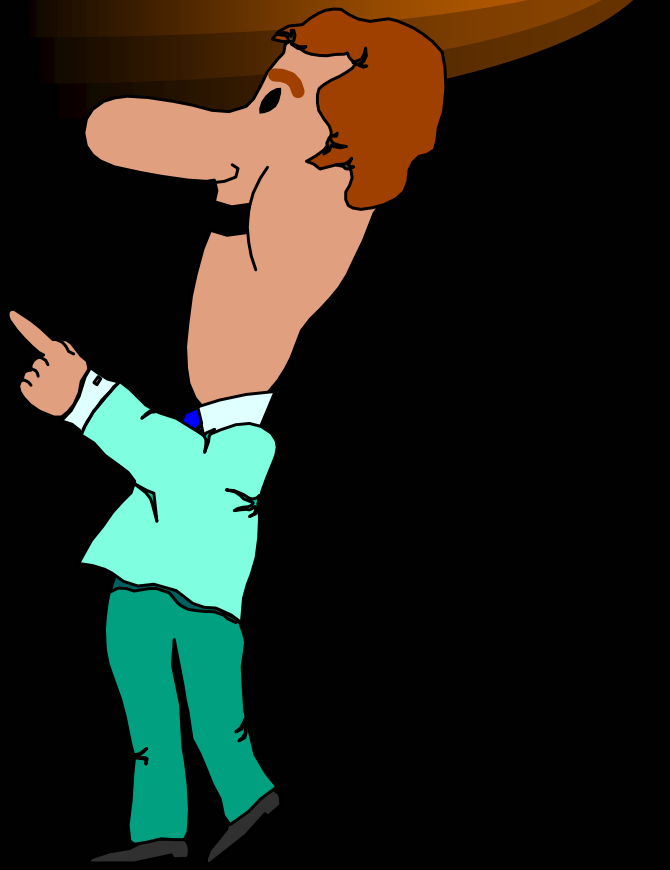
REVIC

SASET

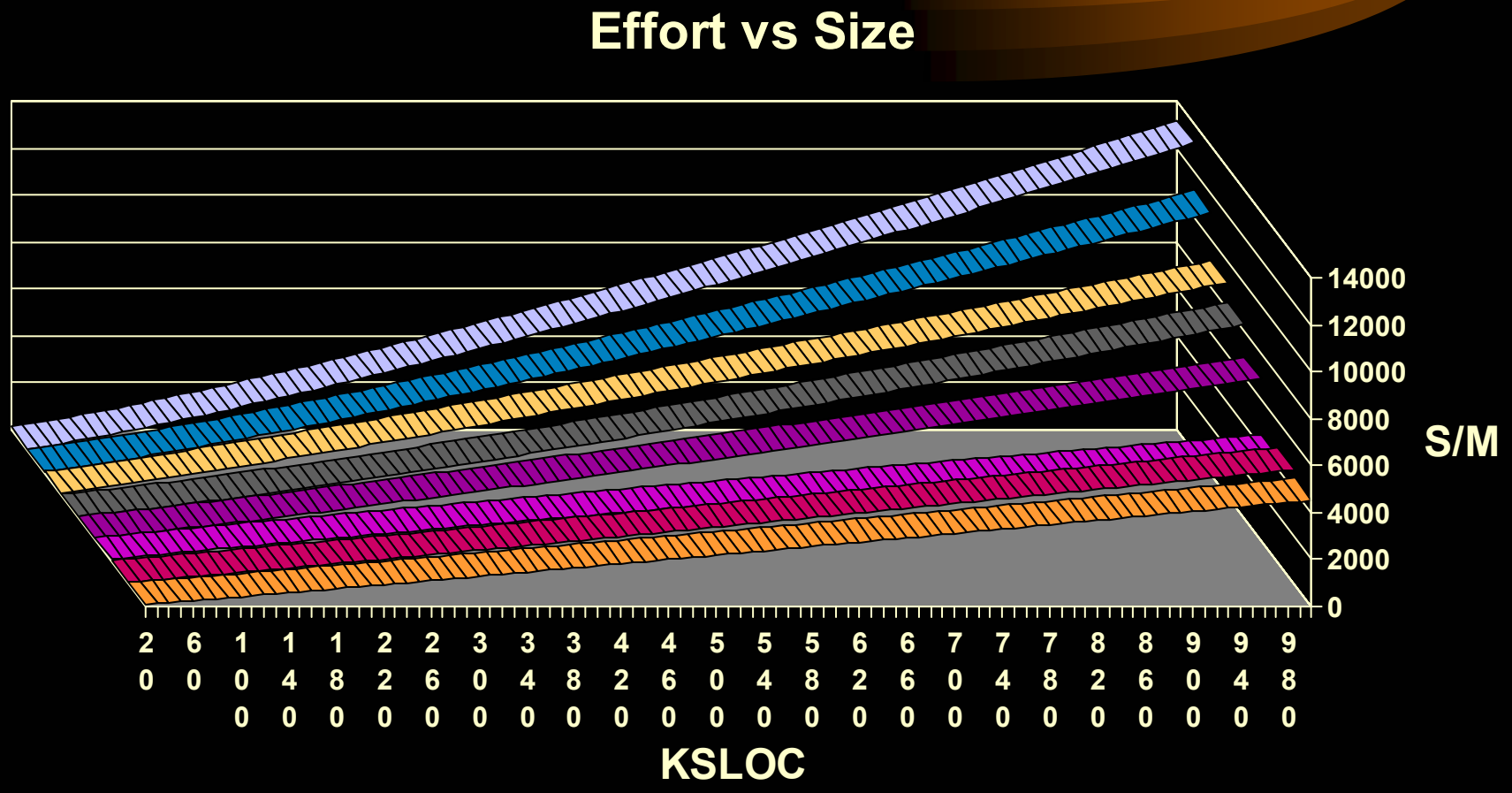
Sage

CostXpert

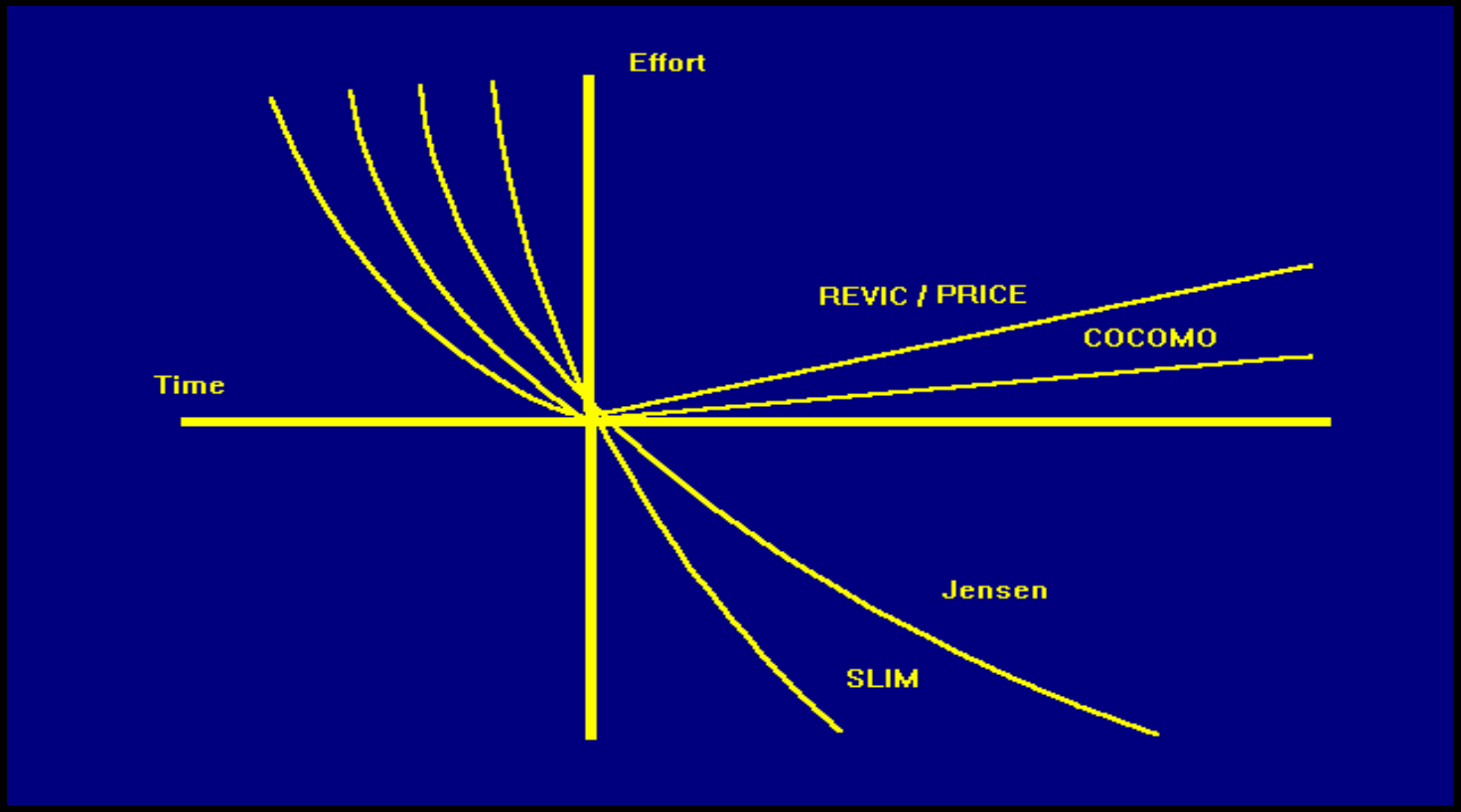
TBD



Which Model Do I Choose?



Which Model Do I Choose?



Software Productivity
The Covert Agenda

- **Apply pressure to developers to get them to work longer and harder.**
  - Promote an ethic of workaholism
  - Get project members to sacrifice personal lives
  - Gull members into accepting hopeless schedules
  - Hold members feet to the fire to make them deliver
- **Minimize product quality**
  - Over constrain projects to force compromised quality
  - Establish unwritten standard that nothing beyond Minimum quality acceptable to customer will be tolerated

What Will Future Models Address?

4GL

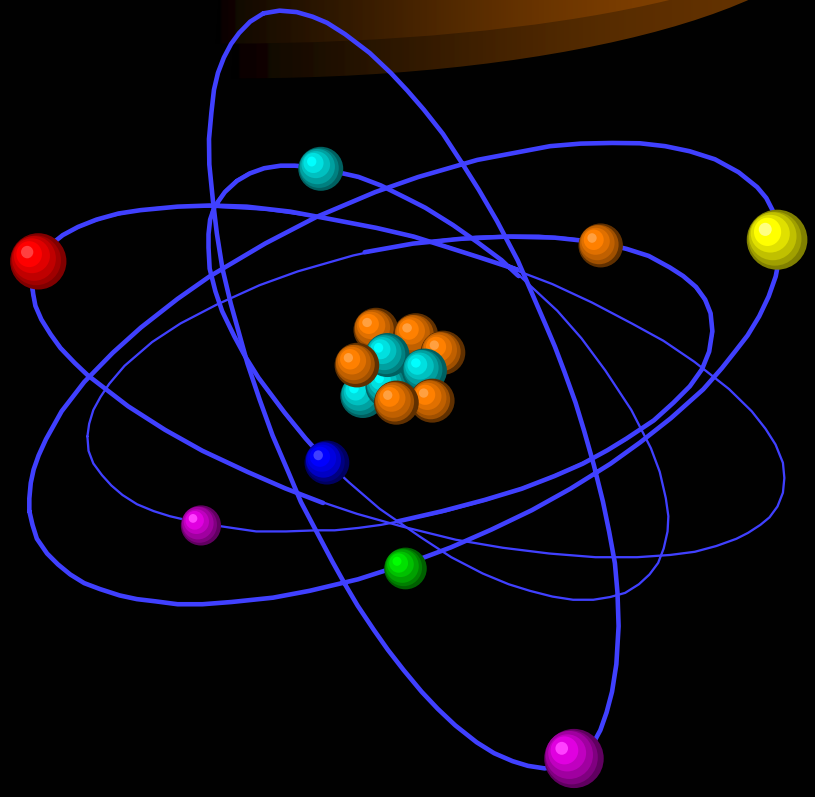
5GL

COTS

GOTS

NDI

Modernization



Using Existing Models 4 & 5GL Adjustments

- If using LOC, estimate only the to be developed code, not the generated code
- Extra effort may have to be added to the design and I&T portions of the model

Using Existing Models COTS, GOTS & NDI

- Estimate the "Glue" code needed to interface the packages to each other and to the operating environment
- Be careful of estimating the code "breakage"

Using Existing Models Modernization

- Estimate new and modified code for the new functionalities
- Estimate the breakage code for the new operating system

Conclusion

- SELECT a model that best suits your environment
- LEARN the model inside and out
- SEARCH and collect data that fits your environment
- VALIDATE & CALIBRATE the model or models you have selected

Think of me as Software