Building a Local Resource Model

Model Validation Study

Walston and Felix build a model of resource estimation for the set of projects at the IBM Federal Systems Division.

They did this by characterizing the relationship between size and effort for projects in their environment.

They studied the relationship of other variables and how they affected effort.

They also characterized other relationships, e.g., the relationship between size and number of pages of documentation.

Do the W/F equations hold in other environments? Do the same parameters affect effort in other domains.

Are there other measures of size that might be of interest? Can there be further tailoring of their model?

MODELING AND MEASURING RESOURCES

Model Validation Study in the SEL

Object of study:relationships among variables, e.g., size and effort
Purpose: Evaluation (Valid for another environment?)
Improvement (Other variables that do better?)
Focus: Quality of relationship/ fit of the data to a relationship
Point of view: Organization (management)/ researcher
Environment: SEL

Goal: Analyze the relationship between size and effort for the purpose of
evaluation (improvement)of the Walston-Felix equations with respect to the quality of the fit of the data to the curve from the point of view of the organization

Object of Study Model: $E = a*Size^b$ specifically $E = 5.2L^{.91}$

Focus Model: Standard Error of Estimate, R^2

Basili/Freburger
University of Maryland, 1979

MODELING AND MEASURING RESOURCES

Model Validation Study in the SEL

Are there other parameters of interest?

e.g., DL = number of developed/delivered source lines of code (new code + 20% reused code)

M = total number of modules = FORTRAN Subroutine I

DM= total number of developed modules (all new or more than 20% new)

P= productivity = L/E

RDTODL =ratio of developed to toatal lines of code (.2 --> all old, I --> all new)

RDTODM= ratio of developed to total modules

Basili/Freburger
University of Maryland, 1979

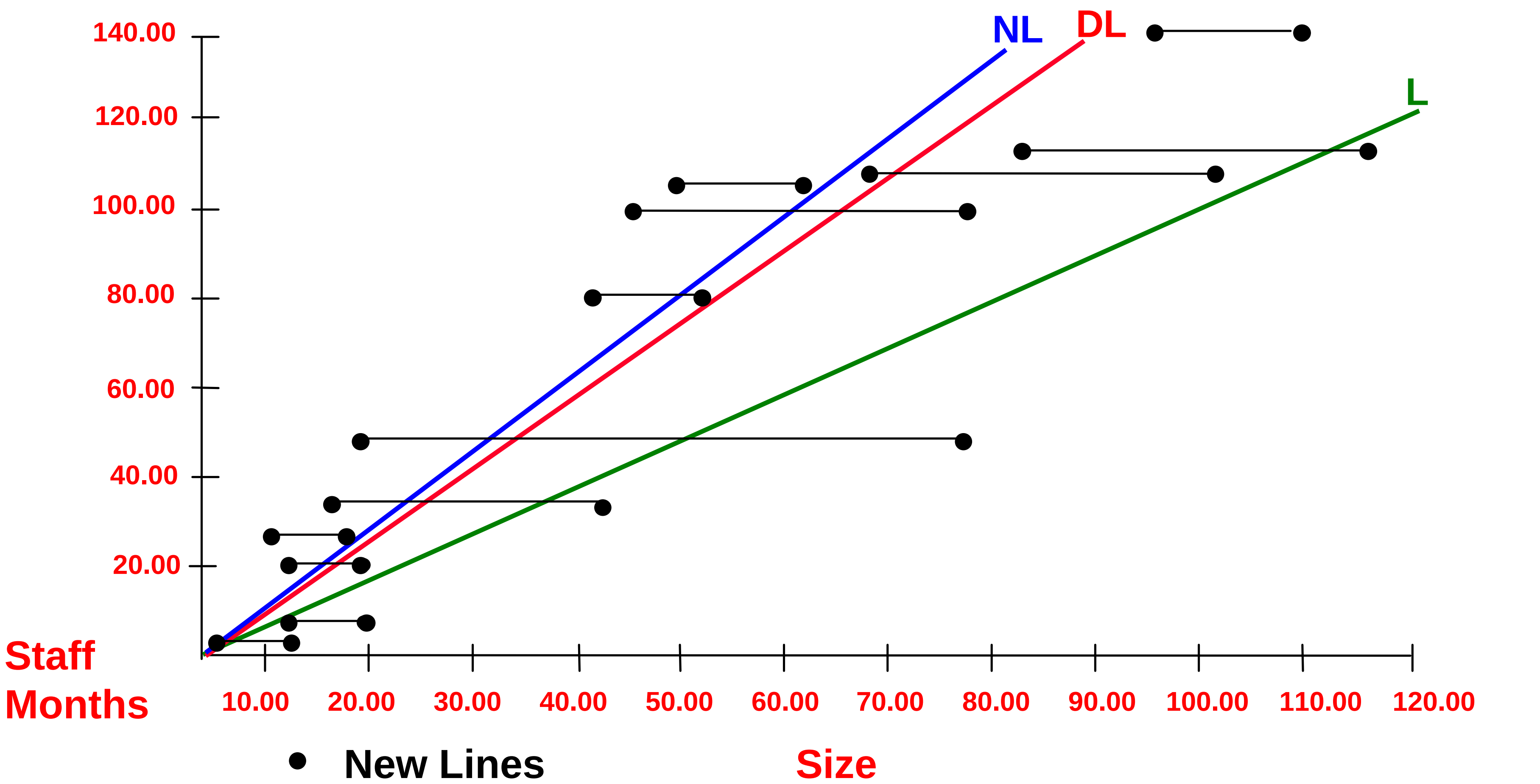
MODELING AND MEASURING RESOURCES
Model Validation Study in the SEL

SEL Data

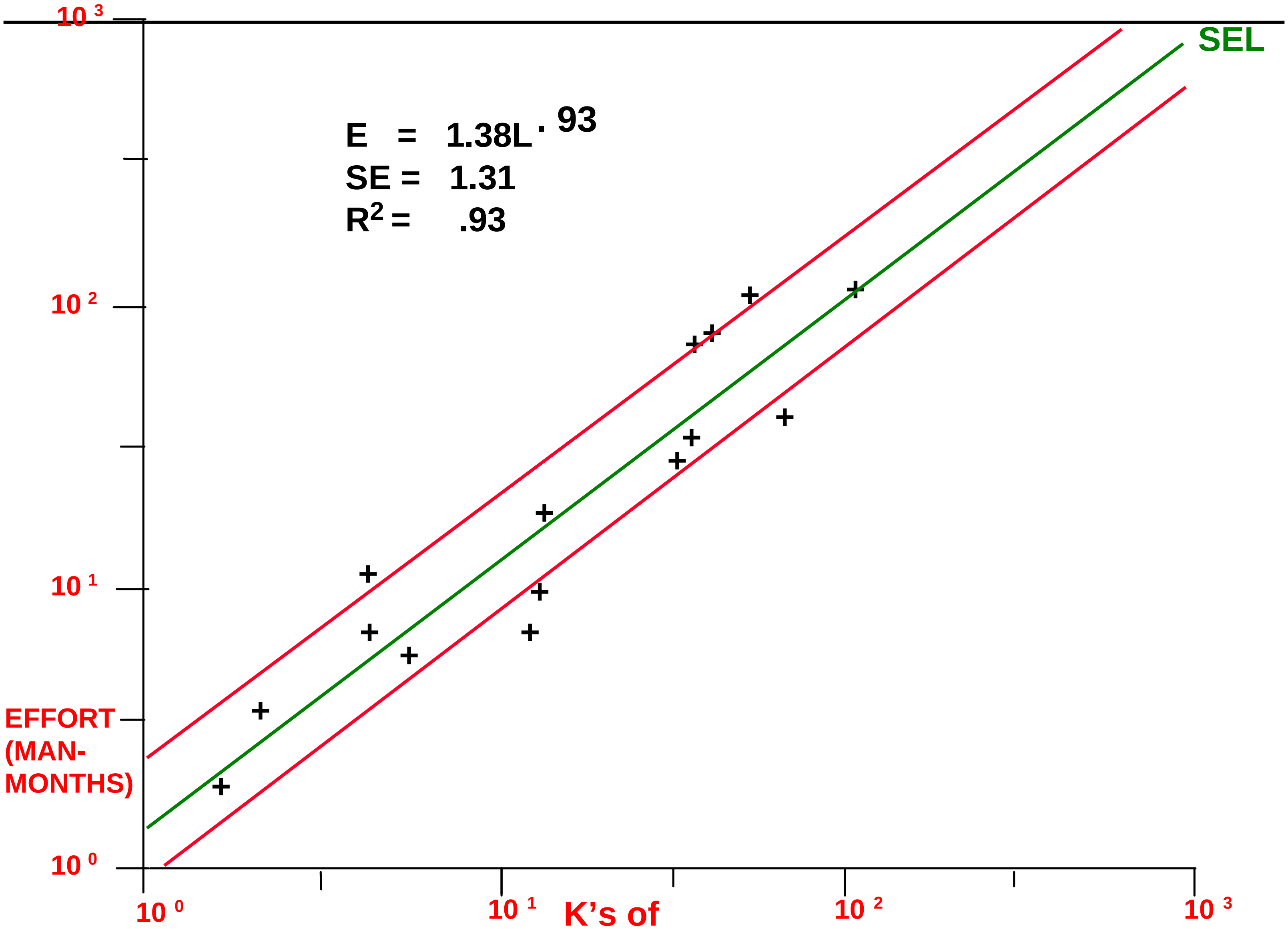
We have examined a set of projects

- Dealing with ground support software
- Ranging from 2K to 101K developed source lines
- Duration ranging from 4.6 to 17.4 months
- Effort ranging from 5 to 138 staff months
- Average staff size from 1 to 8 people
- Productivity from 413 to 1068 developed source lines/staff month with an average of 668 developed source lines/staff month
- Data covers design through acceptance test
- Includes manager, programmer and support staff

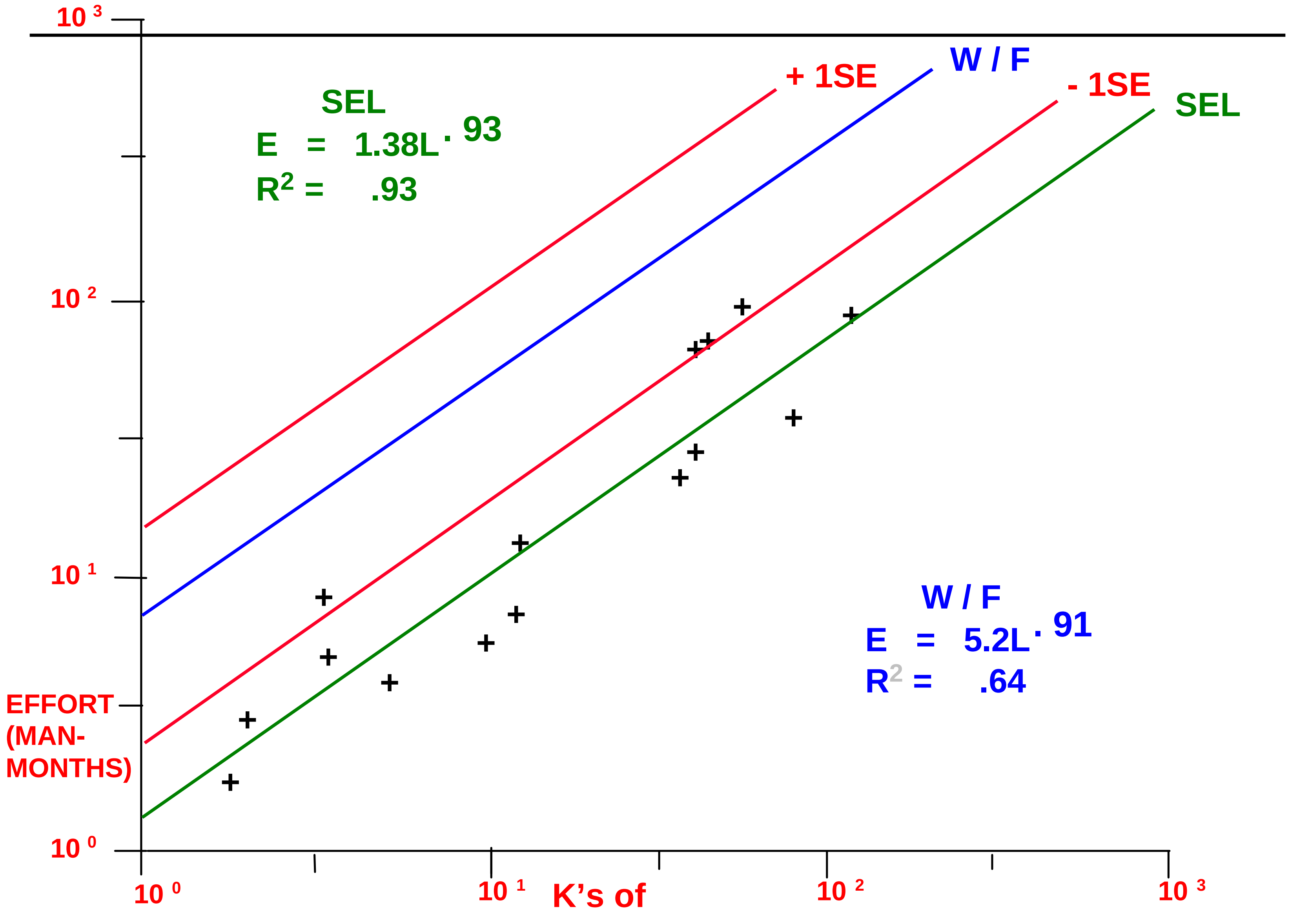
COMPARISON OF NL, DL, L



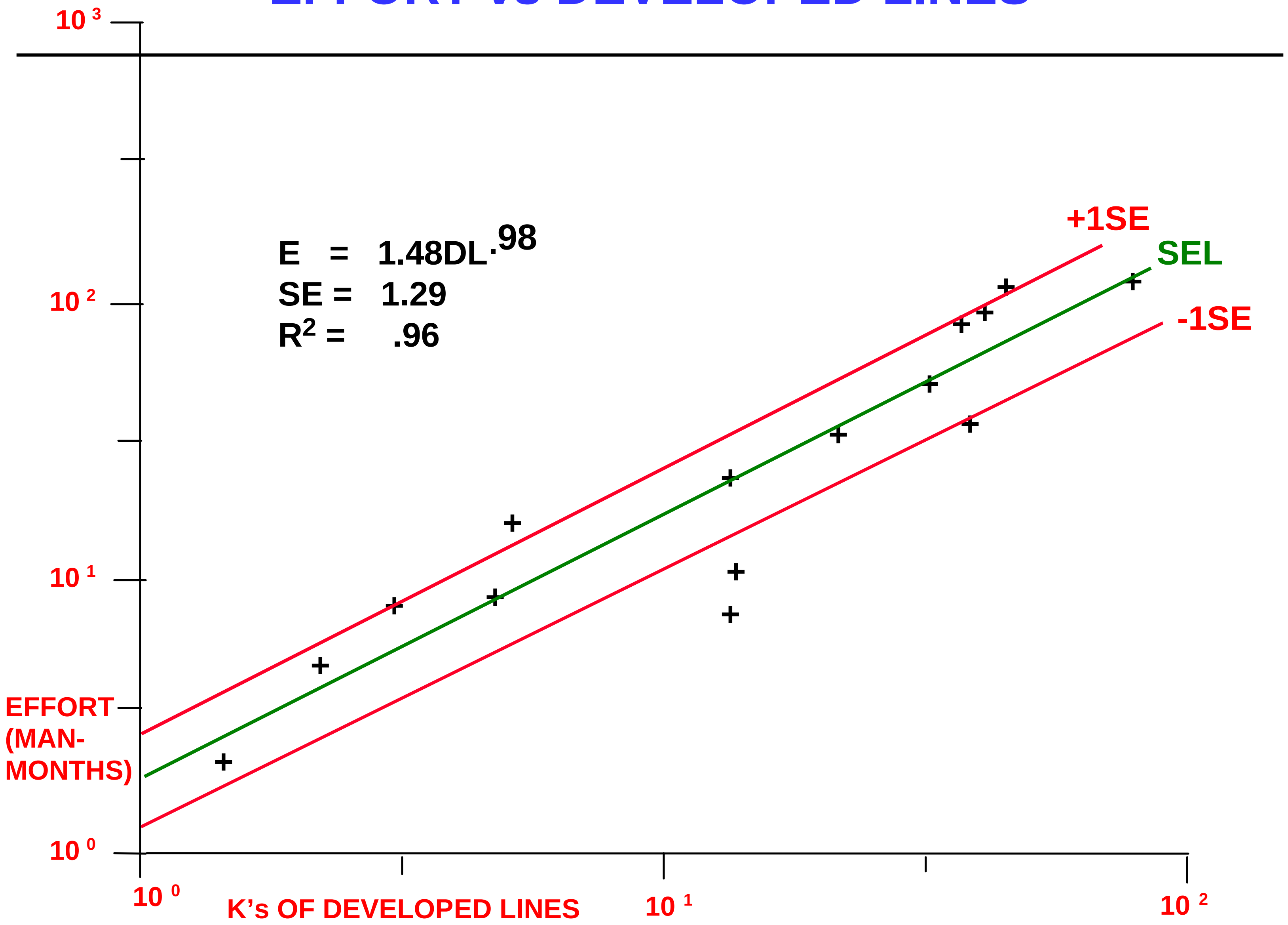
SEL - EFFORT vs LINES OF CODE



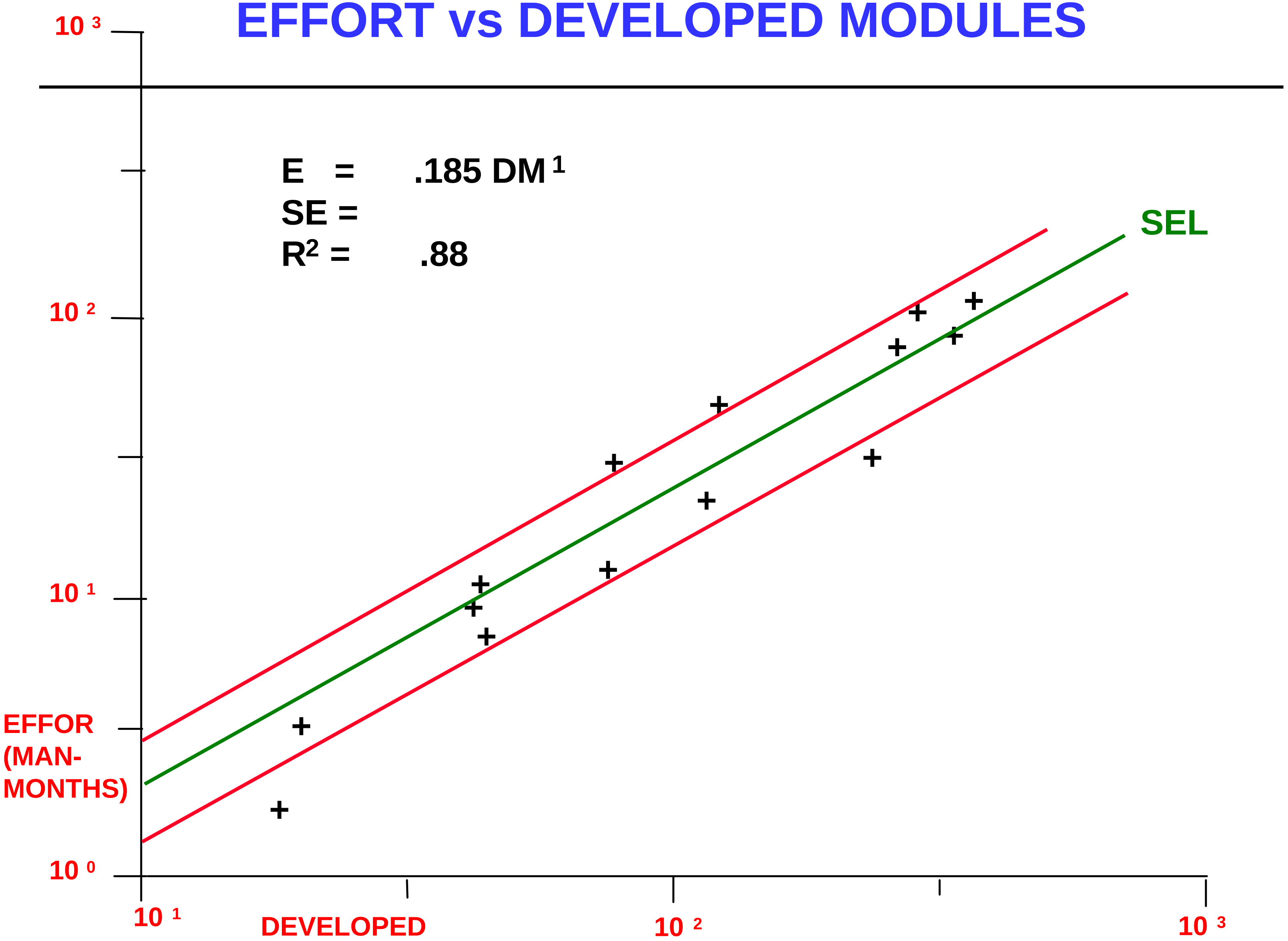
EFFORT vs LINES OF CODE



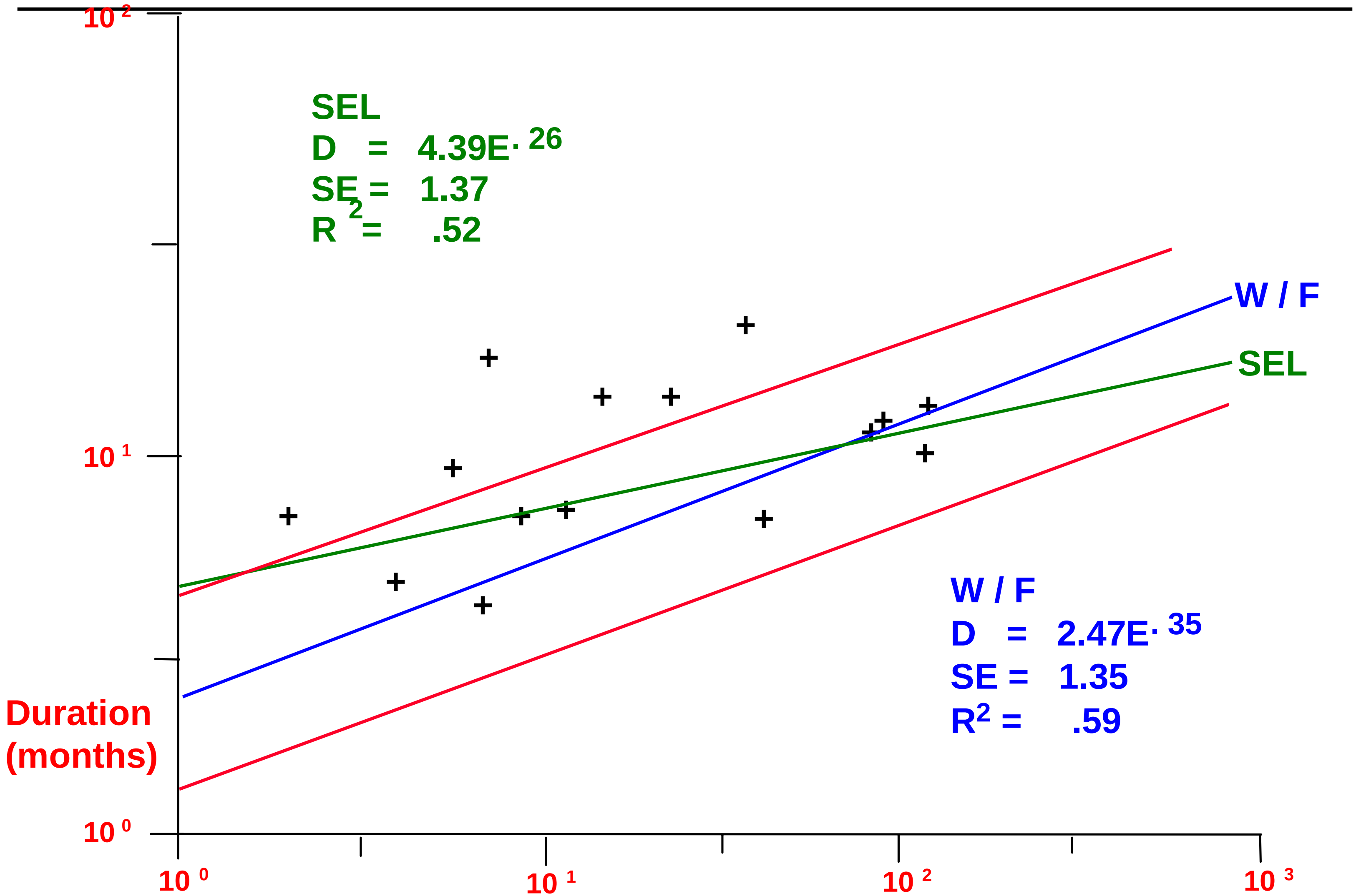
EFFORT vs DEVELOPED LINES



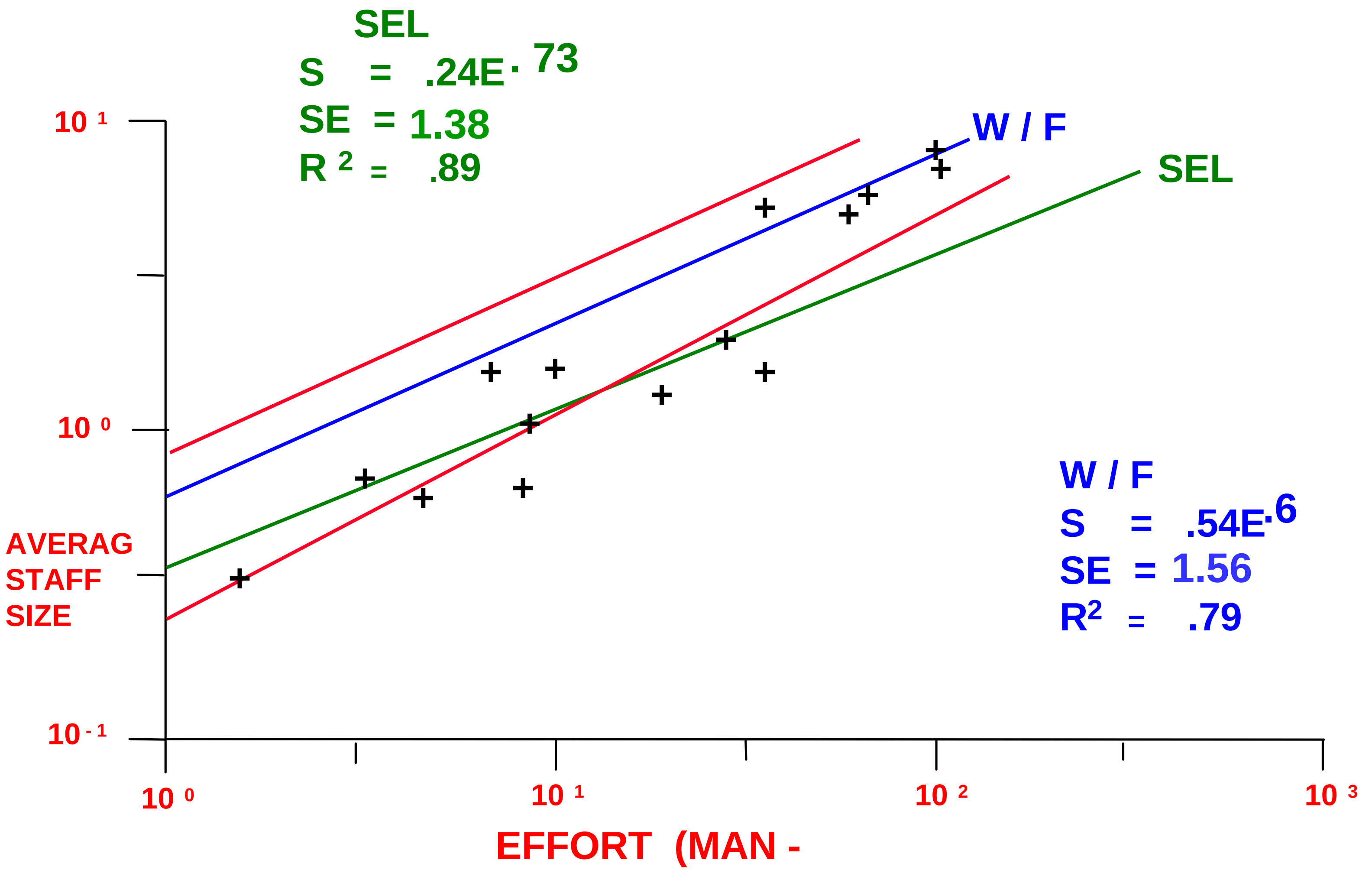
EFFORT vs DEVELOPED MODULES



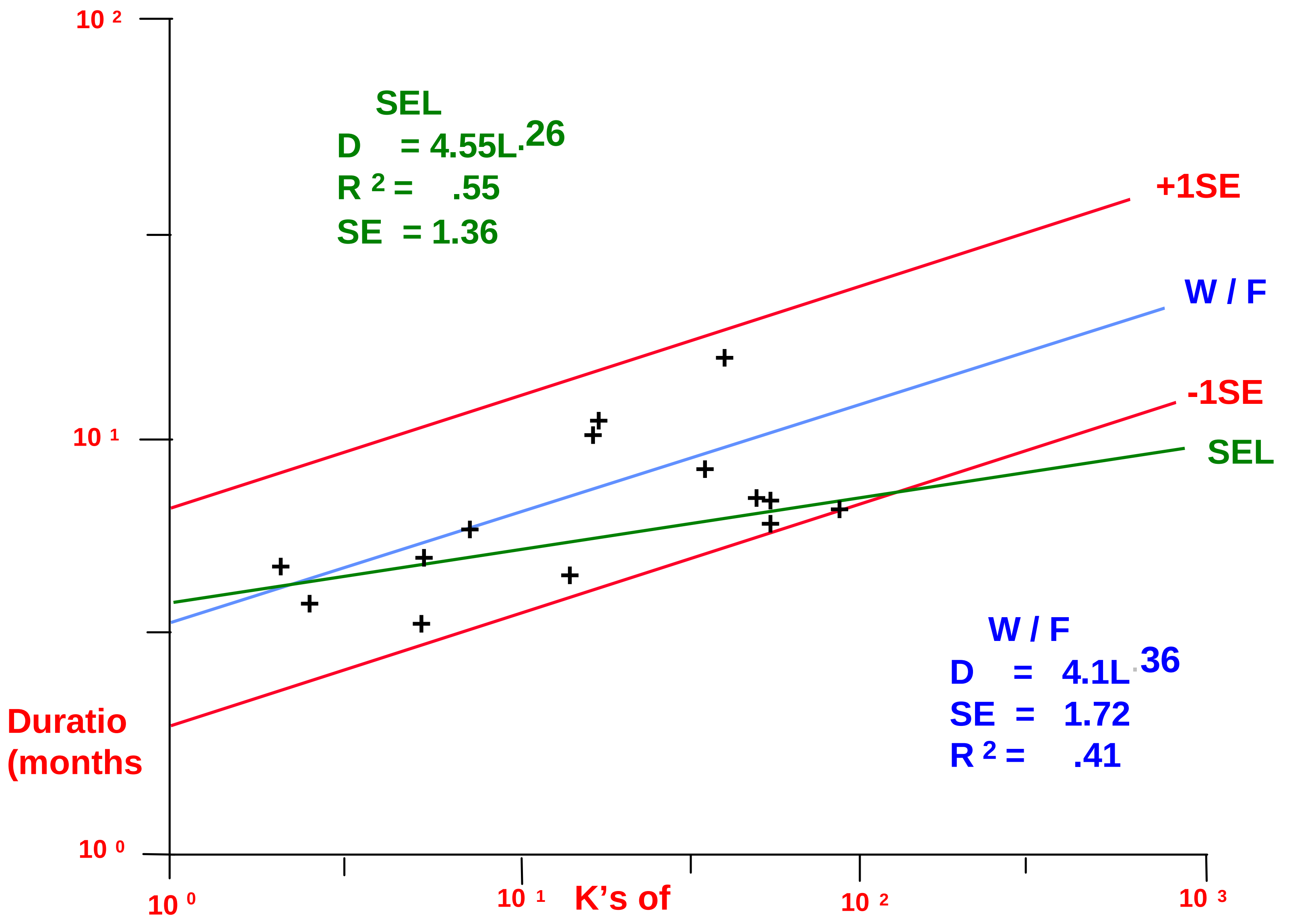
DURATION vs EFFORT



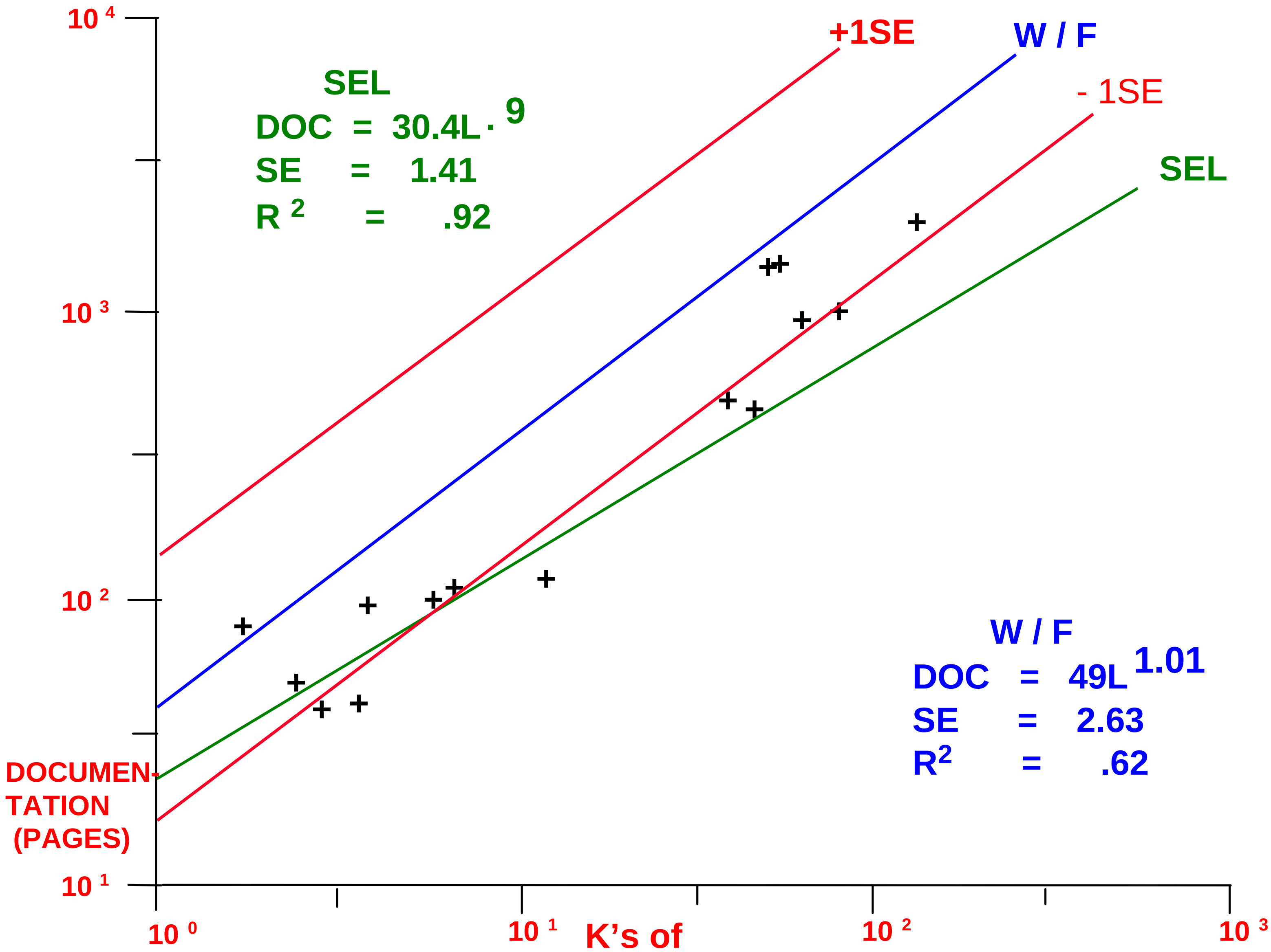
STAFF SIZE vs EFFORT



DURATION vs LINES



DOCUMENTATION vs LINES



Model Validation Study in the SEL

Do the W/F equations hold in other environments?

Results show that:

SEL data fits within the broad context of the Walston and Felix equations

But it would be better for the SEL to build its own base model

Are there other measures of size that might be of interest?

Yes, there are other measures of size that might be more effective in the context of the environment

Do the same parameters affect effort in other domains?

Can there be further tailoring of their model?

MODELING AND MEASURING RESOURCES

Model Creation Study in the SEL

Build a mechanism for creating local models of effort estimation based upon size and other parameters, using variable tailored to the local environment

Goal: Analyze the new resource modeling for the purpose of evaluation with respect to the quality of the fit of the data to the curve from the point of view of the SEL

MODELING AND MEASURING RESOURCES
Building your Own Model

A Resource Meta-Model

Meta-Model Assumptions:

- Each software development environment is different
- There are factors that reflect the organizational environment rather than the project
- There are other factors that reflect the differences in the projects

Approach:

- Compute the background equation
- Analyze the factors available to explain the difference between actual effort and effort as predicted by the background
- Use the model to predict the effort for new project

Bailey/Basili
Univ. of Maryland

MODELING AND MEASURING RESOURCES
Building your Own Model

A Resource Meta-Model

Step 1

Compute the background equation

A) Pick size measure (S)
- lines (comments, executable stmts, new, total,developed)
- modules (new, total, with data blocks)
- function points

B) Choose background relationship for effort (E) and size (S)

$E = aS + b$

$E = aSb$

$E = aSb + c$

C) Calculate best fit based on
- minimizing the absolute error (least squares)
- minimizing the percent error

A Resource Meta-Model

Based upon 18 points

Linear Fit

$E = 1.37\ DL + 3.48$ — see = 29.3%

(minimizing absolute error) — see = 9.54mm

Log Transformation Fit

$E = 1.991\ DL^{.914}$ — see = 27.8%

(minimizing absolute error) — see = 9.11mm

Direct Fit

$E = 1.545\ DL^{.967}$ — see = 24.7%

(minimizing percent error) — see = 9.44mm

Direct Fit (with constant)

$E = .5766\ DL^{1.2} + 3.75$ — see = 20.8%

(minimizing percent error) — see = 11.68mm

MODELING AND MEASURING RESOURCES
Building your Own Model

A Resource Meta-Model

Step 2

Analyze the factors available to explain the difference between actual effort and effort as predicted by the background equation for the available data

A) Choose a set of factors (we used ~ 100)

B) Group the factors based upon relevance and experience

C) Choose a few factor groups based upon intuition and correlations with productivity and the difference between actual and predicted effort

D) Run a forward multiple regression (it will add a factor at a time)

- there is no rule to judge how many factors to use
- a rule of thumb is to limit the number of factors to either
  - 10% of the number of data points or
  - stop adding factors when the new factor accounts for minimal improvement in the explanation of the difference R^2)

E) Iterate A, B, C, D to try to improve the explanation of the difference

MODELING AND MEASURING RESOURCES
Building your Own Model

Factors Grouped

Total Methodology

Tree charts, Top down design, Design formalisms, Code reading
Formal documentation, Chief programmer teams, Formal test plans
Unit development folders, Formal training

Cumulative Complexity

Customer interface complexity, Application process complexity,
Customer-initiated program design changes, Program flow complexity
Internal communication complexity, External communication complexity
Data base complexity

Cumulative Experience

Programmer qualifications, Programmer experience with machine
Programmer experience with language,
Programmer experience with application
Team previously worked together on same type problem

MODELING AND MEASURING RESOURCES
Building your Own Model

Step 3

Use the model to predict the effort for the new project

A. Estimate size

B. Use background equation to predict standard effort

C. Estimate values of the factors used in the regression analysis

D. Compute the difference this project should exhibit based upon the values from C used in the multiple regression equation

E. Apply that difference to the standard effort to compute the improved effort estimate

MODELING AND MEASURING RESOURCES
Application of the Model in the SEL

We Used 17 points to predict the 18th

For our model we chose

$S = DL$ (developed lines of code)

The background relationship

$E = aS^{b} + C$

We chose to minimize the percent error

This yields a background equation (based upon the first 17 data points) of

(1) $E_s = .72DL^{1.17} + 3.4$

with a standard error of estimate - 1.254

For project 18 we estimated 101,000 lines yielding

$E_s = 163$ staff-months with interval (130,204)

We grouped factors to form new factors and chose two:

Meth: Total methodology (11 factors)

Complx: Complexity (5 factors)

MODELING AND MEASURING RESOURCES
Application of the Model in the SEL

We applied the multiple regression routine to compute the effort ratio

Using meth

$ER = -.036meth + 1.0 = -.224$

then the improved estimated $E_I + E_S / 1.224 = 133$ staff-months

with interval (115,154)

Using meth and cmplx

$E_R = -.036meth + .006cmplx + .86 = -.166$

then the improved estimate $E_I = E_S / 1.166 = 140$ staff-months

with interval (121,162)

The actual effort was 138 staff-months

MODELING AND MEASURING RESOURCES
Building Your Own Model

Summary

Background equation tries to express the relationship between size and effort for the average project

It should reflect all those properties that are constant across the environment

Factors reflect local differences among projects within the environment

It is hard to get good factors and good data for the factors

We could only explain half the variation with factors we used

It is a viable approach to estimating software development resource expenditures

Problems

- Need historical data
- Background equation changes with improvement

USING THE SAME DATA FOR OTHER GOALS
The Effect of Various Factors on Productivity

We examined the relationship between productivity and various factors

Found no significant relationship between productivity and size

A large set of methodology factors showed varying degrees of positive correlation with productivity

A combined methodology factor showed a significant positive correlation with productivity

Projects with high methodology rating came from a different population than those with a low methodology rating

No other factors showed a significant positive correlation with productivity

Methodology is correlated with productivity

Bailey/Basili

5th NASA SEL Workshop

USING THE SAME DATA FOR OTHER GOALS
The Effect of Various Factors on Productivity

Projects vary with respect to the set of software development techniques used and the extent to which they were used

There was formal training for some projects

Each project was rated with respect to

- A large set of factors
- Covering environment, methodology, experience, performance, etc.
- Values were given on a six-point scale
- Ratings were subjective
- Relative to the local environment
- Done near end of project without knowledge of the productivity results
- By NASA (McGarry), CSC (Page), and University of Maryland (Basili)

USING THE SAME DATA FOR OTHER GOALS
The Effect of Various Factors on Productivity

Based upon a similar study by Doug Brooks (IBM/FSD)

Tried to see if methodology had a significant effect on productivity

Used a statistical test to see if the projects with high methodology use came from a different environment (with respect to productivity) than the projects with a low methodology use

The data used was based upon a relative ranking rather than an absolute rating

The approach was

Divide the ratings for each technique into 3 categories:
low (-1), medium (0), high (1)   (Done to offset differences in scales)

Add the ratings to get a cumulative methodology rating

Divide projects into groups based upon their rating and analyze using the Mann-Whittney-U test (nonparametric statistics)

USING THE SAME DATA FOR OTHER GOALS
The Effect of Various Factors on Productivity

No significant relationship between productivity and size (no point categorizing by size)

Methodology factors for all that showed a difference among projects,the correlations between methodology and productivity:

PDL	.26
Formal design review*	.62
Design formalism	.38
Design decision notes*	.62
Design walk-through	.28
Code walk-through	.19
Code reading*	.58
Top down design	-.19
Structured code	.02
Librarian use*	.52
Chief programmer team*	.62
Formal test plans**	.51
Heavy management involvement	-.09
Formal training*	.58
Top down code	.29

*sig. < .01 **sig. < .05

USING THE SAME DATA FOR OTHER GOALS
The Effect of Various Factors on Productivity

Group:	Low	Medium	High
Ratings:	(-11,-9,-9,-9)	(2,2,2,1,0,-1,-3-3)	(12,11,8,5,5,3)
Productivity:	535 DL/SM	660 DL/SM	768 DL/SM

Result:

Low different from Medium U High (sig. at 0.5)

High different from Medium U Low (sig. at .03)

Group:	Low	High
Ratings:	(-11,-9,-9,-9,-3,-3,-1)	(0,1,2,2,2,3,5,5,8,11,12)
Productivity:	602 DL/SM	710 DL/SM

Result:

Low different from High (sig. at .05)

USING THE SAME DATA FOR OTHER GOALS
Relation of Other Factors with Productivity

Tried the following:

Customer interface complexity

Customer originated program design changes

Complexity of: application processing, program flow, internal communication, external communication, data base complexity, Jerry's general complexity rating.

Constraints: I/O capability, timing, main store

Programming group experience: machine familiarity, language familiarity, application experience, same type before

Hardware changes during development

Percent real time or interactive*

Percent programmer involved in specifications

All but one showed no significant correlation with productivity

*Showed significant difference at .05 level in wrong direction (i.e., higher % real time --> higher productivity)

USING THE SAME DATA FOR OTHER GOALS
The Effect of Various Factors on QUALITY

We compressed three sets of metrics into three factors:
quality, methodology, and complexity

Methodology and complexity were not significantly correlated

Quality was significantly correlated with
methodology (R = .67) and complexity (R = .64)
at less than .001 significance level

Using methodology alone to predict quality, $R^2 = .65$

There is evidence we can predict quality from methodology and complexity

Methodology is correlated with quality

HOW TO USE THE MODELS

The models should be an aid to software development management and engineering- not be taken as the sole source

An Approach

First do a prediction
Apply one or more models
Examine the range of prediction offered by the model
Compare the results

If they agree

I can be more secure about the estimate

If they don't agree

Examine why not
What model assumptions did we not satisfy
What makes this project different
Am I comfortable with my explanation of the difference

Barry Boehm, Software Engineering Economics, Prentice Hall
Tom DeMarco, Controlling Software Projects, Yourdon Press

References

Boehm, Software Engineering Economics, Prentice Hall, 1981

Boehm, et. al., COCOMO II, Prentice Hall, 2001. l

Tom DeMarco, Controlling Software Projects, Yourdon Press

Walston & Felix: A Method of Programming Measurement and Estimation, IBM Systems Journal, Vol. 16, No. 1, 1977

John Bailey and Victor R. Basili, A Meta-Model for Software Development Resource Expenditures, Proceedings of the Fifth International Conference on Software Engineering, pp 107-116, March 1981..

Victor R. Basili and Karl Freburger, Programming Measurement and Estimation in the Software Engineering Laboratory, Journal of Systems and Software, vol. 2, #l, pp 47-57, 1981.

Basili, Models and Metrics for Software Management and Engineering, IEEE Computer Society Press, 1980.