CSSE 372 Software Project Management:
SW Estimation Case Study and COCOMO-II

Shawn Bohner
Office: Moench Room F212
Phone: (812) 877-8685
Email: bohner@rose-hulman.edu

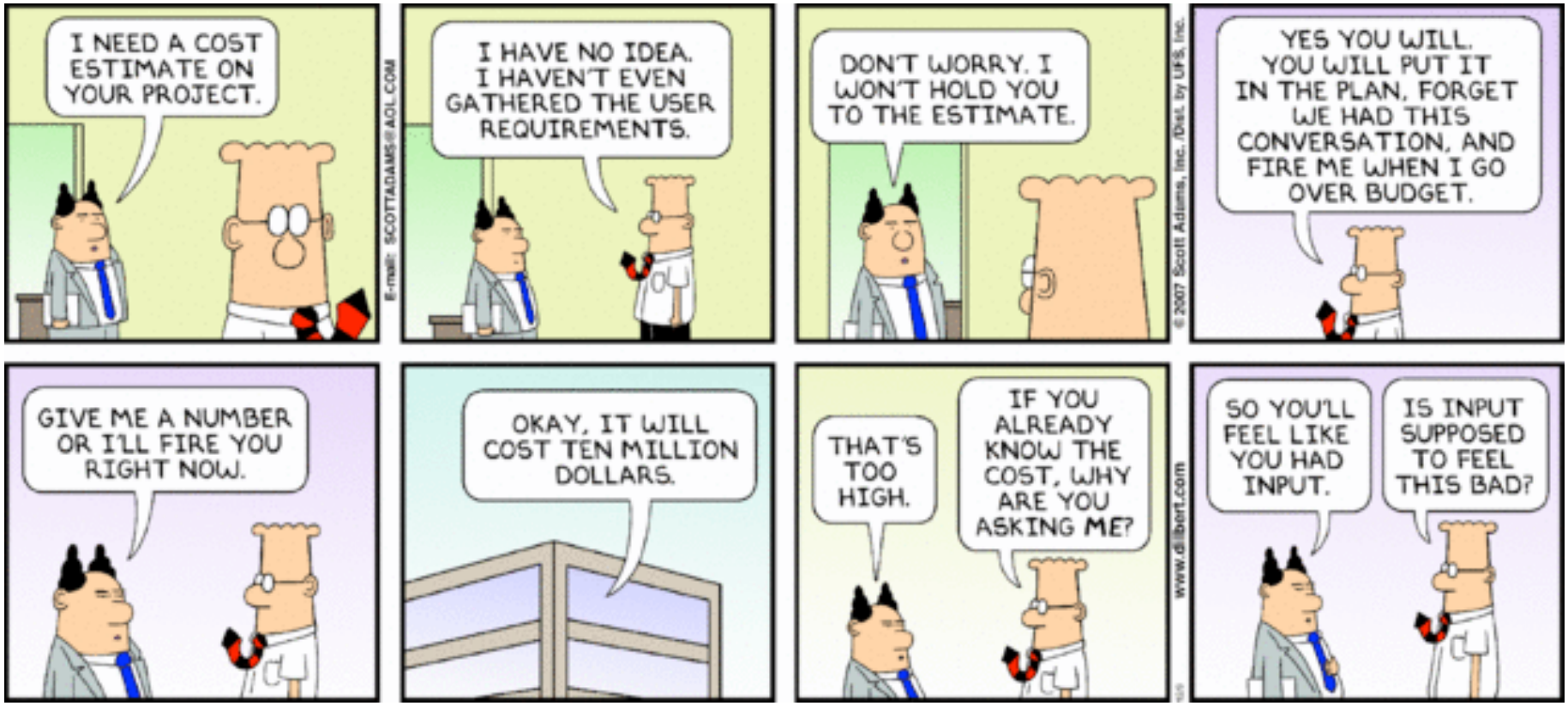
ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Learning Outcomes: Estimation

Estimate software project effort, cost, and schedule for an intermediate size project.

- Look at Homework 3 assignment
- Discuss case study…
- Continue walk through COCOMO-II getting started

Before we get started, What's Dilbert's take on Software Estimation



Paper: "Software Estimation: An Overview" by Richard Stutzke

- What is main thrust or message of the paper?
- What are some of the reasons that estimation can be challenging?
- Why is accuracy of the estimates converging as a project progresses? If they were not converging, what would it mean?
- How has estimation changed over the years?

Software Development Life-Cycle

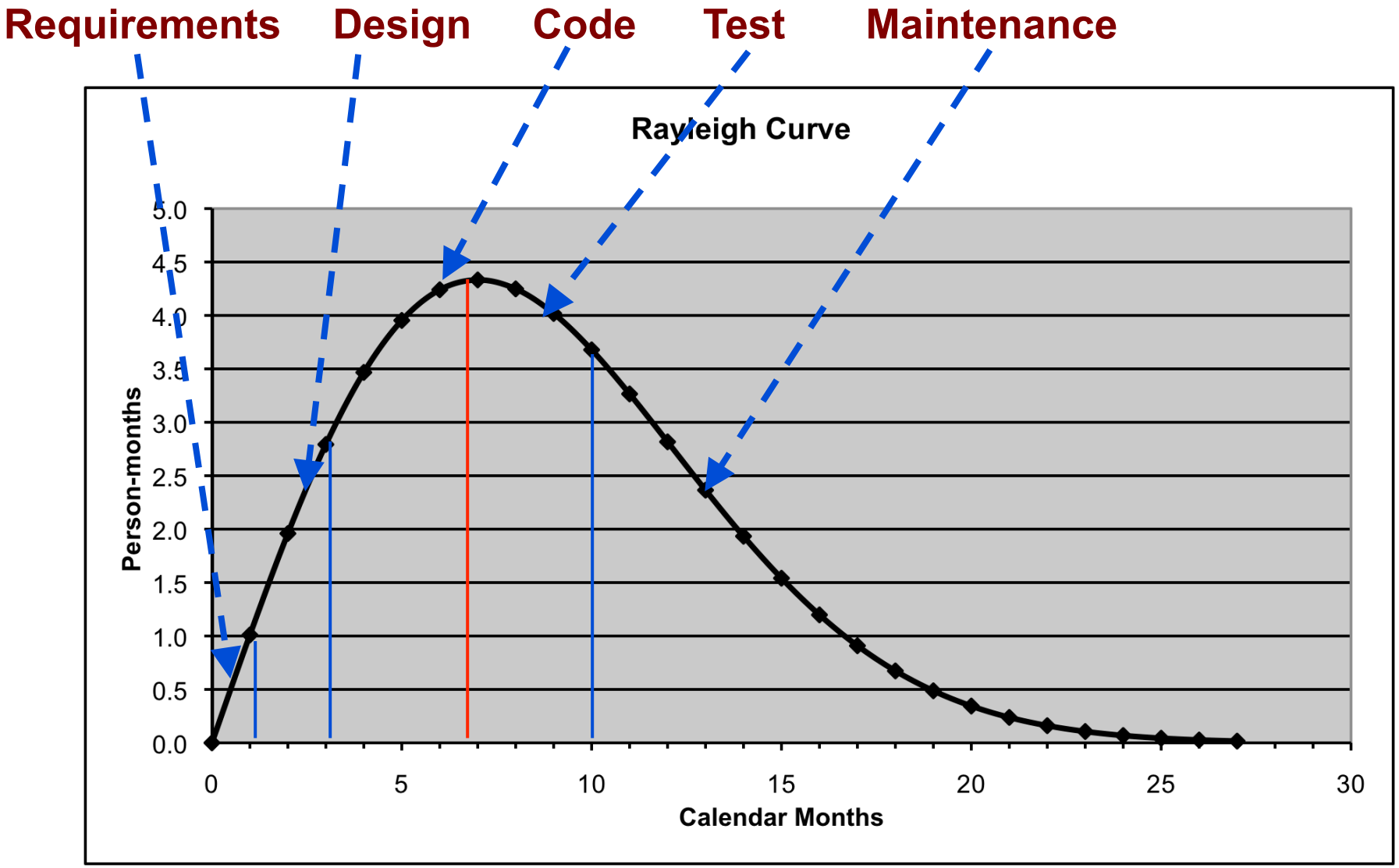
Early researchers found that labor distribution of hardware and software development follows a Rayleigh distribution

So, what does Dilbert think of the numbers?

$$PY = pm\left[\frac{t}{t_D^2}\right]e^{-\left(\frac{t^2}{t_D^2}\right)}$$

Where t = month, t_D = month at peak effort

Rayleigh Curve Relates to SW Life Cycle

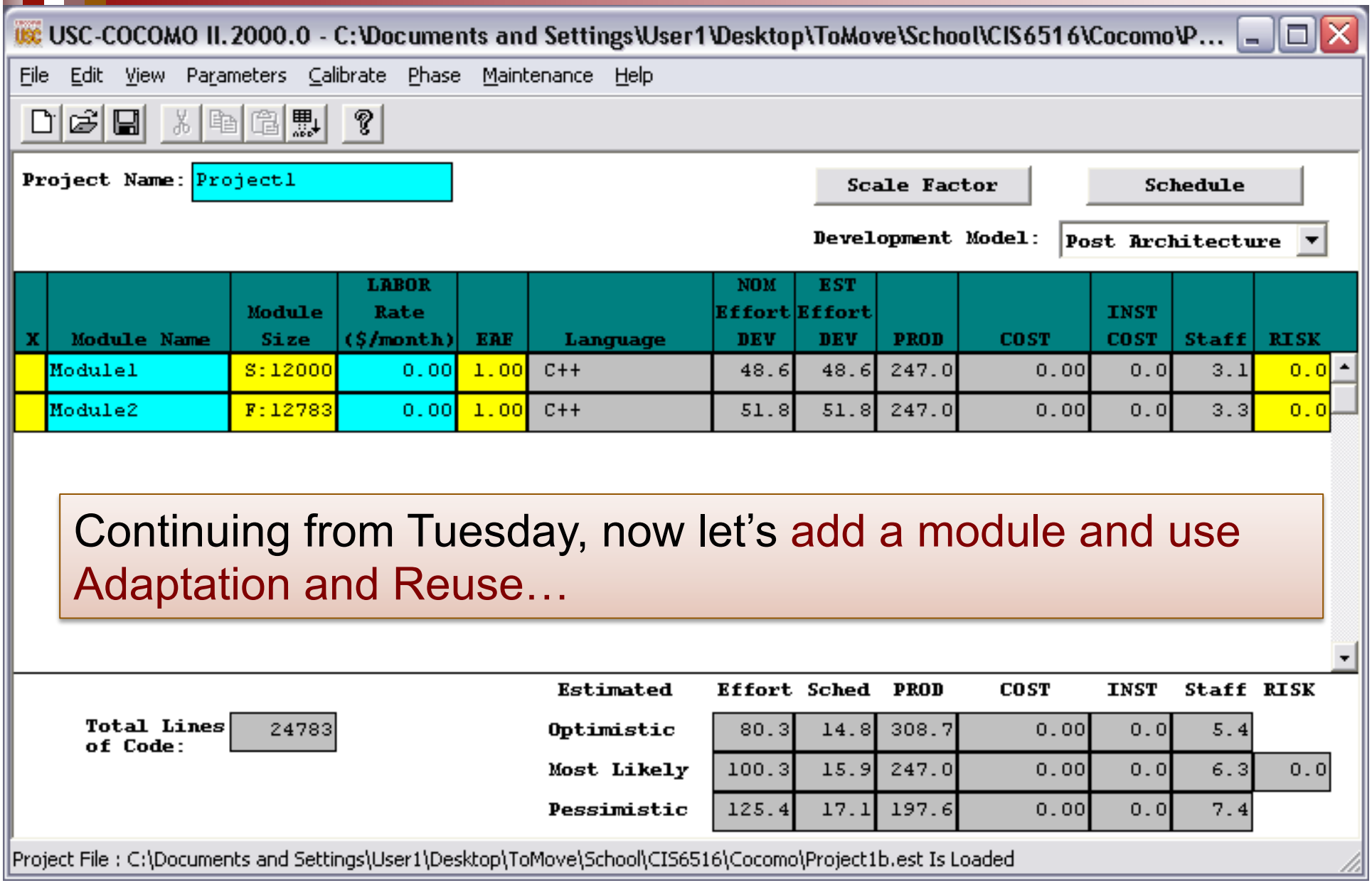


Peak effort at 7 months - Requiring 5 People
50 total person-months of effort required

Default Multiplier values per Language

	Language	Value
✔	Machine Code	640
✔	Assembly, Basic	320
	First Generation	320
✔	Assembly, Macro	213
✔	C	128
✔	Fortran77	107
	Second Generation	107
	Procedural	105
✔	Cobol 85, ANSI	91
	High Level	91
✔	Pascal	91
✔	Modula 2	80
	Report Generator	80
	Third Generation	80
✔	Ada 83	71
✔	Fortran 95	71
✔	Basic, ANSI	64
✔	Lisp	64
✔	Prolog	64
✔	C++	53
✔	Java	53
✔	Ada 95	49
	AI Shell	49
✔	Basic, Compiled	49
✔	Forth	49

	Language	Value
	Simulation Default	46
	Database Default	40
	Access	38
	Visual C++	34
	APL	32
✔	Basic, Interpreted	32
	Object Oriented	29
	Visual Basic 5.0	29
	Perl	21
✔	UNIX Shell	21
	Fourth Generation	20
	PowerBuilder	16
✔	HTML 3.0	15
	Query Default	13
	Spreadsheet Default	6
	Fifth Generation	5

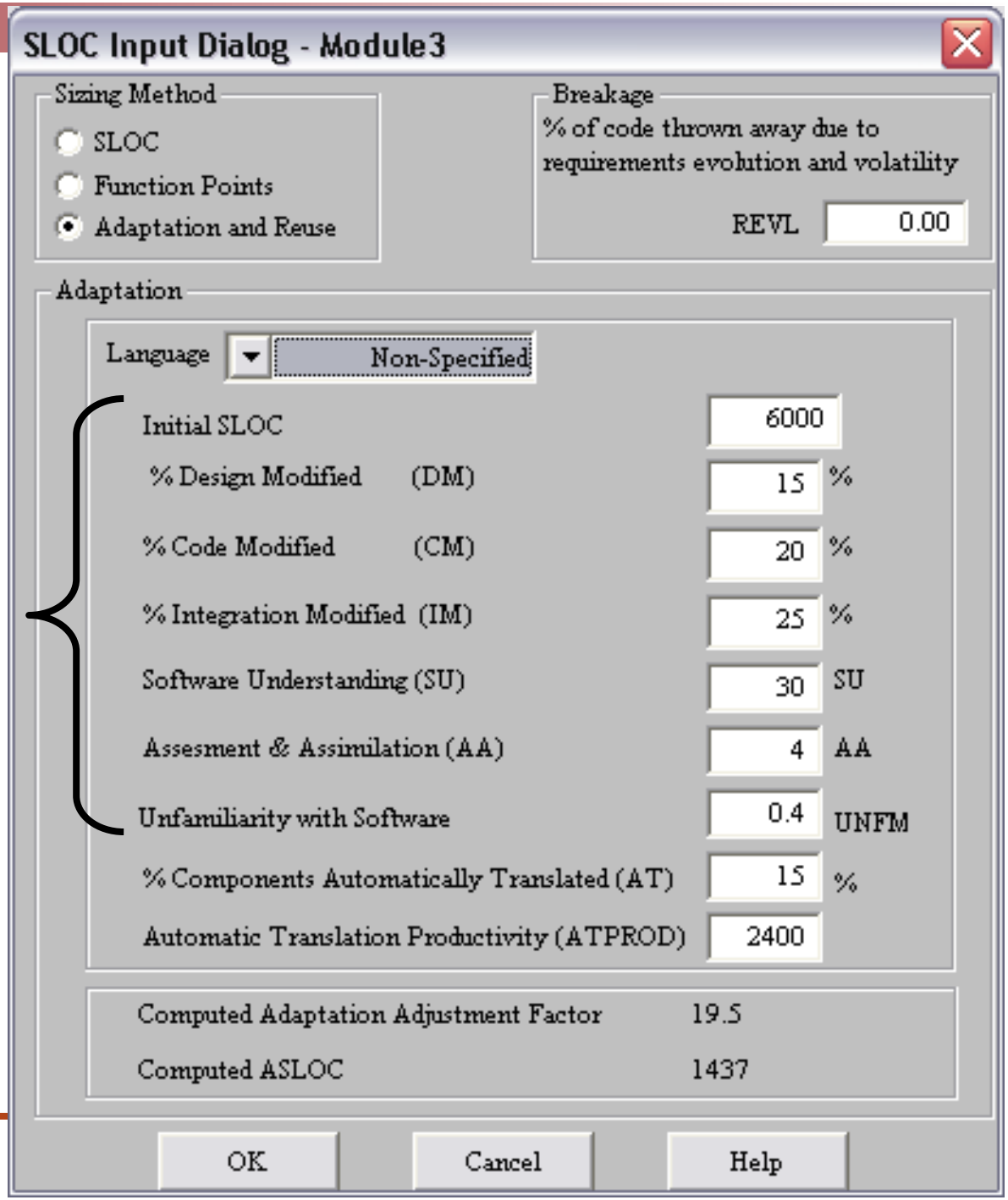

USC-COCOMO II.2000.0 - C:\Documents and Settings\User1\Desktop\ToMove\School\CIS6516\Cocomo\P...
File Edit View Parameters Calibrate Phase Maintenance Help
Project Name: Project1
Scale Factor
Schedule
Development Model: Post Architecture
X | Module Name | Module Size | LABOR Rate ($/month) | EAF | Language | NOM Effort DEV | EST Effort DEV | PROD | COST | INST COST | Staff | RISK
Module1 | S:12000 | 0.00 | 1.00 | C++ | 48.6 | 48.6 | 247.0 | 0.00 | 0.0 | 3.1 | 0.0
Module2 | F:12783 | 0.00 | 1.00 | C++ | 51.8 | 51.8 | 247.0 | 0.00 | 0.0 | 3.3 | 0.0
Continuing from Tuesday, now let's add a module and use Adaptation and Reuse…
Total Lines of Code: 24783
Estimated | Effort | Sched | PROD | COST | INST | Staff | RISK
Optimistic | 80.3 | 14.8 | 308.7 | 0.00 | 0.0 | 5.4
Most Likely | 100.3 | 15.9 | 247.0 | 0.00 | 0.0 | 6.3 | 0.0
Pessimistic | 125.4 | 17.1 | 197.6 | 0.00 | 0.0 | 7.4
Project File : C:\Documents and Settings\User1\Desktop\ToMove\School\CIS6516\Cocomo\Project1b.est Is Loaded

All of these items are associated with a non-linear re-use model.

Why non-linear?

NASA study of 3000 re-used modules found:

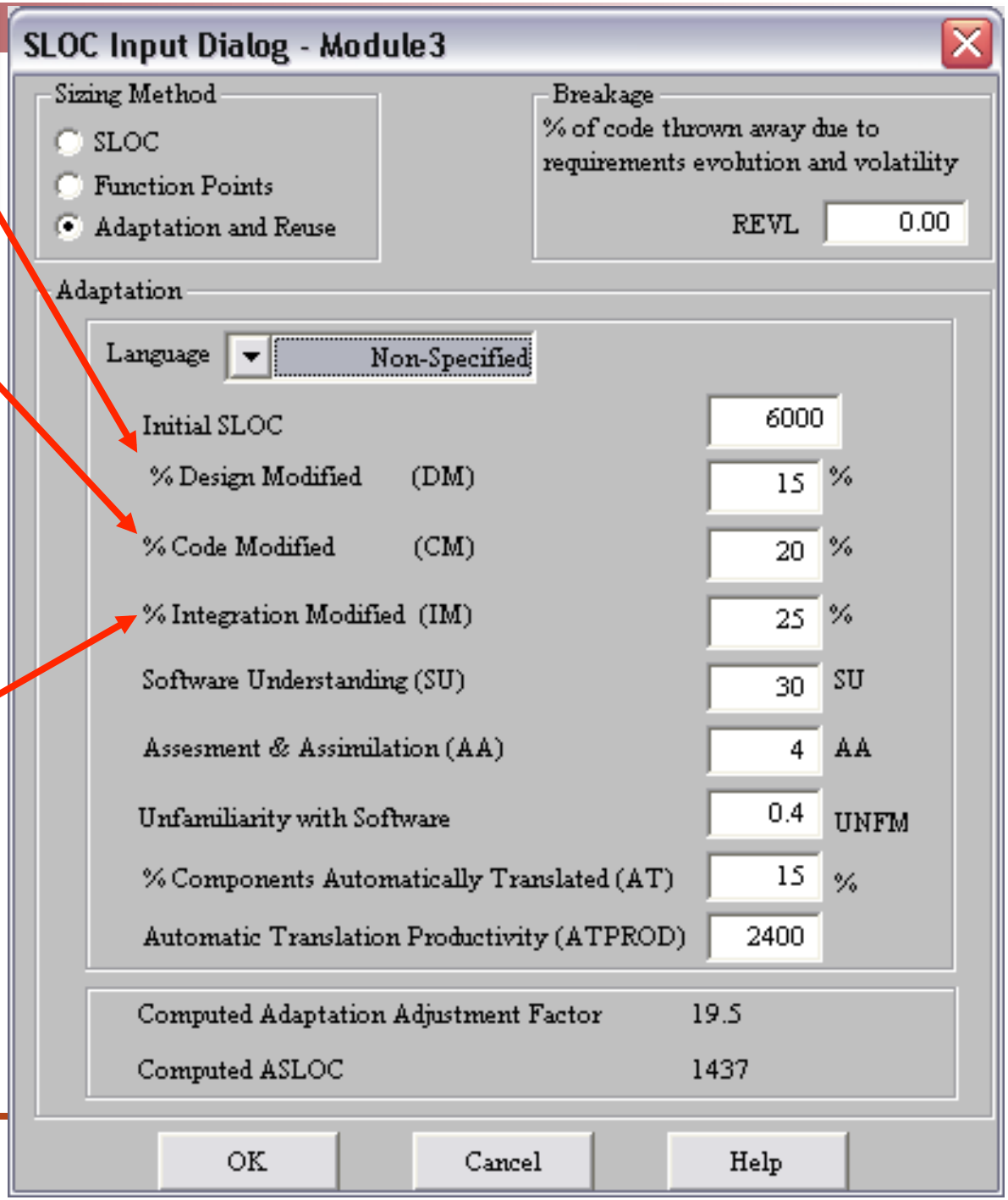
1. There is a cost of about 5% just to assess, select, and assimilate a project.
2. Small modifications generate disproportionately large costs.



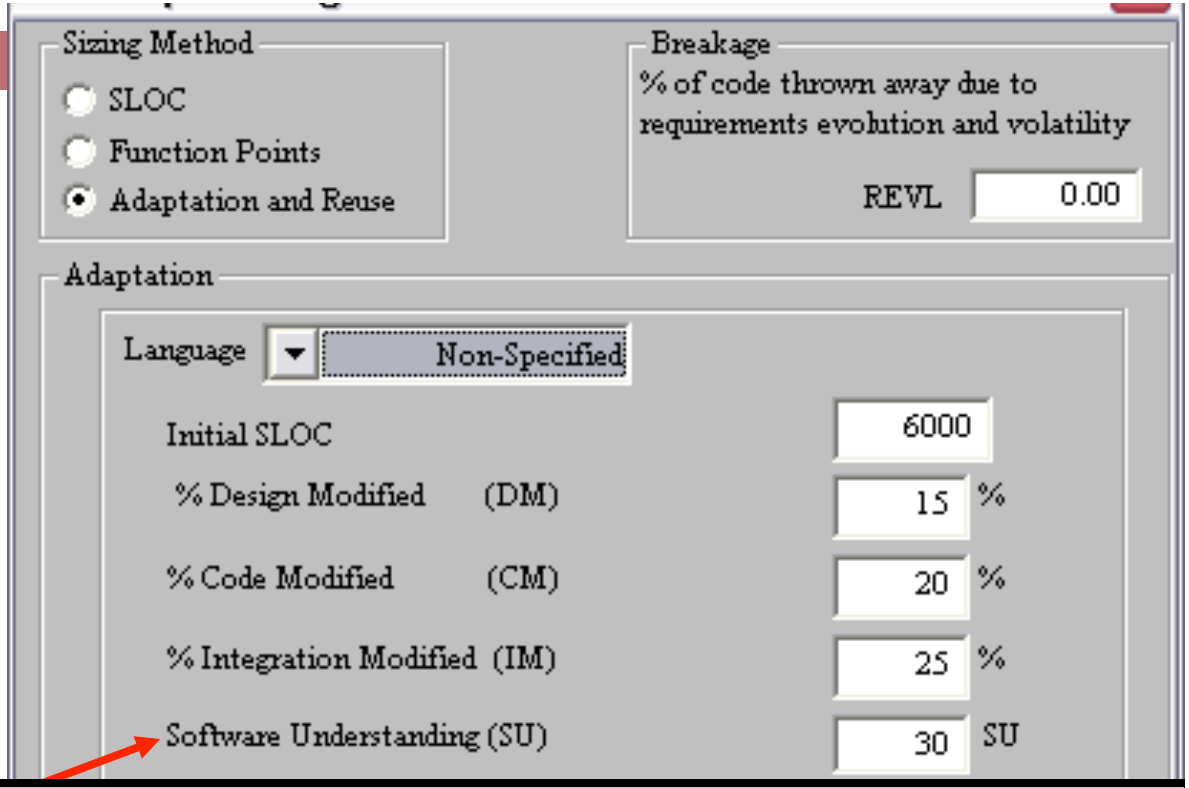
What % of the adapted software's design will change?

… % of code that will change?

% of effort required to integrate the adapted software into an overall product and to test the resulting product as compared to the normal amount of integration and test effort for software of comparable size.

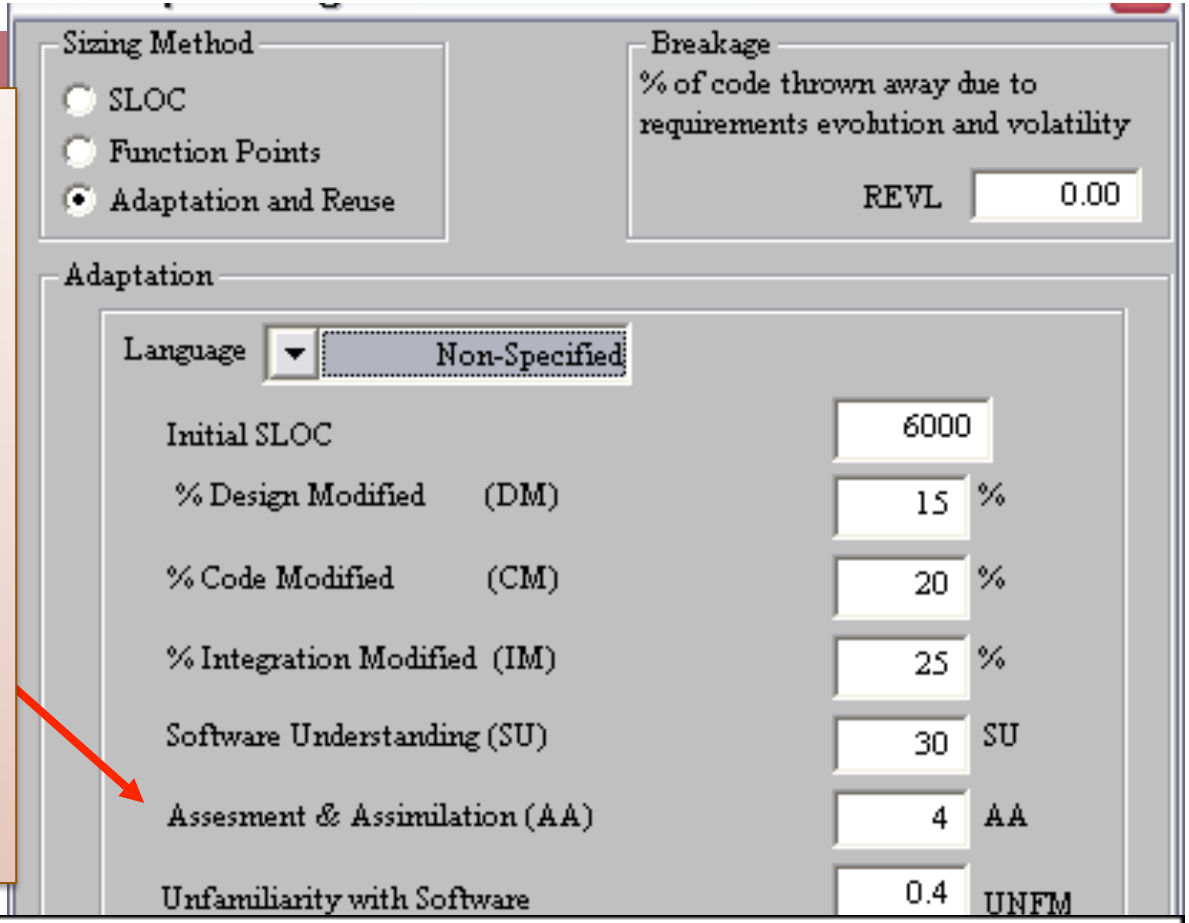


Software Understanding (SU): Use the table below to help you come up with a weighted average based on three key areas…

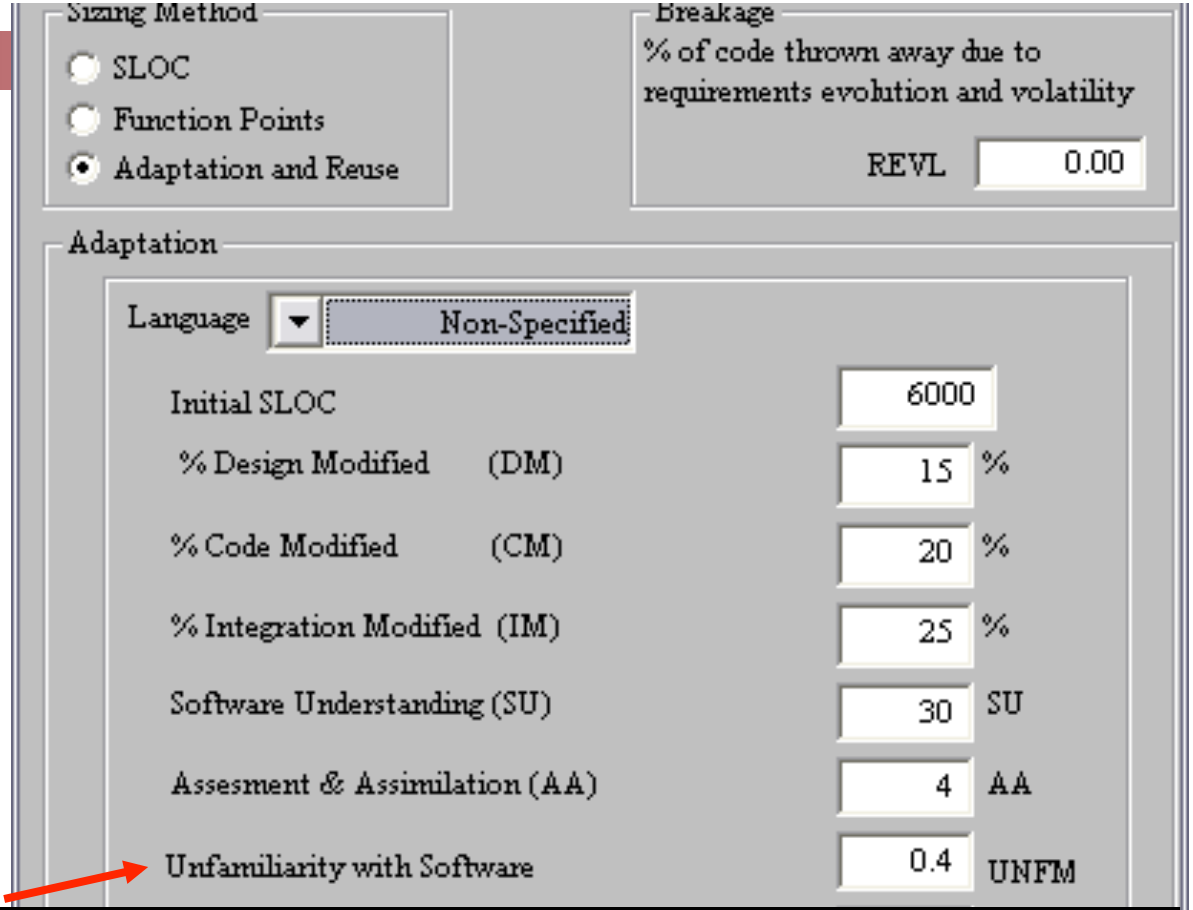


	Choose one column from each row, then average the results.				
	Very Low: 50%	**Low: 40%**	**Nominal: 30%**	**High: 20%**	**VeryHigh: 10%**
Structure	Very low cohesion, high coupling, spaghetti code.	Moderately low cohesion, high coupling.	Reasonably well structured; some weak areas.	High cohesion, low coupling.	Strong modularity, information hiding in data / control structures.
Application Clarity	No match between program and application world views.	Some correlation between program and application.	Moderate correlation between program and application.	Good correlation between program and application.	Clear match between program and application world-views.
Self-Descriptiveness	Obscure code; documentation missing, obscure or obsolete.	Some code commentary and headers; some useful documentation.	Moderate level of code commentary, headers, documentations.	Good code commentary and headers; useful documentation; some weak areas.	Self-descriptive code; documentation up-to-date, well-organized, with design rationale.
Weighted Average:	**(%Structure + %Application Clarity + %Self-Descriptiveness) / 3**				

(AA) Assessment & Assimilation: 0 to 8. Effort to determine whether a fully-reused software module is appropriate to the application, and to integrate its description into the overall product description



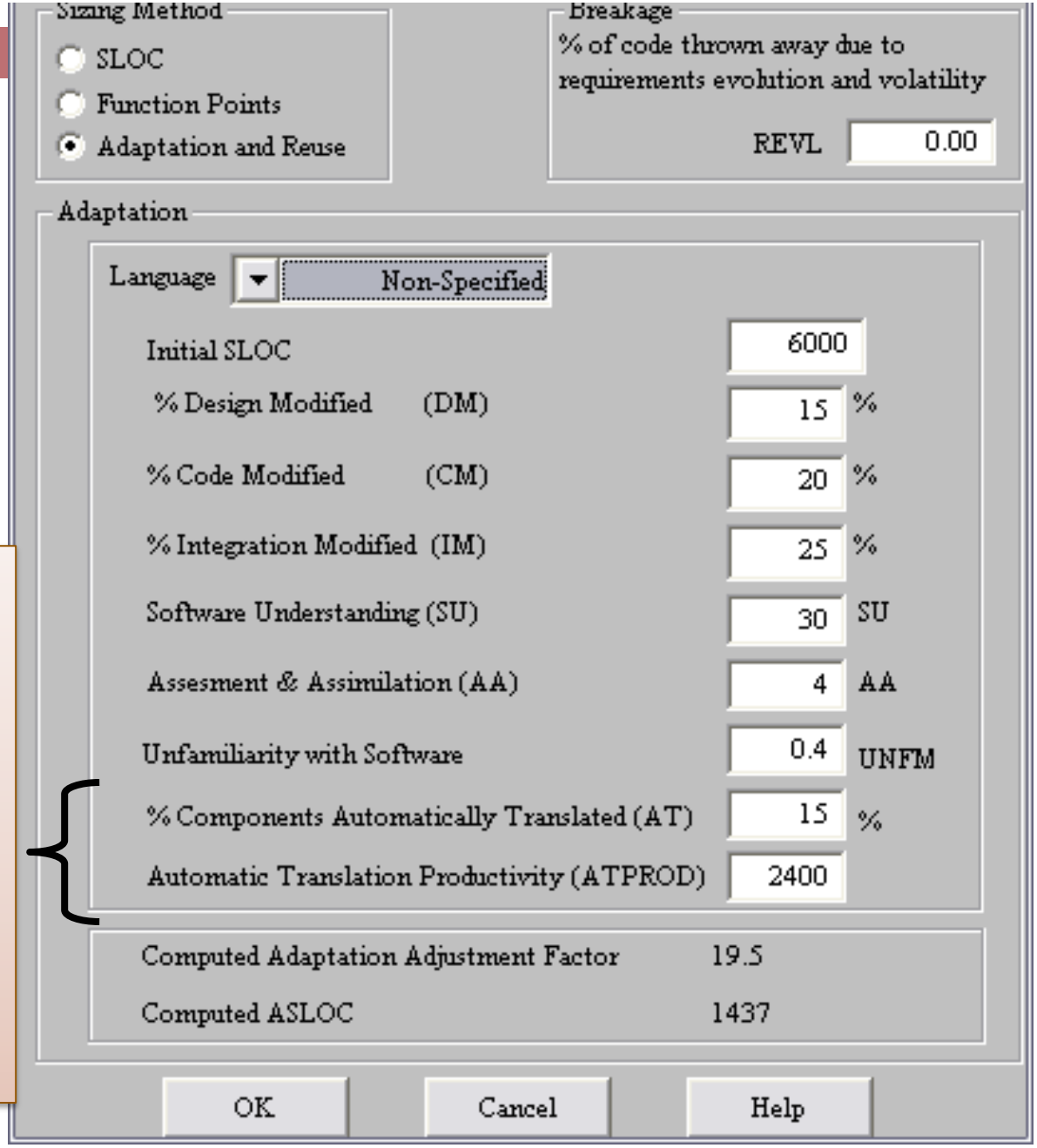
AA Increment	Level of AA Effort
0	None
2	Basic module search and documentation
4	Some module Test and Evaluation (T&E), documentation
6	Considerable module T&E, documentation
8	Extensive module T&E, documentation



UNFM	
Increment	Level of Unfamiliarity
0	Completely familiar
0.2	Mostly familiar
0.4	Somewhat familiar
0.6	Considerably familiar
0.8	Mostly unfamiliar
1	Completely unfamiliar

These last two areas have to deal with automatically translating code.

The ATPROD figure is in source statements / person month.



COCOMO-II: Effort Equation for Post Architecture Model

$$PM = \prod_{i=1}^{17} (EM_i) \cdot A \cdot \left[\left(1 + \frac{REVL}{100}\right) \cdot Size\right]^{\left(0.91 + 0.01 \sum_{j=1}^{5} SF_j\right)} + \left(\frac{ASLOC \cdot \left(\frac{AT}{100}\right)}{ATPROD}\right)$$

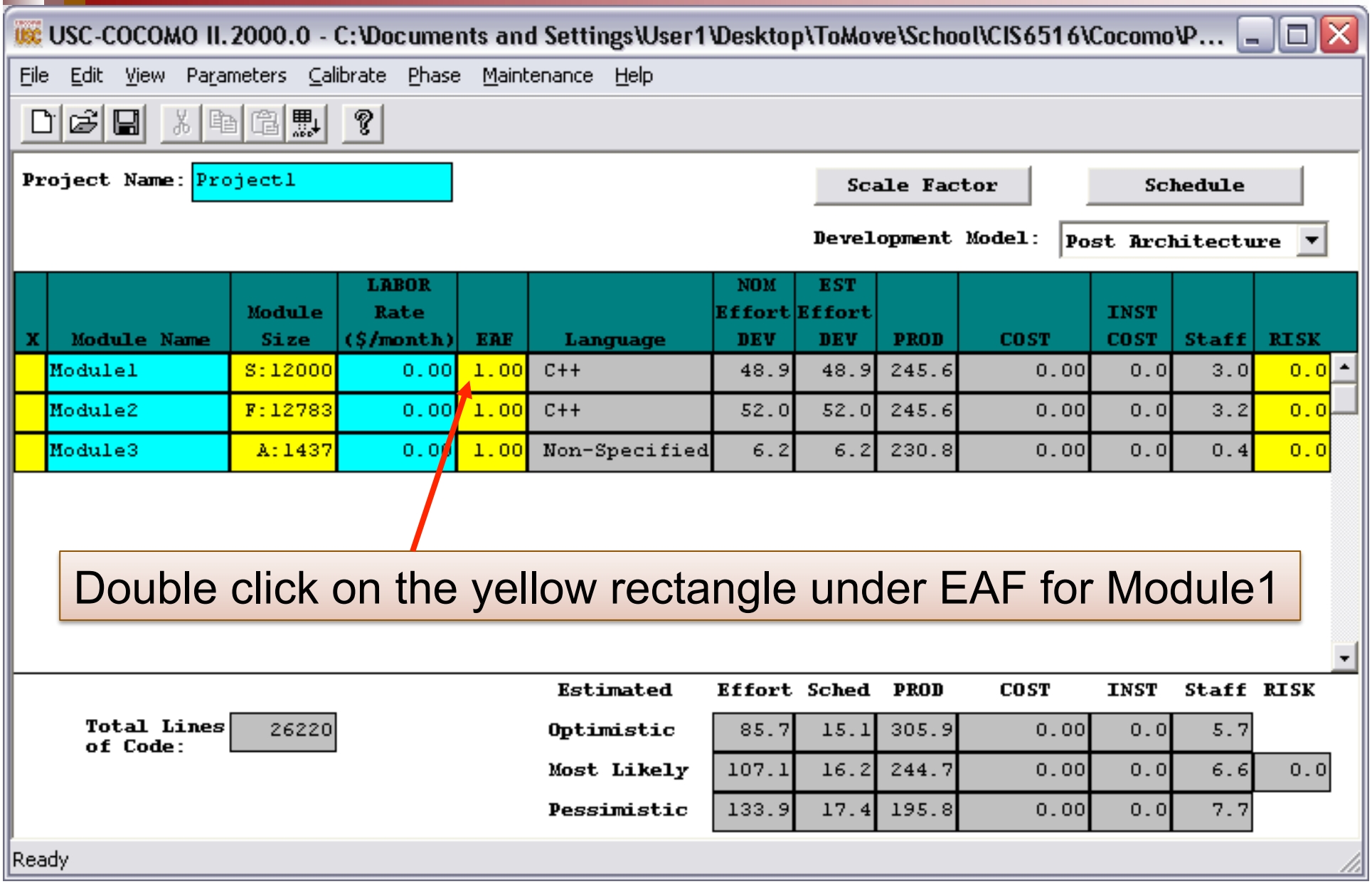
where

$$Size = KNSLOC + \left[KASLOC \cdot \left(\frac{100 - AT}{100}\right) \cdot \frac{(AA + SU + 0.4 \cdot DM + 0.3 \cdot CM + 0.3 \cdot IM)}{100}\right]$$

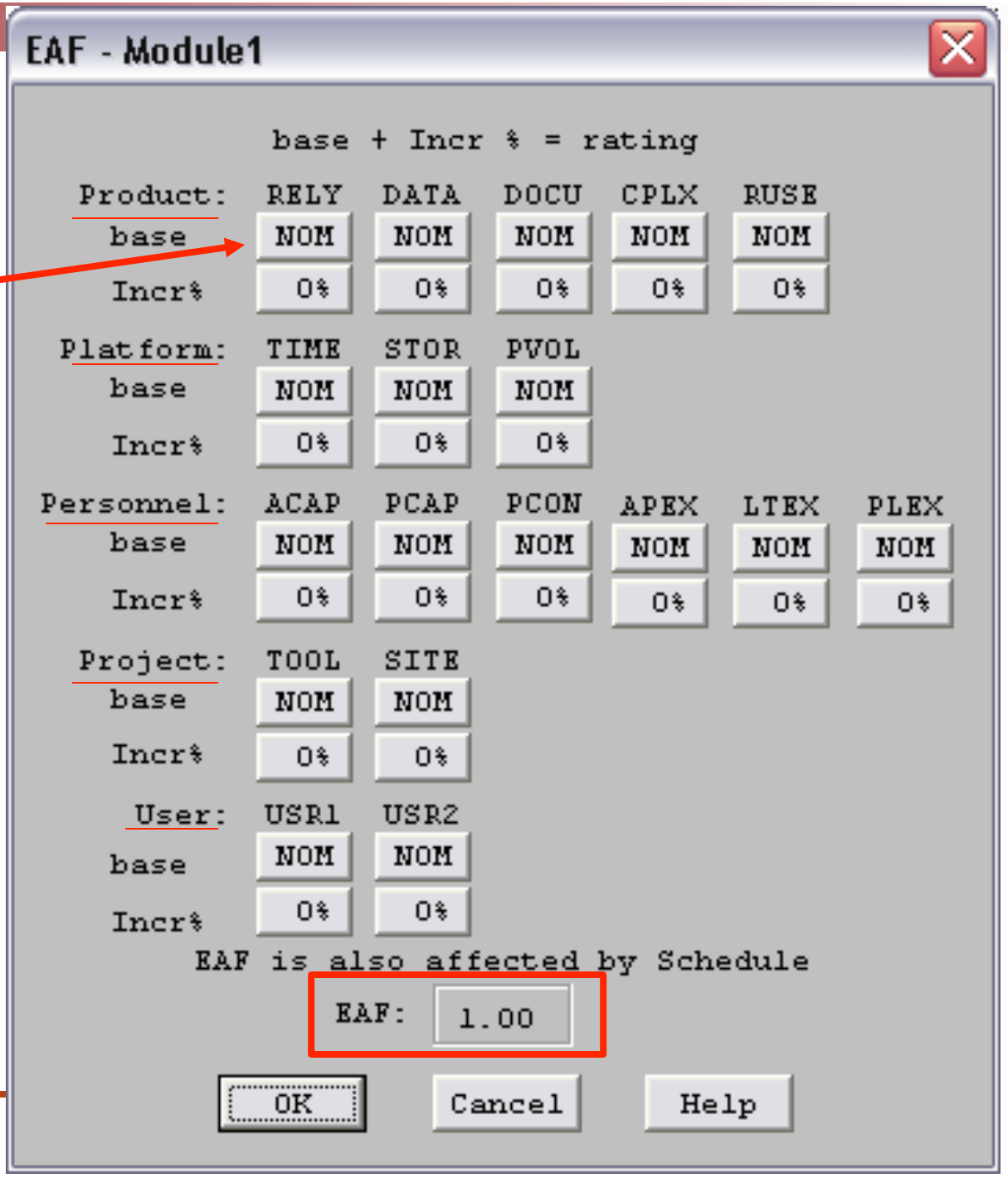
$$B = 0.91 + 0.01 \sum_{j=1}^{5} SF$$

So via Adaptation and Reuse we have now addressed the areas of the calculation in this brown color…

So now let's talk about Effort Adjustment Factors (EAF)


USC-COCOMO II.2000.0 - C:\Documents and Settings\User1\Desktop\ToMove\School\CIS6516\Cocomo\P...
File Edit View Parameters Calibrate Phase Maintenance Help
Project Name: Project1
Scale Factor
Schedule
Development Model: Post Architecture
X | Module Name | Module Size | LABOR Rate ($/month) | EAF | Language | NOM Effort DEV | EST Effort DEV | PROD | COST | INST COST | Staff | RISK
Module1 | S:12000 | 0.00 | 1.00 | C++ | 48.9 | 48.9 | 245.6 | 0.00 | 0.0 | 3.0 | 0.0
Module2 | F:12783 | 0.00 | 1.00 | C++ | 52.0 | 52.0 | 245.6 | 0.00 | 0.0 | 3.2 | 0.0
Module3 | A:1437 | 0.00 | 1.00 | Non-Specified | 6.2 | 6.2 | 230.8 | 0.00 | 0.0 | 0.4 | 0.0
Double click on the yellow rectangle under EAF for Module1
Total Lines of Code: 26220
Estimated | Effort | Sched | PROD | COST | INST | Staff | RISK
Optimistic | 85.7 | 15.1 | 305.9 | 0.00 | 0.0 | 5.7
Most Likely | 107.1 | 16.2 | 244.7 | 0.00 | 0.0 | 6.6 | 0.0
Pessimistic | 133.9 | 17.4 | 195.8 | 0.00 | 0.0 | 7.7
Ready

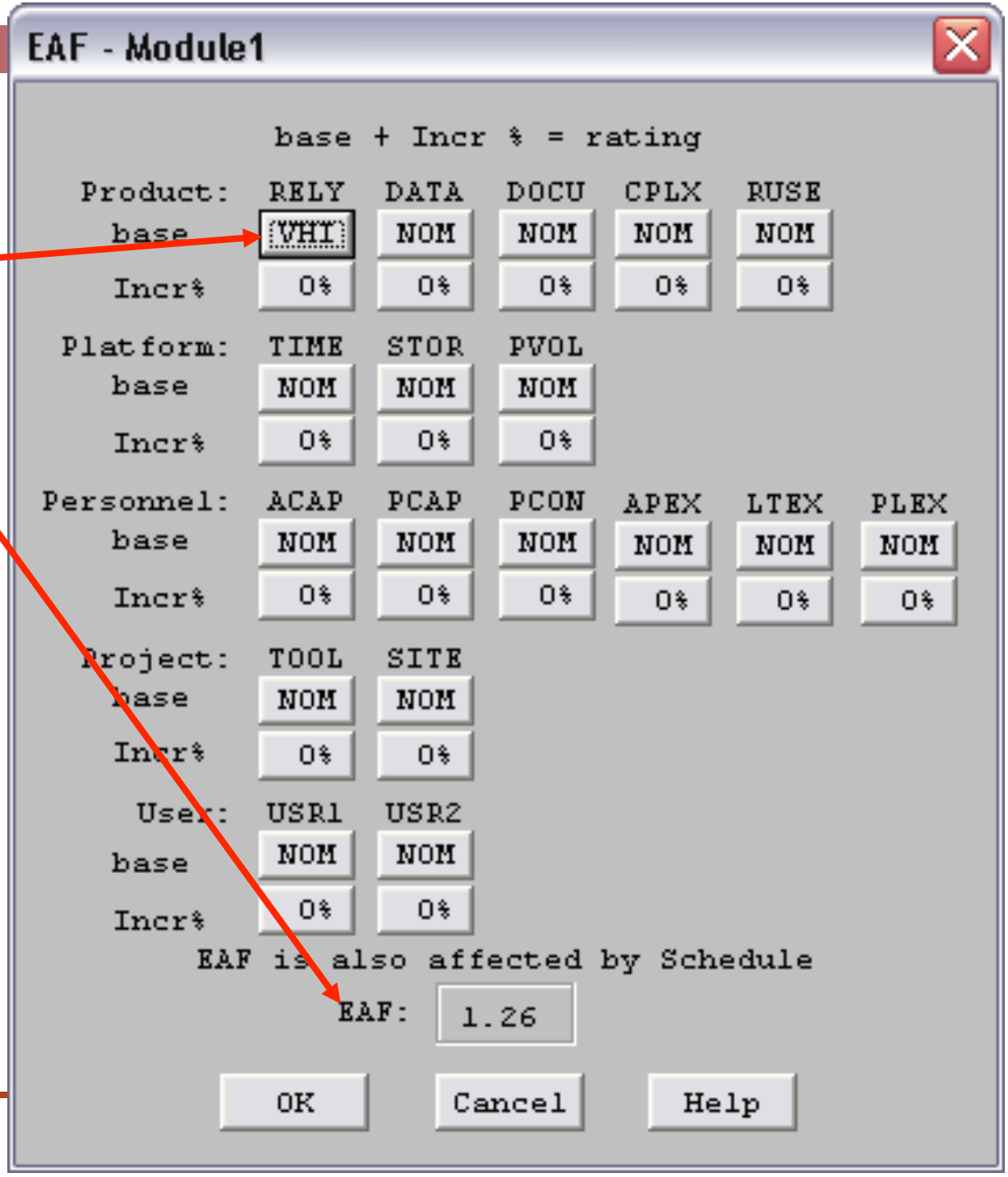
This screen will pop-up. As you click on any given button top row button, the button's title will change (Nom, High,Very High, Very Low, Low, etc) AND you will see the EAF at the bottom of the screen change.

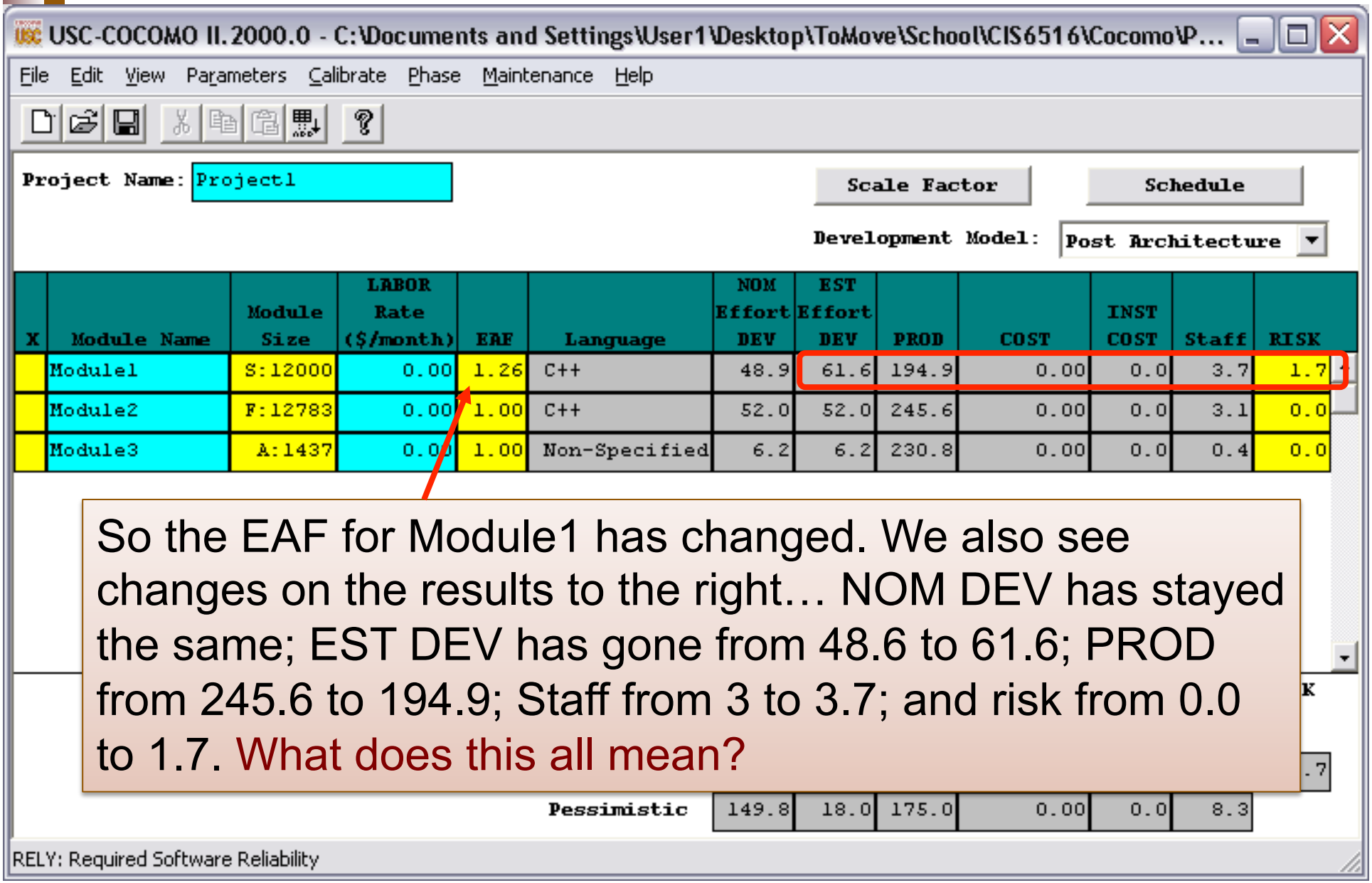


I have changed ONLY the button for RELY to VHI and by doing so, the EAF has changed to 1.26.

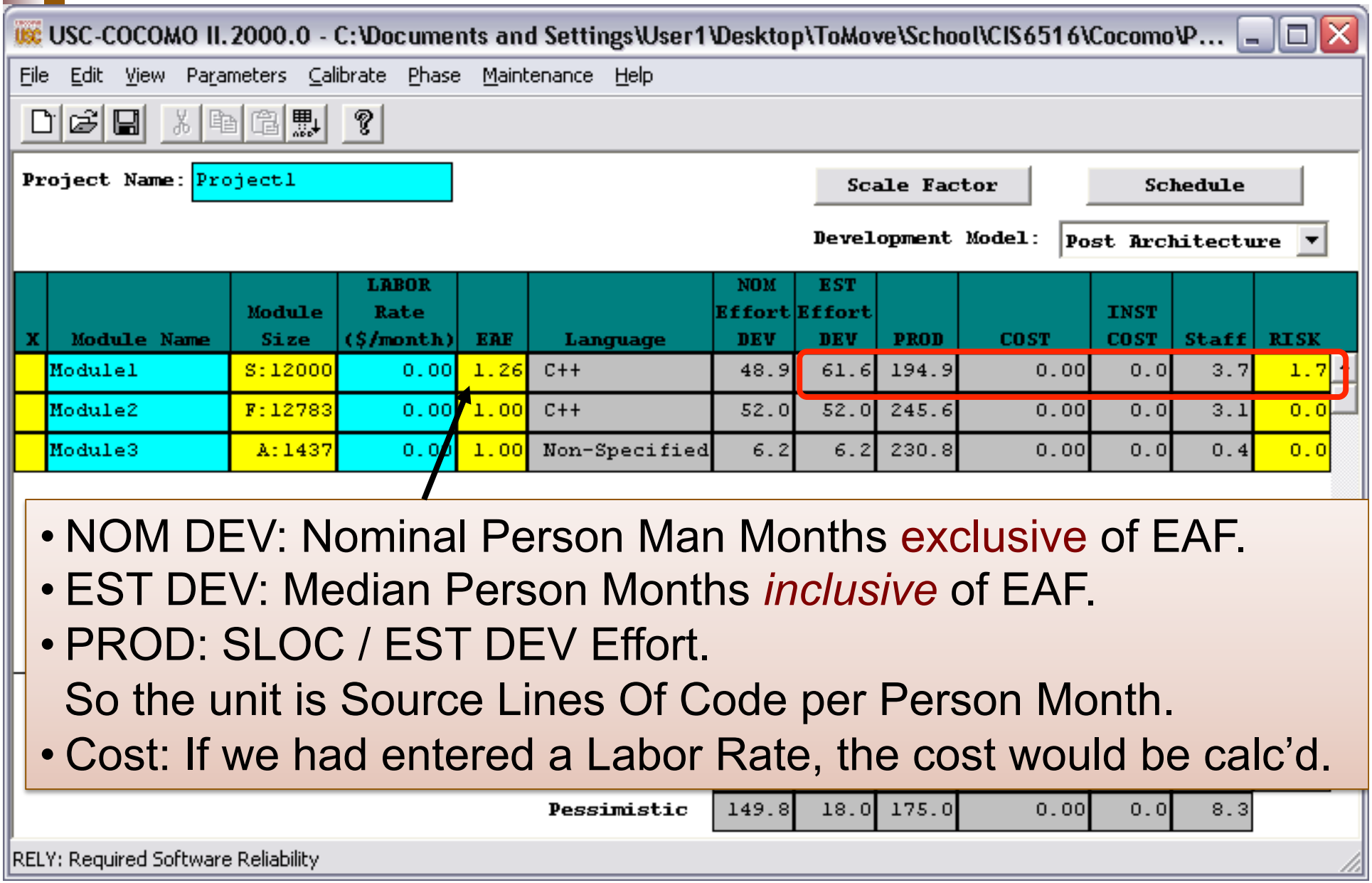
So we just increased the Effort equation by 26%!!!

Click OK to see the result.

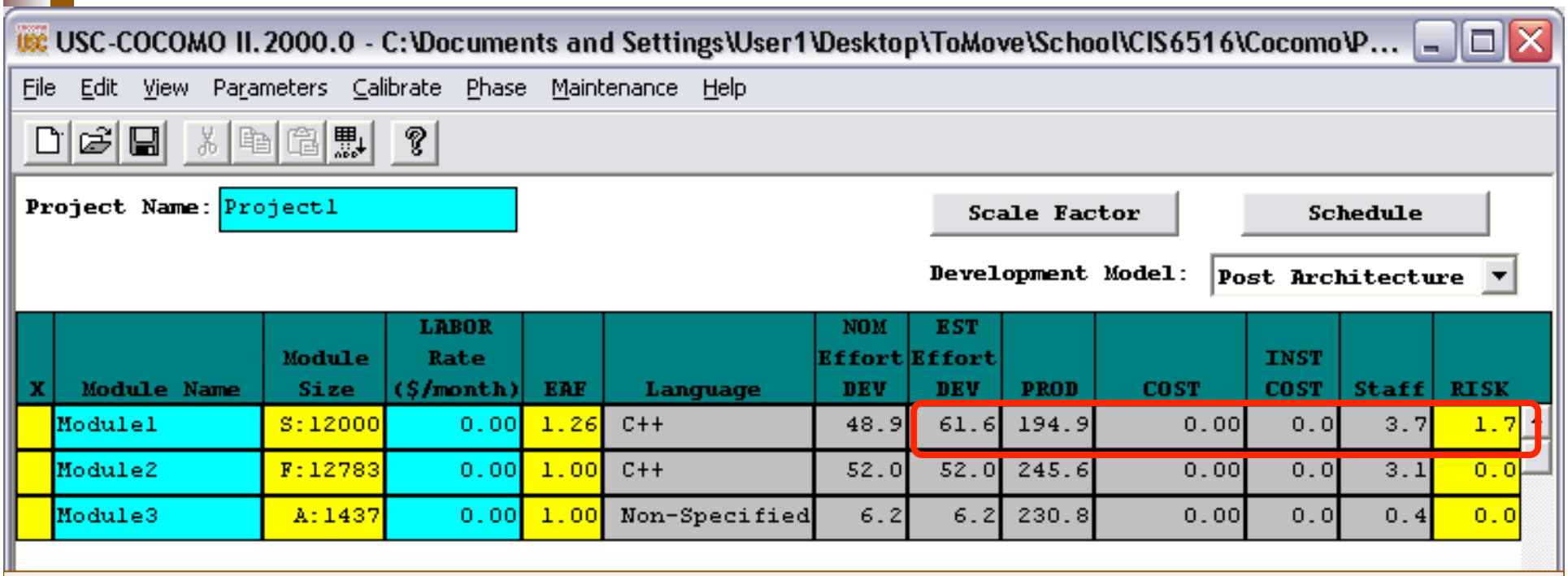




So the EAF for Module1 has changed. We also see changes on the results to the right… NOM DEV has stayed the same; EST DEV has gone from 48.6 to 61.6; PROD from 245.6 to 194.9; Staff from 3 to 3.7; and risk from 0.0 to 1.7. What does this all mean?



- NOM DEV: Nominal Person Man Months exclusive of EAF.
- EST DEV: Median Person Months *inclusive* of EAF.
- PROD: SLOC / EST DEV Effort.
  So the unit is Source Lines Of Code per Person Month.
- Cost: If we had entered a Labor Rate, the cost would be calc'd.

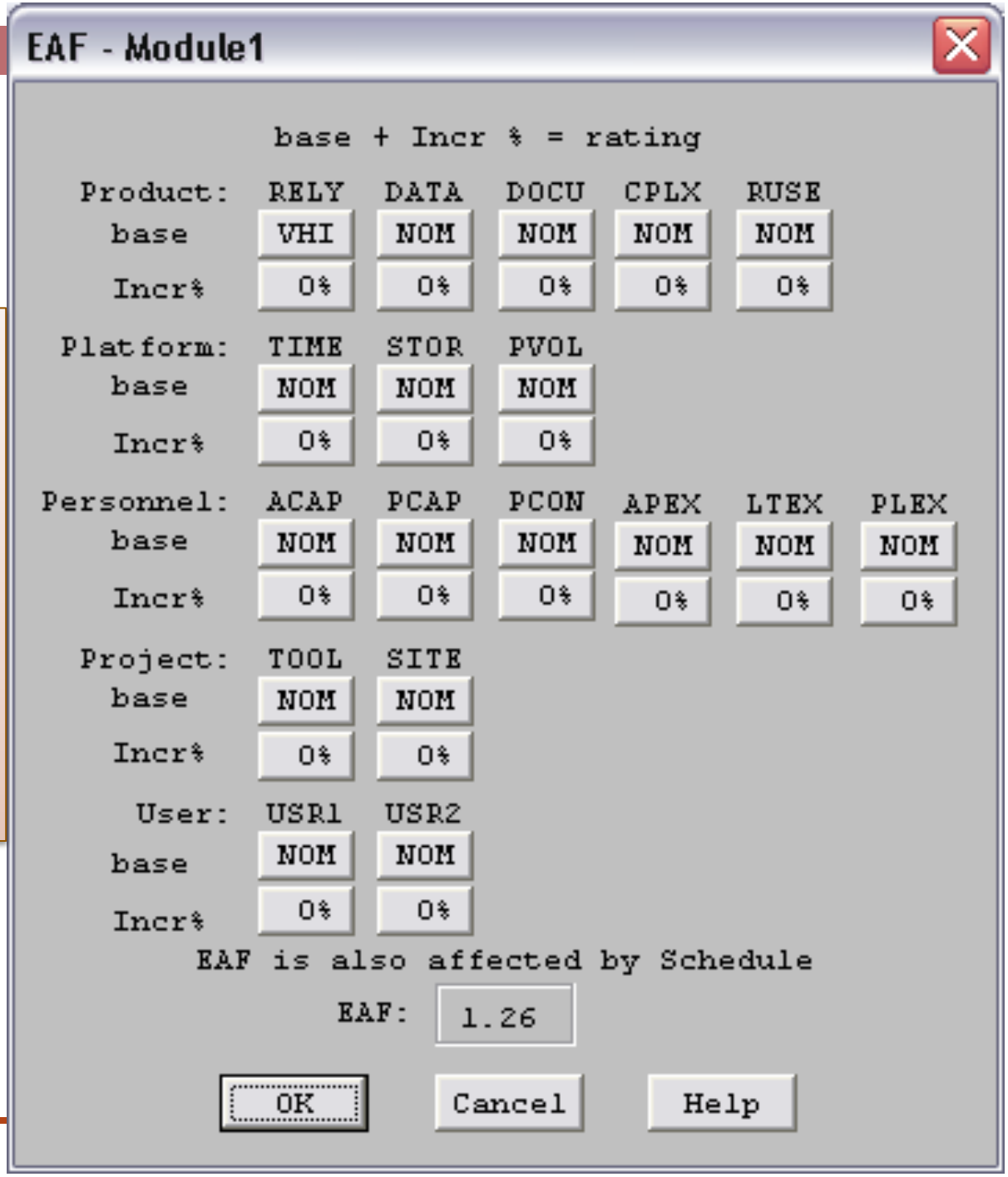


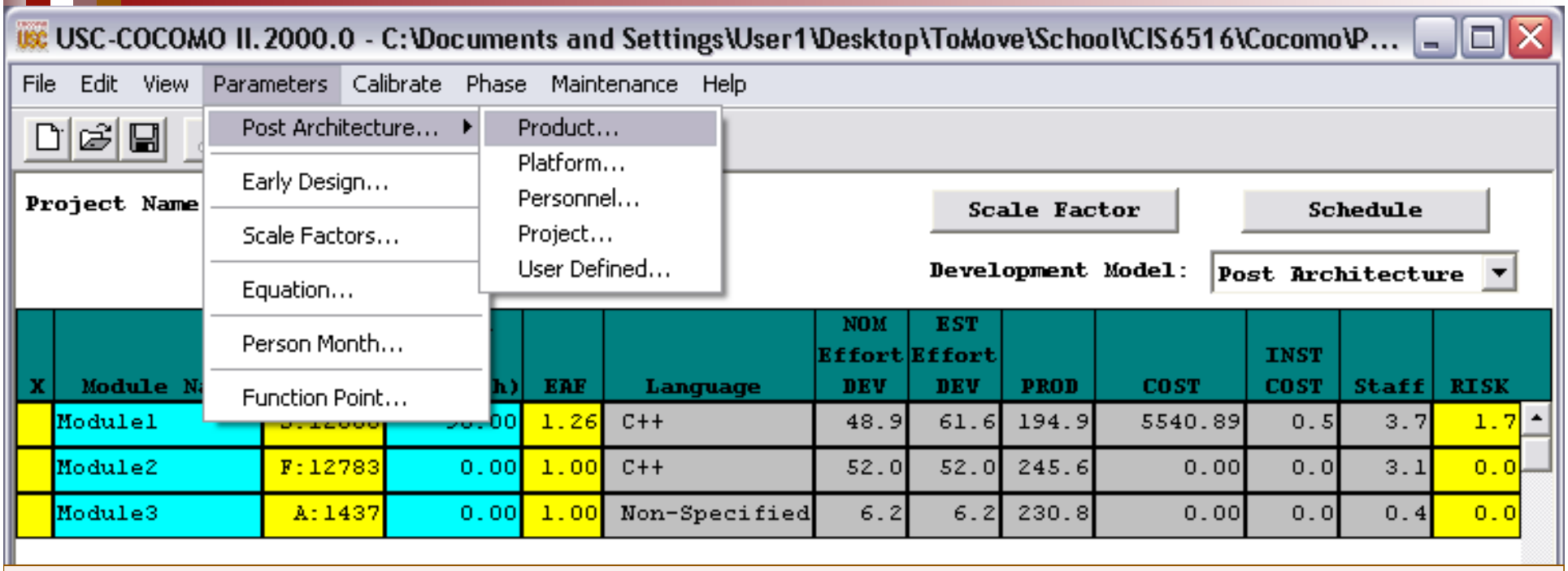
X	Module Name	Module Size	LABOR Rate ($/month)	EAF	Language	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	Module1	S:12000	0.00	1.26	C++	48.9	61.6	194.9	0.00	0.0	3.7	1.7
	Module2	F:12783	0.00	1.00	C++	52.0	52.0	245.6	0.00	0.0	3.1	0.0
	Module3	A:1437	0.00	1.00	Non-Specified	6.2	6.2	230.8	0.00	0.0	0.4	0.0

- INST COST: calculated most likely cost per instruction. This number is calculated from Cost/SLOC in each module.
- Staff: most likely estimate for the number of full-time developers that would be needed to complete a module in the estimated development time.
- RISK: total_risk = schedule_risk + product_risk + personnel_risk + process_risk + platform_risk + reuse_risk. Then total risk of a module=total_risk/373.*100.

In my estimation, Lack of visibility can be risky…

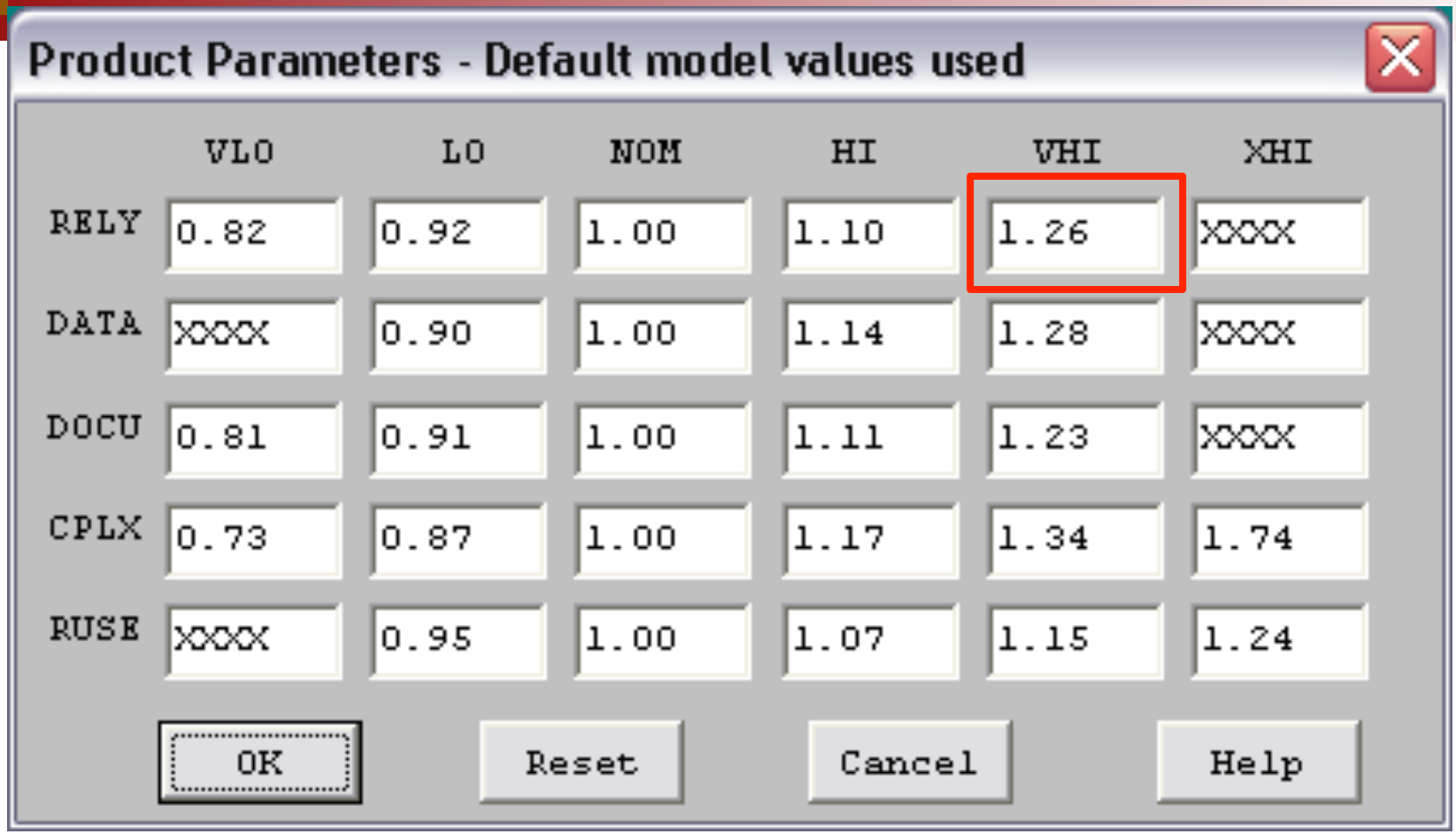
Let's re-visit the EAF screen. What did it mean when we chose "VHI" (Very High)? Qualitatively, that's a nice phrase. But what did it mean *quantitatively*?





To answer that question we need to click on the above menu choices Parameters→Post Architecture→Product.

Note: There are other parameter menus just like the Post-Arch / Product choices. You can see a list of those above. You can adjust Function Point weights, EAF factors for Early Architecture, Scale Factors, the number of hours in a Person Month, etc.



This is where you set the *quantitative* measures associated with your *qualitative* choices. This is how you calibrate COCOMO-II to fit your environment. You can also save your calibrations as a separate loadable module.

"Very High", "Very Low", etc. are ok, but what are the details behind them? To get this answer, you actually have to visit the Model Manual, which is a weakness in COCOMO-II. So for RELY, here is what the Model Manual says:

	Very Low	Low	Nominal	High	Very High	Extra High
RELY	slight incon venience	low, easily recoverable losses	moderate, easily recov erable losses	high finan cial loss	risk to human life	

So selecting "Very High" really meant "Risk to human life" and, through the Parameters→Post Architecture→Product menu choice for RELY changed the EAF contribution by a factor of 1.26.

COCOMO-II: Effort Equation for Post Architecture Model

$$PM = \left(\prod_{i=1}^{17} (EM_i)\right) \cdot A \cdot \left[\left(1 + \frac{REVL}{100}\right) \cdot Size\right]^{\left(0.91 + 0.01 \sum_{j=1}^{5} SF_j\right)} + \left(\frac{ASLOC \cdot \left(\frac{AT}{100}\right)}{ATPROD}\right)$$

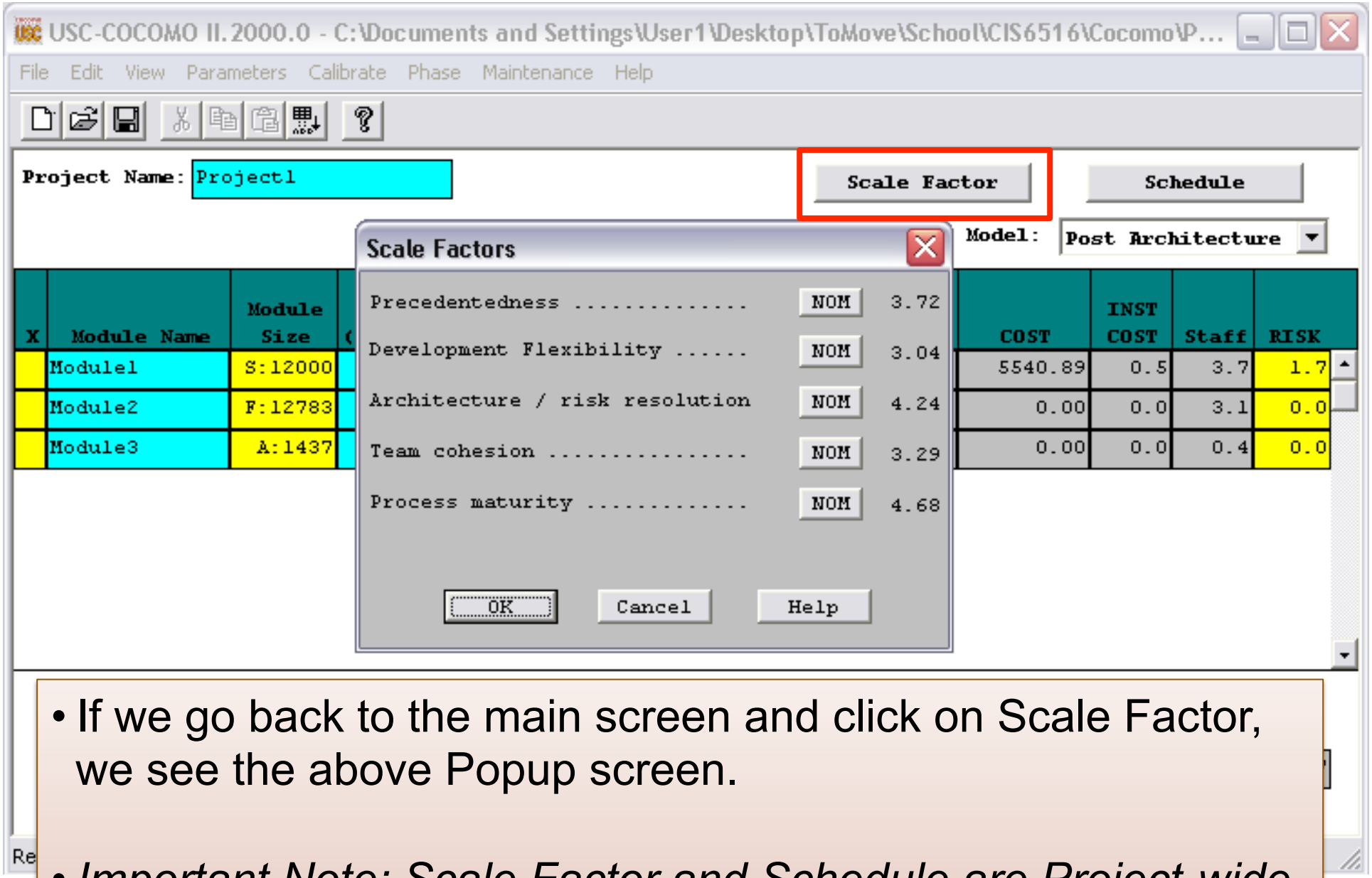
where

$$Size = KNSLOC + \left[KASLOC \cdot \left(\frac{100 - AT}{100}\right) \cdot \frac{(AA + SU + 0.4 \cdot DM + 0.3 \cdot CM + 0.3 \cdot IM)}{100}\right]$$

$$B = 0.91 + 0.01 \sum_{j=1}^{5} SF$$

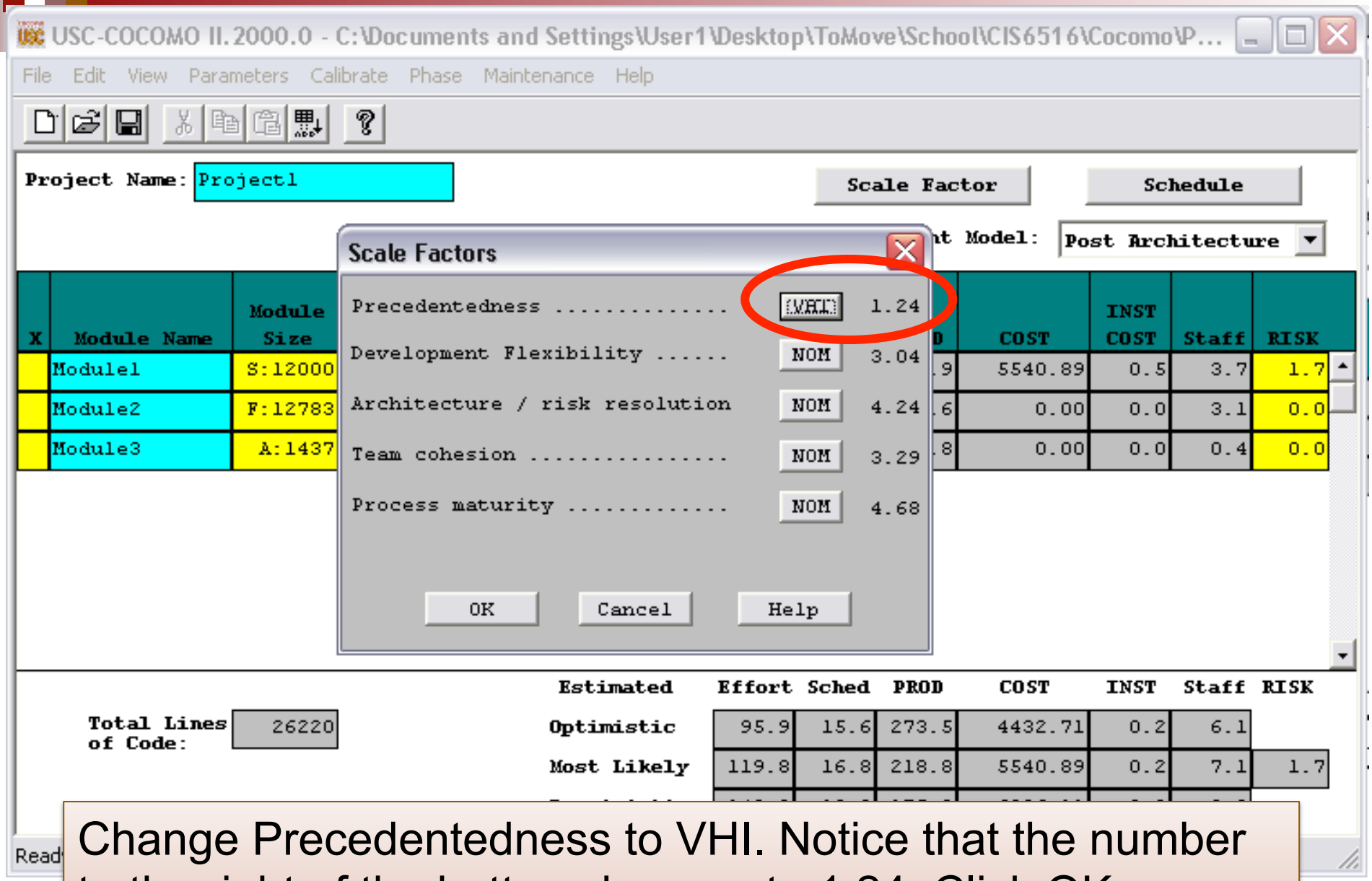
When we adjust the Effort Adjustment Factors, we are impacing the part of the Effort equation in green above…

Lastly in the Effort equation is the Scale Factors.

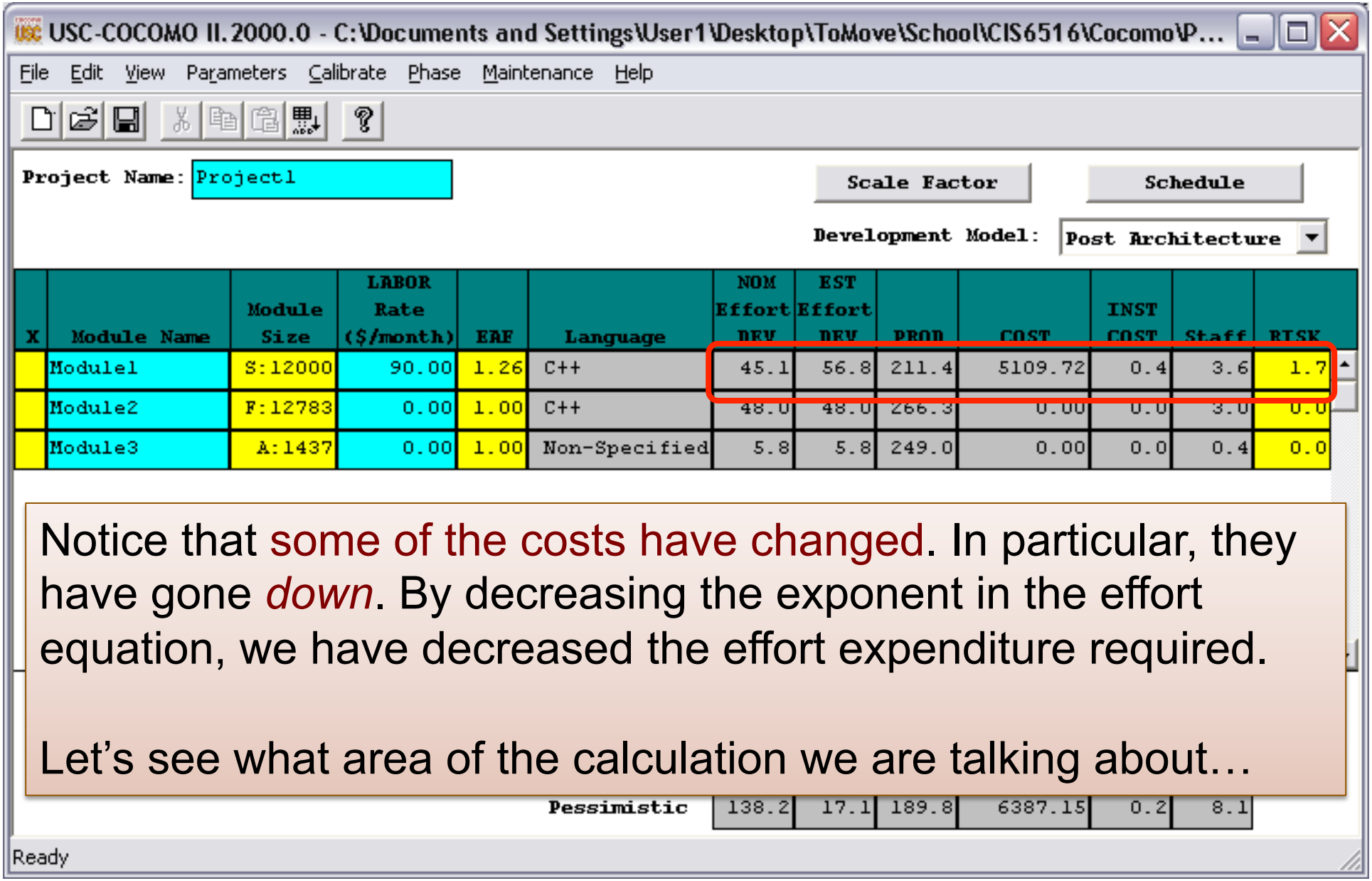


- If we go back to the main screen and click on Scale Factor, we see the above Popup screen.

- *Important Note: Scale Factor and Schedule are Project-wide, not module specific!!!*



Change Precedentedness to VHI. Notice that the number to the right of the button changes to 1.24. Click OK.



Notice that some of the costs have changed. In particular, they have gone *down*. By decreasing the exponent in the effort equation, we have decreased the effort expenditure required.

Let's see what area of the calculation we are talking about…

COCOMO-II Basic Calculations: Effort Equation for Post Architecture Model

Note; this *is* B shown below!

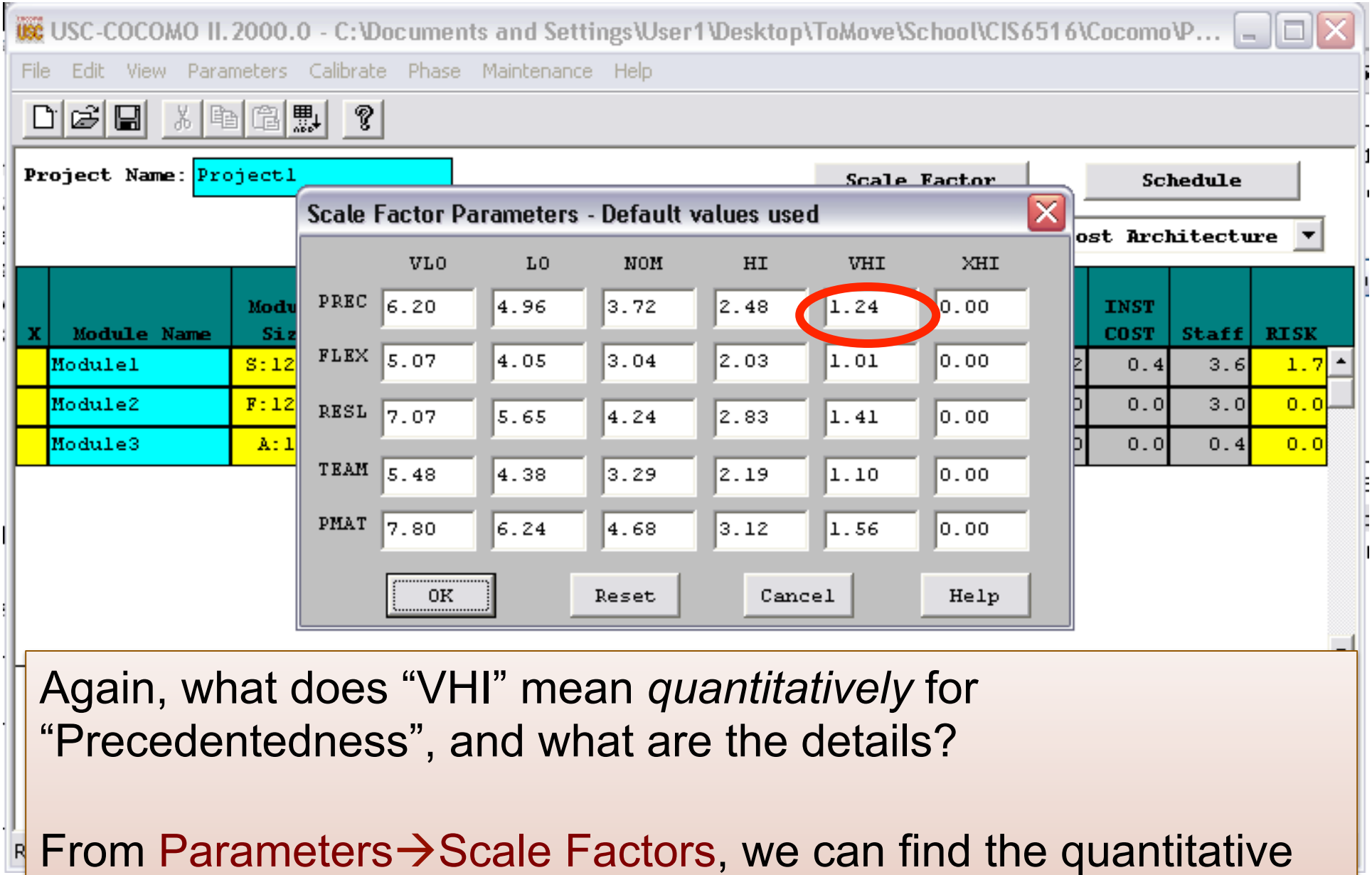
$$PM = \left(\prod_{i=1}^{17} (EM_i)\right) \cdot A \cdot \left[\left(1 + \frac{REVL}{100}\right) \cdot Size\right]^{\left(0.91 + 0.01 \sum_{j=1}^{5} SF_j\right)} + \left(\frac{ASLOC \cdot \left(\frac{AT}{100}\right)}{ATPROD}\right)$$

where

$$Size = KNSLOC + \left[KASLOC \cdot \left(\frac{100 - AT}{100}\right) \cdot \frac{(AA + SU + 0.4 \cdot DM + 0.3 \cdot CM + 0.3 \cdot IM)}{100}\right]$$

$$B = 0.91 + 0.01 \sum_{j=1}^{5} SF$$

So the Scale Factor portion of the Effort Equation is now highlighted in purple!



Again, what does "VHI" mean *quantitatively* for "Precedentedness", and what are the details?

From Parameters→Scale Factors, we can find the quantitative measures shown above in the pop-up…

… and again, from the Model Manual some help on determining how to choose Very Low, Nominal, Very High, etc.

Feature	Very Low	Nominal / High	Extra High
Precedentedness			
Organizational understanding of product objectives	General	Considerable	Thorough
Experience in working with related software systems	Moderate	Considerable	Extensive
Concurrent development of associated new hardware and operational procedures	Extensive	Moderate	Some
Need for innovative data processing architectures, algorithms	Considerable	Some	Minimal

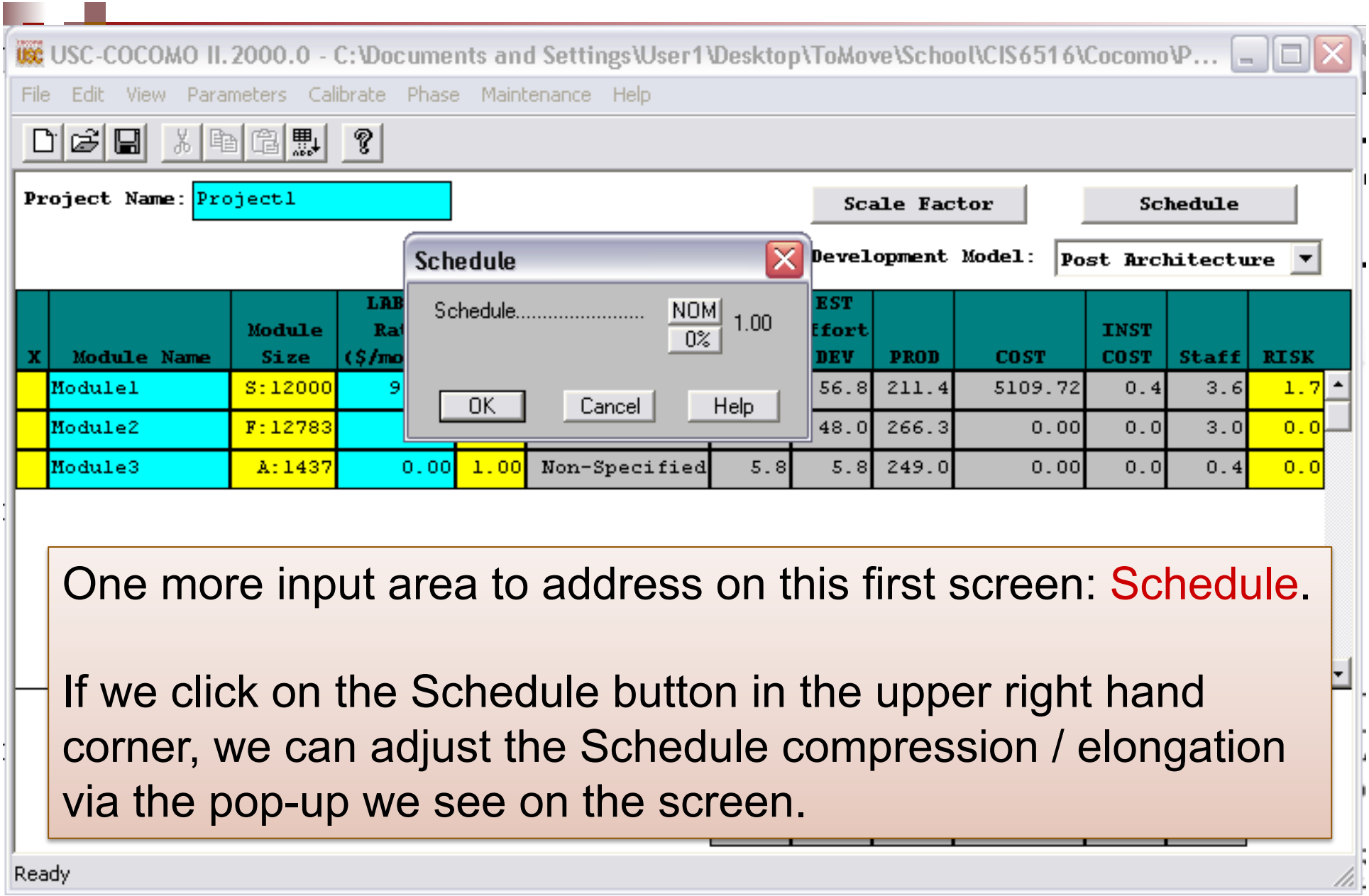
Let's talk about the SCED factor, which applies *project-wide.* The calculation for Schedule is:

$$TDEV = \left[3.67 \times (\overline{PM})^{(0.28 + 0.2 \times (B - 1.01))}\right] \cdot \frac{\text{SCED\%}}{100}$$

where

$$B = 0.91 + 0.01 \sum_{j=1}^{5} SF_j$$

Symbol	Description
PM	Person Months of estimated effort from Early Design or Post-Architecture models (excluding the effect of the SCED effort multiplier).
SF	Scale Factors: PREC, FLEX, RESL, TEAM, PMAT
TDEV	Time to develop (in sequential months)
SCED	Schedule
SCED%	The compression / expansion percentage in the SCED effort multiplier



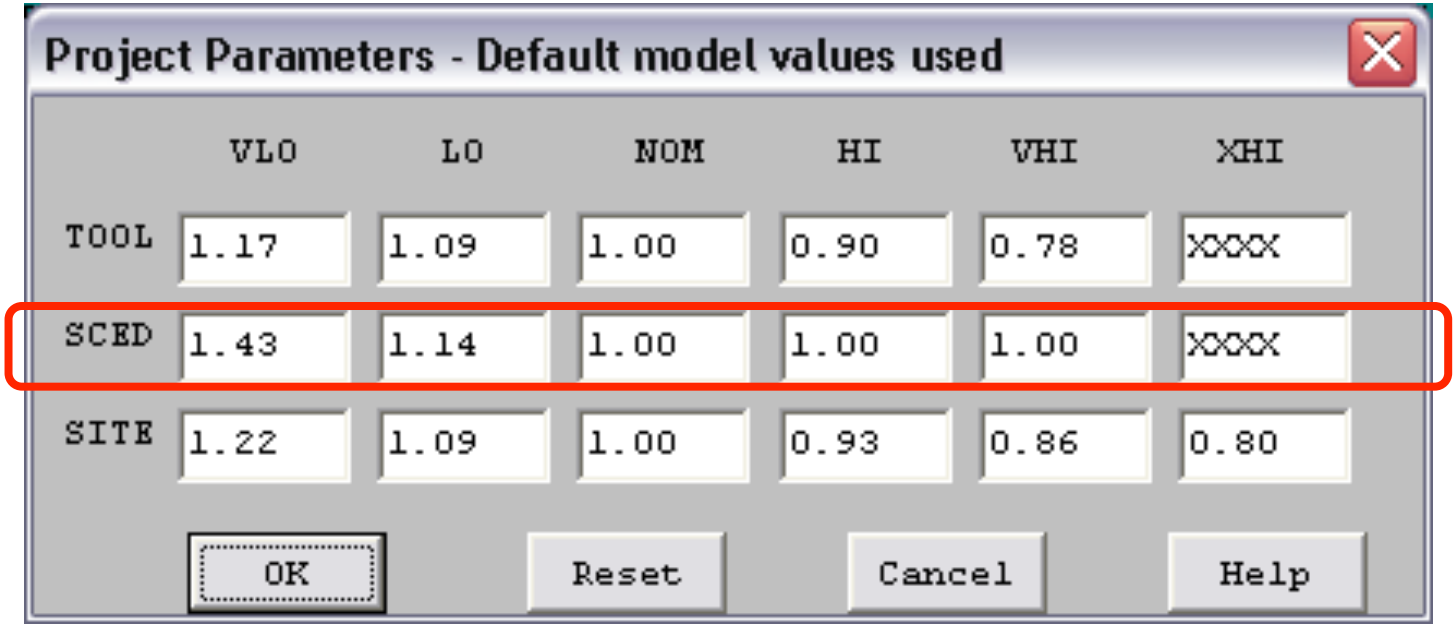
One more input area to address on this first screen: Schedule.

If we click on the Schedule button in the upper right hand corner, we can adjust the Schedule compression / elongation via the pop-up we see on the screen.

Again, from the Model Manual and from the Parameters→Post Architecture→Project menu we can put more quantitative values to our selection.

SCED Rating Level Summary

	Very Low	Low	Nominal	High	Very High	Extra High
SCED	75% of nominal	85%	100%	130%	160%	



Homework and Reading Reminders

- Read Mythical Man Month Paper for Monday's class
- Complete Homework 3 – Software Estimate Using COCOMO-II or Costar
  - Due by 5pm, Tuesday, September 25th, 2012
- Reminder: First Exam is next week (Friday)