Software Cost Automation

Cocomo PostArch Parameters guidelines

Auteur

01/01/2018

Version

SOMMAIRE

Table des matières

1 Precedentedness (PREC) and Development Flexibility (FLEX)

These two scale factors largely capture the differences between the Organic, Semidetached and Embedded modes of the original COCOMO model [Boehm 1981]. Table I-2 reorganizes [Boehm 1981, Table 6.3] to map its project features onto the Precedentedness and Development Flexibility scales. This table can be used as a more in-depth explanation for the PREC and FLEX rating scales given in Table I-1.

Table I-2: Scale Factors Related to COCOMO Development Modes

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
Development Flexibility			
Need for software conformance with preestablished requirements	Full	Considerable	Basic
Need for software conformance with external interface specifications	Full	Considerable	Basic
Premium on early completion	High	Medium	Low

2 Architecture/Risk Resolution (RESL)

This factor combines two of the scale factors in Ada COCOMO, “Design Thoroughness by Product Design Review (PDR)” and “Risk Elimination by PDR” [Boehm and Royce 1989; Figures 4 and 5]. Table I-3 consolidates the Ada COCOMO ratings to form a more comprehensive definition for the COCOMO II RESL rating levels. The RESL rating is the subjective weighted average of the listed characteristics.

Table I-3: RESL Rating Components

Characteristic	Very Low	Low	Nominal	High	Very High	Extra High
Risk Management Plan identifies all critical risk items, establishes milestones for resolving them by PDR.	None	Little	Some	Generally	Mostly	Fully
Schedule, budget, and internal milestones through PDR compatible with Risk Management Plan	None	Little	Some	Generally	Mostly	Fully
Percent of development schedule devoted to establishing architecture, given general product objectives	5	10	17	25	33	40
Percent of required top software architects available to project	20	40	60	80	100	120
Tool support available for resolving risk items, developing and verifying architectural specs	None	Little	Some	Good	Strong	Full
Level of uncertainty in Key architecture drivers: mission, user interface, COTS, hardware, technology, performance.	Extreme	Significant	Considerable	Some	Little	Very Little
Number and criticality of risk items	> 10 Critical	5-10 Critical	2-4 Critical	1 Critical	> 5Non-Critical	< 5 Non-Critical

3 Team Cohesion (TEAM)

The Team Cohesion scale factor accounts for the sources of project turbulence and entropy due to difficulties in synchronizing the project's stakeholders: users, customers, developers, maintainers, interfacers, others. These difficulties may arise from differences in stakeholder objectives and cultures; difficulties in reconciling objectives; and stakeholder's lack of experience and familiarity in operating as a team. Table I-4 provides a detailed definition for the overall TEAM rating levels. The final rating is the subjective weighted average of the listed characteristics.

Table I-4 : TEAM Rating Components

Characteristic	Very Low	Low	Nominal	High	Very High	Extra HIgh
Consistency of stakeholder objectives and cultures	Little	Some	Basic	Considerable	Strong	Full
Ability, willingness of stakeholders to accommodate other stakeholders' objectives	Little	Some	Basic	Considerable	Strong	Full
Experience of stakeholders in operating as a team	None	Little	Little	Basic	Considerable	Extensive
Stakeholder teambuilding to achieve shared vision and commitments	None	Little	Little	Basic	Considerable	Extensive

4 Process Maturity (PMAT)

The procedure for determining PMAT is organized around the Software Engineering Institute's Capability Maturity Model (CMM). The time period for rating Process Maturity is the time the project starts. There are two ways of rating Process Maturity. The first captures the result of an organized evaluation based on the CMM.

Overall Maturity Level

p CMM Level 1 (lower half) p CMM Level 1 (upper half) p CMM Level 2 p CMM Level 3 p CMM Level 4 p CMM Level 5

Key Process Areas

The second is organized around the 18 Key Process Areas (KPAs) in the SEI Capability Maturity Model [Paulk et al. 1993, 1993a]. The procedure for determining PMAT is to decide the percentage of compliance for each of the KPAs. If the project has undergone a recent CMM Assessment, then the percentage compliance for the overall KPA (based on KPA Key Practice compliance assessment data) is used. If an assessment has not been done, then the levels of compliance to the KPA's goals are used (with the Likert scale below) to set the level of compliance. The goal-based level of compliance is determined by a judgement-based averaging across the goals for each Key Process Area. If more information is needed on the KPA goals, they are listed in Appendix C of this document.

Table I-5

Key Process Areas	Almost Always (>90%)	Often (60-90%)	About Half (40-60%)	Occasion -ally (10-40%)	Rarely If Ever (<10%)	Does Not Apply	Don't Know
Requirements Management							
Software Project Planning							
Software Project Tracking and Oversight							
Software Subcontract Management							
Software Quality Assurance							
Software Configuration Management							
Organization Process Focus							
Organization Process Definition							
Training Program							
Integrated Software Management							
Software Product Engineering							
Intergroup Coordination							
Peer Reviews							
Quantitative Process Management							
Software Quality Management							

Defect Prevention
Technology Change Management
Process Change Management

- Check Almost Always when the goals are consistently achieved and are well established in standard operating procedures (over 90% of the time).
- Check Frequently when the goals are achieved relatively often, but sometimes are omitted under difficult circumstances (about 60 to 90% of the time).
- Check About Half when the goals are achieved about half of the time (about 40 to 60% of the time).
- Check Occasionally when the goals are sometimes achieved, but less often (about 10 to 40% of the time).
- Check Rarely If Ever when the goals are rarely if ever achieved (less than 10% of the time).
- Check Does Not Apply when you have the required knowledge about your project or organization and the KPA, but you feel the KPA does not apply to your circumstances.
- Check Don’t Know when you are uncertain about how to respond for the KPA.

After the level of KPA compliance is determined each compliance level is weighted and a PMAT factor is calculated, as in Equation I-3. Initially, all KPAs will be equally weighted.

(EQ I-3)

$$5-\left[\sum_{i=1}^{18}\left(\frac{KPA\%_i}{100}\times\frac{5}{18}\right)\right]$$

5 Required Software Reliability (RELY)

This is the measure of the extent to which the software must perform its intended function over a period of time. If the effect of a software failure is only slight inconvenience, then RELY is low. If a failure would risk human life, then RELY is very high.

	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	

6 Date Base Size (DATA)

This measure attempts to capture the affect large data requirements have on product development. The rating is determined by calculating D/P. The reason the size of the database is important to consider it because of the effort required to generate the test data that will be used to exercise the program.

(EQ II-9)

$$\frac{D}{P} = \frac{DataBaseSize(Bytes)}{ProgramSize(SLOC)}$$

DATA is rated as low if D/P is less than 10 and it is very high if it is greater than 1000.

	Very Low	Low	Nominal	High	Very High	Extra High
DATA		DB bytes/ Pgm SLOC < 10	10 D/P < 100	100 D/P < 1000	D/P 1000	

7 Product Complexity (CPLX)

Table II-15 (found at the end of this section) provides the new COCOMO II CPLX rating scale. Complexity is divided into five areas: control operations, computational operations, device-dependent operations, data management operations, and user interface management operations. Select the area or combination of areas that characterize the product or a sub-system of the product. The complexity rating is the subjective weighted average of these areas.

Table II-15: Module Complexity Ratings versus Type of Module

	Control Operations	**Computational Operations**	**Device dependent Operations**	**Data Management Operations**	**User Interface Management Operations**
Very Low	Straight-line code with a few non-nested structured programming operators: DOs, CASEs, IFTHENELSEs. Simple module composition via procedure calls or simple scripts.	Evaluation of simple expressions: e.g., *A=B+C*(D-E)*	Simple read, write statements with simple formats.	Simple arrays in main memory. Simple COTS-DB queries, updates.	Simple input forms, report generators.
Low	Straightforward nesting of structured programming operators. Mostly simple predicates	Evaluation of moderate-level expressions: e.g., *D=SQRT(B**2-4.*A*C)*	No cognizance needed of particular processor or I/ O device characteristics. I/O done at GET/PUT level.	Single file subsetting with no data structure changes, no edits, no intermediate files. Moderately complex COTS-DB queries, updates.	Use of simple graphic user interface (GUI) builders.
Nominal	Mostly simple nesting. Some intermodule control. Decision tables. Simple callbacks or message passing, including middlewaresupported distributed processing	Use of standard math and statistical routines. Basic matrix/vector operations.	I/O processing includes device selection, status checking and error processing.	Multi-file input and single file output. Simple structural changes, simple edits. Complex COTS-DB queries, updates.	Simple use of widget set.
High	Highly nested structured programming operators with many compound predicates. Queue and stack control. Homogeneous, distributed processing. Single processor soft	Basic numerical analysis: multivariate interpolation, ordinary differential equations. Basic truncation, roundoff concerns.	Operations at physical I/O level (physical storage address translations; seeks, reads, etc.). Optimized I/O overlap.	Simple triggers activated by data stream contents. Complex data restructuring.	Widget set development and extension. Simple voice I/O, multimedia.
	real-time control.				

Very High	Reentrant and recursive coding. Fixed-priority interrupt handling. Task synchronization, complex callbacks, heterogeneous distributed processing. Single-processor hard real-time control.	Difficult but structured numerical analysis: near singular matrix equations, partial differential equations. Simple parallelization.	Routines for interrupt diagnosis, servicing, masking. Communication line handling. Performance intensive embedded systems.	Distributed database coordination. Complex triggers. Search optimization.	Moderately complex 2D/ 3D, dynamic graphics, multimedia.
Extra High	Multiple resource scheduling with dynamically changing priorities. Microcode-level control. Distributed hard real-time control.	Difficult and unstructured numerical analysis: highly accurate analysis of noisy, stochastic data. Complex parallelization.	Device timing dependent coding, micro-programmed operations. Performance-critical embedded systems.	Highly coupled, dynamic relational and object structures. Natural language data management.	Complex multimedia, virtual reality.

8 Required Reusability (RUSE)

This cost driver accounts for the additional effort needed to construct components intended for reuse on the current or future projects. This effort is consumed with creating more generic design of software, more elaborate documentation, and more extensive testing to ensure components are ready for use in other applications.

	Very Low	Low	Nominal	High	Very High	Extra High
RUSE		none	across project	across program	across product line	across multiple product lines

9 Documentation match to life-cycle needs (DOCU)

Several software cost models have a cost driver for the level of required documentation. In COCOMO II, the rating scale for the DOCU cost driver is evaluated in terms of the suitability of the project's documentation to its life-cycle needs. The rating scale goes from Very Low (many life-cycle needs uncovered) to Very High (very excessive for life-cycle needs).

	Very Low	Low	Nominal	High	Very High	Extra High
DOCU	Many lifecycle needs uncovered	Some lifecycle needs uncovered.	Right-sized to life-cycle needs	Excessive for life-cycle needs	Very exces sive for lifecycle needs	

10 Platform Factor

The platform refers to the target-machine complex of hardware and infrastructure software (previously called the virtual machine). The factors have been revised to reflect this as described in this section. Some additional platform factors were considered, such as distribution, parallelism, embeddedness, and real-time operations. These considerations have been accommodated by the expansion of the Module Complexity ratings in Equation II-15.

Execution Time Constraint (TIME)

This is a measure of the execution time constraint imposed upon a software system. The rating is expressed in terms of the percentage of available execution time expected to be used by the system or subsystem consuming the execution time resource. The rating ranges from nominal, less than 50% of the execution time resource used, to extra high, 95% of the execution time resource is consumed.

	Very Low	Low	Nominal	High	Very High	Extra High
TIME			50% use of available execution time	70%	85%	95%

Main Storage Constraint (STOR)

This rating represents the degree of main storage constraint imposed on a software system or subsystem. Given the remarkable increase in available processor execution time and main storage, one can question whether these constraint variables are still relevant. However, many applications continue to expand to consume whatever resources are available, making these cost drivers still relevant. The rating ranges from nominal, less than 50%, to extra high, 95%.

	Very Low	Low	Nominal	High	Very High	Extra High
STOR			50% use of available storage	70%	85%	95%

Platform Volatility (PVOL)

“Platform” is used here to mean the complex of hardware and software (OS, DBMS, etc.) the software product calls on to perform its tasks. If the software to be developed is an operating system, then the platform is the computer hardware. If a database management system is to be developed, then the platform is the hardware and the operating system. If a network text browser is to be developed then the platform is the network, computer hardware, the operating system, and the distributed information repositories. The platform includes any compilers or assemblers supporting the development of the software system. This rating ranges from low, where there is a major change every 12 months, to very high, where there is a major change every two weeks.

	Very Low	Low	Nominal	High	Very High	Extra High
PVOL		major change every 12 mo.; minor change every 1 mo.	major: 6 mo.; minor: 2 wk.	major: 2 mo.; minor: 1 wk.	major: 2 wk.; minor: 2 days	

11 Personnel Factor

Analyst Capability (ACAP)

Analysts are personnel that work on requirements, high level design and detailed design. The major attributes that should be considered in this rating are Analysis and Design ability, efficiency and thoroughness, and the ability to communicate and cooperate. The rating should not consider the level of experience of the analyst; that is rated with AEXP. Analysts that fall in the 15th percentile are rated very low and those that fall in the 95th percentile are rated as very high.

	Very Low	Low	Nominal	High	Very High	Extra High
ACAP	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	

Programmer Capability (PCAP)

Current trends continue to emphasize the importance of highly capable analysts. However, the increasing role of complex COTS packages, and the significant productivity leverage associated with programmers' ability to deal with these COTS packages, indicates a trend toward higher importance of programmer capability as well.

Evaluation should be based on the capability of the programmers as a team rather than as individuals. Major factors which should be considered in the rating are ability, efficiency and thoroughness, and the ability to communicate and cooperate. The experience of the programmer should not be considered here; it is rated with AEXP. A very low rated programmer team is in the 15th percentile and a very high rated programmer team is in the 95th percentile.

	Very Low	Low	Nominal	High	Very High	Extra High
PCAP	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	

Applications Experience (AEXP)

This rating is dependent on the level of applications experience of the project team developing the software system or subsystem. The ratings are defined in terms of the project team's equivalent level of experience with this type of application. A very low rating is for application experience of less than 2 months. A very high rating is for experience of 6 years or more.

	Very Low	Low	Nominal	High	Very High	Extra High
AEXP	2 months	6 months	1 year	3 years	6 years	

Language and Tool Experience (LTEX)

This is a measure of the level of programming language and software tool experience of the project team developing the software system or subsystem. Software development includes the use of tools that perform requirements and design representation and analysis, configuration management, document extraction, library management, program style and formatting, consistency checking, etc. In addition to experience in programming with a specific language the supporting tool set also effects development time. A low rating given for experience of less than 2 months. A very high rating is given for experience of 6 or more years.

	Very Low	Low	Nominal	High	Very High	Extra High
LTEX	2 months	6 months	1 year	3 years	6 year	

Personnel Continuity (PCON)

The rating scale for PCON is in terms of the project's annual personnel turnover: from 3%, very high, to 48%, very low.

	Very Low	Low	Nominal	High	Very High	Extra High
PCON	48% / year	24% / year	12% / year	6% / year	3% / year	

12 Project Factors

Use of Software Tools (TOOL)

Software tools have improved significantly since the 1970's projects used to calibrate COCOMO. The tool rating ranges from simple edit and code, very low, to integrated lifecycle management tools, very high.

	Very Low	Low	Nominal	High	Very High	Extra High
TOOL	edit, code, debug	simple, fron tend, back end CASE, little integra tion	basic lifecy cle tools, moderately integrated	strong, mature life cycle tools, moderately integrated	strong, mature, pro active lifecy cle tools, well inte grated with processes, methods, reuse	

Multisite Development (SITE)

Given the increasing frequency of multisite developments, and indications that multisite development effects are significant, the SITE cost driver has been added in COCOMO II. Determining its cost driver rating involves the assessment and averaging of two factors: site collocation (from fully collocated to international distribution) and communication support (from surface mail and some phone access to full interactive multimedia).

	Very Low	Low	Nominal	High	Very High	Extra High
SITE: Communica tions	Some phone, mail	Individual phone, FAX	Narrowband email	Wideband electronic communica tion.	Wideband elect. comm, occasional video conf.	Interactive multimedia

Required Development Schedule (SCED)

This rating measures the schedule constraint imposed on the project team developing the software. The ratings are defined in terms of the percentage of schedule stretch-out or acceleration with respect to a nominal schedule for a project requiring a given amount of effort. Accelerated schedules tend to produce more effort in the later phases of development because more issues are left to be determined due to lack of time to resolve them earlier. A schedule compress of 74% is rated very low. A stretch-out of a schedule produces more effort in the earlier phases of development where there is more time for thorough planning, specification and validation. A stretch-out of 160% is rated very high.

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

Table II-16: Post-Architecture Cost Driver Rating Level Summary

	Very Low	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
RELY	slight inconvenience	low, easily recoverable losses	Moderate, easily recoverable losses	high financial loss	risk to human life	
DATA		DB bytes/Pgm SLOC < 10	10 D/P < 100	100 D/P < 1000	D/P 1000	
CPLX	see Table II-15					
RUSE		none	Across project	across program	across product line	across multiple product lines
DOCU	Many life-cycle needs uncovered	Some life-cycle needs uncovered.	Right-sized to life-cycle needs	Excessive for life-cycle needs	Very excessive for life-cycle needs	
TIME			50% use of available execution time	70%	85%	95%
STOR			50% use of available storage	70%	85%	95%
PVOL		major change every 12 mo.; minor change every 1 mo.	major: 6 mo.; minor: 2 wk.	major: 2 mo.; minor: 1 wk.	major: 2 wk.; minor: 2 days	
ACAP	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	
PCAP	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	
PCON	48% / year	24% / year	12% / year	6% / year	3% / year	
AEXP	2 months	6 months	1 year	3 years	6 years	
PEXP	2 months	6 months	1 year	3 years	6 year	
LTEX	2 months	6 months	1 year	3 years	6 year	
TOOL	edit, code,	simple, fron	basic lifecycle	strong, mature	strong, mature,	

	debug	tend, backend CASE, little integration	tools, moderately integrated	lifecycle tools, moderately integrated	proactive life cycle tools, well integrated with processes, methods, reuse	
SITE: Collocation	International	Multi-city and Multi-company	Multi-city or Multi-company	Same city or metro. area	Same building or complex	Fully collocated
SITE: Communications	Some phone, mail	Individual phone, FAX	Narrowband email	Wideband electronic communication.	Wideband elect. comm, occasional video conf.	Interactive multimedia
SCED	75% of nominal	85%	100%	130%	160%	