

КОНСТРУКТИВНАЯ МОДЕЛЬ СТОИМОСТИ СОСОМО

COCOMO (Constructive Cost Model) – это конструктивная модель стоимости, разработанная в начале 80-х годов Барри Боэмом для оценки трудоемкости разработки программных продуктов¹. Она основана на статистическом анализе фактических данных по выполнению 63 проектов в компании TRW Aerospace, где Барри Боэм был директором отдела исследований программного обеспечения и технологий. Анализировались проекты объемом от 2 до 100 тысяч строк кода, на языках программирования от ассемблеров до высокоуровневого языка PL/1, основанные на каскадной модели жизненного цикла разработки ПО.

Модель состоит из иерархии трех последовательно детализируемых и уточняемых уровней [3]. На каждом уровне все проекты разбиваются на три группы по уровню сложности:

- 1) распространенный тип (organic projects);
- 2) встроенный тип (embedded projects);
- 3) полунезависимый тип (semidetached projects).

Распространенный тип характеризуется тем, что проект выполняется небольшой группой специалистов, имеющих опыт в создании подобных изделий и опыт применения технологических средств. Условия работы стабильны, и изделие имеет относительно невысокую сложность.

Встроенный тип характеризуется очень жесткими требованиями на программный продукт, интерфейсы, параметры ЭВМ. Как правило, у таких изделий высокая степень новизны и планирование работ осуществляется при недостаточной информации, как о самом изделии, так и об условиях работы. Встроенный проект требует больших затрат на изменения и исправления.

Полунезависимый тип занимает промежуточное положение между распространенным и встроенным – это проекты средней сложности. Исполнители знакомы лишь с некоторыми характеристиками (или компонентами) создаваемой системы, имеют средний опыт работы с подобными изделиями, изделие имеет элемент новизны. Только часть требований к изделию жестко фиксируется, в остальном разработки имеют степени выбора.

Тип той или иной группы можно рассматривать как один из параметров модели COCOMO.

Рассмотрим уровни модели.

¹ Опубликовано в книге Barry Boehm. *Software Engineering Economics* (Englewood Cliffs, NJ: Prentice-Hall, 1981).

2.1.1 Базовый уровень (Basic COCOMO)

Модель этого уровня – двухпараметрическая. В качестве параметров выступают тип проекта и объем программы (число строк кода).

Уравнения базового уровня модели имеют вид:

$$PM = a_i \times (SIZE)^{b_i}$$
$$TM = c_i \times (PM)^{d_i},$$

где

PM (People×Month) – трудоемкость (чел.×мес.);

TM (Time at Month) – время разработки в календарных месяцах;

SIZE – объем программного продукта в тысячах строк исходного текста (Kilo of Source Line of Code – *KSLOC*).

Коэффициенты a_i , b_i , c_i и d_i выбираются из табл. 2.1.

Таблица 2.1. Значения коэффициентов базовой уровня модели COCOMO в зависимости от типа проекта

Тип проекта	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>
Распространенный	2,4	1,05	2,5	0,38
Полунезависимый	3,0	1,12	2,5	0,35
Встроенный	3,6	1,20	2,5	0,32

Модель этого уровня подходит для ранней быстрой приблизительной оценки затрат, но точность её весьма низка, т.к. не учитываются такие факторы, как квалификация персонала, характеристики оборудования, опыт применения современных методов разработки программного обеспечения и современных инструментальных сред разработки и др.

2.1.2 Промежуточный уровень (Intermediate COCOMO)

На этом уровне базовая модель уточнена за счет ввода дополнительных 15 «атрибутов стоимости» (или факторов затрат) *Cost Drivers* (CD_k), которые сгруппированы по четырем категориям:

□ **Характеристики продукта (Product Attributes):**

- Требуемая надежность ПО (Required Software Reliability);
- Размер БД приложения (Size of Application Database);
- Сложность продукта (Complexity of the Product);

□ **Характеристики аппаратного обеспечения (Hardware Attributes):** □

Ограничения быстродействия при выполнении программы (Run-Time Performance Constraints);

- Ограничения памяти (Memory Constraints);

- Неустойчивость окружения виртуальной машины (Volatility of the Virtual Machine Environment);
- Требуемое время восстановления (Required Turnabout Time);
- **Характеристики персонала (Personnel Attributes):**
 - Аналитические способности (Analyst Capability);
 - Способности к разработке ПО (Software Engineer Capability);
 - Опыт разработки (Applications Experience);
 - Опыт использования виртуальных машин (Virtual Machine Experience);
 - Опыт разработки на языках программирования (Programming Language Experience);
- **Характеристики проекта (Project Attributes):**
 - Использование инструментария разработки ПО (Use of Software Tools);
 - Применение методов разработки ПО (Application of Software Engineering Methods);
 - Требования соблюдения графика разработки (Required Development Schedule).

Значения каждого атрибута выбирается из табл. 2.2 в соответствии с его степенью значимости (рейтингом) в конкретном проекте.

Таблица 2.2. Значения атрибутов стоимости в зависимости от их уровня

Атрибуты стоимости, CD_k	Рейтинг					
	Очень низкий	Низкий	Средний	Высокий	Очень высокий	Критический
Характеристики продукта						
1. Требуемая надежность ПО	0,75	0,88	1,00	1,15	1,40	n/a
2. Размер БД приложения	n/a	0,94	1,00	1,08	1,16	n/a
3. Сложность продукта	0,70	0,85	1,00	1,15	1,30	1,65
Характеристики аппаратного обеспечения						
4. Ограничения быстродействия при выполнении программы	n/a	n/a	1,00	1,11	1,30	1,66
5. Ограничения памяти	n/a	n/a	1,00	1,06	1,21	1,56
6. Неустойчивость окружения виртуальной машины	n/a	0,87	1,00	1,15	1,30	n/a
7. Требуемое время восстановления	n/a	0,87	1,00	1,07	1,15	n/a

Характеристики персонала						
8. Аналитические способности	1,46	1,19	1,00	0,86	0,71	n/a
9. Опыт разработки	1,29	1,13	1,00	0,91	0,82	n/a
10. Способности к разработке ПО	1,42	1,17	1,00	0,86	0,70	n/a
11. Опыт использования виртуальных машин	1,21	1,10	1,00	0,90	n/a	n/a
12. Опыт разработки на языках программирования	1,14	1,07	1,00	0,95	n/a	n/a
Характеристики проекта						
13. Применение методов разработки ПО	1,24	1,10	1,00	0,91	0,82	n/a
14. Использование инструментария разработки ПО	1,24	1,10	1,00	0,91	0,83	n/a
15. Требования соблюдения графика разработки	1,23	1,08	1,00	1,04	1,10	n/a

Примечание: n/a (not available) – данные отсутствуют, т.е. соответствующий уровень не оценивается

Формула промежуточного уровня модели имеет вид

$$PM = EAF \times a_i \times (SIZE)^{b_i},$$

где PM – трудоемкость (чел.×мес.);

$SIZE$ – объем программного продукта в тысячах строк исходного текста (Kilo of Source Line of Code – $KSLOC$).

EAF (Effort Adjustment Factor) – произведение выбранных атрибутов

15

стоимости из табл. 2.2: $EAF = \prod_{k=1}^n CD_k$.

$k=1$

Коэффициенты модели a_i и b_i выбираются из табл. 2.3.

Таблица 2.3. Значения коэффициентов промежуточного уровня модели COCOMO в зависимости от типа проекта

Тип проекта, i	Значения коэффициентов	
	a_i	b_i
1. Распространенный	3,2	1,05
2. Полунезависимый	3,0	1,12
3. Встроенный	2,8	1,20

Время разработки рассчитывается по той же формуле, что и для базовой модели.

2.1.3 Детальный уровень (Advanced COCOMO)

Повышает точность оценки за счет иерархической декомпозиции создаваемого ПО и учета стоимостных факторов на каждом уровне иерархии и по фазам работ (здесь не рассматривается).

В 1997 методика была усовершенствована и получила название COCOMO II. Калибровка параметров производилась уже по 161 проекту разработки ПО.

Различаются две стадии оценки проекта: *предварительная* оценка на начальной фазе (Early Design) и *детальная* оценка после проработки архитектуры (Post Architecture).

Формула оценки трудоемкости проекта в чел.×мес. имеет вид:

$$PM = EAF \times A \times (SIZE)^E,$$

5

где $E = B \prod_{i=1}^n SF_i$;

$B = 0,91$; $A = 2,94$ для предварительной оценки; $A = 2,45$ для детальной оценки²;

SF_i – факторы масштаба (Scale Factors) (табл. 5);

$SIZE$ – объем программного продукта в тысячах строк исходного текста (KSLOC – Kilo of Source Line of Code);

EAF (Effort Adjustment Factor) – произведение выбранных множителей

трудоемкости: $EAF = \prod_{j=1}^n EM_j$;

EM_j – множители трудоемкости³ (Effort Multipliers). $n=7$ для предварительной оценки (табл. 6), $n=17$ для детальной оценки (табл. 7).

2.2.1 Факторы масштаба (Scale Factors)

² Приведены параметры модели COCOMO II.2000 (Version 2.1). Источник: COCOMOII: Model Definition Manual. – USC, Center for Software Engineering. – URL: <http://csse.usc.edu/csse/TECHRPTS/2000/usccse2000-500/usccse2000-500.pdf> ³ Иное название: Cost Driver (CD) – атрибут стоимости (фактор затрат).

В методике COCOMO II используются пять факторов масштаба SF_i , описание которых приведено в табл.4.

Таблица 4. Описание уровней значимости факторов масштаба

SF_i	Описание	Уровень значимости фактора					
		Очень низкий	Низкий	Средний	Высокий	Очень высокий	Критический
1. PREC. Precedentedness	Прецедентность, наличие опыта аналогичных разработок	опыт в продукте и платформе отсутствует	продукт и платформа не много знакомы	некоторый опыт в продукте и платформе присутствует	продукт и платформа в основном известны	продукт и платформа в большой степени знакомы	продукт и платформа полностью знакомы
2. FLEX. Development Flexibility	Гибкость процесса разработки	процесс строго детерминирован	допускаются некоторые компромиссы	значительная жесткость процесса	относительная жесткость процесса	незначительная жесткость процесса	определены только общие цели
3. RESL. Architecture / Risk Resolution	Архитектура и разрешение рисков	риски известны/проанализированы на 20%	риски известны/проанализированы на 40%	риски известны/проанализированы на 60%	риски известны/проанализированы на 75%	риски известны/проанализированы на 90%	риски разрешены на 100%
4. TEAM. Team Cohesion	Сработанность команды	формальные взаимодействия	тяжелое взаимодействие до некоторой степени	чаще всего коллективная работа	в основном коллективная работа	высокая степень взаимодействия	полное доверие, взаимозаменяемость и взаимопомощь
5. PMAT. Process Maturity	Зрелость процессов	CMM Уровень 1 (ниже среднего)	CMM Уровень 1 (выше среднего)	CMM Уровень 2	CMM Уровень 3	CMM Уровень 4	CMM Level 5

Примечание. CMM (Capability Maturity Model) — пятиуровневая модель зрелости возможностей компании-разработчика ПО, предложенная SEI (Software Engineering Institute, США).

Эти факторы применяются **на обеих стадиях оценки** проекта.

Числовые значения фактора масштаба в зависимости от оценки его уровня, приведены в таблице 5.

Таблица 5. Значение фактора масштаба в зависимости от оценки его уровня

Фактор масштаба, SF_j	Оценка уровня фактора					
	Very Low	Low	Nominal	High	Very High	Extra High
1. PREC	6,20	4,96	3,72	2,48	1,24	0,00
2. FLEX	5,07	4,05	3,04	2,03	1,01	0,00
3. RESL	7,07	5,65	4,24	2,83	1,41	0,00
4. TEAM	5,48	4,38	3,29	2,19	1,10	0,00
5. PMAT	7,80	6,24	4,68	3,12	1,56	0,00

2.2.2 Множители трудоемкости (Effort Multipliers)

Количество и значения множителей трудоёмкости отличаются для разных стадий оценки проекта.

1. Стадия **предварительной оценки трудоемкости программного проекта (Early Design)**. Для этой оценки необходимо оценить для проекта уровень **семи множителей** трудоемкости EM_j :

– **параметры персонала:**

1. PERS (Personnel Capability) – квалификация персонала (Extra Low – аналитики и программисты имеют низшую квалификацию, текучесть больше 45%; Extra High – аналитики и программисты имеют высшую квалификацию, текучесть меньше 4%);

2. PREX (Personnel Experience) – опыт персонала (Extra Low – новое приложение, инструменты и платформа; Extra High – приложение, инструменты и платформа хорошо известны);

– **параметры продукта:**

3. RCPX (Product Reliability and Complexity) – сложность и надежность продукта (Extra Low – продукт простой, специальных требований по надежности нет, БД маленькая, документация не требуется; Extra High – продукт очень сложный, требования по надежности жесткие, БД сверхбольшая, документация требуется в полном объеме);

4. RUSE (Developed for Reusability) – разработка для повторного использования (Low – не требуется; Extra High – предполагается переиспользование в других продуктах);

– **параметры платформы:**

5. PDIF (Platform Difficulty) – сложность платформы разработки (Extra Low – специальные ограничения по памяти и быстродействию отсутствуют, платформа стабильна; Extra High – жесткие ограничения по памяти и быстродействию, платформа нестабильна);

– **параметры проекта:**

6. FCIL (Facilities)– оборудование (Extra Low – инструменты простейшие, коммуникации затруднены; Extra High – интегрированные средства поддержки жизненного цикла, интерактивные мультимедиа коммуникации);

7. SCED (Required Development Schedule) – требуемое выполнение графика работ (Very Low – 75% от номинальной длительности; Very High – 160% от номинальной длительности).

Значения множителей трудоемкости в зависимости от их уровня приведены в табл. 6.

Таблица 6. Значения множителей трудоемкости в зависимости от оценки их уровня (Early Design)

№	Множитель трудоемкости, <i>EM_i</i>	Оценка уровня множителя трудоемкости						
		Extra Low	Very Low	Low	Nominal	High	Very High	Extra High
1	PERS	2,12	1,62	1,26	1,00	0,83	0,63	0,50
2	PREX	1,59	1,33	1,22	1,00	0,87	0,74	0,62
3	RCPX	0,49	0,60	0,83	1,00	1,33	1,91	2,72
4	RUSE	n/a	n/a	0,95	1,00	1,07	1,15	1,24
5	PDIF	n/a	n/a	0,87	1,00	1,29	1,81	2,61
6	FCIL	1,43	1,30	1,10	1,00	0,87	0,73	0,62
7	SCED	n/a	1,43	1,14	1,00	1,00	n/a	n/a

Примечание: n/a (not available) – данные отсутствуют, т.е. соответствующий уровень не оценивается

2. Стадия детальной оценки после проработки архитектуры (Post Architecture). Для этой оценки необходимо оценить для проекта уровень семнадцати множителей трудоемкости *EM_j*:

– **параметры персонала:**

- 1) Analyst Capability (ACAP) – возможности аналитика;
- 2) Applications Experience (AEXP) – опыт разработки приложений;
- 3) Programmer Capability (PCAP) – возможности программиста;
- 4) Personnel Continuity (PCON) – продолжительность работы персонала;
- 5) Platform Experience (PEXP) – опыт работы с платформой;

6) Language and Tool Experience (LTEX) – опыт использования языка программирования и инструментальных средств. – **параметры продукта:**

7) Required Software Reliability (RELY) – требуемая надежность программы;

8) Database Size (DATA) – размер базы данных;

9) Software Product Complexity (CPLX) – сложность программы;

10) Required Reusability (RUSE) – требуемая возможность многократного использования;

11) Documentation Match to Life-Cycle Needs (DOCU) – соответствие документации потребностям жизненного цикла.

– **параметры платформы:**

12) Execution Time Constraint (TIME) – ограничения времени выполнения;

13) Main Storage Constraint (STOR) – ограничения памяти; 14) Platform Volatility (PVOL) – изменяемость платформы.

– **параметры проекта:**

15) Use of Software Tools (TOOL) – использование инструментальных программных средств;

16) Multisite Development (SITE) – многоабонентская (удаленная) разработка;

17) Required Development Schedule (SCED) – требуемое выполнение графика работ.

Значения множителей трудоемкости в зависимости от их уровня приведены в табл. 7.

Таблица 7. Значения множителей трудоемкости в зависимости от оценки их уровня (Post Architecture)

№	Effort Multiplier, EM_I		Very Low	Low	Nominal	High	Very High	Extra High
	<i>Personnel Factors</i>							
1	ACAP	Analyst Capability	1,42	1,29	1,00	0,85	0,71	n/a
2	AEXP	Applications Experience	1,22	1,10	1,00	0,88	0,81	n/a
3	PCAP	Programmer Capability	1,34	1,15	1,00	0,88	0,76	n/a
4	PCON	Personnel Continuity	1,29	1,12	1,00	0,90	0,81	n/a
5	PEXP	Platform Experience	1,19	1,09	1,00	0,91	0,85	n/a
6	LTEX	Language and Tool Experience	1,20	1,09	1,00	0,91	0,84	n/a

	Product Factors							
7	RELY	Required Software Reliability	0,84	0,92	1,00	1,10	1,26	n/a
8	DATA	Database Size	n/a	0,23	1,00	1,14	1,28	n/a
9	CPLX	Software Product Complexity	0,73	0,87	1,00	1,17	1,34	1,74
10	RUSE	Required Reusability	n/a	0,95	1,00	1,07	1,15	1,24
11	DOCU	Documentation Match to Life-Cycle Needs	0,81	0,91	1,00	1,11	1,23	n/a
	Platform Factors							
12	TIME	Execution Time Constraint	n/a	n/a	1,00	1,11	1,29	1,63
13	STOR	Main Storage Constraint	n/a	n/a	1,00	1,05	1,17	1,46
14	PVOL	Platform Volatility	n/a	0,87	1,00	1,15	1,30	n/a
	Project Factors							
15	TOOL	Use of Software Tools	1,17	1,09	1,00	0,90	0,78	n/a
17	SITE	Multisite Development	1,22	1,09	1,00	0,93	0,86	0,80
16	SCED	Required Development Schedule	1,43	1,14	1,00	1,00	1,00	n/a

Примечание: n/a (not available) – данные отсутствуют, т.е. соответствующий уровень не оценивается

2.2.3 Оценка длительности проекта

Время разработки проекта **TM** в методике COCOMO II для обоих уровней рассчитывается по формуле:

$$TM = SCED \times C \times (PM_{NS})^{D+0,2 \times (E-B)},$$

где $C = 3,67$; $D = 0,28$;

PM_{NS} – рассчитанная трудоемкость проекта без учета множителя $SCED$, определяющего сжатие расписания. Остальные параметры определены выше.