

# Оценка затрат, стоимости и длительности разработки ПО

Технологии разработки программного обеспечения

Виноградова М.В.  
МГТУ им. Н.Э. Баумана  
Кафедра СОИУ (ИУ5)

# Основные оценки

- Затраты (трудозатраты) [чел-месяц]
- Стоимость [тыс.руб.]
- Длительность [месяц]

# Оценка бюджета

- Экспертная оценка
  - различные эксперты
  - обсуждение
- Оценка по аналогии
- На основе графика выполнения
- Алгоритмическая оценка
  - настройки
  - метрики
  - результат практики и опыта
- желательно использовать  $\geq 2$  штук для проверки адекватности

# Оценка бюджета на основе графика выполнения

- план на основе исходных данных
- стоимость ресурсов (зависит от графика ресурсов; может изменяться)
- фиксированные затраты
- нормы и ставки
- накладные расходы
  
- Отслеживание выполнения проекта:
  - план на основе исходных данных.  $\Leftrightarrow$  фактический план
  - бюджет на основе исходных данных.  $\Leftrightarrow$  фактический бюджет
- Причины несоответствия:
  - появление/разбиение задач (  $\Rightarrow$  корректировка плана)
  - недоступность ресурсов
  - система бухучета (не все затраты могут быть учтены) (бухучет и управление проектом не синхронны)

# Алгоритмические методы оценки

- Размерно-ориентированные метрики
- Функционально-ориентированные метрики
- Модель конструктивной стоимости  
COCOMO

# Размерно-ориентированные метрики

- **LOC (Lines of code)** - количество строк в ПО.
- Исходные данные для расчета метрик в таблице (по проектам за последние несколько лет)
- Таблицы аналогов (метрический базис фрмы)

Проект	Затраты чел-мес	Стоимость тыс. \$	KLOC тыс. LOC	Прочие документы стр.	Ошибки	Люди
П1	24	168	12,1	365	29	3
П2	62	440	27,2	1224	86	5
П3	43	314	20,2	1050	64	6

# Метрики

- На основе таблицы аналогов (метрического базиса) вычисляются размерно-ориентированные метрики для любого проекта

$$\text{Производительность} = \frac{\text{Длина}}{\text{Затраты}} \left[ \frac{\text{тыс. LOC}}{\text{чел.-мес.}} \right]$$

$$\text{Качество} = \frac{\text{Ошибки}}{\text{Длина}} \left[ \frac{\text{Единицы}}{\text{тыс. LOC}} \right]$$

$$\text{Удельная стоимость} = \frac{\text{Стоимость}}{\text{Длина}} \left[ \frac{\text{тыс. \$}}{\text{тыс. LOC}} \right]$$

$$\text{Документированность} = \frac{\text{Страниц документа}}{\text{Длина}} \left[ \frac{\text{страниц}}{\text{тыс. LOC}} \right]$$

# Особенности ROM

- Плюсы:
  - Широко распространены;
  - Просты и легко вычисляются
  - Прямо измеряют ПО и процесс разработки.
- Минусы:
  - Зависят от языка программирования;
  - Требуют исходных данных, которые трудно получить на начальной стадии проекта;
  - Не приспособлены к непроцедурным языкам программирования

# Процесс оценки

Шаг 1: Декомпозиция продукта на функции (каждая из них оценивается отдельно).

$f_1, f_2, \dots, f_n$

Шаг 2: Для каждой  $f_i$  определить:

$LOC(FP)_{лучші}$

$LOC(FP)_{худші}$

$LOC(FP)_{вероятні}$

Определяются на основе метрического базиса или интуиции. Диапазон значений ~ предусмотренной неопределенности.

Шаг 3: Для каждой  $f_i$  по  $\beta$  – распределению определить ожидаемую LOC (FP):

$$LOC_{ожі} = (LOC_{лучші} + LOC_{худші} + 4 * LOC_{вероятні}) / 6.$$

# Процесс оценки

Шаг 4: Определить LOC(FP) производительность разработки по вариантам:

А. Для каждой  $f_i$  взять произвольную среднюю из метрического базиса (минимальная точность, максимальная простота вычислений).

Б. Для каждой  $f_i$  вычислить по формуле:

$$\text{ПРОИЗВ}_i = \text{ПРОИЗВ}_{\text{ср}} \times (\text{LOC}_{\text{ср}} / \text{LOC}_{\text{ож}i}),$$

$\text{LOC}_{\text{ср}}$  - из метрического базиса

В. Для каждой  $f_i$  вычислить по формуле:

$$\text{ПРОИЗВ}_i = \text{ПРОИЗВ}_{\text{ан}i} \times (\text{LOC}_{\text{ан}i} / \text{LOC}_{\text{ож}i}).$$

$\text{ПРОИЗВ}_{\text{ан}i}$  и  $\text{LOC}_{\text{ан}i}$  - по аналогу  $f_i$  из метрического базиса.

(Максимальная точность, максимальная сложность вычислений)

# Процесс оценки

Шаг 5: Определить общую оценку затрат на проект:

Для А:

$$\text{ЗАТРАТЫ} = \left( \sum_{i=1}^n \text{LOC}_{\text{ож}} \right) / \text{ПРОИЗВ}_{\text{ср}} \left[ \text{чел. - мес} \right]$$

Для Б и В:

$$\text{ЗАТРАТЫ} = \sum_{i=1}^n \left( \text{LOC}_{\text{ож}} / \text{ПРОИЗВ}_i \right) \left[ \text{чел. - мес} \right].$$

# Процесс оценки

Шаг 6: Определить общую оценку стоимости:

Для А и Б:

$$\text{СТОИМОСТЬ} = \left( \sum_{i=1}^n \text{LOC}_{\text{ожг}i} \right) \times \text{УД\_СТОИМОСТЬ}_{\text{ср}}$$

$\text{УД\_СТОИМОСТЬ}_{\text{ср}}$  — средняя удельная стоимости LOC из метрического базиса.

Для В:

$$\text{СТОИМОСТЬ} = \sum_{i=1}^n (\text{LOC}_{\text{ожг}i} \times \text{УД\_СТОИМОСТЬ}_{\text{анг}i})$$

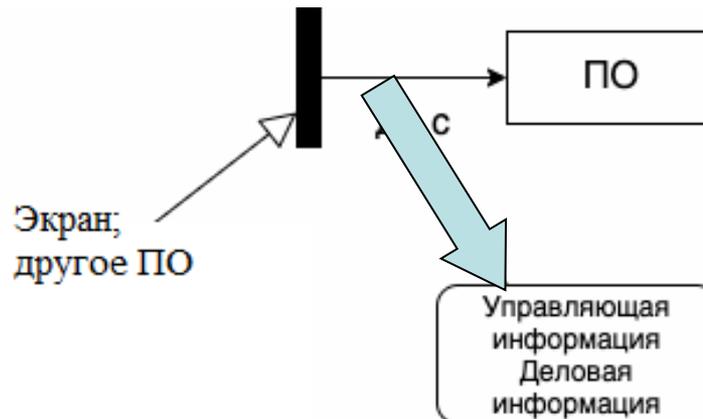
$\text{УД\_СТОИМОСТЬ}_{\text{анг}i}$  — средняя удельная стоимость для аналога из метрического базиса.

# Функционально-ориентированные метрики

- Вместо размера (LOC) учет функциональности или полезности ПО.
- Косвенное измерение ПО и процесса его разработки.
- Информационные характеристики:
  - Количество внешних вводов, выводов и запросов
  - Количество внешних интерфейсных и внутренних логических файлов

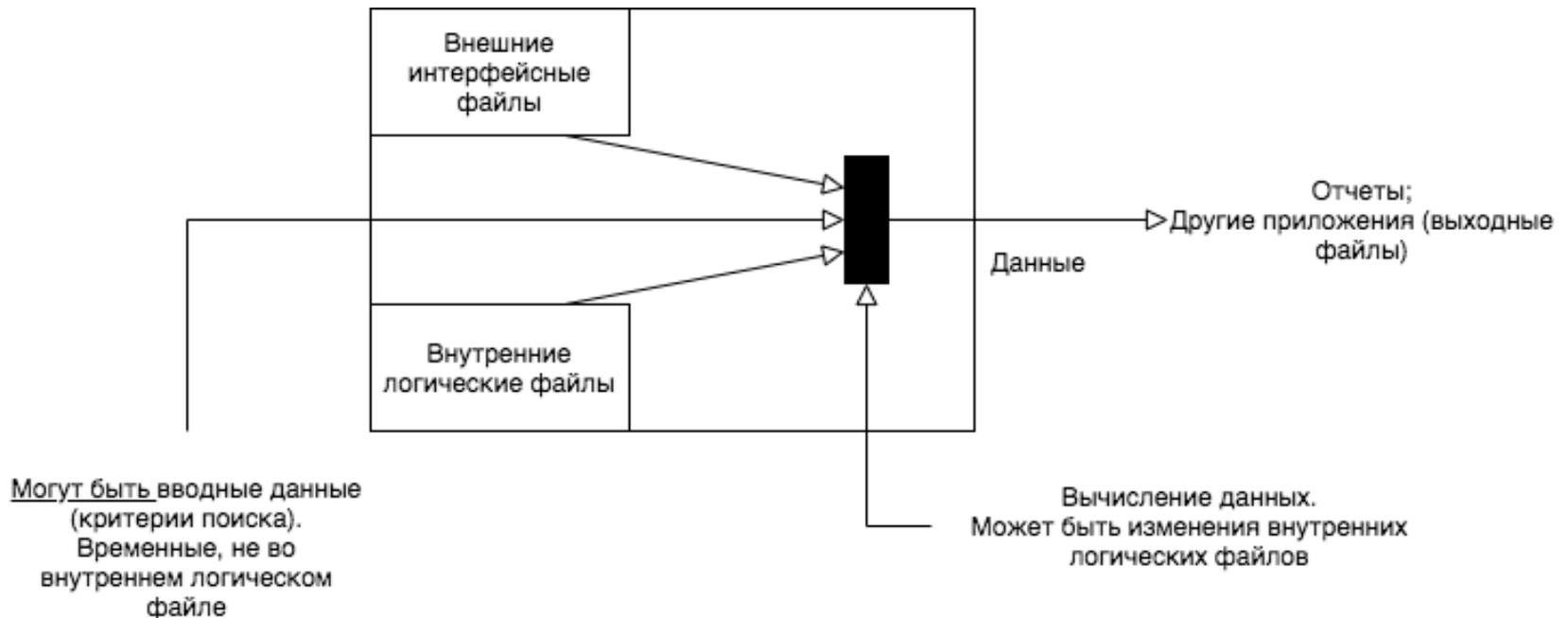
# Количество внешних вводов

- Внешний ввод - элементарный процесс, различаемый пользователем и перемещающий данные между внешней средой и ПО. Изменяет файлы.
- Управляющая информация - не изменяет внутренний логический файл;
- Деловая информация - может изменять внутренний логический файл.



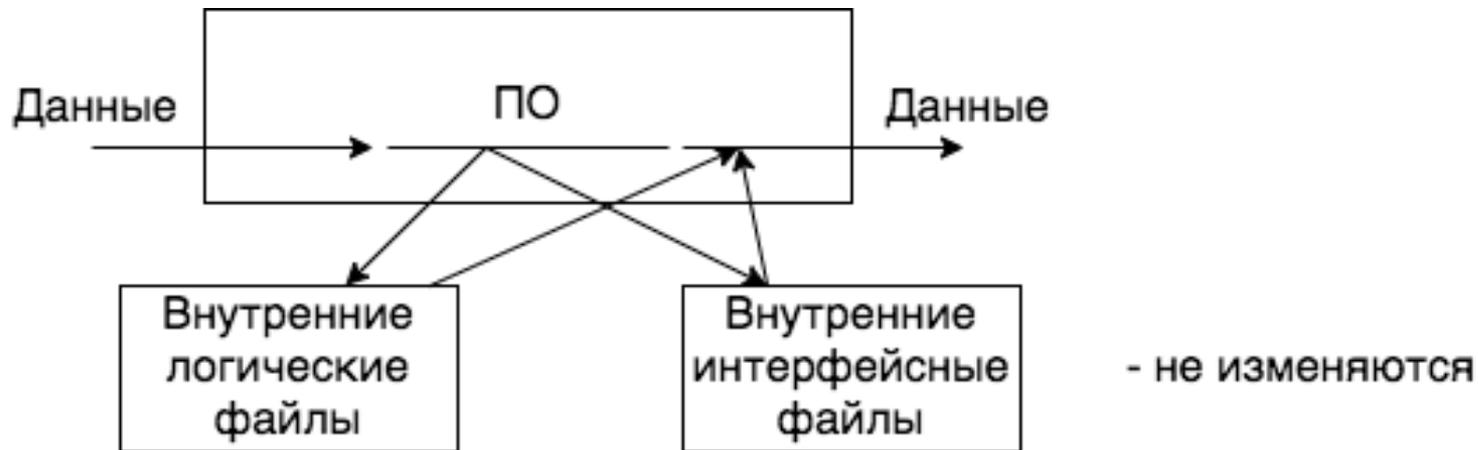
# Количество внешних выводов

- Внешний вывод - элементарный процесс
- Выводит данные из файла. Перед выводом обрабатывает данные



# Количество внешних запросов

- Внешний запрос - элементарный процесс;
- диалоговый ввод -> сразу вывод;
- нет сохранения вводимых данных;
- нет вычисляемых выходных данных.



# Количество внутренних логических файлов

- Распознаваемая пользователем группа логически связанных данных внутри ПО;
- обслуживается через внешние вводы

# Количество внешних интерфейсных файлов

- Распознаваемая пользователем группа логически связанных данных внутри ПО;
- обслуживается через внешние вводы внутри другого приложения, которым поддерживается.
- Внутренний файл приложения 1 - это внешний файл приложения 2.

# Примеры информационных характеристик

Внешний ввод	Добавление / удаление / изменение данных о Персоне
Внешний вывод	Отчет о доходах Персоны за год
Внешний запрос	Раскрывающийся список названий организаций, где работает Персона
Внутренний логический файл	БД Персон; Файл конфигурации профиля
Внешний интерфейсный файл	Сервис налоговой службы

# Оценка ранга и сложности транзакций (ввод, вывод, запрос)

- Для транзакций (ввод, вывод, запрос - коэф. отличаются)
- Р - ранг - низкий, средний, высокий.  
С – сложность (от 3 до 15)

Количество ссылок на файлы	Количество элементов данных		
	1 - 4	5 - 15 (19)	>15 (19)
0 - 1	P(C)		
2 (2 - 3)			
>2 (>3)			

# Оценка ранга и сложности файлов

- Для файлов коэф. отличаются (внутренний - сложнее )
- Р - ранг - низкий, средний, высокий.  
С – сложность (значение)
- Тип элемента записи - подгруппа главных данных в пределах файла, распознаваемая пользователем.
- Элемент данных - уникальное (не рекурсивное) поле, распознаваемое пользователем.

Количество типов элементов записей	Количество элементов данных		
	1 - 19	20 - 50	>50
1			
2 - 5	P(C)		
>5			

# Метрика “Количество функциональных указателей”

FP (Functional Points) Албрехт, 1979г.

Характеристика	Ранг, сложность, количество			
	Низкий	Средний	Высокий	Итого
Внешние вводы	_x3=_	_x4=_	_x6=_	-
Внешние выводы	_x4=_	_x5=_	_x7=_	-
Внешние запросы	_x3=_	_x4=_	_x6=_	-
Внутренние логические файлы	_x7=_	_x10=_	_x15=_	-
Внешние интерфейсные файлы	_x5=_	_x7=_	_x10=_	-
Общее количество				<u>OK</u>

# Количество функциональных указателей

$$FP = \text{Общ\_колич} \times (0,65 + 0,01 * \text{Sum}(Fi))$$

где  $F_i$  — коэффициенты регулирования сложности.  
от 0 — нет влияния, 1 — случайное, 2 — небольшое,  
3 — среднее, 4 — важное, 5 — основное.

(K от 0,65 до 1,35)

# Коэффициенты регулировки сложности

1	Передачи данных	Сколько средств связи требуется для передачи или обмена информацией с приложением или системой?
2	Распределенная обработка данных	Как обрабатываются распределенные данные и функции обработки?
3	Производительность	Нуждается ли пользователь в фиксации времени ответа или производительности?.
4	Распространенность используемой конфигурации	Насколько распространена текущая аппаратная платформа, на которой будет выполняться приложение?
5	Скорость транзакций	Как часто выполняются транзакции? (каждый день, каждую неделю, каждый месяц)
6	Оперативный ввод данных	Какой процент информации надо вводить в режиме онлайн?
7	Эффективность работы конечного пользователя	Приложение проектировалось для обеспечения эффективной работы конечного пользователя?
8	Оперативное обновление	Как много внутренних файлов обновляется в онлайн-транзакции?
9	Сложность обработки	Выполняет ли приложение интенсивную логическую или математическую обработку?
10	Повторная используемость	Приложение разрабатывалось для удовлетворения требований одного или многих пользователей?
11	Легкость инсталляции	Насколько трудны преобразование и инсталляция приложения?
12	Легкость эксплуатации	Насколько эффективны и/или автоматизированы процедуры запуска, резервирования и восстановления?
13	Разнообразные условия размещения	Была ли спроектирована, разработана и поддержана возможность инсталляции приложения в разных местах для различных организаций?
14	Простота изменений	Была ли спроектирована, разработана и поддержана в приложении простота изменений?

# Метрики на основе FP

$$\text{Производитель} = \frac{\text{ФункцУказатель}}{\text{Затраты}} \left[ \frac{\text{FP}}{\text{чел. - мес}} \right];$$

$$\text{Качество} = \frac{\text{Ошибки}}{\text{ФункцУказатель}} \left[ \frac{\text{Единиц}}{\text{FP}} \right];$$

$$\text{Удельная стоимость} = \frac{\text{Стоимость}}{\text{ФункцУказатель}} \left[ \frac{\text{Тыс. \$}}{\text{FP}} \right];$$

$$\text{Документированность} = \frac{\text{СтраницДокумента}}{\text{ФункцУказатель}} \left[ \frac{\text{Страниц}}{\text{FP}} \right].$$

- Область применения - коммерческие приложения.

# Переход от FP к LOC оценкам

$$\text{Loc} = \text{FP} * (\text{коэф.языка})$$

Язык программирования	LOC на FP
Assembler	320
C	128
C++	64
Pascal	90
Java	53
VBasic	32
VC++	34
Delphi	29
Perl	21
HTML	15

# Метрики «указателя свойств»

- Features Points' (FP')
- для системного и инженерного ПО, ПО реального времени, встроенного ПО
- добавляется к функциональным указателям метрика “количество алгоритмов”
- Алгоритм - ограниченная подпрограмма вычислений внутри общей программы (обработка прерываний, инвертирование матрицы, расшифровка битовой строки).
- Для систем реального времени  $FP' > FP$  на 25-30 %.

# Указатель свойств

$$FR' = \text{Общее количество} \times (0,65 + 0,01 * Fi)$$

Характеристика	Количество	Сложность	Итог
Внешние вводы		x4	=
Внешние выводы		x5	=
Внешние запросы		x4	=
Внутренние логические файлы		x7	=
Внешние интерфейсные файлы		x7	=
Количество алгоритмов		x3	= Общее количество

# Особенности ФОМ

- Плюсы функционально-ориентированных метрик:
  - Не зависят от языка программирования;
  - Легко вычисляются на любой стадии проекта.
- Минусы функционально-ориентированных метрик:
  - Результаты основаны на субъективных данных;
  - Используются не прямые, а косвенные измерения.

# Оценка на основе LOC и FP метрик

- Цель - предварительная оценка для
  - корректные требования по стоимости и затратам разработки для предъявления заказчику;
  - составления плана проектных работ.
- Варианты использования LOC и FP метрик:
  - для определения размера любого элемента продукта;
  - для метрического базиса фирмы (архив за прошлые проекты).

# COCOMO II

## (Constructive Cost Model)

- Конструктивная модель стоимости
- Б. Боэм, 1995г.
- Состав:
  - модель композиции приложения;
  - модель раннего этапа проектирования;
  - модель этапа пост-архитектуры
- Информация о размере программного обеспечения: LOC, FP (объекты и функции)

# СОСОМО II.

## Модель композиции приложения

- Этап разработки ПО, когда выполнено:
  - макетирование пользовательского интерфейса;
  - оценка производительности;
  - определение степени зрелости технологии.
- Объектный указатель (OP – Object Points)
  - для оценки размера ПО
  - количество экранов (элементов UI),
  - количество отчетов,
  - количество компонентов

# Оценка количества объектных указателей (ОР)

Типы объекта	Количество	Вес			Итого
		Простой	Средний	Сложный	
Экран	-	x1	x2	x3	= _
Отчет	-	x2	x5	x8	= _
3GL компоненты	-			x10	= _
Объектные указатели					= _

# Оценка веса (сложности) формы и отчета

Количество представлений и секций на экранной форме		Количество клиентских и серверных таблиц		
Сложность экрана		< 4, < 2 серверов, < 3 клиентских таблиц	< 8, 2 - 3 сервер., 3 – 5 клиен. таблиц	>= 8, > 3 сервер., > 5 клиен. таблиц
	< 3	Простой	Простой	Средний
	3 - 7	Простой	Средний	Сложный
	> 8	Средний	Сложный	Сложный
Сложность отчета	0 или 1	Простой	Простой	Средний
	2 - 3	Простой	Средний	Сложный
	>= 4	Средний	Сложный	Сложный

# Количество новых объектных указателей

$$NOP = OP \times [(100 - \%REUSE) / 100]$$

%REUSE – процент повторно используемых компонентов

# Затраты на разработку

$$\text{ЗАТРАТЫ} = \text{NOP} / \text{PROD} \text{ [чел.-мес]}$$

NOP - Количество новых объектных указателей

$$\text{NOP} = \text{OP} \times [(100 - \% \text{REUSE}) / 100]$$

%REUSE – процент повторно используемых компонентов

PROD - Оценка скорости разработки

Опытность / возможности разработчика	Зрелость / возможность среды разработки	PROD
Очень низкая	Очень низкая	4
Низкая	Низкая	7
Номинальная	Номинальная	13
Высокая	Высокая	25
Очень высокая	Очень высокая	50

# СОСОМО II. Модель раннего этапа проектирования

- Этап разработки ПО, когда выполнено:
  - стабилизация требования;
  - определение базисной проектной архитектуры.

$$\text{ЗАТРАТЫ} = A \times \text{РАЗМЕР}^B * M_e + \text{ЗАТРАТЫ}_{\text{auto}} [\text{чел.-мес}].$$

A - 2,5 масштабный коэффициент;

B - нелинейная зависимость затрат от размера ПО (в KLOC);

$M_e$  - множитель поправки (характеризует продукт, процесс, персонал);

ЗАТРАТЫ auto - автоматически генерируемый программный код

B – масштабный фактор ( от 1,01 до 1,26 )

$$M_e = \prod_{i=1}^7 EM_i$$

$$B = 1,01 + 0,01 \sum_{i=1}^5 W_i$$

# Масштабные факторы Wi

- Предсказуемость PREC  
Опыт фирмы в разработке ПО данного типа (5 = нет, 0 = максимальный).
- Гибкость разработки FLEX  
Гибкость процесса разработки (5 = точный, строгий; 0 = заказчик определяет цели).
- Разрешение архитектуры /риска RESL  
Степень анализа риска (5 = малое, ~ 20%, 0 = полное (100%)).
- Связность группы TEAM  
Знакомство и согласованная работа разработчиков (5 = трудное взаимодействие, 0 = безукорезненное взаимодействие).
- Зрелость процесса PMAT
- Зрелость в организации по вопросам CMM. Среднее количество Да на вопросы CMM (5 = очень низкий, 0 = сверх высокий).

# Множители затрат Me

- Возможности персонала PERS
  - Надежность и сложность продукта PCPX
  - Требуемое повторное использование RUSE
  - Трудность платформы PDIF
  - Опытность персонала PREX
  - Средства поддержки FCIL
  - График SCED
- 
- Очень низкий = 6, низкий = 5, номинальный = 4, высокий = 3, очень высокий = 2, сверх высокий = 1.
  - На основе оценки назначают коэф. от 0,7 до 1,3

# Затраты на автоматическую генерацию кода

$$\text{ЗАТРАТЫ}_{\text{auto}} = (\text{KALOC} \times (\text{AT} / 100)) / \text{ATPROD}$$

- KALOC — KLOC автоматически генерируемого кода;
- AT — процент автоматически генерируемого кода (от всего кода системы);
- ATPROD — производительность автоматической генерации кода с учетом соединения автоматически сгенерированного кода с кодом, написанным вручную.

# СОСОМО II.

## Модель этапа пост-архитектуры

- Этап разработки ПО, когда выполнено:
  - сформирована архитектура;
  - дальнейшая разработка ПО.

$$\text{ЗАТРАТЫ} = A \times K_{req} \times \text{РАЗМЕР}^B \times M_p + \text{ЗАТРАТЫ}_{\text{auto}} \text{ [чел.-мес]}$$

A - 2,5 масштабный коэффициент;

B - нелинейная зависимость затрат от размера ПО (в KLOC);

M<sub>p</sub> - множитель поправки (обычно от 0,3 до 3);

ЗАТРАТЫ auto - автоматически генерируемый программный код

B – масштабный фактор ( от 1,01 до 1,26 )

K<sub>req</sub> – коэффициент для учета возможных изменений требований

BRAC = % кода, отброшенного из-за изменений требований

$$K_{req} = 1 + (\text{BRAC}/100). \quad M_p = \prod_{i=1}^{17} EM_i \quad B = 1,01 + 0,01 \sum_{i=1}^5 W_i$$

# Оценка размера кода

$$\text{РАЗМЕР} = \text{РАЗМЕР}_{\text{new}} + \text{РАЗМЕР}_{\text{reuse}} \text{ [KLOC]}$$

- РАЗМЕР<sub>new</sub> — размер нового кода;
- РАЗМЕР<sub>reuse</sub> — размер повторно используемого кода.

$$\text{РАЗМЕР}_{\text{reuse}} = \text{KASLOC} \times ((100 - AT) / 100) \times (AA + SU + 0,4 DM + 0,3 CM + 0,3 IM) / 100$$

KASLOC — повторно используемый код, который должен быть модифицирован [KLOC];

AT — процент автоматически генерируемого кода;

DM — процент модифицируемых проектных моделей;

CM — процент модифицируемого программного кода;

IM — процент затрат на интеграцию для подключения повторно используемого ПО;

SU — фактор понимаемости добавляемого ПО ( 10 = хороший объектно-ориентированный код, 50 = сложный неструктурированный код);

AA — фактор стоимости решения о повторно использовании ПО (необходимость тестирования и оценивания), от 0 до 8.

# Множители затрат $M_p$

- Факторы продукта:
  - требуемая надежность ПО — RELY;
  - размер базы данных — DATA;
  - сложность продукта — CPLX;
  - требуемая повторная используемость — RUSE;
  - документирование требований жизненного цикла — DOCU.
- Факторы платформы (виртуальной машины):
  - ограничения времени выполнения — TIME;
  - ограничения оперативной памяти — STOR;
  - изменчивость платформы — PVOL.
- Факторы персонала:
  - возможности аналитика — ACAP;
  - возможности программиста — PCAP;
  - опыт работы с приложением — AEXP;
  - опыт работы с платформой — PEXP;
  - опыт работы с языком и утилитами — LTEX;
  - непрерывность персонала — PCON.
- Факторы проекта:
  - использование программных утилит — TOOL;
  - мультисетевая разработка — SITE;
  - требуемый график разработки — SCED.

# Пример оценки множителя затрат

<b>TOOL</b>	Редактирование, кодирование, отладка 1,24	Простая входная, выходная CASE утилита, малая интеграция 1,12	Базовые утилиты жизненного цикла, умеренная интеграция 1,00	Развитые утилиты жизненного цикла, умеренная интеграция 0,86	Развитые утилиты жизненного цикла, хорошо интегрированы с процессами, методами, повторным использованием 0,72	
<b>SITE</b>	Один телефон, почта 1,25	Индивидуальные телефоны, FAX 1,10	Узкополосный e-mail 1,00	Широкополосные коммуникации 0,92	Широкополосные коммуникации, иногда видеоконференции	Интерактивные мультимедиа 0,78
<b>PCAP</b>	15% 1,37	35% 1,16	55% 1,00	75% 0,87	90% 0,74	

# Оценка стоимости и длительности по СОСОМО II

- Стоимость разработки

**Стоимость = затраты \* раб.коэф.**

- Длительность разработки

$$\text{Длительность (TDEV)} = 3 * \text{затраты}^{0,33 + 0,2 * (B - 1,01)} * \frac{SCEDPr}{100}$$

B – масштабный фактор.

SCEDPr - Процент увеличения или уменьшения номинального графика от 75 до 160 % . Номинальный график = 100 %.

# Особенности СОСОМО II

- Факторы затрат существенно влияют на выходные параметры
- СОСОМО II учитывает большинство реальных ситуаций
- СОСОМО II преобразовывает качественные обоснования в количественные показатели

# Анализ чувствительности программного проекта

- Имеется проект:
  - Размер = 10 KLOC
  - $V = 1,16$  (номинал)
  - $M_p = 1,088$
  - Затраты auto = 0
  - Раб.коэф. (зарплата) = 60 000 руб/месяц.
  - Затраты = 39 [чел-мес]
  - Стоимость = Затраты \* 60000 = 2 340 000 [руб]

$$\overline{2,5 * 10^{1,16} * 1,088 = 39 \text{ [чел - мес]}}$$

# Оценка изменений

- Уменьшаем раб.коэф. до 50 000 р/мес
- В результате:
  - Возможности аналитика: 0,83 -> 1
  - Возможности разработчика: 0,87 -> 1
  - Мр 1,088 -> 1,507
  - Затраты = 54 [чел - мес]
  - Стоимость = 2 700 000  
(увеличилась на 360 000)

# Сокращение затрат

- Уменьшаем стоимость до 1 800 000 (Затраты до 30)
- Вариант А – уменьшение размера:  
 $2,5 \times (\text{Нов.размер}^{1,16}) = 30 / 1,088$   
 $\text{Нов.размер} = (30 / 1,088 \times 2,5)^{1/1,16}$   
Нов.размер  $\sim 7,9$

Сократить лишние функции

# Сокращение затрат

- Вариант Б – изменение факторов затрат на 23% (т.к. Затраты 39 -> 30)
  - Затраты /  $M_p$  = ЗатратыНов /  $M_{pНов}$
  - $M_{pНов}$  = ЗатратыНов x  $M_p$  / Затраты
  - Например, выбрали коммуникацию:  
 $M_{ip}$  = 1,1 (e-mail)  
 $M_{iНов}$  = 0,84 (широкополосная коммуникация - очное выступление на конференциях)
- Вопрос возможности и связанных факторов
- Корреляция факторов затрат между собой и с раб.коэф.