

Оценка трудозатрат разработки программной компоненты

И. Я. Гольфанд, П. С. Хлебутин

План не предсказывает будущего. Он лишь позволяет должным образом реагировать на неизвестное будущее.

Аноним

В статье дается описание методики расчета трудозатрат для разработки программной компоненты, используемой в практической деятельности по созданию программного обеспечения. Методика обладает свойствами самообучения, что делает ее привлекательной как для оценки модификации существующих, так и при создании новых компонент.

Введение

Проект по созданию программного обеспечения включает в себя множество работ, одной из которых является разработка программных компонент. Интерес именно к стадии разработки программной компоненты обусловлен тем, что это наиболее длительная и трудоемкая стадия проекта, которая оказывает значительное влияние на другие стадии. Оценка трудозатрат по разработке программной компоненты может служить основой для предварительного планирования всего проекта.

Создание программного обеспечения имеет свои особенности. Одной из таких особенностей является изменение требований в процессе работы. Эта особенность оказывает наибольшее влияние именно на стадии разработки. Изменение требований заставляет корректировать план работ, где основой изменений являются работы по разработке программной компоненты.

1. Постановка задачи

Процесс создания программной компоненты начинается с этапа планирования. В основу оценки берется опыт предыдущей работы. Разрабатывается архитектура системы, создаются подробные требования к си-

стеме. Требование может состоять не только в создании нового свойства программной компоненты, но и в изменении существующих свойств. Каждое требование к системе оценивается, получая временное и стоимостное выражение оценки. В результате, полная стоимость всех требований не должна превышать бюджет, выделенный на этап создания программной компоненты.

После начала работы и выполнения некоторого количества требований, становятся известны реальные временные сроки выполнения запланированных работ. Реальное время выполнения работ может отличаться от запланированного времени вследствие ошибок в объеме работ или мощности используемых ресурсов (другие причины изменения времени здесь не учитываем).

В процессе работы требуется регулярно получать прогноз времени выполнения и стоимости работ, который позволил бы принимать своевременные меры по снижению рисков проекта.

Таким образом, задача оценки трудозатрат состоит в нахождении достоверных оценок длительности и стоимости разработки, удовлетворяющих следующим требованиям:

1. Оценка трудозатрат по созданию программной компоненты должна опираться на предшествующий опыт оценки аналогичных компонент.
2. Оценка должна оценивать как новую работу, так и работу по внесению изменений и переработку алгоритмов работы компоненты.
3. Оценка должна учитывать квалификацию сотрудников.
4. Оценка должна учитывать расходы на весь цикл создания программной компоненты, включая руководство процессом и этап тестирования.
5. Необходимо иметь оценку точности.

2. Существующие решения

Существует множество решений по оценке трудозатрат программного обеспечения. Наиболее распространенные:

1. Оценка трудозатрат по количеству строчек кода. Она появилась в середине 1980 гг. и получила название COCOMO (*Constructive Cost Model*), см. [1, с. 380]. В основе этой методики лежит оценка объема работ по количеству строчек кода (*Lines Of Code* — LOC). Базовая формула оценки трудозатрат COCOMO:

$$E = a \cdot \text{KSLOC}^b \cdot \text{EAF}, \quad (1)$$

где E — трудозатраты, выраженные в человеко-месяцах; a и b — константы, определенные на основе регрессионного анализа, в зависимости от проекта. KSLOC — количество тысяч строк кода (под строчками кода

SLOC понимаются логические строки кода, а именно строки в понимании используемого языка программирования, без учета комментариев); *EAF* (*Effort Adjustment Factor*) — фактор корректировки трудозатрат;

Фактор корректировки трудозатрат *EAF* увеличивает или уменьшает трудозатраты в зависимости от факторов среды:

$$EAF = \prod_{i=0}^n C_i, \quad (2)$$

где C_i — один из факторов среды.

Этот способ оценки используется на начальной стадии проекта, для оценки всего проекта в целом. Для небольших проектов и небольших по объему работ его использование затруднительно.

2. Оценка трудозатрат по функциональным точкам. Метод оценки трудозатрат по функциональным точкам основывается на предположении, что существует связь между языком программирования и количеством строчек кода приходящимся на одну функцию. В процессе оценивания, происходит оценка количества функций, приходящихся на реализацию одного функционального требования, с учетом его сложности. В итоге, количество полученных функциональных точек пересчитывается в условные строчки кода или непосредственно в трудозатраты (см. [1, с. 331]).

Этот способ не учитывает возможности ресурсов для выполнения работ. Одна и та же работа может выполняться разными людьми за разное время, в зависимости от квалификации работника.

3. Экспертная оценка или трех точечное оценивание (см. [2, с. 138]). Метод основывается на том, что эксперты дают три оценки затрат для каждого функционального требования системы: наилучшую, наихудшую и наиболее вероятную. Общая оценка определяется по формуле:

$$E = \sum_i \frac{W_i + 4M_i + B_i}{6}, \quad (3)$$

где B_i — наилучшая оценка, W_i — наихудшая оценка, M_i — наиболее вероятная оценка. Исходя из предположения, что оценки имеют нормальное распределение, определяется псевдостандартное отклонение:

$$\sigma = \sqrt{\sum_i \left(\frac{W_i - B_i}{2} \right)^2}. \quad (4)$$

Эта формула позволяет получить диапазон изменения объема трудозатрат для реализации набора функциональных требований.

Этот способ оценки требует привлечения экспертов, поэтому его сложно использовать в новых разработках. Он субъективен, трудоемок и не учитывает возможностей имеющихся ресурсов. Для корректировки

начальных планов в процессе работы требуется повторное привлечение экспертов.

Таким образом каждое из рассмотренных решений обладает рядом недостатков. В описываемом далее решении делается попытка частично устранить отмеченные недостатки.

3. Предлагаемое решение

3.1. Планирование

За основу оценки трудозатрат по разработке программной компоненты возьмем оценку трудозатрат по количеству строчек кода по формуле (1) и (2). Будем оценивать каждое функциональное требование отдельно, при этом примем значение константы b равной 1. Предположим, что объем работ, зависит от вида работы, а время его выполнения зависит от мощности ресурса, необходимого для его выполнения. Тогда формула (1) преобразуется к виду:

$$E = \sum_i \sum_j \frac{\text{KSLOC}(i, j)}{W_j} \cdot \text{EAF}, \quad (5)$$

где W_j — мощность ресурса, необходимого для выполнения j -ого вида работы; $\text{KSLOC}(i, j)$ — количество строк кода j -ого вида работ для реализации i -го требования.

Для определения параметра KSLOC и W необходимо провести анализ аналогичной программной компоненты, имеющей схожую архитектуру и язык программирования. В результате анализа необходимо выделить три вида кода:

- Код, описывающий интерфейс системы. Обычно это код описывающий сам интерфейс. Для компонент работающих в среде Интернет — это HTML-код. Обозначим его как SLOC_R .
- Код, обрабатывающий элементы интерфейса и связывающий его с другими частями системы. Обычно это классы обработки ошибок ввода данных, записи данных в базу и т. п. Обозначим его как SLOC_I .
- Код, обеспечивающий определенную логику работы системы. Обычно, это классы, описывающие свойства объектов системы, взаимосвязь между бизнес функциями. Обозначим его как SLOC_P .

Для расчета KSLOC можно воспользоваться формулой:

$$\text{KSLOC} = \frac{\text{SLOC}_P \cdot K_p + \text{SLOC}_I \cdot K_i + \text{SLOC}_R \cdot K_r}{1000}, \quad (6)$$

где K_p — фактор изменения логики работы системы; K_i — фактор изменения обработки интерфейсных элементов системы; K_r — фактор изменения интерфейса системы; Факторы K подбираются опытным путем.

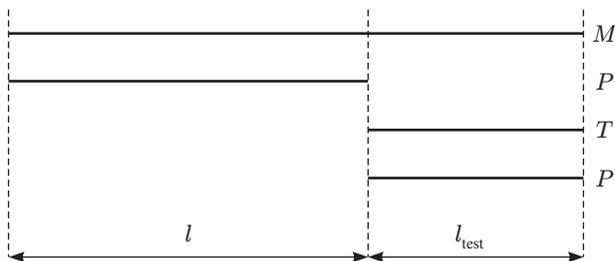


Рис. 1

Например, для изменений менее 20 % фактор изменений может быть равен 0,2; для 50 % изменений — 0,7; для вновь создаваемых компонент — 1,0.

Мощность W определяется опытным путем, исходя из опыта работы над предыдущими компонентами.

Определение факторов среды EAF.

Фактор учета технологии разработки (ФУТР). Под этим фактором учитывается увеличение количества трудозатрат в результате того, что в процесс разработки вовлекается разное количество сотрудников.

Сам процесс можно представить следующим образом. Процесс кодирования длится в течение времени l (см. рис. 1).

В это время задействованы M менеджеров проекта и P программистов. Процесс тестирования начинается сразу после кодирования, длится он в течение времени l_{test} , и в нем участвуют M менеджеров проекта, T тестеров и P программистов. Время тестирования является величиной зависимой от времени кодирования так, что $l_{test}/l = K_{test}$ (5).

Таким образом,

$$\text{ФУТР} = \frac{M \cdot (1 + K_{test}) + T \cdot K_{test}}{P} + 1 + K_{test}. \quad (7)$$

При этом длительность разработки

$$L = l \cdot (1 + K_{test}). \quad (8)$$

Фактор учета сложности и опыта разработки (ФУСОР). Фактор ФУСОР определен эмпирически и состоит из следующих значений:

- надо поправить то, что есть (1,00);
- уже делали раньше, есть опыт (1,10);
- есть опыт и не видно трудностей (1,30);
- опыта нет, но есть помощь (2,00);
- нет ни опыта, ни помощи (4,00).

При использовании этого фактора необходимо учесть, что фактор зависит от конкретной личности или группы. Если в одном проекте есть повторяющиеся работы, тогда одна из них может быть учтена с фактором

«Нет ни опыта, ни помощи», а для другой, аналогичной работы, он должен быть «Уже делали раньше, есть опыт».

Таким образом, длительность кодирования и тестирования можно вычислить по формуле:

$$T = \sum_i \sum_j \frac{KSLOC(i, j)}{W_j} \cdot \PhiУСОР(i) \cdot \frac{1}{P} \cdot (1 + K_{\text{test}}). \quad (9)$$

Трудозатраты на кодирования и тестирования можно вычислить по формуле:

$$E = \sum_i \sum_j \frac{KSLOC(i, j)}{W_j} \cdot \PhiУСОР(i) \cdot \PhiУТР. \quad (10)$$

3.2. Уточнение основных показателей

Предположим, что перед началом выполнения работ, было выполнено планирование трудозатрат по методике описанной выше. Через некоторое время после начала работ, можно получить реальные значения времени выполнения каждой из работ. Используем эти данные для коррекции исходных значений SLOC, SLOCi, SLOCr и мощности W .

Плановый расчет времени, для одного вида работ, описывается линейной зависимостью от объема выполняемой работы и мощности ресурсов:

$$V = tR, \quad (11)$$

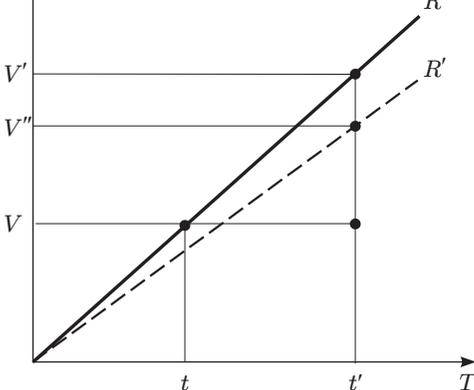


Рис. 2. График зависимости длительности разработки от объема работ и мощности ресурса

где V — объем работ, измеряемый в KSLOC; $R = EAF/W$ — величина, обратно пропорциональная мощности ресурса W и учитывающая факторы корректировки EAF.

Первоначальное значение, вычисленное по (11) для одной из работ имеет точку с координатами (t, V) . После выполнения работы получим время t' . Полученная ошибка может быть следствием двух причин. Первая причина заключается в том, что была допущена ошибка в объеме работ. В этом случае, объем работ должен был бы быть V' (см. рис. 2). Вторая причина заключается в том, что была допущена ошибка в значении R , зависимой от мощности ресурса и факторов EAF. В этом случае, объем

работ должен был остаться равным V . Таким образом, выполненный объем работ V'' находится в промежутке между V и V' .

Введем коэффициенты, учитывающие ошибки в объеме и в мощности K_V и K_R :

$$K_v \cdot K_R = \frac{t'}{t}. \quad (12)$$

Выберем новую точку, находящуюся на отрезке между V и V' , соответствующую времени t' . У новой точки будут координаты (t', V'') , определяемые формулой:

$$t' = t', \quad V'' = V \cdot K_v. \quad (13)$$

Для корректировки параметров уравнения (11), воспользуемся методом оптимизации способом наименьших квадратов. Для этого посчитаем следующие значения:

$$S_{tt} = \sum_{i=1}^n (t'_i - \bar{t})^2. \quad (14)$$

$$S_{tV} = \sum_{i=1}^n ((t'_i - \bar{t})(V_i - \bar{V})), \quad (15)$$

где n — количество сделанных наблюдений, $i = 1, \dots, n$, а

$$\bar{V} = \frac{1}{n} \sum_{i=1}^n V_i, \quad \bar{t} = \frac{1}{n} \sum_{i=1}^n t'_i$$

— средние значения.

Новый скорректированный коэффициент

$$R' = \frac{S_{tt}}{S_{tV}}. \quad (16)$$

Для того чтобы узнать, каков диапазон изменения параметра R для среднего времени выполнения одной работы, построим доверительные интервалы:

$$\sigma_R = \sqrt{\frac{\sum_{i=1}^n (R_i - R'_i)^2}{n - 2}}, \quad (17)$$

$$\Delta R = \pm t_{\varepsilon/2} \cdot \sigma_R \sqrt{\frac{1}{n}}, \quad (18)$$

где $t_{\varepsilon/2}$ — табличное значение распределения Стьюдента с $(n - 2)$ степенями свободы.

Делая расчеты по каждому из видов работ отдельно, можно уточнить предположения об объемах и мощности ресурсов сделанных на этапе планирования.

Уточненное значение мощности с учетом усредненных коэффициентов EAF для выбранного вида работ будет

$$W' = \frac{1}{R' \pm \Delta R}. \quad (19)$$

Уточненное значение базовых объемов для выбранного вида работ можно получить по формуле:

$$\text{SLOC}' = \frac{\sum t'_i R}{\sum t_i R'} \cdot \text{SLOC}. \quad (20)$$

4. Примеры

Пример 1. Определить стоимость работы по созданию макета интернет-системы на основе существующей системы. Работа заключается в изменении дизайна системы (около 50 форм), модификации и создании макета 30 форм. В проекте участвуют один менеджер проекта и три разработчика. Тестирование не требуется.

В результате анализа существующей системы было выявлены следующие закономерности:

| Описание работы | SLOC | | |
|------------------------|-------------------------|----------------------------|------------------|
| | Объем кода логики SLOCP | Объем кода обработки SLOCI | Объем кода SLOCR |
| Средний модуль проекта | 4 000 | 250 | 150 |

Было также определено, что мощность выполнения работ при создании интернет-систем аналогичного вида составляет $W = 2,5$.

Количество строк кода для изменения дизайна интерфейса (из расчета изменения менее 20 % полей формы) и форм макета (создание 100 %-х полей формы):

$$\begin{aligned} \text{SLOC} = & 50 \cdot (4000 \cdot 0 + 250 \cdot 0 + 150 \cdot 0,2) + \\ & + 30 \cdot (4000 \cdot 0 + 250 \cdot 0 + 150 \cdot 1) = 6000. \end{aligned} \quad (21)$$

Фактор учета технологии разработки будет равен:

$$\text{ФУТР} = \frac{1 \cdot (1 + 0) + 0 \cdot 0}{3} + 1 + 0 = 1,333.$$

Фактор учета сложности и опыта примем равным 1,1 (делали раньше и есть опыт).

Таким образом, длительность работы, исходя из расчета работы трех человек, составит:

$$T = \frac{6}{2,5 \cdot 3} \cdot (1 + 0) \cdot 1,1 = 0,88 \text{ [мес]}$$

Полные трудозатраты с учетом тестирования составят:

$$E = \frac{6}{2,5} \cdot 1,333 \cdot 1,1 = 3,52 \text{ [чел/мес]} .$$

Пример 2. В результате выполнения работ, были получены данные представленные в табл. 1, 2. Показатель ΔR вычислен из прогноза на среднюю продолжительность работ. Необходимо провести анализ и уточнить базовые показатели для выполненных видов работ 1, 2, 3 табл. 1, 2.

Анализ работы № 1 табл. 1, 2

Из данных видно, что задержка выполнения работ составляет $1,545 - 1,294 = 0,251$ месяца. Необходимо внести коррекцию равную $1,545/1,294 = 1,194$. Возможны следующие виды коррекций:

1. Коррекция по мощности $0,484/0,4 = 1,21$.
2. Коррекция по объему $2\,463/(2\,500) = 0,985$.

При таких значениях, коррекция по времени составит искомый коэффициент $1,21 \cdot 0,985 = 1,19$.

При корректировке можно пренебречь коррекцией по объему, в запас по времени.

Таблица 1

| Виды работ | SLOCP | SLOCI | SLOCR | R | Время план. | Время реал. |
|------------|-------|-------|-------|-----|-------------|-------------|
| № 1 | 2 500 | 200 | 150 | 0,4 | 1,294 | 1,545 |
| № 2 | 2 500 | 250 | 200 | 0,4 | 1,28 | 0,818 |
| № 3 | 1 000 | 50 | 50 | 0,4 | 0,46 | 0,92 |

Таблица 2

| Виды работ | SLOCP' | SLOCI' | SLOCR' | R' | $\pm \Delta R$ |
|------------|--------|--------|--------|-------|----------------------|
| № 1 | 2 463 | 197 | 148 | 0,484 | 0,05 |
| № 2 | 1 643 | 164 | 131 | 0,389 | 0,05 |
| № 3 | 1 667 | 83 | 83 | 0,48 | $4,0 \cdot 10^{-08}$ |

Анализ работы № 2 табл. 1, 2

Из данных видно, что перевыполнение работ составляет $0,818 - 1,28 = -0,462$ месяца. Необходимо внести коррекцию равную $0,818/1,28 = 0,639$. Возможны следующие виды коррекций:

1. Коррекция по мощности $0,389/0,4 = 0,9725$.
2. Коррекция по объему $1\ 643/(2\ 500) = 0,657$.

При таких значениях, коррекция по времени составит искомый коэффициент $0,9\ 725 \cdot 0,657 = 0,639$.

При корректировке можно пренебречь коррекцией по мощности в запас по времени.

Анализ работы № 3 табл. 1, 2

Из данных видно, что задержка выполнения работ составляет $0,92 - 0,46 = 0,46$ месяца. Необходимо внести коррекцию равную $0,92/0,46 = 2,0$. Возможны следующие виды коррекций:

1. Коррекция по мощности $0,48/0,4 = 1,2$.
2. Коррекция по объему $1\ 667/(1\ 000) = 1,667$.

При таких значениях, коррекция по времени составит искомый коэффициент $1,2 \cdot 1,667 = 2,0$.

Для принятия решения необходимо проанализировать реальный объем выполненных работ. Если не подтвердится предположение о превышении объема сделанной работы в 1,7 раза, то в сочетании с низким значением показателя ΔR это может говорить о необходимости коррекции по мощности в два раза.

5. Выводы

Данная методика оценки трудозатрат может быть использована в практической деятельности по созданию программного обеспечения, при анализе текущего состояния проекта, прогнозировании сроков его окончания и стоимости. В отличие от других методик она, в известной степени, обладает свойством самообучения. Это позволяет с одной стороны, достаточно легко выбирать базовые значения для новых работ, не имеющих аналогов реализации, а с другой стороны позволяет накапливать и использовать опыт предыдущей работы.

Наибольший эффект такая методика может принести для работ, выполняемых этапами. В этих случаях уточнение базовых показателей может быть привязано к завершению этапа работ. Следующий этап можно перепланировать с учетом результатов предыдущего этапа. В результате перепланирования могут быть выявлены проблемы, решение которых возможно лишь административно-управленческими методами, изменением количества работ или изменением бюджета проекта.

Литература

1. *Фатрени Р. Т., Шафер Д. Ф., Шафер Л. И.* Управление программными проектами: достижение оптимального качества при минимуме затрат. М.: Издательский дом «Вильямс», 2003.
2. *Кармайкл Э., Хейвуд Д.* Быстрая и качественная разработка программного обеспечения. М.: Издательский дом «Вильямс», 2003.