

nexign students

#quality

#attributes

#Metrics

Метрики оценки качества ПО

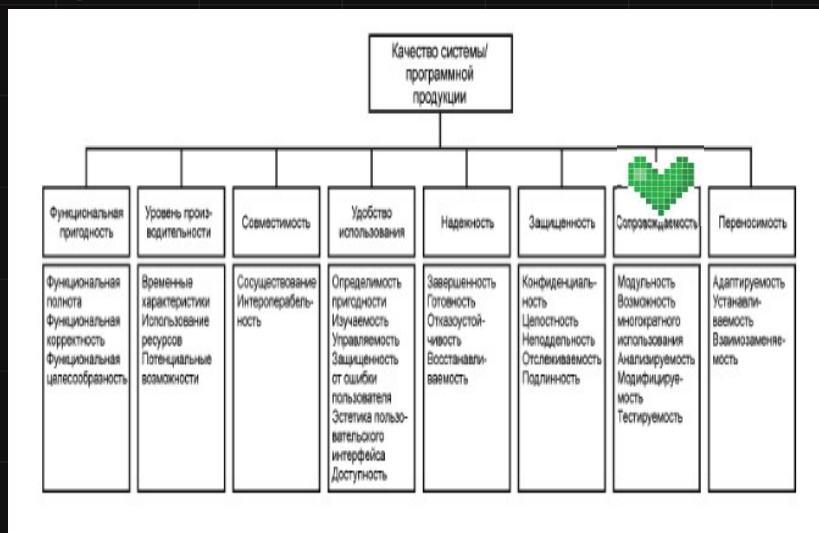
(практика #1)

Дмитрий Юшкевич



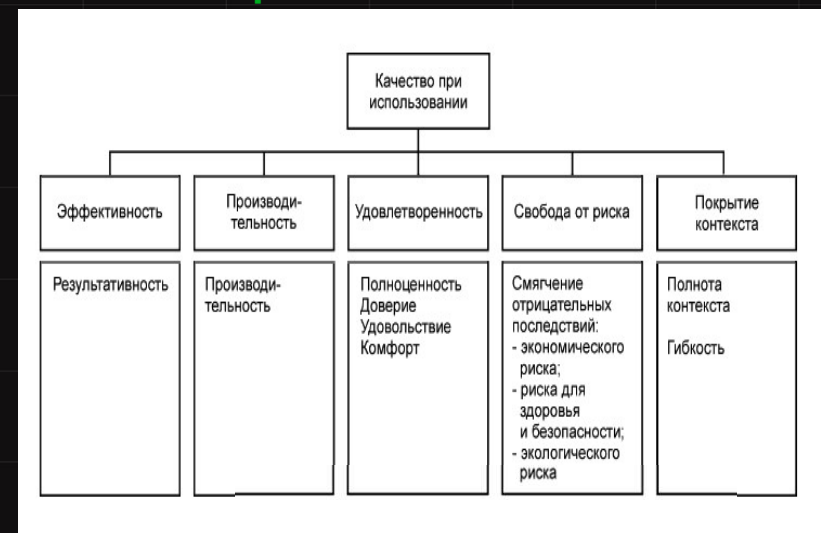
1. Характеристики качества ПО

Внутреннее + внешнее качество




Метрика качества программного обеспечения (ПО) - это мера, позволяющая **количественно** оценить определенное свойство или характеристику ПО

Качество при использовании




2. Классификация метрик качества ПО


По модели качества:

-  **Внутреннее качество** – проявляется в процессе разработки и в других промежуточных этапах жизненного цикла ПО
- **Внешнее качество** – задается требованиями заказчика в спецификациях и отражается в характеристиках конечного продукта
- **Качество при использовании** – определяется результативностью достижения потребностей пользователей в различных контекстах – того качества, которое ощущается пользователями при конкретных сценариях работы ПО


По видимости:

-  **внутренние метрики**, описывающие свойства, доступные только разработчикам.
- **внешние метрики**, описывающие свойства ПО, доступные для конечного пользователя;

По объекту измерения:

-  **метрики продукта**, оценивающие характеристики ПО или артефактов производства
- **метрики использования**, оценивающие свойства ПО при работе с конечным пользователем.
- **метрики процессов**, оценивающие характеристики процессов разработки; .

По способу измерения:

-  **динамические** (требующие выполнения программы);
- **статические**, получаемые анализом проекта ПО, кода или документации.

3. Пример оценки «тестируемости» продукта

Качество ПО

Атрибут \ свойство
Характеристика
Метрика
Шкала
Метод получения
Мера
Множество допустимых значения



Сопровождаемость

Тестируемость (сложность проверки продукта)
Сложность кода продукта
В условных единицах (баллах)
Расчет гибридной метрики сложности кода
= ??? – предстоит вычислить
От ??? до ??? в зависимости от границ слагаемых гибридной метрики



4. Статические метрики качества продукта



1. Количественные метрики.
2. Метрики сложности потока управления программы.
3. Метрики сложности потока управления данными.
4. Объектно-ориентированные метрики.
5. Гибридные метрики.

5. Гибридные метрики

Класс гибридных метрик представляет собой взвешенную сумму более простых метрик. Peter KokoI, 1989, предложил одну из моделей объединения метрик.

Метрика:

$$HM = \frac{M + R_1 M(M_1) + \dots + R_n M(M_n)}{1 + R_1 + \dots + R_n},$$

Где:

- M – основная метрика;
- $M(M_i)$ – функции, связанные с другими представляющими интерес метриками;
- R_i – специально подобранные под конкретный программный продукт коэффициенты (веса).

Интерпретация:

Если взять за основу распределение затрат на разработку программного обеспечения: 17% – архитектура программы, 8% – кодирование, 25% – тестирование, 50% – поддержка, то можно предложить метрику, где:

- C – сложность программы на стадии проектирования, например ОО – Лоренц, Oper Complexity [проектирование];
- D – сложность программы по Холстеду [кодирование];
- M – цикломатическая сложность Маккейб [тестирование];
- Q – метрика Чепина сложность данных [поддержка].

Пример:

$$HM = 0.17C + 0.08D + 0.25M + 0.5Q$$

6. Учебные продукты

Продукт MAIN:

```
main()
{
    int a, b, c, avg;
    scanf("%d %d %d", &a, &b,
    &c);
    avg = (a+b+c)/3;
    printf("avg = %d", avg);
}
```

Продукт BIN:

```
int binary (int c[], int n, int val)
{
    int a,b,m;
    for (a=0, b=n-1; a<=b;)
    {
        m = (a+b)/2;
        if (c[m] == val)
            return m;
        if (c[m] > val)
            b = m-1;
        else
            a = m+1;
    }
    return -1;
}
```

its

6.1. Сложность, Холстед

Отражают реализацию, зависят от языков, не зависят от платформ, вычисляются статически - из кода

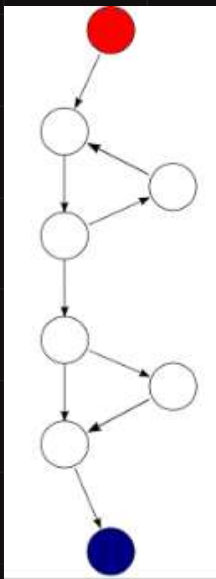
- η_1 = количество различных операторов
- η_2 = количество различных операндов
- N_1 = общее количество операторов
- N_2 = общее количество операндов



• Уровень сложности: $D = \frac{\eta_1}{2} \times \frac{N_2}{\eta_2}$

6.2. Сложность, Маккейб

Показывают структурную сложность кода на основе операторов в исходном коде путем построения графов переходов от одного оператора к другому



$$M = E - N + 2P,$$

где

- E = количество ребер графа.
- N = количество узлов графа.
- P = количество подключенных компонентов.



7.1. Сложность, Лоренц

Сложность операции ОС (Operation Complexity) м.б. вычислена суммированием оценок с весовыми коэффициентами из таблицы.

$$C = \sum_{i=1}^n (M_i * V_i)$$

Где: C – сложность операции; M_i – вес характеристики i -ой группы из n существующих; V_i – количество действий i -ой группы

Действие, V_i	Вес, M_i
Определение (описание) перем.-параметра	0,3
Определение (описание) временной перем.	0,5
Присваивание значения	0,5
Вложенное выражение	0,5
Сообщение без параметров	1
Арифметическая операция	2
Сообщение с параметрами	3
Вызов стандартной функции интерфейса (API)	5
Вызов пользовательской функции (простой вызов)	7

7.2. Сложность, Чепин

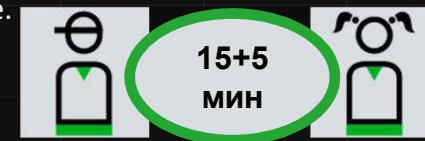
Делит множество переменных на группы в зависимости от их использования в программе

$$Q = a_1 * P + a_2 * M + a_3 * C + a_4 * T$$

Где:

- a_1, a_2, a_3, a_4 - весовые коэффициенты (1, 2, 3 и 0.5)
- P – переменные для расчетов и для обеспечения вывода
- M – переменные модифицируемые или создаваемые внутри кода
- C - управляющие переменные.
- T – не используемые переменные.

** переменные., выполняющие несколько функций, учитываются в каждой группе.



8. Гибридная метрика + выводы

Метрика	Граница	MAIN	BIN
- C – Operation Complexity [проектирование]	< 65 (Лоренц)	14	19,9
- D – сложность программы по Холстеду [кодирование]	< 30 (эталон)	12,85	27,9
- M – цикломатическая сложность Маккейб [тестирование]	< 20 (Маккейб)	1	5
- Q – метрика Чепина сложность данных [поддержка]	< 30 (эталон)	5	25


$$HM = ?C + ?D + ?M + ?Q$$


ВЫВОДЫ



9. Пример оценки «тестируемости» продукта

Качество ПО

Атрибут \ свойство
Характеристика
Метрика
Шкала
Метод получения
Мера
Множество допустимых значения

Сопровождаемость

Тестируемость (сложность проверки продукта)
Сложность кода продукта
В у. е.
Расчет гибридной метрики сложности кода
= 6 (MAIN) и 19 (BIN)
$C < 65 / D < 30 / M < 20 / Q < 30$: BIN приблизился к сложности по D и Q (25 у.е.)

Thank you!

