

Table of Contents

1. 目的	4
2. 简介	4
3. 使用用例点来估计测试规模和工作量	4
4. Case Study	6

1. 目的

本文的目的是解释用用例点的方法来做测试规模和测试人力估计。

使用用例来做估计是一个在软件工业中被普遍接受的方法。当用例越复杂，那么在设计，开发，测试和实现上面花费的时间也会越多。这是一个常识。

业界有很多参考资料是关于用用例点来估计开发人力，但是用用例点来估计测试活动的信息确很少。本文提供了一些用用例点来进行测试估计的步骤。

2. 简介

估计测试的工作量是一个很重要的方面，因为这直接关系到项目的成本。众所周知，有三个主要的方面会影响估计：时间，成本和质量。一个好的估计将会帮助项目按时交付。如果估计不正确，那么将会导致延迟交付，成本增加或者不正确的结果。

也有很多的方法用例对测试活动进行估计，比如测试点，功能点等。用例点方法在估计软件工作量方面也是一个广受欢迎的方法。

3. 使用用例点来估计测试规模和工作量

3.1. 测试规模估计

不管是哪种类型的估计，首先第一个活动就是要计算需要做的活动的规模。在用例点方法中，决定一个项目规模主要有 4 个方面：

1. 参与者权重 (Actor Weight)
2. 用例权重 (Use Case Weight)
3. 技术因子 (Technical Factors)
4. 环境因素 (Environmental Factors)

翻译者注：Actor 是用例里面的一个重要概念。主要是定义系统用户在与系统交互时候可扮演的一组相关角色。可以是个人或者外部系统。

3.1.1 参与者权重

参与者权重已经被定义为 3 类：Simple, Medium 和 Complex。下面的表描述了分类标准和权重值。

参与者权重		
加权类型	加权值	标准
Simple	1	通过定义好的 API 表示另外的系统
Medium	2	可能是通过一些协议比如 TCP/IP 交互的另外的系统或者通过基于结构的一个文本交互的一个人
Complex	3	可能是一个通过 GUI 交互的人

3.1.2 用例权重

计算用例权重的步骤和计算参与者权重类似。所有的用例需要被划分为 Simple, Medium 和 Complex。最终的权重是所有分类用例的权重值的总和。

用例权重		
加权类型	加权值	标准
Simple	1	3 个或者 3 个一下的事务
Medium	2	4-7 个事务
Complex	3	超过 7 个事务

3.1.3 未经调整的用例点 (UUCP)

$UUCP = \text{参与者权重} + \text{用例权重}$

3.1.4 技术复杂度因素

有 7 个技术因素会影响整个项目的估计。下面的表列出了从测试角度出发的因素和权重。

技术因素		
S.N.	描述	分配的值
1	测试工具	2
2	文档输入	2
3	测试件重用	1
4	分布式系统	2
5	性能目标	1
6	安全特征	1
7	接口复杂度	1

技术复杂度因素的计算公式:

$$TCF = C1+C2(\text{Total TF})$$

[对于每个用例点来说, TCF 对于用例点的影响有个范围, 这个范围从 0.6 (-40%) 到 1.3 (+30%)。为了达到这个范围, C2 的系数需要为了测试工作量而进行如下的修正。

$$C2 \text{ for testing} = (1.3 - 0.6)/50 = 0.014$$

-> 50 是 TCF 对于测试来说的一个最大值]

[C1 = Constant with value 0.6 常量

C2 = Constant with value 0.014

Total TF = Total Technical Factor]

3.1.5 环境复杂度因素

有 7 个环境因素会影响整个项目的估计。下面的表列出了从测试角度出发的因素和权重。

环境因素		
S.N.	描述	分配的值
1	测试环境	2
2	测试数据	1
3	应用知识	1
4	测试 Lead 的能力	0.5
5	激励	1
6	需求稳定性	2
7	兼职人员	-1

技术复杂度因素的计算公式:

$$ECF = C1+C2(\text{Total EF})$$

[对于每个用例点来说, ECF 对于用例点的影响有个范围, 这个范围从 0.0425 (-57.5%) 到 1.4 (+40%)。为了达到这个范围, C2 的系数需要为了测试工作量而进行如下的修正。

$$C2 \text{ for testing} = (1.4 - 0.0425)/37.5 = 0.0362$$

-> 37.5 是 ECF 对于测试来说的一个最大值]

[C1 = Constant with value 1.4 常量

C2 = Constant with value 0.0362

Total EF = Total Environment Factor]

3.1.6 计算最终用例点 (UCP)

$$UCP = UUCP*TCF*ECF$$

3.2. 工作量估计

3.2.1 转换因素

一旦项目的规模被计算出来, 总的规模需要用乘以一个转化因子而转化为工作量。这个转化因子 (Effort/Size) 被定义为测试每个用例点需要的总体时间。

这个转化因子能够被得到通过反向工程技术, 比如通过把项目历史数据放入估计模板。对于基于 Java 的项目来说, 这个值是 20 (hrs)。

3.2.2 计算最终工作量

$$\text{Final Effort} = UCP*\text{Conversion factor}$$

4. Case Study

4.1. 参与者

S.N.	参与者名称	权重	因子值
1	Actor 1	Simple	1
2	Actor 2	Medium	2
3	Actor 3	Complex	3
4	Actor 4	Simple	1
		Total	7

4.2. 用例权重

S.N.	用例描述	权重	因子值
1	UC 1	Simple	5
2	UC 2	Medium	10
3	UC 3	Simple	5
4	UC 4	Complex	15
		Total	35

4.3. 没有调整过的用例点

$$UUCP = 7 + 35 = 42$$

4.4. 技术复杂度因子计算

技术因素				
S.N.	描述	权重	复杂度	计算结果
1	测试工具	2	1	2
2	文档输入	2	2	4
3	测试件重用	1	3	3
4	分布式系统	2	4	8
5	性能目标	1	5	5
6	安全特征	1	1	1
7	接口复杂度	1	3	3
			Total	26

$$\begin{aligned} TCF &= 0.6 + (0.01) * \text{Total TF} \\ &= 0.6 + (0.014) * 26 \\ &= 0.96 \end{aligned}$$

4.5. 技术复杂度因子计算

技术因素				
S.N.	描述	权重	复杂度	计算结果
1	测试环境	2	3	6
2	测试数据	1	1	1
3	应用知识	1	5	5
4	测试 Lead 的能力	0.5	1	0.5
5	激励	1	5	5
6	需求稳定性	2	2	4
7	兼职人员	-1	2	-2
			Total	19.5

$$\begin{aligned} ECF &= 1.14 + (-0.03) * \text{Total EF} \\ &= 1.14 + (-0.0362) * 19.5 \end{aligned}$$

www.51testing.com

= 0.69

4.6. 计算最终用例点

$$\begin{aligned} \text{UCP} &= \text{UUCP} * \text{TCF} * \text{ECF} \\ &= 42 * 0.96 * 0.69 \\ &= 28.82 \end{aligned}$$

4.7. 计算最终工作量

$$\text{Final Effort (Hrs)} = \text{UCP} * 20 = 28.82 * 20 = 576.4$$