

一种增量式软件开发工时度量方法研究

王世安

(广东松山职业技术学院 计算机系, 广东 韶关 512126)

摘要: 为了定量地管理增量式软件的迭代开发, 进一步做好增量式软件开发资源分配和成本估算, 需要度量开发工时. 每个迭代阶段增量需求是以一定的概率实现的, 把这个概率作为开发任务完成率的度量, 提出增量式软件开发迭代周期的度量方法, 进而获得对整个软件项目开发工时的预测. 通过实例对方法进行了验证.

关键词: 增量式软件; 迭代周期; 任务完成率; 工时

中图分类号: TP311.5

文献标识码: A

文章编号: 1000-7180(2011)05-0201-03

Research on Measurement of Incremental Software Development Man—Hour

WANG Shi-an

(Department of Computer, Guangdong Songshan Polytechnic College, Shaoguan 512126, China)

Abstract: To manage iterative development of incremental software so that resource allocation and cost estimating can be reasonably finished in incremental software development, the man—hour of development must be measured. Incremental requirements in each phase are implemented according to stated probability which is regarded as task accomplishment rate, a measurement method of iteration periods in incremental software development is provided. Furthermore, the man—hour of development for whole software project can be forecasted. The method is demonstrated with an example.

Key words: incremental software; iteration periods; task accomplishment rate; man—hour

1 引言

在软件开发实践中, 当客户在软件设计或软件实现之后提出新的需求时, 常常用增量式设计方法设计新提出的软件需求^[1-2]. 增量式软件开发的基本思想是在不可能一次就开发出完全满足用户要求的复杂软件系统时, 通过需要迭代地执行核心的软件活动, 增量地实现软件系统. 通过迭代过程, 不但获得了软件系统的增量, 尽快得到反馈, 而且不断降低项目风险.

增量式软件开发是以极限编程和特征驱动开发为代表的敏捷开发方法以及广泛应用的统一软件开发过程的关键组成^[3]. 然而, 这些方法都没有指出迭代的次数和每个增量的大小. 这就造成增量式软件迭代过程的随意性, 使得软件过程难以计划、评估、

控制和管理. 且相关文献对增量式迭代开发方法的研究较多, 缺少对增量迭代开发项目预测和度量方法的关注, 未见关于迭代开发效率、开发周期、迭代工作量预计的方法等有关论述. 为此, 研究增量式软件开发工时具有重要意义.

在增量式软件开发实践中, 把用户需求分解成具体的、明确的增量需求, 这些增量需求能够被估计出开发时间, 采用基于历史数据和项目规模进行估计的方法, 来计算增量式软件开发的迭代周期, 进而获得开发工时.

2 增量需求的划分

在增量式软件开发的每个迭代阶段, 由于执行过程中的变更、用户需求的复杂性、对用户需求的工作量估计过低、对团队的开发速度估计过快、新的需

求与已经实现的功能难以顺利集成、客户代表不能有效沟通等原因,用户需求的实现情况难以精确度量^[4-5],但可以将用户需求划分为不同阶段的增量需求,每个阶段的增量需求的顺利实现是以一定的概率 p 进行的,则另一部分以 $1-p$ 的概率延迟到下一迭代阶段.而 p 值可以视为增量需求的完成率.

在新的开发迭代阶段,总的增量需求由新的增量需求和上一个阶段遗留下来的增量需求组成.对于上一阶段未完成的增量需求,由于上一个阶段已经做了一些工作,多数不再是一个完整的增量需求^[6],将把他们折合成具有规定需求大小的增量,统称为上阶段遗留增量需求.这样,可以搜集各阶段的已实现增量需求和遗留增量需求,当项目进行到某个阶段时,估计出 p 值,并进一步估计出迭代周期.

增量需求应该易于理解,在规定的迭代时间内能实现^[7].这依赖于与客户的充分的沟通,合理的问卷,敏锐的观察,正确的书写等能力.增量需求的质量直接影响 p 值的大小.

3 增量需求的完成率度量

3.1 前提条件

为保证每个阶段的增量需求的顺利实现是以一定的概率 p 进行,应该满足如下条件:

(1) 开发人员在开发过程中保持不变,不增加和减少人员,不更换人员,保证具有相同的开发速度;

(2) 项目管理人员有一定的项目经验,对项目的初始规模(增量需求)把握较为准确;

(3) 每个增量需求的大小基本相同(对于大小不同的用户需求,可以通过对用户需求分解或组合来实现),即完成每个增量需求的工作量大致相同.

3.2 完成率度量

假设增量式软件开发项目进行到在第 k 个迭代阶段初期,用 c_k 表示第 $k-1$ 阶段的遗留增量需求数,用 n_k 表示第 k 阶段新引入的增量需求数.显然,在第一个迭代阶段, $c_1 = 0$. 用 p 来表示增量需求的完成率,则在第二个迭代阶段初,上一阶段遗留增量需求数 c_2 与 p 和 n_1 的关系如下:

$$c_2 = n_1(1-p) \quad (1)$$

同样,在项目完成第 $k-1$ 次迭代后,进入到第 k 次迭代,这时要去估计 P 值,可知从第 1 个迭代阶段到第 $k-1$ 个阶段遗留下来的增量需求数 c_i ($i = 1, 2, \dots, k$) 和每次加入的增量需求数 n_i ($i = 1, 2, \dots, k$) 得到:

$$q = 1 - p = \frac{\sum_{i=2}^k c_i}{\sum_{i=2}^{k-1} c_i + \sum_{j=1}^{k-1} n_j} \quad (2)$$

通过递推可以得到:

$$\sum_{i=2}^k c_i = \sum_{i_1=1}^{k-1} n_{i_1} q + \sum_{i_2=1}^{k-2} n_{i_2} q^2 + \dots + (n_1 + n_2) q^{k-2} + n_1 q^{k-1} \quad (3)$$

在每个阶段,上阶段的遗留增量需求数和本阶段加入的增量需求数都是可以直接从项目实施过程中获得的,因此当项目发生几次迭代以后,可以根据前面若干次(大于等于 2 次,因为第一次迭代不存在上阶段遗留增量,即 $c_1 = 0$) 迭代获得的实际数据,通过式(2) 计算 p 的估计值.

4 迭代周期

为了简化计算,要求迭代时间不允许改变,无论任务是否完成,到了规定的结束日期,迭代任务必须暂时停止.这对项目经理、开发人员和用户来说,在考察项目进度,度量团队的开发效率和迭代周期、工作量方面十分有用.

在做完项目开发的可行性分析和初期需求分析后,根据开发经验,可以大致给出增量式软件项目迭代的估计增量需求数,设为 S . 设总的迭代次数为 m 次,则在第 m 次迭代后,将不再增加新的用户需求,迭代会结束,这时 $n_{m+1} = 0$,但第 m 次迭代后仍然会有遗留的增量需求:

$$c_{m+1} = (c_m + n_m)(1-p) \quad (4)$$

要在 $m+1$ 阶段迭代最后完成,实际迭代次数可以为 $m+1$,也可以视这一部分遗留的增量需求为在一个延时的迭代阶段完成.

根据前面的公式,可知平均每个阶段引入新的增量需求数为:

$$\bar{n} = (\sum_{i=1}^{k-1} n_k) / (k-1) \quad (5)$$

根据式(2) 有:

$$\bar{n} = (\frac{p}{1-p} \sum_{i=2}^{k-1} c_i + c_k \frac{1}{1-p}) / (k-1) \quad (6)$$

则 S 个增量需求要迭代的可能次数为:

$$m = S / \bar{n} \quad (7)$$

将式(6) 代入式(7):

$$m = \frac{S(k-1)}{(\frac{p}{1-p} \sum_{i=2}^{k-1} c_i + c_k \frac{1}{1-p})} \quad (8)$$

可以看出迭代次数 m 可以通过 p 以及每个阶段

遗留的增量需求数 c_i 来估计.

5 应用实例

使用一个实际的项目,其增量需求总数 $S = 135$,表1说明项目刚进行到第9个迭代阶段,前8次迭代已经完成.

表1 项目增量迭代情况

迭代阶段	c_i	n_i
1	0	19
2	3	17
3	5	13
4	6	10
5	4	11
6	2	12
7	3	8
8	4	10
9	2	3

用前面若干次迭代所得数据去估计 p ,并预测总的迭代次数 m .

如图1所示,根据前3次的迭代,应用式(2),解得 $p = 0.6$,根据前4次的迭代,解得 $p = 0.7544$,不难看出, k 越大,获得的数据越多, p 值越趋于稳定,但 k 取得太大则失去预测的意义.

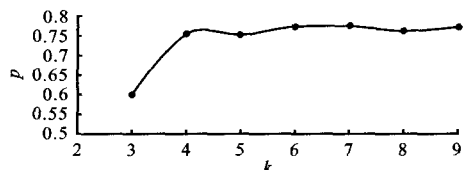


图1 p 随迭代次数的变化情况

取 $p = 0.77$.应用式(8) 计算得:

$$m = 12.3859$$

即对于初步估计需求总数为 135 的增量式软件项目,使用表1提供的经验数据,在开发人员不变的情况下,采用文中方法大约需要经历 13 个迭代阶段才能把用户需求完全加入到最终软件系统中,但由于第 13 次迭代后还有遗留增量需求,可以认为总的迭代次数为 14 次.

完成了对增量式项目迭代周期和迭代次数的度量,很容易进一步做出整个增量式项目开发工时预测^[8],并做出资源分配和成本估算.

6 结束语

文中提供了一个简单的增量式软件开发工时度量方法,其用于计算的数据在项目实施过程中很容易获得^[9],使得用迭代早期开发数据来估计整个项目增量迭代开发周期更容易实现,能够得到较为准确的数据,对进一步做好项目资源分配和成本估算有重要参考价值.其中 p 值也可以作为遗留再造项目团队开发效率评价的参考数据. p 值的估计充分利用了再造迭代开发过程中用户需求被处理情况的数据,因而具有较强的代表性.

参考文献:

- [1] Lucia A D, Ferrucci F, Tortora G, et al. Emerging Methods, Technologies, and Process Management in Software Engineering[M]. NJ: Wiley—IEEE Computer Society, 2008.
- [2] 张天宇. 中小型软件开发质量控制研究[J]. 微电子学与计算机, 2007(6): 88—90.
- [3] Bosch J. Software product families: towards compositionality[C]// Fundamental Approaches to Software Engineering, UNCS 4422, Berlin Heidelberg: Springer—Verlag, 2007: 1—10.
- [4] Rajlich V. Changing the paradigm of software engineering[J]. Communications of the ACM, 2006, 49(8): 67—70.
- [5] 黄全舟. 统一软件开发过程与建模技术研究[J]. 微电子学与计算机, 2005(7): 27—30.
- [6] Purao s, Vaishnavi V. Product metrics for object-oriented systems[J], ACM Computing Surveys, 2003, 35(2): 191—221.
- [7] 张广泉, 戎政, 陆明. 一种面向复用的增量软件开发过程框架[J]. 计算机科学, 2006, 33(10): 252—255.
- [8] Highsmith J, Cockburn A. Agile software development [J]. Computer, 2001, 34(9): 120—122.
- [9] Hetrick W A, Krueger C W, Moore J G. Incremental return on incremental investment: Engenio/S transition to software product line practice [C]// Companion to the 21st ACM Sigplan conference on Object-oriented programming systems, languages, and applications, Portland, Oregon USA: ACM, 2006: 798—804.

作者简介:

王世安 男, (1971—), 硕士, 副教授. 研究方向为算法设计与分析、计算机应用与软件等.