

Les points de fonction et la réutilisation des composants logiciels

Tran-Cao De, Lévesque Ghislain

Résumé : La méthode des points de fonction a été introduite par Allan Albrecht en 1979. Pendant plus de 25 ans, cette méthode n'a cessé de développer, d'être appliquée par un grand nombre d'experts et d'utilisateurs. Plusieurs rapports ont confirmé le rôle de la mesure des points de fonction dans la gestion des projets de logiciels. Les points de fonction sont considérés comme un indice de base pour l'estimation d'effort et du coût de développement ou de maintenance des logiciels. Ils sont aussi une mesure fondamentale pour juger de la productivité du développement ou de la maintenance du logiciel et comparer la productivité de projets de logiciel différents.

Les points de fonction d'un logiciel sont une mesure dérivée de la fonctionnalité du logiciel décrite dans ses spécifications. Cette mesure correspond à la taille du logiciel à développer ou à maintenir. Cette approche a été prouvée expérimentalement et sa validité a été démontrée dans le contexte de « nouveau développement » ou du « tout construit ». Cependant, elle rencontre des problèmes avec la réutilisation de codes sources (bibliothèque) et la réutilisation des composants en accès libre (*open sources*). En fait, avec la réutilisation, on peut construire des logiciels de qualité, très complexes pendant un court délai. Comment peut-on adapter les points de fonction à cette situation?

Cet article propose une façon de mesurer les points de fonction des logiciels développés en faisant appel à la réutilisation des composants.

Mot clés : composant, estimation d'effort, estimation de coût, réutilisation, point de fonction, productivité.

I. INTRODUCTION

En génie logiciel, on croit qu'une bonne gestion de projet du logiciel doit être basée sur la connaissance et la compréhension des processus en cause. Celles-ci sont basées sur des informations collectées régulièrement et sur l'analyse de ces informations. Ces informations sont constituées de données de base et de la collecte de données relatives à différentes mesures. La taille d'un projet logiciel est la plus importante mesure de la collection de données. Elle permet, en l'utilisant comme variable indépendante, d'estimer l'effort et le coût du logiciel à développer ou à entretenir. Elle est aussi très importante pour la gestion du contrat d'un projet de logiciel.

Traditionnellement, le nombre de lignes de code (LOC) ou le nombre de mille lignes de code (KLOC) fut utilisé comme une mesure directe de la taille du logiciel. Cependant, cette mesure est fortement dépendante des langages de programmation, des styles d'écrire du code source. Cela rend plus difficile la comparaison de projets différents, ainsi que la

comparaison de la productivité des équipes de développement du logiciel.

L'approche des points de Fonction fut introduite par Allan Albrecht en 1979 [2] pour mesurer la taille fonctionnelle du logiciel. L'idée de celui-ci est encore aujourd'hui valide et applicable au grand nombre de projets du logiciel actifs en ce monde. Il existe plusieurs propositions sophistiquées pour améliorer la méthode de mesure. Ces propositions se concentrent sur « comment compter de façon précise les points de fonction ». C'est-à-dire, comment on peut définir et mesurer les points de fonction du logiciel qui reflète bien l'effort de développement ou de maintenance du logiciel. Albrecht et Gaffney mettent l'emphase sur la signification de points de fonction comme suit :

« The thesis of this work is that the amount of function to be provided by the application can be estimated from the itemization of the major components data to be used or provided by it. Furthermore, this estimation of function should be correlated to both the amount of LOC to be developed and the development effort needed. » [3]

Il est nécessaire de souligner que les points de fonction d'un logiciel ne sont qu'une mesure grossière du nombre de lignes de code (LOC) parce que celles-ci sont dépendantes du langage de programmation. La recherche [3] montre aussi une conversion simple entre un point de fonction (PF) et LOC, par exemple : 1 PF équivaut à 100 LOC en Fortran ou à 20 LOC dans un langage de 4^{ème} génération.

Aujourd'hui, avec les nouvelles techniques et les nouveaux paradigmes de programmation, notamment les techniques orienté-objet, de paquetage et de réutilisation, on peut réduire significativement le temps nécessaire pour livrer un produit logiciel. Il apparaît que l'effort pour développer un point de fonction ou le coût unitaire d'un point de fonction est de plus en plus bas. Notamment, avec la réutilisation des codes sources disponibles (bibliothèque) et les logiciels d'accès libres (*open sources*), il est possible de construire rapidement des applications très complexes en quelques jours. Comment mesurer la taille du logiciel et fournir des estimations appropriées de l'effort ou du coût dans le contexte de réutilisation ?

II. MESURES DE LA TAILLE FONCTIONNELLE DU LOGICIEL

De nombreux efforts ont été consacrés à consolider la méthode des points de fonction originale d'Albrecht [2,3] ou à établir des nouvelles méthodes inspirées de l'approche des points de fonction. En fait, on peut trouver: