Innovations dans la mesure de la taille fonctionnelle du logiciel

SERGE OLIGNY, ALAIN ABRAN, DENIS SAINT-PIERRE ET JEAN-MARC DESHARNAIS

Résumé : Il a été mentionné à plusieurs reprises dans la littérature que la méthode des pPoints de fFonction (IFPUG), pour la mesure de la taille fonctionnelle des logiciels, ne produit pas des résultats adéquats pour les logiciels de type temps réel et de type embarqué. En 1997, a été publiée une extension à cette méthode ayant pour été publiée dans le but d'améliorer le processus de mesure de ces types de logiciel. Cette extension est connue comme la version 1 des Points de Fonction Étendus (PFE). Par ailleurs, au cours de la dernière année, ISO a publié une nouvelle norme (ISO 14143-1) portant sur les méta- règles et précisant les contraintes à observer pour les méthodes de mesure de la taille fonctionnelle des logiciels, tandis qu'en parallèle un groupe d'experts internationaux a développé les règles pour le contenu descriptif du concept de fonctionnalité auxquelauquels ces contraintes s'appliqueraient. La publication de cette norme ISO de même que les travaux menés par le CoOmmon Software Measurement International Consortium (COSMIC) ont donc fortement influencé la conception de la deuxième version de la méthode PFE tout en la simplifiant et en la rendant plus générique, donc applicable à un plus grand nombre de types de logiciels. Le présent articleCette communication décrit quelques unes des améliorations significatives de la version 2 de la méthode PFE, dorénavant désignée par l'expression PFE-COSMIC.

Mots clés : Mesure du logiciel, taille fonctionnelle, points de fonction, points de fonction étendus, PFE.

1. INTRODUCTION

La mesure de la taille fonctionnelle du logiciel a été introduite, il y a plus de vingt20 ans déjà, par Albrecht [1]. Conçue dans le contexte de l'informatique de gestion, et cette technique a été principalement utilisée dans ce contexte. Plusieurs auteurs ont notamment souligné son inaptitude à représentercr adéquatement la taille fonctionnelle des logiciels de type temps réel ou embarqué [2, 4, 5-, 6, 7, 9, 10, 11, 16]. Bien que quelques alternatives aient été proposées, du milieu des années 80 au milieu des années 90, pour résoudre ce problème, aucune d'entre elles ne semble avoir gagner suffisamment de reconnaissance de la part des praticiens pour être utilisée sur une base régulière à travers un grand nombre d'organisations dans plusieurs pays.

D'autre part, en 1997, la méthode de mesure des points de fonction étendus (PFE) [15] a été introduite pour améliorer la mesure de la taille fonctionnelle des logiciels de types temps réel et embarqués [12, 13, 14].

Au cours des deux dernières années, quelques nouveaux facteurs importants dans l'état de l'art ont permis d'apporter des améliorations significatives à la première version publiée en 1997. La nouvelles version 2,0 est la première méthode de mesure de la taille fonc-

tionnelle du logiciel sur le marché à tenir compte de ces nouvelles exigences : la nouvelle norme ISO [8] sur les mesures de taille fonctionnelle et l'implantation des principes de conception-design fonctionnelle du regroupement Common Software Measurement International Consortium (COSMIC) [3].

Le présent article décrit quelques- unes des améliorations clés proposées par la version 2. Cette version est désignée par l'expression PFE-COSMIC puisqu'elle respecte les principes de conceptiondesign de ce groupe. Le chapitre 2 présente un sommaire des nouveaux besoins de l'industrie. Le chapitre 3 présente les principales améliorations introduites dans la version 2 de la méthode. Enfin, le chapitre 4 replace ces améliorations dans le contexte de l'évolution des mesures de la taille fonctionnelle du logiciel.

2. Pourquoi améliorer ?

Quoiqu'elle n'aite été introduite qu'en 1997, la méthode de mesure des points de fonction étendus a déjà été appliquée non seulement aux seuls logiciels de type temps réel ou embarqué, comme le prévoyait sa conceptionon design, mais également à une variété de logiciels de type technique, de type système et même à des logiciels de type gestion. Ces applications de la méthode ont fait apparaître de nouveaux besoins :

a) Clarifier et rendre plus visibles les principales phases du processus de mesure, tant pour faciliter la tâche des praticiens que pour rendre explicite la façon dont la nouvelle méthode respectait les contraintes définies par la norme ISO [8].

b) Affiner le concept de *frontière*, tel qu'appliqué à la mesure du logiciel, de manière à pouvoir étendre l'applicabilité de la méthode non seulement aux logiciels d'application mais également aux logiciels de support à ceux-ci, qui sont partie intégrante de l'environnement d'opération au sein d'un projet donné ;

c) Simplifier l'ensemble des composants fonctionnels de base utilisés pour mesurer la taille fonctionnelle du logiciel ;

d) Augmenter la souplesse de la méthode de mesure en lui permettant d'offrir des agrégats de la taille à différents niveaux de fonctionnalité (scalability).

Ces nouveaux besoins ont donc engendré de nouvelles exigences pour améliorer la conceptioe de la méthode PFE et ont ainsi guidé l'élaboration de la version 2 de cette dernière. Celle-ci sera dorénavant connue comme la méthode PFE-COSMIC puisqu'elle respecte l'ensemble des exigences des experts du groupe COSMIC.

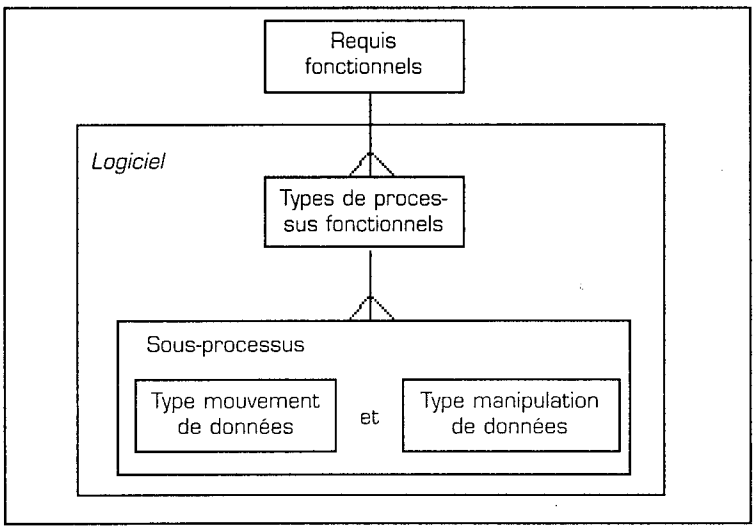
3. La version 2 de la méthode PFE-COSMIC

3.1 Le modèle de fonctionnalité de COSMIC

Le modèle générique de la fonctionnalité du logiciel proposé par le groupe COSMIC est illustré à la Figure 1. Selon ce modèle, les requis fonctionnels exprimés par les utilisateurs du logiciels sont matérialisés par un ensemble de processus fonctionnels. Chacun de ces processus est constitué d'un ensemble ordonné de sous-processus dont la tâche est soit de déplacer des données, soit de les manipuler (transformer).

Le modèle générique de la fonctionnalité du logiciel proposé par la méthode de mesure PFE-COSMIC distingue quatre types de sous-

Figure 1 : Modèle générique de la fonctionnalité du logiciel - COSMIC



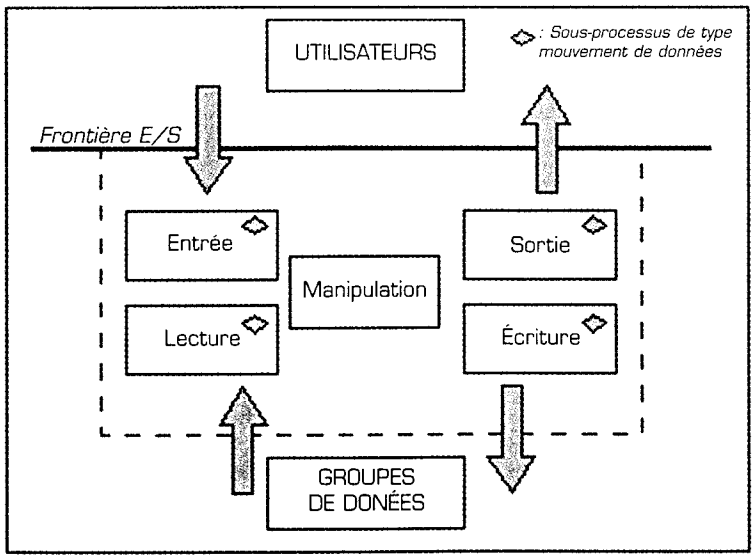


Figure 2 : Les types de sous-processus reconnus par COSMIC

processus déplaçant des données : les entrées, les sorties, les lectures et les écritures. Un sous-processus appartenant à l'un de ces quatre types

déplace des données appartenant à un et un seul groupe de données. Une entrée déplace des données de l'extérieur de la frontière vers l'intérieur de celle-ci, du côté entrée/sortie ; une sortie déplace des données de l'intérieur de la frontière vers l'extérieur de celle-ci du côté entrée/sortie ; une lecture et une écriture déplacent des données à travers les entrepôts de données. Ces relations sont illustrées à la Figure 2.

3.2 Un modèle du processus de mesure

En tant que méthode de mesure de la taille fonctionnelle du logiciel, (FSM), tel que définie par ISO dans [8], la méthode PFE-COSMIC vise à mesurer la taille fonctionnelle du point de vue des requis fonctionnels. En pratique, la méthode de mesure PFE-COSMIC consiste à appliquer un ensemble de règles et de procédures à un logiciel, le résultat de cette application étant un nombre représentant la taille fonctionnelle du logiciel mesuré.

La méthode mesure la taille fonctionnelle du logiciel sous l'angle des requis fonctionnels. Elle est conçue de manière à être indépendante des décisions d'implantation dont les résultats font partie intégrante des artefacts utilisés pour extraire ces requis fonctionnels. Cette caractéristique de la méthode est atteinte en appliquant les règles de mesure sur un modèle du logiciel sur lequel les requis fonctionnels extraits des artefacts des activités de développement des logiciels sont mis en correspondance. Ce modèle du processus de mesure est illustré par la Figure 3.

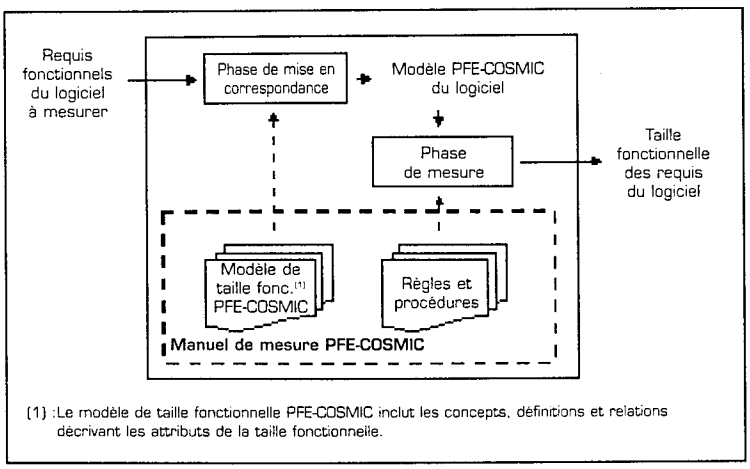


Figure 3 : Modèle du processus de mesure de la méthode COSMIC-PFE

Selon ce modèle, préalablement à l'application des règles et procédures de mesure, le logiciel à mesurer doit être mis en correspondance avec un modèle générique de la fonctionnalité (le modèle COSMIC). Ce dernier capture les concepts, définitions et relations (la structure fonctionnelle, Figures 1 et 2) nécessaires à la mesure de la taille fonctionnelle du logiciel.

3.3 Un modèle de contexte des pièces du logiciel

Un autre aspect clé de la mesure de la taille fonctionnelle du logiciel réside dans l'identification adéquate des fonctionnalités qui font partie du logiciel à mesurer et, de façon complémentaire, dans l'identification des fonctionnalités qui font partie de l'environnement d'exploitation du logiciel mesuré.

Selon la façon dont ces requis fonctionnels sont alloués, le logiciel résultant peut être composé de plusieurs pièces de logiciel. Bien que toutes ces pièces s'échangent des données, elles ne fonctionnent pas toutes au même niveau d'abstraction fonctionnel. Le modèle de contexte du logiciel, présenté à la Figure 4, reconnaît ce fait et la méthode de mesure offre des règles permettant d'identifier les différentes couches de logiciel.

Par exemple, les requis fonctionnels illustrés à la Figure 4 sont alloués à trois pièces de logiciel distinctes. Chacune de ces pièces échange des données avec d'autres selon une organisation spécifique : une des pièces constitue le logiciel d'application et échange des données avec les utilisateurs et avec une seconde pièce faisant partie du niveau « système d'exploitation ». À son tour, cette seconde pièce échange des données avec une troisième pièce qui réside au niveau « pilotes de périphériques ». Cette dernière échange des données directement avec le matériel de support. Le modèle de contexte proposé par la méthode PFE-COSMIC associe chaque niveau fonctionnel à une couche distincte. Chacune de ces couches possède une frontière propre à partir de laquelle les utilisateurs de la couche sont identifiés. La taille fonctionnelle du logiciel, telle que décrite à travers les requis fonctionnels, est donc répartie en trois pièces, chacune recevant une portion des requis fonctionnels.

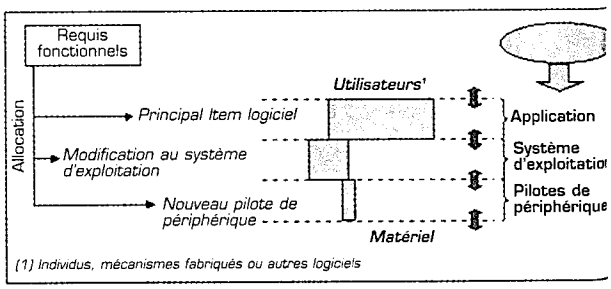


Figure 4 : Modèle de contexte des pièces du logiciel - COSMIC

La méthode PFE-COSMIC définit quatre concepts clés à partir de ce modèle de contexte :

1er concept - notion de requis fonctionnels

Les objectifs du logiciel peuvent être décrits par un ensemble fini de requis. Parmi ceux-ci, la portion décrivant la nature des fonctions fournies est

désignée par l'expression « requis fonctionnels ». C'est à travers la perspective exclusive offerte par cette portion des requis que la taille fonctionnelle du logiciel est mesurée. La portion des requis décrivant comment cette fonctionnalité sera implantée n'est pas considérée aux fins de la mesure de la taille fonctionnelle du logiciel.

2e concept - notion de couche
Le logiciel à mesurer peut être partitionné en une ou plusieurs couches de manière à ce que chacune d'ellese ces couches soit exploitée à un niveau d'abstraction fonctionnelle distinct au sein de l'environnement d'exploitation du logiciel. Il existe une relation entre chacun des niveaux d'abstraction fonctionnelle identifiés, sur la base des données qu'ils s'échangent entre eux. Chaque couche contient la fonctionnalité utile à une couche cliente et utilise la fonctionnalité offerte par une couche subordonnée. Une des couches identifiées interagit avec les utilisateurs externes par l'intermédiaire du matériel d'entrée/sortie et une autre interagit avec le matériel de stockage à travers les pilotes de ces périphériques.

3e concept - notion de frontière
Au sein de chacune des couches identifiées, la fonctionnalité du logiciel à mesurer peut être clairement distinguée, à l'aide d'une frontière, de la fonctionnalité offerte par les autres logiciels exploités sur la même couche. De plus, une frontière implicite existe entre chacune des couches identifiées. La frontière du logiciel mesuré consiste conséquemment en un ensemble de critères, liés à la perception spécifique offerte par les requis fonctionnels, permettant de distinguer sans ambiguïtés la fonctionnalité faisant partie du logiciel mesuré de la fonctionnalité attribuée à l'environnement d'exploitation auquel appartient le logiciel mesuré. Conventionnellement les usagers d'un logiciel sont localisés à l'extérieur de cette frontière.

4e concept - notion d'utilisateurs
Au sein de chaque couche de logiciel, il est possible d'identifier un ou plusieurs utilisateurs, c'est-à-dire des entités bénéficiant de la fonctionnalité offerte par cette couche de logiciel. Par définition, ces utilisateurs peuvent être des individus, des dispositifs fabriqués ou d'autres logiciels. De plus, par définition, les logiciels résidant sur les couches adjacentes à la couche mesurée sont considérés comme des utilisateurs de la fonctionnalité offerte par celle-ci.

En utilisant les concepts et les définitions de la méthode de mesure PFE-COSMIC, les requis fonctionnels identifiés à travers les artefacts produits par les activités de développement du logiciel sont mis en correspondance avec le modèle générique de la fonctionnalité du logiciel et le modèle de contexte des pièces du logiciel, telle que définie par la méthode de mesure PFE-COSMIC. Une fois instancié, ce modèle contiendra tous les éléments requis pour en mesurer la taille fonctionnelle tout en masquant les informations non pertinentes à cet exercice.

Les règles et procédures de mesure proposées par la méthode PFE-COSMIC sont par la suite appliquées à une instance du modèle générique du logiciel de manière à produire une valeur numérique représentant la taille fonctionnelle du logiciel.

Par conséquent, deux phases distinctes et consécutives sont nécessaires pour mesurer la taille fonctionnelle du logiciel, telles qu'illustrées à la Figure 3 :
- *la mise en correspondance* des requis fonctionnels extrait des artefacts produits par les processus du génie logiciel avec les éléments du modèle générique du logiciel ;
- *la mesure* d'éléments spécifiques de ce modèle du logiciel.

3.4 Un système de mesure

La méthode de mesure PFE-COSMIC comprend un système de mesure composé d'un principe de mesure, d'un ensemble de composants fonctionnels de base (BFC) et d'un étalon de mesure.

Principe de mesure : La phase de mesure de la méthode PFE-COSMIC utilise comme intrant une instance du modèle générique du logiciel et, à l'aide d'un ensemble de règles et de procédures, produit une valeur numérique dont l'ordre de grandeur est directement proportionnel à la taille fonctionnelle du modèle. Cette valeur est produite selon le principe suivant :

> *La taille fonctionnelle d'un logiciel est directement proportionnelle à la quantité de sous- processus déplaçant des données qui y sont identifiées.*

Par convention, la signification de la valeur numérique ainsi produite est étendue et représente également la taille fonctionnelle du logiciel lui-même.

Deux éléments supplémentaires caractérisent l'ensemble des règles et procédures gouvernant la production de cette valeur : un ensemble défini de composants fonctionnels de base (BFC) constituant les arguments de la fonction de mesure et un étalon de mesure définissant l'unité de mesure de la taille fonctionnelle (1 unité de taille fonctionnelle COSMIC ou 1 CFSU).

Composants fonctionnels de base : La version 2 de la méthode de mesure n'utilise que quatre types de composants fonctionnels de base (BFC) : l'entrée, la sortie, la lecture et l'écriture. Les sous-processus de manipulation de données ne sont pas utilisés à titre de composants fonctionnels de base. Dans la version 2 de la méthode de mesure, il est posé que ces quatre composants fonctionnels de base constituent une approximation acceptable pour une large gamme de logiciels et que la fonctionnalité spécifique de la manipulation des données est déjà représentée par ces quatre composants fonctionnels de base.

Étalon de mesure : L'étalon de mesure, c'est-à-dire 1 CFSU (COSMIC Functional Size Unit), est défini par convention comme équivalent à un mouvement de données au niveau du sous-processus.

La méthode de mesure PFE-COSMIC ne prétend pas mesurer tous les aspects de la taille du logiciel. Les dimensions qui diffèrent du concept de mouvement de groupe de données ne sont pas prises en compte, par exemple. Un débat constructif en la matière nécessiterait au préalable une définition communément acceptable des autres dimensions au sein du concept ambigu de « «taille » tel qu'applicable au logiciel. De telles définitions font, encore à ce jour, l'objet de recherche de consensus.

3.5 Agrégation des résultats

De par la nature des composants fonctionnels de base (BFC), la méthode de mesure PFE-COSMIC offre l'agrégation des résultats par l'intermédiaire d'une fonction d'agrégation. Cette fonction consiste en l'addition arithmétique de la taille des éléments constituants ; la taille d'un processus fonctionnel étant équivalente à la somme des tailles de ses sous-processus et la taille d'une couche étant équivalente à la somme des tailles de ses processus.

Il est à noter que la plus petite taille fonctionnelle théorique est de deux CFSU puisque tout logiciel montre au moins une entrée ou lecture et une sortie ou écriture. De plus, il n'existe pas de limite théorique à la taille fonctionnelle d'un logiciel et, notamment, il n'existe pas de limite théorique à la taille fonctionnelle d'un processus fonctionnel.

4. Conclusion

La volonté de documenter explicitement les concepts de mesure proposés par la méthode est à la base de plusieurs des améliorations apportées par la version 2 de la méthode PFE-COSMIC. Cette volonté a engendrée une approche permettant de distinguer clairement, d'une part, les concepts du modèles de mesure d'une part et, d'autre part, les règles et les procédures proposées pour les appliquer. La méthode de mesure y gagne en flexibilité en permettant ainsi à ses utilisateurs de saisir rapidement le but des règles et, conséquemment, de les adapter au contexte de mesure propre à leur organisation.

Les exigences présentées au chapitre 2 ont été essentiellement remplies en raffinant le processus de mesure lui-même et en y appliquant le modèle générique de la fonctionnalité tel que défini par COSMIC. Le processus de mesure résultant a été expliqué au chapitre 3, incluant l'enrichissements au modèle de logiciel utilisé par la méthode de mesure, la simplification de l'ensemble des composants fonctionnels de base (BFC) et l'agrégation du résultat à différents niveaux fonctionnels.

En résumé, les améliorations introduites dans la version 2 de la méthode de mesure PFE-COSMIC ont pour but de supporter les praticiens de la mesure du logiciel en offrant une meilleure applicabilité par l'intermédiaire a) d'un cadre métrologique plus rigoureux et b) de concepts explicitement définis permettant aux praticiens de mesurer de manière plus efficace. Des mises à l'essai sont présentement en cours pour déterminer le niveau d'atteinte de ces objectifs lors d'utilisations dans des contextes industriels très variés.

5. Remerciements

Les auteurs remercient les membres du groupe COSMIC [3], notamment Charles Symons, Pam Morris et Grant Rule, pour leurs précieux commentaires et nombreuses suggestions dans le but d'améliorer la version 1 de la méthode de mesure PFE et de donner naissance à la version 2 de cette méthode.

Les auteurs sont membres du Laboratoire de recherche en gestion des logiciels de l'UQAM. Ce Laboratoire est financé par l'intermédiaire d'un partenariat avec Bell Canada. Des fonds supplémentaires sont fournis par le Conseil National de Recherche en Sciences et Génie du Canada.

6. Références

[1] A. J. Albrecht : *Measuring application development productivity* ; Proceedings of the IBM Applications Development Symposium, Monterey, Californiea, 1979.

[2] S. D. Conte, V. Y. Shen et H. E. Dunsmore : *Software engineering metrics and models ;* Benjamin Cummins Publishing, 1986, 396 pages.

[3] Voir www.cosmicon.com pour de plus amples informations.

[4] S. Galea : *The Boeing Company: 3D Function Point Extensions, V. 2.0, Release 1.0* ; Boeing Information and Support Services, Research and Technology Software Engineering, juin 1995.

[5] R. B. Grady : *Practical software metrics for project management and process improvement* ; Prentice Hall, New-Jersey, 1992, 270 pages.

[6] B. Hetzel : M*aking software measurement work* ; QEB Publishing Group, 1993, 290 pages.

[7] D. C. Ince : *History and industrial applications* ; in N. E. Fenton, Software metrics: A rigourous approach, Chapman & Hall, UK, 1991, 337 pages.

[8] ISO/IEC 14143-1 : *Information technology - Software measurement - Functional size measurement - Definition of concepts*, octobre 1997.

[9] C. Jones : *A short history of function points and feature points* ; Software Productivity Research Inc., Cambridge, Mass., 1988.

[10] C. Jones : *Applied software measurement - Assuring productivity and quality* ; McGraw-Hill, 1991, 493 pages.

[11] S. H. Kan : *Metrics and models in software quality engineering* ; Addison-Wesley, 1993, 344 pages.

[12] N. Kececi, M. Li et C. Smidts : *Function point analysis: an application to a nuclear reactor protection system* ; Proceedings of the Probability Safety Assessment '99 Conference, Washington, D.C., 22-25 août 1999.

[13] P. Morris et J.-M. Desharnais : *Measuring all software, not just what the business uses* ; Proceedings of the IFPUG Fall Conference, Orlando, Floridea, 21-25 septembre 1998.

[14] S. Oligny, A. Abran, J.-M. Desharnais et P. Morris : *Functional size of real-time software: overview of field tests* ; Proceedings of the 13th International Forum on COCOMO and software cost modeling, Los Angeles, Californiea, 6-8 octobre 1998.

[15] D. St-Pierre, M. Maya, A. Abran, J.-M. Desharnais et P. Bourque : *Full Function Points: Counting practices manual* ; Technical Report 1997-04, Université du Québec à Montréal, Montréal, Canada, 1997, disponible à www.lrgl.uqam.ca/ffp.html.

[16] S. A. Whitmire : *3-D Function Points: Scientific and real-time extensions to function points* : Proceedings of the 1992 Pacific Northwest Software Quality Conference.