

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/341910074>

# POINTS DE FONCTION ÉTENDUS : GUIDE DE COMPTAGE

Technical Report · July 1998

DOI: 10.13140/RG.2.2.36789.50404

CITATIONS

0

READS

479

5 authors, including:



**Jean-Marc Desharnais**

École de Technologie Supérieure

109 PUBLICATIONS 1,572 CITATIONS

[SEE PROFILE](#)



**Alain Abran**

École de Technologie Supérieure

673 PUBLICATIONS 10,529 CITATIONS

[SEE PROFILE](#)



**Pierre Bourque**

École de Technologie Supérieure

120 PUBLICATIONS 4,113 CITATIONS

[SEE PROFILE](#)

## **POINTS DE FONCTION ÉTENDUS : GUIDE DE COMPTAGE**

Rapport technique 1998-07

Édité par :

Laboratoire de recherche en gestion des logiciels (UQAM)  
et

Laboratoire de métrique appliqué en gestion du logiciel (LMAGL)

En collaboration avec

Nortel

Bell Canada

Hydro-Québec

JECS System Research (Japan)

Préparé par :

Jean-Marc Desharnais

Denis St-Pierre

Alain Abran

Marcela Maya

Pierre Bourque

Juillet 1998

## Table des matières

|   |           |
|---|-----------|
| <b>1. Introduction.....</b>                                       | <b>4</b>  |
| <b>2. Sommaire des points de fonction.....</b>                    | <b>5</b>  |
| <b>3. Logiciels temps réel.....</b>                               | <b>7</b>  |
| <b>4. Limites des points de fonction pour le temps réel.....</b>  | <b>8</b>  |
| 4.1 Limites pour les données.....                                 | 8         |
| 4.2 Limites pour les transactions.....                            | 9         |
| <b>5. Extension au temps réel.....</b>                            | <b>10</b> |
| 5.1 Composants PFEs.....  | 10        |
| 5.2 Assignation des points.....                                   | 14        |
| 5.2.1 Assignation des points pour les composants de gestion.....  | 14        |
| 5.2.2 Assignation des points pour les composants de contrôle..... | 14        |
| <b>6. PFEs : Procédures de comptage et règles .....</b>           | <b>16</b> |
| 6.1 Identification des groupes de données.....                    | 17        |
| 6.2 Comptage des composants de données de contrôle.....           | 18        |
| 6.3 Identification des processus .....                            | 20        |
| 6.4 Comptage des composants transactionnels.....                  | 21        |
| <b>7. Exemple de comptage.....</b>                                | <b>25</b> |
| 7.1 Spécifications des besoins.....                               | 25        |
| 7.2 Comptage des PFEs.....  | 27        |
| 7.2.1 Identification des groupes de données.....                  | 27        |
| 7.2.2 Comptage des composants de données de contrôle.....         | 27        |
| 7.2.3 Identification des processus .....                          | 32        |
| 7.2.4 Comptage des composants transactionnels.....                | 32        |
| 7.2.5 Résumé de l'exemple de comptage .....                       | 45        |
| <b>8. Commentaires sur les PFEs.....</b>                          | <b>47</b> |
| 8.1 Facilité de compréhension .....                               | 47        |
| 8.2 Effort de comptage.....                                       | 48        |
| 8.3 Importance de la documentation .....                          | 48        |
| 8.4 Comptage des PFEs tôt dans le projet.....                     | 48        |
| <b>9. Sommaire.....</b>   | <b>48</b> |
| <b>Glossaire .....</b>  | <b>49</b> |
| <b>Références .....</b>   | <b>50</b> |

Copyright 1997. Tous droits réservés. *Le Laboratoire de Recherche en Gestion des Logiciels* de l'Université du Québec à Montréal et le *Laboratoire de Métriques Appliquées en Gestion du Logiciel* (LMAGL). Il est permis de copier ce document en tout ou en partie, sauf pour des avantages commerciaux, si le titre de la publication, sa date et ses auteurs sont cités et qu'une note à cet effet est transmise au Laboratoire de Recherche en Gestion des Logiciels de l'Université du Québec à Montréal et au Laboratoire de Métriques Appliquées en Gestion du Logiciel (LMAGL). Autrement, il est obligatoire d'obtenir une autorisation spécifique<sup>1</sup>.

---

<sup>1</sup> Vous trouverez une copie des procédures et règles d'une version française qui est du domaine publique dans un autre rapport aux adresses suivantes:

- <http://www.lmagl.qc.ca/PFEregle.pdf> ou
- [http://www.info.uqam.ca/Labo\\_Recherche/Lrgl/publi/268.pdf](http://www.info.uqam.ca/Labo_Recherche/Lrgl/publi/268.pdf)

## 1. INTRODUCTION

Compte tenu de leur importance en tant que produits et services, les logiciels sont devenus un poste budgétaire majeur pour les entreprises. Comme tout autre composant budgétaire, il est important de contrôler les dépenses, d'analyser la performance des montants alloués au développement de logiciels et de permettre l'étalonnage. Pour ce faire, des ratios et des modèles de productivité utilisant ces mesures sont nécessaires.

Les mesures sont nécessaires pour analyser tant la qualité que la productivité du développement et de l'entretien. D'une part, les mesures techniques sont nécessaires pour mesurer la performance technique des produits et services du point de vue des développeurs. Ces mesures techniques peuvent être utilisées pour des analyses d'efficacité pour améliorer, par exemple, la performance de l'architecture des logiciels.

D'autre part, les mesures fonctionnelles sont nécessaires pour mesurer la performance des produits ou services du point de vue de l'utilisateur, et elles sont nécessaires pour des analyses de productivité. Ces mesures fonctionnelles doivent être indépendantes des techniques de développement et des décisions liées à l'implantation. Elles peuvent être utilisées pour comparer la productivité des différentes techniques et technologies.

Une technique de mesure fonctionnelle ayant ces caractéristiques est disponible pour le domaine de l'informatique de gestion : la technique des points de fonction. Cette technique est largement utilisée pour des analyses de productivité et d'estimation [Abran, 1996; Desharnais, 1988; Jones, 1996; Kemerer, 1987]. Cette technique capture relativement bien les caractéristiques fonctionnelles des logiciels de gestion.

D'autre part, un certain nombre de critiques ont été formulées relatives à l'universalité de la technique des points de fonction [Conte, 1986; Galea, 1995; Grady, 1992; Hetzel, 1993; Ince, 1991; Jones, 1988; Jones, 1991; Kan, 1993; Whitmire, 1992]. Voici comment D.C. Ince décrit ce problème lié à la technique des points de fonction :

*«A problem with the function point approach is that it assumes a limited band of application types : typically, large file-based systems produced by agencies such as banks, building societies and retail organizations, and is unable to cope with hybrid systems such as the stock control system with a heavy communication component» [Ince, 1991, page 283].*

Il est reconnu que la technique des points de fonction (PF) ne peut capter toutes les caractéristiques fonctionnelles des logiciels en temps réel. Lorsque la technique des points de fonction est appliquée à ce type de logiciel, elle génère, bien sûr, un résultat, mais ce résultat n'est pas une mesure adéquate de la taille. Depuis 1997 une mesure équivalente à

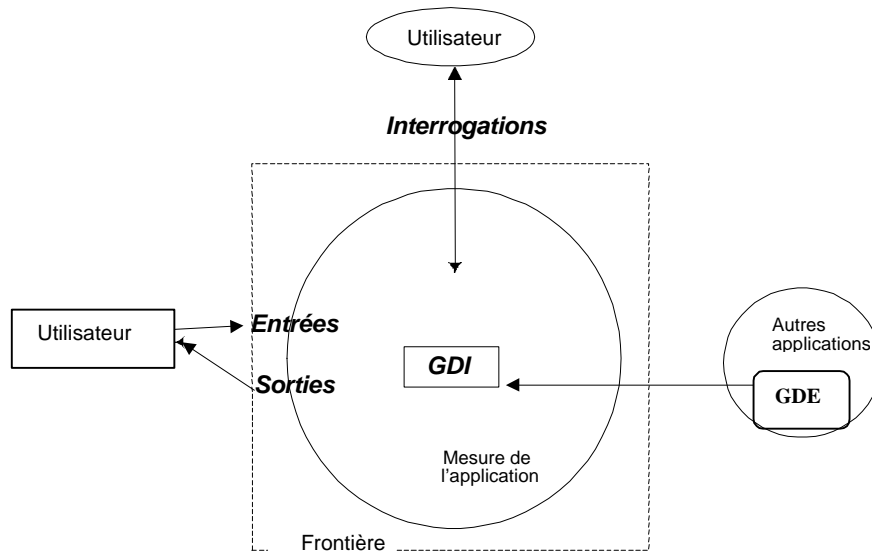
la technique des points de fonction avec des procédures détaillées de mesure pour les logiciels en temps réel est disponible. Cette méthode est présentée dans ce rapport. Elle permet de réaliser des mesures utiles d'étalonnage et de productivité de développement pour fins de modèles d'estimation basées sur la taille fonctionnelle des logiciels en temps réel.

Ce rapport décrit le travail fait par le Laboratoire de Recherche en gestion du logiciel de l'UQAM et son partenaire industriel le LMAGL pour adapter la technique des points de fonction aux logiciels en temps réel. Dans la section 2, nous présentons une courte description de ce qu'est la technique des points de fonction telle que définie par le International Function Point Users Group (IFPUG) version 4.0 [1994]. Ensuite, la section 3 présente quelques définitions de ce que sont les logiciels en temps réel. Dans la quatrième section nous vous présentons les limites de la technique des points de fonction en rapport avec la mesure des logiciels en temps réel. Dans la cinquième section, nous vous présentons une extension de la technique des points de fonction. La section 6 présente les règles et procédure du comptage de l'extension. Dans la section suivante, section 7, un exemple de comptage est présenté. Les aspects pratiques du comptage des avec l'extension proposé sont adressés à la section 8.

## **2. SOMMAIRE DES POINTS DE FONCTION**

La technique des points de fonction, développée par Allan Albrecht d'IBM, a été publiée pour la première fois en 1979. En 1984, IFPUG a été créé pour clarifier les règles, établir les normes et promouvoir son utilisation et son évolution. Dans ce rapport, toutes les références aux définitions de la technique des points de fonction et les règles de mesures ont été tirées de la plus récente publication de normes d'IFPUG, celle de 1994.

La première étape du comptage des points de fonction est d'identifier la frontière du comptage, c'est-à-dire la limite entre l'application ou le projet mesuré et les applications externes du domaine de l'utilisateur. La frontière indique quelles sont les fonctions qui sont incluses dans le comptage des points de fonction. La Figure 1 illustre la frontière entre une application et un utilisateur. L'étape suivante consiste à compter les *points de fonction bruts*, qui reflètent des fonctionnalités spécifiques comptées et fournies à l'utilisateur du projet ou de l'application. Le comptage des points de fonction bruts commence avec le comptage des cinq types de composants d'un projet ou d'une application : deux types de composants pour les données et trois types de composants pour les transactions. Les cinq types de composants comptés sont (Figure 1) :



**Figure 1 – Frontière et composants**

Composants de type données :

**Groupe de données internes (GDI) :** C'est un groupe de paramètres de traitement ou de données logiquement liées, identifiable par l'utilisateur, et maintenu à l'intérieur des frontières de l'application mesurée.

**Groupes de données externes (GDE) :** C'est un groupe de paramètres de traitement ou de données logiquement liées, identifiable par l'utilisateur, et maintenu à l'extérieur des frontières de l'application mesurée. Ceci veut dire qu'un GDE compté dans une application est toujours un GDI dans une autre application.

Composants de type transaction :

**Entrée (ENT) :** Une ENT traite des données ou des paramètres de traitement qui viennent de l'extérieur des frontières de l'application mesurée. Une ENT est un processus élémentaire<sup>2</sup>.

**Sortie (SOR) :** Une SOR génère des données ou des paramètres de traitement qui sont envoyés en dehors des frontières de l'application mesurée.

**Interrogation (INT) :** Une INT représente une combinaison d'une entrée (requête) et d'une sortie (extraction). La partie sortie ne contient pas de données dérivées<sup>3</sup>. Il n'y a pas de GDI maintenu lors de ce processus.

<sup>2</sup> **Processus élémentaire :** C'est la plus petite activité pour l'utilisateur final du point de vue métier. Ce processus élémentaire doit être autonome et laisser l'application en état de cohérence fonctionnelle.

Ces cinq (5) composants sont classés selon leur complexité : bas, moyen ou élevé, en utilisant un ensemble de normes. Après avoir classé chacun des cinq types de composants, les points de fonction bruts (PFB)<sup>4</sup> sont comptés en utilisant des poids prédéfinis pour chaque type de composant.

La dernière étape implique une évaluation des fonctionnalités générales fournies à l'utilisateur de l'application. L'impact des quatorze caractéristiques générales de système (GSC) est comptée sur une échelle de 0 à 5 en termes de leur effet probable sur le projet ou l'application. On obtient ainsi un Facteur d'ajustement (FA) qui ajuste le PFB de  $\pm 35\%$  (maximum) pour produire le nombre de points de fonction net (PFN).

Une description complète de la technique des points de fonction comprenant des définitions, des procédures, des règles de comptage et des exemples, se trouve dans le Guide de comptage des Points de Fonction [IFPUG, 1994]<sup>5</sup>.

### 3. LOGICIELS TEMPS RÉEL

On retrouve dans la littérature des définitions différentes des «logiciels en temps réel» :

Stankovic et Ramamritham [1988] :

*«Real-time systems are defined as those systems in which the **correctness** of the system depends not only on the result of computation, but also on the **time** at which the results are produced».*

Laplante [1993] et Cooling [1991] :

*«Should computer responses exceed these time bounds then performance degradation and/or malfunction results».*

Stankovic et Ramamritham [1988] :

*«Real-time systems are defined as those systems in which the correctness of the system depends not only on the result of computation, but also on the time at which the results are produced».*

---

<sup>3</sup> Les données qui requièrent des traitements autres que l'extraction directe et édition d'information de Groupes logiques de Données Internes ou Externes.

<sup>4</sup> Le nombre de fonctionnalités spécifiques fournies à l'utilisateur par le projet ou l'application avant l'utilisation du Facteur d'ajustement (FA).

<sup>5</sup> Traduction française version 1.0.



The Oxford Dictionary of Computing [Illingworth 1991] :

*«Any system in which the time at which the output is produced is significant. This is usually because the input corresponds to some movement in the physical world, and the output has to relate to that same movement. The lag from input time to output time must be sufficiently small for acceptable timeliness».*

De ces définitions, deux aspects importants des logiciels en temps réel sont adressés : la synchronisation et l'interaction avec les entités externes. Le but de l'interaction est d'obtenir de l'information et/ou de contrôler des entités externes. L'interaction survient dans un contexte de contraintes serrées quant au choix et à la synchronisation des tâches. Les logiciels en temps réel sont différents parce que les contraintes de synchronisation sont explicites. Les logiciels en temps réel incorporent des éléments dédiés pour gérer ces contraintes; un échec dans l'accomplissement de ces contraintes de synchronisation peut résulter en une défaillance. En conséquence, pour les logiciels en temps réel, le bon fonctionnement (absence de conditions d'erreur) est lié à la synchronisation.

#### **4. LIMITES DES POINTS DE FONCTION POUR LE TEMPS RÉEL**

##### **4.1 Limites pour les données**

Structurellement, il y a deux types de contrôle de données : les groupes de données à occurrences multiples et les groupes de données à occurrence simple. Les groupes de données à occurrences multiples peuvent avoir un même type d'enregistrement qui se répète plus d'une fois<sup>6</sup>. Les groupes de données à occurrence simple n'apparaissent qu'une seule fois dans l'application. Les systèmes en temps réel contiennent typiquement un grand nombre de données à occurrence simple qu'il est difficile de grouper en GDI et GDE. Il est donc nécessaire d'élargir les règles des GDI et GDE pour mesurer adéquatement les données à occurrence simple.

---

<sup>6</sup> Par exemple, un moteur de contrôle d'une application peut avoir un groupe de données de contrôle ayant une information sur chaque cylindre (numéro du cylindre, synchronisation de l'allumage, pression, etc.). Un tel groupe de données a une structure d'occurrences multiples (un enregistrement pour chaque cylindre). Autrement dit, l'enregistrement du cylindre est répété plus d'une fois.

## 4.2 Limites pour les transactions

Les processus des logiciels en temps réel ont en commun des caractéristiques transactionnelles spécifiques : le nombre de sous-processus varie considérablement. La technique de mesure fonctionnelle pour le temps réel doit tenir compte du fait que certains processus ont peu de sous-processus, alors que d'autres processus ont un grand nombre de sous-processus. Ce phénomène est illustré par ces deux exemples :

### Exemple 1 – Un système de contrôle de température (processus avec peu de sous-processus)

Le but principal de ce processus est de mettre en marche un système de refroidissement lorsque nécessaire. Un capteur saisie la température dans l'application (sous-processus 1). La température est comparée à un seuil de surchauffe (sous-processus 2). Finalement, un message de mise en marche peut être transmis, si nécessaire, au système de refroidissement (sous-processus 3).

Dans cet exemple, le régulateur de température a 3 sous-processus (Table 1). Ce processus n'est pas une application; c'est seulement un des nombreux processus d'un système de contrôle de la température. L'application n'est pas dans un état de cohérence fonctionnelle tant que tous les sous-processus de régulation de la température ne sont pas complétés. Donc, il n'y a qu'un processus élémentaire [IFPUG, 1994].

| Processus          | Description du sous-processus      | # de sous-processus |
|--------------------|------------------------------------|---------------------|
| Moteur de contrôle | Saisie de la température           | 1                   |
|                    | Lecture du seuil pour comparaison  | 1                   |
|                    | Envoi du message de mise en marche | 1                   |
|                    | <b>Total</b>                       | <b>3</b>            |

**Table 1 – Sous-processus de l'exemple 1**

Selon la norme des points de fonction, un seul composant transactionnel serait identifié puisqu'il n'y a qu'un seul processus élémentaire.

### Exemple 2 – Un processus de diagnostique (processus avec plusieurs sous-processus)

Le but principal de ce processus est de déclencher une alarme lorsque nécessaire. Quinze capteurs (tous différents) envoient des données au processus de diagnostique (15 sous-processus, 1 sous-processus unique pour chaque sorte de capteur). Pour chaque capteur, un ensemble de données externes est reçu et comparé à la valeur de seuil lue à partir d'un fichier interne, un fichier unique pour chaque sorte de capteur (15 autres sous-processus, 1 processus unique pour chaque sorte de capteur). Selon

certaines conditions, une alarme peut être mise en marche sur le tableau de bord (1 sous-processus).

Selon cet exemple, le système de régulation comporte 31 sous-processus (Table 2). Ce processus n'est pas une application; c'est seulement un des nombreux processus d'un système de régulation d'un moteur. Ce système n'est pas dans un état de cohérence fonctionnelle tant que tous les sous-processus ne sont pas complétés.

| Processus                 | Description du sous-processus       | # de sous-processus |
|---------------------------|-------------------------------------|---------------------|
| Régulateur de température | Saisie des données via les capteurs | 15                  |
|                           | Lecture des seuils de comparaison   | 15                  |
|                           | Envoi d'un message d'alarme         | 1                   |
|                           | <b>Total</b>                        | <b>31</b>           |

**Table 2 – Sous-processus de l'exemple 2**

Selon la norme des points de fonction, très peu de composants transactionnels peuvent être comptés puisque les fonctions transactionnelles sont basées sur les processus élémentaires plutôt que sur les sous-processus. Cependant, selon les procédures des points de fonction d'IFPUG, les exemples 1 et 2 ont approximativement le même nombre de points liés aux transactions.

## **5. EXTENSION AU TEMPS RÉEL**

La technique des points de fonction étendues (PFEs) est une mesure fonctionnelle basée sur les normes de la technique des points de fonction. Elle a été conçue tant pour les logiciels de gestion que pour les logiciels en temps réel. Puisque la technique des PFEs est une extension des normes de la technique des points de fonction, toutes les règles d'IFPUG sont inclus dans la nouvelle technique de mesure. De plus, les facettes liées au contrôle des logiciels en temps réel ont été adressées par l'ajout de nouveaux composants.

### **5.1 Composants PFEs**

Pour mesurer adéquatement les caractéristiques d'un nombre variable de sous-processus, il est nécessaire de considérer non seulement les processus tels que définis par les points de fonction (processus élémentaires), mais aussi les sous-processus. De plus, la structure particulière d'une donnée de contrôle typique doit être considérée. Conséquemment, la

technique des PFEs introduit des composants additionnels tant pour les données que pour les transactions.

Les deux nouveaux composants liés aux données de contrôle ont une structure similaire aux composants de données d'IFPUG :

- Groupe de contrôle mis à jour (GCM) : Un GCM est un groupe de données de contrôle qui est mis à jour par l'application mesurée. Il est identifié selon une perspective fonctionnelle<sup>7</sup>. Le groupe de données de contrôle vit pour plus d'une transaction<sup>8</sup>.
- Groupe de contrôle lu (GCL) : Un GCL est un groupe de données de contrôle utilisé, mais pas mis à jour, par l'application mesurée. Il est identifié selon une perspective fonctionnelle. Le groupe de contrôle de données vit pour plus d'une transaction<sup>9</sup>.

Les quatre nouveaux composants liés aux composants transactionnels de contrôle adressent les sous-processus des logiciels en temps réel :

- Entrée de contrôle externe (ECE) : Une ECE est un sous-processus unique. L'ECE est identifiée selon une perspective fonctionnelle. Une Entrée de contrôle externe (ECE) traite des données qui viennent de l'extérieur des frontières de l'application mesurée. C'est la plus petite décomposition d'un processus liée à un seul groupe de données. Conséquemment, si le processus reçoit deux groupes de données, il y a au moins deux ECEs. L'ECE exclue la mise à jour des données, la fonctionnalité de mise à jour étant couverte par un autre type de composant (Écriture de contrôle interne ou ECI).

Dans le diagnostic de moteur (exemple 2 de la section 4), 15 capteurs envoient des données à l'application (la donnée de contrôle traverse la frontière de l'application). Ainsi, il y a un processus unique pour chaque capteur, c'est-à-dire 15 ECEs.

- Sortie de contrôle externe (SCE) : Une SCE est un sous-processus unique. La SCE est identifiée selon une perspective fonctionnelle. Une Sortie de contrôle externe (SCE) traite des données qui sortent à l'extérieur des frontières de l'application mesurée. C'est la plus petite décomposition d'un processus liée à

---

<sup>7</sup> Ce qui veut dire que le groupe de données apparaît dans les spécifications de l'application, en assumant qu'elles sont complètes.

<sup>8</sup> Dans l'exemple 2 de la section 4, la donnée du capteur ne vit que pour une seule transaction; après le diagnostic le système ne la retient pas. Par opposition, les seuils sont réutilisés pour chaque nouvelle entrée, et en conséquence, demeurent pour plus d'une transaction. La section 7 donne des exemples additionnels.

<sup>9</sup> Les règles pour les GCM et GCL sont différentes des règles des GDI et GDE pour les points de fonction afin de permettre à la technique des PFE de mesurer la dimension des données des logiciels en temps réel. Les règles des GDI et GDE sont plutôt orientées vers des données dont la structure est typique aux systèmes de gestion.

un seul groupe de données. Conséquemment, si le processus envoie deux groupes de données, il y a au moins deux SCEs. La SCE exclue la lecture des données, la fonctionnalité de lecture étant couverte par un autre type de composant (Lecture de contrôle interne ou LCI).

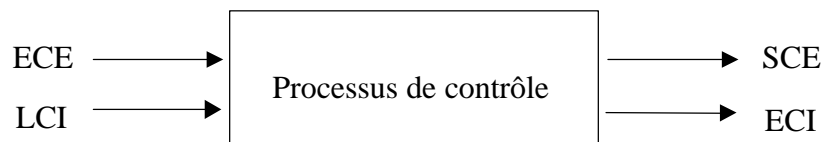
Dans le diagnostic de moteur (exemple 2), le sous-processus qui envoie un message au tableau de bord (la donnée de contrôle est envoyée à l'extérieur de la frontière de l'application) est un SCE.

- Lecture de contrôle interne (LCI) : Une LCI est un sous-processus unique. La LCI est identifiée selon une perspective fonctionnelle. Une Lecture de contrôle interne (LCI) lit les données de contrôle. C'est la plus petite décomposition d'un processus liée à un seul groupe de données. Conséquemment, si l'activité touche à deux groupes de données, il y a au moins deux LCIs.

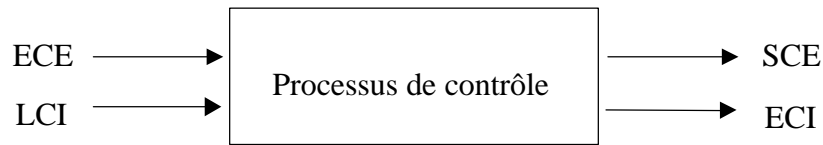
Dans le diagnostic de moteur (exemple 2), les sous-processus qui lisent les valeurs de seuil sont des LCIs. Dans cet exemple, 15 sous-processus uniques lisent différentes sortes de valeurs de seuil à des temps différents pour des fins de comparaison. Il y a donc 15 LCIs.

- Écriture de contrôle interne (ECI) : Une ECI est un sous-processus unique. L'ECI est identifiée selon une perspective fonctionnelle. Une Écriture de contrôle interne (ECI) écrit les données de contrôle. C'est la plus petite décomposition d'un processus liée à un seul groupe de données. Conséquemment, si l'activité touche à deux groupes de données, il y a au moins deux ECIs.

A titre d'exemple le système de diagnostic est étendu avec les fonctionnalités suivantes : «Les 15 ensembles de données de contrôle sont conservés». Ils sont tous conservés séparément à des temps différents et dans des fichiers différents (15 sous-processus différents). Puisqu'il y a 15 sortes de capteurs de contrôle des données mis à jour à des temps différents (15 sous-processus uniques) il y a 15 ECIs selon la technique des PFEs.



Voici comment les composants de contrôle transactionnels sont liés au processus de contrôle :



**Figure 2 - Diagramme des fonctions de contrôle transactionnels**

Comme pour la technique des points de fonction (PF), tous les nouveaux composants sont basés sur la perspective fonctionnelle de l'application, plutôt que sur la perspective technique. La différence entre la norme de la technique des points de fonction et la technique étendue proposée (PFE) consiste en l'addition d'un certain nombre de composants (GCM, GCL, ECE, SEC, LCI et ECI). Ces nouveaux composants sont seulement utilisés pour la mesure des données de contrôle et processus de contrôle. Les autres types de données et processus, appelés données et processus de gestion, sont comptés selon la norme de comptage des points de fonction (Figure 3 et Table 3)

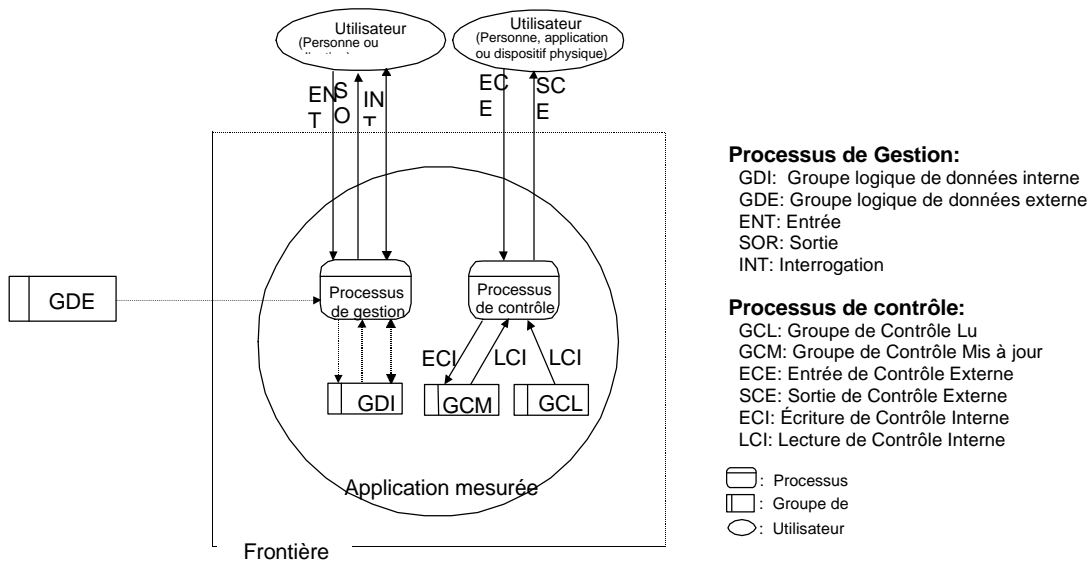
PFE, composants de gestion :

|                                  |  |
|----------------------------------|--|
| Groupe de données internes (ILF) | existe dans les PFs, inchangé dans PFE |
| Groupe de données externes (EIF) | existe dans les PFs, inchangé dans PFE |
| Entrées (ENT)                    | existe dans les PFs, inchangé dans PFE |
| Sorties (SOR)                    | existe dans les PFs, inchangé dans PFE |
| Interrogations (INT)             | existe dans les PFs, inchangé dans PFE |

PFE, composants de contrôle :

|                                    |  |
|------------------------------------|--|
| Groupe de contrôle MAJ (GCM)       | nouveau composant similaire au GDI                                 |
| Groupe de contrôle lu (GCL)        | nouveau composant similaire au GDE                                 |
| Entrée de contrôle externe (ECE)   | nouveau composant, similaire à un sous-ensemble des ENTs           |
| Sortie de contrôle externe (SCE)   | nouveau composant, similaire à un sous-ensemble des ENTs/INTs      |
| Lecture de contrôle interne (LCI)  | nouveau composant, similaire à un sous-ensemble des ENTs/SORs/INTs |
| Écriture de contrôle interne (ECI) | nouveau composant, similaire à un sous-ensemble des ENTs           |

**Table 3 – Liste des composants de PFE**



**Figure 3 – Diagramme des composants des PFEs**

Ainsi, les points bruts d'une application utilisant la technique proposée (PFE) peuvent être comptés de la façon suivante :

$$\begin{aligned} \text{PFE} &= \text{PF de gestion} + \text{PF de contrôle} \\ &= (\text{PF} - \text{paramètres de traitement}) + \text{PF de contrôle} \end{aligned}$$

## 5.2 Assignment des points

La section précédente décrit comment identifier les composants. L'étape suivante consiste à assigner des points à chaque composant identifié dans le comptage de l'application.

### 5.2.1 Assignment des points pour les composants de gestion

Pour les composants IFPUG (GDI, GDE, ENT, SOR, INT : table 3), l'assignation des points reste inchangée dans PFE. Donc, la procédure d'assignation des points d'IFPUG [IFPUG, 1994] s'applique comme tel.

### 5.2.2 Assignment des points pour les composants de contrôle

#### 5.2.2.1 Assignment des points pour les composants de données de contrôle

##### 5.2.2.1.1 Groupe de données à occurrences multiples

Les groupes de données à occurrences multiples (Section 4.1) ont la même structure que les GDIs et GDEs dans la technique des points de fonction. En conséquence, ils sont

comptés exactement de la même façon que les deux composants des PFs, c'est-à-dire en utilisant le nombre de données élémentaires (DEs) et les sous-ensemble logiques de données (SLDs) ainsi que la table de complexité correspondante (voir IFPUG 1994 pour les définitions et les tables).

#### 5.2.2.1.2 Groupe de données à occurrence simple

Le nombre de points assignés pour les groupes de données à occurrence simple (voir section 4.1) dépend uniquement du nombre de DEs. Le nombre de points est compté de la façon suivante pour un GCM :  $(\text{nombre de DEs} / 5) + 5$ . Pour un GCL, la formule est :  $\text{nombre de DEs} / 5$ . Ces formules ont été conçues pour garder la taille d'un groupe de données à occurrence simple en lien avec la taille des GDIs et GDEs des PFs autant que possible.

Une groupe de données mis à jour à occurrence simple (GCM) comprend toutes les valeurs de contrôle mise à jour (selon une perspective fonctionnelle) de l'application mesurée. Puisqu'il contient toutes les valeurs à occurrence simple de l'application, il ne peut y en avoir qu'un seul par application. En conséquence, une application peut avoir plusieurs groupes de données à occurrences multiples mais un seul groupe à occurrence simple. Il en est de même pour les groupes de données lues à occurrence simple (GCLs).

Dans les applications temps réel typiques, le nombre des valeurs varient de quelques-unes à plusieurs centaines. C'est pourquoi une formule est utilisée plutôt qu'une table à 2 niveaux tel dans la technique des PFs. Il permet aux PFEs de considérer un grand nombre de données à occurrence simple. Les tables suivantes présentent des exemples de ces formules :

Exemple d'assignation des points d'un GCM à occurrence simple :  $((\text{nombre de DEs} / 5) + 5)$  :

|                        |        |        |        |
|------------------------|--------|--------|--------|
| <b>Nombre de DEs :</b> | 10 DEs | 35 DEs | 50 DEs |
| <b>Points:</b>         | 7      | 12     | 15     |

**Table 4 : Exemple d'assignation des points d'un GCM à occurrence simple**

Exemple d'assignation des points d'un GCL à occurrence simple :  $(\text{nombre de DEs} / 5)$  :

|                        |        |        |        |
|------------------------|--------|--------|--------|
| <b>Nombre de DEs :</b> | 10 DEs | 35 DEs | 50 DEs |
| <b>Points :</b>        | 2      | 7      | 10     |

**Table 5 : Exemple d'assignation des points d'un GCL à occurrence simple**

La section 7 donne des exemples additionnels de données à occurrence simple



#### 5.2.2.2 Assignment des points pour les composants transactionnels de contrôle

Le nombre de points assignés aux composants transactionnels de contrôle (ECE, SCE, LCI, ECI) dépend du nombre de DEs<sup>10</sup>. Lorsque le nombre de DEs est déterminé, la table suivante est utilisée pour transformer les DEs en points<sup>11</sup> :

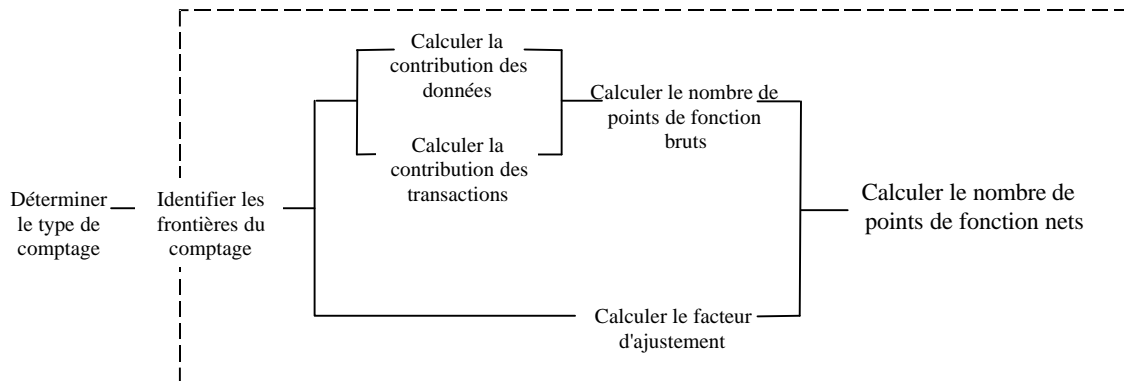
|                 |          |           |         |
|-----------------|----------|-----------|---------|
| Nombre de DEs : | 1-19 DEs | 20-50 DEs | 51+ DEs |
| Points :        | 1        | 2         | 3       |

**Table 6 : Table d'assignation des points des composants des transactions de contrôle**

L'étendue des intervalles (1 à 19, 20 à 50 et 51+) a été choisie afin d'aligner la taille des composants de contrôle transactionnels le plus possible avec les PFs.

## 6. PFES : PROCÉDURES DE COMPTAGE ET RÈGLES

Le diagramme suivant est un résumé de la procédure de comptage des points de fonction :



**Figure 4 – Procédure de comptage des points de fonction**

Les principales étapes de la procédure de comptage des points de fonction sont les suivantes<sup>12</sup> :

1. Déterminer le type de comptage en points de fonction
2. Identifier les frontières du comptage

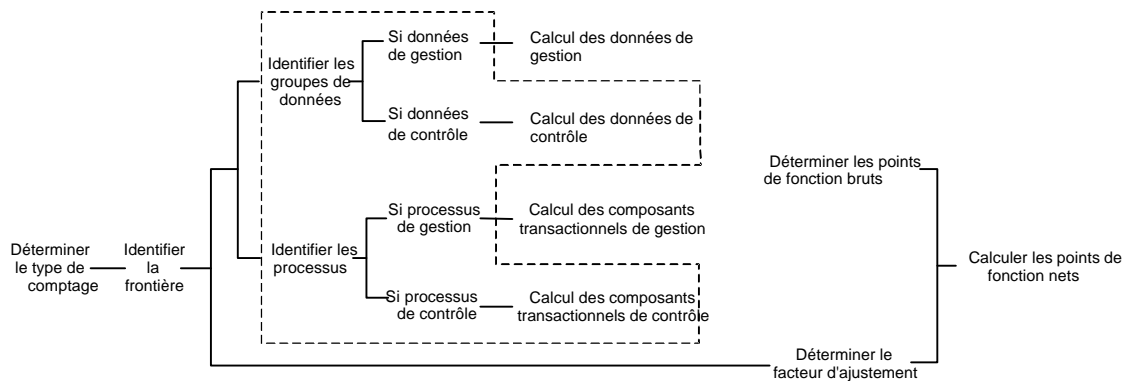
<sup>10</sup> DE : Une unité d'information identifiable et non répétitive (rubrique, champ, attribut, ...) (glossaire IFPUG 1994)

<sup>11</sup> Dans le système de diagnostic par exemple, le sous-processus qui envoie un message d'alarme au tableau de bord a 2 DEs (nombre de boutons, code d'action : ouvrir), le nombre de points est donc 1 (colonne «1 à 19 DEs» de la table 6).

<sup>12</sup> Tiré du «Guide de comptage des Points de Fonction», IFPUG.

3. Déterminer le nombre de points de fonction bruts
  - a. Compter les types de composant relatifs aux données
  - b. Compter les types de composant relatifs aux transactions
4. Déterminer le facteur d'ajustement
5. Compter le nombre de points de fonction ajustés

Pour les PFEs, les étapes 1, 2, 4 et 5 sont les mêmes que précédemment [IFPUG]. L'étape 3 est divisée en composants de gestion et en composants de contrôle.



**Figure 5 – Diagramme de la procédure de comptage des PFEs**

Les portions de la procédure à l'intérieur des lignes pointillées correspondent à de nouvelles étapes des PFEs et sont couvertes dans ce document. Les autres portions sont couvertes par le Guide de comptage des points de fonction [IFPUG, 94].

Référer au même document de la même façon dans un document.

## 6.1 Identification des groupes de données

Cette étape consiste à identifier les groupes de données qui représentent les fonctionnalités livrées aux utilisateurs par l'application. Lorsque ces groupes de données sont identifiés, les définitions et règles associées à ces composants sont appliquées afin de s'assurer que le groupe de données est bien un composant à compter selon PFEs.

### Définitions

**Données de gestion :** Données utilisées par l'application pour supporter l'utilisateur dans sa gestion des informations administratives et d'affaires.

**Données de contrôle :** Données utilisées par l'application pour contrôler directement ou indirectement le comportement d'une application ou d'une machine.

**Groupe de données :** Données identifiées et groupées ensemble d'un point de vue fonctionnel.

### **Procédure d'identification**

La procédure d'identification des groupes candidats de données est la suivante :

1. Rechercher les groupes de données identifiables selon une perspective fonctionnelle.
2. Déterminer si le groupe de données est un groupe de données de gestion ou un groupe de données de contrôle en utilisant les définitions précédentes. Pour les groupes de données de gestion, les règles et procédures liées aux groupes de données internes (GDI) et groupes de données externes (GDE) s'appliquent [IFPUG, 1994]. Pour les groupes de données de contrôle, les procédures et règles qui suivent s'appliquent.

Note : Toutes les données de contrôle à occurrence simple (voir entre autre le seuil de température de l'exemple 1, section 4.2) mises à jour par l'application sont regroupées dans un seul et unique groupe de données.

## **6.2 Comptage des composants de données de contrôle**

### **Définitions**

- Groupe de contrôle mis à jour (GCM) : Un GCM est un groupe de données de contrôle qui est mis à jour par l'application mesurée. Il est identifié selon une perspective fonctionnelle. Le groupe de données de contrôle vit pour plus d'une transaction.
- Groupe de contrôle lu (GCL) : Un GCL est un groupe de données de contrôle utilisé, mais pas mis à jour, par l'application mesurée. Il est identifié selon une perspective fonctionnelle. Le groupe de données de contrôle vit pour plus d'une transaction.

### **Définitions des termes enchâssés**

Perspective fonctionnelle : Point de vue de la fonctionnalité livrée par l'application; exclu toute considération technique ou de mise en œuvre.

Transaction : Tout traitement associé à une occurrence d'un déclencheur externe.

### **Procédure de comptage**

Pour chaque groupe de données identifié à l'étape précédente comme groupe de données de contrôle :

1. Déterminer si le groupe de données de contrôle est un GCM ou un GCL en utilisant les définitions et les règles.
2. Déterminer la contribution du GCM ou du GCL (assignation des points) aux points non ajustés.

### Règles d'identification

- Règles d'identification du GCM (Groupe de contrôle mis à jour) :
  - ☐ Le groupe identifié est un groupe de données de contrôle logique ou un groupe de données à occurrence simple.
  - ☐ Le groupe de données de contrôle est **mis à jour** à l'intérieur de la frontière de l'application.
  - ☐ Le groupe de données de contrôle vit pour plus d'une transaction.
  - ☐ Le groupe de données de contrôle identifié n'a pas déjà été compté comme GCL, GDI ou GDE dans l'application mesurée.

Toutes les règles précédentes doivent s'appliquer selon une perspective fonctionnelle et elles sont toutes obligatoires pour identifier un GCM.

- Règles d'identification du GCL :
  - ☐ Le groupe identifié est un groupe de données de contrôle logique ou un groupe de contrôle à occurrence simple.
  - ☐ Le groupe de données de contrôle **n'est pas mis à jour** à l'intérieur de la frontière de l'application.
  - ☐ Le groupe de données de contrôle est référencé par l'application qui est comptée.
  - ☐ Le groupe de données de contrôle vit pour plus d'une transaction.
  - ☐ Le groupe de données de contrôle identifié n'a pas déjà été compté comme GCM, GDI ou GDE dans l'application mesurée.

Toutes les règles précédentes doivent s'appliquer selon une perspective fonctionnelle et elles sont toutes obligatoires pour identifier un GCL.

### Règles d'assignation des points

Le nombre de points assignés aux GCMs et GCLs dépend du type de groupe de données (occurrence simple ou occurrences multiples). Les groupes de données de contrôle à occurrences multiples ayant la même structure que les GDIs et GDEs sont comptés de la même manière, c'est-à-dire que leur complexité est déterminée par le nombre de données élémentaires (DEs) et par le nombre de sous ensembles logiques de données (SLDs) en utilisant la table de complexité correspondante.

Pour les groupes de données de contrôle ayant une occurrence simple, le nombre de points dépend du nombre de DEs seulement. Lorsque le nombre de DEs a été déterminé sur la

base des règles appliquées aux GDIs et GDEs, le nombre de points est calculé en utilisant les formules suivantes :

$$\text{GCM} : ((\text{nombre de DEs} / 5) + 5)$$

$$\text{GCL} : (\text{nombre de DEs} / 5)$$

### **6.3 Identification des processus**

Après avoir calculé le nombre de points pour les données de gestion et de contrôle, les composants transactionnels doivent être identifiés. Les composants de type «transactions de contrôle» ou composants transactionnels représentent la fonctionnalité fournie à l'utilisateur via le traitement des données de contrôle de l'application. Ainsi, pour identifier des composants transactionnels, il faut d'abord identifier les processus de l'application.

#### **Procédure d'identification**

La procédure d'identification des processus est la suivante :

1. Rechercher les différents processus de l'application selon une perspective fonctionnelle.
2. Déterminer si le processus est un processus de gestion ou un processus de contrôle en utilisant les définitions suivantes :

Processus de gestion : processus dont le but est de supporter l'utilisateur pour gérer ses informations, particulièrement les informations de gestion et d'administration.

Processus de contrôle : processus qui contrôle directement ou indirectement le comportement d'une application ou d'une machine.

3. Si le processus est un processus de contrôle, appliquer les règles d'identification des quatre nouveaux composants de contrôle transactionnel. Si le processus est un processus de gestion, appliquer les définitions et les règles des composants transactionnels des points de fonction [IFPUG, 94].

### **6.4 Comptage des composants transactionnels**

#### **Définitions**

- Entrée de contrôle externe (ECE) : Une ECE est un sous-processus unique. L'ECE est identifiée selon une perspective fonctionnelle. Une Entrée de contrôle externe (ECE) traite des données qui viennent de l'extérieur des frontières de l'application mesurée. C'est la plus petite décomposition d'un

processus liée à un seul groupe de données. Conséquemment, si le processus reçoit deux groupes de données, il y a au moins deux ECEs. L'ECE exclue la mise à jour des données, la fonctionnalité de mise à jour étant couverte par un autre type de composant (Écriture de contrôle interne ou ECI).

- Sortie de contrôle externe (SCE) : Une SCE est un sous-processus unique. La SCE est identifiée selon une perspective fonctionnelle. Une Sortie de contrôle externe (SCE) traite des données qui sortent à l'extérieur des frontières de l'application mesurée. C'est la plus petite décomposition d'un processus liée à un seul groupe de données. Conséquemment, si le processus envoie deux groupes de données, il y a au moins deux SCEs. La SCE exclue la lecture des données, la fonctionnalité de lecture étant couverte par un autre type de composant (Lecture de contrôle interne ou LCI).
- Lecture de contrôle interne (LCI) : Une LCI est un sous-processus unique. La LCI est identifiée selon une perspective fonctionnelle. Une Lecture de contrôle interne (LCI) lit les données de contrôle. C'est la plus petite décomposition d'un processus liée à un seul groupe de données. Conséquemment, si l'activité touche à deux groupes de données, il y a au moins deux LCIs.
- Écriture de contrôle interne (ECI) : Une ECI est un sous-processus unique. L'ECI est identifiée selon une perspective fonctionnelle. Une Écriture de contrôle interne (ECI) écrit les données de contrôle. C'est la plus petite décomposition d'un processus liée à un seul groupe de données. Conséquemment, si l'activité touche à deux groupes de données, il y a au moins deux ECIs.

### **Définition des termes enchâssés**

- Sous-processus : Dans le contexte de ce rapport, c'est la plus petite étape d'un processus qui soit identifiable par l'utilisateur dans une perspective fonctionnelle que ce soit une entrée, une sortie, une lecture ou une écriture.
- Processus de contrôle : Un processus qui contrôle directement ou indirectement le comportement d'une application ou d'une machine.
- Perspective fonctionnelle : Point de vue de la fonctionnalité livrée par l'application; exclue les considérations techniques et de mise en œuvre.
- Données de contrôle : Données utilisées par l'application pour contrôler directement ou indirectement le comportement d'une application ou d'une machine.
- Utilisateur : Une personne, une application ou une machine qui interagit avec l'application mesurée.

### **Procédure de comptage**

Lorsque le processus a été identifié comme étant un processus de contrôle, les étapes suivantes doivent être exécutées :

- Identifier tous les sous-processus fonctionnels (non techniques) du processus de contrôle.
- Identifier le type de sous-processus (ECE, SCE, ECI ou LCI) en appliquant les définitions et règles.
- Déterminer la contribution (assignation des points) de chaque ECE, SCE, ECI ou LCI au comptage des points bruts

Étapes d'identification des sous-processus :

1. Sur la base de l'ordre d'exécution logique des sous-processus à l'intérieur du processus, identifier le premier sous-processus qui entre, sort, lit ou écrit un groupe de données de contrôle.
2. Appliquer l'ensemble de règles pertinentes à chaque type de sous-processus (ECE, SCE, ECI ou LCI).
3. Déterminer la contribution de chaque ECE, SCE, ECI ou LCI (assignation des points) aux points non ajustés.
4. Toujours selon l'ordre d'exécution, identifier le prochain sous-processus qui entre, sort, lit ou écrit un groupe de données de contrôle. Il peut y avoir plus d'un «prochain sous-processus» (ex. : branchement avec un ou plusieurs chemins). Dans ce cas, tous les chemins doivent être explorés afin de s'assurer qu'il ne reste plus de sous-processus à identifier.
5. Répéter les étapes 2 à 4 jusqu'à ce que tous les sous-processus des processus soient identifiés.
6. À la fin du cycle, enlever tous les sous-processus apparaissant plus d'une fois c'est-à-dire même traitement<sup>13</sup> et même DEs

Note : si le même sous-processus est associé à différents processus de contrôle, il peut être compté plus d'une fois.

### **Règles d'identification**

- Règles d'identification des Entrées de contrôle externes (ECE) :

---

<sup>13</sup> Un sous-processus n'est pas seulement une entrée, une sortie, une lecture ou une écriture, mais peut inclure aussi d'autres types de traitements (calculs, tris, comparaisons, etc.) associés aux sous-processus identifiés.

- ☐ Le sous-processus reçoit un groupe de données de contrôle de l'extérieur des frontières de l'application.
- ☐ Le sous-processus ne reçoit qu'un groupe de données. Si plus d'un groupe de données est reçu, il faut alors compter une ECE pour chaque groupe de données.
- ☐ Le sous-processus ne sort, ne lit ou n'écrit pas de données.
- ☐ Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres ECEs associées aux mêmes processus.

Note 1 : Les déclencheurs temporels (provenant de l'horloge interne de l'ordinateur) sont considérés comme externes. Par exemple, un événement qui survient à toutes les 3 secondes est compté comme une Entrée de contrôle externe (ECE) avec une DE. Cependant, le processus qui génère l'événement périodiquement est ignoré.

Note 2 : À moins qu'un processus spécial soit nécessaire, la lecture de l'horloge interne n'est pas comptée. Par exemple, lorsqu'un processus écrit automatiquement une heure (inscription de l'heure) on ne compte pas de Lecture de contrôle interne (LCI) pour l'obtention de la valeur se trouvant dans l'horloge interne.

Toutes les règles de comptage précédentes doivent être appliquées selon une perspective fonctionnelle, et sont toutes obligatoires pour identifier une ECE.

- Règles d'identification des Sorties de contrôle externes (SCE) :
  - ☐ Le sous-processus envoie un groupe de données de contrôle de l'extérieur des frontières de l'application.
  - ☐ Le sous-processus n'envoie qu'un groupe de données. Si plus d'un groupe de données est envoyé, il faut alors compter une SCE pour chaque groupe de données.
  - ☐ Le sous-processus n'entre, ne lit ou n'écrit pas de données.
  - ☐ Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres SCEs associées aux mêmes processus.

Note : Tous les messages sans données d'utilisateur (ex. : confirmation et erreur) sont comptés comme une seule SCE. Le nombre de DEs est le nombre des différents types de messages.

Toutes les règles de comptage précédentes doivent être appliquées dans une perspective fonctionnelle et sont toutes obligatoires pour identifier une SCE.

- Règles d'identification des Lectures de contrôles internes (LCI) :



- ☐ Le sous-processus lit un groupe de données de contrôle.
- ☐ Le sous-processus ne lit qu'un groupe de données. Si plus d'un groupe de données est lu, il faut alors compter un LCI pour chaque groupe de données.
- ☐ Le sous-processus n'entre, ne sort ou n'écrit pas de données.
- ☐ Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres LCIs associés aux mêmes processus.

Toutes les règles de comptage précédentes doivent être appliquées selon une perspective fonctionnelle et sont toutes obligatoires pour identifier un LCI.

- Règles d'identification des Écritures de contrôles internes (ECI) :

- ☐ Le sous-processus écrit un groupe de données de contrôle.
- ☐ Le sous-processus n'écrit qu'un groupe de données. Si plus d'un groupe de données est écrit, il faut alors compter un ECI pour chaque groupe de données.
- ☐ Le sous-processus n'entre, ne sort ou ne lit pas de données.
- ☐ Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres ECIs associés aux mêmes processus.

Toutes les règles de comptage précédentes doivent être appliquées selon une perspective fonctionnelle et sont toutes obligatoires pour identifier un ECI.

## Règles d'assignation des points

Le nombre de points assignés aux composants de contrôle des transactions (ECE, SCE, ECI ou LCI) dépend du nombre de DEs. Les règles suivantes s'appliquent au comptage des DEs :

- Pour les ECEs et SCEs :
  - ☐ Compter une DE pour chaque unité d'information non réursive, identifiable par l'utilisateur, qui *traverse la frontière* de l'application.
- Pour un LCI :
  - ☐ Compter une DE pour chaque unité d'information non réursive, identifiable par l'utilisateur, qui *est lue* à partir d'un GCM, GCL, GDI ou GDE incluant les clefs.
- Pour un ECI :
  - ☐ Compter une DE pour chaque unité d'information non réursive, identifiable par l'utilisateur, qui *est écrite* dans un GCM ou GDI, incluant les clefs.

Lorsque le nombre de DEs est déterminé, la table 6 (section 5.2.2.2) est utilisée pour traduire le nombre de DEs en points.

## 7. EXEMPLE DE COMPTAGE

Exemple simplifié de suivi et de contrôle d'un four

### 7.1 Spécifications des besoins

Groupe de données :

- Groupe de données maintenues par l'application :
  - Registre de messages<sup>14</sup>, champs :
    - Type de message (allumer ou fermer un élément du four)
    - Inscription de l'heure (l'heure d'envoi du message)
  - Champs temporaires :
    - Température désirée (la température que le four doit atteindre et maintenir)
    - Envoi de messages d'avertissement (nombre d'avertissements des messages envoyés)
- Groupes de données lues par l'application :
  - Constantes fonctionnelles utilisées par l'application :
    - Délai d'avertissement (une période de réchauffement continue avant qu'un message d'avertissement soit généré).
    - Seuil de fermeture (nombre de messages d'avertissement envoyés avant que la fermeture se fasse automatiquement).

Processus :

- Contrôle de température :
  - La température du four est reçue à partir d'un capteur.
  - Un message est envoyé à l'élément chauffant, allumer ou éteindre selon la température du four et la température désirée.
  - Une nouvelle entrée est créée dans le registre des messages.

---

<sup>14</sup> Exemple de messages du registre de messages :

| <u>Type de message</u>    | <u>Registre de l'heure</u> |
|---------------------------|----------------------------|
| Allumer l'élément du four | 12:00                      |
| Allumer l'élément du four | 12:01                      |
| Allumer l'élément du four | 12:02                      |
| ...                       |                            |
| Fermer l'élément du four  | 12:15                      |

- Suivi du four

Le processus est déclenché par une horloge externe (en dehors de la frontière de l'application)

En utilisant le délai d'avertissement, le sous-processus recherche les conditions de surchauffe dans le registre des messages<sup>15</sup>. Données utilisées par ce sous-processus :

- Registre des messages : Type de message et inscription de l'heure
  - Groupe de contrôle uniquement lu : Seuil de fermeture et délai d'avertissement

Si une surchauffe est détectée, alors envoyer un message d'avertissement au panneau de contrôle.

Si une surchauffe est détectée, alors il faut incrémenter le compteur de Messages d'avertissement (nombre de messages d'avertissement envoyés), sinon réinitialiser le compteur de Messages d'avertissement

Si le seuil de fermeture est atteint, alors envoyer un message de fermeture du four<sup>16</sup>

Pour garder cet exemple simple, un certain nombre de spécifications de l'utilisateur ne sont pas mentionnées, incluant le processus de mise à jour de la température désirée. La spécification de mise à jour de la température désirée n'est pas illustrée, mais est présumée lorsque l'on compte les groupes de données.

---

<sup>15</sup> Par exemple, si Délai d'avertissement = 30 minutes, alors un message d'avertissement sera envoyé seulement si le registre des messages montre «allumer l'élément» pour 30 minutes. Si le four ne peut atteindre la température désirée en 30 minutes, il y a probablement une défaillance et un message d'avertissement doit être envoyé.

<sup>16</sup> Par exemple, si le seuil de fermeture = 3 messages d'avertissement, alors le four se fermera automatiquement après 3 messages d'avertissement.

## 7.2 Comptage des PFEs

### 7.2.1 Identification des groupes de données

À partir des exigences de l'utilisateur, on peut identifier trois groupes de données :

- Un registre de messages  
Champs : type de message, inscription de l'heure
- Groupe de données à occurrence simple mis à jour  
Champs : température désirée, message d'avertissement envoyé
- Groupe de données à occurrence simple uniquement lues  
Champs : délai d'avertissement, seuil de fermeture

Ces groupes de données sont des données de contrôle parce qu'ils sont utilisés pour contrôler le comportement de l'application. Les règles des GCM/GCL sont utilisées pour leur comptage.

### 7.2.2 Comptage des composants de données de contrôle

La table suivante montre une analyse des règles pour déterminer si le registre des messages est un GCM.

#### Registre des messages

| Règles de comptage du GCM  | Est-ce que la règle s'applique?  |
|--|--|
| Le groupe est un groupe de données logique ou un groupe de données à occurrence simple | Oui. Le registre des messages est un groupe de données logique.  |
| Le groupe de données est maintenu à l'intérieur des frontières de l'application        | Oui, le processus du contrôle de température met à jour le registre des messages.  |
| Le groupe de données vit plus d'une transaction  | Oui, le processus de suivi du four utilise un registre de messages chaque fois que le message est envoyé à l'élément chauffant. Un certain nombre d'entrées enregistrées (de transactions multiples) sont alors utilisées par le processus de suivi du four. |
| Le groupe de données identifié n'a pas été comptabilisé comme un GCL, un GDI ou un GDE | Oui. La règle s'applique parce que le registre de messages n'a pas été compté comme un GCL, un GDI ou un GDE.  |

Sur la base de l'analyse précédente, le registre de messages est un GCM. La table suivante montre les règles de comptage des SLDs et des DEs, et la façon d'assigner les points.

#### SLD

| Règle de comptage du SLD  | Est-ce que la règle s'applique?                 |
|---|---|
| Compter un SLD pour chaque groupe de GCM optionnel ou obligatoire     | Le registre des messages n'a pas de sous-groupe |
| Ou  | Dans ce cas on ne compte qu'un SLD              |
| S'il n'y a pas de sous-groupes, compter un SLD seulement pour le GCM. |   |

1 RET

#### Type de message et inscription de l'heure

| Règles de comptage du DE   | DEs   |
|--|---|
| Compter chaque rubrique unique du GCM, reconnaissable par l'utilisateur, et non récursive.   | - Type de message<br>- Inscription de l'heure |
| Compter une DE pour chaque clef étrangère.   | Il n'y a pas de données de ce type.           |
| Compter les techniques de réalisation comme une seule DE pour le groupe entier de rubriques. | Il n'y a pas de champs de ce type.            |

2 DEs

Sur la base de la table des GDIs [IFPUG, 1994] aussi utilisée pour les GCMs à occurrences multiples, le nombre de SLDs (1), le nombre de DEs (2), le nombre de point est 7 pour le registre des messages.

#### Groupe de données de contrôle mis à jour à occurrence simple

Selon les spécifications des besoins, il y a un groupe de données de contrôle à occurrence simple mis à jour. La table suivante montre l'analyse des règles pour déterminer s'il s'agit d'un GCM.

Champs :

- Température désirée (la température que le four doit atteindre et maintenir)
- Nombre de message d'avertissement envoyé (nombre de messages d'avertissement envoyés)

| Règles de comptage du GCM  | Est-ce que la règle s'applique?  |
|--|--|
| Le groupe est un groupe de données logique ou un groupe de données à occurrence simple.      | Oui. C'est un groupe de données à occurrence simple.   |
| Le groupe de données est maintenu à l'intérieur des frontières de l'application.             | Oui. L'envoi du message d'avertissement a été mis à jour par le processus : suivi du four. La température est maintenue par un autre processus de l'application. Ce processus n'a pas été mentionné dans les spécifications requises pour des fins de simplicité. Cependant il est présumé dans le comptage des données. |
| Le groupe de données vit pour plus d'une transaction.  | Oui. La température désirée et le nombre de message d'avertissement sont utilisés par plusieurs des processus de contrôle de la température et de suivi du four.   |
| Le groupe de données identifiés n'a pas été compté comme GCL, GDI ou GDI pour l'application. | Oui. La règle s'applique parce que la température désirée et le nombre de message d'avertissement n'ont pas été compté comme GCL, GDI ou GDI.  |

Sur la base des analyses de la table précédente, le groupe de données à occurrence simple est un GCM. La table suivante et la formule montrent les règles d'analyse pour compter les DEs, ainsi que l'assignation des points.

### Température désirée et nombre de messages d'avertissements

| Règles de comptage du DE   | DEs   |
|--|---|
| Compter chaque rubrique unique du GCM, reconnaissable par l'utilisateur, et non récursive.   | - Température désirée<br>- Nombre de message d'avertissement envoyé |
| Compter une DE pour chaque clef étrangère.   | Il n'y a pas de données de ce type.                                 |
| Compter les techniques de réalisation comme une seule DE pour le groupe entier de rubriques. | Il n'y a pas de champs de ce type.                                  |

2 DEs

Note : Un GCM à occurrence simple a normalement un grand nombres de DEs parce qu'ils incluent tous les champs à occurrence simple maintenus par l'application comptée. Le GCM précédent a seulement deux DEs dans le but de simplifier cet exemple.

### Assignation des points pour le groupe de données identifié :

$$\text{Points} = ((\text{nombre de DEs} / 5) + 5)$$

$$\text{Points} = ((2/5) + 5) = 5.4$$

### Groupe de données à occurrence simple lues seulement

Selon les spécifications, il n'y a qu'un groupe de données de contrôle à occurrence simple en lecture seulement. La table suivante montre les règles d'analyse pour déterminer s'il s'agit d'un GCL.

Champs :

- Délai d'avertissement (température qui doit être maintenue avant qu'un message d'avertissement soit transmis).
- Seuil de fermeture (nombre de messages d'avertissement envoyés avant que la fermeture se fasse automatiquement).

| Règles de comptage du GCL  | Est-ce que les règles s'appliquent?  |
|--|--|
| Le groupe est un groupe de données logique ou un groupe de données à occurrence simple.        | Oui. Le groupe est un groupe de données à occurrence simple.   |
| Le groupe de données n'est pas mis à jour par l'application comptée.                           | Oui.   |
| Le groupe de données est référencé par l'application comptée.                                  | Oui. Délai d'avertissement et seuil de fermeture sont référencés par le processus de suivi du four.                              |
| Le groupe de données vit plus d'une transaction.   | Oui. Le délai d'avertissement et le seuil de fermeture sont utilisés par plusieurs transactions du processus de suivi du four.   |
| Le groupe de données identifié n'a pas été compté comme un GCM, GDI ou GDE pour l'application. | Oui. La règle s'applique parce que le délai d'avertissement et le seuil de fermeture n'ont pas été compté comme GCM, GDI ou GDE. |

Sur la base des analyses de la table, le groupe de données à occurrence simple est un GCL. La table suivante et la formule montrent les règles d'analyse pour compter les DEs, ainsi que l'assignation des points.

#### Délai d'avertissement et dispositif d'arrêt

| Règles de comptage du DE   | DEs   |
|--|---|
| Compter chaque rubrique unique du GCL, reconnaissable par l'utilisateur, et non récursive.   | - Délai d'avertissement.<br>- Seuil de fermeture. |
| Compter une DE pour chaque clef étrangère.   | Il n'y a pas de données de ce type.               |
| Compter les techniques de réalisation comme une seule DE pour le groupe entier de rubriques. | Il n'y a pas de champs de ce type.                |

2 DEs

Note : Un GCL à occurrence simple possède normalement beaucoup plus de DEs parce qu'il inclut tous les champs à occurrence simple de l'application comptée.



### **Assigination des points pour un groupe de données :**

$$\text{Points} = (\text{nombre de DEs} / 5)$$

$$\text{Points} = (2/5) = 0.4$$

#### 7.2.3 Identification des processus

À partir des besoins fonctionnels on peut identifier deux processus :

- Contrôle de la température.
- Suivi du four.

Ces processus sont des processus de contrôle parce qu'ils contrôlent le comportement de l'application. Les règles des ECE, SCE, LCI et ECI sont utilisées pour les compter.

#### 7.2.4 Comptage des composants transactionnels

Pour chaque processus de contrôle identifié, nous commençons par identifier les sous-processus potentiels et, pour chaque sous-processus potentiel, il faut déterminer le type de sous-processus, le nombre de DEs et les points correspondants. L'analyse du premier processus suit.

#### **Processus : contrôle de la température**

Les sous-processus potentiels de ces processus de contrôle sont :

1. Recevoir du capteur la température du four.
2. Lire la température désirée dans groupe de données à occurrence simple.
3. Comparer la température du four avec la température désirée.
4. Envoyer un message d'allumer ou d'éteindre dépendant du résultat du sous-processus 3.
5. Mettre à jour le registre des messages.

Le premier sous-processus (recevoir la température du four du capteur) reçoit des données de contrôle de l'extérieur de la frontière de l'application. En conséquence, c'est un ECE potentiel. La table suivante montre les règles d'analyse pour déterminer si ce sous-processus est un ECE. Si c'est le cas, les règles de comptages des DEs et les règles d'assignation des points seront appliquées.

### Recevoir du capteur la température du four

| Règles de comptage de l'ECE  | Est-ce que les règles s'appliquent?   |
|--|---|
| Le sous-processus reçoit un groupe de données de contrôle de l'extérieur des frontières de l'application.  | Oui. La température du four, envoyée par un capteur, est reçue par le sous-processus. |
| Le sous-processus ne reçoit qu'un groupe de données. Si plus d'un groupe de données est reçu, il faut alors compter une ECE pour chaque groupe de données.       | Oui. Un seul groupe de données de contrôle est reçu.                                  |
| Le sous-processus ne sort, ne lit ou n'écrit pas de données.   | Oui.  |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres ECEs associées aux mêmes processus. | Oui.  |

Sur la base des analyses de la table, la réception de la température du four est un ECE.

### Température du four

| Règles de comptage du DE   | DE                    |
|--|-----------------------|
| Compter chaque rubrique non récursive, et reconnaissable par l'utilisateur traversant la frontière de l'application. | - Température du four |

1 DE

### Assignment des points

Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et du nombre de DEs (1), le nombre de points du sous-processus de température reçu est 1.

### Lire la température désirée dans le GCM

| Règles de comptage du LCI   | Est-ce que les règles s'appliquent?   |
|---|---|
| Le sous-processus lit un groupe de données de contrôle.   | Oui. La température désirée est lu à partir du groupe à occurrence simple mis à jour (GCM). |
| Le sous-processus lit seulement un groupe de données. Si plus d'un groupe de données différent est lu, compter seulement un LCI pour chaque groupe de données.  | Oui. Seule une donnée de contrôle est lue.  |
| Le sous-processus ne reçoit, ne sort ou n'écrit pas de données.   | Oui.  |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres LCIs associés aux mêmes processus. | Oui.  |

### Température désirée

| Règles de comptage du DE   | DE                    |
|--|-----------------------|
| Compter une DE pour chaque rubrique non réursive, et reconnaissable par l'utilisateur lu par le sous-processus à partir d'un GDI, un GDE, un GCM ou un GCL, incluant les clefs étrangères. | - Température désirée |

1 DE

Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et le nombre de DEs (1), la lecture de la température désirée a 1 point.

Le troisième sous-processus compare deux températures (température du four et température désirée). Il ne reçoit pas, n'envoie pas, ne lit pas ou n'écrit pas de données de contrôle. En conséquence, aucune fonction transactionnelle sera comptée pour cette fonctionnalité.

Le quatrième sous-processus envoie des données de contrôle à l'extérieur de la frontière de l'application. En conséquence, c'est un SCE potentiel. La table suivante montre les règles qui servent à déterminer si ce sous-processus est un SCE. Si c'est le cas, les règles de comptages des DEs et les règles d'assignation des points seront appliquées.

**Envoyer un message : allumer ou éteindre l'élément du four**

| Règles de comptage SCE  | Est-ce que les règles s'appliquent?  |
|---|--|
| Le sous-processus envoie un groupe de données de contrôle de l'extérieur des frontières de l'application  | Oui. Un message est envoyé à l'extérieur de la frontière de l'application. |
| Le sous-processus n'envoie qu'un groupe de données. Si plus d'un groupe de données est envoyé, il faut alors compter une SCE pour chaque groupe de données.     | Oui. Seulement un type de message est envoyé.                              |
| Le sous-processus n'entre, ne lit ou n'écrit pas de données   | Oui.   |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres SCEs associées aux mêmes processus | Oui.   |

**Type d'action**

| Règles de comptage du DE   | DE   |
|--|--|
| Compter chaque rubrique non récursive, et reconnaissable par l'utilisateur traversant la frontière de l'application. | - Type d'action (allumer ou éteindre un élément) |

1 DE

Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et le nombre de DEs (1), le processus d'envoi de messages à l'élément du four a 1 point.

Le cinquième et dernier sous-processus, mettre à jour le Registre des messages, écrit des données de contrôle. C'est un ECI potentiel. La table suivante montre les règles qui servent à déterminer si ce sous-processus est un ECI. Si c'est le cas, les règles de comptages des DEs et les règles d'assignation des points seront appliquées.

#### Mettre à jour le Registre des messages

| Règles de comptage ECI  | Est-ce que les règles s'appliquent?   |
|---|---|
| Le sous-processus écrit un groupe de données de contrôle.   | Oui. Un enregistrement est créé dans le registre des messages pour chaque message envoyé. |
| Le sous-processus n'écrit qu'un groupe de données. Si plus d'un groupe de données est écrit, il faut alors compter une ECI pour chaque groupe de données.       | Oui. Seulement le registre des messages est mis à jour.                                   |
| Le sous-processus ne reçoit, ne sort ou ne lit pas de données.  | Oui.  |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres SCEs associées aux mêmes processus | Oui.  |

#### Type de message et inscription de l'heure

| Règles de comptage du DE   | Des   |
|--|---|
| Compter un DE pour chaque rubrique non réursive, et reconnaissable par l'utilisateur écrit par le sous-processus dans un GDI ou un GDE, incluant les clefs étrangères. | - Type de message<br>- Inscription de l'heure |

2 DEs

Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et le nombre de DEs (2), le processus de mise à jour du registre de messages a 1 point.

## **Élimination des sous-processus dupliqués du Contrôle de la température**

Tel que mentionné dans la procédure de comptage (étape 6 et section 6.4), si pour une raison quelconque, le même sous-processus a été identifié plus d'une fois dans le même processus, seulement un sous-processus doit être conservé.

Dans le processus de contrôle de la température, tous les sous-processus identifiés sont uniques parce qu'ils ont des traitements uniques et des DEs uniques.

Voici l'analyse du second processus :

### **Processus : Suivi du four**

Les sous-processus potentiels de ce processus sont :

- 1 Recevoir un déclencheur d'une horloge externe.
- 2 Lire les délais d'avertissement et de seuil de fermeture .
- 3 Lire le type de message et l'inscription de l'heure à partir du registre des messages.
- 4 Rechercher les conditions de surchauffe en utilisant les données lues.
- 5 Envoyer un message de surchauffe (seulement si des messages d'allumage sont envoyé pour plus longtemps que le délai d'avertissement).
- 6 Lire le nombre de messages d'avertissement envoyé à partir du GCM.
- 7 Mettre à jour le nombre de messages d'avertissement envoyés (incrémenter si surchauffe, remettre à zéro autrement)
- 8 Envoyer un message de fermeture du four si le seuil de fermeture est atteint (si le nombre de messages d'avertissement envoyé est plus grand que le seuil de fermeture).

Le premier sous-processus (recevoir le déclencheur d'une horloge externe) reçoit des données de contrôle à partir de l'extérieur de la frontière de l'application. En conséquence, c'est un ECE potentiel. La table suivante montre l'analyse pour déterminer si le sous-processus est ou non un ECE. Dans ce cas, les règles de comptage des DEs et les points qui s'y rattachent s'appliquent.

### Recevoir un déclencheur d'une horloge externe

| Règles de comptage de l'ECE  | Est-ce que les règles s'appliquent?  |
|--|--|
| Le sous-processus reçoit un groupe de données de contrôle de l'extérieur des frontières de l'application.  | Oui. Un déclencheur est reçu par le sous-processus à partir de l'extérieur de la frontière de l'application. |
| Le sous-processus ne reçoit qu'un groupe de données. Si plus d'un groupe de données est reçu, il faut alors compter une ECE pour chaque groupe de données.       | Oui. Seulement un déclencheur est reçu.  |
| Le sous-processus ne sort, ne lit ou n'écrit pas de données.   | Oui.   |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres ECEs associées aux mêmes processus. | Oui.   |

### Déclencheur

| Règles de comptage du DE   | DE            |
|--|---------------|
| Compter chaque rubrique non récursive, et reconnaissable par l'utilisateur traversant la frontière de l'application. | - Déclencheur |

1 DE

Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et le nombre de DEs (1), le sous-processus de déclenchement à partir d'une horloge externe reçu a 1 point.

Le second processus lit des données. Conséquemment il est potentiellement un LCI.

### Lire le délai d'avertissement et le seuil de fermeture à partir du GCL

| Règles de comptage pour la LCI  | Est-ce que les règles s'appliquent?  |
|---|--|
| Le sous-processus lit un groupe de données de contrôle.   | Oui. La lecture du délai d'avertissement et du seuil de fermeture sont lus par le GCL. |
| Le sous-processus lit seulement un groupe de données. Si plus d'un groupe de données différent est lu, compter seulement un LCI pour chaque groupe de données.  | Oui. Seulement un groupe de données(GCL) est lu.                                       |
| Le sous-processus ne reçoit, ne sort ou n'écrit pas de données.   | Oui.   |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres LCIs associés aux mêmes processus. | Oui.   |

### Délai d'avertissement et seuil de fermeture

| Règle de comptage du DE   | DEs   |
|---|---|
| Compter un DE pour chaque rubrique non réursive, reconnaissable par l'utilisateur et lu à partir d'un GDI, un GDE, un GCM ou un GCL, incluant les clefs étrangères. | - Délai d'avertissement.<br>- Seuil de fermeture. |

2 DEs

Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et du nombre de DEs (2), le sous-processus a 1 point.



Le troisième sous-processus potentiel lit un groupe de données différent : un registre de messages GCM

#### **Lire les types de message et les inscriptions de l'heure dans le registres des messages**

| <b>Règles de comptage pour la LCI</b>   | <b>Est-ce que les règles s'appliquent?</b>   |
|---|--|
| Le sous-processus lit un groupe de données de contrôle.   | Oui. Les types de message et l'inscription de l'heure sont lus à partir du GCM (Registre des messages) |
| Le sous-processus lit seulement un groupe de données. Si plus d'un groupe de données différent est lu, compter seulement un LCI pour chaque groupe de données.  | Oui. Seulement un groupe de données de contrôle (Registre des messages GCM) est lu.                    |
| Le sous-processus ne reçoit, ne sort ou n'écrit pas de données.   | Oui.   |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres LCIs associés aux mêmes processus. | Oui.   |

#### **Type de message et inscription de l'heure**

| <b>Règles de comptage du DE</b>  | <b>DEs</b>                                    |
|--|---|
| Compter un DE pour chaque rubrique non récursive, reconnaissable par l'utilisateur et lu à partir d'un GDI, un GDE, un GCM ou un GCL, incluant les clefs étrangères. | - Type de message<br>- Inscription de l'heure |

2 DEs

Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et du nombre de DEs (2), le sous-processus a 1 point.

Le quatrième sous-processus «Rechercher les conditions de surchauffe en utilisant les données lues» ne reçoit pas, n'envoie pas, ne lit pas ou n'écrit pas de données de contrôle. En conséquence, aucune fonction n'est comptée pour cette fonctionnalité.

Le cinquième sous-processus envoie des données de contrôle à l'extérieur de la frontière de l'application. En conséquence, c'est un SCE potentiel.

#### Envoyer un message d'avertissement en cas de surchauffe

| Règles de comptage SCE  | Est-ce que les règles s'appliquent?  |
|---|--|
| Le sous-processus envoie un groupe de données de contrôle de l'extérieur des frontières de l'application  | Oui. Un message est envoyé à l'extérieur de la frontière de l'application. |
| Le sous-processus n'envoie qu'un groupe de données. Si plus d'un groupe de données est envoyé, il faut alors compter une SCE pour chaque groupe de données.     | Oui. Seulement un type de message est envoyé.                              |
| Le sous-processus n'entre, ne lit ou n'écrit pas de données   | Oui.   |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres SCEs associées aux mêmes processus | Oui.   |

#### Message

| Règle de comptage du DE  | DE        |
|--|-----------|
| Compter chaque rubrique non récursive, et reconnaissable par l'utilisateur traversant la frontière de l'application. | - Message |

1 DE

Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et du nombre de DEs (1), le sous-processus a 1 point.

Le sixième sous-processus lit les données. En conséquence c'est un LCI potentiel.

**Lire le nombre de messages d'avertissement envoyé dans le GCM à occurrence simple**

| Règles de comptage LCI  | Est-ce que les règles s'appliquent?   |
|---|---|
| Le sous-processus lit un groupe de données de contrôle.   | Oui. Les messages d'avertissement envoyés sont lus à partir du GCM.               |
| Le sous-processus lit seulement un groupe de données. Si plus d'un groupe de données différent est lu, compter seulement un LCI pour chaque groupe de données.  | Oui. Seulement un groupe de données de contrôle (GCM à occurrence simple) est lu. |
| Le sous-processus ne reçoit, ne sort ou n'écrit pas de données.   | Oui.  |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres LCIs associés aux mêmes processus. | Oui.  |

**Envoi d'un message d'avertissement**

| Règle de comptage du DE   | DE                                   |
|---|--------------------------------------|
| Compter un DE pour chaque rubrique non réursive, reconnaissable par l'utilisateur et lu à partir d'un GDI, un GDE, un GCM ou un GCL, incluant les clefs étrangères. | - Envoi d'un message d'avertissement |

1 DE

Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et du nombre de DEs (1), le sous-processus a 1 point.

Le septième sous-processus met à jour un GCM à occurrence simple. En conséquence, c'est un ECI potentiel.

**Mettre à jour le nombre de messages d'avertissement transmis**

| <b>Règles de comptage de l'ECI</b>  | <b>Est-ce que les règles s'appliquent?</b>                       |
|---|--|
| Le sous-processus écrit un groupe de données de contrôle.   | Oui. Le GCM à occurrence simple est mis à jour                   |
| Le sous-processus n'écrit qu'un groupe de données. Si plus d'un groupe de données est écrit, il faut alors compter une ECI pour chaque groupe de données.       | Oui. Seulement le simple GCM à occurrence simple est mis à jour. |
| Le sous-processus ne reçoit, ne sort ou ne lit pas de données.  | Oui.   |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres SCEs associées aux mêmes processus | Oui.   |

**Message**

| <b>Règle de comptage du DE</b>  | <b>DE</b> |
|---|-----------|
| Compter un DE pour chaque rubrique non récursive, reconnaissable par l'utilisateur et écrit dans un GDI ou lu dans un GDE, incluant les clefs étrangères. | - Message |

1 DE

Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et du nombre de DEs (1), le sous-processus a 1 point.

Le huitième sous-processus envoie des données de contrôle à l'extérieur de la frontière de l'application. En conséquence, c'est une SCE potentielle.

#### Envoyer un message de fermeture s'il atteint le seuil

| Règles de comptage SCE  | Est-ce que les règles s'appliquent?  |
|---|--|
| Le sous-processus envoie un groupe de données de contrôle de l'extérieur des frontières de l'application  | Oui. Un message est envoyé à l'extérieur de la frontière de l'application. |
| Le sous-processus n'envoie qu'un groupe de données. Si plus d'un groupe de données est envoyé, il faut alors compter une SCE pour chaque groupe de données.     | Oui. Seulement un type de message est envoyé.                              |
| Le sous-processus n'entre, ne lit ou n'écrit pas de données   | Oui.   |
| Le sous-processus est unique, c'est-à-dire que les traitements et données élémentaires identifiés sont différents des autres SCEs associées aux mêmes processus | Oui.   |

#### Message

| Règle de comptage du DE  | DE        |
|--|-----------|
| Compter chaque rubrique non récursive, et reconnaissable par l'utilisateur traversant la frontière de l'application. | - Message |

1 DE

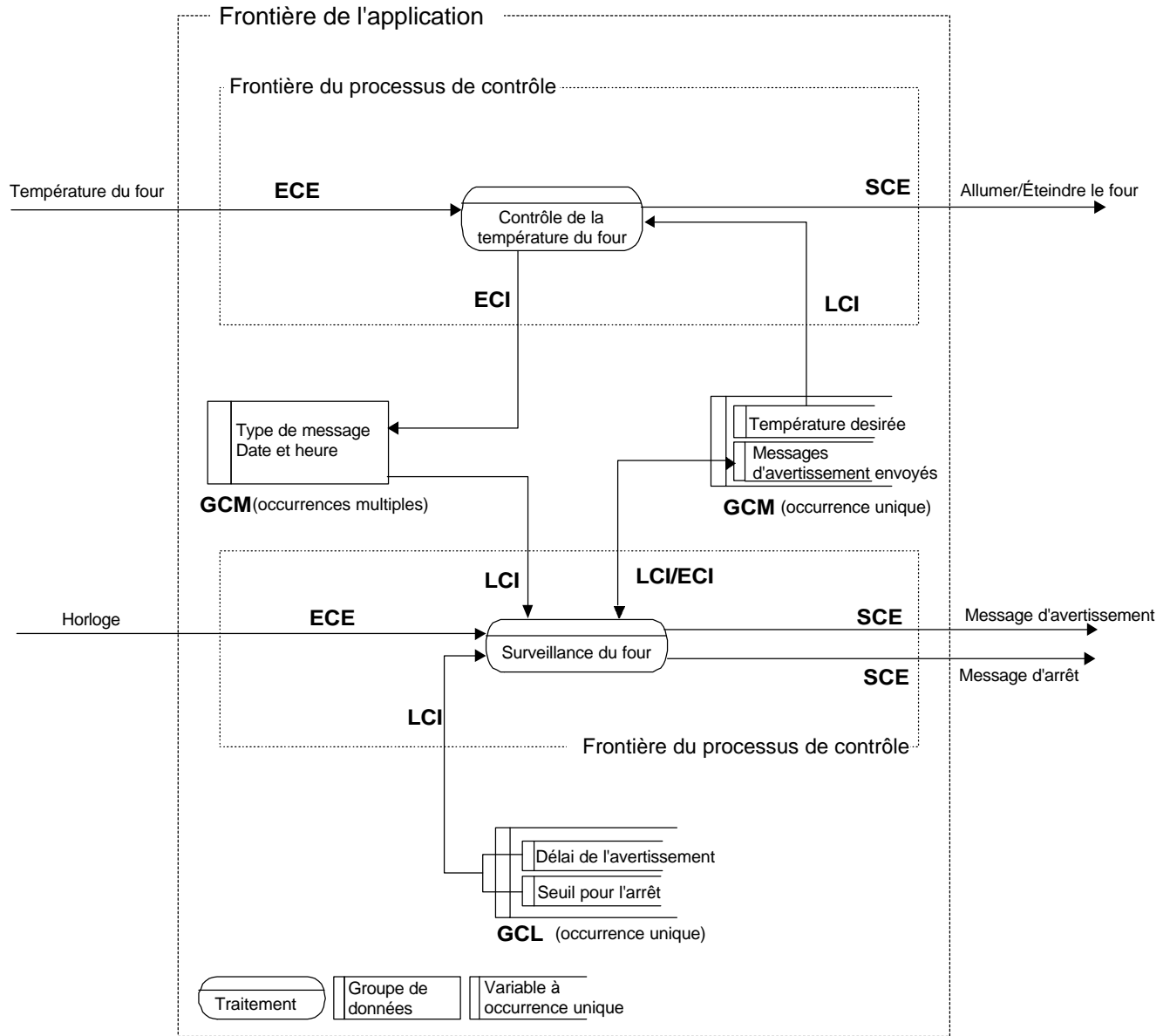
Sur la base de la table d'assignation des composants de contrôle transactionnels (table 6) et du nombre de DEs (1), le sous-processus a 1 point.

#### Élimination des sous-processus dupliqués du processus «Suivi du four»

Tel que mentionné dans la procédure de comptage (étape 6 dans la section 6.4), pour chaque processus de contrôle, les sous-processus dupliqués doivent être enlevés. Dans le processus de suivi du four, tous les sous-processus sont uniques parce qu'ils ont des traitements uniques et des DEs uniques.

### 7.2.5 Résumé de l'exemple de comptage

L'exemple de comptage sommaire est présenté ci-joint dans un format graphique :



**Figure 4 – Diagramme de l'exemple du four**

La table suivante montre les totaux de tous les composants servant à déterminer le total des points non ajustés.

| Description de la fonction  | Composant | GDR/<br>SLD | DE | Point       |
|---|-----------|-------------|----|-------------|
|   |           |             |    |             |
| <b>Composant de données de contrôle</b>                               |           |             |    |             |
| Registre des messages <sup>17</sup>                                   | GCM       | 1           | 2  | 7.0         |
| Groupe de données à occurrence simple mis à jour <sup>18</sup>        | GCM       | ND          | 2  | 5.4         |
| Groupe de données à occurrence simple lues seulement <sup>19</sup>    | GCL       | ND          | 2  | 0.4         |
| <b>Points totaux pour les données :</b>                               |           |             |    | <b>12.8</b> |
|   |           |             |    |             |
| <b>Composants de contrôle transactionnels</b>                         |           |             |    |             |
| Contrôle de la température  |           |             |    |             |
| Recevoir du capteur la température du four                            | ECE       | ND          | 1  | 1           |
| Lire la température désirée du GCM                                    | LCI       | ND          | 1  | 1           |
| Envoyer un message : allumer ou éteindre                              | SCE       | ND          | 1  | 1           |
| Mettre à jour le Registre des messages                                | ECI       | ND          | 2  | 1           |
| <b>Points pour les sous-processus de contrôle de la température :</b> |           |             |    | <b>4</b>    |
|   |           |             |    |             |
| Suivi du four   |           |             |    |             |
| Recevoir un déclencheur d'une horloge externe                         | ECE       | ND          | 1  | 1           |
| Lire le délai d'avertissement et le seuil de fermeture                | LCI       | ND          | 2  | 1           |
| Lire les types de messages et les inscriptions de l'heure             | LCI       | ND          | 2  | 1           |
| Envoyer un message d'avertissement en cas de surchauffe               | SCE       | ND          | 1  | 1           |
| Lire le nombre de messages d'avertissement envoyé                     | LCI       | ND          | 1  | 1           |

<sup>17</sup> Champs d'inscription des messages: type de Message et enregistrement de l'heure.

<sup>18</sup> Champ du groupe de données à occurrence simple mis à jour: température désirée et message d'avertissement envoyé.

<sup>19</sup> Groupe de données à occurrence simple lues seulement: délai d'avertissement et seuil de fermeture.

|  |     |    |   |             |
|--|-----|----|---|-------------|
| dans le GCM à occurrence simple                              |     |    |   |             |
| Mettre à jour le nombre de messages d'avertissement transmis | ECI | ND | 1 | 1           |
| Envoyer un message de fermeture s'il atteint le seuil        | SCE | ND | 1 | 1           |
| <b>Points du processus de suivi du four :</b>                |     |    |   | <b>7</b>    |
|  |     |    |   |             |
|  |     |    |   |             |
| <b>Total de points non ajustés :</b>                         |     |    |   | <b>23.8</b> |
|  |     |    |   |             |

**Table 7 – Exemple du four : sommaire du comptage**

Dans cet exemple, nous avons tenté de présenter une documentation de comptage très détaillé. En industrie, pour produire un comptage, il faut beaucoup moins de détails. Si l'effort requis pour le comptage est une préoccupation, le niveau de détail de la table 7 est suffisant.

Dans le but de conserver un exemple simple, nous avons utilisé une très petite application. Une application temps réel typique a beaucoup plus de sous-processus (ECE, SCE, ECI et LCI) par processus de contrôle. Dans cet exemple, il y a tout au plus 7 sous-processus associés à un processus. Lors des comptages en industrie, certaines applications temps réel avaient jusqu'à 50 sous-processus associés à un seul processus. De plus, les applications en temps réel typiques ont des groupes de contrôle à occurrence simple beaucoup plus complexes (plus de DEs). Nous avons trouvé des groupes de contrôle à occurrence simple avec plus d'une centaine de DEs dans certains sites industriels.

## **8. COMMENTAIRES SUR LES PFEs**

Ce rapport a présenté les concepts des PFEs, les définitions et les procédures, de même que des exemples. Un nombre d'applications industrielles en temps réel ont déjà été comptées en utilisant la technique des PFEs. Nos réflexions et les commentaires du milieu sont rapportés ici.

### **8.1 Facilité de compréhension**

Tous les comptages PFE ont été fait avec le support des experts des applications en temps réel (des personnes familières avec les applications comptées dans leur entreprise). Lorsque les experts des applications ont compris les définitions des composants de contrôle, il n'ont pas de difficultés à les identifier. En fait, après une journée entière de comptage avec la technique PFE, les experts des applications peuvent faire le comptage avec une assistance minimale de l'expert PFE.



## 8.2 Effort de comptage

Sur la base de nos essais industriels, l'effort de comptage des PFEs est similaire à celui des points de fonction. D'une part, plus de fonctions doivent être identifiées avec la technique des PFEs; d'autre part, c'est plus simple<sup>20</sup>, puisque les fonctions de contrôle transactionnels sont identifiées plus facilement.

## 8.3 Importance de la documentation

Une source adéquate d'information fonctionnelle doit être disponible pour compter une application. L'information fonctionnelle est fournie par des experts de l'application et par la documentation de l'application. Tout comme pour les points de fonction, la technique PFE dépend de la qualité de l'information fonctionnelle disponible assurer la précision du comptage.

## 8.4 Comptage des PFEs tôt dans le projet

Il est possible de compter les composants de contrôle assez tôt dans le cycle de développement en assumant que les spécifications fonctionnelles sont documentées. Malgré les nouveaux composants, le niveau de détail des PFEs est similaire à celui des points de fonction. Pour utiliser les points de fonction assez tôt dans le cycle de développement, il faut normalement estimer le compte puisque toutes les spécifications fonctionnelles ne sont pas disponibles. Avec les années, un certain nombre de méthodes de comptage estimées ont été développées pour les points de fonction; des méthodes d'approximation similaires peuvent être développées pour les PFEs<sup>21</sup>

## 9. SOMMAIRE

Le développement d'une mesure fonctionnelle des logiciels en temps réel a été un défi important à rencontrer. Une telle mesure peut être utile à l'étalonnage de la productivité et le développement de modèles de productivité basés sur la taille fonctionnelle. Une des premières étapes, dans la réalisation d'étalonnage des logiciels en temps réel et de modèles d'estimation, est de s'assurer que les spécialistes de la mesure travaillent avec les mêmes règles. Dans une certaine mesure, ceci devrait permettre aux spécialistes de construire à partir des acquis, plutôt que de repartir à zéro avec leurs propres règles et interprétation

---

<sup>20</sup> Les composants de type transactionnels sont plus simples parce qu'elles sont associées à seulement un sous-processus. Un composant transactionnel de gestion peut comprendre plusieurs sous-processus, et les grouper en processus élémentaires peut prendre du temps.

<sup>21</sup> On ne peut évidemment espérer, dès maintenant, des méthodes d'approximation des PFEs aussi matures que celles des points de fonction.

des règles existantes. La mise en place d'un comité international permettant de réguler la pratique est donc une étape importante.

## GLOSSAIRE

Les définitions suivantes sont utilisées avec la technique des points de fonction complémentaires (PFE). Elles sont un complément aux définitions du glossaire d'IFPUG (IFPUG, 1994).

**Composants données :** Les fonctionnalités fournies à l'utilisateur pour répondre à ses spécifications en terme de données. Dans ce document on utilise le plus souvent cette expression avec le mot contrôle. Les composants PFE sont : le GCM, le GCL, le GDI et le GDE.

**Composants transactionnels :** Les fonctionnalités fournies à l'utilisateur pour traiter les données d'une application. Les composants transactionnels sont l'ECE, la SCE, l'ECI, la LCI, l'Entrée, la Sortie et l'Interrogation.

**Déclencheur :** Un événement qui initie un processus selon une perspective fonctionnelle. L'événement vient de l'extérieur de la frontière de l'application. Un événement temporel peut être un déclencheur.

**Données de contrôle :** Données utilisées par l'application pour contrôler directement ou indirectement le comportement d'une application ou d'une machine.

**Données de gestion :** Données utilisées par l'application pour supporter l'utilisateur dans sa gestion de l'information administrative et d'affaires.

**Groupe de données :** Données identifiées et groupées ensemble d'un point de vue fonctionnel.

**Mis à jour :** Possibilité de modifier les données. (Aussi **maintenu**)

**Occurrences multiples :** Groupe de données pouvant avoir plusieurs enregistrements du même type.

**Occurrence simple :** Groupe de données ayant un seul enregistrement du même type.

**Perspective fonctionnelle :** Point de vue de la fonctionnalité livrée par l'application; exclu toute considération technique ou de mise en œuvre.

**Processus de contrôle :** Un processus qui contrôle directement ou indirectement le comportement d'une application ou d'une machine.

**Processus de gestion :** Processus dont le but est de supporter l'utilisateur pour gérer ses informations, particulièrement les informations de gestion et d'administration.

**Processus :** Un ensemble d'opérations ou d'activités qui agissent sur des données d'entrée pour produire un résultat.

**Sous-processus :** Dans le contexte de ce rapport, c'est la plus petite étape d'un processus qui soit identifiable par utilisateur dans une perspective fonctionnelle que ce soit une entrée, une sortie, une lecture ou une écriture.

**Transaction :** Tout traitement associé à une occurrence d'un déclencheur externe.

**Utilisateur :** Une personne, une application ou une machine qui interagit avec l'application mesurée.

## RÉFÉRENCES

Abran, A., and Robillard, P.N., *Function Point Analysis, An Empirical Study of its Measurement Processes*, IEEE Transactions on Software Engineering, Vol. 22, no. 12, pp. 895-909, December 1996.

Albrecht, A.J., *AD/M Productivity Measurement and Estimate Validation*, IBM Corporate Information Systems, IBM Corp., Purchase, N.Y., May 1984.

Conte, S.D., Shen, V.Y., and Dunsmore, H.E., *Software Engineering Metrics and Models*, Benjamin Cummins Publishing, 396 pages, 1986.

Cooling, J.E., *Software Design for Real-Time Systems*, Chapman and Hall, 1991.

Desharnais, J.-M., *Statistical Analysis on the Productivity of Data Processing with Development Projects using the Function Point Technique*, Université du Québec à Montréal, 1988.

Galea, S., *The Boeing Company : 3D Function Point Extensions, V2.0, Release 1.0*, Boeing Information and Support Services, Research and Technology Software Engineering, June 1995.

Grady, R.B., *Practical software metrics for project management and process improvement*, Prentice Hall, New Jersey, 270 pages, 1992.

Hetzel, B., *Making Software Measurement Work*, QEB Publishing Group, 290 pages, 1993.

IEEE, *IEEE Standard Computer Dictionary : A compilation of IEEE Standard Computer Glossaries*, IEEE Std 610-1990, The Institute of Electrical and Electronics Engineers, Inc., New York, NY, 1990.

IFPUG, *Function Point Counting Practices Manual, Release 4.0*, International Function Point Users Group - IFPUG, Westerville, Ohio, 1994.

Illingworth, V. (editor), *Dictionary of Computing*, Oxford University Press, 3rd edition, 510 pages, 1991.

Ince, D.C., *History and industrial applications*, in Fenton, N.E., *Software Metrics : A Rigorous Approach*, Chapman & Hall, UK, 337 pages, 1991.

Jones, C., *A Short History of Function Points and Feature Points*, Software Productivity Research, Inc., Cambridge, Mass, 1988.

Jones, C., *Applied Software Measurement - Assuring Productivity and Quality*, McGraw-Hill, 493 pages, 1991.

Jones, C., *Applied Software Measurement - Assuring Productivity and Quality*, McGraw-Hill, 618 pages, 1996.

Kan, S.H., *Metrics and Models in Software Quality Engineering*, Addison-Wesley, 344 pages, 1993.

Kemerer, C. F., *An Empirical Validation of Software Cost Estimation Models*, Communications of the ACM, Vol. 30, no 5, pp. 416-429, May 1987.

Laplante, P., *Real-Time Systems Design and Analysis : An Engineer's Handbook*, The Institute of Electrical and Electronics Engineers Inc., New York, NY, 339 pages, 1993.

Stankovic, J.A. and Ramamritham, K., *Tutorial Hard Real-Time Systems*, IEEE Computer Society Press, Washington D.C., 618 pages, 1988.

Whitmire, S.A., *3-D Function Points : Scientific and Real-Time Extensions to Function Points*, Proceedings of the 1992 Pacific Northwest Software Quality Conference, 1992.

***Pour plus d'information sur les PFEs :***

**Jean-Marc Desharnais, M.Sc., C.F.P.S.**

Laboratoire de métriques appliqués en gestion du logiciel

175, rue Doody

Chambly (Québec) Canada J3L 1K7

Tél. : (450) 658-9650

E-mail : [desharnais.jean-marc@uqam.ca](mailto:desharnais.jean-marc@uqam.ca)

Web site : <http://www.lmagl.qc.ca>

Monsieur DESHARNAIS, est un spécialiste dans la mesure en gestion des logiciels depuis 1986. Il a réalisé un bon nombre de recherches liées à la mesure en génie logiciel couvrant des aspects d'évaluation de la productivité, d'estimation et de budgétisation. M. Desharnais a aussi évalué les niveaux de productivité de plusieurs organisations et mis en marche des programmes de mesures couvrant l'estimation, la productivité, la qualité et la budgétisation de l'entretien. M. Desharnais a une maîtrise en gestion des logiciels et une maîtrise en administration publique. Il est présentement directeur de compte au LMAGL. Il est certifié comme spécialiste des points de fonction depuis 1993 par IFPUG. Il a participé depuis 1988 à divers comités spécialisés du International Function Point Users Group (IFPUG) dont le Counting Practice Committee et le Management Reporting Committee. Il est présentement membre du Education Committee.

**Denis St-Pierre, M.Sc., C.F.P.S.**

Laboratoire de métriques appliqués en gestion du logiciel

175, rue Doody

Chambly (Québec) Canada J3L 1K7

Tél. : (514) 762-1955

E-mail : [dstpierr@crim.ca](mailto:dstpierr@crim.ca)

Web site : <http://www.lmagl.qc.ca>

Monsieur ST-PIERRE est un consultant senior en métriques de logiciels. En 1988, Il a obtenu une maîtrise portant sur les points de fonction (moyenne parfaite). Il a agi à titre de consultant auprès de corporations nord américaines, asiatiques et européennes. Récemment, il a été impliqué dans l'un des plus gros contrat d'impartition au monde (5 milliards de dollars). Monsieur St-Pierre a publié plusieurs articles scientifiques et industriels. Il a entre autre contribué significativement à plusieurs publications de l'International Function Point Users Group. Il siège d'ailleurs depuis 1993 sur le comité de normes de cette organisation. Voici quelques clients récents de Monsieur St-Pierre : Bell Canada, CBA (Australie), EDS, IBM, Nissan et Nortel. Dans les 10 dernières années, M. St-Pierre a été impliqué dans l'estimation, la comparaison corporative (benchmarking), les programmes de mesures, l'assurance qualité, la formation, les outils de mesures automatique...

**Alain Abran, Ph.D.**

Laboratoire de recherche en gestion des logiciels

Département d'informatique

Université du Québec à Montréal

C.P. 8888, succursale centre-ville

Montréal (Québec) Canada H3C 3P8

Tél. : (514) 987-3000 (8900)

Fax : (514) 987-8477

E-mail : [abran.alain@uqam.ca](mailto:abran.alain@uqam.ca)

Web site : [http://www.info.uqam.ca/Labo\\_Recherche/lrgl.html](http://www.info.uqam.ca/Labo_Recherche/lrgl.html)

Monsieur ABRAN est actuellement professeur en génie logiciel et en informatique de gestion à l'Université du Québec à Montréal. Il enseigne au niveau des études graduées et il est le directeur du Laboratoire de recherche en gestion des logiciels. Membre actif au sein de plusieurs associations professionnelles, il est également président du Centre d'Intérêt sur les Métriques (CIM) et il dirige plusieurs projets de recherche conjoints avec l'Allemagne, le Japon, l'Australie et les États-Unis dans le domaine du génie logiciel. M. Abran était jusqu'à juin 1993 directeur des normes et mesures de la productivité à la division des systèmes et services de soutien du Montréal Trust. Le programme de mesures qu'il a conçu et implanté dans le secteur de la maintenance des logiciels au Montréal Trust a gagné en 1993 un prix «*Best of the Best*» du Quality Assurance Institute aux États Unis. M. Abran est diplômé de l'Université d'Ottawa en informatique, en génie électrique (M.Sc.), en administration (M.B.A.) et il a complété un Ph.D. en génie informatique à l'École Polytechnique de Montréal. Il poursuit actuellement des recherches sur les mesures de productivité, les modèles d'estimation et les méthodes de validation et de vérification des mesures et instruments de mesure en génie logiciel.

**Marcela Maya, M.Sc., C.F.P.S.**

Laboratoire de recherche en gestion des logiciels

Département d'informatique

Université du Québec a Montréal

C.P. 8888, succursale centre-ville

Montréal (Québec) Canada H3C 3P8

Tél. : (514) 987-3000 (3820)

Fax : (514) 987-8477

E-mail : [maya.marcela@uqam.ca](mailto:maya.marcela@uqam.ca)

Web site : [http://www.info.uqam.ca/Labo\\_Recherche/lrgl.html](http://www.info.uqam.ca/Labo_Recherche/lrgl.html)

Madame MAYA travaille actuellement comme assistante de recherche au Laboratoire de Recherche en Gestion des Logiciels de l'Université du Québec à Montréal, Montréal, Canada. Avec plus de 10 années d'expérience dans la gestion de systèmes d'information, elle a joint le Laboratoire après avoir fini le programme de maîtrise en informatique de gestion à la même Université. Ses sujets d'intérêt sont les métriques appliquées à

l'entretien du logiciel et aux systèmes temps réel, la réutilisation du logiciel et les modèles de productivité. Elle est spécialiste certifié de la technique des points de fonction depuis 1996.

**Pierre Bourque, M.Sc., C.F.P.S.**

Laboratoire de recherche en gestion des logiciels

Département d'informatique

Université du Québec à Montréal

C.P. 8888, succursale centre-ville

Montréal (Québec) Canada H3C 3P8

Tél. : (514) 987-3000 (0315)

Fax : (514) 987-8477

E-mail : [Bourque.pierre@uqam.ca](mailto:Bourque.pierre@uqam.ca)

Web site : [http://www.info.uqam.ca/Labo\\_Recherche/lrgl.html](http://www.info.uqam.ca/Labo_Recherche/lrgl.html)

Monsieur BOURQUE est le directeur adjoint du Laboratoire de recherche en gestion des logiciels à l'Université du Québec à Montréal. La mission du laboratoire consiste à mener des recherches dans les domaines du génie logiciel et de la métrologie du logiciel. Ces recherches portent notamment sur la technique de l'analyse des points de fonction. M. Bourque détient une Maîtrise en Mathématiques (Informatique) de l'Université de Sherbrooke (1988). Son mémoire de maîtrise traite d'une application de la métrologie du logiciel. De 1987 à 1995, M. Bourque a oeuvré à la Banque Nationale du Canada où il a contribué à l'avancement de dossiers corporatifs d'évaluation technologique, de modélisation conceptuelle et physique de données, d'administration de données et de génie logiciel. M. Bourque a plusieurs publications scientifiques à son actif dont un certain nombre relatives à la métrologie du logiciel. M. Bourque est présentement co-éditeur d'une initiative parrainée par la direction de la IEEE Computer Society dont le but est de définir le corpus des connaissances spécifiques au domaine du génie logiciel. M. Bourque est également certifié à titre de spécialiste en analyse des points de fonction (CFPS) par l'IFPUG.