

comme activité première de faire la synthèse des besoins exprimés et par la suite, de les transformer en spécifications.

### **6.2.2.3 Ressources**

Le déroulement du processus requiert des ressources. Ces ressources sont, entre autres :

- La main d'œuvre
- Les matériaux (moins applicable au génie logiciel)
- Les outils (logiciel et matériel)
- Les normes
- Les méthodes
- Les livres de référence
- Les produits intermédiaires

Il est à noter qu'un produit intermédiaire issu d'un sous-processus est considéré, lui-même, comme une ressource lorsqu'il est utilisé par un autre sous-processus.

### **6.2.2.4 Produit**

Globalement, le produit final est l'objectif ou le résultat attendu du processus du génie. Dans le cas du génie logiciel, le logiciel et le guide d'utilisation sont les produits finaux issus du processus de développement. Le processus de développement se découpe en sous-processus. Ceux-ci génèrent un ou plusieurs produits intermédiaires qui seront utilisés comme des ressources par les sous-processus suivants.

### **6.2.2.5 Contrôle**

Le contrôle est le pôle orienté management et ingénierie où des décisions sont prises pour gérer le processus, le corriger et l'améliorer. Le contrôle s'assure que le processus se déroule normalement afin que les objectifs (les buts) soient atteints dans le respect des

contraintes. Le contrôle doit s'assurer que le projet se déroule conformément aux plans, aux budgets et aux échéanciers prévus. Le contrôle permet aussi d'améliorer le processus afin, entre autres, d'augmenter la productivité ou la qualité du produit. Afin de jouer pleinement son rôle, le contrôle doit s'alimenter en information provenant de mesures prises au niveau du processus. L'analyse des mesures se fait en conjonction avec les buts et contraintes afin de détecter des écarts. Si les écarts nécessitent un ajustement, le contrôle déclenchera une action corrective vers le processus.

#### **6.2.2.6 Mesures**

Les mesures sont les données (nominale, ordinale, intervalle, ratio, absolue) recueillies au niveau du processus ou des produits. Le contrôle procède à l'analyse des données recueillies en considérant les buts et les contraintes et ceci pour détecter des écarts. Si des écarts sont constatés, le contrôle va générer une action vers le processus afin de diminuer ces écarts.

#### **6.2.2.7 Action**

L'action est une intervention issue du contrôle afin de diriger le processus selon les plans prévus ou de le rectifier suite à la mesure d'un écart. Le contrôle, par la prise de nouvelles mesures, pourra vérifier si l'action a porté fruit.

#### **6.2.2.8 Buts**

Les buts sont principalement des objectifs d'affaires de l'entreprise. Les objectifs suivants sont, entre autres, des exemples de buts qui influencent le contrôle et implicitement le processus d'ingénierie.

- Réalisation de profits, rentabilité de l'entreprise
- Production à faible coûts
  - Favoriser la réutilisation

- Faciliter l'effort de maintenance
- Préserver ou accroître la part de marché
- Satisfaction des clients
- Qualité des produits

#### 6.2.2.9 Contraintes

Le contrôle doit aussi tenir compte de contraintes contextuelles au projet. Les budgets alloués au projet, les délais de livraison, l'arrimage du projet à un autre projet sont, entre autres, des contraintes contextuelles.

#### 6.2.3 Principes et modèle

Dans cette section, nous analysons les 34 propositions retenues en fonction des éléments du modèle de l'ingénierie présenté par Moore (2006).

Proposition no.1: Align incentives for developer and customer

L'explication fournie par Davis (1995) vise essentiellement les développeurs. Ceux-ci doivent être motivés et responsabilisés dans l'atteinte des objectifs, tels le respect des échéanciers. Davis souligne que le management devrait mettre en place des mesures positives et négatives pour l'atteinte ou non des objectifs. La proposition est étroitement liée à la proposition no. 24 (*maintain clear accountability of results*). La mise en place d'incitatifs ne peut se faire que si les résultats et les responsabilités de chacun sont clairement définies et communiqués.

Le sens explicite de la proposition souligne que le processus doit intégrer des incitatifs pour les développeurs afin d'atteindre les résultats prévus. D'une façon implicite, le management veut atteindre ses buts et éviter, à titre d'exemple, les dépassements des

échanciers et des budgets (conséquences). Le contrôle doit intervenir pour vérifier (à l'aide de mesures) si les objectifs sont atteints ou non. Le contrôle, par l'action, soulignera l'atteinte des objectifs (par des récompenses) ou les écarts (réprimandes). Les incitatifs peuvent être vus comme des contraintes sur les individus (ressources)

Tableau CLXIX

Éléments d'ingénierie impliqués pour la proposition no.1

Align incentives for developer and customer	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> <li>• Action</li> <li>• Mesure</li> <li>• Contrainte</li> <li>• Ressource</li> <li>• But</li> </ul>

Proposition no.2: Apply and use quantitative measurements in decision making

Le contrôle doit s'alimenter en information en provenance du processus à l'aide des mesures. La proposition souligne explicitement d'appliquer et d'utiliser les mesures dans la prise de décisions qui est le cœur du contrôle. La proposition est associée à l'élément mesure ainsi qu'à l'élément contrôle du modèle où se fait la prise de décision basée, entre autres, sur l'analyse des mesures. D'une façon implicite, il y aura une action de générée par la prise de décision.

Tableau CLXX

Éléments d'ingénierie impliqués pour la proposition 2

Apply and use quantitative measurements in decision making	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Mesure</li> <li>• Contrôle</li> </ul>	<ul style="list-style-type: none"> <li>• Action</li> </ul>

Proposition no.3: Build software so that it needs a short user manual

Cette proposition concerne le produit final, le logiciel. Le thème de la facilité d'utilisation est central à cette proposition, même si implicite. Si le logiciel est simple à utiliser, le recours à des explications supplémentaires du manuel d'utilisation ne sera pas requis. Le terme « build » implique explicitement le processus. Également, la proposition souligne une contrainte explicite à l'effet que le manuel d'utilisation doit être court. De plus, la proposition mentionne explicitement les deux produits finaux du processus, le logiciel et le guide d'utilisation. La facilité d'utilisation peut être aussi vue, implicitement, comme un besoin exprimé par le client.

Tableau CLXXI

Éléments d'ingénierie impliqués pour la proposition no.3

Build software so that it needs a short user manual	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• Produit</li> <li>• Contrainte</li> </ul>	<ul style="list-style-type: none"> <li>• Besoin</li> </ul>

Proposition no.4: Build with and for reuse

La proposition comporte deux aspects. Le premier concerne le « build with » qui est explicitement une contrainte imposée sur le processus afin de construire le logiciel avec des composants déjà faits. Le deuxième aspect est le « for reuse » qui est une contrainte sur les produits réalisés. Ces produits doivent être conçus pour être réutilisables et agir implicitement comme ressources (implicitement) à un autre projet. D'une façon implicite, la réutilisation peut favoriser une réduction des coûts et la réduction des délais de développement. La réutilisation peut être un but du management et des architectes.

Tableau CLXXII

Éléments d'ingénierie impliqués pour la proposition no.4

Build with and for reuse	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Contrainte</li> <li>• Produit</li> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• But</li> <li>• Ressource</li> </ul>

Proposition no.6: Define software artifacts rigorously

La proposition souligne d'une façon explicite par le qualificatif « rigorously » que chacun des produits intermédiaires (artefact) doit être défini avec rigueur. Les produits intermédiaires sont définis par le processus, mais l'aspect de la rigueur sera imposé et vérifié par le volet contrôle. D'une façon implicite, cette proposition souligne qu'en définissant avec rigueur les produits intermédiaires, des imprécisions pourraient être corrigées, des oublis pourraient être détectés. Implicitement, des conséquences sur les coûts et les délais pourraient être diminuées (but).

Tableau CLXXIII

Éléments d'ingénierie impliqués pour la proposition no.6

Define software artifacts rigorously	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• Produit</li> <li>• Contrôle</li> </ul>	<ul style="list-style-type: none"> <li>• But</li> <li>• Mesure</li> <li>• Action</li> </ul>

Proposition no.9: Design for maintenance

Les activités de maintenance peuvent être coûteuses si le logiciel a été développé sans avoir tenu compte de sa facilité future à être modifié. Pour le management, la facilité

d'entretien du logiciel peut être un but. Celui-ci permettra de réaliser des modifications au logiciel dans des délais plus courts et ainsi réduire les coûts de l'activité de maintenance. La proposition souligne explicitement que le processus de design doit tenir compte des aspects de la maintenance, c'est une forme de contrainte. D'une façon implicite, la proposition touche les ressources, telles les normes, les gabarits et autres, qui permettent, entre autres, d'uniformiser le style des produits et en faciliter la maintenance par la suite. La facilité de maintenance est aussi une caractéristique intrinsèque du produit.

Tableau CLXXIV

Éléments d'ingénierie impliqués pour la proposition no.9

Design for maintenance	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• But</li> <li>• Contrainte</li> </ul>	<ul style="list-style-type: none"> <li>• Produit</li> <li>• Ressource</li> </ul>

Proposition no.10: Determine requirements now

La proposition souligne explicitement de déterminer les exigences et ce « maintenant ». Le premier volet s'associe avec l'élément processus du modèle, tandis que le volet « maintenant » est explicitement une contrainte. D'une façon implicite, la contrainte implique le contrôle afin de vérifier si les exigences sont bien définies à l'étape de la spécification des exigences. Laisser en suspens des exigences peut être considéré comme une incertitude et un risque supplémentaire sur le projet, ce que le contrôle veut éliminer ou réduire.

Tableau CLXXV

Éléments d'ingénierie impliqués pour la proposition no.10

Determine requirements now	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• Contrainte</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> </ul>

Proposition no.11: Don't overstrain your hardware

Davis (1995) souligne que la surcharge du matériel peut entraîner des modifications importantes et coûteuses au logiciel, une fois complété. La proposition souligne explicitement une contrainte sur le produit final. Les modifications à faire afin de rendre le logiciel moins gourmand en ressources matérielles peuvent être coûteuses, retarder sa livraison et même être la cause d'un échec. D'une façon implicite, la proposition ne mentionne pas le produit, mais il y a un lien avec le produit final, puisqu'il est en cause dans la surcharge du matériel. De plus, le contrôle est impliqué afin de s'assurer que le logiciel développé tourne adéquatement sur le matériel prévu.

Tableau CLXXVI

Éléments d'ingénierie impliqués pour la proposition no.11

Don't overstrain your hardware	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Contrainte</li> </ul>	<ul style="list-style-type: none"> <li>• Produit</li> <li>• Contrôle</li> </ul>

Proposition no.13: Don't try to retrofit quality

La qualité ne peut s'ajouter au logiciel une fois le développement terminé. Le processus doit donc intégrer des activités d'assurance qualité à partir du début du processus jusqu'à



la fin. La proposition est donc explicitement orientée vers le processus. D'une façon implicite, la proposition touche à la qualité des produits intermédiaires et finaux. Le contrôle doit s'assurer que les activités d'assurance qualité soient réalisées au niveau du processus et que des mesures soient prises pour s'en assurer.

Tableau CLXXVII

Éléments d'ingénierie impliqués pour la proposition no.13

Don't try to retrofit quality	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• Produit</li> <li>• Contrôle</li> <li>• Mesure</li> </ul>

Proposition no.14: Don't write your own test plans

Davis (1995) souligne qu'aucun développeur ne devrait écrire les plans de test des composants logiciels qu'il a lui-même développés. La proposition exprime explicitement une contrainte sur le processus. D'une façon implicite, le contrôle doit s'assurer que cette contrainte est suivie au niveau du processus.

Tableau CLXXVIII

Éléments d'ingénierie impliqués pour la proposition no.14

Don't write your own test plans	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Contrainte</li> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> </ul>

**Proposition no.15: Establish a software process that provides flexibility**

Le processus mis en place doit avoir la flexibilité de pouvoir s'adapter au contexte spécifique du projet et de l'organisation. La flexibilité peut être vue comme un objectif du management. La proposition vise explicitement un but à atteindre. Également, la proposition mentionne l'élément processus. Il y a aussi un aspect de contrainte sur le processus afin qu'il s'ajuste au contexte du projet.

Tableau CLXXIX

Éléments d'ingénierie impliqués pour la proposition no.15

Establish a software process that provides flexibility	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• But</li> <li>• Processus</li> <li>• Contrainte</li> </ul>	

**Proposition no.16: Fix requirements specification error now**

Davis (1995) souligne que les erreurs au niveau des exigences doivent être corrigées dès l'étape de la spécification des exigences du logiciel et ce, avant même d'entreprendre les étapes subséquentes. Nous identifions explicitement dans cette proposition deux éléments principaux. En premier lieu, le terme « fix » représente une action. Deuxièmement, le terme « now » est une contrainte sur le processus. D'une façon implicite, toute la boucle de contrôle est impliquée. Des mesures doivent être prises pour constater les erreurs dans les spécifications. Par la suite, le contrôle, face à l'écart détecté et à la contrainte (du « now ») doit enclencher une action corrective (« fix ») vers le processus afin de faire corriger les écarts.

Tableau CLXXX

Éléments d'ingénierie impliqués pour la proposition no.16

Fix requirements specification error now	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Action</li> <li>• Contrainte</li> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> <li>• Mesure</li> </ul>

Proposition no.17: Give product to customers early

Davis (1995) souligne que le client peut s'impliquer dans le processus en utilisant des versions prototypes du logiciel. Cependant, l'explication de l'auteur ne stipule pas que le processus doit livrer le produit final plus tôt.

D'une façon explicite, la proposition demande au processus de fournir des prototypes au client. Également, la proposition ajoute une contrainte (« early ») au processus. D'une façon implicite, la proposition est liée au contrôle qui doit vérifier si le processus a bien réalisé le prototype et si la contrainte est respectée.

Tableau CLXXXI

Éléments d'ingénierie impliqués pour la proposition no.17

Give product to customers early	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• Contrainte</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> </ul>

Proposition no.19: Grow systems incrementally

Davis (1995) souligne que le développement du logiciel devrait s'effectuer d'une façon incrémentale par l'ajout successif de fonctions. La proposition peut s'associer explicitement au processus. D'une façon implicite, la voie incrémentale permet au management de diminuer les risques du projet, ce qui peut être considéré comme un but. Également, le développement incrémental peut contenir un aspect de contrainte sur le processus.

Tableau CLXXXII

Éléments d'ingénierie impliqués pour la proposition no.19

Grow systems incrementally	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• But</li> <li>• Contrainte</li> </ul>

Proposition no.20: Implement a disciplined approach and improve it continuously

La proposition comporte une contrainte explicite sur le processus : « disciplined » et une autre sur le contrôle : « continuously ». Le processus doit se dérouler d'une façon disciplinée. La proposition souligne, également, qu'il doit y avoir amélioration continue du processus. Cette contrainte positive porte sur le contrôle qui, à l'aide des mesures, procède à des ajustements, sous forme d'actions, au niveau du processus. D'une façon implicite, le contrôle doit effectuer des mesures afin de vérifier si le processus s'exécute de façon disciplinée et, les cas échéant, des actions peuvent s'en suivre. L'amélioration du processus peut être un but du management afin de diminuer les coûts et les délais de développement et d'augmenter la qualité du produit. En bout de ligne, la qualité du produit devrait être influencée par l'amélioration du processus.

Tableau CLXXXIII

Éléments d'ingénierie impliqués pour la proposition no.20

Implement a disciplined approach and improve it continuously	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Contrainte</li> <li>• Processus</li> <li>• Contrôle</li> </ul>	<ul style="list-style-type: none"> <li>• But</li> <li>• Mesure</li> <li>• Action</li> <li>• Produit</li> </ul>

Proposition no.21: Invest in the understanding of the problem

Le terme investir souligne qu'il faut déployer les efforts et les ressources nécessaires pour comprendre le problème à résoudre. La proposition est associée à l'élément processus du modèle. Le processus doit comporter l'étape d'élucidation du problème. D'une façon implicite, il y a des conséquences à ne pas élucider correctement le problème. À titre d'exemple, une mauvaise élucidation peut entraîner des dépassements de coûts, de délais et la non satisfaction du client envers le produit final. Le management doit aussi consacrer (sous forme d'action) les ressources nécessaires pour mettre en application cette proposition.

Tableau CLXXXIV

Éléments d'ingénierie impliqués pour la proposition no.21

Invest in the understanding of the problem	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• Action</li> <li>• Ressource</li> <li>• Contrôle</li> </ul>

**Proposition no.22: Involve the customer**

La proposition souligne que le processus doit impliquer le client. C'est donc une forme de contrainte imposée au processus. Le client est considéré alors comme une ressource pour le processus. D'une façon implicite, le contrôle doit vérifier si le processus a effectivement impliqué le client et ce au moment prévu.

Tableau CLXXXV

Éléments d'ingénierie impliqués pour la proposition no.22

Involve the customer	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• Contrainte</li> <li>• Ressource</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> </ul>

**Proposition no.23: Keep design under intellectual control**

Davis (1995), dans ses commentaires, limite la portée de cette proposition à l'effet que le design doit être fait et documenté de façon à ce que les développeurs puissent le comprendre dans son ensemble. L'auteur identifie des techniques à utiliser telles la conception hiérarchique et les vues multiples. Selon l'explication fournie par l'auteur la dimension du contrôle intellectuel est plutôt faible. Le sens donné par l'auteur est orienté vers la conception à l'aide des techniques suggérées. La proposition est liée explicitement aux éléments processus et contrôle du modèle. D'une façon implicite, la proposition sous-entend une façon de documenter le design, comme un produit intermédiaire. Également, la mesure pourrait être impliquée afin de contrôler le design et mêmes des actions correctives pourraient être déclenchées.

Tableau CLXXXVI

Éléments d'ingénierie impliqués pour la proposition no.23

Keep design under intellectual control	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• Contrôle</li> </ul>	<ul style="list-style-type: none"> <li>• Produit</li> <li>• Mesure</li> <li>• Action</li> </ul>

Proposition no.24: Maintain clear accountability for results

Boehm (1983) souligne que les individus impliqués dans le processus de développement doivent être informés des résultats attendus et de leur responsabilité face à l'atteinte de ceux-ci. Cette proposition est associée au processus et au contrôle. Le processus doit informer les individus des attentes. D'une façon implicite, le contrôle doit s'assurer que le processus informe les individus, donc les ressources.

Tableau CLXXXVII

Éléments d'ingénierie impliqués pour la proposition no.24

Maintain clear accountability for results	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• Contrôle</li> </ul>	<ul style="list-style-type: none"> <li>• Ressources (individus)</li> </ul>

Proposition no.25: Produce software in a stepwise fashion

La proposition souligne explicitement que le logiciel doit être développé à l'aide d'un processus composé d'étapes. D'une façon implicite, le contrôle doit s'assurer qu'il en soit ainsi.

Tableau CLXXXVIII

Éléments d'ingénierie impliqués pour la proposition no.25

Produce software in a stepwise fashion	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> </ul>

Proposition no.26: Quality is the top priority; long term productivity is a natural consequence of high quality

La proposition est explicitement associée à un but du management où la qualité serait la priorité. De plus, la productivité à long terme est liée directement à l'amélioration du processus. D'une façon implicite, cette proposition implique le produit qui devrait intégrer certaines des caractéristiques de qualité. Wiegers (1996) ajoute également, que l'attitude des individus envers la qualité est cruciale. Le contrôle s'assurera que les activités d'assurance qualité sont faites et que les objectifs de qualité sont atteints.

Tableau CLXXXIX

Éléments d'ingénierie impliqués pour la proposition no.26

Quality is the top priority; long term productivity is a natural consequence of high quality	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• But</li> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• Produit</li> <li>• Ressources (individus)</li> <li>• Contrôle</li> </ul>

Proposition no.27: Rotate people through product assurance

Davis (1995) souligne que certaines organisations envoient les développeurs peu performants vers le groupe d'assurance qualité. Davis suggère plutôt que les



organisations mettent en place un processus afin que les meilleurs développeurs fassent un séjour au sein du groupe d'assurance qualité. Ces développeurs pourront par la suite intégrer les aspects de qualité dans le processus de développement et contribuer à l'amélioration de la qualité du processus et en bout de ligne du produit.

La proposition souligne explicitement une contrainte sur le processus de gestion des ressources humaines. De plus, elle souligne explicitement les individus ciblés. D'une façon implicite, le management cherche à améliorer la qualité du processus et la qualité du produit en bout de ligne. Ainsi, c'est un objectif du management.

Tableau CXC

Éléments d'ingénierie impliqués pour la proposition no.27

Rotate (top performer) people through product assurance	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Contrainte</li> <li>• Processus</li> <li>• Ressource</li> </ul>	<ul style="list-style-type: none"> <li>• But</li> <li>• Produit</li> </ul>

Proposition no.28: Since change is inherent to software, plan for it and manage it

La proposition débute par une constatation à l'effet que le changement est inévitable. De ce fait, la proposition souligne qu'il faut le planifier et le gérer. La proposition mentionne explicitement un aspect lié au contrôle. De plus, la constatation est une forme de contrainte. D'une façon implicite, une action est générée par le contrôle dans le but de gérer le changement au niveau du processus. La mesure apportera au contrôle les demandes de changement et le contrôle décidera, en fonction des buts et contraintes, lesquelles seront effectuées.

Tableau CXCI

Éléments d'ingénierie impliqués pour la proposition no.28

Since change is inherent to software, plan for it and manage it	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Contrôle</li> <li>• Contrainte</li> </ul>	<ul style="list-style-type: none"> <li>• Action</li> <li>• Mesure</li> </ul>

Proposition no. 29: Since tradeoffs are inherent to software engineering, make them explicit and document it

Le processus de développement doit documenter explicitement tous les compromis faits. La proposition est associée explicitement au processus et une exerce une forme de contrainte sur celui-ci. D'une façon implicite, le contrôle peut vérifier si le processus documente dans les faits les compromis.

Tableau CXCII

Éléments d'ingénierie impliqués pour la proposition no.29

Since tradeoffs are inherent to software engineering, make them explicit and document it	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• Contrainte</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> </ul>

Proposition no.30: Strive to have a peer, rather than a customer, find a defect

Wiegiers (1996) souligne que les défauts du logiciel devraient être détectés à l'interne par des collègues, plutôt que par le client. La proposition souligne explicitement une contrainte sur le processus du test du logiciel. D'une façon implicite, le contrôle doit

s'assurer de la qualité du processus de test du logiciel et vérifier par des mesures le taux de défauts trouvés par le client suite à l'installation du logiciel.

Tableau CXCIII

Éléments d'ingénierie impliqués pour la proposition no.30

Strive to have a peer, rather than a customer, find a defect	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Contrainte</li> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> </ul>

Proposition no.31: Tailor cost estimation methods

Davis (1995) souligne que les méthodes d'estimation doivent être adaptées au contexte propre de l'organisation et des projets. La proposition a une portée strictement de niveau management. Elle est donc associée à l'élément contrôle. D'une façon implicite, les méthodes d'estimation s'alimentent de mesures.

Tableau CXCIV

Éléments d'ingénierie impliqués pour la proposition no.31

Tailor cost estimation methods	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Contrôle</li> </ul>	<ul style="list-style-type: none"> <li>• Mesures</li> </ul>

Proposition no.32: To improve design, study previous solutions to similar problems

La proposition souligne que le processus de design tient compte de solutions antérieures à un problème similaire dans le but d'améliorer la conception. La proposition est explicitement associée à l'élément processus du modèle. D'une façon implicite,

l'application de la proposition peut faire en sorte de diminuer les temps de développement, d'éviter des erreurs dans le design, de profiter des expériences antérieures et ainsi économiser des ressources. C'est implicitement un but que le management pourrait avoir.

Tableau CXCIV

Éléments d'ingénierie impliqués pour la proposition no.32

To improve design, study previous solutions to similar problems	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• But</li> </ul>	<ul style="list-style-type: none"> <li>• But (conséquences)</li> </ul>

Proposition no.33: Use better and fewer people

Boehm (1983) souligne qu'il y a une grande variance de productivité entre les développeurs, ainsi il suggère d'utiliser les meilleurs développeurs plutôt que plusieurs développeurs de calibre moyen. La proposition souligne explicitement une contrainte sur le management du projet (le contrôle). D'une façon implicite, il y a des motivations (buts) à appliquer cette proposition afin d'accélérer le processus, de diminuer les coûts et d'améliorer la qualité. La proposition a un lien avec l'élément ressources du modèle puisqu'il est question des ressources humaines.

Tableau CXCVI

Éléments d'ingénierie impliqués pour la proposition no.33

Use better and fewer people	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Contrainte</li> <li>• Contrôle</li> <li>• Ressources</li> </ul>	<ul style="list-style-type: none"> <li>• But</li> </ul>

Proposition no.34: Use documentation standards

Davis (1995) souligne que les normes offrent une structure permettant d'organiser clairement les activités. Les produits intermédiaires sont couverts par les normes telles l'IEEE. Les normes et standards sont considérés comme des ressources utilisées par le processus. D'une façon implicite, l'utilisation de normes peut être aussi vue comme une contrainte du management sur le processus.

Tableau CXC VII

Éléments d'ingénierie impliqués pour la proposition no.34

Use documentation standards	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Ressources</li> </ul>	<ul style="list-style-type: none"> <li>• Contrainte</li> <li>• Produit</li> <li>• Processus</li> </ul>

Proposition no.35: Write programs for people first

Davis (1995) souligne que les programmes n'ont plus à être écrits de façon cryptique à cause du coût du matériel. L'auteur souligne que la ressource coûteuse est maintenant la main d'œuvre. Le développeur (et aussi l'organisation) doit garder à l'esprit que le programme est écrit une fois, mais lu et modifié plusieurs fois au cours de sa vie utile. La proposition peut se diviser en deux volets. En premier lieu, « *write programs* » est explicitement lié au processus de développement et au produit. La deuxième partie « *for people first* » est une contrainte sur l'écriture des programmes visant essentiellement la facilité de maintenance par le personnel dans le futur. La lisibilité des programmes permet au personnel de maintenance de réduire l'effort de compréhension des programmes. La proposition est implicitement un but du management qui pourrait désirer diminuer l'effort de maintenance du logiciel.

Tableau CXCVIII

Éléments d'ingénierie impliqués pour la proposition no.35

Write programs for people first	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> <li>• Contrainte</li> <li>• Produit</li> <li>• Ressources (individus)</li> </ul>	<ul style="list-style-type: none"> <li>• But</li> </ul>

Proposition no.36: Know software engineering's techniques before using development tools

La proposition vise essentiellement les développeurs à l'effet qu'ils doivent posséder les connaissances du génie logiciel avant de maximiser les avantages des outils de développement. D'une façon implicite, le management est impliqué afin de s'assurer (lors de l'embauche, par exemple) que les candidats possèdent les connaissances requises.

Tableau CXCIX

Éléments d'ingénierie impliqués pour la proposition no.36

Know software engineering's techniques before using development tools	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Ressources</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> </ul>

Proposition no.37: Select tests based on the likelihood that they will find faults

Cette proposition est explicitement liée au processus de test du logiciel. La préparation des jeux d'essais devrait tenir compte de cette proposition. D'une façon implicite, le

contrôle est impliqué, car il fera la vérification (par des mesures) de la pertinence des cas de test.

Tableau CC

Éléments d'ingénierie impliqués pour la proposition no.37

Select tests based on the likelihood that they will find faults	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> <li>• Mesure</li> </ul>

Proposition no.38: Choose a programming language to assure maintainability

Le choix du langage, selon Davis (1995), a un impact direct sur l'effort de maintenance du logiciel. La proposition comporte deux volets. Le premier volet concerne la sélection d'un langage de programmation, soit une ressource utilisée par le processus. Le second volet de la proposition souligne la motivation de faire une sélection judicieuse du langage. Ce deuxième volet est à la fois un but du management qui permettrait de diminuer les coûts de la maintenance et une contrainte sur le processus. En effet, certains langages jugés intéressants par les programmeurs pourraient être écartés au profit de d'autres. D'une façon implicite, la proposition a un lien avec le produit final.

Tableau CCI

Éléments d'ingénierie impliqués pour la proposition no.38

Choose a programming language to assure maintainability	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Ressources</li> <li>• Processus</li> <li>• But</li> <li>• Contrainte</li> </ul>	<ul style="list-style-type: none"> <li>• Produit</li> </ul>

Proposition no.39: In face of unstructured code, rethink the module and redesign it from scratch

La proposition débute par une constatation (« In face of unstructured code ») qui est le résultat d'une mesure (implicite) faite sur le code d'un programme. Suite à la constatation, la proposition souligne deux actions que le contrôle (implicite) transmet au processus soit de repenser le module en question et de faire une nouvelle conception au lieu de le modifier. Le produit est aussi impliqué par la proposition.

Tableau CCII

Éléments d'ingénierie impliqués pour la proposition no.39

In face of unstructured code, rethink the module and redesign it from scratch	
Sens explicite	Sens implicite
<ul style="list-style-type: none"> <li>• Action</li> <li>• Processus</li> </ul>	<ul style="list-style-type: none"> <li>• Contrôle</li> <li>• Produit</li> <li>• Mesure</li> </ul>

#### 6.2.4 Synthèse

À la phase 3, les propositions ont été classées selon les catégories processus, produit et individu. Une étape de vérification a été réalisée, dans ce volet, afin de s'assurer de la concordance des résultats avec ce classement. Nous constatons que la catégorie processus regroupe plus d'un élément du modèle de Moore (2006). En effet, cette catégorie regroupe les éléments : processus, contrôle, mesure et action. Le modèle raffine donc encore plus notre catégorie processus. Selon le modèle présenté par Moore (2006), le processus se concentre sur les activités de développement et de maintenance du logiciel alors que le contrôle sur les activités de gestion du processus. Nous remarquons que certaines propositions, classées antérieurement dans la catégorie processus, se retrouvent maintenant associées au contrôle.



Dans le même ordre d'idée, une proposition classée dans la catégorie produit, peut être maintenant associée à l'élément ressource du modèle. Un produit intermédiaire peut être une ressource à un processus subséquent.

Le tableau CCIII présente la synthèse de l'association des propositions avec les éléments du modèle présenté par Moore (2006). Le volet contrôle recueille 21 propositions distinctes, les doublons retranchés du compte. Un doublon est lorsque qu'une proposition est associée aux deux volets. Le volet processus recueille 29 propositions, les doublons également retranchés du compte. Les propositions couvrent majoritairement le volet processus par rapport au volet contrôle. Nous constatons également que 20 propositions couvrent les deux volets. Si nous retranchons ces 20 propositions, le volet contrôle ne conserve que deux propositions et le volet processus que 9. Même en retranchant les propositions communes aux deux volets, le volet processus conserve toujours un plus grand nombre de propositions associées. Cette constatation n'est pas surprenante puisque la majorité des propositions se classe sous la catégorie processus et que ces propositions visent essentiellement le processus de développement du logiciel.

Tableau CCIII

Synthèse de l'association explicite des propositions pour chacun des éléments du modèle de l'ingénierie de Moore (2006)

Volet	Éléments du modèle	Propositions associées explicitement
Contrôle	Contrôle	2, 6, 11, 20, 23, 24, 28, 31, 33
	But	9, 15, 26, 32, 38
	Contrainte	3, 4, 9, 10, 11, 14, 15, 16, 17, 20, 22, 27, 28, 29, 30, 33, 35, 38
	Mesure	2
	Action	16, 39
Processus	Processus	1, 3, 4, 6, 9, 10, 13, 14, 15, 16, 17, 19, 20, 21, 22, 23, 24, 25, 26, 27, 29, 30, 32, 35, 37, 38, 39
	Ressource	22, 27, 33, 34, 35, 36, 38
	Produit	3, 4, 6, 35
	Besoins	Aucune

Nous constatons que l'élément mesure ne recueille qu'une seule proposition, malgré le fait que la mesure est essentielle dans l'alimentation en information du contrôle afin de détecter des écarts. De plus, l'élément besoin ne recueille aucune proposition.

Au cours du processus d'association des propositions aux éléments du modèle, nous avons aussi identifié, pour chacune des propositions, les éléments qui sont implicitement impliqués dans la proposition. Le tableau CCIV présente les associations implicites des propositions avec le modèle d'ingénierie.

Tableau CCIV

Synthèse de l'association implicite des propositions pour chacun des éléments du modèle de l'ingénierie de Moore (2006)

Volet	Éléments du modèle	Propositions associées implicitement
Contrôle	Contrôle	1, 10, 13, 14, 16, 17, 21, 22, 25, 26, 29, 36, 37, 39
	But	1, 4, 6, 19, 20, 27, 32, 33, 35
	Contrainte	1, 19, 34
	Mesure	1, 6, 13, 16, 20, 23, 28, 31, 37, 39
	Action	1, 2, 6, 20, 21, 23, 28
Processus	Processus	34
	Ressource	1, 4, 9, 21, 24, 26
	Produit	9, 11, 13, 20, 23, 26, 27, 34, 38, 39
	Besoins	2

Nous constatons que la majorité des propositions se retrouve maintenant du côté du volet contrôle. Une seule proposition implique le processus d'une façon implicite. L'élément besoin recueille une proposition.

### 6.2.5 Conclusion

L'objectif initial de ce premier volet était de vérifier si les 34 propositions retenues de l'analyse couvraient l'ensemble des éléments du modèle d'ingénierie proposé par Moore (2006).

Le volet processus recueille une forte majorité des propositions qui se concentrent essentiellement sur l'élément *processus*. Cette situation peut s'expliquer du fait que la majorité des propositions retenues à la phase 3 se classe dans la catégorie processus. Même si un certain nombre de propositions sont plutôt orientées contrôle, leur nombre demeure faible comparativement aux propositions de type processus. Également, plusieurs propositions ont un sens explicitement orienté développement du logiciel ou soulignant une contrainte sur le processus de développement. Peu de propositions retenues portent sur le contrôle du processus. L'élément *contrôle* ne recueille que neuf propositions à l'opposé de l'élément *processus* qui en comporte 27.

Nous constatons que l'élément *besoin* du volet processus ne recueille aucune proposition. Cette carence est préoccupante compte tenu de l'importance des besoins dans le processus de développement du produit.

Au niveau du volet contrôle, les éléments *mesure* et *action* ne recueillent respectivement qu'une et deux propositions. C'est probablement insuffisant compte tenu que le contrôle est efficace dans la mesure où il est alimenté en données (mesures) afin de pouvoir détecter les écarts et de les corriger sous forme d'action vers le processus.

### **6.3 Couverture des propositions de principes en fonction du corpus des normes de l'IEEE.**

Le deuxième volet de la vérification porte sur le degré de couverture des 34 propositions retenues en fonction du corpus des normes du génie logiciel de l'IEEE.

#### **6.3.1 Objectif**

L'objectif de ce volet est d'évaluer le degré de couverture des propositions retenues à l'ensemble du corpus des normes du génie logiciel. Nous pourrions observer qu'elles

sont les normes qui sont bien supportées par les propositions de principes et celles qui le sont moins.

### **6.3.2 Méthode**

Nous basons une partie de notre démarche sur l'ouvrage de James W. Moore (2006) qui propose une synthèse et une organisation du corpus des normes de l'IEEE. Dans un premier temps, nous traitons des 32 propositions de catégorie processus. À titre de rappel, à la phase 3, les propositions ont été classées selon les catégories suivantes : processus, produit et individu. De plus, lors de cette phase, chacune des propositions retenues de la catégorie processus a fait l'objet d'une analyse plus détaillée afin d'identifier à quel processus et activité de la norme ISO/IEC 12207, la proposition est associée. Dans ce volet, nous allons mettre à profit les résultats de cette analyse et l'associer au travail fait par Moore (2006).

Comme le montre la figure 40, la phase 3 de notre analyse a fait des liens entre les propositions de catégorie processus et les processus de la norme ISO/IEC 12207. De son côté, Moore (2006) a fait le travail de prendre chacun des processus de la norme ISO/IEC 12207 et de les associer aux normes du corpus. En combinant, les deux volets des travaux, nous avons une opportunité intéressante de vérifier le degré de couverture de nos propositions sur l'ensemble des normes du génie logiciel de l'IEEE.

faite à la pièce, puisque le chemin de la norme ISO/IEC 12207 n'est pas possible pour celles-ci.

Suite au processus d'association, un tableau synthèse sera proposé présentant l'association entre les propositions et les normes du corpus. Également, un autre tableau exposera, pour chacune des normes, quelles sont les propositions qui s'y rattachent.

Une synthèse des observations sera faite afin de conclure ce 2<sup>ième</sup> volet de la phase 4.

### 6.3.3 Corpus des normes du génie logiciel de l'IEEE

Dans son récent ouvrage, James W. Moore (2006) présente le corpus des normes IEEE du génie logiciel qui regroupent 36 normes. Moore présente le corpus selon quatre vues distinctes.

La première vue est l'organisation par sujets des normes. Cette vue a été utilisée par l'IEEE en 1997 pour la publication du corpus. Une contrainte de cette vue est qu'une norme ne peut être classée que dans un seul sujet. Les sujets sont au nombre de huit, tel que présenté au tableau CCV.

Tableau CCV

Sujets pour le classement des normes (Traduit de Moore 2006)

1. Documentation	2. Gestion de projet
3. Processus du cycle de vie	4. Réutilisation
5. Mesures	6. Terminologie
7. Plans	8. Outils

En 1999, l'IEEE propose une organisation différente des normes. Les critères sont le niveau de degré de prescription et les objets. Les niveaux de prescription sont représentés par six catégories présentées à la figure 42.

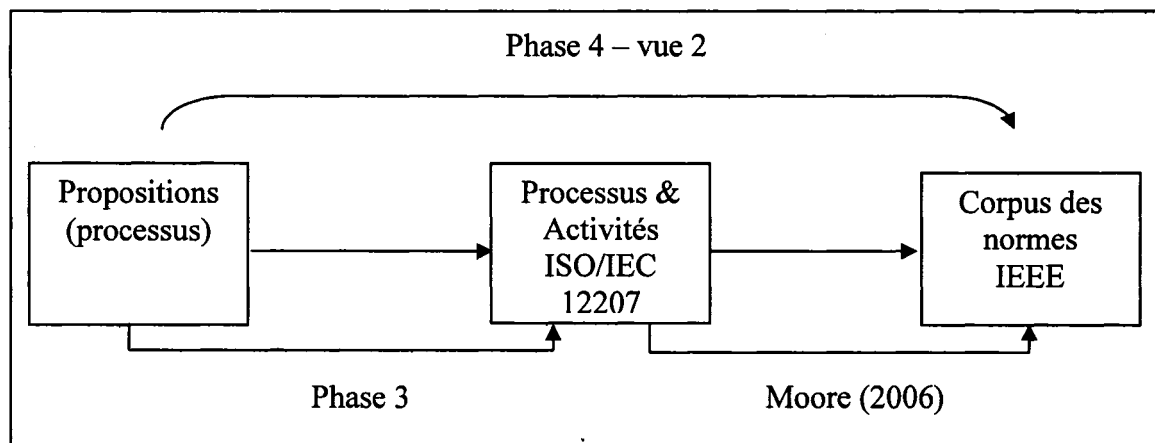


Figure 40 Association des propositions au corpus via la ISO/IEC 12207

La figure 41 présente les éléments de la méthode suivie pour le deuxième volet de la vérification.

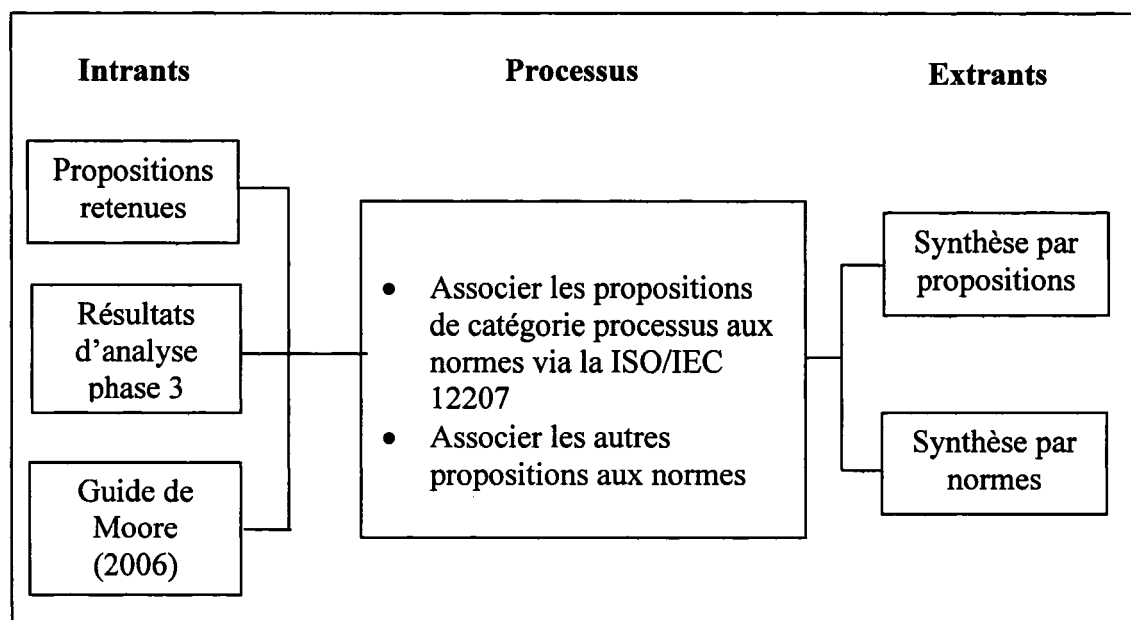


Figure 41 Éléments de la méthode phase 4, volet 2

Du nombre des propositions retenues, seules deux propositions ne sont pas associées à la catégorie processus. Pour ces propositions, la vérification avec le corpus des normes sera

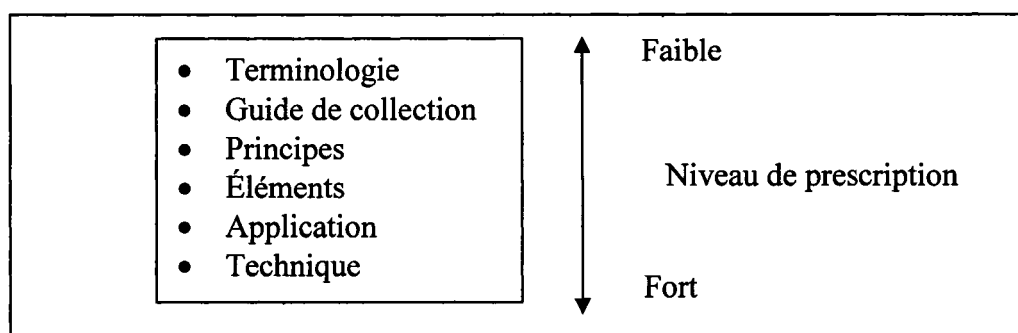


Figure 42 Niveau de prescription (Traduit de Moore 2006)

Les catégories ont un niveau prescriptif allant du plus faible (terminologie) jusqu'au plus fort (technique). Moore (2006) confirme que la grande majorité des normes se situe au niveau de la catégorie élément.

Moore (2006) souligne que les normes peuvent soutenir les quatre principaux objets du génie logiciel interagissant entre eux. Ces objets sont : le client, le processus, les ressources et le produit.

Moore (2006) propose deux nouvelles vues supplémentaires d'organisation du corpus. La troisième vue se fonde sur les dix domaines de connaissances du guide SWEBOK (2004). Chacune des normes est associée à l'un ou l'autre des domaines de connaissances. Une norme peut être associée à plus d'un domaine.

En dernier lieu, Moore propose une quatrième vue du corpus en fonction des processus de la norme parapluie ISO/IEC 12207. Compte tenu que la majorité des propositions que nous avons retenues appartiennent à la catégorie processus, l'organisation par processus du corpus est pertinente pour la vérification du degré de couverture.

Comme nous l'avons déjà présenté antérieurement (à la phase 3), la norme ISO/IEC 12207 structure les processus en trois principales catégories présentées au tableau CCVI.

Tableau CCVI

Catégories de processus ISO/IEC 12207 (Traduction Séguin)

<b>Processus du cycle de vie ISO/IEC 12207</b>		
<b>Primaires</b>	<b>Soutien</b>	<b>Organisationnels</b>
1. Acquisition 2. Approvisionnement 3. Développement 4. Exploitation 5. Maintenance	1. Documentation 2. Gestion des configurations 3. Assurance qualité 4. Vérification 5. Validation 6. Revue 7. Audit 8. Résolution des problèmes	1. Management 2. Infrastructure 3. Amélioration 4. Formation

#### 6.3.4 Association normes et processus

Nous présentons dans cette section, les résultats des croisements faits par Moore (2006) entre les processus de la norme ISO/IEC 12207 et les autres normes du génie logiciel de l'IEEE. Ces résultats nous sont nécessaires afin de compléter le travail fait par Moore en associant les propositions retenues aux normes du corpus.

##### 6.3.4.1 Processus primaires

Moore propose un raffinement afin de mieux catégoriser les cinq processus primaires. Les processus d'acquisition et d'approvisionnement sont catégorisés de niveau entreprise (affaires) tandis que les processus de développement, d'exploitation et de maintenance sont classés de catégorie technique.

Le tableau CCVII suivant présente la ventilation des normes en fonction en fonction de quatre processus primaire. Le cinquième processus, le développement, fera l'objet d'un tableau distinct.



Tableau CCVII

Association processus primaires et normes (adapté de Moore 2006)

<b>Processus</b>	<b>Normes associées</b>
Acquisition	ISO/IEC 12207, IEEE 1517, 1062, 1362
Approvisionnement	ISO/IEC 12207, IEEE 1517
Exploitation	ISO/IEC 12207, IEEE 1517
Maintenance	ISO/IEC 12207, IEEE 1517, 1219, ISO/IEC 14764

Le processus primaire de développement comporte 13 activités. Moore en a fusionné quelques unes puisqu'elles touchaient aux mêmes normes. Le tableau CCVIII suivant présente l'association entre les activités du processus primaire et les normes du corpus.

Tableau CCVIII

Association des activités de développement et normes (adapté de Moore 2006)

<b>Activités du processus de développement</b>	<b>Normes associées</b>
Mise en place du processus	ISO/IEC 12207, IEEE1517, 1074
Analyse des exigences système	ISO/IEC 12207, IEEE1517, 1233, 1320.1, 1320.2
Conception architecturale système	ISO/IEC 12207, IEEE1517, 1471
Analyse des exigences du logiciel	ISO/IEC 12207, IEEE1517, 830, ISO9126-1
Conception architecturale du logiciel	ISO/IEC 12207, IEEE1517, 829, 1063, 1471
Conception détaillée du logiciel	ISO/IEC 12207, IEEE1517, 829, 1016, 1063
Programmation et tests du logiciel	ISO/IEC 12207, IEEE1517, 829, 1008, 1063
Intégration du logiciel et tests	ISO/IEC 12207, IEEE1517, 829, 1063
Intégration système et tests	ISO/IEC 12207, IEEE1517, 829
Installation du logiciel et soutien	ISO/IEC 12207, IEEE1517

### 6.3.4.2 Processus de soutien

La catégorie des processus de soutien comporte huit processus. Moore souligne que les normes ISO/IEC 15939 et l'IEEE 1517 en ajoutent deux autres soit la mesure et la réutilisation. Nous proposons donc d'intégrer les deux processus supplémentaires. Le tableau CCIX suivant présente les résultats de la ventilation des dix processus de soutien en fonction des normes du corpus.

Tableau CCIX

Association des processus de soutien et les normes (Adapté de Moore 2006)

<b>Processus de soutien</b>	<b>Normes associées</b>
Documentation	ISO/IEC 12207, IEEE 1063
Gestion des configurations	ISO/IEC 12207, IEEE 828
Assurance qualité	ISO/IEC 12207, 9001, IEEE 730, 1061, 1465 ISO/IEC 90003
Vérification	ISO/IEC 12207, IEEE 1012
Validation	ISO/IEC 12207, IEEE 1012
Revue	ISO/IEC 12207, IEEE 1028
Audit	ISO/IEC 12207, IEEE 1028
Résolution de problème	ISO/IEC 12207, IEEE 1044
<i>Mesure</i>	ISO/IEC 15939, IEEE 982, 1045, 1061, 14143.1
<i>Réutilisation</i>	IEEE 1517, 1420

### 6.3.4.3 Processus organisationnels

La norme ISO/IEC 12207 définit quatre processus de catégorie organisationnelle. Moore souligne que les normes IEEE 1540 et 1517 en ajoutent chacune un, soit la gestion du risque et la gestion de la réutilisation. Le tableau CCX suivant présente l'association des ces processus avec le corpus des normes de l'IEEE.

Tableau CCX

Association des processus organisationnels et les normes (Adapté de Moore 2006)

<b>Processus</b>	<b>Normes associées</b>
Management	ISO/IEC 12207, IEEE 1058, 1490
Infrastructure	ISO/IEC 12207, IEEE 1175, 1462
Amélioration	ISO/IEC 12207, 15504
Formation	ISO/IEC 12207,
<i>Gestion du risque</i>	IEEE 1540
<i>Gestion de la réutilisation</i>	IEEE 1517

### 6.3.5 Propositions et normes

À la phase 3, les propositions de catégorie processus ont été associées aux processus et aux activités de la norme ISO/IEC 12207. Comme Moore (2006) a fait le lien entre les processus de cette même norme et les autres normes du corpus de l'IEEE, nous allons, dans un premier temps, associer les propositions des catégories processus aux différentes normes du corpus. Dans un deuxième temps, les propositions appartenant uniquement aux normes aux catégories produit et individus seront traitées.

#### 6.3.5.1 Propositions de catégorie processus

Au niveau des phases 2 et 3, nous avons déjà amplement commenté ces propositions. Afin d'éviter la redondance, nous fournissons dans cette section qu'un bref commentaire et le tableau synthèse de l'association entre les processus et les normes pour chacune des propositions. Il est à noter que les normes ISO/IEC 12207 et IEEE 1517 ne seront pas indiquées au niveau des tableaux, compte tenu qu'elles sont associées presque à toutes les propositions

Proposition no.1: Align incentives for developer and customer (Davis 1995)

Le sens donné par Davis (1995) à cette proposition est essentiellement orienté sur le développement du code. Le processus primaire de développement est impliqué avec l'activité de programmation et de tests. De plus, la portée étendue englobe aussi le processus de management. Le tableau CCXI présente les résultats de l'association des processus et des normes.

Tableau CCXI

Association processus et normes pour la proposition no.1

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.7 Software coding and testing	IEEE 829, 1008, 1063
<b>Organisationnel</b> 7.1 Management	7.1.2 Planning 7.1.3 Execution and control 7.1.4 Review and evaluation	IEEE 1490, 1058

Proposition no.2: Apply and use quantitative measurements in decision making  
(Bourque et al. 2002)

La portée de cette proposition touche essentiellement le processus de management et particulièrement deux activités. Il est à noter que la proposition pourrait avoir une portée étendue puisque la prise de décision à l'aide de mesure peut être utilisée dans d'autres processus du cycle de vie.

Tableau CCXII

Association processus et normes pour la proposition no.2

Processus	Activités	Normes associées
<b>Organisationnel</b> 7.1 Management	7.1.3 Execution and control 7.1.4 Review and evaluation	IEEE 1058, 1490

Proposition no.3 : Build software so that it needs a short user manual (Davis 1995)

La proposition implique le processus primaire de développement et plus spécifiquement quatre de ses activités. Comme le montre le tableau CCXIII, sept normes sont associées à la proposition.

Tableau CCXIII

Association processus et normes pour la proposition no.3

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.4 Software requirement analysis 5.3.5 Software architectural design 5.3.6 Software detailed design 5.3.7 Software coding and testing	IEEE 830, ISO/IEC9126 IEEE 829, 1063, 1471 IEEE 829, 1016,1063 IEEE 829, 1008, 1063

Proposition no.4 : Build with and for reuse (Bourque et al. 2002)

La proposition est associée à trois activités du processus primaire de développement. À ces activités sont associées cinq normes tel que présenté par le tableau CCXIV.

Tableau CCXIV

Association processus et normes pour la proposition no.4

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.5 Software architectural design 5.3.6 Software detailed design 5.3.7 Software coding and testing	IEEE 829, 1471, 1063 IEEE 829, 1016,1063 IEEE 829, 1008, 1063

Proposition no.6 : Define software artifacts rigorously (Bourque et al. 2002)

La proposition a une portée sur les produits intermédiaires issus du processus primaire de développement. Elle implique quatre activités du processus de développement.

Tableau CCXV

Association processus et normes pour la proposition no.6

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.4 Software requirement analysis 5.3.5 Software architectural design 5.3.6 Software detailed design 5.3.7 Software coding and testing	IEEE 830, ISO/IEC9126 IEEE 829, 1471, 1063 IEEE 829, 1016,1063 IEEE 829, 1008, 1063

Proposition no.9 : Design for maintenance (Davis 1995)

La proposition implique deux processus primaires soit le développement et la maintenance. Les activités de conception sont ciblées au niveau de ces processus. Même en phase de maintenance, une modification peut nécessiter des changements au niveau de la conception antérieure du logiciel.

Tableau CCXVI

Association processus et normes pour la proposition no.9

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.5 Software architectural design 5.3.6 Software detailed design	IEEE 829, 1471, 1063 IEEE 829, 1016, 1063

Tableau CCXVI (Suite)

Processus	Activités	Normes associées
5.5 Maintenance	5.5.2 Problem and modification analysis 5.5.3 Modification implementation 5.5.4 Maintenance review/acceptance	IEEE 1219

Proposition no.10 : Determine requirements now (Davis 1995).

La proposition est associée à deux activités du processus primaire de développement, soit les activités touchant aux exigences. À ces activités, quatre normes sont associées tel que présenté au tableau CCXVII.

Tableau CCXVII

Association processus et normes pour la proposition no.10

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.2 System requirement analysis 5.3.4 Software requirement analysis	IEEE 1233, 1320.1 1320.2 IEEE 830, ISO/IEC9126

Proposition no.13 : Don't try to retrofit quality (Davis 1995)

La proposition est liée à trois processus de soutien : l'assurance qualité, la vérification et la validation.

Tableau CCXVIII

Association processus et normes pour la proposition no.13

Processus	Activités	Normes associées
<b>Soutien</b> 6.3 Quality Assurance process	6.3.2 Product assurance 6.3.3 Process assurance	IEEE 730, 1061, 1465

Tableau CCXVIII (suite)

Processus	Activités	Normes associées
6.4 Verification	6.4.2 Verification 6.4.2.2 Process verification 6.4.2.3 Requirements verification 6.4.2.4 Design verification 6.4.2.5 Code verification 6.4.2.6 Integration verification 6.4.2.7 Documentation verification	IEEE 1012
6.5 Validation	6.5.2 Validation	IEEE 1012

Proposition no.14 : Don't write your own test plans (Davis 1995)

La proposition est associée aux deux principaux processus primaires, soit le développement et la maintenance.

Tableau CCXIX

Association processus et normes pour la proposition no.14

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.7 Software coding and testing 5.3.9 Software qualification testing 5.3.11 System qualification testing	IEEE 829, 1008, 1063 IEEE 829, 1063 IEEE 829
5.5 Maintenance	5.5.3 Modification implementation	IEEE 1219 ISO/IEC 14764

Proposition no.15 : Establish a software process that provides flexibility (Bourque et al. 2002)



La proposition ne s'applique pas à un groupe de processus en particulier de la norme ISO/IEC12207, mais à l'esprit de la norme dans son ensemble. Cependant, Moore (2006) a identifié un groupe de normes s'appliquant au processus. Ainsi, nous retiendrons donc ces normes pour la proposition tel que présenté au tableau CCXX.

Tableau CCXX

Normes IEEE s'appliquant au processus (Moore 2006)

IEEE 730	IEEE 1045	IEEE 1220
IEEE 828	IEEE 1058	IEEE 12207
IEEE 1008	IEEE 1062	IEEE 1490
IEEE 1012	IEEE 1074	IEEE 1517
IEEE1028	IEEE 1219	IEEE 1540

Proposition no.16 : Fix requirements specification error now (Davis 1995)

La portée de la proposition implique le processus primaire de développement, en particulier, les activités touchant aux exigences. Également, le processus de soutien de vérification est impliqué. Le terme « fix » de la proposition implique qu'une vérification a été faite afin de détecter un écart à combler.

Tableau CCXXI

Association processus et normes pour la proposition no.16

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.2 System requirement analysis 5.3.4 Software requirement analysis	IEEE 1233, 1320.1 1320.2 IEEE 830, ISO/IEC9126
<b>Soutien</b> 6.4 Vérification		IEEE 1012

Proposition no.17 : Give product to customer early (Davis 1995)

La portée de cette proposition, selon Davis (1995), est essentiellement orientée vers le prototypage. Cette technique est utilisée à la phase des exigences afin d'élucider les exigences moins perceptibles et de les valider. Le processus primaire de développement est impliqué, de même que le processus de vérification.

Tableau CCXXII

Association processus et normes pour la proposition no.17

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.4 Software requirement analysis	IEEE 830, ISO/IEC9126
<b>Soutien</b> 6.4 Vérification		IEEE 1012

Proposition no.19: Grow systems incrementally (Davis 1995)

La proposition touche essentiellement le processus primaire de développement. Malgré que Davis (1995) utilise le terme « system » dans la formulation de la proposition, son explication porte essentiellement sur le logiciel.

Tableau CCXXIII

Association processus et normes pour la proposition no.19

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.4 Software requirement analysis	IEEE 830, ISO/IEC9126
	5.3.5 Software architectural design	IEEE 829, 1471, 1063
	5.3.6 Software detailed design	IEEE 829, 1016, 1063
	5.3.8 Software integration	IEEE 829, 1063
	5.3.9 Software qualification testing	
	5.3.10 System integration	IEEE 829
	5.3.11 System qualification testing	IEEE 829
	5.3.12 Software installation	

Proposition no.20: Implement a disciplined approach and improve it continuously (Bourque et al. 2002)

La proposition n'implique pas un processus en particulier de la norme ISO/IEC 12207, mais l'ensemble des processus. La portée de la proposition est donc de niveau global. Tout comme fait antérieurement pour la proposition no.15, nous utiliserons le regroupement fait par Moore (2006) concernant les normes s'appliquant au processus, tel que présenté au tableau CCXXIV.

Tableau CCXXIV

Normes s'appliquant au processus (Moore 2006)

IEEE 730	IEEE 1045	IEEE 1220
IEEE 828	IEEE 1058	IEEE 12207
IEEE 1008	IEEE 1062	IEEE 1490
IEEE 1012	IEEE 1074	IEEE 1517
IEEE1028	IEEE 1219	IEEE 1540

Proposition no.21: Invest in the understanding of the problem (Bourque et al. 2002)

La proposition implique quatre processus primaires : l'acquisition, l'approvisionnement, le développement et la maintenance. Dans chacun de ces processus, il y a au moins une activité qui s'intéresse à la compréhension du problème à résoudre. La proposition implique le processus de soutien de résolution de problèmes, ainsi que le processus de management de niveau organisationnel.

Tableau CCXXV

Association processus et normes pour la proposition no.21

Processus	Activités	Normes associées
<b>Primaire</b>		
5.1 Acquisition	5.1.1 Initiation	IEEE 1062 IEEE 1362
5.2 Approvisionnement	5.2.1 Initiation 5.2.2 Preparation of response	
5.3 Développement	5.3.2 System requirements analysis 5.3.3 System Architectural design 5.3.4 Software Requirements analysis	IEEE 1233, 1320.1 et .2 IEEE 1471 IEEE 830 ISO/IEC 9126
5.5 Maintenance	5.5.2 Problem and modification analysis	IEEE 1219 ISO/IEC 14764
<b>Support</b>		
6.8 Problem resolution	6.8.2 Problem resolution	IEEE 1044
<b>Organisationnel</b>		
7.1 Management	7.1.3 Execution and control	IEEE 1490, 1058

Proposition no.22: Involve the customer (Royce 1970)

L'ensemble des processus primaires est impliqué par la proposition. De plus, trois processus de soutien sont impliqués : vérification, validation et revue.

Tableau CCXXVI

Association processus et normes pour la proposition no.22

Processus	Activités	Normes associées
<b>Primaire</b>		
5.1 Acquisition	5.1.1 Initiation 5.1.2 Request for proposal preparation 5.1.3 Contract preparation 5.1.4 Supplier monitoring 5.1.5 Acceptance and completion	IEEE 1062, 1362
5.2 Approvisionnement	5.2.4 Planning	
5.3 Développement	5.3.2 System requirements analysis 5.3.3 System Architectural design 5.3.4 Software Requirements analysis 5.3.5 Software Architectural design 5.3.10 System integration 5.3.11 System Qualification testing 5.3.12 Software installation 5.3.13 Software Acceptance support	IEEE 1233, 1320 IEEE 1471 IEEE 830, ISO/IEC 9126  IEEE 829, 1471, 1063 IEEE 829 IEEE 829
5.4 Exploitation	5.4.3 System operation 5.4.4 User support	IEEE 1517
<b>Support</b>		
6.4 Verification	6.4.2.1 Contract verification 6.4.2.2 Process verification 6.4.2.3 Requirements verification 6.4.2.4 Design verification 6.4.2.6 Integration verification 6.4.2.7 Documentation verification	IEEE 1012
6.5 Validation	6.5.2 Validation	IEEE 1012
6.6 Joint review process		IEEE 1028

Proposition no.23: Keep design under intellectual control (Davis 1995)

Davis (1995) limite grandement la portée de cette proposition à l'utilisation de certaine technique de documentation du design. De ce fait, le sens donné par l'auteur au contrôle intellectuel est très limitatif.

Trois activités sont visées au niveau du processus primaire de développement. Également, la gestion de configuration est aussi impliquée en tant que processus de soutien. Le tableau CCXXVII présente les résultats.

Tableau CCXXVII

Association processus et normes pour la proposition no.23

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.3 System Architectural design 5.3.5 Software Architectural design 5.3.6 Software Detailed design	IEEE 1471 IEEE 829, 1471, 1063 IEEE 829, 1016, 1063
<b>Support</b> 6.2 Configuration management		IEEE 828

Proposition no.24: Maintain clear accountability for results (Boehm 1993)

La proposition implique le processus organisationnel de management et particulièrement les activités de planification. Deux normes sont ainsi associées à la proposition tel que présenté au tableau CCXXVIII.

Tableau CCXXVIII

Association processus et normes pour la proposition no.24

Processus	Activités	Normes associées
<b>Organisationnel</b> 7.1 Management	7.1.2 Planning 7.1.2.1 a, d et e	IEEE 1058, 1490

Proposition no.25: Produce software in a stepwise fashion (Bourque et al. 2001)

La proposition suggère de développer le logiciel en suivant une séquence d'étapes. À ce sujet, la norme ISO/IEC 12207 découpe le processus de développement en plusieurs activités tel que montré au tableau CCXXIX. La portée de la proposition est essentiellement limitée au développement, ainsi, la maintenance n'est incluse.

Tableau CCXXIX

Association processus et normes pour la proposition no.25

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.2 System requirements analysis 5.3.3 System Architectural design 5.3.4 Software requirement analysis 5.3.5 Software architectural design 5.3.6 Software detailed design 5.3.7 Software coding and testing 5.3.8 Software integration 5.3.9 Software qualification testing 5.3.10 System integration 5.3.11 System qualification testing 5.3.12 Software installation 5.3.13 Software Acceptance support	IEEE 1233, 1320 IEEE 1471 IEEE 830, ISO/IEC9126 IEEE 829, 1471, 1063 IEEE 829, 1016,1063 IEEE 829, 1008, 1063 IEEE 829, 1063  IEEE 829 IEEE 829

Proposition no.26: Quality is the top priority; long-term productivity is a natural consequence of high quality (Wiegiers 1996)

La proposition est liée aux trios groupes de processus de la norme ISO/IEC 12207. Le tableau CCXXX présente les processus et les normes associées.

Tableau CCXXX

Association processus et normes pour la proposition no.26

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement  5.5 Maintenance	5.3.7 Software coding and testing 5.3.8 Software integration 5.3.9 Software qualification testing 5.3.10 System integration 5.3.11 System qualification testing  5.5.3 Modification implementation o 5.5.3.2 a) et b)	IEEE 829, 1008, 1063 IEEE 829, 1063  IEEE 829 IEEE 829  IEEE 1219 ISO/IEC 14764
<b>Support</b> 6.3 Quality assurance process 6.4 Verification 6.5 Validation process 6.6 Joint review 6.7 Audit		ISO/IEC 9001, 90003 IEEE 730, 1061, 1465 IEEE 1012 IEEE 1012 IEEE 1028 IEEE 1028
<b>Organisationnel</b> 7.1 Management  7.3 Process improvement	7.1.2.1 g	IEEE 1490, 1058  ISO/IEC 15504

Proposition no.27 : Rotate people through product assurance (Davis 1995)



La proposition vise essentiellement la gestion des ressources humaines. En effet, elle suggère au gestionnaire de procéder, à l'occasion, à l'envoi d'excellents développeurs faire un séjour au sein du groupe d'assurance qualité. Le tableau CCXXXI présente les processus et les normes associées à la proposition.

Tableau CCXXXI

Association processus et normes pour la proposition no.27

Processus	Activités	Normes associées
<b>Organisationnel</b> 7.1 Management	7.1.2 Planning <ul style="list-style-type: none"> <li>o 7.1.2.1 c), d) et e)</li> </ul>	IEEE 1490, 1058

Proposition no.28 : Since change is inherent to software, plan for it and manage it (Bourque et al. 2002)

La norme ISO/IEC 12207 prévoit certaines activités pour gérer les changements. Les trois groupes de processus de la norme sont impliqués tel que présenté au tableau CCXXXII.

Tableau CCXXXII

Association processus et normes pour la proposition no.28

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.1 Process implementation <ul style="list-style-type: none"> <li>• b)</li> </ul>	IEEE 1074
5.5 Maintenance		IEEE 1219 ISO/IEC 14764

Tableau CCXXXII (Suite)

<b>Processus</b>	<b>Activités</b>	<b>Normes associées</b>
<b>Support</b> 6.2 Configuration management	6.2.3.1 Change control	IEEE 828
<b>Organisationnel</b> 7.1 Management	7.1.3 Execution and Control	IEEE 1490, 1058

Proposition no.29 : Since tradeoffs are inherent to software engineering, make them explicit and document it. (Bourque et al. 2002)

Les compromis soulignés par la proposition sont principalement effectués au niveau du processus de développement et de la maintenance du logiciel.

Tableau CCXXXIII

Association processus et normes pour la proposition no.29

<b>Processus</b>	<b>Activités</b>	<b>Normes associées</b>
<b>Primaire</b> 5.3 Développement	5.3.2 System requirements analysis 5.3.3 System Architectural design 5.3.4 Software requirement analysis 5.3.5 Software architectural design 5.3.6 Software detailed design 5.3.7 Software coding and testing	IEEE 1233, 1320 IEEE 1471 IEEE 830, ISO/IEC9126 IEEE 829, 1471, 1063 IEEE 829, 1016, 1063 IEEE 829, 1008, 1063
5.5 Maintenance	5.5.2 Problem and modification analysis 5.5.3 Modification implementation	IEEE 1219 ISO/IEC 14764

Proposition no.30: Strive to have a peer, rather than a customer, find a defect (Wiegiers 1996)

La proposition vise les activités de développement et de maintenance du logiciel. Également, le processus de soutien avec l'assurance qualité, la vérification et la revue.

Tableau CCXXXIV

Association processus et normes pour la proposition no.30

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.7 Software coding and testing 5.3.8 Software integration 5.3.9 Software qualification testing 5.3.10 System integration 5.3.11 System qualification testing	IEEE 829, 1008, 1063 IEEE 829, 1063  IEEE 829 IEEE 829
5.5 Maintenance	5.5.3 Modification implementation o 5.5.3.2 a) et b)	IEEE 1219 ISO/IEC 14764
<b>Support</b> 6.3 Quality Assurance  6.4 Verification  6.5 Validation 6.6 Joint review	6.3.2 Product assurance  6.4.2.3 Requirements verification 6.4.2.4 Design verification 6.4.2.5 Code verification 6.4.2.6 Integration verification 6.4.2.7 Documentation verification	IEEE 730, 1061, 1465, ISO/IEC 9001, 90003 IEEE 1012  IEEE 1012 IEEE 1028

Proposition no.31: Tailor cost estimation methods (Davis 1995)

L'estimation des coûts et de l'effort est une activité faisant partie de la planification définie par le processus de catégorie organisationnel de management.

Tableau CCXXXV

Association processus et normes pour la proposition no.31

Processus	Activités	Normes associées
<b>Organisationnel</b> 7.1 Management	7.1.2 Planning	IEEE 1490, 1058

Proposition no.32 : To improve design, study previous solutions to similar problems  
(Bourque et al. 2002)

La proposition est principalement liée au processus primaire de développement, particulièrement à l'activité de conception.

Tableau CCXXXVI

Association processus et normes pour la proposition no.32

Processus	Activités	Normes associées
<b>Primaire</b> 5.3 Développement	5.3.3 System Architectural design 5.3.5 Software architectural design 5.3.6 Software detailed design	IEEE 1471 IEEE 829, 1471, 1063 IEEE 829, 1016, 1063

Proposition no.33 : Use better and fewer people (Boehm 1983)

La proposition implique la gestion des ressources humaines d'un projet. Ainsi, l'activité de planification du processus organisationnel de management est associée à la proposition.

Tableau CCXXXVII

Association processus et normes pour la proposition no.33

Processus	Activités	Normes associées
<b>Organisationnel</b> 7.1 Management	7.1.2 Planning	IEEE 1490, 1058

Proposition no.34 : Use documentation standards

La proposition implique le processus de soutien de documentation. La norme IEEE/EIA 12207.1 est un supplément à la norme ISO/IEC 12207 afin de préciser la documentation à produire à chacune des étapes du processus. La norme IEEE 1063 se spécialise dans la documentation utilisateur du logiciel.

Tableau CCXXXVIII

Association processus et normes pour la proposition no.34

Processus	Activités	Normes associées
<b>Soutien</b> 6.1 Documentation		IEEE 1063 ISO/IEC 12207, IEEE/EIA 12207.1

Proposition no.35 : Write programs for people first

La proposition est en lien avec les processus d'écriture du code qui se retrouvent au niveau du développement et de la maintenance tel que présenté au tableau CCXXXIX.

Tableau CCXXXIX

Association processus et normes pour la proposition no.35

Processus	Activités	Normes associées
<b>Primaire</b>		
5.3 Développement	5.3.7 Software coding and testing	IEEE 829, 1008, 1063
5.5 Maintenance	5.5.3 Modification implementation ○ 5.5.3.2 a) et b)	IEEE 1219 ISO/IEC 14764

Proposition no.37 : Select tests based on the likelihood that they will find faults (Davis 1995)

La proposition implique le processus primaire de développement et de maintenance, particulièrement les activités de tests et d'intégration du logiciel ainsi que l'implémentation des changements dans le contexte de la maintenance.

Tableau CCXL

Association processus et normes pour la proposition no.37

Processus	Activités	Normes associées
<b>Primaire</b>		
5.3 Développement	5.3.7 Software coding and testing 5.3.8 Software integration 5.3.10 System integration	IEEE 829, 1008, 1063 IEEE 829, 1063 IEEE 829
5.5 Maintenance	5.5.3 Modification implementation ○ 5.5.3.2 a)	IEEE 1219 ISO/IEC 14764

Proposition no.38 : Choose a programming language to assure maintainability

La portée de la proposition vise essentiellement le choix d'outils de développement. Cette activité est liée au processus organisationnel d'infrastructure tel que présenté au tableau CCCLI

Tableau CCXLI

Association processus et normes pour la proposition no.38

Processus	Activités	Normes associées
<b>Organisationnel</b> 7.2 Infrastructure	Choix des outils et logiciels	IEEE 1175, 1462

Proposition no.39 : In face of unstructured code, rethink the module and redesign it from scratch.

Le sens de la proposition est orienté particulièrement dans un contexte de maintenance, plus particulièrement sur l'implémentation des changements.

Tableau CCXLII

Association processus et normes pour la proposition no.39

Processus	Activités	Normes associées
<b>Primaire</b> 5.5 Maintenance	5.5.3 Modification implementation ○ 5.5.3.2 a)	IEEE 1219 ISO/IEC 14764

### 6.3.5.2 Propositions de catégorie produit et individu

Suite à l'association des 32 propositions de catégorie processus aux corpus des normes, il nous reste maintenant que deux propositions à associer. Pour ce faire, nous utiliserons le classement par objets du génie logiciel proposé par Moore (2006).

Proposition no.11 : Don't overstrain your hardware

À la phase 3, cette proposition a été classée à la catégorie produit. C'est le logiciel qui provoque la surcharge du matériel, donc le produit final. Pour le produit, Moore associe les normes présentées au tableau CCXLIII.

#### Tableau CCXLIII

Normes associées à la proposition no.11

IEEE 982.1, IEEE 1061, IEEE1063 et IEEE1465
---

Proposition no. 36 : Know software engineering's techniques before using development tools

La proposition vise les compétences des développeurs à l'effet qu'ils devraient connaître, comprendre et maîtriser les techniques de base du génie logiciel. Les normes peuvent fournir un guide pour l'utilisation des outils, mais elles présupposent que l'individu maîtrise les techniques sous-jacentes du génie logiciel. De ce fait, nous ne ferons pas d'association de la proposition avec le corpus des normes. La proposition est plutôt orientée vers la formation des individus

#### 6.3.6 Synthèse sur l'association des propositions aux normes

L'objectif de ce second volet de la vérification était d'évaluer le degré de couverture des propositions retenues avec le corpus des normes du génie logiciel de l'IEEE. Les 32 propositions de catégorie processus ont été associées par l'entremise de la norme ISO/IEC 12207. Par la suite, les deux propositions restantes de catégorie produit et individu ont été traitées. Parmi le groupe des 34 propositions, seulement deux n'ont pu être associées (no.31 et no.36) puisque leur signification n'implique pas directement des normes.



Le tableau CCXLIV suivant présente le sommaire des normes associées à chacune des propositions.

Tableau CCXLIV

## Synthèse de l'association des propositions aux normes

No.	Proposition	Catégorie
1	Align incentives for developer and customer IEEE 829, 1008, 1058, 1063, 1490,	Individu Processus
2	Apply and use quantitative measurements in decision making IEEE 1058, 1490	Processus
3	Build software so that it needs a short user manual IEEE 829, 830, 1008, 1016, 1063, 1471, ISO/IEC 9126	Produit Processus
4	Build with and for reuse IEEE 829,1008, 1016, 1063, 1471, 1517	Produit Processus
6	Define software artifacts rigorously IEEE 829, 830, 1008, 1016, 1063, 1471, ISO/IEC 9126	Produit Processus
9	Design for maintenance* IEEE 829, 1016, 1063, 1219, 1471	Processus
10	Determine requirements now* IEEE 830, 1233, 1320.1, 1320.2, ISO/IEC9126	Processus
11	Don't overstrain your hardware IEEE 982.1, 1061, 1063, 1465	Produit
13	Don't try to retrofit quality IEEE 730, 1012, 1061, 1465	Processus
14	Don't write your own test plans* IEEE 829,1008, 1063, 1219, ISO/IEC14764	Processus
15	Establish a software process that provides flexibility IEEE 730, 828, 1008, 1012, 1028, 1045, 1058, 1062, 1074, 1219, 1220, 12207, 1490, 1517, 1540	Processus
16	Fix requirements specification error now* IEEE 830, 1012, 1233, 1320.1, 1320.2, ISO/IEC9126	Processus
17	Give product to customers early* IEEE830, 1012, ISO/IEC9126	Processus
19	Grow systems incrementally IEEE 829, 830, 1016,1063, 1471, ISO/IEC9126	Processus
20	Implement a disciplined approach and improve it continuously IEEE 730, 828, 1008, 1012, 1028, 1045, 1058, 1062, 1074, 1219, 1220, 12207, 1490, 1517, 1540	Processus

Tableau CCXLIV (suite)

No.	Proposition	Catégorie
21	Invest in the understanding of the problem IEEE 830, 1044, 1058, 1062, 1219, 1233, 1320.1, 1320.2, 1362, 1471, 1490, ISO/IEC9126, ISO/IEC14764	Processus
22	Involve the customer IEEE 829, 830, 1012, 1028, 1062, 1063, 1233, 1320.1, 1320.2, 1362, 1471, 1517, ISO/IEC9126	Processus
23	Keep design under intellectual control* IEEE 828, 829, 1016, 1063, 1471	Processus
24	Maintain clear accountability for results* IEEE 1058, 1490,	Processus
25	Produce software in a stepwise fashion* IEEE 829, 830, 1008, 1016, 1063, 1233, 1320.1, 1320.2, 1471, ISO/IEC9126	Processus
26	Quality is the top priority; long term productivity is a natural consequence of high quality IEEE 730, 829, 1008, 1012, 1028, 1058, 1061, 1063, 1219, 1490, ISO/IEC14764, 15504, ISO 9001, 90003,	Individu Processus
27	Rotate (top performer) people through product assurance IEEE 1058, 1490	Processus
28	Since change is inherent to software, plan for it and manage it IEEE 828, 1058, 1074, 1219, 1490, ISO/IEC14764	Processus
29	Since tradeoffs are inherent to software engineering, make them explicit and document it* IEEE 829, 830, 1008, 1016, 1063, 1219, 1233, 1320.1, 1320.2, 1471, ISO/IEC9126, 14764	Processus
30	Strive to have a peer, rather than a customer, find a defect IEEE 730, 829, 1008, 1028, 1061, 1063, 1219, 1465 ISO/IEC14764, ISO 9001, 90003	Processus
31	Tailor cost estimation methods IEEE 1058 et 1490	Processus
32	To improve design, study previous solutions to similar problems IEEE 829, 1016, 1063, 1471,	Processus
33	Use better and fewer people IEEE 1058, 1490	Individu Processus
34	Use documentation standards IEEE 1063, 12207.1	Produit
35	Write programs for people first IEEE 829, 1008, 1063, 1219, ISO/IEC14764	Produit Processus

Tableau CCXLIV (suite)

No.	Proposition	Catégorie
36	Know software engineering's techniques before using development tools Aucune	Individu
37	Select tests based on the likelihood that they will find faults IEEE 829, 1008, 1063 , 1219, ISO/IEC14764	Processus
38	Choose a programming language to assure maintainability IEEE 1175, 1462	Produit Processus
39	In face of unstructured code, rethink the module and redesign it from scratch.* IEEE1219, ISO/IEC14764	Processus

Le tableau CCXLV présente un sommaire des résultats pour chacune des normes.

Tableau CCXLV

## Synthèse de l'association des normes aux propositions

Normes IEEE	Propositions
610.12	Toutes les propositions
730	13, 15, 20, 26
828	15, 20, 23, 28
829	1, 3, 4, 6, 9, 14, 19, 22, 23, 25, 26, 29, 30, 32, 35, 37
830	3, 6, 10, 16, 17, 19, 21, 22, 25, 29
982.1	11
1008	1, 3, 4, 6, 14, 15, 20, 25, 26, 29, 30, 35, 37
1012	13, 15, 16, 17, 20, 22, 26
1016	3, 4, 6, 9, 19, 23, 25, 29, 32
1028	15, 20, 22, 26, 30
1044	13, 21
1045	15, 20
1058	1, 2, 15, 20, 21, 24, 26, 27, 28, 31, 33
1061	11, 13, 26
1062	15, 20, 21, 22
1063	1, 3, 4, 6, 9, 11, 14, 19, 22, 23, 25, 26, 29, 30, 32, 35, 34, 37
1074	15, 20, 28
1175	38
1219	9, 14, 15, 20, 21, 26, 28, 29, 30, 35, 37, 39

Tableau CCXLV (suite)

<b>Normes IEEE</b>	<b>Propositions</b>
1220	15, 20
1228	22
12207.1	34
1233	10, 16, 21, 22, 25, 29
1320	10, 13, 21, 22, 25, 29
1362	21, 22
14143.1	
1420	
1462	38
1465	11, 13, 26
1471	4, 6, 9, 19, 21, 22, 23, 25, 29, 32
1490	1, 2, 15, 20, 21, 24, 26, 27, 28, 31, 33
1517	1, 3, 4, 6, 9, 10, 14, 15, 16, 17, 19, 20, 21, 22, 23, 25, 26, 28, 29, 30, 32, 37
1540	15, 20
2001	
<b>Normes ISO</b>	
12207	1, 2, 3, 4, 6, 9, 10, 13, 14, 15, 16, 17, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 32, 33, 34, 35, 37, 38, 39
14764	9, 14, 21, 26, 28, 29, 30, 35, 37, 39
15504	26
9001	13, 26
90003	13, 26
9126	3, 6, 10, 16, 17, 19, 21, 25, 29

Nous constatons que trois normes ne sont pas associées à aucune proposition :

- IEEE 1420
- IEEE 14143.1
- IEEE 2001

Pour la suite, nous utilisons les catégories (objets du génie logiciel) présentés par Moore (2006) afin d'évaluer le degré de couverture des propositions pour chacune des catégories.

La catégorie « client » comporte cinq normes. Chacune de celles-ci est couverte par au moins une proposition. Au total, huit propositions distinctes (i.e. sans doublon) se rattachent aux normes de la catégorie client, soit 24% des propositions. Le tableau CCXLVI présente les résultats.

Tableau CCXLVI

## Propositions associées aux normes de catégorie Client

<b>Client</b>		
<b>Normes</b>	<b>Nombre de propositions associées</b>	<b>Nombre de propositions distinctes</b>
IEEE 1062	4	8
IEEE 1220	2	
IEEE 1228	1	
IEEE 1233	6	
IEEE 1362	2	

La catégorie « produit » comporte six normes qui sont toutes couvertes par au moins une proposition. Pris d'une façon distincte, 21 propositions sont associées aux normes de la catégorie produit, soit 62% des propositions. Le tableau CCXLVII présente les résultats.

Tableau CCXLVII

## Propositions associées aux normes de catégorie Produit

<b>Produit</b>		
<b>Normes</b>	<b>Nombre de propositions associées</b>	<b>Nombre de propositions distinctes</b>
IEEE 982.1	1	21
IEEE 1044	2	
IEEE 1061	15	
IEEE 1063	2	
IEEE 1465	10	
IEEE 9126	2	

La prochaine catégorie, les « ressources », regroupe dix normes. Trois de ces normes ne recueillent aucune proposition, soit : IEEE 1420, IEEE 14143.1 et IEEE 2001. Le

nombre de propositions distinctes se rattachant à la catégorie ressource est de 21, soit 62% des propositions. Le tableau CCXLVIII présente les résultats.

Tableau CCXLVIII

## Propositions associées aux normes de catégorie Ressources

<b>Ressources</b>		
<b>Normes</b>	<b>Nombre de propositions associées</b>	<b>Nombre de propositions distinctes</b>
IEEE 829	15	21
IEEE 830	10	
IEEE 1016	9	
IEEE 1175	1	
IEEE 1320	6	
IEEE 14143	0	
IEEE 1420	0	
IEEE 1462	1	
IEEE 1471	10	
IEEE 2001	0	

La catégorie « processus » regroupent le plus grand nombre de normes avec 16 normes. Chacune de celles-ci est couverte par au moins une proposition. Le nombre de propositions distinctes se rattachant à cette catégorie est de 32, soit 94% des propositions. Compte tenu que 32 propositions sur 34 sont de catégorie processus, nous ne sommes pas surpris de constater un fort degré de couverture auprès des normes de nature processus. Le tableau CCXLIX présente les résultats.

Tableau CCXLIX

## Propositions associées aux normes de catégorie Processus

<b>Processus</b>		
<b>Normes</b>	<b>Nombre de propositions associées</b>	<b>Nombre de propositions distinctes</b>
IEEE 730	4	32
IEEE 828	4	
IEEE 1008	12	
IEEE 1012	7	

Tableau CCXLIX (suite)

Processus		
Normes	Nombre de propositions associées	Nombre de propositions distinctes
IEEE 1028	5	32
IEEE 1044		
IEEE 1045	2	
IEEE 1058	11	
IEEE 1074	3	
IEEE 1219	11	
IEEE 12207	31	
IEEE 1490	11	
IEEE 1517	22	
IEEE 1540	2	
ISO/IEC 14767	9	
ISO/IEC 90003	2	
ISO/IEC 15504	1	

Le tableau CCL suivant présente les normes qui ont reçues au moins 10 propositions.

Tableau CCL

Normes ayant obtenues 10 propositions et plus

Normes	Nombre de propositions	Catégorie	Normes	Nombre de propositions	Catégorie
IEEE12207	31	processus	IEEE 1058	11	Ressources
IEEE 1517	22	processus	IEEE 1490	11	Ressources
IEEE 1063	15	Produit	IEEE 1471	10	Ressources
IEEE 829	15	Ressource	IEEE 830	10	Ressources
IEEE 1008	12	processus	IEEE 9126	10	Produit
IEEE 1219	11	processus	IEEE 1016	10	Ressources

Nous constatons que trois de quatre catégories de normes sont couvertes par les normes présentées au tableau précédent. Seule la catégorie « client » n'est pas représentée dans ce tableau.

## 6.4 Conclusion

L'objectif de la phase 4 était d'évaluer le degré de couverture des 34 propositions retenues sous deux angles différents. Le premier angle a permis de vérifier le degré de soutien des propositions par rapport aux éléments de base du modèle d'ingénierie présenté par Moore (2006). Le deuxième angle a permis de vérifier la couverture des propositions avec le corpus de normes du génie logiciel de l'IEEE.

Pour le premier angle de vue, nous avons associé les propositions aux différents éléments du modèle du génie de Moore (2006). Le volet processus du modèle (comprenant aussi le produit et les ressources) recueille la majorité des propositions avec 30, tandis que le volet contrôle en comporte 21. Nous avons constaté que les éléments « besoins » et « mesure » recueillent respectivement zéro et une seule proposition. Ces deux éléments du modèle sont les moins couverts par les propositions. Le tableau CCLI présente la synthèse de l'association des propositions aux éléments du modèle du génie de Moore (2006).

Tableau CCLI

Synthèse de l'association des propositions au modèle d'ingénierie

Volet	Éléments du modèle	Nombre de propositions	Nombre de propositions distinctes
Contrôle	Contrôle	9	21
	But	5	
	Contrainte	18	
	Mesure	1	
	Action	2	
Processus	Processus	28	30
	Ressource	7	
	Produit	4	
	Besoins	0	



Il est à noter que l'élément « contrainte » recueille plus de proposition que l'élément contrôle. Cependant, c'est une caractéristique du génie de réaliser des projets en tenant compte d'un ensemble de contraintes.

Nous constatons que l'ensemble des propositions trouve une correspondance avec les éléments du modèle d'ingénierie. Le fait que la majorité des propositions soit associée à l'élément « processus » du modèle confirme la catégorisation faite à la phase 3 de l'analyse.

Le deuxième angle de notre vérification a consisté à associer les propositions au corpus de normes du génie logiciel de l'IEEE. Pour réaliser ce travail, nous avons choisi de prendre la vision par processus, dans un premier temps, et par objets présentées par Jim Moore (2006). La vision par processus a été principalement utilisée, compte tenu que la majorité des propositions sont de catégorie processus.

À la phase 3 de l'analyse des propositions, nous avons fait le travail d'associer chacune des propositions de catégorie processus aux différents processus de la norme ISO/IEC 12207. Moore (2006) a fait un travail d'associer les processus de cette norme aux autres normes du corpus de l'IEEE. En jumelant nos travaux respectifs, nous avons été en mesure d'établir des liens directs entre les propositions et les normes du corpus et d'en vérifier le degré de couverture.

Nous constatons, tout comme au niveau de la vérification avec le modèle du génie, que les normes appartenant à groupe processus sont les plus supportées par les propositions. Cette constatation reconferme de nouveau la justesse de la catégorisation faite à la phase 3. Le tableau CCLII suivant présente une synthèse du degré de couverture selon les quatre regroupements (par objets) présentés par Moore (2006).

Tableau CCLII

Synthèse du degré de couverture par regroupements de l'IEEE

Groupes de normes IEEE	Degré de couverture des propositions
Processus	94%
Produit	62%
Ressources	62%
Client	24%

Les normes des groupes « Produit » et « Ressources » sont aussi bien couvertes par les propositions avec un taux de 62% pour chacun des groupes. Le groupe « Client » ne recueille qu'un taux de couverture de 24%. Cependant, ce plus faible niveau de couverture s'explique par le fait que la majorité des propositions est de catégorie processus.