

RAD 2

Organisation des développements

SEI-CMM & ISO-SPICE



Organisation des développements,
conduite de projets,
ingénierie applicative,
modélisation adaptée.

Jean-Pierre Vickoff, RAD.fr

RAD2

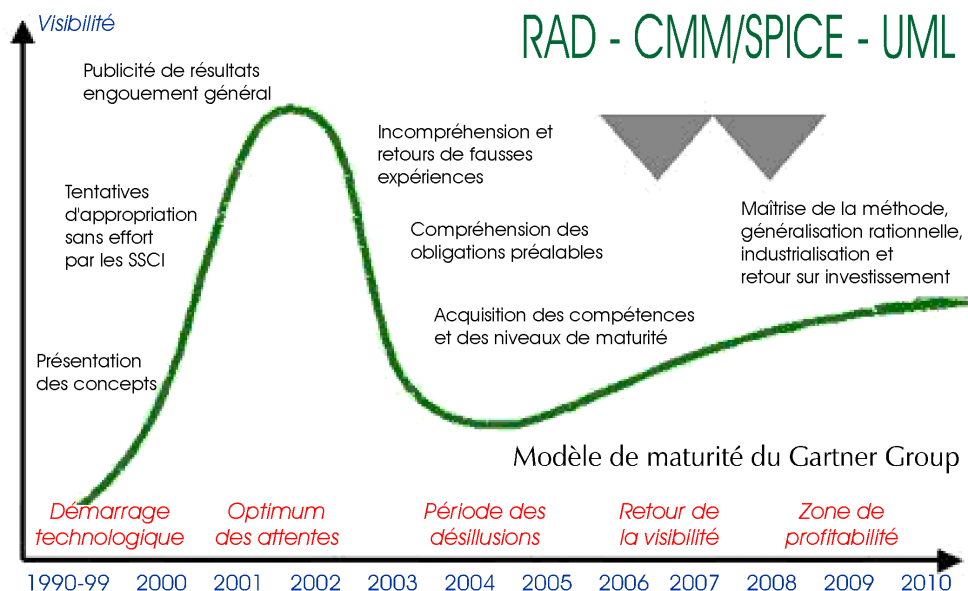
Organisation des développements

SEI-CMM & ISO-SPICE

Jean-Pierre Vickoff
RAD.fr

HORIZON 2010

Les changements les plus difficiles à accompagner sont ceux qui altèrent directement nos habitudes de travail, notre structure de pensée ou notre perception d'une réalité jusqu'ici stable et désormais en évolution. En matière de développement de systèmes d'information, une révolution fondamentale est en cours. Elle se caractérise par l'introduction simultanée de nouvelles méthodes de conduite de projets (comme le RAD), de nouvelles formes de modélisation (comme UML) et elle s'appuie sur des modèles d'évaluation et d'évolution de l'organisation (comme CMM ou ISO-SPICE). Elle fait intervenir l'objet, les composants, le partage ou l'achat de solutions et stratifie la production et l'exploitation du SI en couches d'externalisation. Il n'est donc pas surprenant que les principes régissant l'émergence et la généralisation des nouvelles technologies ou méthodologies prévoient encore, pour certains, des délais de l'ordre d'une dizaine d'années avant de maîtriser ces outils du futur immédiat et d'en tirer tous les bénéfices. En ce qui concerne le champ méthodologique, sa situation est formalisée ci-dessous par l'incontournable modèle de maturité du Gartner Group.



A la base de toute forme d'industrialisation se trouvent des normes et des standards. Pourtant, la plupart des informaticiens actuels hésitent à en adopter la pratique. L'homme est faillible, et des processus formalisés sont indispensables à la maîtrise de son action. Il lui faut ensuite une farouche volonté de les respecter (ou l'obligation), et surtout l'intelligence de les adapter. De ces contraintes naît la capacité de piloter dans la qualité et la performance les activités les plus complexes. C'est au prix de cet *état de l'art* et aussi de celui de la simplification, que le professionnel obtiendra finalement l'amélioration continue d'une qualité qui lui fait actuellement défaut.

Dans un univers de performances où l'obligation de résultats s'impose à tous progressivement, la rigueur devient l'alliée du changement. Les pilotes de SI se doivent alors d'envisager immédiatement l'acquisition d'une nouvelle culture « projet » et de l'appliquer à l'évolution de leur métier. A ce sujet je citerai Jean Monnet, un des initiateurs de la Communauté européenne : « La résistance des hommes est à la mesure de l'ampleur du changement que l'on cherche à apporter, elle est même le signe le plus sûr que l'on est sur la voie de ce changement. » Je connais des organisations qui doivent être en train de changer, car je vous assure qu'elles résistent.

Jean-Pierre Vickoff

www.RAD.fr

SOMMAIRE DE L'OUVRAGE

1. MATURITÉ ET NORMES DE DÉVELOPPEMENT	6
1.1. STANDARDS SEI-CMM ET ISO-SPICE	6
1.2. INTRODUCTION À CMM	8
STRUCTURE DU MODÈLE D'ORGANISATION	9
NIVEAUX DE MATURITÉ	10
Niveau 1 « initial »	10
Niveau 2 « reproductible »	10
Niveau 3 « défini »	10
Niveau 4 « maîtrisé »	10
Niveau 5 « optimisé »	10
SECTEURS CLÉS PAR NIVEAU	11
Secteurs clés niveau 2 « reproductible »	11
Secteurs clés niveau 3 « défini »	11
Secteurs clés niveau 4 « maîtrisé »	13
Secteurs clés niveau 5 « optimisé »	13
1.3. RAD2 ET LES 5 NIVEAUX DE MATURITÉ CMM	15
NIVEAU 1 « INITIAL »	15
NIVEAU 2 « REPRODUCTIBLE »	15
NIVEAU 3 « DÉFINI »	15
NIVEAU 4 « MAÎTRISÉ »	16
NIVEAU 5 « OPTIMISÉ »	16
1.4. INTRODUCTION À ISO-SPICE	16
2. BIBLIOGRAPHIES ET RÉFÉRENCES	20
2.1. BIBLIOGRAPHIE RAD, CONDUITE DE PROJET	20
2.2. BIBLIOGRAPHIE MANAGEMENT, QUALITÉ	20
2.3. DIVERS DOCUMENTS, NORMES, STANDARDS	21
2.4. PRINCIPAUX WEB ET CONTRIBUTEURS	22
2.5. INDEX	23
2.6. FIGURES	24
2.7. TABLEAUX ET LISTES	24



Partie 6

Standards d'évaluation et d'amélioration des capacités de production du logiciel

SEI-CMM / ISO-SPICE

« Troisième précepte : de conduire par ordre mes pensées, en commençant par les objets les plus simples et les plus aisés à connaître, pour monter peu à peu comme par degrés jusques à la connaissance des plus composés, et supposant même de l'ordre entre ceux qui ne se précèdent point naturellement les uns des autres. »

René Descartes, *Discours de la méthode*

1. Maturité et normes de développement

La **maîtrise des processus** d'ingénierie du logiciel se base désormais sur des **cadres de management** normalisés tels que :

- **CMM** (*Capability Maturity Model*) ou
- **SPICE** (*Software Process Improvement Capability dEtermination*).

La **performance** recherchée par cette maîtrise dépend également d'approches comme le Management par la Qualité Totale de Service (**MTQS**¹).

1.1. Standards SEI-CMM et ISO-SPICE

De la maîtrise des processus d'ingénierie du logiciel découle la maîtrise de la Qualité des produits et des services issus de ces processus. Cette Qualité mesurable est une caractéristique démontrée de la réussite économique des entreprises et de la performance des organisations.

CMM : *Capability Maturity Model*, ou modèle d'évaluation et d'évolution des capacités de développement logiciel, a été mis au point par le SEI (*Software Engineering Institute*). et traduit par le CRIM² (<http://www.CRIM.ca>). CMM est un standard de fait. SPICE s'en inspire et se présente sous la forme d'une norme internationale (ISO 15504).

Dans les principes communs aux deux approches, l'évaluation de processus associe un niveau à un processus afin de permettre à une organisation de diagnostiquer son « niveau-qualité ». Le but de cette évaluation est généralement l'engagement d'une démarche d'amélioration (*figure 102*) ou la certification « qualité » des compétences.

Par rapport à son prédécesseur CMM, l'amélioration recherchée par le modèle SPICE est une évaluation plus fine, processus par processus. Chaque processus est évalué individuellement et se voit attribuer son propre niveau de maturité. Le niveau 1 (Initial) de CMM a été scindé en deux niveaux :

- niveau 0 : maîtrise inexistante des processus ;
- niveau 1 : maîtrise informelle des processus.

Cette distinction existe depuis longtemps dans l'outil *Évaluateur* (onglet 4, méthode, Qualité, ...) et différencie l'efficacité des organisations en fonction des résultats réellement obtenus :

- CMM-1-Bas : absence de méthode ou méthode empirique ayant déjà conduit à des débordements ou échecs de projets ;
- CMM-1-Haut : méthode « maison » n'ayant jamais conduit à des débordements ou échecs de projets.

Les objectifs des deux « modèles » SEI-CMM et ISO-SPICE sont globalement identiques. Il en est de même des principes et de la structure d'évaluation qu'ils préconisent. SPICE se souhaite plus généraliste afin de couvrir tous les types de projets. CMM est plus adapté aux projets de développements spécifiques d'application.

¹ Management Total Quality Service. Le MTQS englobe des facteurs subjectifs comme l'engagement effectif des ressources humaines ou la capacité de l'organisation et de ses membres à se former dans le but de faire évoluer ses pratiques.

² Centre de Recherche Informatique de Montréal (Québec, Canada).

Le référentiel ISO-SPICE évolue parallèlement à la révision des normes ISO 9000. Cette révision vise une amélioration de la compatibilité entre les normes de management *qualité* ISO 9001, 9004 et les normes de management *environnemental* ISO 14001, 14004. Une seule norme subsiste (ISO 9001). Les normes ISO 9002 et 9003 disparaissent. L'orientation de la future ISO 9004 positionne cette norme comme un guide d'auto évaluation.

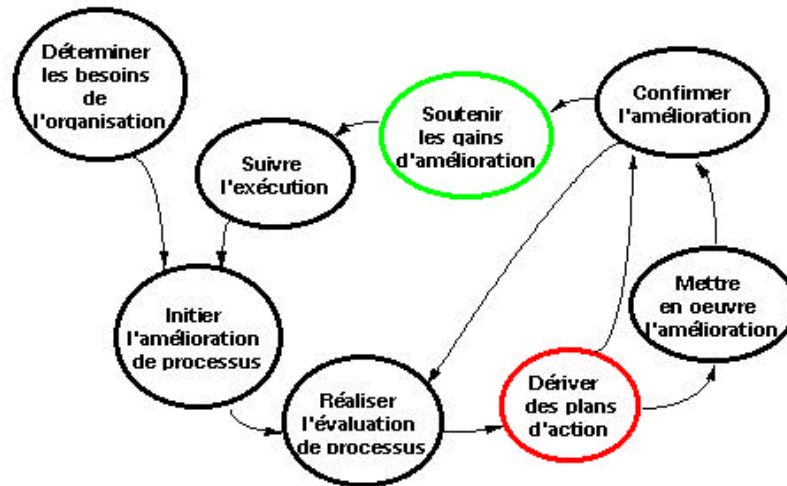


Figure 1. — Cycle d'amélioration continue des compétences

Tableau 1. — Pénétration du standard, source Martinig & Associés

Niveau de maturité	Domaine d'activité ou « préoccupation »	% réponses positives
2	Gestion de configuration	38%
2	Planification de projet	53%
2	Suivi de projet	50%
2	Assurance qualité	30%
2	Gestion des spécifications	59%
2	Gestion des sous-traitants	37%
3	Coordination entre les projets	46%
3	Gestion intégrée du développement	69%
3	Définition du processus	45%
3	Génie logiciel	43%
3	Gestion du processus	47%
3	Inspections et validations	31%
3	Formation	39%
4	Gestion quantitative du processus	18%
4	Qualité du logiciel	15%
5	Prévention des erreurs	40%
5	Gestion du changement	54%
5	Innovation technologique	45%

Les principaux changements dans la philosophie de ces normes se situent dans la recherche d'une qualité effective comprenant :

- la gestion des responsabilités organisationnelles,
- la gestion des ressources humaines et matérielles,
- la gestion des processus de production,

l'évaluation quantitative des améliorations.

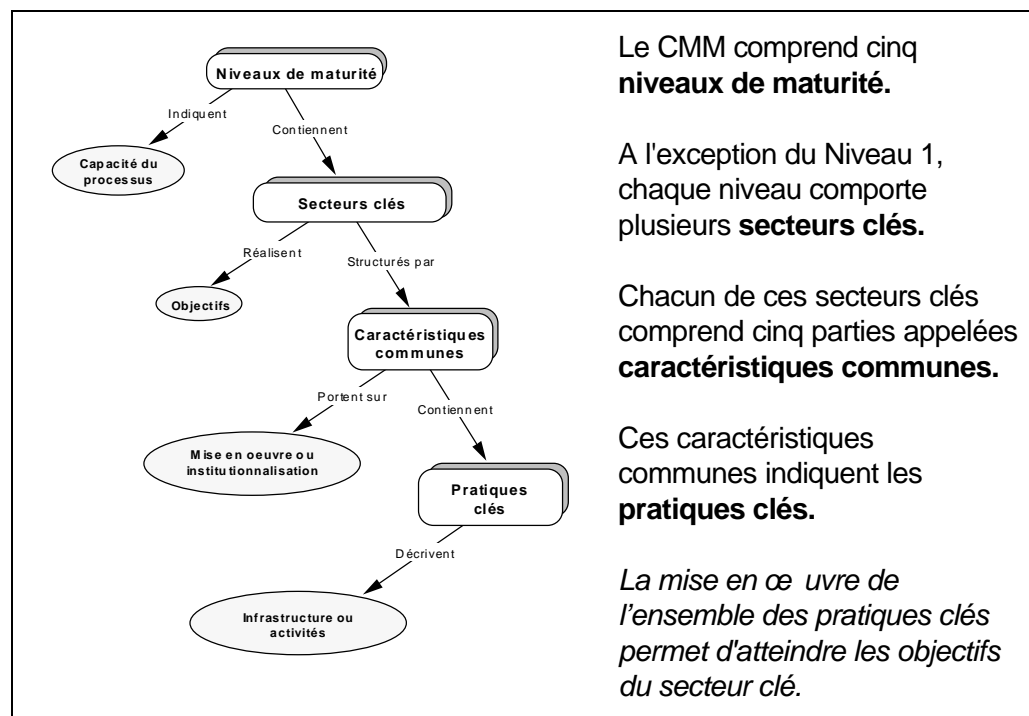
Les chiffres du *tableau 62* reflètent la pénétration des principales pratiques d'amélioration du processus de développement et présentent une vision par « niveau de maturité » et par « préoccupation » des pratiques « qualité du processus de développement » concrètement mises en œuvre par les entreprises ayant adopté une démarche standardisée.

Le *tableau 63* présente un comparatif (limité à la France) de la pénétration réciproque des standards CMM et ISO 9000.

Tableau 2. — CMM / ISO, source : Martinig & Associés

Utilisation des normes et standards	CMM	ISO 9000
Connait	57%	95%
Entendu parler	22%	51%
Utilisation prévue	5%	7%
En cours de mise en œuvre	3%	7%
Certifiés	1%	15%
Analysés et rejetés	5%	15%

1.2. Introduction à CMM



Le CMM comprend cinq **niveaux de maturité**.

A l'exception du Niveau 1, chaque niveau comporte plusieurs **secteurs clés**.

Chacun de ces secteurs clés comprend cinq parties appelées **caractéristiques communes**.

Ces caractéristiques communes indiquent les **pratiques clés**.

La mise en œuvre de l'ensemble des pratiques clés permet d'atteindre les objectifs du secteur clé.

Figure 2. — Principes et structure du CMM

CMM répertorie des pratiques de planification, d'ingénierie et de gestion qui, lorsqu'elles sont appliquées, améliorent la capacité de l'organisation à atteindre des objectifs de coûts, de délais, de qualité et de fonctionnalité. CMM dispose de l'immense avantage d'être modulaire et de discerner plusieurs niveaux de maturité, donc d'évolution souhaitable.

Cinq niveaux précis permettent à l'organisation de se situer et de fixer son objectif d'amélioration (*figures 79, 103 et 104*).

Structure du modèle d'organisation

Tableau 3. — Structures du CMM

Niveaux de maturité	Un niveau de maturité est un palier d'évolution bien défini dans le cheminement vers un processus logiciel mature. Cinq niveaux constituent la structure haute du CMM.
Capacité du processus	Cette notion recouvre la capacité d'obtenir des résultats en respectant un processus logiciel.
Secteurs « clés »	Chaque niveau de maturité comporte divers secteurs clés. Chaque secteur clé identifie un ensemble d'activités.
Objectifs	Les objectifs synthétisent les pratiques d'un secteur clé. L'observation de ces pratiques permet de déterminer si les activités d'un secteur clé sont réellement mises en œuvre.
Caractéristiques communes	Les caractéristiques communes indiquent si la mise en œuvre d'un secteur clé est efficace, durable et reproductible.
Pratiques « clés »	Chaque secteur clé est décrit en termes de pratiques clés (infrastructure et activités) dont la mise en œuvre contribue à satisfaire les objectifs de ce secteur clé. Les pratiques clés sont réparties en cinq caractéristiques communes : <ul style="list-style-type: none"> ▪ engagement de réalisation ; ▪ capacité de réalisation ; ▪ activités réalisées ; ▪ mesures et analyse ; ▪ vérification de mise en œuvre.

A l'exception du niveau 1 (état initial), chaque niveau comporte plusieurs **secteurs clés**. L'articulation de ces niveaux / secteurs est représentée dans la *figure 105*. Chacun de ces secteurs clés comprend cinq parties appelées **caractéristiques communes**.

Ces caractéristiques communes indiquent les **pratiques clés**. La mise en œuvre de l'ensemble des pratiques clés permet d'atteindre les objectifs du secteur clé (*tableau 64*).

Niveaux de maturité

NIVEAU 1 « INITIAL »

L'organisation ne dispose pas de procédures formalisées d'évaluation, de développement et d'évolution de ses applications. L'engagement de ses ressources humaines ne permet pas une capitalisation de l'expérience (turnover). Des crises surviennent en cours de projet. Lorsque l'échec se matérialise, le peu de méthode est abandonné pour tenter des raccourcis dans le processus de réalisation et de validation. Les efforts d'organisation régressent alors vers des pratiques d'engagements purement réactives, de type « codage et tests » qui amplifient la dérive.

NIVEAU 2 « REPRODUCTIBLE »

La gestion des nouveaux projets est fondée sur l'expérience mémorisée à l'occasion de projets semblables. L'engagement permanent des ressources humaines garantit une pérennité du savoir-faire dans la limite de leur présence au sein de l'organisation.

NIVEAU 3 « DEFINI »

Des directives de gestion de projet et des procédures en permettant l'application sont établies. Le processus standard de développement et d'évolution logiciel est documenté. Il intègre en un tout cohérent les procédés d'ingénierie logiciel et de gestion de projet. Un programme de formation est en place dans l'organisation afin que les utilisateurs et les informaticiens acquièrent les connaissances et les compétences nécessaires pour assumer les rôles qui leur ont été confiés.

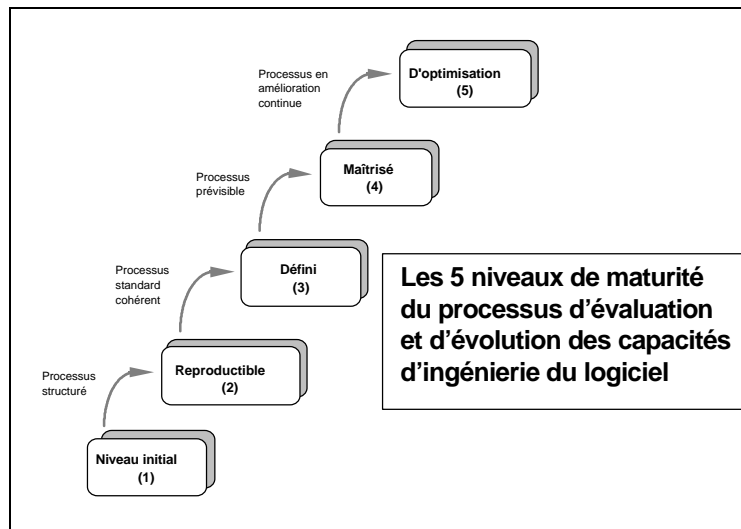


Figure 3. — CMM 5 paliers dans la rigueur

NIVEAU 4 « MAITRISE »

L'organisation se fixe des objectifs quantitatifs et qualitatifs. La productivité et la qualité sont évaluées. Ce contrôle se fonde sur la validation des jalons majeurs du projet dans le cadre d'un programme planifié de mesure.

NIVEAU 5 « OPTIMISE »

L'amélioration continue des processus est la principale préoccupation. L'organisation se donne les moyens d'identifier et de mesurer les faiblesses de ses processus. Une cellule de veille technologique identifie puis acquiert et met en œuvre les produits innovants. Elle recherche les pratiques d'ingénierie logiciel les plus efficaces, particulièrement celles dont la synergie permet l'amélioration continue de la qualité.

Secteurs clés par niveau

SECTEURS CLES NIVEAU 2 « REPRODUCTIBLE »

Au niveau 2, les secteurs clés sont axés sur les aspects relatifs à la mise en place des mécanismes élémentaires du contrôle de gestion de projet. Les secteurs clés du niveau 2 sont décrits ci-dessous :

◇ 2.1 Gestion des exigences

La *Gestion des exigences* vise à établir une compréhension des besoins. Cette compréhension aboutit à un contrat, fondement de la planification et des pratiques de gestion qui en découlent. La planification est décrite dans le secteur clé *Planification de projet*. Les pratiques de gestion dans le secteur clé *Suivi et supervision de projet*. Le contrôle des relations avec le client repose sur un processus efficace de contrôle des changements décrit dans le secteur clé *Gestion de configuration*.

◇ 2.2 Planification de projet

La *Planification de projet* vise à établir des prévisions raisonnables pour la mise en œuvre des travaux d'ingénierie logiciel et la gestion du projet. Ces prévisions sont indispensables à une gestion de projet efficace. L'absence de prévisions réalistes empêche toute gestion concrète du projet.

◇ 2.3 Suivi et supervision de projet

Le *Suivi et la supervision de projet* visent à donner une bonne perspective de l'avancement réel des travaux. Les gestionnaires peuvent ainsi intervenir efficacement lorsque la performance du projet s'écarte de façon significative des prévisions.

◇ 2.4 Gestion de la sous-traitance

La *Gestion de la sous-traitance* vise à sélectionner des sous-traitants qualifiés et à les gérer efficacement. La gestion de base mise en œuvre dans le cadre de ce secteur clé repose sur des considérations de *Gestion des exigences*, de *Planification de projet*, de *Suivi et supervision de projet* et sur la coordination nécessaire des activités *d'assurance qualité* et de *Gestion de configuration*. Ces mécanismes de contrôle sont appliqués de façon appropriée aux activités des sous-traitants.

◇ 2.5 Assurance Qualité

L'*Assurance Qualité logiciel* vise à fournir aux gestionnaires la vision appropriée sur le processus utilisé par le projet et sur les produits en élaboration. L'*Assurance Qualité logiciel* fait partie intégrante de la plupart des processus d'ingénierie logiciel et des processus de gestion.

◇ 2.6 Gestion de configuration

La *Gestion de configuration* vise à établir et à maintenir l'intégrité des produits du projet tout au long de son cycle de vie. La *Gestion de configuration* fait partie intégrante de la plupart des processus d'ingénierie logiciel et des processus de gestion.

SECTEURS CLES NIVEAU 3 « DEFINII »

Au niveau 3, les secteurs clés portent à la fois sur les aspects relatifs au projet et à l'organisation. À ce niveau, l'organisation met sur pied une infrastructure permanente. Elle permet d'institutionnaliser les processus efficaces d'ingénierie et de gestion de façon à ce qu'ils soient appliqués dans le cadre de tous les projets. Les secteurs clés du niveau 3 sont décrits ci-dessous :

◇ **3.1 Focalisation organisationnelle**

La *Focalisation organisationnelle* vise à établir une responsabilité organisationnelle quant aux activités reliées au projet en vue d'améliorer l'ensemble de la capacité du processus logiciel de l'organisation. Les activités de *focalisation organisationnelle* ont pour premier résultat de produire un ensemble d'acquis processus. Ceux-ci sont décrits dans la *Définition du processus de l'organisation* et sont ensuite utilisés dans le cadre de tous les projets logiciel (secteur clé *Gestion logiciel intégrée*).

◇ **3.2 Définition du processus**

La *Définition du processus de l'organisation* vise à développer et à maintenir un ensemble utilisable d'acquis processus logiciel. Ces acquis permettent l'amélioration de la performance obtenue d'un projet à l'autre. Ils constituent, pour l'organisation, une base indispensable à l'obtention de bénéfices cumulatifs sur le moyen terme et le long terme. Ces acquis représentent un fondement stable qui permet d'institutionnaliser la recherche de performance par le biais d'interventions telles que la formation (voir à ce sujet le secteur clé *Programme de formation*).

◇ **3.3 Programme de formation**

Le *Programme de formation* vise à développer les compétences et les connaissances des intervenants pour qu'ils puissent jouer efficacement le rôle qui leur a été attribué. Bien que la formation soit une responsabilité organisationnelle, chaque projet logiciel devrait donner lieu à l'identification des compétences essentielles à sa réalisation et inclure la formation requise pour répondre aux exigences particulières.

◇ **3.4 Gestion logiciel intégrée**

La *Gestion logiciel intégrée* vise à répertorier les activités d'ingénierie et de gestion logiciel sous forme d'un processus défini et cohérent adapté à partir du processus standard de l'organisation et des acquis processus associés (voir le secteur clé *Définition du processus de l'organisation*). Cette adaptation est fonction du contexte organisationnel et des exigences techniques du projet, tels qu'ils sont décrits dans le secteur clé *Ingénierie de produits logiciel*. La *Gestion logiciel intégrée* est le fruit des activités de *Planification de projet* et de *Suivi et supervision de projet* du niveau 2.

◇ **3.5 Ingénierie de produits logiciel**

L'*Ingénierie de produits logiciel* vise à exécuter systématiquement un processus d'ingénierie formellement défini. Il intègre l'ensemble des activités d'ingénierie permettant de produire efficacement des produits logiciel cohérents et corrects. Ce secteur clé décrit les activités techniques du projet, telles que l'analyse des exigences, la conception, la programmation et les tests.

◇ **3.6 Coordination intergroupes**

La *Coordination intergroupes* vise à établir un moyen permettant au groupe d'ingénierie logiciel de collaborer activement avec les autres groupes d'ingénierie. Cette synergie est nécessaire à la performance et à la qualité du projet. La *Coordination intergroupes* est un aspect interdisciplinaire de la *Gestion logiciel intégrée*. Elle dépasse le cadre de l'ingénierie logiciel. Pour être efficace, un processus logiciel doit être intégré et les interactions du groupe d'ingénierie logiciel avec les autres groupes doivent être coordonnées et contrôlées.

◇ **3.7 Revues par les pairs**

Les *Reuves par les pairs* visent à éliminer tôt et efficacement les défauts des produits. La revue par les pairs est une méthode indispensable et efficace utilisée en ingénierie de produits logiciel. Elle consiste en une série de vérifications croisées dont le but est de valider la qualité et la conformité technique du produit.

SECTEURS CLES NIVEAU 4 « MAITRISE »

Au niveau 4, les secteurs clés sont axés sur la compréhension quantitative et qualitative des produits du processus. Les deux secteurs clés à ce niveau, c'est-à-dire la *Gestion quantitative de processus* et la *Gestion de la qualité logiciel*, sont fortement interdépendants, comme on peut le constater ci-dessous :

◇ 4.1 Gestion quantitative de processus

La *Gestion quantitative de processus* vise à contrôler quantitativement la performance du processus appliqué dans le cadre du projet. La performance du processus correspond aux résultats réels obtenus. L'accent est mis sur l'identification des causes spéciales de variation observées dans un projet mesurable et stable. La correction éventuelle des circonstances à l'origine de ces variations transitoires est planifiée. Le secteur clé *Gestion quantitative de processus* permet d'ajouter un programme complet de mesures aux pratiques des secteurs clés *Définition du processus de l'organisation*, *Gestion logiciel intégrée*, *Coordination intergroupes* et *Revue par les pairs*.

◇ 4.2 Gestion de la qualité logiciel

La *Gestion de la qualité logiciel* vise à développer une compréhension mesurée de la qualité des produits logiciel issus du projet. Elle favorise la réalisation d'objectifs de qualité spécifiques. La *Gestion de la qualité logiciel* consiste à appliquer un programme complet de mesures aux produits de travail logiciel décrits dans le secteur clé *Ingénierie de produits logiciel*.

SECTEURS CLES NIVEAU 5 « OPTIMISE »

Au niveau 5, les secteurs clés couvrent la mise en œuvre d'un programme d'amélioration continue et mesurable du processus. Les préoccupations en découlant doivent être abordées simultanément au sein de l'organisation et dans le cadre des projets spécifiques. Les secteurs clés au niveau 5 sont décrits ci-dessous :

◇ 5.1 Prévention des défauts

La *Prévention des défauts* vise à identifier et à éliminer les causes de défauts. Le groupe de projet analyse les défauts, identifie leurs causes et modifie en conséquence le processus défini, tel que décrit dans le secteur clé *Gestion logiciel intégrée*. Les changements de processus de portée générale sont également effectués dans les autres projets, selon la procédure décrite dans le secteur clé *Gestion des changements du processus*.

◇ 5.2 Gestion des changements technologiques

La *Gestion des changements technologiques* vise à identifier les nouvelles technologies performantes (outils, méthodes et processus) et à en effectuer l'introduction ordonnée au sein de l'organisation. Cette pratique est mise en œuvre conformément au processus décrit dans le secteur clé *Gestion des changements du processus*. La priorité est donnée à la mise en œuvre pratique des innovations.

◇ 5.3 Gestion des changements du processus

La *Gestion des changements du processus* vise l'amélioration continue du processus de développement. Elle permet d'améliorer la qualité logiciel, d'augmenter la productivité et de diminuer la durée du cycle de développement de produits. La gestion des changements du processus s'applique à l'ensemble de l'organisation. Elle englobe les améliorations progressives découlant de la *Prévention des défauts* et les améliorations novatrices produites par la *Gestion des changements technologiques*.

Watts Humphrey, le créateur du CMM, travaille sur une adaptation des principes du CMM applicable au niveau de la ressource individuelle de conception-développement. La dénomination de ce futur modèle est « Personal Software Process ».

Ce chapitre se contente d'offrir une synthèse clarifiée de CMM. Lorsque des explications détaillées vous sembleront utiles, référez-vous aux documents d'origine disponibles sur Internet.

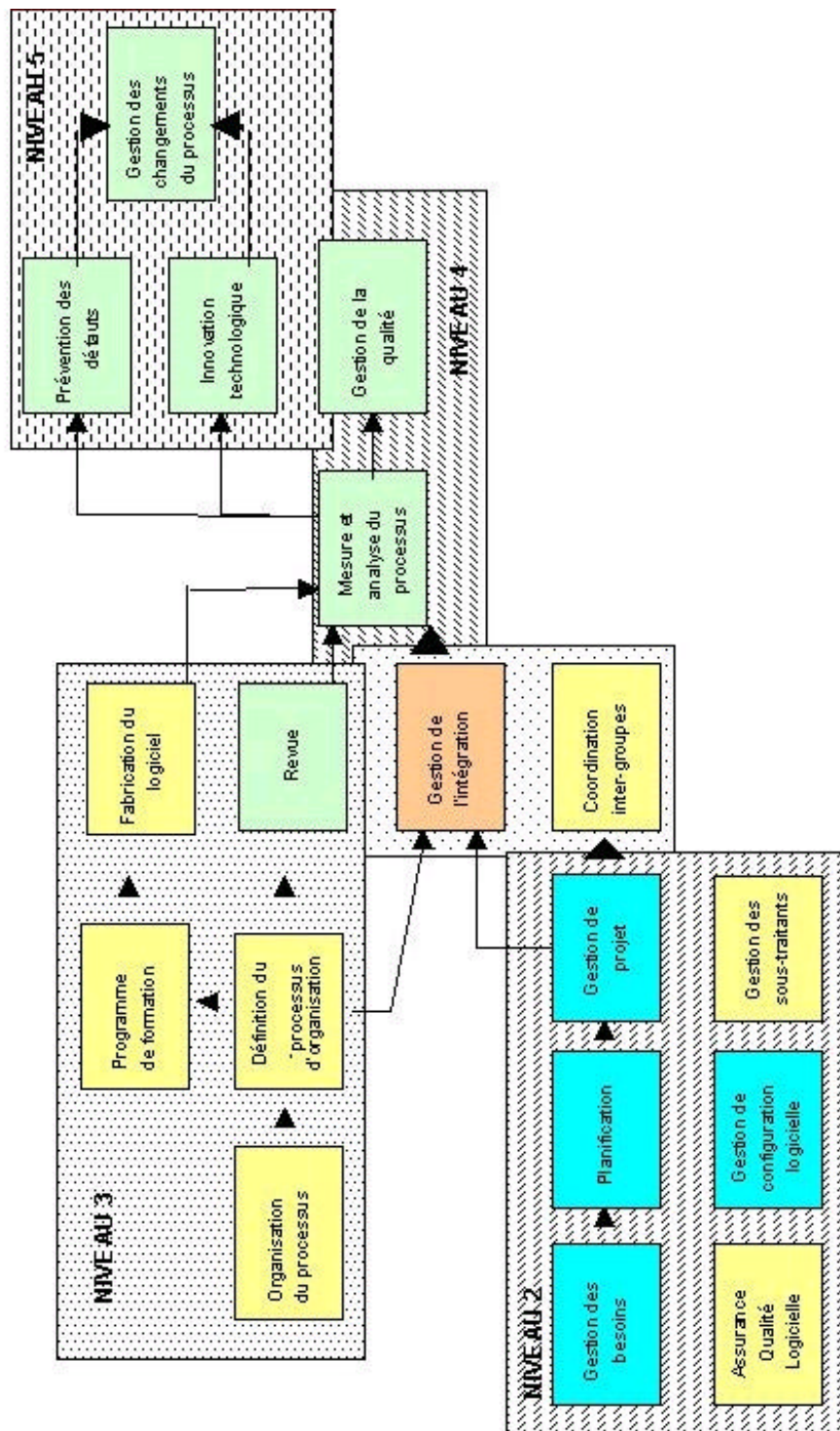


Figure 4. — CMM : articulation des niveaux / secteurs

1.3. RAD2 et les 5 niveaux de maturité CMM

Les principales critiques émises contre le RAD relèvent d'une méconnaissance totale de la méthode. Cette mauvaise presse faite au RAD est avant tout liée à la notion de rapidité qui est souvent considérée comme incompatible avec celle de qualité. Il aurait été logique de traduire RAD par « **développement maîtrisé d'application de qualité approuvée par les utilisateurs** » et la controverse actuelle entre méthode et RAD n'aurait pas eu lieu.

L'évaluation des techniques et du processus RAD2 à l'aune du CMM positionne la méthode RAD comme représentative de l'état de l'art en matière de méthodologie des développements informatiques.

Niveau 1 « initial »

CMM : L'organisation ne dispose pas de procédures formalisées de développement et d'évolution de ses applications. Lorsque l'échec de projet se matérialise, le peu de méthode est généralement abandonné pour tenter des raccourcis dans le processus de réalisation et de validation. Les efforts d'organisation régressent vers des pratiques d'engagements purement réactives de type «codage et tests» qui amplifient la dérive.

La méthode RAD repose sur un phasage et des techniques précises, elle ne relève donc pas de ce niveau.

Niveau 2 « reproductible »

CMM : La gestion des nouveaux projets est fondée sur l'expérience mémorisée à l'occasion de projets semblables.

Avec la méthode RAD, cela correspond au respect de la structuration des phases, de la dimension temporelle, de la validation des livrables, des architectures de conception et de réalisation. La planification s'effectue en phase de Cadrage à partir de la hiérarchie des fonctions à produire.

CMM : La gestion des nouveaux projets est fondée sur l'expérience mémorisée à l'occasion de projets semblables.

Le RAD rencontre ces exigences avec l'usage d'outils comme Évaluateur. Une utilisation partielle de la méthode RAD permet donc de se positionner à ce niveau.

Niveau 3 « défini »

CMM : Le processus standard de développement et d'évolution logiciel est documenté. Il intègre en un tout cohérent les procédés d'ingénierie logiciel et de gestion de projet.

Avec la méthode RAD, c'est le processus qualité d'ingénierie des développements.

CMM : Un programme de formation est en place dans l'organisation afin que les utilisateurs et les informaticiens acquièrent les connaissances et les compétences nécessaires pour assumer les rôles qui leur ont été confiés.

L'animateur RAD impose cette formation en préalable au début du projet et de préférence avant la réunion de lancement. Une application plus fidèle de la méthode RAD permet donc de se positionner à ce niveau.

Niveau 4 « maîtrisé »

CMM : L'organisation se fixe des objectifs quantitatifs et qualitatifs. La productivité et la qualité sont évaluées par le contrôle et la validation des jalons majeurs du projet dans le cadre d'un programme organisationnel de mesure.

Avec la méthode RAD, ces contrôles et ces mesures s'effectuent lors de la mise en place des jalons zéro-défaut et des revues de projet dont l'aboutissement est « l'état de livraison permanente » qui permet la réalisation de « Focus » de validation et d'avancement.

Niveau 5 « optimisé »

CMM : L'amélioration continue des processus est la principale préoccupation. L'organisation se donne les moyens d'identifier et de mesurer les faiblesses de ses processus et renforce ceux-ci de façon proactive afin de prévenir les dysfonctionnements. Une cellule de veille technologique identifie, acquiert et met en œuvre les produits innovants et les meilleures pratiques d'ingénierie logiciel dont la synergie permet l'amélioration continue de la qualité.

Le RAD offre naturellement les moyens de répondre à cette dernière exigence par l'engagement de tous dans le cadre d'équipes d'amélioration de la qualité (interview de groupe, validation permanente). Pour les informaticiens, cette notion se matérialise dans le SWAT (spécialisation, veille technologique, autonomie, réactivité, recherche de performance et de qualité au moindre coût). Pour la maîtrise d'ouvrage, ce sont les techniques d'enquêtes de satisfaction, de Benchmarking, et de veille concurrentielle.

1.4. Introduction à ISO-SPICE

« L'amélioration des processus logiciel doit elle-même être considérée comme un processus continu au sein duquel l'organisation se meut en permanence, où elle introduit de nouvelles pratiques, en modifie ou en supprime d'autres. Une étape importante est la détermination de l'état initial par une collecte de données, qui sera renouvelée pour constater, confirmer et réorienter l'amélioration du processus dans son ensemble. Le cadre de cette amélioration s'inscrit dans celui de l'amélioration de la qualité, décrit par l'ISO 9004-4 » (Thierry Tacquet, Objectif Technologie).

Historique : en janvier 1993, un programme de travail fut approuvé par ISO/IEC JTC1, il devait aboutir aux spécifications d'un standard en matière de pratique de développement d'applications : SPICE (*Software Process Improvement*). En 1995, un rapport technique de type 2 est présenté sous le titre générique de « Software Process Assessment ».

Le référentiel ISO/SPICE propose désormais un modèle de management de processus, ainsi qu'un ensemble cohérent d'exigences et de guides concernant l'évaluation et l'amélioration de ces processus. Le modèle est composé :

- d'une dimension « processus » qui identifie une quarantaine d'activités majeures ;
- d'une dimension « aptitude » qui propose des modalités génériques de mise en œuvre et de management de ces processus, selon une hiérarchie décrite en termes de niveaux d'aptitude.

Dans ses principes, SPICE représente une structure formalisée et normalisée dédiée à la gestion des exigences d'un processus de développement logiciel. Ce modèle peut être utilisé par les organisations soucieuses de maîtriser leurs procédés de planification et de pilotage ainsi que d'améliorer les procédures d'acquisition :

- de fournitures,
- de services de développement,

- de prestations de conduite de projet,
- d'exploitation, de maintenance et de support de leurs produits logiciels.

Dans ses pratiques, SPICE examine les procédés des organisations en matière de production du logiciel, puis détermine à quel niveau de maturité elles se situent. Le résultat de cette analyse en termes de risques et de faiblesses est ensuite utilisé pour orienter le processus d'amélioration.

SPICE est un modèle normatif qui se compose de 9 parties auxquelles correspondent autant de documents :

1. Introduction aux concepts fondamentaux.
2. Modèle de gestion de l'ingénierie des processus.
3. Processus d'évaluation du niveau d'aptitude.
4. Guide de conduite de l'évaluation.
5. Modèle d'évaluation, guide des indicateurs, outils.
6. Guide pour la qualification des évaluateurs.
7. Guide de mise en œuvre de l'amélioration des processus.
8. Guide de détermination des aptitudes des fournisseurs.
9. Dictionnaire, vocabulaire et terminologie.

La *figure 106* décrit les principes d'interaction des 9 parties.

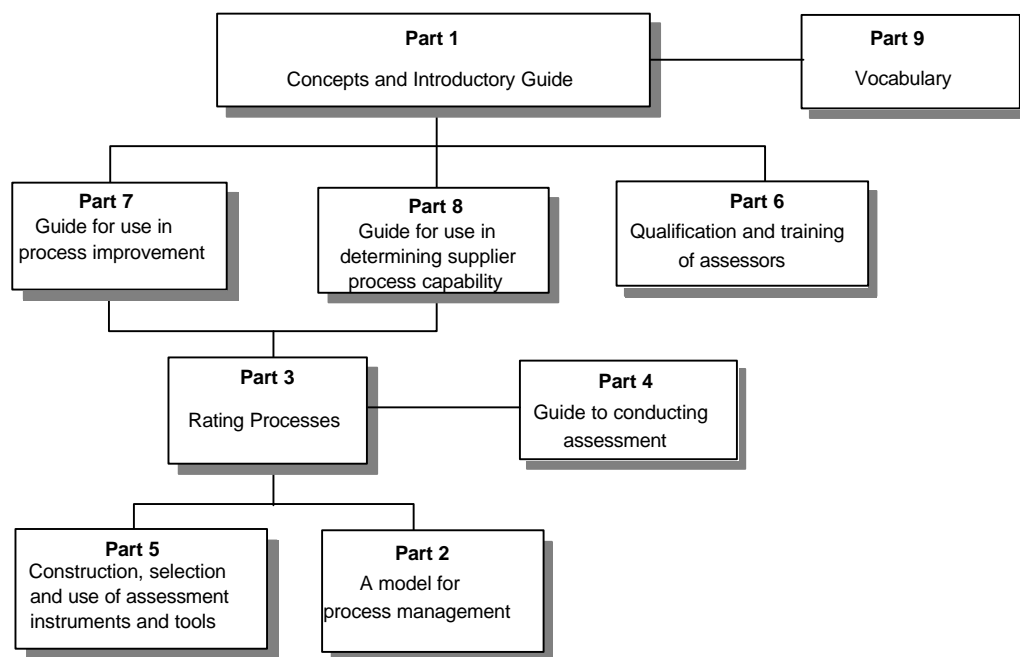


Figure 5. — SPICE : interactions des 9 parties

A ces 9 parties s'ajoute un fascicule de documentation, élaboré en France, dont l'objet est de présenter de manière synthétique l'utilisation de la norme. La référence est : FD Z 67-910 (décembre 1998) « Ingénierie et qualité du logiciel - Introduction au référentiel ISO/SPICE (ISO/CEI TR 15504) et à son utilisation pour le management de la qualité des processus du logiciel ».

SPICE détermine la maturité d'une organisation de production de « logiciel » selon 6 niveaux (*tableau 65*). Le processus se décompose en 5 catégories de préoccupations (*tableau 66*).

Tableau 4. — SPICE : 6 niveaux de maturité

Niv	Définition d'origine	Traduction	Concepts
0	Not-Performed	Non effectué	Mise en œuvre de pratiques empiriques
1	Performed- Informally	Effectué de façon informelle	Mise en œuvre de pratiques définies
2	Planned-and- Tracked	Planifié et Suivi	Planifier des procédures définies, ordonnancer, puis suivre leur exécution
3	Well-Defined	Bien défini	Définir un processus formalisé et standardisé, puis le mettre systématiquement en œuvre
4	Quantitatively- Controlled	Maîtrisé quantitativement	Établir des objectifs-qualité mesurables, mettre en place des indicateurs, puis piloter leur suivi
5	Continuously- Improving	En amélioration permanente	Améliorer les pratiques organisationnelles et accroître l'efficacité du processus

Tableau 5. — SPICE : 5 catégories de préoccupations

Catégorie	Définition d'origine	Zone de processus	Nb de Processus	Nb de pratiques
CUS	Customer- Supplier	Client-Fournisseur	8	39
ENG	Engineering	Ingénierie	7	32
PRO	Project	Projet	8	30
SUP	Support	Support	5	32
ORG	Organization	Organisation	7	48
		TOTAL	35	201

Pour une vue complète du processus ou pour accéder à des communications sur le sujet, visitez le WEB « Toile Francophone du Génie Logiciel » (<http://web.cnam.fr/TFGL>).

Pour conclure sur l'intérêt de la normalisation et de l'industrialisation de la production du logiciel, il n'est pas inutile de rappeler quelques statistiques.

Une étude conduite par *KPMG Peat Marwick* en 1998 démontre que les principales fautes commises en matière de gestion de projets informatiques étaient les suivantes :

- objectifs du projet mal interprétés (51 %),
- estimation et planification déficiente (48%),
- technologies mal maîtrisées (45%),
- inadéquation de la méthode au projet (42 %),
- insuffisance d'expérience des ressources (42 %),
- matériel ou logiciel pénalisant ou déficient (42 %).

Dans le même temps un « survey » du *Standish Group* effectué sur 8000 projets de développement met en évidence que les projets en moyenne dépassent :

- de 90 % les budgets prévus,
- de 120 % les délais escomptés.

Les responsables ont ensuite identifié les facteurs de succès qu'ils comptent mettre en œuvre lors de leur prochain projet :

- formation de compétences en pilotage de projet (86 %) ;
- obtention d'études externes de faisabilité (84 %) ;
- amélioration des communications « utilisateurs » (68 %) ;
- sécurisation par recours à des expertises externes (56 %).

2. Bibliographies et références

2.1. Bibliographie RAD, Conduite de projet

Martin James, *Rapid Application Development*, Macmillan, 1991.

Vickoff Jean-Pierre, *RAD - Développement Rapide d'Applications, Méthodes et outils, les règles à respecter dans le développement d'application client-serveur*, MGI, 1995, puis Macmillan, 1996.

Vickoff Jean-Pierre, *REENGINEERING, Vite fait bien fait, Le Paradigme du futur immédiat*, QI, 1998.

Vickoff Jean-Pierre, *Réingénierie du développement d'application, Méthodes, processus, techniques et pratiques de conduite de projet sous contraintes*, Gartner Group, 1999.

Boehm B. & Bose P. *A Collaborative Spiral Software Process Model*, USC, 1994

Bouchy S. *L'Ingénierie des systèmes d'information évolutifs*, Eyrolles, 1994

Clark B. *The Effects of software process maturity on software development effort*, USC, 1997

Englewood C. *JAD the Group Session, Approach to system design*, Prentice Hall, 1991.

Egyed A. & Boehm B. *Telecooperation Experience with the WinWin System*, USC, 1998

Henry A. & Monkam-Daverat L. *Rédiger les procédures de l'entreprise*, Les Éditions d'organisation, 1995.

Mc Carty J. *54 Règles d'or pour un grand logiciel*, Microsoft Press, 1997.

McConnell S. *Stratégie de développement rapide*, Microsoft Press, 1996.

Panet G. & Letouche R. *Merise 2, modèles et techniques Merise avancés*, Les Éditions d'organisation, 1994.

Yourdon E. *Modern Structured Analysis*, Englewood Cliffs, 1989.

2.2. Bibliographie Management, Qualité

Ballay J-F. *Capitaliser et transmettre les savoir-faire de l'entreprise*, Eyrolles, 1997.

Bartoli A. *Communication et organisation, pour une politique générale cohérente*, Les Éditions d'organisation, 1994.

Beaudoin P. *La Gestion du changement, une approche stratégique pour l'entreprise en mutation*, Stratégies d'entreprise, 1990.

Champy J. *Reengineering Management*, HarperCollins Publishers, 1995.

Chapman C. & Ward S. **Project Risk Management : Processes, Techniques and Insights**, Edition Wiley, 1997.

Courtot E. **La gestion des risques dans les projets**, Edition Economica, 1998

Delafollie G. *Analyse de la valeur*, Hachette, 1991.

Hammer M. & Champy S. *Reengineering the Corporation, A manifesto for Business Revolution*, Harper Business, 1993.

Henry A. & Monkam L., *Rédiger les procédures de l'entreprise*, Éditions d'organisation, 1995.

Laudoyer G. *La Certification, un moteur pour la qualité*, Les Éditions d'organisation, 1993.

Lamprecht J-L. *ISO 9000 Se préparer à la certification*, AFNOR, 1994.

Lefebvre C. *Concevoir et conduire un projet de changement*, Les Presses du management, 1997.

Maders H-P. & Gauthier E. & Le Gallais C. *Conduire un Projet d'Organisation*, Editions d'Organisation, 1998

Nizet J. & Huybrechts C. *Interventions systémiques dans les organisations, Intégration des apports de Mintzberg et de Palo Alto*, Editions DeBoeck Université, 1998

Mucchielli R. *L'Interview de groupe*, ESF, 1987.

Renaud-Coulon A. *La Désorganisation compétitive*, Maxima, 1996

Richard J. & Paula S. *Delivering Results: Evolving BPR from Art to Engineering*, Texas A&M University, 1998

Sary P. *La Stratégie de la programmation neurolinguistique dans l'entreprise*, Editions Retz, 1990.

Wenger E. 1998, *Communities of Practice - Learning, Meaning and Identity*, Cambridge University Press.,

2.3. Divers documents, normes, standards

En ce qui concerne les travaux du SEI sur CMM :

- SEI/CRIM, 1993, Paulk *Modèle d'évolution des capacités logiciel*.
- IEEE Software, 1993, Paulk & M.C. Curtis & B. & Chrissis, M.B. & Weber, C.V. *Capability Maturity Model, Version 1.1*, Vol. 10, No. 4, July 1993, pp. 18-27.
- SEI, 1995, Paulk *The Capability Model : Guidelines for Improving the Software Process*.

En ce qui concerne les normes ISO (Evaluation, Qualité, SPICE) :

- ISO 9001-1994, *Model for quality assurance in design, development, production, installation and servicing*.
- ISO 9000-3-1991, *Quality management and quality assurance standards (Part 3: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software)*.
- ISO 9004-4-1993, *Quality management and quality system elements (Part 4: Guidelines for quality improvement)*.
- ISO 8402 et X 50-125, *Management de la qualité et assurance de la qualité – Vocabulaire*
- ISO/IEC12207-1-1994, *Software life cycle processes* (ingénierie du logiciel)
- ISO/IEC12119-1995, *Software products - Evaluation and test*.
- ISO/IEC 9126-1991, *Software quality characteristics*.
- ISO 9294, *Ligne directrice pour la gestion de la documentation technique du logiciel*

Ainsi que diverses références aux documents suivants :

- Craigmyle, M., and I. Fletcher, *Improving IT effectiveness through software process assessment*, Software Quality Journal, Vol. 2, pp 257-264 (1993).
- Humphrey, W.S., *Managing the Software Process*, Addison Wesley, 1989.
- Kuvaja, P., Simila, J., Krzanik, L., Bicego, A., Koch, G. and Saukkonen, S., *Software Process Assessment and Improvement: The BOOTSTRAP Approach*. Blackwell, 1994.
- Bell Communications Research, Inc., *Quality Program Specification for Surveillance Management Process(SMP) Software (General)*, QPS 88.001, Issue 1.
- AFCIQ-PDL, *Recommandation de plan de développement logiciel*.

- AFICIQ-PAQL, *Recommandation de plan d'assurance qualité logiciel*.
- SCE - Software Capability Evaluation Training Guide / SPA - Software Process Assessment Training Guide, Software Engineering Institute, Pittsburgh Pennsylvania.
- F. Coallier, N. Gammage, A.W. Graydon, Trillium - *Software Process Self-assessment Capability Assessment*, Bell Canada, PI Q008, Issue 4.0, March 1993.

2.4. Principaux WEB et contributeurs

AFAV, Association française pour l'analyse de la valeur, <http://www.afav.asso.fr>

AFITEP, Association francophone de management de projet, <http://www.afitep.fr>

American Society for Quality, <http://www.asqc.org>

Business Processes Ressources Centre, <http://bprc.warwick.ac.uk/bp-gold.html>

Business Process Reengineering Advisory Group,
<http://www.eil.utoronto.ca/tool/list.html>

COCOMO, Constructiv Cost Model,
<http://sunset.usc.edu/CORADMO/coradmo.html>

CORADMO, Constructiv Cost Model,
<http://sunset.usc.edu/CORADMO/coradmo.html>

CRIM, Centre de génie logiciel appliqué de Montréal, <http://www.CRIM.CA/cgla>

DACS, Data Analysys Center for Software (Cost estimation),
<http://mach10.rome.kaman.com/cgi-bin/keylist>

Esprit Project 27599 - RAMSES RAD for MSS,
<http://www.esi.es/ESSI/Reports/All/27599/>

IDEF method, <http://www.idef.com>

IFPUG, International Function Point Users Group, <http://www.ifpug.org>

ISO, Software Process Assessment, <http://www.sqi.gu.edu.au/sc7/wg10/>

SEI, The Software Engineering Institute, <http://www.sei.cmu.edu>

Software Engineering & its Applications (TFGL du CNAM),
<http://web.cnam.fr/TFGL>

Staffordshire University
<http://www.soc.staffs.ac.uk/~cmtrmk/rad/new/course/rad1.htm>

UML, Informations sur UML, <http://www.essaim.univ-mulhouse.fr/uml>

University of California, Davis,
http://sysdev.ucdavis.edu/webadm/document/rad_toc.htm

Pour terminer, je remercie particulièrement les participants assidus des réunions de la commission AFITEP *IM-CP-Processus*. Leur contribution à la formalisation des divers aspects de la conduite de projet, a facilité la rédaction de plusieurs parties de cet ouvrage.

2.5. Index

A	L
AFNOR · 20	livrable · 15
B	livraison · 16
bénéfices · 12	N
C	normalisation · 19
certification · 20	P
D	planification · 11, 16, 19
dérive	planification
pratiques · 10, 15	cadrage · 15
E	pratiques · 9
échec	Planification
crise · 10, 15	de projet · 11
état de livraison permanente · 16	processus
F	CMM · 10
Focus · 16	RAD/CMM · 15
formation	productivité
programme · 10, 12	évaluation · 10, 16
G	R
Gartner Group · 20	raccourci
gestion des exigences · 11	tentative · 10, 15
I	réactivité
ingénierie	optimisation · 16
CMM · 9, 10	René Descartes · 5
ISO · 5, 6, 7, 8, 16, 17, 20, 21, 25	S
J	SEI · 6
James Martin · 20	SPICE · 5, 6, 7, 16, 17, 18, 21
	SWAT · 16
	synergie · 10, 16
	U
	utilisateur
	formation · 10

2.6. Figures

Figure 102. — Cycle d'amélioration continue des compétences	7
Figure 103. — Principes et structure du CMM	8
Figure 104. — CMM 5 paliers dans la rigueur	10
Figure 105. — CMM : articulation des niveaux / secteurs	14
Figure 106. — SPICE : interactions des 9 parties	17

2.7. Tableaux et listes

Tableau 62. — Pénétration du standard, source Martinig & Associés	7
Tableau 63. — CMM / ISO, source : Martinig & Associés	8
Tableau 64. — Structures du CMM	9
Tableau 65. — SPICE : 6 niveaux de maturité	18
Tableau 66. — SPICE : 5 catégories de préoccupations	18

Software Engineering Institute (SEI)

<http://www.sei.cmu.edu/>

Software Process Improvement and Capability dEtermination

<http://www-sqi.cit.gu.edu.au/spice/>

STRUCTURE DU RAPPORT OPERATIONNEL PILOTE 2010

Pilote 2010 n'est pas un livre dans le sens littéraire du terme mais un document d'étude et de référence opérationnelle.

Dans une première approche, il est préférable de parcourir l'ouvrage dans son entier, afin de bénéficier d'une compréhension globale de l'ensemble des éléments traités et de leurs relations.

Il est aussi possible d'accéder directement à une des parties ou à un des chapitres au gré du besoin immédiat en utilisant le sommaire, les index de mots « clés », des figures ou des tableaux.

8 parties structurent Pilote 2010		Chapitres
1	Méthode pour une conduite de projet « haute performance »	1, 2 et 3
2	Plan d'Assurance-Qualité et techniques de pilotage « sécurisé »	4 à 15
3	Processus RAD 2, option « progiciel », choix d'externalisation	16, 17 et 18
4	Modélisation technique (Merise, Flux, UML) et ingénierie « métier »	19 et 20
5	Pilotage « étendu » des risques, évaluation des charges, optimisation de l'environnement	21 et 22
6	Standards d'évaluation et d'amélioration des capacités de production du logiciel SEI-CMM / ISO-SPICE	23 et 24
7	Documents de projet et rapport de pilotage	25
8	Origines et fondement de la méthode RAD	26
	<i>Informations et documents annexes</i>	27

L'aspect pragmatique prévaut dans chaque chapitre. Pourtant, la culture méthodologique française et le sens de la justification théorique n'en sont pas totalement absents. Le style de rédaction est souvent typique du français pratiqué au Canada (structure anglo-saxonne, langue française, phrases courtes, peu de liaisons).

Placé en annexe, le chapitre 25 expose le contexte qui a justifié l'émergence de la méthode RAD. Sa lecture préalable à celle des autres chapitres est conseillée à tous ceux pour qui le RAD est une nouveauté et qui souhaiteraient prendre connaissance de ses origines et de ses fondements.



Jean-Pierre Vickoff est avant tout pilote de projet et concepteur-développeur. Canadien et Français, il s'est spécialisé en Amérique du Nord dans les méthodes de développements sous contraintes.

Consultant en qualité et en performance, il produit des communications pour la presse professionnelle et des rapports pour le Gartner Group. Il est l'auteur de plusieurs ouvrages, dont *RAD* et *Réingénierie du développement d'applications*. Il préside la commission conduite de projet IM-CP de l'AFITEP et intervient auprès des établissements membres de la Conférence des Grandes Ecoles.

Pilote 2010 n'est pas un livre dans le sens littéraire du terme mais un support d'étude, de réflexion et d'action. *Pilote 2010* instrumente l'ensemble des méthodes, pratiques et outils indispensables à la productivité du développement ainsi qu'à la qualité des applications. *Pilote 2010* constitue une source unique de références opérationnelles en matière de :

- Conduite de projet (classique, décisionnel, NET)
- Plan d'Assurance-Qualité du développement
- Pilotage des risques et métrique quantitative des charges
- Méthodes, pratiques, outils de la performance
- Processus d'ingénierie du développement (RAD2-UML)
- Processus spécialisé e-commerce
- Gestion des *Exigences*, des *Validations* et des *Divergences*
- Gestion de la communication et des rapports entre acteurs
- Notations et techniques de modélisation (Merise, Flux, Objet)
- Ingénierie « métier » et accompagnement organisationnel, BPR, MTQS
- Industrialisation, choix de solutions, progiciel, externalisation
- Planification, documentation, plan de tests, gestion de configuration
- Standards d'évaluation et d'amélioration (SEI-CMM/ISO-SPICE)

Ces thèmes sont traités sous une forme directement utilisable.

Pilote 2010 propose aussi dans un format MS Office :

- Un ensemble normalisé de documents de projet
- Un plan d'Assurance-Qualité générique
- Divers processus (RAD2, progiciel, externalisation)
- Un rapport standard de pilotage de projet
- Un logiciel d'évaluation de charge, d'optimisation et de négociation
- Une présentation et une formation à la méthode RAD

Dans un univers de performances, où l'obligation de résultats s'impose à tous progressivement, la rigueur devient l'alliée du changement. Les pilotes de SI se doivent alors d'envisager immédiatement l'acquisition d'une nouvelle culture « projet » et de l'appliquer à l'évolution de leur profession. Dans cette optique, *Pilote 2010* leur propose une couverture complète et en l'état de l'art, des évolutions du métier. Cette ambition positionne l'ouvrage comme la Bible du développement d'applications pour la décennie en cours.