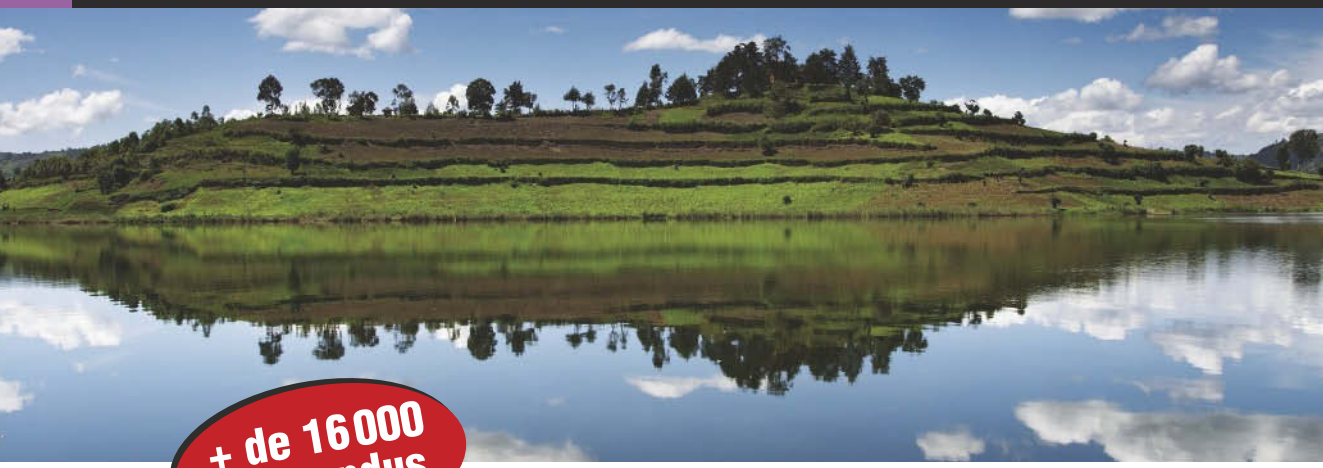
MANAGEMENT SYSTÈMES D'INFORMATION

MANAGEMENT D'UN PROJET SYSTÈME D'INFORMATION

Principes, techniques, mise en œuvre et outils



+ de 16000 ex. vendus

Chantal Morley

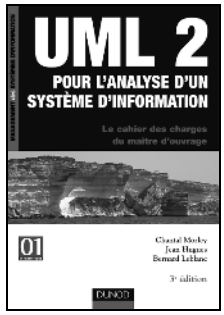
6e édition

DUNOD

MANAGEMENT D'UN PROJET SYSTÈME D'INFORMATION

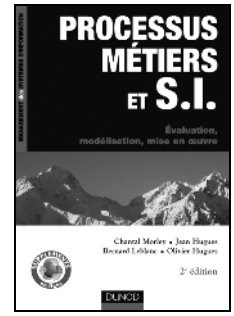
Principes, techniques, mise en œuvre et outils

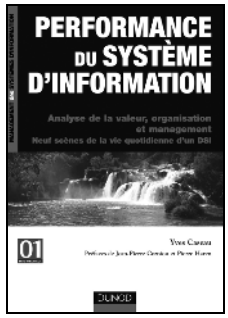
Consultez nos parutions sur dunod.com



UML 2 pour l'analyse d'un système d'information
Le cahier des charges du maître d'ouvrage
3e édition
Chantal Morley, Jean Hugues, Bernard Leblanc
240 pages
Dunod, 2006

Processus métiers et SI
Évaluation, modélisation, mise en œuvre
Chantal Morley, Jean Hugues, Bernard Leblanc, Olivier Hugues
288 pages
Dunod, 2007





Performance du système d'information
Analyse de la valeur, organisation et management
Yves Caseau
272 pages
Dunod, 2007

MANAGEMENT D'UN PROJET SYSTÈME D'INFORMATION

Principes, techniques, mise en œuvre et outils

Chantal Morley
Professeur à l'Institut Télécom/
Télécom & Management Sud Paris

6e édition

DUNOD

Marques déposées

Price-S : RCA Price System
Estimacs : Computer Associates
Winelys : Oresys Lysys
Project 2003 : Microsoft Corporation
PMW : Applied Business Technology Corp
AMI TOOL : Object Technology
PMBOK : Project Management Institute (PMI)

Illustration de couverture : Dmitry Pichugin - Fotolia.com

Le pictogramme qui figure ci-contre mérite une explication. Son objet est d'alerter le lecteur sur la menace que représente pour l'avenir de l'écrit, particulièrement dans le domaine de l'édition technique et universitaire, le développement massif du photocopillage.
Le Code de la propriété intellectuelle du 1er juillet 1992 interdit en effet expressément la photocopie à usage collectif sans autorisation des ayants droit. Or, cette pratique s'est généralisée dans les établissements d'enseignement supérieur, provoquant une baisse brutale des achats de livres et de revues, au point que la possibilité même pour les auteurs de créer des œuvres nouvelles et de les faire éditer correctement est aujourd'hui menacée.
Nous rappelons donc que toute reproduction, partielle ou totale, de la présente publication est interdite sans autorisation de l'auteur, de son éditeur ou du Centre français d'exploitation du droit de copie (CFC, 20, rue des Grands-Augustins, 75006 Paris).

© Dunod, Paris, 2008
© Masson, Paris, 1998 et © Dunod, Paris, 1999, 2001, 2004, 2006

ISBN 978-2-10-053551-4

Le Code de la propriété intellectuelle n'autorisant, aux termes de l'article L. 122-5, 2° et 3° a), d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective » et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause est illicite » (art. L. 122-4).
Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contrefaçon sanctionnée par les articles L. 335-2 et suivants du Code de la propriété intellectuelle.

Ce qu'on ne sait pas, on peut l'apprendre, qui veut s'en donner la peine, bel ami cher. Tout métier requiert effort, courage et persévérance : quand ces trois sont réunis, il n'est rien dont on ne se rende maître. Mais vous à qui cet art fut toujours inconnu et qui ne vîtes jamais homme s'y exercer, comment vous en reviendrait-il honte et blâme, si vous y êtes novice ?

Chrétien de Troyes, *Perceval le Gallois*

Avant-propos

Le présent ouvrage s'adresse à tous ceux qui ont à participer à la gestion de projet dans le domaine des systèmes d'information, aussi bien du côté « métier » que du côté logiciel.

Il vise à présenter et à illustrer à travers de nombreux exercices et cas pratiques, souvent inspirés de cas réels, l'état de l'art des techniques utilisables. Mais ces techniques ne sont pas des éléments indépendants et banalisés d'une boîte à outils du manager de projet. Elles prennent leur sens et leur utilité par rapport à des problèmes ou difficultés surgissant dans la pratique. C'est pourquoi ce livre est organisé en suivant un cycle général de préoccupations rencontrées, d'une manière ou d'une autre, sur tout projet : découper, estimer les charges, planifier, prendre en compte les risques, reconnaître la dimension humaine, intégrer les principes de la qualité… Pour chaque aspect, les techniques seront mises en regard d'un problème dans une situation donnée.

Cependant, la présentation normalisée du cadre du management de projet s'inscrit aujourd'hui dans une approche par processus qui éclaire de façon quasi-indépendante les différentes dimensions à gérer. Des activités d'intégration visent à en assurer la cohérence. Pour faciliter la compréhension et l'apprentissage, nous avons toutefois privilégié une logique séquentielle à dérouler de façon cyclique : planification – suivi – pilotage, jusqu'à la clôture du projet. En effet, certaines techniques peuvent concerner plusieurs aspects : nous situerons leur finalité globale. Par exemple, les techniques d'estimation des charges sont utilisées aussi bien pour le management des délais que pour le management des coûts. C'est pourquoi, l'ordre de présentation est calé sur l'ordre habituel de la première utilisation de la technique dans un cycle projet, les aspects qualité ayant été placés en dernier : découpage, estimation des charges, planification, aspects humains, analyse et contrôle des risques et pilotage.

Cependant, nous avons établi un lien clair avec le cadre par processus. Nous proposons notamment une aide à la certification en management de projet du PMI, qui s'inscrit dans cette perspective processus. Le lecteur qui vise un tel objectif trouvera dans cet ouvrage, non seulement des apports concrets et nuancés sur la façon de mener un projet système d'information, mais pourra situer ces connaissances dans un référentiel structuré en processus. Dans cette 6e édition, nous avons introduit pour chaque aspect du management de projet une perspective particulière sur les méthodes agiles.

Table des matières

Avant-propos vii

Introduction 1

Première partie – Principes et techniques

Chapitre 1 – Problématique du management de projet 5

1.1 Origine et évolution du management de projet 5

1.2 Définitions d'un projet 6

1.2.1 L'approche générale 6

1.2.2 Les définitions normalisées 8

1.3 Qu'est-ce que le management de projet ? 9

1.3.1 L'approche générale du management de projet 9

1.3.2 La définition normalisée du management de projet 12

1.4 Le management des projets système d'information 13

1.4.1 Caractéristiques d'un projet système d'information 13

1.4.2 Les objectifs des projets systèmes d'information 15

1.4.3 La complexité d'un projet système d'information 17

1.5 Management de projet et méthodes agiles 18

Chapitre 2 – Le découpage d'un projet et les modèles de cycle de vie 25

2.1 Les principes du découpage 25

2.2 Les découpages normalisés 28

2.3 Le découpage structurel 31
2.4 Le cycle de vie standard 32
2.4.1 Le contenu du cycle de vie standard 32
2.4.2 Les problèmes posés par le découpage standard 33
2.5 Le découpage classique 35
2.5.1 Le schéma directeur 36
2.5.2 L'étude préalable 36
2.5.3 L'étude détaillée 38
2.5.4 L'étude technique 38
2.5.5 La réalisation 38
2.5.6 La mise en œuvre 39
2.5.7 La qualification 39
2.6 Les modèles de développement 39
2.6.1 Le modèle du code-and-fix 40
2.6.2 Le modèle de la transformation automatique 40
2.6.3 Le modèle de la cascade 41
2.6.4 Le modèle en V 41
2.6.5 Le modèle en W 42
2.6.6 Le modèle de développement itératif 43
2.6.7 Le modèle de la spirale 44
2.7 Les modèles de cycle de vie spécifiques 45
2.7.1 Le cycle ERP 45
2.7.2 Le modèle RUP 46
2.8 Les modèles de cycle de vie des méthodes agiles 48
2.8.1 Le modèle RAD 48
2.8.2 Le modèle DSDM 49
2.8.3 Le modèle XP 50
2.8.4 Le modèle SCRUM 52
2.9 Le découpage structurel dans le cas des méthodes agiles 53

Chapitre 3 – L'estimation des charges 55
3.1 La charge et la durée 55
3.2 Les différents besoins d'estimation 56
3.3 Les différentes méthodes d'estimation 58

3.3.1 Les non-méthodes 58
3.3.2 Les méthodes 58
3.4 Le jugement d'expert : la méthode Delphi 61
3.5 L'estimation par analogie 62
3.5.1 La méthode de répartition proportionnelle 62
3.5.2 La méthode des ratios 64
3.6 L'estimation ascendante : l'évaluation analytique 66
3.7 L'estimation paramétrique : le modèle Cocomo 67
3.8 L'estimation paramétrique : la méthode des points de fonction 71
3.8.1 Les composants fonctionnels 71
3.8.2 La complexité et le nombre de points de fonction 72
3.8.3 L'ajustement de la taille 74
3.8.4 La transformation du nombre de points de fonction en charge 74
3.9 L'estimation probabiliste 75
3.10 Démarche générale d'estimation 76
3.11 L'estimation dans les méthodes agiles 78

Chapitre 4 – Les techniques de planification des délais 81
4.1 L'utilisation de la planification 81
4.2 Les graphes d'ordonnancement 82
4.3 Les types de liens 83
4.4 La méthode du chemin critique 86
4.5 Le diagramme de Gantt 91
4.6 La planification opérationnelle 93
4.6.1 La prise en compte des contraintes 93
4.6.2 L'utilisation des marges 94
4.6.3 Le nivellement 95
4.6.4 Le lissage 97
4.7 Le PERT probabiliste 98
4.8 La planification des délais dans les méthodes agiles 100

Chapitre 5 – La dimension humaine d'un projet 103

5.1 L'organisation du travail 103
5.1.1 La division et la répartition du travail 103
5.1.2 La coordination du travail 104
5.1.3 L'administration de données (AD) 106
5.1.4 Les parties prenantes et les structures-types d'un projet 108
5.1.5 La composition de l'équipe de projet 111

5.2 La participation des utilisateurs 112
5.2.1 La détermination des besoins 113
5.2.2 La prise de décision 113
5.2.3 Le changement 114

5.3 Le rôle du chef de projet 114
5.3.1 Définition du chef de projet 114
5.3.2 Les responsabilités du chef de projet 114
5.3.3 Les styles de management 116
5.3.4 Les théories de la motivation 118
5.3.5 La gestion des conflits 122

5.4 La gestion du changement 123
5.4.1 La nature et le contenu de la gestion du changement 123
5.4.2 La résistance au changement 125
5.4.3 La place de la confiance 127
5.4.4 Les stratégies de déploiement 128
5.4.5 Les activités de la gestion du changement 129

5.5 L'organisation du travail dans les méthodes agiles 131
5.5.1 Le management de l'équipe de développeurs 131
5.5.2 L'implication des utilisateurs 132
5.5.3 Les techniques de travail en session participative 133
5.5.4 Les rôles dans les méthodes agiles 134

Chapitre 6 – Le management des risques 139

6.1 Les risques dans les projets système d'information 139
6.1.1 L'importance des risques dans les projets système d'information 139
6.1.2 La définition du risque 140
6.1.3 Le management des risques 141

6.2 L'analyse des risques 142

6.2.1 *Les différentes approches d'analyse des risques* . . . 142
6.2.2 *L'approche généralisée* . . . 142
6.2.3 *L'approche par les types de risques recensés* . . . 143
6.2.4 *L'approche par le profil de risque* . . . 147

6.3 Le plan de management du projet . . . 150

6.4 Le contrôle des risques . . . 151
6.4.1 *Les stratégies génériques de réponse aux risques* . . . 151
6.4.2 *La stratégie de projet* . . . 152
6.4.3 *L'audit en cours de projet* . . . 155

6.5 Les méthodes agiles et la gestion des risques . . . 157
6.5.1 *Méthodes agiles et profil de risque* . . . 157
6.5.2 *Méthodes agiles et facteurs de succès* . . . 159
6.5.3 *Les conditions d'utilisation d'une méthode agile* . . . 159

Chapitre 7 – Le pilotage du projet . . . 163

7.1 Le concept de pilotage . . . 163

7.2 Le tableau de bord du chef de projet . . . 166
7.2.1 *L'objectif du tableau de bord* . . . 166
7.2.2 *Le suivi individuel* . . . 167
7.2.3 *Le suivi du projet* . . . 171

7.3 Le suivi économique : la méthode de la valeur acquise . . . 173

7.4 Les décisions de pilotage . . . 175
7.4.1 *L'arbre de décision et la valeur monétaire attendue (VMA)* . . . 176
7.4.2 *Le calcul de rentabilité : le ROI* . . . 177

7.5 Le pilotage d'un projet sous-traité . . . 179
7.5.1 *La sous-traitance d'un projet* . . . 179
7.5.2 *Le rôle du chef de projet en cas de sous-traitance* . . . 180

7.6 Le management des connaissances pour les projets . . . 181
7.6.1 *La capitalisation d'expérience sur les projets* . . . 181
7.6.2 *Le knowledge management appliqué aux projets système d'information* 182

7.7 Le pilotage d'un projet en mode agile . . . 186
7.7.1 *Le suivi d'un projet agile* . . . 186
7.7.2 *Les variables d'action pour un projet agile* . . . 187
7.7.3 *La gestion des connaissances dans un projet agile* . . . 188

Chapitre 8 – La maîtrise de la qualité 189

8.1 La problématique de la qualité 189

8.2 Le vocabulaire de la qualité 193

8.3 La normalisation de la qualité : normes AFNOR et ISO 195

8.3.1 La typologie des normes 195

8.3.2 Les normes AFNOR 196

8.3.3 La norme ISO9000 197

8.4 La qualité des systèmes d'information 198

8.5 Les facteurs et les indicateurs de la qualité d'un système d'information . 200

8.5.1 Les facteurs qualité d'un système d'information 200

8.5.2 Les critères qualité d'un système d'information 204

8.5.3 L'utilisation des caractéristiques de la qualité 208

8.6 Le manuel qualité 209

8.7 Le plan qualité 210

8.7.1 La définition du plan qualité 210

8.7.2 Le contenu du plan qualité 211

8.8 Le contrôle et l'audit qualité 212

8.8.1 Le contrôle qualité 212

8.8.2 L'audit qualité 215

8.9 La qualification des entreprises 215

8.9.1 La certification 215

8.9.2 L'évaluation de la maturité des entreprises 217

8.10 La qualité dans les méthodes agiles 220

8.10.1 La qualité du produit dans les méthodes agiles 220

8.10.2 La qualité du processus dans les méthodes agiles 220

Deuxième partie – Mise en œuvre, exercices et études de cas

Chapitre 9 – La maîtrise des risques 223

9.1 Le cas Mécano 223

9.1.1 Description de l'entreprise Mécano 223

9.1.2 Description du projet Mécano 224

9.1.3 Analyse du projet Mécano 224

9.1.4 Profil de risque du projet Mécano 225
9.1.5 Détermination de la stratégie de projet du projet Mécano 226

9.2 Le cas Kalizeau 228
9.2.1 Présentation du projet Kalizeau 228
9.2.2 Énoncé de l'analyse des parties prenantes du cas Kalizeau 229
9.2.3 Corrigé de l'analyse des parties prenantes du cas Kalizeau 230
9.2.4 Énoncé de la structure de découpage de projet Kalizeau 230
9.2.5 Corrigé de la structure de découpage du projet Kalizeau 232
9.2.6 Énoncé de l'analyse de risque du projet Kalizeau 233
9.2.7 Corrigé de l'analyse de risque du projet Kalizeau 234
9.2.8 Énoncé des réponses aux risques du projet Kalizeau 238
9.2.9 Corrigé des réponses aux risques du projet Kalizeau 239

Chapitre 10 – La pratique de l'estimation des charges 241

10.1 Le cas Parking 241
10.1.1 Description du projet Parking 241
10.1.2 Bases d'expériences des experts 242
10.1.3 Éléments pour l'évaluation analytique 245

10.2 Corrigé du cas Parking avec la méthode Delphi 246

10.3 Corrigé du cas Parking avec la méthode de répartition proportionnelle . 247

10.4 Corrigé du cas Parking avec le modèle Cocomo 248

10.5 Corrigé du cas Parking avec la méthode d'évaluation analytique 249

10.6 Corrigé du cas Parking avec la méthode des points fonctionnels 250
10.6.1 Dénombrement des groupes de données référencées 251
10.6.2 Dénombrement des entrées 254
10.6.3 Dénombrement des sorties 254
10.6.4 Dénombrement des interrogations 255
10.6.5 Estimation de la charge 256

10.7 Énoncé de la démarche générale d'estimation 256

10.8 Corrigé de la démarche générale d'estimation 257

10.9 Comparaison des différentes méthodes 258

Chapitre 11 – L'application des techniques de planification des délais 261

11.1 Exercice Conférence 261

11.2 Corrigé de l'exercice Conférence 263
11.3 Exercice Petit déjeuner 264
11.4 Corrigé de l'exercice Petit déjeuner 265
11.5 Exercice Réveil 266
11.6 Corrigé de l'exercice Réveil 266
11.7 Exercice Cursus scolaire 271
11.8 Corrigé de l'exercice Cursus scolaire 272
11.9 Exercice Échéancier 274
11.10 Corrigé de l'exercice Échéancier 275
11.10.1 Graphe des antécédents et paramètres clés 275
11.10.2 Graphe avec date imposée 275
11.10.3 Calendrier du projet 278
11.10.4 Planification avec contraintes sur les ressources 280
11.11 Exercice Recette aléatoire 282
11.12 Corrigé de l'exercice Recette aléatoire 282

Chapitre 12 – Le système d'information de pilotage 285

12.1 Le cas Parking 285
12.2 Corrigé de l'Organisation du projet Parking 287
12.2.1 Graphe des antécédents du projet Parking 287
12.2.2 Affectation des tâches du projet Parking 289
12.2.3 Établissement du planning 291
12.3 Énoncé du récapitulatif mensuel du projet Parking 294
12.4 Corrigé du récapitulatif mensuel du projet Parking 296
12.5 Énoncé du bilan individuel mensuel du projet Parking 297
12.6 Corrigé du bilan individuel mensuel du projet Parking 299
12.7 Énoncé de la gestion des aléas du projet Parking 300
12.8 Corrigé de la gestion des aléas du projet Parking 301
12.9 Énoncé de l'avancement du projet Parking 306
12.10 Corrigé de l'avancement du projet Parking 308
12.11 Énoncé du bilan du projet Parking 310

12.12 Corrigé du bilan du projet Parking 311
12.13 Énoncé de la capitalisation d'expérience du projet Parking 312
12.14 Corrigé de la capitalisation d'expérience du projet Parking 313
12.15 La méthode de la valeur acquise 315
12.15.1 Énoncé de l'exercice sur la valeur acquise 315
12.15.2 Corrigé de l'exercice sur la valeur acquise 317
12.16 La valeur monétaire attendue 319
12.16.1 Énoncé de la VMA 319
12.16.2 Corrigé de la VMA 319
12.17 Le calcul de rentabilité 320
12.17.1 Énoncé des indicateurs du ROI 320
12.17.2 Corrigé des indicateurs du ROI 322

Chapitre 13 – La dimension relationnelle des projets 323

13.1 Le management d'équipe 323
13.1.1 Énoncé de l'affectation de nouvelles ressources 323
13.1.2 Corrigé de l'affectation de nouvelles ressources 324
13.1.3 Énoncé d'un problème avec un groupe de validation 324
13.1.4 Corrigé d'un problème avec un groupe de validation 325
13.1.5 Énoncé d'un retard sur un sous-projet 326
13.1.6 Corrigé d'un retard sur un sous-projet 326
13.1.7 Énoncé d'un retrait de ressources 327
13.1.8 Corrigé d'un retrait de ressources 327
13.1.9 Énoncé de la mise en place d'un intranet pour le projet 328
13.1.10 Corrigé de la mise en place d'un intranet pour le projet 328
13.2 Les stratégies de résolution des conflits 329
13.2.1 Énoncé d'un conflit avec le maître d'ouvrage 329
13.2.2 Corrigé d'un conflit avec le maître d'ouvrage 329
13.2.3 Énoncé d'un conflit avec le responsable qualité 330
13.2.4 Corrigé d'un conflit avec le responsable qualité 330
13.2.5 Énoncé d'un conflit avec un autre chef de projet 331
13.2.6 Corrigé d'un conflit avec un autre chef de projet 331
13.2.7 Énoncé d'un conflit avec le responsable technique 332
13.2.8 Corrigé d'un conflit avec le responsable technique 332
13.2.9 Énoncé d'un conflit avec le chef de projet utilisateur 332

13.2.10 Corrigé d'un conflit avec le chef de projet utilisateur 333

Chapitre 14 – Le plan qualité d'un projet 335

14.1 Le cas Mécano 335

14.2 Corrigé du plan qualité du cas Mécano 336

14.2.1 Première partie : la qualité du produit 336

14.2.2 Deuxième partie : la qualité du processus 338

Chapitre 15 – Un exemple d'utilisation de Microsoft Project 341

15.1 Le cas Parking 341

15.2 Initialisation du projet Parking 341

15.2.1 Création du projet Parking 341

15.2.2 Choix des options du projet Parking 342

15.2.3 Définition du calendrier du projet Parking 345

15.3 Planification du projet Parking 345

15.3.1 Saisie des tâches du projet Parking 345

15.3.2 Saisie des liens du projet Parking 347

15.3.3 Saisie et affectation des ressources du projet Parking 350

15.3.4 Mémorisation du planifié pour le projet Parking 353

15.4 Pilotage du projet Parking 355

15.4.1 Saisie de l'avancement du projet Parking à la fin janvier 355

15.4.2 Récapitulatif de l'avancement du projet Parking à la fin janvier 359

15.4.3 Saisie de l'avancement du projet Parking à la fin février 361

15.4.4 Bilan individuel du projet Parking à la fin février 365

15.4.5 Gestion des aléas du projet Parking 367

15.4.6 Tableau d'avancement du projet Parking à la fin mars 369

15.4.7 Présentation par lots 370

15.5 Bilan du projet Parking 373

15.5.1 Tableau d'avancement final du projet Parking 373

15.5.2 Tableau de bilan du projet Parking 375

Troisième partie – Les référentiels et la certification en management de projet

Chapitre 16 – Associations, normes et cadres de référence 381

16.1 Les associations professionnelles 381

16.2 La norme ISO10006 382

16.3 Le cadre Eurométhode 386

16.3.1 Présentation d'Eurométhode 386
16.3.2 Le plan des livraisons 387
16.3.3 Les fournitures relatives au domaine cible 388
16.3.4 Les fournitures du domaine projet 388

16.4 Le cadre PRINCE2 389

Chapitre 17 – La préparation de la certification PMI 391

17.1 La certification du PMI 391

17.1.1 Le choix du PMI 391
17.1.2 L'apport à la préparation de la certification 392

17.2 Le cadre de la certification du PMI 392

17.2.1 Le référentiel 392
17.2.2 L'examen de certification 392

17.3 La logique générale du PMBOK 393

17.3.1 La structure du PMBOK et les domaines de connaissance 393
17.3.2 Produit et projet 394
17.3.3 Les phases et le cycle de vie du projet 395
17.3.4 Les processus de management de projet et les groupes de processus . . . 396

17.4 Le management de l'intégration du projet 401

17.4.1 Les processus du management de l'intégration 401
17.4.2 Les techniques utiles pour le management de l'intégration 402

17.5 Le management du contenu du projet 402

17.5.1 Les processus du management du contenu du projet 402
17.5.2 Les techniques utiles pour le management du contenu du projet 403

17.6 Le management des délais du projet 403

17.6.1 Les processus du management des délais du projet 403
17.6.2 Les techniques utiles pour le management des délais du projet 404

17.7 Le management des coûts du projet 406
17.7.1 Les processus du management des coûts du projet 406
17.7.2 Les techniques utiles pour le management des coûts du projet 407
17.8 Le management de la qualité du projet 407
17.8.1 Les processus du management de la qualité du projet 407
17.8.2 Les techniques utiles pour le management de la qualité du projet 408
17.9 Le management des ressources humaines du projet 408
17.9.1 Les processus du management des ressources humaines du projet 408
17.9.2 Les techniques utiles pour le management des ressources humaines du projet 409
17.10 Le management des communications du projet 410
17.10.1 Les processus du management des communications du projet 410
17.10.2 Les techniques utiles pour le management des communications du projet 410
17.11 Le management des risques du projet 411
17.11.1 Les processus du management des risques du projet 411
17.11.2 Les techniques utiles pour le management des risques du projet 412
17.12 Le management des approvisionnements du projet 414
17.12.1 Les processus du management des approvisionnements du projet 414
17.12.2 Les techniques utiles pour le management des approvisionnements du projet 414

Chapitre 18 – Méthodes agiles et référentiel PMBOK 417
18.1 Méthodes agiles et logique du PMBOK 417
18.2 Méthodes agiles et management du contenu du projet 418
18.3 Méthodes agiles et management des délais du projet 418
18.4 Méthodes agiles et management des coûts du projet 419
18.5 Méthodes agiles et management de la qualité du projet 419
18.6 Méthodes agiles et management des ressources humaines du projet . . . 420
18.7 Méthodes agiles et management des communications du projet 421
18.8 Méthodes agiles et management des risques du projet 421
18.9 Méthodes agiles et management des approvisionnements du projet . . . 421
18.10 Méthodes agiles et management de l'intégration du projet 422

Conclusion 425

Annexe A – Les outils de la gestion de projet 427

A.1 Outils d'estimation des charges 427

A.2 Outils de planification et de suivi 428

Annexe B – Les facteurs correcteurs de la méthode des points fonctionnels . 429

B.1 Degré d'influence de la communication des données 429

B.2 Degré d'influence de la distribution des données ou des traitements . . . 430

B.3 Degré d'influence de la performance 430

B.4 Degré d'influence de l'intensité d'utilisation de la configuration matérielle 431

B.5 Degré d'influence du taux de transaction 431

B.6 Degré d'influence de la saisie interactive 431

B.7 Degré d'influence de la convivialité 432

B.8 Degré d'influence de la mise à jour en temps réel des GDI 432

B.9 Degré d'influence de la complexité des traitements 432

B.10 Degré d'influence de la réutilisation 433

B.11 Degré d'influence de la facilité d'installation 433

B.12 Degré d'influence de la facilité d'exploitation 434

B.13 Degré d'influence de la portabilité 434

B.14 Degré d'influence de la facilité d'adaptation 434

Annexe C – Grille d'analyse des risques 437

Annexe D – Exemples de questions pour la certification 443

Bibliographie 449

Gestion de projet (français) 449
Gestion de projet (anglais) 450
Sites internet 450
Articles 450

Index 453

Introduction

Conduire un projet de conception et de développement d'un système d'information est une opération complexe. Si tout projet comporte une part d'incertitude, la nature d'un système d'information en accroît les risques. En grande partie immatériel, un système d'information met en jeu des acteurs multiples et entre en interaction avec l'organisation de l'entreprise. L'analyse des difficultés rencontrées dans la pratique montre le besoin d'un management de projet solide. La maîtrise des délais, des coûts et de la qualité passe par la mise en œuvre de techniques, de principes et de dispositifs spécifiques à ce type de projet.

Le but du présent ouvrage, qui prend en compte les normes récentes, est d'offrir une vue complète des différents aspects du management de projet centrée sur les projets système d'information. Il fournit un support pour analyser les situations et décider des options à prendre. Il s'adresse aussi bien à des étudiants qu'à des praticiens. Le point de vue privilégié est celui du chef de projet.

Dépassant le cadre d'une méthode de conception (axée sur le « comment faire ? ») ou de conduite (centrée sur un découpage particulier en étapes), ce livre offre un ensemble d'instruments utilisables quelle que soit la méthode retenue. Il a été conçu avec un objectif opérationnel et inclut de nombreux exemples et exercices résolus. Il couvre le cycle complet, de l'évaluation des risques à la capitalisation d'expérience, et il intègre les préoccupations concernant la dimension humaine d'un projet, la gestion du changement et la qualité. La lecture de cet ouvrage peut être faite de deux manières :

- soit en déroulant les différents aspects, qui sont présentés dans l'ordre où, généralement, ils apparaissent dans un projet ;
- soit en se reportant directement à l'un ou l'autre chapitre selon la technique recherchée.

La première partie décrit les principes et techniques, et fournit un éclairage particulier sur les méthodes agiles ; la deuxième partie propose des exercices et études de cas illustrant la théorie, et présente une utilisation pratique d'un

progiciel de planification et suivi ; la troisième partie est consacrée aux référentiels et à la certification PMI.

PREMIÈRE PARTIE

PRINCIPES ET TECHNIQUES

Après une introduction au management des projets système d'information qui se termine par une rapide présentation des méthodes agiles, l'ouvrage aborde les principes et les modèles de découpage. Les techniques d'estimation des charges sont présentées au chapitre 3. La planification et la prise en compte de la dimension humaine d'un projet font l'objet des deux chapitres suivants. Le chapitre 6 traite de l'analyse des risques et de leur contrôle. Le chapitre 7 décrit les concepts du pilotage et les techniques pour suivre et maîtriser un projet. La première partie s'achève sur un panorama des différents aspects de la problématique qualité.

1

Problématique du management de projet

1.1 ORIGINE ET ÉVOLUTION DU MANAGEMENT DE PROJET

C'est au début des années 1950 que sont nées les premières réflexions publiques sur la conduite d'un projet, principalement dans les pays anglo-saxons. Elles sont liées aux grands projets engagés dans divers domaines industriels : aéronautique, travaux publics, armement. L'objectif était de développer des techniques et des méthodes pour augmenter la maîtrise des travaux et la coordination des différents corps de métier. Historiquement, ce développement s'est inscrit dans le courant de la recherche opérationnelle, qui visait une formalisation mathématique des problèmes de gestion pour prendre les décisions optimales. Il a correspondu également au grand mouvement de planification mis en œuvre dans la plupart des pays développés et qui a influencé certaines théories managériales.

Le corpus méthodologique sur la conduite de projet s'est ensuite constitué en grande partie par la pratique. Les sociétés de services et de conseil, ainsi que de nombreux consultants indépendants, ont développé une offre variée de prestations qui comprennent la direction d'un projet ; l'assistance au chef projet sur certains aspects, tels que la planification et la surveillance des délais ou des coûts, la gestion de la qualité du projet, la gestion administrative du projet… ; le conseil ponctuel pour certaines tâches telles que le découpage et l'organisation du projet ou l'estimation des charges.

Depuis une trentaine d'années, des associations professionnelles ont œuvré pour faire reconnaître le rôle et les compétences particulières des chefs de projet.

Leurs actions ont conduit à une diffusion et une reconnaissance des certifications en management de projet, qui valident l'acquisition de savoirs et savoir-faire spécifiques. Citons notamment l'AFITEP (Association francophone de management de projet), l'IPMA (*International Project Management Association*) d'origine européenne et le PMI (*Project Management Institute*) d'origine nord-américaine. En Grande-Bretagne, les pouvoirs publics ont joué un rôle majeur dans la définition d'une méthode, PRINCE2, qui donne également lieu à une certification. Une brève présentation en est donnée au chapitre 16. La troisième partie de cet ouvrage fournit des éléments complémentaires sur différents référentiels et sur les certifications.

En ce qui concerne les techniques, la planification a occupé pendant longtemps une place centrale dans la conduite d'un projet. Certaines fonctions spécifiques de planificateur ont existé jusqu'à récemment. On a aujourd'hui une vision plus large du rôle du chef de projet. Par ailleurs, l'origine industrielle a fortement marqué le découpage d'un projet ; dans le domaine des systèmes d'information, il a fallu des années pour que d'autres approches apparaissent.

Nous allons maintenant voir comment est défini un projet et quelles sont les activités qui relèvent du management de projet.

1.2 DÉFINITIONS D'UN PROJET

1.2.1 L'approche générale

La langue française a attaché au mot projet des degrés de précision divers. Le terme représente d'abord une intention, souvent floue, dont la réalisation peut être lointaine ; c'est par exemple le cas lorsque l'on évoque un projet de voyage. Une seconde définition le décrit comme une étude préparatoire, parfois exhaustive, qui va être soumise à décision : on parle ainsi d'un projet de loi ou d'un projet d'urbanisme.

Quelle que soit l'acception retenue, le projet précède une réalisation ou un état définitif. C'est une image plus ou moins précise d'un futur que l'on pense atteindre.

Dans cet ouvrage, nous utilisons le mot projet dans un sens orienté vers le management de projet. Les gestionnaires ont ramené ce vocable dans le présent en le reliant à des actions à accomplir.

Un projet est défini comme un ensemble d'activités à effectuer pour atteindre un but défini de façon spécifique. De façon plus précise, on parle de « travail en mode projet » lorsque l'on doit atteindre un *objectif* avec des *moyens ad hoc* et dans un *délai* donné. Le mode projet requiert une organisation et un management adaptés. On le représente parfois sous forme d'un triangle (figure 1.1), ce

qui exprime la contrainte de solidarité entre les sommets : si l'un des sommets bouge et que l'on veut conserver le même triangle, il faut agir sur l'un ou les deux autres sommets. Ainsi, toute évolution du périmètre du projet aura des conséquences soit sur le délai, soit sur les ressources à mettre en œuvre. Un aléa modifiant la disponibilité des ressources se répercutera soit sur le délai, soit sur l'objectif visé.

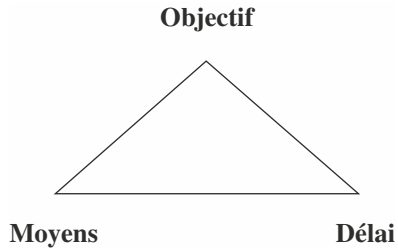


Figure 1.1 — Le triangle Projet.

Le mode Projet se distingue de celui d'une activité répétitive ou d'une mission permanente.

Une activité répétitive est la réponse à une occurrence d'événement déclencheur préalablement identifié dans l'entreprise. On peut définir une procédure stable pour décrire les activités à accomplir. Ainsi un acheteur dans un service Approvisionnement déclenche la procédure *Commande* chaque fois qu'un besoin d'achat se manifeste. À l'inverse, le lancement d'un projet relève d'une décision. Tout projet est unique et ne peut être traité par un dispositif standard. Il nécessite une prise en compte de ses caractéristiques propres.

De son côté, une mission permanente est définie par un but, mais sans mention de délai ; elle subsiste jusqu'à une décision de réorganisation. Ainsi, une mission Qualité dans l'entreprise doit mettre en œuvre un ensemble d'activités pour surveiller et augmenter la qualité : définir des critères qualité, identifier les mesures à effectuer, traiter les écarts… Le responsable de la mission ne peut en prononcer la suppression. Tout projet est, au contraire, temporaire : par nature, il est destiné à s'achever à un horizon visible. Les ressources sont affectées pour une durée limitée. C'est aux responsables du projet qu'il revient de déclarer celui-ci terminé.

Le mode Projet introduit du mouvement dans un dispositif organisationnel stable. Cela provient de ce que l'on entreprend quelque chose de nouveau. L'effet est renforcé par les affectations temporaires des acteurs. Cette instabilité souvent dynamisante est parfois recherchée pour des activités traditionnellement effectuées dans de missions globales. C'est ce que l'on appelle « la gestion par projet ». Des activités diverses sont ainsi menées comme des projets, dans le

sens où l'on formalise précisément l'objectif à atteindre et on y affecte des moyens provisoires. Par exemple, un déménagement, l'élaboration d'un document commercial ou la réorganisation du réseau de distribution peuvent être gérés comme des projets.

1.2.2 Les définitions normalisées

Nous avons retenu trois des principales définitions normalisées : celle de l'ISO, organisation internationale de normalisation, celle du PMI et celle conjointe de l'AFITEP et l'AFNOR, association française de normalisation.

Selon ISO10006 : 2003[1], un projet est un « processus unique, qui consiste en un ensemble d'activités coordonnées et maîtrisées comportant des dates de début et de fin, entrepris dans le but d'atteindre un objectif conforme à des exigences spécifiques telles que les contraintes de délais, de coûts et de ressources ».

L'unicité du processus projet doit être comprise de deux façons. D'une part, la séquence des activités qui permettront d'atteindre l'objectif doit être définie de façon adaptée à chaque projet, même si l'on réutilise des trames générales. D'autre part, les activités qui composent le projet seront exécutées une seule fois. Il y a donc unicité au niveau du type et au niveau de l'instance.

Le référentiel du PMI, appelé Guide du PMBOK (*Project Management Body of Knowledge*), donne d'un projet la définition suivante :

> « entreprise temporaire décidée pour obtenir un produit ou un service unique ».

L'unicité du produit entraînera celle des activités à mettre en œuvre.

L'AFITEP et l'AFNOR définissent un projet comme un « ensemble d'actions à réaliser pour satisfaire un objectif défini, dans le cadre d'une mission précise, et pour la réalisation desquelles on a identifié non seulement un début, mais aussi une fin » [AFITEP, 2000].

Une distinction est introduite entre les *projets d'ingénierie* qui visent l'obtention d'un résultat pour un client, et les *projets produit* débouchant sur un modèle qui fera ensuite l'objet d'une fabrication répétitive. Les projets de système d'information relèvent exclusivement de la première catégorie.

Chacune de ces trois définitions met l'accent sur des activités finalisées et soumises à contraintes, nous y retrouvons les trois éléments du triangle Projet : objectif, moyens, délai.

---

1. Le contenu de cette norme, intitulée « Systèmes de management de la qualité. Lignes directrices pour le management de la qualité dans les projets » est exposé au chapitre 16.

1.3 QU'EST-CE QUE LE MANAGEMENT DE PROJET ?

1.3.1 L'approche générale du management de projet

Le management de projet a pour but de mener un projet à son terme en organisant et en surveillant son déroulement.

Le champ du management de projet est calé sur les caractéristiques génériques d'un projet. Les trois aspects représentés par le triangle Projet doivent être mis sous contrôle. Chacun fait l'objet d'un management spécifique, qui prend en compte l'existence des deux autres ; chaque sommet du triangle Projet en génère un autre, le tout formant un nouveau triangle, celui du management de projet. Ainsi :

- Le délai donne lieu à un *management du temps* dont le rôle est de définir le parcours et de le jalonner, d'établir des calendriers et de maîtriser la consommation de l'enveloppe temps.
- Les moyens affectés constituent le budget du projet, qui est transformé en travail, locaux, matériel, temps machine, déplacement… Cette transformation nécessite un *management des ressources* portant sur les ressources humaines et les moyens matériels. Dans les projets système d'information, les ressources humaines occupent une place primordiale. Utiliser chacun au mieux, constituer des équipes efficaces, affecter les personnes au moment adéquat en fonction de leurs compétences, coordonner les travaux, limiter le nombre d'acteurs sans pour autant exclure… telles sont les attributions de cette fonction.
- L'objectif du projet doit à son terme être concrétisé par une ou plusieurs fournitures. Ce sommet donne naissance au *management de la production*, qui a pour but de suivre et diriger l'avancement vers l'objectif tout au long du projet. On parle parfois de « faire converger le projet » : cela signifie qu'il faut sans cesse s'assurer que l'on se rapproche du but et que l'on ne part pas dans des directions remettant en cause un avancement consolidé.

Ces trois aspects sont représentés par le triangle du management de projet donné à la figure 1.2.

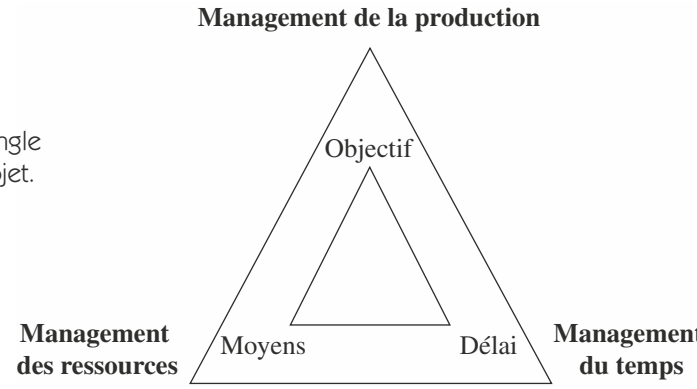


Figure 1.2 — Le triangle Management de projet.

La solidarité des sommets est permanente.

On veut parfois marquer une priorité pour le paramètre temporel : on parle alors d'un « management par les délais ». Il faut toutefois se garder de l'isoler des deux autres, sous peine de conduire le projet à l'échec.

De façon analogue, manager des ressources et une production en dehors de toute contrainte temporelle est une erreur : la prise en compte d'échéances stimule la production et facilite la gestion des ressources.

Ces trois préoccupations sont toujours présentes dans les différentes tâches de management de projet, dont nous allons maintenant examiner le contenu et l'articulation.

En se référant à la description classique d'une fonction managériale, on peut décomposer l'activité de management de projet en trois activités principales autour de la production proprement dite (figure 1.3).

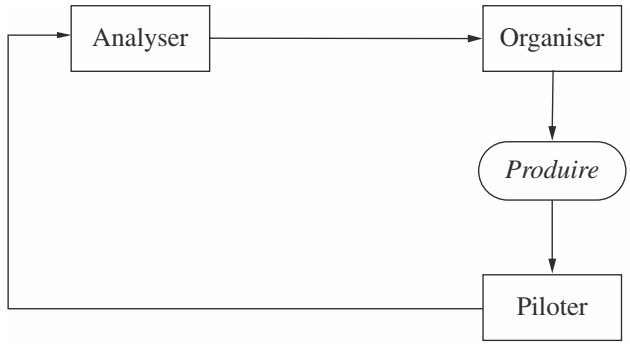


Figure 1.3 — Les activités du management de projet.

- *Analyser* : il s'agit de déterminer le chemin que l'on va emprunter pour avancer vers l'objectif. Pour cela, on étudie les caractéristiques du projet, son contexte, les risques qui le menacent et l'état de son avancement. Cela conduit à un découpage du projet en activités à entreprendre et à une estimation de l'effort nécessaire. La maille de cette analyse varie selon le moment du projet auquel on se place.
- *Organiser* : on repère les contraintes d'enchaînement entre les tâches afin de les ordonnancer. Cela permet d'établir un calendrier. L'organisation recouvre aussi la constitution d'une équipe, c'est-à-dire des personnes qui sont affectées et imputées au projet, en déterminant les bons profils. Les relations avec tous les partenaires nécessaires sont également prises en compte. Dès que la charge est importante, on répartit le travail entre plusieurs personnes, voire plusieurs équipes, ce qui conduit à mettre en place des moyens de partage d'informations pour éviter les incohérences.

- *Piloter* : cette activité comprend le suivi de l'avancement du projet, en quantité et en qualité, ainsi que l'analyse et le traitement des écarts avec ce qui était prévu, les orientations et les décisions à prendre ou à faire prendre. Le pilotage inclut également le management de l'équipe et la gestion des conflits.

Nous n'avons pas isolé une activité spécifique concernant la qualité. En effet, nous considérons qu'elle intervient transversalement dans toute la gestion de projet et qu'elle doit être intégrée aux trois activités principales : analyser, organiser, piloter. Pour des raisons de clarté, nous avons cependant dans cet ouvrage rassemblé ces préoccupations dans un chapitre à part.

La présentation des activités de management de projet donne parfois l'impression d'une répartition inégale dans le temps, la fonction s'exerçant principalement en début de projet et la production prenant ensuite le relais. Cette vision est tout à fait fausse.

Un projet est managé du début à la fin. C'est ce que représente la boucle de rétroaction de la figure 1.3, qui doit être lue de façon dynamique : selon les aléas et le stade d'avancement, on accentue plutôt l'une ou l'autre activité, en adaptant le contenu.

Par exemple, en début de projet, l'activité Analyser comporte une analyse de risque qui permet notamment de choisir un type d'approche : selon le degré de visibilité de l'objectif initial, on choisit soit de faire une étude préalable, soit d'élaborer directement des spécifications, soit de construire un prototype.

Un aléa, interne ou externe, au cours du déroulement du projet peut conduire à une replanification, une nouvelle répartition des tâches, ou même une modification de l'objectif. Prenons l'exemple d'un projet de planification de la production dans une entreprise fabriquant des équipements d'automobile. Supposons que l'entreprise soit rachetée par un constructeur, il faut alors intégrer les prévisions de fabrication du constructeur lui-même, ce qui représente une fonction supplémentaire.

Par ailleurs, si le projet est important, l'équipe constituée au début du projet se renforce et se structure différemment selon les étapes d'avancement.

Les principes et techniques liés aux trois grandes activités du management de projet sont traités dans les différents chapitres.

L'activité Analyser est traitée dans le chapitre 2 (découpage d'un projet), le chapitre 3 (estimation des charges) et le chapitre 6 (analyse des risques). L'aspect qualité de cette tâche figure au paragraphe du chapitre 8 (maîtrise de la qualité) traitant des facteurs et indicateurs.

L'activité Organiser est décrite dans les chapitres 4 (techniques de planification) et 5 (organisation du travail). La qualité spécifique à l'organisation du projet est traitée dans le paragraphe consacré au plan qualité.

L'activité Piloter fait l'objet du chapitre 7 (pilotage du projet), du paragraphe traitant du contrôle des risques au chapitre 6, du paragraphe concernant le rôle du chef de projet au chapitre 5 et du paragraphe sur le contrôle qualité au chapitre 8.

1.3.2 La définition normalisée du management de projet

Le référentiel de l'IPMA [IPMA, 1999] donne une définition générale de l'ensemble des activités de management de projet : « Le management de projet consiste à planifier, organiser, suivre et maîtriser tous les aspects d'un projet, ainsi que la motivation de tous ceux qui sont impliqués dans le projet, de façon à atteindre les objectifs de façon sûre et dans les critères définis de coûts, délais et performance. Cela inclut les tâches de direction nécessaires aux performances du projet. »

L'AFITEP et l'AFNOR [AFITEP, 2000] s'attachent à distinguer plusieurs termes de signification proche, afin d'identifier plusieurs rôles dans le management d'un projet.

Le management est défini comme « *l'ensemble des tâches permettant de conduire une opération quelconque à bonne fin* ». Associé à projet, ce terme est décrit par les tâches qu'il recouvre : le **management de projet** comprend « *les tâches de direction, gestion, maîtrise, pilotage* ». Ces tâches peuvent être assurées « *par une même personne ou plusieurs, appartenant à une même entreprise ou à plusieurs entités, parties prenantes du projet* ».

Le management de projet est donc de la responsabilité du chef de projet, même s'il s'appuie parfois sur un ou plusieurs assistant pour certaines activités. Le management de projet se compose donc de quatre activités, pouvant correspondre à une fonction :

- Direction de projet.
- Gestion de projet.
- Maîtrise.
- Pilotage.

La **direction de projet**, qui est toujours exercée par le chef de projet, a pour mission essentielle de :

1. Fixer les objectifs, la stratégie et les moyens (c'est-à-dire l'itinéraire et l'horaire, les étapes et les ressources qu'on doit y trouver).
2. Coordonner les actions successives et/ou concomitantes.

3. Maîtriser, c'est-à-dire être à tout instant capable, dans tous les domaines, de modifier l'itinéraire et l'horaire (donc les étapes et les ressources) si un objectif évolue, si l'itinéraire (et/ou l'horaire) ne peut être respecté, si une étape doit être grillée, et modifier les étapes suivantes en conséquence.
4. Optimiser la répartition des ressources (en main-d'œuvre, matériel, etc.) en vue d'arriver à une solution optimale, ou de moindre coût, pour l'ensemble du projet.

Pour accomplir la mission de direction de projet, le chef de projet a besoin d'analyser les risques, d'estimer la charge, d'organiser le travail, de le planifier, de le suivre. Pour cela, il doit effectuer, ou faire effectuer, un ensemble de tâches relevant de la **gestion de projet**, c'est-à-dire : « *l'ensemble des tâches de préparation des référentiels, de contrôle de leur respect, d'analyse des causes de déviation, et de reporting.* »

Il est précisé que : « *la gestion de projet a pour objectif essentiel d'apporter à la direction de projet* […] d*es éléments pour prendre en temps voulu toutes les décisions lui permettant de respecter les termes du contrat passé avec le client, en contenu, en qualité, en délai et en coûts (dépenses et recettes)* ». Cette activité comprend également la capitalisation d'expérience.

L'activité de **maîtrise** correspond à des tâches qui ont été extraites de la gestion de projet et sont centrées sur une préoccupation : par exemple, la maîtrise de la qualité ou la maîtrise des coûts. Les décisions relèvent toujours de l'activité de direction de projet.

L'activité de **pilotage**, lorsqu'elle est isolée de la fonction de direction de projet, est limitée à l'activité périodique d'orientation du projet par un Comité de pilotage, dont le rôle est évoqué au chapitre 5.

En conclusion, on peut retenir deux aspects majeurs du management de projet : décider et gérer. Dans cet ouvrage, le terme de pilotage sera utilisé dans le sens cybernétique donné au chapitre 7, c'est-à-dire relevant du chef de projet dans son activité de direction de projet.

1.4 LE MANAGEMENT DES PROJETS SYSTÈME D'INFORMATION

1.4.1 Caractéristiques d'un projet système d'information

Il convient tout d'abord de rappeler la définition d'un système d'information, qui met en lumière son caractère complexe. En effet, dans la mesure où c'est un « ensemble organisé de ressources : matériel, logiciel, personnel, données, procédures… permettant d'acquérir, de traiter, stocker, communiquer des informations (sous formes données, textes, images, sons…) dans des organisations. » [Reix,

2004], il va falloir prendre en compte des dimensions diverses. De plus, son caractère immatériel augmente la difficulté quand on veut en obtenir une description précise. Cela n'est pas sans conséquence sur les projets.

Le triplet {objectif, moyens, délai} présente, dans le domaine système d'information, trois caractéristiques spécifiques, à savoir :

1. *Il y a interaction entre l'objectif d'une part et les moyens/délais d'autre part*. Une première identification de l'objectif conduit à évaluer la charge globale du projet. Cela permet de décider d'une échéance cible théorique et des moyens à affecter. Si d'autres contraintes obligent à limiter le délai ou le budget, on ajuste l'objectif, selon le principe du *design-to-cost* (conception contrainte par le budget disponible). Après décision, on va considérer comme fixes les deux paramètres « moyens » et « délai » initialement alloués et on évaluera l'efficience du projet — composante de son succès — par rapport à ces valeurs.
2. *L'objectif du projet n'est parfaitement défini qu'à l'achèvement du projet*. Un système d'information n'est pas un objet matériel, dont on peut donner une représentation visuelle. Un logiciel est quelque chose d'abstrait. Il est donc décrit par ses fonctions ; cependant, une description exhaustive est longue et coûteuse. Les modèles n'en donnent qu'une vue partielle. La maquette qu'on peut en faire est une analogie, non une miniature. De même un prototype n'est pas, comme en milieu industriel, ce qui précède la série. Cette indétermination est absente des projets industriels qui ont servi de référence à certaines des techniques, notamment le découpage du projet. De plus, les modèles de processus métiers représentant les modifications apportées sont également abstraits et ne rendent pas en compte du vécu des acteurs qui s'exprime progressivement.
3. *Le développement d'un système d'information* ne *se déroule* pas dans un vide organisationnel, mais *dans une organisation*, dont les particularités font partie de la caractérisation du projet lui-même.

   Les comportements des acteurs sont influencés par le système d'organisation dans lequel ils agissent. Celui-ci comprend à la fois la répartition du pouvoir et des ressources, la division des activités, les modes de coordination, les procédures opératoires, les statuts… Les relations personnelles sont régies par un ensemble de normes, fondées sur les valeurs dominantes de l'entreprise, qui contraignent, légitiment ou limitent l'action. Les acteurs ne forment pas un groupe uni vers la réalisation d'un même objectif. Dans les zones d'incertitude se développent les stratégies des groupes ou des individus.

Si le pouvoir est un enjeu dans tout système d'organisation, c'est l'efficacité que l'on invoque le plus souvent lors des choix de conception (rapidité, coût…)

d'un système d'information. La démarche d'élaboration est généralement guidée par une rationalité d'optimisation, faisant abstraction d'autres formes de rationalité. Le processus de développement d'un système d'information est certes un processus *rationnel*, mais qui se double souvent d'un processus *politique* et d'un processus *psychologique*. Leur prise en compte permet d'analyser certaines réactions ou certains conflits, voire de les éviter.

1.4.2 Les objectifs des projets systèmes d'information

Il est parfois utile de distinguer système d'information et système informatique (figure 1.4) :

- Le système d'information est la partie du réel constituée d'informations organisées, d'événements ayant un effet sur ces informations et d'acteurs qui agissent sur ces informations ou à partir de ces informations, selon des processus visant une finalité de gestion et utilisant les technologies de l'information.
- Un système informatique est un ensemble organisé d'objets techniques — matériels, logiciels, applications — dont la mise en œuvre réalise l'infrastructure d'un système d'information.

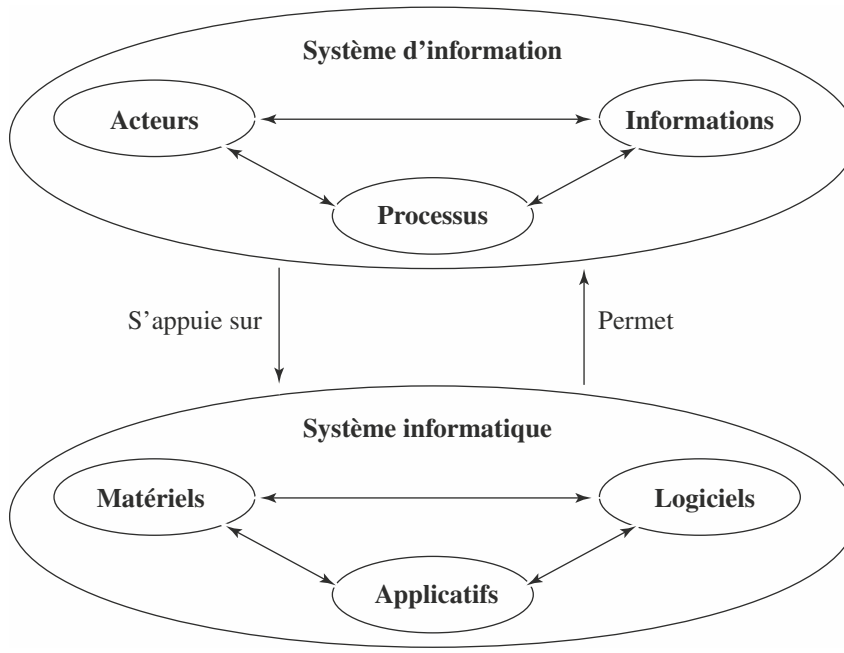


Figure 1.4 — Système d'information et système informatique.

Dans cette perspective, même si un projet système d'information inclut un développement ou un paramétrage de logiciel, les objectifs du projet sont clairement ceux qui sont attachés au système d'information. En effet, c'est l'utilisation que l'on va faire du logiciel — l'aide apportée aux processus et les informations gérées — qui va apporter de la valeur à l'entreprise.

La réflexion sur les objectifs des projets s'inscrit donc aujourd'hui dans la perspective de l'alignement stratégique, selon laquelle la mission du système d'information est d'aider l'entreprise à atteindre ses objectifs. Tout projet système d'information est donc toujours un projet de l'entreprise. Cela implique que les acteurs métiers, que l'on appelle « maîtrise d'ouvrage », décident des évolutions du système d'information. Mais également que les orientations stratégiques soient être traduites en objectifs du système d'information, ce qui fait partie des premières étapes du projet.

Par exemple, diminuer les coûts de gestion peut se traduire par installer un système de workflow ou mettre en place un suivi de la qualité des fournisseurs. L'objectif stratégique d'améliorer l'efficacité des promotions peut conduire à développer un système d'analyse d'effets ou bien un système de suivi des promotions des concurrents (intelligence économique).

Augmenter les ventes peut passer par la mise en place d'un site de vente en ligne ou la conception d'une application d'aide à la vente pour les commerciaux.

Comprendre les objectifs du projet et faire émerger des réponses adéquates est de la responsabilité du chef de projet. On rencontre souvent les grandes catégories suivantes, qui auront des conséquences sur le management du projet. Nous allons les esquisser :

- *Productivité administrative :* la rentabilité du capital investi est recherchée dans la diminution de main d'œuvre grâce à l'automatisation d'une partie des tâches. Le climat social sera tendu et la gestion du changement difficile à mener. La participation des utilisateurs peut conduire à un blocage du projet.
- *Aide au management* : l'objectif majeur du projet est l'amélioration des prises de décision au moyen d'un observatoire au service du management. On va bâtir une mémoire de l'organisation et de ce qui l'entoure, à partir de laquelle on pourra construire des tableaux de bord, faire des analyses, assurer une veille concurrentielle. La conception du système doit être très proche des gestionnaires, faute de quoi le système ne sera guère utilisé.
- *Efficacité opérationnelle* : on attend un meilleur fonctionnement opérationnel par un usage créatif des technologies de l'information et de la communication. L'analyse et la reconstruction des processus sont déterminantes, mais la gestion du changement est un enjeu essentiel.

- *Évolutivité :* on cherche à obtenir un système flexible pouvant être modifié rapidement en cas d'évolution des contraintes et/ou de la stratégie et sachant prendre en compte des adaptations ou des personnalisations non encore identifiées au moment du projet. Cet objectif s'inscrit dans une meilleure maîtrise des investissements informatiques. La compréhension du domaine et de son évolution est importante.
- *Utilisation d'une nouvelle technologie :* le but principal du projet est d'expérimenter une nouvelle technologie, pour voir ce que l'on peut en tirer ou pour obtenir un « effet vitrine » vis-à-vis de l'extérieur. Un délai court est un élément essentiel de la réussite du projet.

1.4.3 La complexité d'un projet système d'information

Un projet système d'information est souvent qualifié de « complexe », dans la mesure où il comporte de nombreux éléments qui sont interdépendants. La complexité des projets a fait l'objet d'un effort de clarification il y a quelques années[1].

Certains distinguent la complexité organisationnelle et la complexité technologique.

La première renvoie notamment à la multiplicité des niveaux hiérarchiques et des entités impliquées, ainsi qu'au degré élevé de la division du travail, induit par la diversité des compétences requises. Dans un projet complexe, les éléments organisationnels ne sont pas indépendants et ils doivent être coordonnés.

La complexité technologique, à une maille plus fine, recouvre la variété des tâches identifiées, ainsi que celle des entrées et des sorties du projet. Si le projet est complexe, les tâches sont interdépendantes, de même que les technologies mises en œuvre et les équipes intervenant dans le projet. Cette forme de complexité augmente lorsque le délai du projet est réduit, car un nombre accru de tâches doivent alors être exécutées en parallèle.

En général, la complexité du projet découle de la complexité du système à réaliser, et plus particulièrement de la structure visée, c'est-à-dire le nombre et la variété de fonctions, et leurs imbrications. C'est pourquoi on appelle parfois cet ensemble — complexité organisationnelle et complexité technologique — la *complexité structurale* du projet. La difficulté générée par cette complexité est liée au fait que tout changement ou option prise dans un sous-système peut avoir non seulement des répercussions sur d'autres sous-systèmes, mais aussi des rétroactions sur d'autres parties du sous-système lui-même. Pour les projets

1. Voir par exemple « The need for new paradigms for complex project », T. M. Williams, *International Journal of Project Management*, 17, 5, pp. 269-273, 1999.

système d'information, la complexité est accrue par le caractère abstrait du logiciel et la difficulté à tester tous les cas de figure pour en éliminer les dysfonctionnements. C'est également le cas lorsque l'on met en place un progiciel intégré avec un haut degré de paramétrage, dont on ne prévoit pas toutes les conséquences organisationnelles.

D'autres auteurs ont ajouté à la complexité structurale une forme de complexité liée à l'*incertitude* du projet. Cette incertitude peut peser sur les buts du projet et sur les méthodes à employer.

Les projets système d'information sont, pour une partie d'entre eux, de bons exemples de projets aux buts incertains. En effet, les exigences des clients/utilisateurs sont difficiles à cerner. Elles sont souvent multiples, voire contradictoires, et surtout elles peuvent varier au cours du temps, entraînant des modifications et retours en arrière, dont les impacts ne sont pas toujours faciles à repérer. D'où une complexité accrue dans le management du projet.

Si les méthodes pertinentes pour le projet ne sont pas perceptibles pour l'équipe de management de projet, par exemple dans le cas de nouveauté technologique, la planification du projet sera difficile et donnera lieu à de nombreux ajustements.

Nous approfondirons l'étude des sources de complexité dans le chapitre 6 consacré à la maîtrise des risques du projet.

Nous allons terminer ce chapitre consacré à la problématique du management de projet en présentant rapidement des approches d'organisation du projet qui se sont progressivement développées depuis une quinzaine d'années pour prendre en compte l'instabilité des besoins et des spécifications des projets système d'information : les méthodes agiles.

1.5 MANAGEMENT DE PROJET ET MÉTHODES AGILES

À partir des années 1970, les grandes entreprises ont commencé à rencontrer de sérieux problèmes de qualité et de productivité dans le développement de leurs applications informatiques. Pour augmenter la rigueur du processus projet et améliorer les relations entre les informaticiens et leurs clients/utilisateurs, différentes méthodes ont été proposées, d'abord dans les années 1980 celles qui ont été appelées « structurées » (SA/SD, SADT, JSD, SSADM, Merise…), puis dans les années 1990 celles dites « orientées objet » (OOD, OMT, OOSE, UML). Elles offraient, en général, un cadre guidant le déroulement du projet et une aide à la représentation du futur système d'information.

Ces méthodes, adoptées à des degrés divers par les entreprises, ont en général permis d'organiser le travail du service informatique en introduisant une culture méthodologique. Cependant, dans le cadre de chaque projet il est nécessaire de passer d'une méthode théorique à une méthode pratique[1], ce qui n'a pas toujours été perçu. La mise en pratique des méthodes a donc souvent donné lieu à un dispositif rigide et contraignant pour les projets, alors que les problèmes de délais et de qualité continuent à se poser. D'où l'émergence d'un courant mettant l'accent sur la souplesse.

Les méthodes dites « agiles » ont officiellement reçu leur nom en 2001 suite à une rencontre entre 17 auteurs de méthodes défendant une organisation des projets de développement logiciel moins structurée et plus légère que les méthodes en vigueur, qui ont cosigné un Manifeste Agile[2]. L'agilité des méthodes fait référence à la capacité qu'elles sont censées donner pour contourner les obstacles et s'adapter aux particularités de chaque projet. L'agilité est largement de nature humaine et organisationnelle.

Donnons quelques repères sur les principales méthodes pouvant être inscrites sous le label « agile » — RAD, DSDM, XP, SCRUM et Crystal — qui ont toutes été élaborées au cours des années 1990.

- RAD

  Cette méthode est la plus ancienne de toutes celles qui peuvent être qualifiées d'agiles. Elle a été proposée par James Martin[3] en 1991 pour promouvoir une utilisation efficace des outils de développement rapide et de prototypage. En effet, la diffusion de tels outils n'avait eu qu'un faible impact sur la longueur des projets. L'idée centrale de RAD, *Rapid Application Development* (développement rapide d'application), est que seule l'introduction des utilisateurs/décideurs dans l'équipe de projet permet à la fois d'améliorer la satisfaction et de raccourcir la durée du processus projet. Alors que les méthodes structurées étaient focalisées sur les règles de découpage et les techniques de modélisation, J. Martin soutient que la dimension organisationnelle des projets est l'élément clé pour le succès d'un projet système d'information. Il propose donc des principes et techniques d'organisation et de pilotage du projet, visant à réduire fortement les délais tout en assurant la qualité du produit.

1. Voir pour plus de détails : C. Morley, « Choisir une méthode de développement de système d'information: problème technique ou problème de management », Colloque Information et Management, AIM, Grenoble, octobre 1991.
2. *Manifesto for Agile Software Development*, http://agilemanifesto.org/
3. James Martin, *Rapid Application Development*, Macmillan Publishing Company, 1991.

- DSDM

  L'apparition de RAD a suscité un vaste mouvement chez les éditeurs d'outils de développement qui se sont assigné une marque « RAD » à des fins commerciales. On a ainsi observé des pratiques de développement, portées par un certain nombre de consultants ou sociétés de conseil en informatique, fort éloignées des principes de la méthode, mais qui revendiquaient le label RAD. Cela a conduit, en 1994, une quinzaine d'entreprises industrielles en Grande-Bretagne, utilisatrices de RAD, à s'associer en consortium pour diffuser un cadre de référence normalisant les pratiques RAD. C'est ainsi qu'est née la méthode DSDM, *Dynamic System Development Method*, avec un premier ouvrage en 1997[1], écrit par la directrice technique du consortium.

- XP

  XP, *eXtreme Programming* est la méthode agile la plus connue, probablement en raison du nombre d'auteurs qui ont contribué à son élaboration. L'expérience initiale était un projet visant à remplacer plusieurs applications de paye dans le groupe Chrysler par une application unique. En 1996, un an après son démarrage, alors qu'il piétinait, ce projet fut placé sous la responsabilité de Ken Beck. Celui-ci a convaincu la Direction informatique de repartir à zéro et, avec l'aide de Ron Jeffries, expérimenta des approches nouvelles d'organisation de l'équipe et de mobilisation des utilisateurs. Ils s'étaient engagés sur un délai de livraison d'une première version un an plus tard, ce qui représentait un défi par rapport aux estimations initiales et aux difficultés rencontrées. Le délai fut globalement respecté, mais l'entreprise Chrysler ayant été rachetée en 1998 par Daimler, le projet fut définitivement arrêté début 2000. Ce demi-succès n'empêcha pas la formalisation de pratiques qui avaient introduit des changements profonds dans le rythme et les modalités de déroulement du projet. De nombreux débats ont eu lieu avec d'autres acteurs méthodologiques, notamment avec Ward Cunningham, inventeur du wiki et interlocuteur de longue date de K. Beck, notamment sur les pratiques de développement. Le nom fut officiellement donné en 1999, lors de la publication des premiers ouvrages[2]. Le terme « extrême » fut choisi pour évoquer une approche radicalement différente, basée sur la systématisation de pratiques considérées comme efficaces, en les poussant jusqu'à leurs limites. Le terme « programming » rappelle le contexte d'application de la méthode, celui des développeurs. La dimension publique des échanges

1. Jennifer Stapleton, *Dynamic System Development Method*, Addison-Wesley, 1997.
2. K. Beck, *Extreme Programming Explained*, Addison-Wesley, 1999 ; R. Jeffries, A. Anderson et C. Hendrickson, *Extreme Programming Installed*, Addison-Wesley, 2000.

autour de XP a contribué à sa diffusion et a conduit d'autres auteurs à se rallier au courant XP.

- SCRUM

  La méthode SCRUM a été élaborée vers 1993 par différents auteurs influencés par les travaux sur l'innovation de H. Takeuchi and I. Nonaka[1]. Ces chercheurs avaient enquêté dans plusieurs entreprises industrielles (Fuji-Xerox, Canon, Honda, NEC, Epson, Brother, 3M, Xerox, and Hewlett-Packard) et ils ont mis en avant que le processus d'innovation ne peut pas être soumis à une planification, compte tenu de nombre d'éléments imprévisibles. Plus précisément, certaines parties du processus peuvent être planifiées, mais d'autres doivent être considérées comme des boites noires, non connaissables a priori. Dans ce cas, le rôle joué par l'équipe de projet, sa motivation, sa capacité d'autonomie et de collaboration sont essentiels pour le succès de projet.

  Ken Schwaber et Jeff Sutherland, les plus actifs dans la formalisation autour de SCRUM, considèrent qu'un projet système d'information comporte également des parties prévisibles et d'autres non, et ils préconisent un mode de management du projet prenant en compte ces deux aspects. Le terme Scrum, utilisé par Takeuchi et Nonaka, est emprunté au vocabulaire du rugby, la « mêlée ». Une mêlée est une phase qui rassemble de façon étroite et ordonnée l'ensemble des joueurs, et qui permet de reprendre le jeu rapidement, en toute sécurité et équitablement, après une faute mineure ou un arrêt de jeu. Ce terme a été adopté comme nom de la méthode d'une part pour mettre l'accent sur la tension entre moments forts de rapprochement et trajectoires plus individuelles, d'autre part pour souligner la nécessité de faire régulièrement des réunions rapides d'équipe visant à continuellement ramener le projet sur une bonne trajectoire. Différents ouvrages sont parus à partir de 2001[2].

- Crystal

  En 1991, IBM a chargé un consultant, Alistair Cockburn, d'élaborer une méthode pour le développement d'application dans un environnement orienté-objet. Après avoir observé de nombreux projets, Cockburn a conclu que la dimension communication était centrale pour la réussite de ces projets[3]. Il a été également sensible à la diversité des situations de projet, et la nécessité d'une adaptation à chaque type de situation. Sous le nom de Crystal, il a donc proposé une famille de méthodes partageant

1. H. Takeuchi and I. Nonaka, « The New New Product Development Game », Harvard Business Review, Jan-Feb 1986.
2. K. Schwaber et M. Beedle, *Agile Software Development with SCRUM*, Prentice Hall, 2001. Voir aussi Scrum : Where Did Rapid Application Development Come From ? IEEE, 2003.
3. A. Cockburn, *Surviving Object-Oriented Projects*, Addison-Wesley Professional, 1998.

certains principes, mais adaptées à la taille de l'équipe et aux enjeux du projet. Ceux-ci sont mesurés par les conséquences en cas d'erreurs dans le logiciel. Chacune porte un qualificatif coloré (Clear, Yellow, Orange, Orange Web, Red, Maroon)[1].

Les méthodes agiles, qui se sont d'abord qualifiées de méthodes « légères », partagent l'objectif de conduire le projet de façon à livrer des applications de qualité dans des délais fortement réduits par rapport aux pratiques observées. Tous les auteurs de ce courant méthodologique sont partis du constat que la dimension humaine est centrale dans les projets système d'information, notamment ceux dont la partie développement de logiciel est importante. Le management de ces projets devrait alors viser à stimuler la créativité des développeurs, qui s'exprime en général dans des activités solitaires, tout en favorisant leur collaboration avec les autres acteurs concernés par le projet (clients, utilisateurs, manageurs…). L'exigence de délai court accroît la difficulté, car l'apprentissage du travail en commun doit être rapide.

Les méthodes agiles ont donné lieu à des controverses méthodologiques entre le camp des « agilistes » et celui des « structuralistes ». Certains proposent depuis quelques années des rapprochements.

Sans nous situer dans ces débats, nous avons adopté une position management de projet, c'est-à-dire que nous considérons d'abord qu'il y a des fondamentaux dans la gestion de tout projet (par exemple, le schéma proposé au § 1.3.1), ensuite que l'unicité de chaque projet appelle des réponses adaptées et enfin que certaines techniques peuvent être ponctuellement utilisées dans des projets très divers. En effet, les recherches sur le management des projets informatiques depuis une vingtaine d'années montrent qu'il n'y a pas de réponse universelle pour tous les projets, mais que seule une approche dite « contingente », prenant notamment en compte les risques et l'incertitude, est pertinente[2]. De plus, une perspective historique fait apparaître des emprunts ou des glissements d'une méthode à l'autre. Certaines techniques de conduite de projet peuvent être utilisées de façon ponctuelle ou accompagnées d'autres éléments méthodologiques. C'est ce qui a permis un rapprochement des cadres de management de projet dans des référentiels à large couverture.

Cependant, une approche contingente et modulaire renvoie sur le responsable du projet de nombreuses décisions : l'objectif de cet ouvrage est d'apporter des éléments qui d'une part facilitent la compréhension de la situation du projet,

1. A. Cockburn, *Crystal Clear : A Human-powered Methodology For Small Teams*, Addison-Wesley Professional, 2004.
2. Voir par exemple, C. Morley, « L'analyse a priori d'un projet système d'information », Colloque AIM, 1999.

d'autre part aide à élaborer les réponses pertinentes. Les principes et les techniques des méthodes agiles rentrent dans cette panoplie d'outils méthodologiques.

C'est pourquoi, nous avons choisi de les présenter, non pas dans un chapitre à part, mais dans les chapitres traitant des différents aspects du management de projet. En particulier :

- les modèles de cycle de vie propres à chacune des principales méthodes agiles sont décrits au chapitre 2, ainsi que les principes de découpage structurel ;
- les approches d'estimation des charges de travail utilisées dans un environnement agile sont indiquées au chapitre 3 ;
- les techniques de maîtrise des délais spécifiques aux méthodes agiles figurent au chapitre 4 ;
- les principes et modalités pour gérer l'équipe de développeurs, pour faire participer les utilisateurs et pour favoriser une communication efficace au cours du projet sont décrits au chapitre 5 ;
- les caractéristiques de gestion des risques des projets menés en mode agile, ainsi que les facteurs de choix d'une méthode agile sont analysés au chapitre 6 ;
- les questions de suivi et de mise sous contrôle d'un projet agile sont abordées au chapitre 7 ;
- la recherche de qualité, objectif majeur des méthodes agiles, donne lieu à des dispositions particulières indiquées au chapitre 8.

De plus, dans la troisième partie de cet ouvrage, un chapitre récapitule les points de convergence et les éventuelles divergences entre le référentiel PMI et les méthodes agiles.

Après cette rapide évocation des courants méthodologiques récents sur la façon de conduire un projet système d'information, nous allons maintenant étudier de façon détaillée les différents aspects de son management.

2

Le découpage d'un projet et les modèles de cycle de vie

2.1 LES PRINCIPES DU DÉCOUPAGE

Nous avons vu au chapitre précédent que le management de projet comporte trois aspects représentés dans le triangle Projet : la production, les ressources et le temps. Une des premières responsabilités du chef de projet est de découper le projet pour pouvoir répartir dans le temps la production et les ressources. Le découpage doit s'appuyer à la fois sur l'approche cartésienne de réduction de la difficulté[1] et sur l'approche systémique de prise en compte des liens entre les éléments[2].

Découper un projet consiste ainsi à identifier des sous-ensembles quasi autonomes, présentant les caractéristiques suivantes :

- chaque sous-ensemble du projet donne lieu à un résultat bien identifié ;
- la charge propre à chacun peut être évaluée ;

1. Le second principe présenté par Descartes dans le *Discours de la méthode* est de « diviser chacune des difficultés (...) en autant de parcelles qu'il se pourrait et qu'il serait requis pour mieux les résoudre ».
2. Jacques Mélèse en présente les principes appliqués à la gestion dans *La gestion par les systèmes*, Édition Hommes et Techniques, 1977.

- les contraintes d'enchaînement entre les sous-ensembles sont repérables : certains sous-ensembles peuvent être réalisés parallèlement, d'autres sont liés entre eux par des contraintes d'antériorité ;
- le découpage est fait à des mailles différentes, un sous-ensemble étant souvent à son tour décomposé.

On utilise deux grands critères pour découper un projet : l'un est temporel, l'autre structurel. Ces deux critères ne sont pas exclusifs.

Le *critère temporel* est utilisé dans la plupart des projets. Il permet de répartir le travail dans le temps : la décomposition fait apparaître une succession d'étapes et de phases. À chacune, on attache une date de début prévue et une date de fin visée. La figure 2.1 en donne une représentation avec le formalisme UML [Morley *et al.*, 2006].

Un projet se compose de phases ; chaque phase comprend un certain nombre d'activités[1] ; une activité est définie par une ou plusieurs tâches à effectuer. À chaque élément de décomposition on attache un résultat à atteindre, appelé livrable, qui peut faire l'objet d'un engagement contractuel.

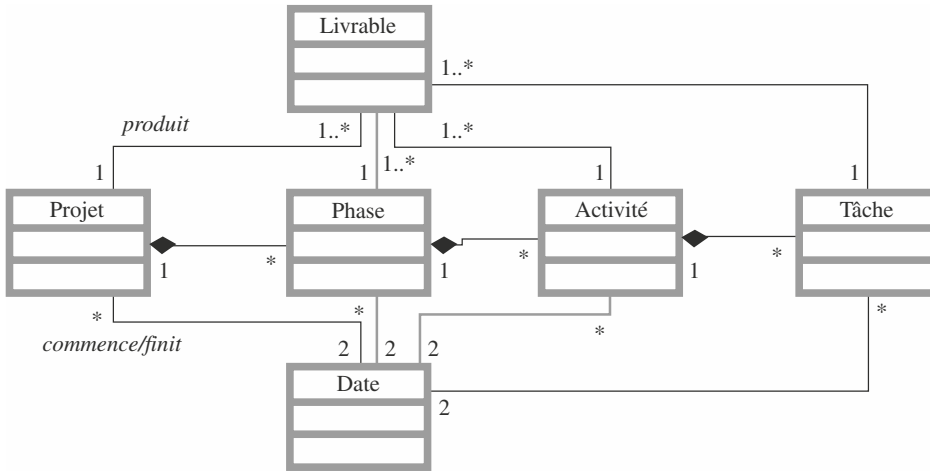


Figure 2.1 — Le découpage temporel d'un projet.

L'ensemble ordonnancé des phases d'un projet s'appelle le **cycle de vie du projet**.

Le découpage temporel poursuit deux objectifs : il balise et il guide. En effet, chaque date représente un jalon permettant de marquer les points de décision du parcours. De plus, la distribution dans le temps reflète un cheminement intellectuel.

1. Parfois, la phase est découpée en étapes, elles-mêmes composées d'activités ou de tâches.

La plupart des méthodes de conception de système d'information proposent un tel cheminement, souvent appelé cycle de développement ou modèle de cycle de vie. Le découpage temporel est souvent de type descendant (*top-down*, du haut vers le bas), c'est-à-dire qu'il favorise :

- une visibilité croissante, car les résultats sont de plus en plus précis et la maille d'étude de plus en plus fine ;
- une progression réelle des travaux, dans la mesure où les résultats consolidés en fin d'une étape ne sont pas remis en question dans les étapes suivantes.

Dans tous les cas, le cheminement intellectuel permet de converger, c'est-à-dire d'éviter de partir continuellement sur de nouvelles pistes. Il doit être économique, évitant de faire et défaire. Le client et le chef de projet ont tous deux intérêt à découper le projet dans le temps, car ainsi :

- *Le client peut valider et orienter le projet.*

  Le découpage temporel permet au client de s'assurer progressivement que ce qui a été conçu traduit bien les objectifs généraux, de faire des choix, éventuellement de réorienter le travail. En général, la fin d'une phase se traduit par la livraison d'une fourniture contractuellement définie. La fin d'une activité donne lieu à la remise de produits intermédiaires.

- *Le chef de projet peut baliser le déroulement du projet.*

  S'il a découpé son projet en tranches, le chef de projet peut effectuer une planification et en suivre pas à pas l'avancement. La fin de chaque tranche est comme un jalon où il regarde plus particulièrement s'il n'y a pas des signaux inquiétants concernant l'état de santé du projet.

Le *critère structurel* permet d'organiser le travail en se basant sur la structure du produit final : la décomposition fait apparaître les différents modules qu'il faut obtenir (figure 2.2). L'utilisation de ce critère requiert une visibilité suffisante sur le résultat à produire.

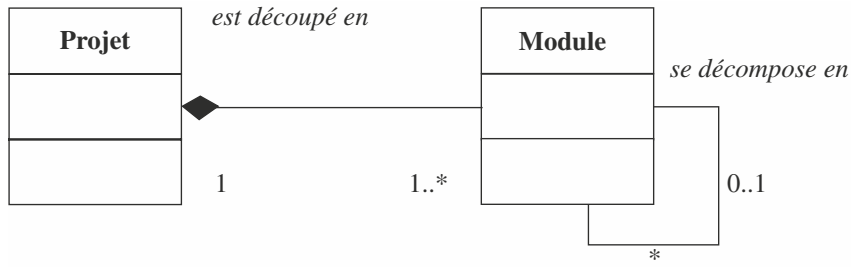


Figure 2.2 — Le découpage structurel d'un projet.

En plus du découpage temporel et surtout si le projet est de taille importante, on recourt au découpage structurel qui présente plusieurs avantages.

- *Maîtriser le projet*.

  Le découpage conduit à des sous-ensembles cohérents d'une taille plus réduite et plus facile à maîtriser.

- *Répartir les responsabilités*.

  L'autonomie des modules autorise leur répartition dans des sous-projets séparés, dont la réalisation est confiée à différents responsables ou éventuellement sous-traitée.

- *Réduire les délais planifiés*.

  Certains modules indépendants sont développés en parallèle, ce qui permet d'avancer la date théorique d'achèvement du projet.

- *Avoir un développement incrémental*.

  Pour différentes raisons (taille, budget, délais), on choisit parfois de développer un système d'information par versions successives (en général trois ou quatre), chaque version comportant un nombre croissant de modules par rapport à la précédente. Le découpage structurel est alors essentiel pour définir le contour de chacune d'elles.

2.2 LES DÉCOUPAGES NORMALISÉS

Les normes internationales proposent trois découpages : PBS, WBS et OBS.

Le découpage PBS, *Product Breakdown Structure* (structure de décomposition du produit), correspond au découpage structurel : ce sont les différents composants du produit final. Soit par exemple un progiciel de gestion des valeurs mobilières à développer (figure 2.3). Le PBS représente le découpage du progiciel en modules, chacun assurant une fonction spécifique. Trois principaux modules ont été identifiés : un référentiel des différents titres (la base Valeur), la tenue de la comptabilité Titres (comptabilité) et la gestion d'un carnet d'ordres passés sur les marchés (ordres de Bourse). Ce dernier module contient à son tour deux parties : l'enregistrement des ordres (carnet d'ordres) et le traitement administratif des ordres effectivement passés (dénouement).

Le PBS est parfois appelé « structure du produit » ou « arborescence produit », *« représentation des liens de composition entre les divers constituants d'un produit complexe »* [AFITEP, 2000].

Le découpage WBS, *Work Breakdown Structure* (structure de décomposition du travail), représente, sous forme d'une arborescence, les différents composants de

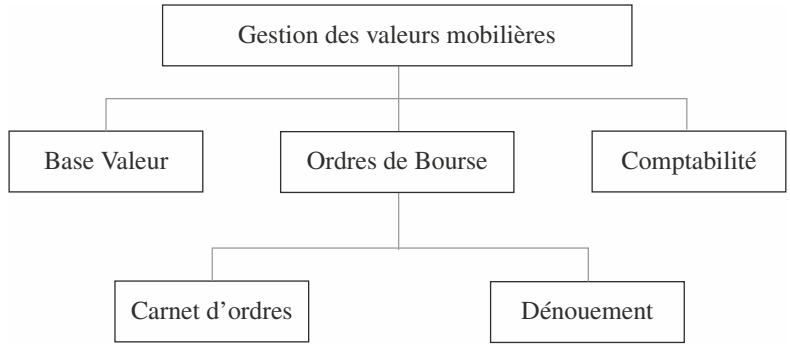


Figure 2.3 — Découpage PBS simplifié.

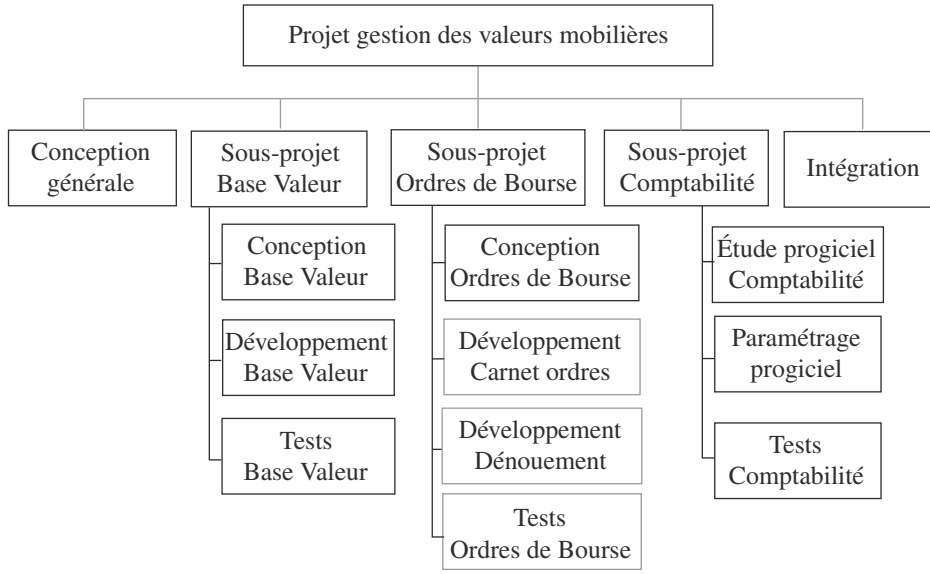


Figure 2.4 — Découpage WBS simplifié.

travail nécessaires pour parvenir au résultat tel qu'il est décrit dans le PBS. Il s'appuie, en général, à la fois sur le critère structurel et sur le critère temporel.

L'AFITEP et l'AFNOR le définissent comme le « découpage hiérarchisé et arborescent du processus de réalisation en éléments plus faciles à analyser et à maîtriser, appelés lots de travaux ou tâches ». Il apporte une réponse aux deux questions : « Que doit-on faire ? Comment doit-on s'y prendre ? ».

En reprenant l'exemple ci-dessus, le WBS pourrait se présenter comme illustré à la figure 2.4.

L'on mène d'abord une conception générale sur l'ensemble sur domaine ; l'on poursuit le travail à travers trois sous-projets, dont on décidera, au moment de les planifier, de les mener ou non en parallèle. Chacun comporte une phase

de conception, puis un développement et une phase de tests. Dans le cas du sous-projet Ordres de bourse, on a distingué deux phases de développement, correspondant à chacun des sous-modules Carnet d'ordres et Dénouement ; dans le cas du sous-projet Comptabilité, on s'appuie sur un progiciel existant qu'il s'agit de découvrir et paramétrer. Enfin, les trois logiciels issus des sous-projets font l'objet d'une intégration.

L'AFITEP traduit le terme anglais WBS par « Organigramme des tâches » ou « OT ». L'IPMA considère cependant que l'OT est la représentation graphique du WBS, sous forme d'un diagramme tel celui que nous présentons au chapitre 4 (réseau PERT ou diagramme des antécédents). Le logiciel MS Project, dans sa version 2003, utilise le terme OT dans cette même acception.

Le PMI propose pour WBS la traduction de SDP (Structure de découpage du projet), comme nous l'indiquons au chapitre 17. Le niveau le plus bas de l'arborescence est appelé « lot de travail ». Chaque lot de travail pourra ensuite être décomposé en activités pour une planification détaillée.

L'OBS, *Organisation Breakdown Structure* (structure de décomposition de l'organisation), reprend le WBS et fait apparaître les noms des personnes responsables de la production des différents éléments.

Dans notre exemple (figure 2.5), le chef du projet global a la responsabilité directe de la conception générale et de l'intégration. Les trois sous-projets sont conduits par d'autres personnes, les différents travaux sont affectés aux ressources du projet.

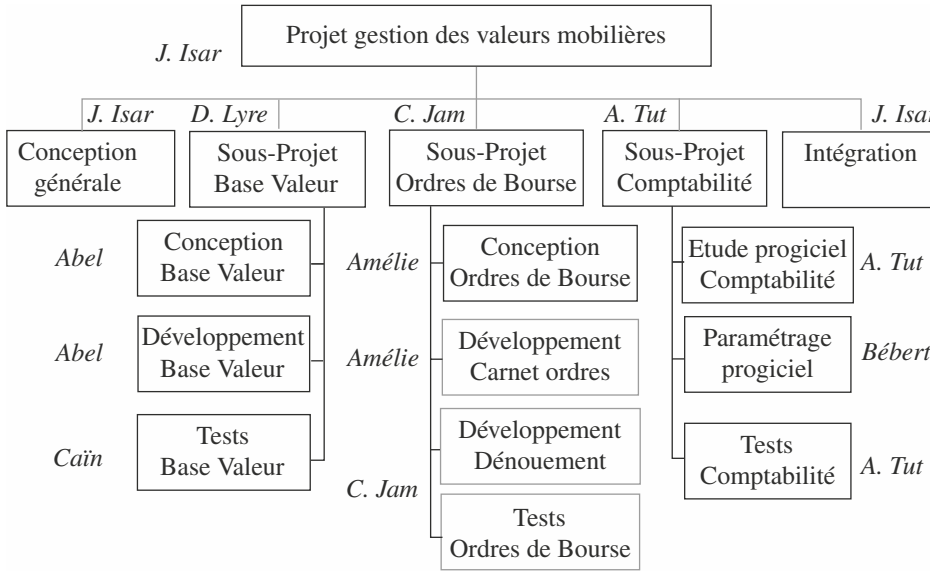


Figure 2.5 — Découpage OBS simplifié.

L'OBS est parfois appelé « organigramme fonctionnel » ou « OF ». Il représente *« la structure des différents niveaux de responsabilités de réalisation de l'ensemble des lots de travaux d'un même organigramme des tâches »* [AFITEP, 2000].

2.3 LE DÉCOUPAGE STRUCTUREL

Le découpage structurel repose sur la possibilité de découper le domaine en sous-ensembles quasi indépendants. Il peut être fait à plusieurs niveaux.

Un premier niveau est celui du découpage d'un système d'information en domaines. Un domaine peut être défini comme un sous-ensemble du système d'information global, qui possède en propre des informations — c'est-à-dire qu'elles sont créées et mise à jour dans le domaine — et des processus. Cela implique que souvent un domaine est transversal à plusieurs entités de l'organisation (site, département, service…).

Un second niveau de découpage est celui qui identifie des modules, comme dans l'exemple donné à la figure 2.3, illustrant le PBS. Chaque module peut être à son tour découpé en sous-modules. Il y a en général deux façons d'approcher ce découpage en modules : par la vision statique ou par la vision dynamique.

Le découpage par la vision statique s'appuie sur une macromodélisation des entités, c'est-à-dire le repérage des principaux concepts d'information à l'aide d'un modèle Entité-Relation ou un diagramme de classes UML. À chaque entité principale correspond un module. Dans l'exemple du domaine Gestion des valeurs mobilières, le découpage en trois premiers modules découle de la vision statique (figure 2.6). En effet, une modélisation des entités ferait apparaître trois principales classes : Valeur mobilière, Ordre de bourse et Écriture comptable.

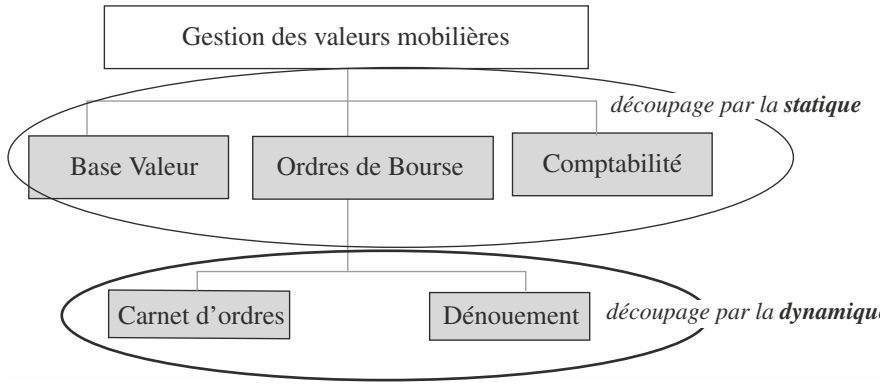


Figure 2.6 — Découpage structurel.

Le découpage par la vision dynamique s'appuie sur une identification des principaux processus du domaine. À chaque processus correspond un module. Dans l'exemple du domaine Gestion des valeurs mobilières, le découpage du module Ordres de bourse en deux sous-modules découle de la vision dynamique (figure 2.6). En effet, on a distingué deux processus : l'entrée d'un ordre dans le carnet et sa transmission à une société de bourse, en réponse à une demande d'un client ; le suivi du traitement de l'ordre, déclenché par la réception de l'avis d'opéré par la société de bourse.

2.4 LE CYCLE DE VIE STANDARD

2.4.1 Le contenu du cycle de vie standard

Le découpage temporel des projets industriels relevant d'une production unitaire est la référence initiale : c'est celui que l'on trouve souvent dans les ouvrages traitant de gestion de projet.

Ce cycle de vie standard se compose des phases suivantes :

- étude de faisabilité ;
- définition des solutions ;
- conception détaillée ;
- réalisation.

L'*étude de faisabilité* comprend des travaux d'analyse, des travaux de recherche, des études sur le terrain. Il s'agit de vérifier si le projet est techniquement réalisable. Par exemple, si l'on veut construire un immeuble, il faut vérifier que le terrain et le sous-sol le permettent, à un coût acceptable.

La *définition des solutions* donne une représentation précise de l'objectif à atteindre. Les solutions possibles sont étudiées de façon détaillée. Pour cela, on utilise différents moyens, par exemple faire des essais, élaborer une maquette, construire un prototype. Au terme de cette phase, une solution est alors choisie et l'on dispose de spécifications exactes.

La *conception détaillée* sert à préparer les contrats de réalisation. En effet, un grand projet industriel fait souvent intervenir des corps de métier différents. Ces contrats contiennent les cahiers des charges pour les sous-traitants. Les spécifications techniques décrivent la mission et les moyens pour la réaliser. Aucune ambiguïté ne doit subsister. Même si l'on ne recourt pas à des fournis-

seurs extérieurs, le principe du cahier des charges exhaustif est fondamental. Les fournitures livrées à l'étape suivante sont acceptées sur la base de son contenu.

La *réalisation* est l'exécution des contrats par les sous-traitants, conformément aux cahiers des charges. Cette phase se termine en général par une procédure d'acceptation officielle.

Dans ce découpage de référence, la réalisation du projet passe par une définition complète, précise et détaillée de l'objectif. Les trois premières phases représentent généralement 10 % des efforts et des dépenses. Le management de projet (planification, organisation, suivi) porte essentiellement sur la phase de réalisation.

2.4.2 Les problèmes posés par le découpage standard

Le découpage temporel standard ne peut guère être utilisé tel quel dans un projet système d'information.

D'abord la notion de cahier des charges y est déclinée à plusieurs moments. En effet, la plupart des phases du cycle de développement peuvent conduire à un cahier des charges qui oriente le travail de l'étape ultérieure. On parle alors d'un « cahier des charges d'analyse » ou d'un « cahier des charges de conception » ou encore d'un « cahier des charges de réalisation ».

Ensuite, dans le découpage temporel standard, on postule la possibilité pour le client d'établir une description complète de ce qu'il attend. Or, dans notre domaine, l'élaboration des spécifications, c'est-à-dire la détermination des besoins et des solutions adéquates, est un problème majeur. Il y a souvent une construction progressive, qui s'appuie sur des allers-retours entre une solution de gestion et des possibilités techniques. Les besoins ne préexistent pas, ils émergent. Pour le chef de projet, les étapes d'analyse et conception sont risquées. Une part non négligeable du budget y est consacrée — jusqu'à 40 % — sans toujours atteindre la qualité espérée. Par conséquent, la gestion de projet commence dès le début du projet.

De plus, l'élaboration d'un cahier des charges de réalisation est un travail coûteux. En effet, l'écriture de spécifications souffre de l'absence de *composants réutilisables*. Par exemple, la définition d'un système de gestion des clients serait allégée si l'on pouvait réutiliser des fonctions standard (création d'un prospect, mise à jour d'une personne…) sans les reconcevoir et les décrire intégralement. Ce n'est malheureusement pas le cas. Dans le domaine industriel, le recours à la CAO (conception assistée par ordinateur), pour concevoir un nouveau produit, favorise l'utilisation de composants existants, élémentaires ou agrégés. On peut les modifier ou les assembler différemment, en obtenant automatiquement les

nouvelles cotes. Les ateliers de génie logiciel[1], qui aident à concevoir un système d'information, offrent un métamodèle et non des modèles concrets, que l'on pourrait stocker dans une bibliothèque de composants conceptuels.

Par ailleurs, on a peu de *spécifications implicites*. Par exemple, avant de dessiner les plans d'un immeuble, on définit sa destination. Il en découle un certain nombre de spécifications standard. Différents critères (standing, segment de marché visé…) donnent des normes orientant la conception (matériaux, taille des pièces, garage…). Si l'on transposait à ce type de projet la démarche système d'information, l'immeuble serait conçu avec la plupart des futurs occupants !

Pour élaborer des spécifications, on est donc conduit à chercher une approche permettant de produire un résultat de qualité, en maîtrisant les coûts et les délais.

Dans les projets de système d'information, le problème de faisabilité se pose aussi de façon particulière. Dans un sens, on peut toujours faire quelque chose, dans la mesure où les contraintes ne sont en général pas d'ordre physique, mais techniques, financières ou organisationnelles. Ce n'est pas tout ou rien, comme la fusée Ariane qui part ou ne part pas. Ainsi, il y a une vingtaine d'années, une grande banque de Madagascar avait lancé un projet de liaison avec ses agences. Compte tenu des infrastructures de télécommunication à l'époque, le projet a mis en place un système reposant sur l'envoi de disquettes entre les sites.

Cela rend le processus de développement plus facile en phase finale, car on peut démarrer un système avec des fonctionnalités réduites ; mais plus périlleux en phase de conception parce qu'il peut s'écouler beaucoup de temps avant d'avoir une vision claire de ce qu'on va faire, et il arrive parfois que le projet s'enlise.

Par exemple, dans la mise en œuvre en 1991 du projet Relit (Règlement-livraison de titres), permettant de relier les acteurs bancaires de la place de Paris, les problèmes de temps de réponse provenant de la capacité des machines et du réseau, ont conduit à fonctionner pendant une période intermédiaire avec un nombre limité de banques et sociétés de bourse.

De façon analogue, au démarrage du projet FNCV (Fichier national des chèques volés, tenu par la Banque de France) une partie de l'alimentation par les différents acteurs s'est faite par disquette, bande magnétique, voire déclaration papier.

En conclusion, on peut dire que dans le domaine qui nous intéresse, le management de projet couvre l'ensemble du cycle et s'appuie sur des découpages temporels pouvant s'éloigner du découpage standard.

1. AGL ou CASE : *Computer Aided System Engineering* (Ingénierie de système assisté par ordinateur). On distingue parfois les AGL de conception (appelés *upper-case* en anglais) qui assistent lors de la partie amont du cycle et les AGL de réalisation (appelés *lower-case* en anglais) qui allègent la partie aval. De plus en plus d'outils couvrent tout le cycle.

Nous allons d'abord décrire le découpage classiquement utilisé, que l'on retrouve dans beaucoup de méthodes de conception. Puis nous présenterons un panorama des découpages temporels génériques utilisés dans notre domaine, souvent appelés modèles de développement. Nous terminerons sur certains découpages spécifiques.

2.5 LE DÉCOUPAGE CLASSIQUE

Durant les deux dernières décennies, les méthodes de développement de système d'information ont proposé un découpage temporel de référence, parfois appelé « cycle de vie classique ». La figure 2.7 présente ce découpage avec les correspondances entre le vocabulaire de la méthode Merise [Nanci, 2001], celui de la méthode SDMS, et celui issu de la méthode MCP qui a servi de référence à la norme AFNOR Z67-101.

NORME AFNOR Z67-101	MERISE	SDMS
	Schéma directeur	
Étude préalable	Étude préalable	
Exploration	Observation	DBS (Définition des besoins du système)
Conception	Conception/Organisation	CAS (Conception de l'architecture du système)
Appréciation	Appréciation	
Conception détaillée	Étude détaillée	SES (Spécifications externes du système)
Réalisation	Étude technique	SIS (Spécifications internes du système)
	Réalisation	Programmation
		Test
Mise en œuvre	Mise en œuvre	Conversion
		Installation
Évaluation	Qualification	Bilan

Figure 2.7 — Découpage temporel classique des méthodes de développement S.I.

2.5.1 Le schéma directeur

SD	EP	ED	ET	REAL	MEO	QUALIF

L'*objectif* d'un schéma directeur est de définir le scénario d'évolution du patrimoine informatique, sous l'un ou l'autre de ces trois angles :

- évolution de l'architecture technique (matériels, réseaux) ;
- évolution de l'architecture applicative (données communes, identification des domaines, évaluation des applications) ;
- évolution de la fonction informatique (méthodes, normes, outils).

Le champ d'un schéma directeur est l'entreprise tout entière ou bien un grand secteur de l'entreprise ; on l'appelle alors schéma directeur sectoriel. Ce type de projet est mené par une petite équipe (2 à 3 personnes), en un temps limité (quelques semaines). Le directeur informatique est directement impliqué, mais aussi les responsables des autres directions.

Le *résultat* d'un schéma directeur dépend de l'objectif majeur. En général on a une photographie de la situation existante, un diagnostic et des orientations d'évolution choisies à partir de deux ou trois scénarios.

Le travail sur l'architecture applicative débouche sur une cartographie des domaines et une modélisation des principaux concepts. Cela conduit à définir des objectifs et des priorités par domaine et par application.

2.5.2 L'étude préalable

SD	EP	ED	ET	REAL	MEO	QUALIF

C'est souvent à partir de ce point que l'on fait partir le cycle de vie d'un système d'information spécifique à un domaine. On aborde une étude préalable soit à l'issue d'un schéma directeur, soit hors de toute opération schéma directeur, pour repenser une application vieillissante sur un domaine bien identifié ou pour répondre à un besoin nouveau.

Par exemple, le domaine du crédit a connu ces dernières années de profondes évolutions réglementaires et commerciales. L'application gestion des prêts est à la limite de ses possibilités d'évolution et doit être refaite. Dans le domaine commercial, on peut mener une étude préalable pour repenser le système d'information dans une optique de *e-commerce*.

L'*objectif* d'une étude préalable est double. D'une part, il s'agit de faire des choix structurants pour la future application : évaluer l'adéquation de la solution aux objectifs, choisir éventuellement entre plusieurs solutions, évaluer l'investissement (budget, temps), ajuster la solution à l'enveloppe si cela est nécessaire. D'autre part, il s'agit de fournir une base de référence pour la suite du projet : le rapport d'étude préalable peut donc être considéré comme un cahier des charges pour l'étude détaillée.

Une étude préalable comporte trois étapes : observation, conception-organisation et appréciation.

L'*objectif* de l'étape observation est de donner une représentation pertinente du domaine étudié, suffisante pour porter un diagnostic et mettre en évidence des besoins. Le *résultat* comprend :

- une structuration du domaine en processus, qui va ensuite guider un éventuel découpage structurel, pour établir un WBS par exemple ;
- le choix d'un sous-ensemble représentatif (SER) : en effet, si le domaine est important, on va se limiter à une partie du domaine, en utilisant la notion de variante de procédure ;
- une description du fonctionnement du SER ;
- une description modélisée des données ;
- un diagnostic.

L'*objectif* de l'étape conception-organisation est de proposer une ou plusieurs solutions, aux niveaux conceptuel et organisationnel, sur tout ou partie du domaine. Le *résultat* comprend un modèle de données consolidé ou enrichi, ainsi qu'une description d'au moins une variante de chaque processus, avec les traitements et les règles de gestion.

L'*objectif* de l'étape appréciation est d'une part de dresser un bilan des avantages attendus et des coûts prévisibles (étude de rentabilité), d'autre part d'élaborer un plan pour la poursuite du projet. On peut ainsi identifier différents sous-projets qu'il faut ordonnancer. Le découpage en sous-projets repose sur un découpage structurel ; par exemple, on peut définir un sous-projet par processus. L'ordonnancement se fait selon :

- une éventuelle priorité stratégique de certains processus ;
- la périodicité (traitements quotidiens, puis mensuels, puis annuels) ;
- les contraintes logistiques (arrivée d'un matériel, mise en place d'un réseau).

2.5.3 L'étude détaillée

SD	EP	**ED**	ET	REAL	MEO	QUALIF

L'*objectif* d'une étude détaillée est de concevoir et décrire de façon exhaustive la solution sur tout le champ de l'étude. Elle sera ensuite complétée par l'étude technique. Les spécifications ainsi obtenues doivent faire l'objet d'un consensus entre futurs utilisateurs et informaticiens. Elles représentent le cahier des charges pour la réalisation.

Le *résultat* comprend toute la vision externe du système : l'interface homme-machine (maquettes d'écran, cinématique), la description des traitements à une maille suffisamment fine pour qu'il n'y ait plus d'ambiguïté fonctionnelle, ainsi que les sorties (maquettes d'état). On y ajoute souvent l'organisation à mettre en place et le planning détaillé.

2.5.4 L'étude technique

SD	EP	ED	**ET**	REAL	MEO	QUALIF

L'*objectif* de cette phase, qui ne concerne que les informaticiens, est d'optimiser les structures physiques de données et de construire les traitements (dossier programmes) en essayant de préparer la réutilisation du code.

Le *résultat* comprend des normes techniques, des dossiers de programme et la structure physique des données. Il complète le cahier des charges de réalisation.

2.5.5 La réalisation

SD	EP	ED	ET	**REAL**	MEO	QUALIF

Cette phase est parfois appelée « développement ». L'*objectif* est de produire un logiciel testé. Elle comprend donc des tâches d'élaboration de jeu d'essai, de programmation et de test.

Elle se termine par une procédure d'acceptation officielle est appelée *recette* : le client fournit un jeu d'essai et vérifie avec le fournisseur la conformité du logiciel à ce qu'il avait demandé. Dans la pratique, la recette fait souvent une étape séparée. On effectue parfois une recette provisoire après la réalisation et une recette définitive après la mise en œuvre. Dans une relation contractuelle donnant lieu à des flux financiers, la recette conditionne le paiement.

2.5.6 La mise en œuvre

SD	EP	ED	ET	REAL	MEO	QUALIF

L'*objectif* est de préparer le démarrage effectif de la nouvelle application. Cette phase comprend notamment le paramétrage, la reprise ou l'alimentation des données, le développement d'interfaces, la formation des utilisateurs, l'installation d'environnement d'exploitation. La gestion du changement décrite au chapitre 5 relève de la mise en œuvre, qui souvent commence dès la fin de l'étude détaillée.

2.5.7 La qualification

SD	EP	ED	ET	REAL	MEO	QUALIF

L'*objectif* est de réaliser des tests dans l'environnement opérationnel et de tirer un bilan du système d'information installé, selon différents critères qualité. Ces derniers sont présentés au chapitre 8 qui regroupe tous les aspects de la qualité.

2.6 LES MODÈLES DE DÉVELOPPEMENT

On considère aujourd'hui qu'on ne peut plus avoir une démarche unique, mais qu'il faut construire le découpage temporel en fonction des caractéristiques de l'entreprise et du projet. On s'appuie pour cela sur des découpages temporels génériques, appelés *modèles de développement* (*process models*) ou modèles de cycle de vie.

Les principaux modèles sont :

- le modèle du *code-and-fix* ;
- le modèle de la transformation automatique ;
- le modèle de la cascade ;
- le modèle en V ;
- le modèle en W ;
- le modèle de développement évolutif ;
- le modèle de la spirale.

2.6.1 Le modèle du code-and-fix

Ce modèle repose sur la possibilité d'une détermination facile des besoins (figure 2.8). Après une étape brève de compréhension de l'objectif, l'application est développée. Ensuite, plusieurs cycles de mise au point, parfois en collaboration avec l'utilisateur du futur système, permettent d'atteindre le résultat visé.

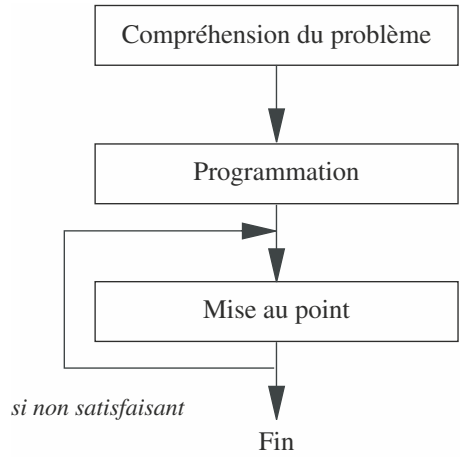


Figure 2.8 — Le modèle du *code-and-fix*.

2.6.2 Le modèle de la transformation automatique

Le modèle de la transformation automatique (*transform model*) est basé sur la possibilité de transformer automatiquement des spécifications en programmes (figure 2.9). L'essentiel de l'effort va donc porter sur une description exhaustive des spécifications. Celles-ci devront être complètement validées. Une succession de cycles de spécification/validation s'achève par la génération de code.

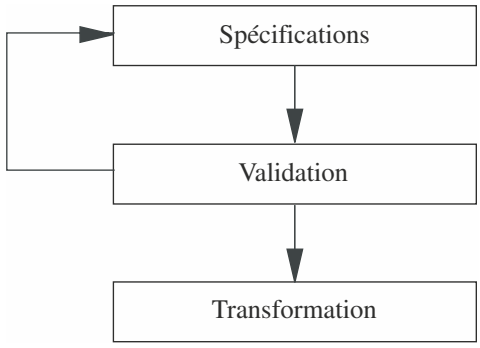


Figure 2.9 — Le modèle de la transformation automatique.

2.6.3 Le modèle de la cascade

Le modèle de la cascade (*waterfall model*) est totalement opposé au modèle du *code-and-fix*. Il a comme objectif majeur de jalonner rigoureusement le processus de développement et de définir de façon précise les rôles respectifs du fournisseur — qui produit un livrable — et du client — qui accepte ou refuse le résultat.

Le découpage temporel se présente donc comme une succession de phases correspondant à une approche descendante (figure 2.10). Chacune donne lieu à une validation officielle : on ne passe à la suivante que si le résultat du contrôle est satisfaisant. Sinon, on modifie le livrable pour qu'il devienne acceptable. En revanche, il n'y a pas de retour possible sur les options validées à l'issue des phases antérieures.

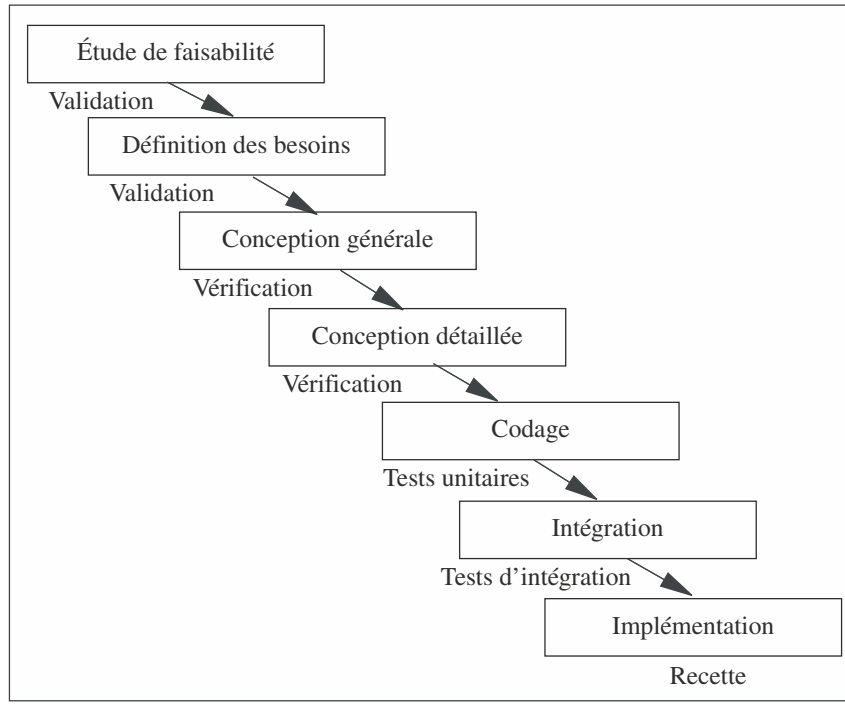


Figure 2.10 — Le modèle de la cascade.

2.6.4 Le modèle en V

Ce modèle (figure 2.11) est une amélioration du modèle de la cascade.

Il vise, d'une certaine façon, à réduire ce que l'on a appelé l'« effet tunnel » : à partir d'un moment donné, les clients perdent la visibilité sur le projet. Quand ce dernier ressort du tunnel, on découvre des livrables qui ne sont pas toujours ceux que l'on attendait, non pas qu'ils ne soient pas conformes aux spécifications,

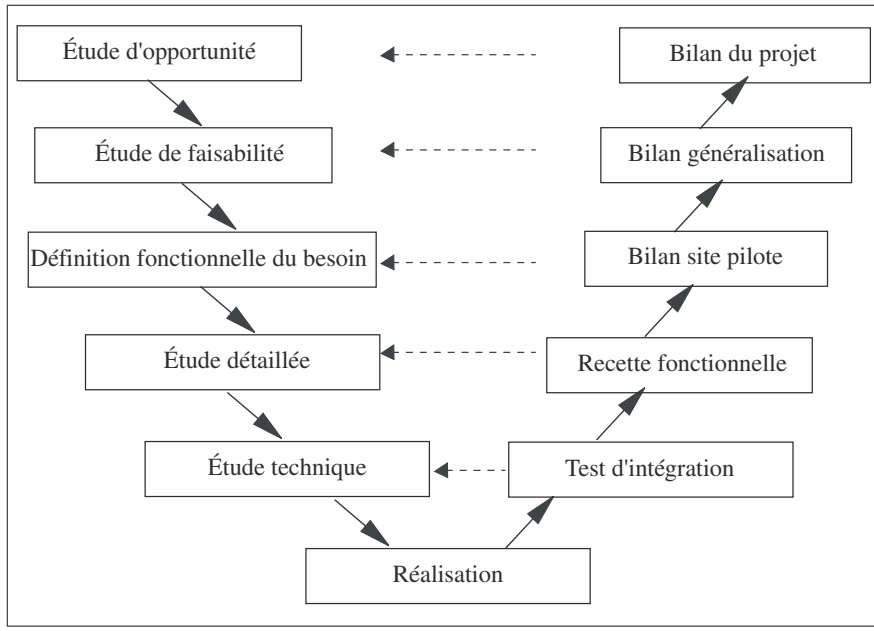


Figure 2.11 — Le modèle en V.

mais parce les spécifications sont parfois impuissantes à décrire les attentes. La seule validation de documents est donc insuffisante.

Dans le modèle en V, on va donc s'attacher dans chacune des phases de la première branche du V à expliciter les critères d'appréciation et d'acceptation du système aux étapes correspondantes de la deuxième branche du V. Par exemple, l'étude détaillée produira un jeu d'essai qui sera utilisé pour effectuer la recette fonctionnelle. L'installation sur un site pilote permettra de valider la définition fonctionnelle du besoin, d'après les critères exprimés dans cette étape. Le bilan global du projet vérifiera que les objectifs initiaux formulés dans l'étude d'opportunité ont bien été atteints.

Pour les projets importants, il est recommandé de décomposer le système en sous-ensembles (découpage structurel) : l'étape de définition fonctionnelle du besoin conduit à un découpage en composants, donnant lieu à études, réalisations, tests et recettes séparés, avant intégration (figure 2.12).

2.6.5 Le modèle en W

Ce modèle enrichit le modèle en V (figure 2.13) dans le même esprit d'anticipation sur le livrable final.

La première partie du W vise à dégager avec les clients des orientations solides pour la conception ou bien à explorer les possibilités d'une nouvelle technique.

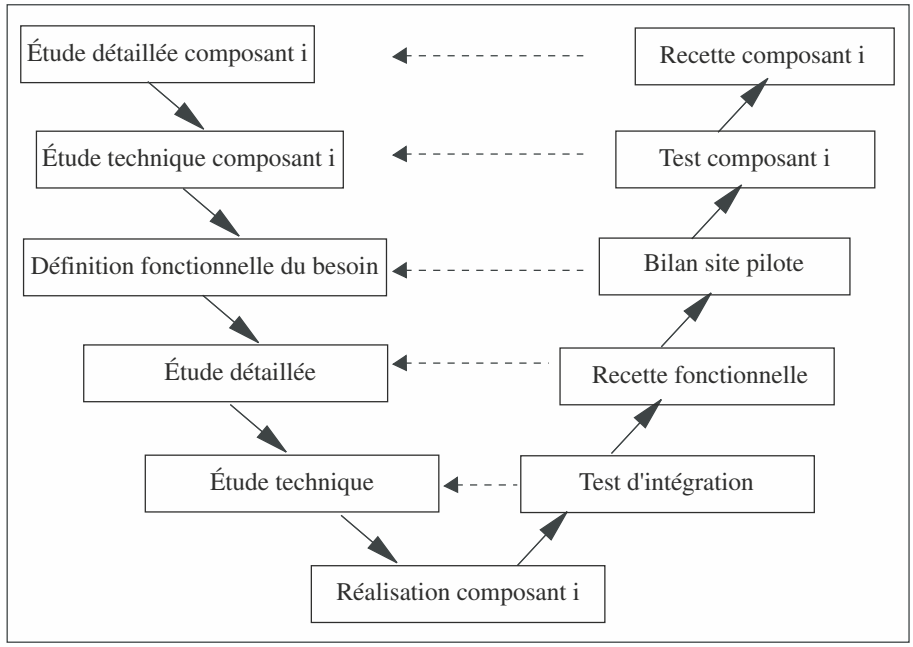


Figure 2.12 — Le modèle en V avec des composants.

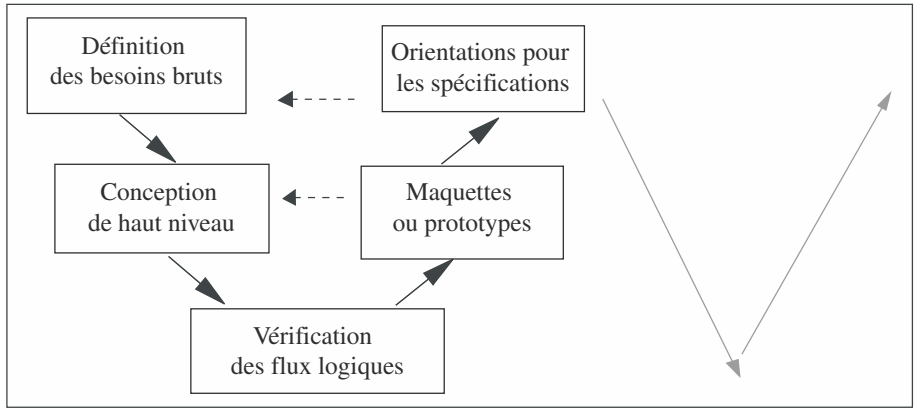


Figure 2.13 — Le modèle en W.

Le développement de maquettes ou prototypes permet une validation plus concrète, voire une expérimentation.

2.6.6 Le modèle de développement itératif

L'objectif du modèle du développement itératif, qui fut initialement appelé évolutif *(evolutionary design model)*, est de construire progressivement le système de façon participative (figure 2.14). Il repose sur l'idée que les besoins ne

peuvent s'exprimer qu'après une expérimentation, même sur un système rudimentaire ou incomplet. Chaque cycle aboutit à une nouvelle version du système : on s'arrête lorsque le client juge le système satisfaisant.

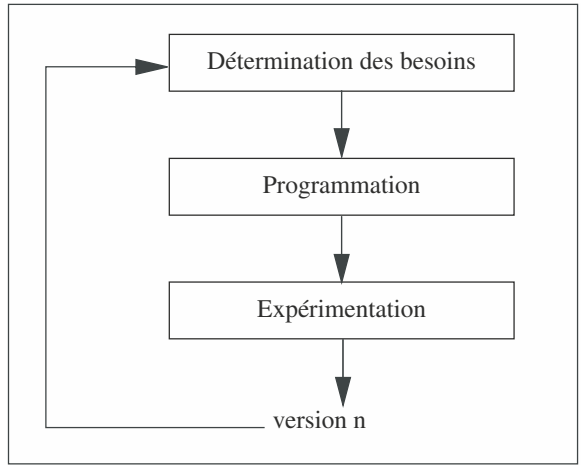


Figure 2.14 — Le modèle du développement itératif.

Lorsque l'on veut indiquer que non seulement chaque itération fournira au commanditaire un ensemble de fonctionnalités qui forment une version exécutable du système cible, mais que chaque version apportera de nouvelles fonctionnalités, on qualifie parfois le modèle d'*incrémental*.

Cependant, tout cycle itératif n'est pas obligatoirement incrémental : la première itération peut livrer un système complet qui est ensuite affiné et ajusté dans les itérations suivantes.

2.6.7 Le modèle de la spirale

Le modèle de la spirale (*spiral model*) repose sur le même principe que le modèle évolutif (figure 2.15), mais il s'inscrit dans une relation contractuelle entre le client et le fournisseur. De ce fait les engagements et validations présentent un caractère formalisé.

Chaque cycle donne lieu à une contractualisation préalable, s'appuyant sur les besoins exprimés lors du cycle précédent. Un cycle peut être considéré comme une phase, qui comporte les six étapes suivantes :

- analyse du risque ;
- développement d'un prototype ;
- simulation et essais du prototype ;

- détermination des besoins (à des mailles différentes selon le cycle), à partir des résultats des essais ;
- validation des besoins par un comité de pilotage ;
- planification du cycle suivant.

Le dernier cycle permet de développer la version finale et d'implémenter le logiciel.

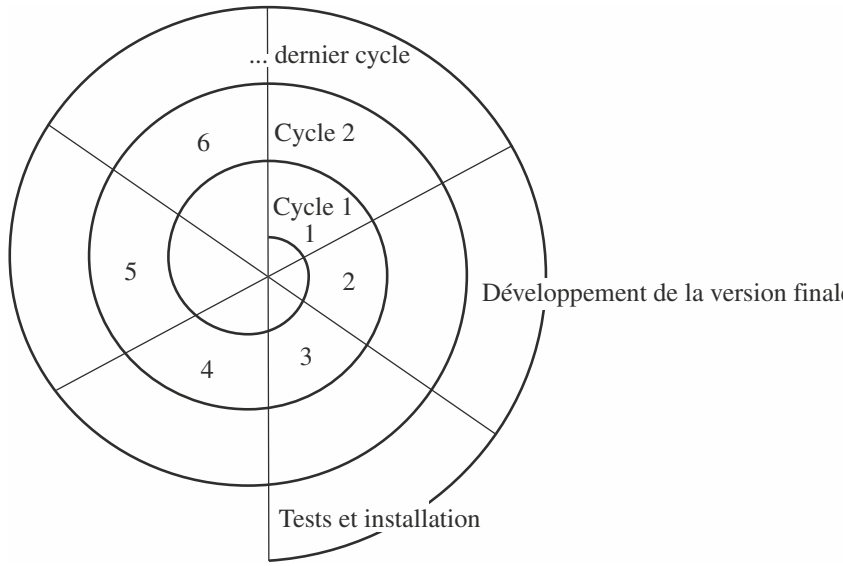


Figure 2.15 — Le modèle de la spirale.

2.7 LES MODÈLES DE CYCLE DE VIE SPÉCIFIQUES

Certains découpages temporels sont liés soit à une méthode, soit à un type de projet bien particulier. Nous en proposons deux exemples : le découpage préconisé pour mettre en place un progiciel intégré et le modèle RUP proposé par la société Rational Software.

Nous présentons dans le paragraphe suivant les modèles de cycle de vie que l'on trouve dans les méthodes agiles, introduites brièvement à la fin du chapitre 1.

2.7.1 Le cycle ERP

La mise en place d'un progiciel de gestion intégré, souvent appelé du terme anglo-saxon ERP (*Enterprise Resource Planning*)[1] s'appuie sur un découpage spécifique (figure 2.16).

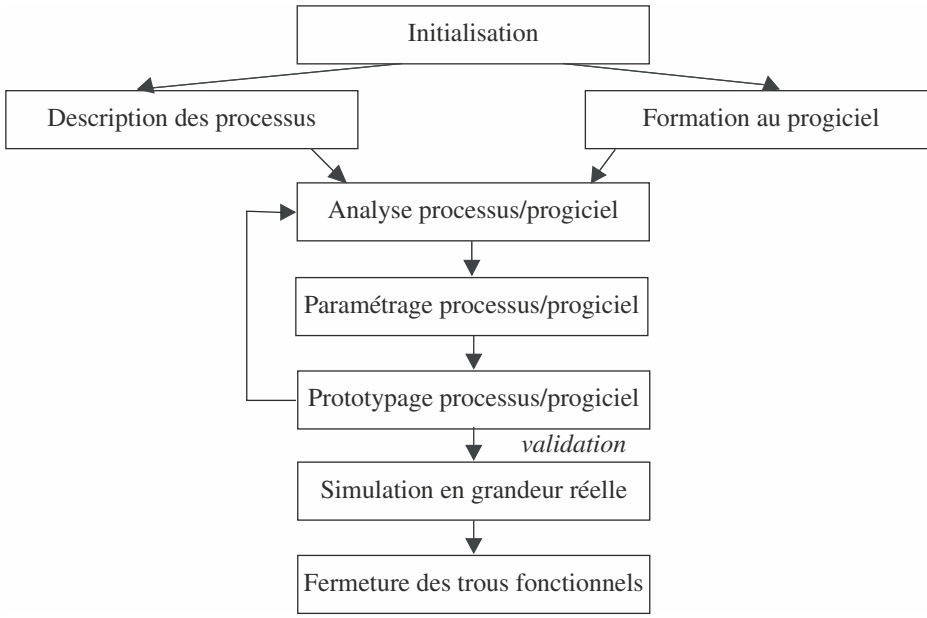


Figure 2.16 — Le cycle ERP.

En effet, il s'agit de construire, en tirant le meilleur parti du progiciel, un système améliorant la performance de l'entreprise. Deux étapes doivent donc être menées en parallèle : Description des processus et Formation au progiciel. Ensuite, il y a autant de cycles d'analyse — paramétrage — prototypage qu'il y a de processus. La validation par le Comité de pilotage permet une simulation en grandeur réelle. Il faut alors prendre en compte ce qui est resté en dehors du champ couvert par le progiciel.

2.7.2 Le modèle RUP

Le modèle RUP (*Rational Unified Process*) est représentatif d'une approche combinant plusieurs modèles. Sa structure fait l'objet d'un assez large accord, notamment parmi les praticiens (figure 2.17). Il peut être lu de la façon suivante :

1. Le cycle est constitué de quatre phases principales, que l'on retrouve globalement dans toutes les approches descendantes : étude préalable (opportunité), conception de la solution détaillée (élaboration), développement de la solution (construction) et mise en œuvre (transition)

1. Pour une description détaillée des ERP, voir l'ouvrage de Jean-Louis Tomas, *ERP et PGI sélection, méthodologie de déploiement et gestion du changement : Les clés du succès, les facteurs de risques*, Dunod, 2007.

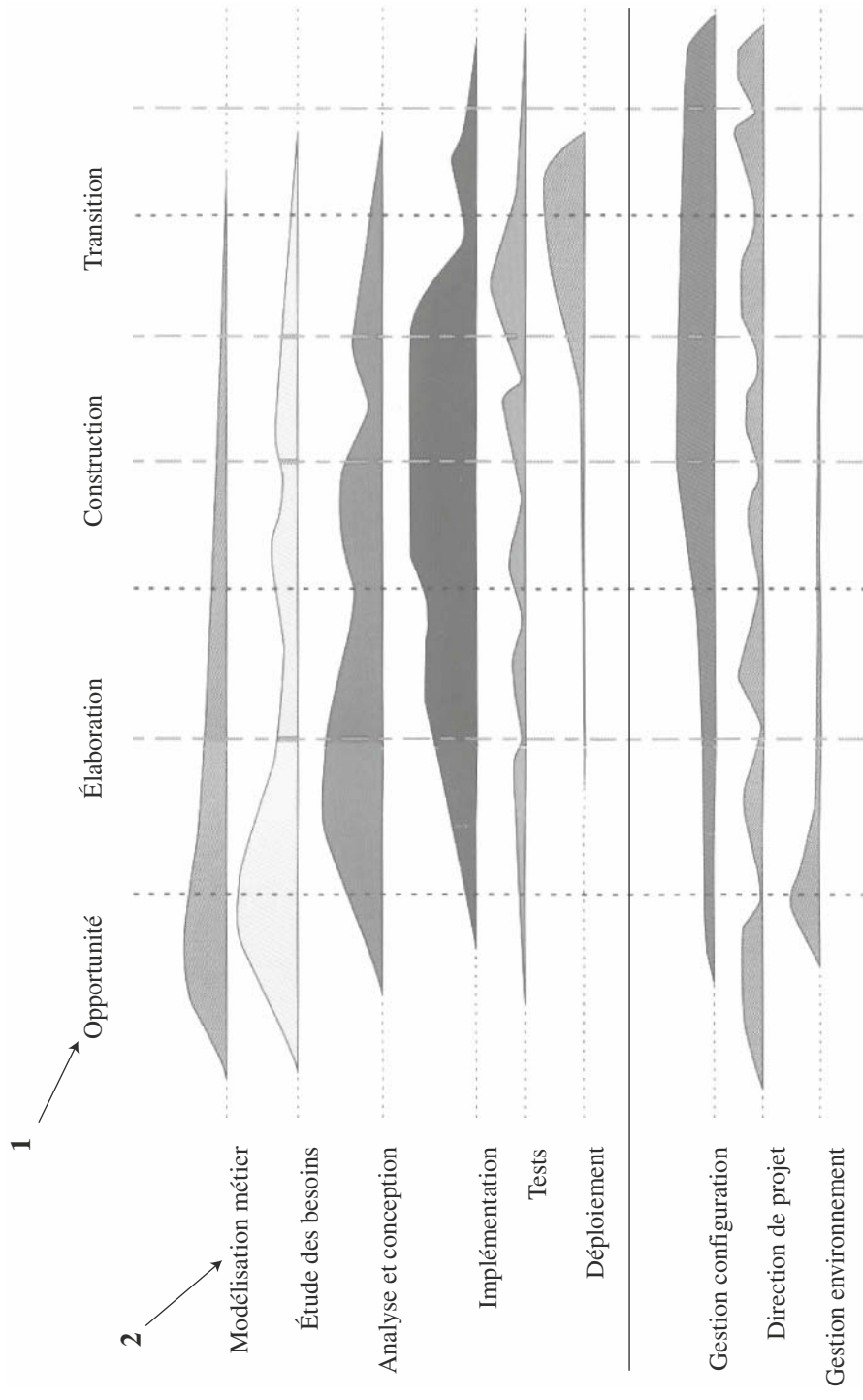


Figure 2.17 — Le cycle RUP.

2. Il existe six types de tâches qui, au lieu d'être affectées exclusivement à une phase, se retrouvent à des degrés divers dans chacune des phases. Par exemple, l'étude des besoins peut apparaître jusqu'à la fin du projet, mais la plus grande partie est effectuée dans les deux premières phases. L'implémentation (développement) a principalement lieu dans la phase de construction, mais on peut réaliser un prototype dès la première phase. Certaines tâches, comme la direction de projet, s'effectuent sur toute la durée du projet.

3. Certaines phases peuvent être menées de façon cyclique. Ainsi, l'élaboration se fait en deux cycles, conduisant par exemple à la production de spécifications externes (vision utilisateur) et spécifications techniques (vision développeur). La construction est itérative et incrémentale.

De plus, l'ensemble du modèle représente un tour de spirale, dans le cas d'une approche globale en spirale.

2.8 LES MODÈLES DE CYCLE DE VIE DES MÉTHODES AGILES

Toutes les méthodes agiles prennent en compte dans leur modèle de cycle de vie trois exigences : une forte participation entre développeurs et utilisateurs, des livraisons fréquentes de logiciel et une prise en compte de possibles changements dans les besoins des utilisateurs au cours du projet. C'est pourquoi toutes font appel, d'une façon ou d'une autre, à un modèle itératif et incrémental.

De plus, elles préconisent en général des durées de cycle de vie des projets ne dépassant pas un an.

2.8.1 Le modèle RAD

Le cycle de vie RAD conjugue modèle linéaire, structuré en cinq phases, et modèle itératif pour la phase Construction du logiciel en plusieurs modules successivement livrés[1] (figure 2.18).

La participation des utilisateurs est placée au cœur du cycle. En effet, le déroulement d'une phase comprend une ou plusieurs sous-phases, et chaque sous-phase présente une structure à trois temps, dans laquelle la tenue d'une session participative joue un rôle central (figure 2.19). Des travaux préparatoires rassemblent et construisent le matériau (modèle ou prototype) qui sera ensuite discuté par les différents acteurs et ajusté.

1. Pour une description détaillée, voir J. Hugues, B. Leblanc et C. Morley, *RAD, une méthode pour développer plus vite*, InterEditions, 1997.

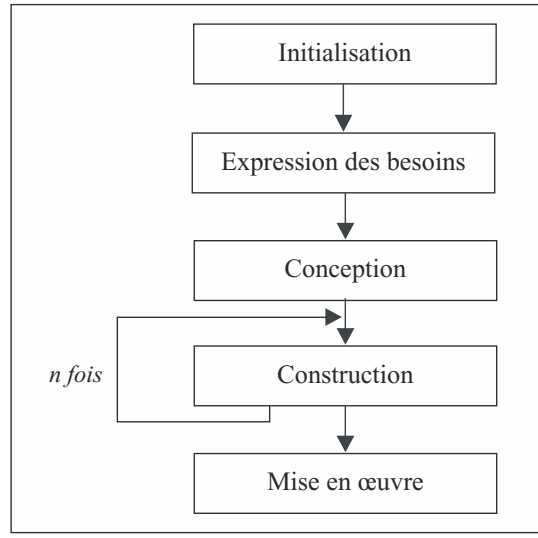


Figure 2.18 — Modèle de cycle de vie RAD.

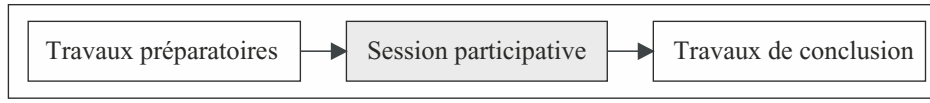


Figure 2.19 — Structure ternaire des sous-phases RAD.

Les techniques de contrôle des délais seront présentées au chapitre 4 et la forme des sessions, spécifiques à RAD, au chapitre 5.

2.8.2 Le modèle DSDM

La méthode DSDM a évolué au cours des années. L'actuelle version[1] distingue le cycle de vie du système et le cycle de vie du projet. Le premier comprend, outre les phases du projet lui-même, une phase de pré-projet qui doit conduire au lancement du projet et une phase post-projet qui recouvre l'exploitation et la maintenance de l'application.

Le cycle de vie du projet comprend cinq phases, dont deux sont cycliques (figure 2.20). Les flèches pleines indiquent un déroulement normal. Les flèches en pointillé montrent des retours possibles à une phase antérieure, soit après la phase Conception et construction, soit après celle de Mise en œuvre.

1. La description de la version 4.2 est donnée sur le site du consortium DSDM : http://www.dsdm.org/version4/2/public.

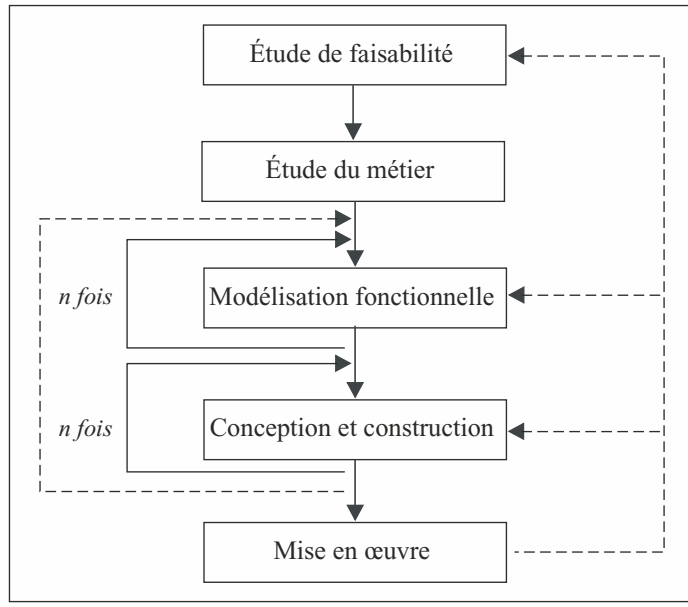


Figure 2.20 — Modèle du cycle de vie du projet DSDM.

Après une Étude de faisabilité, la phase Étude du métier permet, à travers des ateliers (*workshops*) entre équipe de projet et manageurs, de définir le périmètre du projet, avec une liste d'exigences prioritaires et une architecture fonctionnelle et technique du futur système.

La phase Modélisation fonctionnelle est une suite de cycles. Chacun permet de définir précisément les fonctionnalités souhaitées et leur priorité. L'acceptation par toutes les parties prenantes d'un prototype fonctionnel, sur tout ou partie du périmètre, permet de passer à la phase Conception et construction. L'objectif de cette phase est de développer un logiciel testé, par des cycles successifs de développement/acceptation par les utilisateurs.

Les rôles spécifiques à DSDM et la dynamique d'animation sont présentés au chapitre 5.

2.8.3 Le modèle XP

La méthode XP, focalisée sur la partie programmation du projet, propose un modèle itératif avec une structure à deux niveaux : d'abord des itérations de livraison (*release*), puis des itérations de développement (figure 2.21). Les premières conduisent à livrer des fonctionnalités complètes pour le client, les secondes portent sur des éléments plus fins appelés scénarios qui contribuent à la définition d'une fonctionnalité.

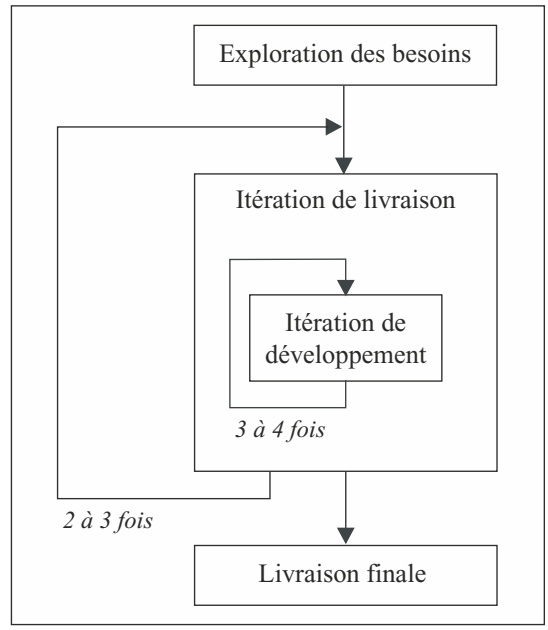


Figure 2.21 — Cycle de vie XP.

Après une phase initiale d'Exploration des besoins, un plan de livraison est défini avec le client. Chaque livraison, d'une durée de quelques mois, se termine par la fourniture d'une version opérationnelle du logiciel. Une itération de livraison est découpée en plusieurs itérations de développement de courte durée (deux semaines à un mois), chacune donnant lieu à la livraison d'une ou plusieurs fonctionnalités pouvant être testées, voire intégrées dans une version en cours.

De façon plus détaillée, chaque itération de développement (figure 2.22) commence par l'écriture de cas d'utilisation ou scénarios (*user stories*), c'est-à-dire des fonctions simples, concrètement décrites, avec les exigences associées, qui participent à la définition d'une fonctionnalité plus globale. Utilisateurs et développeurs déterminent ensemble ce qui doit être développé dans la prochaine itération. Une fonctionnalité est ainsi découpée en plusieurs tâches. Les plans de test sont écrits, les développeurs sont répartis en binôme (comme nous le développons au chapitre 5), ils codent les tâches qui leur sont affectées, puis effectuent avec les utilisateurs des tests d'acceptation. En cas d'échec, on revoit les scénarios et on reprend la boucle. Sinon, on continue jusqu'à avoir développé tous les scénarios retenus. Une version livrable est alors arrêtée et mise à disposition, ainsi que la documentation.

Les approches d'estimation, les rôles et le pilotage XP seront abordés respectivement aux chapitres 3, 5 et 6.

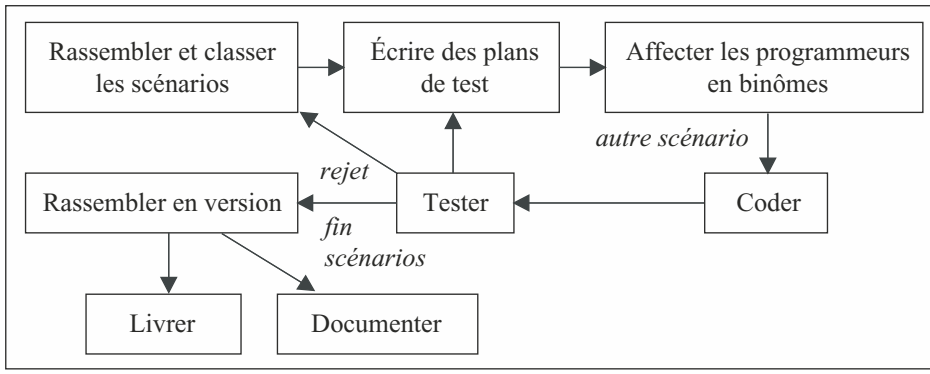


Figure 2.22 — Modèle XP d'une itération de développement.

2.8.4 Le modèle SCRUM

SCRUM emprunte au vocabulaire du jeu le qualificatif des trois phases du cycle préconisé (figure 2.23).

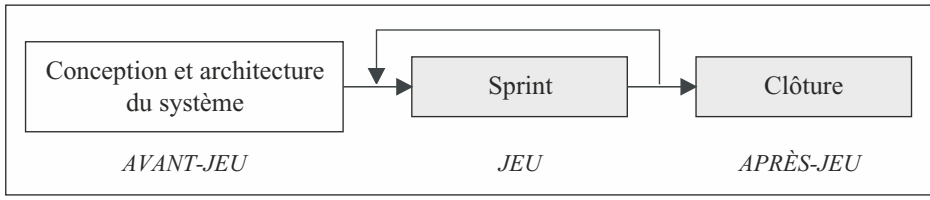


Figure 2.23 — Modèle SCRUM de cycle de vie du projet.

La phase d'Avant-jeu (*pre-game*), Conception et architecture du système, se déroule de façon structurée, en général linéaire, et permet de déterminer le périmètre, la base du contenu du produit à développer et une analyse de haut niveau.

La phase de Jeu (*game*) est itérative et qualifiée d'empirique, dans la mesure où le travail effectué ne fait pas l'objet d'une planification. Une itération, dont la durée oscille entre une et quatre semaines, est appelée un Sprint, en référence à ces poussées rapides et fortes que les joueurs de rugby peuvent effectuer sur le terrain. Un Sprint est découpé en trois sous-phases :

- Développement (*develop*) : il s'agit de déterminer l'objectif visé au terme de l'itération, de le répartir en « paquets » de fonctions élémentaires, de développer et tester chaque paquet.
- Emballage (*wrap*) : on referme les « paquets » et on les assemble pour faire une version exécutable.

- Revue (*review*) : une revue élargie permet de faire le point sur les problèmes et l'avancement.
- Ajustement (*adjust*) : ajuster le travail restant.

La phase d'Après-Jeu (*postgame*), Clôture, vise à livrer un produit complet et documenté. Comme dans la première phase, on peut en planifier les tâches et les dérouler de façon linéaire.

Les rôles, l'animation et le suivi d'équipe, y compris la notion de « mêlée » (*scrum*) sont décrites au chapitre 5.

2.9 LE DÉCOUPAGE STRUCTUREL DANS LE CAS DES MÉTHODES AGILES

Lorsque l'on utilise un modèle itératif et incrémental, notamment dans le cadre d'une méthode agile, le découpage en composants affectés à des itérations différentes est particulièrement important. En effet, d'une part, chaque itération doit pouvoir fournir un logiciel utilisable, d'autre part il est souhaitable que toute livraison produise sur le client un effet démonstratif. Ainsi, la première itération doit convaincre de la pertinence de la conception, même si le système est rudimentaire, et inciter à poursuivre. Les itérations suivantes doivent montrer une progression effective vers un produit stabilisé.

Par exemple, le cycle de vie d'un projet peut être structuré en trois livraisons : la première permet de livrer les fonctionnalités de base, la deuxième l'ensemble des fonctionnalités souhaitées et la troisième d'améliorer les fonctionnalités.

La détermination de l'ordre de livraison des fonctionnalités est parfois une question délicate nécessitant de mettre en balance différents arguments.

Ainsi, le plus confortable pour les développeurs est de commencer par les référentiels et de s'attaquer ensuite au développement des processus utilisant ces structures de données. Cependant, ce choix peut s'avérer peu mobilisateur pour les clients qui s'attendent à voir rapidement des fonctions qu'ils espèrent créatrices de valeur ou pouvant répondre à un besoin urgent. De façon analogue, il est préférable en dehors de toute contrainte, de sélectionner d'abord les fonctionnalités les plus simples pour aborder ensuite les plus complexes, ou les cas standard avant les cas particuliers. Mais ces choix, qui peuvent d'ailleurs être incompatibles, risquent de ne pas emporter l'adhésion des utilisateurs, peu convaincus par l'apport annoncé du nouveau système. De plus, si les utilisateurs ne partagent pas les mêmes objectifs, il est souvent conseillé de commencer par ce qui est le plus consensuel pour bâtir des relations positives de travail en commun, avant de s'attaquer à des fonctions susceptibles de générer des positions conflictuelles.

On peut parfois utiliser la notion de « strate ». Par exemple[1], dans un projet de développement d'un nouveau téléphone mobile chez Motorola, les strates correspondaient aux différents aspects du système (liaison radio, contrôle d'appel, liaison avec la base…) et chaque version du produit implémentait à des degrés divers des éléments de chacune des strates, ce qui a permis d'avoir des retours des commanditaires beaucoup plus tôt dans le déroulement du projet.

La question du choix d'un modèle de cycle de vie est traitée au chapitre 6, dans le cadre plus général du contrôle des risques. On peut cependant souligner un mouvement de rapprochement depuis quelques années, visant à conjuguer un cycle classique global avec un cycle issu d'une méthode agile pour une partie du projet[2]. Ainsi, DSDM expose des rapprochements avec XP, certains auteurs font de même entre XP et RUP[3].

Pour conclure, il convient de poser la question : quel découpage choisir pour un projet donné ? En fait, le choix d'un modèle de développement s'appuie sur l'analyse des caractéristiques du projet, et notamment l'analyse des risques. De plus, il ne suffit pas de sélectionner un modèle : il faut également l'instancier, c'est-à-dire en construire une version concrète adaptée au projet à mener, et le conjuguer avec le découpage structurel et l'organisation du projet. Cela constitue ce que l'on appelle l'établissement d'une stratégie. Cet aspect est traité au chapitre 6, qui couvre l'évaluation des risques et la prise en compte du diagnostic risque dans un plan de management de projet.

1. J. A Stark et R. Crocker, « Trends in Software Process : The PSP and Agile Methods », *IEEE Software* 20, 3, 89-91, 2003.
2. Voir par exemple, les cas de ABB, Ericsson et Vodafone étudiés par D. Karlstrom et P. Runeson, « Combining Agile Methods with Stage-Gate Project Management », *IEEE Software* 22, 3, 43-49, 2005.
3. Voir par exemple P. Y. Cloux, *RUP, XP, Architectures et outils*, Dunod, 2003.

3

L'estimation des charges

3.1 LA CHARGE ET LA DURÉE

Avant de présenter les méthodes permettant d'estimer la charge d'un projet, nous allons préciser certaines notions, puis nous situerons les différents niveaux d'estimation, auxquels sont attachés des objectifs spécifiques.

Il convient de bien distinguer charge et durée.

La *charge* représente une quantité de travail nécessaire, indépendamment du nombre de personnes qui vont réaliser ce travail ; elle permet notamment d'obtenir un coût prévisionnel. Elle s'exprime en jour-personne, mois-personne ou année-personne. Un mois-personne représente l'équivalent du travail d'une personne pendant un mois, en général 20 jours.

Ainsi, un projet de 60 mois-personne représente l'équivalent du travail d'une personne pendant 60 mois. Si on évalue le coût complet du mois-personne à 50 K€ en moyenne, le projet sera estimé à 3 M€.

On mesure la taille des projets à leur charge. Les ordres de grandeur généralement retenus sont les suivants :

- si la charge est inférieure à 6 mois-personne, c'est un très petit projet ;
- si la charge est comprise entre 6 et 12 mois-personne, c'est un petit projet ;
- si la charge est comprise entre 12 et 30 mois-personne, c'est un projet moyen ;
- si la charge est comprise entre 30 et 100 mois-personne, c'est un grand projet ;
- si la charge est supérieure à 100 mois-personne, c'est un très grand projet, souvent mesuré en années-personne.

La *durée* dépend du nombre de personnes. Soixante mois-personne, ce peut être aussi bien une personne pendant cinq ans, cinq personnes pendant un an, dix personnes pendant six mois ou soixante personnes pendant un mois. Cependant, intuitivement on sent bien que l'on ne peut pas faire n'importe quoi. Certaines méthodes d'estimation proposent de respecter un rapport charge/durée. Nous le précisons plus loin.

3.2 LES DIFFÉRENTS BESOINS D'ESTIMATION

Les besoins d'estimation des charges se situent à différents *niveaux*. Prenons l'exemple du domaine Gestion de production par lots (figure 3.1).

Niveau	Exemple
Projet	Projet Gestion de production
Phase	Étude préalable du projet Gestion de production Étude détaillée du sous-projet Ordonnancement
Étape/Activité	Étape observation de l'étude préalable Gestion de production
Tâche	Spécification de la fonction Réservation en stock Écriture du programme Plan de chargement des machines

Figure 3.1 — Exemples de niveaux d'estimation.

Un niveau correspond à un objectif. L'exigence de précision en est dépendante. Souvent le but visé par une évaluation est lié au stade d'avancement du projet. Pour souci de simplification, nous présentons les besoins d'estimation tels qu'on les rencontre aux différents stades du cycle de vie classique détaillé au chapitre précédent.

Au *niveau projet*, on veut estimer la charge du projet complet. L'ordre de grandeur est le mois-personne ou l'année-personne. Il y a plusieurs raisons pour faire une telle estimation :

- déterminer une enveloppe budgétaire ;
- voir ce que « pèse » le projet en termes d'effort ;
- faire une estimation de la rentabilité de l'investissement ;
- évaluer une durée vraisemblable du projet.

Au *niveau phase*, on cherche la charge d'une étape spécifique. L'ordre de grandeur est le mois ou la semaine-personne. Les objectifs d'estimation à ce stade sont les suivants :

- *Ajuster le découpage*. Si la charge est importante, on va chercher à la diviser en deux lots, gérés comme deux sous-projets, qui pourront éventuellement être livrés à des dates différentes.
- *Sous-traiter*. Fournisseur et client font chacun une estimation. Celle-ci est indispensable au premier pour ne pas s'engager à travailler à perte. Mais de son côté, le client peut recevoir en réponse à un appel d'offres des propositions dont les prix varient de un à dix. Retenir le « moins disant » peut constituer un risque réel. Il y a, à ce sujet, le cas célèbre d'un projet de l'US Air Force. Bien que l'estimation initiale fût de 1 500 K$, une société de services remporta le marché à 400 K$. Après développement d'un système incomplet, le fournisseur a négocié plusieurs avenants. Au final, le coût total fut de 3 700 K$, soit près de dix fois le montant du marché initial et plus du double de l'estimation par le client… Ce surcoût s'est accompagné de retards considérables sur le calendrier prévu.
- *Prévoir des délais* pour planifier l'ordonnancement des étapes, ainsi que des opérations complémentaires comme la formation des utilisateurs.
- *Prévoir des ressources*, pour planifier l'affectation d'intervenants (internes ou externes) sur le projet.

Au *niveau étape* (*ou activité*), l'estimation est nécessaire pour :

- faire une planification précise ;
- annoncer un calendrier de remise des différents résultats intermédiaires (entre les livrables de deux phases) ;
- prévoir et effectuer un suivi du projet ou sous-projet, pour surveiller les écarts ;
- prévoir l'affectation des ressources, car il peut y avoir une montée en charge ou une diminution d'une étape à l'autre.

Au *niveau tâche*, il s'agit d'évaluer chacune des tâches qui font généralement l'objet d'une affectation individuelle, par exemple la tâche d'élaboration du jeu d'essai Planification de production. L'ordre de grandeur est le jour-personne.

L'estimation permet une planification au niveau le plus fin, qui est indispensable pour le suivi du travail de l'équipe.

Entre le niveau du projet et celui de la tâche, la visibilité est croissante, ce qui fournit des éléments de plus en plus précis pour l'évaluation. C'est pourquoi on utilisera des techniques différentes, selon le niveau où se situe le besoin d'estimation.

3.3 LES DIFFÉRENTES MÉTHODES D'ESTIMATION

3.3.1 Les non-méthodes

Citons tout d'abord des techniques parfois utilisées, que l'on peut cependant ranger dans la catégorie des « non-méthodes ».

La méthode dite de *Parkinson* n'a rien à voir avec la maladie du même nom : elle fait référence à un auteur, historien et militaire, qui a traduit des observations récurrentes d'un dysfonctionnement administratif sous forme d'une « loi »[1], stipulant que « le travail se dilate jusqu'à remplir le temps disponible ».

Supposons que, pour réaliser une tâche, on dispose d'une personne pendant trois mois. En l'absence d'estimation, la vitesse d'avancement est, dans la plupart des cas, spontanément ajustée à la durée de disponibilité des individus. Plus la taille de l'équipe est importante, plus cette « loi » se trouve vérifiée. L'estimation des charges est difficile, car elle ne résulte pas de la simple application d'une formule. Pour cette raison, certaines entreprises ne font guère d'estimation de leurs projets système d'information, considérant que « cela ne tombe jamais juste ». Cependant, la « loi » de Parkinson nous indique que, sans évaluation, on consomme en général beaucoup plus de temps, si l'on compare avec des projets analogues, et le travail s'achève difficilement.

La méthode du *marché* consiste à considérer que la charge correspond au prix à proposer pour remporter l'appel d'offres. L'exemple du projet de l'US Air Force cité plus haut montre les limites de cette méthode. Il ne s'agit pas de nier les contraintes commerciales dans un marché concurrentiel. Une société de services peut répondre à perte à un appel d'offres, dans l'espoir d'obtenir ultérieurement d'autres contrats dont la rentabilité compenserait la perte sur le premier. Cependant, dans un tel cas, il est indispensable de faire parallèlement une évaluation réaliste, pour pouvoir planifier et organiser le projet sur des bases solides.

3.3.2 Les méthodes

Il existe cinq familles de méthodes d'estimation, qui ne sont pas forcément concurrentes, mais peuvent être utilisées simultanément, ou à des moments différents du cycle de vie du projet : le jugement d'expert, l'estimation par analogie, l'estimation ascendante, l'estimation paramétrique et l'estimation probabiliste.

1. C. N. Parkinson en a publié la première version en 1955 dans le journal *The Economist* : http://alpha1.montclair.edu/~lebelp/ParkinsonsLaw.pdf

1. Le jugement d'expert

Certaines estimations sont basées sur le *jugement d'expert*. Les experts sont, dans le cas des projets système d'information, des consultants ou d'autres chefs de projet expérimentés, qui font appel à leurs connaissances explicites ou implicites, acquises au cours de leur vécu et leurs observations. La méthode Delphi, que nous présentons au paragraphe 3.4, a formalisé un processus de mise en commun d'expertise.

2. L'estimation par analogie

Les estimations s'appuient sur les consommations de projets similaires pour lesquels on a conservé une mémoire. Ces projets peuvent être internes ou externes à l'entreprise. Par exemple, on veut mettre en place une nouvelle Gestion des clients dans une banque A. On sait qu'un projet analogue mené par une banque B a coûté 500 mois-personne et un projet analogue dans une compagnie d'assurance a consommé 350 mois-personne. Ces deux estimations permettent d'avoir un ordre de grandeur. On va situer le projet à estimer par rapport aux deux projets connus, en déterminant si les caractéristiques du projet le rapprochent davantage de la banque B ou de la compagnie C.

On peut aussi se servir d'une référence interne, sur un autre domaine. On a déjà fait un projet Gestion de carte bancaire, qui a coûté 700 mois-personne, et un projet Gestion des crédits qui a coûté 1800 mois-personne. On estime que le domaine Dépôt-épargne est d'une complexité qui le situe entre les deux. On retiendra un chiffre d'environ 1200 mois-personne.

L'estimation basée sur l'analogie doit prendre en compte les facteurs d'environnement. Ainsi, une Gestion des clients dans une petite banque d'affaires coûtera en principe moins cher que dans une grande banque à réseau, notamment à cause des processus de décision et des procédures de qualification.

Nous détaillons au paragraphe 3.5 deux méthodes pouvant se rattacher à cette famille : la méthode de *répartition proportionnelle*, basée sur l'hypothèse que les différentes phases d'un cycle sont liés par une relation de proportionnalité ; la méthode des *ratios* qui s'appuie sur l'observation de pourcentages stables entre différentes activités.

3. L'estimation ascendante

Le principe de cette famille de méthodes est d'estimer la charge d'éléments de travail (lot de travail, activité, tâche…) et de totaliser ces estimations élémentaires. Elle requiert donc d'avoir réalisé une structure de découpage du projet suffisamment fine. Nous décrivons au paragraphe 3.6 la méthode d'évaluation analytique, qui est fréquemment utilisée.

4. L'estimation paramétrique

Certaines méthodes utilisent un modèle mathématique, issu d'analyses statistiques sur la relation entre un ou plusieurs paramètres, qui sont des variables caractéristiques du projet, et la charge prévisible.

L'estimation paramétrique est analogue à la technique des *unités d'œuvre* utilisée en comptabilité analytique pour répartir des coûts généraux de façon simple, facile à calculer, uniforme (la même règle s'applique à tous), et pas trop fausse dans le sens où le résultat n'est pas trop éloigné de celui que l'on obtiendrait si l'on se livrait à un calcul direct et détaillé. Par exemple, les coûts d'entretien d'un immeuble sont répartis entre les différents Départements proportionnellement à la surface occupée ; l'unité d'œuvre est le mètre carré, la valorisation se fait à partir d'un coût standard du mètre carré. De même les coûts d'un Service approvisionnement qui travaille pour toute l'entreprise sont redistribués dans chaque service proportionnellement au nombre de commandes passées ; l'unité d'œuvre est la commande, pour laquelle on détermine un coût standard.

Pour utiliser cette technique, il faut déterminer une ou plusieurs unités d'œuvre pertinentes, affecter un coût unitaire standard à chaque unité d'œuvre (par exemple, en divisant un coût global précédemment mesuré par le nombre total d'unités d'œuvre) et pouvoir dénombrer les unités d'œuvre. Les méthodes basées sur un modèle proposent donc de repérer certains éléments simples, à partir desquels on calculera l'effort nécessaire. Nous allons présenter deux méthodes, diversement utilisées : le modèle Cocomo au paragraphe 3.7 et la méthode des points de fonction au paragraphe 3.8.

Les méthodes d'estimation paramétriques s'inscrivent, totalement ou partiellement, dans un schéma général (figure 3.2) comportant les étapes suivantes :

- faire une estimation de la taille du projet à l'aide d'une unité de mesure ;
- convertir la taille en charge ;
- éventuellement ajuster la taille ou la charge brute en raison de certaines particularités du projet.

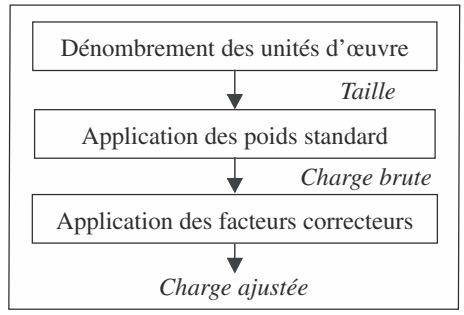


Figure 3.2 — Schéma général de l'estimation des charges à base de modèles.

5. L'estimation probabiliste

Nous avons regroupé dans cette famille des approches utilisées conjointement avec d'autres méthodes lorsque l'incertitude est élevée et que l'on cherche à réduire le risque d'une évaluation erronée. Nous présentons au paragraphe 3.9 l'estimation à trois points et la méthode de Monte-Carlo.

3.4 LE JUGEMENT D'EXPERT : LA MÉTHODE DELPHI

Les méthodes basées sur le jugement d'expert peuvent être utilisées à différents moments et à différents niveaux (pour un projet entier ou pour une activité). On y recourt généralement lorsque l'on dispose de peu d'éléments. Par exemple lors d'une étude de faisabilité ou d'opportunité, le jugement d'expert peut apporter une aide à la décision de poursuivre ou non le projet.

La méthode Delphi fut élaborée en 1948 par la Rand Corporation[1], pour améliorer les prévisions économiques. C'est un domaine où l'avenir est généralement incertain ; les avis des experts, bien que parfois divergents, apportent des éclairages précieux. Le nom Delphi fait référence à la pythie de Delphes, à qui l'on posait des questions et qui, en rendant son oracle, se faisait l'interprète de la réponse des dieux. Sa parole était donc comme celle des experts empreinte d'une grande autorité.

Cette méthode propose une démarche pour mettre en commun des jugements d'experts. Elle est utilisée dans différents domaines, dont l'estimation des charges d'un projet système d'information. La démarche est la suivante :

- Dans un premier temps, chaque expert propose une estimation en utilisant sa propre expérience.
- Dans un second temps, tous les jugements sont rendus publics, mais restent anonymes. Chaque expert peut alors modifier sa propre estimation ou la confirmer. Chacun doit bien sûr considérer que tous les autres sont également des experts, ce qui le conduit à se poser des questions si ses propres estimations sont très éloignées de celles des autres.
- Dans un troisième temps, les nouvelles estimations sont dévoilées et chacun peut justifier son propre jugement.
- Dans un dernier temps, chacun propose une révision de son estimation. Normalement, sans que l'on arrive toujours à un consensus, les résultats

1. Une version est téléchargeable sur le site de RAND : http://www.rand.org/pubs/research_memoranda/RM5888/

sont moins divergents qu'au premier tour. Dans tous les cas, cela permet de construire en commun une première analyse du projet.

Cette méthode, utilisée notamment dans les sociétés de services, peut comporter différentes variantes, selon le mode de communication entre experts (oral, écrit, rapproché ou à distance…), le nombre de cycles de prévision et la conduite de la discussion (avec ou sans animateur extérieur…). Bien que fortement dépendante de l'expérience des experts, elle permet de dépasser la dimension strictement individuelle en organisant la confrontation collective des estimations. On considère, par ailleurs, qu'elle évite que certains exercent une trop grande influence sur les autres, ce qui biaiserait le résultat.

3.5 L'ESTIMATION PAR ANALOGIE

3.5.1 La méthode de répartition proportionnelle

La méthode de répartition proportionnelle est utilisée au niveau du projet ou d'un ensemble de phases. Elle est basée sur des observations de projets antérieurs : la charge de chaque phase d'un projet devrait correspondre, avec un certain intervalle de tolérance, à un pourcentage de la charge globale. Il existe plusieurs versions qui donnent des pourcentages indicatifs, chacune s'appuie sur un cycle de vie de référence. Cette méthode est utilisée de trois façons :

- On a fait une estimation de la charge globale du projet, que l'on cherche à répartir dans le temps.
- On a évalué l'une des phases, au moyen d'une autre méthode, et l'on veut en déduire la charge des autres étapes.
- On est en cours de déroulement du projet, on a observé le temps déjà consommé et on veut estimer celui des phases à venir.

L'exemple de répartition donné à la figure 3.3 fait référence au cycle RUP (§ 2.7.3) et distingue la répartition en charge et la répartition en durée. Notons qu'il est toutefois rare d'évaluer la charge de la phase de mise en œuvre au moyen d'un ratio. En effet, celle-ci comprend des activités diverses dont une partie est décrite au paragraphe traitant de la gestion du changement (5.4). Il faut donc comprendre ici la transition comme réduite à son aspect informatique de préparation de l'exploitation.

Une autre répartition classique est donnée à la figure 3.4. Ces ratios sont issus d'expériences positives et négatives de projet. Ils doivent être considérés en partie comme des recommandations et en partie comme des règles. Ainsi, une étude préalable ne doit pas consommer plus de 10 % des ressources d'un projet ; ou bien on doit s'attendre à consommer deux fois plus en réalisation qu'en étude détaillée.

Phase	Charge	Durée
Opportunité	5 %	10 %
Élaboration	20 %	30 %
Construction	65 %	50 %
Transition	10 %	10 %

Figure 3.3 — Répartition d'après le cycle RUP.

Il s'agit donc moins de les appliquer de façon aveugle que de les prendre comme base, en comprenant les relations qui unissent les différentes phases.

Phase	Ratio
Étude préalable	10 % du total du projet (hors mise en œuvre)
Étude détaillée	20 à 30 % du total du projet
Étude technique	5 à 15 % de la charge de réalisation
Réalisation	deux fois la charge d'étude détaillée

Figure 3.4 — Répartition proportionnelle de la charge entre les phases du cycle classique.

Pour ce qui est de la charge de l'*étude préalable*, on utilise également une répartition proportionnelle entre les étapes (figure 3.5).

Étapes	Pourcentage de la charge d'étude préalable
Observation	30 à 40 %
Conception/Organisation	50 à 60 %
Appréciation	10 %

Figure 3.5 — Répartition proportionnelle entre les phases.

Par certains côtés, l'*étude détaillée* est la phase la plus difficile à évaluer, car il n'y a pas de relation constante entre étude préalable et étude détaillée. Deux facteurs principaux sont sources de variation : la couverture et la maille.

- La *couverture* est la partie du domaine déjà étudiée par le sous-ensemble représentatif. Parfois, le nombre de variantes de traitement restant à étudier est élevé. À l'inverse, dans le cas de petits projets sans variante de

procédure, étude préalable et étude détaillée sont confondues sans surcharge pour l'étude préalable.

L'utilisation des principes de l'analyse de la valeur en conception conduit à limiter la variété des procédures élaborées et à favoriser la réutilisation d'éléments de procédures déjà conçues ; ceci réduit la charge à la fois de l'étude détaillée et de la réalisation.

- La *maille d'étude* mesure la précision de la description. Dans certains cas, le modèle conceptuel représentant les entités est presque complet, dans d'autres cas la structuration statique doit encore être affinée. De même les traitements peuvent-ils n'avoir été écrits que dans les grandes lignes.

La charge de l'*étude technique* est liée à la charge de réalisation, mais peut être augmentée par exemple en cas de particularité ou de nouveauté technique.

La charge de la phase de *réalisation* est liée à celle de l'étude détaillée. Le ratio Charge d'étude détaillée/charge de réalisation est basé sur l'idée qu'il y a une relation entre l'effort pour produire un cahier des charges et celui pour le réaliser. Le ratio est parfois utilisé à l'envers. On fait une estimation de la charge de réalisation en utilisant une autre méthode et on divise par deux pour avoir la charge d'étude détaillée correspondante.

3.5.2 La méthode des ratios

La méthode des ratios est également une forme d'estimation par analogie, c'est une variante de la répartition proportionnelle qui vise la relation entre deux activités. Elle est souvent utilisée pour estimer des *charges complémentaires* au développement de l'application (figure 3.6). Il s'agit par exemple de l'encadrement du projet, de la recette et de l'élaboration de la documentation utilisateur.

Activité	Ratio
Encadrement du projet : – à la phase de réalisation, – aux autres phases.	 20 % de la charge de réalisation 10 % de la charge de la phase
Recette	20 % de la charge de réalisation
Documentation utilisateur	5 % de la charge de réalisation

Figure 3.6 — Répartition proportionnelle des charges complémentaires.

Pour ce qui est de la *charge d'encadrement*, le ratio est plus élevé à la phase de réalisation. En effet, classiquement, l'équipe de projet est la plus nombreuse à ce stade. La coordination assurée par le responsable augmente de façon significative.

La *charge de recette* dépend de la taille du logiciel. En effet, la recette est la procédure par laquelle le client maître d'ouvrage et le fournisseur maître d'œuvre constatent ensemble que le produit (logiciel) livré correspond aux spécifications.

Les utilisateurs sont souvent impliqués dans la procédure, ce qui prépare le démarrage du futur système. Les tâches de recette comprennent :

- l'élaboration des *jeux d'essai de recette*, qui doivent permettre de vérifier que le système fonctionne et réagit correctement ; on bâtit de préférence des cas simulant le traitement entier d'un événement de gestion. Les jeux d'essai de recette sont destinés à faire des tests de fonctionnalité et non de volume. Ces derniers ont pour objectif de vérifier que l'application supporte bien un volume important de données et de transactions simultanées ; ils ne sont pas toujours nécessaires ;
- la préparation de l'*environnement de recette* : il faut extraire des parties de bases de données réelles pour tous les objets qui ne sont pas créés par l'application ;
- l'*exécution* de la recette : il faut exécuter successivement les traitements par lots et pointer les résultats ; les traitements transactionnels sont testés en présence des deux parties, client et fournisseur. Un procès-verbal est dressé.

La *charge d'élaboration de la documentation* utilisateur est proportionnelle à la taille du logiciel. La plupart des logiciels développés ont des aides utilisateurs intégrées pour la partie transactionnelle. Il faut néanmoins généralement livrer un guide sur papier pour l'utilisateur de l'application.

La charge de la phase de *mise en œuvre* ne relève pas d'une estimation standard. La mise en œuvre est le passage de l'ancien système au nouveau. L'effort est souvent proportionnel au nombre et à la complexité des programmes écrits, mais les évolutions organisationnelles (rôles, responsabilités, apprentissages…) sont déterminantes pour évaluer la charge. Si le futur système est implanté sur plusieurs sites, il faut prévoir une charge additionnelle variant avec le nombre de sites.

On applique parfois un ratio de 40 % sur la charge de réalisation, ce qui indique une relation de proportionnalité avec la taille du logiciel. Mais cette estimation est en général affinée en fonction du nombre de sites et de leurs caractéristiques techniques, organisationnelles et humaines.

De plus, elle doit être complétée par les problèmes de *basculement*, c'est-à-dire de passage de l'ancien système au nouveau. La charge peut notamment varier en fonction de deux critères : la complexité et le scénario.

La *complexité* du passage situation avant/situation après se mesure par le nombre d'anciens fichiers (entités logiques) touchés et par le degré de restructuration des informations apportée par le nouveau système. Il faut en effet assurer ce qu'on appelle la « reprise » ou la « conversion », c'est-à-dire reprendre le contenu des anciens fichiers et les convertir selon les nouvelles structures de données.

Le *scénario* retenu peut être ou non progressif. Si l'on bascule d'un coup, il y a moins d'interfaces pour gérer des situations intermédiaires. Le risque est plus grand et la charge plus faible.

Pour les grands systèmes d'information, la mise en œuvre est généralement un sous-projet à part entière, dans la mesure où l'on doit développer des programmes spécifiques de reprise et de bascule. De plus, le scénario de mise en œuvre peut concerner plusieurs projets.

3.6 L'ESTIMATION ASCENDANTE : L'ÉVALUATION ANALYTIQUE

La méthode d'évaluation analytique est souvent utilisée pour évaluer la charge de réalisation, lorsque l'on a une visibilité sur les composants à développer. Elle s'appuie sur une typologie des types de composants, classés par nature et souvent par degré de difficulté, spécifiques de l'environnement : développement classique, paramétrage de progiciel, développement d'un intranet… Les poids (en jours-personne) des types de composants peuvent varier en fonction de la productivité de l'équipe.

Le tableau de la figure 3.7 donne l'exemple d'une typologie classique, séparant programmes transactionnels et programmes par lots. Les poids indiqués, en jours-personne, correspondent à des temps moyens.

Type de transaction	Facile	Moyen	Difficile
Menu	0,25	0,5	1
Consultation	1	2,5	4
Mise à jour	1,5	3	5
Édition en temps réel	1	2	4

Type de programme par lots	Facile	Moyen	Difficile
Extraction	0,5	1	1,5
Mise à jour	2	3	5
Édition	1,5	2,5	4

Figure 3.7 — Exemple de typologie pour l'évaluation analytique

Ce type de méthode sert notamment, pour répondre à un appel d'offres, à évaluer le travail nécessaire à partir d'un cahier des charges. On fait une estimation du nombre de composants à réaliser en les ventilant par nature.

La charge obtenue, appelée *charge de réalisation*, couvre la programmation, l'élaboration de jeux d'essai de test et la mise au point des programmes. Pour obtenir la charge complète, on rajoute en général :

- pour les tests d'intégration : 10 % de la charge de réalisation.
- pour l'encadrement : 20 % de la charge de réalisation.

Nous donnons à la figure 3.8 un exemple d'utilisation de la méthode d'évaluation analytique, avec des composants d'une application intranet Notes/Domino.

Complexité	**Facile**			**Moyen**			**Difficile**			**Total**
Type composant	**N**	**P**	**C**	**N**	**P**	**C**	**N**	**P**	**C**	
Vue	8	0,25	2	20	0,5	10	12	1	12	24
Masque	9	0,5	4,5	6	1	6	10	2	20	30,5
Navigateur web				12	0,5	6				6
Agents							20	0,5	10	10
Accès bases données externes				4	0,25	1				1
Programme lotusScript	6	0,5	3	3	2,5	7,5	5	3	15	25,5
Reprise données							1	9	9	9
Total Réalisation										106
Intégration								10 %		10,6
Encadrement								20 %		21,2
TOTAL PROJET										138

Figure 3.8 — Exemple d'évaluation analytique
N = nombre, P = poids C = charge

3.7 L'ESTIMATION PARAMÉTRIQUE : LE MODÈLE COCOMO

Le modèle Cocomo (*Constructive Cost Model*, modèle de construction des coûts) a été proposé en 1981 par B.W. Boehm [Boehm, 1984]. Ce modèle repose sur deux hypothèses :

- d'une part, un informaticien chevronné sait plus facilement donner une évaluation de la taille du logiciel à développer que faire une estimation du travail nécessaire ;

- d'autre part, il faut toujours le même effort pour écrire un nombre donné de lignes de programme, quel que soit le langage de troisième génération employé.

Ces deux hypothèses ont conduit B.W. Boehm à calculer des coefficients de corrélation entre la taille du logiciel et la charge consommée à partir d'observations sur une centaine de projets déjà réalisés.

Le paramètre utilisé par ce modèle est *l'instruction source*. Il faut donc disposer du nombre présumé de lignes de programme en langage source, en dehors d'éventuels commentaires.

Le modèle permet d'obtenir la *charge de la réalisation* en mois-personne, ainsi que le *délai* « normal », recommandé si on ne veut pas prendre de risques supplémentaires. Cela nous indique la taille moyenne de l'équipe qui est égale à charge/délai.

Les formules de calcul de la charge et du délai sont les suivantes :

$$\text{Charge en mois-personne} = a\ (\text{kisl})^b$$

$$\text{Délai normal en mois} = c\ (\text{charge en mois-personne})^d$$

avec kisl = nombre de milliers d'instructions sources livrées, c'est-à-dire le nombre de milliers de lignes de programme source testées.

Les paramètres a, b, c et d prennent des valeurs différentes selon la catégorie de projet. En effet, l'analyse statistique a conduit à distinguer trois types de projet : simple, moyen et complexe.

- Un projet est *simple* si le logiciel comporte moins de 50 000 instructions, si les spécifications sont stables et le développement effectué par une petite équipe.
- Un projet est *moyen* si le logiciel comporte entre 50 000 et 300 000 instructions.
- Un projet est *complexe* si le logiciel comporte plus de 300 000 instructions et si l'on prévoit une équipe nombreuse ; il s'applique souvent à un domaine nouveau.

Les valeurs des paramètres sont les suivantes (figure 3.9) :

Type projet	Charge en mois-personne	Délai en mois
Simple	Charge = 2,4 (kisl) ^{1,05}	D = 2,5 (Charge) ^{0,38}
Moyen	Charge = 3 (kisl) ^{1,12}	D = 2,5 (Charge) ^{0,35}
Complexe	Charge = 3,6 (kisl) ^{1,2}	D = 2,5 (Charge) ^{0,32}

Figure 3.9 — Formules du modèle Cocomo.

Soit par exemple un projet simple visant à développer un logiciel estimé à 40 000 instructions sources :

Charge $= 2{,}4 \times (40)^{1{,}05}$ = 116 mois-personne (arrondi)

Délai normal $= 2{,}5 \times (116)^{0{,}38}$ = 12 mois (arrondi)

Taille moyenne de l'équipe = 116 / 12 = 9 personnes.

La méthode Cocomo prend en compte des caractéristiques propres au projet pouvant modifier l'estimation de la charge. Ces caractéristiques[1] se traduisent par un coefficient de pondération appelé « facteur correcteur ». Les valeurs des coefficients données par B.W. Boehm et obtenues de façon statistique sur un échantillon limité et biaisé (les projets de sa propre société TRW) sont de moindre intérêt que la démarche de correction. Celle-ci répartit les facteurs selon quatre sources de risques (figure 3.10) : les exigences attendues du logiciel, les caractéristiques de l'environnement technique (matériel), les caractéristiques de l'équipe projet et l'environnement du projet lui-même.

Logiciel	**Personnel**
Fiabilité	Compétence des concepteurs
Base de données	Expérience des concepteurs
Complexité	Compétence des développeurs
Temps d'exécution	Connaissance de l'environnement technique
	Expérience du langage
Matériel	**Projet**
Taille mémoire	Utilisation d'une méthode
Stabilité de l'environnement	Utilisation d'un atelier de génie logiciel
Disponibilité de la machine de test	Contrainte de délai

Figure 3.10 — Facteurs correcteurs du modèle Cocomo.

La *fiabilité* est le degré de régularité exigé ; dans le cas de systèmes mettant en jeu la vie humaine (contrôle du trafic aérien) ou pouvant causer des pertes financières importantes (système temps réel de réservation), ce facteur aura une influence forte.

Le facteur *base de données* est mesuré par le ratio :

(volume de données gérées en octets) sur (taille du logiciel en lignes).

1. En anglais, on les appelle des *cost-drivers*, des générateurs de coût.

Ainsi, si on gère 100 kilo-octets pour 1 000 lignes de programme, le ratio sera 100. La méthode considère que l'influence du facteur est faible si le ratio est inférieur à 10, moyenne entre 10 et 100, forte entre 100 et 1 000 et très forte si le ratio est supérieur à 1 000.

La *complexité* est celle des algorithmes à développer.

Le *temps d'exécution* peut être crucial pour certains systèmes temps réel, comme le contrôle du trafic aérien ou le système de cotation des valeurs mobilières.

Le facteur *taille mémoire* signifie qu'il y aura besoin d'optimiser l'utilisation de la mémoire.

La *stabilité de l'environnement* est celle du logiciel de base.

La *contrainte de délai* se mesure par rapport au délai « normal ».

Chaque facteur peut prendre ses valeurs dans un intervalle spécifique, dont les valeurs résultent de l'analyse statistique des projets de référence. Ils modifient la charge brute de façon multiplicative :

Charge nette = Produit(valeurs des facteurs correcteurs) * Charge brute

La méthode Cocomo propose donc une démarche en cinq temps :

- estimation du nombre d'instructions sources c'est-à-dire en langage de programmation ;
- calcul de la charge dite « brute » ;
- sélection des facteurs correcteurs ;
- application des facteurs correcteurs à la charge brute pour obtenir la charge dite « nette » ;
- évaluation du délai normal.

Les facteurs correcteurs sont à utiliser avec précaution pour deux raisons : d'une part, avec les valeurs proposées par B.W. Boehm, l'estimation peut varier de 1 à 17, à cause de l'effet multiplicatif ; d'autre part, les valeurs proposées résultent d'un échantillon de projets spécifiques. B.W. Boehm propose que chaque entreprise constitue sa propre base d'expériences.

La méthode Cocomo a fait l'objet d'évolutions sous le nom de COCOMO II[1], rendues publiques à partir de 1997. Elle propose un affinement des paramètres de contexte et inclut des projets au stade de l'analyse, en s'appuyant notamment sur la méthode des points de fonction. Ce point est développé plus loin au paragraphe 3.8.

1. Ces évolutions se font principalement à l'université de Southern California, sous la direction de B.W. Boehm, qui y dirige un centre d'ingénierie logicielle. Beaucoup d'informations sont accessibles *via* internet, notamment à l'adresse : http://sunset.usc.edu/COCOMOII.

3.8 L'ESTIMATION PARAMÉTRIQUE : LA MÉTHODE DES POINTS DE FONCTION

Présentée pour la première fois par Alan Albrecht d'IBM en 1979, cette méthode a été expérimentée par beaucoup de grandes entreprises clientes d'IBM, avec quelques difficultés d'interprétation et de dénombrement des unités d'œuvre. Un guide de comptage, publié en 1984, a favorisé sa diffusion. Des associations d'utilisateurs se sont constituées : l'IFPUG *(International Function Point Users Group)* fédère des associations nationales, et diffuse des versions mises à jour du guide de comptage. En France la FFPUG *(French Function Point Users Group)* s'est constituée en 1992, regroupant principalement des grandes entreprises, des constructeurs de matériel informatique et des sociétés de services. Elle a organisé pendant plusieurs années des partages d'expérience et traduit les guides de l'IFPUG. En 1995, l'AFNOR a édicté une norme sur la mise en œuvre de la méthode[1].

Le principe de la méthode des points fonctionnels est de faire une estimation à partir d'une description externe du futur système, de ses « fonctions » ou composants « fonctionnels ». On identifie cinq types avec trois degrés de complexité. À chaque type et chaque degré est affecté un nombre de « points », ce qui permet de calculer, pour un projet donné, son poids en *points de fonction*. Différentes règles permettront de passer des points de fonction à une charge.

La méthode propose de faire un comptage des points au début du projet en faisant des hypothèses sur les fonctions du futur système et de refaire un comptage en fin de projet basé sur les fonctionnalités livrées. L'écart éventuel est appelé « changement d'envergure » du champ d'étude.

3.8.1 Les composants fonctionnels

Cinq types de composants fonctionnels servent d'unités d'œuvre. Deux sont relatifs à l'aspect statique d'un système d'information (GDI et GDE) et trois à l'aspect dynamique (ENT, SOR et INT).

Un **GDI** (Groupe de Données Interne) est un groupe de données que l'utilisateur perçoit comme logiquement liées. Il est créé et mis à jour à l'intérieur du domaine d'étude. Si l'on se réfère au formalisme entité/association d'une modélisation conceptuelle, on peut assimiler le GDI à une entité du modèle conceptuel ou à une association porteuse de propriétés, qui correspond à un objet de gestion.

Ainsi, une entité Date n'est pas un GDI. Un lien de composition entre deux entités conduit à retenir un seul GDI pour les deux entités. Si par exemple, dans le domaine Crédit, une entité *échéance à recouvrer* se décompose en plusieurs entités *composante d'échéance* (capital, intérêts, assurance.), on considère l'ensemble comme un seul GDI. De façon plus large, les entités reliées à d'autres par une

1. AFNOR XP Z67-160 : « Technologies de l'information - Mesure de la taille de logiciel - Guide opératoire de comptage des points de fonction ».

relation de cardinalité (1,1) sont susceptibles d'être regroupées en un seul GDI si elles traduisent un même concept de gestion.

Un **GDE** (Groupe de Données Externe) est un groupe de données que l'utilisateur perçoit comme logiquement liées. Le domaine d'étude ne fait que l'interroger. Il est créé et mis à jour par un autre domaine. Un GDE est donc obligatoirement un GDI dans un autre domaine.

Une **ENT** (Entrée) est une fonction élémentaire, significative pour l'utilisateur, qui permet d'introduire des données à l'intérieur du domaine. Ces données peuvent être des données spécifiques du domaine ou des paramètres de traitement. L'entrée des données déclenchera un traitement comprenant des mises à jour. C'est une fonction autonome qui laisse l'application dans un état de cohérence fonctionnelle. Par simplification, une entrée correspond à un écran de saisie ou à une réception de données provenant d'une autre application et provoquant une mise à jour. À chaque GDI doit correspondre au moins une entrée permettant sa mise à jour. Chaque entrée (ENT) permet de saisir un certain nombre de champs, qui sont des données élémentaires (DE). On compte une seule DE pour un champ répétitif.

Une **SOR** (Sortie) est une fonction élémentaire, significative pour l'utilisateur, qui envoie des données vers l'extérieur du domaine et qui n'effectue aucune mise à jour à l'intérieur du domaine. C'est une fonction autonome qui laisse l'application dans un état de cohérence fonctionnelle. Par simplification, une sortie correspond à la génération d'un état élémentaire, d'un écran de visualisation ou d'un message à destination d'une autre application, avec des données calculées ou dérivées, c'est-à-dire obtenues à partir d'autres données. Sur chaque sortie figurent un certain nombre de champs, qui sont des données élémentaires (DE). On compte une seule DE pour un champ répétitif. Si deux sorties ont la même logique de traitement et les mêmes DE (par exemple, sortie papier et sortie écran), on ne comptera qu'une seule SOR.

Une **INT** (Interrogation) est une fonction élémentaire, qui a pour résultat l'extraction de données. La demande d'interrogation peut être saisie ou provenir d'une autre application. Le résultat de l'interrogation ne contient aucune donnée calculée ou dérivée, c'est-à-dire obtenue à partir d'autres données. L'INT ne met à jour aucun GDI. Sur le résultat de l'interrogation figurent un certain nombre de champs, qui sont des données élémentaires (DE). On compte une seule DE pour un champ répétitif. Si deux interrogations ont la même logique de traitement et les mêmes DE (par exemple, résultat papier et résultat écran), on ne comptera qu'une seule INT.

3.8.2 La complexité et le nombre de points de fonction

Un GDI ou un GDE est composé de *données élémentaires* (DE), qui correspondent aux propriétés d'une entité conceptuelle ou logique. On compte une DE par champ, y compris pour les clés étrangères.

On peut parfois identifier plusieurs *sous-ensembles logiques de données* (SLD) à l'intérieur d'un GDI/GDE. Un SLD est un sous-groupe de données élémentaires reconnaissable par l'utilisateur. Dans un modèle de classes UML, un SLD peut être assimilé à une entité spécialisée : on compte un SLD pour l'entité générale et un SLD par entité spécialisée.

La complexité d'un **GDI** ou d'un **GDE** est fonction du nombre de DE et du nombre de SLD, selon la figure 3.11.

Une ENT utilise, en lecture ou mise à jour, différents groupes logiques de données, internes ou externes (GDI ou GDE), que l'on appelle de façon générale des GDR, *groupes de données référencées*. Une SOR ou une INT utilise, en lecture uniquement, différents GDR.

La complexité d'une **ENT**, d'une **SOR** ou d'une **INT** est fonction du nombre de DE et du nombre de GDR impliqués, selon le tableau de la figure 3.11.

GDI ou GDE		**1 à 19 DE**	**20 à 50 DE**	**51 DE ou plus**
	1 SLD	Faible	Faible	Moyenne
	2 à 5 SLD	Faible	Moyenne	Élevée
	6 SLD ou plus	Moyenne	Élevée	Élevée
ENT		**1 à 4 DE**	**5 à 15 DE**	**16 DE ou plus**
	0 ou 1 GDR	Faible	Faible	Moyenne
	2 GDR	Faible	Moyenne	Élevée
	3 GDR ou plus	Moyenne	Élevée	Élevée
SOR ou INT		**1 à 5 DE**	**6 à 19 DE**	**20 DE ou plus**
	0 ou 1 GDR	Faible	Faible	Moyenne
	2 à 3 GDR	Faible	Moyenne	Élevée
	4 GDR ou plus	Moyenne	Élevée	Élevée

Figure 3.11 — Évaluation de la complexité des composants.

Le tableau de la figure 3.12 donne, selon son degré de complexité, le nombre de points de fonction correspondant à chaque composant fonctionnel.

Degré de complexité du composant	Faible	Moyenne	Élevée
Nombre de points de fonction du GDI	7	10	15
Nombre de points de fonction du GDE	5	7	10
Nombre de points de fonction de l'ENT	3	4	6
Nombre de points de fonction de la SOR	4	5	7
Nombre de points de fonction de l'INT	3	4	6

Figure 3.12 — Nombre de points de fonction des composants.

3.8.3 L'ajustement de la taille

La première étape de la méthode des points fonctionnels est un *dénombrement* des différents composants fonctionnels et de leur degré de complexité : la somme des points de fonction de chaque composant est appelée *nombre de points de fonction brut* (PFB).

La seconde étape, qui est facultative, consiste à ajuster l'évaluation obtenue. On corrige le nombre de points de fonction brut, en appréciant les spécificités du projet pouvant influer sur l'effort. Ce sont des fonctionnalités techniques et ergonomiques fournies à l'utilisateur de l'application. La méthode en identifie quatorze, appelées CGS, *caractéristiques générales du système*. Chacune se voit attribuer une note, comprise entre 0 et 5, en fonction de l'influence qu'elle exerce : on l'appelle le DI, *degré d'influence*. Les CGS et les valeurs associées sont données en annexe B.

L'analyse des caractéristiques permet de calculer un degré d'influence total, le DIT, compris entre 0 et 70 :

$$DIT = \text{Somme } (DI_{i)} \text{ avec } i = 1 \text{ à } 14.$$

Le *facteur d'ajustement FA* permet d'ajuster le nombre de points de fonction brut (PFB) de + ou – 35% :

$$FA = 0{,}65 + \frac{DIT}{100}$$

pour obtenir le nombre de points de fonction ajusté (PFA) :

$$PFA = FA * PFB$$

Chaque caractéristique isolément ne peut faire varier le nombre de PFB que de 35/14 = 2,5 %. Ce point a fait l'objet de discussions, car certaines caractéristiques peuvent peser beaucoup plus lourd.

En pratique, l'application du facteur d'ajustement n'est pas systématiquement réalisée, surtout quand on fait l'estimation au stade de l'étude préalable. Certains choisissent d'évaluer la complexité globale du projet sans passer par l'évaluation analytique des CGS. D'autres préfèrent s'en tenir au nombre brut.

3.8.4 La transformation du nombre de points de fonction en charge

Le coefficient de transformation est variable selon l'environnement matériel et humain. Il est recommandé que chaque entreprise établisse sa base de projets pour déterminer ses propres coefficients.

Certains auteurs ont considéré qu'on pouvait établir une correspondance entre la taille « fonctionnelle » de l'application et la taille du logiciel correspondant. Ils proposent une formule de transformation du nombre de points de

fonction en nombre d'instructions sources livrées[1] qui prend en compte le langage de programmation que l'on envisage d'utiliser. Cette transformation permet alors d'appliquer le modèle Cocomo II.

En général, on calcule la charge en convertissant directement le nombre de points de fonction. Les estimations généralement retenues sont les suivantes :

- En fin d'étude préalable : 3 jours par point de fonction, 2 jours s'il s'agit d'un petit projet, 4 jours s'il s'agit d'un grand projet.
- En fin d'étude détaillée : 1 à 2 jours par point de fonction selon l'environnement.

Ainsi, en fin d'étude détaillée, un projet de 80 points de fonction « pèse » en général 160 jours-personne en environnement grand système et 80 en environnement client-serveur.

La référence aux points de fonction dans les méthodes agiles est développée au § 3.11.

3.9 L'ESTIMATION PROBABILISTE

Lorsque l'incertitude est élevée il est parfois préférable de prendre en compte plusieurs valeurs d'estimation. On va présenter deux approches : l'estimation à trois points et la méthode de Monte-Carlo.

L'*estimation à trois points* peut être utilisée à tous les niveaux, projet, phase, activité ou tâche. Elle consiste à rechercher non pas avec une, mais trois valeurs d'estimation :

- une valeur optimiste (f), c'est-à-dire faible car on fait l'hypothèse qu'on ne rencontrera aucun problème (disponibilité des ressources, productivité, relations avec le client ou les utilisateurs…) ;
- une valeur pessimiste (F), qui est élevée car on suppose que l'on va se heurter à de nombreuses difficultés ou imprévus ;
- une valeur considérée comme la plus probable (p), avec une part vraisemblable de difficulté.

1. Voir C. JONES, *Applied Software Measurement, Assuring Productivity and Quality*, McGraw-Hill, New York, 1991.

On prend alors comme estimation la valeur calculée suivante :

$$\text{moyenne} = (f + F + p)/3$$

Souvent, notamment si l'on a un ensemble d'estimations qui se répartissent autour d'une valeur la plus probable, on prend une moyenne pondérée :

$$\text{moyenne} = (f + F + 4p)/6$$

La méthode de *Monte-Carlo* peut être utilisée avec une méthode ascendante. Elle nécessite d'avoir, pour chaque activité ou composant, une connaissance fine de la répartition des valeurs possibles de la charge. Compte tenu de la quantité de calculs, on s'appuie sur un logiciel, dont la logique de fonctionnement est la suivante :

1. Pour chaque activité, on tire au hasard une valeur de charge possible, selon la loi de distribution qui lui est attachée.
2. On fait ensuite la somme de toutes les valeurs obtenues, et on obtient une valeur totale du projet ou de la phase.
3. On recommence n fois les pas 1 et 2.
4. On prend la moyenne des n valeurs obtenues.

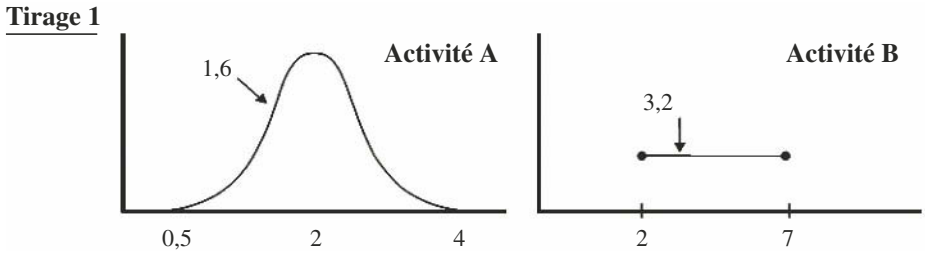


Figure 3.12 — Premier tirage dans la méthode de Monte-Carlo

3.10 DÉMARCHE GÉNÉRALE D'ESTIMATION

Lorsque l'on doit estimer la charge d'un projet pour lequel les méthodes répertoriées (évaluation analytique, points fonctionnels) ne peuvent pas s'appliquer, on procède de la façon suivante.

On recherche tout d'abord à déterminer des paramètres pertinents, c'est-à-dire des éléments que l'on pourra dénombrer et auxquels on pourra attacher une quantité de travail. Pour cela, on effectue un découpage du projet en tâches, en distinguant les tâches uniques et les tâches répétitives.

Pour les **tâches répétitives**, on recherche un élément qui est proportionnel au travail à effectuer, c'est-à-dire un paramètre. Soit, par exemple, dans un projet de Mise en œuvre la tâche « Former les utilisateurs ». La charge de travail pour cette tâche qui incombe à l'équipe de projet est proportionnelle au nombre de sessions de formation à assurer, chaque session comportant un nombre maximum d'utilisateurs. Le paramètre est donc la session de formation. En fonction du nombre d'utilisateurs sur chaque site, on pourra dénombrer le nombre de sessions à assurer.

Ensuite, on va déterminer le poids des paramètres associés aux tâches répétitives. On peut utiliser différents moyens, notamment le prototypage et l'échantillonnage.

Le prototypage revient à réaliser sommairement une des tâches répétitives, ce qui donne une indication sur l'effort nécessaire à ce type de tâche.

L'échantillonnage consiste, pour chaque type de tâche, à faire réaliser réellement quelques unités, ce qui permet de prendre comme poids la valeur moyenne. Dans l'exemple de la formation des utilisateurs, le poids sera la durée de la session de formation : on la déterminera en formant deux ou trois utilisateurs pilotes.

Pour estimer la charge des **tâches uniques**, différents moyens peuvent être utilisés.

On peut rechercher un paramètre propre à la tâche. Par exemple, pour la tâche « Construire le séminaire de formation », on peut considérer qu'il y a un rapport de proportionnalité à peu près constant entre le nombre de fonctions qu'il faut présenter et la charge de travail nécessaire à l'élaboration de cette présentation. Le paramètre retenu est donc la fonction du futur système pour laquelle il faut préparer une formation. On peut éventuellement classer les fonctions selon leur degré de complexité et déterminer une charge de préparation de formation correspondante.

Sinon, on peut rapprocher cette tâche de tâches analogues sur des projets antérieurs et se baser sur les charges consommées.

On peut également utiliser la méthode Delphi, par exemple avec trois ou quatre personnes de l'équipe projet.

Si l'on hésite entre un éventail de valeurs, on peut retenir une approximation probabiliste, par exemple la moyenne pondérée d'une estimation que l'on ramène à trois points (paragraphe 3.9).

Cette approche permet ainsi d'élargir les méthodes d'estimation des charges à une large variété de types de projet autour des systèmes d'information.

3.11 L'ESTIMATION DANS LES MÉTHODES AGILES

Plusieurs principes partagés par les méthodes agiles ont des conséquences sur les approches d'estimation des charges.

D'abord, elles font toutes référence à un modèle incrémental et itératif, ce qui implique impérativement une estimation globale en début de projet et des estimations renouvelées à chaque itération.

Ensuite, elles donnent une place importante aux utilisateurs durant tout le cycle de vie. S'ils sont autorisés à demander des modifications fonctionnelles jusque dans les phases aval du projet, ils sont tenus d'accompagner au quotidien le travail des développeurs par des indications et des évaluations des différents composants du produit en cours de construction. De ce fait, ils sont parties prenantes des estimations de charge.

Enfin, comme le respect d'un délai court est un impératif, la principale variable d'ajustement est le périmètre du projet. Les clients/utilisateurs sont ainsi amenés à affecter des priorités aux fonctionnalités et exigences demandées et à redéfinir le contenu du produit en fonction des estimations.

Les méthodes qui préconisent explicitement une phase de planification générale du projet et de construction de son architecture, RAD et DSDM notamment, suggèrent une session participative pour déterminer l'enveloppe globale du projet.

Les méthodes d'estimation préconisées sont soit une approche basée sur l'*analogie*, par exemple la méthode Delphi, avec de préférence des estimations à trois points, soit la méthode des *points de fonction* après une détermination générale des besoins.

RAD suggère une productivité de 0,5 jour par point de fonction pour l'ensemble du projet. DSDM indique une productivité moyenne de 0,25 jour par point de fonction pour la phase de Conception et Construction, et de 0,7 jour pour une équipe nouvellement formée qui n'est pas encore habituée à travailler ensemble[1].

Lorsque l'on entre dans les itérations de développement, les méthodes de types *analytiques* sont privilégiées, c'est-à-dire que l'on s'appuie sur le dénombrement et la qualification à différents niveaux de granularité. On dénombre d'abord les fonctionnalités retenues, ensuite les scénarios définissant les fonctionnalités, et plus finement les tâches nécessaires pour réaliser les scénarios.

1. DSDM, version 4.2, Management, Estimating DSDM projects, http://www.dsdm.org/version4/2/public/Estimating.asp

Certains peuvent utiliser une estimation paramétrique de type COCOMO simplifié, c'est-à-dire en nombre de lignes de code, qui ensuite sera converti en jours. L'évaluation peut prendre en compte la difficulté intrinsèque de la fonction à développer, sa dépendance avec d'autres composants, ainsi que la productivité des développeurs.

XP propose d'utiliser une notion de « point », qui pourrait être qualifié de point de développement. Chaque scénario est estimé en nombre de points, à partir des indications fournies par les utilisateurs et d'un rapprochement avec des scénarios similaires, antérieurement rencontrés dans le projet.

En début des itérations de développement, s'il s'avère particulièrement difficile de faire une évaluation, notamment par méconnaissance de l'outil de développement, on développe rapidement un prototype à des fins d'estimation, ce qui est appelé un *spike*.

On peut considérer que cette approche s'apparente à la méthode analytique, mais il faut toutefois souligner sa dimension dynamique dans la mesure où :

- la typologie peut être redéfinie pour chaque projet ;
- les poids des types de composants peuvent être réévalués au cours du projet.

Le passage par une métrique en points évite de faire la confusion entre charge et durée. XP parle de « jour-idéal » pour faire référence à des journées de travail complètes, c'est-à-dire sans réunion ni autre interruption. Le point permet de convertir, par un facteur de 2 ou 3, un jour-idéal en durée. La valeur du point est ensuite ajustée en fonction de l'observation de la productivité de l'équipe.

En effet, après une première itération de développement, le nombre de points produits par l'équipe est considéré comme une bonne mesure de sa productivité. XP parle de *vélocité* de l'équipe. Cette valeur correspond à la capacité de production de l'équipe à chaque itération, les itérations étant en général d'une durée fixée.

La vélocité de l'équipe permet de déterminer le travail possible et de faire des ajustements. En début de chaque itération de développement, l'équipe rassemble un nombre de scénarios, les évalue et vérifie que leur somme en points correspond à sa capacité de production, sinon le client doit indiquer ses priorités. Si la définition d'un scénario excède la vélocité de l'équipe, il faudra le scinder en plusieurs scénarios de taille réduite.

Ensuite, les scénarios sont éclatés en tâches, qui sont à leur tour estimées et réparties entre les développeurs de façon à équilibrer leur charge. Si la somme de ces estimations élémentaires est différente de la valeur retenue en début d'itération,

l'équipe se retourne vers le client pour ajouter ou retirer un scénario de l'itération de développement.

En conclusion, l'estimation des charges apparaît, dans un contexte agile, avoir un objectif un peu particulier. Bien sûr, comme dans tous les projets, l'estimation initiale sert à avoir une enveloppe et déterminer un budget. Mais ensuite, il ne s'agit pas de faire une planification détaillée des différentes tâches et d'en suivre l'avancement. Les estimations permettent principalement d'une part de négocier sur des bases très précises avec les clients/utilisateurs pour qu'ils ajustent leurs exigences ; d'autre part de maintenir un rythme de production stable dans l'équipe de développement.

4

Les techniques de planification des délais

4.1 L'UTILISATION DE LA PLANIFICATION

La planification des délais d'un projet comporte deux aspects, l'ordonnancement des activités et l'élaboration de l'échéancier, marqués par l'utilisation de deux techniques complémentaires : la technique des graphes et la technique Gantt (figure 4.1).

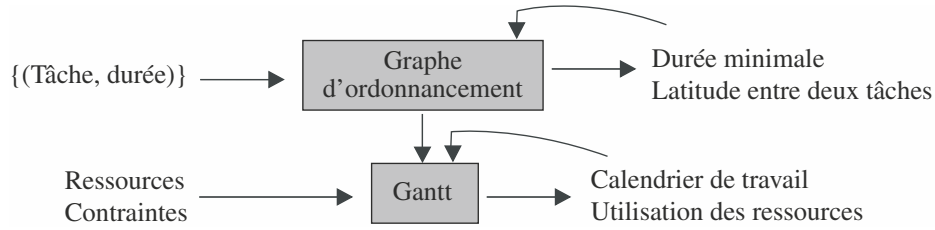


Figure 4.1 — Les deux techniques de planification.

En premier lieu, indépendamment de toute affectation du travail, on prend en compte les caractéristiques de chacun des éléments résultant d'un découpage de type WBS (paragraphe 2.2) en général affiné. On part donc d'une liste d'activités avec la durée estimée de chacune d'elles. On commence la planification par une réflexion sur les *contraintes d'ordonnancement* de ces activités et sur les possibilités de parallélisme. La technique des graphes d'ordonnancement permet de calculer la durée minimum du projet, ainsi que les temps d'attente éventuels entre deux activités. Après une première planification, il est possible d'ajuster le

découpage ou de chercher à assouplir certaines contraintes. En effet, un parallélisme faible offre peu de marge de manœuvre dans la répartition du travail. On peut également utiliser des graphes d'ordonnancement pour traduire plusieurs hypothèses de fin de projet souhaitée (figure 4.1).

Dans un second temps, il s'agit d'établir un *calendrier de travail*. La durée minimum obtenue précédemment est à rapprocher du délai « normal » ou « raisonnable » proposé par certaines méthodes d'estimation des charges (paragraphe 3.7). On utilise ici le diagramme de Gantt. Pour cela, il faut faire une hypothèse sur les ressources dont on disposera et, parfois, prendre en compte les contraintes de disponibilité attachées à ces ressources. On peut établir plusieurs scénarios qui correspondent à différentes hypothèses : les diagrammes aident à décider quel est le scénario souhaitable (figure 4.1).

Dans ce chapitre, après une présentation des techniques des graphes d'ordonnancement et de celle du diagramme de Gantt, nous nous centrerons sur l'établissement d'une planification opérationnelle. Nous terminerons sur la prise en compte de l'incertitude, *via* une utilisation probabiliste du graphe d'ordonnancement.

4.2 LES GRAPHES D'ORDONNANCEMENT

Il existe deux formalismes de représentation de l'ordonnancement des activités : la méthode des antécédents et la méthode du diagramme fléché.

Dans le graphe de la méthode des *antécédents* (figure 4.2), les activités figurent sur les rectangles, les flèches ne représentant que les liens. Les ronds représentent des jalons, c'est-à-dire des activités de durée nulle.

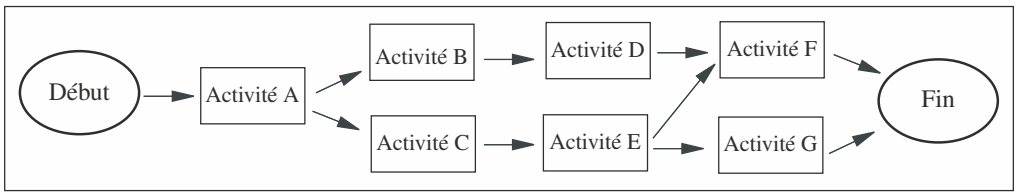


Figure 4.2 — Graphe de la méthode des antécédents.

Dans le graphe de la méthode du *diagramme fléché* (figure 4.3), les activités figurent sur les flèches, les ronds représentent des événements appelés jalons. Ceux-ci n'ont pas de durée. Pour traduire de façon correcte les contraintes d'enchaînement, on est parfois amené à introduire des activités fictives, représentées par des flèches en pointillés.

Ces deux notations sont équivalentes. Dans cet ouvrage, nous utilisons le formalisme du graphe des antécédents, qui est plus simple et qui est retenu par la majorité des progiciels de planification. Ce type de réseau peut être utilisé à

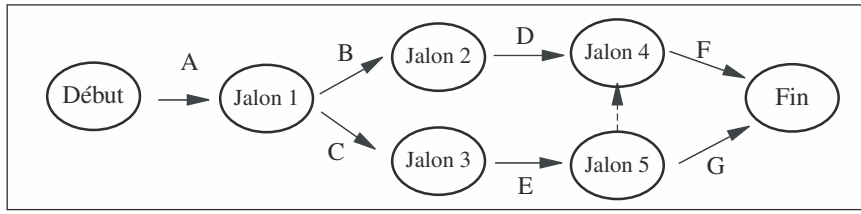


Figure 4.3 — Graphe de la méthode du diagramme fléché.

différentes mailles de décomposition du projet. Dans la suite, nous employons le terme « tâche » de façon générique, celle-ci pouvant aussi bien être une tâche élémentaire, qu'une activité ou même une phase.

Soulignons qu'il y a eu ces dernières années un changement de vocabulaire : ce graphe a longtemps été appelé réseau PERT (comme nous l'avons fait dans de précédentes éditions de cet ouvrage). Aujourd'hui, lorsque l'on emploie le terme PERT, c'est souvent pour faire référence au PERT probabiliste exposé en fin de ce chapitre. De son côté, l'IPMA ainsi que le logiciel MS Project, dans sa version 2003, utilisent le terme organigramme des tâches, OT, pour désigner le graphe des antécédents.

4.3 LES TYPES DE LIENS

Les liens entre les tâches représentent des contraintes provenant de la nature des tâches elles-mêmes. Il existe quatre types de liens : le lien fin-début, le lien fin-fin, le lien début-début, le lien début-fin (figure 4.4).

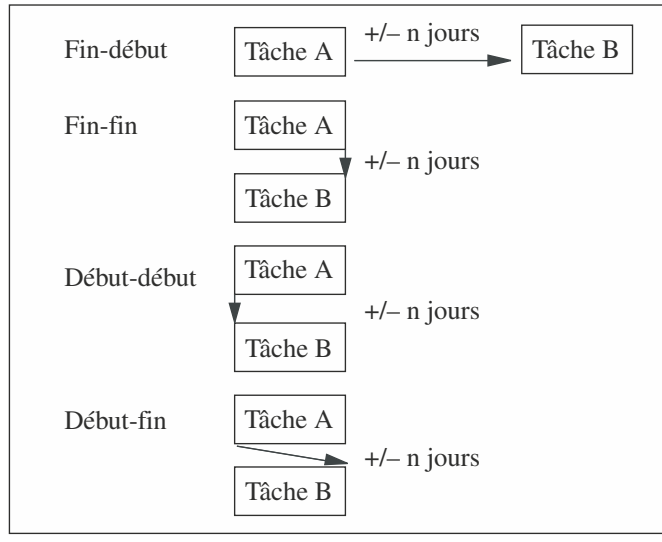


Figure 4.4 — Les types de lien.

Le lien *fin-début* est le plus courant : la tâche A doit être terminée pour que la tâche B puisse commencer. La tâche A est le *prédécesseur* de la tâche B. La tâche B est le *successeur* de la tâche A. On dit parfois que A est antécédente et B subséquente.

Par exemple (figure 4.5), la tâche de Programmation doit être terminée pour que la tâche de Test puisse commencer.

Figure 4.5 — Lien fin-début.

Le lien peut être caractérisé par un *délai*, exprimé en jours. Si le délai est négatif (– n jours), on parle d'une *avance*[1]. *L'avance peut être exprimée en pourcentage de la charge restante (– x %)*.

Par exemple (figure 4.6), la tâche de Test peut commencer 10 jours avant la fin de la tâche de Programmation (– 10 jours), pour préparer l'environnement de test.

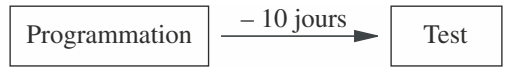


Figure 4.6 — Lien fin-début avec avance.

Si le délai est positif (+ *n* jours), on parle d'un *retard* [2].

Par exemple (figure 4.7), la tâche de Prise en compte des suggestions peut commencer 10 jours après la tâche d'Installation du prototype.

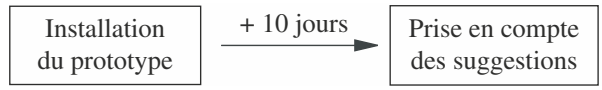


Figure 4.7 — Lien fin-début avec retard.

Le lien *fin-fin* signifie : c'est la fin de la tâche A qui commande la fin de la tâche B. La tâche B ne peut s'arrêter que lorsque A s'arrête.

Par exemple (figure 4.8), la tâche d'Encadrement de programmation dure tant que la tâche de Programmation n'est pas terminée.

1. *Lead* en anglais.
2. *Lag* en anglais.

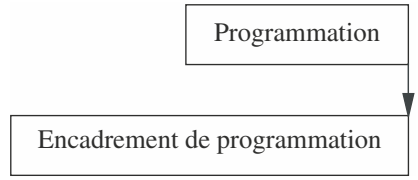


Figure 4.8 — Lien fin-fin.

Ce lien peut être également caractérisé par une avance ou un retard.

Par exemple (figure 4.9), la tâche d'Encadrement de mise en œuvre se termine 20 jours après la fin de la tâche de Mise en œuvre d'un logiciel, pour assurer une aide au démarrage.

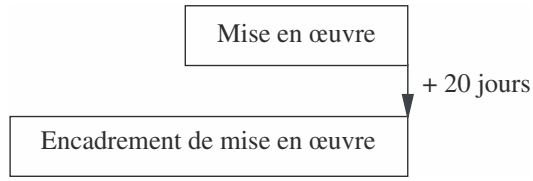


Figure 4.9 — Lien fin-fin avec retard.

Le lien *début-début* signifie : c'est le début de la tâche A qui déclenche le début de la tâche B. B doit obligatoirement commencer quand A commence.

Par exemple (figure 4.10), la tâche de Modélisation doit commencer en même temps que la tâche Interviews.

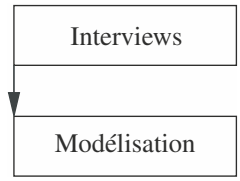


Figure 4.10 — Lien début-début.

Ce lien peut être également caractérisé par une avance ou un retard.

Par exemple (figure 4.11), la tâche Préparation de l'environnement technique doit commencer 10 jours avant le début de la tâche de Programmation et la tâche d'Alimentation du dictionnaire de données doit commencer 10 jours après le début de la tâche de Modélisation.

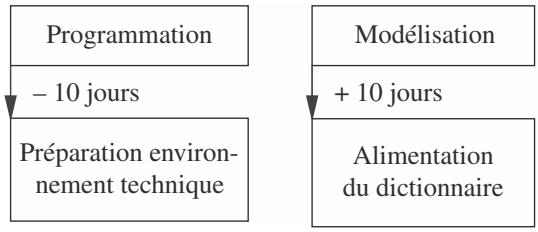


Figure 4.11 — Liens début-début avec délai.

Le lien *début-fin* signifie : c'est le début de la tâche A qui marque la fin de la tâche B. La tâche B ne peut s'arrêter tant qu'A n'a pas commencé.

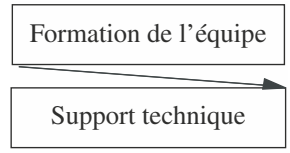


Figure 4.12 — Lien début-fin.

Par exemple (figure 4.12), le début de la tâche Formation de l'équipe marque la fin de la tâche Support technique.

Ce lien peut être également caractérisé par une avance ou un retard.

Par exemple (figure 4.13), la tâche Exploitation de l'ancien logiciel s'arrêtera 15 jours après le début de la tâche Exploitation du nouveau logiciel.

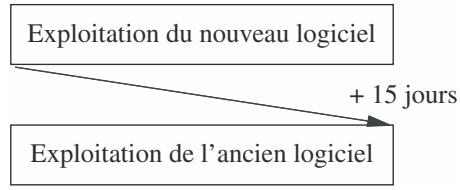


Figure 4.13 — Lien début-fin avec retard.

4.4 LA MÉTHODE DU CHEMIN CRITIQUE

L'analyse d'un graphe s'effectue avec une méthode appelée la méthode du chemin critique[1], car elle permet de mettre en évidence des chemins qui comportent des tâches critiques, dans le sens où elles vont retarder la fin du projet si elles sont elles-mêmes en retard.

1. Ou CPM : *Critical Path Method*.

Pour mettre en évidence ce chemin critique, on calcule les paramètres clés attachés à chaque tâche du graphe.

Pour chacune, on veut obtenir :

- les dates au plus tôt : Début au plus tôt et Fin au plus tôt ;
- les dates au plus tard : Début au plus tard et Fin au plus tard[1] ;
- les marges[2] : marge totale et marge libre.

La signification des dates au plus tôt (D+tôt, F+tôt) et au plus tard (D+tard, F+tard) est la suivante.

Compte tenu des contraintes d'enchaînement, de la durée des tâches et de la date de début de projet, la tâche T_i ne peut pas commencer avant D+tôt et ne peut se terminer avant F+tôt.

Par ailleurs, compte tenu des contraintes d'enchaînement, de la durée des tâches et de la date de fin de projet, elle ne doit pas se terminer après F+tard sans mettre le projet en retard. De même, elle ne doit pas commencer après D+tard, sinon la date de fin du projet serait dépassée.

Pour calculer ces dates, nous devons avoir la durée d_i de chaque tâche T_i. On va supposer dans un premier temps qu'il n'y a que des liens de type fin-début.

Pour calculer les dates au plus tôt de chacune des tâches, on va faire l'hypothèse d'une date de début de projet (t_0) et on va parcourir le graphe vers l'avant en respectant les liens.

Si la tâche T_i se situe en début de projet, la date de début au plus tôt est t_0. Sa date de fin au plus tôt est $(t_0 + d_i - 1)$.

$$\text{D+tôt } (T_i) = t_0$$
$$\text{F+tôt } (T_i) = t_0 + d_i - 1$$

Par exemple, en supposant tous les jours ouvrables (figure 4.14) :

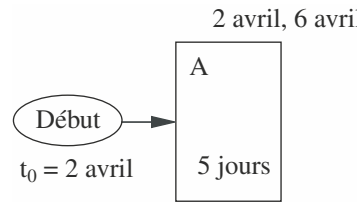


Figure 4.14 — Dates au plus tôt.

1. En anglais : *Early start* et *Early finish*, *Late start* et *Late finish*.
2. *Slack* en anglais.

Si la tâche T_i ne se situe pas en début de projet, elle a des prédécesseurs. Sa date de début au plus tôt est égale à la plus grande des dates de fin au plus tôt de tous ses prédécesseurs plus une période. Sa date de fin au plus tôt est obtenue en ajoutant la durée de la tâche moins une période.

$$D+tôt\ (T_i) = sup\ \{F+tôt\ (prédécesseurs)\} + 1$$

$$F+tôt\ (T_i) = D+tôt\ (T_i) + d_i - 1$$

Par exemple (figure 4.15), en supposant tous les jours ouvrables :

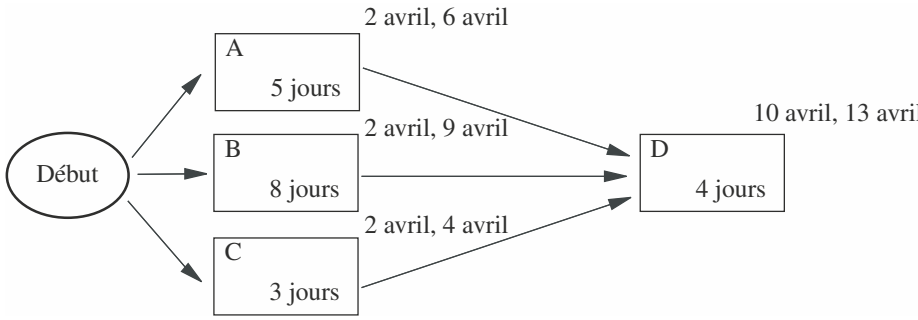


Figure 4.15 — Dates au plus tôt avec prédécesseurs.

Pour calculer les *dates au plus tard* de chacune des tâches, on va faire l'hypothèse d'une date de fin de projet et on va parcourir le graphe vers l'arrière en respectant les liens.

Soit t_f la date de fin de projet. Si la tâche T_i se situe en fin de projet, la *date de fin au plus tard* est t_f. Sa *date de début au plus tard* est $(t_f - d_i + 1)$.

$$F+tard\ (T_i) = t_f$$

$$D+tard\ (T_i) = t_f - d_i + 1$$

Par exemple (figure 4.16), en supposant que le projet se termine le 15 décembre et que tous les jours sont ouvrables :

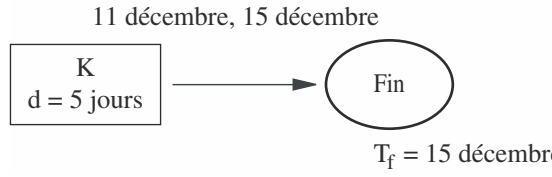


Figure 4.16 — Dates au plus tard.

Si la tâche T_i ne se situe pas en fin de projet, elle a des successeurs. Sa date de fin au plus tard est égale à la plus petite des dates de début au plus tard de tous ses successeurs moins 1. Sa date de début au plus tard est obtenue en soustrayant la durée de la tâche et en ajoutant 1.

$$F{+}tard\ (T_i) = \inf\{D{+}tard\ (successeurs)\} - 1$$

$$D{+}tard\ (T_i) = F{+}tard\ (T_i) - d_i + 1$$

Par exemple, en supposant tous les jours ouvrables (figure 4.17) :

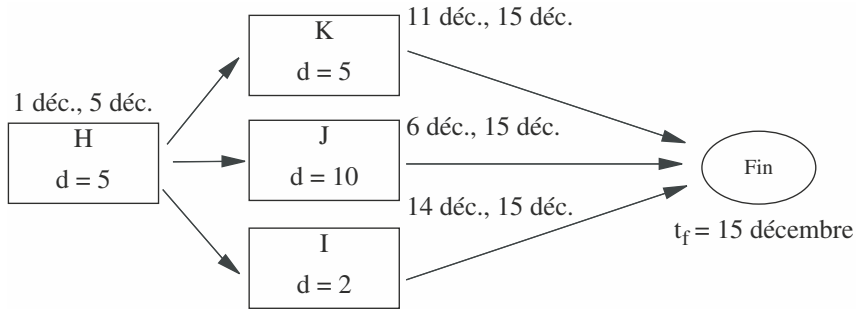


Figure 4.17 — Dates au plus tard avec successeurs.

S'il y a d'autres types de liens, c'est la tâche maître qui impose les dates (début et/ou fin) de la tâche dépendante, aussi bien pour les dates au plus tôt que pour les dates au plus tard.

Par exemple, considérant que les dates attachées à chaque tâche se lisent dans l'ordre :

Début au plus tôt, fin au plus tôt

Début au plus tard, fin au plus tard

et que l'on mesure le temps en semaines, on a les paramètres portés sur la figure 4.18.

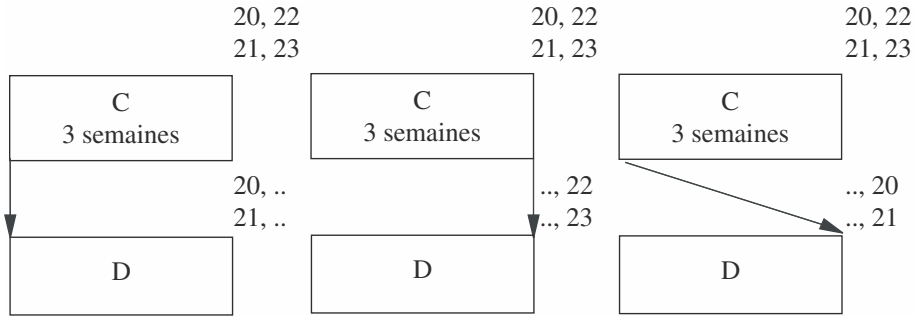


Figure 4.18 — Dates clés avec liens particuliers.

La *marge* attachée à chaque tâche est la différence entre date au plus tard (T_i) et date au plus tôt (T_i). En l'absence de liens autres que des liens fin-début, elle peut être calculée indifféremment sur les dates de début ou sur les dates de fin. Sinon, on peut avoir deux marges différentes sur une tâche, l'une attachée au début de la tâche, l'autre attachée à la fin de la tâche. La marge représente la latitude dont on dispose quand on élabore le planning.

On peut faire des simulations avec différentes dates de fin du projet. La marge ne doit jamais être négative : sinon, il faut revoir le graphe, en modifiant la tâche (éclatement de la tâche en deux pour réduire la durée de la tâche et augmenter le parallélisme), en levant certaines contraintes (découplement de tâches) ou en modifiant la date de fin du projet visée.

La marge, telle que définie ci-dessus, est parfois appelée *marge totale*, lorsque l'on veut l'opposer à une marge plus réduite appelée *marge libre*. On dispose d'une marge libre *ml* sur T_i si on peut planifier la T_i à la date (D+tôt (T_i) + ml) sans que cela ait de conséquence sur ses successeurs (c'est-à-dire qu'on peut toujours les planifier au plus tôt).

$$\text{Marge libre } (T_i) = \inf \{\text{D+tôt (successeurs)}\} - \text{F+tôt } (T_i) - 1$$

Le schéma de la figure 4.19 illustre la différence entre marge libre et marge totale. La tâche A dure deux périodes : sa planification au plus tôt se fait aux périodes 1 et 2, et sa planification au plus tard aux périodes 5 et 6. La marge totale de A est donc de 4. La tâche B commence au plus tôt en période 6. Si l'on planifie la tâche A aux périodes 4 et 5, cela est sans conséquence sur les choix de la planification pour la tâche B. En revanche, si l'on fait une planification en utilisant la marge totale, c'est-à-dire aux périodes 5 et 6, on ne pourra planifier la tâche B qu'à partir de la période 7. La marge libre de A est donc de 3.

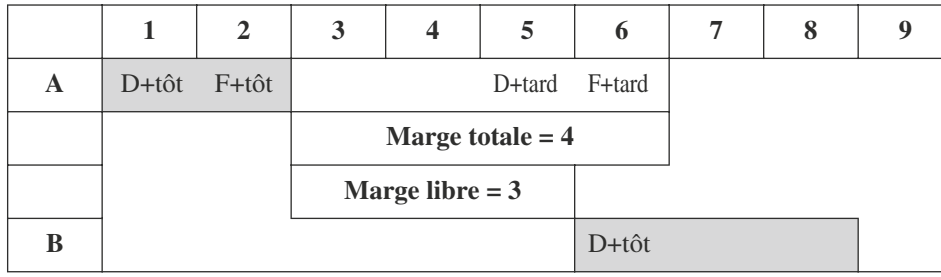


Figure 4.19 — Chemin critique.

Les marges (totale et libre) figurent sur le réseau de la figure 4.20.

Le *chemin critique* est le chemin du graphe sur lequel les marges totales sont nulles. La marge libre de toutes les tâches du chemin critique est donc nulle.

S'il n'y a que des liens fin-début, le chemin critique est le chemin le plus long : il est donc à surveiller tout particulièrement.

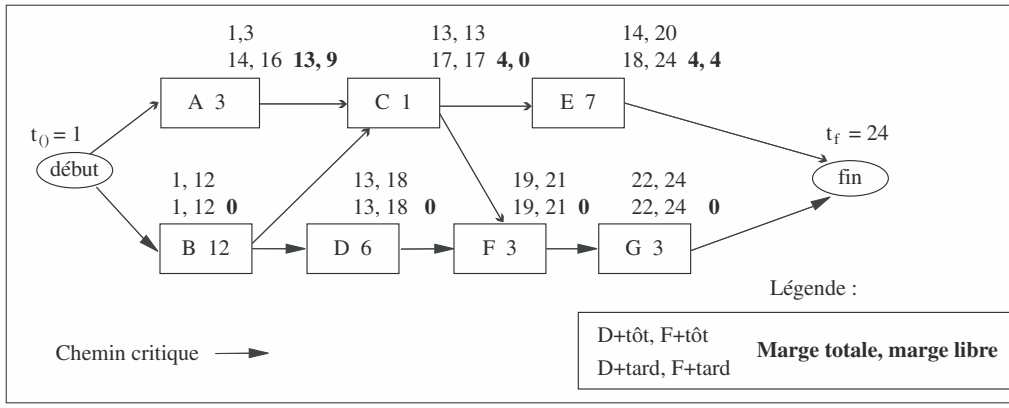


Figure 4.20 — Chemin critique.

S'il y a d'autres liens que des liens fin-début dans un réseau, le chemin critique peut ne pas être complet, c'est-à-dire ne pas parcourir le graphe du début à la fin.

C'est également le cas si l'on a des contraintes temporelles, c'est-à-dire des dates imposées (début imposé ou fin imposée) pour certaines tâches. Ce point sera illustré au chapitre 11 (paragraphe 11.9).

4.5 LE DIAGRAMME DE GANTT

Le graphe des antécédents permet de faire apparaître les possibilités de parallélisme dans l'exécution des tâches et donne les dates de fin du projet possibles en dehors des contraintes de ressources. Pour passer à un planning (calendrier du projet), il faut faire des hypothèses de ressources et affecter les tâches à des personnes ou à des profils de personnes. On pourra faire plusieurs simulations selon la taille de l'équipe envisagée. On prend également en compte les contraintes de calendrier (jours non ouvrables, jours fériés…).

On utilise, pour représenter le planning, le diagramme de Gantt, qui se construit de la façon suivante :

- en abscisse, on a l'axe du temps ;
- en ordonnée, on peut avoir soit les tâches, soit les personnes affectées aux tâches.

Selon que l'on utilise ou non les marges pour effectuer la planification, on parlera de *planification au plus tôt* ou de *planification au plus tard*.

Le diagramme de Gantt, qui porte le nom de son concepteur, a été élaboré au début du XXe siècle pour représenter de façon graphique la répartition du travail en atelier. C'est probablement la technique de gestion de projet la plus largement utilisée.

Voici des exemples de planning à partir du réseau ci-dessus (figure 4.20).

- Planification au plus tôt : on planifie les tâches en s'appuyant sur les dates au plus tôt (figure 4.21). On utilise deux personnes : ressource 1 (R1) et ressource 2 (R2). R2 doit attendre pendant 9 périodes, car la tâche C ne peut démarrer avant la fin de la tâche B. Les parties grisées représentent la marge.

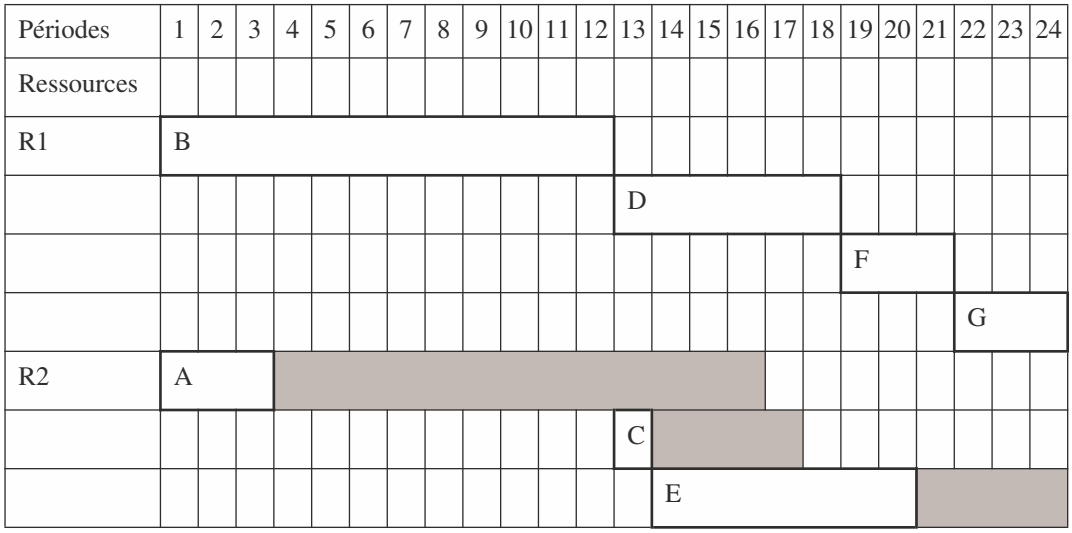


Figure 4.21 — Diagramme de Gantt avec planification au plus tôt.

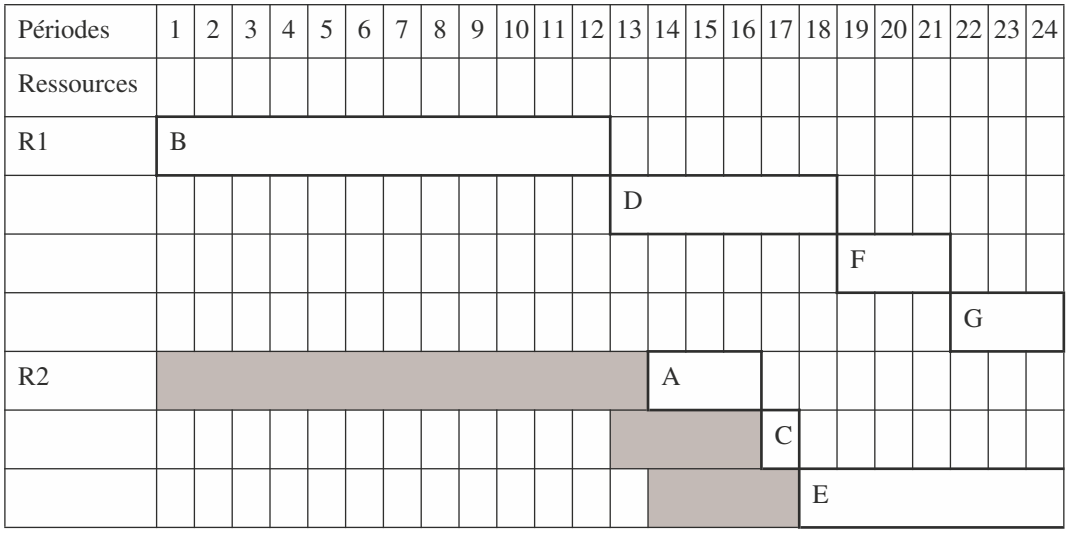


Figure 4.22 — Diagramme de Gantt avec planification au plus tard.

- Planification au plus tard : on planifie les tâches en s'appuyant sur les dates au plus tard (figure 4.22). On utilise deux personnes : ressource 1 (R1) et ressource 2 (R2). Aucune tâche ne peut prendre de retard, sinon le projet ne pourra s'achever à la date visée.
- Amélioration du diagramme au plus tôt de la figure 4.21 : on utilise trois ressources, R1, R2 et R3 ; et on place la tâche A de façon à éviter une attente pour R3 (figure 4.23).

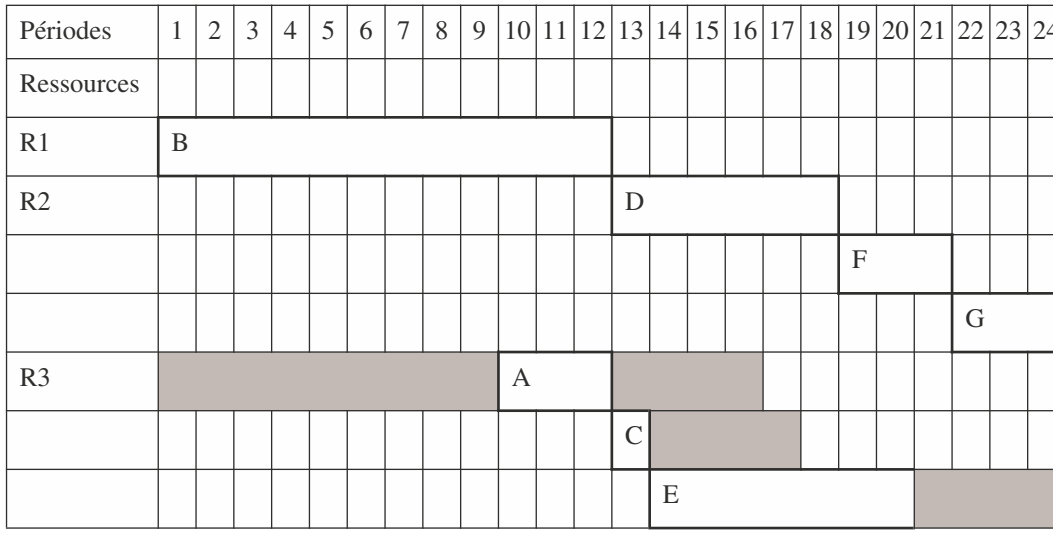


Figure 4.23 — Diagramme de Gantt avec planification améliorée.

4.6 LA PLANIFICATION OPÉRATIONNELLE

4.6.1 La prise en compte des contraintes

Pour établir un diagramme Gantt en vue d'une planification opérationnelle, on doit prendre en compte toutes les contraintes, en commençant par celles qui sont les plus fortes et en terminant par les tâches sur lesquelles on a la latitude la plus élevée. Souvent, on cherche une planification satisfaisante, qui, souvent, n'est pas la seule possible.

On distingue plusieurs types de contraintes :

- Les *contraintes de liens entre tâches* ont été mises en évidence sur le graphe des antécédents, et en particulier le chemin critique. Si l'on veut terminer dans le délai minimum, on planifie en premier les tâches qui sont sur le chemin critique. On planifiera ensuite les tâches qui sont liées aux tâches du chemin critique par des liens du type début-début, fin-fin ou début-fin, et enfin on placera les tâches qui sont des prédécesseurs des tâches critiques.

- Les *contraintes temporelles*, c'est-à-dire les dates imposées pour une ou plusieurs tâches, figurent également sur le graphe des antécédents. Ce sont des contraintes très fortes, puisqu'elles ne laissent pas de choix.
- Les *contraintes liées à la disponibilité des ressources* correspondent soit à une ressource spécialisée, seule à même d'effectuer certaines des tâches du projet, soit à une pénurie de ressources. Dans les deux cas, cela peut conduire à revoir le graphe pour supprimer des possibilités de parallélisme que l'on n'est pas à même d'exploiter. Par exemple, si une personne R1 se voit confier deux tâches A et B pouvant être faites en parallèle, il faudra choisir dans quel ordre ces tâches seront effectuées. Cela correspond à la technique du lissage exposée au paragraphe 4.6.4.
- Les *contraintes d'exclusion* indiquent que des tâches indépendantes ne doivent pas être planifiées simultanément, souvent pour des raisons de sécurité. Ces contraintes sont assez rares dans notre domaine. On peut donner comme exemple les deux tâches suivantes : Tests volumétriques et Recette fonctionnelle. Si elles s'effectuent sur la même machine de test, on risque de pénaliser la recette et d'indisposer le client.

4.6.2 L'utilisation des marges

Pour limiter les risques de retard provoqué par des aléas, le chef de projet a souvent tendance à gonfler la durée des tâches, c'est-à-dire à introduire de la marge cachée un peu partout dans la planification. E. Goldratt[1] appelle cette marge un *tampon (buffer)* et conseille de ne pas répartir les tampons sur toutes les tâches, mais de le faire judicieusement pour qu'ils contribuent effectivement à la maîtrise des délais. Sinon, ces marges seront systématiquement consommées sans amélioration de la performance du projet.

Il fait en particulier trois recommandations.

D'abord, il est souhaitable de prévoir un tampon en fin de projet pour absorber les fluctuations du chemin critique.

Ensuite, il est prudent de placer un tampon en fin des tâches qui sont des prédécesseurs des tâches du chemin critique. En effet, certaines de ces tâches peuvent présenter des risques de dérapages, dont les conséquences toucheront le

1. E. Goldratt est l'auteur d'une approche de gestion de production basée sur l'identification des contraintes dominantes et notamment des goulets d'étranglement, dont la prise en compte permet de limiter les stocks et de réduire le cycle de production. Cette approche décrite dans *The Goal* (1984) a été transposée à la planification de projet sous le nom de CCPM, *critical chain project management*, gestion de projet par la chaîne critique. Pour plus de détails, voir le dossier paru dans *La Cible* n° 80 (la revue de l'AFITEP) de juin 2000.

chemin critique. Si le tampon n'est pas entièrement consommé, cela pourra éventuellement permettre d'anticiper sur le démarrage de la tâche critique.

Enfin, si une ressource doit intervenir sur une tâche critique et si elle ne travaille pas déjà sur le projet, il est bon de lui notifier son intervention quelques jours avant, pour qu'elle ne prenne aucun retard au démarrage de la tâche critique.

Le tampon se présente comme une tâche fictive, liée à la tâche qu'elle doit protéger, comme prédécesseur ou comme successeur.

4.6.3 Le nivellement

La technique du nivellement consiste à maintenir le nombre de personnes travaillant simultanément sur le projet en dessous d'une certaine limite. On va donc, en général, augmenter la durée du projet. Le nivellement vise l'ensemble des ressources du projet.

Plusieurs raisons peuvent conduire à utiliser cette technique. Le nivellement évite d'avoir une taille d'équipe de projet trop importante par rapport à la durée totale du projet. Une première hypothèse de planification qui exploite au maximum le parallélisme peut conduire à une taille d'équipe risquant de générer des surcharges de coordination. La disponibilité des ressources (personnes, matériel, locaux…) peut être telle que l'on doit renoncer à utiliser toutes les possibilités d'exécuter des tâches en parallèle, telles qu'elles figurent sur le graphe des antécédents. Le nivellement permet enfin d'étaler dans le temps les dépenses liées au projet.

Soit par exemple le projet structuré en tâches selon le réseau de la figure 4.24.

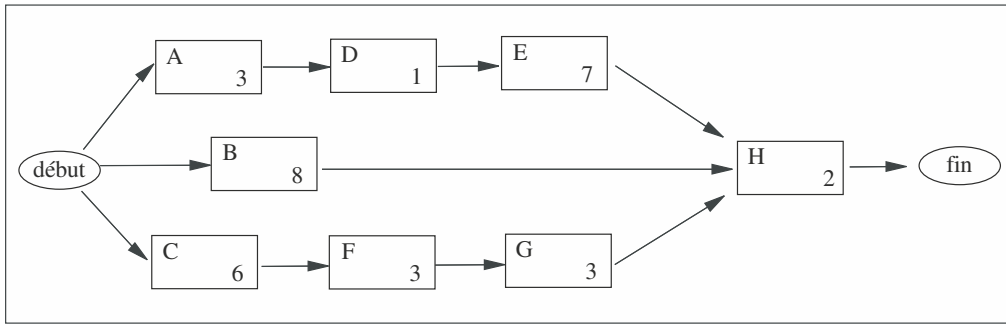


Figure 4.24 — Projet à planifier.

Le graphe fait apparaître trois chemins en parallèle. On va donc faire une première planification avec trois ressources (figure 4.25).

Périodes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Ressources																								
R1					B																			
													H											
R2	C																							
							F																	
										G														
R3	A																							
				D																				
					E																			

Figure 4.25 — Diagramme de Gantt avant nivellement.

Supposons qu'on veuille limiter la taille de l'équipe à deux personnes : on va donc niveler le diagramme, en allongeant la durée du projet (figure 4.26). On en profite pour étaler la montée en charge, en intégrant les deux ressources non pas simultanément mais successivement : R2 n'arrive sur le projet qu'en fin de période 8, quand R1 achève sa première tâche.

Périodes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Ressources																								
R1	B																							
									C															
															F									
																		G						
																					H			
R2									A															
												D												
													E											

Figure 4.26 — Diagramme de Gantt après nivellement.

4.6.4 Le lissage

La technique du lissage consiste à répartir pour chaque ressource sa charge de travail, de telle façon qu'elle ne se trouve à aucun moment en surcharge ou en sous-charge. On va jouer sur les marges pour décaler les tâches. Contrairement à ce qui se passe quand on cherche à niveler, on s'intéresse ici à la répartition de la charge affectée à chaque ressource. Une opération de lissage peut conduire à allonger les délais.

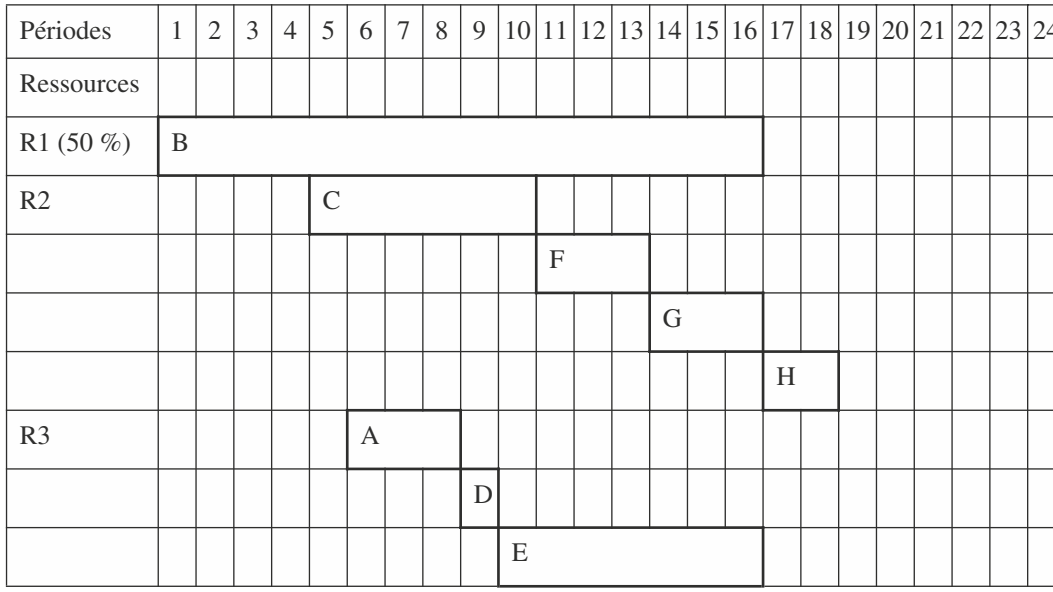


Figure 4.27 — Diagramme de Gantt après lissage.

Les raisons du lissage sont le plus souvent des contraintes liées à l'utilisation des personnes. Parfois, on peut vouloir lisser à cause de la disponibilité réduite d'un matériel.

Nous avons repris le même exemple que ci-dessus (figure 4.25) et nous avons fait une opération de lissage (figure 4.27).

On a supposé que la ressource R1 ne pouvait travailler qu'à mi-temps. Le diagramme de la figure 4.25 était donc en surcharge pour R1. Nous avons dû replanifier en doublant la durée des tâches affectées à R1. On a alors décidé de réaffecter la tâche H, située sur le chemin critique, à R2. On a planifié la tâche B au plus tôt, car elle était maintenant sur le chemin critique. Par contre, on a utilisé les marges des tâches affectées à R3 afin d'étaler la montée en charge du projet.

4.7 LE PERT PROBABILISTE

Cette technique, qui s'appuie sur le graphe des antécédents, permet d'inclure dans la planification le risque et l'incertitude attachés à chaque tâche et d'en déduire une durée du projet assortie d'un niveau de probabilité. La durée de chaque tâche peut être considérée comme une variable aléatoire, c'est-à-dire que l'on peut faire plusieurs estimations (plus ou moins probables) de la durée de la tâche. Alors, la durée de tout chemin dans le graphe PERT (*Program Evaluation and Review Technique*) est également considérée une variable aléatoire, puisque c'est une somme de variables aléatoires.

Sous réserve des conditions suivantes :

- un nombre suffisamment élevé de tâches (minimum 4 sur le chemin) ;
- un ordre de grandeur semblable pour toutes les tâches ;
- l'indépendance entre les durées des tâches ;

la durée probable du chemin obéit à une loi de distribution proche de la loi normale (Laplace-Gauss), dont la représentation graphique est la courbe en cloche dite de Gauss.

Le calcul des paramètres du PERT probabiliste se fait en trois étapes :

La première étape consiste à déterminer la loi de probabilité attachée à chaque tâche. En pratique on retient souvent une loi de distribution Bêta, c'est-à-dire qu'on est capable de donner trois estimations :

1. t_{opt} = estimation optimiste de la durée, c'est le temps minimal possible, si tout se déroule mieux que prévu ;
2. t_{pes} = estimation pessimiste de la durée, c'est le temps maximal possible, si tout se déroule au plus mal (hors catastrophe) ;
3. t_{vrai} = estimation vraisemblable de la durée, c'est la valeur que l'on donnerait si on devait n'en donner qu'une seule.

La deuxième étape consiste à calculer trois valeurs pour chaque tâche i :

- *sa durée probable* $t_{prob}(i)$: c'est une moyenne statistique, c'est le temps moyen de la tâche si elle était répétée un grand nombre de fois ;
- *sa variance* $v_{(i)}$ *et son écart-type* $e_{(i)}$: plus les estimations optimistes et pessimistes sont éloignées, plus elles présentent d'incertitude. C'est la variance qui mesure cette incertitude. Si elle est faible, l'estimation de la durée probable de la tâche sera assez précise.

$$t_{prob}(i) = \frac{t_{opt}(i) + 4t_{vrai}(i) + t_{pes}(i)}{6}$$

$$e(i) = \frac{t_{pes}(i) - t_{opt}(i)}{6}$$

$$v_{(i)} = e_{(i)}^2$$

La troisième étape consiste à calculer pour chaque chemin :

- sa durée estimée D_{est} ;
- sa variance estimée V_{est} ;
- son écart-type estimé E_{est}.

$$D_{est} = Somme\{t_{prob(i)}\} \text{ pour toutes les tâches i du chemin}$$

$$V_{est} = Somme\ \{e_{(i)}^2\} \text{ pour toutes les tâches i du chemin}$$

$$E_{est} = (V_{est})^{1/2}$$

Comme la durée du chemin obéit à une loi normale, on peut utiliser la table de Gauss (figure 4.28) pour obtenir la durée D_p du chemin avec une probabilité p :

$$D_p = D_{est} + G(p)\ E_{est}$$

La durée estimée du projet est probable à 50 % (car G(50) = 0).

Probabilité = p	G(p)	Probabilité = p	G(p)	Probabilité = p	G(p)
99,9	3,00	90	1,28	42,1	– 0,2
99	2,31	89,1	1,23	34,5	– 0,4
98	2,06	85,1	1,04	27,4	– 0,6
97	1,88	80,2	0,85	21,2	– 0,8
96	1,75	75,2	0,68	15,9	– 1
95	1,65	70,2	0,53	11,5	– 1,2
94,1	1,56	65,2	0,39	8,1	– 1,4
93,1	1,48	60,3	0,26	5,5	– 1,6
92,1	1,41	55,2	0,13	3,6	– 1,8
91	1,34	50	0,00	2,3	– 2

Figure 4.28 — Extrait de la table de Gauss.

Soit par exemple un chemin de durée estimé D_{est} = 100 et d'écart-type estimé E_{est} = 15. On peut en déduire soit une durée probable pour une probabilité donnée, soit une probabilité d'achèvement dans un délai donné.

Ainsi,

La durée probable à 90 % est :

$D90 = 100 + G(90) \times 15$

$D90 = 100 + 1{,}28 \times 15 = 119$ jours.

La durée probable à 60 % est :

$D60 = 100 + G(60) \times 15$

$D60 = 100 + 0{,}26 \times 15 = 104$ jours.

La probabilité de terminer en 80 jours est :

$80 = 100 + G(p) \times 15$

d'où : $G(p) = -20/15 = -1{,}33$

d'où : la probabilité est comprise entre 9 et 10 %.

On peut donner une mesure du risque par le ratio :

$$R = \frac{t_{pes} - t_{opt}}{t_{pes}}$$

et considérer que :

$R < 0{,}25$: risque faible

$0{,}25 < R < 0{,}5$: risque moyen

$R > 0{,}5$: risque fort

Le PERT probabiliste est intéressant pour les projets à forte incertitude, particulièrement sur le chemin critique, ainsi que sur les chemins ayant les plus forts risques.

Si l'incertitude est faible, la différence $[t_{pes} - t_{opt}]$ est faible par rapport à la durée estimée de la tâche. Dans ce cas, les variances des tâches seront faibles, de même que la variance du chemin, donc E_{est}. On n'a donc pas besoin d'un PERT probabiliste. On peut se contenter d'un graphe de la méthode des antécédents.

4.8 LA PLANIFICATION DES DÉLAIS DANS LES MÉTHODES AGILES

Les techniques de planification sont probablement, de toutes les techniques, celles qui ont été les plus utilisées dans les projets. En particulier, le diagramme de Gantt, avec plus ou moins de sophistication, est connu de tous les chefs de projet.

Cependant, les méthodes agiles présentent trois principales particularités dans la planification des délais : la négociation des délais, la planification sélective et l'utilisation de la technique de la *timebox*.

1. Tout d'abord, au niveau du cycle de vie complet du projet, la date de livraison finale ainsi que celle des principaux jalons — les itérations de livraison dans XP présentées au § 2.8.3 — sont négociées avec le client/commanditaire. Le nombre et la durée des itérations de développement en dépendent. Le respect de ces dates est ensuite impératif. Cela entraîne souvent l'utilisation d'un tampon (cf. § 4.6.2) avant ces échéances-clés.
2. Ensuite, les tâches confiées aux développeurs dans une itération de développement ne font pas l'objet d'une planification en bonne et due forme. Il est plutôt considéré que la réalisation de ces tâches dans la durée planifiée du cycle relève de l'engagement de l'équipe. Les membres, selon le Manifeste Agile, sont incités à réfléchir régulièrement sur de possibles augmentations de l'efficacité collective et à modifier leurs comportements en conséquence.

   Cette planification sélective, laissant une place à l'auto-organisation, est bien représentée par le schéma de déroulement de la méthode SCRUM donné par K. Schwaber[1] (figure 4.29) dont le cycle de vie a été présenté au § 2.8.4. En dehors des itérations (les *sprints*), le travail peut et doit être planifié rigoureusement. En revanche, compte tenu à la fois de la durée des itérations et de l'impératif d'ajustement, il serait lourd et inutilement contraignant de faire une planification détaillée de toutes les tâches. Au chapitre 7, on indique les moyens utilisés pour piloter l'avancement.

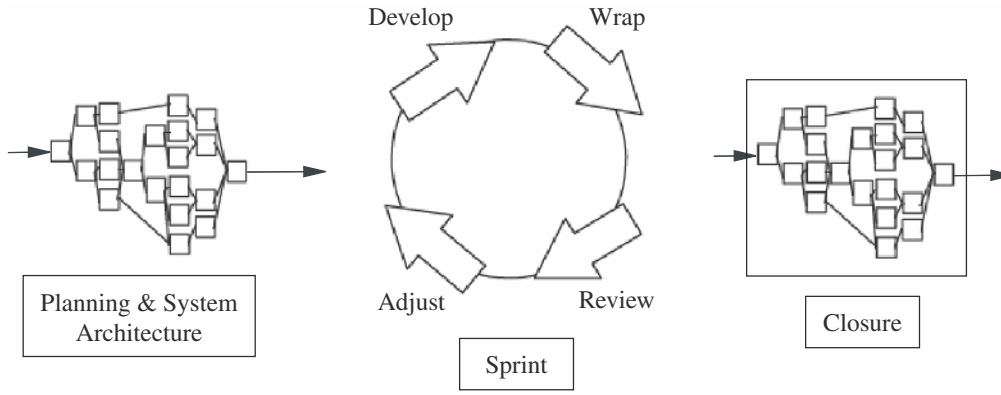


Figure 4.29 — Phases soumises à la planification dans SCRUM selon K. Schwaber.

1. Ken Schwaber, SCRUM Development process, http://jeffsutherland.com/oopsla/schwapub.pdf

3. Enfin, les itérations sont soumises à la technique de la *timebox*.

   La maîtrise du délai requiert la fixation d'une date limite, surtout lorsque le projet comporte une marge d'indétermination élevée sur le résultat à fournir. Le principe de la technique timebox est d'établir une enveloppe temps, de durée fixe et limitée, qui ne doit être dépassée sous aucun prétexte, pour éviter des glissements dont on ne maîtriserait pas l'ampleur. On peut la considérer comme une contrepartie à la possibilité offerte aux utilisateurs de demander des changements tout au long du projet. En effet, si l'enveloppe temps doit être impérativement respectée, cela impose des ajustements négociés sur le périmètre fonctionnel. Certaines fonctions ou améliorations sont donc abandonnées ou repoussées dans une version ultérieure.

   Le principe de la timebox fut élaboré après une expérience remarquée dans l'entreprise chimique DuPont de Nemours en 1989[1]. La Division Fibres était en train de passer à un système hautement automatisé, ce qui obligeait à réécrire toutes les applications de gestion de production, soit 100 000 lignes de code. Le projet avait été planifié sur deux ans par la Direction Informatique, alors que la Direction générale voulait que le changement se fasse en moins de six mois. Après discussion entre Direction Informatique et Direction de Production, on convint de construire un système robuste, mais sommaire, qui évoluerait ensuite par versions. Toutes les applications devaient être développées en 90 jours.

   La réussite de cette pratique, transformée en technique sous le nom de timebox, conduisit d'autres entreprises à l'utiliser. La plupart des méthodes agiles l'ont adoptée pour leurs phases de Construction.

   Les durées sont calées sur la capacité de l'esprit humain à se représenter l'échéance et d'une certaine façon à déclencher intérieurement un compte à rebours. C'est pourquoi l'enveloppe temps allouée est en général comprise entre deux et quatre mois, le nombre recommandé d'itérations variant entre trois et neuf, chacune durant entre deux et trois semaines.

1. J. Martin en fait l'historique dans *Rapid Application Development*, Macmillan, 1991.

5

La dimension humaine d'un projet

Nous allons dans ce chapitre nous intéresser plus particulièrement à la dimension humaine et relationnelle des projets : nous examinerons d'abord les bases de l'organisation du travail, c'est-à-dire la division et la coordination du travail ; nous traiterons ensuite de la participation des utilisateurs ; puis nous préciserons le rôle du chef de projet, particulièrement dans le management d'équipe et la gestion des conflits ; nous donnerons des éléments sur la gestion du changement lié à un nouveau système d'information. Nous terminerons par les particularités du management des ressources humaines et de la communication dans un contexte de méthode agile.

5.1 L'ORGANISATION DU TRAVAIL

5.1.1 La division et la répartition du travail

Tout projet d'une certaine taille fait l'objet d'un découpage en travaux à exécuter : ce point a été développé au chapitre 2. Les tâches ainsi identifiées peuvent parfois être exécutées simultanément : ces possibilités sont mises en évidence par le graphe des antécédents et le diagramme de Gantt, présentés au chapitre 4.

La répartition du travail entre des personnes différentes vise plusieurs objectifs. Elle permet de raccourcir la durée du projet. Quand on développe un système d'information, le délai est souvent critique. Si, par exemple, la réglementation bancaire autorise le lancement d'un nouveau produit d'épargne, que l'on prévoit attractif, il faut pouvoir disposer au jour j du système d'information permettant

la gestion et la souscription : les banques qui se positionnent les premières emporteront la plus grande part du marché. La répartition autorise le partage de compétences spécialisées. Certaines personnes peuvent intervenir ponctuellement ou partiellement sur plusieurs projets. Elle répartit les risques liés aux personnes. En cas de défaillance d'un acteur, il faut pouvoir poursuivre le projet. C'est plus facile si toute la connaissance du projet n'est pas concentrée sur la personne défaillante. Enfin, la répartition garantit une diversité de points de vue, chaque acteur apportant sa propre ouverture.

Les inconvénients de la division du travail dans le cadre d'un projet ne sont pas aussi graves que ceux qui ont été abondamment analysés pour les travaux répétitifs de l'organisation taylorienne. La participation à un projet, en effet, n'est jamais identique à la précédente : ennui et démotivation sont donc moins à craindre.

Quels sont les critères de répartition du travail ? Globalement, on retrouve les deux principes de base : spécialisation ou polyvalence. La spécialisation consiste à confier à une même personne des tâches d'une seule nature. Par exemple, l'élaboration des modèles, les études techniques, la constitution d'un jeu d'essai, l'écriture de programmes… On augmente ainsi la productivité. On limite les risques liés à la défaillance d'un acteur : le remplacement sur des tâches bien circonscrites est généralement plus aisé. La polyvalence consiste à confier à une seule personne un sous-ensemble cohérent de tâches diverses, conduisant souvent à l'élaboration d'un produit livrable. C'est ce que l'on appelle parfois un « lot ». La polyvalence favorise l'apprentissage des acteurs en élargissant leur palette de compétences. Elle minimise la charge de mise au courant du dossier. Elle réduit le besoin de coordination, puisque le lot entier est pris en main par une seule personne.

Les choix de répartition s'appuient également sur les caractéristiques propres aux individus : expériences, souhaits, disponibilités…

5.1.2 La coordination du travail

La division et la répartition du travail génèrent un besoin de coordination. Le travail peut être réparti dans le temps : il faut alors coordonner les phases et étapes successives. À un moment donné, il faut éviter redondances ou incohérences entre plusieurs sous-projets ou à l'intérieur d'un projet entre les différentes personnes de l'équipe.

Le but de la coordination est d'intégrer les différentes activités et les différents participants pour atteindre les objectifs du projet global.

Il y a deux sortes de coordination, personnelle ou impersonnelle, qui peuvent chacune prendre différentes formes.[1]

1. H. Mintzberg, *Structure et dynamique des organisations*, éd. Organisation, 1982.

La coordination *personnelle* repose sur les personnes :

On parle d'*ajustements mutuels* quand la coordination est assurée par des échanges informels lors de contacts directs entre les individus, par exemple, des demandes ponctuelles d'information entre concepteurs.

On parle de *supervision directe* lorsqu'une personne assure la liaison : il s'agit d'une fonction d'encadrement qui revient souvent au chef de projet. La supervision directe vise généralement à contrôler l'avancement des travaux et à s'assurer de leur convergence et cohérence.

La coordination *impersonnelle* repose sur un dispositif :

La *standardisation des procédés* consiste à imposer les mêmes procédés à tous pour réduire les risques d'incohérence, faciliter l'intégration et la compréhension mutuelle. L'utilisation de méthodes communes (techniques de modélisation, modèle de développement, démarche d'évaluation des charges) est un exemple de standardisation des procédés.

La *standardisation des résultats* consiste à établir des normes, visant à donner une forme standard à la présentation des résultats. Par exemple, les normes de documentation (plan-type, préimprimé) ou les normes de présentation des interfaces homme-machine et des sorties du futur système.

La *standardisation des qualifications* doit assurer des comportements analogues. Elle peut être obtenue par une politique recrutement de personnels ayant une culture commune ou par des formations méthodologiques.

À ces deux sortes de coordination, il faut ajouter : les *mécanismes de liaison*, (figure 5.1) qui permettent d'établir une coordination latérale. Le *comité de pilotage* du projet en est un exemple : il est composé de personnes n'appartenant pas à l'équipe de projet qui donnent les grandes orientations ; leur angle de vue est celui du projet dans son ensemble, même si le travail a été réparti en plusieurs sous-projets.

Personnelle	**Impersonnelle**
Ajustement mutuel	Standardisation des procédés
Supervision directe	Standardisation des résultats
	Standardisation des qualifications
Mécanismes de liaison	
Comité de pilotage	Administration de données

Figure 5.1 — Formes de coordination.

On distingue parfois le comité de pilotage client et le comité de pilotage interne [AFITEP, 2000]. Le premier est défini comme un « *groupe de personnes responsables, chez le client, jouant le rôle de maître d'ouvrage vis-à-vis du chef de projet* ». Le second vise à apporter une aide au chef de projet, dans certains projets difficiles : « *composé de responsables de niveau suffisant, il prendra, face au client, les grandes options stratégiques et les décisions correspondantes* ».

Un dispositif de liaison joue un rôle particulièrement important dans les projets de développement de système d'information : il s'agit de *l'administration de données*. Nous allons en développer les différents aspects.

5.1.3 L'administration de données (AD)

Cette fonction a pour mission de construire et gérer un système d'information sur le système d'information de l'entreprise, souvent appelé *référentiel*. Elle peut assurer la coordination à l'intérieur d'un projet : c'est alors un dispositif temporaire. Si elle est au contraire permanente, elle permet de coordonner différents projets, et fonctionne également en dehors de tout projet. On distingue quatre formes d'AD :

- l'AD technique ;
- l'AD projet ;
- l'AD coordination ;
- l'AD pilotage.

L'*AD technique* gère des descriptions proches de la structure des données informatiques. Ces descriptions font partie de la documentation des applications. Si ces informations sont absentes, ou bien n'ont pas intégré les évolutions de l'application, on peut les produire au moyen d'une opération de rétro-documentation. Le champ couvert est circonscrit à celui d'une application. Il est parfois élargi dans le cas d'une base de données partagée par plusieurs applications. Les outils utilisés sont des dictionnaires de données techniques, soit liés à un système de gestion de base de données (par exemple, le dictionnaire DB2 ou le dictionnaire Oracle), soit liés à un atelier de génie logiciel générateur de code.

L'*AD projet* gère les données spécifiques d'un projet, telles qu'elles sont définies au niveau sémantique. La structure n'est intéressante que par le sens dont elle est porteuse. Ainsi, l'identification des propriétés d'une entité permet de préciser sa signification et son rôle dans le système d'information. Cette forme d'administration de données, que l'on pourrait appeler plus justement administration des informations, découle du besoin de concevoir les applications à un niveau proche du système de gestion, en y associant les gestionnaires. Elle accompagne la conception et assure la coordination entre tous les intervenants du projet. On utilise pour cela les dictionnaires liés aux ateliers de génie logiciel

de conception. L'articulation entre dictionnaire des données physiques (celles de l'AD technique) et dictionnaire des données conceptuelles n'est pas toujours mise en place. Certaines opérations de rétroconception ont pour objectif de reconstruire *a posteriori* ce lien.

L'*AD coordination* gère des informations conceptuelles consolidées. Les données y sont définies au même niveau que dans l'AD projet et avec la même maille de description, mais le champ couvert est multiprojet. Ainsi, dans le cas d'une refonte d'une partie importante du système d'information de l'entreprise, l'AD coordination mettra en cohérence les définitions des données issues de chaque sous-projet. On peut aussi installer une AD coordination pour éviter des divergences entre les différentes études détaillées issues d'une même étude préalable. Ou encore, pour fédérer tous les projets d'un département. L'apport de l'AD coordination est de réduire les phénomènes d'autisme (« je ne parle avec personne ») ou de babélisation (« on se cause, mais on ne se comprend pas ») de certaines applications. En soulevant les barrières des projets, elle étend le champ de la réutilisation.

L'*AD pilotage* fournit des représentations synthétiques, supports d'une planification stratégique des systèmes d'informations. C'est par exemple à ce niveau que l'on réfléchira sur les liens Contrat-Client (contrat multiclient) ou les liens Contrat-Produit (contrat multiproduit). Sa maille de description dépend du niveau auquel le besoin de pilotage se situe. Ce peut être un département, une filiale, toute l'entreprise, en fait tout sous-système quasi autonome ayant une capacité de pilotage.

Chaque forme représente une réponse complète à un type de problème. Dans une perspective dynamique, certaines contraintes d'enchaînement doivent être néanmoins respectées (figure 5.2).

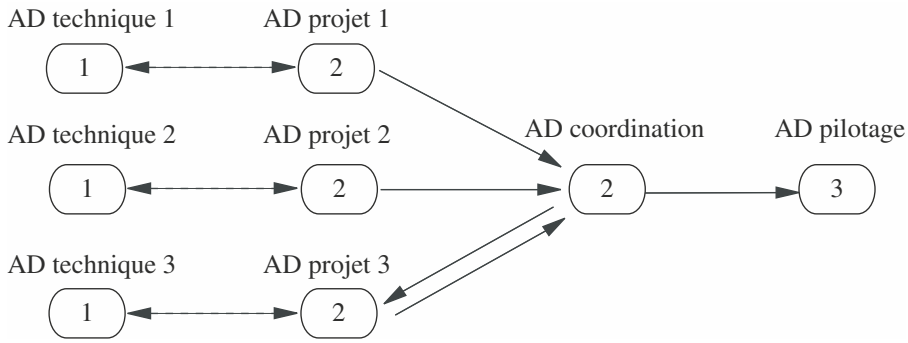


Figure 5.2 — Enchaînement des formes d'administration de données.

Les formes 1 et 2 sont indépendantes. En effet, l'AD projet peut être suivie ou non d'une AD technique. Inversement, l'AD projet peut être précédée d'une AD technique, par exemple si le projet est la refonte d'une application existante, sur laquelle on mène une opération de rétrodocumentation.

Par contre, l'AD pilotage doit être précédée d'une AD coordination stabilisée. De façon analogue, l'AD coordination doit pouvoir s'appuyer sur des AD projet solides. Ensuite, elle alimente partiellement l'administration des nouveaux projets.

L'AD pilotage peut être mise en place sur une partie seulement de l'entreprise, pour laquelle on est déjà en forme 3.

La plupart des entreprises sont aujourd'hui en forme 2 ou 3.

5.1.4 Les parties prenantes et les structures-types d'un projet

Longtemps, on s'est principalement intéressé aux acteurs d'un projet, c'est-à-dire à ceux qui ont des responsabilités vis-à-vis du travail à effectuer. Aujourd'hui, on élargit la perspective à l'ensemble de ceux qui sont concernés ou peuvent être impactés par le projet : c'est ce que l'on appelle les *parties prenantes*. Nous proposons au chapitre 9, dans le cadre de la maîtrise des risques, un exercice d'analyse des parties prenantes d'un projet (§ 9.2). Nous allons maintenant approfondir la notion d'acteur.

Un acteur est quelqu'un qui joue un *rôle* dans le déroulement du projet. La notion de rôle est importante dans un univers projet. C'est une fonction temporaire, déconnectée de la place qu'occupe la personne dans l'organigramme de l'entreprise ; on attache au rôle des activités à effectuer et une responsabilité.

Quand on organise un projet, on commence par déterminer les rôles nécessaires. On distingue trois types d'acteurs (figure 5.3) participant à un projet de développement de système d'information :

- le couple maître d'œuvre-maître d'ouvrage ;
- l'équipe de projet ;
- les utilisateurs.

Typologie des acteurs
Couple maître d'œuvre-maître d'ouvrage
Équipe de projet chef de projet concepteur développeur
Utilisateur final gestionnaire décideur

Figure 5.3 — Typologie des acteurs d'un projet système d'information.

Les rôles de *maître d'œuvre* et *maître d'ouvrage* sont issus des grands projets industriels. Ils sont liés par une relation contractuelle. Dans le domaine des systèmes d'information, les modalités de cette relation ont fait l'objet d'un projet européen, *Eurométhode* : nous en présentons les résultats au chapitre 16. Le maître d'ouvrage représente le client. Partant d'une demande des futurs utilisateurs du système, il établit un cahier des charges qui peut servir de base à un appel d'offres. Après discussions sur une ou plusieurs propositions reçues, il passe contrat avec un fournisseur qui jouera le rôle de maître d'œuvre. Celui-ci est responsable de la conduite du projet. Le maître d'ouvrage assure un suivi de l'avancement du projet, selon des modalités contractuellement définies. À chaque fourniture contractuelle par le maître d'œuvre, il procède à la recette et prononce l'acceptation ou le refus. Il pilote la mise en œuvre du projet.

Rigoureusement parlant, l'ouvrage est le résultat concret d'un projet : construction, matériel, progiciel. L'AFITEP et l'AFNOR [AFITEP, 2000] définissent le maître d'ouvrage comme « *la personne physique ou, le plus souvent, personne morale qui sera le propriétaire de l'ouvrage. Il fixe les objectifs, l'enveloppe budgétaire et les délais souhaités pour le projet* ». À ce titre, il « *assure le paiement des dépenses liées à la réalisation* ».

L'œuvre est définie comme « *le processus de réalisation de l'ouvrage, c'est-à-dire la mise en place des moyens nécessaires à cette réalisation et à leur conduite* ». Par conséquent, elle « *est constituée de l'ensemble des tâches, regroupées ou non en lots de travaux* ».

Le maître d'œuvre est « *la personne physique ou morale qui réalise l'ouvrage pour le compte du maître d'ouvrage et qui assure la responsabilité globale de la qualité technique, du délai et du coût* ».

La transposition de ces deux rôles quand il s'agit de système d'information renvoie davantage à la répartition des responsabilités qu'à la notion de propriété. En effet, même si l'on parle parfois du « p*ropriétaire d'une application informatique* », il est difficile d'étendre ce droit de propriété aux processus et aux acteurs.

L'*équipe de projet* rassemble différents acteurs, qui sont affectés au projet. D'après l'AFITEP et l'AFNOR, c'est « *l'ensemble des personnes placées sous l'autorité directe (et quelquefois indirecte) du chef de projet* ». Mais, on peut parfois considérer que l'équipe de projet s'étend à « *toutes les personnes participant à la réalisation du projet* ».

Le *chef de projet* est responsable devant le maître d'œuvre de l'avancement du projet. Nous décrivons plus précisément son rôle et ses outils au paragraphe 5.3 et au chapitre 7 qui traite du pilotage du projet. Si l'on a découpé le projet en plusieurs sous-projets, un chef de projet est responsable de l'ensemble du projet et chacun des sous-projets a son propre responsable.

Le rôle de *concepteur* peut être tenu par un informaticien, un organisateur ou un gestionnaire selon le stade d'avancement : sa responsabilité est de concevoir le futur système aux étapes étude préalable et étude détaillée.

Le rôle de *développeur* est tenu par un informaticien : sa responsabilité est d'écrire les programmes ou de réaliser un prototype. Pour certains développements réalisés en langage de 4e génération, le rôle peut être tenu par un gestionnaire.

Concepteurs et développeurs sont les producteurs de l'équipe. Ils sont assistés par d'autres acteurs occupant des rôles fonctionnels.

La figure 5.4 donne un exemple de structure de projet.

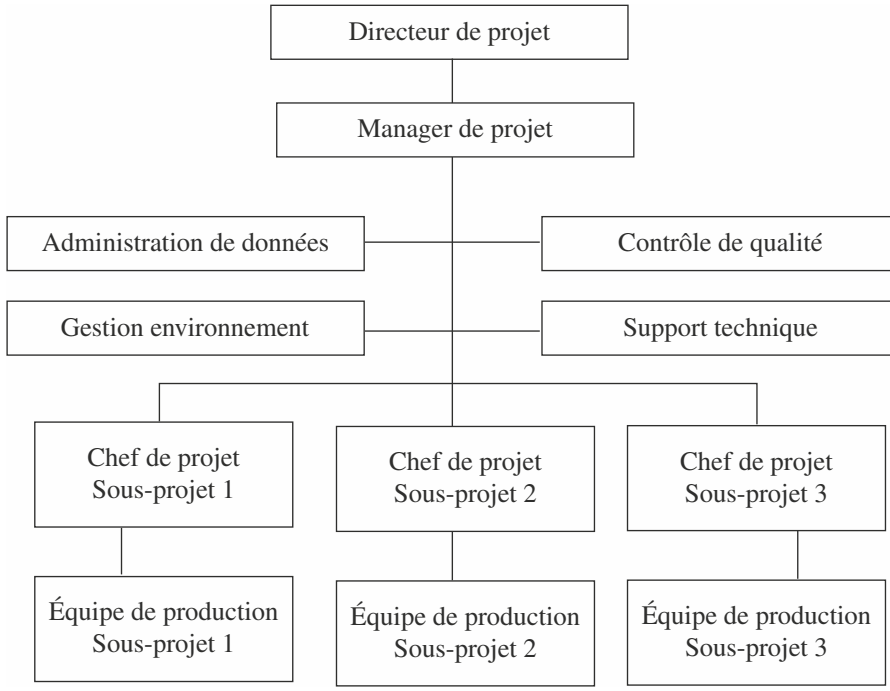


Figure 5.4 — Exemple de structure de projet.

On a dédoublé le rôle du chef de projet. Le directeur de projet est responsable des relations contractuelles avec le client. Le manager est responsable de l'avancement du travail. Le premier s'occupe des affaires extérieures, le second des affaires internes.

Par ailleurs, on voit figurer quatre rôles fonctionnels. La fonction administration de données correspond à l'AD projet présenté ci-dessus. Le support technique apporte ponctuellement conseil et assistance aux équipes de production sur une technologie ou une méthode employée. Le gestionnaire de l'environnement est responsable de l'atelier de génie logiciel utilisé ; parfois, c'est lui prépare les environnements de test et de recette. La fonction contrôle qualité s'assure de la

qualité de la production. Ce point sera détaillé au chapitre 8 qui traite de la qualité.

Le rôle de l'*utilisateur* peut être mis en correspondance avec des fonctions permanentes. On va repérer des acteurs qui vont jouer pour le projet les rôles des utilisateurs, qu'ils soient ou non les véritables utilisateurs du futur système. Ils auront la responsabilité de s'exprimer en tant qu'utilisateurs. On peut distinguer trois catégories.

L'*utilisateur final* est celui qui va utiliser les transactions et les éditions du futur système dans son travail au quotidien. C'est, par exemple, l'employé de l'agence bancaire qui se sert de son terminal pour effectuer les opérations avec la clientèle. La responsabilité attachée à ce rôle est d'exprimer des besoins et des contraintes liées au travail courant.

L'*utilisateur gestionnaire* est un opérationnel qui a des responsabilités d'encadrement. Il utilisera le futur système pour obtenir des informations permettant de prendre des décisions de gestion. Par exemple, un responsable produit attendra des informations permettant de suivre une vente promotionnelle. La responsabilité attachée à ce rôle est d'exprimer des besoins en informations favorisant la gestion à moyen terme de l'activité.

L'*utilisateur décideur* a le pouvoir de modifier les règles du système de gestion, pour utiliser au mieux les possibilités technologiques. Sa responsabilité est de prendre des décisions sur l'évolution des règles de gestion.

5.1.5 La composition de l'équipe de projet

On a évoqué au paragraphe 5.1.1 les critères de répartition du travail à l'intérieur d'une équipe de projet. Nous allons approfondir ce point sous l'angle des profils psychologiques, à partir de la méthode des rôles de M. Belbin[1]. Ce dernier a mené pendant plus de 20 ans à Cambridge des études sur le fonctionnement des groupes de travail et a conclu qu'il n'existe qu'un nombre limité de positionnements dans l'équipe. La performance d'une équipe dépend de l'équilibre de ces positionnements, appelés « rôles », parmi les membres. Au-delà de la répartition des activités, chacun endosse en général un « rôle », et des « rôles » complémentaires favoriseront la synergie, alors que des « rôles » non endossés manqueront dans la dynamique du groupe.

M. Belbin a identifié neuf types de « rôles » regroupés en trois catégories, selon l'orientation majeure : vers les personnes, vers l'action ou vers la production intellectuelle.

1. Meredith Belbin, *Les rôles en équipe*, Ed. Organisation, 2006.

Les « rôles » orientés vers les personnes sont les suivants :

1. Le « coordinateur a tendance à clarifier les buts et favoriser la prise de décision dans le groupe.
2. Le « coéquipier » est attentif aux autres et les encourage.
3. Le « pourvoyeur de ressources » dispose d'un carnet d'adresses qu'il est prêt à ouvrir pour les membres du groupe.

Les « rôles » orientés vers l'action sont les suivants :

1. L'« organisateur » a tendance à traduire les idées en tâches concrètes.
2. L'« entraîneur » aide l'équipe à conserver le cap vers les objectifs et maintient la pression.
3. Le « contrôleur » vérifie les détails, il a le souci du respect du calendrier et des engagements.

Les « rôles » orientés vers l'intellect sont les suivants :

1. L'« innovateur » propose de nouvelles idées, mais il peut être irréaliste et peu communicateur.
2. L'« évaluateur » est au contraire réaliste et il aime analyser les problèmes complexes.
3. Le « spécialiste » possède des connaissances précises, utiles à l'équipe.

La méthode Belbin s'appuie sur une série de questionnaires traités par un système expert, INTERPLACE. Avoir recours à la méthode Belbin peut être intéressant lorsque l'on constate que, sans raison apparente, une équipe ne fournit pas les résultats escomptés. Si l'on a identifié les rôles endossés par chacun, on peut augmenter l'efficacité de l'équipe en jouant, à bon escient, sur sa composition ou sur son fonctionnement interne.

5.2 LA PARTICIPATION DES UTILISATEURS

Quand on évoque l'intervention d'utilisateurs dans un projet, on emploie souvent le terme générique de *participation*. En réalité, la nature et les modalités de participation peuvent présenter une grande variété. On ne peut pas parler de participation dans l'absolu ; elle peut prendre différentes formes et servir différents buts. Dans une perspective de pilotage de projet, on considère que la participation peut avoir un effet sur la détermination des besoins, sur le processus de décision ou sur une évolution organisationnelle.

5.2.1 La détermination des besoins

La participation peut avoir pour objectif de transférer à l'équipe de projet une connaissance sur un domaine, un système de gestion, une organisation existante. Ceci est en général facile à obtenir. Cependant, les utilisateurs sollicités sont ensuite dans un état d'attente : ils savent qu'un projet a démarré, ils expriment davantage qu'un existant, mais également des remarques, critiques, propositions, souhaits…

Si on veut limiter ces effets, il est préférable de réduire le nombre d'informateurs, voire de chercher un informateur externe. Si au contraire on veut déstabiliser, c'est-à-dire provoquer un besoin, il faut élargir la consultation, avec cependant la nécessité de gérer ensuite cette déstabilisation.

5.2.2 La prise de décision

La participation peut faciliter le processus de décision de plusieurs façons. Tout d'abord, supposons que le système de gestion doive être pensé ou repensé. Il s'agit de faire des choix importants de conception, c'est-à-dire qui auront un impact sur le futur système d'information. La participation des utilisateurs est nécessaire, parce que ce sont souvent des choix stratégiques qui ne relèvent pas du domaine des informaticiens.

Ensuite, tout projet est jalonné de prises de décision lors d'activités plus ou moins formalisées de validation. Mais les décisions sont parfois difficiles à obtenir. La participation peut faciliter l'obtention de décisions sur des options de conception et d'organisation. Il s'agit de faire en sorte que les choix soient faits par ceux qui mettront en œuvre le futur système d'information. Souvent, un ou deux utilisateurs sont intégrés au groupe de conception et un groupe plus large est sollicité pour avis et validation. Si les utilisateurs sont multiples, la prise de décisions passe par la construction d'un consensus.

Enfin, la participation peut rechercher des décisions d'orientation et de poursuite du projet. Les utilisateurs visés sont des décideurs, ils ne s'impliqueront pas dans les choix fins de conception, mais la poursuite du projet pourra dépendre de leur appréciation et de leur engagement. Cette participation se fait généralement à travers un comité de pilotage, mais elle peut être plus informelle. Prendre une décision, c'est souvent prendre un risque. Celui-ci est d'autant plus grand que les conséquences du choix se font sentir à moyen ou long terme. C'est pourquoi on observe parfois une réticence à s'engager. La participation permet de réduire cette crainte, en donnant aux décideurs une meilleure connaissance du projet et en offrant des possibilités d'échange et de discussion avec d'autres acteurs.

5.2.3 Le changement

La participation peut faciliter l'utilisation du futur système par l'utilisateur final. C'est un moyen pour que les changements soient acceptés positivement par ceux qui auront à les vivre. Cette participation n'a généralement pas d'impact sur la conception.

De plus, si le futur système peut être contourné, il faut obtenir l'engagement de l'utilisateur, pour qu'ensuite il l'alimente et en utilise les sorties possibles. Cet engagement sera obtenu s'il a participé au projet. Comme la totalité des utilisateurs finaux ne peut en général pas être associée pour des raisons pratiques, coût de la participation et difficulté de faire converger des demandes nombreuses, on s'appuie souvent sur un groupe phare, dont le comportement aura un effet exemplaire.

La participation peut enfin provoquer un changement de comportement des utilisateurs. L'association aux tâches de production et aux décisions peut modifier les comportements, soit entre groupes d'utilisateurs (par un dialogue autour du futur système), soit en rendant possible une autre organisation.

5.3 LE RÔLE DU CHEF DE PROJET

5.3.1 Définition du chef de projet

Le chef de projet est défini par l'AFITEP et l'AFNOR comme : « *la personne physique chargée par le maître d'œuvre, dans le cadre d'une mission définie, d'assurer la maîtrise du projet, c'est-à-dire de veiller à sa bonne réalisation dans les objectifs de technique, de coût et de délai.* »

Cette définition est insuffisante pour les projets système d'information. Heureusement, la définition est élargie : « *Souvent le maître d'ouvrage identifie, en son sein, un interlocuteur au chef de projet du maître d'œuvre. Cet interlocuteur est aussi appelé chef de projet* » [AFITEP, 2000].

Compte tenu de la définition d'un système d'information donnée au paragraphe 1.4, le chef de projet devrait toujours être du côté maître d'ouvrage, avec un chef de projet informatique comme interlocuteur.

5.3.2 Les responsabilités du chef de projet

Le chef de projet doit faire en sorte que le projet réussisse. Pour cela, sa responsabilité est multiple (figure 5.5) : il a la charge d'une bonne gestion du groupe et des individus ; il pilote la production des livrables pour un achèvement en temps et en heure ; il participe à la gestion du changement, en particulier auprès des acteurs clés du projet ; il lui revient enfin de veiller à ce que les processus de décision ne soient pas bloqués.

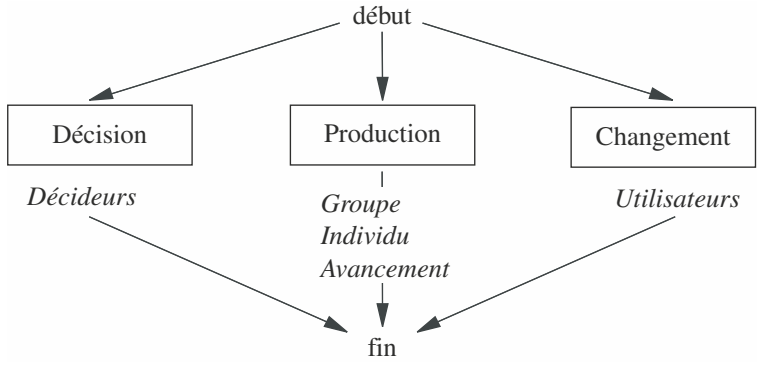


Figure 5.5 — Responsabilités du chef de projet.

Il est *responsable du groupe*. D'un ensemble d'individus, il doit constituer une équipe, l'animer et en maintenir la cohésion. L'équipe prend corps lorsque l'objectif (le projet) devient collectif. Pour obtenir cette adhésion et cette synergie, il faut que chacun soit — et se sente — responsable en propre d'une partie du travail ; il faut favoriser les échanges latéraux grâce notamment à des points réguliers de coordination. L'esprit de groupe se construit par des échanges directs d'individu à individu sur un but commun. C'est pourquoi il est souhaitable que la taille de l'équipe reste limitée. Au-delà de quinze à vingt personnes, le sentiment d'appartenance se dilue.

Le chef de projet est *responsable des individus* membres de l'équipe, qu'il doit valoriser et soutenir. Ce domaine est l'un de ceux où, par chance, le travail est souvent un moyen de se réaliser. Or, la satisfaction qu'un acteur retire de son activité sur le projet est le plus puissant des moteurs. Chercher à ce que chacun se sente gratifié par le travail accompli est une excellente approche managériale.

Le chef de projet doit prendre en compte les souhaits des individus et l'enrichissement apporté, lorsqu'il attribue les tâches. L'estimation de la charge affectée doit être négociée avec celui qui va accomplir la tâche correspondante. S'il y avait désaccord dès le départ, le risque serait grand de dépasser le délai.

L'avancement doit être suivi de près, pour ne pas laisser l'un ou l'autre piétiner. En cas de difficulté, le dialogue est indispensable. Un retard sur une tâche peut être dû à une cause exogène (panne de la machine de développement) ou organisationnelle (demande de fonctionnalité supplémentaire), qui est du ressort du chef de projet.

Le chef de projet est également *responsable de l'avancement des travaux* et doit convaincre l'équipe de la nécessité d'un suivi adéquat. Le système d'information projet est indispensable à la réussite : il faut que les chiffres de base soient le plus juste possible. Pour cela, le chef de projet doit utiliser avec perspicacité les chiffres issus du bilan individuel. Les informations du compte rendu d'avancement doivent être traitées immédiatement.

Le chef de projet est un *acteur du changement parmi les utilisateurs*. Il doit pour cela organiser une participation adaptée afin de créer un noyau qui portera le changement. Si aucune dynamique n'a été déclenchée parmi les utilisateurs, la mise en œuvre risque d'être difficile et le succès du projet compromis. Il appartient au chef de projet de repérer et d'associer au projet certains utilisateurs convaincus de son utilité et qui s'en feront les hérauts.

Le chef de projet est, à certains égards, le *pilote des décisions* touchant au contenu du projet. Toute décision à prendre, si elle n'est pas prise immédiatement, doit lui être signalée. Toutes celles qui sont en suspens doivent figurer sur son tableau de bord, pour qu'il en suive le traitement. Il doit parfois aider à choisir en apportant un éclairage aux décideurs ou en proposant des solutions flexibles si un choix définitif ne peut être fait.

Nous allons développer trois aspects qui concernent un chef de projet : d'abord, les styles de management d'équipe et leur adéquation ou inadéquation à différentes situations pour atteindre les objectifs énoncés ci-dessus ; ensuite, la motivation des membres de l'équipe ; enfin la gestion des conflits à l'extérieur du groupe de projet.

5.3.3 Les styles de management

Dans les travaux intellectuels, les écarts de productivité peuvent être très importants, aussi bien dans les études que dans le développement. Les méthodes d'analyse et les méthodes de programmation visent à réduire ces écarts qui peuvent aller couramment de 1 à 5. Cependant, le chef de projet peut avoir une influence décisive sur la productivité :

- d'abord en estimant les charges, c'est-à-dire en donnant une base de référence ;
- ensuite en assurant un suivi régulier de l'avancement ;
- enfin, en manageant l'équipe avec discernement.

Il n'existe pas un style de management universellement efficace. Le style le plus adéquat dépend à la fois de la personnalité du chef de projet et de la situation au moment où l'on agit (figure 5.6). Un style de management est adéquat s'il permet d'augmenter à la fois la productivité et la satisfaction de l'équipe de projet.

On distingue souvent cinq styles de management, selon le degré et la forme de participation des membres de l'équipe aux décisions de gestion du projet :

- le style directif ;
- le style consultatif ;
- le style participatif ;
- le style persuasif ;
- le style par délégation.

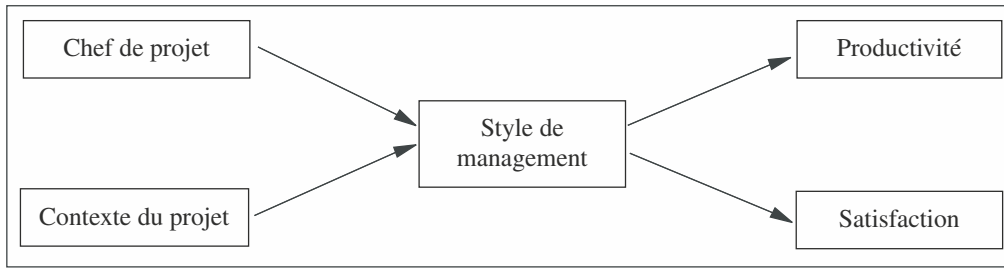


Figure 5.6 — La place du style de management.

Le *style directif* donne au chef de projet une place prépondérante dans la prise de décision : il décide seul de l'organisation du travail, des règles de fonctionnement et des solutions à apporter aux différents problèmes et aléas. On distingue parfois deux formes.

Si le chef de projet étudie les problèmes et prend seul les décisions, on parle du style *directif seul*.

Si le chef de projet recueille des éléments d'information auprès de l'équipe, puis décide seul, il s'agit du style *directif après information*.

Le *style consultatif* consiste à susciter l'avis des membres de l'équipe. Le chef de projet prend ensuite une décision qui n'est pas forcément le reflet des opinions du groupe. Comme le précédent, il peut prendre deux formes.

Si le chef de projet recueille les opinions en discutant individuellement avec chacun, on l'appelle style *consultatif individuel*.

Si le chef de projet recueille les avis de l'équipe lors d'une discussion de groupe, c'est le style *consultatif groupe*.

Le *style participatif* conduit le chef de projet à associer les membres de l'équipe aux décisions, notamment sur l'organisation du travail et la gestion des aléas. Cela demande au chef de projet des capacités d'écoute, de collaboration et de recherche de consensus. Ce style peut comme les précédents prendre deux formes.

Si le chef de projet associe individuellement chacun des membres et recherche une décision satisfaisante pour le plus grand nombre de personnes, on le nomme style *participatif individuel*.

Si le chef de projet réunit son équipe pour une recherche et une évaluation collective des différentes solutions, on parle de style *participatif groupe*.

Le *style persuasif* vise à augmenter l'adhésion des membres de l'équipe au projet. Le chef de projet passe un temps important à parler à l'équipe des objectifs du projet, de son importance, du rôle de chacun et de la pertinence de l'organisation et des décisions prises au cours du projet. Il suscite des réactions et cherche à convaincre.

Le *style par délégation* laisse une grande part à la responsabilité personnelle notamment dans la conduite des travaux à mener : le chef de projet fournit les informations à un membre de l'équipe et lui laisse complète initiative de traiter le problème. Ce style nécessite une confiance partagée, ce qui ne signifie pas aveuglement car il s'accompagne d'un système de suivi précis. Il favorise l'apprentissage et l'autonomie des membres de l'équipe.

Pour choisir un style adéquat lorsque l'on doit prendre une décision ou traiter un problème, il faut se poser les questions suivantes :

- La nature de la décision ou solution retenue peut-elle avoir un impact important sur les trois sommets du triangle projet : coût, délais, qualité ?
- Le chef de projet dispose-t-il des informations nécessaires pour décider ?
- Si le chef de projet décide seul, sa décision sera-t-elle acceptée par l'équipe ?
- L'acceptation de la décision par les membres de l'équipe peut-elle avoir un effet sur l'efficacité de sa mise en œuvre ?
- Peut-il y avoir des conflits entre les membres de l'équipe sur la décision à prendre ?
- Les membres de l'équipe peuvent-ils avoir des avis pertinents sur la décision à prendre ?
- Les contraintes de délais sont-elles fortes ?

Selon les réponses apportées à ces différentes questions, on est conduit à privilégier ou à écarter un style de management pour traiter le problème ou prendre la décision. Le tableau de la figure 5.7 résume les situations pour lesquelles il faut privilégier et/ou éviter chacun des styles de management. Le choix n'est pas forcément unique pour une situation donnée.

5.3.4 Les théories de la motivation

La motivation, comme disposition incitant l'individu à fournir sans contrainte une prestation satisfaisante en qualité et quantité, a fait l'objet de nombreuses théories depuis plus d'un demi-siècle. Nous allons les parcourir rapidement pour montrer les différents angles sous lesquels on peut l'aborder.

Une des théories les plus anciennes est celle de la *hiérarchie des besoins* d'A. Maslow[1]. Les êtres humains sont soumis à des besoins ordonnés : tant qu'une strate n'est pas satisfaite, l'individu ne peut pas se préoccuper des niveaux supérieurs.

1. Abraham Maslow, *Devenir le meilleur de soi-même : Besoins fondamentaux, motivation et personnalité*, Ed. Organisation, 2008. L'ouvrage initial, *Motivation and personality*, fut publié en 1954.

Style de management	À utiliser si	À éviter si
Directif seul	Les contraintes de délai sont très fortes. L'équipe est composée de débutants.	Le chef de projet manque d'information.
Directif après information	On est en situation de crise.	
Consultatif individuel	Le chef de projet manque d'information.	
Consultatif groupe		Il y a risque de conflit entre les membres de l'équipe.
Participatif individuel	Le collaborateur est expérimenté. Le collaborateur adhère aux objectifs du projet, mais il peut y avoir un conflit avec lui sur la décision.	L'implication du collaborateur a peu d'influence sur la mise en œuvre de la décision.
Participatif groupe	Les membres de l'équipe sont expérimentés. Les membres de l'équipe adhèrent aux objectifs du projet et ils pourraient être en désaccord entre eux sur la solution.	L'équipe manque d'information ou d'expérience. Les membres de l'équipe n'adhèrent pas pleinement aux objectifs du projet.
Persuasif	L'équipe est composite, avec des membres venant de départements ou d'entreprises différentes.	
Par délégation		Le collaborateur n'adhère pas pleinement aux objectifs. Le collaborateur est inexpérimenté.

Figure 5.7 — Le choix d'un style de management.

Il propose la classification croissante :

1. Les besoins physiologiques (faim, soif, santé..) doivent être assurés en premier.
2. Puis, l'individu recherche la sécurité (protection contre dangers, menaces…).
3. Ensuite apparaissent les besoins sociaux (, appartenance à un groupe, acceptation sociale, échanges…).
4. L'individu peut alors rechercher des activités et des situations qui lui permettent d'être reconnu et développent l'estime de soi.
5. Enfin, l'être humain cherche à se réaliser par l'exploration, la découverte, les connaissances…

Cette classification, bien qu'un peu sommaire, attire l'attention sur les multiples sources de motivation et sur la nécessité de faire une analyse de ce qui apparaît comme primordial à un moment donné.

La théorie des *facteurs d'hygiène* de F. Herzberg[1], un des théoriciens de l'enrichissement du travail en milieu industriel, a affiné l'approche hiérarchique. Il a proposé de distinguer entre « facteurs d'hygiène » et facteurs de motivation. L'analogie s'appuie sur le fait qu'un manque d'hygiène ouvre la porte à la maladie, mais une bonne hygiène n'est pas suffisante pour garantir la santé.

Les facteurs d'hygiène sont sources potentielles d'insatisfaction et sont liés au contexte du travail : règles administratives, salaire, conditions matérielles, relations avec l'encadrement… Les facteurs de motivation sont des sources potentielles de satisfaction, donc de motivation. Ils touchent au contenu du travail : responsabilité, réussite, promotion…

La *théorie X/Y* de D. McGregor[2] s'intéresse aux managers et au regard que ceux-ci portent sur l'être humain au travail. Les actes de management des personnes reposent sur des hypothèses, en général implicites, chez les managers. Il distingue ainsi deux styles de management.

Selon la « théorie X », l'être humain ressent une aversion pour le travail, a généralement peu d'ambition et évite responsabilités. Pour obtenir un travail satisfaisant, il faudra donc contraindre, menacer, contrôler…

Selon la « théorie Y », le travail peut, au contraire, être une source de satisfactions. L'être humain est capable aussi bien d'autonomie que d'apport créatif, il est donc recommandé d'utiliser ce potentiel d'engagement, notamment en associant le collaborateur aux objectifs poursuivis.

1. Frederick Herzberg, *Le travail et la nature de l'homme*, Entreprise moderne d'édition, 1978. L'ouvrage initial, *The motivation to work*, fut publié en 1959.
2. Douglas McDonald, *La dimension humaine de l'entreprise*, Bibliothèque du Management, 1970. L'ouvrage initial, *The human side of enterprise*, fut publié en 1960.

Sans réduire l'ensemble des managers à deux catégories, il peut être intéressant que le chef de projet s'interroge sur ses propres orientations de management individuel de l'équipe.

La *théorie Z* de W. Ouchi[1] est un clin d'œil à celle de McGregor. Il propose d'utiliser les principes d'organisation des entreprises japonaises, qui conduisent à une performance individuelle supérieure à celle des entreprises américaines. Les valeurs prônées sont orientées vers le groupe : inclusion totale de l'individu dans l'entreprise, primauté absolue au groupe, prise décision participative… Le principe défendu est que la motivation des individus dépend de leur intégration dans le corps collectif. Cette position était notamment liée à la pratique de l'emploi à vie dans les grandes entreprises japonaises, en partie remis en question aujourd'hui.

Plusieurs théories ont étudié le caractère relatif et subjectif de la satisfaction dans le groupe. Selon les théories de *l'équité*[2], le comportement (en quantité et en qualité) dépend des sanctions et récompenses attendues en regard du travail fourni, ainsi que de paramètres comme l'âge, l'expérience, le niveau d'éducation… Le point central est que l'individu élabore des comparaisons avec d'autres ou avec situations antérieures.

La perception d'une injustice, c'est-à-dire d'un déséquilibre entre sa contribution et le résultat obtenu (salaire, responsabilités, rapports sociaux dans le travail, avantages indirects…) est une source importante d'insatisfaction et de réactions négatives.

Dans la même veine, la théorie des *attentes* ou théorie de l'*expectation* de V.H. Vroom[3] a développé l'idée que tout individu élabore des attentes sur les résultats de son travail. Il propose des principes, dont l'application devrait conduire à ce que les motivations individuelles coïncident avec celles de l'organisation. En particulier, les règles de distribution des récompenses devraient être clairement comprises ; les récompenses sont en proportion des efforts nécessaires ; l'individu peut contrôler la partie de sa performance qui est liée aux récompenses ; et l'individu ne perçoit pas de conflit entre une récompense à court terme (liée par exemple à une bonne productivité) et une sanction à long terme (élévation des standards).

En conclusion, la motivation est un phénomène complexe, et les facteurs qui y contribuent peuvent être d'ordre personnel, interpersonnel ou organisationnel.

1. *Theory Z : How American Business Can Meet the Japanese Challenge*, 1981. *Théorie Z, faire face au defi japonais*, InterEditions, 1997.
2. G. Homans, *Social behavior, its elementary forms*, 1961 ; F. Heider, *The psychology of interpersonal relations*, 1958 ; J.S. Adams, *Inequity in social exchange*, 1965.
3. *Industrial social psychology*, 1969.

5.3.5 La gestion des conflits

L'analyse des risques, présentée au chapitre 6, permet d'anticiper des conflits possibles et de prendre des mesures préventives. Cependant, les imprévus du projet peuvent prendre parfois une forme conflictuelle. On fait ici référence à des conflits qui se produisent entre pairs, avec un autre chef de projet, ou avec un maître d'ouvrage, ou encore avec un responsable transversal à la Direction informatique.

On distingue en général cinq modes de résolution des conflits[1] :

- le retrait ;
- l'apaisement ;
- la force ;
- le compromis ;
- la collaboration.

S'il adopte la stratégie du *retrait*, le chef de projet se retire du conflit, il se tient à distance en attendant que les choses se calment. Face à son équipe, il peut aller jusqu'à nier l'existence du conflit.

C'est une stratégie à court terme, pertinente pour des conflits de faible intensité et dont l'objet n'a que peu d'impact sur les activités et la réussite du projet. On peut également l'adopter comme une attitude d'attente, lorsque le contexte n'est pas propice à un traitement de fond du conflit.

Si le chef de projet opte pour l'*apaisement*, il va rencontrer la personne avec laquelle il est en conflit et tenter de minimiser les oppositions, en soulignant les éléments de rapprochement.

Cette stratégie suppose que l'autre personne — ou les autres personnes — ne souhaite pas non plus aborder le conflit de front. Le chef de projet mise sur les effets du temps, qui relativise les oppositions. Il est soucieux de ne pas détériorer la communication. Ce peut être une stratégie temporaire, en attendant de traiter l'objet du conflit en profondeur. Cela est particulièrement vrai si l'apaisement conduit à une harmonie superficielle, donc peu durable.

Si le chef de projet décide de recourir à une stratégie de *force*, cela signifie qu'il va utiliser une position de pouvoir, la sienne ou celle d'un tiers qui lui est favorable, pour imposer son point de vue et faire adopter la décision qu'il préconise.

Cette stratégie peut avoir des conséquences à long terme sur la communication avec les partenaires avec lesquels on est en conflit. Ce mode de résolution

1. La présentation est adaptée de [Kezsbom *et al.*, 1989].

peut faire l'objet d'un accord entre les parties : s'en remettre à la décision d'une autorité, par exemple l'arbitrage d'un Comité de pilotage.

S'il suit la stratégie du *compromis*, le chef de projet va entrer en négociation avec la personne avec laquelle il est en conflit. Chacun va céder partiellement pour aboutir à une solution acceptable. Il peut parfois faire intervenir un médiateur, c'est-à-dire une personne non engagée dans le conflit, pour mener la négociation.

Cette stratégie ne conduit pas toujours à la meilleure solution, quand il s'agit d'un conflit portant sur des options techniques ou des exigences de qualité.

S'il choisit la stratégie de *collaboration*, le chef de projet doit reconnaître la valeur des positions adverses. Le conflit peut être source d'enrichissement et d'amélioration de la qualité pour le projet.

Ce mode de résolution suppose, en général, qu'il ne s'agit pas d'un face-à-face entre deux adversaires, mais que différentes positions se sont exprimées. Un groupe de travail peut, par un travail d'échange et de prise en compte de points de vue adverses, conduire à une solution satisfaisante pour toutes les parties. Cela suppose un engagement important dans le projet et une volonté partagée d'atteindre au mieux l'objectif. Si le temps est compté, ce mode de résolution peut être dangereux.

Les styles de management et la gestion des conflits seront illustrés dans la deuxième partie au chapitre 13.

5.4 LA GESTION DU CHANGEMENT

5.4.1 La nature et le contenu de la gestion du changement

Les technologies de l'information peuvent être un facteur clé d'amélioration du fonctionnement de l'entreprise et de sa création de valeur : réduction des cycles de traitement, augmentation de la qualité, nouveau service, efficacité accrue… Cependant, ce ne sont pas les technologies en elles-mêmes qui sont garantes d'amélioration, mais les systèmes d'information qui en font une utilisation particulière. Or, la définition d'un système d'information intègre les acteurs participant au système :

Le **système d'information** est la partie du réel constituée d'informations organisées, d'événements ayant un effet sur ces informations et d'acteurs qui agissent sur ces informations ou à partir de ces informations, selon des processus visant une finalité de gestion et utilisant les technologies de l'information. Il s'appuie sur un système informatique, ensemble organisé d'objets techniques — matériels, logiciels, applications — dont la mise en œuvre réalise son infrastructure.

Par conséquent, la réussite d'un système d'information rend nécessaire son appropriation par les acteurs concernés. Cependant, les changements apportés par le nouveau système, en particulier dans ses aspects informatiques et organisationnels, peuvent parfois conduire à un rejet, rendant le nouveau dispositif inefficace. Il est donc important de comprendre les phénomènes en jeu, pour pouvoir adopter une attitude appropriée.

On peut distinguer deux attitudes selon le degré d'engagement apporté par les dirigeants de l'entreprise et les responsables du projet[1] :

- Le *soutien* consiste à encourager l'adoption du nouveau système ou de la nouvelle technologie par une attitude positive et bienveillante. Il y a peu d'actions planifiées de façon centralisée et le changement va s'opérer à partir des évolutions et initiatives individuelles ou locales. Ce fut le cas dans les années 1980 pour l'introduction de la micro-informatique dans beaucoup de grandes entreprises, ou bien dix ans plus tard pour l'utilisation du courrier électronique.
- L'*implication* consiste à prendre des mesures pour assurer l'assimilation du nouveau système et un fonctionnement efficace et efficient. Lorsqu'il s'agit de mettre en œuvre un système d'information tel que nous l'avons défini ci-dessus, le soutien ne permet généralement pas d'atteindre les avantages attendus du nouveau système ; il est même parfois impératif que le jeu soit collectif, sous peine de dysfonctionnement graves dans l'activité ou le pilotage de l'entreprise. On parle généralement de « gestion du changement » pour faire référence aux activités mise en œuvre quand il y a implication, ce qui n'exclut pas une approche participative et partiellement décentralisée.

L'objectif de la gestion du changement est de faciliter le passage d'un système socio-technique existant à un système visé. Pour cela, on peut identifier plusieurs axes d'appréhension :

- Le changement est un processus d'apprentissage individuel et collectif.
- Le changement implique une évolution dans la représentation que se font les acteurs du système, ainsi que de la place et du rôle qu'ils occupent.
- Le changement comporte une dimension politique, tous les acteurs ne percevant pas ou n'ayant pas le même intérêt dans l'évolution du système.

1. R. Agarwal, M. Tanniru et D. Wilemon, *IEEE transactions on EM*, 44, 4, nov 1997.

Du point de vue des acteurs, cela revient à savoir comment faire (utilisation du système), comprendre pourquoi (appréhension de la logique globale du système) et participer (convergence entre les stratégies des acteurs et les objectifs du système).

La gestion du changement implique la planification et la réalisation d'un ensemble d'activités, On identifie classiquement : la communication, la formation, la documentation, l'organisation des sites, la migration, l'expérimentation. Avant de les décrire, nous allons développer la notion de « résistance au changement » et le rôle de la confiance, ainsi que les stratégies de déploiement dans lesquelles ces activités s'inscrivent.

5.4.2 La résistance au changement

Le concept de résistance au changement a parfois été présenté comme un facteur explicatif de l'échec de certains systèmes d'information : les personnes préfèrent la stabilité et refusent a priori toute évolution. C'est peut-être oublier que la destinée humaine est faite de changement tout au long de son existence, et que le cadre dans lequel elle s'inscrit est lui aussi en constante évolution. Il est préférable de considérer la résistance comme un symptôme, pouvant se manifester dès le début du projet jusque dans le fonctionnement opérationnel du nouveau système, et de s'interroger sur ses origines, qui relèvent souvent plusieurs facteurs.

De façon schématique, on distingue souvent trois catégories de causes, non exclusives :

- La résistance est due au *système* lui-même : il s'agit d'une problématique de « besoins ».

  Le système, en particulier le système informatique, apparaît insatisfaisant en termes techniques (temps de réponses, bugs…), fonctionnels (fonctions inadéquates, informations manquantes, erreurs, tâches lourdes…), ergonomiques (interfaces inadaptées au contexte métier, apprentissage difficile…).

  Pour éviter ce type de résistance, des actions sont mises en œuvre tout au long du projet pour s'assurer de l'adéquation du futur système : participation des utilisateurs à l'analyse et à la conception, expérimentation sur des sites pilotes, planification de la migration de l'ancien vers le nouveau système.

- La résistance est liée aux *acteurs* eux-mêmes : il s'agit d'une problématique d'attitude individuelle. Deux interprétations ont été données.
  La première considère qu'il existe des différences intrinsèques entre les individus face au changement. Ces différences sont liées au profil psychologique (innovateur ou conservateur). Il s'agit alors de rassurer et préparer

ces utilisateurs, par des actions de communication, de formation et d'accompagnement.

La seconde estime que les différences d'attitude sont contingentes à une situation donnée, c'est-à-dire que la résistance ou l'engagement dépendent de la perception par l'individu du changement proposé. Cette interprétation s'inscrit dans le courant théorique appelé Modèle d'acceptation technologique[1]. Dans sa proposition initiale, cette théorie considère que l'acceptation et l'utilisation volontaire d'une nouvelle technologie sont déterminées par deux perceptions : l'utilité perçue (« le nouveau système va-t-il améliorer ma performance au travail ? ») et la facilité d'usage perçue (« quel effort devrais-je dépenser pour apprendre et utiliser le nouveau système ? »). Différentes recherches ont permis un affinement du modèle initial. L'utilité perçue semble en général avoir davantage de poids que la facilité d'usage, et elle peut être influencée par des facteurs personnels (âge, expérience, formation…), par l'entourage (attitude des pairs, attentes des supérieurs…) et par le système lui-même (qualité des résultats fournis, pertinence par rapport au poste de travail…).
Si l'on diagnostique une telle origine à la résistance, il s'agit de faire évoluer les perceptions, par la formation et la documentation pour agir sur la facilité perçue, et par la communication et la participation à la conception pour développer la perception de l'utilité.

- La résistance est liée à l'*organisation* : il s'agit d'une problématique impact collectif. Le nouveau système modifie la division du travail, la coordination, la collaboration ou coopération requise, les activités de contrôle… ce qui peut entraîner des oppositions. Deux approches explicatives ont été proposées.

  Selon la perspective dite « socio-technique », certains acteurs manifestent une résistance parce qu'ils pensent (à tort ou à raison) que le nouveau système dégradera leurs conditions de travail. Il s'agit, si l'on veut prévenir une telle résistance, de s'appuyer lors l'analyse des processus sur des principes d'ergonomie (par exemple, appréciation du stress causé par certaines tâches) et d'organisation des postes (par exemple, degré de responsabilité et d'autonomie), et de favoriser une participation à l'analyse des processus et à l'organisation des sites.

  Selon la perspective dite « politique », certains acteurs s'opposent parce qu'ils pensent (à tort ou à raison) que le nouveau système leur enlèvera du pouvoir. L'opposition peut provenir de différents niveaux hiérarchiques. Un cadre peut voir diminuer son pouvoir de décision et d'action, ou la taille du groupe placé sous sa responsabilité. Un opérationnel peut voir

1. TAM : *Technology Acceptance Model*. F. D. Davis, « Perceived Usefulness, Perceived Ease Of Use, And User Acceptance », MIS Quarterly, 3, 3, 319-340, 1989.

réduire sa « zone d'incertitude », liée à la détention de son savoir-faire ou d'informations clés. Certains projets système d'information conduisent, pour atteindre des objectifs d'efficacité, à remettre en question des territoires. Cette situation doit être prise en compte, en particulier dans l'analyse des risques et l'élaboration d'une stratégie telles que développées au chapitre 6. Seule la construction d'une vision d'un « bien commun », partagée par les acteurs en conflit, peut débloquer la situation : un engagement de la direction de l'entreprise et/ou la référence à des valeurs collectives sont des moyens d'y parvenir.

Ces différents points de vue sur la résistance au changement peuvent faciliter la compréhension du contexte du projet et l'élaboration d'une stratégie adéquate. Ils apportent des éléments à l'évaluation du potentiel de réussite lors d'un audit en cours de projet, tel qu'il est présenté au § 6.4.3.

5.4.3 La place de la confiance

Dans un projet système d'information, les acteurs de la conduite du changement sont ceux qui ont le pouvoir d'influencer la nature des changements en préparation. L'équipe de projet, la direction de l'entreprise, la maîtrise d'ouvrage, le chef de projet… en font partie. Les utilisateurs, dont le rôle et/ou le travail sera affecté par le futur système, ne sont que partiellement impliqués dans les décisions sur le changement, sauf ceux qui ont été sélectionnés pour le mener ou y participer. Ceux qui restent extérieurs aux décisions sont dans une situation d'incertitude, c'est-à-dire qu'ils ne peuvent pas adopter un comportement fondé sur des éléments qu'ils tiennent pour certains. En l'absence de certitude, le changement peut apparaître comme une menace et provoquer des attitudes de repli.

C'est pourquoi la réussite d'une action de changement est souvent liée à l'établissement d'une relation de confiance entre les acteurs de la conduite du changement et les utilisateurs.

La *confiance* des utilisateurs peut être définie comme la croyance dans le fait que les acteurs de la conduite du changement accompliront des actions qui auront des effets positifs pour eux et qu'ils n'accompliront pas d'actions non désirables qui auraient des résultats négatifs.

La confiance et l'engagement des utilisateurs interagissent dynamiquement. Si les acteurs soumis au changement ont confiance dans la compétence de ceux qui organisent le changement, mais également dans leurs intentions, ils auront tendance à adopter une attitude positive face au projet. L'observation des actions contribue à développer ou diminuer la confiance dans les compétences. Par ailleurs, la confiance dans les intentions tend à se renforcer au fur et à mesure que les faits confirment le bien-fondé de la décision de faire confiance.

La communication et les opérations d'expérimentation du futur système (prototype ou site pilote) peuvent développer la confiance. Cette disposition d'esprit améliore l'engagement vis-à-vis du projet, notamment lors des formations ou de participations ponctuelles. Il peut ainsi se créer une dynamique positive d'implication active des utilisateurs.

5.4.4 Les stratégies de déploiement

On appelle « déploiement » le fait de passer de l'ancien au nouveau système. Le terme, qui est aujourd'hui d'un emploi généralisé, renvoie à une image selon laquelle le système, d'abord limité à une application informatique, change de dimension quand il est installé pour un fonctionnement opérationnel.

Différentes stratégies peuvent être adoptées, qui dépendent de plusieurs facteurs :

- La taille du projet ;
- La modularité du système (indépendance des fonctions ou des processus) ;
- Le contexte (géographie, organisation…) ;
- Les contraintes d'origine interne ou externe (coûts, calendrier, réglementation, autres projets…) ;
- Les risques de différentes natures (commercial, social, politique, image…).

La stratégie comporte deux dimensions : le déploiement structurel et/ou géographique, et le déploiement fonctionnel. Elle peut inclure un fonctionnement en double ancienne/nouvelle application pendant une certaine durée. Les différentes modalités retenues pour la première dimension sont souvent :

- Couverture complète : tous les sites concernés changeront de système simultanément.
- Couverture par région : le changement de système s'effectuera à des dates différentes selon les pays ou les régions.
- Couverture par strates : les différentes entités dans l'organigramme (agences, succursales, concessionnaires…) rentreront dans le nouveau système à des moments différents.

Le déploiement fonctionnel peut être d'un bloc (l'ensemble des processus, variantes et fonctions) ou progressif.

On appelle souvent « stratégie big bang » la couverture complète et d'un bloc. C'est la solution la plus rapide, qui simplifie la migration des données, mais en cas d'échec les conséquences sont plus graves. Une couverture progressive par strates ou régions répartit les risques, mais fait cohabiter deux systèmes. Un

déploiement progressif est moins lourd à gérer, mais oblige à développer des passerelles entre ancien et nouveau système informatique.

La stratégie de déploiement est en partie liée à la stratégie de projet, telle que présentée au § 6.4.2, et a des impacts sur les activités de la gestion du changement, notamment la communication, la formation, la migration et l'organisation des sites.

5.4.5 Les activités de la gestion du changement

La gestion du changement comporte différentes activités, dont certaines visent les compétences et les comportements des utilisateurs (communication et formation), et d'autres la construction d'un contexte permettant une utilisation réussie du nouveau système (documentation, migration, organisation des sites).

- L'objectif de la *communication* est de réduire les risques d'un rejet qui aurait pour cause la non préparation ou la désinformation des acteurs concernés par le futur système. Les cibles peuvent être internes (décideurs, utilisateurs, hiérarchie locale, équipe projet, syndicats, Comité d'entreprise…) ou externes (partenaires, fournisseurs, clients, grand public…). La cible influe sur le choix du média, le contenu du message et le calendrier de communication. Les opérations de communication font généralement l'objet d'un suivi quantitatif et qualitatif, permettant d'améliorer les opérations ultérieures :
- L'objectif de la *formation* est de faciliter la prise en main du nouveau produit par les utilisateurs et leur appropriation des nouvelles règles et procédures. La prise en compte du contexte humain, ainsi que des contraintes de nombre, de budget et de calendrier, conduit à l'établissement d'une stratégie de formation, c'est-à-dire des modalités (auto-formation, formation directe, formation de formateurs, accompagnement) et de la couverture. Elle est ensuite concrétisée par un plan de formation (profils, thèmes, filières, modules, supports, planning).
- L'objectif de la *documentation* est de fournir aux utilisateurs une assistance après passage au nouveau système. Elle peut être un moyen d'auto formation pour certains. On peut distinguer la description des processus organisationnels (rôles, flux, documents, règles de gestion…), le manuel d'utilisation du logiciel (en mode débutant et en mode avancé) et l'aide en ligne.
- Formation et documentation peuvent être complétées par des actions d'*accompagnement* lors des premiers temps du fonctionnement du nouveau système et de l'utilisation quotidienne du logiciel. Elles peuvent prendre la forme de personnes ressources à la disposition des différents sites ou bien d'un centre d'appels interne.

- L'*organisation* des sites consiste, pour chaque entité structurelle et/ou géographique, à étudier de façon détaillée les nouvelles procédures organisationnelles et à préparer la mise en place des moyens humains et matériels. Elle peut comprendre une analyse de l'organisation existante, un inventaire des matériels et un diagnostic des compétences. Elle est à articuler avec le plan de communication et le plan de formation. Une délégation locale et une large implication des acteurs sont souvent préconisées.
- L'objectif de la *migration* est de mettre en place les données (c'est-à-dire les informations gérées sous forme électroniques, fichiers, bases de données, bases de documents) nécessaires au démarrage du nouveau système, en limitant la charge imposée aux utilisateurs, en réduisant les risques de pertes ou de discontinuité du service et en minimisant les perturbations des autres systèmes. La migration comprend un volet technique (lié au passage d'une technologie à une autre) et un volet fonctionnel (écarts dans la nature et la structure des données gérées entre l'ancien et le nouveau système). On s'appuie au maximum sur des outils logiciels du marché ou sur des programmes développés spécifiquement. Les utilisateurs peuvent cependant être sollicités pour définir des règles (codification) ou pour des tâches non automatisables (vérification de la validité des données). Les conséquences de la non qualité d'une migration peuvent être importantes. Rappelons à cet égard les difficultés du système Socrate de la SNCF, dont les bases de données recensant les différentes destinations étaient incomplètes, ce qui bloquait certaines ventes. C'est pourquoi il est recommandé d'établir un plan de migration, comprenant les actions à mener, leur planification et les responsabilités.
- L'*expérimentation* a pour objectif de réduire les risques de perturbations lors du passage au nouveau système. Il s'agit de tester l'adéquation et la pertinence de ce qui a été conçu (produit et procédures) par rapport aux objectifs visés par le projet. L'expérimentation peut se faire en mode simulation — on parle alors de « vérification d'aptitude » — ou en mode réel sur un site pilote — on parle alors de « vérification de service régulier ». Le choix d'un site d'expérimentation obéit à des critères de représentativité du point de vue de l'activité du groupe (fonctions assurées, volumes traités, organisation), à des critères de calendrier (disponibilité, absence de conflit avec des surcharges de travail) et à des critères politico-stratégiques (visibilité du site, légitimité du site, influence sur les autres acteurs…). L'expérimentation se termine par un bilan, à partir d'indicateurs préalablement déterminés, suivi d'une décision : poursuite du déploiement, aménagement du déploiement incluant des corrections du système, ou bien, dans certains cas graves de rejet ou de dysfonctionnement, abandon de la mise en œuvre du système.

5.5 L'ORGANISATION DU TRAVAIL DANS LES MÉTHODES AGILES

La dimension humaine est une préoccupation majeure pour les concepteurs des méthodes agiles. Nous allons en indiquer les principaux points : une attention particulière aux développeurs, un souci d'intégration des utilisateurs, des techniques pour faciliter l'expression des exigences, ainsi que des définitions de rôles particuliers.

5.5.1 Le management de l'équipe de développeurs

Les méthodes agiles, particulièrement orientées vers les projets de développement d'application, ont cherché des dispositions pour éviter ou réduire les problèmes de compétence, de communication ou de motivation chez les développeurs.

Une des principales dispositions est la *programmation en binôme* (*pair programming*). La pratique de la programmation, longtemps considérée comme un art, a fait l'objet de dispositions méthodologiques dans le sens d'une certaine industrialisation, visant à obtenir une qualité en grande partie indépendante des talents des développeurs. L'approche par composants et la réutilisation, rendues possibles par les langages à objets, traduisent cette orientation. Cependant, même si elle porte sur des éléments plus petits, la programmation est le plus souvent restée une activité solitaire.

Les méthodes agiles, en particulier XP, ont rompu avec cette pratique et proposé d'organiser le développement en binômes, alliant souvent un développeur confirmé avec un autre moins expérimenté. La qualité du logiciel devrait s'en trouver accrue pour deux raisons. La collaboration obligeant à expliciter ses choix conduit à une écriture plus simple et plus robuste. De plus, le regard d'autrui peut permettre de détecter plus rapidement d'éventuelles erreurs, abaissant ainsi leur coût de correction. Les binômes sont incités à de fréquentes recompositions, à des fins de la circulation de connaissances par compagnonnage.

La programmation en binôme a donné lieu à discussions, notamment sur le droit éventuel à refuser de travailler sur ce mode[1].

Un second principe, qui vise également le décloisonnement entre développeurs, est celui de *propriété collective du code*, chacun étant responsable de l'ensemble du logiciel et pouvant intervenir si nécessaire sur des programmes

1. Voir par exemple : J.Kendall et K.E.Kendall, « Agile methodologies and the lone systems analyst : when individual creativity and organizational goals collide in the global IT environment », *Journal of Individual Employment Rights*, 11, 4, 333-347, 2004-2005.

écrits par d'autres. Cela peut poser le problème de la dilution de responsabilité, d'où la recommandation d'entretenir régulièrement l'idée d'engagement collectif[1].

Ce partage permet la mise en place d'une *intégration continue*, c'est-à-dire que le résultat de chacune des tâches élémentaires est livré pour être intégré à un ensemble qui progressivement mettra en œuvre le scénario, puis la fonctionnalité demandée. Cette pratique évite les problèmes d'incohérence que l'on rencontre fréquemment lorsque l'on consolide les produits d'un travail morcelé. Elle assure une coordination régulière entre développeurs.

Un troisième principe est l'attention aux *conditions de travail* des développeurs. Ces derniers, arrivant en bout de chaîne dans les projets, sont souvent soumis à des tensions et dépassements d'horaire pour tenir les engagements contractuels. Le Manifeste Agile s'inscrit dans une philosophie dite de développement durable et considère que le rythme de travail adopté dans le projet doit pouvoir être maintenu dans la durée. Le respect d'une charge de travail hebdomadaire de 40 heures est souvent évoqué. De plus, en contrepartie aux contraintes de collaboration, une attention particulière devrait être portée à l'agrément de l'environnement de travail et à l'harmonie du groupe par l'adoption participative de règles de fonctionnement collectif, voire d'activités extra-professionnelles de détente.

5.5.2 L'implication des utilisateurs

L'implication des utilisateurs poursuit un double objectif : une pertinence accrue du système produit et le maintien du projet dans un délai plus court que ce que l'on observe habituellement.

Dire qu'un système d'information correspond à un besoin est une formulation qui masque une réalité généralement plus complexe. En effet, les attentes sont souvent mal définies et émergent au fil du déroulement du projet. Elles sont susceptibles de changer au fur et à mesure que les commanditaires ont une vision plus claire de ce que les technologies sont susceptibles d'apporter ou parce que les conditions extérieures se sont modifiées. Par ailleurs, pour développer un système, il est quand même nécessaire à un moment donné de stabiliser des spécifications : c'est ce que vise classiquement l'établissement d'un cahier des charges.

Les méthodes agiles ont une position différente, considérant que l'explicitation exhaustive des exigences est à la fois coûteuse et vouée à l'échec. Elles préconisent donc de la remplacer par une participation soutenue des utilisateurs, qui accompagnent le travail des développeurs. Ceci implique une localisation

1. Voir, par exemple, A. Fruhling et G.J. De Vreede, « Field Experiences with eXtreme Programming: Developing an Emergency Response System », *Journal of Management Information Systems*, 22, 4, 39-68, 2006.

géographique proche, des rencontres fréquentes et une communication régulière en face à face. Les utilisateurs ainsi impliqués doivent avoir délégation pour représenter la maîtrise d'ouvrage, les décisions prises ne pouvant être ensuite soumises à validation et remises en question par d'autres acteurs.

Cette forme de participation, qui consiste à réagir immédiatement à une fonction logicielle, permet d'orienter et d'ajuster les options de conception. Pour définir une architecture fonctionnelle, une autre forme de participation est privilégiée, celle qui consiste à travailler en session participative.

5.5.3 Les techniques de travail en session participative

Deux techniques, JRP et JAD, initialement promues par RAD visent à construire des exigences partagées dans un groupe de commanditaires/utilisateurs. Elles ont été reprises notamment par DSDM qui a élargi à une notion d'atelier coopératif et animé, sous le terme *Facilitated workshop*[1].

JRP signifie *Joint Requirement Planning*, définition participative des besoins, et JAD correspond à *Joint Application Design*, conception participative d'application. Ces deux techniques furent proposées dans les années 1980 par IBM, qui les a initialement mises au point pour ses projets d'informatisation internes, suite au mécontentement de ses utilisateurs jugeant être trop peu associés à la définition des spécifications. Dans sa présentation initiale, IBM[2] utilise le terme JAD comme appellation globale des deux techniques JRP et JAD. Lorsque J. Martin les a intégrées dans RAD, il les a considérées comme distinctes et complémentaires.

Le principe commun aux deux techniques est de réunir des utilisateurs pour les faire collaborer sur un objectif précis, de façon intensive, en général pendant un ou deux jours, et dans un lieu les mettant à l'abri des sollicitations quotidiennes. L'isolement temporaire et organisé des acteurs peut favoriser l'émergence d'un esprit de groupe, orienté vers l'objectif commun.

La session est conduite par un animateur, en position neutre par rapport aux enjeux, qui a pour rôle de surveiller les temps de parole, de recentrer vers l'objectif et de veiller à la participation de tous. Elle doit avoir été préparée avec soin, pour obtenir une implication maximum des participants et un résultat solide. La tenue de sessions remplace des processus de décision/validation souvent lourds et longs quand plusieurs décideurs, ayant chacun un point de vue, en sont parties prenantes. C'est pourquoi les utilisateurs participant à une session doivent avoir délégation de décision.

1. Voir http://www.dsdmfrance.com
2. IBM Overview Pamphlet, JAD, « Joint Application Design », White Plains, New York, 1984.

De façon plus précise, une session JRP vise à exprimer les grandes attentes d'un système partagé par différentes entités organisationnelles. La préparation est effectuée par les futurs participants, à l'aide d'un membre de l'équipe projet, par exemple le responsable de projet. Durant la session, chacun présentera son point de vue sur le système existant et ses manques ou disfonctionnements, selon un plan horaire rigoureusement établi. L'animateur doit alors s'assurer que tous les participants partagent l'analyse proposée. Les travaux de préparation devraient permettre d'éviter que surgissent des contradictions irréductibles, et conduire à un bilan global du fonctionnement du domaine. Ensuite, chaque participant expose ses attentes, sous l'angle qui lui est propre. À la lumière de ces éléments, certains peuvent évoluer dans leur expression initiale. On arrive ainsi à construire une première vision du futur système. L'animateur s'attache à dégager un consensus sur le degré de priorité des fonctions.

Dans le cas de JAD, la session vise à ajuster une architecture fonctionnelle ou une fonctionnalité concernant plusieurs entités organisationnelles, suite à une proposition par l'équipe projet. L'horaire d'une session JAD, même s'il s'inscrit dans un cadre rigoureux, est plus souple que celui d'une session JRP. L'animateur adapte le déroulement en prenant en compte la dynamique du groupe. La présence simultanée de toutes les parties prenantes évite des allers-retours. En général, les conflits liés à des positions divergentes trouvent plus facilement une issue. L'utilisation d'outils de maquettage et/ou prototypage, ainsi que de moyens de visualisation et d'impression, renforcent l'engagement des utilisateurs dans une conception collaborative.

5.5.4 Les rôles dans les méthodes agiles

Chaque méthode prévoit des rôles spécifiques visant les deux objectifs majeurs de qualité et délai. Ils orientent les relations entre commanditaires et producteurs, et organisent le travail de l'équipe.

XP identifie six rôles.

Le *programmeur* a une double responsabilité de conception technique et de codage, pour des raisons de maintenabilité du code. Il doit en effet construire son programme de façon à ce qu'il puisse évoluer facilement par modification ou ajout de spécifications.

Le rôle du *client* consiste à exprimer ses exigences, sous forme de scénarios (*user stories*) et en dialoguant avec les développeurs, et à écrire des jeux de tests fonctionnels qui seront utilisés lors de la recette.

Le *testeur* aide le client à élaborer des jeux de test. Il peut intervenir dans les choix de conception pour éviter des options qui augmentent les difficultés de test, donc les risques de non détection d'erreur. Ce rôle délicat nécessite une bonne

connaissance du métier et des compétences en développement. Il s'appuie le plus souvent sur des outils automatisés.

Le *tracker* est le gestionnaire du temps. Il intervient notamment en début de chaque itération, lors de l'estimation, et ensuite il contrôle la progression par des échanges avec les développeurs. Cette tâche correspond à un pilotage opérationnel et quotidien du projet, dont nous décrivons les principes et techniques au chapitre 7. Le tracker n'a pas d'autorité hiérarchique. Son rôle est de maintenir le souci du respect des engagements par son activité quotidienne, mais aussi de détecter d'éventuels problèmes suffisamment tôt pour alerter le coach ou le manager. Chaque matin, il anime une brève réunion, appelée *standup meeting* car l'on reste debout, au cours de laquelle chaque développeur annonce ce qu'il va faire durant la journée et d'éventuelles difficultés auxquelles il se heurte.

Le *coach* a un rôle de soutien technique des membres de l'équipe, notamment lors des premières itérations de livraison. Il peut recadrer certaines options de conception technique ou de codage, organiser des réunions de remue-méninges pour résoudre un problème ou demander des changements d'un binôme, en cas de blocage ou disfonctionnement.

Le rôle du *manager* correspond à un rôle de chef de projet. Il est le responsable des développeurs, qu'il soutient et encourage. Il doit rendre des comptes au client sur la base des informations fournies par le coach et le tracker.

Ces rôles peuvent être combinés. Un chef de projet peut ainsi réunir les rôles de manager, tracker et coach. De façon générale, on peut dire que les deux styles, participatif et par délégation, sont privilégiés, mais qu'ils s'accompagnent d'un pilotage très rigoureux qui demande une discipline quotidienne.

Pour **SCRUM**, la notion d'équipe, au sens sportif du terme, est centrale. C'est pourquoi la méthode n'identifie que trois rôles dans l'équipe**.**

Le *propriétaire* (*product owner*) représente le client en ce qui concerne les spécifications et les acceptations.

Le *scrum master* est un capitaine d'équipe. Il anime au quotidien, recadre le projet en cas de difficulté, notamment en rappelant les priorités et en débloquant les situations. Ce rôle donne lieu à une certification.

Tous les autres acteurs producteurs sont considérés comme des *membres d'équipe*, quelle que soit leur affectation spécifique (architecte, développeur, testeur, administrateur de base de données…). Les membres d'équipe disposent d'une certaine autonomie dans l'organisation de leurs travaux.

SCRUM recommande comme XP une brève réunion quotidienne, appelée « mêlée » (*scrum*) car elle rassemble le propriétaire, membres de l'équipe et son capitaine. Animée par le scrum master, elle permet de répondre à trois

questions : quelle a été la production de la veille ? quelle est la production prévue pour la journée ? qui rencontre un problème ? Un tableau affichant ce qu'il reste à faire pour obtenir le livrable associé à l'itération (le sprint) est alors mis à jour. Son rôle est d'encourager et de stimuler l'équipe de projet par la visualisation de la progression ou d'un piétinement dont il faut sortir. Appelé *Scrum Burndown Chart*, cet outil peut également être utilisé pour montrer plus globalement la progression du projet à chaque nouvelle itération.

Pour bien distinguer la place l'équipe, SCRUM parle parfois de « cochons » et de « poulets », en référence à une histoire imageant les différentes conséquences d'un rôle, et rapportée par K.Schwaber[1]. Un poulet et un cochon envisagent d'ouvrir un restaurant et cherchent un nom. « Pourquoi pas 'Œufs et Jambon' ? » suggère le poulet ? « Certainement pas, répond le cochon, car je serais engagé dans l'affaire, alors que tu ne serais que concerné ». De façon semblable, dit le concepteur de SCRUM, les membres de l'équipe, le Scrum master et le propriétaire sont engagés dans la production des livrables, alors que les autres parties prenantes du projet n'ont qu'une contribution marginale, et de ce fait ne doivent pas être un poids pour le projet. Par exemple, ils sont invités à rester en retrait s'ils assistent aux réunions quotidiennes et à ne pas interférer dans les mêlées. Ils seront informés et consultés lors de la remise des livrables. Par exemple, un architecte système peut intervenir dans la phase de planification, ou les futurs utilisateurs dans la phase d'après-jeu (cf. § 2.8.4).

DSDM identifie neuf rôles et affine davantage que XP ou SCRUM les positions de la maîtrise d'ouvrage[2].

Le *sponsor exécutif* correspond au plus haut niveau de décision du côté de la maîtrise d'ouvrage, puisqu'il s'agit en particulier de financer le projet.

Le *visionnaire* est le porteur du projet par rapport à des objectifs du métier. Il en a une vision globale et intervient en particulier dans les sessions de conception et de révision. Sa responsabilité s'étend à la mise à disposition d'utilisateurs nécessaires au travail de la maîtrise d'œuvre.

L'*utilisateur ambassadeur* peut correspondre à ce que RAD identifie sous le terme Chef de projet utilisateur. Affecté à temps plein sur le projet, il coordonne les demandes et apports des utilisateurs, et il travaille en étroite collaboration avec les responsables opérationnels de la maîtrise d'œuvre.

Le rôle d'*utilisateur conseiller* est en général tenu par un futur utilisateur du système en cours de construction. Il n'intervient que lorsqu'il est sollicité pour fournir des informations sur le fonctionnement opérationnel ou tester les prototypes d'un point de vue utilisation pratique.

1. Voir http://jeffsutherland.com/scrum/2004/05/scrum-pigs-and-chickens.html
2. Voir http://www.dsdmfrance.com

Le *chef de projet* est un rôle de gestionnaire de projet classique. Il peut être assuré aussi bien par quelqu'un venant du métier ou ayant une expérience technique.

Le *coordinateur technique* est garant de la cohérence des travaux des développeurs, auxquels il apporte un soutien technique.

Le *chef d'équipe* est un rôle plus managérial que celui de coordinateur technique. Il porte la responsabilité de la livraison du système documenté, et il assure l'interface avec les utilisateurs, en particulier pour l'organisation et la tenue des sessions de prototypage et ateliers.

Les *développeurs* ne sont pas obligatoirement organisés en binôme, ils peuvent rencontrer les utilisateurs, mais travaillent initialement à partir de documents de l'étude du métier.

Le rôle de *facilitateur* intervient lors de la tenue des ateliers participatifs pour en assurer le travail préparatoire et l'animation.

Le *rapporteur* est un rôle qui permet une transcription et une conservation des résultats des réunions et des ateliers. Dans RAD, ce rôle restreint aux sessions est appelé scribe.

On voit, à travers ces brèves présentations, que selon la méthode un intérêt différent est porté à l'organisation du travail. DSSM se situe à un niveau englobant l'ensemble maîtrise d'ouvrage — maîtrise d'œuvre. À l'inverse, SCRUM se focalise sur la notion d'équipe et cherche à augmenter son engagement. XP structure davantage les rôles de producteurs, avec une exigence particulière autour des tests.

Ainsi s'achève la présentation de différents aspects de la dimension humaine d'un projet. Nous les retrouvons en partie dans le management des risques présenté au chapitre suivant, aussi bien dans l'analyse des facteurs de risque que dans la stratégie et les dispositions visant à contrôler ces risques.

6

Le management des risques

6.1 LES RISQUES DANS LES PROJETS SYSTÈME D'INFORMATION

6.1.1 L'importance des risques dans les projets système d'information

Les risques liés aux projets informatiques sont fréquents et importants.

Le Standish Group a mené en 1994 une première étude statistique sur plus de 8 000 projets informatiques de toutes tailles et dans tous les secteurs d'activité, aux États-Unis. Le résultat a été publié sur Internet sous le titre « Chaos[1] ». Il en ressort qu'un tiers des projets sont abandonnés avant la fin, plus des trois quarts ont dépassé leur budget et/ou délai et près de la moitié n'ont pas complètement atteint leur objectif ! D'autres études ont confirmé le taux d'échec particulièrement élevé des projets système d'information.

KPMG, dans une étude récente menée auprès de plus de 600 organisations dans 22 pays du globe[2], constate que dans les 12 derniers mois, 49 % des organisations interrogées ont subi au moins un échec de projet informatique. De plus, 86 % des organisations interrogées font état d'un bénéfice réel inférieur de 25 % par rapport à celui attendu, chiffre que les analystes trouvent optimiste.

Différents exemples témoignent des difficultés rencontrées. Citons notamment le célèbre système Socrate de la SNCF, dont les dysfonctionnements ont

1. On peut la trouver notamment sur : http://www.projectsmart.co.uk/docs/chaos-report.pdf ou sur la partie publique du site http://www.standishgroup.com.
2. *Global IT Project Management Survey 2005*, téléchargeable notamment sur http://www.kpmg.co.nz.

été longuement commentés, ou le Plan informatique du ministère de la Justice abandonné en 1994 alors que 850 millions de francs y avaient été consacrés, qui ont donné « des résultats très médiocres », selon le rapport de la Cour des comptes. Le projet Taurus de liaison informatique entre tous les acteurs financiers de la place de Londres est stoppé en 1993 après dix ans d'efforts et une dépense officiellement évaluée à 400 millions de livres ; un projet analogue (projet Relit) à Paris a été livré avec trois ans de retard. En 1996, le distributeur de produits pharmaceutiques Fox-Meyer a fait faillite, suite aux dépenses considérables entraînées par la tentative d'installation du progiciel intégré SAP.

Ces quelques exemples ne sont pas des cas isolés, on en retrouve dans différents pays et secteurs d'activité, dans des entreprises ou des institutions réputées par ailleurs pour leur professionnalisme. Des enquêtes récentes ont montré des améliorations[1], mais des progrès restent à accomplir. Le management des risques est donc de la plus grande actualité.

6.1.2 La définition du risque

Les assureurs définissent le risque par les conséquences financières d'un événement redouté et sa fréquence probable :

Risque = Coût des conséquences d'un événement
× fréquence de cet événement

Si, par exemple, les conséquences sont de 500 € et que l'événement a une certaine probabilité de se produire trois fois tous les 5 ans, on dira que le risque est de :

500 € × 3/5 = 300 € /an

Cette définition n'est pas retenue dans le management des projets, où l'on s'attache moins à quantifier les risques qu'à en faire une analyse fine.

C'est pourquoi on préfère définir le risque comme *la possibilité qu'un projet ne s'exécute pas conformément aux prévisions de dates d'achèvement, de coût et de spécifications, ces écarts par rapport aux prévisions étant considérés comme difficilement acceptables voire inacceptables* [AFITEP, 2000].

Le PMBOK le considère comme une menace dont la concrétisation, incertaine, aurait un impact positif ou négatif sur au moins un objectif du projet tel que les délais, le coût, le contenu ou la qualité [PMI, 2003].

Le risque prend, en général, soit la forme d'un événement (départ d'une ressource-clé, changement dans la réglementation…), soit celle d'une situation dans laquelle le projet se retrouve (parties prenantes divisées, équipe démotivée,

1. *Projet planté, à qui la faute ?*, 01 Informatique, n°1830, 7 octobre 2005, p.40-43.

perte de la maîtrise du contenu du projet…). La réalisation des risques peut porter sur le processus ou sur le résultat.

Dans le premier cas, il s'agit, par exemple, des cas de figure suivants : le projet n'aboutit pas ; il a consommé trop de ressources ; il a duré trop longtemps…

Dans le deuxième cas, on peut donner comme exemples : le système ne remplit pas la fonction attendue ; il n'est pas accepté par l'utilisateur ; son fonctionnement est trop coûteux.

6.1.3 Le management des risques

Le risque résulte de l'incertitude attachée au projet, d'imprévus ou d'aléas. L'incertitude correspond à une « *insuffisance d'information* » qui empêche de prendre des décisions de façon assurée. Les imprévus sont des "*événements qui n'ont pas été envisagés*" [AFITEP, 2000]. notamment lors de l'analyse de risques. Les aléas sont des événements « *imprévisibles* » ayant des conséquences négatives sur les délais et/ou les coûts.

Le management des risques consiste d'une part à réduire l'incertitude par une analyse plus approfondie des caractéristiques internes et environnementales du projet, d'autre part à élaborer des stratégies pour faire face aux risques les plus graves et/ou les plus probables.

L'origine des risques étant l'incertitude, celle-ci peut aussi bien apporter des éléments favorables que des éléments défavorables. Cependant, dans la plupart des cas, on va s'attacher à éliminer ou à réduire les probabilités d'apparition d'éléments négatifs et/ou à atténuer leurs conséquences possibles sur le projet.

La démarche générale de management des risques[1] comprend cinq étapes :

1. identifier les risques ;
2. évaluer leur impact possible sur les coûts, le délai et la qualité ;
3. définir des actions ou prendre des dispositions aptes à réduire les risques jugés inacceptables ;
4. suivre les actions ou la mise en œuvre des dispositions, et surveiller régulièrement l'état des risques ;
5. capitaliser l'expérience.

On définit parfois un rôle spécifique : le manager des risques, mais en général les responsabilités sont réparties au sein de l'équipe de projet.

1. Cette démarche est représentée notamment par le projet européen RiskDriver, qui a eu comme objectif de promouvoir en Europe le management du risques et a donné lieu à l'outil Riskman : Opl Uk Limited, Riskman Reference Manual: The European Project Risk Management Methodology, Blackwell Pub, 1996.

Nous allons présenter le management des risques en nous focalisant sur deux aspects majeurs : l'analyse des risques (qui correspond aux étapes 1 et 2) et le contrôle des risques (qui correspond aux étapes 3 et 4). L'étape 5 sera abordée de façon plus large au chapitre 7 (paragraphe 7.6).

6.2 L'ANALYSE DES RISQUES

6.2.1 Les différentes approches d'analyse des risques

Dans le domaine des systèmes d'information, on a longtemps cru que le strict respect de dispositions méthodologiques permettrait de mener les projets avec succès. C'était mal comprendre les particularités de ces projets.

Les trois principales sources d'échec sont la *définition des besoins*, l'*estimation des charges* et la possibilité de *nombreux aléas* dans le déroulement du projet. En effet, la plupart du temps les objectifs sont loin d'être clairs et ce que l'on attend du futur système se clarifie progressivement ; or, parfois, au lieu de converger vers des spécifications de plus en plus précises, on repart sur de nouvelles pistes ; ou bien l'on se retrouve dans l'incapacité d'assembler des éléments de solution élaborés séparément et d'en faire un tout cohérent et répondant à l'objectif visé. Par ailleurs, l'estimation des charges n'est pas une science exacte : les écarts de productivité individuelle peuvent être importants et de nombreux facteurs peuvent conduire à un dépassement de charge. De plus, divers événements imprévus ou imprévisibles peuvent également avoir une incidence sur le déroulement du projet.

Pour repérer et prévenir ces difficultés, différentes approches d'analyse des risques ont été proposées.

Nous allons d'abord présenter une approche généralisée d'identification des risques. Puis nous donnerons une deuxième approche basée sur une liste de risques particuliers correspondant à des caractéristiques de projet. Ensuite, nous développerons l'approche par le profil de risque du projet.

6.2.2 L'approche généralisée

Il s'agit de produire une liste de risques la plus large possible, et ensuite de sélectionner ceux qui semblent les plus sérieux.

Différentes techniques de stimulation des idées peuvent être utilisées :

- des entretiens ou des séances de remue-méninges avec les parties prenantes du projet ;
- la méthode Delphi, en faisant appel à des personnes expérimentées ;
- l'analyse du projet selon une grille FFOM (forces, faiblesses, opportunités, menaces).

On peut éventuellement utiliser un diagramme cause-effet, dit d'Ishikawa, pour rechercher les sources d'un effet redouté (figure 6.1).

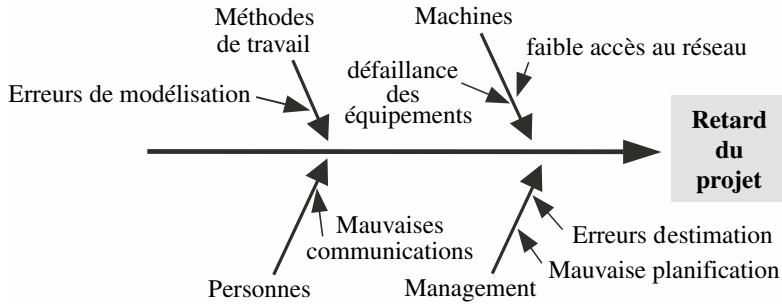


Figure 6.1 — Utilisation d'un diagramme cause-effet pour l'identification des risques

Ensuite, on peut classer les risques en catégories à l'aide d'une matrice, parfois appelée matrice de Pareto, selon leur probabilité d'occurrence et les impacts prévisibles sur le projet (figure 6.2).

Impact Probabilité occurrence	Faible	Important
Élevée		risques inacceptables
Faible	Risques mineurs	

Figure 6.2 — Matrice de probabilité/impact

6.2.3 L'approche par les types de risques recensés

Des listes de risques ont parfois été établies. Nous allons donner deux exemples de recensement raisonné de risques propres aux projets informatiques.

Le premier a été proposé par Eurométhode, dont nous présentons le cadre au chapitre 16 consacré aux associations professionnelles et aux normes. Le second résulte d'un travail mené au sein de l'AFITEP.

Eurométhode considère que les risques du projet peuvent provenir de différentes sources (figure 6.3). Dans les phases amont, les spécifications peuvent être difficiles à élaborer (système d'information) pour différentes raisons. Ensuite, on peut buter sur des obstacles dans la construction du système informatique. Différentes activités du projet peuvent être entachées d'incertitude. Les liens structurels du projet avec d'autres ou entre les parties prenantes présentent parfois des caractéristiques difficiles à gérer. Enfin la taille et la compétence de l'équipe de projet, ainsi que les caractéristiques des outils mis en œuvre dans le projet sont également des causes de difficultés.

système d'information	complexité de l'information, complexité des processus métiers, incertitude sur la compétence des utilisateurs, incertitude sur la stabilité des spécifications…
système informatique	complexité de la technologie cible, complexité des algorithmes…
projet : les activités	complexité de la migration, incertitude liée aux sous-traitants…
structure	nombre d'interfaces avec d'autres projets, degré de formalisation de la relation MOA-MOE
acteurs	nombre d'acteurs directement impliqués, compétences de l'équipe de projet…
technologie	complexité de l'atelier de génie logiciel, disponibilité de l'outil de gestion de projet

Figure 6.3 — Les sources de risques selon Eurométhode

Plus récemment, la commission informatique de l'AFITEP[1] a proposé une typologie des projets informatiques et une liste des risques principaux associés à chaque type de projet.

Ainsi, un projet peut être caractérisé par un type d'objectif, un type de cible et un type de solution envisagé (figure 6.4).

	Définition	**Exemples**
Objectif		
Stratégie	Il s'agit d'un projet dont les enjeux relèvent de la Direction générale. Il s'appuie souvent sur une technologie novatrice.	Messagerie dans les années 1970 GPAO dans les années 1980 Commerce électronique
Efficience	Il s'agit d'un projet de remplacement ou d'outillage d'un processus existant, en vue d'accroître la productivité. Il s'appuie souvent sur des technologies éprouvées.	Mise en place d'un workflow Mise en place d'un progiciel de paie Automatisation de gestion de stock
Obligation	Il s'agit d'un projet lié à une obligation extérieure, légale ou de fait. Il s'appuie souvent sur une technologie banalisée.	Projet an 2000 Projet Euro Fusion/absorption d'entreprise

Figure 6.4 — Critères de classification des projets.

1. Pour plus de détails, voir la revue de l'AFIPEP : *La Cible*, n° 80 et n° 81 (déc. 1999 et janv. 2000).

	Définition	Exemples
Cible		
Client	Le résultat du projet accroît, pour les clients, la valeur ajoutée de l'organisation (entreprise, administration, association…). Le client peut être l'usager, le consommateur, le prescripteur, l'adhérent	Bon de commande sur Internet Informatique embarquée TGV Gestion de la relation client
Support	Le résultat du projet améliore le fonctionnement interne de l'organisation. L'impact sur le client est indirect.	Progiciel comptable
Transversal	Le résultat du projet touche l'ensemble de l'organisation. Il s'agit principalement de la mise en place d'outils communs.	Messagerie Systèmes bureautiques Intranet
Type de solution		
Progiciel applicatif	Projet de mise en place d'un produit logiciel standard commercialisé par un éditeur.	CAO, gestion intégrée, Gestion documentaire électronique
Type de solution ***(suite)***		
Développement applicatif	Projet de conception et de développement d'un logiciel spécifique dont le code source appartient à l'entreprise (même si le développement est sous-traité en tout ou partie).	Création d'un site web commercial Gestion de bourses
Intégration de systèmes	Projet d'assemblage et d'interface des différents éléments matériels et immatériels.	Projet « Socrate » de la SNCF et son matériel
Maintenance évolutive	Projet d'évolution d'un système existant (matériel ou logiciel).	Nouvelle version d'un logiciel interne
Infrastructures techniques	Projet concernant la mise à disposition d'une ossature technique.	Déploiement de réseau, Système de sécurité

Figure 6.4 — Critères de classification des projets. *(Suite)*

Selon les valeurs prises par ces trois critères, des risques ont pu être fréquemment observés (figure 6.5).

Objectif du projet	**Risques**
Stratégie	Faible niveau d'implication de la Direction Générale Changement de l'environnement Non-remise en cause de l'existant Communication déficiente
Efficience	Appropriation insuffisante du SI par les utilisateurs Sous-estimation globale du projet, minimisation des coûts Dérive technologique
Obligatoire	Manque d'attractivité du projet Cahier des charges incomplet Non-respect des délais
Cible du projet	**Risques**
Client	Mauvaise perception des attentes client Non-remise en cause du fonctionnement interne Détérioration de la performance de l'organisation
Support	Non-remise en cause de l'existant Sous-estimation des travaux Modification de l'environnement Rejet par les opérationnels
Cible du projet	**Risques**
Transversal	Définition insuffisante de l'objectif Structuration inadéquate du projet Sous-estimation de l'utilisation
Type de solution	**Risques**
Progiciel applicatif	Non-pérennité du produit sélectionné Sous-estimation de la charge/complexité de l'intégration Pas de remise en cause de l'existant Gestion du changement déficiente Pas de prise en compte des évolutions du progiciel
Développement applicatif	Insuffisance du cahier des charges (besoins et solutions) Manque de compétences ou de pérennité du prestataire
Intégration de système	Erreurs dans le choix des composants Sous-estimation des travaux de migration et d'interfaçage

Figure 6.5 — Liste des risques principaux selon le type de projet.

Type de solution	Risques
Maintenance	Absence ou indisponibilité des ressources (humaines et/ou documentaires) Mauvaise analyse d'impacts des modifications envisagées
Infrastructure technique	Réduction du projet à sa dimension technique Technologie incompatible avec la maturité technologique de l'entreprise

Figure 6.5 — Liste des risques principaux selon le type de projet. *(Suite)*

Cette typologie représente, pour le chef de projet, une aide à l'identification des risques de son propre projet.

6.2.4 L'approche par le profil de risque

Cette approche s'apparente à la précédente, dans le sens où elle s'appuie sur une liste de facteurs de risque reconnus, mais elle approfondit l'analyse à l'aide de critères et métriques pour obtenir une mesure de chaque source de risque, ce qui permet de dresser le profil du projet.

Les facteurs de risque

L'origine des risques pouvant peser sur une situation de projet est variée, cependant six facteurs, qui caractérisent le projet, jouent un rôle déterminant :

1. la taille du projet ;
2. la difficulté technique ;
3. le degré d'intégration ;
4. la configuration organisationnelle ;
5. le changement ;
6. l'instabilité de l'équipe de projet.

La *taille du projet* est un facteur retenu par la plupart des auteurs. Le risque est lié aux limites de l'individu. Un grand projet signifie une large étendue du domaine couvert, qui impose une division du travail entre un nombre élevé de personnes. La semi-autonomie nécessaire à l'avancement de chaque sous-groupe conduit naturellement à un morcellement. En l'absence d'un dispositif imposant une synthèse et un dépistage des incohérences, il y a perte de la maîtrise du processus de production qui ne converge plus et n'est plus sous contrôle.

La majorité des échecs de projet système d'information n'ont pas une cause liée à la *difficulté technique* ; cependant ce facteur est souvent cité comme générateur de risque. Il correspond à une nouveauté technologique ou bien à une difficulté technique provenant des contraintes imposées au projet. Ce sont souvent des contraintes de performance. Le risque est celui de l'absence de compétences techniques nécessaires, qui pénalise la production.

Le *degré d'intégration* agit sur la complexité du projet, puisqu'il mesure le degré de dépendance ou d'autonomie du futur système. Des flux nombreux et variés entre le système d'information que l'on développe et les autres systèmes d'information de l'entreprise rendent plus difficile l'identification claire des impacts de choix de conception. D'autres entités, d'autres projets, d'autres équipes de développement sont concernés, ce qui augmente le nombre d'acteurs du projet.

La *configuration organisationnelle* est un facteur correspondant à l'étendue de l'entreprise qui est touchée par le projet (*organizational scope*). Le risque provient de la lourdeur des procédures de participation et de décision quand plusieurs grandes entités (des directions, par exemple) sont parties prenantes du projet. La production en est ralentie. S'y ajoutent d'éventuels motifs de conflits, qui alimentent un processus politique latent, bloquant les prises de décision.

Le *changement* visé par le projet signifie que les systèmes de gestion et/ou d'organisation existants ne peuvent pas être pris comme référence stable et que l'effort de conception/innovation va être important. L'abandon du statu quo crée une instabilité qui favorise le processus politique. Les risques de rejet ou de mauvaise définition du futur système sont élevés.

L'*instabilité de l'équipe de projet*, quelle qu'en soit la cause, pose le problème du transfert de connaissances : les concepteurs engrangent un ensemble de connaissances implicites sur le projet et son domaine que des modèles formalisés ne suffisent pas à transmettre. Cette connaissance est à reconstituer et les erreurs d'interprétation peuvent avoir des conséquences sur les délais et sur la cohérence de la conception.

La grille d'analyse des risques et le profil de risque d'un projet

Ces différents facteurs de risque ne peuvent pas être gérés de façon semblable. Leur variété requiert une analyse du *profil de risque*, dans laquelle chaque type reste identifié. Chaque facteur est précisé par plusieurs critères, et chaque critère est mesuré sur une échelle de 1 à 4 par des métriques. La grille détaillée est donnée en annexe C. Elle est utilisée au chapitre 9, pour illustrer la mise en œuvre de l'analyse de risque.

- La taille du projet est mesurée par sa charge estimée, sa durée prévue et l'ampleur de sa couverture fonctionnelle.

- La difficulté technique est mesurée par l'expérience capitalisée de l'entreprise sur les techniques à utiliser, la diffusion de ces techniques sur le marché, les contraintes de performance et l'existence de la direction informatique interne.
- Le degré d'intégration est mesuré par les flux entre la future application et d'autres applications, ainsi que par le nombre d'applications connexes en cours d'évolution.
- La configuration organisationnelle est mesurée par le nombre de Directions assurant la maîtrise d'ouvrage, l'appui de la Direction Générale, ainsi que par l'implication et la pérennité du promoteur.
- Le changement est mesuré par les degrés d'évolution fonctionnelle, organisationnelle et technique, ainsi que par le nombre de sites concernés et les éventuels impacts sociaux.
- L'instabilité de l'équipe de projet est influencée par plusieurs critères : la durée du projet, la part de sous-traitance de la maîtrise d'œuvre, le degré de rotation ou de mobilité du personnel dans l'entreprise, la qualité des relations entre maîtrise d'ouvrage et maîtrise d'œuvre, l'attrait des techniques utilisées, la conjoncture du marché de l'emploi et l'image du projet.

Ces métriques permettent d'établir le profil de risque du projet, dont un exemple est donné à la figure 6.6.

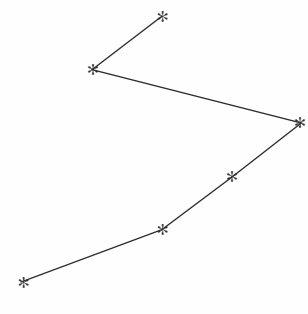
Nature du risque	Degré du risque pour le projet 0 1 2 3 4
Taille du projet	
Difficulté technique	
Degré d'intégration	
Configuration organisationnelle	
Changement	
Instabilité de l'équipe de projet	

Figure 6.6 — Profil de risque d'un projet.

Si plusieurs risques majeurs apparaissent sur le profil, leur réduction est une condition de la réussite du projet. C'est la première mesure du contrôle des risques, qui est développé au paragraphe 6.4.

6.3 LE PLAN DE MANAGEMENT DU PROJET

L'identification des risques conduit, en général, à prendre des mesures qui vont modifier le contenu du projet, voire dans certains cas le contenu du produit avec l'accord du commanditaire. Tous les aspects du projet, tel qu'ils ont été planifiés, peuvent être touchés. Lorsque le projet est géré de façon rigoureuse, les résultats de la planification, et leurs modifications ultérieures, sont consignés dans un document appelé le *plan de management du projet*.

Celui-ci prend en compte l'unicité de tout projet et représente le passage de la théorie à la pratique. Les différents choix issus de la planification (découpage, délais, ressources, contenu…) sont donc traduits dans un plan propre à chaque projet. Par exemple, le choix d'un modèle de développement en cascade doit préciser le nombre et le contenu des étapes ; les modalités de participation doivent spécifier les acteurs concrets avec leurs rôles…

La norme ISO10006 définit le plan de management de projet comme *un document qui spécifie les exigences permettant d'atteindre les objectifs du projet*, et comprend les aspects qualité, ressources, planning, budget et gestion des risques.

L'AFITEP et l'AFNOR précisent que *le plan de management de projet (PMP) est un document essentiel établi au début du projet sous l'autorité du chef de projet* et qu'il comprend la description de l'objectif, l'exposé des moyens nécessaires au cours du projet, les activités envisagées, l'analyse des risques et les moyens pour les traiter, le planning général et le budget, les procédures et normes qui s'appliquent au projet. Ce plan *peut être révisé en fonction des évolutions du projet, mais la traçabilité doit être strictement assurée* [AFITEP, 2000].

Le PMBOK attire l'attention sur le caractère officiel de ce plan entre les parties prenantes. Il est en effet défini comme un *document formel et approuvé qui définit les modes d'exécution, de surveillance et de maîtrise du projet* [PMI]. Tous les aspects planifiés peuvent s'y retrouver dans des plans subsidiaires (plan qualité, échéancier…).

La prise en compte de l'analyse des risques, c'est-à-dire leur contrôle, conduit souvent à modifier le plan de management du projet. Nous allons d'abord présenter de façon générale les types de réponses aux risques et nous ciblerons ensuite sur la prise en compte des risques dans les projets système d'information, suite à l'établissement du profil de risque décrit au paragraphe 6.2.4.

6.4 LE CONTRÔLE DES RISQUES

Le contrôle des risques suppose souvent de faire appel à des solutions imaginatives. On peut citer à ce propos la « Pensée latérale » proposée par le médecin et psychologue Edward de Bono[1]. Cette technique de recherche de solution consiste à approcher le problème non pas de face (verticalement), selon des schémas classiques, mais sous différents angles (latéralement), en osant des solutions peu orthodoxes. Le principe général est le suivant. Après avoir tenté de reconnaitre les idées dominantes ou polarisantes, c'est-à-dire qui orientent vers une même catégorie de solutions, on recherche de nouvelles façons d'envisager les choses. Certaines idées provocantes (*provocative operation*) peuvent ainsi être énoncées, non pour donner une solution, mais pour modifier la vision des choses et introduire une certaine liberté, favorisant ainsi le relâchement du contrôle de la pensée verticale.

Sans nous étendre davantage sur les techniques de créativité, nous allons passer en revue les mécanismes classiques de réponse aux risques.

6.4.1 Les stratégies génériques de réponse aux risques

On distingue cinq types de comportement face à un risque identifié : éviter, transférer, atténuer, accepter et élaborer une réponse conditionnelle.

Éviter

On modifie le plan de management du projet pour éliminer la menace. Le risque a donc disparu.

Par exemple, si le risque majeur provient de la nouveauté technologique, on évite le risque en se repliant sur une technologie éprouvée. On peut aussi acquérir de l'expérience, c'est-à-dire former l'équipe de projet concernée par les aspects techniques, de façon à lever les incertitudes. Si la menace vient d'un sous-traitant qui pourrait se montrer défaillant (délai, qualité), on l'évitera en renonçant à externaliser le développement. Si enfin l'on craint le refus du système par les utilisateurs, on peut chercher à l'éviter en associant les utilisateurs à la conception.

Transférer

On modifie le plan de management du projet pour détourner la menace et ses conséquences possibles sur un tiers. Le risque n'a donc pas disparu.

Par exemple, si la menace est de dépasser le budget, un transfert pourra consister à sous-traiter au forfait[2] (pour le client) et inversement à travailler en

1. E. de Bono, *Au service de la créativité dans l'entreprise : la pensée latérale*, EME, 1973.
2. Les aspects concernant la sous-traitance sont décrits au chapitre 7.

régie (pour le maître d'œuvre). Si le sous-traitant risque de dépasser le délai, on peut utiliser des clauses contractuelles pour se garantir financièrement (pénalités de retard). Si l'on craint des vols de matériel utilisé sur le projet, on peut prendre une assurance moyennant une prime : les conséquences financières seront supportées par l'assureur.

Atténuer

On modifie le plan de management du projet pour réduire la probabilité et/ou les conséquences d'un événement à risque. C'est une stratégie fréquemment utilisée.

Par exemple, si l'on redoute une faible qualité des livrables logiciels, on peut prévoir d'augmenter le nombre de tests. S'il peut y avoir des erreurs de compréhension des spécifications détaillées par le client, on peut réduire ce risque en élaborant un prototype ou mieux gérer ses conséquences en prévoyant des ressources supplémentaires au moment de la recette pour effectuer des ajustements.

Accepter

On ne modifie pas le plan de management du projet et on accepte le risque tel qu'il se présente, soit parce qu'on ne pense pas qu'il va se réaliser ou qu'on ne redoute pas ses conséquences, soit parce qu'on ne trouve pas d'autre réponse.

On distingue deux formes d'acceptation. L'acceptation passive consiste à ne rien faire du tout et à prendre des mesures à chaud en cas de concrétisation du risque, c'est-à-dire élaborer le cas échéant un plan dit « palliatif ». L'acceptation active consiste à prévoir une provision pour faire face aux aléas, par exemple une part de budget ou des ressources qui seront utilisés si le risque se réalise.

Élaborer une réponse conditionnelle

Sans modifier le plan de management du projet, on prépare un plan annexe qui ne sera mis en œuvre que sous certaines conditions de réalisation du risque. Celui-ci est appelé « plan de secours », il modifiera le déroulement du projet. On peut, par sécurité si les conséquences du risque peuvent être très graves, préparer également un « plan de repli » au cas où le plan de secours serait inefficace.

Par exemple, si l'on craint que le fournisseur de matériel ait du retard, on peut prévoir dans un plan de secours une action de pression sur le fournisseur, et en repli un nouvel échéancier.

Pour contrôler les risques, il s'agit souvent de prendre en compte un ensemble de risques dans une stratégie globale que l'on peut appeler « stratégie de projet ».

6.4.2 La stratégie de projet

Après établissement d'un profil de risque et une réduction éventuelle, le chef de projet doit élaborer une *stratégie de projet*, ce qui signifie :

- choisir un modèle de cycle de vie et l'adapter au projet ;
- mettre en place un dispositif de coordination, avec ses composantes personnelles et impersonnelles ;
- choisir les modalités de participation des utilisateurs ;
- mettre en place un dispositif de contrôle, c'est-à-dire un tableau de bord permettant le pilotage du projet.

Les modèles de cycle de vie ont été décrits au chapitre 2. La coordination et la participation ont été présentées au chapitre 5. Le dispositif de contrôle figure au chapitre 7 qui décrit les facettes du pilotage.

Le risque lié à la *taille* se gère de trois façons.

Si la visibilité est faible, les engagements doivent être limités mais précis. Le modèle de développement privilégié est celui de la spirale : chaque cycle comprenant une analyse du risque permettra un affinement ou une redéfinition des objectifs.

Pour contrôler l'avancement d'une équipe importante, il faut s'appuyer sur un dispositif de coordination formelle et sur un système formalisé de contrôle de l'avancement du projet, comme une nécessaire extension de la mémoire humaine individuelle.

Dans la maîtrise de la cohérence, la coordination personnelle qui trouve ses limites devant le nombre, doit être soutenue par des moyens formels, comme une administration de système d'information.

Le risque *technique*, s'il provient de la complexité de mise en œuvre d'un outil, se gère comme un problème de chargement, classique en gestion de projet industriel : il s'agit de repérer et de faire intervenir la bonne ressource au bon moment. Si les besoins sont stables et que le risque majeur est lié à la programmation, on choisira un modèle de développement en cascade avec deux étapes majeures : spécifications précises, suivies d'une étape programmation sans modification des spécifications. Sinon, un modèle en W est préférable, le prototype permettant de stabiliser les besoins.

Si le risque découle de la nouveauté, il faut alors utiliser un modèle en W, dont la première partie est axée sur la maîtrise de l'outil ou de la performance.

Le risque lié à l'*intégration* appelle une coordination personnelle en renfort d'un système d'information sur le futur système. Un exemple de ce dernier (administration des données) est présenté au chapitre 5. Le modèle de développement en V facilite l'intégration modulaire.

Le risque lié à la *configuration organisationnelle* (les utilisateurs) conduit à rechercher un consensus décisionnel, sur un champ qui pourra être limité. La participation a pour objectif la prise de décision. Le modèle du cycle RAD peut la favoriser.

Le risque lié au *changement* se gère par la participation des différents acteurs. Il faut user de modalités adaptées à l'objectif, qui peut être la créativité (conception d'un nouveau système), l'adhésion (éviter les rejets) ou la prise de décision. Si les contraintes de budget et délai sont faibles et s'il n'y a pas de problème d'intégration, alors le modèle de développement évolutif sera le plus adapté.

Le risque lié à l'*instabilité de l'équipe de projet* (les concepteurs) peut être limité à la fois par un système d'information sur le futur système (documentation de projet) et par une coordination personnelle de type supervision directe qui favorise la transmission des connaissances.

Dans le cas particulier où l'on veut mener le projet dans un délai très bref, sans sacrifier la qualité, une méthode agile sera privilégiée. Un tel choix est examiné du point de vue des risques au § 6.5.

Ainsi, le développement d'un système d'information comporte trois processus (figure 6.7) : l'un, rationnel, vise la production d'une application ; le deuxième, politique, cherche à ce que les décisions nécessaires soient prises ; le troisième, psychologique, gère les changements individuels et sociaux liés au nouveau système d'information.

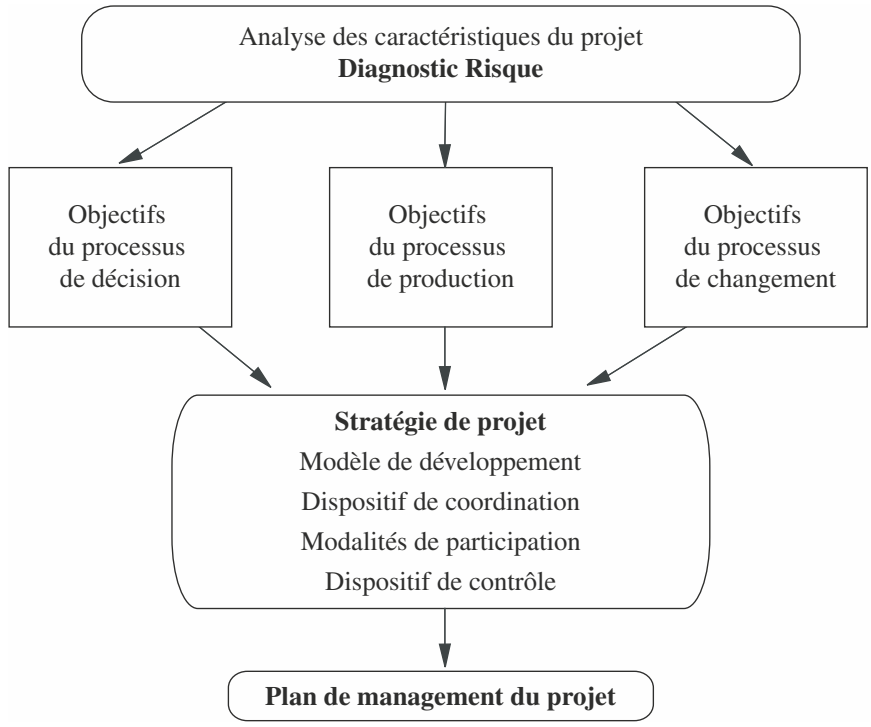


Figure 6.7 — Élaboration d'un plan de management de projet.

Les processus sociaux (décision, motivation) introduisent pour le même problème une variété importante. La stratégie de projet doit articuler les trois processus. Ainsi, certaines tâches de production (élaborer un modèle, construire un prototype…) pourront-elles parfois avoir un objectif politique ou social.

6.4.3 L'audit en cours de projet

Le contrôle des risques peut conduire à une réévaluation périodique de la situation du projet au regard des risques, notamment sous forme de revue structurée à mener en interne par l'équipe de management de projet. Nous présentons ci-dessous un exemple d'une telle démarche d'audit du projet sous l'angle des risques.

Le Standish Group a proposé, à partir d'une soixantaine d'interviews approfondies menées avec des responsables informatiques, une grille d'évaluation du potentiel de réussite d'un projet en cours[1]. Cette grille comprend dix critères, pesant chacun d'un certain poids dans la réussite du projet (figure 6.8). Il apparaît ainsi que l'engagement de la maîtrise d'ouvrage, utilisateurs et décideurs, avec une définition claire des besoins pèse pour moitié dans le succès d'un projet.

Critères de réussite	Poids du critère
Implication des utilisateurs	19
Soutien de la hiérarchie	16
Définition claire des besoins	15
Plan de développement correct	11
Attentes réalistes	10
Découpage du projet en petites étapes	9
Compétences dans l'équipe de projet	8
Appropriation du projet par les acteurs du projet	6
Vision claire de la raison d'être et des objectifs du projet	3
Productivité et motivation de l'équipe de projet	3
	100

Figure 6.8 — Critères de réussite d'un projet.

1. On peut la trouver sur la partie publique du site http://www.standishgroup.com.

Pour déterminer la valeur de ces critères lors de l'audit d'un projet particulier, les auteurs de l'étude proposent une série de questions qui précisent la signification de chaque critère. Chaque question pèse d'un poids équivalent dans l'évaluation du critère.

1. Implication des utilisateurs
   – A-t-on réellement les utilisateurs qu'il faut ?
   – Les a-t-on impliqués suffisamment tôt et suffisamment souvent ?
   – Les relations avec les utilisateurs sont-elles de bonne qualité ?
   – A-t-on facilité leur implication ?
   – A-t-on réellement trouvé leurs besoins ?
2. Soutien de la hiérarchie
   – A-t-on mobilisé les responsables clés ?
   – La réussite du projet est-elle un enjeu important pour les responsables clés ?
   – L'échec est-il acceptable ?
   – A-t-on un plan de projet bien défini ?
   – Y a-t-il un enjeu fort pour l'équipe de projet ?
3. Définition claire des besoins
   – A-t-on une vision concise de la stratégie, des enjeux, du court, moyen et long terme ?
   – A-t-on fait une analyse des fonctions attendues ?
   – A-t-on fait une analyse de risque ?
   – A-t-on fait un dossier d'opportunité ?
   – A-t-on défini des métriques pour évaluer la réussite du projet ?
4. Plan de développement correct
   – A-t-on formalisé une description du projet ?
   – A-t-on formalisé une description de la solution ?
   – A-t-on une équipe de projet adéquate ?
   – Les spécifications sont-elles stabilisées ?
   – A-t-on établi une planification avec des étapes réalistes ?
5. Attentes réalistes
   – A-t-on des spécifications claires ?
   – A-t-on établi des priorités entre les besoins ?
   – A-t-on fait un découpage en petites étapes ?
   – Peut-on gérer le changement ?
   – Peut-on faire un prototype ?
6. Découpage du projet en petites étapes
   – Utilise-t-on la règle des 80/20 pour se centrer sur les 20 % des fonctions qui vont satisfaire 80 % des besoins des utilisateurs ?
   – Utilise-t-on une conception descendante ?
   – A-t-on arrêté des dates butoirs ?
   – Utilise-t-on un outil de prototypage ?
   – Peut-on mesurer l'avancement ?
7. Compétences de l'équipe de projet
   – A-t-on une bonne connaissance des compétences requises ?
   – A-t-on rassemblé les bonnes ressources ?

– A-t-on défini un plan de formation ?
– A-t-on prévu des incitations pour motiver l'équipe ?
– L'équipe va-t-elle rester jusqu'à la fin du projet ?

8. Appropriation du projet par les acteurs du projet
   – Les rôles ont-ils été correctement définis ?
   – L'organisation a-t-elle été clairement annoncée ?
   – Chaque acteur du projet connaît-il son rôle et ses responsabilités ?
   – Les incitations pour motiver les acteurs du projet contribuent-elles à la réussite du projet ?
   – Y a-t-il un engagement de tous les acteurs du projet ?
9. Claire vision de la raison d'être et des objectifs du projet
   – Tous les acteurs partagent-ils la même vision du projet ?
   – Cette vision est-elle en phase avec les objectifs de l'entreprise ?
   – Les objectifs du projet sont-ils réalistes et peuvent-ils être atteints ?
   – Les objectifs du projet sont-ils mesurables ?
   – A-t-on prévu des contrôles de bons sens, honnêtes et réguliers ?
10. Productivité et motivation de l'équipe de projet
   – A-t-on mis en place des incitations (primes, augmentations, promotions) ?
   – Se concentre-t-on sur les livrables annoncés ?
   – Chacun s'est-il approprié le projet ?
   – Y a-t-il un travail d'équipe ?
   – A-t-on bâti un climat de confiance ?

La somme des valeurs pondérées de chaque critère donne le potentiel de réussite du projet. Cette grille d'analyse permet, le cas échéant, de redresser la barre et de mettre en œuvre des actions pour augmenter les chances de succès.

6.5 LES MÉTHODES AGILES ET LA GESTION DES RISQUES

Pour étudier la relation entre management des risques et méthodes agiles, on peut d'abord se demander si les préconisations agiles modifient le profil de risque du projet, tel qu'exposé au § 6.2.4. On peut aussi examiner si leur utilisation pourrait avoir un effet sur les facteurs de succès du projet présentés au paragraphe précédent. Il est enfin nécessaire de réfléchir sur les conditions nécessaires à la mise en œuvre d'une méthode agile.

6.5.1 Méthodes agiles et profil de risque

La taille du projet est le premier facteur de risque de la figure 6.6. Si l'on regarde sa valorisation avec les métriques données à l'annexe C, on peut considérer qu'un projet agile devrait généralement se situer à un niveau bas, compte tenu des préconisations en matière de durée visée (moins de 18 mois) et de taille d'équipe (moins de dix personnes).

Le deuxième facteur, la difficulté technique, comporte différents critères qui peuvent être regroupés en trois grandes rubriques : expérience technique, contrainte de performance et présence d'une responsabilité technique interne à l'entreprise. Pour la première, l'excellence technique fait partie des principes du Manifeste agile, ce qui se traduit dans les méthodes par une attention à la progression des compétences, mais aussi à la sélection des développeurs. Pour la seconde, certains projets peuvent inclure des exigences de performance élevées. Même si elles peuvent être discutées avec le client et si le groupe peut être mobilisé pour trouver des solutions efficaces, ce critère est variable selon les projets, et le risque ne peut être considéré a priori comme faible.

Le troisième facteur, le degré d'intégration, ne peut être considéré comme systématiquement réduit, même s'il est pris en compte par une intégration continue des composants livrés, qui conduit à faire plus rapidement des tests d'interopérabilité avec les systèmes devant communiquer avec le futur produit.

Une faible valeur du quatrième facteur de risque, la configuration organisationnelle, peut être considérée comme une condition à l'utilisation d'une méthode agile. Cependant, les méthodes visant à faire collaborer des décideurs (sessions JRP ou JAD, ateliers facilités de DSDM…) ont en principe un effet positif sur l'allègement des procédures de décision et l'obtention d'un consensus. De plus, un rôle correspondant à la notion de promoteur est identifié dans plusieurs méthodes. Il est donc permis de penser que ce facteur de risque, dont le degré peut être élevé dans certains projets, sera réduit par une application réussie des techniques agiles concernant les utilisateurs.

Le risque lié au changement peut être réduit, en ce qui concerne la plupart des critères, par la participation aux itérations de développement, si toutefois des relais sont mis en place pour l'ensemble des utilisateurs. Sans remplacer une gestion du changement, l'utilisation d'une méthode agile peut être un élément facilitant le changement.

Le dernier facteur, l'instabilité de l'équipe, est soumis à différents critères, internes (par exemple, l'intérêt du projet) ou externes (par exemple, les possibilités offertes par le marché de l'emploi). On peut toutefois relever que plusieurs préconisations des méthodes agiles réduisent l'apparition ou les conséquences d'un turn-over de l'équipe : une courte durée du projet, les pratiques de programmation en binômes, la rotation des équipes, ainsi que le souci des conditions de travail de l'équipe.

En conséquence, on peut a priori considérer qu'il devrait y avoir une réduction du risque lorsque l'on applique une méthode agile. Elle est liée d'une part à la définition du contenu et du périmètre de ces projets, d'autre part aux principes et techniques de management de la dimension humaine qui sont proposés par ces méthodes.

6.5.2 Méthodes agiles et facteurs de succès

Prenant un autre point de vue, nous allons examiner ce que l'on peut dire du potentiel de réussite d'un projet mené dans un environnement agile.

Le premier facteur de la figure 6.8, l'implication des utilisateurs, qui pèse pour 19 % dans le succès d'un projet, devrait être vérifié, dans la mesure où les méthodes agiles insistent sur l'importance des relations avec les utilisateurs et proposent différents moyens pour favoriser leur implication (sessions, ateliers, prototypes…).

Le deuxième facteur, le soutien de la hiérarchie, est moins évident et semble difficilement pouvoir être garanti de la simple utilisation d'une méthode agile.

Le troisième facteur, la définition claire des besoins, doit être bien compris. En effet, si les méthodes agiles ne conduisent pas à la production de dossier d'analyse et de spécifications, elles mettent au cœur de l'organisation préconisée une prise en compte à la fois précise et dynamique des exigences. On peut donc considérer que ce facteur, d'un poids de 15, est en général vérifié.

Un raisonnement similaire s'applique pour le quatrième facteur, un plan de développement correct, qui compte pour 11 %. L'approche itérative permet de visualiser progressivement les différents aspects du futur produit, ce qui compense l'absence de formalisation de la solution et l'instabilité des spécifications. De plus, la priorisation des fonctionnalités par le client à chaque itération, ainsi que l'exigence d'un délai court, sont des caractéristiques des méthodes agiles qui peuvent conduire à qualifier la planification de réaliste.

Le cinquième et le sixième facteur, attentes réalistes et découpage du projet en petites étapes, sont très vraisemblablement vérifiés si les préconisations agiles sont suivies, ce qui apporte une contribution respective de 10 % et 9 % au succès du projet.

Les quatre facteurs suivants concernent la répartition des rôles et les qualités de l'équipe de projet. Comme une attention particulière est portée à l'organisation du travail et à l'animation d'équipe, on peut retenir que ces critères sont validés.

Il en découle qu'un projet se déroulant selon des préconisations agile bénéficierait a priori d'un potentiel de réussite de 84 %. Il faut toutefois garder à l'esprit que l'adoption d'une méthode agile suppose certaines conditions.

6.5.3 Les conditions d'utilisation d'une méthode agile

Les méthodes agiles apparaissent prometteuses en ce qui concerne les chances de réussite du projet. Cependant, en 2006, selon une recherche du groupe Forrester[1],

1. Cette étude est vendue sur http://www.forrester.com.

seuls 17 % des entreprises nord-américaines et européennes, utilisent une méthode agile. Ce paradoxe apparent s'explique par les défis[1] que représente le passage à une méthode agile, que l'on peut regrouper en plusieurs catégories :

- La culture de l'entreprise et ses méthodes de management, en particulier les pratiques de délégation et de collaboration, ainsi que les styles de leadership privilégiés. En ce qui concerne la motivation, la théorie des facteurs d'hygiène de Herzberg, brièvement décrite au § 5.3.4, peut expliquer des difficultés à mobiliser les développeurs.
- Les caractéristiques des équipes de développement peuvent être un obstacle au passage à des approches agile. Le travail en binôme, les tests croisés entre développeurs ou l'intégration fréquente des composants développés modifient profondément le quotidien d'un développeur travaillant habituellement en solitaire. Par ailleurs, compte tenu des expériences observées, il est difficile d'affirmer que de telles approches ne demandent pas des compétences au-dessus de la moyenne. De plus, un travail efficient entre développeur et utilisateur suppose une confiance de la part du chef de projet, qui traditionnellement est responsable de la plupart des décisions sur le contenu du projet.
- L'appui systématique sur des procédures entre en partie en conflit avec l'esprit d'adaptation promu par les méthodes agiles. Plus précisément, l'approche itérative et la tolérance aux changements de spécifications requièrent une souplesse, avec laquelle sont peu familiers les acteurs des projets gérés selon un cycle de vie qui est planifié et piloté de façon plus fine. Ainsi dans XP, les itérations de livraison correspondent en général à des jalons fixés et dont les dates seront respectées, alors que les itérations de développement peuvent fluctuer.
- Même si les approches agiles sont essentiellement des principes et techniques d'organisation, elles présupposent un environnement technique particulier. D'une part, la décomposition du futur système doit pouvoir aller jusqu'à un niveau élémentaire de développement, sans générer un besoin de coordination insoutenable. Ceci est possible avec des technologies orientées objet, mais difficilement compatible avec des grands systèmes. D'autre part, le modèle itératif suppose l'utilisation d'outils de développement rapide.

Au-delà de cette vision un peu dichotomique, on peut aussi considérer que cohabitent dans une même entreprise, différentes cultures, des niveaux de compétence variés et fluctuants, ainsi que des environnements technologiques

1. Voir S. Nerur, R. Mahapatra et G. Mangalaraj, « Challenges of migrating to agile methodologies », Communication of the ACM, 48, 5, 2005.

divers. Se pose alors la question du choix d'une approche agile pour un projet particulier.

Différents points portant sur les trois sommets du triangle projet peuvent ainsi être soulignés.

- Si les objectifs du projet sont très incertains et le nombre de parties prenantes élevé, l'utilisation d'une méthode agile sera vraisemblablement inefficace. En effet, toutes les méthodes agiles considèrent, d'une façon ou d'une autre, que la mission du futur système doit être claire au départ, même si elle peut se décliner à travers des fonctionnalités prenant des formes diverses. Sinon, l'incertitude demeurera dans les itérations, ou bien les utilisateurs pourront indiquer des directions peu compatibles entre elles.
- La faiblesse et l'insuffisance des ressources par rapport à l'objectif, aussi bien du côté de la maîtrise d'ouvrage (utilisateurs pertinents) que de la maîtrise d'œuvre (particulièrement les développeurs), posent problème, quelle que soit l'approche retenue. Cependant, dans une approche agile, le problème est amplifié à cause de l'exigence de mobilisation collective et celle de qualité élevée, et devrait faire l'objet d'une analyse de risque préalable.
- Le délai est le point sur lequel les méthodes agiles se sont historiquement différenciées. Cependant, des méthodes comme XP ou SCRUM considèrent que la taille limite de l'équipe de développement est de dix personnes, ce qui implique un périmètre limité pour le projet. Cela suppose, dans certains cas, une redéfinition du contenu du produit à développer. D'autres méthodes, comme Crystal, considèrent toutefois une taille plus élevée. Nous évoquons la question de l'application des méthodes agiles à des projets de grande taille au chapitre 18.

Ainsi, une analyse des caractéristiques du projet et de son environnement peut conduire à retenir une méthode agile et permettre un changement culturel local.

7

Le pilotage du projet

7.1 LE CONCEPT DE PILOTAGE

Le concept de pilotage a été étudié par la cybernétique, science du contrôle des systèmes. Étymologiquement, la cybernétique est l'art du pilote. Nous allons en rappeler les principes de base.

Un système est classiquement représenté comme une transformation. Celle-ci traduit la valeur ajoutée entre un flux entrant et un flux sortant (figure 7.1). Un projet système d'information peut ainsi être représenté comme un système transformant une demande de gestion d'information en un dispositif logiciel/réseau/organisation installé.

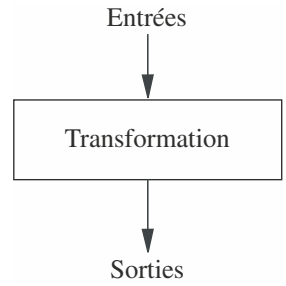


Figure 7.1 — Schéma générique d'un système.

Un système est *déterminé* si, connaissant les valeurs des entrées, on peut prédire les valeurs des sorties. Les projets système d'information ne sont pas des systèmes déterminés. Même si les actions de la transformation sont prévues et

planifiées (normes, règles, procédures, calendrier.), les projets ne se déroulent jamais exactement comme prévu. L'indétermination peut avoir différentes causes :

- l'ignorance de l'existence de certaines entrées (besoins mal identifiés) ;
- l'ignorance de certaines sorties, notamment l'effet retour de certaines sorties (résultats des points de validation par le maître d'ouvrage) ;
- l'incertitude de certaines règles de transformation (sous-estimation des charges) ;
- l'ignorance des effets de l'environnement (modification de la réglementation en cours de déroulement de projet).

Les aléas exigent de modifier le réglage de la transformation si l'on veut obtenir les sorties désirées : c'est le rôle du pilotage. Pour cela, le système de pilotage a besoin de moyens lui permettant de voir et d'agir. Ce sont les variables essentielles et les variables d'action.

- Les *variables essentielles* sont des sorties particulières du système : ce sont des critères permettant de mesurer la réussite de la mission.
- Les *variables d'action* sont des entrées particulières du système, qui modifient le fonctionnement prévu. Le résultat de la transformation varie suivant les valeurs assignées aux variables d'action (figure 7.2).

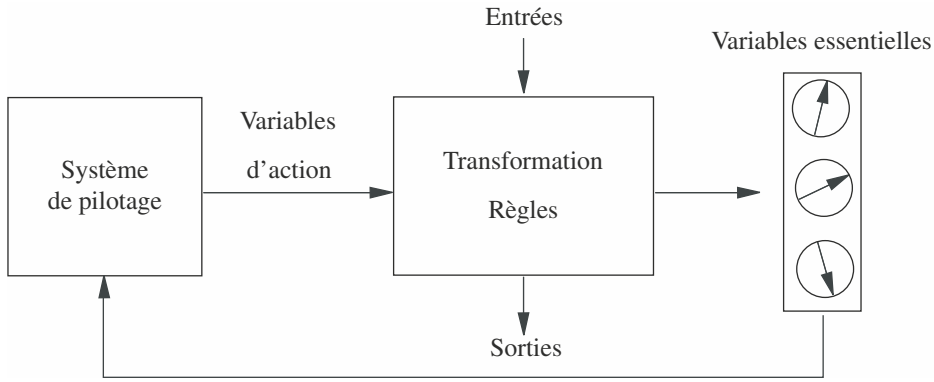


Figure 7.2 — Schéma de pilotage d'un système.

Pour pouvoir piloter son projet, le chef de projet a besoin de variables essentielles : c'est son *tableau de bord*. Il lui permet de détecter le plus rapidement possible d'éventuels problèmes et d'éviter ainsi des situations irrémédiables. Il doit également disposer de variables d'action à actionner en fonction des résultats figurant sur son tableau de bord. Par exemple, si un membre de l'équipe de

projet se casse la jambe au ski, il réaffecte ses tâches à d'autres personnes ou fait appel à des ressources supplémentaires. Si des difficultés imprévues se traduisent par une charge de travail supérieure à celle qui avait été estimée, il faut éventuellement négocier avec le maître d'ouvrage une modification du périmètre fonctionnel pour pouvoir respecter un délai impératif.

Le pilotage d'un système est l'ensemble des processus qui permettent de maîtriser et de guider son fonctionnement et son évolution. Les deux concepts clés du pilotage sont le contrôle et la régulation.

Le *contrôle* comprend :

- la prise en compte d'objectifs, c'est-à-dire l'établissement de variables essentielles (ce sont les variables du tableau de bord) et des plages admissibles pour chaque variable (identification des « zones rouges ») ;
- la détermination de moyens d'actions pouvant faire varier les résultats (fixation de variables d'action).

La *régulation* vise à maintenir le système dans les limites de fonctionnement que le système de contrôle a désignées. C'est le suivi (examen du tableau de bord) et la prise en compte des écarts (utilisation des variables d'action).

La difficulté de pilotage d'un système est proportionnelle à sa variété. On appelle *variété* d'un système le nombre d'états différents qu'il peut prendre. Il ne peut être totalement maîtrisé a priori que si le système de pilotage possède une variété au moins égale, c'est-à-dire qu'il y a autant de réponses que d'états possibles du système. C'est ce qu'on appelle la « loi de la variété requise ».

Il est illusoire et coûteux de chercher a priori une maîtrise complète des systèmes complexes. Plutôt que de construire des dispositifs de pilotage avec une variété élevée, on va miser sur des systèmes de pilotage pouvant s'adapter et apprendre.

L'*adaptation* est la faculté de faire face à de nouvelles situations. Si un système de pilotage possède à un instant t un registre R(t) de réponses à des états E(t) du système à contrôler S, il y a adaptation si S présentant un nouvel état e(t+1), pour lequel R(t) ne contient pas de réponse, le système de pilotage est capable de découvrir une réponse r(t+1) convenable.

Un système de contrôle est adaptatif si sa gamme de réponses croît, ainsi que sa capacité de sélection, c'est-à-dire sa variété. L'être humain est le plus grand générateur de variété.

L'*apprentissage* est la faculté de mémoriser et de cumuler l'adaptation. Il y a apprentissage si r(t+1) est mémorisée dans le registre des réponses, de telle façon que si e(t+1) se présente à nouveau, le système de contrôle propose la réponse r(t+1). Pour qu'il y ait adaptation, il faut des moyens de mémorisation.

Le rôle du chef de projet est rendu nécessaire par le besoin d'adaptation aux situations imprévues, requis tout au long du projet. Il s'appuie sur un tableau de bord. L'apprentissage nécessite un dispositif de capitalisation du savoir-faire dans l'entreprise.

Nous allons présenter une proposition de structure et de contenu d'un tableau de bord pour le chef de projet (paragraphe 7.2 à 7.4). Une ouverture sur le pilotage nécessaire en cas de sous-traitance est ensuite apportée (paragraphe 7.5). Ce chapitre se termine par quelques éléments de base pour une capitalisation d'expérience sur les projets (paragraphe 7.6).

7.2 LE TABLEAU DE BORD DU CHEF DE PROJET

7.2.1 L'objectif du tableau de bord

Le projet est planifié et organisé : le processus de production va démarrer. On a préalablement effectué un diagnostic sur les risques qui le menacent, ce qui a permis d'élaborer une stratégie de projet en conséquence. On a découpé le projet en tâches, dont on a évalué la charge. On a élaboré une planification des actions, en fonction du délai imparti, des charges, des contraintes d'enchaînement et des ressources, en utilisant notamment la méthode du chemin critique et le diagramme de Gantt.

La planification détaillée va servir de repère pour suivre l'avancement des travaux. Suivre l'avancement, pour un chef de projet, c'est pouvoir répondre à n'importe quelle question sur :

- ce qui a été produit : c'est l'avancement réel du projet ;
- ce qui a été consommé : ce sont les ressources utilisées ;
- les écarts entre le planifié et le réalisé ;
- l'origine des écarts, que ce soit une cause ayant des effets sur plusieurs tâches, par exemple l'indisponibilité d'une machine, ou un problème ponctuel lié à une tâche ou à une personne ;
- ce qu'il reste à faire.

Pour informer la maîtrise d'ouvrage et pour prendre les décisions de pilotage, le chef de projet a besoin de gérer un ensemble d'informations que l'on appelle le *système d'information projet* ou *système d'information de pilotage*. Celui-ci comprend :

- un *tableau de bord* ;
- en général complété par un *journal de bord*, journal où sont consignés au quotidien les événements, incidents ou faits spéciaux du projet.

Certaines actions de pilotage sont internes au projet, la décision en est prise par le chef de projet, d'autres relèvent d'un comité de pilotage du projet.

Les charges des projets système d'information sont principalement celles des ressources en personnel ou leur sont directement proportionnelles (locaux, équipement…). Par ailleurs, les écarts de spécification vont se traduire par des écarts (à venir) sur le calendrier. Nous allons donc étudier principalement le pilotage des délais. Nous verrons ensuite les indicateurs normalisés du pilotage des coûts.

Le tableau de bord ne doit contenir que le minimum d'information que l'on a l'intention d'analyser. Une erreur fréquemment rencontrée consiste à collecter périodiquement une masse d'informations que l'on est ensuite incapable d'analyser. Il s'agit de réduire la variété du tableau de bord à la capacité d'adaptation du chef de projet.

Le dispositif mis en place a un coût : il doit être adapté aux caractéristiques du projet.

Ainsi, le degré de formalisation doit proportionnel à la taille de l'équipe ; au-delà de trois ou quatre personnes, la coordination personnelle s'avère insuffisante.

La fréquence des mesures est dépendante de la capacité de réaction : elle varie entre la semaine et le mois selon les projets.

Le dispositif doit inclure un suivi individuel, afin de responsabiliser chacun des membres de l'équipe : la réussite collective passe par l'engagement individuel dans le projet commun.

Le tableau de bord contient ainsi deux niveaux :

- le *suivi individuel*, qui permet de détecter d'éventuelles difficultés pour un intervenant ou sur une tâche ;
- le *suivi du projet*, qui sert de base à un point d'avancement périodique avec le maître d'ouvrage.

7.2.2 Le suivi individuel

Il se base sur la *liste des tâches*, qui sont affectées individuellement. Le descriptif de chaque tâche comprend un identifiant — chaque tâche est repérée de façon unique — et les éléments ayant servi pour l'évaluation de sa charge (méthode, unités d'œuvre, poids standard, ratio…). À chaque tâche, on associe trois types de charge :

- la *charge initiale* : c'est celle de l'estimation qui a servi à faire la planification détaillée. Cette valeur doit toujours être conservée. Sa modification priverait l'entreprise d'une possibilité d'apprentissage ;

- la *charge affectée* : c'est la personnalisation de la charge initiale en fonction de l'expérience et de la compétence de celui qui va l'effectuer. Elle peut être supérieure ou inférieure à la charge initiale. Cette valeur représente le contrat entre le chef de projet et l'acteur concerné. Elle n'est en général pas visible pour le maître d'ouvrage. Elle doit être utilisée avec précaution ;
- la *charge actualisée* : en cours de déroulement du projet, mais toujours avant que la tâche ne soit commencée, des éléments nouveaux peuvent conduire à revoir l'estimation initiale. Par exemple, l'étude technique a fait apparaître des difficultés imprévues et l'on sait maintenant que la charge de programmation a vraisemblablement été sous-estimée de 20 %. On ne modifie pas la charge initiale, mais on négocie avec le maître d'ouvrage une actualisation. La valeur de la charge actualisée est alors prise en compte pour une nouvelle planification.

La base d'alimentation du tableau de bord est le *compte rendu d'activité*, aussi appelé *compte rendu d'avancement*, rédigé périodiquement, en général en fin de semaine, par chaque intervenant affecté au projet. Le compte rendu doit être régulier. Les chiffres doivent être le plus exacts possible, sinon tous les éléments calculés du tableau de bord seront faussés. Il comprend, par intervenant et par tâche :

- le *temps passé* T : c'est la consommation qui sera imputée au projet ;
- le *reste à faire* R : c'est l'estimation par l'intervenant du temps nécessaire à l'achèvement de la tâche. Ce chiffre peut être égal, inférieur ou supérieur à la différence (charge affectée – temps passé).

Les tâches hors projet figurent également sur le compte rendu d'activité (figure 7.3).

Mois : juin 2001 – semaine 1	**Tâche**	**Charge affectée**	**Temps passé**	**Reste à faire**
Jean-Claude	Réalisation jeu d'essai du module Comptabilité	10	3	6
	Maladie		1	
	Représentation syndicale		1	
Roger	Programmation du lot Imputation	8	4	5
	Congé		1	

Figure 7.3 — Exemples de compte rendu d'activité.

Le *récapitulatif mensuel* permet un suivi au plus fin. On y trouve pour chaque tâche et chaque semaine du mois :

- le temps passé ;
- le reste à faire ;
- l'*avancement*, calculé comme une différence entre les deux dernières évaluations du reste à faire :

$$\textit{avancement en fin de période } n = \textit{reste à faire en fin de période } (n-1) - \textit{reste à faire en fin de période } n$$

Cette définition de l'avancement est spécifique des productions immatérielles, où l'on ne dispose pas d'une mesure physique permettant de mesurer objectivement la production effective. Par exemple, on peut avoir écrit la moitié du nombre présumé de lignes de programme sans pouvoir dire que le travail est à moitié fait. C'est parfois plus, parfois moins. Aussi, doit-on se baser sur la diminution du travail restant (figure 7.4).

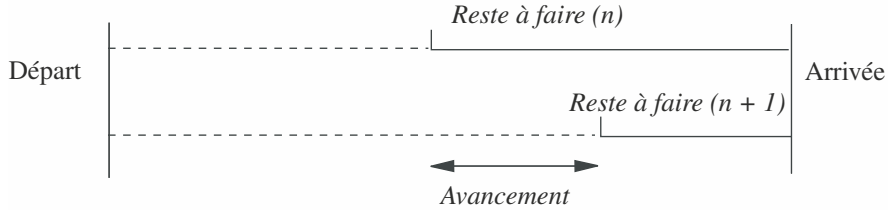


Figure 7.4 — La mesure de l'avancement.

Cette définition implique que l'avancement pourra être supérieur, inférieur ou égal au temps passé.

Un exemple de récapitulatif est donné à la figure 7.5 pour deux tâches A et B dont la charge affectée est respectivement de 12 et 10 jours. Dans la partie droite du tableau, on a effectué le total du mois. L'attention du chef de projet est attirée quand l'avancement est inférieur au temps passé.

Mai 2001	Tâche	Semaine 1			Semaine 2			Semaine 3			Semaine 4			Total mois		
		T	R	A	T	R	A	T	R	A	T	R	A	T	R	A
Jean-Claude	A (12 J)	4	8	4	5	3	5	1	0	3				10	0	12
	B (10 J)							3	7	3	5	2	5	8	2	8
														18		20

Figure 7.5 — Exemple de récapitulatif mensuel.

Le *bilan individuel mensuel* donne pour chaque intervenant une photographie de sa performance.

La partie gauche du tableau de la figure 7.6 donne les chiffres clés du mois n, par tâche et pour le total des tâches du mois. On y trouve :

- la charge affectée ;
- le reste à faire à la fin du mois précédent (R_{n-1}) ;
- le temps passé (T_n) ;
- le reste à faire à la fin du mois n (R_n) ;
- l'avancement du mois n : $A_n = R_{n-1} - R_n$;
- le *coefficient d'utilisation* de la ressource pendant le mois n :

$$\frac{T_n}{\text{Nombre de jours ouvrables du mois qui auraient dû être consacrés au projet}}$$

  Ce ratio mesure la part du temps de l'intervenant consacrée au projet. Il est à comparer avec la disponibilité escomptée de la ressource quand on a élaboré le planning à l'aide du diagramme de Gantt.

- la *vitesse* du mois n est :

$$\frac{A_n}{T_n}$$

  Ce ratio compare l'avancement et le temps passé. Il représente la vitesse d'avancement de l'intervenant. Inférieur à 1, il doit attirer l'attention.

La partie droite donne un récapitulatif depuis le début du projet. Elle comprend :

- le temps total passé, soit pour une tâche donnée si elle est à cheval sur deux mois, ou toutes tâches confondues. C'est la somme de tous les temps consommés ;
- le *coefficient d'utilisation* : c'est le ratio entre le temps passé par l'intervenant et le nombre de jours ouvrables, calculé à partir de la date de son arrivée sur le projet. Il est analogue à celui du mois écoulé, mais porte sur toute la durée de présence ;
- la *performance* : ce ratio mesure le degré d'atteinte des objectifs. Il n'a de sens que pour la totalité des tâches en cours ou achevées. On n'y inclut pas les tâches non encore ouvertes. Il compare la charge affectée avec la charge qui a été ou qui sera vraisemblablement consommée :

$$\frac{\text{Charge affectée} \times 100}{\text{Temps total passé + Reste à faire des tâches ouvertes}}$$

Mois 3 (20 J)								**Récapitulatif depuis le début du projet**		
Roger	Charge affectée	R(2)	T(3)	R(3)	A(3)	Coef. utilisation	Vitesse	Temps total	Coef. utilisation	Performance
Tâche A	14	0								
Tâche B	21	18	14	0	18		1,29	24		88 %
Tâche C	15	15	2	14	1		0,50			
Total	50		16		19	0,8	0,75	51	81 %	77 %

Figure 7.6 — Exemple de bilan individuel.

7.2.3 Le suivi du projet

Le chef de projet a besoin d'avoir périodiquement une vue de synthèse de l'état du projet. C'est sur cette synthèse que le maître d'œuvre, responsable contractuel du projet, fera le point avec le maître d'ouvrage.

C'est ce qu'on appelle le *tableau d'avancement du projet*. En général, la maille est plus large que pour le suivi individuel : on ne raisonne plus sur des tâches, mais sur des *lots* de tâches (figure 7.7). Un lot est un regroupement de tâches donnant lieu à des livrables qui forment un ensemble cohérent. Un lot correspond à une ou plusieurs fonctionnalités et peut faire l'objet d'une recette. Les engagements contractuels de livraison portent le plus souvent sur des lots. Un exemple de répartition en lots est donné au chapitre 10 (paragraphe 10.1.3).

	Mois n-1		**Mois n**				**Récapitulatif depuis le début du projet**			
Lots	T	R	T	R	A	Évolution charge restante	Charge initiale	Temps total passé	Évolution globale charge %	% Avancement

Figure 7.7 — Structure du tableau d'avancement du projet.

Ce tableau, alimenté par les récapitulatifs mensuels, comprend trois parties :

1. Rappel des éléments du mois n – 1 : pour toutes les tâches, même non commencées en fin de moins n – 1, on trouve le temps passé (T) et le reste à faire (R).
2. Les éléments du mois n : pour toutes les tâches, même non commencées, on a le temps passé (T), le reste à faire (R) et l'avancement (A).

   Cela permet de calculer la tendance du passé récent entre le mois n-1 et le mois n :

$$\textit{Évolution de la charge restante} = T(n) - A(n)$$
$$= T(n) - (R(n-1) - R(n))$$
$$= (T(n) + R(n)) - R(n-1)$$

   Ce paramètre indique si durant le mois la charge restante du projet augmente ou non. Si sa valeur est négative, la charge s'allège, si elle est positive la charge s'alourdit.
3. Les éléments récapitulatifs depuis le début du projet : la charge initiale est la somme de toutes les charges initiales, éventuellement actualisées ; le temps total passé représente le temps de travail affecté au projet depuis le démarrage.

   Cela permet de faire une comparaison globale entre la charge estimée et la charge consommée :

$$\textit{Évolution globale de la charge} = \textit{Temps total passé} + R(n) - \textit{Charge initiale}$$

   Cet indicateur compare la charge du projet avec l'avancement. S'il est positif, cela signifie qu'avec les éléments connus au jour du calcul, on prévoit déjà de dépasser la charge prévue.

On calcule enfin deux ratios, le *pourcentage d'évolution* qui donne le pourcentage de l'avance ou du dépassement par rapport à la charge initiale :

$$\frac{\textit{Évolution globale de la charge} \times 100}{\textit{Charge initiale}}$$

et le *pourcentage d'avancement* qui compare l'avancement à la fin du mois n avec la charge initiale :

$$\frac{\textit{Charge initiale} - R(n) \times 100}{\textit{Charge initiale}}$$

7.3 LE SUIVI ÉCONOMIQUE : LA MÉTHODE DE LA VALEUR ACQUISE

La méthode de la valeur acquise (*Earned Value Method)* permet de mesurer les performances du projet à une date t et d'effectuer des prévisions. Elle s'appuie sur trois indicateurs normalisés (calculés par la plupart des logiciels de suivi de projet) :

- *Valeur planifiée* (VP)[1] : c'est le coût du travail planifié, basé sur l'estimation des charges et le coût prévu des ressources. C'est qu'on a planifié de faire jusqu'à la date t. La VP jusqu'à la fin prévue du projet correspond au budget du projet et s'appelle le *Budget à l'achèvement* (BAA)[2].
- *Coût réel* (CR)[3] : ce sont les dépenses provenant du temps consommé. C'est le cumul de ce que l'on a effectivement dépensé jusqu'à la date t.
- *Valeur acquise* (VA)[4] : cet indicateur est basé sur l'avancement réel du projet et représente donc ce que l'on aurait dû dépenser pour le travail effectivement réalisé jusqu'à la date t.

Ces trois indicateurs peuvent être calculés pour une activité, un ensemble d'activités ou l'ensemble du projet. Ils sont cumulatifs, c'est-à-dire que c'est toujours la valeur cumulée depuis le début du projet jusqu'à la date t fixée.

On en déduit deux indicateurs complémentaires : l'écart de coût et l'écart de délai.

- *Écart de coût* (EC)[5] = VA – CR. S'il est négatif, on dépense plus que prévu ; s'il est positif, on sous-consomme, et s'il est nul, la productivité réelle correspond exactement à la productivité planifiée.
- *Écart de délai* (ED)[6] = VA – VP. S'il est négatif, on avance moins vite que ce que l'on avait planifié ; s'il est positif, on a pris de l'avance, et s'il est nul, le travail ce réalise conformément à ce qui avait été planifié.

La figure 7.8 donne une illustration. En abscisse, on a représenté l'axe du temps et en ordonnée les coûts cumulés. On suppose que l'on se situe au jour 10, date à laquelle on va calculer la performance.

Le budget total du projet est de 150 K€.

La VP au jour 10 est de 80 K€. C'est ce que l'on a prévu de faire et de dépenser, par exemple 8 jours de travail pour l'ensemble de l'équipe de projet à 10 K€ par jour.

1. En anglais : *Planned value* (PV). Anciennement appelé : CBTP (BCWS en anglais).
2. En anglais : *Budget at completion* (BAC).
3. En anglais : *Actual cost* (AC). Anciennement appelé : CRTE (ACWP en anglais).
4. En anglais : *Earned value* (EV). Anciennement appelé : CBTE (BCWP en anglais).
5. En anglais : *Cost Variance* (CV).
6. En anglais : *Schedule Variance* (SV).

Le CR est de 95 K€. On a donc dépensé plus que ce que l'on avait planifié. Mais peut-être a-t-on avancé davantage que prévu ? La VA va nous permettre de le savoir. Elle est de 70 K€, c'est-à-dire que, compte tenu de l'avancement réel, on aurait dû dépenser uniquement 70 K€.

L'écart de coût est de : 70 K€ – 95 K€ = – 25 K€.

L'écart de planning est de : 70 K€ – 80 K€ = – 10 K€.

Le projet, au jour 10, est donc en retard et en surconsommation.

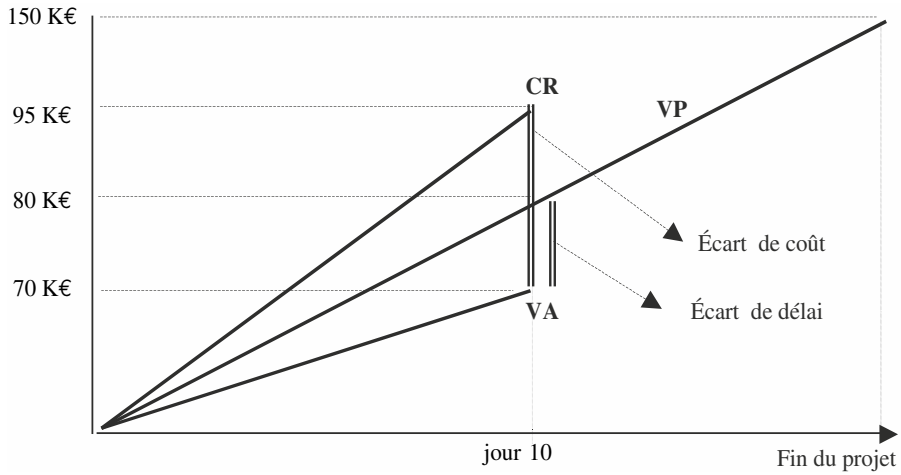


Figure 7.8 — Les indicateurs de la méthode de la valeur acquise

On peut traduire les écarts sous forme d'indicateurs de performance :

- *Indice de performance des coûts* (IPC)[1] = VA/CR
- *Indice de performance des délais* (IPD)[2] = VA/VP

On calcule parfois un indicateur qui lisse les variations de performance et qui est souvent considéré comme stable, c'est-à-dire variant au maximum de 10 %, lorsque l'on a dépassé 20 % d'achèvement du projet :

- *Indice de performance des coûts cumulé* (IPCC) =
  Somme VA (calculées à chaque fin de période/Somme CR (calculés à chaque fin de période).

On peut alors faire une prévision des coûts à l'achèvement du projet :

- *Prévision à l'achèvement* (PAA)[3] : c'est l'estimation du coût total du projet, compte tenu des performances jusqu'à ce jour et des risques prévisibles.

1. En anglais : *Cost performance index* (CPI).
2. En anglais : *Schedule performance index* (SPI).
3. En anglais : *Estimate at completion* (EAC).

Pour calculer la PAA, on distingue trois cas de figure.

- 1. On considère que les écarts sont accidentels et on n'en tient pas compte pour le futur. Alors : PAA = Coûts réels à ce jour + Travail restant = CR + (Budget à l'achèvement – Valeur acquise).
- 2. On considère que les écarts sont représentatifs du futur et on fait une projection. Alors : PAA = Coûts réels à ce jour + (Travail restant/indice de performance des coûts) = CR + (BAA – VA)/IPC.

  On peut aussi utiliser l'IPC^c.
- 3. On considère que les conditions ont changé, et l'on fait une nouvelle estimation. Alors : PAA = Coûts réels à ce jour + Coût estimé à l'achèvement[1] = CR + CEA.

On peut enfin calculer l'écart prévisible en fin de projet :

- *Écart à l'achèvement* (EAC)[2]

  EAC = Budget à l'achèvement – Prévision à l'achèvement = BAA – PAA.

Éventuellement, on calcule l'indice de performance nécessaire pour terminer le projet dans le budget :

- *Indice de performance requis* (IPR)[3] :

  IPR = Travail restant/Budget restant = (BAA – VA)/(BAA – CR).

L'utilisation de l'ensemble de ces indicateurs est proposée au chapitre 12 (paragraphe 12.15).

7.4 LES DÉCISIONS DE PILOTAGE

De nombreuses décisions sont prises en cours de projet, suite à l'analyse de l'avancement et des performances, et à divers imprévus. Le processus de décision ne s'appuie pas sur des techniques génériques. Nous allons toutefois présenter deux techniques générales qui peuvent être utilisées dans des contextes décisionnels : l'arbre de décision, accompagné de la technique de la valeur monétaire attendue, et le calcul de rentabilité selon le ROI.

1. En anglais : *Estimate to complete* (ETC).
2. En anglais : *Variance at completion* (VAC).
3. En anglais : *To complete performance index* (TCPI).

7.4.1 L'arbre de décision et la valeur monétaire attendue (VMA)

Un arbre de décision est une structure arborescente qui permet de représenter des alternatives à plusieurs niveaux. On a ainsi différents chemins, que l'on souhaite comparer. Pour cela, on va attacher des gains et pertes possibles pour chaque alternative. La technique de la Valeur monétaire attendue permet de calculer la valeur des différents scénarios selon les décisions prises. Nous avons le schéma de la figure 7.9 :

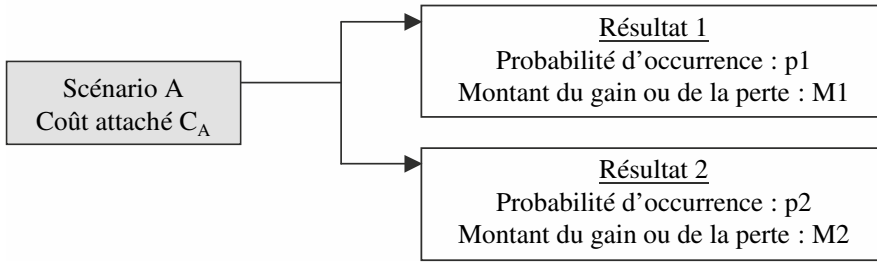


Figure 7.9 — Schéma de la méthode de la valeur acquise

La valeur du scénario est égale à : Σ (probabilité *gain/perte) – Coût attaché au scénario.

La décision, sous l'angle financier, va privilégier le scénario ayant la valeur la plus élevée.

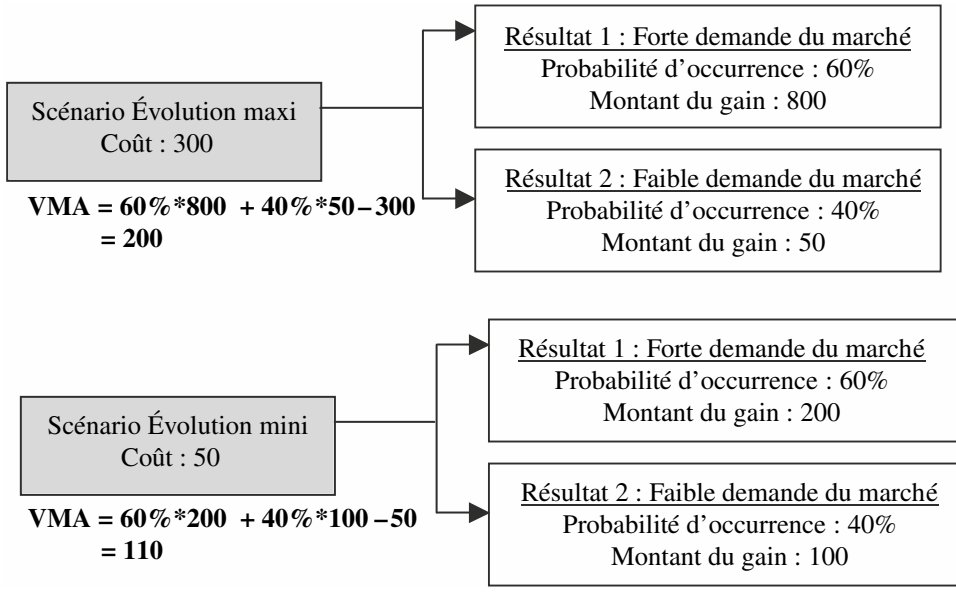


Figure 7.10 — Exemple d'application de la méthode de la valeur acquise

Soit par exemple un progiciel de paie que l'on doit faire évoluer pour prendre en compte une nouvelle réglementation. Il s'agit de choisir entre faire une évolution en profondeur avec ajout de nouvelles fonctionnalités, qui devrait permettre de gagner de nouveaux clients, et faire une évolution minimum. On a fait des estimations du coût de chacune des options, et on a fait des hypothèses sur la demande des entreprises (figure 7.10). Le calcul montre que le scénario « Évolution maxi » est à privilégier.

7.4.2 Le calcul de rentabilité : le ROI

Le calcul du ROI (*Return on Investment*) est une technique classique pour évaluer la rentabilité d'un investissement, compte tenu de son coût, de sa durée de vie et des gains attendus. Pour pouvoir l'utiliser, il faut au départ disposer de trois éléments d'information :

1. Une estimation de la durée de vie de l'investissement.
2. Des prévisions de dépenses, période par période. La période est souvent annuelle, et les dépenses sont aussi bien l'investissement que les coûts de fonctionnement à venir.
3. Des prévisions de recettes, qui peuvent être des diminutions de dépenses, pendant toute la durée de vie de l'investissement, période par période.

On commence par établir un tableau des flux de trésorerie, période par période, que l'on appelle *chronique du cash-flow*. Pour chaque année, le cash-flow = Recettes – Dépenses. Un exemple est donné dans le tableau de la figure 7.11.

Année	2007	2008	2009	2010	2011	2012
Dépenses	50	60	70	10	10	20
Recettes	0	0	0	60	100	200
Cash-flow (recettes-dépenses)	**– 50**	**– 60**	**– 70**	**50**	**90**	**180**

Figure 7.11 — Exemple de chronique de cash-flow

On souhaite résumer cette chronique par un seul chiffre pour pouvoir comparer des projets d'investissement. On va le faire par trois variables : le délai de remboursement, la valeur actuelle nette et le taux de rendement interne.

Le *délai de remboursement* est délai au bout duquel les recettes cumulées dépassent les dépenses cumulées. Sur notre exemple, le délai est de 5 ans (figure 7.12).

Année	2007	2008	2009	2010	2011	2012
Dépenses	50	60	70	10	10	20
Recettes	0	0	0	60	100	200
Cash-flow	– 50	– 60	– 70	50	90	180
Cash-flow cumulé	**– 50**	**– 110**	**– 180**	**– 130**	**– 40**	**140**

Figure 7.12 — Exemple de calcul de délai de remboursement

Pour calculer la *valeur actuelle nette* (VAN), il faut faire une hypothèse sur le taux de placement de l'argent, que l'on appelle taux d'actualisation (t_{act}). On actualise la chronique de cash-flow en ramenant chaque somme future M (acquise ou dépensée période n) à son équivalent actuellement. La somme actualisée M_{act} est celle que l'on devrait avoir pour obtenir M à la période n en la plaçant au taux t_{act}.

$$M_{act} = M/(1+ t_{act})^{n}.$$

La valeur actuelle nette de l'investissement est la somme des cash-flows actualisés jusqu'à la fin de la durée de vie de l'investissement. Dans notre exemple, avec un taux d'actualisation de 6 %, la VAN est de 74 (figure 7.13)

Année	2007	2008	2009	2010	2011	2012
Dépenses	50	60	70	10	10	20
Recettes	0	0	0	60	100	200
Cash-flow	– 50	– 60	– 70	50	90	180
Cash-flow actualisé (6 %)	– 47	– 53	– 59	40	67	127
Cash-flow actualisé cumulé	**– 47**	**– 101**	**– 159**	**– 120**	**– 52**	**74**

Figure 7.13 — Exemple de calcul de valeur actuelle nette

Le *taux de rendement interne* (TRI) est le taux d'actualisation qui rend la VAN nulle à la fin de la durée de vie de l'investissement. Cette variable est équivalente à la VAN, mais peut être plus parlante. Sur notre exemple, le TRI est de 19 % (figure 7.14).

Des exercices d'application de la VMA et du ROI sont proposés au chapitre 12.

Année	2007	2008	2009	2010	2011	2012
Dépenses	50	60	70	10	10	20
Recettes	0	0	0	60	100	200
Cash-flow	– 50	– 60	– 70	50	90	180
Cash-flow actualisé (19 %)	– 42	– 42	– 42	25	38	63
Cash-flow actualisé cumulé	**– 42**	**– 84**	**– 126**	**– 101**	**– 63**	**0**

Figure 7.14 — Exemple de calcul du taux de rendement interne

7.5 LE PILOTAGE D'UN PROJET SOUS-TRAITÉ

7.5.1 La sous-traitance d'un projet

On peut sous-traiter tout ou partie d'un projet (étude, réalisation, formation…) à une entreprise tierce, ce qui donne lieu à un contrat. Il y a trois grandes formes de sous-traitance.

1. La sous-traitance au forfait consiste à faire réaliser des travaux pour un montant forfaitaire. Le client attend du fournisseur un engagement de résultat.
2. La sous-traitance en régie consiste à acheter pour une durée définie une mise à disposition de personnel appartenant à une entreprise tiers. Il s'agit de la part du fournisseur d'un engagement de moyens.
3. La sous-traitance d'expertise consiste à acheter une compétence dans un domaine pour traiter un problème précis (estimation de charge, planification). Le fournisseur intervient ponctuellement ; son intervention se termine par des recommandations.

Dans les deux derniers cas, le chef de projet bénéficie de ressources supplémentaires. Son rôle de pilote n'en est pas affecté. En revanche, dans le premier cas, une partie de la responsabilité de conduite de projet est transférée au sous-traitant.

Le choix d'un sous-traitant prend généralement en compte des critères divers, auxquels on peut accorder des poids différents. Ils peuvent être d'ordre technique (approche technique proposée, compétences techniques…), commercial (prix, clause de propriété…), institutionnel (taille et type d'entreprise, capacité financière…) ou managérial (procédures, méthodes, système qualité…). On les

présente parfois dans une *grille multicritère* assortie d'un système de pondération. Un exemple de notation de trois fournisseurs par une telle grille est donné à la figure 7.15.

Critère	Poids	Fournisseur A		Fournisseur B		Fournisseur C	
Technique	3	3	**9**	7	**21**	5	**15**
Commercial	2	4	**8**	5	**32**	5	**10**
Méthode	1	1	**1**	3	**3**	3	**3**
Gestion projet	3	1	**3**	3	**9**	1	**3**
Note			**21**		**65**		**31**

Figure 7.15 — Exemple d'utilisation d'une grille multicritère

7.5.2 Le rôle du chef de projet en cas de sous-traitance

La première responsabilité du chef de projet est d'établir un cahier des charges indiquant les résultats attendus, accompagnés d'éventuelles contraintes (format de remise des livrables, sécurité, volumétrie, portabilité…) et exigences (documents de suivi de projet…). Le cahier des charges fait parfois l'objet d'un dialogue et d'un accord entre les deux parties.

La rédaction du cahier des charges suppose que le chef de projet a effectué un découpage structurel, voire un découpage temporel si une partie seulement du projet est sous-traitée.

Il est fortement conseillé d'effectuer une analyse de risques, car l'externalisation des risques ne les diminue pas — sauf éventuellement sur le facteur Instabilité de l'équipe de projet — et les chances de succès peuvent être compromises, sans que l'on ne puisse plus rien faire.

Il serait dangereux de ne faire aucune évaluation des charges, comme cela a été évoqué au paragraphe 3.2. Il est en effet difficile de sélectionner un fournisseur, parmi des offres qui comportent des prix très éloignés. Il importe donc de faire une estimation globale, soit par la méthode Delphi, soit par la méthode des points fonctionnels, soit par analogie avec un projet comparable.

La planification du projet relève de la responsabilité du sous-traitant. Cependant, le chef de projet qui sous-traite doit avoir une visibilité sur le jalonnement du projet : pour cela, il est conseillé de privilégier la remise de livrables intermédiaires, ce qui favorise le suivi des étapes du projet.

Le pilotage d'un projet sous-traité s'exerce à deux niveaux. Un premier niveau est de la responsabilité exclusive du sous-traitant. Un second niveau est partagé par le chef de projet et le sous-traitant. Lors de la passation du contrat, un format de tableau de bord doit être défini, par exemple analogue à celui de la figure 7.7. Des réunions de suivi régulières et fréquentes doivent être tenues et faire l'objet d'un engagement contractuel entre les deux parties. Il est préférable d'avoir une réunion d'une heure chaque semaine, plutôt que de trois heures tous les mois. Cette proximité permet d'anticiper, ou du moins de découvrir suffisamment tôt certaines difficultés pour pouvoir éventuellement prendre des décisions d'orientation. De plus, le chef de projet doit pouvoir à son tour informer son maître d'ouvrage sur l'avancement du projet, donc en avoir une connaissance précise et à jour.

Ainsi, la charge la plus importante pour le chef de projet qui sous-traite, concerne le pilotage conjoint avec le sous-traitant.

7.6 LE MANAGEMENT DES CONNAISSANCES POUR LES PROJETS

7.6.1 La capitalisation d'expérience sur les projets

Nous allons clore ce chapitre sur le pilotage en situant un projet particulier dans le contexte plus large de l'entreprise. On peut considérer que la responsabilité d'un chef de projet s'étend à la capitalisation de la nouvelle expérience qu'il vient d'acquérir. En effet, la mémorisation des éléments de synthèse d'un projet permet l'apprentissage collectif. Nous allons en donner quelques éléments.

La mémoire sur un projet comporte d'abord un *bilan de projet*, avec les principales caractéristiques du projet :

- nom du projet ;
- date de début, date de fin ;
- nombre total d'intervenants ;
- domaine d'application.

Le bilan des ressources utilisées peut être présenté comme dans le tableau de la figure 7.16 qui s'appuie sur la typologie utilisée pour l'évaluation des charges.

Il s'agit ensuite de suivre les bases de valorisation standard. Par exemple, pour le suivi de ratios, on élabore le tableau de la figure 7.17 et pour le suivi des unités d'œuvre celui de la figure 7.18.

Type de tâche	Nombre	Charge initiale	Charge constatée	Nombre jours écart	% écart par rapport à la charge initiale
Type 1					
Type 2					
Total					

Figure 7.16 — Tableau de bilan de projet.

Type de tâche	Charge initiale	Charge constatée	Ratio appliqué	Ratio constaté
Programmation	85	100	–	–
Jeu d'essai	17	18	20 %	18 %

Figure 7.17 — Tableau de suivi des ratios.

Type de tâche	Poids de l'unité d'œuvre	Nombre d'unités d'œuvre	Charge moyenne constatée	Écart	Écart-type (avec c_i = charges réelles)
Type	p	N	m	p – m	$1/n\ [\text{Somme}_{i=1 \text{ à } n}(c_i - m)^2]^{1/2}$

Figure 7.18 — Tableau de suivi des unités d'œuvre.

7.6.2 Le knowledge management appliqué aux projets système d'information

La capitalisation de l'expérience acquise sur les projets s'inscrit aujourd'hui dans une perspective plus vaste qu'il y a quelques années, celles du « knowledge management », c'est-à-dire du management des connaissances. On en attend un meilleur partage entre les chefs de projets des bonnes pratiques et des erreurs à éviter, pour une maîtrise accrue des projets. Celle-ci se traduit par des améliorations tangibles — on peut en voir les effets — ou non, et mesurables — on peut définir une métrique — ou non. On peut en donner quelques exemples.

- L'augmentation du respect des délais dans les projets est un gain tangible et mesurable.
- L'amélioration de la qualité des dossiers est un gain tangible, mais difficilement mesurable.
- La satisfaction des utilisateurs, évaluée par des enquêtes de satisfaction, est mesurable, mais pas toujours tangible.
- La qualité de la communication entre les chefs de projet n'est ni mesurable, ni tangible à court terme.

Le knowledge management participe d'une nouvelle approche des organisations orientée vers le mode de travail en réseau et la mise en commun des informations. Pour cela, il s'agit de construire une **mémoire** organisationnelle qui pourra être utilisée par tous ceux qui en ont besoin. On peut distinguer plusieurs catégories de mémoire :

- la mémoire individuelle : les connaissances, expériences et savoir-faire des chefs de projet ou des autres participants aux projets ;
- la mémoire du projet : le dossier de projet contenant les caractéristiques du projet, son déroulement, son bilan ;
- la mémoire de l'entreprise : les normes, méthodes et paramètres retenus dans l'entreprise ;
- la mémoire de la profession : l'état de l'art formalisé, les méthodes diffusées…

Différents **obstacles** freinent la capitalisation. Souvent, il n'y a pas responsable de la capitalisation ; or, l'analyse de ce qu'il faut retirer du projet est parfois difficile. En fin de projet, le chef de projet est rarement disponible pour des tâches additionnelles ; de plus, il n'a pas envie d'exposer des écarts négatifs de charge ou délai. S'il n'y a pas de structure de stockage, l'expérience reste exprimée dans des documents hétérogènes C'est pourquoi la capitalisation nécessite un **dispositif** qui comprend un contenu, une organisation, des procédures et des technologies.

Contenu

Il s'agit de définir ce sur quoi on veut capitaliser. Le contenu couvre les différentes facettes de la gestion de projet et comprend notamment : des découpages structurels ; des plans types ; des organisations types ; des fiches contenant les bonnes pratiques ; des fiches techniques…

Ce contenu doit être structuré en thèmes et en éléments de connaissance. Par exemple, la méthode REX (Retour d'Expérience) utilisée dans le domaine industriel et commercialisée avec un outil par la société Euriware, repose sur l'architecture (simplifiée) suivante :

- un référentiel terminologique identifiant les éléments-types sur lesquels on souhaite capitaliser (équipement, défaillance, tâche…) ;
- des éléments de connaissance associés à un ou plusieurs termes du référentiel terminologique ;
- une structure de description de chaque élément de connaissance, pouvant être de différentes natures (documentation, formalisation d'une expérience, savoir-faire acquis lors de la réalisation d'une tâche…).

De façon générale, pour que les résultats d'expérience puissent être partagés, il est conseillé de respecter les règles suivantes :

- la connaissance doit être découpée en éléments manipulables ;
- les éléments doivent être référencés de façon claire ;
- les éléments doivent être ciblés sur un problème précis (analyser les risques, évaluer une charge, rédiger un dossier d'analyse…) ;
- les éléments doivent être organisés avec différents niveaux de détail.

Organisation

Il s'agit de préciser les rôles et les responsabilités. Certains auteurs ont étudié les facteurs qui influent sur la performance d'une entreprise. Il en ressort que la performance individuelle n'intervient qu'en sixième position, alors que le support que peut apporter l'organisation aux individus vient en second rang (le premier étant la définition claire des produits ou services offerts). Cela signifie que l'organisation mise en place en vue de capitaliser est essentielle. On distingue souvent les rôles suivants :

L'*expert* est porteur de connaissances. C'est un chef de projet, plus ou moins expérimenté.

Le *knowledge manager*, gestionnaire de connaissances, responsable du processus de création et d'évolution de la connaissance partagée. Dans le contexte informatique, ce peut être le responsable qualité à qui on confie la définition des procédures de capitalisation.

Le *responsable de thème* est responsable du contenu de son domaine, par exemple dans le cas d'un intranet il est en charge de la connaissance nouvelle qui doit apparaître dans l'espace collectif. Il analyse les bilans de projets, est à l'affût de nouvelles connaissances de la profession, fait évoluer certains paramètres (poids des unités d'œuvre…). Son rôle comporte une dimension relationnelle, toute nouvelle connaissance devant être expliquée aux experts et validée avec eux. Il est également responsable de la rédaction des pages et de la validité des informations. Si la connaissance est de taille réduite, le responsable de thème est responsable de l'ensemble du contenu de tous les thèmes.

Cette organisation repose sur la distinction de quatre formes de connaissances (figure 7.19), selon deux critères : elle est individuelle ou collective ; elle est explicite — le détenteur en est conscient — ou tacite — le détenteur utilise la connaissance sans en être conscient, de façon implicite ou automatique[1].

1. Nonaka I., *A dynamic theory of organizational knowledge creation*, Organization Science, 5, 1, fév. 1994, pp. 14-37.

Le passage du collectif à l'individuel correspond à un processus d'appropriation ou d'apprentissage.

Le passage de l'individuel au collectif correspond à un enrichissement des connaissances et/ou des pratiques collectives.

Le passage du tacite à l'explicite correspond à de l'explicitation et/ou de la formalisation.

Le passage de l'explicite au tacite correspond à l'assimilation.

La capitalisation d'expérience sur les projets système d'information signifie d'une part un passage des connaissances individuelles explicites vers des connaissances collectives, d'autre part une explicitation de certaines connaissances tacites, en général individuelles. Pour passer au stade du knowledge management, il faut ensuite organiser les processus d'apprentissage et d'appropriation.

	Individuelle	**Collective**
Explicite	Connaissance dont le détenteur est conscient Par exemple : ma façon de calculer les charges pour un projet groupware dans un environnement Notes/Domino 5.	Connaissances accessibles à tous Par exemple : l'état de l'art de l'estimation des charges.
Tacite	Connaissance pratique que le détenteur utilise sans qu'elle fasse l'objet d'une « règle » Par exemple : pour certains projets que je ne « sens » pas bien, je rajoute automatiquement 10 % de charge.	Connaissances partagées sans que les personnes en soient conscientes Par exemple : un projet se mène de façon « top-down ».

Figure 7.19 — Formes de connaissance.

Procédures

Il s'agit de définir comment capitaliser et comment réutiliser la connaissance. La logique du management des connaissances doit prendre en compte à la fois le producteur et le consommateur d'informations. Ceci est d'autant plus vrai dans le cas des projets informatiques, car ces deux populations se recouvrent largement. Les procédures doivent donc :

- proposer au producteur des procédures simples de capitalisation de ses projets ;
- définir des procédures claires d'évolution et d'enrichissement des connaissances collectives ;

- fournir au consommateur des informations pertinentes, c'est-à-dire ciblées sur les problèmes rencontrés ;
- donner au consommateur des moyens d'accéder facilement et rapidement à l'information nécessaire.

Les procédures doivent prendre en compte le fait que les différents modes de communication fournissent une information de qualité différente, comme indiqué dans le tableau de la figure 7.20.

Mode de communication	Qualité de l'information
Échange en face à face	Très riche
Échange à distance (téléphone, mail)	Riche
Réunion	Riche si la réunion est bien conduite (ordre du jour et choix des participants)
Écrit (document, dossier…)	Moyenne

Figure 7.20 — Modes de communication et qualité de l'information.

Technologies

Tout dispositif de management des connaissances doit s'appuyer sur des outils qui allègent les contraintes liées aux processus d'alimentation, qui facilitent l'exploration et qui permettent un stockage d'informations devant continuellement être enrichies et actualisées. Trois types d'outils complémentaires peuvent être utilisés :

- des outils d'infrastructure et de diffusion (messagerie, intranet…) ;
- des outils de mémorisation pour structurer et stocker la connaissance (AGL, bases de données, gestion électronique de document…) ;
- des outils d'exploitation de la connaissance (moteur de recherche, outils d'analyse…).

7.7 LE PILOTAGE D'UN PROJET EN MODE AGILE

7.7.1 Le suivi d'un projet agile

La formalisation du suivi n'est pas un aspect préconisé par les méthodes agiles. Il ne faudrait pas en conclure que le suivi est négligé, bien au contraire.

Le suivi individuel est assuré de façon étroite, notamment lors des itérations de développement, par le principe des brèves réunions quotidiennes, préconisées notamment par XP et SCRUM, qui permettent de faire un point sur la progression. Certains rôles favorisent une proximité avec l'équipe, comme le *tracker* dans XP, *le scrum master* ou le chef d'équipe dans DSDM.

Selon le Manifeste Agile, l'indicateur d'avancement le plus pertinent est la mise à disposition de logiciel qui marche. Cette position se justifie dans la mesure où la structure de découpage du projet a permis d'identifier, à une maille fine, des composants pouvant être testés de façon indépendante. Certains tableaux simples, comme le *burndown chart* de SCRUM qui visualise la diminution du reste à faire, assurent une certaine formalisation de l'avancement.

En ce qui concerne le suivi du projet vis-à-vis du client, on peut relever que les engagements vis-à-vis des jalons que représentent les fins de cycle de livraison ont un caractère impératif, comme énoncé par la *timebox* (cf. § 4.8). En revanche, le calendrier des itérations de développement est réajusté en fonction du déroulement réel, ce qui limite une communication formelle à ce niveau-là. Le responsable du projet a donc un rôle accru vis-à-vis de la maîtrise d'ouvrage, puisqu'il ne peut pas compter sur la médiation d'un tableau de bord bien formalisé.

7.7.2 Les variables d'action pour un projet agile

Les aléas ne sont pas considérés comme des événements négatifs, mais comme partie intégrante du déroulement des projets. En particulier, les changements de spécifications et les difficultés rencontrées par les développeurs sont pris en compte dans le pilotage du projet.

On peut ainsi mentionner les variables d'action suivantes :

- La réduction du périmètre fonctionnel, si le champ déterminé au début d'une itération rend impossible le respect de la *timebox*.
- Le renvoi dans une itération de livraison ou une livraison ultérieure.
- La tenue d'un « conclave », défini comme une session extraordinaire qui conduit à des décisions importantes n'ayant pu être prises lors des sessions normales. Ces décisions portent en général sur des options de conception qui bloquent l'avancement du projet car elles demandent un consensus. La session dure jusqu'à ce que la décision soit prise.
- Le développement non planifié d'un petit prototype pour éclairer une incertitude technique et permettre une estimation des charges plus précise, appelé *spike* par XP.

7.7.3 La gestion des connaissances dans un projet agile

L'origine des méthodes ou techniques agiles est souvent empirique, expérience réussie ou malheureuse, dont on a tiré les leçons. Même si toutes les méthodes ne mettent pas l'accent sur le bilan des projets et la capitalisation d'expérience, la gestion des connaissances occupe une place non négligeable dans les approches agiles.

Il y a en particulier trois dispositions d'apprentissage collectif qui doivent être mentionnées.

D'abord, le compagnonnage qui favorise une transmission de savoir-faire entre développeurs. La mise en binôme, qui oblige à expliciter ses propres choix, ainsi que l'accompagnement des juniors par des développeurs plus expérimentés, augmentent la circulation des connaissances.

Ensuite, la mobilisation ponctuelle de l'équipe, notamment en cas de problème important lors d'une itération de développement, peut profiter à tous.

Enfin, un des principes du Manifeste Agile est une incitation à la réflexivité pour l'équipe, c'est-à-dire une réflexion collective périodique pour améliorer le fonctionnement du groupe.

Il convient cependant de souligner que généralement une telle dynamique d'apprentissage reste circonscrite au groupe de projet, donc temporaire, et qu'on ne peut compter que sur une transmission individuelle d'un projet à l'autre.

Ainsi s'achève la présentation des différents aspects du pilotage. La mise en pratique est décrite au chapitre 12 et illustrée au chapitre 15 avec l'outil MS Project.

8

La maîtrise de la qualité

8.1 LA PROBLÉMATIQUE DE LA QUALITÉ

Sous des formes différentes, liées au contexte de production, la qualité est une notion très ancienne[1]. Elle comporte deux aspects complémentaires, que nous percevons clairement dans l'organisation des corporations au moyen âge : un volet collectif et un volet individuel. Chaque corps de métier édicte des règles, dont la transmission est assurée par le système de formation des compagnons ; celle-ci s'achève par la réalisation d'un chef-d'œuvre, qui, pour personnel qu'il soit, doit respecter les normes de la profession. La satisfaction individuelle de la belle ouvrage, du travail bien fait, s'intègre dans l'objectif de défense du métier.

Cette organisation de la qualité a disparu avec la production industrielle. L'ouvrier n'est plus en contact avec le client ; il n'agit que sur une parcelle du produit ; les règles jadis transmises par les organisations professionnelles se sont perdues. Les enjeux économiques de la qualité sont cependant perçus assez vite[2]. Les premières normes nationales de fabrication apparaissent dans le domaine militaire : sous la Révolution, le gouvernement fait fabriquer du matériel de mesure devant être utilisé dans toutes les fabriques de munitions. Cela conduit à imposer des dimensions standard permettant l'interchangeabilité des fusils et des munitions. Au XX^{e} siècle, la préoccupation de la qualité va connaître un

1. « Si un maçon a construit une maison et que celle-ci n'est pas suffisamment solide et que la maison s'écrase et tue ses occupants, le maçon devra être tué » trouve-t-on dans le code d'Hammourabi, roi de Babylone ayant vécu plus de 17 siècles avant notre ère ! Les Phéniciens coupaient la main des artisans ayant violé de façon répétitive leurs standards de qualité...
2. « Si nos usines, par un travail soigné, assurent la qualité de nos produits, il sera de l'intérêt des étrangers de s'approvisionner chez nous et l'argent affluera dans le royaume » écrit Colbert en 1664 dans un rapport à Louis XIV.

développement sans précédent. On peut distinguer quatre étapes que nous avons qualifiées d'après la théorie apparue durant cette période : Inspection, Contrôles statistiques, Qualité totale, Assurance Qualité.

Première étape (1900-1920) : Inspection. Les inventions techniques ont permis un foisonnement d'applications industrielles. La productivité passe au premier rang des préoccupations. Aux États-Unis, les grandes entreprises telles la compagnie Ford mettent en œuvre à une large échelle les principes d'organisation scientifique du travail (OST) édictés par F.W. Taylor. L'objectif est de faire fabriquer des produits de plus en plus complexes par des ouvriers non qualifiés. L'autocontrôle est remplacé par une inspection systématique permettant d'isoler les produits défectueux. Celle-ci est souvent une détection visuelle des défauts apparents : elle s'effectue en bout de chaîne sur les produits finis. Le dispositif de production distingue nettement concepteurs, ouvriers et inspecteurs.

Deuxième étape (1920-1950) : Contrôles statistiques. En 1920, la Western Electric doit renoncer à mettre en œuvre un nouveau central téléphonique, à cause du nombre de défauts qu'il contient. Et pourtant, durant toute la production de ce central, les inspecteurs étaient aussi nombreux que les ouvriers ! La Compagnie Bell Telephone crée alors un département Qualité, avec notamment G. Edwards et W. Shewhart. Le premier va développer l'idée que la qualité doit être gérée : il crée la notion d'assurance qualité. Le second introduit la statistique comme moyen de maîtrise de la qualité. Son objectif est de déterminer le nombre minimal d'unités à contrôler sur un lot de produits finis avant de pouvoir raisonnablement déclarer le lot bon ou mauvais. Ces contrôles statistiques se répandent, particulièrement dans l'industrie d'armement pendant la deuxième guerre mondiale, sous le nom de contrôle statistique de la qualité[1]. Ces méthodes servirent à établir les *military standards*, normes militaires s'imposant à tous les fournisseurs. Les lots ayant un pourcentage d'unités défectueuses plus élevé qu'une limite contractuelle sont rejetés ; les autres sont acceptés. Le taux dépend de l'historique des livraisons de chaque fournisseur. À partir des années cinquante, ces méthodes vont se répandre dans les industries européennes. En France, elles seront particulièrement soutenues par l'AFCIQ, Association française pour le contrôle industriel et la qualité.

Troisième étape (1950-1960) : Qualité totale. En 1950, A.V. Feigenbaum publie un ouvrage relatant ses expériences de développement de la *qualité totale* à la General Electric[2]. Il sera nommé quelques années plus tard directeur de toutes les unités de production de la General Electric (électronique, automobile, chimie, navigation aérienne, etc.). Le terme « total » s'oppose aux contrôles

1. SQC : *Statistical Quality Control*.
2. TQC : *Total Quality Control*.

partiels de la qualité, où celle-ci est conçue comme une activité de contrôle technique spécialisée. Les dix principes de la qualité totale sont les suivants :

1. La qualité ne relève pas du seul critère technique et doit faire l'objet d'une application systématique dans toute l'entreprise.
2. La qualité ne pourra être l'affaire de tous que si les dispositions qualité sont conçues de manière à soutenir simultanément le travail individuel et celui des équipes.
3. L'amélioration de la qualité concerne non seulement la fabrication, mais également la vente, le marketing, la conception des produits et les services fonctionnels.
4. Le processus d'application de la qualité doit se faire en pensant constamment à l'acheteur et non en fonction des impératifs de vente ou d'efficacité de la production.
5. La qualité et le coût ne doivent pas être perçus comme concurrents : le meilleur moyen de fabriquer plus vite et moins cher consiste à améliorer la qualité des produits.
6. L'amélioration de la qualité n'est pas une opération ponctuelle, comme par exemple éliminer les vieilles méthodes de contrôle de la qualité. Elle se conçoit, s'évalue et se gère continûment.
7. L'amélioration de la qualité ne repose pas sur le seul travail d'un spécialiste ; elle ne peut être atteinte qu'avec la participation active de tout le personnel.
8. L'amélioration de la qualité augmente la productivité car elle élimine des dysfonctionnements existants.
9. La qualité doit faire l'objet d'une gestion aussi directe et efficace que la production ou la finance.
10. Les principes ci-dessus découlent de la mise en place d'une politique claire de gestion de la qualité orientée vers la clientèle.

La qualité totale, qui élargit et complète le contrôle statistique, reçoit un accueil très favorable au Japon. L'industrie japonaise d'après-guerre, ruinée, dirigée par de nouvelles équipes issues de la production ou du commercial, fournissait des produits de qualité médiocre au regard des standards américains. Elle ne pouvait envisager la conquête des marchés occidentaux qu'au prix d'une forte amélioration de la qualité. Avec deux statisticiens disciples de Shewhart, W.E. Deming et J.M. Juran, A.V. Feigenbaum se rend fréquemment au Japon pour former les cadres à la maîtrise de la qualité. Une association d'ingénieurs[1]

1. La JUSE : *Japanese Union of Scientists and Engineers*.

créée après guerre diffuse les méthodes américaines. Cependant, les Japonais comprennent vite l'impasse d'ordre méthodologique dans laquelle on s'enferme si l'on assimile qualité et science exacte. En effet, on va alors multiplier les procédures de vérification. Les pesanteurs administratives imposées par les inspecteurs entraînent à la fois pertes de temps et rejet par les acteurs. Ce constat conduit à la création en 1962 sous l'impulsion du chercheur K. Ishikawa, alors à la tête de la JUSE, du premier cercle de qualité. Vingt ans plus tard, il y en aura 140 000.

Le *cercle de qualité* est défini comme « un petit groupe permanent et homogène, composé de cinq à dix volontaires appartenant à une même unité organique (atelier, bureau, service, laboratoire, réseau de vente) ou ayant des préoccupations professionnelles communes. Animé par le plus proche responsable hiérarchique direct et agissant en liaison avec un spécialiste facilitant le dialogue, le cercle se réunit régulièrement afin d'identifier, analyser et résoudre les problèmes de son choix concernant la qualité, la sécurité, la productivité, les conditions de travail, etc. »[1]. Les cercles de qualité mettent près de vingt ans à pénétrer en France : il y en a eu jusqu'à 40 000, avec des degrés d'activité divers. Ils ont permis de faire sortir de façon pratique la préoccupation de la qualité des seuls départements Qualité. Ils ont favorisé le déplacement de l'attention du produit vers le processus de production. L'AFCIQ et l'AFCERQ fusionnent au début des années 90 pour former le MFQ, Mouvement français pour la qualité.

Quatrième étape (1960-1990) : Assurance Qualité. La diffusion des principes qualité à tous les acteurs va franchir un nouveau pas. En 1961, P.B. Crosby, en collaboration avec G. Borel de LMT, lance le concept de zéro défaut dans la société Martin Marietta : les premiers échecs dans le domaine spatial ont montré que la majorité des défauts sont le résultat d'erreurs humaines ; c'est donc sur l'homme qu'il faut consacrer ses efforts. Crosby a ensuite, en tant que directeur qualité d'ITT, élaboré un programme zéro défaut. Celui-ci comprenait notamment la suppression de nombreux contrôles, afin de responsabiliser l'opérateur pour qu'il fasse bien dès la première fois. Le zéro défaut représente la conformité complète aux spécifications.

On passe ainsi progressivement d'une forme inactive de la qualité (détecter le défaut a posteriori) à une forme active de prévention méthodique et systématique des sources de non-qualité que l'on appelle assurance qualité. En 1988, se crée l'AFAQ, Association française d'assurance qualité. Cette évolution dans la conception de la qualité s'accompagne d'une évolution des normes : des normes de spécification (caractéristiques techniques du produit), on est passé d'abord à des normes modélisant l'attente de l'utilisateur, ce qui permet d'éviter que la norme subisse la même obsolescence que la technologie (modèle OSI), puis à des normes portant sur la gestion de la qualité dans l'entreprise (ISO9000). On

1. Brochure de présentation de l'AFCERQ, Association française pour les cercles de qualité.

distingue ainsi plusieurs niveaux de norme. Ce point est développé plus loin (paragraphe 8.3).

Aujourd'hui, la qualité totale, préventive et couvrant l'ensemble des activités de l'entreprise, est pratiquée par plus de 80 % des entreprises nippones. Elle rencontre un intérêt croissant dans les grandes entreprises européennes et américaines. Quelques exemples : dans le secteur automobile, Renault, après un redressement spectaculaire, fait maintenant figure de référence ; un Institut Renault de la Qualité a été mis en place pour les cadres. Rank Xerox, en dix ans, a divisé par treize le nombre de pièces défectueuses et amélioré de 40 % la performance de ses équipements grâce à son programme qualité totale.

On peut dire en conclusion que les concepts qualité apparus depuis près d'un demi-siècle prennent forme lentement dans les entreprises ; ils correspondent à un changement profond dans les relations internes de l'entreprise comme dans ses relations avec ses partenaires (clients, fournisseurs, prescripteurs…). Ainsi le professeur Ishikawa expose le *Company Wide Quality Control*[1] (CWQC) à travers quatre principes opposés à des pratiques courantes :

1. La qualité d'abord (*quality first*) au lieu du profit tout de suite (*profit first*).
2. Le marché doit rentrer dans l'entreprise (*market in*) et non l'entreprise écouler sur le marché des produits conçus sans référence au marché (*product out*).
3. Tout processus est le fournisseur d'un autre processus (*next process is a consumer*) par opposition au produit conçu par un seul processus : c'est la mise en place de relations de type client-fournisseur à l'intérieur de l'entreprise.
4. La maîtrise des faits (*fact control*) : toute production doit pouvoir être contrôlée par l'identification de grandeurs mesurables.

8.2 LE VOCABULAIRE DE LA QUALITÉ

La norme AFNOR[2] X50-120, qui définit les principaux termes relatifs à la qualité, propose la définition suivante. La qualité est « l'ensemble des propriétés et caractéristiques d'un produit ou service qui lui confèrent l'aptitude à satisfaire des besoins exprimés ou implicites ». Ainsi la qualité ne se mesure pas dans l'absolu, mais par rapport à un besoin. Que l'on soit ou non dans un contexte

1. Maîtrise de la qualité dans l'entreprise tout entière.
2. Association Française de Normalisation, Tour Europe, Paris La Défense.

contractuel, les clients et les besoins doivent être identifiés. Cette définition est le résultat d'une réflexion qui s'est faite en trois étapes.

D'abord, on définit des normes et on vérifie la conformité. Puis, on découvre que le nombre de rebuts est élevé, on améliore alors le processus de fabrication. Enfin, on s'aperçoit que les clients ne sont pas satisfaits par des produits techniquement bons et que les attentes doivent être prises en compte. Ni le client, ni le fournisseur ne peuvent attendre la livraison pour découvrir que le produit n'est pas satisfaisant. L'un et l'autre ont intérêt à préciser les *facteurs* qui serviront de base à l'appréciation de la qualité. Cela conduit à une négociation devant aboutir à un compromis. En effet, quand le client cherche à préciser les critères de jugement, il est conduit à identifier des fonctionnalités qui s'ajoutent à l'expression brute du besoin (exemple : aide en ligne) ainsi que des procédés de fabrication (exemple : utilisation d'un atelier de génie logiciel pour assurer la portabilité). Ces fonctionnalités et ces procédés induisent souvent une charge supplémentaire (coûts, délais). Il faut donc trouver un compromis entre le niveau des exigences et le prix consenti.

L'approche négociée de la qualité conduit à identifier deux types de rôle :

- le maître d'ouvrage (le client) — en anglais *project owner* — définit les facteurs qualités qui caractérisent le produit attendu ;
- le maître d'œuvre (le fournisseur) — en anglais *project manager* — définit les facteurs qualité du processus de développement à mettre en œuvre pour produire ce qui a été convenu, c'est-à-dire des fonctionnalités assorties d'un couple coût-délai.

Les dispositions négociées se traduiront par un *plan qualité*, « document énonçant les modes opératoires, les ressources et la séquence des activités liées à la qualité, se rapportant à un produit, service, contrat ou projet particulier ». Précédemment appellé *plan assurance qualité* (PAQ), il contient d'une part les caractéristiques qualité du produit et d'autre part les dispositions qualité portant sur le processus. Il est présenté plus longuement au paragraphe 8.7.

La relation maître d'œuvre-maître d'ouvrage s'inscrit dans les dispositions d'assurance qualité (figure 8.1), définie comme « l'ensemble des actions préétablies et systématiques nécessaires pour donner la confiance appropriée en ce qu'un produit ou un service satisfera aux exigences données relatives à la qualité ».

La gestion de la qualité s'est développée ces dernières années et a fait l'objet d'un ensemble de normes applicables à tout secteur d'activité notamment la norme ISO9000, que nous présentons ci-dessous (paragraphe 8.3.3).

Les dispositions prises par le fournisseur pour assurer et gérer la qualité sont rassemblées dans un *manuel qualité*. On en décrit le contenu au paragraphe 8.6.

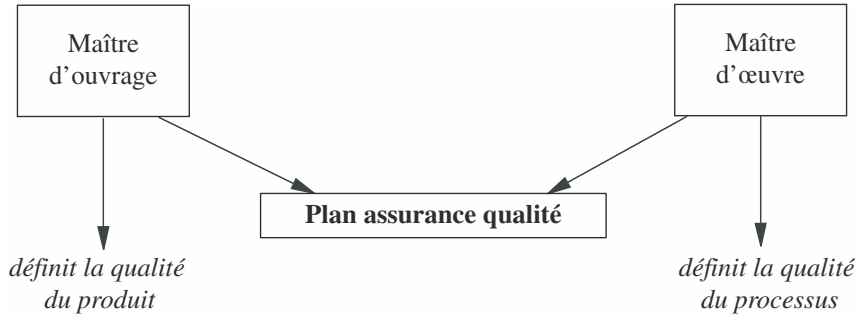


Figure 8.1 — Les dispositions de l'assurance qualité.

8.3 LA NORMALISATION DE LA QUALITÉ : NORMES AFNOR ET ISO

8.3.1 La typologie des normes

La normalisation trouve son origine dans le besoin d'interchangeabilité. Si l'on a pu acheter une pellicule photographique et la mettre sans problème dans n'importe quel appareil, c'est parce que leurs dimensions et leurs tolérances ont fait l'objet de normes internationales.

Ce type de norme, que l'AFNOR qualifie de norme du *premier type*, décrit l'état d'une technique, en spécifiant les caractéristiques des éléments constitutifs et leur assemblage.

Pour certaines exigences, ce type de norme est aujourd'hui dépassé. Ainsi, il y a quelques décennies, les normes de fabrication d'une chaise portaient sur les caractéristiques du bois, la résistance de la colle utilisée, etc. Cet énoncé présente l'inconvénient majeur d'être lié à la technologie utilisée : une évolution de l'état de l'art oblige à modifier la norme.

C'est pourquoi on a préféré définir des normes du *deuxième type*, se présentant sous la forme d'un modèle. Les jouets destinés aux enfants de moins de trois ans et portant la norme NF doivent satisfaire à des exigences de sécurité et d'hygiène, quels que soient leurs dimensions ou matériaux. Une cuisinière à gaz obéit généralement à des normes de premier type (dimensions normalisées) et à des normes de deuxième type (sécurité, répartition des températures dans le four, etc.).

Un *troisième type* de norme est apparu, pour répondre à une exigence de régularité dans la production d'une entreprise : ce sont des normes portant sur l'organisation et la gestion de la qualité elle-même. Elles ne sont donc plus liées à un produit donné, mais s'appliquent à tout type d'entreprise et de projet.

Dans le domaine système d'information, nous retrouvons ces trois types de normes :

- Les normes de type « caractéristiques » sont celles qui sont spécifiques à une entreprise : présentation des écrans, plans types de documentation, utilisation d'outils spécifiques (atelier de génie logiciel, outil de suivi de projet, de maquettage…), structure des noms (de rubrique, de programme, de document).

  Ces normes sont définies par référence à des standards de fait — l'ergonomie Windows — ou par un organisme de normalisation — l'OMG[1] pour le langage de modélisation UML, l'ISO[2] pour le modèle Entité/relation. Elles obéissent à un besoin d'interchangeabilité : pouvoir passer de façon transparente d'un projet à l'autre, ou d'une application à l'autre. Elles doivent être stables et rigoureusement respectées.

- Parmi les normes de type « modèle », on trouve d'abord le modèle général de communication OSI. On trouve également les méthodes de conception, qui ne décrivent pas la solution mais des exigences (élaboration de modèles, de maquettes, de prototypes, approche par niveau…). Elles créent une culture commune.

- La principale norme de type « système qualité » est ISO9000, dont la dernière version date de 2000 et qui sert de base à la certification qualité. La norme ISO10006, qui porte sur la qualité du management de projet, peut également être rangée dans cette catégorie. Elle est décrite au chapitre 16.

8.3.2 Les normes AFNOR

NF X50-120 (septembre 1987) définit le vocabulaire qualité et les termes anglais correspondants.

NF X50-126 (octobre 1986) propose un guide d'évaluation des coûts de la non-qualité. Des études ont montré que la non-qualité, variable d'une entreprise à l'autre, représente en moyenne 10 % du chiffre d'affaires. La norme propose un cadre d'évaluation (figure 8.2).

Elle distingue :

- les anomalies internes : ce sont les défauts détectés au cours du processus de production, avant la mise à disposition du produit ;

1. *Object Management Group*.
2. *International Standard Organisation*.

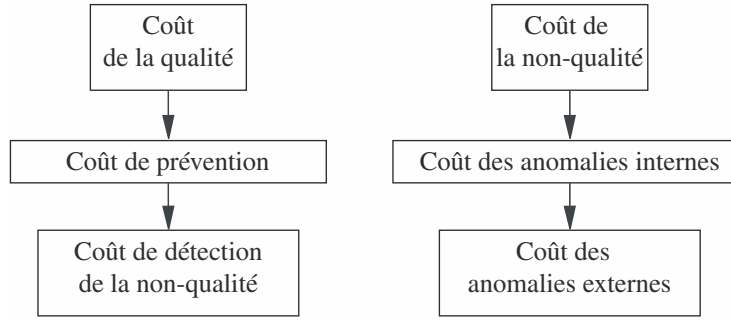


FIGURE 8.2 — Évaluation des coûts de la qualité et de la non-qualité.

- les anomalies externes : elles sont signalées par le client et nécessitent une prise en charge des réclamations et de la maintenance corrective ;
- la détection des défauts, c'est-à-dire les moyens matériels et humains pour rechercher d'éventuelles anomalies ;
- la prévention, c'est-à-dire le dispositif nécessaire pour prévenir les anomalies.

8.3.3 La norme ISO9000

La croissance du commerce international après la Deuxième Guerre mondiale a poussé les gouvernements à s'accorder sur un minimum de normes techniques pour faciliter les échanges internationaux. Des instances internationales mais non gouvernementales se sont mises en place à l'ONU, notamment l'ISO. Ces normes constituent une référence d'application non obligatoire.

Sur le plan européen, le CEN[1] a vu son rôle s'amplifier en 1985, quand le conseil des ministres de la CEE a décidé que la réglementation communautaire élaborée par la Commission devrait se borner à fixer les exigences principales de sécurité, en laissant le soin aux organismes de normalisation de définir les normes européennes. Certains appels d'offres, notamment communautaires, exigent le respect des normes.

La particularité de la norme ISO9000 est de s'intéresser non aux produits mais à l'organisation de l'entreprise. Le principe de base est qu'un système de production, comme le déroulement d'un projet, se trouve placé dans un environnement qui a un impact non négligeable sur les résultats. D'où l'idée que la qualité doit être gérée de façon permanente, en dehors de tout projet, par un système qualité, c'est-à-dire l'« ensemble de la structure organisationnelle, des responsabilités, des procédures, des procédés et des ressources pour mettre en œuvre la gestion de la qualité ». La prévention est donc privilégiée, au lieu de se limiter à la recherche de non-conformités.

1. Comité européen de normalisation, situé à Bruxelles.

Les principaux concepts sont définis de la façon suivante :

- *Politique qualité* : engagement et principes de la direction générale en matière de qualité.
- *Gestion de la qualité* : ensemble des activités qualité qui relèvent de la direction générale, c'est-à-dire la planification des actions qualité, l'allocation de ressources pour la qualité, l'organisation de la qualité et l'évaluation. Son objectif est de rechercher un équilibre économique, en évitant aussi bien la sous-qualité que la sur-qualité.
- *Système qualité* : c'est le dispositif organisationnel (responsabilités et procédures).
- *Maîtrise de la qualité* : ce sont les activités et techniques mises en œuvre pour suivre un processus et éliminer les causes d'anomalie. Par exemple, les activités mises en œuvre sur un projet spécifique relèvent de la maîtrise.
- *Assurance qualité* : ce sont les dispositions qui visent à donner confiance dans le fait que le produit du service sera de qualité, soit à la direction générale (assurance qualité interne) soit au client (assurance qualité externe). L'assurance qualité implique la formalisation et le contrôle.

8.4 LA QUALITÉ DES SYSTÈMES D'INFORMATION

C'est une préoccupation relativement récente. Les premières normes relatives au logiciel sont apparues au début des années soixante-dix aux États-Unis sous l'impulsion des militaires, puis de l'IEEE[1]. Elles portent sur les langages (Cobol-ANS) et les structures de données (Codasyl). Elles se sont ensuite largement développées. Citons notamment la norme ISO12207 qui définit les étapes de développement[2], selon le découpage classique présenté au chapitre 2 (paragraphe 2.5).

La notion de qualité des logiciels est issue des industries aéronautiques et spatiales, préoccupées de fiabilité. À la fin des années quatre-vingt, l'importance des budgets informatiques, la place des applications dans le fonctionnement de l'entreprise, la part dominante consacrée à la maintenance des logiciels et les problèmes rencontrés dans la plupart des grands projets informatiques ont porté la qualité des systèmes d'information au rang des préoccupations majeures dans les entreprises.

1. *Institute of Electronics and Electrical Engineers*.
2. *Software life-cycle process*.

On peut considérer que la partie logicielle d'un système d'information est un produit ; il présente toutefois des caractéristiques qui marquent la problématique de sa qualité :

- il est immatériel ;
- il est reproductible ;
- il nécessite une maintenance ;
- il possède une dimension subjective.

C'est un produit *immatériel* : personne n'a jamais vu un logiciel, on ne le perçoit qu'à travers les représentations que sont le code et la documentation. On peut s'en faire une idée en essayant d'imaginer de façon précise, par exemple pour en faire une image, une maison, à partir d'une simple description littérale. Cela demande un effort d'abstraction important, pour un objet par ailleurs familier, ce qui donne une idée de la difficulté à laquelle on se heurte dans le cas d'un logiciel.

Par ailleurs, le passage d'une représentation à l'autre (documentation/code) pose le problème de leur fidélité : des erreurs sont susceptibles de s'introduire.

C'est un produit *reproductible* : la reproduction en série d'un logiciel ne pose aucun problème. Le processus de fabrication est un processus de fabrication unitaire, ce qui est profondément différent des industries de masse où est née et s'est ancrée la réflexion qualité. Dans le domaine industriel, le processus de production devait assurer au meilleur coût la reproduction d'un modèle. La qualité est mesurée par la ressemblance du produit avec le modèle. Pour un logiciel, tout l'effort doit porter sur la conception et la mise au point du modèle. Même si les méthodes de conception ont formalisé des règles, des techniques, une démarche, réutilisables en partie ou en totalité sur chaque projet, l'application de la méthode ne garantit pas le résultat. La différence de nature entre méthode de conception de système d'information et méthode industrielle, au sens d'un bureau des méthodes dans une usine, peut être représentée par la différence entre les deux consignes : « élaborer le modèle conceptuel du futur système » et « reproduire fidèlement le modèle conceptuel client ».

La qualité étant définie par l'aptitude d'un produit ou d'un service à satisfaire les besoins des utilisateurs, une des grandes difficultés en système d'information, du fait de l'immatérialité des logiciels, est d'identifier les utilisateurs et leurs besoins.

Le coût d'un logiciel ne se limite pas au coût de développement, mais doit tenir compte du *coût de maintenance*. Différentes études ont montré que lorsqu'une erreur coûte un franc si elle est détectée à l'étape de conception, elle en coûtera près de quarante à l'étape de réalisation et plus de cent en maintenance corrective.

Par ailleurs, une partie de la maintenance évolutive provient des limites de la conception.

Un système d'information fait partie d'un système socio-technique : la qualité perçue comporte, outre sa dimension objective qui est la conformité aux spécifications du cahier des charges, une *dimension subjective* dépendant du contexte relationnel. Cet aspect fait qu'une partie de la qualité d'un système d'information, comme dans le domaine des services, est concomitante à la production et ne se retouche pas *a posteriori*.

Les caractéristiques d'un système d'information expliquent la lenteur de diffusion de la démarche qualité, ainsi que les évolutions dans l'interprétation des principes qualité. Les années quatre-vingt ont été marquées par des essais-erreurs. Les années 90 sont celles de la mise en œuvre effective.

8.5 LES FACTEURS ET LES INDICATEURS DE LA QUALITÉ D'UN SYSTÈME D'INFORMATION

8.5.1 Les facteurs qualité d'un système d'information

Dans une première approche, on peut considérer un logiciel sous deux angles : les fonctions qu'il réalise et les caractéristiques de son utilisation.

Un logiciel remplit certaines *fonctions* : calcul de paie, suivi des consommations, élaboration de facture… Elles sont décrites au travers des différents modèles de représentation du système d'information. On ne peut parler de qualité si la fonction attendue n'est pas exécutée correctement.

L'*utilisation* d'un logiciel comprend sa manipulation, les conditions de son exploitation, la correction des erreurs résiduelles (maintenance corrective) et les évolutions fonctionnelles (maintenance évolutive).

Les exigences vont être différentes selon :

- la durée de vie du logiciel ;
- le caractère critique ou non des erreurs (logiciel de jeu ou logiciel de comptabilité) ;
- le type d'utilisateur.

Ces caractéristiques correspondent à des besoins du client tout aussi fondamentaux que les fonctions réalisées par le logiciel. Or, elles sont en général exprimées de façon partielle.

B.W. Boehm a proposé, vers le milieu des années soixante-dix, une première caractérisation de la qualité.

Un logiciel doit être :

- utilisable en l'état ;
- maintenable ;
- portable.

J. McCall, de la General Electric, a raffiné la définition de Boehm dans son ouvrage *Factors in Software Quality* (Facteurs de la qualité du logiciel) paru en 1978. Il a introduit deux éléments :

- une typologie des facteurs, selon différents points de vue ;
- des niveaux d'expression allant du qualitatif au quantitatif.

On distingue ainsi trois niveaux de description :

1. les *facteurs* correspondent à un besoin ;
2. les *critères* sont des caractéristiques du logiciel, associées aux facteurs ;
3. les *métriques* sont des variables permettant de mesurer l'atteinte d'un critère.

La démarche consiste donc, au cours d'un dialogue maître d'œuvre-maître d'ouvrage, à expliciter les facteurs correspondant à la qualité attendue, puis à préciser les critères correspondants et enfin à déterminer les métriques.

Celles-ci doivent être simples, fiables et faciles à auditer. La simplicité de la mesure signifie que le phénomène ou l'effet retenu peut être mesuré de façon peu coûteuse, dans un temps court et de façon aisément compréhensible. La fiabilité est la capacité à reproduire la même valeur sur les mêmes éléments se rapportant à la même période. Elle ne doit pas dépendre de celui qui effectue la mesure. La facilité d'audit signifie qu'un tiers indépendant peut vérifier la bonne application des règles et procédures de mesure. Celles-ci doivent être explicites.

Les **facteurs qualité** (d'après Boehm-McCall) sont répartis selon quatre points de vue : fonctionnel, utilisation, maintenance et économique.

Le point de vue **fonctionnel,** parfois appelé conceptuel, est celui des besoins fonctionnels.

Il comporte trois facteurs :

1. la pertinence ;
2. l'adéquation ;
3. la généralité.

La *pertinence* est la capacité de répondre au problème de l'entreprise. C'est le facteur dont la réponse est la plus spécifique de chaque projet. Pour trouver critères

et métriques associés, il faudra s'interroger sur l'impact du système d'information sur la gestion de l'entreprise.

L'*adéquation* est celle du logiciel à l'organisation et aux procédures. Applications informatiques et procédures de travail doivent faire un tout. Deux points sont à étudier particulièrement : la prise de décision et les contrôles. L'automatisation complète est rarement possible. Certains cas particuliers relèvent d'une décision humaine, moins coûteuse et capable de traiter des situations imprévues. De même, la sécurité du système d'information, tout en s'appuyant sur les dispositifs classiques, doit être renforcée par des interventions humaines.

La *généralité* est l'aptitude de la solution à résoudre des problèmes de portée plus large que le contexte particulier du projet. Ce facteur est particulièrement important dans le cas où l'on conçoit un progiciel. Il est souvent recherché pour des domaines présentant plusieurs variantes (gestion des contrats d'assurance, gestion des ordres de bourse, gestion des crédits…).

Le point de vue de l'**utilisation** est celui de la mise en œuvre et de l'exploitation du logiciel. Il comporte cinq facteurs :

1. la maniabilité ;
2. la fiabilité ;
3. l'efficience ;
4. la confidentialité ;
5. la couplabilité.

La *maniabilité* est l'aptitude d'un logiciel à être convivial et facile d'emploi pour le type d'utilisateur auquel il est destiné. Ce facteur concerne les interfaces homme-machine, mais aussi le degré de paramétrisation accessible à l'usager, les possibilités d'autoformation, etc.

La *fiabilité* est l'aptitude d'un logiciel à accomplir sans défaillance l'ensemble des fonctions spécifiées dans un document de référence, pour une durée d'utilisation donnée[1]. La fiabilité suppose que le logiciel est stable et constant, même en présence d'événements non souhaités.

L'*efficience* est l'aptitude d'un logiciel à minimiser l'utilisation des ressources disponibles.

La *confidentialité* est l'aptitude d'un logiciel à être protégé contre tout accès par des personnes non autorisées, aussi bien en que hors exploitation.

1. Norme AFNOR Z61-102.

La *couplabilité*, ou interopérabilité, est l'aptitude d'un logiciel à communiquer ou interagir avec d'autres systèmes, logiciel de base ou autre applicatif. Cette caractéristique permet notamment l'échange de données avec d'autres systèmes et leur utilisation.

Le point de vue de la **maintenance** du logiciel est celui de son évolution. Il comporte trois facteurs :

1. la maintenabilité ;
2. l'adaptabilité ;
3. la portabilité.

La *maintenabilité* est degré de facilité avec laquelle on peut localiser et corriger des erreurs résiduelles. Cette caractéristique a un effet direct sur les délais de maintenance corrective. Elle est liée à la durée de vie du logiciel.

L'*adaptabilité* est l'aptitude d'un logiciel à évoluer facilement, si l'on veut modifier ou ajouter des fonctionnalités. Cette caractéristique a un effet direct sur les délais de maintenance évolutive.

La *portabilité* est le degré de facilité avec laquelle on peut transférer le logiciel dans un autre environnement, logiciel (système d'exploitation, système de gestion de bases de données) ou matériel. Cette caractéristique a un effet direct sur les modifications nécessaires en cas de transfert.

Le point de vue **économique** est celui de la rentabilité des applications. Il se traduit par un seul facteur prenant en compte, à la fois les coûts de développement, les coûts d'exploitation et les gains effectifs dus au logiciel : c'est l'*efficacité* du logiciel.

Certains facteurs qualité sont antinomiques. Ainsi, une exigence élevée en matière de confidentialité est souvent peu compatible avec la maniabilité. De même la recherche de l'efficience maximale ne garantit ni la maintenabilité, ni l'adaptabilité, ni la portabilité.

Fonctionnel	Utilisation	Maintenance	Économique
Pertinence	Maniabilité	Maintenabilité	Efficacité
Adéquation Généralité	Fiabilité Efficience Confidentialité Couplabilité	Adaptabilité Portabilité	

Figure 8.4 — Les facteurs qualité d'un système d'information selon les différents points de vue.

Il s'agit avant tout de faire un choix et de préciser ce qui fait qu'on jugera le système « bon ».

La définition des caractéristiques qualité a fait l'objet de la norme ISO9126, qui s'appuie sur les travaux de Boehm-McCall. Elle retient six facteurs : capacité fonctionnelle, fiabilité, facilité d'utilisation, maintenabilité, portabilité et efficacité.

8.5.2 Les critères qualité d'un système d'information

Chaque facteur peut être caractérisé par plusieurs *critères qualité* ; un critère peut concourir à l'obtention de plusieurs facteurs. Certains critères sont propres au projet ; c'est notamment le cas de ceux qui sont associés aux facteurs du point de vue fonctionnel. D'autres ont un caractère plus général. Nous allons présenter ces derniers, en les regroupant par facteur.

Le facteur **maniabilité** comporte trois critères *:*

1. la communicabilité ;
2. l'exploitabilité ;
3. la facilité d'apprentissage.

La *communicabilité* est la capacité du logiciel de permettre un dialogue aisé entre l'homme et la machine. Ce critère doit être apprécié en fonction des utilisateurs potentiels et des conditions d'utilisation. La présentation des résultats ou accessibilité aux commandes doivent être adaptées.

L'*exploitabilité* est la facilité à mettre en œuvre et à utiliser le logiciel. Ce critère peut induire le développement d'aides en ligne, de reprise en cas d'erreur, etc.

La *facilité d'apprentissage* est mesurée par le temps moyen nécessaire pour utiliser le logiciel de façon autonome. On peut parfois demander que la majeure partie de l'apprentissage résulte d'une autoformation.

Le facteur **fiabilité** comporte également trois critères :

1. la complexité ;
2. la tolérance aux fautes ;
3. l'auditabilité.

Le critère de *complexité* mesure l'effort nécessaire pour comprendre et analyser le logiciel. La fiabilité décroit quand la complexité augmente. Celle-ci dépend de sa construction algorithmique et de sa taille.

Plusieurs métriques ont été proposées pour la mesurer :

- le nombre de lignes de code, car il est corrélé avec le nombre d'erreurs probables ;
- le nombre d'arcs divisé par le nombre de sommets sur l'organigramme du programme : c'est l'indicateur de complexité de structure de MacCabe ;
- le nombre d'opérateurs arithmétiques et logiques, ajouté au nombre de variables ; cette valeur mesure la complexité textuelle d'un programme : c'est la mesure de Halstead.

Le critère de *tolérance aux fautes* renvoie à la possibilité de limiter les effets d'une perturbation, d'origine interne ou externe, sur logiciel. Les Japonais utilisent le concept de *poka-yoké*, signifiant anti-erreur humaine. Il est illustré par ces fiches de connexion, dont les formes rendent impossible les erreurs de branchement.

Le principe général de la tolérance aux fautes est d'éviter que des erreurs soient commises (en posant ce que l'on appelle des « chiens de garde »), de vérifier que des fautes n'ont pas été commises (par exemple en recoupant certaines informations) et de pouvoir corriger les erreurs.

L'*auditabilité* est la capacité du logiciel de permettre de retrouver rapidement et sans difficulté la trace d'une opération ; la recherche doit pouvoir être menée du haut vers le bas (repérer toutes les conséquences d'un événement donné) et du bas vers le haut (retrouver ce qui a permis d'arriver à un résultat).

Le facteur **efficience** traduit la capacité à utiliser au mieux les ressources informatiques.

Les ressources dont on peut chercher à optimiser l'utilisation sont :

- la consommation en *place mémoire* ;
- la taille des *périphériques* et leur vitesse d'accès ;
- le *temps d'exécution* des programmes.

Le facteur **confidentialité** est mis en œuvre à travers deux critères :

1. la protection du code et des données ;
2. la mémorisation des accès.

La *protection du code et des données* est la limitation, en exploitation ou non, des accès à des personnes autorisées.

La *mémorisation des accès* permet de retrouver la trace des manipulations. Pour cela, les accès à un programme, une zone de données ou un fichier doivent être historisés.

Deux critères favorisent la **couplabilité** :

1. la standardisation des données ;
2. la standardisation des interfaces.

La *standardisation des données* est la compatibilité des données avec des standards de représentation. On peut ainsi utiliser pour les échanges monétaires internes d'une banque des normes interbancaires.

La *standardisation des interfaces* : dans la logique de l'approche objet, il faut s'attacher à distinguer entre les fonctions dites publiques, c'est-à-dire pouvant être appelées par d'autres applicatifs, et les fonctions privées. Les appels doivent être normalisés.

Le facteur **maintenabilité** comporte quatre critères :

1. la lisibilité ;
2. la modularité ;
3. la traçabilité ;
4. l'adaptabilité.

La *lisibilité* est la capacité d'un programme ou de sa documentation de pouvoir être lus par d'autres personnes que leur auteur. Ils sont considérés comme lisibles si on peut les comprendre par simple lecture. La lisibilité dépend d'une part du vocabulaire utilisé et de l'expression écrite, d'autre part de l'absence d'éléments mal définis, c'est-à-dire nécessitant pour être compris des connaissances préalables ou des informations complémentaires. Les éléments superflus doivent être évités. La lisibilité facilite la maintenance des programmes.

La *modularité* d'un logiciel représente l'indépendance de ses composants. On peut distinguer deux niveaux. La modularité des spécifications se traduit par des fonctionnalités bien identifiées avec leurs relations et par une autonomie par rapport à l'environnement matériel et logiciel. La modularité des composants logiciels permet de modifier un composant sans effet sur les autres**.**

Le critère de *traçabilité* vient des industries où il est parfois nécessaire de retrouver l'origine de chaque élément monté sur le produit fini (quel fournisseur ? quelle livraison ? quel lot ?…), notamment l'aéronautique. Dans notre domaine, la traçabilité traduit l'existence de liens entre les différentes représentations

graphiques et textuelles d'un logiciel. Par exemple, elle permet de savoir où se retrouve tel objet ou telle opération du modèle conceptuel.

Le critère *adaptabilité* met en jeu la notion d'expansibilité, c'est-à-dire la possibilité d'augmenter les zones de données et la taille des programmes. C'est un des avantages majeurs des systèmes de gestion de base de données relationnels par rapport aux systèmes hiérarchiques ou réseaux. Ce critère inclut la possibilité d'identifier facilement les impacts dans tous les programmes touchés.

Le facteur **portabilité** se décline selon trois critères :

1. la banalité d'emploi ;
2. l'indépendance ;
3. la qualité de la documentation.

Le critère de *banalité d'emploi* s'applique à une fonction et traduit son indépendance par rapport à une application. Certaines fonctions sont classiquement banalisées, par exemple, les transformations de format de date ou le calcul du nombre de jours entre deux dates données. Les paramètres de référence sont gérés dans des tables externes.

Le critère d'*indépendance* est le degré de liberté du logiciel par rapport à l'environnement logiciel ou matériel et l'absence de liens structurels. Ainsi, l'indépendance par rapport à un système de gestion de bases de données nécessitera par exemple l'externalisation de tous les accès et leur localisation dans un sous-programme unique.

La *qualité de la documentation* porte sur le fond ou sur la forme. Les documents sont les principaux produits des étapes de conception et constituent des représentations du logiciel (documentation d'utilisation, d'exploitation et de maintenance).

Maniabilité	**Fiabilité**	**Efficience**	**Confidentialité**
Communicabilité Exploitabilité Facilité d'apprentissage	Complexité Tolérance aux fautes Auditabilité	Consommation de place mémoire Taille et vitesse d'accès aux périphériques Temps d'exécution	Protection du code et des données Mémorisation des accès

Couplabilité	**Maintenabilité**	**Portabilité**
Standardisation des données Standardisation des interfaces	Lisibilité Modularité Traçabilité Adaptabilité	Banalité d'emploi Indépendance Qualité de la documentation

Figure 8.5 — Les critères qualité associés aux différents facteurs qualité d'un système d'information.

Sur le fond, il faut éviter trois écueils : la *contradiction* entre deux parties d'un document ou entre deux documents ; le *silence*, c'est-à-dire l'absence d'éléments importants ou permettant la compréhension et l'*ambiguïté*, à savoir une expression donnant lieu à des interprétations multiples.

Sur la forme, il faut minimiser quatre travers : la *redondance*, marquée par la présence d'éléments identiques dans le même document ou dans deux documents ; le *bruit* que représente la présence d'éléments superflus ; le *sur-détail*, c'est-à-dire la présence d'éléments d'un niveau de détail trop fin par rapport à l'objectif du document et le *non-respect des normes* telles qu'un plan type ou des normes de présentation.

8.5.3 L'utilisation des caractéristiques de la qualité

Le premier intérêt de la *démarche facteur-critère-métrique* est d'offrir un cadre dans l'approche négociée des caractéristiques qualité. Par un examen systématique des différents aspects, il permet d'expliciter les attentes qualité des acteurs, ceux qui utilisent le système d'information (gestionnaire, utilisateur final) et ceux qui le font vivre (exploitation et maintenance).

Par ailleurs, la démarche fait apparaître d'éventuelles contradictions dans les exigences, résultant de la présence de facteurs antinomiques. La démarche oblige donc à faire des choix, à hiérarchiser facteurs et critères. Pour les facteurs fonctionnels, elle favorise une projection en posant non pas la question « que voulez-vous ? » mais « qu'est-ce qui fera que vous serez satisfait ? ». C'est pourquoi, la méthode de l'analyse de la valeur[1] est souvent considérée comme un des éléments pouvant concourir à la qualité fonctionnelle des systèmes d'information.

Le second intérêt de la démarche facteur-critère-métrique est de mettre en place une liste de non-conformités à traquer tout au long du processus de développement par le dispositif de contrôle.

Dans un projet, les facteurs, critères et métriques résultent de la négociation entre le maître d'œuvre et le maître d'ouvrage. Celle-ci intègre les préoccupations de coût et délais. Le résultat figure dans le plan assurance qualité.

Voici un exemple de facteurs qualité qui sont des objectifs exprimés par la direction générale d'une grande entreprise de services :

La qualité du système d'information devient un facteur prépondérant pour la réalisation des objectifs majeurs de l'entreprise. Son évolution sera conduite en respectant les principes suivants :

- *ne conduire que des projets dont la rentabilité soit clairement établie et dont la taille garantisse la maîtrise dans des délais courts ;*

1. Normes AFNOR X 50-150, 151, 152, et 153.

- *dans tous les nouveaux projets, mettre systématiquement en œuvre une approche et une architecture client-serveur en s'appuyant sur un poste de travail banalisé ;*
- *s'appuyer sur le système existant en le déformant progressivement et continuellement ;*
- *rechercher une meilleure homogénéité et une plus grande cohérence, conduisant à l'abandon progressif des solutions régionales particulières.*

Ainsi, dans l'exemple ci-dessus, la rentabilité des applications peut être recherchée dans la réduction des coûts, par exemple, la réduction des coûts administratifs, l'amélioration de la logistique, la gestion de stock en flux tendus. Il faut d'autre part développer des d'activités rentables, par exemple, un soutien à l'action commerciale, une amélioration de la gestion du contentieux, la facturation d'un service nouveau…

Pour promouvoir la couplabilité, on parle parfois *du* système d'information, tous les systèmes devant pouvoir communiquer. Ainsi, à partir d'une station de travail unique, on doit pouvoir accéder à l'ensemble des informations.

L'évolutivité demandée correspond à une réalité très forte dans certains secteurs industriels comme l'automobile où, d'une année sur l'autre, le même modèle se modifie. C'est également le cas dans l'urbanisme lorsque l'on rénove progressivement un quartier, au lieu de le raser. Dans le domaine des systèmes d'information, cela se traduit notamment par des habillages ergonomiques, qui évitent de refaire les applications.

La généralité des solutions obéit à un compromis entre cohérence et complexité. La cohérence totale dans une grande organisation est complexe. La cohérence de l'ensemble des parties communes doit coexister avec des parties privées spécifiques.

8.6 LE MANUEL QUALITÉ

La notion et le contenu du *manuel qualité* ont évolué au fur et à mesure que les principes de l'assurance qualité étaient interprétés et expérimentés pratiquement.

Il y a quelques années, le manuel qualité contenait le savoir-faire formalisé de l'entreprise : il se concentrait sur les aspects techniques et méthodologiques du métier. Certaines dispositions représentaient l'état de l'art, d'autres étaient spécifiques à l'entreprise.

Pour l'AFITEP et l'AFNOR, le manuel qualité est « un document décrivant les dispositions générales prises par l'entreprise pour obtenir la qualité de ses produits ou services » [AFITEP, 2000]. Son objectif est double : d'une part informer le personnel sur l'organisation d'ensemble de l'activité et notamment sur le

système qualité de l'entreprise ; d'autre part, résumer pour les clients les mesures adoptées pour assurer la qualité. Ce manuel est parfois considéré comme un outil interne et externe de promotion de la qualité de l'entreprise.

Dans l'ingénierie informatique, on distingue parfois deux volets complémentaires du manuel qualité :

- Le *manuel qualité interne* comporte un volet technique (principes de mise en œuvre des technologies) et un volet méthodologique (par exemple méthodes de conception, méthodes d'évaluation des charges, guide de conception d'interface homme-machine…).
- Le *manuel assurance qualité* contient d'une part la description de l'organisation de la qualité dans l'entreprise (direction qualité, correspondant qualité…), d'autre part l'interprétation que l'entreprise a faite des différents points de la norme ISO9000. Certains points renvoient à des points du manuel qualité interne.

Le manuel qualité sert de base aux audits qualité, internes ou externes.

8.7 LE PLAN QUALITÉ

8.7.1 La définition du plan qualité

L'assurance qualité cherche à généraliser le principe de la *relation client-fournisseur*. Cela signifie que chacun dans l'entreprise (individu ou service) a toujours au moins un client et un fournisseur. Cette approche permet de clarifier et d'améliorer les relations entre partenaires de travail.

La relation client-fournisseur peut être représentée par un schéma simple : le fournisseur est celui qui procure quelque chose dont a besoin le client. Derrière cet aspect de troc, se cache une relation souvent ambiguë. La règle du jeu consiste à obtenir le plus possible de l'autre. Mais c'est aussi un échange de bons procédés dans lequel le fournisseur conseille le client. Il garantit l'absence de vices cachés. Il doit fournir un produit ou un service qui réponde aux besoins exprimés, en s'assurant que cette expression est conforme aux besoins réels du client. Celui-ci s'engage à exprimer son attente de la manière la plus claire possible. Ils conviennent ensemble d'une mesure pertinente du résultat. Cette volonté d'une relation de coopération se matérialise par un contrat.

Le plan qualité joue ce rôle. C'est un outil de clarification entre maître d'œuvre et maître d'ouvrage.

Le plan qualité se distingue du contrat commercial sous trois aspects :

1. Le contrat commercial n'existe que dans les relations entre l'entreprise et son environnement, alors que le plan qualité est utilisable également dans les relations internes.
2. Après description du produit/service à fournir, des conditions financières et modalités de paiement, et des conditions de livraison, le contrat contient toutes les clauses visant les défaillances possibles de l'une ou l'autre partie. Le plan qualité, au contraire décrit pour une large partie des dispositions visant à prévenir les défaillances.
3. Le contrat commercial est en général de diffusion confidentielle, alors que le plan qualité constitue un outil de travail pour l'ensemble des acteurs du projet.

L'AFITEP et l'AFNOR définissent le plan qualité comme un *« document énonçant les modes opératoires, les ressources et la séquence des activités liées à la qualité, se rapportant à un produit, service, contrat ou projet particulier »* [AFITEP, 2000].

Aujourd'hui, le plan qualité est inclus dans le plan de management de projet.

8.7.2 Le contenu du plan qualité

Le plan qualité contient généralement deux parties. On y trouve d'une part la description — suite aux négociations entre maître d'œuvre et maître d'ouvrage — de la qualité attendue du futur système d'information, exprimée par un ensemble de facteurs hiérarchisés, assortis de critères et métriques, ainsi que, d'autre part, le cycle de développement retenu avec pour chaque étape :

- les résultats attendus ;
- les conditions d'acceptation de chaque résultat ;
- les modalités de contrôle ;
- la planification ;
- l'organisation des équipes ;
- les relations entre acteurs (rôles, responsabilités, circulation d'information et de documents…) ;
- les méthodes, normes et outils utilisés.

Le plan qualité ne doit contenir que ce qui est propre au projet. Toute norme, procédure, méthode, etc., qui s'applique à tous les projets relève du manuel qualité interne ou du manuel qualité. On peut en revanche faire référence à certains éléments de ces manuels.

Selon la norme ISO9000-3 (point 5.5.2), le plan qualité doit comprendre la description des points suivants ou leur référence dans d'autres documents s'ils ont été décrits séparément :

- la qualité attendue du produit, exprimée à l'aide de facteurs, critères et métriques ;
- les entrées et sorties de chaque étape du cycle retenu ;
- la nature des tests et des contrôles de la production ;
- le planning des activités de vérification, comprenant le calendrier et les ressources qui sont affectées ;
- l'identification des responsables des différentes activités qualité, telles que la maîtrise des modifications, la gestion des actions correctrices, etc.

De par son contenu, le plan qualité est un document qui est une référence dynamique durant le déroulement du projet : il s'enrichit progressivement selon les phases et peut être modifié si les besoins du client évoluent. Il constitue néanmoins un cadre stable.

8.8 LE CONTRÔLE ET L'AUDIT QUALITÉ

8.8.1 Le contrôle qualité

Le contrôle qualité fait partie des dispositions d'assurance qualité. Il doit être mené en continu tout au long du projet sous différentes formes. Il porte sur les produits élaborés à chaque étape du projet. Ces produits sont de deux natures : des documents et le logiciel.

La qualité des documents

Les documents doivent satisfaire aux critères qualité touchant le fond (absence de contradiction, silence ou ambiguïté) et la forme (absence de redondance, bruit, non-respect des normes).

Selon les acteurs impliqués, les contrôles sont *internes* (les acteurs du contrôle font partie de l'équipe projet) ou *externes* (les acteurs ne font pas partie de la maîtrise d'œuvre). La philosophie générale est de tendre vers une situation où l'on fait bien du premier coup. Les dispositions doivent laisser une part importante à l'*autocontrôle* : celui-ci sera favorisé par une gestion correcte du système qualité, notamment tous les points concernant la diffusion des normes, règles, principes. Cependant, les contrôles effectués par d'autres acteurs obligent en permanence le rédacteur à considérer son produit comme devant servir à d'autres.

Le plan qualité du projet prévoit différents types de contrôles.

L'*inspection* permet d'apprécier la qualité d'un document en fonction d'un référentiel. Le moyen utilisé est la lecture du document par un individu, interne ou extérieur à l'équipe projet. Ce contrôle peut être effectué par le chef de projet, par un utilisateur privilégié ou par le responsable qualité du projet. Il est parfois assuré par un autre concepteur du projet ; l'objectif est de favoriser un autocontrôle ultérieur, par intériorisation des règles et principes d'harmonisation, de cohérence et de lisibilité.

La *lecture croisée* met en jeu deux acteurs appartenant à des projets différents, par exemple deux sous-projets du même projet global. Elle favorise la cohérence entre projets ou sous-projets.

La *revue* est une évaluation collective d'un document. Elle peut être considérée comme un moyen de conforter des pratiques organisationnelles souhaitables. L'organisation, la conduite et le suivi d'une revue sont soumis à des règles générales de bon fonctionnement. Les revues impliquent plusieurs personnes jouant un rôle différent, par exemple :

- Le présentateur du document est en général l'auteur ; c'est parfois le chef de projet.
- Le coordinateur organise la revue ; il envoie préalablement le document à tous les participants, date, lieu… C'est souvent le responsable qualité du projet.
- Le rapporteur est chargé du compte rendu de la revue : diagnostic et actions correctives décidées. C'est souvent le responsable qualité du projet.
- Le lecteur fait une lecture critique du document. Selon que le contrôle est interne ou externe, c'est soit un membre de l'équipe projet (ou éventuellement d'une autre équipe) soit un représentant du maître d'ouvrage. Il peut y avoir plusieurs lecteurs, chacun pouvant représenter un point de vue spécifique.

Pendant la conduite de la revue, tous les participants sont responsables de l'évaluation et l'amélioration du produit contrôlé. Ils doivent garder à l'esprit qu'ils jugent un document et non une personne : les critiques doivent être ciblées et mettre en relief des points nécessitant un complément de travail. La revue ne doit pas se transformer en réunion de conception. Elle ne dépasse pas deux heures.

À l'issue de la revue, un compte rendu rédigé par le rapporteur contient trois parties : identification de la revue, de son objet et de ses participants ; liste des points et problèmes soulevés ; décisions et calendrier de prise en compte des décisions. Ce compte rendu fait partie du système d'information projet.

Les améliorations doivent être apportées par l'auteur initial du document. Le responsable qualité, ou à défaut le chef de projet, est responsable de leur mise en œuvre : si les modifications sont mineures, une inspection sera suffisante. Dans certains cas, une nouvelle revue est organisée. Ces dispositions figurent dans le compte rendu.

Le plan qualité, ou le plan de contrôle si celui-ci est séparé du plan qualité, doit prévoir les différents contrôles prévus :

- type de contrôle ;
- objet du contrôle ;
- date ;
- acteurs ou type d'acteurs.

Il doit être mis à jour, pour intégrer les aléas du projet (contrôle supplémentaire, nouvelle revue, modification d'acteur…).

La qualité des programmes

Le plan de test comprend le calendrier et les modalités de préparation et d'exécution des tests.

Les *tests unitaires*, dits de « boîte blanche » mettent en évidence les erreurs de chacun des composants ; ils sont préparés par le réalisateur avant la programmation. Ils permettent de tester les différents chemins et la validité des conditions d'accès à chaque branche, par exemple, examen des valeurs conditionnelles, vérification des valeurs limites…

Les *tests d'intégration* détectent des erreurs d'interface entre les composants logiciels. Les fonctions testées unitairement sont introduites successivement.

Les *tests de validation fonctionnelle*, dits de « boîte noire » recherchent les erreurs de conception, les omissions et les non-conformités par rapport aux exigences spécifiées.

Ces trois types de test correspondent à trois niveaux de *validation*. Selon le modèle de développement en V, les jeux d'essais sont constitués dans l'ordre inverse de leur utilisation. À l'issue d'une phase de spécification fonctionnelle, on élabore le jeu d'essai de validation fonctionnelle. Celui-ci doit donner lieu à une revue externe avec la maîtrise d'ouvrage. En phase d'analyse technique, on élabore (si possible par enrichissement d'un sous-ensemble du jeu d'essai précédent) les jeux de tests d'intégration. Enfin, avant la programmation, on élabore les jeux de tests unitaires, à partir également des jeux de tests d'intégration.

Les tests effectués sur le site ou dans un environnement opérationnel donnent lieu à une *qualification*.

8.8.2 L'audit qualité

L'*audit qualité* du projet est un moyen d'évaluer la qualité du processus. L'AFNOR[1] le définit comme « l'examen méthodique d'une situation relative à un produit, procédé ou organisation, réalisé en coopération avec les intéressés en vue de vérifier la conformité de cette situation aux dispositions préétablies et l'adéquation de ces dernières à l'objectif recherché ».

Dans les projets systèmes d'information, l'audit peut prendre différentes formes :

- l'audit qualité porte sur le respect des dispositions du plan assurance qualité ;
- l'audit d'avancement porte sur l'état réel du projet par rapport à son avancement annoncé ;
- l'audit fonctionnel porte sur l'adéquation du système conçu aux objectifs annoncés.

Les audits ne sont pas planifiés. Ils s'appuient sur des questionnaires précis et concis, préparés à l'avance. Les audits internes au projet se font à la demande du chef de projet. Les audits externes se font à la demande de la maîtrise d'ouvrage ou de la direction générale. Certains audits de projet peuvent être faits lors d'une procédure de certification ISO9000. Nous allons aborder ce point dans le cadre plus large de la qualification des entreprises.

8.9 LA QUALIFICATION DES ENTREPRISES

8.9.1 La certification

La qualité comporte deux aspects temporellement différents. La *qualité nominale* des produits est l'acception habituelle de la qualité : il y a correspondance avec les besoins et absence de défauts pour un produit précisément identifié. La *régularité de la qualité*, qu'il s'agisse du respect des délais, des caractéristiques du produit ou de la facturation… est un autre aspect. C'est celui qui est visé par la norme ISO9000.

Dès la fin des années soixante-dix, certaines grandes entreprises ont commencé à faire pratiquer des audits qualité, fondés sur des normes sectorielles ou spécifiques, chez leurs sous-traitants. Cela a conduit la plupart des pays occidentaux à mettre en place des systèmes de certification de façon à attribuer une

1. AFNOR X61-102.

certification unique pour une entreprise donnée, homogène d'une entreprise à l'autre et impartiale.

En France l'AFAQ, dont le conseil d'administration comprend l'AFNOR, des acheteurs, des fournisseurs, des sous-traitants…, délivre les certifications. Elle se compose de différents comités sectoriels.

La Grande-Bretagne a instauré un organisme central accréditant les organismes de certification ; il y en a une dizaine. Les Britanniques ont, par ailleurs, créé une certification spécifique pour les technologies de l'information, basée non plus directement sur ISO9000 mais sur l'interprétation qu'en a faite le projet TickIT.

La certification est attribuée au terme d'un audit qualité de l'entreprise, « examen méthodique et indépendant en vue de déterminer si les activités et les résultats relatifs à la qualité satisfont aux dispositions préétablies, et si ces dispositions sont mises en œuvre de façon efficace et aptes à atteindre les objectifs »[1].

L'audit du système qualité dans le domaine ingénierie des systèmes d'information, se déroule en trois phases :

- La *préparation*. Elle comprend notamment la délimitation du champ de la certification (toute l'entreprise, une entité, une activité dans une entité ?) et la constitution de l'équipe d'audit (au moins la moitié de ses membres doivent être des spécialistes logiciel).
- *L'exécution*. Elle comprend l'examen du manuel assurance qualité et l'établissement de sa correspondance avec ISO9000 ; la recherche de non-conformités au niveau des procédures et des instructions de travail ; l'examen, à titre de confirmation, de quelques projets, afin de vérifier l'application effective des dispositions du système qualité ; et la présentation orale des résultats.
- Le rapport *d'audit*. Si le rapport a mis en évidence un nombre limité de non-conformités, d'importance limitée, l'entreprise obtient la certification après correction des non-conformités et vérification par un nouvel audit. Si des anomalies graves ou nombreuses ont été décelées, l'entreprise doit mener une opération de mise en conformité de son système qualité.

La certification est accordée pour trois ans. La plupart des sociétés de services, grandes ou moyennes, ont obtenu la certification pour tout ou partie de leurs entités et de leurs activités. Certaines demandes de certification proviennent des services informatiques internes. Ces derniers recherchent une reconnaissance externe qui leur apportera en retour la légitimité interne.

1. AFNOR X50-120.

8.9.2 L'évaluation de la maturité des entreprises

Mal comprise, la démarche de certification peut conduire à une sur-formalisation (mythe du contrôle total), à la bureaucratisation (la règle prime sur l'intelligence) et à la dispersion des efforts dans l'entreprise (l'objectif devient la certification et non l'amélioration constante de la qualité).

Les entreprises japonaises trouvent le système trop contraignant et contraire à leur approche.

Aux États-Unis, le SEI[1] a mis au point à la demande du DOD[2] une démarche d'évaluation de la maturité des sous-traitants. C'est le SW-CMM[3]. La première ébauche date de 1987, elle s'est affinée au fur et à mesure de son utilisation, surtout en Amérique du Nord et au Japon.

Ce modèle évalue les entreprises selon deux critères (figure 8.7) :

- la maîtrise des processus ;
- la maîtrise technologique.

Les niveaux de maîtrise des processus sont au nombre de cinq : le début, la répétition, la définition, la gestion et l'optimisation.

1. *Au niveau début*, les procédures et les responsabilités sont mal définies. Les principes ne sont pas appliqués de façon cohérente. L'entreprise à ce niveau rencontre d'importants problèmes de coûts et de délais.
2. Au niveau *répétition*, l'entreprise a su mettre en œuvre des procédures de gestion des coûts et des délais, standard d'un projet à l'autre : évaluation des coûts, techniques de planification, gestion des modifications.
3. Au niveau *définition*, l'entreprise a mis en place un groupe spécialisé en génie logiciel, qui définit et met en œuvre des normes et des méthodes, ainsi que des plans de formation pour les différents acteurs. Les procédures sont bien comprises.
4. Au niveau *gestion*, l'entreprise sait mesurer l'impact des actions entreprises. Elle base ses décisions opérationnelles ainsi que les évolutions en matière de normes, organisation et méthodes, sur des données quantifiées qui sont analysées.
5. Au niveau *optimisation*, l'entreprise a orienté son action vers l'amélioration de ses processus, par des analyses rigoureuses des données collectées sur les erreurs, les coûts et les écarts par rapport aux prévisions.

1. *Software Engineering Institute*, de l'Université Carnegie-Mellon, située à Pittsburgh (Pennsylvanie).
2. *Department of Defense* : ministère de la Défense.
3. *Software Capability Maturity Model*.

Les deux niveaux de maîtrise technologique sont qualifiés d'inefficace et efficace.

- Au niveau *inefficace*, des outils peuvent être utilisés de façon généralisée, mais l'entreprise n'en retire qu'un faible bénéfice, faute d'objectifs clairs, de formation suffisante, de cohérence entre les outils et l'exigence de rentabilité.
- Au niveau *efficace*, l'entreprise sait obtenir de bons résultats de l'utilisation généralisée et cohérente de ses outils et techniques. Selon son niveau de maîtrise des processus, elle sera plus ou moins régulière dans ses performances.

Maîtrise des processus / **Maîtrise technologique**	**Début**	**Répétition**	**Définition**	**Gestion**	**Optimisation**
Inefficace					
Efficace					

Figure 8.7 — Modèle de maturité des entreprises.

On évalue la maturité d'une entreprise avec un double objectif. On cherche d'une part à identifier les forces et faiblesses. On veut d'autre part susciter une dynamique d'amélioration des pratiques. La boucle d'amélioration peut être représentée par un schéma tel que celui de la figure 8.8.

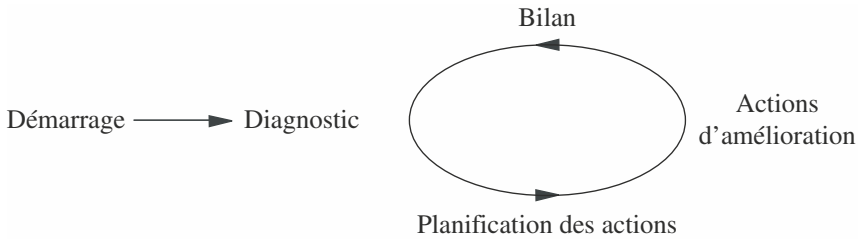


Figure 8.8 — Boucle d'amélioration de la qualité.

Cette approche permet de concrétiser des objectifs de progression. L'expérience montre qu'une majorité d'entreprises se situent au niveau un ou deux pour la maîtrise des processus et au premier niveau de maîtrise technologique.

Une opération d'évaluation dure en général entre quatre et huit mois. Elle représente un investissement d'environ 200 jours-personne pour une entreprise de taille moyenne. Elle est basée sur un questionnaire d'évaluation des pratiques de l'entreprise, notamment en ce qui concerne la définition des besoins, la planification des projets, le suivi des projets, la gestion des sous-traitants, la gestion du changement, etc.

Le modèle du SEI a été repris et adapté par les Canadiens de Bell Canada[1] et par un projet Esprit[2]. Ces deux modèles ne sont pas de nature publique.

Le projet SPICE[3], né d'une initiative britannique en 1992, a été mené par l'ISO. Il intègre l'acquis en matière qualité, notamment les normes ISO9000 et ISO12207. Il a donné lieu à la norme ISO/IEC 15504 qui propose une démarche d'amélioration des processus[4].

Nous pouvons dire en conclusion que l'assurance qualité et la nécessité de fiabiliser durablement tous les processus de l'entreprise reposent sur :

- des objectifs explicités ;
- une organisation du travail claire et souple ;
- un personnel compétent et motivé ;
- des méthodes et des procédés de travail maîtrisés ;
- une documentation précise, concise et complète ;
- un autocontrôle efficace ;
- le maintien d'un contrôle minimal ;
- la traçabilité des actions ;
- le recueil et le traitement des anomalies et des défauts ;
- l'analyse des causes de cette non-qualité et la mise en place de la prévention nécessaire.

L'amélioration de la qualité doit se concevoir comme un processus dynamique et non comme l'atteinte d'un objectif définitif. La certification est un des moyens de cette dynamique.

Elle s'accompagne d'une évolution des concepts fondant les relations entre les acteurs, passant progressivement de la directivité à la participation, de la centralisation à la délégation, de la notion de chef à celle de manager favorisant l'initiative et le développement des membres de son équipe, des rapports de force aux rapports de négociation, de la mise en concurrence au partenariat pour durer ensemble, de la faute à l'obligation de progrès, du bricolage au professionnalisme.

1. Modèle Trillium.
2. Modèle Bootstrap.
3. *Software Process Improvement and Capability Determination*.
4. Pour plus de détails, voir Morley et al., *Processus métiers et S.I.*, Dunod, 2007, particulièrement au chapitre 4.

8.10 LA QUALITÉ DANS LES MÉTHODES AGILES

Les approches agiles sont parfois opposées aux démarches d'amélioration des processus et de maturité des entreprises. En effet, la question de répétition rigoureuse de procédures, périodiquement revues et améliorées, entre dans une certaine contradiction avec l'exigence d'adaptation nécessaire pour atteindre l'agilité.

Par ailleurs, la qualité du logiciel livré est un des objectifs majeurs des méthodes agiles.

Nous allons examiner comment sont prises en compte la qualité du produit et la qualité du processus projet.

8.10.1 La qualité du produit dans les méthodes agiles

La définition de la qualité, rappelée au paragraphe 8.2, met l'accent sur la satisfaction des besoins implicites ou exprimés. Dans cette perspective la démarche itérative et participative des méthodes agiles, avec des ajustements dynamiques aux évolutions des exigences des clients, est parmi les plus aptes à produire des systèmes de qualité.

De plus, les tests sont fréquents, ils sont effectués à différents niveaux, et ils impliquent l'utilisateur qui doit en écrire les jeux d'essai. Ces dispositions jouent largement en faveur de la qualité du produit[1].

L'intégration continue évite les surprises tardives, comme on peut en avoir parfois avec un cycle en V, lorsque plusieurs équipes ont longtemps travaillé en parallèle et que le moment est venu d'intégrer le résultat de leurs travaux.

8.10.2 La qualité du processus dans les méthodes agiles

Les méthodes apparaissent discrètes sur ce point, dans le sens où l'accent n'est pas mis sur la formalisation des dispositions dans un Plan Qualité ou sur des audits qualité, même si ces derniers sont évoqués par DSDM.

En revanche, le contrôle qualité est fort développé, à travers les validations des utilisateurs, à chaque itération de développement.

On peut donc considérer que les méthodes agiles ont une approche de la qualité qui repose davantage sur les personnes que sur des procédures.

1. Pour une description des dispositions spécifiques aux tests dans XP, voir par exemple J.L. Bénard, L. Bossavit, R. Médina et D. Williams, *Gestion de projet eXtreme Programming*, Eyrolles, 2005, chapitre 4.

DEUXIÈME PARTIE

MISE EN ŒUVRE, EXERCICES ET ÉTUDES DE CAS

La deuxième partie de cet ouvrage illustre les principes et les techniques du management d'un projet système d'information exposés dans la première partie.

Le chapitre 9 est consacré à l'examen d'un projet, utilisant le diagnostic des risques. Cette analyse sert à prendre des décisions sur la stratégie de projet et conduit à établir un plan de management de projet. Deux cas sont proposés, tous deux issus de situations réelles.

Le chapitre 10 permet d'appliquer sur le même cas les différentes techniques d'estimation des charges : méthode Delphi, répartition proportionnelle, modèle Cocomo, évaluation analytique, méthode des points fonctionnels.

Le chapitre 11 propose une série d'exercices de planification mettant en œuvre les techniques du réseau des antécédents et du diagramme de Gantt. Ils illustrent l'utilisation pratique de ces outils. Tous les exercices sont corrigés.

Le chapitre 12 expose une démarche simplifiée de suivi de projet.

Le chapitre 13 offre une mise en pratique des choix concernant la management d'équipe et la gestion des conflits.

Le chapitre 14 propose un plan qualité associé à l'un des cas analysés au chapitre 9.

La deuxième partie s'achève sur une illustration de l'utilisation d'un outil de planification et suivi de projet qui reprend le déroulement du cas Parking du chapitre 12.

9

La maîtrise des risques

9.1 LE CAS MÉCANO

Le cas Mécano va permettre d'illustrer l'analyse de risque présentée au chapitre 6 (paragraphe 6.3), ainsi que l'établissement d'une stratégie de management de projet (paragraphe 6.4).

9.1.1 Description de l'entreprise Mécano

Mécano est une filiale d'un grand groupe automobile, spécialisée dans la construction de boîtes de vitesse. L'organigramme simplifié de la figure 9.1 comporte les principaux services, avec le nombre de personnes affectées à chacun.

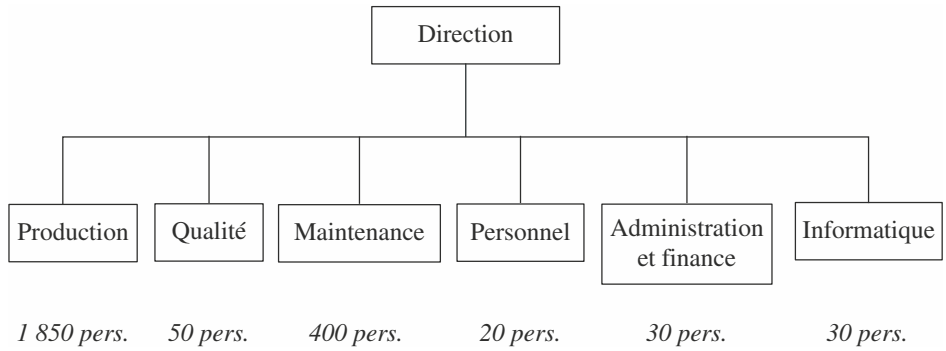


Figure 9.1 — Organigramme de l'entreprise Mécano.

Cette entreprise de 2 400 personnes est en permanence à la limite de ses capacités de production. Le directeur de l'entreprise et le chef du service informatique ont décidé de mettre à disposition du service maintenance, chargé de la réparation et de l'entretien des machines de fabrication, un outil informatique de maintenance assistée par ordinateur (MAO). Ce service passe un temps important à rechercher des informations, notamment sur les machines en panne (plans) et sur la disponibilité des pièces nécessaires. Il y a ainsi, en permanence, une trentaine de personnes à la recherche d'un plan. Le délai moyen d'intervention est de deux heures et la durée moyenne d'une intervention de trois heures.

9.1.2 Description du projet Mécano

Le futur système ne sera pas indispensable pour effectuer les tâches concrètes et immédiates d'entretien et de réparation des machines. Il comprend les fonctions suivantes : saisie des demandes d'intervention, planification des interventions, suivi des interventions en cours, solde d'une intervention. Les données concernées sont actuellement gérées sur papier, notamment la nomenclature des machines et les plans.

On devra pouvoir réserver des pièces si elles sont disponibles en stock ou déclencher automatiquement des commandes. Un certain nombre de statistiques sur les pannes seront fournies. Un module permettra de mettre en place une maintenance préventive. Un système de gestion de base de données objet pourrait être expérimenté à cette occasion.

Les utilisateurs possibles sont divers, la plupart peu familiers des étapes de conception informatique (ouvriers de chaîne de fabrication, chefs d'équipe…) et de l'utilisation des moyens informatiques.

Différents services de l'usine sont touchés par le projet : le service production, le service maintenance, le bureau des méthodes (par exemple des statistiques pour expliquer des baisses de rendement de la production) et le service administratif et financier (statistiques pour calculer le coût d'une intervention).

Après avoir analysé le projet et élaboré votre diagnostic du risque, choisissez un modèle de cycle de vie et proposez une stratégie de projet.

9.1.3 Analyse du projet Mécano

L'objectif du projet est d'augmenter la capacité réelle de production, en réduisant les temps de panne.

L'observation d'une opération de maintenance fait apparaître la *pertinence* du projet. La durée moyenne de traitement d'une panne est longue. Elle se construit en deux temps : d'abord la réaction à l'événement panne, puis l'intervention

elle-même. Or, le service maintenance représente plus de 20 % du personnel directement productif. On relève une perte sensible de temps et d'énergie dans la recherche d'informations non disponibles ou éparpillées. On peut donc considérer qu'en organisant la gestion et la circulation de l'information, on réduira sensiblement la longueur du cycle de traitement.

Mais l'ambition du projet dépasse le raccourcissement des circuits et la mise à disposition immédiate d'information. Il vise également à développer une maintenance préventive, basée sur l'analyse des incidents sur les machines.

La *légitimité* du projet dans l'entreprise est garantie par l'engagement du directeur.

Puisque l'objectif du projet est clair et pertinent et que son positionnement assure que des moyens y seront affectés, évaluons les risques qu'il présente.

9.1.4 Profil de risque du projet Mécano

Le profil de risque est donné à la figure 9.2.

Le projet est de taille plutôt importante. En effet les informations de base (nomenclatures machines, outillages…) ne sont pas encore structurées et font partie de l'étude. De plus, le degré d'automatisation pour la maintenance préventive est a priori élevé. La première exigence est impérative : on ne peut pas faire l'économie des informations périphériques aux interventions. En revanche, le périmètre de la prévention est à géométrie variable.

La difficulté technique provient de la nouveauté du système de gestion de base de données et la possible utilisation du multimédia pour gérer les plans.

Le degré d'intégration est élevé. L'application devra être interfacée avec la gestion des stocks et la gestion des achats, comme aussi probablement avec la base personnel, l'application contrôle de gestion, voire la planification de production si celle-ci devait intervenir dans une affectation de priorité.

La configuration organisationnelle est très large : le nombre d'utilisateurs concernés est de plusieurs centaines, et leur diversité recouvre la plus grande partie de l'usine. Quantité et variété génèrent un risque important.

Le changement envisagé est important. Il porte sur deux points. D'une part, le futur système va modifier le rôle des ouvriers de production et de maintenance lors du constat d'une panne. La transparence introduite par un suivi automatisé va diminuer d'éventuelles « zones d'incertitude[1] ». Ceci risque de provoquer une résistance ou un détournement dans l'utilisation du système. Si ce dernier n'est

1. Selon le concept développé par Michel Crozier dans *L'Acteur et le système*, Seuil, 1977.

pas un passage obligé pour effectuer les réparations, il faudra amener les futurs acteurs à l'utiliser et à l'alimenter correctement. En effet, la qualité des informations de gestion que l'on pourra obtenir en sous-produit dépend de l'exactitude et la complétude des informations opérationnelles.

D'autre part, tout le volet maintenance préventive est entièrement nouveau. Il y a donc là un effort de conception très important.

L'équipe de projet ne présente a priori guère de risques, puisque l'on s'appuie sur des ressources internes. La seule interrogation porte sur les compétences autour des nouvelles techniques : faut-il miser sur la formation ou s'appuyer sur une ressource externe ?

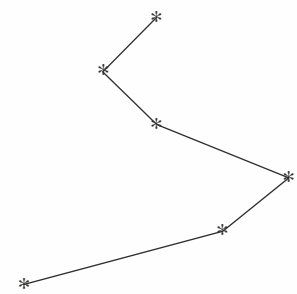
Nature du risque	Degré du risque pour le projet 0 1 2 3 4
Taille du projet	
Difficulté technique	
Degré d'intégration	
Configuration organisationnelle	
Changement	
Instabilité de l'équipe de projet	

Figure 9.2 — Profil de risque du projet Mécano.

9.1.5 Détermination de la stratégie de projet du projet Mécano

Examinons tout d'abord les risques les plus importants et voyons s'il est possible de les réduire.

La configuration organisationnelle peut représenter une pesanteur dans la conduite du projet : il faut s'arranger pour découper le projet en modules, qui chacun ne touchera qu'un nombre limité d'entités. En effet, le cœur du système concerne les services production et maintenance, les autres services sont utilisateurs des données produites. Il faudra néanmoins s'appuyer sur des utilisateurs délégués. Il serait souhaitable de trouver dans le service production ou le service maintenance un cadre ayant une bonne connaissance de l'usine, qui puisse être détaché sur le projet pour participer à la conception générale. Ce cadre devrait être convaincu de l'utilité du projet pour s'en faire ensuite l'avocat auprès de ses pairs.

Ensuite, la gestion du changement doit être prise sérieusement en compte. On va donc retenir une approche itérative, basée sur des cycles comprenant l'élaboration d'un prototype suivie de tests par les utilisateurs. Les utilisateurs finaux étant trop nombreux pour être tous associés directement au projet, on peut constituer un groupe témoin d'une vingtaine de personnes. Celles-ci, sur la base du volontariat, seront prises dans les deux services clés : maintenance et production. Les temps d'expérimentation seront de l'ordre d'une ou deux semaines, les suggestions étant ensuite prises en compte après une synthèse.

Il est souhaitable qu'au terme d'une étape de conception générale, on ait déterminé, comme dans le cycle XP, un nombre limité de cycles de livraison pouvant faire l'objet d'une mise en œuvre incrémentale. L'apprentissage des nouvelles technologies (objet et multimédia) doit commencer dès le début du projet.

On peut donc proposer le découpage structurel suivant :

- module 1 : gestion administrative d'une intervention ;
- module 2 : gestion des données techniques (nomenclatures et plans) ;
- module 3 : statistiques.

Le module maintenance préventive doit s'appuyer sur des résultats statistiques, il est donc souhaitable de différer son développement. Il fera partie d'une deuxième version de l'application.

Cela nous donne l'ébauche de plan de la figure 9.3.

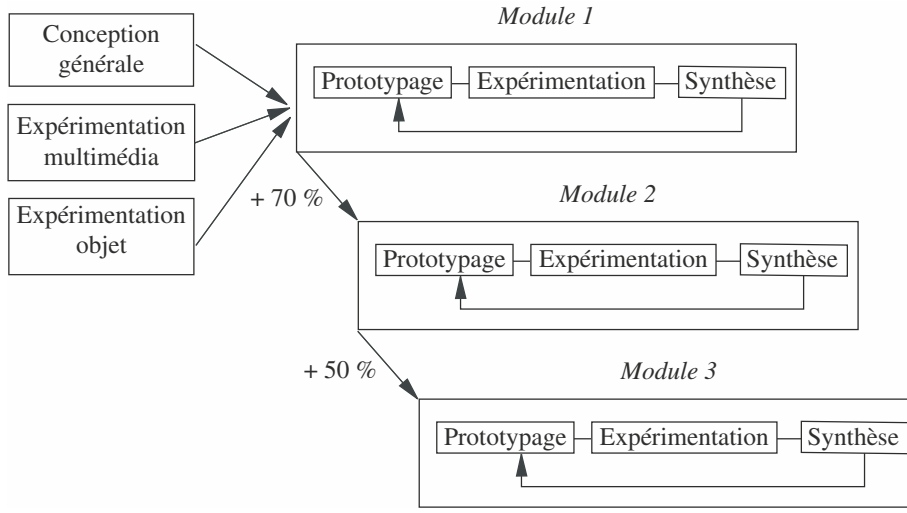


Figure 9.3 — Plan de projet du projet Mécano.

La phase d'exploration des besoins inclut l'élaboration de deux prototypes, dont l'objectif est double. Elle offre aux commanditaires une première concrétisation du futur système sur laquelle ils peuvent se prononcer. Elle permet que les développeurs prennent en main des technologies avec lesquelles ils sont peu familiers. Ensuite, on retient trois itérations de livraison, chacune aboutissant à un module qui peut déjà être mis en exploitation. Les itérations de livraison sont composées d'itérations de développement, avec la participation active des utilisateurs du groupe témoin.

Pour réduire le délai, on peut commencer le développement du module 2 quand on a réalisé 70 % du module 1. Le module 3 peut démarrer quand on a réalisé 50 % du module 2.

9.2 LE CAS KALIZEAU

Le cas Kalizeau va permettre d'illustrer l'analyse des parties prenantes, l'identification des alternatives, le découpage structurel, l'analyse de risque et le repérage des types de réponse.

9.2.1 Présentation du projet Kalizeau

L'état de Valbourg[1], nouvellement intégré dans l'Europe, vient de terminer un schéma directeur qui définit les principes de gestion de l'état écologique des cours d'eau situés sur son territoire. En particulier, pour s'aligner sur les exigences européennes, il a été décidé que la qualité des eaux devra être mesurée systématiquement, lors de campagnes à intervalles réguliers. Pour chaque cours d'eau, on a sélectionné des tronçons — points de recueil — sur lesquels se feront les prélèvements. L'analyse des échantillons recueillis permettront de connaître :

1. la composition chimique de l'eau ;
2. la quantité et l'état de santé des poissons recueillis, excellents indicateurs de la qualité de l'eau.

Pour mettre en œuvre ce schéma directeur, il est nécessaire de construire un système d'information permettant de gérer les informations issues des prélèvements. Le projet Kalizeau a donc été lancé.

1. Ce cas, lointainement inspiré d'un projet réel, a été largement remanié à des fins pédagogiques.

Les fiches suivantes sont données à titre indicatif dans l'annexe du schéma directeur.

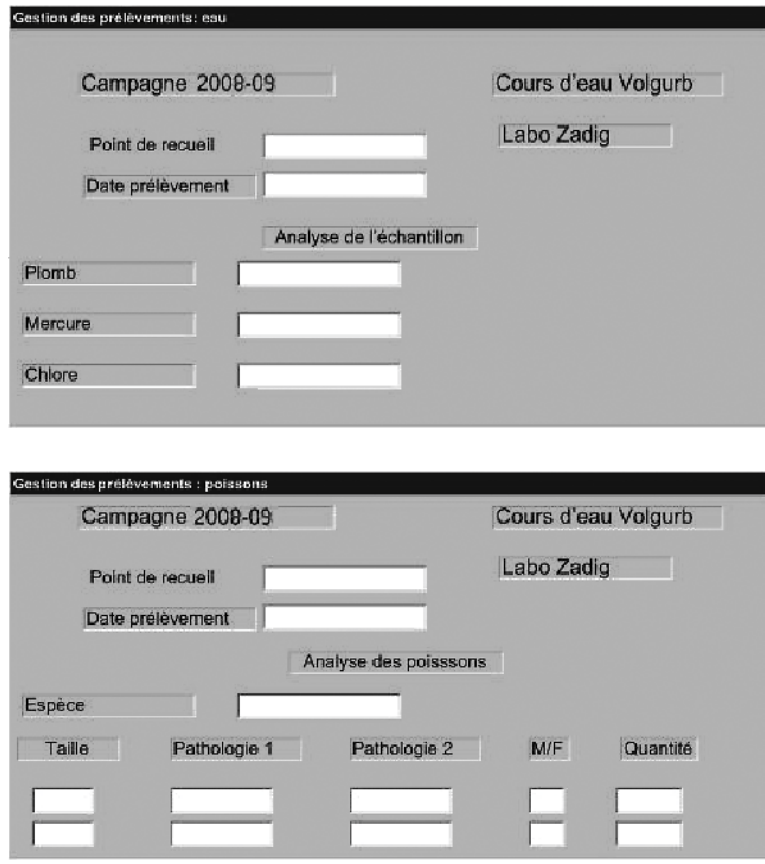


Figure 9.4

9.2.2 Énoncé de l'analyse des parties prenantes du cas Kalizeau

Le projet est mené sous la responsabilité du service Qualité au sein du ministère du Développement Durable (MDD). Deux responsables ont été nommés, M^me^ Olga, chimiste, et M. Bernard, biologiste. Le Ministère de la Pêche est également impliqué, en la personne de M. Pedro. La maîtrise d'œuvre est confiée à la Direction informatique du MDD, placée sous la responsabilité de M. Blake. Les différents avis ont été recueillis.

M^me^ *Olga* : « J'attendais depuis longtemps que l'on s'occupe de la qualité des eaux. Le projet Kalizeau va nous permettre, par la gestion organisée des informations, de vraiment progresser, car nous saurons exactement où nous en sommes. »

M. *Bernard* : « C'est un projet qui rentre tout à fait dans les préoccupations de notre Ministère. Il faudra concevoir un système techniquement à la hauteur. On ne doit pas être à la traîne. Notamment, toutes les informations seront transmises par voie électronique, directement par les laboratoires d'analyse chargés des prélèvements. »

M. *Pedro* : « Ce projet va permettre à notre beau pays de développer une activité touristique autour des zones aquatiques. L'Association des Pêcheurs en Valbourg (APV) avait contacté mes services, à plusieurs reprises, pour me signaler des problèmes sanitaires touchant certaines espèces. Bon, eux ils sont partants, mais tout n'est pas aussi simple. Le syndicat des entreprises chimiques (SEC), nombreuses dans notre pays, s'est manifesté. Il faudra faire très attention aux données que l'on publiera, pour que cela ne fasse pas trop de vagues. »

M. *Blake* : « C'est un projet intéressant, cela nous permettra de mettre en œuvre les technologies internet. Mais mes équipes sont encore pour beaucoup dédiées aux systèmes anciens. Il va falloir que je monte un plan de formation solide pour mettre à niveau mes développeurs. Comme je ne suis là que depuis l'été, je n'ai pas droit à l'erreur ! »

Faire l'analyse de chacune des parties prenantes, en analysant :

- ses relations avec les autres acteurs ;
- ce qu'il/elle peut gagner ou perdre avec le projet (enjeux) ;
- ses possibilités d'action sur le projet ;
- sa stratégie probable.

9.2.3 Corrigé de l'analyse des parties prenantes du cas Kalizeau

Le tableau de la figure 9.5 résume la position des principales parties prenantes.

9.2.4 Énoncé de la structure de découpage de projet Kalizeau

En première approche, on a identifié les processus suivants :

- Gestion des données sur l'empoissonnement.
- Gestion des analyses de l'eau.
- Étude des évolutions.
- Mise à disposition des informations.

	M^me Olga	**M. Bernard**	**M. Pedro**	**M. Blake**	**Féd. labos**	**APV**	**SEC**
relations	concurrence M. Bernard	concurrence M^me Olga	politique	recherche légitimité	fournisseur	pression sur ministères	pression sur ministères
enjeux	idéologique	intérêt technologique	craint vagues	carrière	activité accrue, mais incertitude technique	associatif	économiques
action	engagement dans le projet décisions fonctionnelles	engagement dans le projet décisions fonctionnelles	décisions fonctionnelles	décision d'ordre technique	coopération ou désintérêt	campagne publique	lobbying politique
stratégie	positive poussera le projet	globalement positif techniquement maximaliste	favorable, mais prudent frein si redoute réactions	suivra le projet avec attention techniquement prudent	favorable coopération	favorable si transparence	

Figure 9.5 — Les parties prenantes dans le projet Kalizeau.

On prévoit quatre référentiels différents pour gérer les informations permanentes de Kalizeau. Il s'agit des informations sur les cours d'eau, sur les points de recueil, sur les composants chimiques de l'eau et sur les catégories de poissons.

On envisage le découpage en phases suivant :

- Analyse globale
- Conception technique et élaboration d'un prototype de saisie des prélèvements
- Développement modulaire
- Mise en œuvre en deux temps, d'abord au MDD et dans un laboratoire pilote, puis généralisation à l'ensemble des laboratoires concernés.

Proposer deux structures de découpage de projet, selon que l'on privilégie le découpage structurel ou le découpage temporel.

9.2.5 Corrigé de la structure de découpage du projet Kalizeau

Le découpage selon l'axe composant se présente comme à la figure 9.6.

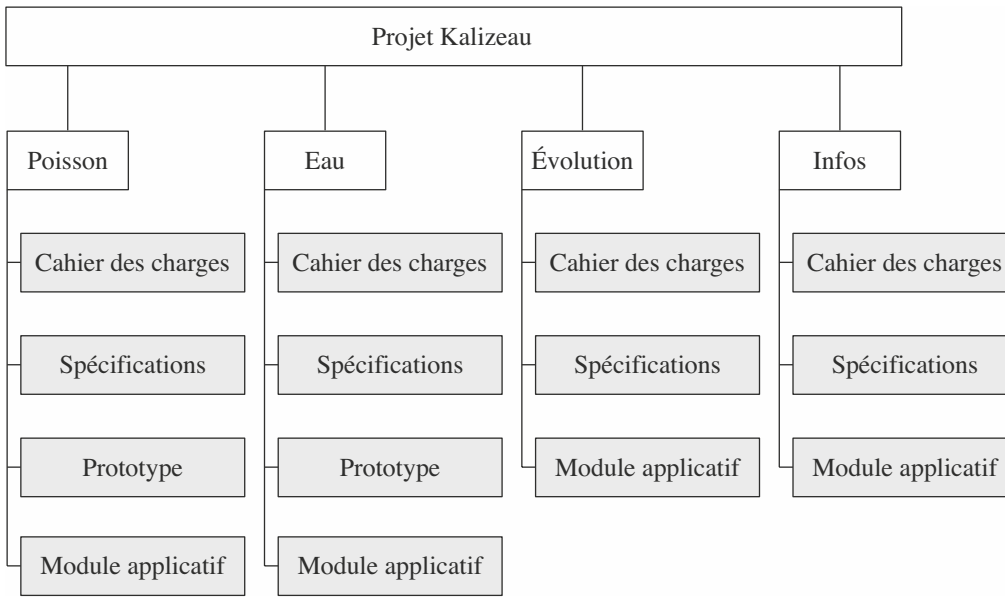


Figure 9.6 — Découpage du projet Kalizeau selon l'axe composant.

Le découpage selon l'axe cycle de vie se présente comme à la figure 9.7.

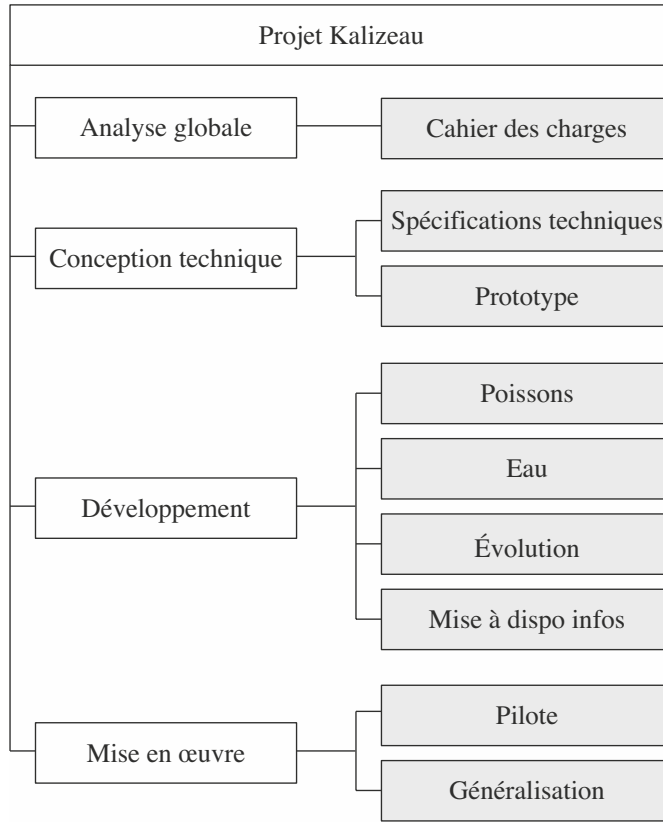


Figure 9.7 — Découpage du projet Kalizeau selon l'axe cycle de vie.

9.2.6 Énoncé de l'analyse de risque du projet Kalizeau

On souhaite mettre en œuvre le système Kalizeau d'ici un an. L'environnement technique retenu utilisera au maximum les standards du marché : protocole de réseau TCP/IP, serveur sous NT, base de données Oracle, langages de programmation Visual Basic et Java. Selon les estimations effectuées, la charge de travail totale prévue est de 500 jours-personne.

Les bases de données CoursEau et PointsRecueil sont gérées sur un serveur au niveau national. Les référentiels des Poissons et des Composants chimiques sont mis à disposition sous forme électronique par les instances européennes ; la structure et le contenu sont stables, et si jamais des évolutions devaient apparaître (nouveau composant chimique ou nouveau poisson), la Communauté européenne mettra un nouveau contenu à disposition. Le récapitulatif de chaque campagne est fourni, sous forme électronique, au Ministère de la Pêche et à la Communauté européenne.

Rappelons que les responsabilités de conduite du projet ont été réparties comme suit.

Le projet est mené sous la responsabilité du Service Qualité au sein du ministère du Développement Durable (MDD), en liaison avec le ministère de la Pêche. Deux responsables ont été nommés par le MDD, Mme Olga, chimiste, et M. Bernard, biologiste, qui ont chacun un point de vue sur le projet. De plus, la vision du ministère de la Pêche est plus politique que celle du MDD. Un représentant des laboratoires d'analyse fait également partie du Comité de pilotage.

La maîtrise d'œuvre est confiée à la Direction Informatique du MDD, placée sous la responsabilité de M.Blake. Les compétences techniques de la Direction Informatique relèvent en grande partie des systèmes informatiques de type AS400, et peu sont familiers des technologies internet. C'est pourquoi le développement sera sous-traité à une SSII. En revanche, plusieurs assistants à maîtrise d'ouvrage ont une bonne connaissance des métiers. La saisie des prélèvements se fera directement dans les laboratoires partenaires (une dizaine). Il faudra prévoir une formation sur la partie précise de la saisie des résultats, car leur familiarité avec internet est limitée.

Faire l'analyse de risques et dresser le profil du projet.

9.2.7 Corrigé de l'analyse de risque du projet Kalizeau

L'analyse, basée sur une grille inspirée de celle de l'annexe C, se présente ainsi.

Taille du projet = 2

Critère	Degré	Métrique	
Durée	1	=< 6 mois	
	2	=< 18 mois	2
	3	=< 30 mois	
	4	> 30 mois	
Charge	1	=< 20 mois-personne	
	2	=< 120 mois-personne	2
	3	=< 300 mois-personne	
	4	> 300 mois-personne	

Difficulté technique = 1

Critère	Degré	Métrique	
Expérience du marché sur l'architecture ciblée	1	Plus de 200 références	1
	2	Plus de 100 références	
	3	Plus de 50 références	
	4	Moins de 10 références	
Expérience du marché sur le langage de programmation	1	Plus de 200 références	1
	2	Plus de 100 références	
	3	Plus de 50 références	
	4	Moins de 10 références	
Expérience du marché sur le SGBD	1	Plus de 200 références	1
	2	Plus de 100 références	
	3	Plus de 50 références	
	4	Moins de 10 références	
Contrainte de performance	1	Normale	1
	2	Un peu élevée	
	3	Élevée	
	4	Très élevée	

Degré d'intégration = 1,5

Critère	Degré	Métrique	
Nombre de flux (type) inter applicatifs	1	=< 1	
	2	=< 5	2
	3	=< 10	
	4	> 10	
Nombre d'applications connexes en chantier	1	Aucune	1
	2	= 1	
	3	= 2	
	4	> 2	

Configuration organisationnelle = 4

Critère	Degré	Métrique	
Nombre de personnes (de point de vue différent) représentant la maîtrise d'ouvrage	1	= 1	
	2	= 2	
	3	= 3	
	4	> 3	4
Nomination d'un promoteur	1	Nommé, impliqué, légitime	
	2	Nommé, impliqué	
	3	Nommé	
	4	Pas de promoteur	4
Appui de la Direction générale	1	Suivi par la DG	
	2	DG informée	
	3	Pas d'implication de la DG, mais une seule MOA	
	4	Pas d'implication de la DG et plusieurs MOA	4

Changement = 2,75

Critère	Degré	Métrique	
Degré d'évolution organisationnelle	1	Pas d'écart	
	2	Écart faible	2
	3	Écart moyen	
	4	Écart fort	
Degré d'évolution fonctionnelle	1	Pas d'écart	
	2	Écart faible	
	3	Écart moyen	3
	4	Écart fort	

Critère	Degré	Métrique	
Degré d'évolution technique	1	Pas d'écart	
	2	Écart faible	
	3	Écart moyen	3
	4	Écart fort	
Nombre de sites concernés	1	1	
	2	Moins de 10	
	3	Moins de 50	3
	4	Plus de 50	

Équipe = 1,3

Critère	Degré	Métrique	
Compétences techniques	1	Bonnes	1
	2	Moyennes	
	3	Faibles	
	4	Très faibles	
Compétences fonctionnelles	1	Bonnes	1
	2	Moyennes	
	3	Faibles	
	4	Très faibles	
Image du projet	1	Très valorisante	
	2	Valorisante	2
	3	Moyennement valorisante	
	4	Faiblement valorisante	

Le profil du projet peut être ainsi visualisé comme à la figure 9.8.

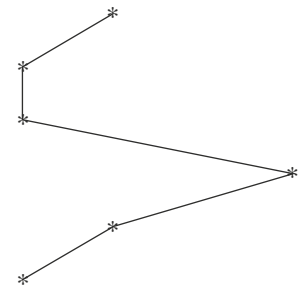
Nature du risque	Degré de risque 1 2 3 4
Taille	
Difficulté technique	
Degré d'intégration	
Configuration organisationnelle	
Changement	
Équipe de projet	

Figure 9.8 — Profil du projet Kalizeau

9.2.8 Énoncé des réponses aux risques du projet Kalizeau

Après analyse des risques, on a conclu que le projet Kalizeau doit faire face à une menace de dépassement de délai pour les raisons suivantes. On pourra rencontrer des difficultés pour faire converger les maîtres d'ouvrage sur les fonctionnalités nécessaires. Les opérationnels du MDD manifestent des inquiétudes face à la future gestion des campagnes. Le manque de technicité des développeurs du MDD pourra poser problème, dans la mesure où l'on a finalement décidé de conserver en interne le développement de la mise à disposition des informations.

Déterminer à quel type de réponse au risque se rattache chacune des mesures ci-dessous, en considérant que les mesures sont indépendantes entre elles.

1. Acheter le système proposé par la communauté européenne qui comporte les fonctionnalités de base.
2. Négocier un délai supplémentaire d'un mois.
3. Proposer d'élaborer l'expression des besoins sous forme d'une session JRP.
4. Obtenir que le Ministère de la Pêche exprime par écrit ses exigences.
5. Attendre que les MOA tombent d'accord.
6. Créer une cellule de préparation au changement.
7. Embaucher un spécialiste Java et adopter un modèle en W.
8. Obtenir la nomination du Directeur du Service Qualité comme promoteur du projet.
9. Si les fonctionnalités ne sont pas validées au 15 novembre, demander une réunion d'urgence du Comité de pilotage.
10. Adopter une approche de développement de l'application par version.

9.2.9 Corrigé des réponses aux risques du projet Kalizeau

La première mesure peut être envisagée de deux façons. On peut considérer qu'acheter un système existant va faire disparaître le problème de la détermination des besoins. Il s'agit donc d'un évitement. Si toutefois, on envisage d'enrichir les fonctionnalités de base, il s'agit d'une atténuation.

La deuxième mesure est une acceptation active si l'on ne modifie pas l'échéancier et que le délai supplémentaire est gardé comme un tampon de réserve, utilisé uniquement si on dépasse la date de fin planifiée. Si, au contraire, on refait la planification et que l'on recule la date de fin visée d'un mois, il s'agit d'une atténuation, puisque l'on a introduit des marges dans l'ensemble du projet. La première disposition est celle qui est recommandée par la méthode de la chaîne critique (cf. § 4.6.2).

Le travail en session sur les besoins, qui est planifié dans la troisième mesure, devrait permettre, dans la mesure où il n'y a pas d'antagonisme majeur entre les décideurs, d'atténuer voire d'éviter le risque.

La quatrième disposition correspond à un transfert du risque lié à l'expression des exigences provenant du Ministère de la Pêche. Celles-ci ne pourront être prises en compte que dans la mesure où elles sont formalisées.

Attendre qu'un consensus se fasse entre les décideurs correspond à une acceptation passive.

La création d'une cellule de préparation au changement peut être plus ou moins efficace. Selon les espérances que l'on a, il s'agit d'un évitement si l'on pense que le risque va disparaître, ou d'une atténuation s'il semble que le risque n'a pas disparu.

La septième mesure devrait permettre de former les développeurs, grâce à l'accompagnement d'un spécialiste et la construction d'un prototype. Selon la confiance que l'on a dans la capacité et la rapidité d'apprentissage des développeurs, il s'agit d'un évitement ou d'une atténuation.

Si le Directeur du Service Qualité est nommé promoteur du projet, il aura une responsabilité particulière dans le projet, ce qui atténuera les risques de non convergence sur les fonctions du futur système.

La neuvième mesure ne sera mise en œuvre que si l'on constate l'absence de validation, il s'agit donc d'une réponse conditionnelle.

La dernière mesure qui s'inscrit dans une perspective itérative, avec une implémentation à chaque itération de livraison, devrait permettre d'atténuer le risque, aussi bien en ce qui concerne les besoins qu'en ce qui concerne la mise à niveau des développeurs.

10

La pratique de l'estimation des charges

10.1 LE CAS PARKING

Nous allons estimer la charge du projet décrit ci-dessous (paragraphe 10.1.1) et utiliser pour ce faire les différentes méthodes d'estimation.

Nous simulerons d'abord la méthode Delphi pour confronter le jugement des experts, en supposant les expériences indiquées ci-après (paragraphe 10.1.2).

Puis, nous utiliserons la méthode de répartition proportionnelle du cycle de développement classique.

Nous appliquerons le modèle Cocomo en prenant une valeur de la taille du logiciel de 2 400 lignes.

Nous estimerons la charge de réalisation par la méthode d'évaluation analytique en nous basant sur l'identification des programmes, les ratios et poids standard donnés ci-dessous (paragraphe 10.1.3).

Nous utiliserons la méthode des points fonctionnels en faisant des hypothèses sur les fonctionnalités.

Nous illustrerons enfin la démarche générale d'estimation et nous terminerons en comparant les différentes estimations.

10.1.1 Description du projet Parking

Nous sommes sur le centre de production d'un constructeur automobile. Deux chaînes effectuent le montage. Les véhicules sont ensuite transportés chez les

distributeurs par un service Livraison/logistique. Les employés sont répartis dans des bâtiments parfois éloignés les uns des autres. On veut gérer l'accès aux différents parkings.

On définit, pour chaque parking, les bâtiments qui sont accessibles à partir de ce parking. L'attribution des places de parking se fera en fonction du lieu d'affectation de l'employé. L'attribution dépend également de la marque du véhicule : certains parkings sont interdits aux véhicules de marques concurrentes.

Les employés peuvent obtenir des autorisations exceptionnelles de parking, par exemple s'ils participent à une réunion dans un autre bâtiment que leur bâtiment habituel.

L'organigramme simplifié du centre de production est donné à la figure 10.1. C'est le service Divers qui gérera l'attribution des parkings.

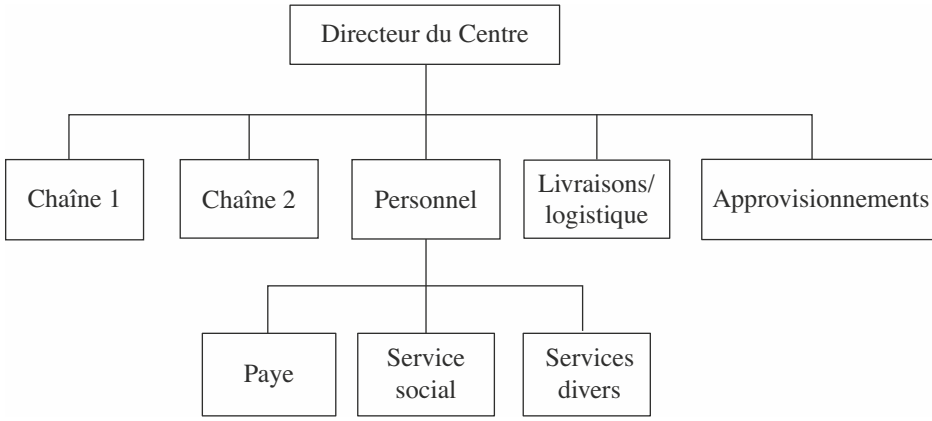


Figure 10.1 — Organigramme du cas Parking.

10.1.2 Bases d'expériences des experts

Supposons quatre experts A, B, C et D, ayant chacun sa base d'expériences constituée des projets précédemment menés. Certains projets ont été menés en commun par plusieurs d'entre eux. Analysons d'abord les expériences.

La base d'expériences de l'expert A comprend trois projets, nommés D04, K67 et RESA.

Le projet D04 « Description des commandes de véhicules » traite de la prise de commande, avec les options et couleurs choisies. Un catalogue devait renseigner sur les choix autorisés. Le projet a duré quatre mois, mais pendant deux mois et demi un concepteur et un stagiaire sont venus renforcer A.

Le projet K67 « Gestion des concours » est un projet commun à plusieurs écoles de gestion. Le système comprenait les fonctions suivantes : enregistrer toutes les inscriptions, puis les notes obtenues aux différentes épreuves écrites, calculer la moyenne (avec des coefficients de pondération variables). Selon les écoles, le système produisait : soit deux listes « admis à l'oral » et « refusés » ; soit trois listes « grands admissibles », « petits admissibles » et « refusés ». Les admissibles passent ensuite un entretien de motivation, les « petits admissibles » ayant en plus des oraux sur différentes matières présentes à l'écrit. Le système enregistrait les notes d'oral et produisait alors une liste définitive. Une école située à Evry avait des règles particulièrement subtiles qui n'ont pas facilité la tâche. A a commencé en septembre, avec deux autres concepteurs, pensant avoir fini fin janvier. Les deux autres concepteurs ont terminé leur partie mi-décembre, mais A a dû poursuivre jusqu'à fin février.

Le projet RESA « Gestion des réservations de chambres d'hôtel » a permis à A de passer deux mois à La Clusaz, avec un de ses collègues, puisque le client était le syndicat d'initiative de cette station. La réservation est faite de façon centralisée par le syndicat pour tous les hôtels de la station. De plus, A a travaillé seul à Paris pendant trois mois.

La base d'expériences de l'expert B comprend également trois projets : IA55, SCI8 et BIB1.

Le projet IA55 « Gestion des vols de l'INT-Airlines » a été mené deux ans auparavant. Cette compagnie aérienne gère des parcours-types, identifiés par des codes IATA (par exemple Paris-Varsovie). Un parcours-type, selon sa distance et la fréquentation de la destination, peut être effectué par un ou plusieurs types d'avion (par exemple Airbus A320). Chaque vol effectué par la Compagnie est caractérisé par l'affectation d'un avion (repéré par son numéro d'immatriculation) à un parcours-type pour une date donnée. Le projet a permis de gérer l'ensemble de ces données. B a dû développer des interfaces avec deux autres applications : la maintenance des avions et la base des aéroports dans le monde. B était aidé par une seconde personne sur ce projet qui a duré trois mois.

Le projet SCI8 « Gestion des conférences scientifiques » avait pour objectif de faciliter l'organisation des conférences par une société scientifique. Chaque conférence est préparée par un comité d'organisation et un comité de programme, dont les présidents sont des adhérents. Après appel à communication, des articles sont proposés par des chercheurs. Lorsqu'un article arrive, le président du comité de programme l'attribue à des lecteurs, choisis parmi les membres du comité de programme, qui lui donnent une note. Selon la barre fixée, les articles peuvent être sélectionnés, refusés ou faire l'objet d'une délibération par le comité de programme. Le système après enregistrement des notes directement par les jurés sur un serveur minitel, produit les lettres de réponse.

B a mis cinq mois pour mettre en place ce nouveau système. La société disposait déjà d'un serveur utilisé par ses adhérents.

Le projet BIB1 « Gestion des prêts », destiné à la bibliothèque de son ancienne école, a rappelé à B de bons souvenirs. Il pensait n'en avoir que pour trois mois. Cependant, les règles de gestion variaient selon le type d'emprunteur : élèves en formation initiale, participants à des modules de formation continue, mastères, professeurs permanents, professeurs vacataires, professeurs invités ou sabbatiques, thésards. De plus il a fallu établir une liaison avec la base de données livres. Ces deux éléments ont fait qu'il y a finalement passé cinq mois.

La base d'expériences de l'expert C comprend les projets APP et ASS8.

Le projet APP « Gestion des approvisionnements » visait à faciliter le suivi des campagnes d'approvisionnement de DISTRINT. Périodiquement, l'entreprise lance des campagnes d'approvisionnement ou de réapprovisionnement : enregistrement des commandes, suivi des réceptions, rapprochement avec les factures. C avait développé un petit système expert pour proposer un plan de réapprovisionnement pour les articles peu soumis aux modes, prenant en compte la saison et les consommations antérieures (d'un article et des articles de substitution). Le projet intégrait un suivi des fournisseurs prenant en compte la qualité des marchandises livrées et l'évolution de leur prix. Le projet a duré un an. Après une étude préalable de deux mois, il a été rejoint par un collègue qui l'a aidé à élaborer un prototype et à le faire évoluer vers le système finalement implanté.

Le projet ASS8 « Gestion des polices d'assurance automobile » couvrait les fonctions classiques d'une compagnie d'assurance : élaboration d'un devis, suivi d'une proposition fixant les termes du contrat ; transformation de la proposition en contrat après le versement de la prime, émission périodique des quittances, élaboration d'avenants et résiliation. C y a travaillé avec deux collègues pendant trois mois, puis a été remplacé par une équipe de quatre personnes pendant six mois.

La base d'expériences de l'expert D comprend les projets K67, APP et ASS8.

D a mené le projet K67 « Gestion des concours » jusqu'à mi-décembre, il y a fait la connaissance de l'expert A.

D a rejoint l'expert C sur le projet APP « Gestion des approvisionnements » après l'étude préalable.

D a travaillé les six derniers mois sur le projet ASS8 « Gestion des polices d'assurance automobile ».

10.1.3 Éléments pour l'évaluation analytique

Pour l'étape de réalisation, nous avons identifié quatre lots homogènes et indépendants.

1. Le lot *Parking* comprend un complément des bases de données bâtiment et personnel, avec l'affectation des salariés, ainsi qu'un complément de la base parking, avec les proximités et les interdictions liées aux marques. Ce lot est composé de 10 programmes temps réel de difficulté moyenne.
2. Le lot *Véhicule* comprend le répertoire des marques et la description des véhicules. Il est composé de 5 programmes temps réel de difficulté moyenne.
3. Le lot *Autorisation* comprend les demandes d'autorisation habituelle et exceptionnelle. Il est composé de 5 programmes temps réel difficiles.
4. Le lot *Édition* comprend différentes listes croisées entre Véhicules, Salariés et Autorisations. Il est composé de 10 programmes batch faciles.

Chaque lot comprend quatre types de tâches : élaboration d'un jeu d'essai, étude technique, programmation et test qualité. Après élaboration des lots, on prévoit une tâche d'intégration.

La méthode d'évaluation analytique est basée sur des poids standard et des ratios. Les unités d'œuvre sont les programmes, les ratios permettant d'estimer la charge des autres tâches s'appliquent à la charge de programmation.

La tâche d'intégration pèse entre 10 % et 15 % de la charge de programmation des quatre lots.

Les poids standard affectés aux différentes unités d'œuvre et mesurés en jours-personne sont donnés à la figure 10.2.

Type de programme	Facile	Moyen	Difficile
Temps réel	2	3	5
Batch	1,5	2,5	3,5

Figure 10.2 — Poids standard pour l'évaluation analytique du cas Parking.

Les ratios utilisés sont donnés à la figure 10.3.

On décide de faire quelques ajustements. Les tests du lot Parking devraient être assez rapides, de même que l'étude technique du lot Autorisation. On retire respectivement 3 et 2 points au ratio permettant d'obtenir leur charge. L'étude technique du lot Véhicule devrait être plus difficile : on décide de rajouter 3 points au ratio.

Type de tâche	% de la charge de programmation
Jeu d'essai	20
Étude technique	10
Tests	10

Figure 10.3 — Ratios pour l'évaluation analytique du cas Parking.

On convient d'élaborer un jeu d'essai unique pour les lots Véhicule et Autorisations.

10.2 CORRIGÉ DU CAS PARKING AVEC LA MÉTHODE DELPHI

L'exercice peut être réalisé en donnant à quatre personnes (ou quatre groupes) le rôle des experts. Chacun doit reconstituer, si nécessaire, la charge des projets de référence.

L'expertise de A s'appuie sur des projets dont la charge est comprise dans un intervalle [7 ; 13] mois-personne. En effet :

D04 : 9 mois-personne ;

K67 : 13 mois-personne ;

RESA : 7 mois-personne.

L'expertise de B s'appuie sur des projets dont la charge est comprise dans un intervalle [5 ; 6] mois-personne. En effet :

IA55 : 6 mois-personne ;

SC18 : 5 mois-personne ;

BIB1 : 5 mois-personne.

L'expertise de C s'appuie sur des projets dont la charge est comprise dans un intervalle [22 ; 33] mois-personne. En effet :

APP : 22 mois-personne ;

ASS8 : 33 mois-personne.

L'expertise de D s'appuie sur des projets dont la charge est comprise dans un intervalle [13 ; 33] mois-personne.

Il s'agit donc pour chacun de comparer la représentation qu'il se fait du domaine Parking avec les projets antérieurement menés.

On observe souvent que ceux qui ont des petits projets en référence (expert A ou expert B) donnent une première évaluation inférieure à celle de ceux qui ont connu des projets plus importants (expert C ou expert D). Ces derniers projettent sur le projet une richesse ou une complexité fonctionnelle qu'ils ont déjà rencontrée.

Le premier tour peut donner des estimations allant parfois du simple au quadruple, par exemple 4 mois-personne pour certains et 16 mois-personne pour d'autres. En général, l'estimation se rapproche ensuite avec une concentration autour de 8 mois-personne.

10.3 CORRIGÉ DU CAS PARKING AVEC LA MÉTHODE DE RÉPARTITION PROPORTIONNELLE

Nous allons estimer la charge de l'étude préalable (par approche paramétrique simplifiée) et nous en déduirons celle du projet complet.

Pour cela, il faut ébaucher le plan d'interviews nécessaire pour mener l'étude et affecter des poids à chaque interview. Ces poids (entre 0,5 et 3 jours) ne sont pas des durées d'interview, mais comprennent le travail de modélisation, synthèse et diagnostic qui sera fait avec la matière recueillie.

On peut proposer de rencontrer les personnes suivantes :

- Le directeur donnera ses orientations et ses ambitions en ce qui concerne le futur système. Envisage-t-il de l'étendre aux visiteurs ? De le proposer à d'autres usines du groupe ? On affecte à cette interview un poids de 1 jour.
- Le service Divers est le futur gestionnaire : on rencontrera les trois personnes. On affecte un poids de 1 jour à chacun.
- Le responsable du personnel est à l'origine des règles de gestion, il nous renseignera sur la mobilité des employés… On affecte un poids de 2 jours à cette interview.
- On rencontrera également un représentant du service Logistique, le responsable des approvisionnements, ainsi qu'un cadre d'une des deux chaînes de production. On affecte un poids de 0,5 jour à chacun.

La charge de l'étape observation de l'étude préalable se répartit comme indiqué à la figure 10.4.

La charge de l'étape observation (7,5 jours-personne) représente environ un tiers de la charge de l'étude préalable. On estime donc celle-ci à :

22 jours-personne.

Interview	Poids
Directeur	1
Service Divers	3
Responsable personnel	2
Logistique	0,5
Approvisionnements	0,5
Chaîne de production	0,5
Total	7,5

Figure 10.4 — Charge de l'étape observation du cas Parking.

Si l'on considère que la charge de l'étude préalable représente 10 % du total, on obtient :

charge du projet = 220 jours-personne.

Ces 11 mois-personne se répartissent d'après les ratios de distribution dans le cycle de développement. Le résultat est donné à la figure 10.5.

Phase	Charge
Étude préalable	22 jours-personne
Étude détaillée	65 jours-personne
Étude technique	3 jours-personne
Réalisation	130 jours-personne
Total	220 jours-personne

Figure 10.5 — Répartition de la charge globale du cas Parking.

10.4 CORRIGÉ DU CAS PARKING AVEC LE MODÈLE COCOMO

Supposons qu'un informaticien ait évalué la taille du logiciel à 4 000 lignes, dans un langage de 3e génération.

On applique la formule pour les petits projets :

Charge = $2{,}4 \times 4^{1{,}05}$

Charge = $2{,}4 \times 4{,}28$

Charge = 10,3 mois-personne

10.5 CORRIGÉ DU CAS PARKING AVEC LA MÉTHODE D'ÉVALUATION ANALYTIQUE

Nous appliquons, lot par lot, les poids standard de chaque type de programme, et nous appliquons à la charge de programmation ainsi obtenue les ratios des tâches périphériques. L'évaluation analytique, arrondie à la demi-journée près, figure dans le tableau de la figure 10.6.

Cela nous donne la charge de l'étape de réalisation, incluant les tâches d'étude technique. Nous utilisons la méthode de répartition proportionnelle pour extrapoler la charge du projet complet (figure 10.7).

Nous obtenons : 10,6 mois-personne

Lot	Type de tâche	Nombre d'unités	Poids standard ou ratio	Charge estimée en jours-personne
Parking	Programme en temps réel moyen	10	3 jours-personne	30
	Étude technique		10 %	3
	Jeu essai		20 %	6
	Tests		7 %	2
Total lot Parking				**41 j/p**
Véhicule	Programme en temps réel moyen	5	3 jours-personne	15
	Étude technique		13 %	2
	Tests		10 %	1,5
Total lot Véhicule				**18,5 j/p**
Autorisation	Programme en temps réel difficile	5	5 jours-personne	25
	Étude technique		8 %	2
	Jeu essai		20 %	8
	Tests		10 %	2,5
Total lot Autorisation				**37,5 j/p**

Figure 10.6 — Évaluation analytique du cas Parking.

Édition	Programme batch facile	10	1,5 jours-personne	15
	Étude technique		10 %	1,5
	Jeu essai		20 %	3
	Tests		10 %	1,5
Total lot Édition				**21 j/p**
Total programmation				**85 j/p**
Intégration				10
TOTAL				**128 j/p**

Figure 10.6 — Évaluation analytique du cas Parking. *(Suite)*

Étape	Règle de calcul	Charge
Charge réalisation		128 jours-personne
Charge étude détaillée	La moitié de la charge de réalisation	64 jours-personne
Charge étude préalable	Entre la moitié et le tiers de la charge d'étude détaillée : on prend le tiers.	21 jours-personne
Total		**213 jours-personne,** soit **10,6 mois-personne**

Figure 10.7 — Extrapolation de la charge globale du cas Parking.

10.6 CORRIGÉ DU CAS PARKING AVEC LA MÉTHODE DES POINTS FONCTIONNELS

La démarche consiste à identifier et dénombrer tous les composants fonctionnels du futur système, en les classant par type d'unité d'œuvre et degré de complexité. Pour cela, nous ferons certaines hypothèses sur le futur système, que nous expliciterons.

Nous commençons par structurer rapidement les données, puis nous ferons une projection des différents traitements, afin de remplir les deux colonnes manquantes du tableau standard de la figure 10.8.

Entité	Complexité	Nombre de composants	Poids	Nombre de points de fonction brut
GDI	Faible		**7**	
	Moyenne		**10**	
	Élevée		**15**	
GDE	Faible		**5**	
	Moyenne		**7**	
	Élevée		**10**	
ENT	Faible		**3**	
	Moyenne		**4**	
	Élevée		**6**	
SOR	Faible		**4**	
	Moyenne		**5**	
	Élevée		**7**	
INT	Faible		**3**	
	Moyenne		**4**	
	Élevée		**6**	
Total				

Figure 10.8 — Tableau de dénombrement des points de fonction.

10.6.1 Dénombrement des groupes de données référencées

Pour identifier les groupes de données (GDI et GDE), nous élaborons un modèle des données simplifié (figure 10.9).

La gestion des parkings suppose que l'on connaisse les parkings utilisables et pour chacun d'eux les éventuels interdits. Les bâtiments et leur situation par rapport aux parkings doivent être répertoriés. Pour pouvoir faire l'affectation, il faut connaître la localisation habituelle de chaque employé propriétaire d'un véhicule.

Pour les autorisations exceptionnelles, nous avons fait le choix de ne pas relier les demandes à un calendrier des réunions avec les salles correspondantes.

La pertinence d'une demande exceptionnelle reste donc en dehors du système informatisé.

Enfin, on a choisi de ne pas suivre l'utilisation réelle de la place attribuée.

On suppose que :

- chaque bâtiment est proche d'au moins un parking ;
- pour toute marque, il y a au moins un parking accessible ;
- un employé n'est basé à un moment donné que dans un seul bâtiment ;
- un seul propriétaire du véhicule est déclaré ;
- un employé ne peut pas déclarer plus de deux véhicules ;
- le système propose une ou plusieurs affectations, s'appuyant sur les règles en vigueur ;
- toute affectation doit être confirmée manuellement.

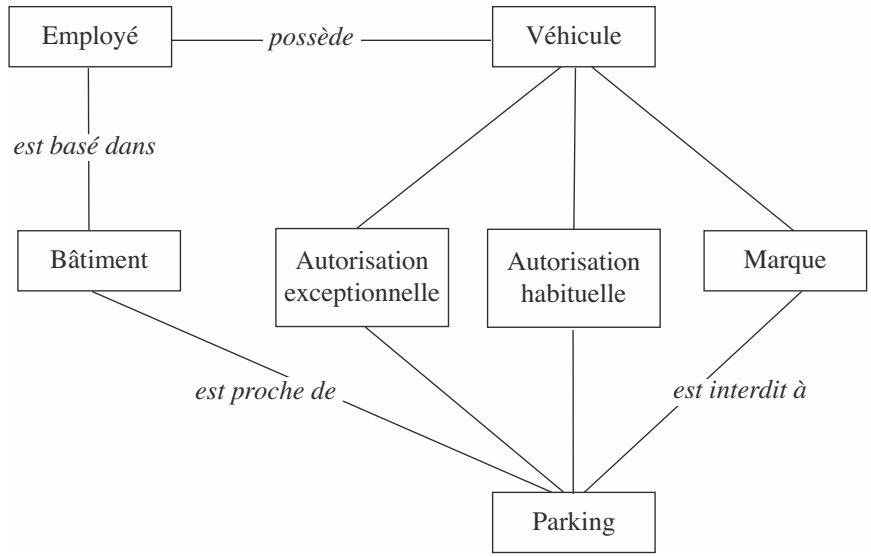


Figure 10.9 — Modèle des données du cas Parking.

Les groupes de données référencées sont :

- employé ;
- véhicule ;
- bâtiment ;

- parking ;
- marque ;
- autorisation habituelle ;
- autorisation exceptionnelle.

Les groupes de données externes (GDE) sont ceux qui sont déjà gérés par d'autres domaines. Il est clair que l'on n'a pas attendu la gestion des parkings pour constituer un référentiel *employé* : ce groupe de données est donc externe.

La question se pose pour les données *bâtiment* et *parking* : sont-elles déjà gérées sous forme informatisée par le service Logistique ? Nous faisons ici l'hypothèse qu'elles le sont.

On a donc la répartition donnée à la figure 10.10.

GDE	GDI
employé bâtiment parking	véhicule marque autorisation habituelle autorisation exceptionnelle

Figure 10.10 — GDR du cas Parking.

À part *employé*, tous les groupes de données, internes ou externes, sont de faible complexité. En effet, n'ayant pas de sous-type, ils n'ont qu'un seul sous-ensemble logique ; de plus le nombre de leurs propriétés élémentaires ne devrait pas dépasser 110.

On fait l'hypothèse vraisemblable que le groupe *employé* est d'une complexité moyenne.

On obtient donc pour les données le dénombrement de la figure 10.11.

Type	Complexité	Nombre
GDI	Faible	4
GDE	Faible	2
	Moyenne	1

Figure 10.11 — Dénombrement des données du cas Parking.

10.6.2 Dénombrement des entrées

Pour repérer les entrées de l'application (ENT), on s'appuie sur les données internes. Tous les GDI doivent faire l'objet d'au moins une entrée.

On peut donc dresser la liste suivante :

- un écran de saisie des marques, avec le cas échéant les parkings interdits ;
- un écran de saisie des véhicules, avec le propriétaire et la marque ;
- un écran d'affectation d'une autorisation habituelle ;
- un écran de saisie d'une demande exceptionnelle ;
- un écran d'affectation d'une autorisation exceptionnelle.

La complexité de ces entrées est donnée à la figure 10.12.

Libellé de l'entrée (ENT)	Données utilisées (GDI ou GDE)	Nombre de données élémentaires (DE)	Complexité
Saisie des marques	marque parking	moins de 15	moyenne
Saisie des véhicules	véhicule employé marque	moins de 5	moyenne
Autorisation habituelle	autorisation habituelle bâtiment parking	moins de 5	moyenne
Demande exceptionnelle	employé véhicule bâtiment	moins de 5	moyenne
Autorisation exceptionnelle	autorisation exceptionnelle parking	moins de 5	moyenne

Figure 10.12 — Dénombrement des entrées du cas Parking.

10.6.3 Dénombrement des sorties

Les sorties sont principalement des états de gestion permettant d'évaluer la pertinence des règles d'attribution et de les faire éventuellement évoluer. Il s'agit de statistiques sur l'occupation des parkings, les autorisations exceptionnelles…

Il est difficile d'aller plus loin avant de mener l'étude. Par précaution, on compte 5 sorties, dont 3 de faible complexité et 2 de complexité moyenne (figure 10.13).

Type	Complexité	Nombre
SOR	Faible	3
	Moyenne	2

Figure 10.13 — Dénombrement des sorties du cas Parking.

10.6.4 Dénombrement des interrogations

Chaque groupe de données interne doit pouvoir être consulté. On prévoit au minimum les listes (écran ou papier) suivantes :

- véhicule ;
- marque ;
- parkings avec et sans interdiction ;
- autorisation habituelle ;
- autorisation exceptionnelle.

Il faut y ajouter un certain nombre de listes croisées, telles que :

- véhicules d'une marque donnée ;
- véhicules affectés à un parking donné ;
- employés ayant reçu des autorisations exceptionnelles ;
- etc.

On compte donc une dizaine d'interrogations (INT), dont 2 de complexité moyenne et les autres de faible complexité (figure 10.14).

Type	Complexité	Nombre
INT	Faible	8
	Moyenne	2

Figure 10.14 — Dénombrement des interrogations du cas Parking.

10.6.5 Estimation de la charge

Le dénombrement complet, conduisant au nombre de points de fonction brut, est donné à la figure 10.15.

Entité	Complexité	Nombre de composants	Poids	Nombre de points de fonction brut
GDI	Faible	4	7	28
GDE	Faible	2	5	10
	Moyenne	1	7	7
ENT	Moyenne	5	4	20
SOR	Faible	3	4	12
	Moyenne	2	5	10
INT	Faible	8	3	24
	Moyenne	2	4	8
Total				119

Figure 10.15 — Estimation complète du nombre de points de fonction du cas Parking.

Nous ne faisons pas d'ajustement. La taille du logiciel est donc de 119 points de fonction. En prenant une valeur moyenne de 2 jours par point de fonction, on obtient :

charge du projet = 238 jours-personne

soit 11,9 mois-personne.

10.7 ÉNONCÉ DE LA DÉMARCHE GÉNÉRALE D'ESTIMATION

L'application informatique du projet Kalizeau, dont l'énoncé a été donné au paragraphe 9.2, va faire l'objet d'une mise en œuvre pilote. Dix personnes du ministère du Développement Durable (MDD) sont concernées, ainsi que cinq personnes d'un laboratoire partenaire du projet.

Un nouveau matériel va être installé au MDD pour l'occasion, soit un serveur et des postes pour tous les acteurs du ministère participant à cette première mise en œuvre. Le laboratoire prend en charge son propre équipement. L'installation du matériel comprend la réception du matériel et une vérification de son fonctionnement, ce qui prend environ un quart de journée pour chaque poste. La

charge d'installation du serveur a donné lieu à diverses évaluations, obtenues par la méthode Delphi. La valeur la plus fréquemment indiquée se situe autour de 1,5 jour, la plus basse étant de 0,5 jour et la plus élevée de 5,5 jours.

La recette se fera sur un jeu d'essai composé de cinq cas d'utilisation. Chaque cas mobilisera deux personnes pendant un quart de journée.

On prévoit de monter deux formations. La première concerne le MDD et le laboratoire, et présentera la saisie des prélèvements. La seconde portera sur les interrogations et les sorties et ne s'adresse qu'au MDD.

Le temps de préparation des formations, c'est-à-dire le choix du contenu et l'élaboration du support, est évalué en se basant sur les résultats de l'estimation du projet par la méthode des points de fonction. Il est proportionnel au nombre d'entrées (ENT) pour la première et au nombre de sorties et d'interrogations (SOR et INT) pour la seconde. Chaque composant fonctionnel nécessite soit 0,5 jour, 1 jour ou 2 jours de travail de préparation du support de formation selon son degré de complexité.

Le décompte en entités fonctionnelles est le suivant :

Type composant	Faible complexité	Complexité moyenne	Complexité élevée
ENT	3	2	2
SOR			2
INT			1

Figure 10.16 — Dénombrement des entités fonctionnelles du projet Kalizeau.

La première formation devra durer une journée, la seconde durera une demi-journée. Compte tenu du nombre de postes dans la salle de formation, les sessions comprendront au maximum dix personnes.

La charge de coordination s'élève à 20 % de l'ensemble.

Estimer la charge totale de la mise en œuvre pilote.

10.8 CORRIGÉ DE LA DÉMARCHE GÉNÉRALE D'ESTIMATION

La charge totale correspond à 28 jours-personnes, à répartir entre les différents acteurs et leurs compétences, selon le tableau ci-dessous (figure 10.17).

	Unité d'œuvre	Nombre	Poids	Charge en jour-pers.
Installation des postes	Poste	10	0,25	2,5
Installation du serveur	Unique	1	2	2
Recette	Cas d'utilisation	5	0,5	2,5
Préparation formation	Entrées simples	3	0,5	1,5
	Entrées moyennes	2	1	2
	Entrées complexes	2	2	4
	Sorties et interrogations complexes	3	2	6
Animation formation	1re formation	2	1	2
	2e formation	1	0,5	0,5
Sous-total				**23**
Coordination			20 %	5
TOTAL				**28**

Figure 10.17 — Estimation de la charge de mise en œuvre pilote du projet Kalizeau.

10.9 COMPARAISON DES DIFFÉRENTES MÉTHODES

Le tableau de la figure 10.18 récapitule les estimations obtenues. Celles-ci ne comprennent ni les charges d'encadrement, ni les charges de recette, ni les charges de mise en œuvre.

Méthode	Charge du projet en mois-personne
Delphi	8
Répartition proportionnelle	11
Cocomo	10,3
Évaluation analytique	10,6
Points fonctionnels	11,9

Figure 10.18 — Comparaison des différentes estimations du cas Parking.

Plusieurs remarques s'imposent.

L'évaluation Delphi est la plus incertaine ; elle est fortement dépendante des expériences de l'expert ; c'est souvent un compromis entre des compréhensions parfois éloignées de l'ambition du projet. Il est bon que l'animateur de la confrontation fasse expliciter les raisons du classement par rapport aux projets de référence.

L'évaluation Cocomo recouvre essentiellement l'étape de réalisation. En effet cette méthode date d'une époque où la conception était limitée. L'évaluation des charges ne la prenait en compte que pour une part très réduite. Il faut donc comparer la valeur obtenue, 10,3 mois-personne, à celle provenant directement de l'estimation analytique avant extrapolation, c'est-à-dire 6,4 mois-personne.

L'évaluation analytique est précise, mais elle oblige à déterminer par type de programme le nombre prévisible. Il est intéressant de comparer en fin de projet avec ce qui a été réellement programmé. Elle permet éventuellement d'ajuster le périmètre fonctionnel de l'application avant son développement, si le délai visé est incompatible avec la charge estimée.

La méthode des points fonctionnels oblige à expliciter l'idée que l'on se fait du futur système. La démarche de dénombrement des composants fonctionnels ne remplace pas les étapes de conception du système d'information. C'est une simple projection d'une solution vraisemblable. On dispose ainsi d'une ébauche dont le développement correspondra à la charge estimée. Cela permet de dire : « Voilà ce que je peux vous faire pour le prix annoncé ! ».

On pourra ainsi faire un dénombrement a posteriori en fin de projet. Il permettra de vérifier si un éventuel écart de charge correspond à une différence fonctionnelle.

11

L'application des techniques de planification des délais

11.1 EXERCICE CONFÉRENCE

Il s'agit d'organiser une conférence scientifique, avec appel à communications et recherche de partenaires et sponsors.

Cette activité comprend les tâches suivantes :

Il faut tout d'abord déterminer le thème, ainsi que la cible visée. Ensuite, un premier groupe s'occupe des aspects scientifiques et du contenu de la conférence, tandis qu'un deuxième groupe recherche des partenaires et des sponsors.

Après la détermination du budget et la sélection des conférenciers, on peut constituer le programme et le diffuser, tout en préparant la logistique.

Les tâches du premier groupe comprennent :

- la rédaction d'un descriptif de la conférence ;
- la constitution d'un comité scientifique et d'un comité de programme ;
- le choix d'une date et d'un lieu ;
- la rédaction et la diffusion des appels à communication ;
- la réception des communications et l'envoi aux notateurs dès le lendemain ;
- la réception des notes, la sélection des communications et la réponse aux auteurs.

Les tâches du deuxième groupe consistent à :

- repérer les appuis potentiels ;
- faire appel à des sponsors possibles et recueillir leurs réponses ;
- prendre contact avec des partenaires, puis dresser un bilan des contacts ;
- déterminer le budget.

Le réseau des antécédents donné à la figure 11.1 représente les contraintes d'enchaînement. Les durées figurent dans les rectangles représentant les tâches. Trouvez la durée minimale de ce projet en respectant les types de liens.

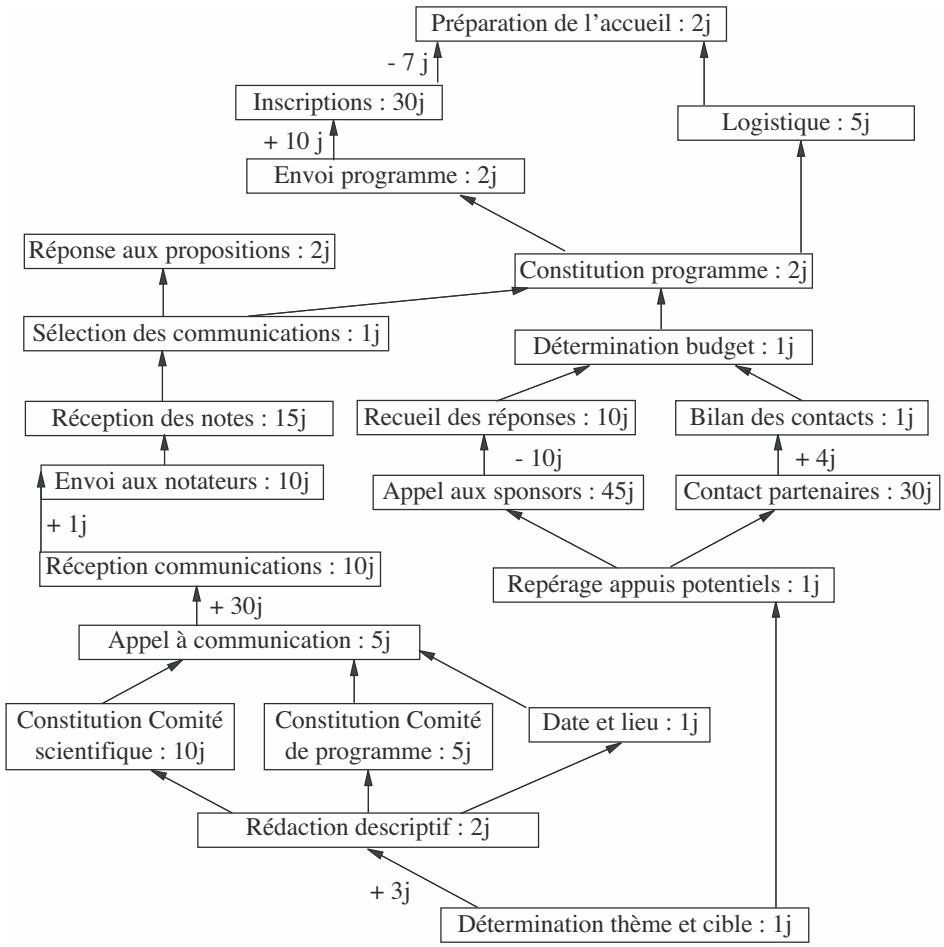


Figure 11.1 — Réseau des antécédents de l'exercice conférence.

11.2 CORRIGÉ DE L'EXERCICE CONFÉRENCE

Le corrigé est donné à la figure 11.2. Tous les liens sont de type fin-début, sauf celui qui unit *Réception communications* et *Envois aux notateurs* qui est de type début-début, avec un retard d'un jour. On relève cinq retards et deux avances. La durée totale est de 122 jours, comme le montre le graphe ci-dessous, où l'on a fait figurer les dates de début et fin à côté de chaque tâche, en supposant que le projet commence au temps 1. Elles doivent se lire ainsi, par exemple pour la tâche *Détermination thème et cible* : **1**

1

La tâche commence en début du jour 1 et se termine en fin du jour 1.

La tâche *Inscriptions* est celle qui se termine en dernier.

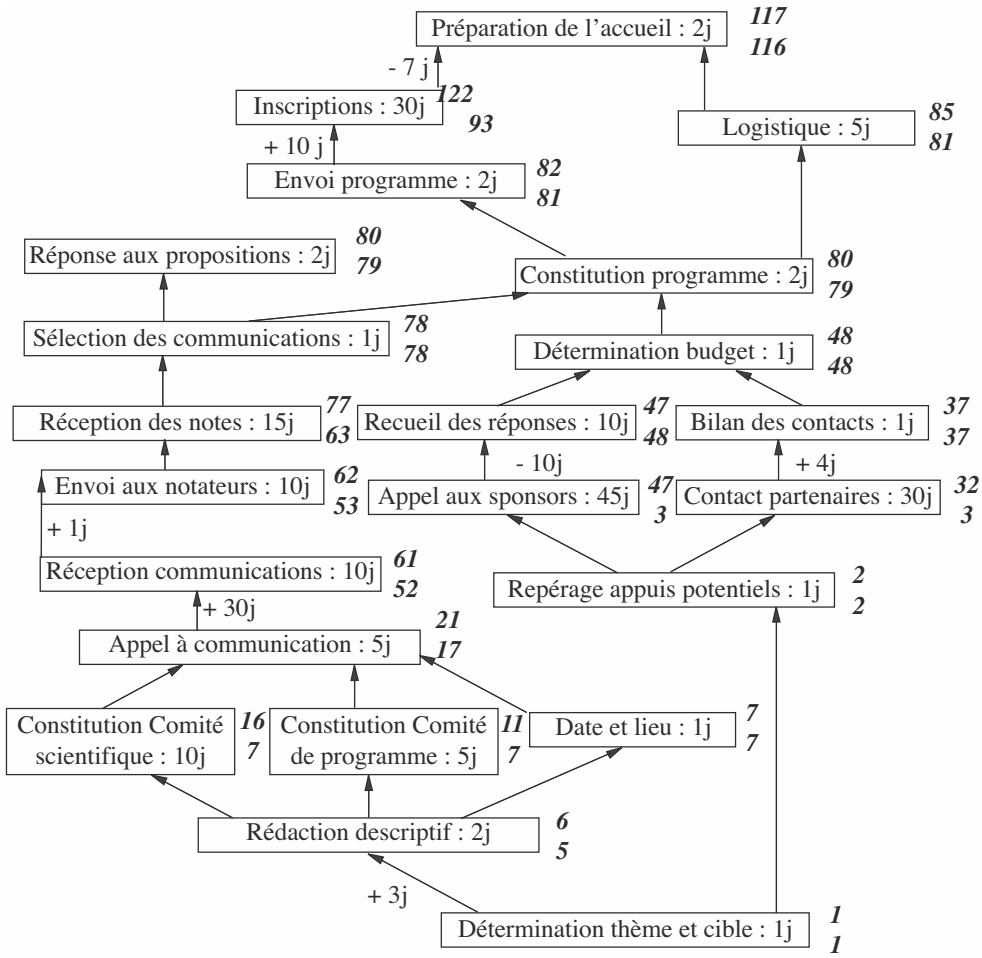


Figure 11.2 — Corrigé de l'exercice Conférence.

11.3 EXERCICE PETIT DÉJEUNER

Vous vous préparez chaque matin un petit déjeuner comportant : une saucisse, deux œufs mollets (que vous mettez directement dans l'eau froide), deux toasts et un verre de lait. Vous cherchez à réduire au minimum le temps de préparation. Pour cela, vous devez établir une stratégie d'ordonnancement de vos tâches. Pour la représenter, tracez un graphe représentant l'enchaînement des tâches identifiées à la figure 11.3, sachant que :

- vous êtes seul pour tout faire ;
- vous préparez à la cuisine, mais vous mangez dans la salle à manger ;

N° tâche	Description de la tâche	Durée en secondes
1	Prendre verre, assiette et couverts dans le buffet	20
2	Poser les couverts sur la table	15
3	Prendre le pain dans le garde-manger	10
4	Mettre deux tranches de pain dans le grille-pain	5
5	Faire griller le pain	140
6	Poser les toasts sur l'assiette	5
7	Beurrer les toasts dans l'assiette	15
8	Sortir les œufs, le beurre et la saucisse du réfrigérateur	25
9	Mettre de l'eau dans la casserole	10
10	Mettre les œufs dans l'eau	5
11	Faire cuire les œufs	170
12	Mettre les œufs dans l'assiette	20
13	Sortir le lait du réfrigérateur	10
14	Verser du lait dans le verre	5
15	Poser le verre de lait sur la table	10
16	Mettre la saucisse dans la poêle	5
17	Faire cuire la saucisse	210
18	Poser la saucisse dans l'assiette	15
19	Poser l'assiette pleine sur la table	10
20	Se mettre à table	2

Figure 11.3 — Tâches de l'exercice Petit déjeuner.

- n'ayant que vos deux mains, vous ne pouvez faire qu'une seule chose (tâche) à la fois ;
- vous disposez de quatre plaques de cuisson ;
- vous devez intervenir sans attendre dès que la cuisson est terminée ;
- vous préférez beurrer vos toasts quand ils sont très chauds.

11.4 CORRIGÉ DE L'EXERCICE PETIT DÉJEUNER

L'analyse du problème fait apparaître les points suivants.

- On doit préparer quatre éléments indépendants : saucisses, œufs, pain et lait.
- La seule possibilité de parallélisme est celle des tâches de cuisson.

Les tâches de cuisson sont celles qui prennent le plus de temps.

La stratégie d'ordonnancement sera donc de commencer par la tâche de cuisson la plus longue.

On obtient le graphe de la figure 11.4, avec une durée minimale de *277 secondes*.

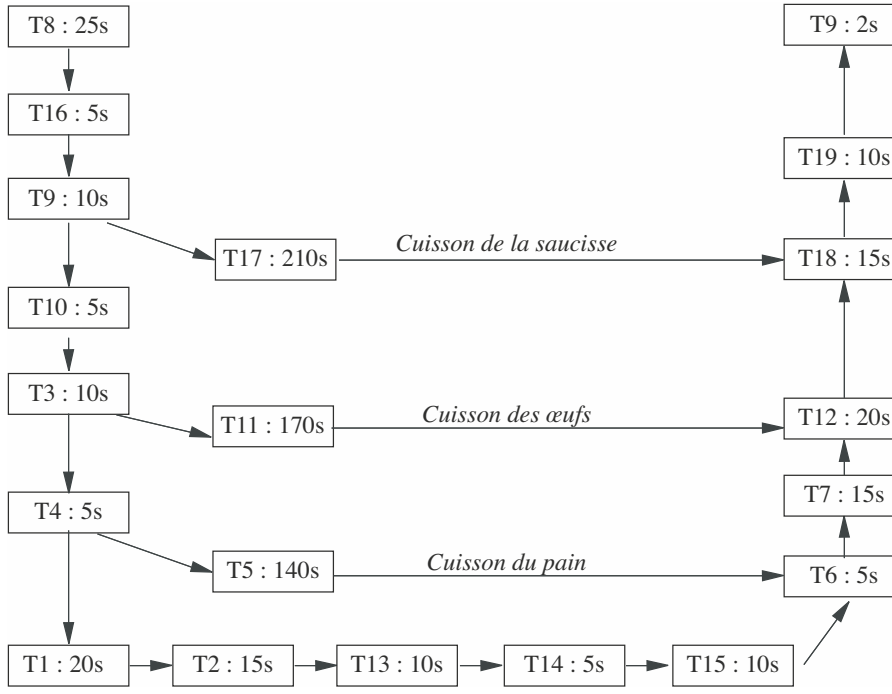


Figure 11.4 – Graphe de l'exercice Petit déjeuner.

La sortie du pain du garde-manger (T5) ne peut être faite pendant la cuisson des œufs, sinon on ne peut terminer le pain avant la fin des œufs. Cela oblige à remplir la casserole d'eau (T2) avant de commencer à cuire la saucisse. Sinon on ne pourrait finir les œufs avant la fin de la cuisson de la saucisse.

11.5 EXERCICE RÉVEIL

Les frères Marx se préparent tous les jours pour partir rejoindre les studios selon un ordre désormais coutumier (figure 11.5). Tracez un graphe de type réseau des antécédents pour détecter d'éventuels inconvénients. Proposez, le cas échéant, des améliorations.

11.6 CORRIGÉ DE L'EXERCICE RÉVEIL

À partir de la sonnerie du réveil de Groucho et jusqu'à leur départ de la maison, les préparatifs des trois frères peuvent être représentés comme à la figure 11.6. Il n'y a que des liens de type fin-début. Par simplification, nous n'avons pas fait apparaître les paramètres clés, mais uniquement les dates au plus tôt et les temps d'attente de certains (en gras). Ces derniers correspondent à la marge. Le chemin critique correspond ici au chemin le plus long. C'est celui de Harpo.

L'*analyse* du réseau fait apparaître les points suivants :

- La durée totale depuis le réveil du premier jusqu'au départ est de 69 minutes : c'est le chemin le plus long.
- Les chemins correspondant à chacun des trois frères sont de longueur inégale. Un chemin se compose d'une part des tâches que chacun effectue et d'autre part de temps d'attente. Si l'on appelle *temps utile* la somme des tâches d'un acteur, on obtient :

  – temps utile de Groucho : 44 minutes ;

  – temps utile de Harpo : 67 minutes ;

  – temps utile de Chico : 29 minutes.

  Les temps d'attente se répartissent de la façon suivante :

  – Groucho : 24 minutes ;

  – Harpo : 0 minute et 2 minutes de sommeil supplémentaire ;

  – Chico : 34 minutes et 6 minutes de sommeil supplémentaire.

N°tâche	Description	Durée en minutes	Prédécesseur
Groucho			
G1	Se réveille	1	–
G2	Réveille Harpo	1	G1
G3	Réveille Chico	4	G2
G4	Se coiffe	1	G3
G5	Prend une douche	5	G4, H2
G6	Choisit ses habits	5	G5
G7	S'habille	2	G6
G8	Va promener le singe	5	G7
G9	Prépare le repas du singe	10	G8
G10	Prend son petit déjeuner	10	G9, H6
G11	Prend son chapeau	1	G10
Harpo			
H1	Bondit de son lit	1	G2
H2	Prend un bain	9	H1
H3	Fait de la gymnastique	20	H2
H4	S'habille	10	H3
H5	Fume une cigarette	2	H4
H6	Prépare le petit déjeuner	5	H5
H7	Prend son petit déjeuner	10	H6
H8	Prépare le pique-nique	10	H7
Chico			
C1	S'étire	2	G3
C2	Se douche	15	C1, G5
C3	Fait des assouplissements	1	C2
C4	S'habille	5	C3
C5	Prend son petit déjeuner	5	C4, H6
C6	Prend son écharpe	1	C5

Figure 11.5 — Tâches de l'exercice Réveil.

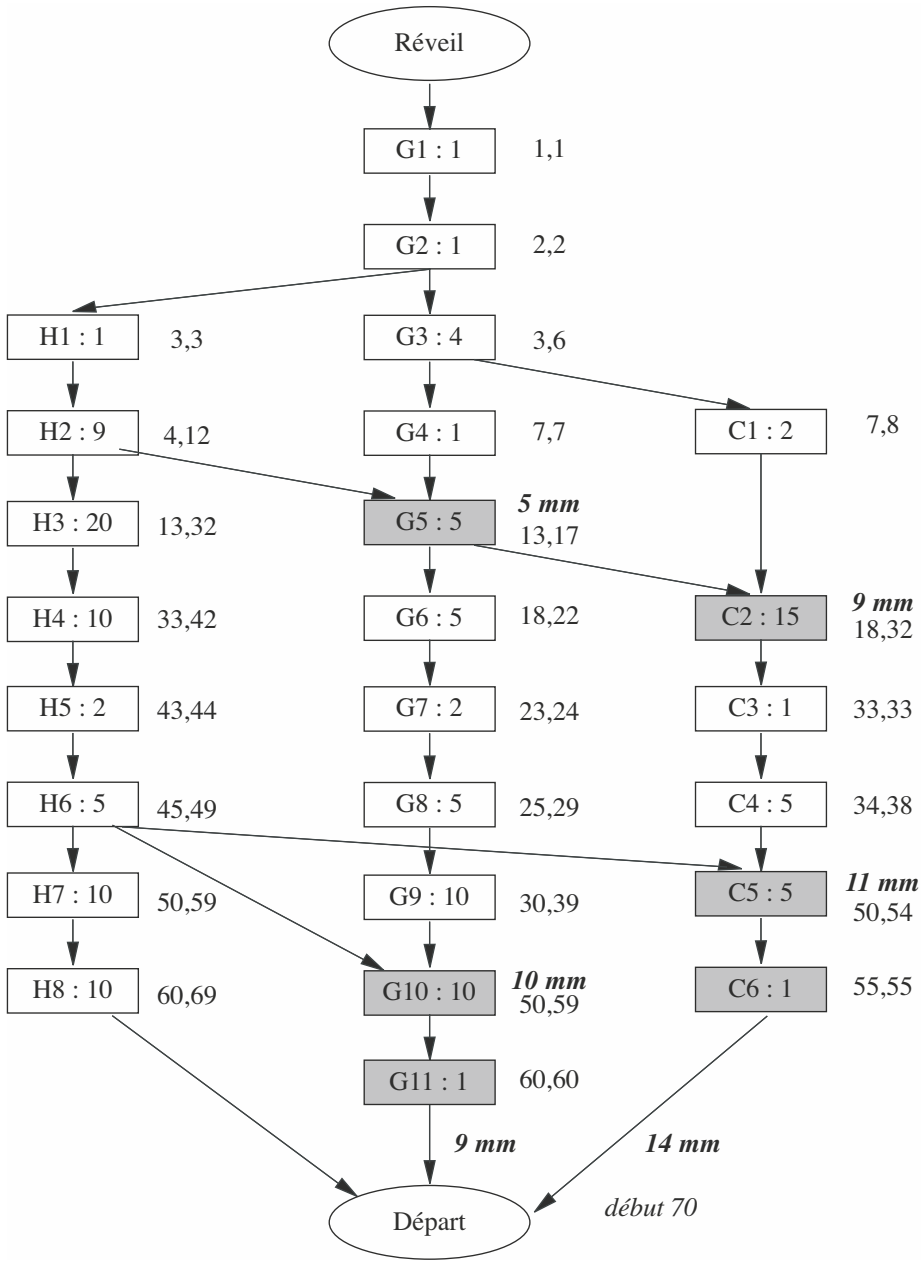


Figure 11.6 — Graphe de l'exercice Réveil.

Ainsi, Harpo est très actif et n'attend jamais. À l'inverse, depuis son réveil, Chico passe plus de temps à attendre qu'à s'activer. Le chemin de Groucho se répartit en près de deux tiers de temps actif pour plus d'un tiers de temps d'attente.

- La douche est une ressource partagée qui peut être considérée comme un point de blocage, puisque Groucho et Chico se retrouvent tous deux en situation d'attente.
- Le petit déjeuner est pris en commun, ce qui est une excellente chose. Mais on peut également considérer que c'est un point de blocage. En effet, Groucho et Chico attendent que Harpo ait fini de le préparer pour tout le monde.
- Le pique-nique préparé par Harpo après le petit déjeuner retarde le départ.

Les inconvénients sont directement liés aux critères utilisés pour juger d'un « bon » réseau. En dehors des habitudes personnelles des trois frères, on peut faire deux suggestions.

D'une part, les temps d'attente pourraient être réduits ou transformés en temps de sommeil supplémentaire.

D'autre part, la durée totale de préparation, entre la sonnerie du réveil et le départ, pourrait elle-même être raccourcie.

Pour cela, plusieurs moyens sont à notre disposition.

Pour réduire le chemin critique, il faut essayer soit de réduire la durée de certaines de ses tâches, soit de supprimer certaines tâches ou de les transférer à d'autres.

On peut ainsi suggérer à Harpo :

- soit de s'abstenir de fumer ;
- soit de préparer le pique-nique la veille ou bien de le confier à un de ses frères ;
- soit de prendre une douche, ce qui devrait être plus rapide qu'un bain.

Pour réduire les temps d'attente, il faut travailler sur les points de blocage. La solution consiste soit à modifier certaines contraintes, soit à rajouter des ressources. Ainsi, on peut imaginer :

- que Groucho et Chico prennent successivement leur douche pendant que Harpo fait sa gymnastique, ce dernier ne se baignant qu'ensuite (interversion des tâches H2 et H3) ;
- l'aménagement d'une douche complémentaire, ce qui supprime l'attente de Harpo ; Chico pourrait alors être réveillé après la toilette de ce dernier ;
- que le petit déjeuner soit préparé par Chico, qui a tout le temps après s'être habillé.

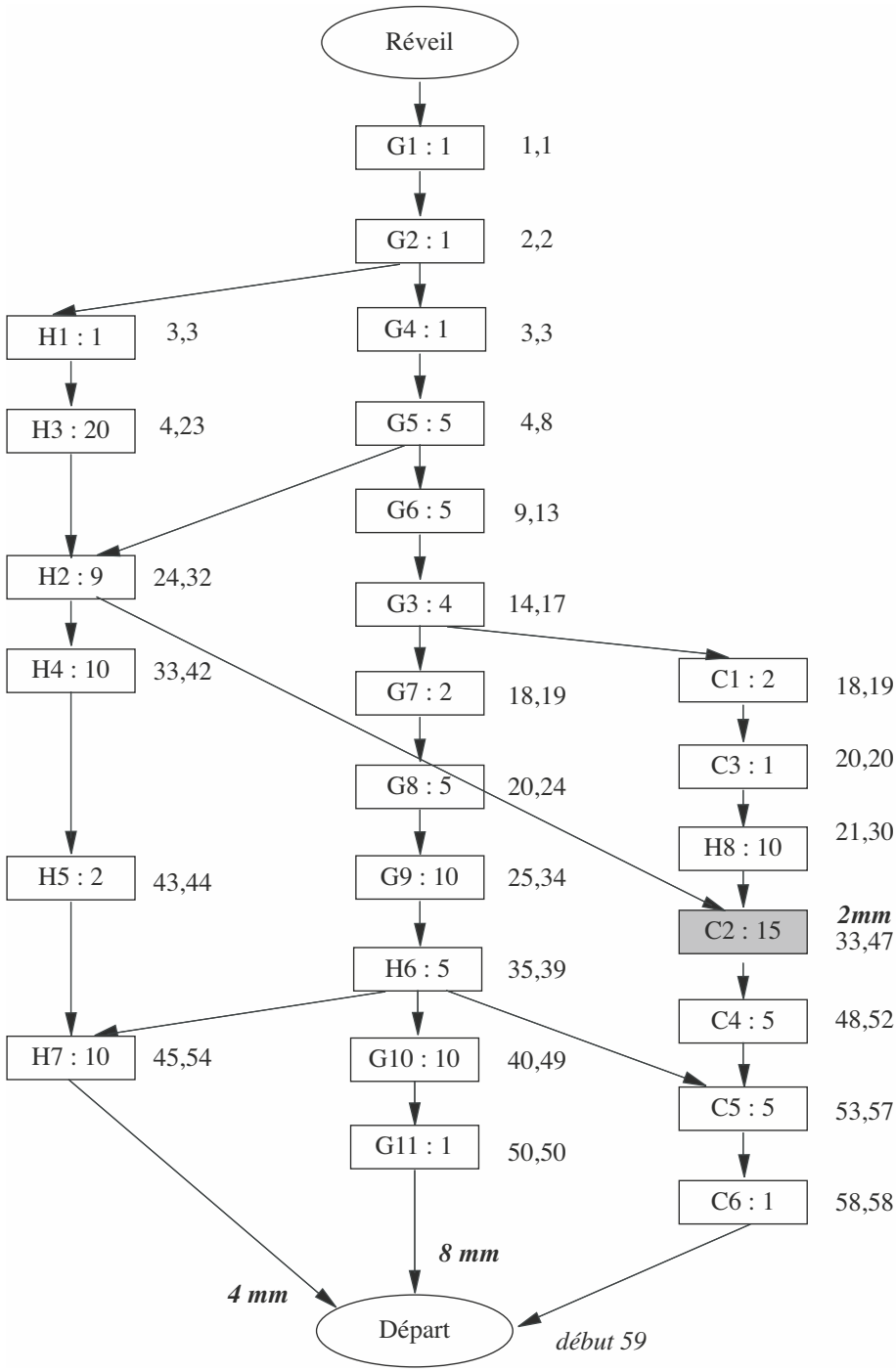


Figure 11.7 – Graphe remanié de l'exercice Réveil.

La première représentation des tâches d'un projet sous forme de graphe des antécédents permet ainsi de réfléchir, pour éventuellement ajuster la définition des tâches et leurs contraintes.

Nous avons refait (figure 11.7) le graphe du réveil des trois frères.

C'est un exemple de réseau remanié, sans ajout de ressource ni suppression de tâche :

- Chico est réveillé plus tard ;
- il prépare le pique-nique en attendant que Harpo ait fini de prendre son bain ;
- il arrive à la table du petit déjeuner cinq minutes après ses frères, mais heureusement il mange plus vite ou moins qu'eux ;
- Harpo fait sa gymnastique avant de prendre son bain, ce qui évite à Groucho l'attente de la salle de bains ;
- il est déchargé du petit déjeuner qui est pris en charge par Groucho ;

On a, sur une durée totale ramenée à 58 minutes, la répartition suivante.

Le temps utile de Groucho est de 50 minutes ; il attend 8 minutes au départ.

Le temps utile de Harpo est de 52 minutes ; il attend 4 minutes au départ et bénéficie de 2 minutes de sommeil supplémentaire.

Le temps utile de Chico est de 39 minutes. Il attend la douche 2 minutes et profite de 17 minutes de sommeil supplémentaire.

11.7 EXERCICE CURSUS SCOLAIRE

Pour obtenir son diplôme, l'étudiant doit avoir suivi avec succès douze modules, d'une durée d'un semestre. Il peut organiser ses études librement, sous réserve des contraintes d'enchaînement données à la figure 11.8.

Tracez le graphe des antécédents. Quelle est la durée minimum des études ?

Proposez, au moyen de diagrammes de Gantt, trois plans d'études :

1. en suivant les modules le plus tôt possible ;
2. en suivant les modules le plus tard possible ;
3. en répartissant au mieux la charge de travail sur toute la durée des études.

Module	Prédécesseurs
A : Comptabilité I	
B : Comptabilité II	A
C : Comptabilité analytique	A
D : Analyse financière	K, B
E : Système d'information I	C, B
F : Système d'information II	H
G : Marketing I	I
H : Marketing II	G
I : Gestion de Production	B, J
J : Économie	
K : Gestion Ressource humaine	J
L : Stratégie	E, G

Figure 11.8 — Contraintes d'enchaînement du cursus scolaire.

11.8 CORRIGÉ DE L'EXERCICE CURSUS SCOLAIRE

Le graphe des antécédents et les paramètres du chemin critique sont donnés à la figure 11.9.

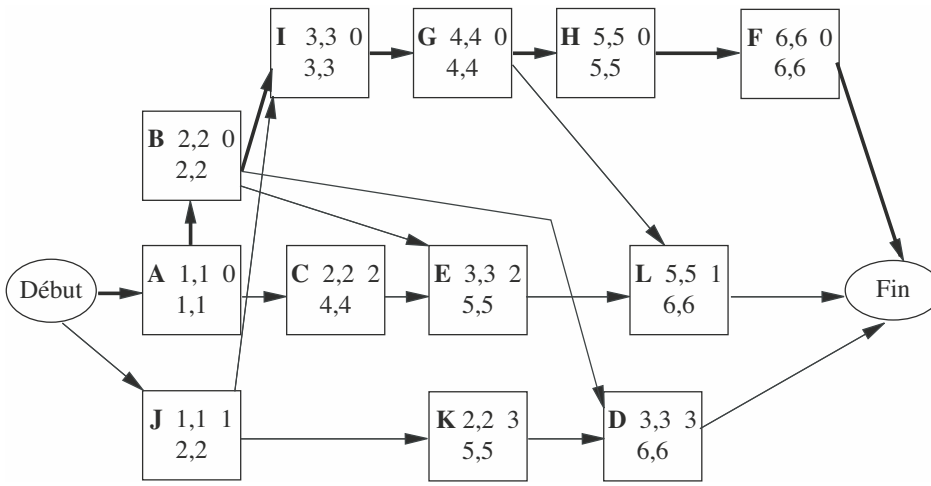


Figure 11.9 — Réseau de l'exercice Cursus scolaire.

Le chemin critique est indiqué en gras : la durée minimum des études est de six semestres.

Le scénario au plus tôt est donné à la figure 11.10. La charge s'allège au bout d'un an et demi.

Semestre	1	2	3	4	5	6
	A	B	I	G	H	F
	J	C	E		L	
		K	D			

Figure 11.10 — Diagramme de Gantt de l'exercice Cursus scolaire (scénario au plus tôt).

Le scénario au plus tard est donné à la figure 11.11. La charge s'alourdit la dernière année.

Semestre	1	2	3	4	5	6
	A	B	I	G	H	F
		J		C	E	L
					K	D

Figure 11.11 — Diagramme de Gantt de l'exercice Cursus scolaire (scénario au plus tard).

Le scénario lissé est donné à la figure 11.12. La charge est uniformément répartie sur les trois années.

Semestre	1	2	3	4	5	6
	A	B	I	G	H	F
	J	C	E	K	L	D

Figure 11.12 — Diagramme de Gantt de l'exercice Cursus scolaire (scénario lissé).

11.9 EXERCICE ÉCHÉANCIER

Soit le projet représenté dans le tableau de la figure 11.13.

Tâche	Durée en semaines	Lien
t1	5	fin t1 - début t3
t2	2	fin t2 - début t4, t5
t3	10	fin t3 - début t6, t8
t4	8	fin t4 - début t6
t5	10	fin t5 - début t7
t6	25	fin t6 - début t11
t7	4	fin t7 - début t11
t8	10	fin t8 - début t9, t10, t11
t9	2	fin t9 - début t13
t10	1	fin t10 - début t13
t11	15	début t11 - début t12 fin t11 - début t13
t12	10	fin t12 - début t14
t13	12	fin t13 - fin
t14	30	fin t14 - fin

Figure 11.13 — Énoncé de l'exercice Échéancier.

Calculez les paramètres clés et faites-les figurer sur un graphe des antécédents. Pour chaque tâche indiquez les dates de début au plus tôt et de fin au plus tôt, de début au plus tard et de fin au plus tard, ainsi que les marges.

Déterminez le chemin critique.

Que devient le chemin critique avec la contrainte :

t9 ne peut pas commencer avant t = 71 ?

Proposez deux échéanciers sans date imposée, d'abord en planifiant au plus tôt, puis au plus tard. Vous les représenterez sur un diagramme de Gantt.

On dispose des ressources R1, R2 et R3, ayant les contraintes suivantes :

- R1 est absent entre les périodes 26 et 50.
- R2 ne peut pas commencer avant la période 8.
- R3 travaille à 50 %.

Établissez un calendrier du projet que vous représenterez avec un diagramme de Gantt.

11.10 CORRIGÉ DE L'EXERCICE ÉCHÉANCIER

11.10.1 Graphe des antécédents et paramètres clés

Le graphe est donné à la figure 11.14. Le chemin critique est indiqué en gras.

Les paramètres clés se lisent ainsi : par exemple, la tâche t3 commence au plus tôt en début du jour 6 et finit au plus tôt en fin du jour 15.

Le lien début-début entre t11 et t12 fait apparaître deux marges sur la tâche t11. En effet, dans le calcul des dates au plus tard, la tâche t13 permet de finir au plus tard en t = 68.

Ceci permettrait que t11 commence au plus tard en t = 68 – 15 + 1 = 54.

Mais si l'on considère la tâche t12, son début au plus tard est égal à 41. Par conséquent, t11 qui déclenche t12, doit s'aligner sur ce début au plus tard. Il en résulte que t11 ne peut pas commencer avant t = 41, mais peut se terminer en t = 68.

Cela signifie que l'on peut envisager un travail à temps partiel sur cette tâche : la durée calculée sur les dates au plus tard l'autorise.

La seconde conséquence est que l'on a deux valeurs de marge : sur les dates de début, la marge est nulle ; en revanche sur les dates de fin, elle est égale à 13.

11.10.2 Graphe avec date imposée

Le graphe modifié est donné à la figure 11.15.

La date imposée sur t9 a des conséquences sur :

- les dates au plus tôt de toutes les tâches subséquentes ;
- la date de fin de projet, puisque t_f passe à 84 ;
- toutes les dates au plus tard, donc sur les marges.

Le chemin critique du deuxième réseau ne parcourt pas tout le graphe, du début à la fin : il se situe uniquement entre t9 et la fin du projet. Les chemins critiques partiels sont fréquents lorsque l'on impose des dates. En l'absence de date imposée, le chemin critique est toujours complet.

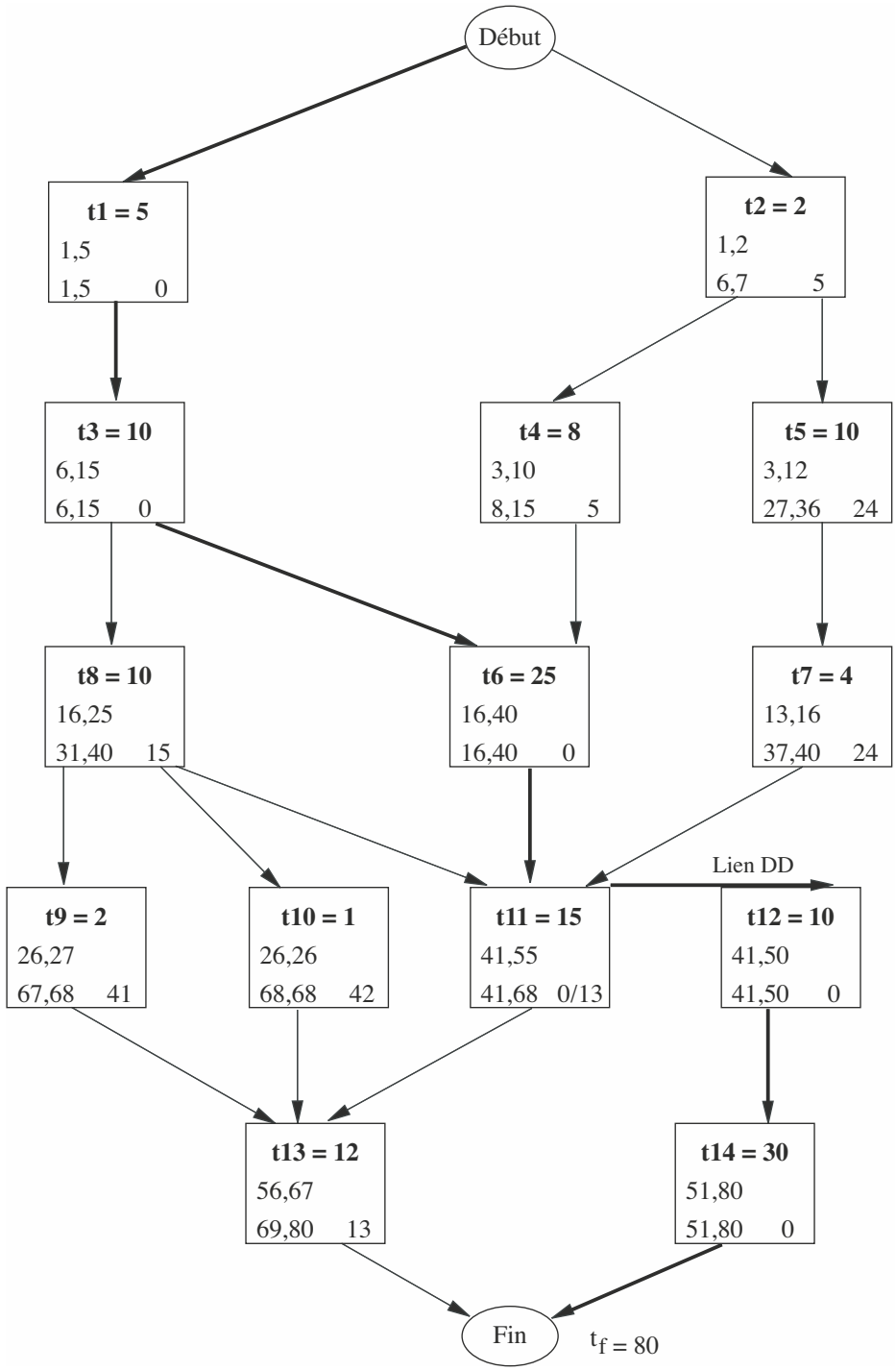


Figure 11.14 — Graphe des antécédents de l'exercice Échéancier.

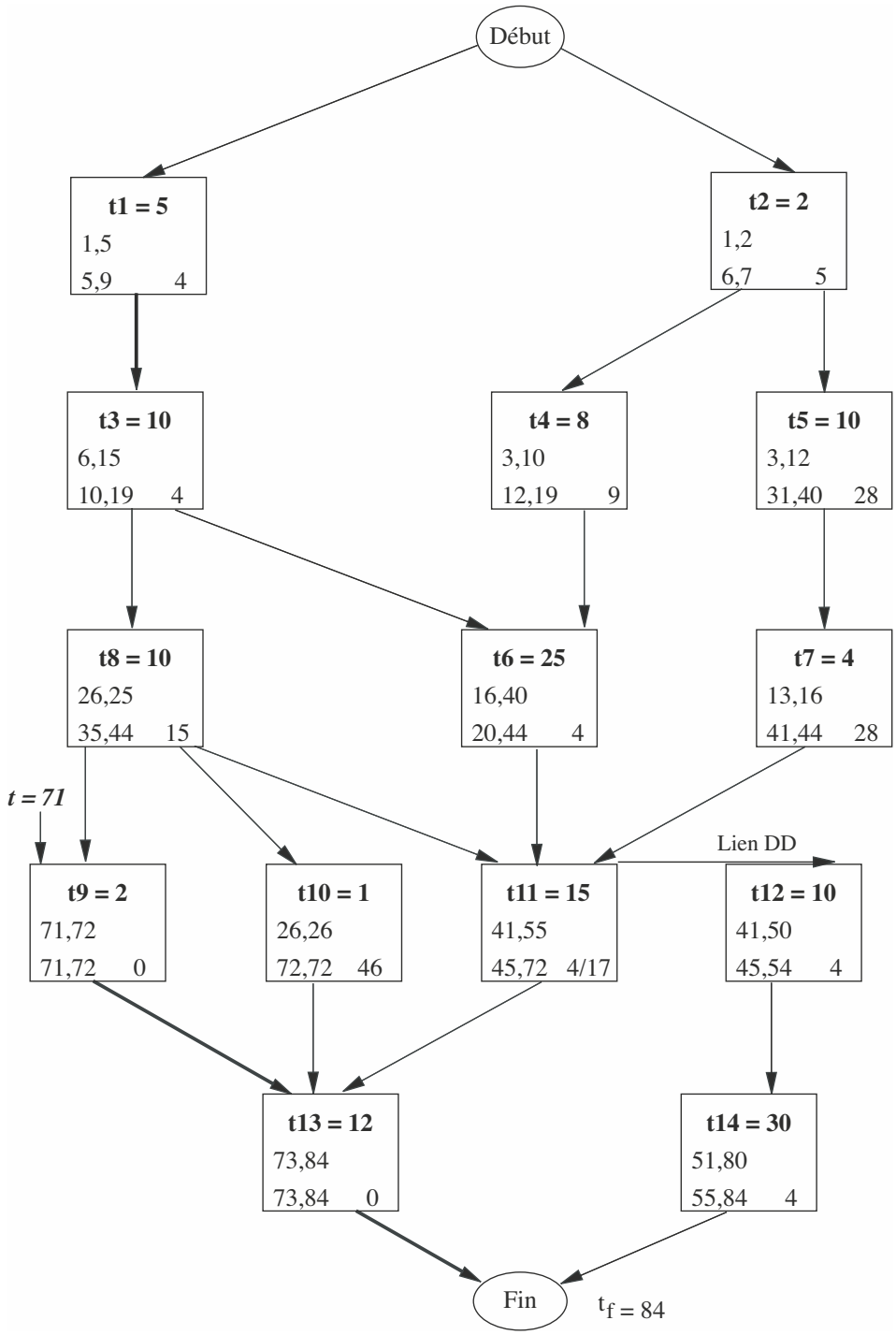


Figure 11.15 — Graphe des antécédents avec date imposée.

11.10.3 Calendrier du projet

La *planification au plus tôt* donne le planning de la figure 11.16.

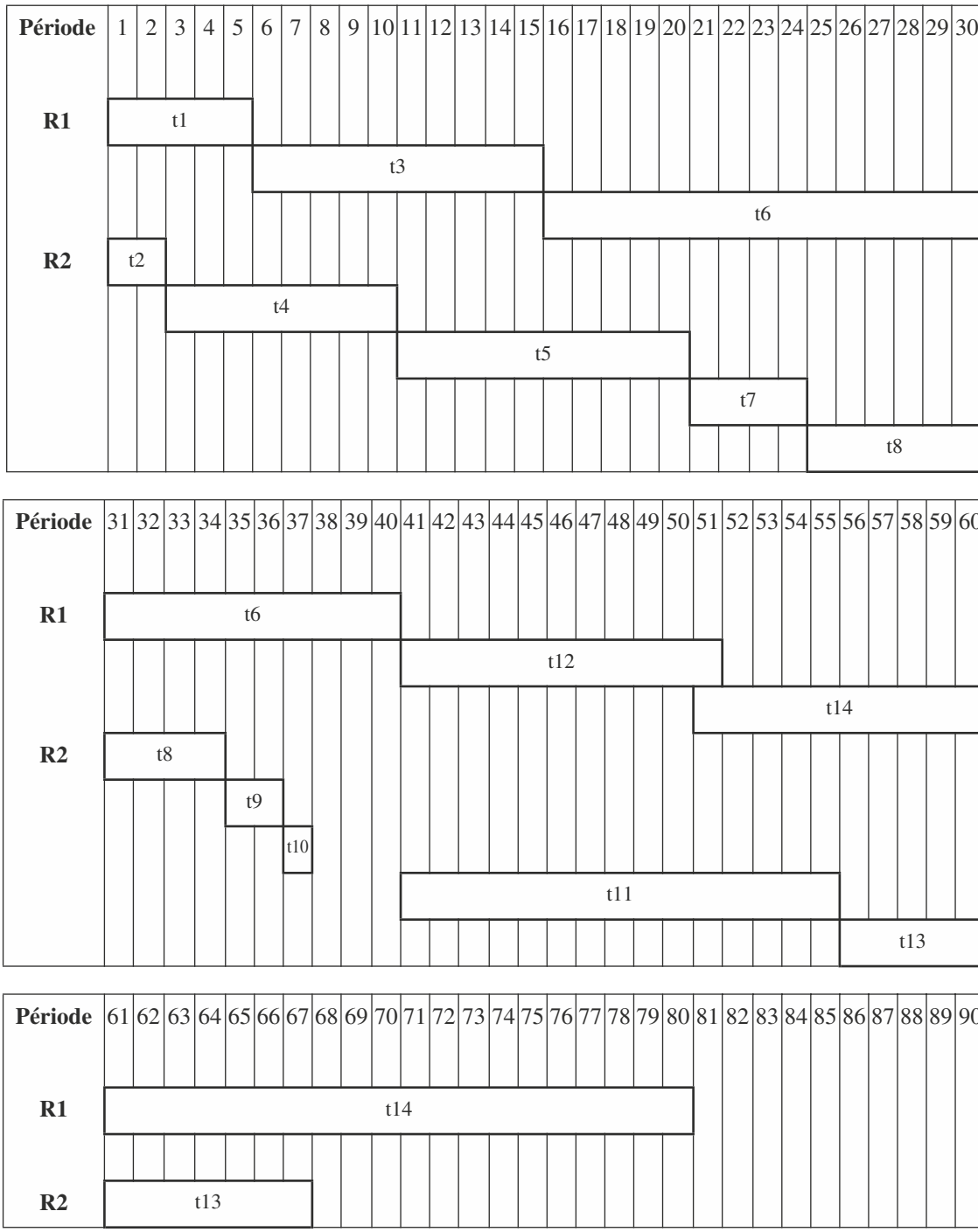


Figure 11.16 — Diagramme de Gantt avec planification au plus tôt.

On planifie d'abord le chemin critique, que l'on affecte à la ressource R1 : t1, t3, t6, t12 et t14.

On repart ensuite au début et on place le plus tôt possible les autres tâches, que l'on affecte à une seconde ressource R2 : d'abord t2, t4 ; puis t5, t7 ; puis t8, t9, t10 ; après une interruption de trois périodes, t11 qui doit impérativement commencer en 40 ; enfin t13.

La planification au plus tard donne le planning de la figure 11.17.

Le chemin critique reste positionné au même endroit. On commence à affecter les tâches à R2 par la fin : d'abord t13 ; puis t10 et t9, en utilisant leurs marges ; puis t11 qui doit commencer en t = 40 ; ensuite on place la séquence t8, t7 et t5 ; la tâche t4 devant s'achever au plus tard en t = 15, on a une période d'attente entre t4 et t5 ; on place enfin t2.

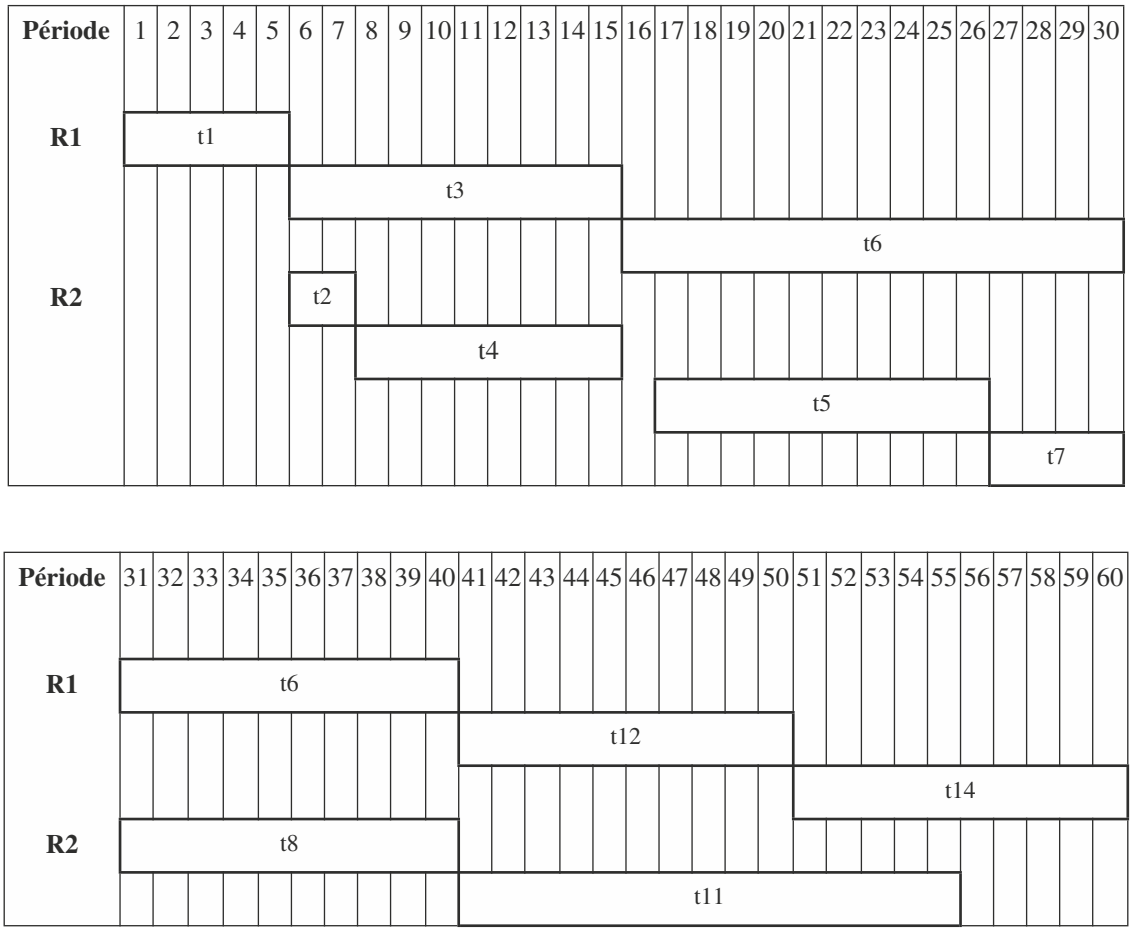


Figure 11.17 — Diagramme de Gantt avec chargement au plus tard.

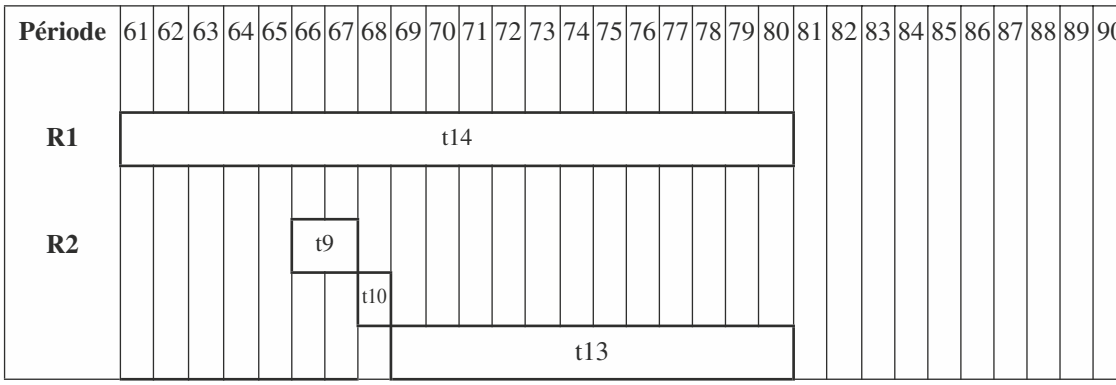


Figure 11.17 — Diagramme de Gantt avec chargement au plus tard. *(Suite)*

11.10.4 Planification avec contraintes sur les ressources

Le diagramme de Gantt est donné à la figure 11.18. Il faut commencer immédiatement les tâches du chemin critique. On y affecte donc R1 jusqu'à son départ en congé. Comme il laisse la tâche t6 inachevée, on préfère la confier à R2 ainsi que la suite du chemin critique.

R1 fait donc t1 et t3. R2 fait t6, t12 et t14. R2 fera t4 avant de commencer t6.

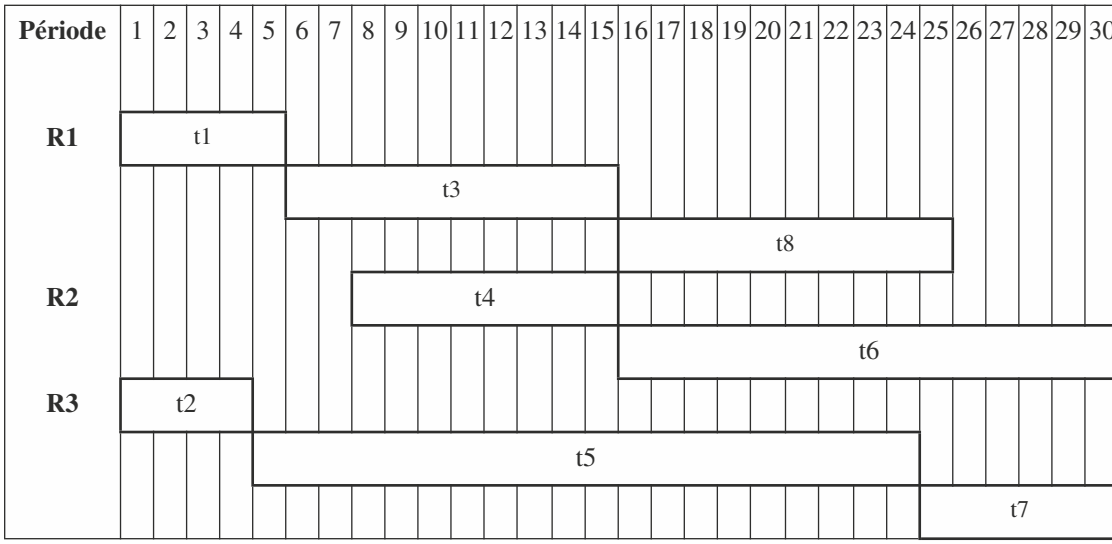


Figure 11.18 — Diagramme de Gantt avec contraintes sur les ressources.

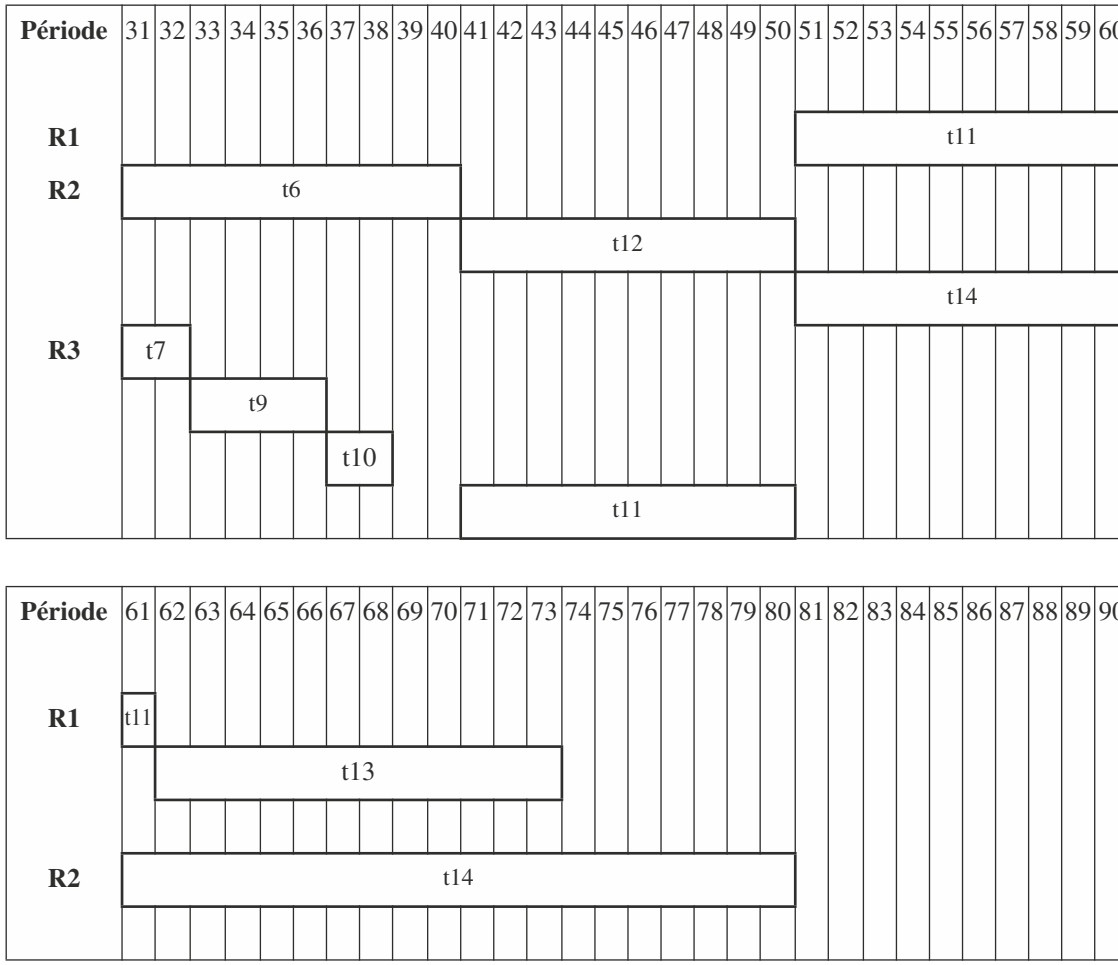


Figure 11.18 — Diagramme de Gantt avec contraintes sur les ressources. *(Suite)*

On cherche à affecter à R3 des tâches dont la marge soit au moins double de la durée : en effet, comme R3 ne travaille qu'à mi-temps, on considère que les tâches qu'il effectue ont une durée calendaire double.

Les tâches éligibles sont : t2, t5, t7, t9 et t10. On les place le plus tôt possible en évitant les interruptions.

Les tâches restantes sont : t11 et t13. Cette dernière est affectée à R1 ; t11 pose un problème car elle doit impérativement démarrer en même temps que t12. On va donc la partager entre :

- R3 qui produit une charge de cinq périodes en dix périodes, entre 41 et 50.
- R1 qui prend la suite à son retour de congé : on compte un jour supplémentaire pour prendre connaissance du travail de R3.

11.11 EXERCICE RECETTE ALÉATOIRE

Soit à estimer la durée d'une recette de gâteau au chocolat. Une incertitude pèse sur la durée de la préparation. Elle provient de différents facteurs : la qualité du four, la disponibilité de certains outillages (batteur électrique, par exemple) et la rapidité des pâtissiers.

Le tableau de la figure 11.19 donne la liste des tâches, ainsi que les estimations optimiste, pessimiste et vraisemblable de chaque tâche.

Tâches		t_{opt}	t_{pes}	t_{vrai}
R1	Faire fondre le beurre et le chocolat	6	9	7,5
R2	Séparer les œufs en jaunes et blancs	1	4,5	3
R3	Ajouter les jaunes au mélange fondu et faire cuire 2 minutes	6	8	7
R4	Monter les blancs en neige	2	12	5
R5	Retirer le mélange du feu et incorporer les blancs	2	6	3
R6	Faire cuire au four	16	22	18

Figure 11.19 — Tâches de l'exercice Recette aléatoire.

Tracez le graphe des antécédents, sans contrainte de ressource (main d'œuvre). Calculez le risque, la durée probable, l'écart-type et la variance de chaque tâche. Quel est le chemin critique ?

Quelle est la durée estimée de préparation du gâteau :

- avec une probabilité de 95 % ?
- avec une probabilité de 90 % ?

Quelle est la probabilité que l'on termine en 37 minutes ?

11.12 CORRIGÉ DE L'EXERCICE RECETTE ALÉATOIRE

Le réseau est donné à la figure 11.20.

Les paramètres probabilistes sont donnés à la figure 11.21.

La durée estimée des chemins est donnée à la figure 11.22.

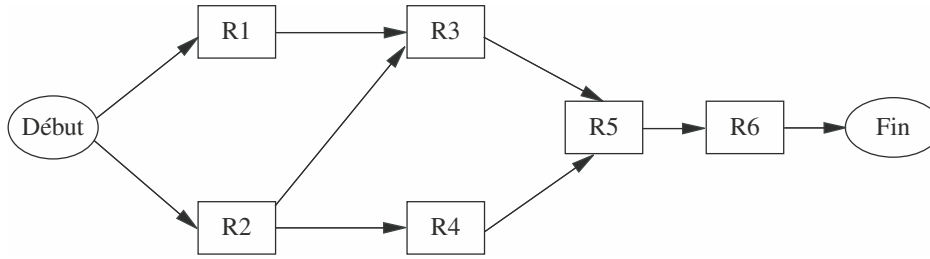


Figure 11.20 — Réseau de l'exercice Recette aléatoire.

Tâches	t_{opt}	t_{pes}	t_{vrai}	risque	$t_{prob}(i)$	e(i)	v(i) = e(i)2
R1	6	9	7,5	0,33	7,50	1/2	0,25
R2	1	4,50	3	0,78	2,92	7/12	0,34
R3	6	8	7	0,25	7,00	1/3	0,11
R4	2	12	5	0,83	5,67	5/3	2,78
R5	2	6	3	0,67	3,33	2/3	0,44
R6	16	22	18	0,27	18,33	1	1,00

Figure 11.21 — Paramètres probabilistes de l'exercice Recette aléatoire.

Chemins	D(est)	V(est)	E(est)
R1R3R5R6	36,17	1,81	1,34
R2R3R5R6	31,58	1,90	1,38
R2R4R5R6	30,25	4,56	2,14

Figure 11.22 — Durée estimée des chemins de l'exercice Recette aléatoire.

Le chemin critique est R1R2R3R6 si l'on considère la durée estimée. Cependant, la variance de R2R4R5R6 étant double de celle du chemin critique (pour une durée estimée de 30,25), il convient de surveiller également ce chemin.

Les durées probables des différents chemins sont données à la figure 11.23. En effet :

- D(95) = D(est) + 1,65 × E(est)
- D(90) = D(est) + 1,28 × E (est)

Chemins	D(95)	D(90)
R1R3R5R6	38,38	37,89
R2R3R5R6	33,86	33,35
R2R4R5R6	33,77	32,98

Figure 11.23 — Durées probables de la Recette aléatoire.

La probabilité d'une durée de 37 minutes se calcule sur le chemin critique.

$$D(p) = D(est) + G(p) \times E(est), \text{ soit : } 37 = 36{,}17 + G(p).1{,}34$$

G(p) = 0,619, d'où l'on déduit que la probabilité est d'environ 72 %.

12

Le système d'information de pilotage

Nous allons maintenant simuler partiellement le déroulement d'un petit projet. Vous en êtes le chef. Après avoir organisé votre projet avec les ressources qui y ont été affectées, vous allez le suivre régulièrement au moyen de votre système d'information de pilotage. À la différence des cas proposés dans les autres chapitres de cette deuxième partie, nous allons poser les questions au fur et à mesure de l'avancement simulé du projet. Nous fournirons progressivement les informations nécessaires. Le corrigé fera suite à la question.

En fin de chapitre, nous proposons un exercice illustrant la méthode de la valeur acquise (paragraphe 12.15), un autre sur la valeur monétaire attendue (paragraphe 12.16) et un troisième sur le calcul de rentabilité avec la méthode du ROI (paragraphe 12.17).

12.1 LE CAS PARKING

Le projet à piloter est la réalisation du projet Parking, dont vous avez estimé la charge au chapitre 10. Vous commencez par organiser votre projet.

Pour cela, vous ferez apparaître les contraintes d'enchaînement entre les tâches en établissant un graphe des antécédents.

Vous disposez de deux ingénieurs, Claude et Camille. Le premier débute, la seconde a deux ans d'expérience. Sur quels critères allez-vous baser la répartition du travail ? Présentez l'affectation des tâches dans un tableau.

Votre client veut mettre en service son application le plus tôt possible. Vous allez planifier le projet en conséquence. Établissez le diagramme de Gantt. Vous ferez les hypothèses suivantes :

Vous commencez en début du mois 1. Le mois 1 compte 31 jours. Le premier jour du mois 1 est un dimanche. Le mois 2 comprend 28 jours et aucun autre jour chômé que les week-ends. Le mois 3 a 31 jours. Les 17 et 20 sont des jours fériés. Le mois 4 est de 30 jours. Seuls les week-ends sont chômés.

Par ailleurs, Camille souhaite partir dix jours en congé : il s'agit d'un reliquat de l'année précédente que le service du personnel lui demande de prendre dans les trois prochains mois.

Vous reprenez les résultats de votre évaluation analytique[1]. Nous les avons reproduits dans la figure 12.1.

Remarques

Les contraintes d'enchaînement sont celles qui résultent de la nature même des tâches. Les lots seront considérés comme quasi indépendants. Le graphe des antécédents fera donc apparaître toutes les possibilités de parallélisme.

Les tests se font sur la base de cas de figure choisis par le client. Ils préparent la recette officielle, qui aura lieu à la livraison de l'application.

Lot	Type de tâche	Nombre d'unités	Poids standard ou ratio	Charge estimée en jours-personne
Parking	Programmes temps réel moyen	10	3 jours-personne	30
	Étude technique		10 %	3
	Jeu essai		20 %	6
	Tests		7 %	2
Véhicule	Programmes temps réel moyens	5	3 jours-personne	15
	Étude technique		13 %	2
	Tests		10 %	1,5

Figure 12.1 — Résultat de l'évaluation analytique du projet Parking.

1. Elle figure au chapitre 10. Les données de base se trouvent au paragraphe 10.1.3 et le corrigé au paragraphe 10.5.

Autorisation	Programmes temps réel difficiles	5	5 jours-personne	25
	Étude technique		8 %	2
	Jeu essai		20 %	8
	Tests		10 %	2,5
Édition	Programmes batch faciles	10	1,5 jours-personne	15
	Étude technique		10 %	1,5
	Jeu essai		20 %	3
	Tests		10 %	1,5
Intégration				10
TOTAL				128 jours-personne

Figure 12.1 — Résultat de l'évaluation analytique du projet Parking. *(Suite)*

12.2 CORRIGÉ DE L'ORGANISATION DU PROJET PARKING

12.2.1 Graphe des antécédents du projet Parking

Le réseau est donné à la figure 12.2. La notation est la suivante : *date début*, *date fin*. Lorsque les dates comportent des demi-journées, elles dont suivies de *m* (matin) ou *a* (après-midi).

Les quatre lots étant quasi indépendants, les seules contraintes à respecter sont les suivantes :

- l'intégration ne peut se faire que sur les modules terminés et testés ;
- dans chaque lot, étude technique, programmation et test doivent se suivre ;
- le jeu d'essai doit être terminé avant les tests.

Le graphe possède donc sept possibilités de parallélisme au départ, que nous n'exploiterons que partiellement.

Le délai normal donné par la formule de la méthode Cocomo[1] est :

$$\text{délai normal} = 2{,}5 \times (6{,}4 \text{ mois-personne})^{0{,}38} = 5 \text{ mois.}$$

1. Cette méthode est présentée au paragraphe 3.7.

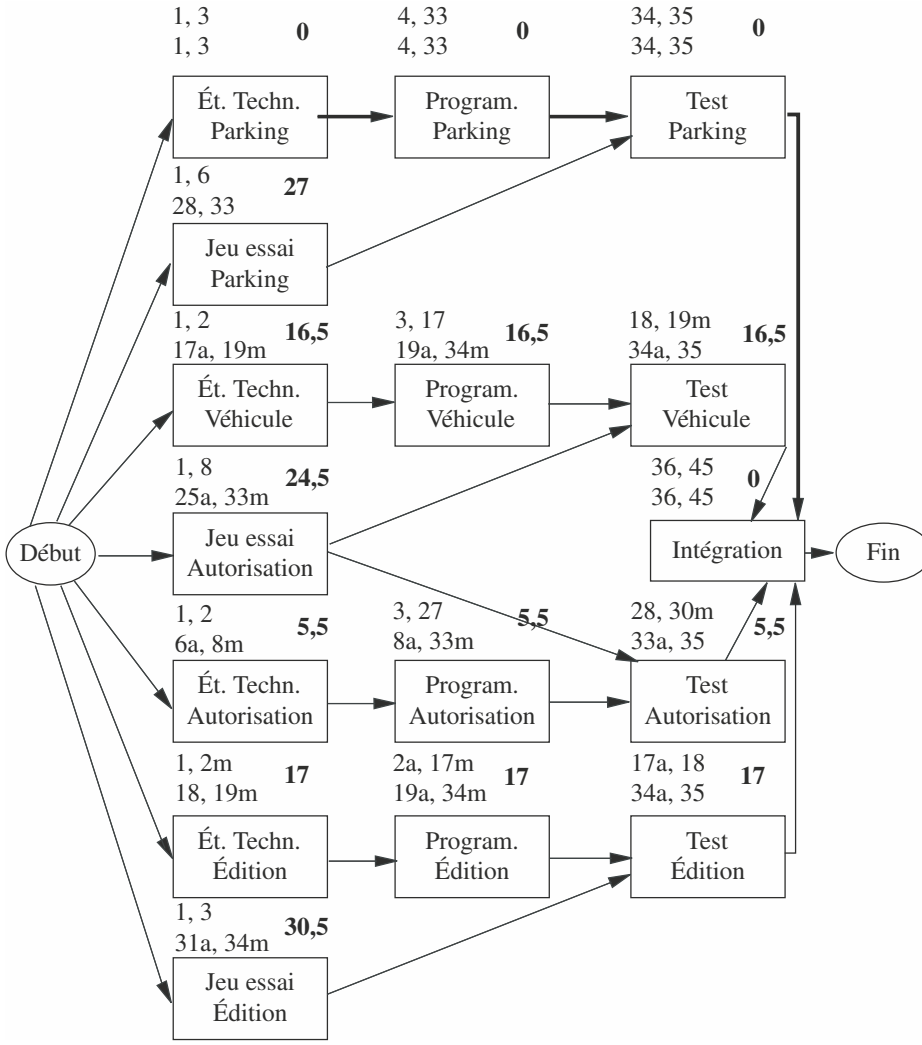


Figure 12.2 — Graphe des antécédents du projet

La formule empirique pour les petits projets[1] de réalisation donne un délai incompressible légèrement inférieur :

$$\text{délai} = 2{,}5 \times (6{,}4 \text{ mois-personne})^{1/3} = 4{,}6 \text{ mois.}$$

Si on choisit d'effectuer le projet dans un délai plus court, on augmente de façon importante les risques de dépassement : en voulant raccourcir, on aboutit parfois au résultat inverse !

1. Proche de la formule proposée par la méthode COCOMO.

12.2.2 Affectation des tâches du projet Parking

L'affectation (figure 12.3) est un compromis à trouver entre les différentes contraintes.

Intervenant	Tâche	Charge initiale	Charge affectée	Charge actualisée
Claude	**Lot Parking**			
	Jeu essai	6	8	
	Étude technique	3		
	Programmation	**30**		
	Tests qualité	2		
	Total		**43**	
	Lot Édition			
	Jeu essai	3		
	Étude technique	1,5		
	Programmation	**15**		
	Tests qualité	1,5		
	Total		21	
	Total Claude		**6**4	
Camille	**Lot Véhicule**			
	Étude technique	2		
	Programmation	**15**		
	Tests qualité	1,5		
	Total		**18,5**	

Figure 12.3 — Affectation des tâches du projet Parking.

	Lot Autorisation			
	Jeu essai	8		
	Étude technique	2		
	Programmation	**25**		
	Tests qualité	2,5		
	Total		**37,5**	
	Intégration	10	**10**	
	Total Camille		**66**	

Figure 12.3 — Affectation des tâches du projet Parking. *(Suite)*

Claude débute : que faire de lui ? Plusieurs solutions se présentent :

- lui donner les tâches les moins risquées ; souvent les débutants sont affectés aux tâches de programmation ; on mise sur le fait que les tests débusqueront les fautes ;
- lui confier les tâches les plus faciles : la typologie des programmes nous y aide ; les éditions sont souvent plus simples que le transactionnel ;
- éviter de le mettre sur le chemin critique.

Par ailleurs, le découpage en lots doit être pris en compte :

- Si une même personne prend en charge un lot complet, cela minimise le travail de mise au courant du dossier. Si on a déjà fait l'étude technique, il est plus rapide de commencer la programmation.
- En revanche, d'éventuelles erreurs ne seront pas débusquées par celui qui prendrait la suite du lot.
- Mener à terme un lot complet est très formateur et constitue une expérience riche.

Si l'on considère qu'un intervenant présente un risque important, deux points de vue s'opposent :

- Lui confier un lot complet confine les risques sur un seul élément de livraison.
- À l'inverse, l'affecter à une tâche semblable sur plusieurs lots favorise un apprentissage spécialisé, mais diffuse le risque.

Un autre critère doit intervenir : l'équilibre entre les intervenants dans la répartition de la charge. Elle est ici de 128 jours. Les deux ingénieurs sont à temps plein, mais Camille a des vacances à prendre. Il faudra intégrer cette contrainte dans la planification.

Pour le cas présent, nous avons choisi une répartition minimisant l'effort de mise au courant et privilégiant l'intérêt pour les développeurs. En effet, chacun a souhaité être responsable d'une partie du projet :

- La répartition se fait par lot complet.
- Les lots *Véhicule et Autorisation* partagent le même jeu d'essai. Celui qui a travaillé sur l'un passera facilement à l'autre.
- Les deux lots ci-dessus représentent 56 jours-personne, soit un peu moins de la moitié de la charge globale qui est de 128 jours.
- Le lot *Édition* étant le plus facile, nous le confions à Claude.
- Nous confions les deux lots *Véhicule et Autorisation* à Camille, qui fera également l'intégration. Sa charge est donc de 66 jours.
- Le lot restant, *Parking*, d'une charge de 46 jours, est confié à Claude. Il faut remarquer que ce lot se situe sur le chemin critique, car c'est le plus grand des quatre. Cependant, sa difficulté étant moyenne, nous prenons le risque de le donner à un débutant.
- Pour prendre en compte la mise au courant de Claude, nous proposons, après discussion avec lui, de lui affecter deux jours supplémentaires pour la tâche d'élaboration de jeu d'essai. C'est en effet une tâche un peu délicate, qu'il n'a encore jamais faite. En effet, au cours de sa formation d'ingénieur il a surtout été formé aux tâches d'étude technique, programmation et test.
- La charge de Claude représente donc 64 jours.

Le tableau d'affectation se présente donc comme à la figure 12.3.

12.2.3 Établissement du planning

Compte tenu de l'affectation précédemment réalisée, nous avons établi un planning serré, sans aucun jour de battement sauf les jours fériés.

De plus, nous n'avons prévu aucun moment pour faire le point : la faible taille de l'équipe favorisera la coordination personnelle. Nous veillerons à la rendre facile par une clarification des rôles et une localisation géographique qui s'y prête. Si possible, les deux ingénieurs seront dans le même bureau.

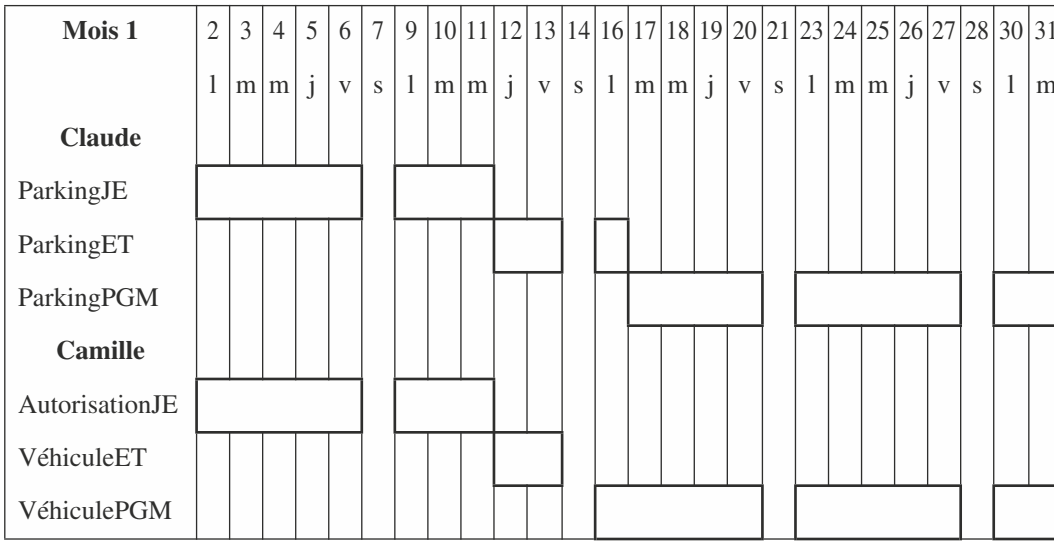
Nous avons préféré planifier les congés de Camille juste avant l'intégration, car sa charge hors intégration est de 56 jours, alors que celle de Claude est de

64 jours. Il y a donc déjà un décalage de 8 jours. Or, Camille ne peut commencer l'intégration avant que Claude ait terminé ses deux lots.

Cela nous donne le calendrier de la figure 12.4, présenté mois par mois. Les demi-journées sont indiquées par une case barrée transversalement.

Le projet commence le lundi 2 du premier mois. Il se termine au milieu du mois 4, ce qui satisfait notre client.

La durée totale du projet est de 106 jours, pour une charge de 130 jours. Le premier mois est le plus chargé comme on le voit sur la figure 12.5.



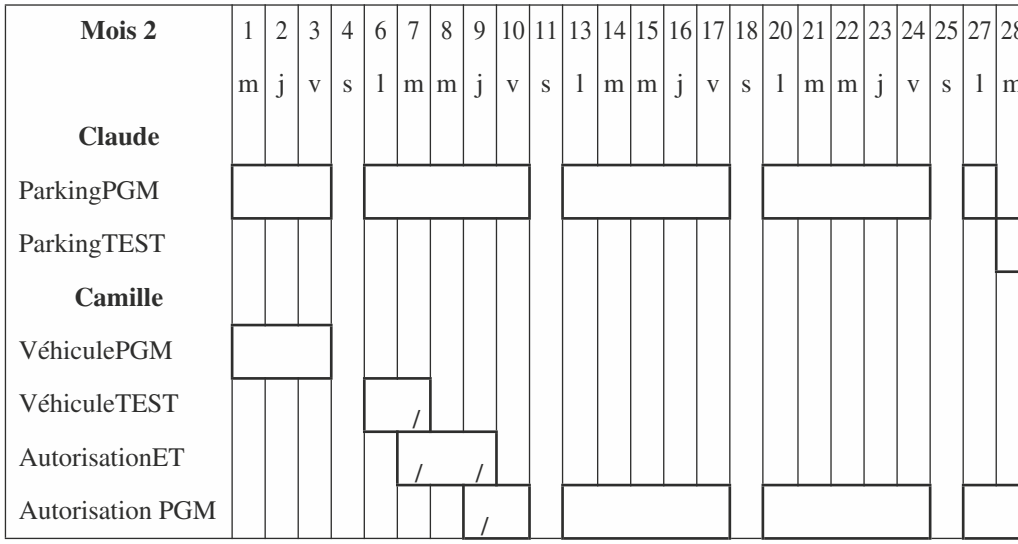
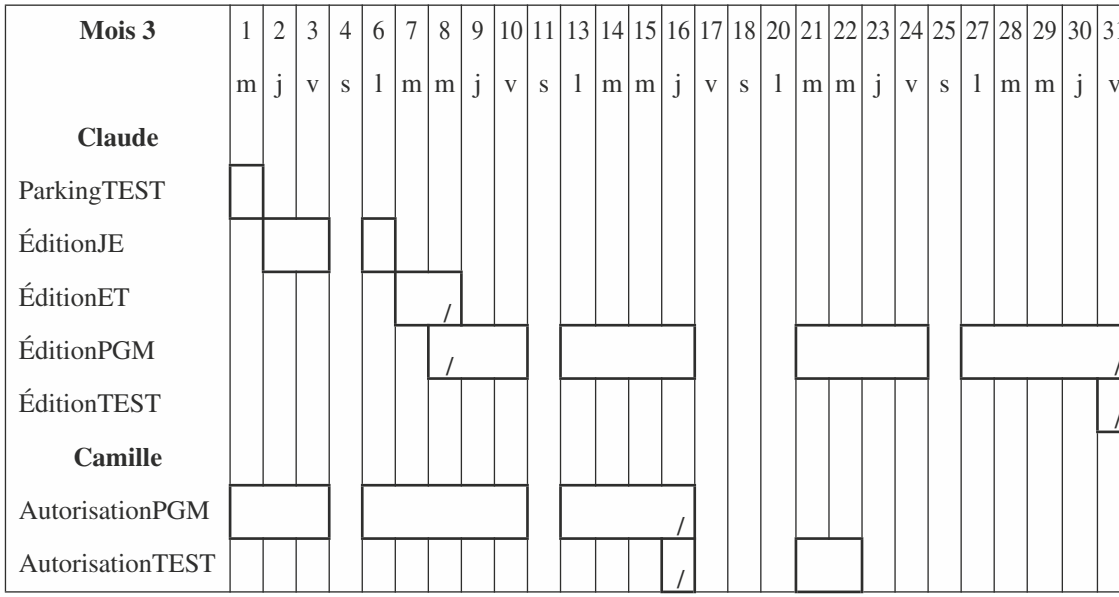


Figure 12.4 — Planning du projet Parking.



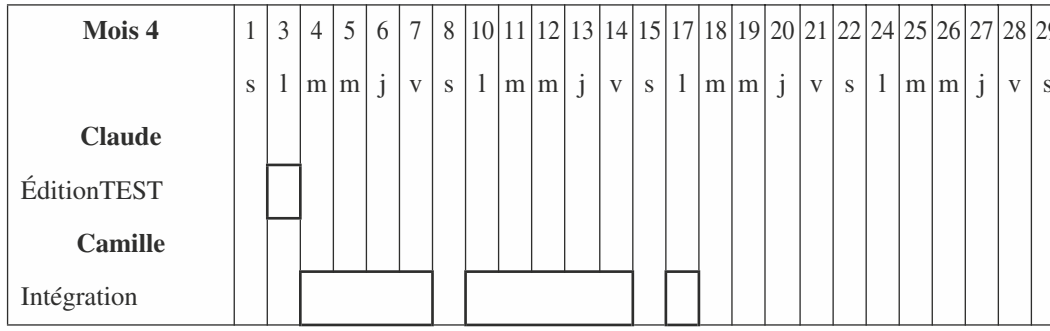


Figure 12.4 — Planning du projet Parking. *(Suite)*

Mois	Durée projet	Nombre de jours travaillés	Charge du mois
1	30	22	44
2	28	20	40
3	31	21	35
4	17	11	11
Total	**106**	**74**	**130**

Figure 12.5 — Répartition par mois de la charge et de la durée du projet Parking.

12.3 ÉNONCÉ DU RÉCAPITULATIF MENSUEL DU PROJET PARKING

Claude et Camille vous remettent leur compte rendu d'avancement chaque semaine. Dresser les récapitulatifs mensuels par intervenant pour le mois 1.

Les comptes rendus d'avancement de la semaine du 2/1 au 8/1 sont donnés à la figure 12.6. Les comptes rendus d'avancement de la semaine du 9/1 au 15/1 sont donnés à la figure 12.7. Les comptes rendus d'avancement de la semaine du 16/1 au 22/1 sont donnés à la figure 12.8. Les comptes rendus d'avancement de la semaine du 23/1 au 29/1 sont donnés à la figure 12.9. Les comptes rendus d'avancement de la semaine du 30/1 au 5/2 sont donnés à la figure 12.10.

Mois 1 Claude	lundi 2	mardi 3	mercredi 4	jeudi 5	vendredi 6	samedi 7	dimanche 8	Reste à faire
ParkingJE	1	1	0,5	1	0,5			4
Visite médicale			0,5					
Réunion nouveaux embauchés					0,5			
Mois 1 Camille	lundi 2	mardi 3	mercredi 4	jeudi 5	vendredi 6	samedi 7	dimanche 8	Reste à faire
AutorisationJE	1	1	1	1	1			0

Figure 12.6 — Comptes rendus d'avancement de la semaine 1.

Mois 1 Claude	lundi 9	mardi 10	mercredi 11	jeudi 12	vendredi 13	samedi 14	dimanche 15	Reste à faire
ParkingJE	1	1	1	1	1			0
Mois 1 Camille	lundi 9	mardi 10	mercredi 11	jeudi 12	vendredi 13	samedi 14	dimanche 15	Reste à faire
VéhiculeET	1	1						0
VéhiculePGM			1	1	1			13

Figure 12.7 — Comptes rendus d'avancement de la semaine 2.

Mois 1 Claude	lundi 16	mardi 17	mercredi 18	jeudi 19	vendredi 20	samedi 21	dimanche 22	Reste à faire
ParkingET	1	1	1					0
ParkingPGM				1	1			28
Mois 1 Camille	lundi 16	mardi 17	mercredi 18	jeudi 19	vendredi 20	samedi 21	dimanche 22	Reste à faire
Congé maladie	1	1	1	1	1			

Figure 12.8 — Comptes rendus d'avancement de la semaine 3.

Mois 1 Claude	lundi 23	mardi 24	mercredi 25	jeudi 26	vendredi 27	samedi 28	dimanche 29	Reste à faire
ParkingPGM	1	1	1	1				24
Journée syndicale					1			
Mois 1 Camille	lundi 23	mardi 24	mercredi 25	jeudi 26	vendredi 27	samedi 28	dimanche 29	Reste à faire
AutorisationJE	1	1	1	1	1			8

Figure 12.9 — Comptes rendus d'avancement de la semaine 4.

Mois 1 Claude	lundi 30	mardi 31						Reste à faire
ParkingPGM	1	1						22
Mois 2 Claude			mercredi 1	jeudi 2	vendredi 3	samedi 4	dimanche 5	Reste à faire
ParkingPGM			1	1	1			19
Mois 1 Camille	lundi 30	mardi 31						Reste à faire
VéhiculePGM	1	1						2
Mois 2 Camille			mercredi 1	jeudi 2	vendredi 3	samedi 4	dimanche 5	Reste à faire
VéhiculePGM			1	1				0
VéhiculeTEST					1			1,5

Figure 12.10 — Comptes rendus d'avancement de la semaine 5.

Le récapitulatif mensuel sera présenté, par intervenant, comme à la figure 12.11. La légende est la suivante :

- t = temps passé ;
- r = reste à faire ;
- a = avancement.

Ces trois valeurs seront calculées pour chaque tâche ouverte pendant la semaine, puis en récapitulatif dans la colonne *total mois*. Dans cette dernière colonne, on récapitulera également pour l'ensemble des tâches du mois, le temps total passé et l'avancement total.

Mois 1		Semaine 1			Semaine 2			Semaine 3			Semaine 4			Semaine 5			Total mois		
Nom	Tâche	t	r	a	t	r	a	t	r	a	t	r	a	t	r	a	t	r	a

Figure 12.11 — Modèle de récapitulatif mensuel.

12.4 CORRIGÉ DU RÉCAPITULATIF MENSUEL DU PROJET PARKING

À partir des valeurs des comptes rendus d'avancement du mois 1, on obtient le tableau récapitulatif de la figure 12.12.

Claude a mis un peu plus de temps que prévu sur le jeu d'essai. En revanche, il semble avoir pris un bon rythme sur les tâches suivantes.

Camille a été très rapide sur le jeu d'essai ; elle avance également plus vite que prévu sur la programmation véhicule.

Ainsi l'avancement n'est pas forcément égal au temps passé. Il peut être supérieur (*AutorisationJE*) ou inférieur (*ParkingJE*).

Mois 1		Semaine 1			Semaine 2			Semaine 3			Semaine 4			Semaine 5			Total mois		
Nom	Tâche	t	r	a	t	r	a	t	r	a	t	r	a	t	r	a	t	r	a
Claude																			
	ParkingJE	4	4	4	5	0	4										9	0	8
	ParkingET							3	0	3							3	0	3
	ParkingPGM							2	28	2	4	24	4	2	22	2	8	22	8
	Total																20	22	19
Camille																			
	AutorisationJE	5	0	8													5	0	8
	VéhiculeET				2	0	2										2	0	2
	VéhiculePGM				3	13	2				5	8	5	2	2	6	10	2	13
	Total																17	2	23

Figure 12.12 — Récapitulatif du mois 1 pour le projet Parking.

Quelque chose doit attirer notre attention : c'est le temps qui n'est pas consacré au projet. Claude a manqué un jour la première semaine et de nouveau un jour en semaine 4. Camille a été malade toute la semaine 3. C'est à suivre de près.

12.5 ÉNONCÉ DU BILAN INDIVIDUEL MENSUEL DU PROJET PARKING

Nous sommes à la fin du mois 2. Vous venez de dresser le récapitulatif mensuel. Dressez le bilan individuel mensuel du mois 2 pour chacun des intervenants.

Le récapitulatif mensuel du mois 2 se présente comme à la figure 12.13.

Le bilan individuel mensuel se présentera comme à la figure 12.14 avec :

coefficient d'utilisation = temps passé/durée planifiée
(pour le mois ou depuis le début du projet)

vitesse = avancement/temps passé

performance = charge affectée × 100/(temps passé + reste à faire)

Mois 2		Semaine 1			Semaine 2			Semaine 3			Semaine 4			Semaine 5			Total mois		
Nom	Tâche	t	r	a	t	r	a	t	r	a	t	r	a	t	r	a	t	r	a
Claude																			
	ParkingPGM	3	19	3	4	18	1	5	14	4	4	10	4	2	8	2	18	8	14
	Total																18	8	14
Camille																			
	VéhiculePGM	2	0	2													2	0	2
	VéhiculeTEST	1	1,5	0	2	0	1,5										3	0	1,5
	AutorisationET				2	0	2										2	0	2
	AutorisationPGM				1	24	1	4	20	4	5	15	5	2	13	2	12	13	12
	Total																19	13	17,5

Figure 12.13 — Récapitulatif du mois 2 pour le projet Parking.

Bilan individuel de : *nom de l'intervenant* pour le mois de : *mois*										
	Rappel		**Mois**					**Récapitulatif projet**		
Tâche	Charge affectée	Reste mois n – 1	Temps passé	Reste à faire	Avancement	Coefficient d'utilisation	Vitesse	Temps passé	Coefficient d'utilisation	Performance

Figure 12.14 — Modèle de bilan individuel.

12.6 CORRIGÉ DU BILAN INDIVIDUEL MENSUEL DU PROJET PARKING

Le bilan de Claude se présente comme résumé à la figure 12.15.

Bilan individuel de : *Claude* pour le mois de : *mois 2*										
	Rappel		**Mois**					**Récapitulatif projet**		
Tâche	Charge affectée	Reste mois n – 1	Temps passé	Reste à faire	Avancement	Coefficient d'utilisation	Vitesse	Temps passé	Coefficient d'utilisation	Performance
ParkingPGM	30	22	18	8	14	0,9	0,77	26		88 %
Total	41	22	18	8	14	0,9	0,77	38	0,9	89 %

Figure 12.15 — Bilan de Claude pour le mois 2.

Celui de Camille est donné à la figure 12.16.

Bilan individuel de : C*amille* pour le mois de : *mois 2*										
	Rappel		**Mois**					**Récapitulatif projet**		
Tâche	Charge affectée	Reste mois n – 1	Temps passé	Reste à faire	Avancement	Coefficient d'utilisation	Vitesse	Temps passé	Coefficient d'utilisation	Performance
VéhiculePGM	15	2	2	0	2			12		125 %
VéhiculeTEST	1,5	1,5	3	0	1,5			3		50 %
AutorisationET	2	2	2	0	2			2		100 %
AutorisationPGM	25	25	12	13	12			12		100 %
Total	53,5	30,5	19	13	17,5	0,95	0,92	36	0,85	109 %

Figure 12.16 — Bilan de Camille pour le mois 2.

Les deux intervenants ne travaillent pas à temps plein sur le projet, comme le montrent leurs coefficients d'utilisation. Ceci est à surveiller de près.

La performance de Camille est satisfaisante, elle équilibre des lenteurs sur certaines tâches par une rapidité sur d'autres.

La vitesse de Claude sur la programmation *Parking* doit nous alerter : il faut faire le point avec lui et voir les difficultés qu'il rencontre.

12.7 ÉNONCÉ DE LA GESTION DES ALÉAS DU PROJET PARKING

Nous sommes au mois 3. Le mardi 21 au soir, Claude se foule la cheville. Vous faites le point de son avancement au téléphone, puis vous allez voir Camille pour savoir où elle en est. Vous disposez de leurs comptes rendus d'avancement.

Dressez le diagramme de Gantt permettant de comparer avancement planifié et déroulement du projet.

Quel problème y a-t-il ? Que faites-vous ?

Les comptes rendus d'activité de Claude sont donnés à la figure 12.17. Il est arrêté pendant trois jours. Il pourra reprendre le lundi 27.

Mois 3 Claude	lundi	mardi	mercredi 1	jeudi 2	vendredi 3	samedi 4	dimanche 5	Reste à faire
Parking PGM			1	1	1			6

Mois 3 Claude	lundi 6	mardi 7	mercredi 8	jeudi 9	vendredi 10	samedi 11	dimanche 12	Reste à faire
Parking PGM	1	1	1	1	1			3

Mois 3 Claude	lundi 13	mardi 14	mercredi 15	jeudi 16	vendredi 17	samedi 18	dimanche 19	Reste à faire
Parking PGM	1	1	1	1	férié			0

Mois 3 Claude	lundi 20	mardi 21	mercredi 22	jeudi 23	vendredi 24	samedi 25	dimanche 26	Reste à faire
Parking TEST	férié	1	absent	absent	absent			2

Figure 12.17 — Comptes rendus d'avancement de Claude pour le mois 3.

Camille vous rappelle qu'elle a prévu de prendre quelques jours de congé. Ses comptes rendus d'activité se présentent comme à la figure 12.18.

Mois 3 Camille	lundi	mardi	mercredi 1	jeudi 2	vendredi 3	samedi 4	dimanche 5	Reste à faire
Autorisation PGM			1	1	1			10

Mois 3 Camille	lundi 6	mardi 7	mercredi 8	jeudi 9	vendredi 10	samedi 11	dimanche 12	Reste à faire
Autorisation PGM	1	1	1	1	1			7

Mois 3 Camille	lundi 13	mardi 14	mercredi 15	jeudi 16	vendredi 17	samedi 18	dimanche 19	Reste à faire
Autorisation PGM	1	1	1	1	férié			2

Mois 3 Camille	lundi 20	mardi 21	mercredi 22	jeudi 23	vendredi 24	samedi 25	dimanche 26	Reste à faire
Autorisation PGM	férié	1	?	?	?			1

Figure 12.18 — Comptes rendus d'avancement de Camille pour le mois 3.

12.8 CORRIGÉ DE LA GESTION DES ALÉAS DU PROJET PARKING

Le déroulement du projet peut se suivre sur des diagrammes de Gantt (figures 12.19, 12.20 et 12.21). On a repris la planification initiale. L'avancement réel figure en grisé.

À la fin du premier mois (figure 12.19), on voit que Claude a pris trois jours de retard par rapport à la planification : il a commencé la programmation *Parking* le 18 au lieu du 16 et il n'a pas travaillé sur le projet le 27.

Camille a attaqué la programmation *Véhicule* avec trois jours d'avance, le 11 au lieu du 16. Mais sa semaine d'absence la met finalement en retard de deux jours sur le calendrier.

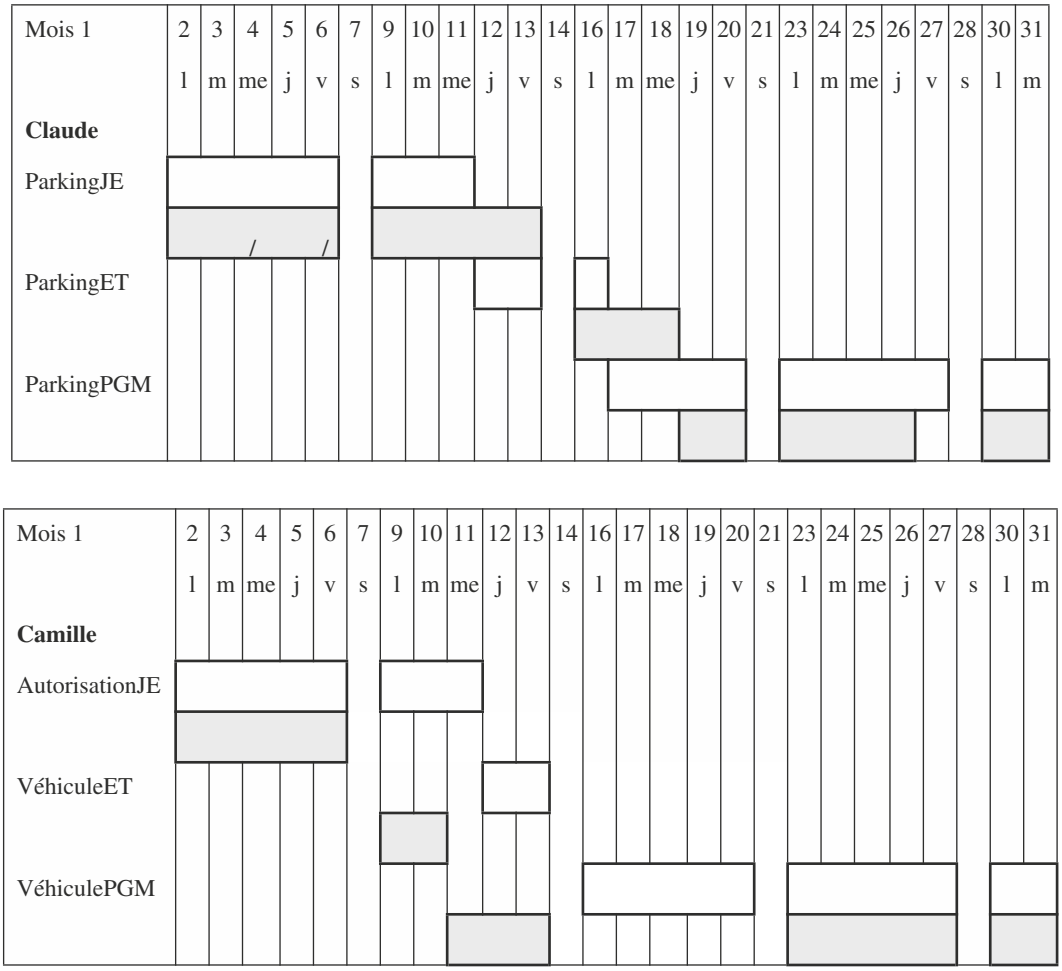


Figure 12.19 — Déroulement du projet Parking (mois 1).

En fin du deuxième mois (figure 12.20), Claude a augmenté son retard par ses absences répétées : 10, 22, 27 et 28. Avec un rythme normal, c'est-à-dire conforme à ce qui était prévu, il a maintenant sept jours de retard.

Camille a largement rattrapé son retard sur la tâche de programmation *Véhicule*, puisqu'elle termine avec un jour d'avance. En revanche, elle consomme davantage que prévu sur les tests, ce qui la place à un jour et demi de retard en fin de mois.

Où en sommes-nous le mercredi 22 au matin (figure 12.21) ? Claude aurait dû commencer les tests de *Parking* le dernier jour du mois 2, le mardi 28. En fait, il termine la programmation *Parking* le mardi 21 du mois 3. Par rapport au planifié, il a 14 jours de retard. Comme il est sur le chemin critique, cela signifie que si l'on ne fait rien, le projet est en retard de près de trois semaines calendaires.

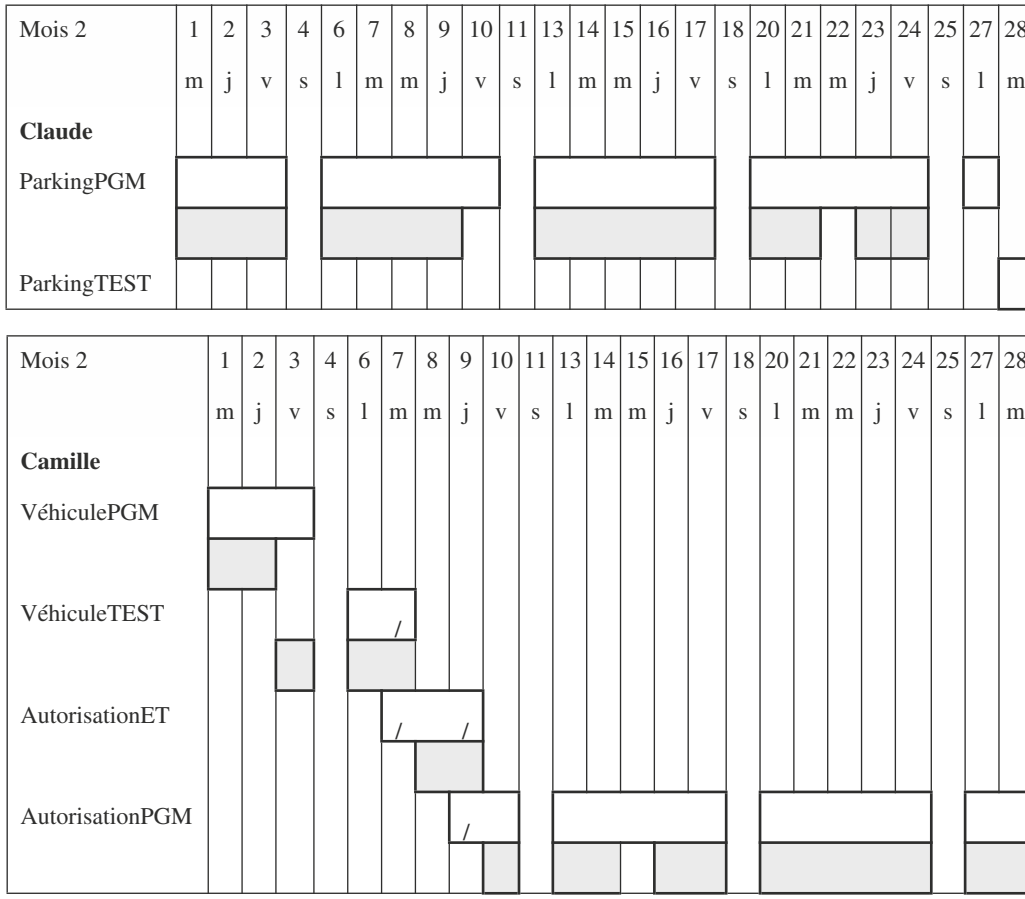


Figure 12.20 — Déroulement du projet Parking (mois 2).

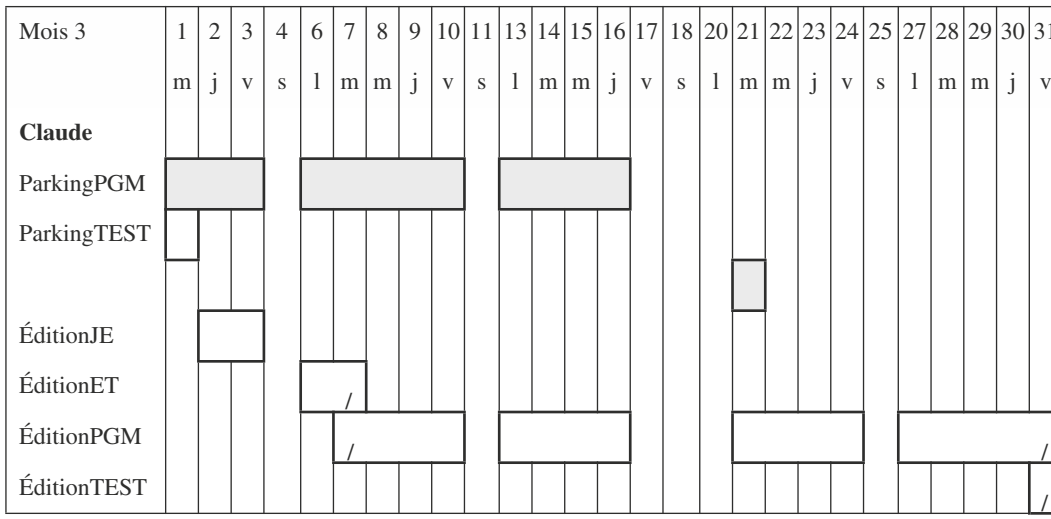


Figure 12.21 — Déroulement du projet Parking (mois 3).

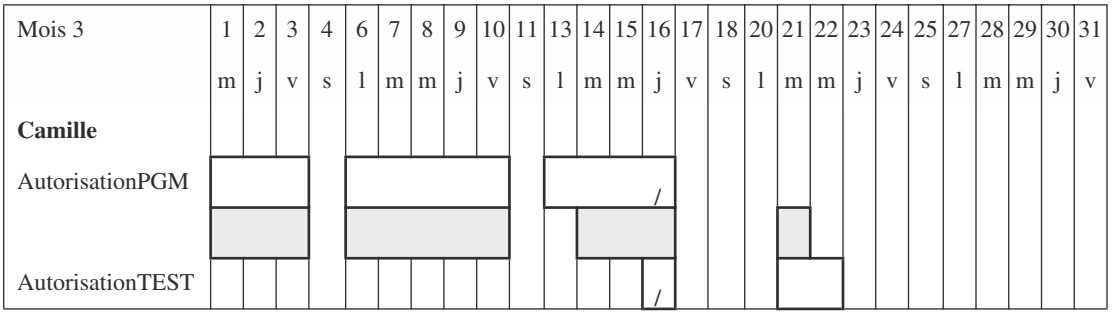


Figure 12.21 — Déroulement du projet Parking (mois 3). *(Suite)*

Quant à Camille, elle est en retard d'un jour et demi sur la fin planifiée de la programmation *Autorisation*. Comme elle annonce un reste à faire d'un jour, elle a en fait deux jours et demi de retard. Par ailleurs, elle part en congé le soir même jusqu'au mardi 4 du mois suivant.

L'essentiel est de réduire le chemin critique. Il faut réaffecter certaines des tâches de Claude. Vous jugez préférable de le laisser terminer le lot *Parking*, puisqu'il ne reste que les tests. En revanche, il faut démarrer immédiatement le lot *Édition*.

Camille accepte de différer légèrement ses vacances, car elle avait prévu de ne voyager que le dimanche. Elle viendra travailler jeudi et vendredi matin pour terminer la programmation *Autorisation* et faire l'étude technique *Édition*. En échange, elle ne rentrera que le mercredi 5 après-midi. Elle prendra son reliquat de congé après la fin du projet ; vous allez le négocier avec le service du personnel.

Vous vous chargez vous-même du jeu d'essai *Édition*.

De son côté, Claude accepte de venir travailler tous les samedis jusqu'à la fin du projet. Il les récupérera par la suite.

La tâche d'intégration commencera avant la fin du lot *Édition*. Celui-ci sera intégré en dernier.

La nouvelle planification se présente comme à la figure 12.22.

Selon le nouveau calendrier, le projet se termine donc avec quatre jours de retard. Il faut soumettre immédiatement cette proposition au maître d'ouvrage et obtenir son accord.

En cas de désaccord, il faudra soit négocier le report de certaines éditions, soit chercher en urgence une ressource supplémentaire. Cette dernière solution est coûteuse, aussi bien en temps que financièrement. Elle présente également un risque en ce qui concerne la qualité.

Mois 3	1	2	3	4	6	7	8	9	10	11	13	14	15	16	17	18	20	21	22	23	24	25	27	28	29	30	31
	m	j	v	s	l	m	m	j	v	s	l	m	m	j	v	s	l	m	m	j	v	s	l	m	m	j	v
Chef de projet																											
ÉditionJE																											

Mois 3	1	2	3	4	6	7	8	9	10	11	13	14	15	16	17	18	20	21	22	23	24	25	27	28	29	30	31
	m	j	v	s	l	m	m	j	v	s	l	m	m	j	v	s	l	m	m	j	v	s	l	m	m	j	v
Camille																											
AutorisationPGM																											
ÉditionET																				/							

Mois 4	1	3	4	5	6	7	8	10	11	12	13	14	15	17	18	19	20	21	22	24	25	26	27	28	29
	s	l	m	m	j	v	s	l	m	m	j	v	s	l	m	m	j	v	s	l	m	m	j	v	s
Claude																									
ÉditionPGM																									
ÉditionTEST														/											

Mois 4	1	3	4	5	6	7	8	10	11	12	13	14	15	17	18	19	20	21	22	24	25	26	27	28	29
	s	l	m	m	j	v	s	l	m	m	j	v	s	l	m	m	j	v	s	l	m	m	j	v	s
Camille																									
AutorisationTEST				/																					
Intégration																									

Figure 12.22 — Nouvelle planification du projet Parking.

Tout événement de la vie du projet doit être consigné sur un *journal de bord*, qui permet ensuite de reconstituer ce qui s'est passé ou de retrouver la cause de certains retards. Un extrait d'un tel journal est donné à la figure 12.23.

Mois 3
Mardi 14 *La machine de test n'a pu démarrer qu' 9 h 50.* Mercredi 15 *Appel de M. Dentre pour l'organisation de la recette. Je dois lui envoyer un calendrier la semaine prochaine.* Jeudi 16 RAS. Vendredi 17 *Férié.* Lundi 20 *Férié.* Mardi 21 *Envoi du calendrier de recette (document numéro 1996/45, daté du 21/3).* Mercredi 22 *Claude s'est foulé la cheville. Il sera absent jusqu'à la fin de la semaine. Une nouvelle planification a été établie qui reporte la fin du projet au 21/4. Réunion à 14 heures avec M^me^ Zort : le report est accepté.*

Figure 12.23 — Extrait du journal de bord du projet Parking.

12.9 ÉNONCÉ DE L'AVANCEMENT DU PROJET PARKING

En fin du mois 3, comme chaque mois, vous dressez le tableau d'avancement du projet. Les comptes rendus d'activité des deux dernières semaines du mois 3 sont ceux des figures 12.24, 12.25 et 12.26. Les autres ont été donnés avec l'énoncé de la question précédente au paragraphe 12.7.

En ce qui vous concerne (figure 12.24), vous avez passé le mercredi et le jeudi à négocier avec votre client et avec le service du personnel. De plus le jeu d'essai *Édition* vous a pris plus de temps que prévu. Vous êtes donc venu travailler samedi 25.

Mois 3 **Chef de projet**	lundi 20	mardi 21	mercredi 22	jeudi 23	vendredi 24	samedi 25	dimanche 26	Reste à faire
ÉditionJE	férié				1	1		2

Mois 3 **Chef de projet**	lundi 27	mardi 28	mercredi 29	jeudi 30	vendredi 31	samedi	dimanche	Reste à faire
ÉditionJE	1	1						0

Figure 12.24 — Compte rendu d'avancement du chef de projet après l'accident de Claude.

Le compte rendu de Camille (figure 12.25) montre qu'elle a passé tout le vendredi 24 sur l'étude technique du lot *Édition*.

Mois 3 Camille	lundi 20	mardi 21	mercredi 22	jeudi 23	vendredi 24	samedi 25	dimanche 26	Reste à faire
AutorisationPGM	férié	1	1					0
ÉditionET				1	1			0

Mois 3 Camille	lundi 27	mardi 28	mercredi 29	jeudi 0	vendredi 31	samedi	dimanche	Reste à faire
	congé	congé	congé	congé	congé			

Figure 12.25 — Compte rendu d'avancement de Camille après l'accident de Claude.

Le compte rendu de Claude est conforme à ce qui était prévu (figure 12.26).

Mois 3 Claude	lundi 20	mardi 21	mercredi 22	jeudi 23	vendredi 24	samedi 25	dimanche 26	Reste à faire
ParkingTEST	férié	1	absent	absent	absent			2

Mois 3 Claude	lundi 27	mardi 28	mercredi 29	jeudi 30	vendredi 31	samedi	dimanche	Reste à faire
ParkingTEST	1	1						0
ÉditionPGM			1	1	1			12

Figure 12.26 — Compte rendu d'avancement de Claude après son accident.

Le tableau d'avancement du projet sera présenté par lots. Il aura la forme du tableau de la figure 12.27, avec :

Évolution de charge pour le mois écoulé
= temps passé – avancement

Évolution de charge en jours depuis le début du projet
= temps passé + reste à faire à ce jour – charge initiale.

Évolution de charge depuis le début du projet en pourcentage
= évolution de charge en jours × 100/charge initiale.

Achèvement en pourcentage
= (Charge initiale – reste à faire à ce jour × 100/charge initiale

Tableau d'avancement du projet : *nom* en fin de mois : *mois*											
	Mois n – 1		**Mois n**				**Récapitulatif projet**				
Lots	Temps passé	Reste à faire	Temps passé	Reste à faire	Avancement	Évolution de charge	Charge initiale	Temps passé	Évolution en jours	Évolution en %	Achèvement en %

Figure 12.27 — Modèle de tableau d'avancement.

12.10 CORRIGÉ DE L'AVANCEMENT DU PROJET PARKING

Nous allons d'abord établir le récapitulatif mensuel du mois 3 (figure 12.28).

Le tableau d'avancement établi à partir des récapitulatifs mensuels des trois premiers mois se présente comme à la figure 12.29.

Les deux colonnes Reste à faire (mois 2 et mois 3) comprennent toutes les tâches du lot sur lesquelles il reste du travail, donc en particulier toutes celles qui ne sont pas commencées.

Ce tableau présenté au maître d'ouvrage lui permet de voir que les deux premiers lots sont terminés, que le troisième est en bonne voie et qu'il reste principalement le lot *Édition* et l'intégration.

Le projet est achevé à 80 %. Jusqu'à ce jour, il a été consommé 10 jours de plus que prévu, soit 8 % de la charge hors encadrement.

Mois 3		Semaine 1			Semaine 2			Semaine 3			Semaine 4			Semaine 5			Total mois		
Nom	Tâche	t	r	a	t	r	a	t	r	a	t	r	a	t	r	a	t	r	a
Claude																			
	ParkingPGM	3	6	2	5	3	3	4	0	3							12	0	8
	ParkingTEST										1	2	0	2	0	2	3	0	2
	ÉditionPGM													3	12	3	3	12	3
	Total																18	12	13
Camille																			
	Autorisation.PGM	3	10	3	5	7	3	4	2	5	2	0	2				14	0	13
	ÉditionET										2	0	1,5				2	0	1,5
	Total																16	0	14,5
Chef de projet																			
	ÉditionJE										2	2	1	2	0	2	4	0	3

Figure 12.28 — Récapitulatif du mois 3 pour le projet Parking.

Tableau d'avancement du projet Parking en fin de mois 3												
	Mois 2		Mois 3				Récapitulatif projet				
Lots	Temps passé	Reste à faire	Temps passé	Reste à faire	Avancement	Évolution de charge	Charge initiale	Temps passé	Évolution en jours	Évolution en %	Achèvemen en %
Parking	18	10	15	0	10	**+ 5 jours**	41	53	**+ 12 jours**	+ 29	100
Véhicule	5	0					18,5	17	**– 1,5 jour**	– 8	100
Autorisation	14	15,5	14	2,5	13	**+ 1 jour**	37,5	33	**– 2 jours**	– 5	93
Édition		21	9	13,5	7,5	**+ 1,5 jours**	21	9	**+ 1,5 jour**	+ 7	36
Intégration		10		10			10	0			0
Total	37	56,5	38	26	30,5	**7,5 jours**	128	112	**10 jours**	+ 8 %	80 %

Figure 12.29 — Avancement du projet Parking au mois 3.

12.11 ÉNONCÉ DU BILAN DU PROJET PARKING

En fin de projet, vous dressez le bilan de projet.

Le récapitulatif mensuel du mois 4 est donné à la figure 12.30. La semaine 1 ne comporte qu'un seul jour, le samedi. Le projet s'est terminé le 22 du mois 4.

Mois 4		**Semaine 1**			**Semaine 2**			**Semaine 3**			**Semaine 4**			**Semaine 5**			**Total mois**		
Nom	Tâche	t	r	a	t	r	a	t	r	a	t	r	a	t	r	a	t	r	a
Claude																			
	ÉditionPGM(1)	1	6	0	6	3	3	6	0	3							13	0	6
	ÉditionTEST										2	0	1,5				2	0	1,5
	Total																15	0	7,5
Camille																			
	AutorisationTEST				3	0	2,5										3	0	2,5
	ÉditionPGM(2)				1	2,5	0,5	3,5	0	2,5							4,5	0	3
	Intégration							2,5	7	3	6	0	7				8,5	0	10
	Total																16	0	15,5
Chef de projet																			
	ÉditionPGM(3)				3	0	3										3	0	3

Figure 12.30 — Récapitulatif du mois 4 pour le projet Parking.

Vous aviez sous-estimé la complexité du générateur d'édition. Claude vous a fait part de ses difficultés pour générer des listes un peu compliquées. Il restait huit programmes à écrire sur les dix.

Vous avez décidé de partager le travail entre vous trois. Vous avez confié deux programmes du lot Édition (durée estimée à trois jours) à Camille, vous en avez pris deux autres et laissé les quatre derniers à Claude.

12.12 CORRIGÉ DU BILAN DU PROJET PARKING

Nous dressons d'abord un tableau d'avancement final (figure 12.31).

Tableau d'avancement du projet Parking en fin de mois 4											
	Mois 3		Mois 4				Récapitulatif projet				
Lots	Temps passé	Reste à faire	Temps passé	Reste à faire	Avancement	Évolution de charge	Charge initiale	Temps passé	Évolution en jours	Évolution en %	Achèvement en %
Parking	15	0					41	53	+ 12 jours	+ 29	100
Véhicule	0	0					18,5	17	– 1,5 jour	– 8	100
Autorisation	14	2,5	3	0	2,5	+ 0,5 jour	37,5	36	– 1,5 jour	– 4	100
Édition	9	13,5	23,5	0	13,5	+ 10 jours	21	31,5	+ 10 ,5 jours	+ 50	100
Intégration	0	10	8,5	0	10	– 1,5 jour	10	8,5	– 1,5 jour		0
Total	38	26	35		26	+ 9 jours	128	146	+ 18 jours	+ 14	

Figure 12.31 — Avancement du projet Parking au mois 4.

Globalement, le projet a consommé 146 jours hors encadrement, soit 14 % de plus que prévu. Nous allons étudier par type de tâche où s'est créé l'écart (figure 12.32). Nous reprenons pour cela la typologie qui a servi de base à l'estimation analytique.

Type tâche	Nombre	Charge estimée	Charge constatée	Écart	%
Jeu essai	3	17	18	1	6
Étude technique temps réel	3	7	7	0	0
Étude technique batch	1	1,5	2	0,5	33
Programme moyen temps réel	15	45	50	5	11
Programme difficile temps réel	5	25	26	1	4

Figure 12.32 — Écarts de charge du projet Parking.

Programme batch facile	10	15	23,5	8,5	57
Test	4	7,5	11	3,5	47
Intégration	1	10	8,5	– 1,5	– 15
Total		128	146	18	14

Figure 12.32 — Écarts de charge du projet Parking. *(Suite)*

L'écart le plus important a été observé sur la programmation batch, que l'on jugeait facile.

12.13 ÉNONCÉ DE LA CAPITALISATION D'EXPÉRIENCE DU PROJET PARKING

Vous allez capitaliser l'expérience issue de ce projet.

1. Faites le suivi du poids des unités d'œuvre (paramètres) et de la valeur des ratios appliqués. Le suivi des unités d'œuvre se présentera comme à la figure 12.33. Le suivi des ratios se fera ainsi selon la figure 12.34.
2. Terminez par une fiche récapitulative du projet. Celle-ci comprendra un bref résumé du projet ainsi que les caractéristiques de son déroulement.

Suivi des unités d'œuvre du projet					
Type tâche	**Valeur de l'unité d'œuvre**	**Nombre d'unités d'œuvre**	**Charge moyenne constatée**	**Écart**	**Écart-type**
Programme moyen temps réel					
Programme temps réel difficile					
Programme batch facile					

Figure 12.33 — Modèle de suivi des unités d'œuvre.

Suivi des ratios du projet						
Type tâche	**Charge estimée de programmation**	**Ratio standard**	**Charge réelle de programmation**	**Charge constatée du type de tâche**	**Ratio constaté**	**Écart**
Jeu essai						
Étude technique						
Test						

Figure 12.34 — Modèle de suivi des ratios.

12.14 CORRIGÉ DE LA CAPITALISATION D'EXPÉRIENCE DU PROJET PARKING

Le suivi des unités d'œuvre fait apparaître les écarts donnés à la figure 12.35.

Pour les *programmes temps réel de difficulté moyenne*, nous ne disposons pas du détail de charge constatée pour chacun des quinze programmes : dix dans le lot Parking et cinq dans le lot Véhicule. En effet, on a fait une tâche unique pour l'ensemble des programmes de chaque lot. On calcule donc la moyenne entre les deux tâches :

Charge moyenne constatée = 50/15 = 3,3 jours.

L'écart-type mesure la distance à la moyenne. Ici, nous n'avons que deux valeurs. Pour le lot *Parking*, la durée moyenne pour le lot est de : 38/10 = 3,8 jours. Pour le lot *Véhicule*, la durée moyenne pour le lot est de : 12/5 = 2,4 jours. Donc un écart important entre les deux. La dispersion par rapport à la moyenne est de :

$$\text{Écart-type} = [10(3,8 - 3,3)^2 + 5(2,4 - 3,3)^2]^{1/2} / 15$$
$$= 0,17.$$

Pour les *programmes temps réel difficiles*, l'écart-type n'a pas de sens puisque nous n'avons qu'une seule valeur, celle du lot *Autorisation*.

Pour les *éditions batch*, nous n'avions au départ qu'une seule tâche comprenant les dix programmes. Mais la répartition sur plusieurs personnes au cours du déroulement du projet (voir paragraphes 12.8 et 12.9) donne une visibilité plus fine.

On a les chiffres suivants : Claude a fait six programmes en deux fois, Camille deux et le chef de projet deux.

D'après les récapitulatifs mensuels des mois 3 et 4, nous avons les valeurs suivantes :

- Claude au mois 3 : 2 programmes en 3 jours, soit 1,5 jour en moyenne.
- Claude au mois 4 : 4 programmes en 13 jours, soit 3,25 jours en moyenne.
- Camille au mois 4 : 2 programmes en 4,5 jours, soit 2,25 jours en moyenne.
- Chef de projet au mois 4 : 2 programmes en 3 jours, soit 1,5 jours en moyenne.

La charge moyenne constatée est de 23,5/10 = 2,4 jours (après arrondi).

L'écart entre les moyennes est important, puisqu'elles vont de 1,5 à 3,25. La dispersion est mesurée par :

$$\text{écart-type} = [2(1,5 - 2,4)^2 + 4(3,25 - 2,4)^2 + 2(2,25 - 2,4)^2 + 2(3 - 2,4)^2]^{1/2} / 10 = 0,23.$$

Type tâche	Valeur de l'unité d'œuvre en jours-personne	Nombre d'unités d'œuvre	Charge moyenne constatée	Écart en jours/ homme	Écart -type
Programme moyen temps réel	3	15	3,3	0,3	0,66
Programme temps réel difficile	5	5	5,2	0,2	
Programme batch facile	1,5	10	2,4	0,9	0,23

Figure 12.35 — Suivi des unités d'œuvre du projet Parking.

Le suivi des ratios (figure 12.36) fait apparaître des écarts faibles (un ou deux points) : cela signifie qu'il y a effectivement une relation de proportionnalité forte entre la charge de programmation et celle des charges périphériques.

Type tâche	Charge estimée de programmation	Ratio standard	Charge réelle de programmation	Charge constatée du type de tâche	Ratio constaté	Écart
Jeu essai	85	20	99,5	18	18	2
Étude technique	85	10	99,5	9	9	1
Test	85	10	99,5	11	11	1

Figure 12.36 — Suivi des ratios du projet Parking.

La fiche récapitulative de projet se présente comme illustrée à la figure 12.37.

Projet Parking
Déroulement : 2/1/96 au 22/4/96 *Nature du projet* : Il s'agissait d'un projet de réalisation. L'application permet de gérer l'attribution de places de parking, en prenant en compte des règles spécifiques. Pour faciliter l'évolution du système, les règles ont été externalisées et peuvent être mises à jour directement par les gestionnaires. De nombreux états de gestion ont été fournis. Le développement s'est fait dans un environnement AS400. *Caractéristiques* : Le dossier de spécification était correctement documenté et toutes les maquettes fournies. *Remarque* : L'écart de charge sur les programmes d'édition a deux causes. D'une part, la prise en main du nouveau générateur d'états n'a pas été immédiate. D'autre part, ces éditions ont été en grande partie confiées à un débutant, qui a mis beaucoup plus de temps que prévu. La charge moyenne de 2,4 jours au lieu des 1,5 prévus s'explique ainsi.

Figure 12.37 — Fiche récapitulative du projet Parking.

Les trois éléments de capitalisation du savoir-faire (suivi des unités d'œuvre, suivi des ratios et fiche récapitulative) viennent alimenter la base d'expériences des participants, notamment si cette dernière a fait l'objet d'une informatisation. Ils enrichissent et affinent les évaluations de charge qui seront faites ultérieurement.

12.15 LA MÉTHODE DE LA VALEUR ACQUISE

12.15.1 Énoncé de l'exercice sur la valeur acquise

Soit le projet de la figure 12.38.

Tâches	Prédécesseur	Durée (semaines)	Coût planifié (en K€)
A		2	5
B	A	6	12
C	A	5	10
D	C	3	9
			36

Figure 12.38 — Tâches du projet à suivre.

Ce qui donne le réseau des antécédents de la figure 12.39.

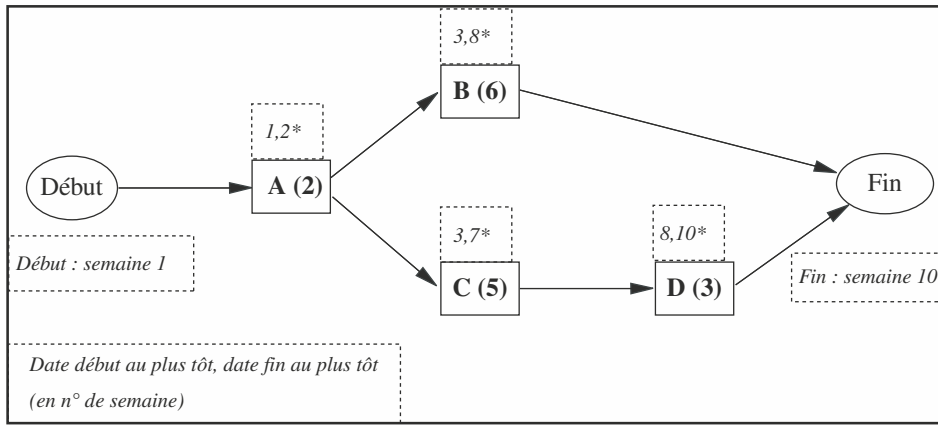


Figure 12.39 — Réseau du projet à suivre.

1) Calculer la **VP en fin de chaque semaine**, en supposant une répartition proportionnelle des coûts en fonction du temps. On indiquera pour chaque tâche le coût généré chaque semaine. Dans une ligne Total on calculera la VP du projet, c'est-à-dire les coûts cumulés en fin de chaque semaine.

2) Les coûts réels et le pourcentage d'achèvement à la fin de la semaine 5 sont donnés à la figure 12.40. Calculer le **CR et la VA à la fin de chaque semaine**, ainsi que l'**écart de délai et l'écart de coût** (VC :*Value cost*).

Tâche		1	2	3	4	5
A	*réel* *% achevé*	3 40 %	2 100 %			
B	*réel* *% achevé*			1 25 %	4 35 %	2 55 %
C	*réel* *% achevé*			2 20 %	2 40 %	2 50 %

Figure 12.40 — Avancement réel du projet à suivre.

3) Calculer, en fin de semaine 5 :

- l'indice de performance des coûts (**IPC**),
- l'indice de performance des délais (**IPD**),
- l'indice de performance des coûts cumulé (**IPCC**)
- l'indice de performance requis (**IPR**) pour respecter l'enveloppe budgétaire.

- la **PAA** (prévision à l'achèvement), d'abord si les écarts sont accidentels, puis si les écarts sont représentatifs du futur.

12.15.2 Corrigé de l'exercice sur la valeur acquise

Le corrigé de la VP pour l'ensemble du projet est donné au tableau de la figure 12.41.

Semaine – Tâche	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
A	2,5	2,5								
B			2	2	2	2	2	2		
C			2	2	2	2	2			
D								3	3	3
VP du projet (K€)	**2,5**	**5**	**9**	**13**	**17**	**21**	**25**	**30**	**33**	**36**

Figure 12.41 — Coût du travail planifié pour l'ensemble du projet.

Le corrigé des indicateurs économiques jusqu'à la fin de la semaine 5 est donné au tableau de la figure 12.42. Pour la VA, on a indiqué la formule de calcul.

	1	**2**	**3**	**4**	**5**
A (5 K€) ***planifié*** ***réel*** ***% achevé***	 2,5 **3** 40 %	 2,5 **2** 100 % (+ 60 %)			
B (12 K€) ***planifié*** ***réel*** ***% achevé***			 2 **1** 25 %	 2 **4** 35 % (+ 10 %)	 2 **2** 55 % (+ 20 %)
C (10K€) ***planifié*** ***réel*** ***% achevé***			 2 **2** 20 %	 2 **2** 40 % (+ 20 %)	 2 **2** 50 % (+ 10 %)

Figure 12.42 — Indicateurs économiques du projet à la semaine 5.

	1	2	3	4	5
CR ***ce qu'on a dépensé***	**3**	**5**	**8**	**14**	**18**
VA ***ce qu'on aurait dû dépenser pour le travail achevé***	**2** 40 %.5	**5** 100 %.5	**10** 5 + 25 %.12 + 20 %.10	**13,2** 10 + 10 %.12 + 20 %. 10	**16,6** 13,2 + 20 %.12 + 10 %.10
VP ***ce qu'on a budgété et planifié de faire***	2,5	5	9	13	17
Écart de délai : **VA – VP**	**– 0,5** (retard)	**0**	**1** (avance)	**0,2** (avance)	**– 0,4** (retard)
Écart de coût : **VA – CR**	**– 1** sur-consom-mation	**0**	**2** sur-consom-mation	**– 0,8** sur-consom-mation	**– 1,4** sur-consom-mation

Figure 12.42 — Indicateurs économiques du projet à la semaine 5. *(Suite)*

En fin de semaine 5, nous obtenons les valeurs suivantes :

- Indice de performance des coûts (**IPC**) = VA/CR = 16,6/18 = **0,92**
- Indice de performance des délais (**IPD**) = VA/VP = 16,6/17 = **0,98**
- Indice de performance des coûts cumulés (**IPCC**) = (Σ VA)/(Σ CR) = (2 + 5 + 10 + 13,2 + 16,6)/(3 + 5 + 8 + 14 + 18) = 46,8 / 48 = **0,98**
- Indice de performance requis (**IPR**) = Travail restant/Budget restant = (45 % B) + (50 % C) + D/(36 – 18) = 5,4 + 5 + 9 / 18 = 19,4 / 18 = **1,08**

La prévision à l'achèvement se présente comme suit.

- Si les écarts sont accidentels,
  PAA = Coûts réels + Travail restant = 18 + 19,4 = **37,4**
- Si les écarts sont représentatifs du futur,
  PAA = Coûts réels + Travail restant/IPC = 18 + (19,4/0,92) = 18 + 21,09 = **39,09**
- ou bien avec l'indice de performance des coûts cumulé :
  PAA = Coûts réels + Travail restant/IPC^c = 18 + (19,4/0,98) = 18 + 19,8 = **37,8**

12.16 LA VALEUR MONÉTAIRE ATTENDUE

12.16.1 Énoncé de la VMA

Vous êtes attentif aux coûts de votre projet, mais vous avez aussi des contraintes de délai (intéressement/pénalités en cas d'avance/retard). Vous êtes obligés de faire appel à une équipe extérieure que vous payez proportionnellement au temps travaillé et au profil des ressources (régie simple). Vous hésitez entre trois équipes sur lesquelles vous avez les renseignements suivants.

Équipe CapMidi

Le prix annoncé est de 9000. Ce prestataire a l'habitude de terminer dans les temps, mais de facturer plus cher que le prix annoncé, en affectant des ressources plus qualifiées, mais plus chères. La probabilité qu'il respecte le budget est de 10 %. Il a cependant 20 % de chances d'augmenter le prix de 15 %, 40 % de l'augmenter de 30 % et 30 % de l'augmenter de 50 %.

Équipe InfoCom

Le prix annoncé est de 12 000. Le prix sera respecté, mais ces intervenants peuvent être en retard. La probabilité qu'ils respectent le délai est de 30 %. Ils ont 20 % de chances d'être en retard de 20 jours, ce qui représente pour vous une pénalité de 2000 pour votre projet, et ils ont 50 % de chances d'être en retard de 10 jours, ce qui correspond à une pénalité de 1000.

Équipe Dafné

Le prix annoncé est de 13 000. Le prix annoncé sera respecté, mais ce prestataire peut aussi bien être en avance qu'en retard. Plus précisément, il a 60 % de chances d'être dans les délais, 20 % de chances d'avoir un léger retard correspondant pour vous à une pénalité de 500, et 20 % de chance d'être en avance ce qui vous procurerait un intéressement de 800.

Dressez un arbre de décision pour montrer les raisons du choix que vous allez faire en vous appuyant sur la valeur monétaire attendue de chaque scénario.

12.16.2 Corrigé de la VMA

Le tableau de la figure 12.43 récapitule le calcul de la valeur du scénario attaché à chacun des trois fournisseurs potentiels. Nous voyons que le choix devrait se porter sur CapMidi qui probablement coûtera le moins cher.

Scénario	Dépenses	Probabilité gain ou perte	Montant gain ou perte	Probabilité * Montant gain ou perte	VMA
Équipe CapMidi		10 %	0		
		20 %	– 1 350	– 270	
		40 %	– 2 700	– 1 080	
		30 %	– 4 500	– 1 350	
	– 9 000			**– 2 700**	**– 11 700**
Équipe InfoCom		30 %	0		
		20 %	– 2 000	– 400	
		50 %	– 1 000	– 500	
	– 12 000			**– 900**	**– 12 900**
Équipe Dafné		60 %	0		
		20 %	– 500	– 100	
		20 %	800	160	
	– 13 000			**60**	**– 12 940**

Figure 12.43 — Calcul de la VMA

12.17 LE CALCUL DE RENTABILITÉ

12.17.1 Énoncé des indicateurs du ROI

Soit à évaluer la rentabilité d'un projet ERP (SAP/R3) dans une entreprise internationale (France, Belgique, Allemagne et Suède). Le périmètre du projet comprend quatre modules : SD (*Sales and Distribution*), FI (*Financial Accounting*), PP (*Production planning*) et MM (*Material management*). Les objectifs financiers de l'implémentation sont les suivants. On cherche à réduire les coûts du service Administration des ventes (automatisation, réduction du besoin de coordination personnelle) par la suppression de trois postes de travail et la diminution des

besoins de transport. On veut aussi réduire les délais de paiement par les clients et passer de 75 à 70 j. On va revoir les procédures de la fonction Comptabilité (automatisation, disponibilité de l'information), ce qui devrait diminuer le nombre de réunions internationales. On vise enfin de réduire le volume des stocks de 40 % en moyenne sur l'année et d'augmenter leur rotation des stocks en passant de 30 jours à 15 jours.

Les estimations (en K€) sont données au tableau de la figure 12.44 :

Année	Dépenses		Gains
1	Modules SD, MM et FI	700	0
2	Assistance externe Redevance SAP	1 400	300
3	Assistance externe Module PP Redevance SAP	1 300	1 000
4	Assistance externe Redevance SAP	1 150	2 000
5	Redevance SAP Maintenance	950	2 500
6	Redevance SAP Maintenance	900	3 000
7	Redevance SAP Maintenance	1 000	3 000
8	Redevance SAP Maintenance	1 500	3 000
Total		8 900	14 800

Figure 12.44 — Estimations pour le calcul du ROI

En supposant un taux actualisation de 4,5 % et une durée de vie de l'investissement de 8 ans, calculer le délai de remboursement, les gains et dépenses actualisés, la VAN et le TRI.

Si l'on utilise un tableau Excel, les fonctions se paramètrent de la façon suivante :

- VAN (taux % ; cellule début Cash-flow ; cellule fin Cash-flow).
- TRI (cellule début Cash-flow ; cellule fin Cash-flow).

12.17.2 Corrigé des indicateurs du ROI

Le délai de remboursement se situe entre l'année 4 et l'année 5 (figure 12.45). Les chiffres donnés au tableau de la figure 12.46 montrent que la valeur actuelle nette est de 4154, ce qui correspond à taux de rentabilité interne de 37 %.

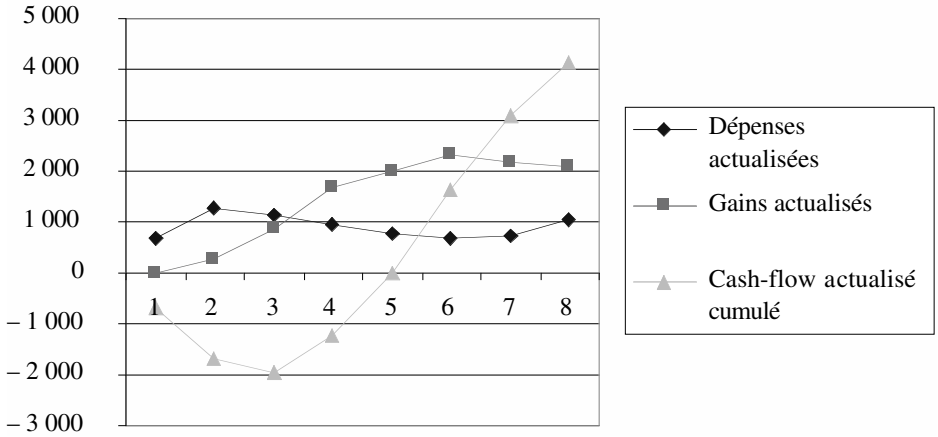


Figure 12.45 — Délai de remboursement

Année	1	2	3	4	5	6	7	8
Dépenses	700	1 400	1 300	1 150	950	900	1 000	1 500
Gains	0	300	1 000	2 000	2 500	3 000	3 000	3 000
Cash-flow (gains-dépenses)	– 700	– 1 100	– 300	850	1 550	2 100	2 000	1 500
Cash-flow cumulé	– 700	– 1 800	– 2 100	– 1 250	300	2 400	4 400	5 900
Cash-flow actualisé	– 670	– 1 007	– 263	713	1 244	1 613	1 470	1 055
Cash-flow actualisé cumulé	– 670	– 1 677	– 1 940	– 1 227	17	1 629	3 099	**4 154**

Figure 12.46 — Calcul du ROI

Ainsi s'achève la mise en pratique du pilotage d'un projet, avec le dispositif de suivi, le calcul de la performance et une illustration de quelques techniques d'aide aux décisions d'ordre financier.

13

La dimension relationnelle des projets

Nous allons illustrer la dimension relationnelle des projets sous deux angles. Le premier concerne le management de l'équipe de projet. Le second présente les difficultés que l'on peut rencontrer avec d'autres acteurs de l'entreprise.

13.1 LE MANAGEMENT D'ÉQUIPE

Vous êtes confronté à différentes situations qui appellent une prise de décision. Différents comportements vous sont proposés.

Il vous est demandé :

- d'indiquer à quel style de management correspond chacune des conduites ;
- de présenter les avantages et les inconvénients de chaque choix ;
- d'éventuellement conseiller une solution, si elle vous semble préférable ;
- et d'éventuellement déconseiller une des solutions si elle vous paraît trop risquée ou inadéquate.

13.1.1 Énoncé de l'affectation de nouvelles ressources

Bernard et Jean, ayant environ deux ans d'expérience, viennent d'être recrutés et sont affectés à votre projet. Vous hésitez quant aux tâches à leur confier. Bernard est « fana » de technique, mais il vous semble important qu'il s'implique

également dans les jeux d'essais et dans les séances de validation avec les utilisateurs. Jean est polyvalent, mais un peu timide.

1. Vous dialoguez avec Jean et Bernard pour leur montrer l'importance d'une activité diversifiée pour le projet comme pour leur propre évolution de carrière.
2. Vous leur confiez deux lots complets avec une date de fin des deux lots et une charge estimée pour chaque tâche, charge à eux de se répartir le travail et de vous donner le planning détaillé le lendemain.
3. Vous les rencontrez chacun séparément pour leur demander leur avis. Si Bernard reste déterminé à ne faire que du développement, vous effectuez une répartition spécialisée des tâches.
4. Vous confiez à chacun un lot complet et vous demandez à Henri, un membre de votre équipe qui a une grande expérience, de prévoir une demi-journée de formation pour Bernard.

13.1.2 Corrigé de l'affectation de nouvelles ressources

La qualification des attitudes est donnée à la figure 13.1.

	Style	Avantages	Risque ou inconvénients
1	Persuasif	Favorise l'intégration des nouveaux. *Conseillé*	Pas de décision.
2	Par délégation		Dangereux pour le projet. Risque de conflits entre Bernard et Jean. *Déconseillé*
3	Participatif individuel	Augmente la satisfaction individuelle	Baisse de la polyvalence de l'équipe. Réduction de souplesse pour le projet.
4	Directif seul	Décision rapide.	Ne favorise pas l'adhésion au projet.

Figure 13.1 — Styles de management pour l'affectation des ressources.

13.1.3 Énoncé d'un problème avec un groupe de validation

Vous avez adopté une démarche RAD, et le projet se trouve à l'étape de construction. Charles, responsable du premier lot, vient d'avoir la première session de validation de prototype. L'équipe de validation est composée de cinq personnes : une n'est pas venue, deux ont passé une partie de la session à traiter un problème

opérationnel sans rapport avec leur prototype. Seules deux personnes sur cinq ont vraiment participé à la session. La prochaine doit avoir lieu dans deux semaines.

1. Vous organisez une réunion avec l'ensemble de l'équipe de projet (hors utilisateurs) et vous recherchez ensemble des moyens d'agir.
2. Vous allez voir chaque personne du groupe de validation, accompagné du chef de projet utilisateur, pour leur expliquer à nouveau l'importance du projet et de leur rôle.
3. Vous expliquez à Charles qu'il doit être plus ferme. Vous lui conseillez d'envoyer un mail de rappel au groupe de validation quelques jours avant la prochaine session.
4. Vous faites le point avec Charles et avec le chef de projet utilisateur, puis vous demandez au président du Comité de pilotage d'organiser une réunion avec le groupe de validation et le chef de projet utilisateur. Au cours de cette réunion, vous rappelez que l'objectif de délai et qualité doit être partagé par tous les acteurs du projet et que cela implique un respect des règles du jeu par chacun. Vous insistez sur le fait que le groupe de validation doit être libéré pour participer aux sessions.

13.1.4 Corrigé d'un problème avec un groupe de validation

La qualification des attitudes est donnée à la figure 13.2.

	Style	Avantages	Risque ou inconvénients
1	Participatif groupe		Ne touche pas les acteurs concernés (utilisateurs). Risque de dresser l'équipe de projet contre les utilisateurs. *Déconseillé*
2	Persuasif	Évite de créer des perturbations chez la maîtrise d'ouvrage.	Résultat incertain si les contraintes opérationnelles du groupe de validation sont fortes.
3	Par délégation		Risque de déstabiliser Charles. Résultat peu garanti. *Déconseillé*
4	Directif après information	Les règles ont besoin d'être rappelées aussi bien à l'équipe de validation qu'au Comité de pilotage. *Conseillé*	

Figure 13.2 — Styles de management pour un problème avec un groupe de validation.

13.1.5 Énoncé d'un retard sur un sous-projet

Votre projet comporte quatre sous-projets ayant chacun un responsable. Vous avez une réunion d'avancement tous les lundis matins. Vendredi après-midi, vous prenez connaissance du tableau de bord hebdomadaire qui vient de vous parvenir et vous constatez un retard important du sous-projet dirigé par Dominique. Jusque-là, vous n'avez connu aucun problème majeur.

1. Vous attendez la réunion de lundi pour que Dominique vous explique les raisons du retard et les actions qu'elle va mettre en œuvre.
2. Vous appelez immédiatement Dominique pour la rencontrer avec son équipe afin d'analyser le retard et de mettre en place des mesures de redressement.
3. Vous convoquez les quatre responsables de sous-projets pour une réunion exceptionnelle et vous recueillez les explications de Dominique et les avis des autres, en vue de prendre des mesures que vous annoncerez lundi matin.
4. Vous envoyez un mail à Dominique pour lui rappeler l'importance des délais et lui demander de préparer des propositions d'action pour la réunion de lundi.

13.1.6 Corrigé d'un retard sur un sous-projet

La qualification des attitudes est donnée à la figure 13.3.

	Style	Avantages	Risque ou inconvénients
1	Par délégation	Pas de dramatisation. Mise sur la confiance. *Conseillé*	Retard à réagir si Dominique n'a rien préparé.
2	Participatif groupe		Dominique est mise en difficulté devant son équipe. Risque d'une réaction de bloc de l'équipe de sous-projet. *Déconseillé*
3	Consultatif groupe + Directif seul		Brutal et éprouvant pour Dominique. Risque de démotivation de l'équipe de Dominique. Risque de mauvais climat dans l'équipe de projet. *Déconseillé*
4	Persuasif	Dominique sera préparée et aura eu le temps de réfléchir, éventuellement avec son équipe.	

Figure 13.3 — Styles de management pour un retard sur un sous-projet.

13.1.7 Énoncé d'un retrait de ressources

Votre projet comporte quatre sous-projets ayant chacun un responsable. Le Directeur informatique vient de vous informer que deux personnes (travaillant chacune sur un sous-projet) sont retirées immédiatement du projet à cause d'une urgence sur un projet prioritaire. La date d'achèvement de votre projet (dans un mois) est maintenue.

1. Vous réfléchissez à différentes solutions, puis vous convoquez l'ensemble de l'équipe pour décider de la meilleure solution.
2. Vous convoquez les responsables de sous-projet. Vous leur expliquez pourquoi l'autre projet est prioritaire. Vous leur demandez de trouver une solution pour le lendemain.
3. Vous demandez leur avis à chaque responsable de projet et vous cherchez la solution la moins perturbatrice que vous exposez le lendemain.
4. Vous rencontrez successivement les responsables des deux sous-projets touchés et vous cherchez avec eux une solution acceptable.

13.1.8 Corrigé d'un retrait de ressources

La qualification des attitudes est donnée à la figure 13.4.

	Style	Avantages	Risque ou inconvénients
1	Participatif groupe		Court-circuite les responsables de sous-projet. Risque de déstabiliser l'équipe de projet. *Déconseillé*
2	Persuasif + Par délégation	Fait confiance aux responsables de sous-projet.	Risque de conflit entre les responsables de sous-projet. Risque de retard accru pour le projet.
3	Consultatif + Directif seul	Rapidité de la décision. Évite des conflits entre les responsables de sous-projet. *Conseillé*	
4	Participatif individuel	Évite de perturber deux des quatre sous-projets.	Besoins d'allers-retours entre les deux responsables. Il est peut-être nécessaire d'utiliser certaines ressources des sous-projets non touchés.

Figure 13.4 — Styles de management pour un retrait de ressources.

13.1.9 Énoncé de la mise en place d'un intranet pour le projet

Le projet a démarré depuis peu et vous allez mettre un intranet, déjà utilisé sur d'autres projets, pour favoriser la communication et faciliter le suivi du projet.

1. Vous étudiez la façon dont l'intranet est utilisé sur les autres projets et vous définissez les procédures et fonctions les plus utiles pour améliorer l'efficacité du projet.
2. Vous demandez à un des autres chefs de projet utilisant déjà l'intranet de faire une présentation devant l'ensemble de l'équipe. Puis vous organisez une réunion avec votre équipe pour décider des modalités d'utilisation de l'intranet.
3. Vous chargez un membre expérimenté de l'équipe, avec lequel vous avez déjà travaillé, de mettre en place l'intranet dans un délai d'une semaine et de prévoir une information/formation à l'ensemble de l'équipe.
4. Vous expliquez à l'ensemble de l'équipe l'intérêt d'utiliser un intranet sur le projet et vous constituez un groupe de volontaires pour proposer une utilisation.

13.1.10 Corrigé de la mise en place d'un intranet pour le projet

La qualification des attitudes est donnée à la figure 13.5.

	Style	Avantages	Risque ou inconvénients
1	Directif après information	Cohérent avec les choix de gestion du chef de projet	Peut être mal accepté si l'équipe n'adhère pas pleinement au projet ou si l'intranet est contraignant.
2	Participatif groupe	Facilite l'utilisation ultérieure. *Conseillé*	
3	Par délégation	Ne disperse pas les énergies.	Risque de rejet par le reste de l'équipe.
4	Persuasif + Par délégation	Responsabilise les volontaires.	Risque de dispersion de l'équipe. Risque de retard. *Déconseillé*

Figure 13.5 — Styles de management pour la mise en place d'un intranet pour le projet.

13.2 LES STRATÉGIES DE RÉSOLUTION DES CONFLITS

Vous êtes engagés dans les cinq conflits suivants. On vous propose, chaque fois, deux comportements à adopter. Indiquez, pour chaque conflit et pour chaque comportement proposé :

1. La stratégie de résolution correspondante.
2. Les avantages et les risques de cette stratégie.

13.2.1 Énoncé d'un conflit avec le maître d'ouvrage

Le projet a commencé depuis quatre mois. Un nouveau maître d'ouvrage vient d'être nommé, en remplacement de celui avec lequel vous avez planifié un développement en spirale, qui a été appelé à d'autres fonctions.

Le nouveau maître d'ouvrage remet en question les priorités de planification des fonctions, ce qui a des impacts sur le travail en cours et vous oblige en particulier à revoir votre planification et l'attribution des tâches.

Comportement 1 : Vous évitez le maître d'ouvrage. Le prochain Comité de pilotage a lieu dans un mois. Vous espérez que d'ici là, le maître d'ouvrage sera devenu raisonnable, ou bien que votre directeur informatique se montrera intransigeant.

Comportement 2 : Vous faites le point avec votre équipe. Vous analysez le travail déjà commencé sur les modules à repousser, ainsi que les compétences des personnes disponibles. Vous acceptez de commencer très rapidement le module 4 pour lequel vous disposez des bons profils. En revanche vous ne cédez pas sur le module 6 : il sera fait dans la version 2. Le maître d'ouvrage vous demande d'intégrer une fonction du module 6 dans le module 4. Vous finissez par accepter à contrecœur.

13.2.2 Corrigé d'un conflit avec le maître d'ouvrage

La qualification des stratégies est donnée à la figure 13.6.

	Stratégie	Avantages	Risques
1	Retrait + Force	Ne pas heurter de front le maître d'ouvrage.	À terme, pas de relation de confiance entre le chef de projet et le MOA. Risque de replanification *in extremis* si le MOA impose son point de vue en Comité de pilotage.
2	Compromis	Le chef de projet amorce une relation de négociation ouverte avec le MOA. L'équipe est engagée dans la recherche d'une solution.	Le chef de projet a accepté des modifications de planification qui touchent le découpage en modules : risques sur la qualité.

Figure 13.6 — Stratégies face à un conflit avec le maître d'ouvrage.

13.2.3 Énoncé d'un conflit avec le responsable qualité

Vous allez bientôt terminer l'analyse du projet. Vous avez utilisé différents modèles pour représenter la partie statique et la partie dynamique des traitements. Le responsable qualité vient de mettre en vigueur de nouvelles procédures d'assurance qualité : il vous demande de rajouter dans votre dossier d'analyse les modèles correspondant à la description des traitements sans les choix d'organisation, ce que vous trouvez superflu.

Comportement 1 : Vous rencontrez le responsable qualité. Vous le questionnez sur les nouvelles procédures, vous lui montrez qu'elles vont dans le sens de vos pratiques, comme en témoigne le PAQ de votre projet. Vous arrivez à le convaincre que les modèles supplémentaires seront rajoutés après la livraison du dossier d'analyse.

Comportement 2 : Vous organisez une réunion avec le responsable qualité et toute votre équipe. Vous demandez au responsable qualité d'exposer les principes et le contenu des nouvelles procédures. Vous recherchez ensemble une solution à moindre coût pour mettre à niveau les dossiers.

13.2.4 Corrigé d'un conflit avec le responsable qualité

La qualification des stratégies est donnée à la figure 13.7.

	Stratégie	Avantages	Risques
1	Apaisement + Compromis	Le chef de projet préserve ses engagements. Pas de conflit ouvert : l'équipe peut continuer à travailler	Le problème est renvoyé après la livraison : s'il n'est pas traité, risque d'augmentation du conflit.
2	Collaboration	Établissement de bonnes relations avec le responsable qualité.	Risque de dérive du projet, dont le chef de projet sera tenu pour responsable.

Figure 13.7 — Stratégies face à un conflit avec le responsable qualité.

13.2.5 Énoncé d'un conflit avec un autre chef de projet

Bernard est affecté à temps partiel sur votre projet, et à 50 % sur un projet de Datawarehouse qui doit se terminer dans un mois. Le partage se fait par demi-semaine. L'autre projet a pris du retard et depuis deux semaines, Bernard n'a travaillé que deux jours sur votre projet (au lieu de 5).

Comportement 1 : Vous espérez que cela va aller mieux à l'issue du projet Datawarehouse.

Comportement 2 : Vous allez voir votre maître d'ouvrage. Vous lui exposez la dérive dans laquelle le projet risque de s'engager. Vous obtenez de lui qu'il intervienne auprès du directeur informatique pour que Bernard intervienne conformément aux engagements ou bien qu'il soit remplacé.

13.2.6 Corrigé d'un conflit avec un autre chef de projet

La qualification des stratégies est donnée à la figure 13.8.

	Stratégie	Avantages	Risques
1	Retrait	Pas de conflit entre collègues.	Le problème peut ne pas s'arranger si rapidement et le projet va prendre un retard important.
2	Force	Le conflit se fait par personnes interposées.	Le conflit risque d'augmenter si la ressource est retirée de l'autre projet.

Figure 13.8 — Stratégies face à un conflit avec un autre chef de projet.

13.2.7 Énoncé d'un conflit avec le responsable technique

Vous avez commencé l'étude technique, et le responsable technique attaché à la Direction informatique vous annonce que tous les projets qui devaient être développés dans un environnement Windows NT4 doivent maintenant se faire dans l'environnement Windows NT5. Il vous assure que cela ne posera pas de problèmes.

Comportement 1 : Vous allez voir le directeur informatique et lui exposez les risques et perturbations générés. Vous obtenez une ressource support à mi-temps pendant un mois pour aider votre équipe à faire le passage.

Comportement 2 : Vous annoncez à votre équipe que votre projet n'est pas concerné par le changement d'environnement. Parallèlement, vous essayez de négocier le report du changement pour la version 2.

13.2.8 Corrigé d'un conflit avec le responsable technique

La qualification des stratégies est donnée à la figure 13.9.

	Stratégie	Avantages	Risques
1	Force + Compromis	Le chef de projet se décharge de la responsabilité d'un retard. Le chef de projet préserve ses relations avec le responsable technique.	La charge de support a mal été évaluée.
2	Retrait + Compromis	L'équipe est tenue à distance des perturbations. Le projet n'est pas impacté directement.	La négociation peut échouer. Le changement de système peut être imposé par la direction et perturber fortement le projet.

Figure 13.9 — Stratégies face à un conflit avec le responsable technique.

13.2.9 Énoncé d'un conflit avec le chef de projet utilisateur

Vous avez commencé l'étude technique, et le responsable technique Le chef de projet a organisé, à votre insu, une réunion avec les utilisateurs, au cours de laquelle certaines options de conception ont été discutées et légèrement modifiées. Vous n'avez pas apprécié lorsque vous l'avez appris.

Comportement 1 : Après quelques jours de réflexion, vous rencontrez le chef de projet utilisateur et vous essayez de comprendre les raisons de sa démarche,

ainsi que la pertinence des modifications apportées. Vous les annoncez ensemble à l'ensemble de l'équipe.

Comportement 2 : Vous jugez les changements peu importants. Pour le principe, vous n'en acceptez que la moitié, en invoquant diverses raisons techniques.

13.2.10 Corrigé d'un conflit avec le chef de projet utilisateur

La qualification des stratégies est donnée à la figure 13.10.

	Stratégie	Avantages	Risques
1	Retrait + Collaboration	Préservation des relations avec le chef de projet utilisateur.	Le chef de projet utilisateur risque de faire à nouveau cavalier seul.
2	Compromis	Message envoyé au chef de projet utilisateur : les initiatives individuelles ne doivent pas se prendre au détriment du travail d'équipe.	Le chef de projet utilisateur est insatisfait et la relation se dégrade.

Figure 13.10 — Stratégies face à un conflit avec le chef de projet utilisateur.

14

Le plan qualité d'un projet

14.1 LE CAS MÉCANO

Le cas Mécano a été présenté au chapitre 9. Nous en étudions ici les aspects concernant la maîtrise de la qualité.

Vous ferez un certain nombre d'hypothèses et proposerez un plan qualité. Vous distinguerez deux parties :

1. La qualité souhaitée du futur système à l'aide des facteurs et critères qualité.
2. Les dispositions préconisées pour donner à votre client l'assurance de la qualité du processus que vous allez conduire.

Vous prendrez comme base la stratégie adoptée suite à l'analyse du projet et de ses risques, qui figure au paragraphe 9.1.5. Votre plan qualité peut être structuré en deux parties.

- *Première partie* : la qualité du produit

  Cette partie est faite en liaison étroite avec le client. Elle lui permet d'expliciter les caractéristiques sur lesquelles il jugera l'application. Vous choisirez parmi les facteurs et critères de Boehm-McCall ceux qui vous paraissent pertinents et vraisemblables dans le contexte du projet. Leur nombre doit être limité et ils ne doivent pas être contradictoires.

- *Deuxième partie* : la qualité du processus

Cette deuxième partie, liée au plan de management de projet, comprendra les éléments suivants :

– responsabilités qualité ;
– documents de référence ;
– organisation ;
– démarche de développement ;
– document de suivi de projet ;
– plan de contrôle ;
– mise en œuvre.

14.2 CORRIGÉ DU PLAN QUALITÉ DU CAS MÉCANO

Un exemple de plan qualité pour le projet Mécano est donné ci-dessous.

14.2.1 Première partie : la qualité du produit

L'application MAO doit d'une part améliorer l'efficience de la fonction maintenance, d'autre part favoriser une diminution des temps de panne.

Nous attendons qu'elle satisfasse aux exigences qualité suivantes.

D'un point de vue fonctionnel, l'application doit être *pertinente*, c'est-à-dire qu'elle doit être adaptée à la taille de l'entreprise. Son périmètre doit être limité aux fonctions de base des deux processus Suivi d'intervention et Aide à l'intervention.

Le premier comprendra les fonctions suivantes :

- demande d'intervention ;
- planification des interventions ;
- suivi des interventions en cours ;
- solde d'une intervention ;
- statistiques.

Le second se compose également de cinq fonctions :

- gestion des nomenclatures des machines ;
- gestion des plans ;
- gestion des gammes opératoires ;

- gestion des outillages ;
- aide à la recherche de pièces.

Les *métriques* retenues pour ce facteur mesurent la diminution des temps d'intervention.

Le délai avant intervention est actuellement de deux heures, il doit être ramené à une heure pour 95 % des interventions. Le gain sera obtenu par un circuit d'information court entre la production et la maintenance, ainsi que par une planification rigoureuse des interventions.

La durée moyenne d'intervention est de trois heures ; elle doit être réduite à deux heures et demie dans 95 % des cas, sauf indisponibilité des pièces. Le gain sera possible par l'accès aisé aux informations sur les machines, les outillages et les gammes opératoires, ainsi que par la disponibilité immédiate des plans.

L'application doit être *adéquate*, c'est-à-dire qu'elle doit s'adapter aux procédures existantes, dont elle doit augmenter la fluidité. Toutes les décisions sur les opérations de maintenance à mener sont, comme actuellement, du ressort de l'ouvrier de maintenance. On ne souhaite pas renforcer la centralisation des décisions, ni augmenter leur degré d'automatisation. Au contraire, les informations communes doivent être partageables. Les communications entre production et maintenance doivent être facilitées le plus possible. Seules sont centralisées la gestion des données de référence stables et la planification des interventions.

La mesure de l'atteinte de ce facteur sera la proximité des modèles organisationnels de traitements entre l'ancien et le nouveau système.

D'un point de vue utilisation, l'application doit être *maniable*. Les utilisateurs au quotidien sont principalement des ouvriers. Le vocabulaire manipulé dans les interfaces homme-machine doit être celui qui est utilisé couramment.

Les deux critères souhaités sont la *communicabilité* et la *facilité d'apprentissage*. Le premier porte sur l'interface utilisateur, qui doit être très simple d'usage. La métrique sera une mesure d'opinion : un échantillon représentatif d'utilisateurs finaux, utilisant l'application depuis deux mois, sera interrogé. Le deuxième critère sera mesuré par le temps d'apprentissage de la manipulation de l'application : il ne devra pas dépasser une demi-journée pour les ouvriers comme pour les cadres.

La *confidentialité* de l'application doit être assurée de deux façons. Les données de références sont mises à jour par un responsable unique. Les données concernant une intervention ne sont pas modifiables a posteriori, dans le sens que toutes les rectifications sont enregistrées sans effacer les données initiales. Ce facteur sera mesuré par des tests utilisateurs.

L'application devra être *couplée* avec trois applications. Elle se servira des données de l'application Personnel pour l'identification des différents intervenants sur une opération de maintenance. Elle alimentera sans nouvelle saisie l'application de comptabilité analytique. Elle pourra interroger la gestion des stocks, effectuer des réservations et si nécessaire déclencher une commande urgente. L'arrivée des pièces commandées doit donner lieu à un signal dans le module planification des interventions. L'atteinte de ce facteur sera mesurée par des tests de recette.

14.2.2 Deuxième partie : la qualité du processus

En ce qui concerne les *responsabilités qualité*, le chef de projet jouera le rôle de responsable qualité sur le projet. Les différents modules feront l'objet d'une qualification par la section qualité du service informatique. Le client est responsable de la qualité des jeux d'essais de recette.

Les *documents de référence* sont les suivants. Le plan de management de projet a été décrit dans le document MAO-2. L'application respectera les normes de présentation de l'interface utilisateur décrites dans le document INF95-34. La documentation de l'application sera semblable à celle de l'application Gestion des stocks mise en service l'an dernier : GS16. Le manuel utilisateur sera réduit au minimum et remplacé par une aide en ligne, conformément à la norme interne INF92-8.

Pour ce qui est de l'*organisation*, l'équipe de projet comprendra quatre personnes en tout : MM. Alain, Jean et Bon du service informatique, ainsi que M^me^ Legrand du service maintenance.

M. Alain sera chef de projet. MM. Jean et Bon seront concepteurs-développeurs. Le premier sera spécialisé sur l'utilisation du multimédia pour la gestion des plans. Le second sera spécialiste de la base de données objet. Tous deux feront à la fois de la conception et du développement. M^me^ Legrand sera affectée à temps plein sur le projet pendant l'étape de conception générale et à mi-temps ensuite.

Le développement des modules comprendra des phases d'expérimentation d'une durée de cinq jours ouvrables. Les deux concepteurs-développeurs se tiendront à disposition des utilisateurs témoins pour leur apporter de l'aide s'ils le souhaitent. La liste des participants à l'expérimentation sera composée de la façon suivante :

- service maintenance : cinq personnes ;
- production chaîne A : cinq personnes ;
- production chaîne B : cinq personnes.

Il est à noter que ces trois chaînes ont été choisies, en fonction des contraintes de production. Si ces dernières changeaient, le client pourrait mettre à disposition des personnes choisies dans une autre chaîne.

La *démarche de développement* a été décrite dans le plan de management de projet (voir paragraphe 9.1.6). Celui-ci donnera lieu à la livraison des fournitures mentionnées à la figure 14.1.

Point de livraison	Contenu de la fourniture	Validation
Fin de l'étape de conception générale	Rapport de conception : – architecture matérielle – architecture logicielle – modèle de données avec le formalisme entité/relation – modèles organisationnels de traitement (MOT)	Comité de pilotage
Fin de la phase prototypage de chaque cycle de réalisation	Prototype installé sur trois postes en libre service dans la salle B202	

Figure 14.1 — Livraison des fournitures dans le projet Mécano.

Le planning des premières étapes, c'est-à-dire jusqu'à la livraison du premier module figure en annexe. Il sera complété par le calendrier de développement des modules 2 et 3, à la fin de la conception générale.

Le *document de suivi de projet* est structuré selon le modèle de la figure 14.2, avec les indicateurs suivants :

Évolution de charge pour le mois écoulé = temps passé – avancement

Évolution de charge en jours depuis le début du projet =
temps passé + reste à faire à ce jour – charge initiale.

Évolution de charge depuis le début du projet en pourcentage =
évolution de charge en jours × 100/charge initiale.

Achèvement en pourcentage =
(Charge initiale – reste à faire à ce jour) × 100/charge initiale

Il sera fourni tous les lundis matins par le chef de projet au client. Il donnera lieu à une mini-réunion d'une demi-heure.

Tableau d'avancement du projet : *nom* en fin de mois : *mois*											
	Mois n-1		Mois n				Récapitulatif projet				
Lots	Temps passé	Reste à faire	Temps passé	Reste à faire	Avancement	Évolution de charge	Charge initiale	Temps passé	Évolution en jours	Évolution en %	Achèvement en %

Figure 14.2 — Document de suivi du projet Mécano.

Le *plan de contrôle* comporte trois volets :

- Le chef de projet fera une vérification systématique et complète de tous les produits livrés : études, programmes et documentation.
- Les programmes du premier prototype feront l'objet d'une lecture croisée préalable à sa mise à disposition.
- Le client fournira les jeux d'essais de recette qui seront utilisés lors de la production du logiciel.

La *mise en œuvre* sera pilotée par M^me^ Legrand. Elle fera l'objet d'un plan assurance qualité séparé.

Ce plan qualité engage le maître d'œuvre et le maître d'ouvrage. Toute évolution souhaitée par l'une ou l'autre partie devra être négociée et donnera lieu, en cas d'accord, à une nouvelle version.

15

Un exemple d'utilisation de Microsoft Project

15.1 LE CAS PARKING

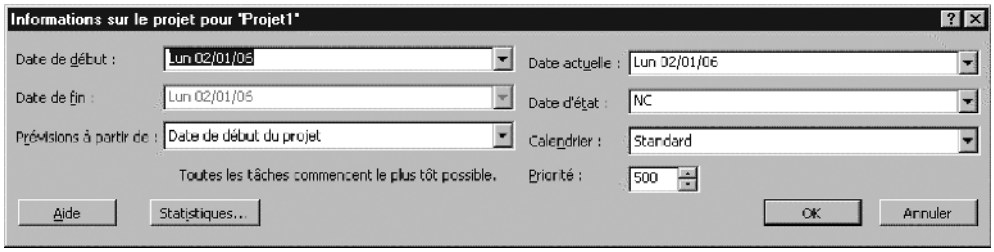
Nous allons reprendre l'énoncé du cas Parking figurant au chapitre 12, qui a permis de simuler le déroulement d'un petit projet. Il va nous servir à donner un exemple de système d'information de pilotage à partir de l'outil Microsoft® Project 2003. Nous n'avons pas cherché à reproduire les indicateurs présentés au chapitre 7 et illustrés au chapitre 12, qui sont destinés à un suivi avec un tableur comme Excel, mais nous proposons un choix parmi ceux proposés par MS Project et nous indiquerons les correspondances.

Le projet se déroule en l'**an 2006** et commence début janvier. Toutes les contraintes de calendriers et de ressources sont identiques à celles qui figurent au chapitre 12 (paragraphe 12.1).

15.2 INITIALISATION DU PROJET PARKING

15.2.1 Création du projet Parking

On va pour commencer, indiquer ses caractéristiques générales : **Projet**, **Informations sur le projet…** On saisit la Date de début : **2/1/06**, et on sélectionne « Prévisions à partir de » : **Date de début du projet**.



On sauvegarde le projet par Fichier, Enregistrer sous : **Parking. mpp**,

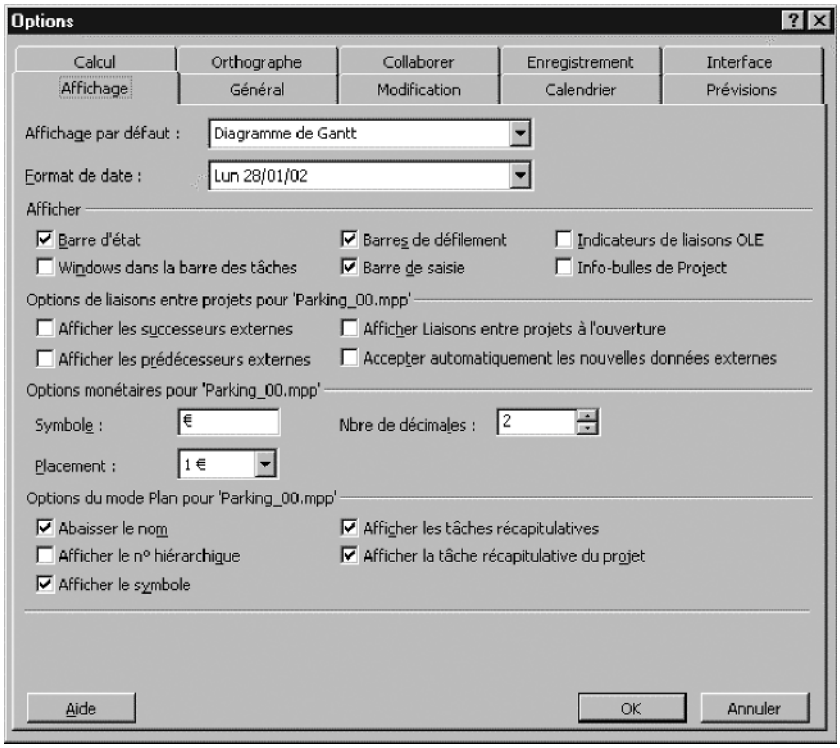
Puis, dans **Fichier**, **Propriétés**, onglet '**Résumé**', on saisit le **Sujet** et le nom de l'**Auteur**.

15.2.2 Choix des options du projet Parking

On sélectionne les options de planification par **Outils**, **Options…**

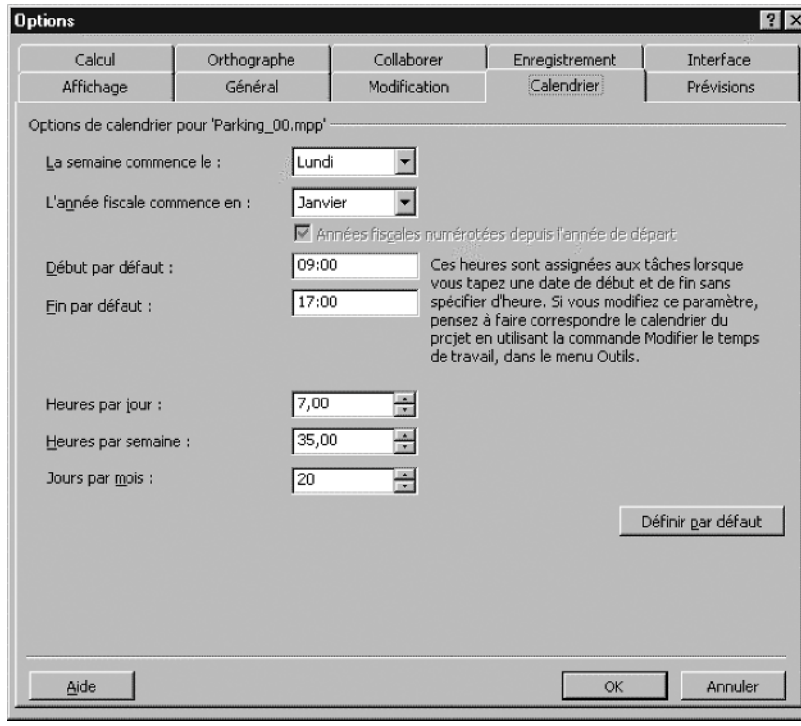
Onglet Affichage

On choisit d'afficher une tâche récapitulative du projet, précédée d'un symbole, et en décalant vers la droite les autres tâches par « Abaisser le nom ». On va par ailleurs gérer les coûts de notre projet en Euros.



Onglet Calendrier

On table sur une semaine de 35 heures réparties uniformément sur 5 jours.



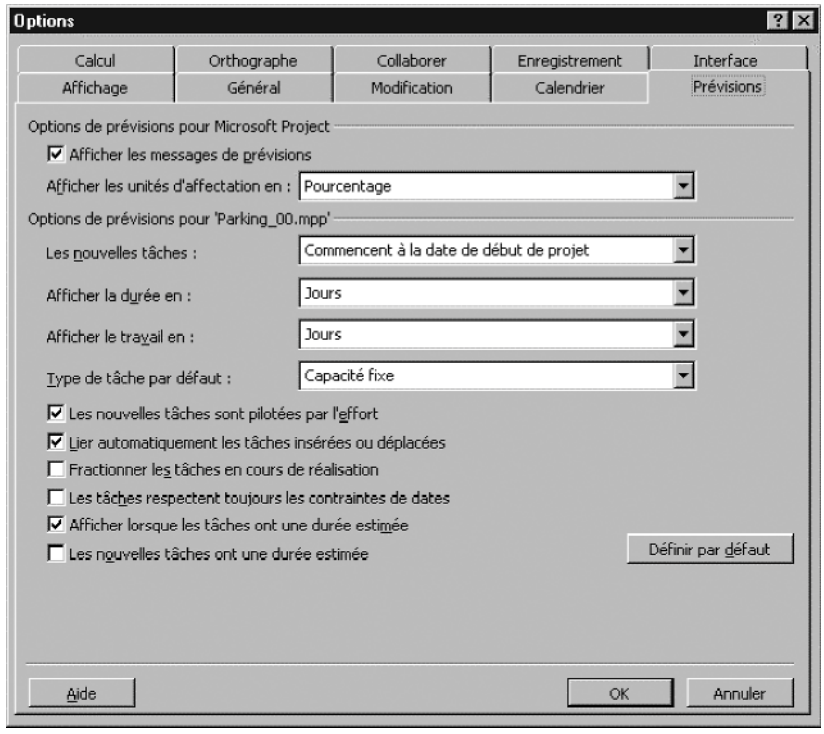
Onglet Prévisions

Le type de tâche par défaut **Capacité fixe** signifie que l'information « Durée », saisie lors de la planification des tâches, est en fait une quantité de travail — c'est-à-dire une charge — et que la durée de la tâche pourra être modifiée par MS Project en fonction du nombre de personnes affectées.

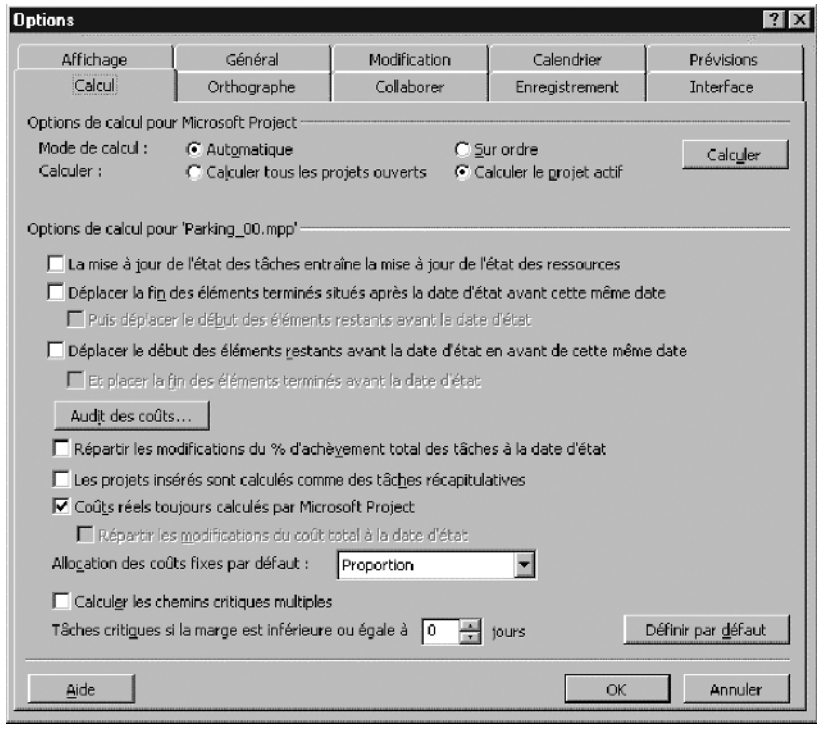
Le pilotage **par l'effort** signifie que le suivi s'effectuera à partir des comptes rendus d'avancement des ressources affectées au projet. Dans le cas contraire, on saisirait directement les données d'avancement d'une tâche, sans passer par les ressources. Pour des projets principalement basés sur du travail intellectuel, il est conseillé de suivre le travail de chaque intervenant dans l'équipe de projet.

Onglet Calcul

Par **Calcul Automatique**, on choisit de laisser MS Project déclencher le calcul de la planification ou de l'avancement du projet dès la saisie de nouvelles informations. De même, les **coûts réels**, que l'on sous-entend proportionnels aux ressources utilisées, seront calculés automatiquement à la saisie de l'avancement.



Onglet Prévisions



Onglet Calcul

15.2.3 Définition du calendrier du projet Parking

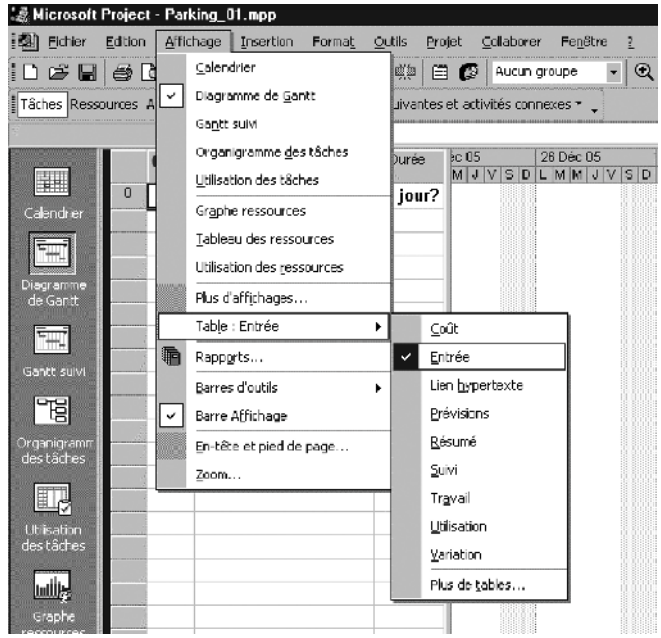
On modifie le calendrier par **Outils**, **Modifier le temps de travail…** On sélectionne du lundi au vendredi (**L** à **V**) et on modifie les horaires de travail de façon à avoir des demi-journées de même durée (3 h 30).

On va ensuite indiquer les deux jours fériés, vendredi saint et lundi de Pâques, soit les **17 et 20 mars**.

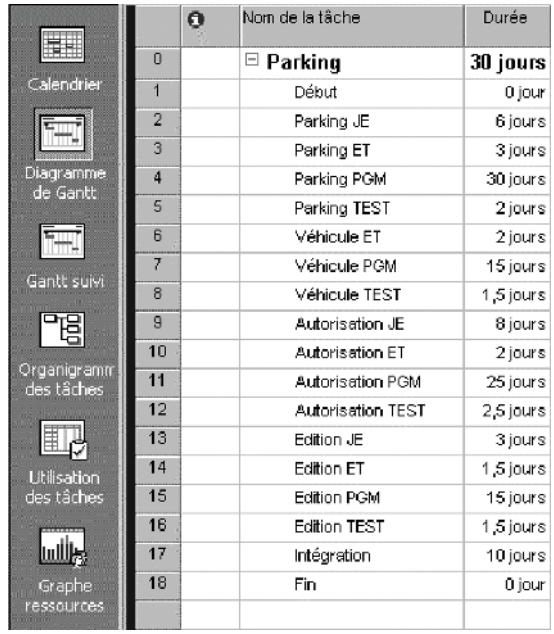
15.3 PLANIFICATION DU PROJET PARKING

15.3.1 Saisie des tâches du projet Parking

La saisie se fait par **Affichage**, **Diagramme de Gantt** avec le choix complémentaire **Table : Entrée**.



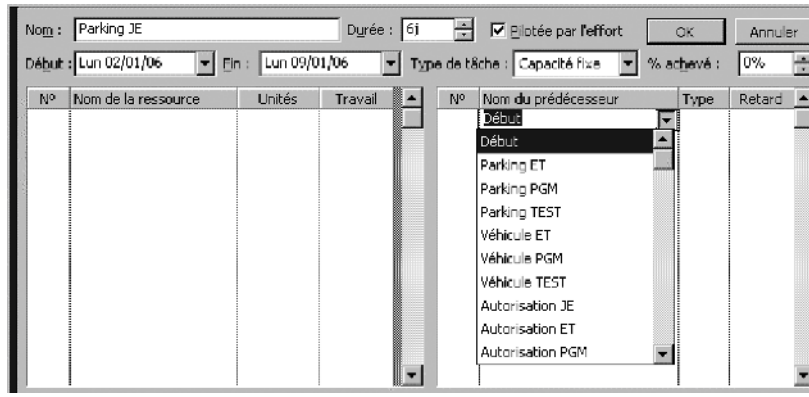
On saisit la liste des tâches et leurs durées, telle qu'elles figurent ci-dessous, ainsi que les tâches de type jalon (Début et Fin) avec une durée nulle. On obtient le tableau suivant, avec en tête une tâche récapitulative, la tâche 0, qui a été insérée automatiquement avec la durée de la tâche la plus longue, soit 30 jours conformément à l'option d'affichage choisie précédemment.



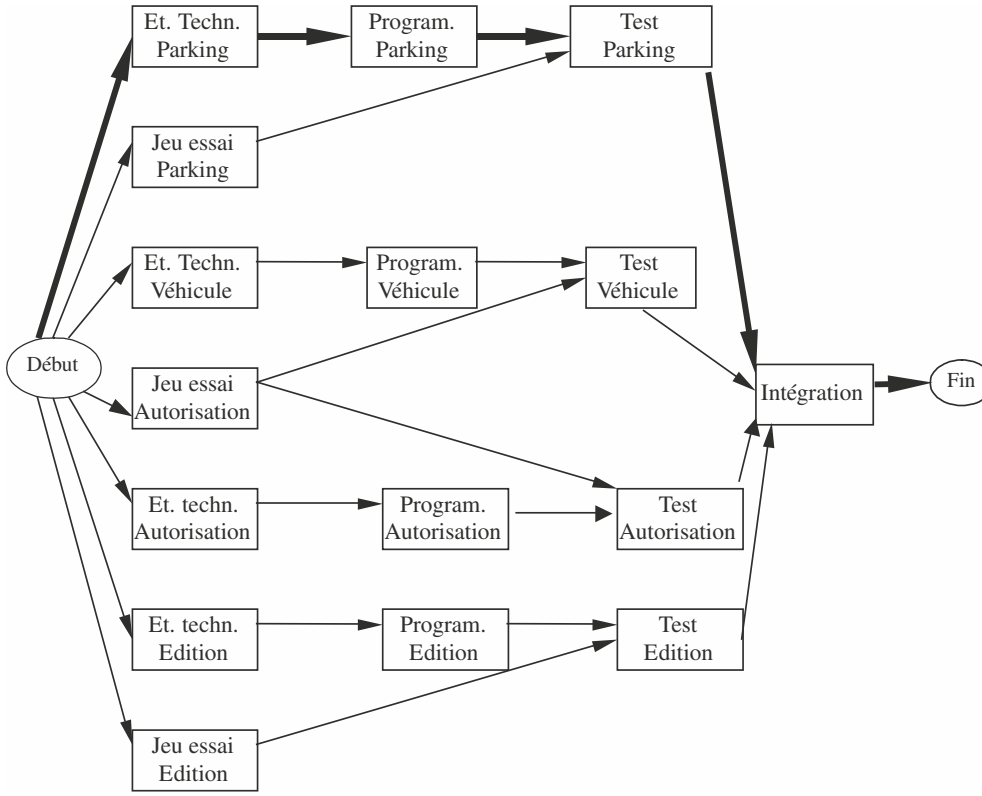
	Nom de la tâche	Durée
0	**Parking**	**30 jours**
1	Début	0 jour
2	Parking JE	6 jours
3	Parking ET	3 jours
4	Parking PGM	30 jours
5	Parking TEST	2 jours
6	Véhicule ET	2 jours
7	Véhicule PGM	15 jours
8	Véhicule TEST	1,5 jours
9	Autorisation JE	8 jours
10	Autorisation ET	2 jours
11	Autorisation PGM	25 jours
12	Autorisation TEST	2,5 jours
13	Edition JE	3 jours
14	Edition ET	1,5 jours
15	Edition PGM	15 jours
16	Edition TEST	1,5 jours
17	Intégration	10 jours
18	Fin	0 jour

15.3.2 Saisie des liens du projet Parking

Pour saisir les liens entre tâches, on ouvre d'abord une boite de détail en bas de l'écran par **Fenêtre**, **Fractionner**. Puis, on se positionne sur une tâche et on indique toutes les tâches qui la précèdent directement.



On saisit les liens tels qu'ils figurent dans le graphe des antécédents ci-dessous :



On peut constater que la tâche récapitulative du projet a été mise à jour et que la durée du chemin le plus long — en temps travaillé, c'est-à-dire hors jours chômés — est de **45 jours**.

	Nom de la tâche	Durée	Début	Fin	Prédécesseurs
0	**Parking**	**45 jours**	**Lun 02/01/06**	**Ven 03/03/06**	
1	Début	0 jour	Lun 02/01/06	Lun 02/01/06	
2	Parking JE	6 jours	Lun 02/01/06	Lun 09/01/06	1
3	Parking ET	3 jours	Lun 02/01/06	Mer 04/01/06	1
4	Parking PGM	30 jours	Jeu 05/01/06	Mer 15/02/06	3
5	Parking TEST	2 jours	Jeu 16/02/06	Ven 17/02/06	2;4
6	Véhicule ET	2 jours	Lun 02/01/06	Mar 03/01/06	1
7	Véhicule PGM	15 jours	Mer 04/01/06	Mar 24/01/06	6
8	Véhicule TEST	1,5 jours	Mer 25/01/06	Jeu 26/01/06	7;9
9	Autorisation JE	8 jours	Lun 02/01/06	Mer 11/01/06	1
10	Autorisation ET	2 jours	Lun 02/01/06	Mar 03/01/06	1
11	Autorisation PGM	25 jours	Mer 04/01/06	Mar 07/02/06	10
12	Autorisation TEST	2,5 jours	Mer 08/02/06	Ven 10/02/06	9;11
13	Edition JE	3 jours	Lun 02/01/06	Mer 04/01/06	1
14	Edition ET	1,5 jours	Lun 02/01/06	Mar 03/01/06	1
15	Edition PGM	15 jours	Mar 03/01/06	Mar 24/01/06	14
16	Edition TEST	1,5 jours	Mar 24/01/06	Mer 25/01/06	13;15
17	Intégration	10 jours	Lun 20/02/06	Ven 03/03/06	5;8;12;16
18	Fin	0 jour	Ven 03/03/06	Ven 03/03/06	17

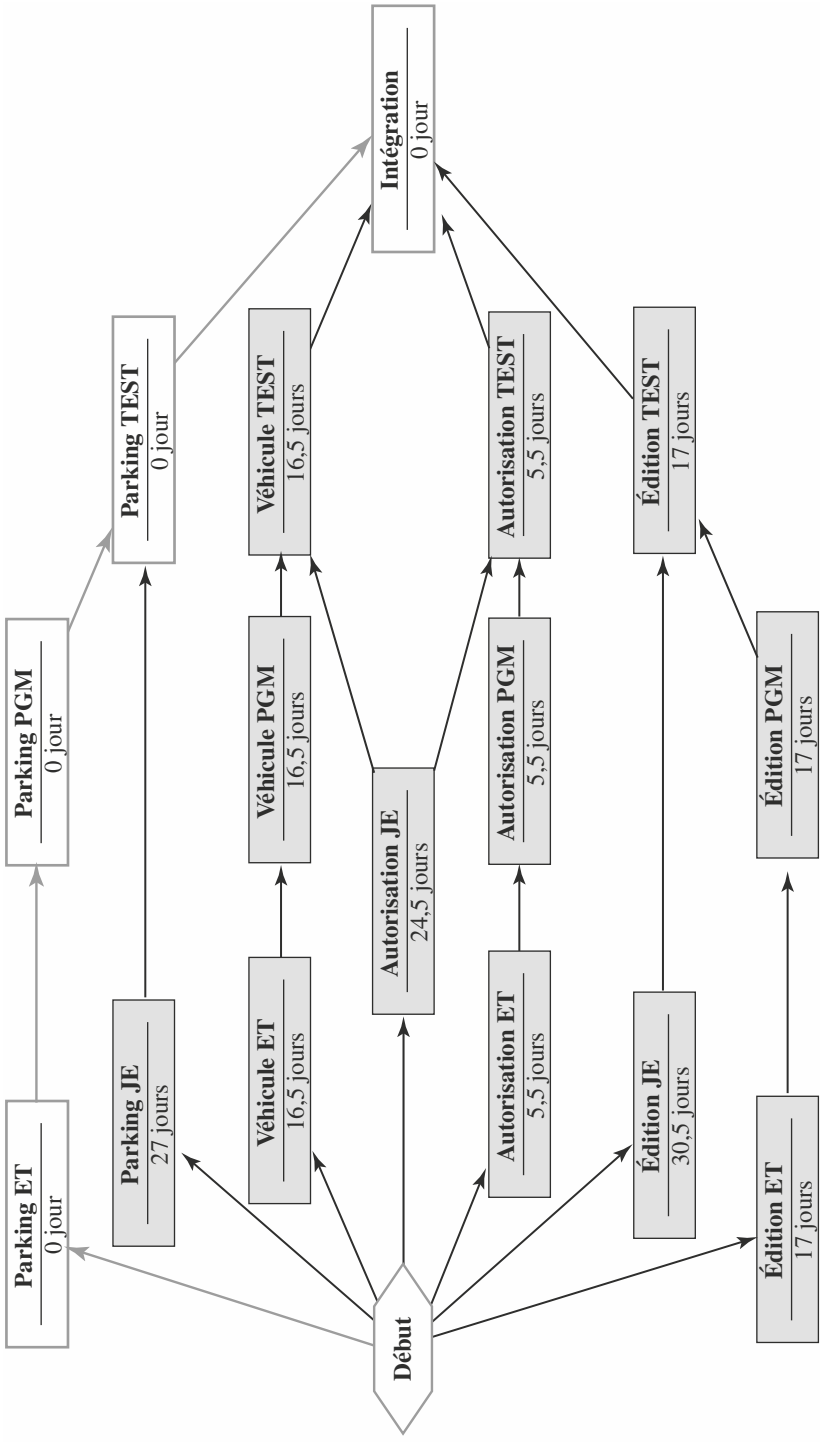
Si l'on affiche l'ensemble du diagramme réseau (Affichage, Organigramme des tâches) par **Affichage**, **Zoom…** avec le choix **Ensemble du projet,** MS Projet organise automatiquement la disposition des taches.

Cependant en sélectionnant dans **Format, Disposition, Mode de disposition,** « **Positionnement manuel des cases** » on peut déplacer manuellement les tâches puis aligner les cases par **Format**, **Réorganiser maintenant**. On peut en modifier le style des liaisons entre les tâches par **Format**, **Disposition**, « **Styles des liaisons** ».

Les champs Début et Fin correspondent aux dates au plus tôt. Par **Format, Style de case**, **Plus de modèles**, on peut choisir les champs à afficher dans le graphique. On peut par exemple remplacer le N° par la marge totale.

On obtient le réseau ci-contre.

On peut vérifier sur l'écran que les marges totales affichées sont bien égales à celles du diagramme ci-dessous.

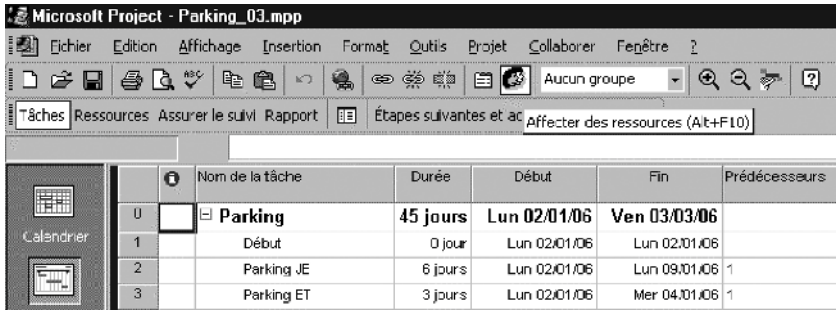

Début
Parking ET
0 jour
Parking PGM
0 jour
Parking JE
27 jours
Parking TEST
0 jour
Véhicule ET
16,5 jours
Véhicule PGM
16,5 jours
Véhicule TEST
16,5 jours
Autorisation JE
24,5 jours
Autorisation ET
5,5 jours
Autorisation PGM
5,5 jours
Autorisation TEST
5,5 jours
Édition JE
30,5 jours
Édition ET
17 jours
Édition PGM
17 jours
Édition TEST
17 jours
Intégration
0 jour

15.3.3 Saisie et affectation des ressources du projet Parking

On crée nos trois ressources Claude, Camille et Boss (le chef de projet) et on indique leur coût horaire (respectivement 50, 80 et 150 Euros) par **Affichage**, **Tableau des ressources**, avec le choix **Table : Entrée**. Ce qui nous donne :

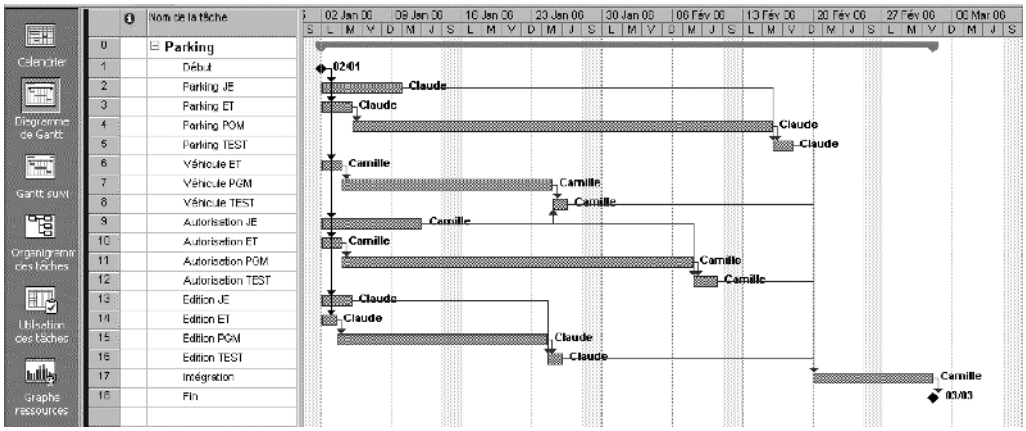
	Nom de la ressource	Type	Étiquette Matériel	Initiales	Groupe	Capacité max	Tx. standard	Tx. hrs. sup.	Coût/Utilisation	Allocation	Calendrier de base	Code
1	Claude	Travail		CL		100%	50,00 €/hr	0,00 €/hr	0,00 €	Proportion	Standard	
2	Camille	Travail		CA		100%	80,00 €/hr	0,00 €/hr	0,00 €	Proportion	Standard	
3	Boss	Travail		BO		100%	150,00 €/hr	0,00 €/hr	0,00 €	Proportion	Standard	

On indique l'attribution des lots Parking et Édition à Claude, tandis que Camille fera les lots Véhicule et Autorisation, ainsi que l'Intégration. Pour cela, on utilise **Affichage**, **Diagramme de Gantt**, avec le choix **Table : Entrée**. On sélectionne les tâches à affecter (en utilisant la touche **Ctrl** si les tâches ne sont pas contiguës), puis on clique sur l'icône d'affectation en haut à droite de l'écran et on choisit la ressource.



On aurait pu utiliser également Outils, Ressources, Affectation des ressources…

Si l'on regarde le diagramme de Gantt, on voit que l'exécution des tâches a été planifiée par MS Project sans tenir compte de la surutilisation des ressources.



Le détail des surutilisations est donné par **Affichage**, **Utilisation des ressources**, avec le choix **Table : Résumé**.

Si l'on conserve uniquement les colonnes qui nous intéressent et que l'on filtre l'affichage par **Projet**, **Filtré pour : Ressources surutilisées**, on obtient le tableau récapitulatif ci-dessous. Claude et Camille sont par moments chargés respectivement jusqu'à 400 et 300 % !

	Nom de la ressource	Groupe	Capacité max	Pointe	Tx. standard	Tx. hrs. sup.	Coût	Travail
1	⊞ Claude		100%	400%	50,00 €/hr	0,00 €/hr	21 700,00 €	62 jours
2	⊞ Camille		100%	300%	80,00 €/hr	0,00 €/hr	36 960,00 €	66 jours

Les jours de surutilisation peuvent être visualisés avec la commande **Format**, **Détails** et le choix **Surutilisation**.

Pour supprimer la surutilisation, nous devons décider de l'ordre dans lequel chacun effectuera les tâches (cf. ci-dessous). Ensuite, le plus simple est d'introduire les modifications dans les caractéristiques des tâches, en changeant les prédécesseurs. Cela revient à effectuer un nivellement manuel.

On modifie la durée de la tâche 'Parking JE'puisque l'on a attribué 8 jours à Claude. MS Project ne gère pas la notion de charge affectée. On peut conserver l'estimation standard (6 jours) en commentaire de la tâche.

La saisie des nouvelles données peut se faire directement dans la table **Entrée** du **Diagramme de Gantt**. Le nivellement effectué ici se résume à la modification des prédécesseurs des tâches. Vous observerez que les dates sont automatiquement changées.

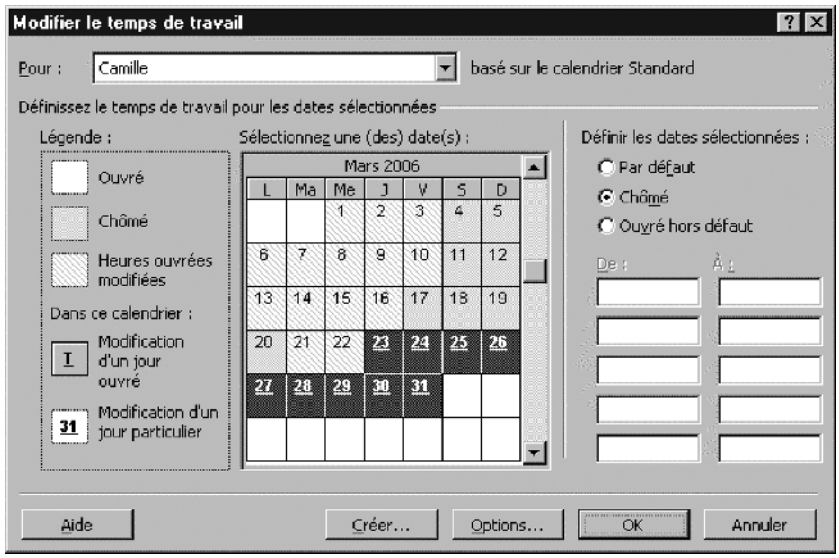
Après modifications, la nouvelle liste est la suivante :

	Nom de la tâche	Durée	Début	Fin	Prédécesseurs	Noms ressources
0	⊟ **Parking**	**74 jours**	**Lun 02/01/06**	**Lun 17/04/06**		
1	Début	0 jour	Lun 02/01/06	Lun 02/01/06		
2	Parking JE	8 jours	Lun 02/01/06	Mer 11/01/06	1	Claude
3	Parking ET	3 jours	Jeu 12/01/06	Lun 16/01/06	2	Claude
4	Parking PGM	30 jours	Mar 17/01/06	Lun 27/02/06	3	Claude
5	Parking TEST	2 jours	Mar 28/02/06	Mer 01/03/06	4	Claude
6	Véhicule ET	2 jours	Jeu 12/01/06	Ven 13/01/06	9	Camille
7	Véhicule PGM	15 jours	Lun 16/01/06	Ven 03/02/06	6	Camille
8	Véhicule TEST	1,5 jours	Lun 06/02/06	Mar 07/02/06	7	Camille
9	Autorisation JE	8 jours	Lun 02/01/06	Mer 11/01/06	1	Camille
10	Autorisation ET	2 jours	Mar 07/02/06	Jeu 09/02/06	8	Camille
11	Autorisation PGM	25 jours	Jeu 09/02/06	Jeu 16/03/06	10	Camille
12	Autorisation TEST	2,5 jours	Jeu 16/03/06	Mer 22/03/06	11	Camille
13	Edition JE	3 jours	Jeu 02/03/06	Lun 06/03/06	5	Claude
14	Edition ET	1,5 jours	Mar 07/03/06	Mer 08/03/06	13	Claude
15	Edition PGM	15 jours	Mer 08/03/06	Ven 31/03/06	14	Claude
16	Edition TEST	1,5 jours	Ven 31/03/06	Lun 03/04/06	15	Claude
17	Intégration	10 jours	Mar 04/04/06	Lun 17/04/06	12;16	Camille
18	Fin	0 jour	Lun 17/04/06	Lun 17/04/06	17	

On va ensuite rechercher la période où Camille pourra prendre ses congés.

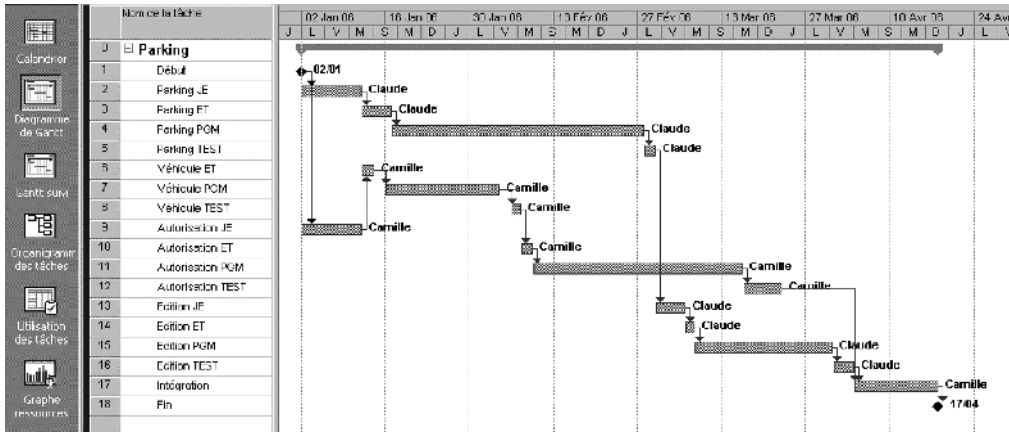
On peut utiliser le graphe des ressources par : **Affichage**, **Tableau des ressources** ; on sélectionne Camille, puis **Affichage**, **Graphe ressources** ; on voit, après avoir choisi par **Format**, **Echelle de temps…** un format adéquat, qu'entre le 23 mars et le 3 avril compris, Camille n'est pas utilisée.

On peut alors modifier son calendrier pour planifier son absence par **Outils**, **Modifier le temps de travail…** pour Camille.



Pour visualiser le Gantt définitif sur une seule page écran, on peut réduire la hauteur des lignes en sélectionnant l'ensemble des lignes et en ajustant avec le pointeur la hauteur de n'importe quelle cellule située dans la première colonne :

On obtient :

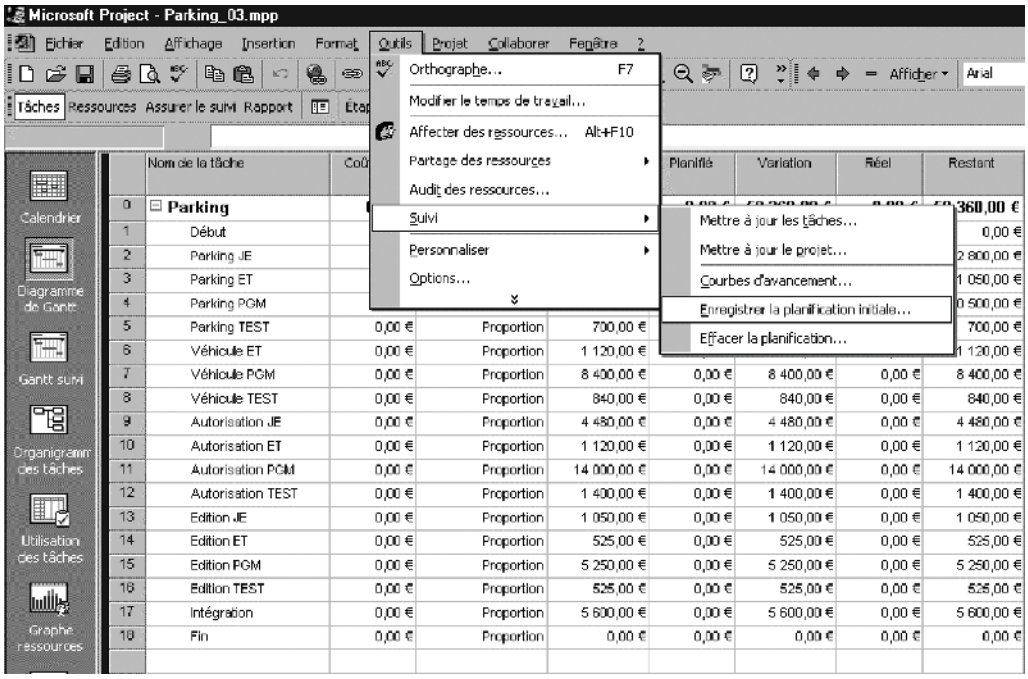


On peut obtenir le coût du projet résultant de l'utilisation prévue de Claude et Camille par **Affichage**, **Diagramme de Gantt** et **Table : Coût**. Il est de 59 360 euros qui se répartissent ainsi :

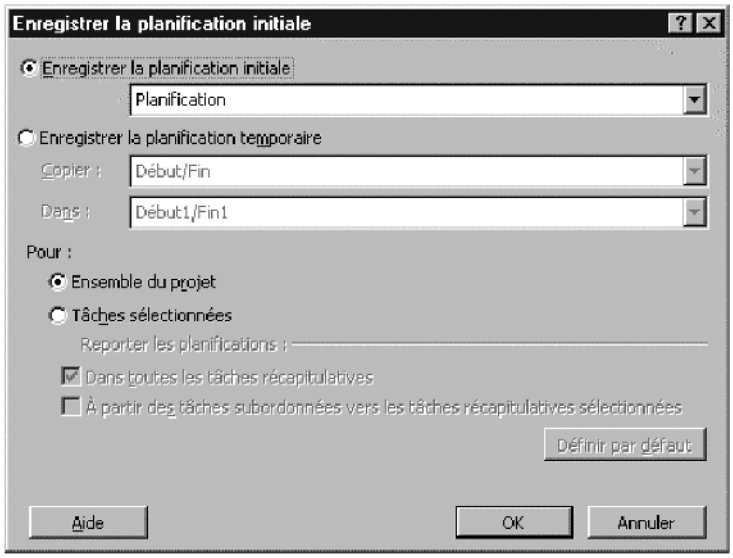
	Nom de la tâche	Coût fixe	Allocation des coûts fixes	Coût total	Planifié	Variation	Réel	Restant
0	**Parking**	**0,00 €**	**Proportion**	**59 360,00 €**	**0,00 €**	**59 360,00 €**	**0,00 €**	**59 360,00 €**
1	Début	0,00 €	Proportion	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €
2	Parking JE	0,00 €	Proportion	2 800,00 €	0,00 €	2 800,00 €	0,00 €	2 800,00 €
3	Parking ET	0,00 €	Proportion	1 050,00 €	0,00 €	1 050,00 €	0,00 €	1 050,00 €
4	Parking PGM	0,00 €	Proportion	10 500,00 €	0,00 €	10 500,00 €	0,00 €	10 500,00 €
5	Parking TEST	0,00 €	Proportion	700,00 €	0,00 €	700,00 €	0,00 €	700,00 €
6	Véhicule ET	0,00 €	Proportion	1 120,00 €	0,00 €	1 120,00 €	0,00 €	1 120,00 €
7	Véhicule PGM	0,00 €	Proportion	8 400,00 €	0,00 €	8 400,00 €	0,00 €	8 400,00 €
8	Véhicule TEST	0,00 €	Proportion	840,00 €	0,00 €	840,00 €	0,00 €	840,00 €
9	Autorisation JE	0,00 €	Proportion	4 480,00 €	0,00 €	4 480,00 €	0,00 €	4 480,00 €
10	Autorisation ET	0,00 €	Proportion	1 120,00 €	0,00 €	1 120,00 €	0,00 €	1 120,00 €
11	Autorisation PGM	0,00 €	Proportion	14 000,00 €	0,00 €	14 000,00 €	0,00 €	14 000,00 €
12	Autorisation TEST	0,00 €	Proportion	1 400,00 €	0,00 €	1 400,00 €	0,00 €	1 400,00 €
13	Edition JE	0,00 €	Proportion	1 050,00 €	0,00 €	1 050,00 €	0,00 €	1 050,00 €
14	Edition ET	0,00 €	Proportion	525,00 €	0,00 €	525,00 €	0,00 €	525,00 €
15	Edition PGM	0,00 €	Proportion	5 250,00 €	0,00 €	5 250,00 €	0,00 €	5 250,00 €
16	Edition TEST	0,00 €	Proportion	525,00 €	0,00 €	525,00 €	0,00 €	525,00 €
17	Intégration	0,00 €	Proportion	5 600,00 €	0,00 €	5 600,00 €	0,00 €	5 600,00 €
18	Fin	0,00 €	Proportion	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €

15.3.4 Mémorisation du planifié pour le projet Parking

Maintenant, nous sommes satisfaits du planning, nous allons donc enregistrer la planification comme planification initiale. Pour ce faire : **Outils, Suivi, Enregistrer la planification initiale**… :



	Nom de la tâche	Coût		Planifié	Variation	Réel	Restant
0	Parking						360,00 €
1	Début						0,00 €
2	Parking JE						2 800,00 €
3	Parking ET						1 050,00 €
4	Parking PGM						0 500,00 €
5	Parking TEST	0,00 €	Proportion	700,00 €			700,00 €
6	Véhicule ET	0,00 €	Proportion	1 120,00 €			1 120,00 €
7	Véhicule PGM	0,00 €	Proportion	8 400,00 €	0,00 €	8 400,00 €	0,00 € 8 400,00 €
8	Véhicule TEST	0,00 €	Proportion	840,00 €	0,00 €	840,00 €	0,00 € 840,00 €
9	Autorisation JE	0,00 €	Proportion	4 480,00 €	0,00 €	4 480,00 €	0,00 € 4 480,00 €
10	Autorisation ET	0,00 €	Proportion	1 120,00 €	0,00 €	1 120,00 €	0,00 € 1 120,00 €
11	Autorisation PGM	0,00 €	Proportion	14 000,00 €	0,00 €	14 000,00 €	0,00 € 14 000,00 €
12	Autorisation TEST	0,00 €	Proportion	1 400,00 €	0,00 €	1 400,00 €	0,00 € 1 400,00 €
13	Edition JE	0,00 €	Proportion	1 050,00 €	0,00 €	1 050,00 €	0,00 € 1 050,00 €
14	Edition ET	0,00 €	Proportion	525,00 €	0,00 €	525,00 €	0,00 € 525,00 €
15	Edition PGM	0,00 €	Proportion	5 250,00 €	0,00 €	5 250,00 €	0,00 € 5 250,00 €
16	Edition TEST	0,00 €	Proportion	525,00 €	0,00 €	525,00 €	0,00 € 525,00 €
17	Intégration	0,00 €	Proportion	5 600,00 €	0,00 €	5 600,00 €	0,00 € 5 600,00 €
18	Fin	0,00 €	Proportion	0,00 €	0,00 €	0,00 €	0,00 € 0,00 €



On sélectionne, « Enregistrer la planification initiale » pour « Ensemble du projet ».

Après validation, nous pouvons constater que le **coût total** du projet est égal au **planifié** et que la **variation** est à 0.00 €.

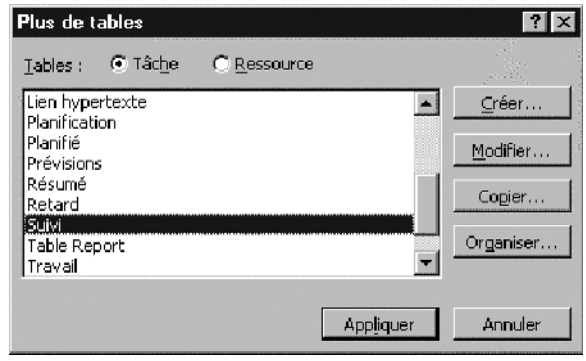
	Nom de la tâche	Coût fixe	Allocation des coûts fixes	Coût total	Planifié	Variation	Réel	Restant
0	**Parking**	**0,00 €**	**Proportion**	**59 360,00 €**	**59 360,00 €**	**0,00 €**	**0,00 €**	**59 360,00 €**
1	Début	0,00 €	Proportion	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €
2	Parking JE	0,00 €	Proportion	2 800,00 €	2 800,00 €	0,00 €	0,00 €	2 800,00 €
3	Parking ET	0,00 €	Proportion	1 050,00 €	1 050,00 €	0,00 €	0,00 €	1 050,00 €
4	Parking PGM	0,00 €	Proportion	10 500,00 €	10 500,00 €	0,00 €	0,00 €	10 500,00 €
5	Parking TEST	0,00 €	Proportion	700,00 €	700,00 €	0,00 €	0,00 €	700,00 €
6	Véhicule ET	0,00 €	Proportion	1 120,00 €	1 120,00 €	0,00 €	0,00 €	1 120,00 €
7	Véhicule PGM	0,00 €	Proportion	8 400,00 €	8 400,00 €	0,00 €	0,00 €	8 400,00 €
8	Véhicule TEST	0,00 €	Proportion	840,00 €	840,00 €	0,00 €	0,00 €	840,00 €
9	Autorisation JE	0,00 €	Proportion	4 480,00 €	4 480,00 €	0,00 €	0,00 €	4 480,00 €
10	Autorisation ET	0,00 €	Proportion	1 120,00 €	1 120,00 €	0,00 €	0,00 €	1 120,00 €
11	Autorisation PGM	0,00 €	Proportion	14 000,00 €	14 000,00 €	0,00 €	0,00 €	14 000,00 €
12	Autorisation TEST	0,00 €	Proportion	1 400,00 €	1 400,00 €	0,00 €	0,00 €	1 400,00 €
13	Edition JE	0,00 €	Proportion	1 050,00 €	1 050,00 €	0,00 €	0,00 €	1 050,00 €
14	Edition ET	0,00 €	Proportion	525,00 €	525,00 €	0,00 €	0,00 €	525,00 €
15	Edition PGM	0,00 €	Proportion	5 250,00 €	5 250,00 €	0,00 €	0,00 €	5 250,00 €
16	Edition TEST	0,00 €	Proportion	525,00 €	525,00 €	0,00 €	0,00 €	525,00 €
17	Intégration	0,00 €	Proportion	5 600,00 €	5 600,00 €	0,00 €	0,00 €	5 600,00 €
18	Fin	0,00 €	Proportion	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €

15.4 PILOTAGE DU PROJET PARKING

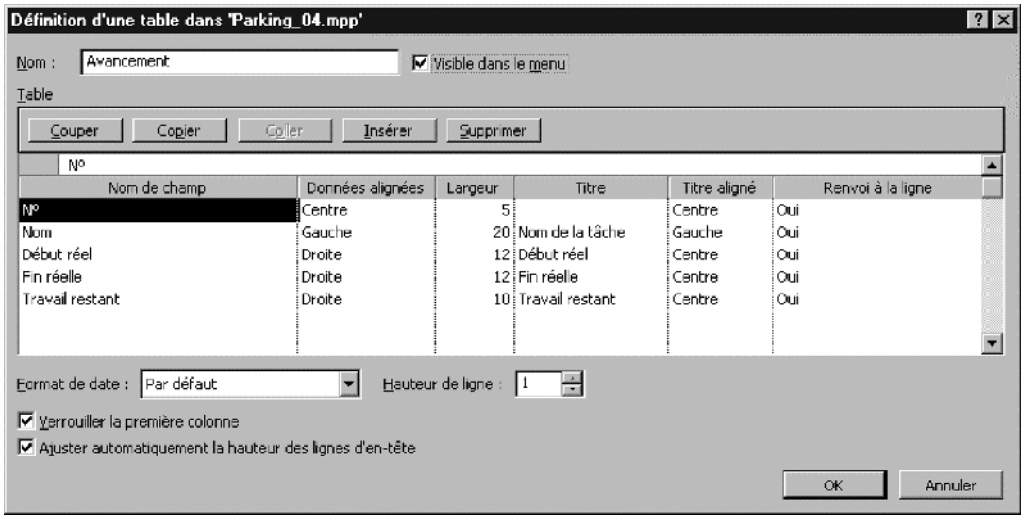
15.4.1 Saisie de l'avancement du projet Parking à la fin janvier

Pour alléger notre illustration, nous allons effectuer la saisie des comptes rendus d'avancement non pas à la fin de chaque semaine, mais en fin de mois. On commence par modifier la date du jour par **Projet**, **Informations sur le projet…** : on se positionne au **1/2/06 (Date actuelle)**.

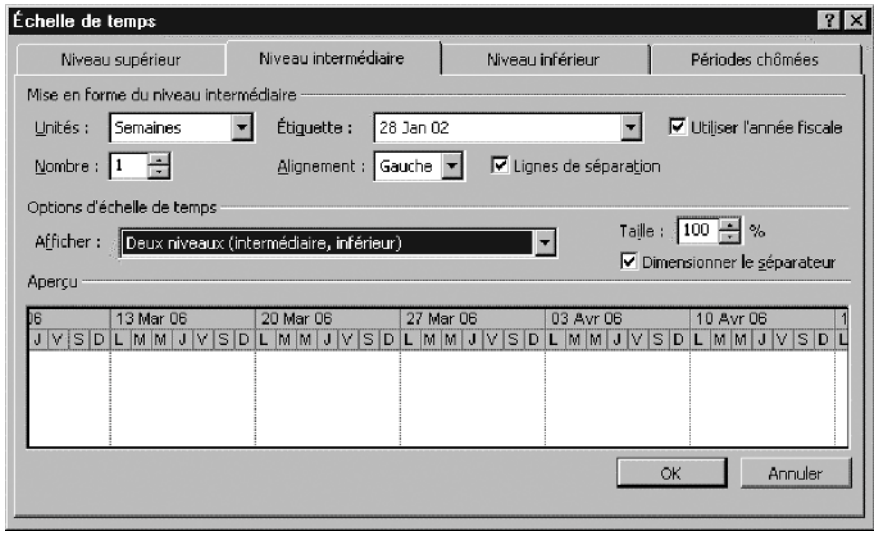
On utilise la grille de saisie donnée par **Affichage**, **Gantt suivi**. On va personnaliser la table utilisée par **Table : Plus de tables…** On copie la table **Suivi** que l'on va ajuster :



On appelle la nouvelle table **Avancement**. Elle contient les cinq champs suivants : N°, Nom, Début réel, Fin réelle et Travail restant (reste à faire). Il faut également renseigner les Titres des colonnes et cocher la case « Visible dans le menu ». Puis OK et Appliquer.



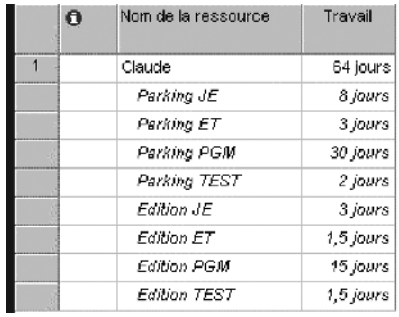
Ensuite on ajuste les échelles de temps par **Format, Echelle de temps**… de façon à avoir une colonne par jour. Par exemple, au niveau intermédiaire on choisit la Semaine et au niveau inférieur le Jour (on peut également modifier les **Étiquettes**) :

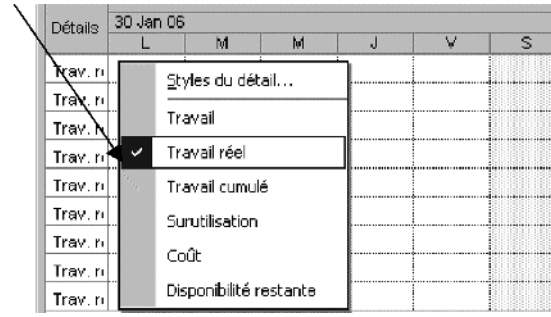


On définit une fenêtre permettant de saisir le travail effectué quotidiennement par **Fenêtre**, **Fractionner**. On clique dans la fenêtre du bas et on sélectionne **Affichage**, **Utilisation des ressources** avec **Table : Utilisation**.

En cliquant sur la tâche ParkingJE de la fenêtre du haut, on voit apparaître toutes les tâches de la ressource Claude.

On se positionne alors dans le calendrier en bas à droite et on clique avec le bouton droit de la souris ; on peut sélectionner ce que l'on veut saisir : le **Travail réel** (vérifier une deuxième fois avec le bouton droit que seul « **Travail réel** » est coché !).



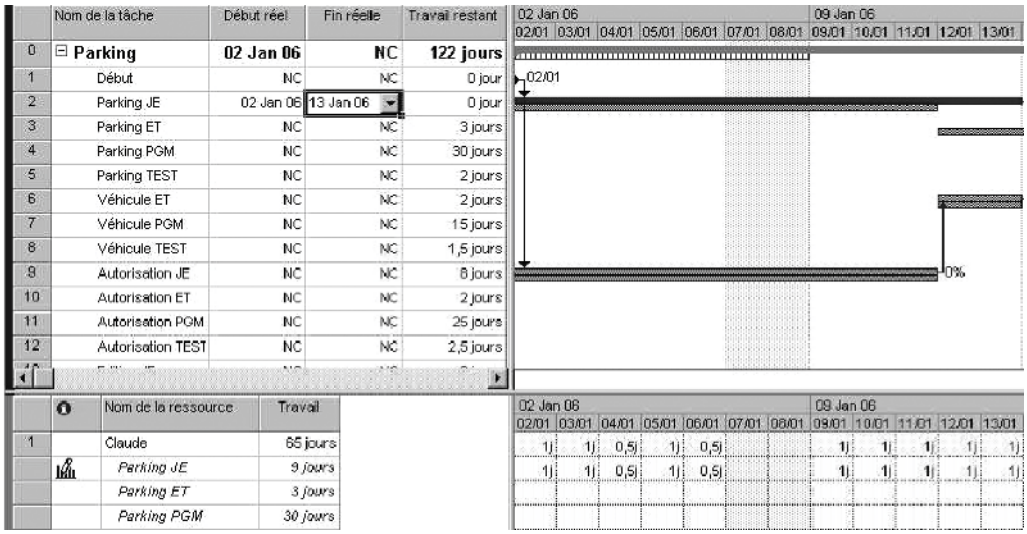


On va maintenant saisir les jours de travail réels des ressources sur chaque tâche du mois de janvier (d'après les comptes rendus d'avancement donnés plus bas), de la façon suivante :

- on se positionne dans le tableau en bas à gauche sur la tâche sur laquelle la ressource a travaillé ;
- on indique les journées ou demi-journées travaillées en tapant **1** ou **0,5** dans les cellules correspondant aux jours du calendrier, en bas à droite : la

date de début réel du tableau en haut à gauche est automatiquement renseignée ;

- *si la tâche est terminée* (reste à faire = 0), on saisit la date de fin réelle sur le tableau en haut à gauche ;
- *si la tâche n'est pas terminée* à la fin du mois et que le reste à faire est différent de celui qui a été calculé par MS Project et qui figure dans le champ Travail restant du tableau en haut à gauche, on saisit le reste à faire indiquer dans le compte-rendu d'avancement.



On saisit successivement l'avancement de Claude et Camille pour le mois de janvier, selon les éléments de leurs comptes rendus hebdomadaires, donnés ci-dessous.

Claude	2/1	3/1	4/1	5/1	6/1	Reste à faire
Parking JE	1	1	0,5	1	0,5	4
	9/1	10/1	11/1	12/1	13/1	Reste à faire
Parking JE	1	1	1	1	1	0
	16/1	17/1	18/1	19/1	20/1	Reste à faire
Parking ET	1	1	1			0
Parking PGM				1	1	28
	23/1	24/1	25/1	26/1	27/1	Reste à faire
Parking PGM	1	1	1	1		24

	30/1	31/1				Reste à faire
Parking PGM	1	1				22

Camille	2/1	3/1	4/1	5/1	6/1	Reste à faire
Autorisation JE	1	1	1	1	1	0
	9/1	10/1	11/1	12/1	13/1	Reste à faire
Véhicule ET	1	1				0
Véhicule PGM			1	1	1	13
	16/1	17/1	18/1	19/1	20/1	
Maladie						
	23/1	24/1	25/1	26/1	27/1	Reste à faire
Véhicule PGM	1	1	1	1	1	8
	30/1	31/1				Reste à faire
Véhicule PGM	1	1				2

15.4.2 Récapitulatif de l'avancement du projet Parking à la fin janvier

On va préparer un tableau synthétique de l'avancement mensuel.

On supprime le fractionnement par **Fenêtre, Supprimer le fractionnement** et on sélectionne **Affichage**, **Gantt suivi**, avec **Table : Plus de tables…**

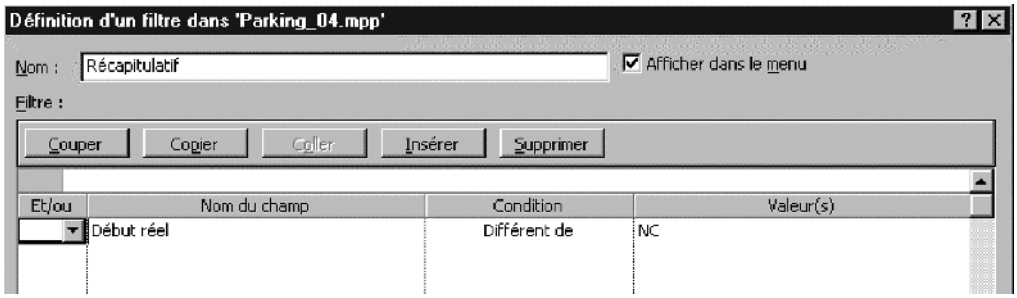
On copie la table **Suivi** que l'on va ajuster en la renommant **Récapitulatif**, avec dans l'ordre les quatorze champs suivants, qui permettent de suivre la production et les écarts par rapport au planifié :

N°, Initiales de la ressource, Nom, % Travail achevé, Travail réel, Travail restant, Travail planifié, Variation de travail, Début réel, Début planifié, Variation date de début, Fin réelle, Fin planifiée, Variation date de fin.

Pour faciliter le suivi, on va modifier l'affichage en triant et en filtrant les données, de la façon suivante.

Pour présenter ce tableau par intervenant, on utilise **Projet**, **Trier, Trier par…** en choisissant **Initiales de la ressource** comme premier critère de tri (décroissant) et **N°** comme second critère de tri (croissant).

Pour n'afficher que les tâches qui ont été actives en janvier, c'est-à-dire ayant une date de début réel, on utilise **Projet**, **Filtré pour : Plus de filtres…** et on crée un nouveau filtre que l'on appelle **Récapitulatif** et qui contient la condition suivante :



On applique le filtre et on obtient le tableau de bord suivant, où l'on peut suivre les consommations en charge :

N°	Init. Ress.	Nom de la tâche	% Trav.Achevé	Trav.Réel	Trav.Restant	Trav.Planifié	Var.Travail
0		**Parking**	**30%**	**37 jours**	**88 jours**	**130 jours**	**-5 jours**
2	CL	Parking JE	100%	9 jours	0 jour	8 jours	1 jour
3	CL	Parking ET	100%	3 jours	0 jour	3 jours	0 jour
4	CL	Parking PGM	27%	8 jours	22 jours	30 jours	0 jour
6	CA	Véhicule ET	100%	2 jours	0 jour	2 jours	0 jour
7	CA	Véhicule PGM	83%	10 jours	2 jours	15 jours	-3 jours
9	CA	Autorisation JE	100%	5 jours	0 jour	8 jours	-3 jours

Si l'on regarde les colonnes les plus à droite, on voit que le projet a commencé à la date prévue, mais a pris trois jours de retard global par rapport à la fin planifiée :

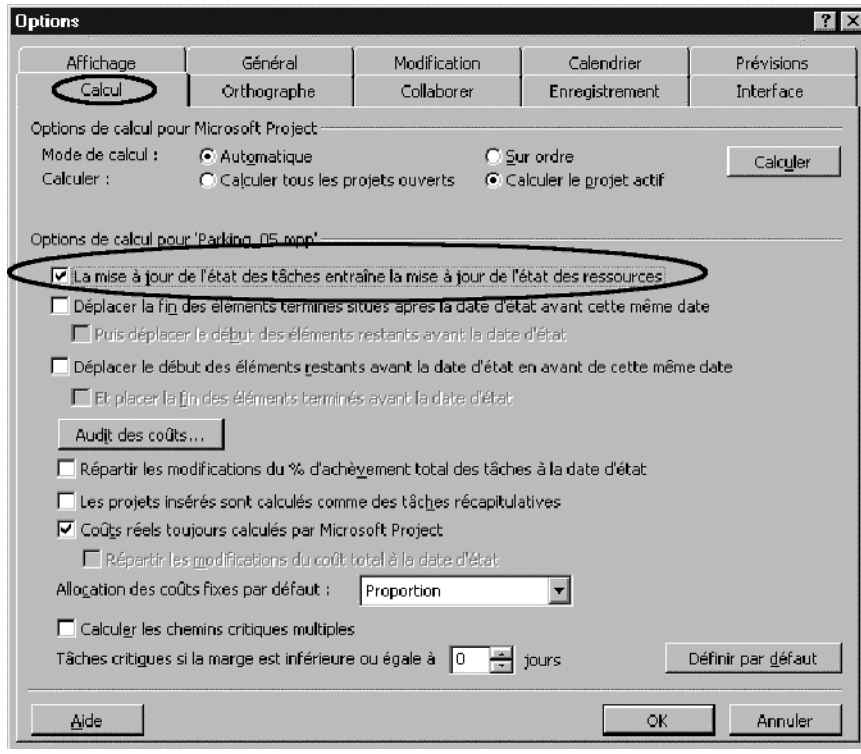
N°	Début réel	Début planifié	Var.D.Début	Fin réelle	Fin planifiée	Variation D.Fin
0	**Lun 02/01/06**	**Lun 02/01/06**	**0 jour**	**NC**	**Lun 17/04/06**	**3 jours**
2	Lun 02/01/06	Lun 02/01/06	0 jour	Ven 13/01/06	Mer 11/01/06	2 jours
3	Lun 16/01/06	Jeu 12/01/06	2 jours	Mer 18/01/06	Lun 16/01/06	2 jours
4	Jeu 19/01/06	Mar 17/01/06	2 jours	NC	Lun 27/02/06	3 jours
6	Lun 09/01/06	Jeu 12/01/06	-3 jours	Mar 10/01/06	Ven 13/01/06	-3 jours
7	Mer 11/01/06	Lun 16/01/06	-3 jours	NC	Ven 03/02/06	-1 jour
9	Lun 02/01/06	Lun 02/01/06	0 jour	Ven 06/01/06	Mer 11/01/06	-3 jours

NB : Si l'on veut que les valeurs du champ « % de Travail achevé » soient visibles sur le diagramme de Gantt, il faut le spécifier, par exemple à l'aide de l'Assistant Diagramme de Gantt du menu Format dans la rubrique « Infos sur la tâche personnalisée ».

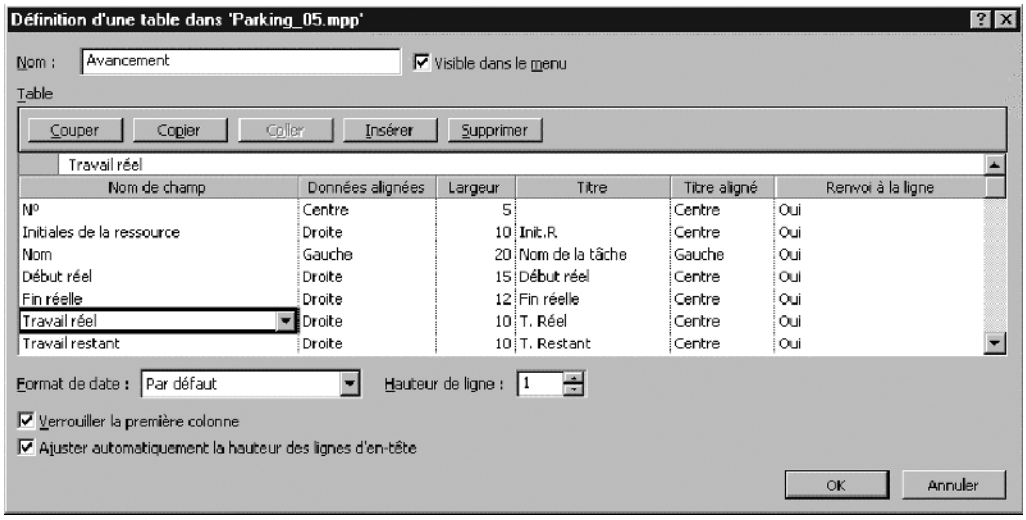
15.4.3 Saisie de l'avancement du projet Parking à la fin février

À titre d'illustration, on va maintenant montrer une saisie simplifiée, lorsque l'on ne veut pas suivre les jours calendaires précis pendant lesquels le travail réel a été effectué.

On choisit une option permettant de mettre à jour automatiquement le travail des ressources par **Outils**, **Options…** avec l'onglet **Calcul** :



On choisit **Affichage**, **Gantt suivi** et **Table : Avancement**. On modifie la table Avancement par **Table : Plus de tables…** pour y rajouter les champs Initiales de la ressource et Travail réel par le bouton Insérer :



L'application de cette table nous fournit tous les champs nécessaires à la saisie simplifiée. On procède de la façon suivante :

- On commence par changer la date du jour par **Projet**, **Informations sur le projet…** et on met la « Date actuelle » au **1/3/06**.
- Pour plus de commodité, on restreint l'affichage aux tâches inachevées par **Projet**, **Filtré pour : Tâches inachevées**, tout en laissant les tâches triées sur l'Initiale de la ressource (décroissant), puis par N° (croissant).
- Pour les tâches ayant commencé pendant le mois, on saisit la date de **Début réel**.
- Pour toutes les tâches sur lesquelles la ressource a travaillé pendant le mois de février, on saisit dans la colonne **Travail réel** la somme du temps déjà passé en janvier (qui est affiché) et du temps passé en février. Par exemple, sur la tâche 'Parking PGM', on a déjà 8 jours dans Réel, il faut remplacer cette valeur par 26 puisque Claude a travaillé 18 jours sur cette tâche en février.
- Pour les tâches terminées, on saisit **0** dans la colonne **Travail restant** si ce n'est pas déjà la valeur affichée ainsi que la date de **Fin réelle**.
- Pour les tâches non terminées, on modifie éventuellement la valeur affichée dans **Travail restant**, si elle ne correspond pas au Reste à faire réel en fin de mois.

On saisit l'avancement de février correspondant aux données suivantes :

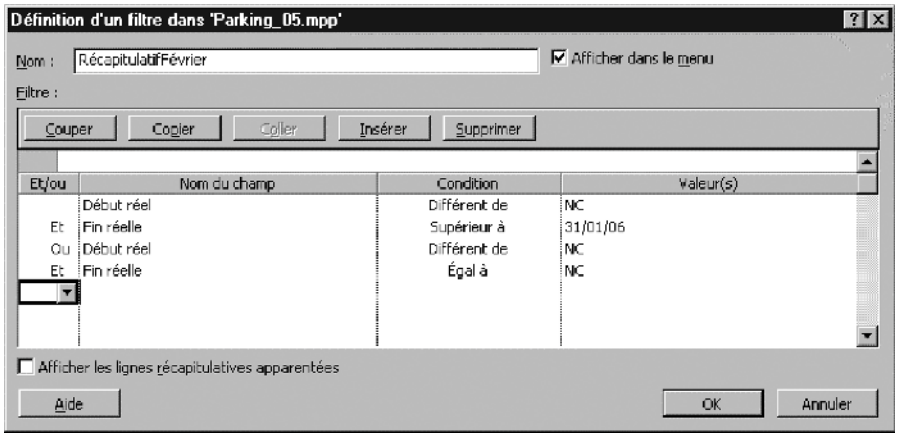
Nom	Tâche	Date début	Date fin	Temps passé	Reste à faire
Claude					
	Parking PGM			18	8
Camille					
	Véhicule PGM		2/2/06	2	0
	Véhicule TEST	3/2/06	7/2/06	3	0
	Autorisation ET	8/2/06	9/2/06	2	0
	Autorisation PGM	10/2/06		12	13

Ce qui nous donne :

	Init.R	Nom de la tâche	Début réel	Fin réelle	T. Réel	T. Restant
0		⊟ **Parking**	**Lun 02/01/06**	**NC**	**74 jours**	**56,5 jours**
4	CL	Parking PGM	Jeu 19/01/06	NC	26 jours	8 jours
5	CL	Parking TEST	NC	NC	0 jour	2 jours
13	CL	Edition JE	NC	NC	0 jour	3 jours
14	CL	Edition ET	NC	NC	0 jour	1,5 jours
15	CL	Edition PGM	NC	NC	0 jour	15 jours
16	CL	Edition TEST	NC	NC	0 jour	1,5 jours
7	CA	Véhicule PGM	Mer 11/01/06	Jeu 02/02/06	12 jours	0 jour
8	CA	Véhicule TEST	Ven 03/02/06	Mar 07/02/06	3 jours	0 jour
10	CA	Autorisation ET	Mer 08/02/06	Jeu 09/02/06	2 jours	0 jour
11	CA	Autorisation PGM	Ven 10/02/06	NC	12 jours	13 jours
12	CA	Autorisation TEST	NC	NC	0 jour	2,5 jours
17	CA	Intégration	NC	NC	0 jour	10 jours
1		Début	NC	NC	0 jour	0 jour
18		Fin	NC	NC	0 jour	0 jour

Pour étudier le récapitulatif de février, on veut restreindre l'affichage aux tâches qui ont été actives durant le mois.

Pour cela, on définit un nouveau filtre, appelé **RécapitulatifFévrier** par **Projet**, **Filtré pour : Plus de filtres…** On fait une copie du filtre Récapitulatif que l'on modifie ainsi :



Avec la table Avancement, le tableau obtenu est le suivant :

	Ress.	Nom de la tâche	Début réel	Fin réelle	Travail réel	Travail restant
0		**Parking**	**02 Jan 06**	**NC**	**74 jours**	**56,5 jours**
4	CL	Parking PGM	19 Jan 06	NC	26 jours	8 jours
7	CA	Véhicule PGM	11 Jan 06	02 Fév 06	12 jours	0 jour
8	CA	Véhicule TEST	03 Fév 06	07 Fév 06	3 jours	0 jour
10	CA	Autorisation ET	08 Fév 06	09 Fév 06	2 jours	0 jour
11	CA	Autorisation PGM	10 Fév 06	NC	12 jours	13 jours

On obtient le récapitulatif suivant avec la table Récapitulatif :

N°	Init. Ress.	Nom de la tâche	% Trav.Achevé	Trav.Réel	Trav.Restant	Trav.Planifié	Var.Travail
0		**Parking**	**57%**	**74 jours**	**56,5 jours**	**130 jours**	**0,5 jour**
4	CL	Parking PGM	76%	26 jours	8 jours	30 jours	4 jours
7	CA	Véhicule PGM	100%	12 jours	0 jour	15 jours	-3 jours
8	CA	Véhicule TEST	100%	3 jours	0 jour	1,5 jours	1,5 jours
10	CA	Autorisation ET	100%	2 jours	0 jour	2 jours	0 jour
11	CA	Autorisation PGM	48%	12 jours	13 jours	25 jours	0 jour

On peut relever que le mode de calcul du champ % **Travail achevé** diffère de la définition de l'avancement que nous avons donnée et illustrée, comme une différence entre deux Reste à faire.

En effet, pour MS Project, la définition est :

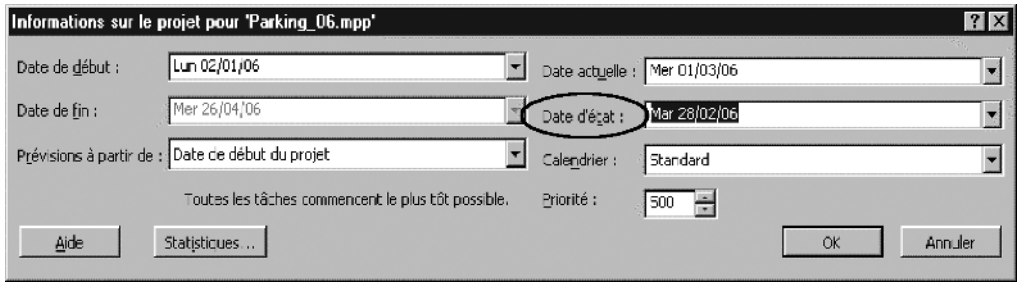
%Travail achevé = [Travail réel/(Travail réel + Travail restant)] x 100.

C'est donc la part réalisée de la nouvelle prévision. Si l'on prend ParkingPGM, on a bien :

$$76\ \% = (26/(26 + 8)) \times 100.$$

15.4.4 Bilan individuel du projet Parking à la fin février

On va utiliser une table standard de suivi des coûts, telle que fournie par MS Project. On a besoin pour cela d'indiquer la date à laquelle on veut faire le bilan, c'est-à-dire la fin février, et ceci correspond à la date d'état que l'on trouve dans **Projet, Informations sur le projet…**



On peut alors utiliser **Affichage, Utilisation des ressources** avec la **Table : Audit des coûts**, dont nous allons expliquer la signification des champs qui y figurent. Ils sont tous en devises (Euros), mais on peut établir des correspondances avec les indicateurs proposés au chapitre 12.

	Nom de la ressource	CBTP	CBTE	CRTE	VS	VC	FAC	BAC	VAC
	⊞ Non affecté	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €
1	⊞ Claude	14 700,00 €	11 879,41 €	13 300,00 €	-2 820,59 €	-1 420,59 €	24 150,00 €	22 400,00 €	-1 750,00 €
2	⊞ Camille	23 520,00 €	22 680,00 €	20 160,00 €	-840,00 €	2 520,00 €	34 440,00 €	36 960,00 €	2 520,00 €
3	Boss	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €

Si l'on traduit ces montants en devises en jours de travail, en les divisant par le coût journalier de chaque ressource (350 euros pour Claude et 560 euros pour Camille), cela donne les valeurs suivantes :

	CBTP	CBTE	CRTE	VS	VC	FAC	BAC	VAC
CL	42	33,94	38	– 8,05	– 4,05	69	64	– 5
CA	42	40,5	36	– 1,5	4,5	61,5	66	4,5

Nous allons rappeler la signification et le mode de calcul de ces indicateurs liés à la méthode de la valeur acquise, exposée au paragraphe 7.3., mais avec l'ancienne terminologie :

- Le **CBTP,** Coût Budgété du Travail Prévu, correspond à la Valeur planifiée (VP) : c'est le coût qui aurait dû être engagé selon la planification initiale. Ramené en jours, c'est le temps qui aurait dû être consacré au projet. Il peut aussi être considéré comme l'avancement planifié. Pour les deux intervenants, la valeur est de 42 jours, c'est-à-dire un plein temps en janvier et février 2006.
- Le **CBTE**, Coût Budgété du Travail Effectué, correspond à la Valeur acquise (VA) : cet indicateur représente un avancement réel. Il est calculé en multipliant, pour chaque tâche affectée, le pourcentage de travail achevé par la charge initiale valorisée au coût des ressources. C'est donc : [Travail réel/(Travail réel + Travail restant)] * Charge initiale.
- Le **CRTE**, Coût Réel du Travail Effectué, correspond au Coût réel (CR) : c'est le temps réellement passé sur le projet.

  Si le CRTE est supérieur au CBTE, cela signifie que la performance n'est pas bonne : les intervenants consomment trop de temps par rapport à leur avancement.

  Si le CRTE est inférieur au CBTP, cela signifie que ce que nous avions appelé le « coefficient d'utilisation » au chapitre 12 est inférieur à 1 : les intervenants consacrent moins de temps que prévu au projet !
- **VS**, *Value Schedule*, appelé maintenant Écart de délai, mesure la différence entre l'avancement réel et l'avancement planifié. Il est calculé par :

  CBTE – CBTP.

  Si le VS est négatif, c'est que la production n'est pas aussi avancée que prévu.
- **VC**, *Value Cost*, appelé maintenant Écart de coût, mesure la différence entre l'avancement réel et le temps passé. Il est calculé par :

  CBTE – CRTE.

  Si le VC est négatif, c'est que la production consomme plus de temps que ce que l'on avait planifié.
- **FAC**, *Forecast At Completion*, appelé maintenant Prévision à l'achèvement : c'est l'estimation de coût la plus vraisemblable compte tenu des informations dont on dispose à la date t. Cet indicateur correspond à la charge prévisible à ce jour, c'est-à-dire la somme du Travail réel (Temps passé) et du Travail restant (Reste à faire). Il prend en compte toutes les tâches affectées, y compris celles qui ne sont pas commencées.

  Pour Claude, les 69 jours correspondent aux 38 jours de travail réel, plus 8 jours de travail restant sur ParkingPGM, plus les 23 jours des tâches affectées et non commencées (ParkingTEST et tout le lot Édition).

Pour Camille, les 61,5 jours correspondent aux 36 jours déjà consommés, auxquels s'ajoute le reste de AutorisationPGM (13 jours), Autorisation-TEST (2,5 jours) et Intégration (10 jours).

- **BAC**, *Budgeted At Completion*, maintenant appelé Budget à l'achèvement : c'est le coût total planifié. Cet indicateur correspond à la totalité de la charge affectée.

  Les chiffres correspondent bien aux 64 jours de charge initialement planifiés pour Claude, et 66 pour Camille.

- **VAC**, *Variance At Completion*, maintenant appelé Écart à l'achèvement, est la différence entre le coût total planifié et le coût prévisible, c'est-à-dire BAC – FAC. Cet indicateur correspond à l'évolution positive ou négative de la charge : ramené en jours, c'est la différence entre le temps passé et l'avancement. On peut donc le rapprocher de ce que nous avons appelé la « vitesse » au paragraphe 11.5.

  Pour Claude, on trouve –5 jours qui correspondent bien à :
  38 jours (temps total passé) – 14 jours (avancement de janvier) – 19 jours (avancement de février).

  Pour Camille, on a 4,5 jours qui correspondent bien à : 36 jours (temps total passé) – 23 jours (avancement de janvier) – 17,5 jours (avancement de février).

15.4.5 Gestion des aléas du projet Parking

Nous allons maintenant illustrer le traitement d'aléas obligeant à modifier la planification : le 21 mars au soir, Claude a un accident l'immobilisant pendant trois jours.

On positionne la **date actuelle** au 21/3/06 (**Projet, Informations sur le projet…**). Les éléments au 21 au soir sont les suivants :

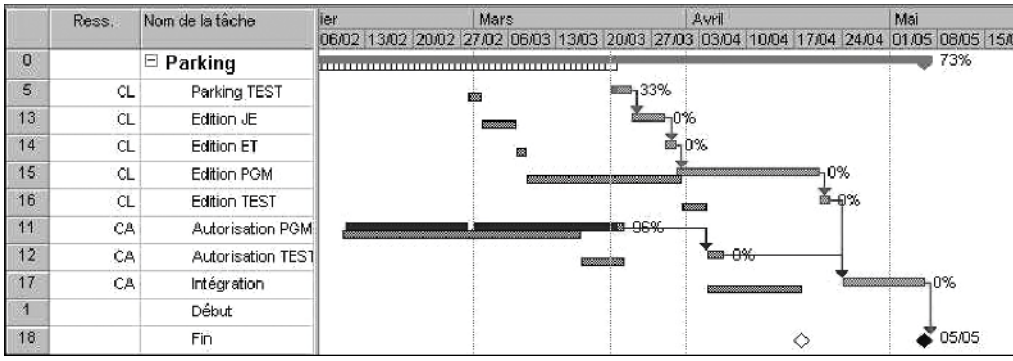
Claude			1/3	2/3	3/3	Reste à faire
Parking PGM			1	1	1	6
	6/3	7/3	8/3	9/3	10/3	Reste à faire
Parking PGM	1	1	1	1	1	3
	13/3	14/3	15/3	16/3	17/3	Reste à faire
Parking PGM	1	1	1	1		0
	20/3	21/3	22/3	23/3	24/3	Reste à faire
Parking TEST		1				2

Camille			1/3	2/3	3/3	Reste à faire
Autorisation PGM			1	1	1	10
	6/3	7/3	8/3	9/3	10/3	Reste à faire
Autorisation PGM	1	1	1	1	1	7
	13/3	14/3	15/3	16/3	17/3	Reste à faire
Autorisation PGM	1	1	1	1		2
	20/3	21/3	22/3	23/3	24/3	Reste à faire
Autorisation PGM		1				1

On saisit l'avancement par la méthode 1 (en commençant par enlever l'option de calcul : **Outils, Options** : « La mise à jour de l'état des tâches entraîne la mise à jour de l'état des ressources ») :

Pour analyser la situation après avoir fait la saisie, on regarde le Gantt par **Affichage**, **Gantt suivi** et **Projet**, **Filtré pour : Tâches inachevées**.

On obtient :



	Ress.	Nom de la tâche	Début réel	Fin réelle	Travail réel	Travail restant
0		**Parking**	**02 Jan 06**	**NC**	**100 jours**	**36,5 jours**
5	CL	Parking TEST	21 Mar 06	NC	1 jour	2 jours
13	CL	Edition JE	NC	NC	0 jour	3 jours
14	CL	Edition ET	NC	NC	0 jour	1,5 jours
15	CL	Edition PGM	NC	NC	0 jour	15 jours
16	CL	Edition TEST	NC	NC	0 jour	1,5 jours
11	CA	Autorisation PGM	10 Fév 06	NC	25 jours	1 jour
12	CA	Autorisation TEST	NC	NC	0 jour	2,5 jours
17	CA	Intégration	NC	NC	0 jour	10 jours

On voit que la fin du projet est repoussée du 17 avril au 5 mai, à cause du retard de Claude.

Pour résorber ce retard, on modifie donc les affectations (et les prédécesseurs) par **Affichage, Diagramme de Gantt** avec **Table : Entrée** :

- La tâche 'Édition JE' est confiée à Boss, le chef de projet, qui commencera dès le lendemain (22/3/06). La liaison doit être supprimée.
- La tâche 'Édition ET' est confiée à Camille : elle la fera juste après 'Autorisation PGM'et avant 'Autorisation TEST'.
- La tâche 'Edition PGM' aura comme prédécesseur la tâche 'Parking TEST'.
- La tâche 'Intégration' commencera avant la fin du lot Édition.

On modifie les calendriers personnels par Outils, Modifier le temps de travail… :

- Camille, qui devait partir le 22 mars au soir, viendra travailler les 22 et 23 toute la journée, ainsi que le 24 au matin. En revanche, elle ne reviendra de congé que le 5 avril après-midi, au lieu du 4 au matin.
- Claude viendra travailler les trois premiers samedis d'avril : 1er, 8 et 15. Il sera absent : les 22, 23 et 24 mars.

Après avoir modifié l'affectation des ressources et le ou les prédécesseurs de chaque tâche, on obtient :

	Nom de la tâche	Durée	Début	Fin	Prédécesseurs	Noms ressources
0	**Parking**	**78 jours**	**Lun 02/01/06**	**Ven 21/04/06**		
1	Début	0 jour	Lun 02/01/06	Lun 02/01/06		
5	Parking TEST	3 jours	Mar 21/03/06	Mar 28/03/06	4	Claude
11	Autorisation PGM	26 jours	Ven 10/02/06	Mer 22/03/06	10	Camille
12	Autorisation TEST	2,5 jours	Mer 05/04/06	Ven 07/04/06	14	Camille
13	Edition JE	3 jours	Mer 22/03/06	Ven 24/03/06		Boss
14	Edition ET	1,5 jours	Jeu 23/03/06	Ven 24/03/06	11	Camille
15	Edition PGM	15 jours	Mer 29/03/06	Ven 14/04/06	5	Claude
16	Edition TEST	1,5 jours	Sam 15/04/06	Lun 17/04/06	15	Claude
17	Intégration	10 jours	Lun 10/04/06	Ven 21/04/06	12	Camille
18	Fin	0 jour	Ven 21/04/06	Ven 21/04/06	17	

15.4.6 Tableau d'avancement du projet Parking à la fin mars

Fin mars, nous allons présenter un tableau d'avancement pour le maître d'ouvrage, qui sera une synthèse par lots.

L'avancement réel a été le suivant :

Claude	20/3	21/3	22/3	23/3	24/3	Reste à faire
Parking TEST		1				2
	27/3	28/3	29/3	30/3	31/3	Reste à faire
Parking TEST	1	1				0
Édition PGM			1	1	1	12

Camille	20/3	21/3	22/3	23/3	24/3	Reste à faire
Autorisation PGM		1	1			0
Édition ET				1	1	0
	27/3	28/3	29/3	30/3	31/3	Reste à faire
Congé						

Boss	20/3	21/3	22/3	23/3	24/3	Reste à faire
Édition JE				1	1	2
	27/3	28/3	29/3	30/3	31/3	Reste à faire
Édition JE	1	1				0

On effectue la mise à jour par la méthode 1.

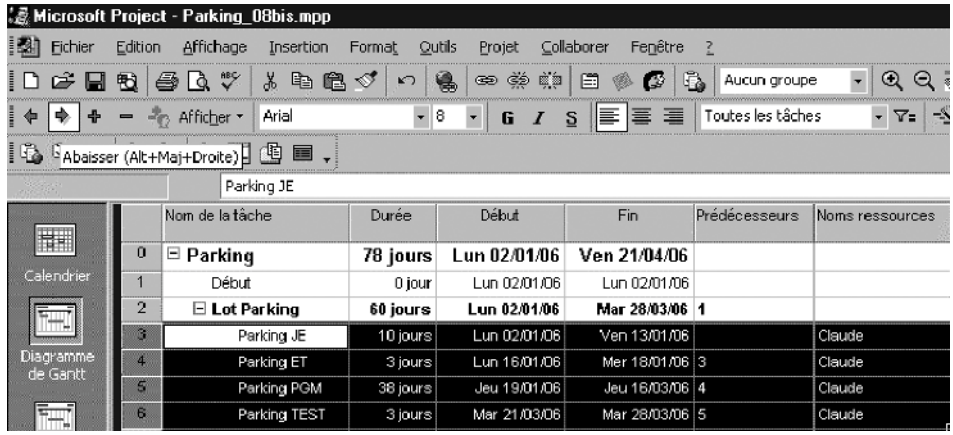
15.4.7 Présentation par lots

Ensuite, on prépare une présentation de l'avancement par lot. Pour cela, on va introduire quatre tâches récapitulatives : Lot Parking, Lot Édition, Lot Véhicule, Lot Autorisation et Lot Intégration.

On sélectionne **Affichage, Diagramme de Gantt** avec **Table : Entrée**.

On se place sur la tâche 'Parking JE' et on crée une nouvelle tâche que l'on appelle **Lot Parking** par **Insertion, Insérer une tâche**. On insère ensuite les cinq autres tâches récapitulatives au-dessus de la première tâche du lot.

Ensuite, on sélectionne les quatre tâches du lot Parking et on descend leur niveau hiérarchique en cliquant sur l'icône Abaisser en haut à gauche de l'écran :



	Nom de la tâche	Durée	Début	Fin	Prédécesseurs	Noms ressources
0	**Parking**	**78 jours**	**Lun 02/01/06**	**Ven 21/04/06**		
1	Début	0 jour	Lun 02/01/06	Lun 02/01/06		
2	**Lot Parking**	**60 jours**	**Lun 02/01/06**	**Mar 28/03/06**	1	
3	Parking JE	10 jours	Lun 02/01/06	Ven 13/01/06		Claude
4	Parking ET	3 jours	Lun 16/01/06	Mer 18/01/06	3	Claude
5	Parking PGM	38 jours	Jeu 19/01/06	Jeu 16/03/06	4	Claude
6	Parking TEST	3 jours	Mar 21/03/06	Mar 28/03/06	5	Claude

Les quatre tâches ont été décalées par rapport à la tâche récapitulative Lot Parking.

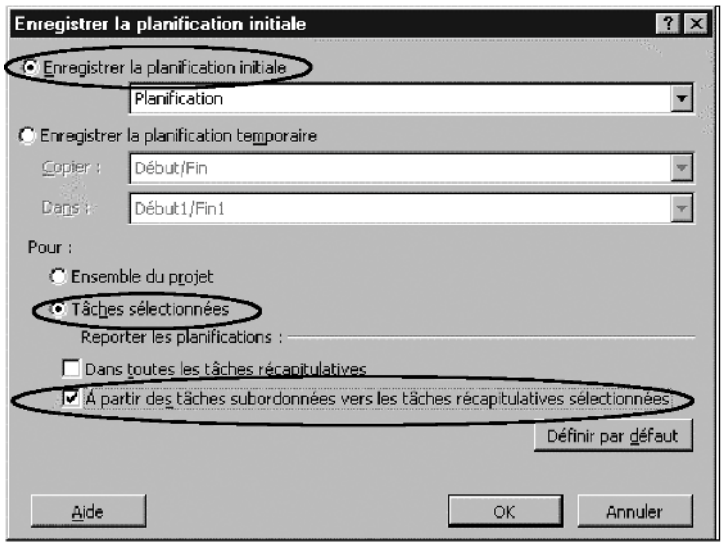
On fait de même pour les quatre autres lots : Édition, Véhicule, Autorisation, Intégration. On obtient la présentation suivante par **Affichage**, **Diagramme de Gantt** avec **Table : Travail** :

	Nom de la tâche	Travail	Planifié	Variation	Réel	Restant
0	**Parking**	**138 jours**	**130 jours**	**8 jours**	**112 jours**	**26 jours**
1	Début	0 jour	0 jour	0 jour	0 jour	0 jour
2	**Lot Parking**	**53 jours**	**0 jour**	**53 jours**	**53 jours**	**0 jour**
3	Parking JE	9 jours	8 jours	1 jour	9 jours	0 jour
4	Parking ET	3 jours	3 jours	0 jour	3 jours	0 jour
5	Parking PGM	38 jours	30 jours	8 jours	38 jours	0 jour
6	Parking TEST	3 jours	2 jours	1 jour	3 jours	0 jour
7	**Lot Véhicule**	**17 jours**	**0 jour**	**17 jours**	**17 jours**	**0 jour**
8	Véhicule ET	2 jours	2 jours	0 jour	2 jours	0 jour
9	Véhicule PGM	12 jours	15 jours	-3 jours	12 jours	0 jour
10	Véhicule TEST	3 jours	1,5 jours	1,5 jours	3 jours	0 jour
11	**Lot Autorisation**	**35,5 jours**	**0 jour**	**35,5 jours**	**33 jours**	**2,5 jours**
12	Autorisation JE	5 jours	8 jours	-3 jours	5 jours	0 jour
13	Autorisation ET	2 jours	2 jours	0 jour	2 jours	0 jour
14	Autorisation PGM	26 jours	25 jours	1 jour	26 jours	0 jour
15	Autorisation TEST	2,5 jours	2,5 jours	0 jour	0 jour	2,5 jours
16	**Lot Edition**	**22,5 jours**	**0 jour**	**22,5 jours**	**9 jours**	**13,5 jours**
17	Edition JE	4 jours	3 jours	1 jour	4 jours	0 jour
18	Edition ET	2 jours	1,5 jours	0,5 jour	2 jours	0 jour
19	Edition PGM	15 jours	15 jours	0 jour	3 jours	12 jours
20	Edition TEST	1,5 jours	1,5 jours	0 jour	0 jour	1,5 jours
21	**Lot Intégration**	**10 jours**	**0 jour**	**10 jours**	**0 jour**	**10 jours**
22	Intégration	10 jours	10 jours	0 jour	0 jour	10 jours
23	Fin	0 jour	0 jour	0 jour	0 jour	0 jour

Le champ Planifié des tâches récapitulatives n'a pas été mis à jour et reste à 0 pour les cinq lots. On va introduire ces tâches dans la planification.

On sélectionne les cinq tâches récapitulatives (à l'aide de la touche Ctrl).

On utilise **Outils, Suivi, Enregistrer la planification initiale…** et on choisit :



On observe alors que le planifié des tâches récapitulatives a été mis à jour et correspond à la somme (Réel + Restant). Si on filtre le projet par **Projet**, **Filtré pour** : **Tâches récapitulatives**, on obtient le tableau suivant :

	Nom de la tâche	Travail	Planifié	Variation	Réel	Restant
0	⊟ **Parking**	**138 jours**	**130 jours**	**8 jours**	**112 jours**	**26 jours**
2	⊞ **Lot Parking**	**53 jours**	**43 jours**	**10 jours**	**53 jours**	**0 jour**
7	⊞ **Lot Véhicule**	**17 jours**	**18,5 jours**	**-1,5 jours**	**17 jours**	**0 jour**
11	⊞ **Lot Autorisation**	**35,5 jours**	**37,5 jours**	**-2 jours**	**33 jours**	**2,5 jours**
16	⊞ **Lot Edition**	**22,5 jours**	**21 jours**	**1,5 jours**	**9 jours**	**13,5 jours**
21	⊞ **Lot Intégration**	**10 jours**	**10 jours**	**0 jour**	**0 jour**	**10 jours**

Le lot Parking a une valeur incorrecte, car juste avant la planification initiale on a ajouté 2 jours de charge supplémentaire à Claude pour la tâche 'Parking JE': on modifie le planifié en reprenant les valeurs initiales et en conservant la valeur de 6 jours pour la tâche 'Parking JE', c'est-à-dire :

Lot Parking	41

On peut maintenant filtrer le projet par **Projet**, **Filtré** pour : **Tâches récapitulatives** et on obtient le tableau d'avancement suivant par **Affichage**, **Gantt suivi** avec **Table : Travail** :

	Nom de la tâche	Travail	Planifié	Variation	Réel	Restant
0	⊟ Parking	138 jours	130 jours	8 jours	112 jours	26 jours
2	⊞ Lot Parking	53 jours	41 jours	12 jours	53 jours	0 jour
7	⊞ Lot Véhicule	17 jours	18,5 jours	-1,5 jours	17 jours	0 jour
11	⊞ Lot Autorisation	35,5 jours	37,5 jours	-2 jours	33 jours	2,5 jours
16	⊞ Lot Edition	22,5 jours	21 jours	1,5 jours	9 jours	13,5 jours
21	⊞ Lot Intégration	10 jours	10 jours	0 jour	0 jour	10 jours

Ce tableau pourra être fourni périodiquement au maître d'ouvrage.

15.5 BILAN DU PROJET PARKING

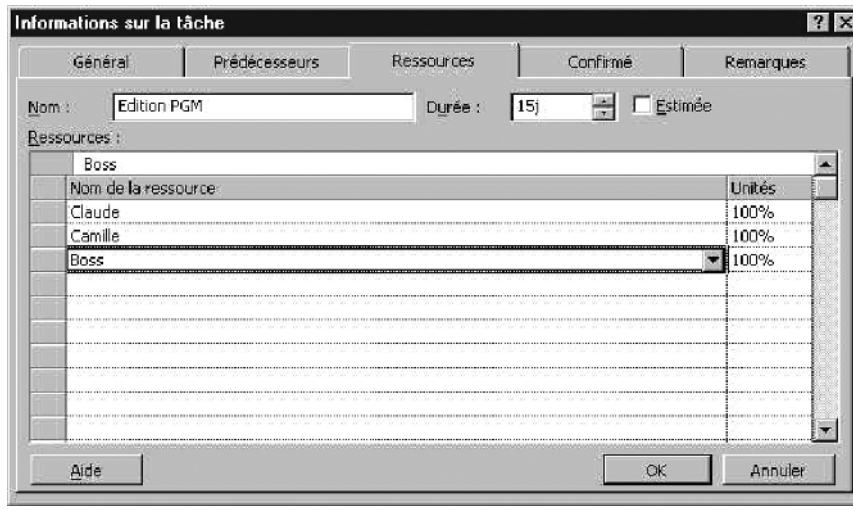
15.5.1 Tableau d'avancement final du projet Parking

On effectue les dernières mises à jour des travaux d'avril. Compte tenu de l'échéance, plusieurs décisions ont été prises :

Camille est rentrée dès le 5 avril au matin et elle est venue travailler trois samedis : les 8, 15 et 22 avril. On met donc à jour son calendrier.

La tâche 'Édition PGM' a été répartie entre trois personnes : Claude, Camille et Boss. On procède de la façon suivante :

Sur la vue **Affichage**, **Diagramme de Gantt** avec **Table : Entrée**, on rajoute la tâche 'Édition PGM' comme prédécesseur de la tâche Intégration. On modifie les ressources affectées à la tâche 'ÉditionPGM' par :



On peut observer que la durée a été réduite automatiquement par MS Project :

	Nom de la tâche	Durée	Début	Fin	Prédécesseurs	Noms ressources
0	**Parking**	**76,75 jours**	**Lun 02/01/06**	**Jeu 20/04/06**		
1	Début	0 jour	Lun 02/01/06	Lun 02/01/06		
11	**Lot Autorisation**	**67,5 jours**	**Lun 02/01/06**	**Ven 07/04/06**		
15	Autorisation TEST	2,5 jours	Mer 05/04/06	Ven 07/04/06	18	Camille
16	**Lot Edition**	**13,25 jours**	**Jeu 23/03/06**	**Mar 11/04/06**		
19	Edition PGM	9,75 jours	Mer 29/03/06	Sam 08/04/06	6	Claude;Camille;Boss
20	Edition TEST	1,5 jours	Sam 08/04/06	Mar 11/04/06	19	Claude
21	**Lot Intégration**	**8,75 jours**	**Sam 08/04/06**	**Jeu 20/04/06**		
22	Intégration	10 jours	Sam 08/04/06	Jeu 20/04/06	15;19	Camille
23	Fin	0 jour	Jeu 20/04/06	Jeu 20/04/06	22	

Nous allons maintenant **Mettre à jour le projet**. L'avancement réel de chaque ressource se présente comme suit :

Claude						1/4	Reste à faire
Édition PGM						1	6
	3/4	4/4	5/4	6/4	7/4	8/4	Reste à faire
Édition PGM	1	1	1	1	1	1	3
	10/4	11/4	12/4	13/4	14/4	15/4	Reste à faire
Édition PGM	1	1	1	1	1	1	0
	17/4	18/4	19/4	20/4	21/4	22/4	Reste à faire
Édition TEST	1	1					0

Camille						1/4	Reste à faire
Congé						1	
	3/4	4/4	5/4	6/4	7/4	8/4	Reste à faire
Autorisation TEST			1	1	1		0
Édition PGM						1	2,5
	10/4	11/4	12/4	13/4	14/4	15/4	Reste à faire
Édition PGM	1	1	1	0,5			0
Intégration				0,5	1	1	7,5
	17/4	18/4	19/4	20/4	21/4	22/4	Reste à faire
Intégration	1	1	1	1	1	1	0

Boss						1/4	Reste à faire
	3/4	4/4	5/4	6/4	7/4	8/4	Reste à faire
Édition PGM	1	1	1				0

Après mise à jour, le récapitulatif, filtré sur les tâches récapitulatives, se présente ainsi :

N°	Init. Ress.	Nom de la tâche	% Trav.Achevé	Trav.Réel	Trav.Restant	Trav.Planifié	Var.Travail
0		⊟ Parking	100%	146 jours	0 jour	130 jours	16 jours
2		⊟ Lot Parking	100%	53 jours	0 jour	41 jours	12 jours
7		⊟ Lot Véhicule	100%	17 jours	0 jour	18,5 jours	-1,5 jours
11		⊟ Lot Autorisation	100%	36 jours	0 jour	37,5 jours	-1,5 jours
16		⊟ Lot Edition	100%	31,5 jours	0 jour	21 jours	10,5 jours
21		⊟ Lot Intégration	100%	8,5 jours	0 jour	10 jours	-1,5 jours

On peut y observer que le Travail planifié de la tâche récapitulative du projet Parking est de 130 jours, et non de 128. C'est en effet la valeur de la planification initiale, avec une charge affectée différente de la charge estimée. On peut, si on le souhaite, ramener cette valeur à **128** jours.

15.5.2 Tableau de bilan du projet Parking

Pour construire un bilan des écarts de charge, on va supprimer les tâches récapitulatives et opérer un regroupement par nature de tâche.

Pour cela, on sélectionne la vue **Affichage**, **Diagramme de Gantt** avec **Table : Entrée**. Ensuite, on procède de la façon suivante :

On remonte toutes les tâches de second niveau : on obtient une liste avec toutes les tâches au même niveau.

On peut alors supprimer les quatre tâches : Lot Parking, Lot Autorisation, Lot Véhicule et Lot Édition, sans supprimer les tâches figurant dans les lots.

On crée de nouvelles tâches récapitulatives, correspondant aux types de tâche utilisés pour l'estimation et pour lesquels on veut suivre les écarts entre le planifié et le réel.

On déplace les tâches par Couper-Coller de façon à ce que les nouvelles tâches récapitulatives soient situées juste au-dessus des tâches qu'elles récapitulent.

La typologie est la suivante :

Type de tâche	Tâches
Jeu essai	Parking JE Autorisation JE Édition JE
Étude technique temps réel	Parking ET Véhicule ET Autorisation ET
Étude technique batch	Édition ET
Programme moyen temps réel	Parking PGM Véhicule PGM
Programme difficile temps réel	Autorisation PGM
Programme batch facile	Édition PGM
Test	Parking TEST Véhicule TEST Autorisation TEST Édition TEST
Intégration	Intégration

On obtient une nouvelle liste hiérarchisée qui se présente ainsi :

N°	Init. Ress.	Nom de la tâche	% Trav.Achevé	Trav.Réel
0		**Parking**	**100%**	**146 jours**
1		Début	0%	0 jour
2		**JEU ESSAI**	**100%**	**18 jours**
3	CL	Parking JE	100%	9 jours
4	CA	Autorisation JE	100%	5 jours
5	BO	Edition JE	100%	4 jours
6		**ETUDE TECHNIQUE**	**100%**	**7 jours**
7	CL	Parking ET	100%	3 jours
8	CA	Véhicule ET	100%	2 jours
9	CA	Autorisation ET	100%	2 jours
10		**ETUDE TECHNIQUE**	**100%**	**2 jours**
11	CA	Edition ET	100%	2 jours
12		**PROGRAMME facil**	**100%**	**23,5 jours**
13	CL;CA;BO	Edition PGM	100%	23,5 jours
14		**PROGRAMME moy**	**100%**	**50 jours**
15	CL	Parking PGM	100%	38 jours
16	CA	Véhicule PGM	100%	12 jours
17		**PROGRAMME diffi**	**100%**	**26 jours**
18	CA	Autorisation PGI	100%	26 jours
19		**TEST**	**100%**	**11 jours**
20	CL	Parking TEST	100%	3 jours
21	CA	Véhicule TEST	100%	3 jours
22	CA	Autorisation TES	100%	3 jours
23	CL	Edition TEST	100%	2 jours
24		**INTEGRATION**	**100%**	**8,5 jours**
25	CA	Intégration	100%	8,5 jours
26		Fin	0%	0 jour

Pour visualiser les écarts, on utilise **Affichage**, **Gantt suivi** ; on crée une table **Bilan** contenant les cinq champs **N°**, **Nom**, **Travail planifié**, **Travail réel** et **Variation de travail**. On ajuste les valeurs du travail planifié des tâches récapitulatives. En filtrant sur les tâches récapitulatives, on obtient le tableau suivant :

N°	NOM	Trav. Planifié	Trav. Réel	Var. Travail
0	**Parking**	**130 jours**	**146 jours**	**16 jours**
1	Début	0 jour	0 jour	0 jour
2	**JEU ESSAI**	**19 jours**	**18 jours**	**-1 jour**
6	**ETUDE TECHNIQUE temps réel**	**7 jours**	**7 jours**	**0 jour**
10	**ETUDE TECHNIQUE batch**	**1,5 jours**	**2 jours**	**0,5 jour**
12	**PROGRAMME facile batch**	**15 jours**	**23,5 jours**	**8,5 jours**
14	**PROGRAMME moyen temps réel**	**45 jours**	**50 jours**	**5 jours**
17	**PROGRAMME difficile temps réel**	**25 jours**	**26 jours**	**1 jour**
19	**TEST**	**7,5 jours**	**11 jours**	**3,5 jours**
24	**INTEGRATION**	**10 jours**	**8,5 jours**	**-1,5 jours**
26	Fin	0 jour	0 jour	0 jour

Ainsi s'achève l'illustration de l'outil MS Project, avec lequel nous avons proposé des choix de planification et de suivi cohérents avec les principes présentés au chapitre 12. Le progiciel offre d'autres possibilités, pour lesquelles nous renvoyons à sa documentation propre. Nous avons essayé de concilier d'une part la simplicité — des manipulations et de la lecture des différents tableaux de bord — et d'autre part les informations de suivi indispensables au chef de projet. Le cas Parking a permis de simuler un avancement et des actions de pilotage réalistes, de la planification jusqu'au bilan.

Troisième partie

Les référentiels et la certification en management de projet

Cette troisième partie s'adresse au lecteur qui souhaite approfondir sa connaissance des référentiels et/ou préparer une certification en management de projet, et tout particulièrement celle du PMI.

Le chapitre 16 présente les principales associations professionnelles dans le domaine de la gestion de projet, ainsi que la norme ISO 10006 qui traite de la qualité spécifique au management de projet. On y a ajouté le cadre proposé par Eurométhode pour les appels d'offres publics au niveau européen.

Le chapitre 17 constitue une bonne introduction à la certification PMI : il comporte de nombreux renvois aux différentes techniques exposées dans la première partie de cet ouvrage.

16

Associations, normes et cadres de référence

16.1 LES ASSOCIATIONS PROFESSIONNELLES

L'AFITEP (*Association Francophone de Management de Projet*)[1] anciennement Association Française des Ingénieurs et Techniciens en Évaluation et Planification de Projet, fondée en 1982, organise des rencontres et publie une revue professionnelle, *La Cible*. Elle est associée aux principales fédérations internationales de management de projet

Le PMI (*Project Management Institute*)[2], association américaine de professionnels, fondée en 1969, est aujourd'hui fédérée en chapitres nationaux et regroupe près de 100 000 membres dans le monde. Son référentiel a été reconnu comme une norme aux États-Unis par l'ANSI, institut de normalisation américain.

L'IPMA (*International Projet Management Association*)[3], association internationale fondée en 1967, fédère près de 40 associations nationales. Elle organise des congrès et des échanges, et regroupe environ 25 000 membres.

L'ICEC (*International Cost Engineering Council*)[4] est une fédération d'associations orientées vers la maîtrise des coûts, qui comprend environ 50 000 adhérents dans 38 pays.

---

1. http://www.afitep.fr.
2. http://www.pmi.org.
3. Association internationale de gestion de projet, http :// www.ipma.ch.
4. Conseil international de l'ingénierie des coûts, http://www.icoste.org.

L'objectif de certification a conduit à une définition normalisée du vocabulaire et des compétences requises pour conduire des projets. De plus, compte tenu de l'importance prise par les organisations du travail en mode projet, les organismes de normalisation ont également travaillé sur la définition des termes et des activités.

Pendant longtemps, la gestion de projet n'a pas été considérée comme une discipline académique, au même titre que les autres spécialités de la gestion. Depuis quelques années, la gestion de projet n'est plus étrangère à la recherche académique, ce que l'on peut observer à travers deux phénomènes : d'une part des revues spécialisées, comme l'*International Journal of Project Management* ou *Project Management Journal* font une large place à la recherche ; d'autre part, les revues scientifiques, notamment dans les domaines système d'information et informatique, s'intéressent de plus en plus à la gestion des projets. Une sélection d'articles est donnée dans la bibliographie.

16.2 LA NORME ISO10006

La norme ISO10006[1] traite de la qualité dans le management de projet et décrit, de façon structurée, les activités à accomplir pour gérer un projet avec des exigences de qualité. La version 2003[2] est alignée sur la norme ISO9000 version 2000, qui traite du management de la qualité et défend une vision par processus de la gestion d'une organisation. Le processus est défini comme un « *ensemble d'activités corrélées ou interactives qui transforme des éléments d'entrée en éléments de sortie* ».

Pour la **norme ISO10006 : 2003**, les processus sont regroupés en quatre familles.

- La première ne comprend qu'un seul processus, appelé *processus stratégique*, qui est de la responsabilité de la direction de l'entreprise.
- La deuxième famille comprend des processus qui touchent à la *gestion des ressources*, en particulier le personnel.
- La troisième famille de processus touche à la *réalisation* du produit, c'est-à-dire la production du bien ou service visé par l'objectif. Elle comprend la coordination des activités, la maîtrise des tâches de production, celle des délais et des coûts, la gestion de la communication, la maîtrise des risques et celles des achats pour le projet.
- La quatrième famille de processus touche à *l'amélioration*, c'est-à-dire le management des connaissances issues des projets achevés.

1. Elle s'intitule « Systèmes de management de la qualité. Lignes directrices pour le management de la qualité dans les projets ».
2. Cette version est actuellement en cours de révision, avec notamment la participation du PMI.

Nous allons détailler le contenu de chaque famille.

1. Famille du processus stratégique.

Le but de ce processus est d'inciter la direction de l'entreprise à mettre en place un dispositif qualité adapté au projet. Après avoir nommé un chef de projet, la norme recommande que la direction soit attentive à plusieurs aspects dans le déroulement du projet, notamment :

- la prise en compte des besoins et des attentes du client et des autres parties prenantes du projet (personnel, fournisseurs, banquiers, syndicats, partenaires, société…) ;
- l'implication des acteurs à tous les niveaux ;
- l'établissement de relations mutuellement bénéfiques avec les fournisseurs.

De plus, chaque projet doit être l'occasion d'améliorations dans le management de projet.

2. Famille des processus relatifs aux ressources et au personnel.

Les *processus relatifs aux ressources* ont pour but d'une part leur planification, c'est-à-dire identifier, estimer et allouer celles qui seront nécessaires ; d'autre part leur contrôle, c'est-à-dire s'assurer qu'elles sont suffisantes pour atteindre les objectifs du projet.

Les ressources doivent s'entendre au sens large : équipements, installations, finance, informations, matériels, logiciels, personnel, services et locaux…

Les *processus relatifs au personnel* ont pour but de créer un environnement dans lequel le personnel peut apporter une contribution efficace et efficiente au projet. Cela implique de définir la structure du projet (les rôles et les responsabilités), de sélectionner et affecter le personnel, et de former l'équipe pour obtenir les compétences individuelles et collectives nécessaires.

3. Famille des processus relatifs au produit.

Les *processus relatifs à la coordination* ont pour but de gérer les interdépendances entre les travaux et les actions. Pour cela, il faut :

- lancer le projet et élaborer le plan de management de projet, document qui décrit comment on va atteindre les objectifs du projet ;
- gérer les interactions entre les travaux des différents acteurs ;
- gérer les modifications, notamment celles qui touchent le périmètre ou les objectifs du projet ;
- clore le projet et faire le bilan.

Les *processus relatifs au contenu* du projet ont pour objectif de s'assurer de l'achèvement et de la qualité du produit. Pour cela, il faut :

- traduire les besoins et attentes du client et des autres parties prenantes en activités nécessaires pour accomplir les objectifs du projet, et organiser ces activités (définition des activités)
- s'assurer que le personnel travaille dans le périmètre du projet durant l'exécution des activités (contrôle des activités)
- s'assurer que les résultats satisfont aux exigences (contrôle des activités).

Les *processus relatifs aux délais* ont pour but de maîtriser les délais. Pour cela, il faut :

- identifier les liaisons entre les tâches ;
- estimer les durées des tâches ;
- élaborer le planning ;
- comparer le réel et le planifié, et prendre des mesures correctrices (contrôle des délais).

Les *processus relatifs aux coûts* ont pour but de maîtriser les coûts du projet. Pour cela, il faut :

- estimer les coûts ;
- faire un budget, qui montre la répartition des dépenses dans le temps ;
- suivre les dépenses réelles, estimer les tendances, identifier l'origine des écarts par rapport au budget et prendre des mesures (contrôle des coûts).

Les *processus relatifs à la communication* ont pour but de faciliter les échanges internes et externes d'information, nécessaires au projet. Pour cela, il faut :

- planifier les actions de communication (quoi, qui, comment) ;
- gérer l'information (utiliser des procédures pour contrôler la préparation de l'information, la collecte, l'identification, la classification, la mise à jour, la distribution, le stockage, l'archivage, la protection, l'accès, la durée de conservation, la suppression) ;
- contrôler la communication, pour éviter les malentendus et les conflits.

Les *processus relatifs aux risques* ont pour but de minimiser l'impact d'événements potentiellement négatifs pour le projet et de tirer parti d'opportunités d'amélioration.

Pour cela, il faut :

- identifier les risques (coût, délai, produit, sécurité, santé, environnement…) ;
- évaluer les risques (en utilisant l'expérience et les données mémorisées de projets antérieurs) ;
- traiter les risques, c'est-à-dire élaborer des solutions pour éliminer, réduire, transférer, partager ou supporter les risques ;
- contrôler les risques, c'est-à-dire suivre et gérer les aléas tout le long du projet.

Les *processus relatifs aux achats* ont pour but d'obtenir des produits/services pour le projet. Pour cela, il faut :

- planifier les achats, en élaborant un plan d'achat ou de sous-traitance ;
- documenter les exigences d'achat ;
- évaluer les fournisseurs ;
- contracter, c'est-à-dire évaluer les offres et établir le contrat ;
- contrôler le contrat, c'est-à-dire s'assurer que les conditions du contrat sont satisfaites.

4. Famille des processus relatifs à l'amélioration.

Les *processus relatifs à l'amélioration* ont pour but tirer les leçons du projet. Pour cela, il faut :

- utiliser les résultats des mesures et de l'analyse des données du projet (actions correctives, actions préventives) et stocker les enseignements (prévention de la perte d'expérience).

Les *processus relatifs à la mesure et l'analyse* ont pour but de produire une trace et une évaluation de chaque projet. Pour cela, il faut :

- Mesurer, collecter et valider les données pour une amélioration efficace et efficiente de la performance (audit, évaluation des activités individuelles, évaluation des processus, mesure de l'atteinte des objectifs du projet…)

Les *processus relatifs à l'amélioration continue* ont pour but d'améliorer en permanence la qualité globale des projets de l'entreprise. Pour cela, il est recommandé de mettre en place un cycle d'amélioration continue basé sur la roue de Deming : Plan-Do-Check-Act (PDCA), planifier-faire-vérifier-agir.

16.3 LE CADRE EUROMÉTHODE

16.3.1 Présentation d'Eurométhode

Eurométhode est un projet européen qui a été financé par le Groupe des Marchés Publics de la Commission Européenne[1] afin de définir un cadre contractuel pour les projets système d'information. Il prend en compte tout contexte d'adaptation d'un système d'information, c'est-à-dire « toute forme de modification, amélioration, automatisation, etc. du système d'information, (ce qui inclut…) développement, maintenance, conception, rétroconception, installation de système, etc. ». La définition donnée d'un projet est plus restrictive que les définitions normalisées et correspond à l'organisation humaine temporaire permettant de le mener à bien : « une adaptation est réalisée par une équipe qu'on appelle le projet ».

Eurométhode décrit les obligations contractuelles respectives du client et du fournisseur sous l'angle des éléments à livrer et fournit ainsi un cadre de référence des différentes catégories de fournitures susceptibles d'être produites, par le fournisseur et/ou par le client, au cours d'un projet (figure 16.1).

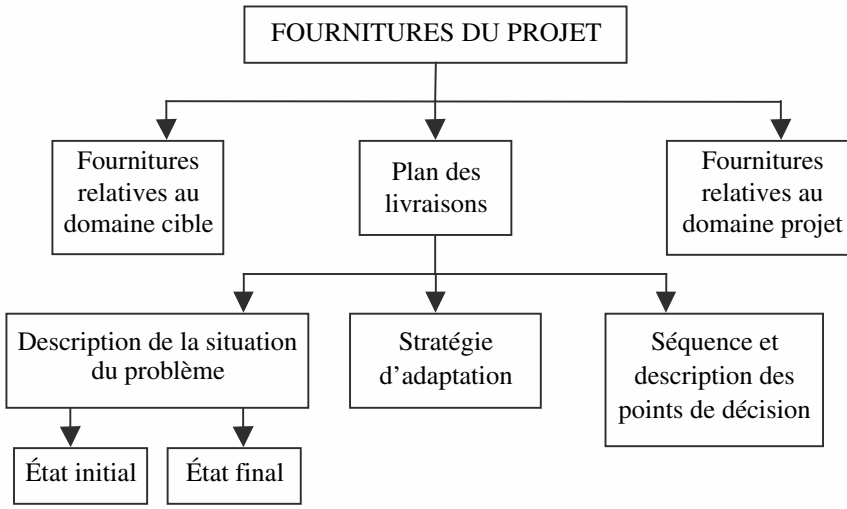


Figure 16.1 — Fournitures d'un projet.

1. PPG : *Public Procurement Group*, qui fait partie de la DGXIII de la Commission Européenne. En France sa diffusion s'est effectuée en 1995 sous le contrôle de la Commission Centrale des Marchés.

Le *plan des livraisons* est élaboré au moment de la passation du marché ou au début du projet. Il peut être affiné par la suite. Il décrit les engagements réciproques du client et du fournisseur en termes d'informations à fournir ou de produits à livrer.

Les *fournitures relatives au domaine cible* sont celles pour lesquelles le client paie réellement et pour lesquelles il fait appel à un fournisseur. Elles correspondent :

- soit à tout ou partie du système d'information opérationnel, c'est-à-dire l'application ou une partie de l'applicatif ;
- soit à des descriptions du système d'information considéré : description de l'existant, description du système cible, documentation, plan de formation…

Les *fournitures relatives au domaine du projet* permettent au client d'avoir une visibilité sur l'avancement et la qualité des travaux effectués par le fournisseur.

Nous allons préciser le contenu de ces trois types de fourniture.

16.3.2 Le plan des livraisons

Le plan des livraisons contient tout d'abord la *stratégie d'adaptation* qui correspond en partie au choix d'un modèle de cycle de vie et au choix d'un modèle de mise en œuvre. Eurométhode propose différentes options (figure 16.2).

Stratégie d'adaptation	Choix
Démarche de description du système	Analytique (modélisation et rédaction de spécifications) Expérimental (approche par prototypage) Type de coopération : projet conduit par des experts Type de coopération : participative
Démarche de construction du système	En une seule fois Incrémentale (le système est construit progressivement par modules) Évolutionnaire (le système est construit par versions successives)
Démarche de mise en service du système	En une seule fois Incrémentale (le système est mis en œuvre progressivement par modules) Évolutionnaire (les différentes versions du système sont installées successivement)

Figure 16.2 — Options de la stratégie d'adaptation selon Eurométhode.

La *séquence et la description des points de* décision correspondent aux différents jalons que l'on trouve dans un plan de développement. Chaque jalon comprend la remise d'une fourniture au client (note, rapport, dossier, programmes…) et l'attente d'une prise de décision par le client (choix entre plusieurs propositions, validation, décision sur la suite du projet…).

La *description de la situation du problème* est fournie par le client. Elle comprend la description de l'*état initial* et celle de l'*état final*.

L'état initial doit permettre au fournisseur de situer le projet dans son contexte et de savoir ce qui s'est passé antérieurement. Il doit en particulier être informé sur l'origine du projet, le champ couvert, les structures concernées, les résultats d'éventuelles études antérieures, l'état de l'application existante, les contraintes.

L'état final correspond aux objectifs visés et résultats attendus par le client.

16.3.3 Les fournitures relatives au domaine cible

Les fournitures relatives au domaine cible sont soit des fournitures qui décrivent le système d'information (modèles, rapports, spécifications…), soit des éléments opérationnels (modules, application…) qui relèvent du système informatique. Dans la perspective d'une approche agile, on mettrait aujourd'hui l'accent sur les livraisons de composants logiciels.

16.3.4 Les fournitures du domaine projet

Eurométhode distingue deux sous-catégories : les plans de projet et les comptes rendus de projet.

Les fournitures relatives aux *plans de projet* permettent de contrôler le processus de production interne, de s'assurer que les objectifs sont atteints et de définir le niveau de qualité nécessaire pour les fournitures du domaine cible. Le plan de projet fait référence à des fournitures décrites dans le plan de livraison.

Les fournitures relatives au *compte rendu de projet* sont utilisées pour évaluer l'adéquation des fournitures du domaine cible ainsi que leur contenu, pour déterminer si les exigences définies pour le processus sont satisfaites et pour documenter la façon dont les activités prévues par les plans ont été accomplies.

16.4 LE CADRE PRINCE2

Le référentiel PRINCE2[1] (*PRojects IN Controlled Environments*) a été commandité par le gouvernement britannique, à l'origine pour les projects publics. Repris plus largement dans le secteur privé, en Grande-Bretagne d'abord, puis en Europe, il donne aujourd'hui lieu à une certification. Son objectif majeur est de réduire les risques des projets, par une maîtrise accrue.

Le référentiel présente un modèle des processus qui doivent être mis en œuvre pour maîtriser la conduite du projet (figure 16.3).

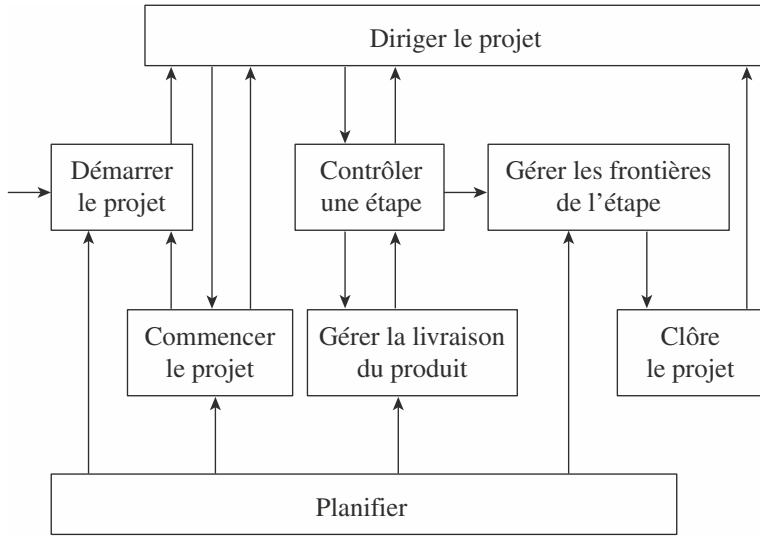


Figure 16.3 — Modèle des processus de PRINCE2.

« Démarrer le projet » correspond au lancement et n'intervient qu'une seule fois dans le cycle de vie du projet.

« Diriger le projet » interagit avec tous les autres processus durant toute la durée du projet. Ce processus est placé sous la responsabilité du comité de direction du projet.

« Commencer le projet » n'est exécuté qu'une seule fois dans le cycle de vie d'un projet. Il vise à produire un premier document de management de projet qui comprend notamment les objectifs du projet, le contenu du projet, les livrables, ainsi que l'organisation du projet et un début de plan de management de projet (cf. § 6.3).

1. http://www.prince2.com/

« Planifier » est un processus commun pour la planification des différents aspects du projet.

« Contrôler une étape » est un processus itératif, qui est répété pour chacune des étapes du cycle de vie du projet. Il guide le chef de projet dans son pilotage au quotidien.

«Gérer la livraison du produit » propose un mécanisme pour suivre la production depuis l'affectation des tâches jusqu'à la fourniture du livrable correspondant.

«Gérer les frontières de l'étape » permet de gérer la transition d'une étape à l'autre, et inclut la capitalisation d'expérience.

« Clôre le projet » est le processus qui officialise l'achèvement du projet.

Par ailleurs, PRINCE2 décrit huit « composants », auxquels sont attachés des connaissances qui seront mises en œuvre dans les différents processus de management de projet. On y retrouve des aspects classiques : gestion des risques, gestion de la qualité, pilotage du projet… Citons en particulier *Business case*, composant qui oriente tout le projet, dans le sens où il décrit les attentes métier liées au projet et sert de base aux revues du projet.

Le référentiel n'est pas en contradiction avec le référentiel du PMI présenté au chapitre 17. Il est globalement moins complet que le PMBOK et parfois plus directif. Il fournit un ensemble de documents types qui peuvent être utilisés dans l'administration d'un projet.

17

La préparation de la certification PMI

17.1 LA CERTIFICATION DU PMI

17.1.1 Le choix du PMI

La certification en management de projet rencontre un intérêt croissant depuis quelques années. Nous avons indiqué au début de cet ouvrage (paragraphe 1.1) l'existence d'organisations professionnelles délivrant des certifications. Aujourd'hui, on peut hésiter entre deux certifications internationales : celle de l'IPMA (*International Project Management Association*), axée sur la reconnaissance des compétences du candidat à exercer correctement la fonction de chef de projet ou directeur de projet (paragraphe 1.3.2) ; et celle du PMI (*Project Management Institute*) qui repose sur la validation de la compréhension et l'assimilation des connaissances de référence. Cet ouvrage étant consacré aux principes et techniques de management de projet, nous avons fait le choix de proposer une solide introduction pour ceux qui, se rattachant au domaine des systèmes d'information, ont l'objectif d'obtenir une certification PMI.

Pour cela, nous avons d'abord exposé les règles de la certification. Ensuite, nous décrivons les concepts clés et la logique propre au PMI, qui ne contredisent pas l'ensemble de ce qui figure de façon détaillée dans cet ouvrage. Enfin, nous reprenons les différents aspects du management d'un projet avec la terminologie PMI et nous établissons une correspondance avec les techniques exposées en détail et illustrées dans les deux premières parties.

17.1.2 L'apport à la préparation de la certification

Ce chapitre 17 se veut une présentation claire et structurée de la démarche un peu abstraite portée par PMI. Ses principes et sa logique apparaîtront plus facilement au lecteur déjà averti par son parcours dans nos deux premières parties.

Il est également possible de lire cet ouvrage en enchaînant les chapitres 1 et 17, et en se reportant ensuite aux chapitres intermédiaires en fonction des renvois aux techniques particulières indiquées dans les paragraphes 17.4 à 17.12 ci-dessous.

Après lecture du chapitre 17, le lecteur pourra aborder de façon efficiente et efficace l'étude du document de référence du PMI, accompagné éventuellement d'un ouvrage spécialisé et de préférence en s'appuyant sur une base de simulation de tests qui permet de se familiariser avec la durée imposée et le style des questions posées. Certains titres choisis (ouvrage et logiciel) sont donnés dans la bibliographie en annexe

17.2 LE CADRE DE LA CERTIFICATION DU PMI

17.2.1 Le référentiel

L'examen de certification du PMI s'appuie sur le PMBOK, *Project Management Body of Knowledge*, qui rassemble un ensemble de connaissances ayant fait l'objet d'une large reconnaissance. Les principes et techniques exposés sont applicables à la majorité des projets et leur mise en œuvre devrait améliorer les chances de succès des projets. Cependant, le référentiel du PMI n'offre pas, sauf exception, de description des. techniques de management de projet recommandées.

Par ailleurs, le référentiel est général et n'introduit pas certaines particularités liées à des caractéristiques du projet, notamment le domaine d'application. Or, si certaines techniques sont communes à tous les projets (techniques de planification des délais, par exemple), d'autres se déclinent de façon spécifique selon le domaine (techniques d'estimation de la charge de travail, par exemple). Dans ce chapitre, nous mettons en relation les techniques génériques indiquées par le PMBOK et les techniques applicables aux projets de système d'information décrites dans les huit premiers chapitres de cet ouvrage.

La version en cours du PMBOK est la 3e et elle date de 2004. Elle a fait l'objet de traduction en différentes langues, dont le français.

17.2.2 L'examen de certification

Le PMI propose deux types de certification : PMP, *Project Management Professional*, et CAPM, *Certified Associate in Project Management*.

Pour obtenir la certification PMP, qui est la plus répandue actuellement, il faut d'abord soumettre une demande d'éligibilité. Pour être accepté, il faut satisfaire à deux exigences : d'une part, avoir suivi au moins 35 heures de cours en management de projet : d'autre part, faire état d'expériences de travail en mode projet dont la durée est variable selon le niveau d'études (trois années durant les six dernières années si l'on est titulaire d'un diplôme d'enseignement supérieur, ou cinq années durant les 8 dernières années sinon). La demande s'effectue par internet auprès du siège américain de l'association, et la réponse est envoyée au bout de quelques jours. Il s'agit alors de s'inscrire auprès d'un des centres d'examen habilités. L'épreuve est individuelle et informatisée : elle comporte 200 questions à choix multiples, aussi bien à caractère théorique que portant sur des mises en situation, et le temps imparti est de quatre heures. Les questions, qui sont les mêmes quel que soit l'endroit du globe où l'on se présente, sont rédigées en anglais, mais une traduction peut être affichée sur commande.

Le PMI propose également une certification CAPM qui est destinée aux assistants du chef de projet. Le référentiel est le même, mais l'examen ne comporte que 150 questions, plutôt théoriques, et se déroule sur trois heures.

17.3 LA LOGIQUE GÉNÉRALE DU PMBOK

17.3.1 La structure du PMBOK et les domaines de connaissance

Le guide du PMBOK comprend trois chapitres introductifs qui présentent les notions clés, notamment la distinction entre le projet et le produit, le principe de structuration temporelle du projet et l'architecture des connaissances concernant les activités de management de projet. Celles-ci sont organisées en processus, selon la préconisation de la norme ISO9000 et ISO10006 (Cf. paragraphe 16.2). Les processus sont regroupés autour de huit points de vue devant être pris en compte dans la gestion d'un projet : le contenu, les délais, les coûts, la qualité, les ressources humaines, les communications, les risques, les approvisionnements, ainsi qu'un point de vue intégrateur où peuvent se faire les synthèses et les arbitrages.

Ces neuf points de vue, appelés « domaine de connaissance », comportent plusieurs processus. Chaque processus est décrit selon la structure de la figure 17.1, c'est-à-dire comme une boite noire visant à produire des sorties à partir de certaines entrées (pouvant être elles-mêmes des sorties d'un autre processus) en utilisant éventuellement des outils (moyens tangibles, par exemple logiciel) ou des techniques (activités organisées de façon procédurale pour résoudre un

problème ou apporter une aide, par exemple technique de calcul de la performance du projet) :

Figure 17.1 — Structure de description d'un processus dans le PMBOK.

17.3.2 Produit et projet

Le *produit* est un terme qui prend deux significations différentes selon le niveau auquel on se situe. À un niveau générique, il désigne le résultat visé par le projet. De façon plus détaillée, un produit peut prendre trois formes différentes : ce peut être un *produit* (dans le deuxième sens), c'est-à-dire un artefact utilisable qui possède une certaine matérialité (par exemple, un objet ou un logiciel) ; ce peut aussi être un *service*, c'est-à-dire l'accomplissement d'un travail utile (par exemple, la formation des utilisateurs) ; ce peut enfin être un *résultat*, c'est-à-dire un élément issu du projet et pouvant être constaté (par exemple, l'intégration de deux applications ou un document d'étude).

Le produit, dans le sens générique, doit être soigneusement distingué du *projet*, notamment lorsque l'on parle de contenu. Le *contenu du produit*, dont la description est de la responsabilité du commanditaire du projet, renvoie aux caractéristiques et spécifications du produit/service/résultat demandé. Le *contenu du projet*, qui est identifié par l'équipe de projet, comprend le travail nécessaire pour obtenir le produit visé.

Le passage du contenu du produit au contenu du projet s'effectue schématiquement en quatre étapes, qui peuvent se dérouler de façon cyclique :

1. Le commanditaire élabore une *Description du contenu du produit*, c'est-à-dire une description, parfois succincte, de ce qui est attendu. Cette description peut être enrichie au cours du projet. Si la description est susceptible d'évoluer profondément, il est recommandé de mettre en place une *Gestion de configuration*, c'est-à-dire une procédure d'enregistrement des caractéristiques du produit et de chacune des modifications.

   La description du contenu du produit fait en général partie d'un document plus global, fourni en entrée du projet et appelé *Énoncé des travaux du projet*, qui comporte notamment des éléments sur le contexte du projet. Le commanditaire peut ainsi indiquer les raisons qui ont conduit au lancement du projet et sa contribution au plan stratégique de l'entreprise.

2. L'équipe de projet va alors traduire, avec l'accord du commanditaire, le contenu du produit en *livrables*, c'est-à-dire en éléments précisant le produit

demandé et pouvant faire l'objet d'une vérification. Lorsque le commanditaire constatera officiellement la réalisation satisfaisante de tous les livrables du projet, celui-ci sera considéré comme achevé.

Certains livrables peuvent être liés au management de projet, par exemple la situation financière du projet à la fin de chaque mois.

À ces livrables « externes » viennent parfois s'ajouter des livrables « internes » visibles uniquement par l'équipe de projet.

3. Lorsque les livrables sont clairement identifiés, l'équipe de projet détermine les travaux à réaliser pour les obtenir et les consignes avec la liste des livrables dans un *Énoncé du contenu du projet*. Ce document, qui mentionne également les contraintes (budget, délai, ressources) et certaines exigences (sécurité, performance), constitue une référence commune et va guider le travail de l'équipe de projet.
4. Dans un dernier temps, l'équipe de projet planifie le travail à réaliser en élaborant une SDP, *Structure de découpage du projet* (aussi appelée WBS, *Work Breakdown Structure*), c'est-à-dire une décomposition hiérarchique, guidée par les livrables à produire, des travaux à réaliser à une maille parfois plus fine que celle de l'Énoncé du contenu du projet. On appelle *lot de travail* le niveau de décomposition le plus bas de la SDP.

   Celle-ci est la référence pour la planification et le suivi. Les logiciels de gestion de projet proposent de gérer des codes hiérarchisés permettant des regroupements dans les états de suivi des délais ou des coûts.

17.3.3 Les phases et le cycle de vie du projet

Les travaux identifiés dans la SDP vont ensuite être organisés dans le temps, à deux principaux niveaux. Au niveau le plus détaillé, ils vont être ordonnancés, et éventuellement donner lieu à des calculs de chemin critique. On peut parfois, à un niveau plus global, regrouper ces travaux en *phases* pour placer des jalons dans le déroulement du projet et réduire ainsi l'incertitude. Celles-ci sont des ensembles d'activités, débouchant sur un ou plusieurs livrables importants. La phase prend parfois le nom du livrable principal, par exemple Phase de Conception. Le contenu du projet apparaît alors comme structuré en phases qui sont généralement effectuées de façon séquentielle.

Le *cycle de vie du projet* est l'ensemble des phases d'un projet, s'il en comporte plusieurs. Une phase s'achève, en général, par une revue du travail accompli et des livrables, à l'issue de laquelle on décide de clore le projet ou de démarrer la phase suivante.

Une approche alternative au phasage peut être de distinguer plusieurs projets consécutifs. L'évaluation de l'incertitude guide le choix entre les deux options. Par

exemple, si le projet commence par une Étude de faisabilité et que la suite est très incertaine, on préférera en faire un projet complet avec une phase unique. Si au contraire, la suite est très probable, ce sera la première phase du cycle de vie d'un projet global.

Le cycle de vie du projet peut représenter une étape dans ce que l'on appelle le « cycle de vie du produit ». Par exemple, un système d'information commence à vivre dès son identification dans un plan stratégique jusqu'à l'abandon de l'application correspondante : le passage correspondant à la conception et mise en œuvre d'une nouvelle application (ou d'un nouveau progiciel) se fait sur le mode d'un projet avec son cycle de vie (figure17.2).

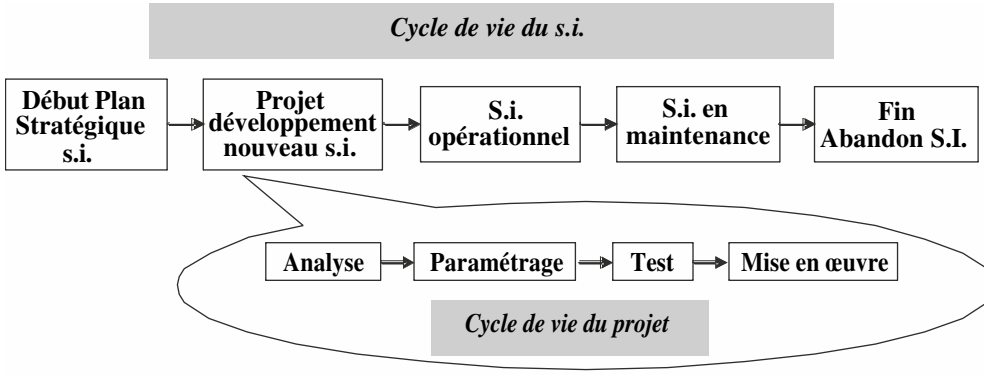


Figure 17.2 — Cycle de vie du projet et cycle de vie du produit.

Les modèles de développement décrits au chapitre 2 représentent des modèles de cycle de vie d'un projet.

17.3.4 Les processus de management de projet et les groupes de processus

Le Guide du PMBOK propose une répartition des activités de management de projet en 44 processus, liés entre eux via les données d'entrée et de sortie et répartis dans les 9 domaines de connaissance présentés au paragraphe 17.3.1. Ils font l'objet d'une seconde catégorisation, orthogonale à celle des domaines de connaissances et basée sur une logique séquentielle : les *groupes de processus*.

Les groupes de processus sont liés entre eux par une dépendance logique forte. Ils structurent la gestion de chacune des phases du projet. On distingue cinq groupes. Le groupe *Démarrage* comprend les processus à exécuter en début de chaque phase du projet dans le but d'autoriser officiellement le démarrage des travaux (par exemple élaborer une première version de l'Énoncé du contenu du projet). Le groupe *Planification* rassemble les processus de tous les domaines de connaissances qui visent à planifier les actions requises, sous un angle ou un autre

(contenu, coût, délais, ressources humaines, qualité, etc.). Le groupe *Exécution* comprend ceux des processus qui visent à faire réaliser les travaux prévus (affectation des ressources, sélection d'un sous-traitant, etc.). Le groupe *Surveillance et Maîtrise* sont les processus de pilotage du projet, qui permettent de surveiller les écarts entre le planifié et le réel et de prendre éventuellement des mesures correctives (par exemple, vérifier le contenu ou établir un rapport d'avancement). Le groupe *Clôture* vise à formaliser l'achèvement d'une phase.

En principe, pour la gestion de chacune des phases, on retrouve au moins un processus de chaque groupe dans l'ordre présenté à la figure 17.3 : les processus de Démarrage et de Clôture ne sont exécutés qu'une seule fois, alors que l'on peut avoir plusieurs cycles de processus de Planification - Exécution - Surveillance et Maîtrise.

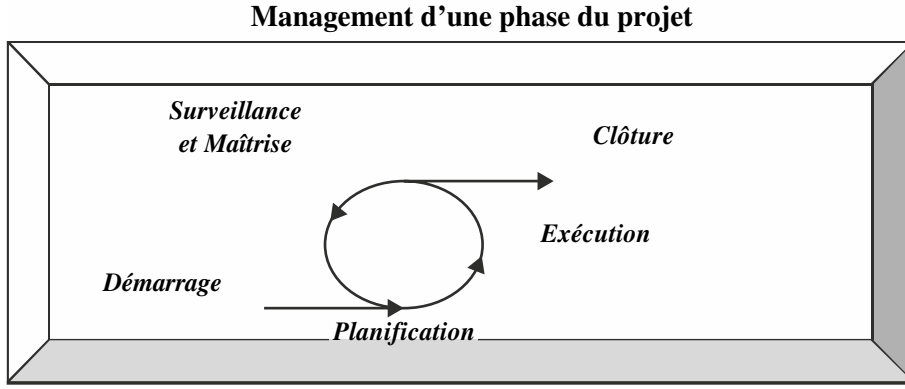


Figure 17.3 — Séquencement des groupes de processus d'une phase.

Au-delà de cet ordonnancement global, les processus de chaque groupe possèdent certains liens, que nous allons évoquer. Pour ce faire, nous avons utilisé la codification du PMBOK, qui attribue à chacun des 9 domaines de connaissance le numéro du chapitre dans lequel il est présenté :

- 4 = Management de l'intégration du projet
- 5 = Management du contenu du projet
- 6 = Management des délais du projet
- 7 = Management des coûts du projet
- 8 = Management de la qualité du projet
- 9 = Management des ressources humaines du projet
- 10 = Management des communications du projet
- 11 = Management des risques du projet
- 12 = Management des approvisionnements du projet.

Chaque processus est ensuite numéroté séquentiellement à l'intérieur du domaine.

Le groupe **Démarrage** comporte les processus 4.1 et 4.2 du domaine 4 (intégration) (figure 17.4).

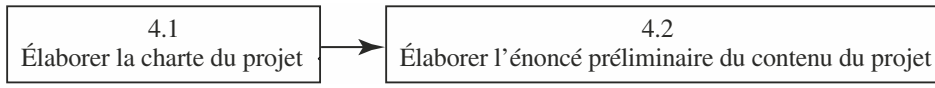


Figure 17.4 — Les processus du groupe Démarrage.

Le groupe **Planification** comporte vingt-et-un processus des neuf domaines (figure 17.5), ce qui signifie que tous les aspects d'un projet sont planifiés.

Il convient de souligner que la démarche de planification n'est pas totalement linéaire : non seulement la planification du contenu interagit avec la définition du contenu, mais la planification de certains aspects (risques ou approvisionnements par exemple) peut conduire à faire évoluer le budget initial. Par ailleurs, on peut relever une légère différence entre cette dynamique générale de planification et celle qui est généralement mise en œuvre dans les projets de système d'information où les coûts majeurs (et les plus fortement entachés d'incertitude) sont ceux de matière grise. C'est pourquoi, comme cela a été exposé dans la première partie de cet ouvrage, le plus souvent on estime d'abord la quantité de travail, qui correspondra à un coût après affectation à une ressource ou une ressource-type. Au niveau d'une activité élémentaire la quantité de travail équivaut à la durée compte tenu de la difficulté, sauf exception, à diviser un travail de production intellectuelle entre plusieurs personnes.

Le groupe **Exécution** comporte sept processus des domaines 4 (intégration), 8 (qualité), 9 (ressources humaines), 10 (communications) et 12 (approvisionnements) (figure 17.6).

Le groupe **Surveillance et Maîtrise** comporte douze processus des neuf domaines de connaissances (figure 17.7). La logique générale est la suivante : tous les aspects d'un projet sont mis sous contrôle, notamment les modifications survenues ou demandées. Comme la modification d'un aspect (contenu, risque, etc.) peut avoir des répercussions sur les autres (coûts, équipe, etc.), il s'agit de rechercher une cohérence et un équilibre de l'ensemble du projet.

Le groupe **Clôture** comporte deux processus des domaines 4 (intégration) et 12 (approvisionnements) (figure 17.8). La phase — ou le projet si l'on est dans le cas d'une phase unique ou d'une phase finale — fait l'objet d'une fermeture officielle, qui inclut le cas échéant les activités de solde du contrat.

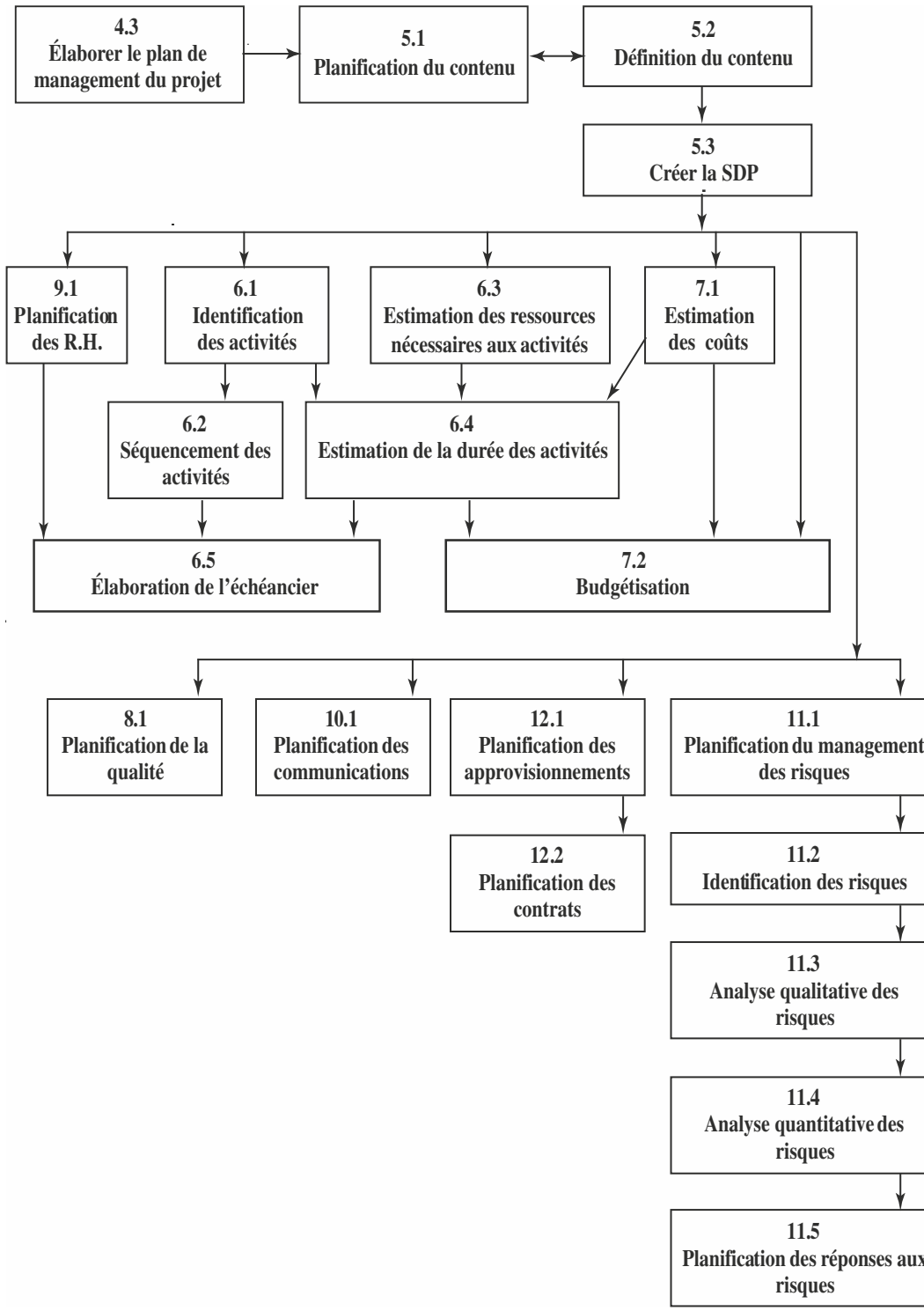


Figure 17.5 — Les processus du groupe Planification.

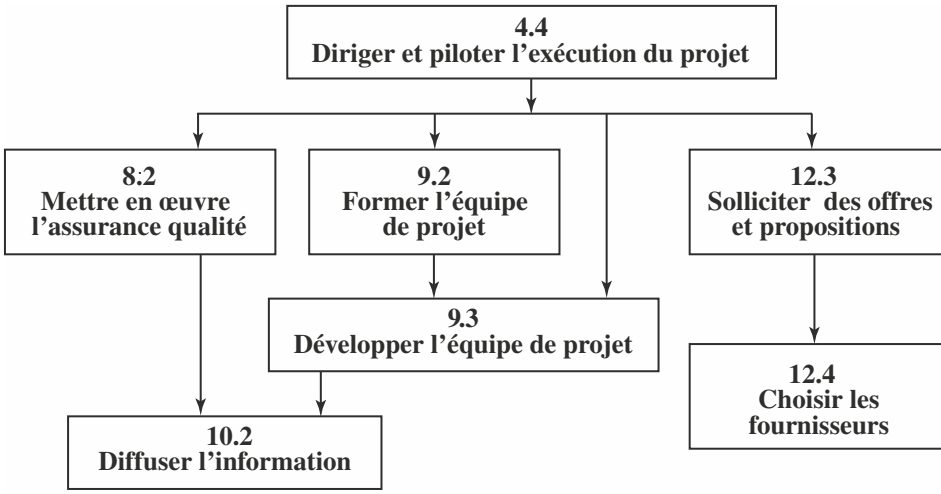


Figure 17.6 — Les processus du groupe Exécution.

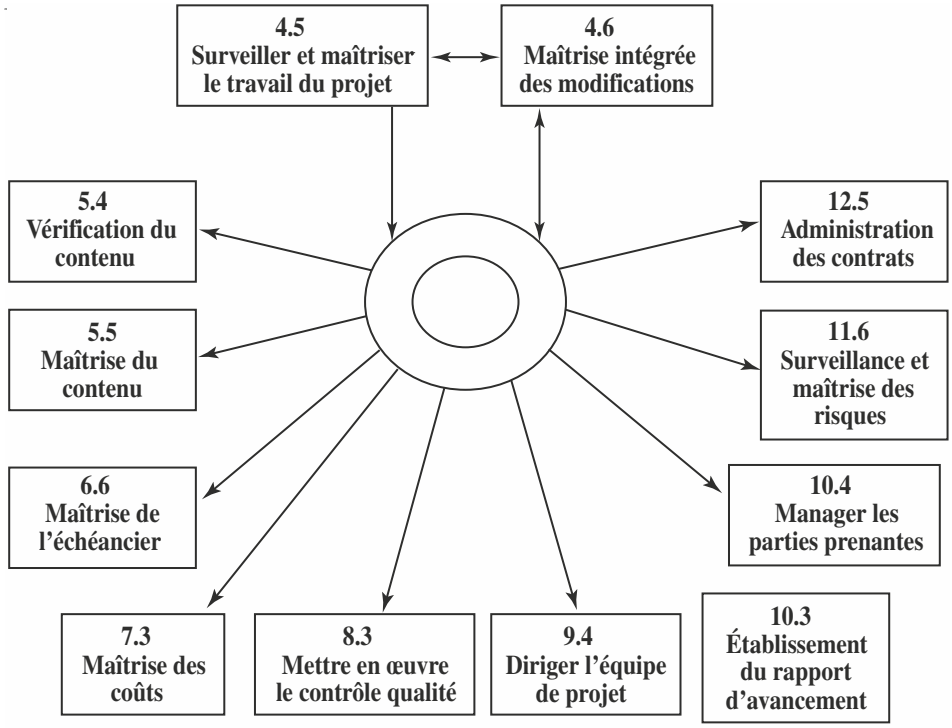


Figure 17.7 — Les processus du groupe Surveillance et Maîtrise.

Figure 17.8 — Les processus du groupe Clôture.

Nous allons maintenant reprendre les 44 processus en les présentant par domaine de connaissances et en rappelant leur appartenance à un groupe de processus. L'objectif est d'indiquer les techniques utilisables dans l'exécution de chaque processus en renvoyant à la description correspondante dans la première partie de cet ouvrage.

17.4 LE MANAGEMENT DE L'INTÉGRATION DU PROJET

17.4.1 Les processus du management de l'intégration

L'objectif commun de ces processus est de prendre en compte toutes les dimensions du projet durant sa vie.

Il s'agit d'abord de lancer officiellement le projet, ce qui relève en général du commanditaire et peut se matérialiser par une charte par laquelle le chef de projet est officiellement nommé. Ensuite, un premier énoncé du contenu du projet avec les principaux livrables (paragraphe 17.3.2) est élaboré. Le résultat de la planification des différents aspects vient alimenter un plan de management de projet, pouvant être structuré selon les domaines de connaissances, et décrit au paragraphe 6.3. Ce plan est exécuté par l'équipe de projet. Régulièrement, l'équipe de management de projet calcule les performances et les éventuelles modifications (faites ou à faire) sont prises en compte selon une vision globale du projet. À l'achèvement, le projet est clos officiellement.

La figure 17.9 récapitule ces processus, classés par groupe d'appartenance.

Démarrage	Planification	Exécution	Surveillance et maîtrise	Clôture
4.1. Élaborer la charte du projet 4.2. Élaborer l'énoncé préliminaire du contenu du projet	4.3. Élaborer le plan de management de projet	4.4. Diriger et piloter l'exécution du projet	4.5. Surveiller et maîtriser le travail du projet 4.6 Maîtrise intégrée des modifications	4.7. Clore le projet

Figure 17.9 — Les processus de management de l'intégration.

17.4.2 Les techniques utiles pour le management de l'intégration

Dans le *processus 4.1*, on peut utiliser pour arbitrer entre deux scénarios du même projet la technique du retour sur investissement (ROI) ou celle de la valeur monétaire attendue (VMA) décrites au chapitre 7 et illustrées au chapitre 12.

Un tableau de bord tel que celui exposé dans les paragraphes 12.3 à 12.10 peut être utilisé dans les *processus 4.4 et 4.5*.

Dans le *processus 4.5*, on peut également utiliser, pour évaluer la performance et calculer les prévisions, les indicateurs de la technique de la valeur acquise (EVT), décrite au chapitre 7 et illustrée au chapitre 12.

Notons enfin qu'un outil de planification et suivi du projet, tel que celui décrit au chapitre 15, peut être utilisé dans tous les processus du management de l'intégration.

17.5 LE MANAGEMENT DU CONTENU DU PROJET

17.5.1 Les processus du management du contenu du projet

L'objectif commun de ces processus est de mettre en place un dispositif pour définir et exécuter uniquement les activités nécessaires pour achever le projet de façon satisfaisante.

Pour cela, on définit d'abord comment on va s'organiser pour gérer le contenu, puis on élabore un énoncé détaillé du contenu du projet, c'est-à-dire du travail nécessaire (paragraphe 17.3.2) et on crée la structure de découpage du projet (SDP) comme décrit au paragraphe 2.2. Le niveau de décomposition visé est celui des lots de travail, tel que présentés au paragraphe 17.3.2.

Après réalisation des livrables, on officialise leur acceptation, et le cas échéant on prend en compte les modifications de contenu qui seront ensuite traitées par le processus de Maîtrise intégrée des modifications.

La figure 17.10 récapitule ces processus, classés par groupe d'appartenance.

Démarrage	Planification	Exécution	Surveillance et maîtrise	Clôture
	5.1. Planification du contenu 5.2. Définition du contenu 5.3. Créer la structure de découpage du projet		5.4. Vérification du contenu 5.5. Maîtrise du contenu	

Figure 17.10 — Les processus de management du contenu.

17.5.2 Les techniques utiles pour le management du contenu du projet

Pour bien cerner les caractéristiques du projet dans le *processus 5.2*, on peut utiliser l'analyse des parties prenantes, dont la notion a été introduite au chapitre 5 et la technique illustrée au chapitre 13.

Dans le *processus 5.3*, on peut élaborer le découpage du travail en s'appuyant sur les approches de structuration et modèles de développement présentés tout au long du chapitre 2.

Le *processus 5.5* peut faire intervenir un Système de gestion de configuration (paragraphe17.3.2) si les modifications sont fréquentes et importantes.

17.6 LE MANAGEMENT DES DÉLAIS DU PROJET

17.6.1 Les processus du management des délais du projet

L'objectif commun de ces processus est de s'organiser pour achever le projet en temps voulu.

Il s'agit d'abord de compléter la définition du contenu du projet en identifiant le plus finement possible les activités nécessaires pour produire les livrables, puis de repérer les dépendances entre ces activités, qu'elles soient dues à la nature du travail, à des choix de l'équipe ou à des contraintes extérieures. On estime d'une part les types et quantités de ressources nécessaires et d'autre part la durée des activités. On peut alors élaborer le calendrier du projet qui servira de référence. Les écarts par rapport à cet échéancier de référence seront détectés, voire anticipés, et pourront donner lieu à des recommandations de modification du calendrier pour que celles-ci soient prises en compte de façon intégrée, en considérant les autres aspects du projet.

La figure 17.11 récapitule ces processus, classés par groupe d'appartenance.

Démarrage	Planification	Exécution	Surveillance et maîtrise	Clôture
	6.1. Identification des activités 6.2. Séquencement des activités 6.3. Estimation des ressources nécessaires aux activités 6.4. Estimation de la durée des activités 6.5. Élaboration de l'échéancier		6.6. Maîtrise de l'échéancier	

Figure 17.11 — Les processus de management des délais.

17.6.2 Les techniques utiles pour le management des délais du projet

Pour identifier les activités élémentaires (*processus 6.1*), on reprend la Structure de Découpage du Projet (SDP) et on décompose chaque lot de travail. On peut utiliser modèles de décomposition par exemple un lot de type « Réalisation » comprend de façon standard les activités d'élaboration de jeu d'essai, d'étude technique, de programmation et de test unitaire. Sur les projets de taille importante, on recourt généralement à la « planification par vagues », c'est-à-dire que l'on n'identifie les activités élémentaires que pour les lots de travail devant être planifiés à court terme.

Pour séquencer les activités (*processus 6.2*), on utilise de façon générale la technique des graphes permettant de représenter les liens logiques et les contraintes d'enchaînement. Il en existe différentes formes, dont les deux principales ont été introduites au paragraphe 4.2 : le diagramme des antécédents (de loin le plus utilisé et parfois connu sous le nom de PERT) et le diagramme fléché.

Les liens entre activités peuvent être documentés selon l'origine : une dépendance obligatoire, parfois appelée dépendance logique forte, est inhérente à la nature des activités ; une dépendance optionnelle, ou lien logique préféré, est décidée par l'équipe de management de projet, souvent à partir d'expériences antérieures ; une dépendance externe est une contrainte liée à des activités extérieures au projet, par exemple une date imposée.

Sur chaque lien, on peut appliquer des avances ou des retards (paragraphe 4.3).

Pour estimer les ressources nécessaires (*processus 6.3*), on peut utiliser une méthode relevant de ce que l'on appelle l'« estimation ascendante », qui consiste obtenir une estimation en totalisant les charges estimées des composants (lot de travail, activité, tâche…). Son application requiert une définition précise du travail. On en a vu deux exemples au chapitre 3 : la méthode d'évaluation analytique (paragraphe 3.6) et la généralisation de l'approche par unité d'œuvre (paragraphe 3.10). On peut aussi s'appuyer sur le jugement d'expert, éventuellement avec la méthode Delphi (paragraphe 3.4).

Parfois, on souhaite comparer deux scénarios (par exemple en sous-traitant une partie) sous l'angle des ressources nécessaires : on peut alors appliquer la technique de la Valeur Monétaire Attendue (VMA) décrite au chapitre 7 et illustrée au chapitre 12.

L'estimation de la durée des activités (*processus 6.4*) fait appel à toutes les techniques du chapitre 3, avec les appellations génériques suivantes.

L'estimation dite « par analogie », ou « évaluation descendante », consiste à utiliser les coûts réels de projets antérieurs similaires comme base d'estimation des totaux des coûts d'un projet en cours, ce qui implique d'avoir capitalisé. Ces

méthodes sont adéquates lorsque l'on dispose de peu d'informations sur le projet ou que l'on est dans un cadre normatif. On a vu l'exemple de la méthode de répartition proportionnelle (paragraphe 3.5.1) et la méthode des ratios (paragraphe 3.5.2).

L'estimation dite « paramétrique » correspond à la famille des méthodes basées sur une unité d'œuvre. On en a vu deux exemples : le modèle Cocomo (paragraphe 3.7) et la méthode des points de fonction (paragraphe 3.8).

Si l'incertitude sur les valeurs estimées est élevée, on peut utiliser l'estimation à trois points (valeur optimiste, valeur pessimiste, valeur la plus probable) et prendre la moyenne, pondérée ou non, comme on l'a vu au paragraphe 3.9. Signalons que dans les appellations récentes, on appelle réseau PERT un diagramme des antécédents sur lequel on a utilisé des estimations de durée calculées ainsi :

[(estimation optimiste + estimation pessimiste
+ 4*(estimation la plus probable)]/6

On peut également faire une provision pour d'éventuels aléas, en pourcentage de la valeur estimée ou en nombre de jours, que l'on appelle « réserve de temps » ou « tampon ».

Pour élaborer l'échéancier (*processus* 6.5) et produire un diagramme à barres, tel que le diagramme de Gantt présenté au paragraphe 4.5, on effectue une analyse du graphe des activités, représenté le plus souvent par un diagramme des antécédents. La technique la plus connue est celle du chemin critique, avec les dates au plus tôt, dates au plus tard, marge totale et marge libre, tel que détaillé au paragraphe 4.4.

On peut alors décider d'introduire des modifications, notamment de compresser l'échéancier pour réduire le délai global. Ceci peut se faire en réduisant la durée des activités, notamment celles qui se trouvent sur le chemin critique : c'est ce que les Anglo-Saxons appellent le « crashing ». On peut également utiliser l'exécution accélérée par chevauchement, « fast tracking », qui consiste à faire exécuter en parallèle des phases ou des activités qui devraient normalement se dérouler en séquence. La compression de l'échéancier peut augmenter les coûts, que ce soit ceux d'une coordination accrue ou ceux liés à une non-qualité des résultats et à l'obligation de reprise.

On fait parfois une analyse des éventualités, c'est-à-dire que l'on teste plusieurs scénarios de délai, soit en considérant un événement particulier (par exemple, retard dans le démarrage d'une tâche à cause de la défaillance d'un sous-traitant), soit en s'appuyant sur des distributions de probabilités. La méthode de Monte-Carlo s'inscrit dans ce deuxième cas et fonctionne de la façon suivante : on suppose que l'on connaît la fonction des probabilités de durées possibles de chaque activité (par exemple une courbe de Gauss entre une valeur minimale, une valeur maximale et une valeur moyenne). On effectue de nombreuses itérations du calcul de la durée totale du projet, en tirant au hasard une valeur de la

durée de chaque activité en fonction de la loi de distribution qui lui est attachée. On dispose ainsi d'un ensemble de durées calculées du projet, dont on va prendre la moyenne (paragraphe 3.9).

Après stabilisation du diagramme des antécédents, on fait des hypothèses sur la disponibilité des ressources. Cela peut conduire à ne pas utiliser toutes les possibilités de parallélisme. Le PMBOK parle de nivellement dans deux cas : soit parce le nombre de personnes dans l'équipe doit être maintenu au-dessus d'un certain niveau (ce qui est l'emploi classique du terme), soit parce que l'on doit résoudre des problèmes d'indisponibilités partielles, par exemple une ressource rare et affectée à des tâches en parallèle ou bien une ressource partagée. On a présenté cette technique sous le terme souvent utilisé de lissage (paragraphe 4.6.4).

La méthode dite de la chaîne critique est due à E.M. Goldratt, qui préconise de se focaliser sur une chaîne critique qui est composée des activités pouvant être des goulets d'étranglement, soit qu'elles soient porteuses d'une incertitude élevée, soit qu'elles soient affectées à des ressources rares dont l'indisponibilité pourrait avoir un impact important. Au lieu de saupoudrer de la marge sur la plupart des activités, il est préférable de planifier au plus tard et de rajouter des tâches-tampons après les activités de la chaîne critique, ou en fin de phase pour absorber des retards (paragraphe 4.6.2).

Le calendrier du projet est ensuite représenté, éventuellement à l'aide d'un outil de planification, sous forme d'un diagramme de Gantt (paragraphe 4.5).

Pour suivre les délais et contrôler les écarts (*processus* 6.6), on s'appuie sur un tableau de bord, par exemple un récapitulatif produit chaque semaine, sur les indicateurs de la valeur acquise, notamment l'écart de délai, l'indice de performance requis et les prévisions à l'achèvement (paragraphe 7.3). Les logiciels de suivi de projet fournissent également des tableaux de suivi de performance et des diagrammes à barre dans lesquels figurent la planification de référence et les performances réelles (paragraphe 15.4).

17.7 LE MANAGEMENT DES COÛTS DU PROJET

17.7.1 Les processus du management des coûts du projet

L'objectif commun de ces processus est de s'organiser pour que le projet soit réalisé dans les limites d'un budget imposé ou négocié.

Il s'agit de déterminer une valeur approximative des coûts des activités et d'établir un budget, c'est-à-dire un coût global du projet par agrégation des coûts des activités. La maîtrise des coûts suppose le suivi des dépenses et la connaissance de leur origine : les écarts par rapport au budget sont alors détectés, voire

anticipés, et peuvent donner lieu à des recommandations ou propositions de modifications qui seront traitées de façon intégrée.

La figure 17.12 récapitule ces processus, classés par groupe d'appartenance.

Démarrage	Planification	Exécution	Surveillance et maîtrise	Clôture
	7.1. Estimation des coûts 7.2. Budgétisation		7.3. Maîtrise des coûts	

Figure 17.12 — Les processus de management des coûts.

17.7.2 Les techniques utiles pour le management des coûts du projet

L'estimation des coûts (*processus 7.1*) étant souvent liée à l'estimation de la quantité de travail nécessaire, on retrouve certaines techniques déjà mentionnées pour les processus 6.3. (estimation des ressources) et 6.4 (estimation des durées). En particulier, l'estimation par analogie (par ex. la répartition proportionnelle), l'estimation ascendante (par ex. l'évaluation analytique) et l'estimation paramétrique (par ex. les points de fonctions).

Comme pour les délais, on peut prévoir une enveloppe financière pour d'éventuels aléas, si le budget l'autorise.

L'établissement du budget de référence (*processus 7.2*) peut, notamment si l'on n'est pas passé par une estimation des activités ou pour conforter cette dernière, être effectuée au moyen d'une méthode paramétrique : on répartira ensuite ce coût global entre les différentes activités.

La maîtrise des coûts (*processus 7.3*) s'appuie essentiellement sur l'établissement et l'analyse des indicateurs de performance, en particulier ceux fournis par la technique de la valeur acquise (paragraphe 7.3).

17.8 LE MANAGEMENT DE LA QUALITÉ DU PROJET

17.8.1 Les processus du management de la qualité du projet

L'objectif commun de ces processus est de s'organiser pour mettre en œuvre des actions visant la qualité du processus projet, ainsi que celle du produit visé.

Pour cela, on va élaborer un plan qualité qui viendra s'insérer dans le plan de management de projet, indiquant d'une part les actions et dispositions pour donner confiance dans la qualité du processus, d'autre part les critères qualité du

produit explicités avec le commanditaire. Ensuite, on vérifie périodiquement que l'équipe de projet effectue les activités de gestion de la qualité conformément au plan qualité), et on contrôle la qualité des résultats du projet avant livraison.

La figure 17.13 récapitule ces processus, classés par groupe d'appartenance.

Démarrage	Planification	Exécution	Surveillance et maîtrise	Clôture
	8.1. Planification de la qualité	8.2. Mettre en œuvre l'assurance qualité	8.3. Mettre en œuvre le contrôle qualité	

Figure 17.13 — Les processus de management de la qualité.

17.8.2 Les techniques utiles pour le management de la qualité du projet

La planification de la qualité (*processus 8.1*) s'appuie sur les principes qualité, notamment les facteurs et indicateurs qualité (paragraphe 8.5) et le contenu d'un plan qualité (paragraphe 8.7).

La mise en œuvre de l'assurance qualité (*processus 8.2*) consiste à vérifier que les dispositions prévues sont effectivement appliquées. Elle porte donc sur les processus de management du projet. La technique principalement utilisée est l'audit qualité (paragraphe 8.8.2), c'est-à-dire une revue structurée et indépendante de la façon dont le travail est organisé et effectué, qui peut déboucher sur des recommandations de correction ou d'amélioration.

Le contrôle qualité (*processus 8.3*) porte sur les résultats du projet : il consiste à vérifier qu'ils sont bien conformes aux normes et critères qualité retenus (paragraphe 8.8.1). Il s'applique bien entendu aux livrables, mais également aux documents produits par le management de projet.

17.9 LE MANAGEMENT DES RESSOURCES HUMAINES DU PROJET

17.9.1 Les processus du management des ressources humaines du projet

L'objectif commun de ces processus est d'organiser et gérer l'équipe de projet afin d'obtenir l'engagement de chacun dans le projet.

Pour cela, il s'agit définir les rôles (responsabilités d'activités et autorité éventuelle) et d'obtenir le personnel pouvant assurer ces rôles. Puis, on cherche à établir et maintenir une bonne collaboration et un climat positif dans l'équipe, par des mesures individuelles ou concernant tout le groupe. Par ailleurs, les performances

font l'objet d'un suivi, qui permet parfois d'identifier des problèmes qui doivent être gérés.

La figure 17.14 récapitule ces processus, classés par groupe d'appartenance.

Démarrage	Planification	Exécution	Surveillance et maîtrise	Clôture
	9.1. Planification des ressources humaines	9.2. Former l'équipe de projet 9.3. Développer l'équipe de projet	9.4. Diriger l'équipe de projet	

Figure 17.14 — Les processus de management des ressources humaines.

17.9.2 Les techniques utiles pour le management des ressources humaines du projet

Pour planifier les ressources humaines (*processus 9.1*), c'est-à-dire pour identifier et décrire les rôles, les responsabilités et les relations d'autorité, on s'appuie sur des descriptions classiques reproduites à la figure 17.15 : organigramme hiérarchique de la répartition d'encadrement de l'autorité dans l'équipe de projet, matrice d'affectation des responsabilités croisant les différents acteurs et les différentes activités (et pouvant dans le cas d'une grille dite RACI, faire apparaître à l'intersection le type d'intervention sur l'activité : Responsabilité d'accomplissement, Autorité pour prendre des décisions, Consultation sur les décisions, Information transmise à l'acteur sur les décisions), et description du poste des membres de l'équipe de projet.

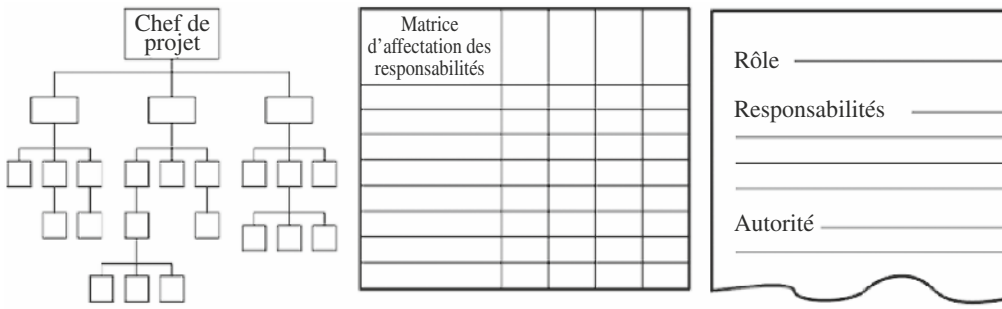


Figure 17.15 — Les outils de planification des ressources humaines (*d'après Guide du PMBOK, figure 9.4*).

Sur un projet nécessitant de nombreuses ressources, on peut utiliser un histogramme pour représenter, par type de profil, les besoins (nombre de jours-personne) sur la durée du projet, période par période.

Pour former l'équipe de projet (*processus 9.2*), on s'appuie sur les principes d'organisation du travail tels que ceux développés au paragraphe 5.1, notamment les critères de composition de l'équipe et les formes de coordination à privilégier.

Le développement de l'équipe de projet (*processus 9.3*) conduit à mettre en œuvre des actions qui s'inspirent des théories de la motivation (récompenses, formation, salle commune, etc.) présentées au paragraphe 5.3, notamment une utilisation judicieuse des styles de management.

La direction de l'équipe de projet (*processus 9.4*) comprend une évaluation des performances (par exemple, à partir des suivis individuels du paragraphe 7.2.2) et éventuellement une gestion des conflits (paragraphe 5.3.5).

17.10 LE MANAGEMENT DES COMMUNICATIONS DU PROJET

17.10.1 Les processus du management des communications du projet

L'objectif commun de ces processus est de déterminer les informations à diffuser à chaque partie prenante, notamment sur l'avancement du projet, et de s'organiser pour qu'elles soient mises à disposition en temps voulu.

Il s'agit pour cela d'établir un plan de gestion des communications, c'est-à-dire de déterminer quelles informations, pour qui, quand et sous quelle forme. Ensuite, les éléments sont rassemblés et présentés dans les différents formats prévus. Au-delà de cette remise systématique, le chef de projet s'attache généralement à établir et maintenir des relations de travail satisfaisantes avec les acteurs du projet (rencontres, réunions…) et à résoudre avec eux les problèmes rencontrés.

La figure 17.16 récapitule ces processus, classés par groupe d'appartenance.

Démarrage	Planification	Exécution	Surveillance et maîtrise	Clôture
	10.1. Planification des communications	10.2. Diffusion de l'information	10.3. Établissement des rapports d'avancement 10.4. Manager les parties prenantes	

Figure 17.16 — Les processus de management des communications.

17.10.2 Les techniques utiles pour le management des communications du projet

Pour élaborer un plan de management des communications (*processus 10.1*), on analyse les besoins en communication à partir des éléments précédemment

élaborés (analyse des parties prenantes, organigramme du projet…). Le choix d'une technologie support dépend de notamment de l'urgence de l'information, de la durée du projet et de son environnement.

Dans la diffusion de l'information (*processus 10.2*), on utilise différents canaux (réunions, documents papiers ou électroniques, intranet…) pour collecter et diffuser l'information planifiée ou faisant l'objet de demandes ponctuelles.

Pour établir les rapports d'avancement (*processus 10.3*), on peut s'appuyer sur des indicateurs et une structure de tableau de bord prédéfinis (paragraphe 7.2) ou normalisés (paragraphe 7.3).

Dans le management des parties prenantes (*processus 10.4*), on peut avoir à résoudre des problèmes (par exemple en utilisant les stratégies de résolution de conflit), dont on gardera éventuellement trace dans un journal de bord (exemple de la figure 12.23 à la fin du paragraphe 12.8)

17.11 LE MANAGEMENT DES RISQUES DU PROJET

17.11.1 Les processus du management des risques du projet

L'objectif commun de ces processus est de réduire la probabilité et les conséquences d'événements défavorables au projet et d'éventuellement augmenter celles qui lui sont favorables. Il s'agit, pour ce faire, d'identifier les risques du projet, de les analyser et, si besoin est, d'élaborer des parades, ce qui est consigné dans un plan de management des risques. Ensuite, les mesures prévues sont, le cas échéant, appliquées et un suivi régulier de l'état des risques est mis en place.

La figure 17.17 récapitule ces processus, classés par groupe d'appartenance.

Démarrage	Planification	Exécution	Surveillance et maîtrise	Clôture
	11.1. Planification du management des risques 11.2. Identification des risques 11.3. Analyse qualitative des risques 11.4. Analyse quantitative des risques 11.5. Planification des réponses aux risques		11.6. Surveillance et maîtrise des risques	

Figure 16.17 — Les processus de management des risques.

17.11.2 Les techniques utiles pour le management des risques du projet

L'élaboration du plan de management des risques (*processus 11.1*), qui vient enrichir le plan de management de projet, permet, à travers des réunions de l'équipe de projet élargie éventuellement à d'autres parties prenantes, de se mettre d'accord sur les méthodes d'analyse des risques à utiliser ainsi que sur les niveaux qui seront considérés comme critiques. En effet, il y a toujours un arbitrage à faire entre prise de risque et coût de prévention des risques, et la sensibilité aux risques diffère d'une organisation à l'autre. Il s'agit de définir et attribuer des rôles et responsabilités de gestion des risques, ainsi que la fréquence d'application des analyses et opérations de surveillance.

Les trois processus suivants (identification et analyse) correspondent à l'approche généralisée que nous avons décrite au paragraphe 6.2.2 : élaborer une liste très large de risques possibles et restreindre progressivement sur les quelques risques que l'on voudra contrôler. L'identification des risques (*processus 11.2*) s'appuie sur différentes techniques présentées au paragraphe 6.2.2.

On peut aussi s'appuyer sur différentes typologies, comme celle proposée par Eurométhode ou par l'AFITEP pour les projets système d'information (paragraphe 6.2.3), ou plus générale comme celle du PMBOK (figure 17.18). On peut aussi cibler sur les sources majeures, comme nous l'avons montré avec la grille d'analyse des risques (paragraphe 6.2.4).

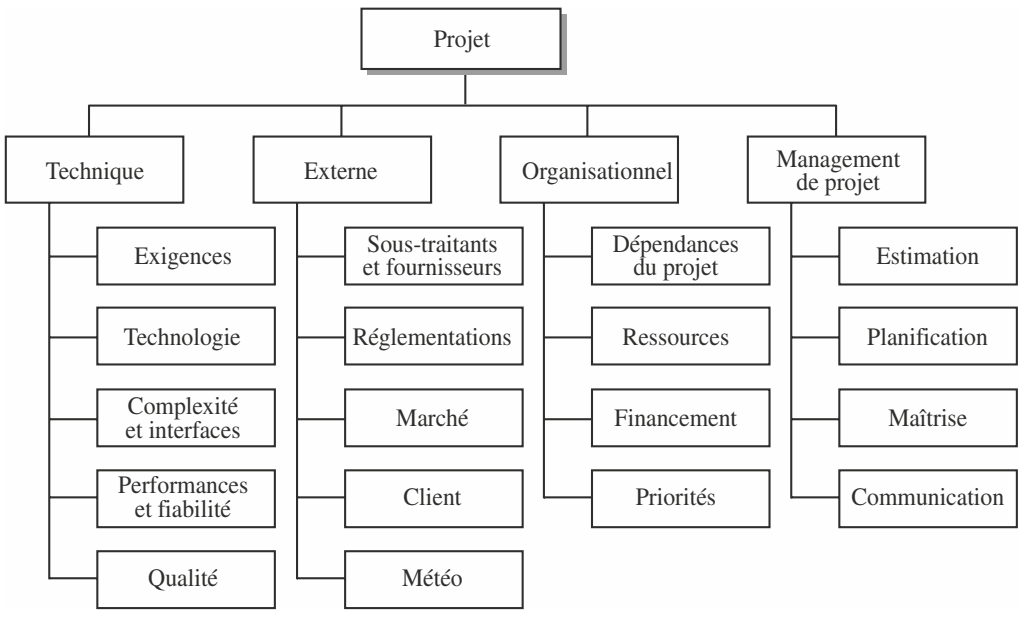


Figure 17.18 — Typologie structurée des risques (Source : PMBOK, 2004).

L'analyse qualitative des risques (*processus 11.3*) vise à évaluer l'importance de chacun des risques précédemment identifiés pour ne retenir que les plus importants. On peut ainsi analyser l'impact d'un risque sur les objectifs (coût, délai, contenu, qualité) ou classer les risques par probabilité d'occurrence et impact (paragraphe 6.2.3). Dans le cas d'un projet système d'information, on peut utiliser les critères et métriques d'une grille d'analyse des risques, comme présenté au paragraphe 6.2.4.

L'analyse quantitative des risques (*processus 11.4*), qui consiste à effectuer l'analyse chiffrée des effets des risques identifiés, n'est pas utilisée pour les projets systèmes d'information où l'incertitude est en général trop élevée. Elle recourt à des techniques appliquées à l'analyse de processus répétitifs pour lesquels on peut établir des distributions de probabilités. Ainsi, une analyse de sensibilité qui aide à classer les facteurs ayant un impact potentiel sur un effet redouté. Par exemple, le diagramme en tornade présenté à la figure 17.19 montre l'influence (positive ou négative) de différents facteurs sur le respect des délais.

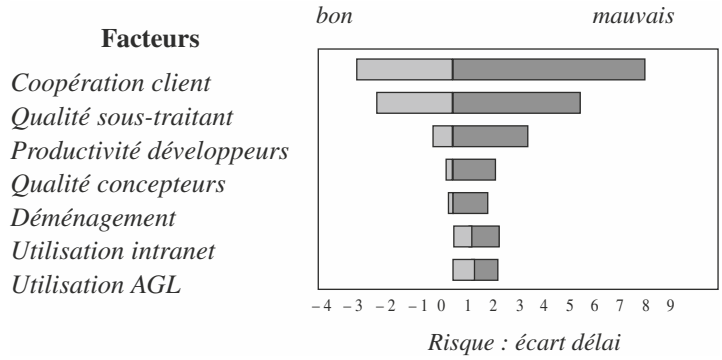


Figure 17.19 — Diagramme en tornade pour classer les facteurs de risque.

On pourrait également, pour quantifier la comparaison entre deux scénarios présentant tous deux des risques, utiliser la technique de la Valeur monétaire attendue décrite au paragraphe 7.4.

La planification des réponses aux risques retenus au terme de l'analyse (*processus 11.5*) vise à élaborer des options et des actions pour réduire les menaces et améliorer les opportunités favorables au projet. Les différents types de réponse ont été présentés au paragraphe 6.4.1. Pour les projets de système d'information, on parle de stratégie de projet prenant en compte de façon intégrée l'ensemble des risques majeurs identifiés, comme décrit au paragraphe 6.4.2.

La surveillance et maîtrise des risques (*processus 11.6*) consistent à réanalyser périodiquement la situation des risques et les actions entreprises. On s'appuie pour cela sur les réunions de revue de projet et l'analyse des écarts et tendances

(par exemple, ceux donnés par les indicateurs de la valeur acquise dans le suivi des coûts et des délais). On peut également faire un audit, c'est-à-dire une revue ponctuelle et structurée, par exemple selon le schéma du Standish Group présenté au paragraphe 6.4.3.

17.12 LE MANAGEMENT DES APPROVISIONNEMENTS DU PROJET

17.12.1 Les processus du management des approvisionnements du projet

L'objectif commun de ces processus est d'acquérir de façon efficace et efficiente des biens ou services fournis par une entreprise extérieure.

Pour cela, on commence par déterminer ce qu'il est nécessaire ou préférable d'acheter ou sous-traiter et à quel moment dans le projet : cela donne lieu à un plan de management des approvisionnements, qui vient s'inscrire dans le plan de management de projet. La nature des contrats est ensuite étudiée (par exemple régie ou forfait). Aux moments planifiés, il s'agit de lancer l'appel d'offres correspondant et de sélectionner le fournisseur. Après contractualisation, l'exécution du contrat fait l'objet d'un suivi de réalisation et d'une gestion des paiements. En fin de phase (ou fin de projet), le contrat est clos officiellement après règlement des points en suspens.

La figure 17.20 récapitule ces processus, classés par groupe d'appartenance.

Démarrage	Planification	Exécution	Surveillance et maîtrise	Clôture
	12.1. Planifier les approvisionnements 12.2. Planifier les contrats	12.3. Solliciter des offres/propositions 12.4. Choisir les fournisseurs	12.5. Administration du contrat	12.6. Clôture du contrat

Figure 17.20 — Les processus de management des approvisionnements.

17.12.2 Les techniques utiles pour le management des approvisionnements du projet

Pour planifier les approvisionnements du projet (*processus 12.1*), on peut recourir à la technique de l'arbre de décision et de la valeur monétaire attendue si l'on hésite devant l'alternative « faire ou acheter ? ». Le type de contrat est également discuté et sélectionné. Ces deux aspects sont présentés au paragraphe 7.4.1.

Lors de la planification des contrats (*processus 12.2*), on recourt souvent à des contrats-types et on prépare les critères d'évaluation des réponses aux appels d'offres en s'appuyant sur des listes de critères que l'on adapte et complète.

Pour obtenir des offres ou des propositions (*processus 12.3*), on fait connaître son besoin de façon publique ou auprès de fournisseurs présélectionnés (par exemple, uniquement des fournisseurs certifiés ISO9000), et on organise parfois une réunion pour les fournisseurs intéressés afin de clarifier le besoin et donner à tous le même niveau d'information.

Le choix d'un fournisseur (*processus 12.4*) est souvent un processus à deux temps : on élabore d'abord une liste restreinte, on effectue ensuite une évaluation détaillée des deux ou trois restants. On fait parfois appel des experts indépendants, notamment pour juger de la pertinence du coût annoncé, et on s'appuie en général sur une grille multicritère, comme celle présentée au paragraphe 7.5.1.

L'administration du contrat (*processus 12.5*) consiste à gérer les relations avec le fournisseur, et tout particulièrement de suivre l'avancement afin de pouvoir intégrer les informations dans le tableau général d'avancement du projet, y compris les prévisions d'achèvement. Sur des sous-traitances importantes, des revues du travail accompli, voire des audits, peuvent être effectués.

Ainsi s'achève cette mise en correspondance entre les processus de management de projet, tels que structurés par le Guide PMBOK, et les principes et techniques applicables aux projets de système d'information. La préparation de la certification passe ensuite par une prise de connaissance approfondie du Guide de référence, et surtout son application dans des questions de mise en situation. Quelques exemples de questions sont donnés en annexe D.

18

Méthodes agiles et référentiel PMBOK

Les méthodes agiles se sont démarquées des approches plus classiques de gestion d'un projet informatique, en particulier en ce qui concerne le cycle de développement (cf. § 2.8) et la place des utilisateurs (cf. § 5.5.2). La radicalité des auteurs du courant agile explique peut-être l'utilisation limitée de telles approches. Nous avons défendu, tout au long de cet ouvrage, une idée centrale dans le management des projets, qui est celle de l'adéquation. Loin de défendre une ligne de conduite unique, nous avons cherché à donner au lecteur les moyens d'une approche adaptée à chaque situation de projet. C'est pourquoi nous proposons, dans ce chapitre final, de replacer de la conduite d'un projet en mode agile dans la perspective du PMBOK.

18.1 MÉTHODES AGILES ET LOGIQUE DU PMBOK

Examinons d'abord la compatibilité entre une approche agile et la logique fondamentale du PMBOK.

La distinction entre produit et projet, avec les responsabilités spécifiques du client et du fournisseur, est le premier principe. La détermination du contenu du produit par le client est d'autant plus respectée que les spécifications évoluent à sa seule demande. Lorsque des arbitrages doivent être faits, lorsqu'une priorité doit être établie, seuls les utilisateurs peuvent en décider. Pour ce qui est du contenu du projet, la planification des itérations de livraison est souvent menée

en coopération entre le client, le chef de projet et son équipe. Cette participation n'enlève pas la responsabilité du maître d'œuvre.

Le principe du cycle de vie, c'est-à-dire d'un découpage en phases, est tout à fait respecté, et l'identification progressive des phases est admise par le PMBOK qui parle de « planification par vagues ».

L'idée que les groupes de processus s'appliquent à chacune des phases est tout à fait respectée si l'on considère les itérations de livraison : Démarrage et Clôture sont bien exécutés une seule fois, alors l'on a plusieurs cycles de Planification - Exécution - Surveillance et Maîtrise. Selon la méthode, chacune des itérations de développement peut également respecter l'exécution de processus des cinq groupes, c'est-à-dire inclure des activités de démarrage et de clôture.

Examinons maintenant comment les méthodes agiles par rapport aux différents domaines de connaissances.

18.2 MÉTHODES AGILES ET MANAGEMENT DU CONTENU DU PROJET

Si l'objectif de ce domaine de connaissance est de « s'assurer que le projet contient tout le travail requis, et uniquement celui-ci, pour assurer la bonne fin du projet », on peut considérer que les méthodes agiles s'attachent à l'atteindre, en recherchant les approches les plus efficientes.

Il peut toutefois y avoir des écarts importants avec une approche classique dans la mesure où les processus de planification du contenu sont fortement allégés, alors que ceux de surveillance et maîtrise, bien que peu formalisés, sont renforcés. En effet, on ne cherche pas à expliciter le plus tôt possible un contenu stable du produit, mais on accepte une définition progressive, avec de nombreux aménagements et une vérification continue par les utilisateurs participant aux itérations.

18.3 MÉTHODES AGILES ET MANAGEMENT DES DÉLAIS DU PROJET

Le souci du respect des engagements calendaires du projet étant une préoccupation centrale des méthodes agiles, on peut retenir que l'objectif d'« assurer la réalisation du projet en temps voulu » est bien recherché.

Cependant, les processus de planification diffèrent fortement d'une approche PMBOK, notamment en ce qui concerne les itérations de développement. D'abord, les activités élémentaires, si elles sont bien identifiées par l'équipe, ne font pas l'objet d'une planification formalisée. Chaque développeur ou binôme

de développeurs est chargé d'un ensemble de tâches et bénéficie d'une latitude pour les mener à bien. La séquence « estimation des ressources — estimation de la durée — élaboration de l'échéancier » n'est pas respectée pour deux raisons principales. D'une part, les objectifs délais dominent et conduisent à évaluer en conséquence la taille du contenu du projet. D'autre part, la capacité de production de l'équipe, sa « vélocité », détermine les livraisons possibles à chaque itération.

La technique de la timebox, ainsi que les principes de motivation et d'animation d'équipe sont d'un apport majeur dans la maîtrise de l'échéancier. Mais le suivi des délais s'effectue en grande partie par une communication informelle et une coordination personnelle (*standup meetings* de XP ou mêlées de SCRUM), ou avec des outils plus orientés vers la dynamisation de l'équipe (*burndown chart* de SCRUM) que vers le reporting.

18.4 MÉTHODES AGILES ET MANAGEMENT DES COÛTS DU PROJET

L'objectif de respect d'un budget est globalement partagé par les méthodes agiles, dans la mesure où les coûts sont principalement liés au travail consommé.

Toutefois, la planification est moins formalisée et la justification économique du travail en binôme est parfois difficile à apporter.

De même, le suivi des coûts n'entre pas dans les préoccupations majeures du chef de projet, qui se situe plutôt dans une logique de « design to cost », c'est-à-dire une adaptation du périmètre fonctionnel en cas de sur-consommation. L'utilisation d'indicateurs de performance n'est pas mise en avant, ce qui se comprend pour des projets de petite taille. Soulignons, par ailleurs, que nombre de projets système d'information ne font pas, dans la pratique, l'objet d'un suivi des coûts.

18.5 MÉTHODES AGILES ET MANAGEMENT DE LA QUALITÉ DU PROJET

L'objectif majeur du management de la qualité est que « le projet réponde aux besoins pour lesquels il a été entrepris ». De ce point de vue, les méthodes agiles sont tout à fait conformes au PMBOK.

Si l'on regarde le détail du domaine de connaissance, on peut considérer que les trois processus sont inclus, sous une forme ou une autre, dans les méthodes agiles.

La qualité est planifiée, en particulier à travers des activités et des rôles spécifiques. La qualité attendue du produit est déterminée progressivement, mais finement, par la collaboration avec les commanditaires/utilisateurs.

La mise en œuvre de l'assurance qualité est intégrée à l'organisation du travail, qui comprend une surveillance de l'exécution des actions visant la qualité. Celles-ci comprennent non seulement les tests des développeurs, mais aussi le respect de principe de simplicité et la recherche d'excellence technique. Le partage des connaissances dans un binôme ou au sein de l'équipe vise aussi à donner confiance dans la qualité du futur produit.

La surveillance des livrables, par l'équipe comme par l'utilisateur conseiller, selon le terme de DSDM, ainsi que le principe d'effectuer une intégration systématique de tout composant élémentaire, dès son achèvement, assurent a priori une mise en œuvre effective du contrôle qualité.

18.6 MÉTHODES AGILES ET MANAGEMENT DES RESSOURCES HUMAINES DU PROJET

La gestion des ressources humaines est une préoccupation centrale des méthodes agiles, et l'on peut considérer que tous les processus y figurent à un titre ou l'autre.

En effet, les ressources humaines sont planifiées, avec notamment les rôles nécessaires.

L'objectif du processus de développement de l'équipe, qui est d'« améliorer les compétences et la coopération des membres de l'équipe afin d'améliorer les performances du projet », est manifestement poursuivi par les méthodes agiles en ce qui concerne les développeurs, à travers les différentes dispositions dont nous avons parlé au paragraphe 5.5.

De même, le processus de direction de l'équipe de projet, qui consiste à « suivre la performance des membres de l'équipe, effectuer des retours d'information, résoudre les problèmes et coordonner les modifications en vue d'améliorer la performance du projet », se retrouve à des degrés divers dans les méthodes agiles. Bien sûr, la délégation et la participation sont des styles privilégiés, et le suivi de la performance est en général effectué de façon non formalisée. Cependant, on peut considérer que l'encadrement de l'équipe y est souvent plus rigoureux que dans des approches classiques. C'est d'ailleurs la raison invoquée par le concepteur de la méthode Crystal pour proposer une méthode, certes moins efficace que XP, mais plus facile à accepter par l'équipe en termes de discipline imposée.

18.7 MÉTHODES AGILES ET MANAGEMENT DES COMMUNICATIONS DU PROJET

Les préconisations des méthodes agiles semblent éloignées de celle du PMBOK telle qu'elle apparaît à travers les quatre processus liés aux communications. Cependant, si l'on retient que « les processus de management des communications du projet apportent les liens indispensables entre les gens et les informations nécessaires à une communication réussie », on doit reconnaître que cet objectif est le plus souvent atteint dans les projets agiles, et qu'il est revendiqué par les méthodes. Mais les approches préconisées diffèrent, là-aussi, par le degré de formalisation.

Selon le Manifeste Agile, la vraie mesure de l'avancement est la fourniture d'un logiciel qui fonctionne correctement. De ce fait, les tableaux d'avancement ne sont pas une priorité, voire même une perte de temps. La communication orale et en face à face est privilégiée.

18.8 MÉTHODES AGILES ET MANAGEMENT DES RISQUES DU PROJET

Nous avons développé au § 6.5, la question des risques dans les méthodes agiles et montré à la fois que le profil est a priori plus favorable et que le potentiel de réussite est plus élevé. Par conséquent, l'objectif du management des risques est pris en compte par ces méthodes.

La réduction des risques est liée aux dispositions, ce qui explique en grande partie pourquoi les processus d'analyse des risques (identification, analyse qualitative et quantitative) ne font pas partie des recommandations agiles. La surveillance et maîtrise des risques s'effectue par un suivi rapproché d'une équipe, qui est de taille moyenne, et la proximité du client/commanditaire évite les risques dus à une mauvaise communication.

18.9 MÉTHODES AGILES ET MANAGEMENT DES APPROVISIONNEMENTS DU PROJET

Les méthodes agiles évoquent peu les questions de sous-traitance du projet. Il est toujours implicite que les utilisateurs doivent être géographiquement proches, et que les estimations et la planification sont revues à chaque itération.

Certaines expériences font toutefois état de l'utilisation d'une approche itérative sous-traitée à distance. Dans de tels cas, l'intégration est placée sous

le contrôle du client, et les spécifications font l'objet d'une formalisation plus poussée.

18.10 MÉTHODES AGILES ET MANAGEMENT DE L'INTÉGRATION DU PROJET

Le management de l'intégration du projet est abordé de façon différente par les méthodes agiles sur deux points principaux : le plan de management de projet, ainsi que la surveillance et maîtrise.

Comme on l'a déjà évoqué, notamment à propos du contenu du projet, la planification du projet est non seulement moins formalisée, mais elle est aussi progressive. Par conséquent, ce qui tient lieu de plan de management de projet est susceptible d'évolutions au gré du déroulement et pas forcément à des moments planifiés, tels que les fins de phase.

La surveillance et la maîtrise du projet reposent davantage sur les personnes que sur des procédures et des documents. Les modifications sont maîtrisées à travers la négociation avec les clients/utilisateurs et le respect commun d'un objectif délai.

Pour les petits projets, il est généralement admis que les personnes peuvent suppléer à un déficit de formalisation. Mais depuis quelques années, praticiens et méthodologues ont cherché à combiner méthodes agiles et approches de management plus structurées[1]. L'objectif est en particulier d'introduire une certaine agilité dans les grands projets pour faire face de façon plus réactive aux aléas, et bénéficier d'une maîtrise accrue des délais et de la qualité.

Plusieurs questions ont été soulevées.

Quel est le niveau de planification requis ? Jusqu'où faut-il aller dans la définition du système global, c'est-à-dire son architecture fonctionnelle ?

Peut-on faire fonctionner une équipe de plus de dix développeurs ? Comment coordonner plusieurs équipes travaillant en parallèle ?

Peut-on utiliser des méthodes agiles pour des projets dont tous les acteurs ne sont pas situés sur le même lieu ?

Les approches agiles sont-elles aussi adaptées lorsque la composante paramétrage ou intégration de systèmes est plus importante que le développement ?

Des réponses définitives n'ont pas encore été apportées, mais plusieurs pistes se dessinent.

1. Voir par exemple D. J. Reifer, F. Maurer et H. Erdogmus, « Scaling agile methodes », IEEE Software, 0740-7459/03, 2003.

Une observation fine des pratiques laisse penser que les approches traditionnelles, sans revendiquer un label agile, intègrent une certaine souplesse et que la dichotomie agile/non agile ne serait pas si marquée[1].

Par exemple, la coopération et la communication en face à face occupent souvent une place centrale, en particulier dans les petites équipes. Lorsque des changements sont jugés importants, les parties prenantes n'hésitent pas à intégrer des changements dans les spécifications.

On peut donc considérer que la culture de l'agilité se répand, et que l'utilisation ponctuelle de certaines techniques dépasse le périmètre des projets agiles.

Certaines recommandations ont été émises concernant les grands projets[2]. On peut notamment relever deux principes, déjà évoqués dans le chapitre 6 sur la gestion des risques.

D'une part, avant de s'engager dans une démarche d'itérations accélérées et participatives, il importe d'avoir solidement construit les fondations du futur système. Une définition des processus métiers concernés par le projet et une représentation conceptuelle des informations peuvent être de bons outils pour construire une vision claire du contenu du produit.

D'autre part, une analyse des caractéristiques du projet, y compris son environnement technique et les parties prenantes, semble être hautement nécessaire pour décider du degré d'agilité à introduire dans le projet.

1. Voir par exemple C. Hansson, Y. Dittrich, B. Gustafsson et S. Zarnak, « How agile are industrial software development practices ? », *The Journal of Systems and Software*, 79, 1295-1311, 2006.
2. Voir par exemple B. Boehm et R. Turner, « Management challenges to implement agile processes in traditional development organizations », IEEE Software, 0740-7459, 2005.

Conclusion

Le champ du management d'un projet système d'information est large. La mise en œuvre des principes et techniques appelle quelques remarques de clôture.

La *prise en main* d'un projet commence par un examen de ses caractéristiques et un diagnostic des risques. Le découpage structurel nécessite une connaissance du domaine. Le découpage temporel est choisi parmi les différents modèles existants en fonction des risques. Cette double démarche débouche sur la mise en place d'une stratégie adaptée.

Les *techniques d'estimation des charges* ont des degrés de précision divers. Elles s'appuient en grande partie sur une base de références, dont il faut encourager le développement dans toutes les entreprises : l'estimation des charges est indissociable de la capitalisation du savoir-faire. La plupart des méthodes conduisent à expliciter l'idée que l'on se fait du projet, donc à justifier l'estimation. Elles permettent les comparaisons a posteriori favorisant l'apprentissage.

Les *techniques de planification* sont beaucoup plus précises que les techniques d'estimation. Cependant, il ne faut pas oublier que les calculs se font sur un découpage proposé et sur des durées estimées. Les contraintes d'ordonnancement sont elles aussi à déterminer. L'important dans un travail de planification est de pouvoir éventuellement ajuster le découpage en tâches ou modifier des contraintes d'ordonnancement. Par ailleurs, il importe de bien séparer une planification faisant apparaître les contraintes liées aux tâches et un planning établi en prenant en compte la disponibilité des ressources.

L'*organisation du projet* est un point délicat, qui influence fortement sa dynamique. Il faut être particulièrement attentif à trois points. La constitution d'une équipe doit faire partager la responsabilité d'atteinte des objectifs. La participation des utilisateurs doit être appropriée aux caractéristiques du projet : c'est pourquoi elle se décide souvent à l'établissement du plan de management de projet. Sa mise

en œuvre ne doit pas être sous-estimée. Les moyens de coordination doivent être adaptés à la taille de l'équipe et au volume d'informations partagées. Les dispositifs formalisés de coordination sont encore trop souvent négligés.

Le *pilotage du projet* se fait au quotidien. Le chef du projet doit être en permanence à proximité (réelle ou virtuelle) de l'équipe de production. Il doit être informé de tout événement pouvant avoir des conséquences sur le déroulement du projet. Sa rapidité de réaction face à des chiffres inquiétants du tableau de bord conditionne la précision de ces chiffres : en effet les membres de l'équipe de projet y porteront une attention d'autant plus grande qu'ils en perçoivent la nécessité. En ce qui concerne les actions de pilotage, nous avons choisi, dans l'exemple présenté au chapitre 12, une affectation du chef de projet lui-même à des tâches de production pour faire face à une défaillance. Cette réponse ne présente aucun caractère obligatoire et ne se rencontre guère que sur les petits projets. En revanche, le chef de projet doit pouvoir suivre de près toute la production. Par ailleurs, dans la prise de décision concernant l'équipe comme dans la gestion d'aléas pouvant conduire à des oppositions avec des tiers, il s'agit pour le chef de projet de choisir le mode de management adéquat ou la stratégie de résolution de conflit appropriée.

La *qualité* est une préoccupation transversale au projet. Elle comporte une dimension organisationnelle, notamment dans la clarification des procédures et l'explicitation des rôles. Elle a également un caractère relationnel, puisqu'en dernier ressort seul le client jugera de la qualité du produit.

La *capitalisation du savoir-faire* est en développement dans la plupart des entreprises, dans la perspective élargie du knowledge management. Elle nécessite un dispositif de mémorisation et permet ensuite l'apprentissage collectif. Nous avons vu qu'elle repose sur l'estimation des charges.

Le management d'un projet système d'information met en œuvre différentes qualités dont le développement est accéléré — si ce n'est conditionné, nombre de cas malheureux nous le prouvent — par l'usage de principes et de techniques robustes.

Annexe A

Les outils de la gestion de projet

La liste ci-dessous est donnée à titre indicatif, pour illustrer les types de logiciels pouvant apporter une aide à la gestion de projet, en liaison avec les techniques présentées dans cet ouvrage. Elle n'a aucune prétention à l'exhaustivité.

A.1 OUTILS D'ESTIMATION DES CHARGES

Estimacs, distribué par Computer Associates, s'appuie sur une vingtaine de questions touchant à l'organisation et aux caractéristiques du projet. Il fonctionne ensuite de façon fermée.

Winelys, distribué par Oresys Lysys, est également une boîte noire basée sur une description externe du projet, proche de celle des points fonctionnels, ainsi que sur une répartition proportionnelle du cycle qui peut être personnalisée.

CA-FPXpert, également de Computer Associates, et *Function Point Workbench*, de Software Productivity Research Inc., sont basés sur la méthode des points de fonction.

ESTIMATE Professional, de Software Productivity Center, s'appuie sur le modèle COCOMO II.

A.2 OUTILS DE PLANIFICATION ET DE SUIVI

Il existe un grand nombre de logiciels permettant de planifier et suivre les projets. Citons notamment *Project* de Microsoft — dont nous avons donné un exemple au chapitre 15 — et PMW (*Projet Manager Workbench*) de Applied Business Technology Corp.

La plupart offrent les fonctions suivantes :

- saisie du découpage d'un projet à plusieurs niveaux, avec indication des durées et des liens entre les tâches ;
- élaboration automatique du graphe des antécédents, avec calcul des paramètres clés et repérage du chemin critique ;
- saisie de l'affectation des ressources aux tâches ;
- personnalisation du calendrier de référence (jours fériés) ;
- élaboration automatique du diagramme de Gantt ;
- saisie des coûts par catégorie : coûts fixes ou coûts variables liés à une durée ou une quantité consommée, coûts standard et coûts réels ;
- calcul automatique des coûts à différents niveaux : projet, tâche, ressource, etc. ;
- saisie périodique des temps passés et des restes à faire ;
- comparaison automatique entre le planifié et le réalisé ;
- analyse du projet (avancement, tâches en retard, prévision de coût…) ;
- modification éventuelle de la planification.

Une offre commence à apparaître en Open Source. Citons notamment *Open Workbench*, de Niku, ou *Imendio Planner*, développé par Codefactory et promu par le gouvernement québécois, ou encore *Gantt project*.

Annexe B

Les facteurs correcteurs de la méthode des points fonctionnels

La méthode des points fonctionnels fournit des directives pour déterminer l'influence de certaines caractéristiques du futur système informatique. Chaque tableau présente le degré d'influence, compris entre 0 et 5, associé à une caractéristique particulière.

B.1 DEGRÉ D'INFLUENCE DE LA COMMUNICATION DES DONNÉES

0	L'application ne comporte pas de télécommunications : elle tourne sur un PC autonome ou est principalement composée de traitements par lots.
1	L'application permet la saisie ou l'impression à distance.
2	L'application permet la saisie et l'impression à distance.
3	L'application contient la collecte interactive des données en frontal à un traitement par lots.
4	L'application utilise un seul type de protocole de communication.
5	L'application utilise plusieurs types de protocole de communication.

B.2 DEGRÉ D'INFLUENCE DE LA DISTRIBUTION DES DONNÉES OU DES TRAITEMENTS

0	Pas de répartition.
1	L'application prépare les données qui seront transférées pour traitement local par l'utilisateur final.
2	L'application prépare les données qui seront transférées pour traitement par une autre composante du système.
3	Les données et les traitements sont répartis sur plusieurs composantes : la distribution des traitements et le transfert des données sont opérationnels dans un seul sens.
4	Les données et les traitements sont répartis sur plusieurs composantes : la distribution des traitements et le transfert des données sont opérationnels dans les deux sens.
5	Les fonctions de traitement sont exécutées de façon dynamique sur la composante la plus appropriée du système.

B.3 DEGRÉ D'INFLUENCE DE LA PERFORMANCE

0	Aucune exigence particulière.
1	Des exigences ont été spécifiées mais aucune action n'est nécessaire.
2	Le temps de réponse ou le débit est crucial aux heures de pointe. L'échéance du traitement est le jour ouvrable suivant.
3	Le temps de réponse ou le débit est crucial pendant toutes les heures ouvrables. L'échéance du traitement des systèmes en interface n'est pas contraignante.
4	Les exigences demandées exigent une analyse de la performance.
5	Les exigences demandées exigent des outils d'analyse de la performance.

B.4 DEGRÉ D'INFLUENCE DE L'INTENSITÉ D'UTILISATION DE LA CONFIGURATION MATÉRIELLE

0	Les installations ne sont pas très utilisées.
1	Des limites existent, mais aucune action n'est nécessaire.
2	Quelques considérations de sécurité ou de synchronisation doivent être prises en considération.
3	Des contraintes particulières pour certains processeurs doivent être prises en compte pour une partie de l'application.
4	Les limitations de fonctionnement imposent des contraintes particulières à l'utilisation de l'unité centrale ou des processeurs dédiés.
5	Outre les limitations des matériels, la distribution des composants impose des contraintes particulières à l'application.

B.5 DEGRÉ D'INFLUENCE DU TAUX DE TRANSACTION

0	Aucune période de pointe.
1	Une période de pointe est prévue : mensuelle, trimestrielle ou annuelle.
2	Une période de pointe est prévue : hebdomadaire.
3	Une période de pointe est prévue : quotidienne.
4	Les taux de transaction ou les exigences du contrat de service nécessitent une analyse de la performance.
5	Les taux de transaction ou les exigences du contrat de service nécessitent des outils d'analyse de la performance.

B.6 DEGRÉ D'INFLUENCE DE LA SAISIE INTERACTIVE

0	Toutes les transactions sont traitées par lots.
1	De 1 à 7 % des transactions sont des saisies de données interactives.
2	De 8 à 15 % des transactions sont des saisies de données interactives.
3	De 16 à 23 % des transactions sont des saisies de données interactives.
4	De 24 à 30 % des transactions sont des saisies de données interactives.
5	Plus de 30 % des transactions sont des saisies de données interactives.

B.7 DEGRÉ D'INFLUENCE DE LA CONVIVIALITÉ

La convivialité peut être assurée par différentes fonctions : menu, navigation facile entre les écrans (touches de fonctions, débranchements…), aide en ligne, déplacement automatique du curseur, défilement de l'écran, sélection des données à l'écran avec le curseur, utilisation des formats particuliers (couleur, surbrillance…), interface pour souris, fenêtres déroulantes… Le multilinguisme de l'application est équivalent à quatre (pour deux langues) ou six fonctions (plus de deux langues) de convivialité.

0	Aucune fonction de convivialité.
1	Une à trois fonctions.
2	Quatre ou six fonctions.
3	Plus de six fonctions.
4	Plus de six fonctions et des exigences élevées d'efficacité de l'utilisateur final demandant une conception spécifique (par exemple : minimiser l'utilisation des touches).
5	Plus de six fonctions et des exigences d'efficacité de l'utilisateur final demandant l'utilisation de traitements spéciaux pour démontrer l'atteinte des objectifs.

B.8 DEGRÉ D'INFLUENCE DE LA MISE À JOUR EN TEMPS RÉEL DES GDI

0	Aucune.
1	Un à trois GDI, avec un faible volume et une restauration facile.
2	Quatre GDI, avec un faible volume et une restauration facile.
3	Tous les principaux GDI.
4	Tous les principaux GDI, avec une protection spécifique contre la perte.
5	Tous les principaux GDI, avec une protection spécifique contre la perte, des volumes importants et des procédures de restauration automatisées.

B.9 DEGRÉ D'INFLUENCE DE LA COMPLEXITÉ DES TRAITEMENTS

Plusieurs types de traitements peuvent être complexes : contrôle, traitement logique, algorithme, traitement des interruptions, traitement d'entrées/sorties multiples.

0	Aucun traitement complexe.
1	Un type de traitement complexe.
2	Deux types de traitement complexe.
3	Trois types de traitement complexe.
4	Quatre types de traitement complexe.
5	Tous les types de traitement complexe.

B.10 DEGRÉ D'INFLUENCE DE LA RÉUTILISATION

0	Aucune réutilisation.
1	L'application intègre des modules réutilisables.
2	Moins de 10 % des modules sont conçus multi-utilisation.
3	Plus de 10 % des modules sont conçus multi-utilisation.
4	L'application a été conçue multi-utilisation. L'utilisateur peut personnaliser le code source.
5	L'application a été conçue multi-utilisation. La personnalisation est paramétrée.

B.11 DEGRÉ D'INFLUENCE DE LA FACILITÉ D'INSTALLATION

0	Aucune exigence particulière.
1	Un réglage spécial est nécessaire.
2	Des exigences de conversion et d'installation sont spécifiées.
3	Des exigences de conversion et d'installation sont spécifiées et l'impact de la conversion sur le projet est important.
4	Les exigences spécifiées comprennent la fourniture d'outils automatisés de conversion et d'installation.
5	Les exigences spécifiées comprennent la fourniture d'outils automatisés de conversion et d'installation et l'impact de la conversion sur le projet est important.

B.12 DEGRÉ D'INFLUENCE DE LA FACILITÉ D'EXPLOITATION

La facilité d'exploitation se mesure en points. Elle peut comprendre : des procédures de démarrage, de sauvegarde et de restauration avec intervention manuelle (un point) ou sans (deux points), une réduction du montage des bandes (un point), une réduction des tâches de manipulation du papier (un point).

0	Aucune exigence particulière.
1	Un point.
2	Deux points.
3	Trois points.
4	Quatre points.
5	L'application doit fonctionner de façon autonome : l'opérateur n'intervient que pour démarrer et arrêter l'application.

B.13 DEGRÉ D'INFLUENCE DE LA PORTABILITÉ

0	Aucune exigence particulière.
1	Portabilité dans des environnements matériel et logiciel identiques.
2	Portabilité dans des environnements matériel et logiciel similaires.
3	Portabilité dans des environnements matériel et logiciel différents.
4	Des plans de maintenance multisite sont demandés pour des environnements matériel et logiciel identiques ou similaires.
5	Des plans de maintenance multisite sont demandés pour des environnements matériel et logiciel différents.

B.14 DEGRÉ D'INFLUENCE DE LA FACILITÉ D'ADAPTATION

La facilité d'adaptation se mesure en points. Elle permet à l'utilisateur de modifier les sorties obtenues ou les paramètres de l'application. Elle comprend les caractéristiques suivantes : possibilité de modifier facilement des demandes d'interrogation ou d'édition simples (un point), moyennes (deux points) ou complexes (trois points) ; mise à jour des paramètres utilisateur (par exemple :

valeur d'un taux de TVA) par traitement interactif avec prise en compte le prochain jour ouvrable (un point) ou immédiatement (deux points).

0	Aucune exigence particulière.
1	Un point.
2	Deux points.
3	Trois points.
4	Quatre points.
5	Cinq points.

Annexe C

Grille d'analyse des risques

La grille proposée permet de déterminer le degré de risque de chaque facteur.

Le degré de risque du facteur est calculé de la façon suivante :

Degré de risque d'un facteur
=
*Somme (degré de chaque critère * pondération) / Somme des pondérations*

Souvent, on prend la moyenne arithmétique de la valeur des critères correspondants. Parfois, une *pondération* peut être définie pour majorer l'importance du critère dans le facteur étudié. Par exemple pour le facteur instabilité de l'équipe de projet le critère image du projet (pondération : 2) peut être apprécié comme deux fois plus important que le critère technique attrayant (pondération : 1).

Par ailleurs un critère peut parfois être qualifié de dominant. Dans ce cas le degré calculé du facteur ne peut être inférieur au degré de ce critère. Par exemple si le degré du critère durée du projet du facteur taille du projet est de 3 le degré du facteur ne pourra être inférieur à 3.

Les critères et les métriques sont donnés ci-dessous.

Critères pour : Taille du projet	Degré	Métrique	
Durée du projet	1	=< 6 mois	
	2	=< 18 mois	
	3	=< 30 mois	
	4	> 30 mois	
Charge du projet	1	=< 20 mois/personne	
	2	=< 120 mois/personne	
	3	=< 300 mois/personne	
	4	> 300 mois/personne	
Couverture fonctionnelle	1	=< 2 processus	
	2	=< 6 processus	
	3	=< 10 processus	
	4	> 10 processus	
Critères pour : Difficulté technique	**Degré**	**Métrique**	
Expérience de l'entreprise sur l'architecture ciblée	1	2 projets	
	2	1 projet	
	3	Projet(s) technologie voisine	
	4	Aucune expérience	
Expérience de l'entreprise sur le langage de programmation	1	2 projets	
	2	1 projet	
	3	Projet(s) technologie voisine	
	4	Aucune expérience	
Expérience de l'entreprise sur le SGBD	1	2 projets	
	2	1 projet	
	3	Projet(s) technologie voisine	
	4	Aucune expérience	
Expérience du marché sur l'architecture ciblée	1	Plus de 200 références	
	2	Plus de 100 références	
	3	Plus de 50 références	
	4	Moins de 10 références	

Critères pour : Difficulté technique *(suite)*	**Degré**	**Métrique**	
Expérience du marché sur le langage de programmation	1	Plus de 200 références	
	2	Plus de 100 références	
	3	Plus de 50 références	
	4	Moins de 10 références	
Expérience du marché sur le SGBD	1	Plus de 200 références	
	2	Plus de 100 références	
	3	Plus de 50 références	
	4	Moins de 10 références	
Contrainte de performance (par rapport aux contraintes volumétriques habituelles de l'entreprise)	1	Pas de contrainte	
	2	Normale	
	3	Élevée	
	4	Très élevée	
Place et responsabilité de la Direction informatique	1	La Direction informatique assure la responsabilité technique	
	2	La Direction informatique anime la responsabilité technique	
	3	Pas de Direction informatique. La responsabilité technique est sous-traitée.	
	4	Pas de Direction informatique. La responsabilité technique n'est pas assurée.	
Critères pour : Degré d'intégration	**Degré**	**Métrique**	
Nombre de flux (type) avec des applications connexes	1	=< 1	
	2	=< 10	
	3	=< 20	
	4	> 20	
Nombre d'applications connexes en chantier	1	Aucune	
	2	= 1	
	3	= 2	
	4	> 2	

Critères pour : Configuration organisationnelle	Degré	Métrique	
Nombre de directions assurant la maîtrise d'ouvrage (MOA)	1	= 1	
	2	= 2	
	3	= 3	
	4	> 3	
Existence et implication d'un promoteur (parmi les MOA)	1	Nommé, impliqué, légitime	
	2	Nommé, impliqué	
	3	Nommé	
	4	Pas de promoteur	
Pérennité du promoteur (s'il existe)	1	Couvrira tout le projet	
	2	Pourra être relayé avec la même ligne (par une émanation de la DG)	
	3	Pourra être relayé avec une ligne différente	
	4	Pourra ne pas être relayé	
Appui de la Direction générale (DG)	1	Suivi par la DG	
	2	DG informée	
	3	Pas d'implication de la DG, mais une seule MOA	
	4	Pas d'implication de la DG et plusieurs MOA	
Critères pour : Changement	**Degré**	**Métrique**	
Degré d'évolution organisationnelle (pour les utilisateurs)	1	Pas d'écart	
	2	Écart faible	
	3	Écart moyen	
	4	Écart fort	
Degré d'évolution fonctionnelle (pour les utilisateurs)	1	Pas d'écart	
	2	Écart faible	
	3	Écart moyen	
	4	Écart fort	

Critères pour : Changement *(suite)*	**Degré**	**Métrique**	
Degré d'évolution technique (pour les utilisateurs)	1	Pas d'écart	
	2	Écart faible	
	3	Écart moyen	
	4	Écart fort	
Impact social	1	Aucune conséquence	
	2	Réduction d'effectif	
	3	Réduction salaire	
	4	Réduction effectif et salaire	
Nombre de sites concernés	1	1	
	2	Moins de 10	
	3	Moins de 50	
	4	Plus de 50	
Critères pour : Instabilité de l'équipe	**Degré**	**Métrique**	
Durée du projet	1	=< 6 mois	
	2	=< 18 mois	
	3	=< 30 mois	
	4	> 30 mois	
Sous-traitance de la maîtrise d'œuvre	1	0 %	
	2	40 %	
	3	70 %	
	4	100 %	
Turn-over/mobilité dans l'entreprise	1	Très faible	
	2	Faible	
	3	Moyenne	
	4	Forte	
Relation MOA - MOE	1	Excellente	
	2	Bonne	
	3	Moyenne	
	4	Mauvaise	
Techniques attrayantes	1	Très	
	2	Normale	
	3	Peu	
	4	Pas	

Critères pour : Instabilité de l'équipe *(suite)*	**Degré**	**Métrique**	
Conjoncture (marché de l'emploi)	1	Difficile (pour les employés)	
	2	Moyenne	
	3	Favorable	
	4	Très favorable	
Image du projet	1	Très valorisante	
	2	Valorisante	
	3	Moyennement valorisante	
	4	Faiblement valorisante	

Annexe D

Exemples de questions pour la certification

1.	Vous êtes le chef d'un projet d'un nouvel intranet pour une compagnie d'assurance. Bien entendu, la direction veut que votre projet rapporte beaucoup (en réduisant les coûts de communication interne) et coûte peu. Le Directeur informatique est particulièrement exigeant sur les questions de sécurité. Les développeurs pressentis souhaitent expérimenter un nouvel outil de développement. Quand vous allez travailler avec les différentes parties prenantes du projet, vous devriez : A. Répartir les parties prenantes en catégories pour les identifier facilement. B. Être attentif au fait que les parties prenantes peuvent avoir des attentes très différentes et leur gestion est parfois délicate. C. Reconnaître que les rôles et responsabilités peuvent se chevaucher. D. Réduire les activités des parties prenantes qui peuvent avoir un effet négatif sur le projet.
2.	Quelle forme de résolution de conflit utilise-t-on lorsque l'on déclare « Je ne veux pas traiter ce problème maintenant » ? A. Compromis. B. Apaisement. C. Retrait. D. Coopération.

3.	Vous construisez un centre d'exploitation informatique de 500 serveurs en Papouasie pour mettre en œuvre un grand projet de e-commerce principalement en direction de l'Asie. Cet endroit offre des avantages économiques certains, mais le risque de typhon vous a conduit à prévoir un repli sur le site de Manille en cas d'inondation du centre. Ceci est un exemple de : A. Acceptation passive. B. Évitement. C. Acceptation active. D. Réponse conditionnelle.
4.	Vous venez d'être nommé responsable d'un grand projet de télécommunication qui a commencé depuis un an et met en jeu 25 personnes. Vous voulez savoir précisément qui est responsable de quoi dans ce projet. Où pouvez-vous trouver cette information ? A. Diagramme hiérarchique. B. Grille RACI. C. Structure de découpage du projet. D. Fiche de compétence.
5.	Vous travaillez sur un projet de site web visant les propriétaires d'animaux exotiques. Au début, le produit a été défini par votre client comme « un portail offrant des liens avec tous les sites intéressants ». Puis il a été décrit comme « un portail offrant des liens avec tous les sites intéressants, ainsi que des informations juridiques sur le transport de ces animaux ». Ceci montre que l'élaboration progressive des spécifications du produit doit être soigneusement coordonnée avec : A. Les parties prenantes. B. Un système de maîtrise des modifications du projet. C. Le plan stratégique du client. D. Une définition précise du contenu du projet.
6.	Une activité a comme date de début au plus tôt le 3 avril, comme date de début au plus tard le 13 avril, comme de date de fin au plus tôt le 9 avril, comme date de fin au plus tard le 19 avril. Sa marge libre est nulle. Vous en déduisez que cette activité : A. est sur le chemin critique. B. a un retard. C. avance correctement. D. n'est pas sur le chemin critique.
7.	On vous demande de terminer votre projet trois jours plus tôt que prévu. Quelle est la *meilleure* chose à faire ? A. Préparer un plan de secours. B. Demander à votre équipe de travailler plus vite. C. Réunir l'équipe et regarder les possibilités de compression des délais ou de chevauchement des tâches qui sont sur le chemin critique. D. Demander une ressource supplémentaire.

8.	Votre client vous demande de concevoir une application facile d'apprentissage. Vous convenez ensemble que la formation nécessaire à la prise en main ne doit pas dépasser deux heures. Ceci est *avant tout* un exemple de : A. Cycle en V. B. Bonne communication avec les parties prenantes. C. Détermination d'une métrique de la qualité produit. D. Participation.
9.	Un membre de votre équipe vient de démissionner brusquement et a été autorisé à ne pas effectuer son préavis. Votre projet est dans une situation critique. En attendant l'arrivée d'une nouvelle ressource, vous devez réorganiser le travail et éviter que le moral de l'équipe, jusque-là très motivée, ne baisse. Le style de leadership à privilégier est : A. Directif après information. B. Consultatif. C. Directif, puis persuasif. D. Persuasif et participatif groupe.
10.	Vous êtes responsable d'un projet robotique. Une nouvelle technologie vient d'être mise sur le marché, qui va améliorer la précision des connecteurs et réduire le travail nécessaire. On vous demande une nouvelle estimation du coût total de votre projet (PAA). La meilleure méthode consiste à : A. Effectuer une analyse de la VA pour déterminer l'indice de performance des coûts. B. Ajouter les coûts réels et le budget restant modifié d'un facteur de performance. C. Ajouter les coûts réels et le résultat d'une nouvelle estimation pour le travail restant. D. Ajouter le budget restant à ce qui a déjà été dépensé.
11.	En vous basant sur tout un ensemble d'estimations, vous savez que votre projet va probablement durer 50 jours, au pire 120 et au mieux 40 jours. Quelle valeur retenez-vous ? A. 45 jours. B. 70 jours. C. 50 jours. D. 60 jours.
12.	Pour réduire la durée de votre projet, vous décidez de confier une activité à deux personnes au lieu d'une seule. Cette approche tend à : A. Améliorer la communication dans l'équipe. B. Augmenter les coûts de coordination. C. Améliorer la valeur acquise. D. Utiliser la marge totale.

13.	Un membre de votre équipe manque d'expérience dans le langage de programmation utilisé, ce qui se traduit par une faible performance. Personne d'autre n'est disponible dans l'équipe. Quelle est la MEILLEURE solution pour le chef de projet ? A. Proposer une prime de productivité. B. Obtenir une ressource externe qualifiée. C. Obtenir une formation pour la ressource. D. Replanifier les tâches de développement pour les confier à quelqu'un de plus expérimenté.
14.	Votre projet comporte plusieurs commanditaires et vous savez que la description du contenu du produit est susceptible de nombreuses modifications. Que devez-vous faire de façon impérative ; A. Mettre en place une gestion de configuration. B. Refuser les changements inutiles. C. Demander un correspondant unique parmi vos commanditaires. D. Prévoir des ressources additionnelles.
15.	Un membre de votre équipe, basé à Evry, vous informe vendredi midi qu'il a commencé lundi dernier la tâche « Développement de l'interface utilisateur », sur laquelle il a travaillé toute la semaine, qu'il est convoqué l'après-midi pour une visite périodique de la médecine du travail à Paris et qu'il lui reste quatre jours de travail pour terminer cette tâche que vous aviez estimée à 9 jours. Vous en déduisez que son avancement de la semaine est de : A. 4,5 jours. B. 5 jours. C. 4 jours. D. 5,5 jours.

Les réponses sont les suivantes :

- 1.B : Les parties prenantes doivent être gérées.
- 2.C : On évite de rencontrer l'autre partie et d'aborder le problème.
- 3.D : Le plan est mis en œuvre sous condition.
- 4.B : La grille RACI est plus précise que l'organigramme hiérarchique en ce qui concerne les responsabilités sur les activités.
- 5.D : Le travail à faire (contenu du projet) dépend des demandes exprimées (contenu du produit).
- 6.D : Les dates au plus tôt étant inférieures aux dates au plus tard, la marge totale est positive, la tâche n'est donc pas sur le chemin critique.

- 7.C : Rajouter une ressource pour gagner trois jours n'est généralement pas très efficace. La meilleure solution est de chercher à réduire la durée du chemin critique.
- 8.C : On détermine une façon de mesurer l'atteinte d'une exigence. Cette métrique pourra être utilisée dans un cycle en V, mais pas nécessairement.
- 9.C : En situation de crise, il est préférable d'agir rapidement, mais sans oublier de communiquer. Une autre possibilité aurait été de privilégier un style participatif, mais elle ne figurait pas dans les choix.
- 10.C : La prévision à l'achèvement inclut toujours les coûts réels : compte tenu du changement dans la technique utilisée, il faut faire une nouvelle estimation.
- 11.D : On considère que la valeur la plus probable a été fréquemment citée, on prend donc la formule [(au pire) + (au mieux) + 4*(la plus probable)]/6.
- 12.B : Diviser un travail visant un livrable unique augmente les coûts de coordination.
- 13.C : Pour favoriser le développement de l'équipe, il est préférable d'augmenter, si possible, les compétences de ses membres.
- 14.A : La gestion de configuration permettra de suivre les évolutions dans la description du produit attendu.
- 15.B : L'avancement est la différence entre deux « reste à faire » successifs, soit 9 – 4.

Bibliographie

Gestion de projet (français)

AFITEP — *Dictionnaire de management de projet*, 4e édition, Afnor, 2000.

AFITEP — *Le Management de projet, Principes et Pratique*, Afnor, 1998.

AUTISSIER D. et MOUTOT J.-M.. — *Pratiques de la conduite du changement*, Dunod, 2003.

BENNATAN E.-M. — *Management des projets informatiques*, Afnor, 1995.

CORBEL J.-C. — *Le Management de projet : fondamentaux, méthodes, outils*, Éd. Organisation, 2003.

GILBERT P., GUÉRIN F. et PIGEYRE F. — *Organisations et comportements*, Dunod, 2005.

HUGUES J., LEBLANC B. et MORLEY C. — *RAD, une méthode pour développer plus vite*, InterEditions, 1996.

IPMA — ICB, IPMA *Competence Baseline*, Version 2.0, IPMA, www.afitep.fr, (1999)

MADERS H.P. et CLET E. — *Comment manager un projet : Les sept facettes du management de projet*, Éd. Organisation, 2003.

MARCINIAK R. et CARBONEL M. — *Management des projets informatiques. Études de cas*, Afnor, 1996.

MORLEY C., HUGUES J. et LEBLANC B. — *UML 2 pour l'analyse d'un système d'information, Le cahier des charges du maître d'ouvrage*, Dunod, 2006.

NANCI D. et ESPINASSE B., avec la collaboration de COHEN B., ASSELBORN J.-C. et HECKENROTH H. — *Ingénierie des systèmes d'information : Merise deuxième génération*, 4e éd., Vuibert, 2001.

PMI — *Guide du corpus des connaissances en management de projet* (Guide du PMBOK), 2004.

REIX R — *Systèmes d'information et management des organisations*, 5e éd., Vuibert, 2004.

Gestion de projet (anglais)

BOEHM B.W. — *Software Engineering Economics*, Prentice Hall, 1984.

CADLE J. et YEATES D. — *Project management for information systems*, Prentice Hall, 2001.

DREGER J.B. — *Function Points Analysis*, Prentice Hall, 1989.

HUGHES, B., and COTTERELL, M. — *Software Project Management*, McGraw Hill, 1999.

KEZSBOM D.S., SCHILLING D.L. et EDWARD K.A. — *Dynamic Project Management. A practical Guide for Managers and Engineers*, John Wiley&Sons, 1989.

MULCAHY R. — *PMP Exam Prep*, 5e édition, RMC Publications, 2005.

ROYCE, W. — *Software Project Management : A Unified Framework*, Addison Wesley, 1998.

STAPLETON J. — *DSDM, dynamic system development method*, Addison Wesley, Harlow, 1997.

Sites internet

http://www.afitep.fr.

http://www.ipma.ch.

http://www.4pm.com.

http://www.pmi.org.

http://www.spmn.com.

http://www.standishgroup.com.

http://www.cw360ms.com/pmsurveyresults/index.asp.

Articles

Général

ANDERSON D.K. et MERNA T. — « Project Management Strategy – project management represented as a process based set of management domains and the consequences for project management strategy », *International Journal of Project Management*, 2003.

DELISLE C.L. et OLSON D. — « Would the real project management language please stand up ? », *International Journal of Project Management*, Available online 4 November 2003.

DVIR D., RAZ T. et SHENHAR A.J. — « An empirical analysis of the relationship between project planning and project success », *International Journal of Project Management*, Volume 21, Issue 2, February 2003, p. 89-95.

SCHINDLER M. et EPPLER M.J. — « Harvesting project knowledge : a review of project learning methods and success factors », *International Journal of Project Management*, 21, 2003, p.219-228.

SCOTT L. et *alii*. — « Understanding the use of an electronic process guide », *Information and Software Technology*, Vol. 44, 2002, p. 601-616.

SÖDERLUND J. — « Building theories of project management : past research, questions for the future », *International Journal of Project Management*, In Press, Corrected Proof, Available online 5 December 2003.

WHITE D. et FORTUNE J. — « Current practices in project management – an empirical study », *International Journal of Project Management*, 20, 2002, p. 1-11.

WILSON J.M. — « Gantt charts : A centenary appreciation », *European Journal of Operational Research*, Volume 149, Issue 2, 1 September 2003, p. 430-437.

Management équipe/Ressources humaines/Culture

BARBER E. — « Benchmarking the management of projects : a review of current thinking », *International Journal of Project Management*, Available online 13 November 2003.

BELOUT A. et GAUVREAU C. — « Factors influencing project success : the impact of human resource management », *International Journal of Project Management*, Vol. 22, Issue 1, January 2004, p. 1-11.

EL-SABAA S. — « The skills and carreer path of an effective project manager », *International Journal of Project Management*, Vol. 19, 2001, p. 1-7.

MAHANEY R.C. et LEDERE A.L. — « Information system project management : an agency theory interpretation », *Journal of Systems and Software*, Vol. 68, Issue 1, October 2003, p. 1-9.

MCCHESNEY I.R. et GALLAGHER S. — « Communication and co-ordination practices in software engineering projects », *Information and Software Technology*, In Press, Corrected Proof, Available online 3 December 2003.

MURIITHI N. et CRAWFORD L. — « Approaches to project management in Africa : implications for international development projects », *International Journal of Project Management*, Vol. 21, Issue 5, July 2003, p. 309-319.

Estimation

BIELAK J. — « Improving size estimaztes using historical data », *IEEE Software*, nov.déc. 2000.

BOOTSMA F. — *How to Obtain Accurate Estimates in a Real-Time Environment using full function points*, 2000, IEEE.

CAIVANO D., LANUBILE F., VISAGGIO G. — « Software Renewal Process Comprehension Using Dynamic Effort Estimation », *Software Maintenance*, 2001. Proceedings, IEEE International Conference, 7-9 Nov. 2001.

HÖST M. et WOHLIN C. — *An experimental study of individual subjective effort estimations and combinations of estimates*, 1998, IEEE.

MENDES, E., MOSLEY, N., COUNSELL, S. — « The application of case-based reasoning to early Web project cost estimation », *Computer Software and Applications Conference*, 2002. COMPSAC 2002. Proceedings. 26th Annual International, 26-29 Aug. 2002.

MOSES J. — *Learning how to improve effort estimation in small software development companies*, 2000, IEEE.

OHLSSON, M.C., WOHLIN, C. — « An empirical study of effort estimation during project execution », *Software Metrics Symposium*, 1999. Proceedings. Sixth International, 4-6 Nov. 1999, p. 91-98.

PASSING U. et SHEPPERD M. — *An experiment on software project size and effort estimation*, Proceedings of the 2003 International Symposium on Empirical Software Engineering (ISESE'03), 2003, IEEE.

STAMELOS I., ANGELIS L., MORISIO M., SAKELLARIS E. et BLERIS G.L. — « Estimating the development cost of custom software », *Information & Management*, Vol. 40, Issue 8, September 2003, p. 729-741.

Participation/Décision

BEYNON-DAVIES P. et HOLMES S. — « Design breakdown, scenarios and rapid application development », *Information and Software Technology*, Vol.44, Issue 10, July 2002, p. 579-592.

HOWCROFT D. et WILSON M. — « Paradoxes of participatory practices : the Janus role of the system developer », *Information and Organization*, 13, 2003, p. 1-24.

MEKHILEF M. et STAL LE CARDINAL J. — « A pragmatic methodology to capture and analyse decision dysfunctions in development projects », *Technovation*, Available online 24 October 2003.

Risques

ADDISON T. — « E-commerce project development risks : evidence from a Delphi survey », *International Journal of Information Management*, Vol. 23, 2003, p.25-40.

AGARWAL R.,.TANNIRU M. et WILEMON D. — « Assimilating Information Technology Innovations : Strategies and Moderating Influences », *IEEE transactions on Engineering Management*, Vol. 44, n°4, novembre1997.

BARROS M.O., WERNER C.M.L. et TRAVASSOS G.H. — « Supporting risks in software project management », *Journal of Systems and Software*, Available online 28 September 2003.

DAVIS F.D. — « Perceived Usefulness, Perceived Ease of Use and User acceptance of Information Technology », *MIS Quarterly*, vol. 13, n° 3, p. 319-340, 1989.

ELKINGTON P. et SMALLMAN C. — « Managing project risks : a case study from the utilities sector », *International Journal of Project Management*, Vol. 20, 2002, p. 49-57.

JIANG J. *et alii*. — Reducing user-related risks : during and prior to system development », *International Journal of Project Management*, Vol. 20, 2002, p. 507-515.

KWAK Y. H. et STODDARD J. — « Project risk management : lessons learned from software development environment », *Technovation*, Available online 26 March 2003.

NA K.S., LI X., SIMPSON J.T. et KIM K.Y. — « Uncertainty profile and software project performance : A cross-national comparison », *Journal of Systems and Software*, Available online 28 September 2003.

SCOTT J.E. et Vessey I. — « Managing risks in enterprise systems implementations », *Communication of the* ACM, Vol. 45, n° 4, April 2002, p. 74-81.

SUH B. et HAN I. — « The IS risk analysis based on a business model », *Information & Management*, 2003.

WARD S. et CHAPMAN C. — « Transforming project risk management into project uncertainty management », *International Journal of Project Management*, Vol. 21, Issue 2, February 2003, p. 97-105.

Méthodes agiles

ABRAHAMSSON, P., WARSTA, J., SIPONEN, M.T., RONKAINEN, J. — « New directions on agile methods : comparative analysis », *Software Engineering*, 2003. Proceedings. 25th International Conference on, 3-10 May 2003, p. 244-254.

FOWLER M. — « The New Methodology », http://martinfowler.com/articles/, 2003.

MÉDINA R. — « L'Extreme Programming », www.design-up.com, 2000.

PAETSCH F., EBERLEIN A. et MAURER F. — « Requirements engineering and agile software development », *Enabling Technologies : Infrastructure for Collaborative Enterprises*, 2003. WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on, Jun 9-11, 2003, p. 308-313.

SPAYD M.K. — « Evolving agile in the enterprise : implementing XP on a grand scale », *IEEE*, Proceedings of the Agile Development Conference, 2003.

Index

A

achèvement 308
acteur 14, 104, 108, 125, 157, 323
activité 26, 57, 75
Actual Cost (AC) 173
adaptation 165, 167
administration de données 106
AFAQ (Association française d'assurance qualité) 192, 216
AFITEP 6, 8, 12, 29, 30, 31, 94, 106, 109, 114, 140, 141, 143, 150, 209, 381, 412
AFNOR 8, 12, 29, 35, 109, 114, 150, 193, 195, 196, 209
AGL (atelier de génie logiciel) 34, 106, 110, 186
ajustements mutuels 105
Albrecht 71
aléa 11, 164, 300, 367, 405
alignement stratégique 16
amélioration 385
analyse
  de la valeur 64
  des risques 142, 180, 223
  qualitative des risques 413
  quantitative des risques 413
appréciation (étape) 37, 63
apprentissage 165, 185
arbre de décision 176, 414
assurance qualité 190, 192, 198, 408
audit 155, 212, 216, 408, 414, 415
  qualité 215
avance 84, 85, 86, 404
avancement 115, 116, 166, 296, 355, 361

B

basculement 65
Belbin 111
Bell Telephone 190
bilan
  de projet 181, 310
  du projet Parking 373
  individuel 170, 297, 365
Boehm B.W. 67, 70, 200
Borel G. 192
budget 384, 406
  à l'achèvement (BAA) 173, 367
Budgeted At Completion (BAC) 173, 367
burndown chart 187, 419

C

cahier des charges 33, 38, 64, 66, 109, 180
calendrier 406
  de travail 82
capitalisation d'expérience 181, 312
CBTE 173, 366
CBTP 173, 366
CCPM (*Critical Chain Project Management*) 94
cercle de qualité 192
certification vi, 215, 379, 391, 443
changement 114, 123, 154
charge 55, 69, 74, 167

chef de projet 5, 16, 25, 106, 109, 114, 180, 331
chemin critique 90, 97, 166, 272, 275, 405
chronique du cash-flow 177
COCOMO (COCOMO II) 60, 67, 70, 75, 241, 248, 287
coefficient d'utilisation 170, 297
cohérence (cohérent) v, 28, 209
comité de pilotage 13, 105
communication 129, 186, 384, 410
composants réutilisables 33
compte rendu (avancement, activité) 168, 294, 388
concepteur 110
conception détaillée 32
conception-organisation (étape) 37, 63
confiance 127, 407
conflit 122, 329, 410, 411
contenu
  du produit 394
  du projet 384, 394, 395, 402, 403
contrôle 165
  des risques 151
  qualité 212, 408
coordination 103, 104, 115, 153, 383, 410
Cost Variance (CV) 173
coût 384, 406
coût réel (CR) 173, 316, 366
critère 201, 437
  qualité 39, 204, 407
Crosby P.B. 192
CRTE 173, 366
Crystal 19, 21
cybernétique 163
cycle de vie 62, 395
  classique 35, 56
  du produit 396
  du projet 26
  standard 32

D

date
  au plus tard 87, 88, 405
  au plus tôt 87, 405
décision (prise de) 114, 118, 154, 323
découpage 25, 26, 28, 81, 156, 180, 403
définition des solutions 32
délai 68, 69, 82, 84, 103, 287, 288, 403, 405
  de remboursement 177
Delphi (méthode) 59, 61, 77, 142, 180, 241, 246, 404
Deming W.E. 191, 385
déploiement 128
Descartes 25
design-to-cost 14
détermination des besoins 33, 112, 113
développement
  de l'équipe 410
  incrémental 28
développeur 110
diagramme
  cause-effet 143
  des antécédents 30, 404
  en tornade 413
directeur de projet 110
direction de projet 12
division du travail 104
documentation 64, 129
domaine de connaissance 393
DSDM 19, 20, 49, 50
durée 56, 81, 95, 98, 103

E

Earned Value (EV) 173
Earned Value Method 173
écart 166, 406
  à l'achèvement (EAC) 175, 367
  de coût 173, 366
  de délai 173, 366
écart-type 98, 282
échantillonnage 77
échéancier 81, 405
Edwards G. 190
effet tunnel 41
encadrement du projet 64
ENT (entrée) 72, 73, 254
équipe (de projet) 111, 115, 116, 156, 323, 408
ERP 45, 320
Estimate At Completion (EAC) 174
Estimate To Complete (ETC) 175
estimation 55, 180, 256, 407, 451
  à trois points 61, 75, 405
  ascendante 59, 66, 404

des charges 241
par analogie 59, 62
paramétrique 60, 71
probabiliste 61, 75
étude
de faisabilité 32, 396
détaillée 38, 62, 63
préalable 36, 62, 63, 247
technique 38, 63
Eurométhode 109, 143, 386, 412
évolution de la charge 172, 307

F

facteur 194, 201, 437
correcteur 69, 429
d'hygiène 120
de risque 147
qualité 200
Feigenbaum A.V. 190, 191
FFOM (grille) 142
FFPUG 71
FNCV (fichier national des chèques volés) 34
forfait 179

G

Gantt 81, 82, 91, 93, 96, 103, 166, 273, 351, 405, 406
Gauss 98, 99, 405
GDE (groupe de données externe) 72, 73, 251
GDI (groupe de données interne) 71, 73, 251
GDR 73
General Electric 190, 201
gestion
de configuration 403
de projet 13
par projet 7
Goldratt E. 94, 406
graphe (d'ordonnancement, des antécédents) 82, 265, 266, 268, 270, 272, 275, 287, 404
grille
d'analyse des risques 148, 412, 437
multicritère 180, 415
groupe de processus 396

H

Herzberg F. 120
hiérarchie des besoins 118

I

IBM 71
IFPUG 71
incertitude 1, 98, 141, 282, 395
incrémental 44, 48, 53, 78
indicateur normalisé 167, 173
indice de performance
des coûts (IPC) 174, 316
cumulé (IPCC) 174, 316
des délais (IPD) 174, 316
requis (IPR) 175, 316
INT (interrogation) 72, 73, 255
intégration 401
INTERPLACE 112
IPMA 6, 12, 30, 381, 391
Ishikawa K. 143, 192, 193
ISO/IEC 15504 219
ISO10006 8, 150, 196, 382, 393
ISO9000 192, 196, 197, 215, 382, 393, 415
itératif 43, 44, 48, 50, 53, 78, 390

J

jalon 27, 388
journal de bord 305, 411
jugement d'expert 59, 61
Juran J.M. 191
JUSE 191, 192

L

La Cible 94, 144, 381
lien 83, 404
début-début 85
début-fin 86
fin-début 84
fin-fin 84
lissage 97, 406
livrable 26, 41, 114, 180, 394, 402, 403
lot 171, 245, 370
de travail 30, 395, 402

M

maître
  d'œuvre 109, 114, 194, 210
  d'ouvrage (maîtrise d'ouvrage, MOA) 16, 106, 109, 114, 155, 166, 181, 194, 210, 329
maîtrise 5, 13
management
  de projet 9, 10, 12, 396
  des connaissances 181
  des risques 141
  par les délais 10
manuel qualité 194, 209
marge
  libre 87, 90, 405
  totale 87, 90, 405
Maslow A. 118
matrice d'affectation des responsabilités 409
maturité 217
McCall 201
McGregor D. 120
mêlée 21, 135
Mélèse J. 25
mémoire 183
Merise (méthode) 35
méthode
  agile 452
  d'estimation 58, 241
  d'évaluation analytique 59, 66, 241, 245, 249, 404
  de Monte-Carlo 61, 75, 405
  de répartition proportionnelle 59, 62, 241, 247, 405
  des antécédents 82
  des points de fonction (fonctionnels) 60, 71, 180, 241, 250, 405, 429
  des ratios 59, 64, 405
  des rôles 111
  du chemin critique 86
  du diagramme fléché 82
  REX 183
métrique 201, 205, 337, 437
migration 130
military standards 190
Mintzberg H. 104
mise en œuvre 39, 65, 109, 387
mode Projet 7
modèle
  d'acceptation technologique 126
  de développement 39, 153, 387, 403
    en cascade 153
    en V 153
    évolutif 43
  de la cascade 41
  de la spirale 44
  de la transformation automatique 40
  du *code-and-fix* 40, 41
  en V 41
  en W 42, 153
motivation 118, 157, 410
MS Project 30, 83, 188, 341

N

niveaux d'estimation 56
nivellement 95, 406
Nonaka 184
norme 195, 198, 381

O

objectifs du projet 16, 157
OBS (*Organisation Breakdown Structure*) 30
observation (étape) 37, 63, 247
ordonnancement 81
organigramme
  des tâches (OT) 30
  fonctionnel (OF) 31
  hiérarchique 409
organisation (du travail) 103, 126, 130, 184, 338
Ouchi W. 121

P

Pareto (matrice) 143
Parkinson 58
participation 112, 153, 451
partie prenante 108, 150, 411
PBS (*Product Breakdown Structure*) 28
Pensée latérale 151
performance 170, 297, 402, 410
personnel 383
PERT 83, 98, 100, 404, 405
phase 26, 32, 57, 75, 395
pilotage 13, 153, 163, 181, 355

plan
de management de projet 150, 339, 407
qualité 194, 210, 335, 407
planification
des délais 81, 261
par vagues 404
Planned Value (PV) 173
PMBOK 8, 140, 150, 392
PMI vi, 6, 8, 30, 381, 391
point de fonction (fonctionnel) 71, 256
polyvalence 104
prévision à l'achèvement (PAA) 174, 317, 366
prise de décision 113
probabilité 282, 405, 411
probabilité (loi) 98
probabilité/impact 143
procédure 185, 198
processus 8, 15, 31, 217, 219, 336, 338, 382, 393
produit 394
profil de risque 147, 225
projet 6, 8, 386, 394
système d'information 13, 163, 182
prototype (prototypage) 43, 48, 77

Q

qualification 39, 215
qualité 11, 189, 193, 215, 335, 407
totale 190

R

RACI (grille) 409
RAD (*Rapid Application Developpement*) 153, 227
Rand Corporation 61
réalisation 32, 38, 63, 67, 68, 249
récapitulatif mensuel 169, 294
recette 64, 109
recherche opérationnelle 5
régie 179
régulation 165
Relit (projet) 34, 140
rentabilité 37, 56, 177, 285, 320
ressource humaine 408, 451
reste à faire 168, 358, 366
retard 84, 85, 86, 404
risque 140, 223, 384, 411, 437, 452
ROI (*Return On Investment*) 177, 320, 402
RUP (*Rational Unified Process*) 45, 46

S

SAP 140, 320
Schedule Variance (SV) 173
schéma directeur 36
SCRUM 19, 21, 52, 101, 135, 136, 137, 187
SDMS 35
SDP 395
SDP (structure de découpage du projet) 30, 404
SEI 217
session 133
Shewhart W. 190, 191
SNCF 130, 139
SOR (sortie) 72, 73, 255
sous-ensemble représentatif 37
sous-projet 28, 37, 57
sous-traitance (sous-traiter) 57, 179, 385, 414
spike 79, 187
SQC (*Statistical Quality Control*) 190
standardisation 105
Standish Group 139, 155, 414
standup meeting 135, 419
stratégie de développement 152, 226, 413
structurel (critère, découpage) 27, 31, 37, 59, 227
style de management 116, 323, 410
suivi 167, 171, 173, 180
supervision directe 105
SW-CMM 217
système
d'information 1, 15, 31, 123, 143, 198
projet (de pilotage) 166, 285, 341
informatique 15, 143
qualité 198

T

tableau
d'avancement du projet 171, 306, 369
de bord 153, 164, 166, 181, 406, 411
tâche 26, 57, 75, 83, 167

tampon 94, 405
Taurus (projet) 140
taux de rendement interne (TRI) 178, 321
Taylor F.W. 190
temporel (critère, découpage) 26, 33, 35
temps passé 168, 366
théorie
  de l'équité 121
  de l'expectation 121
  X/Y 120
  Z 121
timebox 102, 187, 419
TQC (*Total Quality Control*) 190
triangle Projet 7, 9

U

unité d'œuvre 60, 71, 181, 245, 312, 404
US Air Force 57, 58
utilisateur 111, 155, 156

V

valeur
  acquise (VA) 173, 285, 315, 316, 366, 402, 406, 407, 414
  actuelle nette (VAN) 178, 321
  monétaire attendue (VMA) 285, 319, 402, 404, 413, 414
  planifiée (VP) 173, 316
variable
  d'action 164
  essentielle 164
variance 98, 282
Variance at completion (VAC) 175, 367
variété 165
VC (*Value Cost*) 316, 366
version 28, 44, 45
vision
  dynamique 31
  statique 31
vitesse 170, 297
Vroom V.H. 121
VS (*Value Schedule*) 366

W

WBS (*Work Breakdown Structure*) 28, 37, 81, 395
Western Electric 190

X

XP 19, 20, 50, 51, 52

Z

zéro défaut 192
zone d'incertitude 127

INFOPRO

TYPE D'OUVRAGE

L'ESSENTIEL | SE FORMER | **RETOURS D'EXPÉRIENCE**

- **MANAGEMENT DES SYSTÈMES D'INFORMATION**
- APPLICATIONS MÉTIERS
- ÉTUDES, DÉVELOPPEMENT, INTÉGRATION
- EXPLOITATION ET ADMINISTRATION
- RÉSEAUX & TÉLÉCOMS

Chantal Morley

MANAGEMENT D'UN PROJET SYSTÈME D'INFORMATION

Principes, techniques, mise en œuvre et outils

6e édition

Cet ouvrage s'adresse aux responsables de systèmes d'information et aux chefs de projets, ainsi qu'aux étudiants en informatique ou système d'information et aux élèves ingénieurs.

Quelle est la meilleure façon de **conduire un projet** système d'information ? Ce livre répond à cette interrogation en analysant les **outils** et les **méthodes** de gestion du domaine à partir des points clés que sont :

– l'analyse et le découpage d'un projet ;
– l'évaluation des risques ;
– l'estimation des charges ;
– les techniques de planification ;
– l'organisation du travail ;
– la dimension humaine et relationnelle du projet ;
– le pilotage du projet ;
– la maîtrise et la qualité du projet.
– les principales normalisations internationales.

Chacun de ces points clés fait l'objet d'exemples de mise en oeuvre, d'exercices et d'études de cas détaillés et explicités. La planification et le pilotage d'un projet sont illustrés avec le **progiciel MS Project 2003**. De plus, l'ouvrage apporte une aide à la préparation de la certification en management de projet du **PMI**.

Cette sixième édition introduit pour chaque aspect du management de projet une perspective particulière sur les **méthodes agiles**.

CHANTAL MORLEY

docteur ès sciences de gestion (HEC), a exercé de nombreuses responsabilités autour des systèmes d'information (développement, conception, modélisation, gestion de projet). Elle est actuellement professeur à l'Institut Télécom / Télécom & Management Sud Paris.

ISBN 978-2-10-053551-4

www.dunod.com

DUNOD