

EXERCICES DIRIGES DE GÉNIE LOGICIEL

- **GESTION DE PROJET**
- **Langage unifié de Modélisation (LUM/UML)**
- **TEST DU LOGICIEL (méthode des couvertures)**

**ANNALES DES EXAMENS
POUR LA PERIODE 1994-2006**

**RECUEIL EXHAUSTIF DES SUJETS
PROPOSES
AVEC UNE LARGE SELECTION DE
CORRIGES**

PREMIERE PARTIE

GESTION DE PROJET (avec le modèle cocomo)

Dans cette section tous les corrigés sont fournis

LES PARAMETRES DU MODELE

COCOMO

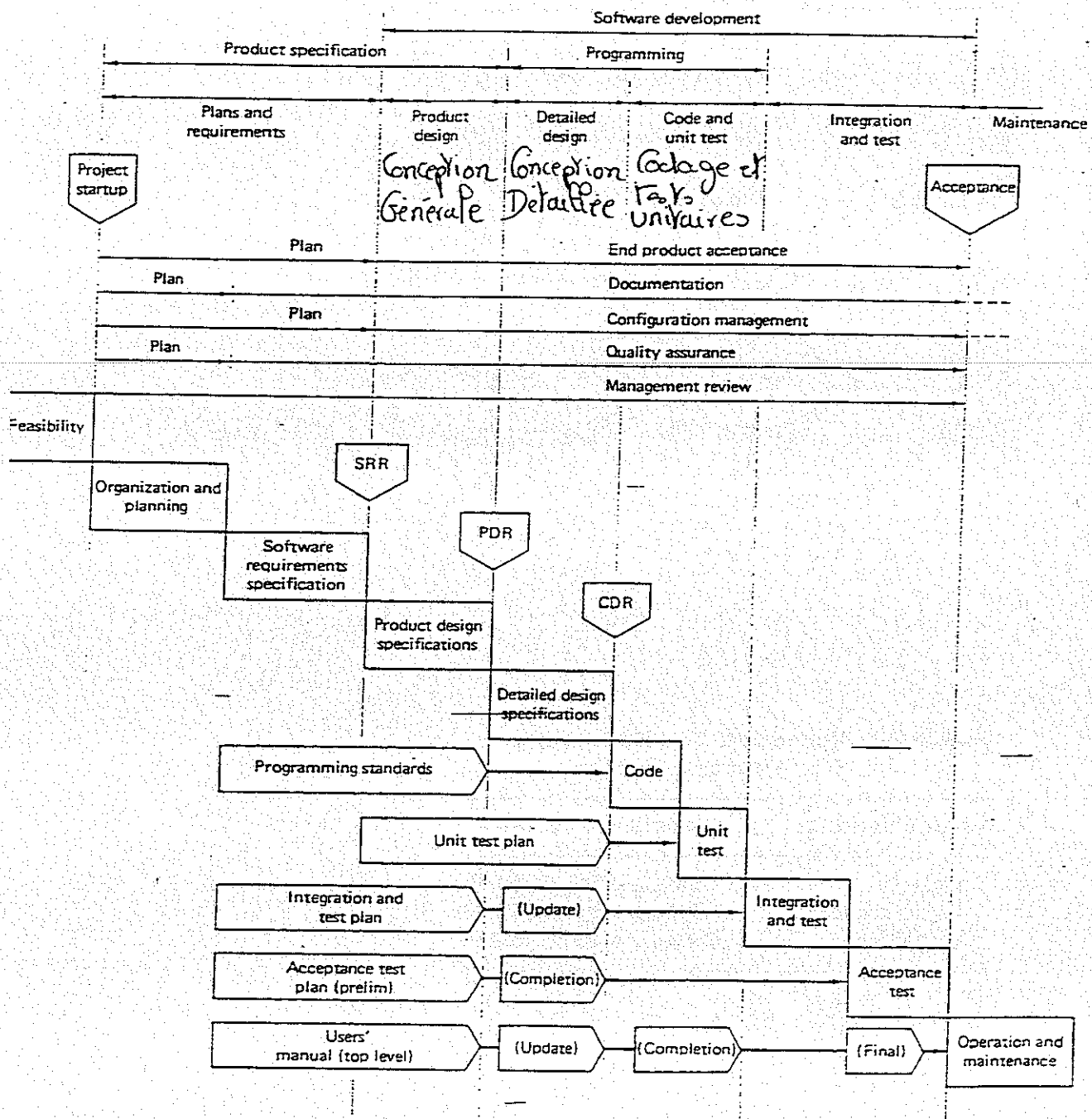


TABLE 6-1 Basic COCOMO Effort and Schedule Equations

Mode	Effort	Schedule
Organic	$MM = 2.4(KDSI)^{1.05}$	$TDEV = 2.5(MM)^{0.38}$
Semidetached	$MM = 3.0(KDSI)^{1.12}$	$TDEV = 2.5(MM)^{0.25}$
Embedded	$MM = 3.6(KDSI)^{1.20}$	$TDEV = 2.5(MM)^{0.32}$

Organic → programmes de type S

Semidetached → programmes de type P

Embedded → programmes de type E

MM = Man Month = Homme Mois

KDSI → KLS (kilo lignes Source)

TDEV → Temps de Developpement (en mois)

TABLE 8-1 Intermediate COCOMO Nominal Effort Estimating Equations

Development Mode	Nominal Effort Equation
Organic	$(MM)_{NOM} = 3.2 (KDSI)^{1.05}$
Semidetached	$(MM)_{NOM} = 3.0 (KDSI)^{1.12}$
Embedded	$(MM)_{NOM} = 2.8 (KDSI)^{1.20}$

(voir tables 8-2 et 8-3)

TABLE 5-1 Overall Project Profiles: Organic-Mode Projects

Product Size	Effort	Productivity	Schedule (months) (Durée)	Average Staffing (Taille de l'équipe)
Small: 2 KDSI	5.0 MM	400 DSI/MM	4.6	1.1 FSP
Intermediate: 8 KDSI	21.3 MM	376 DSI/MM	8.0	2.7 FSP
Medium: 32 KDSI	91.0 MM	352 DSI/MM	14.0	6.5 FSP
Large: 128 KDSI	392.0 MM	327 DSI/MM	24.0	16.0 FSP

TABLE 5-2 Phase Distribution of Effort and Schedule: Organic Mode

Phase	Product Size			
	Small (2 KDSI)	Intermediate (8 KDSI)	Medium (32 KDSI)	Large (128 KDSI)
<i>Effort</i>				
Plans and requirements	6%	6%	6%	6%
Product design	16	16	16	16
Programming	68	65	62	59
Detailed design	26	25	24	23
Code and unit test	42	40	38	36
Integration and test	16	19	22	25
Total	100%	100%	100%	100%
<i>Schedule (Durée)</i>				
Plans and requirements	10%	11%	12%	13%
Product design	19	19	19	19
Programming	63	59	55	51
Integration and test	18	22	26	30
Total	100%	100%	100%	100%

TABLE 5-3. Basic Project Profiles: Organic Mode

Quantity	Product size / Taille du produit			
	Small/petit (2 KDSI)	Intermediate (8 KDSI)	Medium/moyen (32 KDSI)	Large/grand (128 KDSI)
Total Effort (MM)	5.0	21.3	91	392
Plans and Requirements	0.3	1.3	5	24
Product Design	0.8	3.4	15	63
Programming	3.4	13.8	56	231
Detailed Design	1.3	5.3	22	90
Code and Unit Test	2.1	8.5	34	141
Integration and Test	0.8	4.1	20	98
Total Schedule (months)	4.6	8	14	24
Plans and Requirements	0.5	0.9	1.7	3.1
Product Design	0.9	1.5	2.7	4.6
Programming	2.9	4.7	7.7	12.2
Integration and Test	0.8	1.8	3.6	7.2
Average Personnel (FSP)				
Plans and Requirements	0.6	1.4	2.9	8
Product Design	0.9	2.3	5.6	14
Programming	1.2	2.9	7.3	19
Integration and Test	1.0	2.3	5.6	14
Project Average (FSP)	1.1	2.7	6.5	16
Percent of Project Average				
Plans and Requirements	60%	55%	50%	46%
Product Design	84	84	84	84
Programming	108	110	113	116
Integration and Test	89	87	85	83
Productivity (DSI/MM)	400	376	352	327

FSP = Full time Software Personnel = personnel à temps plein

TABLE 6-8 Phase Distribution of Effort and Schedule: All Modes

(0/0)

Effort distribution		Size				
Mode	Phase	Small 2 KDSI	Inter- mediate 8 KDSI	Medium 32 KDSI	Large 128 KDSI	Very Large 512 KDSI
Organic (S)	Plans and requirements (%)	6	6	6	6	
	Product design	16	16	16	16	
	Programming	68	65	62	59	
	Detailed design	26	25	24	23	
	Code and unit test	42	40	38	36	
	Integration and test	16	19	22	25	
Semidetached (P)	Plans and requirements (%)	7	7	7	7	7
	Product design (<i>Con. générale</i>)	17	17	17	17	17
	Programming	64	61	58	55	52
	Detailed design (<i>Con. détaillée</i>)	27	26	25	24	23
	Code and unit test (<i>Codage / test</i>)	37	35	33	31	29
	Integration and test (<i>Intégr. / tests</i>)	19	22	25	28	31
Embedded (E)	Plans and requirements (%)	8	8	8	8	8
	Product design	18	18	18	18	18
	Programming	60	57	54	51	48
	Detailed design	28	27	26	25	24
	Code and unit test	32	30	28	26	24
	Integration and test	22	25	28	31	34
Schedule distribution		2 KDSI	8 KDSI	32 KDSI	128 KDSI	512 KDSI
Organic (S)	Plans and requirements (%)	10	11	12	13	
	Product design	19	19	19	19	
	Programming	63	59	55	51	
	Integration and test	18	22	26	30	
Semidetached (P)	Plans and requirements (%)	16	18	20	22	24
	Product design	24	25	26	27	28
	Programming	56	52	48	44	40
	Integration and test	20	23	26	29	32
Embedded (E)	Plans and requirements (%)	24	28	32	36	40
	Product design	30	32	34	36	38
	Programming	48	44	40	36	32
	Integration and test	22	24	26	28	30

Phase	Plans and Requirements					Product Design					Programming					Integration and Test					Development					Maintenance				
Product Size	S	I	M	L	VI.	S	I	M	L	VI.	S	I	M	L	VI.	S	I	M	L	VI.	S	I	M	L	VI.					
(Overall Phase Percentage)	B	B	B	B	B	18	18	18	18	18	60	57	54	61	48	22	25	28	31	34	S <td>I<td>M<td>L<td>VI</td></td></td></td>	I <td>M<td>L<td>VI</td></td></td>	M <td>L<td>VI</td></td>	L <td>VI</td>	VI					
Activity percentage																														
Requirements analysis	50	48	46	44	42	10	10	10	10	10	3	3	3	3	3	2	2	2	2	2	4	4	4	4	4					
Product design	12	13	14	15	16	42	42	42	42	42	6	6	6	6	6	4	4	4	4	4	12	12	12	12	12					
Programming	2	4	6	8	10	10	11	12	13	14	55	55	55	55	55	32	36	40	44	48	42	43	43	44	45					
Test planning	2	3	4	5	6	4	5	6	7	8	4	5	6	7	8	3	3	4	4	5	4	4	5	6	7					
Verification and validation	6	7	8	9	10	6	7	8	9	10	8	9	10	11	12	30	28	25	23	20	12	13	14	14	14					
Project office	16	14	12	10	8	15	13	11	9	7	9	8	7	6	5	10	9	8	7	6	10	9	8	7	6					
CM/QA	5	4	4	4	3	4	3	3	3	2	8	7	7	7	6	10	9	9	9	8	8	7	7	7	6					
Manuals	7	7	6	5	5	9	9	8	7	7	7	7	6	5	5	9	9	8	7	7	8	8	7	6	6					
														</																

TABLE 7-2 Project Activity Distribution by Phase: Semidetached Mode

Phase	Plans and Implementations										Product Design										Programming										Integration and Test										Development										Maintenance									
	Product Size					Typical Plans Percentage					S	I	M	L	VI	S	I	M	L	VI	S	I	M	L	VI	S	I	M	L	VI	S	I	M	L	VI	S	I	M	L	VI																				
Actively percentage	40	47	46	45	44	12.5	12.5	12.5	12.5	12.5	4	4	4	4	4	2.5	2.5	2.5	2.5	2.5	5	5	5	5	5	13	13	13	13	13	12	12	12	12	12	12	12	12	12	12																				
	Product design	18	16.5	17	17.5	18	41	41	41	41	41	8	8	8	8	8	5	5	5	5	5	13	13	13	13	13	12	12	12	12	12	12	12	12	12	12	12	12	12																					
	Programming	2.5	3.5	4.5	5.5	6.5	12	12.5	13	13.5	14	68.5	58.5	56.5	56.5	56.5	33	35	37	39	41	45	45	44.5	44.5	44.5	41.5	41.5	41	41	41	41	41	41	41	41	41	41	41																					
	Test planning	2.5	3	3.5	4	4.5	4.5	5	5.5	6	6.5	4	4.5	5	5.5	6	2.5	2.5	3	3	3.5	4	4	4.5	5	5.5	3	3	3.5	4	4.5	3	3	3.5	4	4.5	3	3																						
	Verification and validation	6	8.5	7	7.5	8	6	6.5	7	7.5	8	7	7.5	8	8.5	9	32	31	29.5	28.5	27	11	12	13	13.5	14	11	12	13	13.5	14	11	12	13	13.5	14	11	12																						
Project office	15.5	14.5	13.5	12.5	11.5	13	12	11	10	9	7.5	7	6.5	6	6.5	8.5	8	7.5	7	6.5	6.5	6	6	5.5	5	5	6	6	6	6	6	6	6	6	6	6	6	6																						
	CMM/CA	3.5	3	3	3	2.5	3	2.5	2.5	2	7	6.5	6.5	6.5	6	0.5	0	0	0	7.5	6.5	6	6	5.5	5	5	6	6	6	6	6	6	6	6	6	6	6	6																						
Manuals	6	6	5.5	5	5	8	8	7.5	7	7	6	6	5.5	5	5	8	8	7.5	7	7	7	7	6.6	6	6	11	11	10.5	10.5	10.5	10.5	10.5	10.5	10.5	10.5	10.5	10.5	10.5																						

TABLE 7-3 Project Activity Distribution by Phase: Organic Mode

Phase	Plans and Requirements			Product Design			Programming			Integration and Test			Development			Maintenance		
	S	I	M L	S	I	M L	S	I	M L	S	I	M L	S	I	M L	S	I	M L
Product Size																		
Overall Phase Percentage		6				16		68	65	62	59	16	19	22	25			
Activity percentage																		
1 Requirements analysis		46				15		5		3				6			7	
2 Product design		20				40		10		6				14			13	
3 Programming		3				14		58		34				48	47	46	45	45
4 Test planning		3				5		4		2				4			3	
5 Verification and validation		6				6		6		34				10	11	12	13	10
6 Project office		15				11		6		7				7			7	
7 CM/QA		2				2		6		7				5			5	
8 Manuals		5				7		5		7				6			10	

TABLE 0-2 Software Development Effort Multipliers

Facteurs de Coûts	Ratings / Niveaux					
	Very Low	Low	Normal	High	Very High	Extra High
Product Attributes						
RELY Required software reliability						
DATA Data base size	.75	.88	1.00	1.15	1.40	
CPLX Product complexity	.70	.85	1.00	1.15	1.30	1.65
Computer Attributes						
TIME Execution time constraint						
STOR Main storage constraint			1.00	1.11	1.30	1.66
VIPT Virtual machine volatility*		.87	1.00	1.06	1.21	1.56
TURN Computer turnaround time		.87	1.00	1.15	1.30	
Personnel Attributes						
ACAP Analyst capability	1.46	1.19	1.00	.86	.71	
AEXP Applications experience	1.29	1.13	1.00	.91	.82	
PCAP Programmer capability	1.42	1.17	1.00	.86	.70	
VEXP Virtual machine experience*	1.21	1.10	1.00	.90		
LEXP Programming language experience	1.14	1.07	1.00	.95		
Project Attributes						
MODP Use of modern programming practices	1.24	1.10	1.00	.91	.82	
TOOL Use of software tools	1.24	1.10	1.00	.91	.83	
SCED Required development schedule	1.23	1.08	1.00	1.04	1.10	

* For a given software product, the underlying virtual machine is the complex of hardware and software (OS, DBMS, etc.) it calls on to accomplish its tasks.

TABLE 0-3 Software Cost Driver Ratings

Cost Driver	Ratings / Niveaux				
	Very Low	Low	Nominal	High	Very High
Product attributes					
RELY	Effect: slight inconvenience	Low, easily recoverable losses DB bytes < 10 Prog. DSI	Moderate, recoverable losses $10 \leq \frac{D}{P} < 100$	High financial loss $100 \leq \frac{D}{P} < 1000$	Risk to human life $\frac{D}{P} \geq 1000$
DATA					
CPLX	See Table 0-4				
Computer attributes					
TIME			$\leq 50\%$ use of available execution time	70%	85%
STOR			$\leq 50\%$ use of available storage	70%	85%
VIRT		Major change every 12 months Minor: 1 month	Major: 6 months Minor: 2 weeks	Major: 2 months Minor: 1 week	Major: 2 weeks Minor: 2 days
TURN		Interactive	Average turnaround < 4 hours	4-12 hours	> 12 hours
Personnel attributes					
ACAP	15th percentile ^a	35th percentile	55th percentile	75th percentile	90th percentile
AEXP	≤ 4 months experience	1 year	3 years	6 years	12 years
PCAP	15th percentile ^a	35th percentile	55th percentile	75th percentile	90th percentile
VEXP	≤ 1 month experience	4 months	1 year	3 years	
IEXP	≤ 1 month experience	4 months	1 year	3 years	
Project attributes					
MODP	No use	Beginning use	Some use	General use	Routine use
TOOL	Basic microprocessor tools	Basic mini tools	Basic mid-level tools	Strong max-level programming, test tools	Add requirements, design, management, documentation tools
SCED	75% of nominal	85%	100%	130%	160%

^a Team rating criteria: analysis (programming) ability, efficiency, ability to communicate and cooperate

Règles d'emploi du modèle COCOMO

Voir en annexe les paramètres du modèle ainsi que les problèmes et les corrigés.

1. Équations de base

Les équations de base du modèle relient entre elles les 3 quantités:

- Homme-Mois (MM)
- Kilo-ligne-source (KDSI)
- Durée de développement (TDEV)

Une quelconque de ces 3 quantités permet de déterminer les 2 autres, soit par les fonctions directes, soit par les fonctions inverses.

NB : l'inverse d'une fonction

$$y = a \times (x)^b$$

est :

$$\text{Log} \frac{y}{a} = b \times \text{Log}(x) \text{ soit : } x = e^{\frac{1}{b} \times \text{Log} \frac{y}{a}}$$

Généralement, une application est un assemblage de programmes de type S, P, et E. qui interviennent respectivement dans des proportions p, q, r telles que :

$$p + q + r = 1$$

L'effort, pour une application composite, est donc une moyenne pondérée (c'est un barycentre), soit :

$$\text{Effort moyen pondéré} = p \times \text{Eff}(S) + q \times \text{Eff}(P) + r \times \text{Eff}(E)$$

en prenant comme valeurs de *Eff* celles correspondant respectivement aux 3 équations S, P, E appliquées à la totalité des lignes source.

L'utilisation du modèle sur des parties d'application revient à négliger l'effort d'intégration de ces parties. C'est donc une erreur. Voir problème Juin 95.

La table 5-1 donne des profils types à partir desquels on peut faire des extrapolations linéaires.

À titre d'exercice vous pourrez comparer les résultats obtenus par extrapolation avec ceux fournis par les équations.

2. La ventilation par phase

Pour des commodités de calcul, le modèle fait l'hypothèse que les phases (voir schéma page 1) n'ont aucun recouvrement.

Les valeurs correspondantes en effort et durée sont décrites dans les tables :

5-2, 5-3, 6-8.

Sur ces tables, il convient de remarquer que la somme :

$$\text{Product Design} + \text{Programming} + \text{Intégration and Test} = 100\%$$

La partie *Plans and requirements* est comptabilisée comme une fraction (ou un pourcentage) de ce décompte et recalculée sur une base 100% (c'est comme un prix HT auquel on applique la TVA pour trouver le prix TTC).

Si l'on donne un coût total incluant les plans, et que l'on demande de calculer le nombre de KLS, il faut donc d'abord retrancher le coût des plans, avant d'appliquer les équations (voir exercice Avril 96).

Dans la réalité il y a recouvrement; c'est ce qui est pris en compte dans les tables :

7-1, 7-2, 7-3

qui donne la ventilation par phase et par période, pour une activité donnée.

Si l'on prend la table 7-3 on voit que l'activité de programmation totale provient de:

3% des Plans, 14% de Product Design, 58% de Programming, 34% de Integration-Test.

Pour chiffrer une activité particulière, il faut donc d'abord calculer la répartition par phase (au sens COCOMO) puis appliquer les clés de répartition des tableaux 7-1, 7-2, 7-3. On peut bien sûr procéder de façon inverse.

3. Le modèle COCOMO intermédiaire

C'est un raffinement du modèle de base. Il permet de prendre en compte des paramètres (appelés Cost Drivers, c'est à dire facteurs de coût) qui caractérisent:

- La complexité de l'application et/ou du produit à réaliser (*Product attributes*),
- Le qualité de service du centre de calcul et de l'ordinateur cible (*Computer attributes*),
- L'expérience et/ou maturité des programmeurs, individuellement et collectivement (*Personnel attributes*),
- Le processus et/ou la méthode de développement adoptés pour le projet (*Project attributes*).

On part cette fois des équations données par le tableau 8-1 (page 2) que l'on corrige à l'aide d'un paramètre k , calculé à partir du tableau 8-2.

$$\text{Soit : } Eff_{r\acute{e}l} = k \times Eff_{nominal}$$

Exemple :

Soit la combinaison de facteurs suivante :

Facteurs de coût		Niveau	Remarques
Produit	RELY	1.15	L'application a des contraintes de fiabilité.
	DATA	0.94	Pas de base de données.
	CPLX	1.15	Les algorithmes à utiliser sont complexes.
Ordinateur cible	TIME	1.11	—
	STOR	1.00	
	VIRT	1.00	
	TURN	1.00	
Expérience du personnel	ACAP	0.86	L'architecte est bon.
	AEXP	1.13	Il n'a pas une forte expérience de ce type d'application.
	PCAP	1.17	Les programmeurs sont débutants.
	VEXP	0.90	Ils connaissent bien le modèle de programmation.
	LEXP	1.07	Le langage de programmation est nouveau pour eux.
Projet	MODP	0.91	Ils connaissent bien les méthodes de programmation.
	TOOL	0.91	Ils connaissent bien les outils de développement.
	SCED	1.10	Le planning de réalisation est très tendu.
Valeur de k :		1.38	

L'équation d'effort résultant de la rectification est donc

$$S \quad MM = 1.38 \times 3.2(KDSI)^{1.05} = 4.16(KDSI)^{1.05}$$

$$P \quad MM = 1.38 \times 3.0(KDSI)^{1.12} = 4.14(KDSI)^{1.12}$$

$$E \quad MM = 1.38 \times 2.8(KDSI)^{1.20} = 3.86(KDSI)^{1.20}$$

En comparant ces équations à celles du tableau 6-1, on voit que nous sommes assez proche de l'équation standard de type E; les facteurs *Produit* laissent penser qu'il pourrait effectivement s'agir d'une application temps réel.

Chaque combinaison de facteur donne naissance à une équation particulière. Pour déterminer le niveau du facteur, on peut s'aider du tableau 8-2. Dans la pratique, il faut être particulièrement lucide sur le maniement de ce tableau. Un chef de projet optimiste pourrait très facilement sous-estimer la difficulté du produit, tout en surestimant la capacité de son personnel, ce qui conduirait immédiatement à une sous-estimation très importante de l'effort. On peut s'en rendre compte en multipliant les termes extrêmes du tableau 8-2.

Le calcul de TDEV et la répartition par phase restent les mêmes.

EXAMEN DE GENIE LOGICIEL

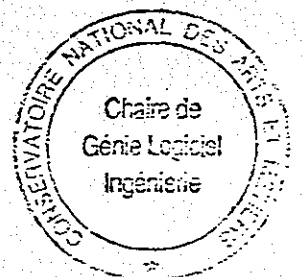
1ère session du 26 juin 1995

Ci-joint

. Question de cours

Annexe :

Les paramètres du modèle COCOMO



QUESTION DE COURS

Le Président de la société FMP (Fabrique de Moteurs Parisiens), séduit par les offres de matériels en client-serveur, a décidé de faire migrer l'application de gestion de production (GP) de son usine de Courbevoie. Cette application tournait sur un vieux main-frame acheté il y a 3 ans.

L'usine dispose déjà d'un réseau local utilisé par des applications de bureautique; la jeune équipe réseau, très compétente, est parfois un peu optimiste.

Les nouveaux matériels ont les mêmes langages de programmation et les mêmes SGBD que l'ancien matériel. Apparemment, l'offre de prix paraît particulièrement séduisante.

Pris d'un doute, le Président, avant de prendre sa décision, décide de confier l'analyse du dossier de migration à un architecte conseil, de façon à disposer d'un avis indépendant de celui de son directeur informatique.

L'étude du dossier fait apparaître les éléments suivants:

L'équipe de développement-maintenance oscille entre 3 et 4 personnes. L'application GP est un peu ancienne car le code n'a pas subi d'évolutions majeures depuis quelques années. Seules les erreurs résiduelles ont été corrigées, il y a eu quelques modifications pour connecter de nouveaux terminaux. L'application GP, bien qu'un peu lente, n'a pas de problème de performance grave; elle est administrée par l'opérateur du main-frame qui connaît bien tous les messages qu'elle émet. L'application GP n'est pas gérée par un AGL; un dictionnaire de données (gestion de configuration) a été utilisé à l'époque du développement initial, mais il n'a pas été mis à jour depuis. Les tests de l'application ont disparu (il s'agit de tests manuels).

Configuration du main-frame:

