

# GESTION DE PROJETS



Hiver 2017

Réal Gendron

## **Table des matières**

Outils et techniques pour la gestion de projets ..... 7 pages

PERT ..... 2 pages

Diagramme de GANTT ..... 3 pages

Gestion du développement des systèmes d'information ..... 14 pages

# Outils et techniques pour la gestion de projets

Il existe plusieurs techniques pour assister la planification et le contrôle de projet. Parmi celles-ci nous verrons plus particulièrement PERT, CPM et GANTT.

## PERT et CPM

La méthode de programmation optimale est mieux connue sous le nom de **PERT** (**P**rogram **E**valuation **R**evue **T**echnique). Elle a été développée en 1958 pour le projet de missiles Polaris de la Marine américaine (projet complexe impliquant plus de 3000 contractants). C'est à la firme d'experts conseils Booz, Allen & Hamilton, Lockheed Aircraft et un groupe de spécialistes de la Marine que l'on doit son développement.

La méthode du chemin critique ou **CPM** (**C**ritical **P**ath **M**ethod) a été mise au point à peu près au même moment par la compagnie Dupont. Elle a été développée pour contrôler et planifier la maintenance des usines chimiques et subséquemment des fonctions d'ingénierie.

PERT et CPM sont des techniques de réseaux temps-événement qui mettent en évidence grâce à des graphiques, les relations de dépendances logiques entre les activités nécessaires à la réalisation d'un projet. Ces techniques permettent :

- 1- de planifier efficacement l'utilisation des ressources du projet (personne, équipement, local ou tout ce qui sert à réaliser un projet) ;
- 2- de veiller à ce que les objectifs du projet se réalisent dans les délais et selon les coûts prévus;
- 3- d'administrer les projets non répétitifs ;
- 4- d'administrer les différentes variables d'un projet et de réagir rapidement aux facteurs qui peuvent influencer les délais et les coûts ;
- 5- d'évaluer la main d'œuvre, les matériaux et l'équipement requis.

**Construction d'un diagramme PERT ou CPM** (appelé dans Microsoft Project, **Réseau de tâches** [anciennement **organigramme des tâches**])

### Étape 1 Analyse des tâches

Préparez une liste de toutes les activités ou tâches du projet. L'activité est un élément de travail à exécuter afin de mener le projet à bonne fin. La liste est habituellement présentée

sous forme d'un tableau où l'on retrouve entre autres, pour chaque activité, un code ou numéro, une description, des prédécesseurs pour indiquer l'ordre à respecter pour son exécution par rapport à d'autres activités et une durée.

### Exemple

Projet : analyse préalable

Étape 1 Liste des activités

Code ou Numéro	Description	Prédécesseurs	Durée (m)
A	Prendre connaissance de la demande	-	2
B	Examiner les systèmes impliqués	A	4
C	Confirmer/prioriser les buts, contraintes et bénéfices	A	6
D	Évaluer l'ampleur du projet	A	3
E	Identifier les facteurs de risque	B, C, D	3
F	Établir la méthode de développement	C, D	9
G	Planifier l'analyse préliminaire	F	5
H	Réviser le travail effectué	E, G	2
I	Rédiger le rapport d'étude préalable	H	8
J	Présenter le rapport	I	2

Chaque activité reçoit un code ou un numéro de façon à la distinguer facilement des autres ; quand le projet compte peu d'activités, on peut utiliser les lettres de l'alphabet en commençant par « A » mais dans la plupart des cas on utilise des numéros. La description permet de donner un nom à l'activité afin de préciser ce qu'elle réalise ; étant donné qu'il s'agit d'une action, le nom de l'activité devrait être exprimé sous la forme d'un verbe à l'infinitif suivi d'un complément. Pour les prédécesseurs, on indique le code ou le numéro des activités qui doivent être complétées avant de débiter l'activité ; les prédécesseurs permettent de contrôler l'ordre d'exécution des activités du projet ; certains préfèrent travailler avec des successeurs.

L'estimation des durées des tâches différencie PERT et CPM. CPM utilise une méthode déterministe qui considère uniquement la durée la plus probable de l'activité (désignée par « m » par certains auteurs). PERT effectue une estimation probabiliste qui prend en considération trois facteurs pondérés comme le montre sa formule :

$$\text{durée} = (a + (4 * m) + b) / 6$$

« a » représente la durée la plus optimiste c'est-à-dire le temps minimum (probabilité de 1% du moins de temps que prend la tâche). « m » est la durée la plus probable ou qu'il faut le plus souvent (identique à celle utilisée par CPM). « b » constitue la durée la plus pessimiste (probabilité de 1% du plus de temps que prend la tâche).

Dans les deux cas, on recommande d'utiliser une seule unité de temps (heures ou jours habituellement) pour toutes les durées d'activités ; on peut aussi utiliser des fractions de l'unité comme par exemple 0,5 heure (1/2 heure).

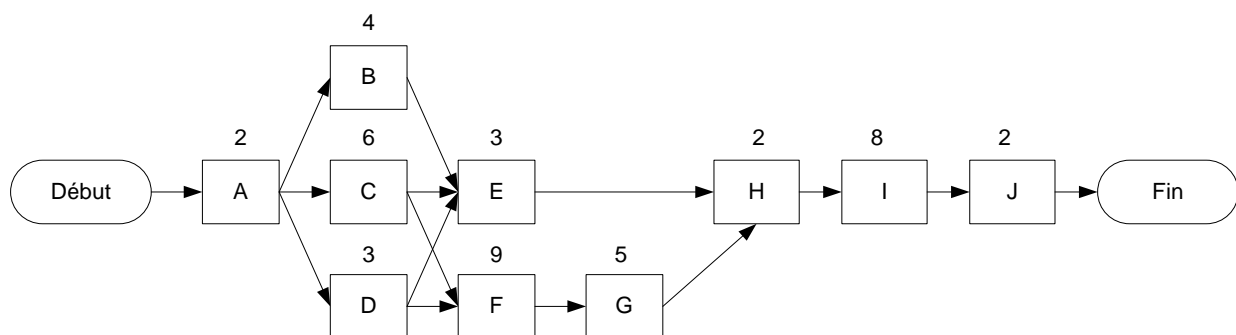
Code ou Numéro	a	m	b	Durée CPM = m	Durée PERT = $(a + (4 * m) + b) / 6$
A	1	2	3	2	$(1 + (4 * 2) + 3) / 6 = 2$
B	3	4	4	4	$(3 + (4 * 4) + 4) / 6 = 3,83$
C	4	6	7	6	$(4 + (4 * 6) + 7) / 6 = 5,83$
D	1	3	4	3	$(1 + (4 * 3) + 4) / 6 = 2,83$
E	2	3	4	3	$(2 + (4 * 3) + 4) / 6 = 3$
F	6	9	10	9	$(6 + (4 * 9) + 10) / 6 = 8,67$
G	4	5	5	5	$(4 + (4 * 5) + 5) / 6 = 4,83$
H	2	2	3	2	$(2 + (4 * 2) + 3) / 6 = 2,167$
I	6	8	10	8	$(6 + (4 * 8) + 10) / 6 = 8$
J	1	2	2	2	$(1 + (4 * 2) + 2) / 6 = 1,83$

## Étape 2 Construction du diagramme

Le graphique est constitué de boîtes (rectangles ou cercles) appelés « nœuds » et de lignes (arcs ou flèches) montrant les dépendances des activités (prédécesseurs). Le diagramme possède à l'extrémité gauche une étape appelée « Début » et à l'extrémité droite une étape « Fin ». Les activités sont reliées selon les relations de dépendance, de façon à former des chemins entre « Début » et « Fin ». Quand le nombre d'activités à représenter est élevé, on utilise des niveaux de diagramme en regroupant les activités par étapes (équivalent aux **tâches récapitulatives** dans Microsoft Project).

- La description de l'activité est inscrite dans le nœud qui prend souvent la forme d'un rectangle ou d'un cercle.
- Les lignes montrent seulement les séquences d'activités.
- On utilise parfois des rectangles aux coins arrondis ou des losanges comme nœud pour indiquer des étapes ou points de vérification du projet (aussi appelé « **jalón** »); le jalón correspond en fait à une tâche de durée 0 (zéro) qui permet de repérer un moment précis dans un projet (par exemple, début, fin, etc.)
- La durée est indiquée à l'intérieur ou en haut du nœud mais on peut retrouver aussi d'autres informations de l'activité (par exemple début au plus tôt, etc.)

## Exemple



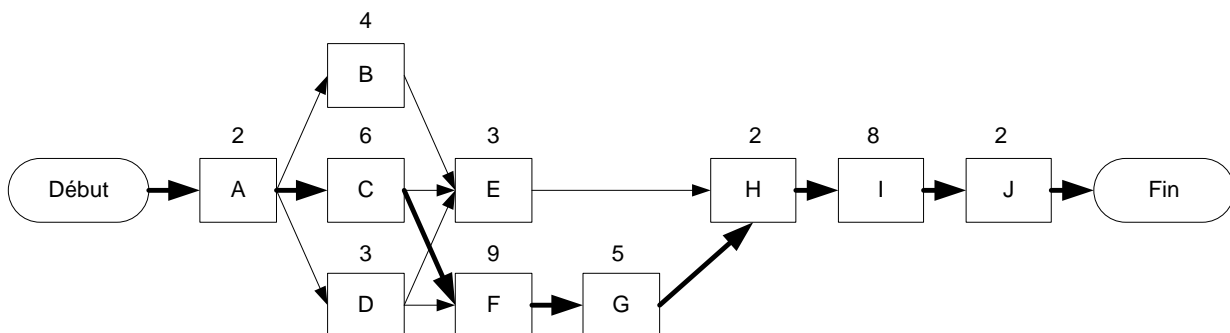
Le diagramme montre les activités qui se déroulent en parallèle; ce parallélisme peut ainsi aider à réduire la durée du projet.

### Étape 3 Identification du chemin critique

Le chemin critique est la séquence d'activités consécutives entre Début et Fin, qui présente la plus longue durée. Pour ce faire, il faut additionner la durée de toutes les activités d'un chemin et répéter l'opération pour chacun des chemins. Le plus grand total désigne le chemin critique qu'on indique sur le diagramme par des lignes en gras ou en couleur. Les activités de ce chemin ont une influence directe sur la durée du projet.

*Exemple (pour CPM)*

Chemin	Durée	Chemin critique?
A - B - E - H - I - J	$2 + 4 + 3 + 2 + 8 + 2 = 21$	
A - C - E - H - I - J	$2 + 6 + 3 + 2 + 8 + 2 = 23$	
A - C - F - G - H - I - J	$2 + 6 + 9 + 5 + 2 + 8 + 2 = 34$	oui
A - D - E - H - I - J	$2 + 3 + 3 + 2 + 8 + 2 = 20$	
A - D - F - G - H - I - J	$2 + 3 + 9 + 5 + 2 + 8 + 2 = 31$	



### Étape 4 Précision d'informations complémentaires

Quand on dispose d'un logiciel on peut parfois ajouter des informations supplémentaires aux activités. Outre les ressources affectées (personne, équipement, local ou tout ce qui sert à réaliser le projet et peut posséder un horaire) et leur coût, on peut préciser divers temps. Ainsi, pour chaque activité on peut connaître son début au plus tôt (ES = Earliest Start), son début au plus tard (LS = Lastest Start), sa fin au plus tôt (EF = Earliest Finish) et sa fin au plus tard (LF = Lastest Finish). Les équations qui suivent montrent les relations entre ces données et la durée de l'activité. Le calcul de ces temps doit tenir compte des prédécesseurs ou des successeurs.

$EF = ES + \text{Durée}$ $LS = LF - \text{Durée}$
---

*Exemple (pour CPM)*

Code ou Numéro	Durée (m)	Prédécesseurs	ES	EF = ES + Durée	LS = LF - Durée	LF	Ressources
A	2	-	0	0 + 2 = 2	2 - 2 = 0	2	Sam, Ada, Vic
B	4	A	2	2 + 4 = 6	19 - 4 = 15	19	Sam
C	6	A	2	2 + 6 = 8	8 - 6 = 2	8	Ada
D	3	A	2	2 + 3 = 5	8 - 3 = 5	8	Vic
E	3	B, C, D	8	8 + 3 = 11	22 - 3 = 19	22	Sam
F	9	C, D	8	8 + 9 = 17	17 - 9 = 8	17	Vic
G	5	F	17	17 + 5 = 22	22 - 5 = 17	22	Sam
H	2	E, G	22	22 + 2 = 24	24 - 2 = 22	24	Vic
I	8	H	24	24 + 8 = 32	32 - 8 = 24	32	Ada
J	2	I	32	32 + 2 = 34	34 - 2 = 32	34	Sam

Pour les activités directement reliées à « Fin », leur fin au plus tôt et leur fin au plus tard ont la même valeur que la durée du chemin critique.

Étape 5 Analyse du réseau

Afin d'aider le gestionnaire du projet, on peut déterminer le **temps de battement** (ou **jeu de temps** ou **marge**) (TS = time slack) à partir d'une des équations :

$$\boxed{TS = LS - ES \text{ ou } LF - EF}$$

Le temps de battement est l'écart de temps dont on dispose pour compléter la tâche sans occasionner de retard dans le projet (correspond à la **marge totale** dans Microsoft Project). Toutes les activités du chemin critique n'ont pas de jeu de temps (TS = 0). (Microsoft Project peut aussi calculer la **marge libre** qui est la période de temps pendant laquelle une tâche peut glisser avant qu'elle ne retarde une autre tâche).

*Exemple (pour CPM)*

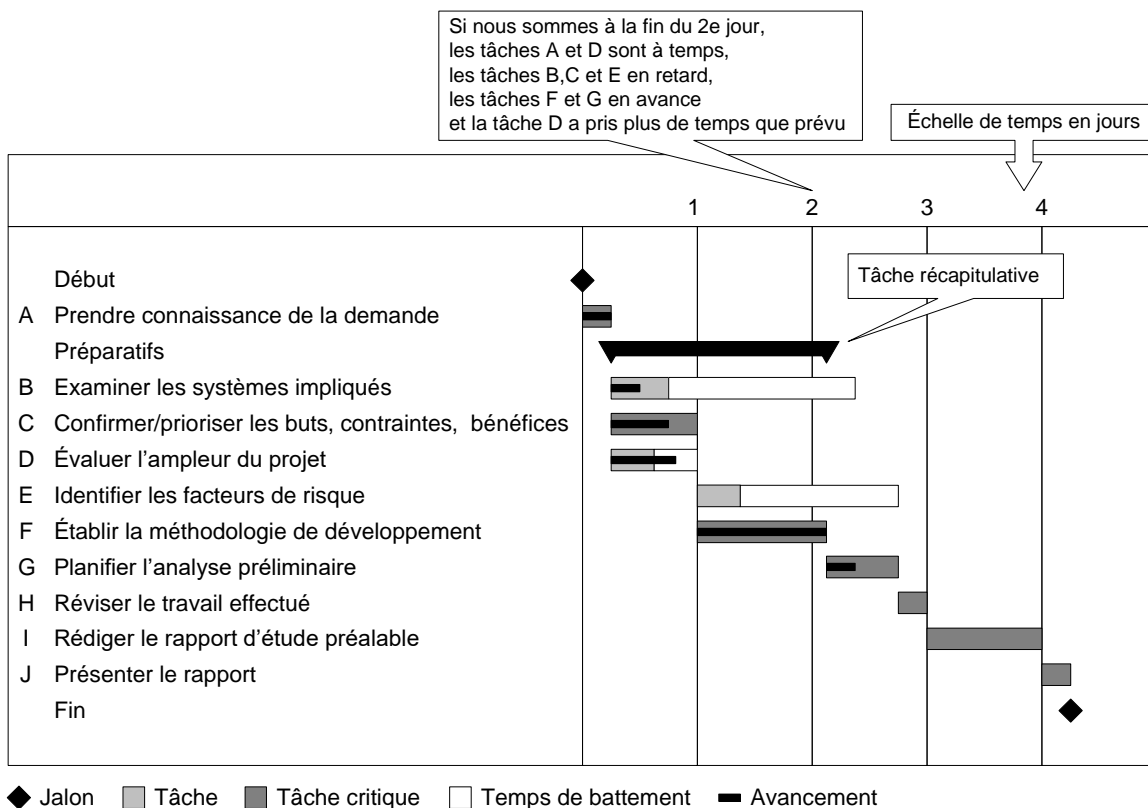
Code ou Numéro	ES	EF	LS	LF	TS = LS - ES = LF - EF
A	0	2	0	2	0 - 0 = 2 - 2 = 0
B	2	6	15	19	15 - 2 = 19 - 6 = 13
C	2	8	2	8	2 - 2 = 8 - 8 = 0
D	2	5	5	8	5 - 2 = 8 - 5 = 3
E	8	11	19	22	19 - 8 = 22 - 11 = 11
F	8	17	8	17	8 - 8 = 17 - 17 = 0
G	17	22	17	22	17 - 17 = 22 - 22 = 0
H	22	24	22	24	22 - 22 = 24 - 24 = 0
I	24	32	24	32	24 - 24 = 32 - 32 = 0
J	32	34	32	34	32 - 32 = 34 - 34 = 0

Lors du déroulement du projet, il est souvent nécessaire de reprendre des activités antérieures afin de régler les problèmes rencontrés. Grâce au diagramme, le gestionnaire peut alors voir l'effet d'un changement sur le projet.

## GANTT

Le diagramme de Gantt ou diagramme à barres est la méthode la plus simple de planification d'un projet. Mise au point par Henry L. Gantt, cette technique qui décèle la façon dont les activités d'un projet s'intègrent les unes aux autres en fonction du temps et des événements a servi à élaborer les réseaux temps-événement (PERT, CPM).

Comme PERT et CPM, la première étape consiste à identifier chaque activité et ensuite d'estimer le temps à leur consacrer. Après quoi, la liste des tâches est portée dans le diagramme, dans la partie gauche, de haut en bas, dans l'ordre dans lequel elles seront entreprises. Le temps-calendrier est indiqué de gauche à droite au haut du diagramme. Une barre horizontale sur l'échelle de temps à droite du nom de la tâche indique le moment où elle débute, le moment où il est prévu qu'elle soit terminée et du même coup sa durée. L'absence de barre signifie qu'aucun travail n'est associé à cette activité pendant cette période. Certaines tâches de durée nulle, appelées « **jalons** », étapes ou points de contrôle, sont représentées par un losange. D'autres tâches dites « **récapitulatives** » résument un groupe d'activités mais leur durée n'est pas la somme des durées de leurs tâches constitutives car il peut y avoir du parallélisme de tâches.



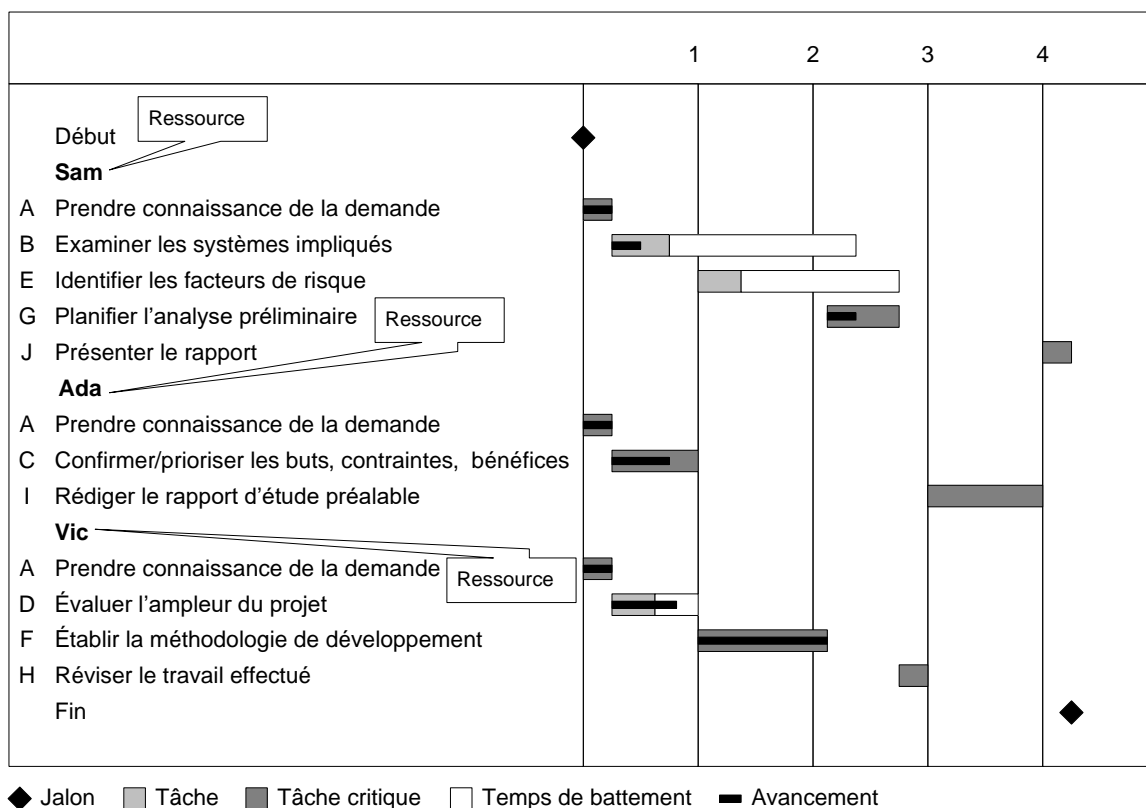
Le diagramme de Gantt peut représenter les mêmes informations que PERT et CPM. Ainsi, le temps estimé pour chaque activité peut être représenté par une barre. Au fur et à mesure que la tâche se complète, le degré de progression (avancement) est indiqué par un trait épais dans la barre. Dans les versions élaborées, on distingue visuellement



les barres des activités du chemin critique ; pour les autres activités, la barre comporte une partie indiquant le temps le plus tôt suivi d'une barre présentant le temps de battement (time slack).

Un coup d'œil sur le graphique de la page précédente nous renseigne sur les activités qui sont en avance, à temps ou en retard par rapport aux estimations. Cette technique facile à comprendre et à utiliser en raison de son format tableau, convient pour des projets présentant un nombre limité d'activités. Dans le cas contraire, comme leur taille devient excessive, on peut utiliser des tâches récapitulatives ou plusieurs niveaux de diagrammes allant du général au particulier. Comme PERT et CPM, le diagramme de Gantt montre le parallélisme des tâches. Malheureusement, il est difficile à maintenir et il indique mal l'interdépendance des activités.

Le diagramme de Gantt peut servir à d'autres usages en présentant d'autres informations relatives au projet. Ainsi, au lieu de montrer les tâches, on peut produire un diagramme regroupant les activités par ressource (personne ou machine, etc.) affectée à leur réalisation. Pour ce faire, il suffit de remplacer le nom des activités à gauche par le nom des ressources. De cette façon, on peut connaître les activités dont est responsable chaque ressource tout comme le temps où elle n'est pas affectée.



En conclusion, les techniques décrites précédemment montrent des aspects divers de la gestion d'un projet qui sont compatibles entre eux : disposant d'un de ces modèles (PERT, CPM ou GANTT), il est possible d'en dériver les autres.

# Diagramme de GANTT

## Présentation

Un projet comporte un nombre de tâches plus ou moins grand à réaliser dans les délais impartis et selon un agencement bien déterminé. Le diagramme de GANTT est un outil permettant de planifier le projet et de rendre plus simple le suivi de son avancement. Le diagramme de GANTT permet de visualiser facilement le déroulement du projet, ainsi que de prévoir suffisamment à l'avance les actions à penser ou à faire comme la commande de matériel qui prend parfois beaucoup de temps. On pourra aussi gérer plus facilement les conflits de ressources et les éventuels retards en visualisant l'impact de ceux-ci sur le déroulement du projet. En outre, le diagramme de GANTT est un bon outil de communication avec les différents acteurs du projet.

## Méthodologie

Le diagramme de GANTT est un planning présentant une liste de tâches en colonne et en abscisse l'échelle de temps retenue. La mise en œuvre de technique de planification nécessite que :

- les tâches soient identifiées
- les tâches soient quantifiées en terme de délais, de charges ou de ressources
- la logique de l'ensemble des tâches ait été analysée.

Ces éléments sont issus de l'analyse d'un projet, qui se situe en amont de la planification. Afin de mener à bien la réalisation de cet outil, il faut donc procéder comme suit :

### **1) Déterminer et structurer la liste des tâches à réaliser pour mener à bien le projet**

Cette identification peut se faire par des techniques comme le « brainstorming » ou les groupes nominaux. La liste obtenue doit être ensuite structurée : on tente de regrouper les tâches, de les hiérarchiser par lots de travail.

### **2) Estimer les durées et les ressources**

Il faut remplir un tableau présentant, pour chaque tâche, la durée de celle-ci et les ressources affectées. L'unité de temps pour exprimer la durée est fonction du type de projet réalisé. Elle peut aller de la minute (pour le déroulement d'un concert important), à l'année (pour les projets d'urbanisation). Le tout est d'utiliser la même unité de temps pour toutes les tâches dans un souci d'harmonisation du diagramme de GANTT. Quand aux ressources, elles peuvent être humaines ou matérielles. Il faut cependant faire attention à utiliser les mêmes types de ressources pour toutes les tâches.

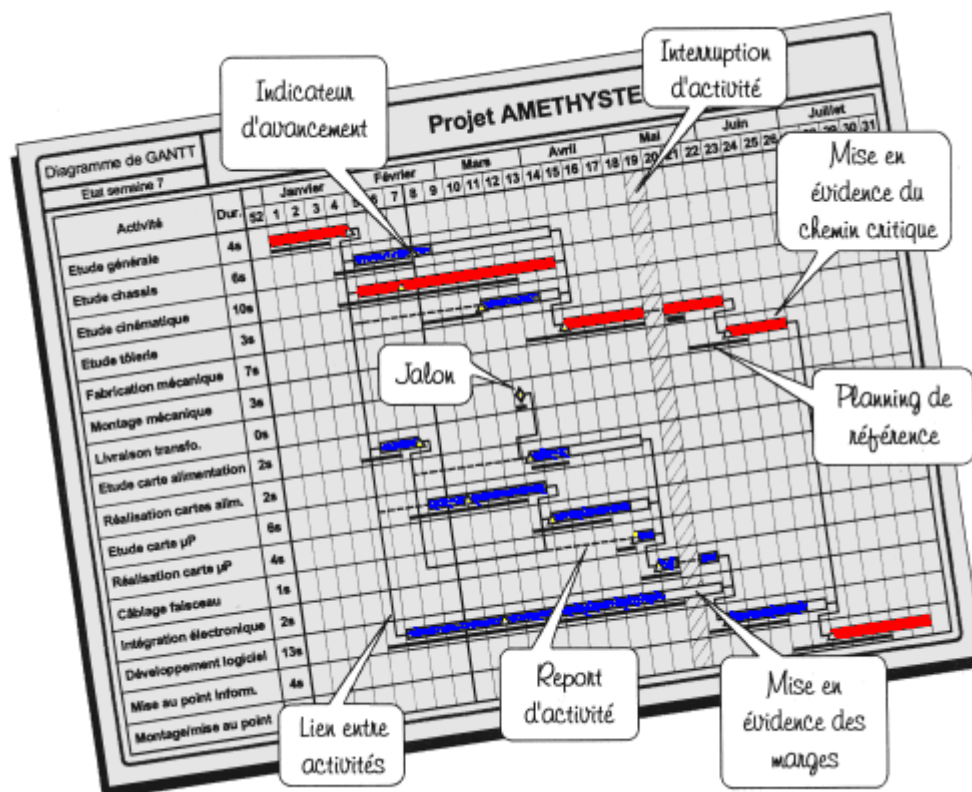


Image in <http://perso.wanadoo.fr/masscom/Pganttt.htm>

### 3) Réaliser le réseau logique

Le réseau doit reprendre les hypothèses de priorités des tâches. Il se présente souvent sous la forme de tâches reliées entre elles par des liens logiques. Pour chaque tâche, il est primordial de trouver les relations d'antécédence et de succession. Une fois le réseau tracé, on retrouvera la chronologie du projet.

### 4) Tracer le diagramme de GANTT

Avec en abscisse l'échelle de temps et en ordonnée la liste des tâches, il faut tracer un rectangle d'une longueur proportionnelle à sa durée, le tout suivant la logique d'ordre d'exécution du réseau. Il peut apparaître des tâches se réalisant pendant la même période. En effet, c'est un peu le but recherché car plus les tâches pourront se faire simultanément plus la durée du projet sera courte et plus le coût du projet risque de baisser. Il faut toutefois vérifier la disponibilité des ressources en regardant dans le tableau d'affectation, les ressources utilisées plusieurs fois.

D'après : <http://www.univ-lyon1.fr/qualite/cadregeneral.html>

# PERT

## Présentation

Il existe un certain nombre d'outils de planification, qui se différencient essentiellement par leur représentation graphique et leur type de liaisons entre tâches ou événements. L'outil PERT (Program of Evaluation and Review Technique) ou technique d'établissement et de remise à jour des programmes, créée en 1957 pour l'US Navy, n'utilise que des liaisons directes (fin, début) non explicitées, exprimées implicitement dans des étapes, et représentées graphiquement par un diagramme fléché.

L'outil des potentiels, créée en 1958 par M. B. Roy utilise systématiquement des relations d'ordre initiales (début, début) et fait partie des méthodes dites potentiels-tâches où les activités sont représentées par les sommets et les relations d'ordre entre activités successives par des liaisons. L'outil des antécédents a été créé ultérieurement pour compléter l'outil des potentiels et il prend en compte les liaisons fin-début.

## Comment ?

### ***Établir la liste des tâches***

Cette étape consiste à donner la liste exhaustive des tâches à exécuter :

- pour un premier planning indicatif, il faudrait probablement le décomposer en quelques dizaines de tâches ;
- pour établir un planning d'exécution détaillé, il faudrait sans doute pousser l'analyse au niveau de plusieurs centaines de tâches ;
- Les études importantes comportent parfois plusieurs milliers de tâches.

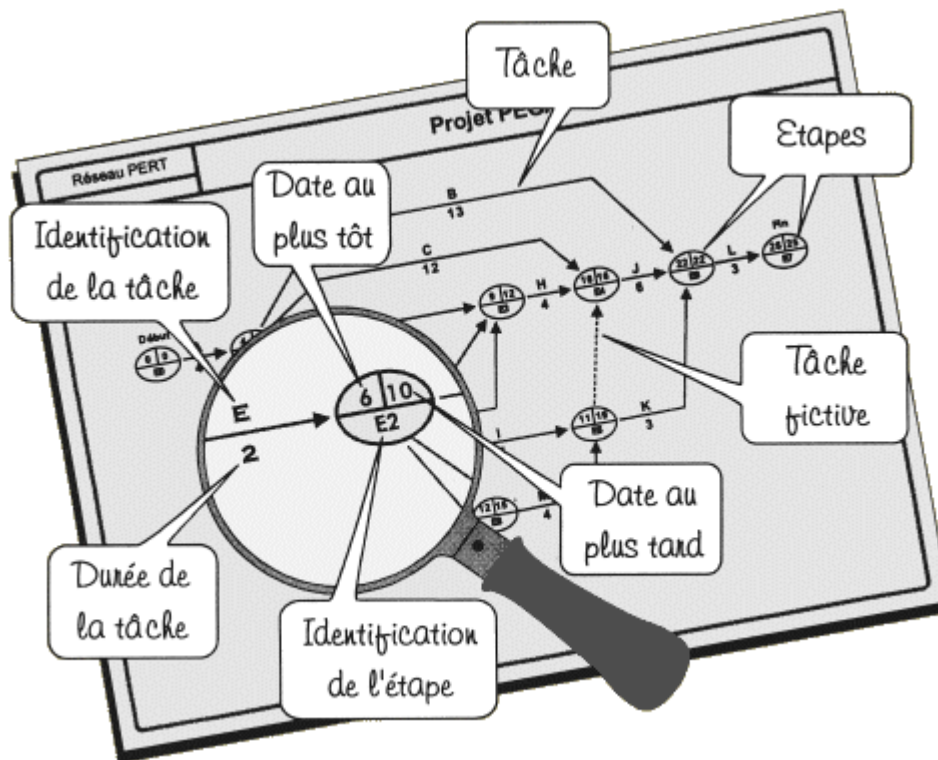
### ***Définir une tâche***

- pour définir correctement les tâches, certaines conditions sont à observer : une tâche décrit une action ou un événement, à entreprendre ou à subir. Son libellé doit être clair et précis.
- une tâche doit avoir des limites chronologiques bien définies. Pour cela, il faut pouvoir répondre aux deux questions suivantes :
  - Cette tâche, au vu de son énoncé, a-t-elle un commencement et une fin ?
  - Quels événements concrets (document, matériel ...) constituent son début et sa fin ?
- une tâche doit être associée à un responsable acceptant et assumant la responsabilité de l'exécution. À contrario, un événement peut être un point de

rencontre de plusieurs responsables. On entend par événement une tâche de durée nulle.

De manière pratique, il est conseillé :

- D'énoncer une tâche en commençant par un verbe à l'infinitif.
- De décrire la tâche en présence de son responsable.
- D'utiliser un support préétabli pour l'établissement de la liste des tâches.
- Dans certains contextes (tâches globales, de durée relativement longue car horizon de planification important : 5 à 10 ans), il est important d'associer à chaque activité une fiche descriptive. Elle comportera, outre les informations relevées sur le modèle ci-avant :
  - les caractéristiques des événements de début et de fin.
  - les intervenants.
  - les conditions d'exécution.



in <http://perso.wanadoo.fr/masscom/>

### Codifier les tâches

Pour passer à l'étape suivante (examen des contraintes d'enchaînement), il est nécessaire de repérer, de coder les tâches. Ceci va faciliter la construction du réseau.

Au début du projet, il est recommandé d'avoir recours à des codes simples.

En réalité, aucune règle ne peut être édictée, mais une codification faisant apparaître domaine et nature de la tâche est très appropriée. L'application du découpage de l'Organigramme Technique est un bon élément de génération des numéros de code.

Pour l'établissement de la liste des tâches, l'ordre selon lequel elles apparaissent n'a aucune importance. L'important est de ne pas en oublier. L'ordonnancement de celles-ci se fera dans l'étape suivante.

### ***Quel est le bon nombre de tâches à définir ?***

Ce nombre est bien évidemment fonction de l'ampleur du projet, mais avec un même nombre d'activités, on peut planifier :

- la construction d'une centrale nucléaire (7 à 8 ans) ou le changement d'outillage sur une machine (7 à 8 heures) ;
- la préparation sur 30 ans de sa retraite ou l'assemblage sur machine automatique de cinq composants en 30 secondes ;
- Diviser beaucoup conduit à éviter les oublis, à obtenir un scénario détaillé, à repérer toutes les contraintes, mais rend difficile le tracé du réseau et le suivi des activités par la suite. Peu diviser produit les effets inverses. Un juste équilibre est à trouver en fonction des objectifs visés : exécution, pilotage, appréciation de la durée globale ou optimisation de la durée. Ce qui est certain, c'est que l'effort de planification est proportionnel au nombre de tâches à programmer. Une bonne règle est la règle du 1/10ème :
  - si le projet s'étale sur 1 an, la tâche moyenne aura une durée de 1 mois.
  - si le projet s'étale sur 5 ans, la tâche moyenne aura une durée d'environ 6 mois.

Source : Dictionnaire de management de projet, AFITEP, AFNOR

# **GESTION DU DÉVELOPPEMENT DES SYSTÈMES D'INFORMATION**

ESTIMATION ET GESTION DU TEMPS DE DÉVELOPPEMENT .....	2
Estimation de temps .....	2
MÉTHODES D'ESTIMATION DE TEMPS .....	3
TEMPS-HEURES NÉCESSAIRE AU PROJET .....	4
Estimation du temps des diverses activités .....	4
Éléments variables du développement d'un programme .....	5
Calcul des estimations de temps de programmation .....	9
Diagrammes à barres .....	10
Diagrammes-jalons .....	11
Diagrammes de PERT .....	11

## QUESTIONS SOULEVÉES DANS CE CHAPITRE

- Comment les analystes estiment-ils le temps nécessaire au développement d'un système d'information ?
- Quels sont les facteurs qui déterminent la durée du développement ?
- Est-il possible d'estimer l'influence de l'expérience du personnel dans le temps de développement des programmes ?
- Quelles sont les méthodes de conduite de projet les plus courantes ?
- Comment le travail en équipe permet-il de mieux utiliser le potentiel du personnel impliqué dans un projet ?

La qualité de la conduite d'un projet est à la base de sa réussite. À l'inverse, on verra des projets tourner mal si le facteur temps n'a pas été correctement estimé, si les plans de développement ne tiennent pas compte de la nature critique de certaines tâches et si le personnel n'est pas utilisé de façon efficace et rationnelle. Ce chapitre expose des méthodes d'estimation et de gestion de la planification de projets. Il présente la formulation des estimations de temps de développement, leur intégration dans une planification et leur utilisation pour l'évaluation des performances.

## ESTIMATION ET GESTION DU TEMPS DE DÉVELOPPEMENT

Un projet mal planifié ne respectera pas les délais et décevra les utilisateurs les plus enthousiastes. Dans certaines entreprises, les retards sont courants, mais dans d'autres, ils sont exceptionnels. Les projets réalisés à temps présentent toutes les caractéristiques suivantes : (1) une estimation de temps formulée avec beaucoup de soin, (2) une méthodologie de suivi de l'avancement du projet, (3) des moyens permettant de comparer les performances réelles aux performances prévues et (4) suffisamment d'informations pour régler les problèmes éventuels.

### Estimation de temps

L'estimation et la formulation du temps nécessaire au développement d'un système constituent l'un des aspects les plus ardues de la conduite d'un projet. Les estimations, comme leur nom l'indique, sont des évaluations des heures, des jours ou des mois d'efforts nécessaires à la production du système désiré. Leur exactitude dépend pour beaucoup de la compétence et de l'expérience de la personne qui les prépare, généralement le chef de projet. Nous verrons que ces estimations dépendent de plusieurs facteurs : capacité des analystes et des programmeurs, complexité du système et interruptions qui ne sont pas directement liées au projet et qui de ce fait ne sont pas totalement maîtrisées par le chef de projet.



## MÉTHODES D'ESTIMATION DE TEMPS

La durée de développement d'un projet peut être estimée selon trois méthodes courantes. La méthode historique est basée sur des notes soigneusement tenues à l'occasion de développements précédents. Ces notes indiquent les caractéristiques du programme ou du projet, les affectations du personnel à des tâches, le temps qui est nécessaire pour les effectuer et les problèmes ou événements inusités. Le projet envisagé est comparé aux renseignements précédents pour estimer le temps de développement par comparaison. La mise à jour de ces notes demande beaucoup de temps. Cette méthode n'est valable que si tout est noté de façon rigoureuse, et ne s'avère utile que si l'ancien et le nouveau projet présentent les mêmes caractéristiques.

Rien ne vaut l'expérience. C'est donc sur l'expérience des responsables que repose la méthode intuitive. Ces derniers, s'appuyant sur leur expérience personnelle, estiment le temps de développement prévu. Bien entendu, l'expérience de l'individu responsable de l'estimation est primordiale. À l'inverse de la méthode historique, la méthode intuitive n'utilise pas directement de notes détaillées.

La méthode intuitive est la plus répandue. Toutefois, peu de personnes peuvent faire des estimations avec une grande précision. Son succès s'explique donc par le fait que c'est une méthode d'estimation rapide et facile.

La méthode de la formule standard aborde l'estimation d'une façon plus concrète. Les facteurs individuels qui influencent le plus le temps de développement sont identifiés et quantifiés. Des coefficients de pondération sont affectés à des facteurs tels que les caractéristiques du personnel, les détails du système et la complexité du projet. Une formule arithmétique combine les divers éléments pour produire une estimation du temps de développement en heures, jours ou semaines. La création de formules efficaces a fait l'objet de nombreuses tentatives, y compris de la part de plusieurs grands fabricants informatiques, à grand renfort de publicité. Mais, ces formules suscitent quelque peu de scepticisme. Nous élaborerons une estimation de programmation, à partir d'une formule IBM, plus loin dans cette section. Toutefois, de nombreuses entreprises ont leurs propres normes ou procédures pour estimer le temps de développement.

Les estimations de temps sont nécessaires pour que la direction sache quand le projet sera terminé et le système mis en oeuvre. De plus, ces estimations permettent au chef de projet de planifier l'affectation du personnel aux diverses tâches et de procéder aux ajustements nécessaires pendant le déroulement du projet. L'estimation du temps nécessaire à la réalisation d'un projet se présente sous deux aspects : le temps-heures et le temps-calendrier.

## TEMPS-HEURES NÉCESSAIRE AU PROJET

Le *temps-heures* est le temps demandé par l'ensemble des opérations : mener l'étude du système, formuler la conception externe, coder le logiciel et préparer les fichiers, créer les données d'essai, tester le logiciel, commander et installer le matériel. Autrement dit, chaque activité intervenant dans le développement demande un certain temps qu'il faut estimer et intégrer dans la planification du projet.

### Estimation du temps des diverses activités

Le temps d'analyse découle du nombre de personnes à interviewer et du temps nécessaire à la création, la distribution, la réception et l'analyse des données des questionnaires, aux observations et à l'étude des renseignements. Bien que l'analyste dépende des autres personnes (ceux qui sont interviewés ou qui doivent remplir et retourner les questionnaires), le temps d'analyse est, à bien des égards, plus facile à estimer que les autres activités.

La conception externe fait appel à la créativité de l'analyste et nécessite le développement de nombreux détails : définition des états, organisation des fichiers, validation de l'entrée, méthodes de contrôle et procédures d'exploitation. Ces activités sont plus difficiles à estimer que l'activité d'analyse.

Mais, le plus souvent, l'estimation la plus ardue concerne le temps d'écriture et de test des programmes. L'élaboration de la logique de programmation représente environ 35 % du temps de programmation total, tout comme les essais et la mise au point (figure 1). L'écriture du code proprement dit et le développement de données d'essai demandent 25 % du temps. Environ 5 % du temps est consacré à la documentation (IBM).

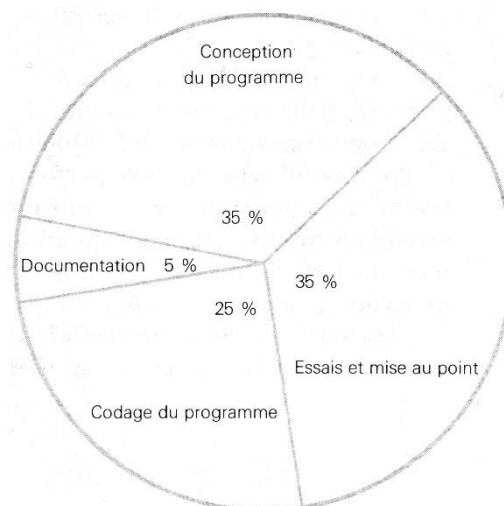


Figure 1 Répartition type du temps pour le développement du logiciel.

Cette estimation dépend de trois éléments principaux :

- 1 Niveau de compétence du programmeur
- 2 Niveau de complexité du programme
- 3 Niveau de compréhension du programme spécifique par le programmeur

La compétence et l'expérience du programmeur peuvent être très variables. Dans des projets de grande envergure, se côtoient des programmeurs en formation, un personnel expérimenté et des spécialistes chevronnés. Le chef de projet doit affecter les individus à des tâches spécifiques en fonction de leur niveau de compétence et estimer le temps qu'une tâche particulière demandera à chaque personne.

### **Éléments variables du développement d'un programme**

Certains chefs de projet utilisent des coefficients de pondération pour évaluer les talents d'un individu et les intégrer au temps. Voici certains de ces critères :

- 1 Connaissance du langage de programmation utilisé pour le projet ;
- 2 Expérience de l'ordinateur sur lequel le système fonctionnera ;
- 3 Expérience de programmation ;
- 4 Capacité logique ;
- 5 Créativité et imagination ;
- 6 Patience ;
- 7 Maturité ;
- 8 Persévérance ;
- 9 Formation.

Un coefficient est attribué à chaque individu, en principe entre 1 et 5, pour chacun des critères ci-dessus. Par exemple, un nouveau programmeur recevra 1 ou 2 en expérience de programmation, tandis qu'un programmeur très expérimenté recevra 5.

On peut supposer qu'un programmeur en formation a besoin pour développer un programme, de trois à quatre fois plus de temps qu'un programmeur ayant entre cinq et sept ans d'expérience. Par contre, un programmeur très expérimenté a besoin de 50 % à 75 % moins de temps qu'un programmeur de bon niveau.

La *complexité du programme* dépend du niveau des fonctions du système, telles que les méthodes d'entrée et de sortie, et de la complexité des programmes constituant le logiciel. Ainsi, un programme qui se contenter de lire un seul type de carte perforée, de mettre à jour un fichier séquentiel sur bande magnétique et d'imprimer la liste des transactions traitées est beaucoup moins complexe qu'un système en ligne qui reçoit des entrées diverses à partir de sites multiples. Fichiers multiples et traitement réparti ne feront qu'accroître cette complexité.

Les variables qui constituent la complexité d'un programme (modifiées à partir d'un système proposé par IBM) se présentent ainsi :

VARIABLE	COEFFICIENT
TYPE D'ENTRÉE	
● Une seule carte	1
● Cartes multiples	2
● Donnée simple, entrée directe	1
● Données multiples, entrée directe	2
● Entrée magnétique (bande, disque, MICR, etc. voir TYPE DE FICHIERS)	1
TYPE DE SORTIE	
● Imprimante, une ligne par enregistrement	1
● Imprimante, plusieurs lignes par enregistrement	2
● Affichage, écran unique, ligne unique	2
● Affichage, écran unique, lignes multiples	3
● Affichage, écrans multiples	4
● Cartes perforées, format unique	1
● Cartes perforées, formats multiples	2
● Sortie magnétique (voir TYPES DE FICHIERS)	
TYPES DE FICHIERS	
● Fichier séquentiel, type d'enregistrement unique	1
● Fichier séquentiel, type d'enregistrements multiples	2
● Fichier séquentiel, enregistrement de longueur variable	3
● Fichier séquentiel indexé, type d'enregistrement unique	2
● Fichier séquentiel indexé, type d'enregistrements multiples	3
● Fichier séquentiel virtuel	4
● Fichier en accès direct, type d'enregistrement unique	3
● Fichier en accès direct, type d'enregistrements multiples	4
LANGAGE DE PROGRAMMATION (voir TABLEAU 1)	

**TABLEAU 1 COEFFICIENTS DE PONDÉRATION DES ACTIVITÉS DE PROGRAMMATION (FOURNIS PAR INTERNATIONAL BUSINESS MACHINES CORPORATION)**

Système de programmation	Fonction	Coefficients			Limite <sup>1</sup>	
		Simple	Complexe	Très complexe	Min.	Max.
COBOL	● Restructuration des données	1	3	4		
	● Vérification de conditions	1	4	7		
	● Lecture et présentation des données	2	5	8		
	● Calcul	1	3	5		
	● Chaînage	1	2	3		
	<b>Total</b>	6	17	27	4	27
Langage assembleur	● Restructuration des données	4	5	6		
	● Vérification de conditions	4	7	9		
	● Lecture et présentation des données	4	7	9		
	● Calcul	3	5	8		
	● Chaînage	2	3	5		
	<b>Total</b>	17	27	37	12	37

Système de programmation	Fonction	Coefficients			Limite <sup>1</sup>	
		Simple	Complexe	Très complexe	Min.	Max.
Programmes utilitaires IBM (tri, fichier à fichier, Copie-restauration, etc.)	● Seules les cartes de contrôle changent	1	N/A	N/A		
	● Codage particulier nécessaire	2	3	4		
	<b>Total</b>	3	3	4		
RPG		2	8	13		
PL/1	● Restructuration des données <sup>2</sup>	1	2	4		
	● Vérification de conditions <sup>3</sup>	0,5	3	6		
	● Lecture et présentation des données <sup>4</sup>	0,5	4	8		
	● Calcul <sup>5</sup>	1	2	4		
	● Chaînage <sup>6</sup>	0,5	1	2		
	<b>Total</b>	5,5	20	37	2	24

1 Les limites représentent le coefficient minimum et maximum utilisable sur ces tableaux pour un programme unique. Il faut contrôler l'exactitude historique de ces coefficients pour des estimations plus précises.

2 Restructuration des données = Combinaison, compression, réagencement, suppression des données et mise à jour des fichiers.

3 Contrôle des conditions = Contrôles, tels que routines d'en-tête et de bas de page, test de vraisemblance, test de limite et routines d'erreur associées à ces procédures.

4 Lecture et présentation des données = Recherche d'enregistrements spécifique dans un fichier unique, recherche en table, techniques de nombres aléatoires pour accès aux enregistrements et indexation associée à ces activités.

5 Calcul = Opérations arithmétiques de tout type, à l'exclusion des étapes simples associées à d'autres catégories (par exemple, ajouter 1 à un compteur).

6 Chaînage = Routines de recouvrement de programmes, procédures de point de contrôle et de redémarrage et routines permettant l'interface avec un système de programmation ou avec un autre programme ou module de programme.

Pour un programme donné, un coefficient est attribué à chaque facteur, le nombre étant proportionnel au niveau de complexité. Généralement, on adopte une échelle de 1 à 5. La figure 2 démontre l'utilisation de ces coefficients pour un système avec fichiers séquentiels indexés et compte rendu d'erreurs.

CARACTÉRISTIQUES DU PROGRAMME	COEFFICIENT
FONCTIONS D'ENTRÉE	
Séquentiel indexé : un type d'enregistrement	2
Fichier séquentiel de détail sur bande magnétique	1
Fichier séquentiel de données exceptionnelles erronées sur bande magnétique	1
FONCTIONS DE SORTIE	
Listage de sortie en format simple	1
Listage de transactions en format simple	1
État des erreurs en format simple	1
	<u>7</u>
COMPLEXITÉ	
Langage COBOL : complexité moyenne	17
CARACTÉRISTIQUES DU PROGRAMMEUR	
Expérience de programmation	3,5
Connaissance du travail à exécuter	0,75
FACTEUR DE PERTE DE PRODUCTION	60%
RÉSULTAT DE LA FORMULE <sup>1</sup>	100,8 jours

1  $[(17 + 7) - (3,5 \times 0,75)] \times 1,60 = 100,8$  jours

Figure 2 Durée de développement d'un programme en utilisant une formule d'estimation

La connaissance des programmeurs vis-à-vis du système ou du programme qu'ils doivent développer peut se situer à plusieurs niveaux (IBM) :

- 1 Connaissance détaillée du travail demandé ;
- 2 Connaissance générale du travail demandé ;
- 3 Connaissance générale des sujets de même nature ;
- 4 Aucune connaissance générale des sujets de même nature.

À cela s'ajoute une autre pondération, selon le niveau de connaissance nécessaire à l'exécution du travail. Ainsi, tel individu ayant une excellente connaissance d'un travail dans un contexte qui se satisfait d'un niveau moyen sera évalué différemment que tel autre ayant une excellente connaissance d'un système pour lequel ce niveau de connaissance est nécessaire (tableau 2).

**TABLEAU 2 EXEMPLE DE CRITÈRES DE PRODUCTIVITÉ D'UN PROGRAMMEUR**

(a) Niveau d'expérience d'un programmeur (fourni par <i>International Business Machines Corporation</i> )			
NIVEAU	EXPÉRIENCE GÉNÉRALE DE PROGRAMMATION	PERSONNE/JOURS PAR PROGRAMME : COEFFICIENT DE PONDERATION	
Programmeur expérimenté	Expérience d'écriture et de mise au point de nombreux programmes sur divers types de matériel. Très expérimenté en configuration d'ordinateur et en systèmes de programmation.	0,50—0,75	
Programmeur	Expérience d'écriture et de mise au point de programmes de complexité diverse. Expérience d'un type particulier de configuration et de système de programmation.	1,00—1,50	
Débutant	A écrit et mis en œuvre plusieurs programmes. Expérience limitée des configurations et des systèmes de programmation particuliers.	2,00—3,00	
Novice	A suivi les cours de programmation. A écrit des programmes de formation. Expérience très limitée.	3,50—4,00	

(a) Tableau des connaissances du travail (fourni par <i>International Business Machines Corporation</i> )			
CONNAISSANCE DU TRAVAIL RÉEL	Connaissance du travail demandée		
	EXCELLENTE	MOYENNE	NULLE
Connaissance détaillée de ce travail	0,75	0,25	0,00
Connaissance générale de ce travail avec des connaissances détaillées fragmentaires	1,75	0,50	0,00
Bonne connaissance générale de ce travail, mais peu ou pas de connaissances détaillées	1,50	0,75	0,00
Pas de connaissance du travail mais connaissance générale de sujets de même nature	1,75	1,00	0,25
Pas de connaissance du travail et pas de connaissance générale de sujets de même nature	2,00	1,25	0,25

## Calcul des estimations de temps de programmation

Chaque programme doit être évalué séparément. Pour déterminer le nombre de jours nécessaires à chaque programme, le chef de projet doit combiner les données identifiées ci-dessus. Si une démarche quantitative stricte est adoptée, la complexité du programme est multipliée par la somme de l'expérience et de la compréhension du programmeur. Dans l'exemple de la figure 2, les caractéristiques du programme donnent un total de coefficients de 7. Le programme COBOL, jugé de complexité moyenne selon les critères du tableau 1, a un coefficient total de 17. Le programme lui-même nécessite une connaissance générale mais détaillée de la part du programmeur qui, dans cet exemple, est supposé être un débutant. En outre, on estime devoir ajouter 60 % au temps de programmation pure, en raison de réunions et de perte de temps. D'après la formule de la figure 2, on aboutit à une estimation de 100 jours pour le programme modèle.

Le risque d'erreurs d'estimation est toujours grand. Donc, certains dirigeants préfèrent utiliser la combinaison des trois estimations de temps de développement : le plus rapide, le plus long et le plus probable. Cette formule fournit une nouvelle estimation :

$$\text{Estimation} = \frac{\text{le plus pessimiste} + 4 \times (\text{le plus vraisemblable}) + \text{le plus optimiste}}{6}$$

Dans certaines entreprises, on exige l'utilisation d'une formule spécifique pour obtenir les estimations à soumettre à la direction. Chaque entreprise a sa propre politique en ce qui a trait aux procédures d'estimation.

## TEMPS-CALENDRIER

Il ne suffit pas d'identifier le temps requis pour un projet pour savoir le nombre de jours, de semaines ou de mois à intégrer dans la planification du projet. Ce n'est qu'un facteur. Dans la plupart des projets, il faut ajouter du temps supplémentaire pour diverses raisons : réunions de direction, revue de projet, éducation et formation, interaction avec les utilisateurs, maladie, congés et indisponibilité de l'ordinateur. Tout cela vient allonger les estimations précédentes. En fait, dans des projets d'envergure, le temps nécessaire à ces activités ajoute de 50 % à 100 % au temps de développement total. D'après ces critères, un projet estimé à 250 jours de temps de programmation représentera de 375 à 500 jours de projet.

Un *jour de projet* (parfois appelé jour-personne) permet de mesurer le nombre de jours nécessaires à un projet sur la base du travail effectué en un jour par un individu. Le nombre de personnes travaillant sur le projet affecte le temps-calendrier, mais pas proportionnellement. Le fait d'ajouter des personnes à un projet réduit le temps-calendrier total, mais il faut aussi prévoir le temps qui n'est pas consacré au projet.

Le temps-calendrier est généralement planifié selon trois méthodes diagrammes à barres, diagrammes-jalons et diagrammes de PERT.

## Diagrammes à barres

La méthode de planification la plus simple est celle des diagrammes à barres. Chaque activité du projet Y est représentée accompagnée de son temps de réalisation. Cette méthode, mise au point par Henry L. Gantt (ces diagrammes sont parfois appelés *diagrammes de Gantt*), indique le temps consacré à chaque tâche, sous forme de barres horizontales (figure 3). L'analyse identifie d'abord chaque tâche et estime le temps à lui consacrer. Après quoi, la liste des tâches est portée sur le diagramme, de haut en bas, à gauche du diagramme, dans l'ordre où elles seront entreprises. Le temps-calendrier est indiqué de gauche à droite. Une barre horizontale indique le moment où chaque tâche doit débuter et celui où il est prévu qu'elle soit terminée. Ainsi, la première tâche de la figure 3 devrait durer deux semaines. Trois activités distinctes, commençant la troisième semaine, nécessitent entre deux et quatre semaines chacune. L'absence de barres signifie qu'aucun travail n'est associé à cette activité pendant une certaine période.

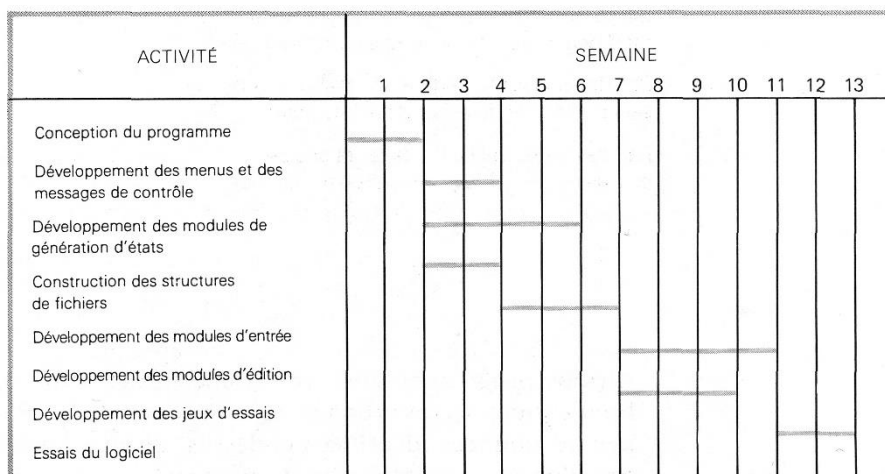


Figure 3 Diagramme à barres pour le plan de développement d'un système

Les diagrammes à barres conviennent pour des projets présentant un nombre limité de tâches ou d'activités. Sinon, leur taille devient excessive et les barres sont trop nombreuses pour qu'ils soient facilement utilisables.

On utilise souvent plusieurs niveaux de diagrammes pour présenter l'information de planification. Un diagramme général présente les activités principales. La personne responsable de la totalité du projet veut un diagramme plus général que, par exemple, le chef programmeur qui doit subdiviser chaque tâche principale en tâches plus petites, gérées séparément. Un diagramme à barres sera donc plus ou moins détaillé selon l'utilisation qui en est prévue.



## Diagrammes-jalons

Tous les projets sont parsemés d'événements, appelés *jalons*, qui représentent des stades de développement significatifs. Ce sont soit des obstacles à franchir, soit des tâches critiques qui doivent être réalisées dans un certain délai. Les diagrammes-jalons présentent les événements importants de réalisation d'un projet et l'ordre dans lequel ils doivent être réalisés. À la différence des diagrammes à barres, ils représentent des stades de réalisation, et non des tâches individuelles à entreprendre.

Voici par exemple quelques jalons importants dans la mise en oeuvre d'un système : arrivée du matériel, installation du matériel, formation des utilisateurs, conversion des fichiers et passage au nouveau système.

L'avancement d'un projet est mesuré en comparant l'état d'une activité à un jalon ou date prévue d'achèvement ; ainsi, le chef de projet peut respecter les délais. Cette méthode a pour inconvénient de concentrer l'attention sur le temps, plutôt que sur l'interdépendance des tâches et des événements ou le contrôle des coûts du projet. Il faut bien entendu que les jalons soient significatifs et mesurables. Des jalons tels que « interviews réalisées à 50 % », « écriture de la moitié du code », ou « analystes maîtrisant bien la conception externe » sont inutiles et doivent être évités.

## Diagrammes de PERT

La méthode PERT (*Program Evaluation and Review Technique*) est la méthode de planification la plus élaborée. Développée en 1958 par la U.S. Navy et une firme de consultants en gestion, Booze, Allen and Hamilton, cette méthode a souvent été utilisée dans des projets complexes nécessitant une planification et une supervision rigoureuses. Ainsi, pour mettre au point le système d'arme Polaris, il fallait à la U. S. Navy une méthode pour planifier et intégrer des milliers d'activités industrielles et scientifiques, une procédure pour déterminer l'avancement du projet et un moyen d'évaluer l'influence des changements sur l'exécution du projet. Le diagramme de PERT fut conçu à cet effet. Grâce à cette méthode, la U.S. Navy put mener à bien le projet Polaris avec deux ans d'avance, tandis que certains autres de ses projets, n'utilisant pas le diagramme de PERT, étaient en retard et dépassaient le budget prévu.

Bien que la meilleure méthode de gestion d'un projet consiste à le subdiviser en sections plus faciles à suivre, on risque toujours de superviser les tâches de niveau inférieur et de perdre de vue le projet global. Les activités d'un projet sont généralement interdépendantes. Cette interdépendance n'est pas très évidente dans les méthodes du diagramme à barres ou du diagramme-jalons. Les tâches critiques, qui doivent être réalisées à temps et dans un certain ordre, ne sont pas non plus très apparentes. Un chef de projet souhaite : indiquer à un individu les activités et le temps alloué à chacune, présenter l'interdépendance entre les activités, identifier la bonne séquence, donner une estimation de temps, isoler les domaines susceptibles de présenter des problèmes

ou des retards (et indiquer les domaines à suivre particulièrement de près) et pouvoir suivre l'avancement du projet. Il faut savoir par exemple quelle sera l'influence du retard d'une certaine activité sur l'ensemble du projet, ou par quelle activité il est le plus difficile d'envisager un retard. Des diagrammes de PERT fournissent cette information tandis que des diagrammes à barres n'impliquent que l'interdépendance des tâches.

Les projets sont constitués d'événements et d'activités. Le diagramme de PERT utilise des noeuds et des chemins pour représenter la relation entre les diverses activités du projet (figure 4). Les noeuds représentent des événements et les chemins montrent les activités nécessaires pour progresser d'un événement à un autre. Les nombres entre parenthèses indiquent le temps nécessaire à l'exécution de chaque activité. Dans un grand projet, ce maillage de lignes et de noeuds sera très étendu.

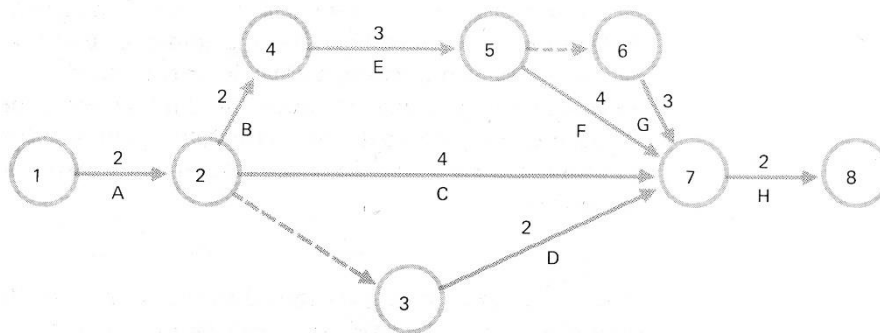


Figure 4 Modèle de diagramme de PERT pour le développement d'un système.

Le diagramme de PERT présente tout son intérêt au moment de la planification et de la conception d'un projet. La grille étant terminée, on l'étudie pour déterminer le chemin critique, c'est-à-dire le parcours du début à la fin du projet, sur lequel le temps total nécessaire sera le plus long (présenté en gras à la figure 5). Si les activités de ce chemin ne sont pas terminées à temps, tout le projet sera retardé. On doit donc porter une attention particulière à ces activités.

PERT illustre également l'interdépendance des tâches et permet de répondre à trois questions de gestion courante :

- 1 Quelles sont les autres activités qui doivent précéder ou être terminées avant de débiter une activité spécifique ?
- 2 Quelles sont les autres activités qui peuvent être réalisées pendant qu'une activité spécifique est en cours ?
- 3 Quelles sont les activités qui ne peuvent débiter qu'après l'exécution d'une activité spécifique ?

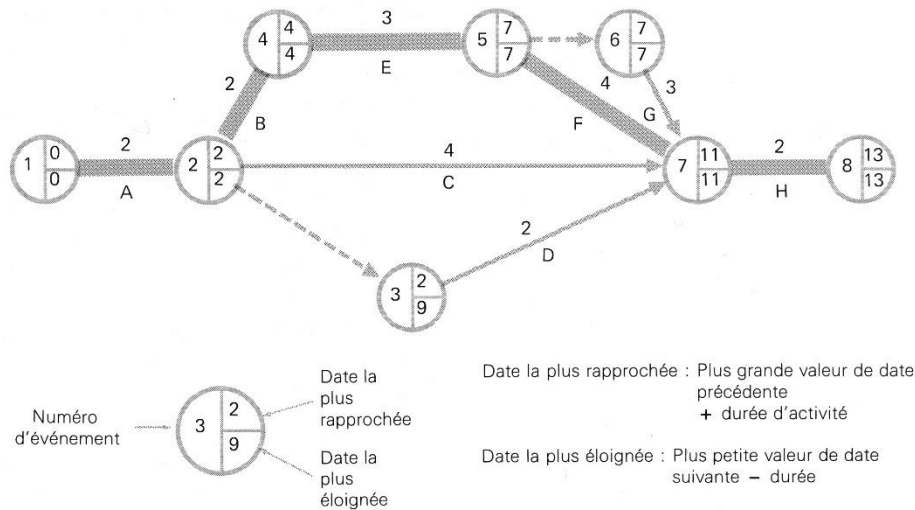


Figure 5 Temps d'activités dans le diagramme de PERT

Avant de créer un diagramme de PERT pour un projet de système d'information, il faut identifier les tâches et les temps d'exécution associés à chaque activité. Le temps de chaque activité représente la durée. Ensuite, il faut dresser la liste des activités en soulignant la place de certaines tâches devant en précéder d'autres et de certaines activités pouvant se dérouler simultanément. Cette information apparaît dans le diagramme de la figure 5 : les lignes représentent les activités de A à H et le chiffre sur chaque ligne est le nombre de semaines prévu par le chef de projet pour cette activité (c'est-à-dire sa durée). La longueur des lignes n'a pas d'importance.

Dans cet exemple, les événements 2-4, 2-7 et 2-3 ne peuvent pas commencer avant la fin de la tâche A, comme l'indique la position des événements sur la grille. La ligne pointillée qui relie les événements 2 et 3 a une signification spéciale. Aucune notion de temps n'est associée à la ligne pointillée et aucune identification de tâche (lettre ou nom) ne lui est attribuée. Lorsqu'il y a une dépendance entre deux événements, sans utilisation de temps ni de ressource, c'est une *activité fictive*. Avant que la tâche D de la figure 5 ne puisse commencer, l'événement 2 doit être terminé. De même, l'événement 5 doit avoir lieu avant que l'activité G ne soit entreprise. Les activités fictives permettent par conséquent de relier des activités parallèles. (Les activités C et D peuvent se produire en parallèle.)

L'étape suivante consiste à analyser le calendrier du projet. Nous voulons connaître (1) la date la plus rapprochée où l'événement peut débiter et (2) la date la plus éloignée où l'événement peut débiter sans affecter le calendrier global du projet. La date la plus rapprochée (désignée par ET = *earliest time*) est de 0 pour le premier événement. Pour les autres événements, c'est la *somme la plus élevée de la durée d'activité et du ET de tout événement lui précédant immédiatement*. Par exemple, quatre activités précèdent immédiatement l'événement 7. Le ET est calculé à partir du ET et de la durée de chaque événement précédent.

Activité	ET précédent	Durée	ET calculé
3-7	2	3	5
2-7	2	4	6
5-7	7	4	11
6-7	7	3	10

Comme l'événement 7 ne peut pas commencer tant que les événements précédents ne sont pas achevés, nous nous intéressons au plus grand facteur de temps, 11 dans ce cas. La figure 5 présente les ET pour toute la grille. Une notation fréquente présente le ET (*earliest time* = date la plus rapprochée) dans la partie supérieure droite du noeud et le LT (*latest time* = date la plus éloignée) dans la partie inférieure droite.

La date la plus éloignée (LT) représente la date limite à laquelle l'événement peut débuter *sans retarder* le projet. Pour déterminer cette date, il faut reculer dans la grille, à partir de la droite. La date la plus éloignée est la plus *petite différence entre le LT du dernier événement moins la durée de l'activité*. Par exemple, pour déterminer le LT de l'événement 7, le LT du noeud suivant, chiffre 8 dans la figure 5, présente la valeur 13. En soustrayant la durée, la date la plus éloignée est :  $13 - 2 = 11$ . Le nombre en bas et à droite du noeud 7 est 11.

Activité	LT suivant	Durée	LT calculé
2-4	4	2	2
2-7	11	4	7

Comme le plus petit LT est 2, cette date devient le LT pour l'événement 2.

Le *chemin critique* regroupe les activités auxquelles les responsables doivent porter une attention particulière : il s'agit d'événements qui doivent être commencés et terminés à temps et dont l'exécution ne doit pas dépasser la durée estimée, sous peine de retarder l'ensemble du projet.

La figure 5 présente le chemin critique en gras. Il a été obtenu en reliant les noeuds dont les ET et les LT sont égaux, ce qui signifie qu'aucun changement ou déviation n'est toléré. Autrement dit, il n'y a pas de *temps de battement*.

Le temps de battement associé à un événement peut être obtenu en soustrayant la durée et le ET du noeud initial du LT du noeud final. Par exemple, pour l'événement 3-7, le temps de battement est de sept semaines ( $11 - 2 - 2 = 7$ ). Autrement dit, l'activité 3-7 peut commencer n'importe quand entre la deuxième et la neuvième semaine et, si la durée de deux semaines est maintenue, le projet sera exécuté dans le délai fixé. Ce type de souplesse peut faciliter l'affectation du personnel.

Les diagrammes de PERT sont très utiles en conception, comme nous l'avons indiqué. Sur de grands systèmes, il est difficile de tenir à jour ces diagrammes pour présenter l'état exact de toutes les activités. C'est pourquoi de nombreux chefs de projet recourent à une combinaison de diagrammes : les diagrammes de PERT pour planifier le développement et pour illustrer l'interdépendance, et les diagrammes à barres pour présenter le calendrier.