

Séquence 7

Conduite d'un projet informatique

Durée indicative : 4heures

Dès que vous entendez parler un informaticien, vous êtes quasiment sûr d'entendre le mot « projet » : chef de projet, travailler sur le projet X, s'inscrire dans le projet X, avoir des interactions avec le projet X... En développement d'applications informatiques ou dans les réseaux, on retrouve la nécessité de gérer des projets

► Prérequis

Comprendre les différentes étapes de la gestion de projet.
Planifier un projet en utilisant le diagramme de Gantt et le graphe MPM.

► Contenu

1. Qu'est-ce qu'un projet informatique ?	2
1A. Pourquoi définir un projet ?	68
1B. Exemple des caractéristiques d'un projet	68
1C. Qu'est-ce qu'un projet réussi ?	69
2. Les étapes de la gestion de projet	4
2A. Étapes de la gestion de projet, méthode Merise	70
2B. Étapes de la gestion de projet, langage de conception UML.....	71
3. La planification de projet.....	7
3A. Exemple de planning.....	72
3B. Le diagramme de Gantt.....	73
3C. Le graphe MPM	75
3D. La méthode PERT	77
4. Le suivi des projets	11

Séquence 4

Conduite d'un projet
informatique

Page 1

1. Qu'est-ce qu'un projet informatique ?

Mais alors qu'est-ce qu'un projet ?

C'est un ensemble cohérent d'activités développées pour atteindre un objectif défini dans un certain domaine de l'organisation.

Il est placé sous la responsabilité d'une personne disposant de moyens définis (humains, matériels, financiers et organisationnels) et de contraintes (délais, budgets, autres éléments prédéfinis).

Exemples de projets :

- gestion des horaires flexibles et des repos compensateurs (domaine ressources humaines) ;
- profilage et suivi des clients (domaine commercial) ;
- refonte de la comptabilité pour qu'elle soit homogène avec celle d'un grand groupe financier (domaine comptable) ;
- sécurisation d'un Intranet (domaine sécurité des systèmes d'information).

Tout service ou société informatique fonctionne par projets.

1A. Pourquoi définir un projet ?

Il n'est pas possible de réaliser des évolutions du système d'information de l'entreprise au cas par cas sans se préoccuper de leurs interactions avec le système d'information en place. On va gérer l'ensemble de travaux à mener comme un tout, en constituant un projet de développement. Il est nécessaire de centraliser et coordonner ces diverses modifications et évolutions pour :

- analyser leurs interactions ;
- définir des priorités entre les projets par rapport aux objectifs globaux de l'organisation ;
- regrouper certains projets car ils représentent un même type d'intervention sur le système d'information ;
- assurer la rentabilité de la maintenance et l'évolution du système d'information ;
- organiser la gestion du personnel travaillant sur des projets liés au système d'information.

1B. Exemple des caractéristiques d'un projet

Pour définir un projet, on va spécifier quelques paramètres :

- son objectif ;
- son domaine fonctionnel ;
- le résultat attendu ;
- les contraintes ;
- les moyens ;
- les délais ;
- les critères spécifiques au projet.

Exemple : pour un projet relatif au développement du commerce électronique d'une entreprise, on peut retenir les éléments suivants :

- son objectif : augmenter nos ventes et les diversifier ;
- son domaine fonctionnel : gestion commerciale ;
- le résultat attendu : se faire connaître par d'autres clients, modifier notre image de marque... ;
- les contraintes : contraintes juridiques sur les ventes sur Internet, pas de personnel de l'entreprise suffisamment compétent pour développer le logiciel désiré ;
- les moyens : trois développeurs, outils de développement de sites de e-commerce, outils de sécurisation ;
- le délai : un mois.

1C. Qu'est-ce qu'un projet réussi ?

On peut déterminer quelques caractéristiques de la réussite d'un projet.

- Satisfaction des besoins des utilisateurs en maîtrisant les phases de développement, la réalisation et la mise en œuvre. La réponse aux besoins des utilisateurs est l'élément primordial de tout projet informatique.
- Apports positifs pour l'organisation : bénéfique, retour sur investissement, amélioration de l'organisation générale, avantage (ou rattrapage) concurrentiel.
- Coûts et délais maîtrisés.
- Maîtrise du projet tout au long de son déroulement : anticipation des risques et mise en place de mesures préventives, connaissance quasi-instantanée des dérives et prises de décisions adéquates.

Voici quelques problèmes qui peuvent exister lors de la durée de vie d'un projet et des exemples de réactions appropriées :

Risques	Causes	Réactions
Retard Glissement de budget	<ul style="list-style-type: none"> • Mauvaise estimation de la charge globale du projet • Mauvais suivi de projet • Mauvaise répartition des tâches • Facteurs humains (inexpérience, mauvaise entente entre personnes, mauvaise affectation,...) 	<ul style="list-style-type: none"> • Méthodes d'évaluation appropriées • Utilisation de tableaux de bord et d'outils normalisés • Choix et suivi des ressources humaines et techniques • Planification à surveiller et à adapter • Formation
Mauvaise qualité Non satisfaction des besoins	<ul style="list-style-type: none"> • Mauvaises spécifications • Mauvaises réalisations • Centrage exclusivement sur la technique 	<ul style="list-style-type: none"> • Associer étroitement les utilisateurs tout au long du projet et leur faire valider les étapes • Identifier les fonctions métier essentielles • Méthodes de tests et de recette • Limiter les innovations au sein d'un même projet
Abandon du projet (n'est pas obligatoirement une mauvaise décision)	<ul style="list-style-type: none"> • Peu de valeur ajoutée du projet • Aléas fonctionnels (modification des règles de gestion,...) • Aléas techniques (défaillance, retard) • Changement d'avis ou mauvaise description des besoins des utilisateurs • Coûts trop élevés 	<ul style="list-style-type: none"> • Ne jamais hésiter à abandonner un projet non rentable • Capitaliser l'expérience • Miser sur la réutilisation (modules logiciels par exemple) • Utiliser un progiciel à la place d'en développer un nouveau.

2. Les étapes de la gestion de projet

L'important dans un projet est de définir des étapes intermédiaires, qui sont un guide pour les divers acteurs comme une occasion de faire le point entre eux sur l'opportunité de continuer le projet. Chaque fin d'étape (ou de sous-étape) est un moment pour se mettre d'accord sur les actions correctrices qu'il est nécessaire de mener.

Ce découpage permet de dresser la liste des travaux à exécuter et des résultats attendus pour chaque étape, cela va aider le chef de projet à y voir plus clair, à faire des plannings et à évaluer les ressources et budgets nécessaires tout au long de la vie du projet.

Les diverses étapes dépendent du type de projet, des volumes (budgets et personnel) traités... Cependant, nous allons vous présenter deux types de découpages préconisés par des outils suivants : la méthode Merise et le langage de modélisation UML (unified modeling language).

2A. Étapes de la gestion de projet, méthode Merise

2A1. Le schéma directeur

Il fait l'état des lieux global du système d'information et fixe ses grandes orientations pour le futur.

2A2. L'étude préalable

Cette phase se limite à une partie particulière du système d'information, telle que la refonte ou la création d'une application.

Elle aboutit à une synthèse des scénarii retenus et à leurs estimations en termes de délais et de budgets.

2A3. L'étude détaillée

L'étude détaillée se consacre à un projet, tout en ne négligeant pas ses interactions avec le reste du système d'information. Elle décrit la solution retenue en la détaillant.

Le rapport de cette phase doit donner une vision externe du système (interface homme-machine, description des traitements réalisés...).

2A4. L'étude technique

C'est à partir de cette phase que les informaticiens vont devenir les acteurs principaux, sans négliger pour autant les diverses phases de validation par les utilisateurs.

Elle produit le document de base du développement de l'application : descriptions précises des structures de données et des traitements, normes techniques, indications de réutilisation...

2A5. La réalisation

Dans cette phase on va produire le logiciel en diverses étapes, allant de programmes séparés et testés individuellement, qui vont peu à peu s'inter-relier pour produire le logiciel global. Les premiers tests sont réalisés par les informaticiens alors que les suivants (et notamment lors de la phase de « recette ») nécessitent d'être réalisés par les utilisateurs sur la base de documents réels.

Lors de la recette, les informaticiens guident les utilisateurs, ce qui sert souvent de premier niveau de formation au logiciel.

2A6. La mise en œuvre

Il s'agit de l'installation et de l'intégration du logiciel dans le système d'information de l'entreprise. Elle concerne le paramétrage, la reprise ou l'alimentation des données et la modification d'interfaces.

2A7. La qualification et la livraison

C'est la dernière phase de la recette. Les tests seront réalisés dans l'environnement opérationnel (de production) de l'entreprise sans que ce dernier ne serve réellement d'environnement de production à ce moment. On va rédiger un bilan du projet selon divers critères de qualité.

La signature de ce rapport final par les représentants des utilisateurs (maître d'oeuvre, responsable du domaine concerné ou personnes détachées pour la recette) signifiera que le projet a été « livré ». Il peut alors rentrer en exploitation et le « client » atteste qu'il est conforme aux besoins qu'il était censé combler.

La mise en exploitation d'un projet ne signifie pas la disparition complète des informaticiens. Ils participeront ensuite à la formation et veilleront à sa maintenance (corrective ou évolutive).

2B. Étapes de la gestion de projet, langage de conception UML

UML se base sur un processus nommé itératif et incrémental, impliquant l'utilisateur, centré sur l'architecture technique et piloté par les risques. Il débute par des analyses séparées du niveau fonctionnel et du niveau technique qui se rejoignent ensuite.

2B1. Itérativité

Basé sur le même principe que le RAD (rapid application development) UML préconise un processus en spirale où une maquette, ou une petite partie du projet final, est développée à chaque stade. Le projet évolue en détaillant et complétant le résultat de la précédente production.

On peut ainsi présenter et faire valider une partie du produit fini aux utilisateurs à chaque étape.

De plus, la production d'une partie (prototype ou maquette) à chaque itération oblige les développeurs à intégrer l'environnement technique. Une maquette sert à montrer à l'utilisateur une partie importante du logiciel pour qu'il ait une vue d'ensemble et puisse prendre conscience des liens entre les diverses fonctionnalités. Un prototype, quant à lui n'a aucun objectif d'exhaustivité, il sert à expérimenter des cas précis, à se mettre d'accord sur des procédures et points de détails importants.

2B2. Orienté selon deux types d'utilisateurs

UML considère deux types d'acteurs parmi les utilisateurs.

L'utilisateur au sens classique, qui se servira des résultats du projet pour remplir le rôle qu'il a à tenir dans l'organisation. Il permettra de valider l'adéquation aux besoins.

L'utilisateur exploitant le système, correspondant à un rôle plus technique et opérationnel commun à la plupart des systèmes informatiques.

2B3. Piloté par les risques

Ce développement incrémental et la participation des utilisateurs au projet permettent d'éviter le risque majeur de non adéquation des besoins.

De plus, la prise en compte au plus tôt de l'architecture technique permet d'éviter les « études » trop théoriques, les développeurs étant directement confrontés à l'environnement de développement.

2B4. Basé sur un système en Y : la méthode TTUP

Le processus de développement est représenté sous forme de Y car on part tout d'abord sur deux branches séparées (le modèle fonctionnel et l'architecture technique) qui vont ensuite se rejoindre pour la réalisation ultime.

Le modèle fonctionnel permet de se focaliser sur le métier des utilisateurs. Cette partie de l'analyse définit les besoins réels du système en termes de métier.

L'étude de l'architecture technique recense toutes les contraintes techniques liées à l'existant et construit les bases de l'architecture technique intégrée dans le système d'information global. Lors de cette phase on réalise souvent un prototype.

Lorsque les deux branches ci-dessus se rejoignent on passe ensuite par des phases assez traditionnelles, à savoir :

- l'étude préliminaire qui conforte l'adéquation et l'interaction des deux branches ci-dessus ;
- l'étude détaillée, similaire à celle de la méthode Merise, mais sur la base de spécificités techniques plus détaillées ;
- le codage et les premiers tests faits par les informaticiens ;
- la recette.

3. La planification de projet

Pour faire un **planning**, la **première chose** est de faire la **liste des tâches relatives à un projet** donné **en évaluant le temps nécessaire à la réalisation de chacune** et **leur enchaînement** (telle tâche nécessite que telle autre soit terminée pour pouvoir commencer, la fin de telle tâche implique qu'une ou plusieurs autres peuvent commencer...).

3A. Exemple de planning

Commençons par un exemple simple de planning : comment se préparer le matin.

Supposons que le matin, avant de partir de chez vous vous ne fassiez que trois choses :

- vous préparer (se lever, se laver, s'habiller...) qui prend 20 minutes ;
- préparer votre café : 15 minutes ;
- prendre le petit déjeuner (céréales froides puis café) : 10 minutes.

De plus, supposons que vous ne puissiez pas prendre votre petit déjeuner avant de vous être préparé. Comment organiser vos tâches pour aller au plus vite... afin de faire sonner le réveil au dernier moment ? Pour cela nous allons réaliser un mini planning.

Il est évident ici qu'il faudra réaliser la tâche numéro **2** (préparer le café), puis la numéro **1** puis la **3**. Si le café met plus de **20** minutes à se préparer, on peut encore gagner du

temps en séparant la tâche numéro 3 en deux : céréales et café. On pourra ainsi, en mangeant les céréales réduire le temps d'attente du café.

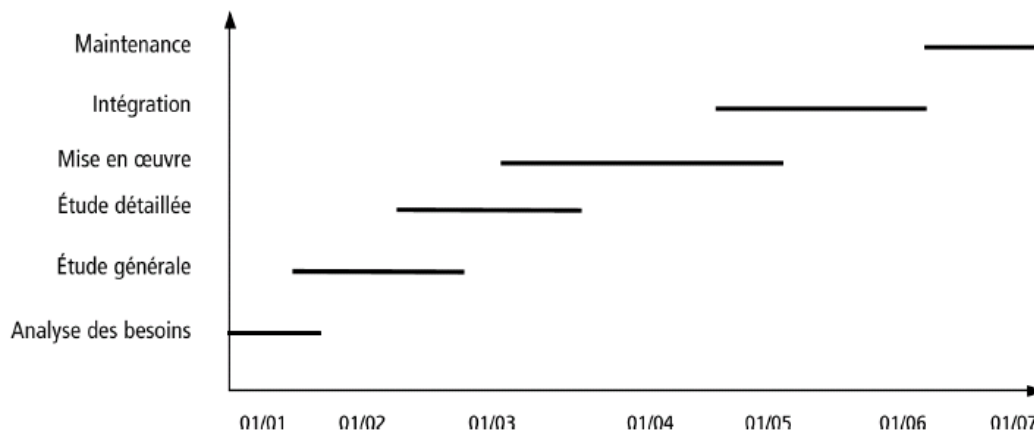
C'est ce type de planning et d'ajustement en fonction d'un aléa (ici le retard dans la disponibilité du café) que nous allons voir dans cette partie.

3B. Le diagramme de Gantt

Il répond à la question : « quelles sont les tâches qui doivent se réaliser à une date donnée ? ».

Ce diagramme se présente sous la forme d'un graphique avec la liste de tâches en ordonnée et le temps en abscisse.

Par exemple, si on suit les étapes préconisées par Merise pour un « petit » projet démarrant le 1er janvier d'une certaine année, on aura :



Séquence 4

Conduite d'un projet informatique

Page 7

On voit sur ce schéma que la mise en oeuvre peut commencer dès lors que les 2/3 (environ) de l'étude détaillée sont réalisés. En effet, l'étude détaillée s'étend du 15/02 au 01/04 et la mise en oeuvre débute vers le 10/03 pour se terminer fin mai.

Ce type de schéma va nous permettre, en cas de retard ou d'allongement d'une tâche, de décaler toutes les autres en conséquence.

De nombreux logiciels de planification permettent de réaliser un diagramme de Gantt, ils sont couramment employés dans le monde industriel, en gestion de projet. Comme « Gestion de Projet » ou Open Workbench dans le monde du libre, Project de Microsoft, Gnt Planning de SodeaSoft... Certains sont téléchargeables, je vous conseille d'aller les consulter. Ils permettent de réaliser la planification initiale mais surtout de l'adapter au cours du temps en fonction des réalisations concrètes et des aléas que se présentent.

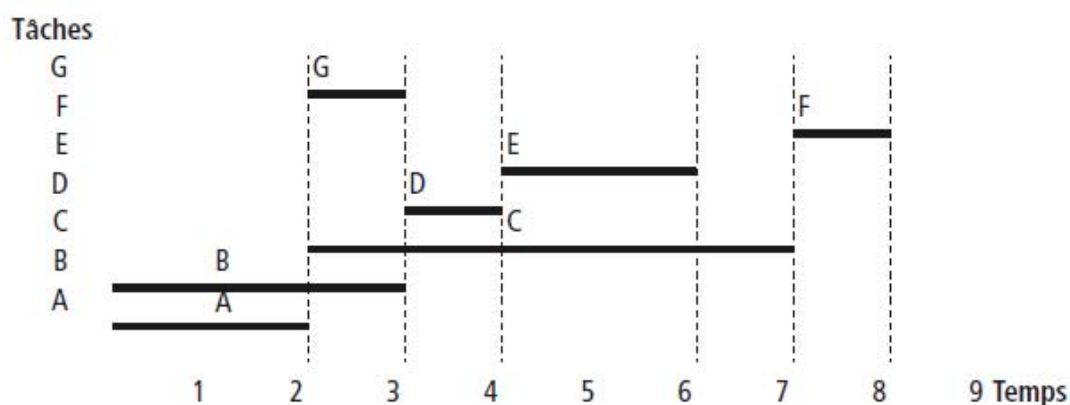
Exemple

Soit un projet pour lequel les diverses tâches (nommées ici A à G) ont été répertoriées ainsi que les relations d'antériorité entre elles. Nous allons réaliser le diagramme de Gantt correspondant.

Tâche	Durée en jours	Antériorité
A	2	-
B	3	-
C	5	A
D	1	B
E	2	D
F	1	E, C
G	1	A

La lecture de ce tableau nous informe que les tâches A et B peuvent commencer dès le début, qu'une fois que A est terminée on peut débiter en parallèle les tâches C et G et que la tâche F est obligée d'attendre que les tâches E et C soient terminées pour débiter.

Le diagramme de Gantt se présente alors ainsi :



Séquence 4

Conduite d'un projet informatique

Page 8

On voit ici que le projet se terminera à la fin du 8^e jour.

Les traits verticaux représentent les antériorités.

Ce diagramme nous permet de voir que seule la tâche G peut être retardée de 5 jours sans que le projet ne soit globalement retardé. On pourrait retrouver ceci par le calcul en faisant : 8 (date de fin de projet au plus tôt) - 1 (durée de G) - 2 (date de début au plus tôt de G, sachant qu'elle ne peut commencer que lorsque A est achevée). 5 jours est la marge de manoeuvre dont le responsable de projet dispose sur cette tâche (appelée la marge totale, voir ci-dessous).

De même à la lecture du diagramme on voit que les tâches A, C et F ne peuvent prendre aucun jour de retard sans retarder globalement le projet. Elles représentent ce que nous appellerons le chemin critique.

La tâche E peut prendre un jour de retard (on peut la translater d'une case vers la droite sans modifier le délai total de réalisation du projet). Mais si elle prend 3 jours de retard, le projet se terminera alors 2 jours plus tard (soit à la fin du 10^e jour). En effet le retard de 3 jours absorbe la marge de 1 jour qui était disponible et retarde donc le projet de 2 jours (3-1). Si cela était le cas E se situerait alors sur le chemin critique.

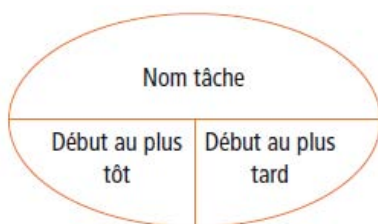
Le diagramme de Gantt est un bon instrument graphique pour la lecture des dates de début et de fin de tâches, il est relativement moins performant pour les prévisions en fonction des marges. Nous allons donc voir maintenant un outil un peu plus complet pour le calcul des marges : le MPM (méthode des potentiels métra)

3C. Le graphe MPM

3C1. Présentation et construction

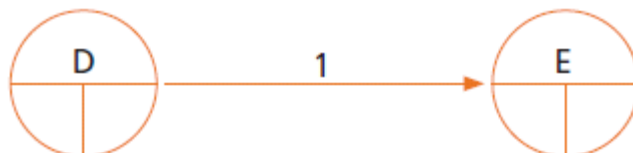
Cet outil se présente sous forme d'un graphe, il a donc des sommets, qui représentent les tâches, et des arcs (flèches) qui indiquent les relations d'antériorité.

Chaque sommet est présenté sous forme d'un rond avec le nom de la tâche, sa date de début au plus tôt, ainsi que sa date de début au plus tard.



On réalise tout d'abord le graphe en ne mettant que le nom des tâches et les flèches, sur lesquelles on porte la durée de la tâche.

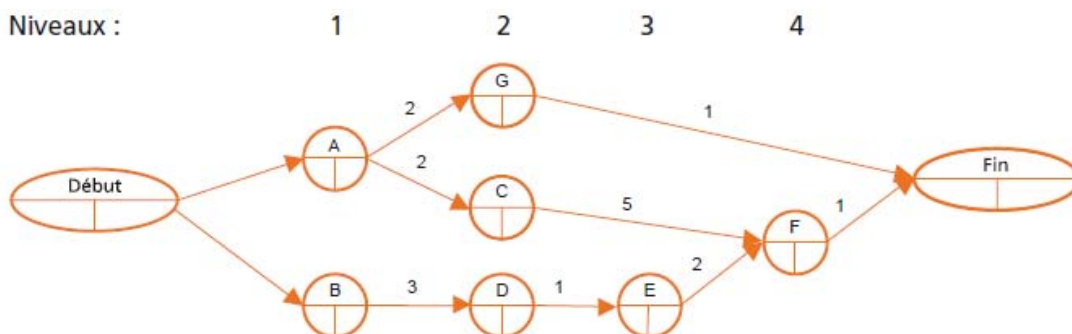
Si on reprend l'exemple ci-dessus, la tâche D, qui dure un jour et qui précède la tâche E, est représentée ainsi :



De plus, on dessine le graphe en tenant compte des niveaux. Le 1^{er} niveau représente l'ensemble des tâches qui peuvent se réaliser dès le début, le niveau 2 celui qui peut se réaliser dès qu'une tâche de niveau 1 est terminée...

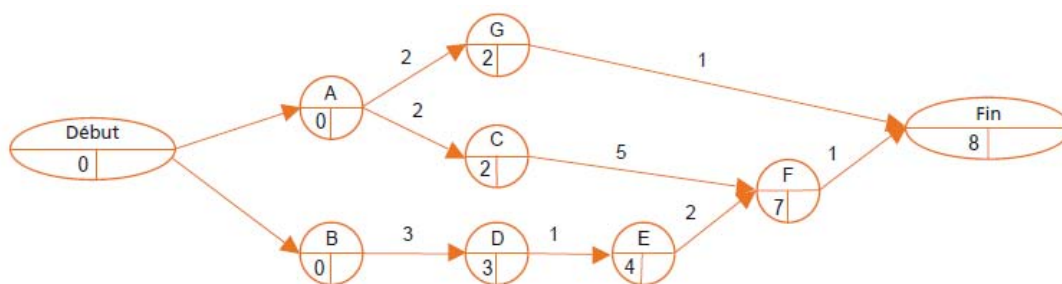
On rajoute deux tâches « début » et « fin » auxquelles sont reliées respectivement les tâches n'ayant pas d'antériorité ni de successeur.

On obtient donc pour le graphe suivant :



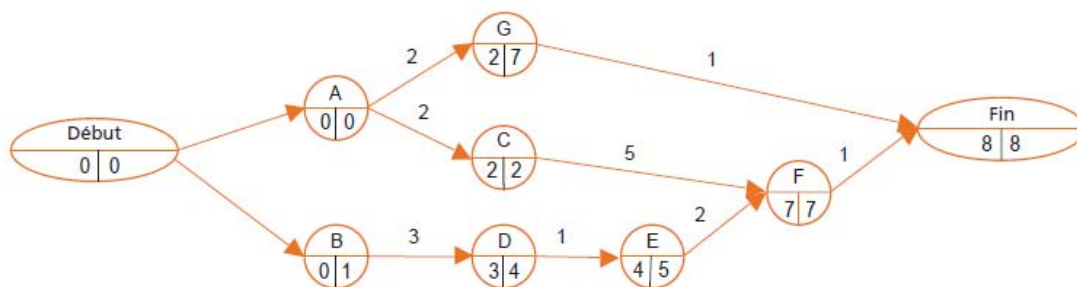
On s'occupe ensuite des dates de début au plus tôt. Pour cela on met 0 à la tâche de « début » puis, **pour chaque tâche, on additionne le début au plus tôt et la durée de la tâche précédente**, pour obtenir le début au plus tôt de la tâche courante. Par exemple pour E on a 4, soit 3 le début au plus tôt de D et 1 la durée de D.

Si une tâche a plusieurs prédécesseurs, on prendra la valeur la plus grande obtenue par chacune des flèches qui arrivent à la tâche. En effet la tâche doit attendre que tous ses prédécesseurs soient terminés, et donc, entre autre, le prédécesseur qui termine le plus tard. Par exemple pour F on a C et E comme tâches antérieures. Par le chemin qui provient de C, on obtient 7 (2 + 5) comme date au plus tôt. Par le chemin qui provient de E, on obtient 6 (4 + 2) comme date au plus tôt. **On prend donc 7 pour attendre la fin de C.**



Nous savons maintenant que le projet peut être réalisé, au plus juste, en **8 jours**, nous allons donc calculer les dates de début au plus tard de chaque tâche pour ne pas retarder le projet global. On va faire ceci comme pour les dates au plus tôt sauf qu'on va partir de la fin et prendre à chaque fois les données les plus faibles obtenues.

Pour cela on part de la « fin » dans laquelle on reporte 8, la durée globale, puis on en déduit de proche en proche les dates de fin au plus tard en se servant des durées de chaque tâche. Pour G cela nous donne 7 (8 - 1). **Si une tâche a plusieurs successeurs on prendra alors la valeur la plus petite obtenue**, comme A pour laquelle on a le choix entre 5 (7 - 2) si on considère G comme successeur et 0 (2 - 2) si on considère C comme successeur, le début au plus tard est donc 0.



3C2. Indicateurs d'ordonnancement

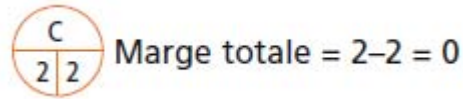
Ce graphe MPM nous permet, pour chaque tâche, de calculer aisément **la marge totale d'une tâche** qui **représente le retard qu'elle peut prendre sans retarder le projet global**.

La marge totale est obtenue en faisant, pour chaque tâche, la différence entre la date de début au plus tard et la date au plus tôt.

$$\begin{array}{|c|c|} \hline G \\ \hline 2 & 7 \\ \hline \end{array} \quad \text{Marge totale} = 7 - 2 = 5$$

La marge libre représente, pour chaque tâche du projet, la marge de manoeuvre disponible sans modifier la date de début au plus tôt de tous les successeurs. Elle se calcule donc par « date de début au plus tôt » de la tâche suivante moins durée de la tâche considérée.

Une tâche critique est une tâche dont aucun retard n'est possible sans affecter la durée globale du projet. Sur le graphe MPM on les repère par le fait que les deux dates sont égales.



Dans le graphe précédent, les tâches critiques sont donc **A, C et F**.

On appelle chemin critique la succession des tâches critiques. En général, sur le graphe MPM, on met en valeur ce chemin par des doubles flèches ou une couleur différente des flèches. Les tâches situées hors du chemin critique peuvent, dans une certaine mesure, être allongées ou retardées sans répercussion sur la durée totale du projet.

3D. La méthode PERT

Dans une première approximation, on peut dire que le graphe associé à la méthode PERT (program evaluation and review technic) est quasi identique au graphe MPM. Simplement, on va mettre le nom des tâches sur les arcs, suivi de la durée entre parenthèse. À la place du nom de la tâche (en haut du rond) on met un numéro de tâche.

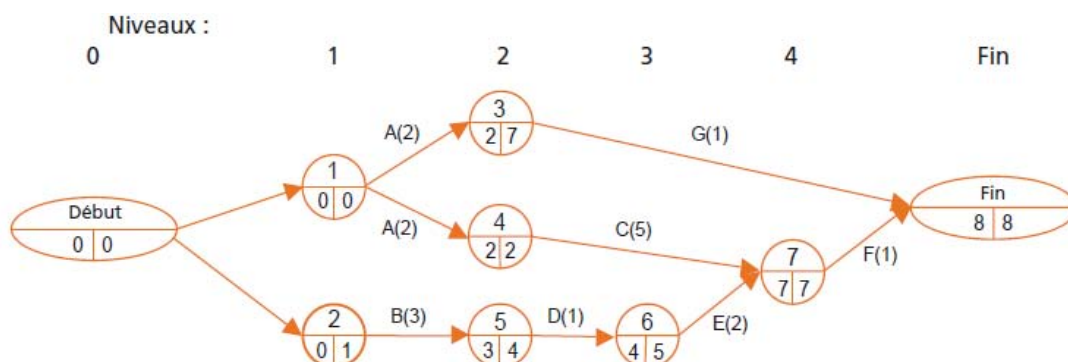
Il est important d'y faire apparaître le niveau, ce que nous avons déjà fait pour notre exemple de graphe MPM. Le niveau de chaque tâche est le fait qu'elle puisse être réalisée dès le début (niveau 1), ou après que les premières tâches soient terminées (niveau 2)...

Si on lit « verticalement » notre ancien graphe MPM, on voit que le niveau 1 est composé des tâches A et B qui n'ont aucun prédécesseur et donc peuvent commencer au début.

Le niveau 2 est composé des tâches G, C et D, qui ont toutes un prédécesseur de niveau 1. Le niveau 3 est composé de la tâche E, qui a comme seul prédécesseur D, de niveau 2. Le niveau 4 est composé de la tâche F dont le niveau le plus élevé parmi ses prédécesseurs (C et E) est 3.

Dans un graphe PERT la tâche fictive « fin » est toujours apparente, alors qu'on peut se passer de la tâche « début ».

Le graphe PERT de notre exemple se présente donc ainsi :



4. Le suivi des projets

Pour toutes les étapes que nous avons vues ci-dessus, l'essentiel n'est pas de prévoir toutes les opérations et leur planification, mais la maîtrise du déroulement du projet.

Pour cela le chef de projet doit suivre les conditions et l'état d'avancement du projet et agir par des actions correctrices au bon moment. De nombreux tableaux de bord l'aident pour l'alerter des dérives. C'est à lui de prendre les bonnes décisions au bon moment. Ces décisions sont fonction des risques à prendre ou à éviter, de la rapidité de leur mise en œuvre, des marges de manœuvre données à chaque acteur du projet, de la mise à jour des informations...

Exercice 1

Supposons que vous travailliez au service informatique d'un grand hypermarché avec une méthode de conception basée sur le langage UML et respectant les consignes de la méthode TTUP. Préciser, pour les exemples ci-dessous, la branche du processus TTUP (fonctionnelle ou technique) qui sera prise en compte.

1. L'hypermarché ouvre une section « agence de voyage », il veut créer un système d'information indépendant de celui de l'hypermarché, en pouvant assurer des échanges de données entre les deux systèmes d'information.
2. La section « agence de voyage » passe un accord de revente de certains voyages organisés avec une compagnie qui monte ces voyages. Des conditions particulières doivent être mises en place pour cette activité de revente.
3. La section « agence de voyage » décide d'ouvrir une rubrique de commerce électronique.
4. Le partenariat avec la compagnie précédente impose la mise en place d'un extranet avec des moyens techniques de synchronisation des deux systèmes.

Exercice 2

Dans une entreprise, on désire mettre en place un réseau local qui prendra en charge une soixantaine de postes clients. Le responsable informatique a dégagé les diverses tâches à réaliser pour mener à bien le projet. Il a quantifié leurs durées en jours de travail.

Référence tâche	Tâche	Durée en jours	Antériorité
A	Description détaillée des besoins	4	-
B	Détermination du budget	2	-
C	Recherche et choix des fournisseurs intervenant	6	A et B
D	Commande et réception du logiciel d'administration réseau	1	C
E	Commande et réception des équipements réseaux	4	C
F	Installation du logiciel d'administration réseau	1	D
G	Câblage	10	C
H	Formation de l'administrateur réseau	4	C
I	Mise à jour des postes au niveau matériel	3	E et G
J	Mise à jour des postes clients (logiciel et sécurité)	2	I
K	Intégration et mise au point serveurs et clients	5	F, H et J

Remarques

Le câblage est réalisé par une société extérieure qui a proposé deux devis : un premier de 3 000 € avec un délai de réalisation de 6 jours, et un second (celui retenu) de 2 000 € en 10 jours.

Pour la formation de l'administrateur réseau, l'organisme de formation qui la dispense n'organise que deux sessions par mois, les prochaines auront lieu le lundi 05/03/12 et le lundi 26/03/12 (soit exactement 3 semaines après la première session).

L'entreprise travaille cinq jours par semaine, du lundi au vendredi.

Au niveau du personnel, tous les travaux sont réalisés par des fournisseurs externes ; seul un administrateur réseau est nouvellement embauché pour assurer son administration (en plus de travaux de maintenance informatique). Il doit impérativement être présent pour les tâches A, C et K de ce projet.

1. Représenter graphiquement l'enchaînement des tâches de ce projet.(Gantt).
2. À l'aide du graphe MPM, déterminer le nombre de jours que va durer la mise en place du réseau. Donner la durée en semaine.
3. Quelle est la tâche, sur le chemin critique, qui dure le plus longtemps ? Dans le cas où l'entreprise voudrait réduire le temps global d'installation du réseau, que proposez-vous pour cette tâche ?

4. Sachant que le choix des fournisseurs (tâche C) auquel participe l'administrateur réseau, se termine le jeudi 1 mars 2012 et que ce dernier est inscrit pour la formation du lundi 5 mars, pourra-t-il suivre cette session ? La participation à ce stage du 05/03/12 va-t-elle retarder le projet ?
5. Si la session de formation du 05/03/12 affiche complet et que l'administrateur ne peut assister qu'à la suivante, quelle conséquence cela aura sur la durée globale de mise en place du réseau ?
6. Pour aborder, comme sur un calendrier, la planification des tâches en termes de dates-précises, quelle autre représentation (que le MPM) semblerait plus adaptée ?
7. Quelles sont les marges totales des tâches D, E et G ?
8. Lors de la réalisation concrète de ce projet, alors que l'entreprise est en train de réaliser la mise à jour des postes au niveau matériel (tâche I), on constate que les tâches ci-dessous ont pris du retard :
 - suite à l'absence du principal décideur, la description détaillée des besoins a duré 7 jours ;
 - retard de réception des équipements réseau de 2 jours ;
 - allongement d'une journée pour l'installation du câblage.Quelle est l'incidence de ces trois retards sur la date de disponibilité générale du réseau ?