



LE SYSTEME D'INFORMATION

1) Définition du Système d'Information :

C'est l'organisation (homme, organigramme, règle de gestion, procédures, ...) et les outils (applications informatiques, méthodes, règles de calcul, matériels, ...) qui permettent aux acteurs d'une entreprise de communiquer, de traiter et de stocker des informations. Il représente l'ensemble des éléments participant à la gestion, au stockage, au traitement, au transport et à la diffusion de l'information au sein d'une organisation ou entreprise.

En informatique et en télécommunications et plus généralement dans le monde de l'entreprise, le terme **Système d'Information (SI)** possède les significations suivantes :

✓ Un ensemble organisé de ressources (personnel, données, procédures, matériel, logiciel, ...) permettant d'acquérir, de stocker, de structurer et de communiquer des informations sous forme de textes, images, sons, ou données codées dans des organisations. Selon leur finalité principale, on distingue des **Systèmes d'Information** supports d'opération (traitement de transaction, contrôle de processus industriels, supports d'opérations de bureau et de communication) et des **Systèmes d'Information** supports de gestion (aide à la production de rapports, aide à la décision, ...).

✓ Un système ou sous-système d'équipements, d'informatique ou de télécommunication, interconnectés dans le but de l'acquisition, du stockage, de la structuration, de la gestion, du déplacement, du contrôle, de l'affichage, de l'échange (transmission ou réception) de données sous forme de textes, d'images, de sons, et/ou, faisant intervenir, du matériel et des logiciels.

Le système informatique ne comprend que l'outil informatique (matériel et logiciel) qui mécanise, en partie ou en totalité, le système d'information.

2) Composition d'un Système d'Information

2-1) Composition classique

Dans un **Système d'Information** classique d'une grande entreprise, on trouve généralement :

✓ un **ERP - Enterprise Resource Planning** (en français : **PGI** pour **Progiciel de Gestion Intégré**) - qui intègre tous les systèmes informatisés permettant d'aider le travail dans l'entreprise ;

✓ des systèmes appelés spécifiques (non standards, développés sur mesure, que l'on ne trouve pas sur le marché,...), où l'on trouvera davantage d'applications dans les domaines de la facturation, de l'aide à la production, ou de fonctions annexes.



La proportion entre **ERP** et systèmes spécifiques est très variable d'une entreprise à l'autre.

Dans les **ERP**, on trouve des modules de différents domaines.

Il est fréquent qu'une entreprise soit équipée de plusieurs progiciels différents selon ses domaines d'activité. Dans ce cas, les progiciels ne sont pas totalement intégrés comme dans un **PGI**, mais interfacés entre eux ainsi qu'avec des applications spécifiques. On trouvera par exemple des applications de :

✓ **CRM - Customer Relationship Management** (en français : **GRC** pour **Gestion de la Relation Client**) : regroupe toutes les fonctions permettant d'intégrer les clients dans le système d'information de l'entreprise

✓ **SCM - Supply Chain Management** (en français : **GCL** pour **Gestion de la Chaîne Logistique**) : regroupe toutes les fonctions permettant d'intégrer les fournisseurs et la logistique au système d'information de l'entreprise

✓ **HRM - Human Resource Management** (en français : **GRH** pour **Gestion des Ressources Humaines**)

✓ **PDM - Product Data Management** (en français : **SGDT** pour **Système de Gestion de Données Techniques**) : fonctions d'aide au stockage et à la gestion des données techniques. Surtout utilisé par les bureaux d'études.

2-2) Tendances actuelles

Le domaine des **Systèmes d'Information** a certes une forte composante technologique et informatique. Mais c'est seulement un aspect de ce domaine qui est en fait beaucoup plus vaste. Il s'agit de concevoir comment circule et est stockée l'information de façon efficace et cohérente pour toutes les activités d'une entreprise, d'un réseau d'entreprises, d'une administration publique, des relations entre entreprises, citoyens, gouvernements...

Le champ est vaste et concerne tous les domaines des activités humaines. Malgré cette ampleur, ce domaine a son unité scientifique, construit autour de concepts, de constructions abstraites et concrètes, de composants de méthodes notamment qui sont indépendantes des activités concernées. Sans doute, un des maîtres mots de ce domaine des **Systèmes d'Information** est-il celui de modèle accompagné de celui de modélisation.

Par conséquent, dans les entreprises actuelles, le **Système d'Information** tend à s'orienter vers des ensembles plus globaux, l'information traitée par l'homme étant une connaissance à gérer.

En complément du **SI** classique, on a donc une **Ingénierie des connaissances** qui s'articule autour des deux composantes suivantes, que l'on peut retrouver dans chaque domaine d'activité de l'entreprise :

✓ la **Gestion de Contenu** (en anglais : **Content Management**) , destinée à gérer les informations brutes à transformer en **connaissances** mieux structurées ;

✓ la **Gestion des Accès**, c'est-à-dire la gestion des flux et des protocoles d'échange dans les réseaux de télécommunications internes ou partagés avec les partenaires.



2-3) Autres composants possibles

D'autres composants peuvent être inclus dans un système d'information :

- ✓ Bases de données de l'entreprise,
- ✓ Applications métiers,
- ✓ Infrastructure réseau,
- ✓ Serveurs de données et systèmes de stockage,
- ✓ Serveurs d'application,
- ✓ Postes de travail informatique,
- ✓ Dispositifs de sécurité.

3) Le management de projet du SI :

3-1) Les évolutions du développement de systèmes informatiques

De nos jours, les organisations ou entreprises ne veulent plus s'engager dans des développements informatiques très longs et très coûteux. Elles ne sont plus intéressées par des réalisations déjà obsolètes à la livraison. De plus en plus, elles se tournent vers des solutions clés en mains, mais cela n'est pas possible quand il s'agit de mettre en œuvre des solutions destinées à offrir des avantages spécifiques à l'entreprise.

Intuitivement, nous sentons que les méthodes traditionnelles pour le développement d'applications informatiques ne sont plus adaptées au climat économique dans lequel nous vivons :

- ✓ Les temps de développement sont trop longs. Lorsque le système informatique est enfin livré l'environnement de l'entreprise a lui déjà largement évolué. Si le système correspondait aux besoins initiaux, il ne répond plus aux besoins actuels. Il est déjà obsolète avant d'être utilisé. L'analogie peut être faite avec l'embauche d'un salarié qui prendrait plusieurs années. Les cycles de vie d'outils de management sont devenus trop courts pour permettre un temps d'adaptation de plusieurs années.
- ✓ Les applications livrées ne répondent pas aux besoins des Utilisateurs, parce que les Utilisateurs n'ont pas participé suffisamment à l'élaboration de leur outil. C'est un peu comme si l'équipage d'un avion n'avait pas contribué à l'aménagement de leur appareil. Les équipes modernes doivent intégrer la logique du marché, sinon le client en personne.
- ✓ Les personnes chargées du développement d'applications informatiques sont trop rarement consultées par les analystes et les concepteurs. Le développeur a trop peu d'influence sur l'architecture du produit. Cela revient à la situation où une entreprise devait construire un bâtiment tout en se servant de plans inutilisables. Une équipe responsable et décloisonnée doit prendre en compte les connaissances spécifiques de tous ses membres.
- ✓ Enfin, quand un produit connaît des difficultés de fonctionnement, personne n'est capable de diagnostiquer le problème, dans la mesure où personne ne possède une vue complète de l'outil depuis le système opératoire jusqu'à son application dans l'entreprise. Les apports individuels à la réalisation du système sont trop fragmentés et l'environnement de fonctionnement trop fragile. Tout comme une voiture livrée avec un certain nombre de défauts connus (le moteur coupe, les freins ne fonctionnent plus si on utilise la radio ou les essuie-glaces se mettent en marche lorsque l'on descend les vitres).



Il faut admettre que chaque entreprise a ses spécificités et tout particulièrement l'informatique, étrange mélange de science et d'artisanat. Cependant, une chose est certaine, la réalisation d'applications informatiques doit s'accélérer pour rattraper la vitesse à laquelle les technologies et les marchés évoluent.

3-2) Méthodologie type de développement de systèmes informatiques

Le développement des projets du Système d'information se fait, généralement, par prototypage. Du prototypage au produit généralisé, on adopte la démarche EQUIPAGE pour répondre au besoin des Utilisateurs en respectant les étapes ci-dessous.

- ✓ Conception/prototypage.
- ✓ Industrialisation - MPP.
- ✓ Généralisation.
- ✓ Utilisation.
- ✓ Bilan.

3-3) Une nouvelle approche de développement de systèmes informatiques

Les approches de l'ingénierie concourante, des équipes pluridisciplinaires, des plateaux projets et de conception guidée par le marché sont devenues des pratiques standards pour accélérer le développement. Il est temps de revoir ces approches dans l'informatique, et pourtant, les méthodes traditionnelles basées sur la logique du séquentiel continuent à dominer dans l'informatique. Par conséquent, elles se présentent comme un frein à l'approche de développement d'applications rapides.

Le développement rapide d'application est une approche qui n'impose pas qu'une étape soit atteinte avant de démarrer l'étape suivante. Tous les acteurs du projet sont fortement impliqués à toutes les étapes. L'expression de besoins peut évoluer d'une manière itérative par un processus de prototypage, de valorisation de fonctions et de priorité d'objectifs. Le projet est découpé en phases. Chaque phase vise la livraison d'un maximum de fonctions prioritaires. Néanmoins, le développement rapide d'application n'était pas une approche tout à fait acceptable pour l'industrie informatique, car il lui manquait une méthodologie. Ce n'est plus le cas.

3-4) La méthodologie dynamique de développement de systèmes informatiques

La méthodologie dynamique de développement de systèmes informatiques est une méthodologie non-propriétaire de développement d'applications rapides. Elle représente la synthèse de l'expérience d'un nombre important de fournisseurs et d'Utilisateurs. Elle offre un canevas pour la réalisation et l'entretien de systèmes informatiques de gestion par l'emploi d'une démarche de prototypage. La méthode s'adresse à toutes les parties prenantes du système, aussi bien les Utilisateurs, que les responsables de projet, les responsables de la qualité et les réalisateurs.

Un principe fondamental de l'approche dynamique est le fait que rien n'est construit parfaitement du premier coup, mais que 80% de la solution est réalisable en 20% du temps qu'il prendra pour une solution complète.



3-5) Pourquoi une méthodologie dynamique de développement ?

Les approches traditionnelles pour le développement informatique essaient d'atteindre la précision du premier coup. Dans la démarche classique et séquentielle, l'étape suivante ne peut pas commencer avant que l'étape précédente soit complètement terminée et validée. En pratique, cela engendre énormément d'investissement en temps et en effort pour passer du stade de 80% de réalisé à l'achèvement complet, en respectant le principe que l'on ne doit pas revenir sur une étape précédente. Pourtant, les besoins ne sont que très rarement spécifiés correctement du premier coup. Ce qui se passe dans la maille c'est le retard du projet. On passe un temps important pour retourner aux étapes « terminées » et pour corriger les erreurs dans le travail déjà validé.

Dans la méthode dynamique le droit de revenir sur les étapes précédentes fait partie de la démarche itérative. Par conséquent, l'étape en cours ne doit pas avancer au-delà du point nécessaire pour démarrer l'étape suivante, car on peut y revenir dans une prochaine itération. Le principe est que les besoins auront probablement évolué de toute façon, et donc des dépenses supplémentaires auront été gaspillées !

Des applications réalisées selon la méthode dynamique visent les besoins actuels de l'entreprise. Or, l'approche traditionnelle a souvent eu tendance à préconiser une solution pour toutes les possibilités. Un système réalisé avec la méthode dynamique répond mieux aux véritables besoins de l'entreprise, est plus facile à tester, et a plus de chance d'être utilisé.

3-6) Qu'est-ce que c'est la méthodologie dynamique de développement ?

La méthodologie dynamique de développement couvre toutes les étapes de la réalisation d'un système informatique pour un projet informatique qui doit être réalisé rapidement. La plupart des méthodes informatiques n'adressent qu'une seule activité telle que l'analyse et la conception, ou le pilotage de projet. La méthode dynamique prend en compte le cycle de vie entier d'un système informatique avec les outils et les contrôles nécessaires pour assurer son succès.

Le cycle de vie de la réalisation basé sur une démarche de prototypage itérative et incrémentale est associé avec les fournitures issues des activités de gestion et de suivi technique. La méthode dynamique spécifie des catégories de prototypes et leur application la plus efficace durant le projet.

Le développement rapide d'applications doit être considéré comme une approche légitime et sérieuse. La méthode dynamique contient une description des activités de management de projet nécessaire pour maîtriser l'évolution rapide d'un système. Le principe du découpage en phases est l'une de ces approches, mais il en existe d'autres selon l'activité. Par exemple, la structure et les rôles au sein de l'équipe doivent être adaptés à la démarche. Les équipes qui respectent la méthode dynamique doivent recevoir des objectifs clairs et réalisables dans un temps donné. Toutes les fournitures sont associées à des critères de qualité pour s'assurer d'atteindre les objectifs.

Les structures des équipes comprennent plusieurs rôles pour les Utilisateurs. On attend que certains Utilisateurs s'intègrent dans l'équipe de développement afin d'effectuer des décisions journalières de manière plus rapide et plus efficace. D'autres Utilisateurs assument les rôles de clients externes plus traditionnels. L'apport efficace du temps et de connaissances de la part des Utilisateurs est un des éléments clés de la méthode dynamique.



Conforme à la volonté de sérieux, la méthode dynamique détaille l'approche préconisée pour la gestion de la configuration technique. La gestion de la configuration est importante pour maîtriser l'état de produits très évolutifs dans un projet conçu avec la méthode dynamique. Une approche adaptée pour les tests et le contrôle de la qualité est aussi élaborée.

On ne retrouve pas dans la méthodologie dynamique de développement un mandat pour ce qui concerne les techniques d'analyse, de conception et de programmation. En revanche, les techniques de l'analyse et de la conception structurées, et l'analyse orientée objets sont dessinées. Il est vrai que certaines organisations peuvent préférer utiliser leurs techniques habituelles dans le mesure du possible. De telles techniques peuvent être évaluées par rapport aux spécificités de la méthode dynamique à l'aide des critères spécifiés dans le guide de la méthodologie dynamique de développement.

La méthodologie dynamique de développement décrit également les conditions de développement idéal avec une « check-list » pour aider une entreprise dans l'évaluation et la spécification d'un tel environnement. Parmi d'autres aspects traités en détail dans la méthode dynamique se trouvent :

- ✓ l'estimation des coûts et des durées,
- ✓ des paramètres pour évaluer le succès des projets,
- ✓ des recommandations pour la conduite de la sous-traitance,
- ✓ enfin, la méthodologie dynamique de développement offre des conseils pour l'introduction du développement rapide d'applications au sein d'une organisation.

3-7) Un exemple de méthodologie dynamique de développement

Cet exemple de méthode pour livrer des systèmes rapides et souples pour répondre à des besoins urgents dans un monde qui évolue rapidement, repose sur :

3-7-1) Des grands principes

- 1) Associer les utilisateurs à toutes les étapes.
- 2) Autoriser l'équipe à prendre des décisions.
- 3) Satisfaire les besoins du « business » avant tout.
- 4) Développer de manière itérative.
- 5) Assurer la réversibilité des modifications.
- 6) Manager le produit, et non la tâche.
- 7) Intégrer les tests à toutes les étapes.
- 8) Délivrer des fournitures fréquentes.
- 9) Estimer par rapport aux besoins globaux.
- 10) Evaluer les risques par rapport aux fonctions.
- 11) Spécifier un plan de référence global.
- 12) Promouvoir des relations coopératives.

3-7-2) Quatre types de prototypes

- ✓ « Business » - vérifier la compréhension.
- ✓ Ergonomie - vérifier la praticabilité.
- ✓ Performance - vérifier les dimensions.
- ✓ Capacité - vérifier l'architecture.



3-7-3) Trois étapes de prototypage

- ✓ Explorer les besoins.
- ✓ Améliorer la fonctionnalité.
- ✓ Consolider.

Une approche pour développer les prototypes doit : identifier, planifier, créer et capitaliser.

3-7-4) Un relationnel et une équipe

- ✓ Associer les utilisateurs dès le début : construire des relations collaboratrices, se mettre d'accord sur les rôles.
- ✓ Autoriser l'équipe à prendre des décisions les ressources, les priorités, les motifs.
- ✓ Localiser l'équipe à un seul endroit là où le système sera utilisé.
- ✓ Employer des compétences clés : communication et « business », technique et management.
- ✓ Déterminer tous les rôles clés : managers, Utilisateurs, conseillers, chargés techniques et développeurs.

3-7-5) Un canevas

- ✓ L'étude de faisabilité : est-ce que la méthodologie dynamique de développement est adaptée ?
- ✓ L'étude du « business » : la définition des fonctions, les besoins en information et les fournitures prioritaires.
- ✓ Le modèle fonctionnel itératif les fonctions clés spécifiées en détail.
- ✓ La conception et le développement itératifs : l'apport des Utilisateurs et les priorités mises à jour.
- ✓ La mise en œuvre : l'installation et l'entraînement terminés.
- ✓ Tout est itératif.

3-7-6) Des outils et des techniques de développement rapide d'application

- ✓ Permettre un développement rapide.
- ✓ Associer les utilisateurs dans la conception.
- ✓ Permettre un développement itératif.
- ✓ Construire une ergonomie excellente.
- ✓ Supporter la gestion de la configuration.
- ✓ Faciliter la réutilisation fournir une documentation intégrée.

3-7-7) Des approches préconisées

- ✓ Structurées et orientées objets.
- ✓ Atelier logiciel recommandé.
- ✓ Le comment est plus important que l'outil.

3-7-8) Un management

- ✓ Planifier le projet de façon itérative : viser les fournitures clés, estimer à partir des fonctions les plus importantes, employer le concept de découpage en phases.
- ✓ Tester durant toutes les étapes du projet : tester les fonctions « business » en priorité, se concentrer sur les fonctions clés, répéter et enregistrer. Les concepteurs n'effectuent pas de tests.
- ✓ Capitaliser sur les causes des réussites et des échecs et apprendre à ne pas culpabiliser.
- ✓ Se baser sur des mesures fiables.