



Le projet **DELIRE**  
**Développement par Equipe**  
**de Livrables Informatiques**  
**et Réalisation Encadrée**

**GP8 – Indicateurs et tableaux de bord**



## Je dis ce que je fais, je fais ce que j'ai dit

Un projet où on plonge directement dans la phase de réalisation se révèle généralement un désastre. Néanmoins, vous continuez et vous continuerez à avoir des ingénieurs qui adorent cela. On les appelle généralement des pompiers pyromanes : ils adorent se trouver dans une situation difficile et lutter contre des incendies, mais ils oublient simplement qu'ils sont eux-mêmes à l'origine de cette situation.

A contrario une phase de spécification et de structuration qui n'est pas suivie d'une phase d'implémentation puis d'une phase de convergence n'est pas d'un grand intérêt : ce n'est qu'une étude papier, élégante probablement, mais durant laquelle on n'a pas identifié les problèmes intrinsèques.

Le bon projet

1. On a compris les besoins du client
2. On a clairement défini l'architecture fonctionnelle et l'architecture logique de la solution
3. On a défini quand et comment on allait réaliser la solution
4. On implémente ce qui a été défini
5. On fait converger le système

Et le miracle s'accomplit : le produit marche et satisfait le client. Le tout sans avoir dépasser le délai ou le budget.

Utopique ? Pas toujours.



Mais ne rêvons pas ni ne faisons de sophisme: ce n'est pas parce qu'on a fait un vrai travail d'architecture fonctionnelle et d'architecture logique que la phase d'implémentation puis la phase de convergence vont se révéler de véritables parties de plaisir : il va y avoir des problèmes. On voit mal sinon pourquoi on aurait construit un PM (Plan de Management) et un PGR (Plan de Gestion des Risques).

Il va donc falloir traiter les problèmes au fur et à mesure qu'ils vont se révéler. Mais dire qu'ils vont se révéler signifie surtout qu'il faut savoir les identifier. C'est la notion de suivi de projet



## **Principes de base du suivi de projet**

Pour identifier un problème, il faut avoir un détecteur de problème : un indicateur

Un indicateur n'est pas une mesure. Beaucoup d'étudiants pensent que présenter des courbes de mesure est suffisant. Il n'en est rien.

Un indicateur c'est :

1. Une prévision
2. Une mesure
3. Une comparaison entre la prévision et la mesure.

Et c'est parce que la mesure diverge de la prévision que l'indicateur va devenir système d'alerte.

Ce qui est important n'est pas que l'indicateur soit simple (même si c'est souhaitable), mais qu'il soit prévisible et mesurable

Une solution c'est

1. Une architecture : c'est l'objectif de la phase de spécification;
2. Les moyens de la mettre en œuvre : ils sont définis lors de la structuration, et sont déployés en phase de réalisation et en phase de convergence

Les indicateurs sont définis lors de la phase de structuration

1. Des indicateurs de qualité dans le PQ
2. Des indicateurs de délai, de coût et de charge dans le planning et le budget
3. Des indicateurs de risques dans le Plan de Gestion Des Risques
4. Des indicateurs de problèmes humains dans le Plan de management.

Et ils sont suivis lors des phases de réalisation et de convergence.

On assurera, dans le cadre de la gestion du changement, une attention particulière aux indicateurs de suivi des tâches innovantes du projet

On assure, par le suivi de projet, la convergence tout au long de l'implémentation. Objectif majeur : on ne veut pas avoir de problème majeur détecté en fin de projet, en tests de solution.

On va donc ramener les indicateurs globaux, tels qu'ils pourront être validés sur le produit en fin d'assemblage, en un ensemble d'indicateurs sur chacun des composants de la solution.

De même que l'architecte, une fois le poids du système global défini, va décomposer partager ce poids entre les différents sous-systèmes, de même l'architecte informatique, une fois le temps de réponse d'une commande défini, va décomposer ce temps entre les différents modules de la solution logicielle.

Conformément aux notions d'ingénierie (voir G3 – De l'ingénierie), le point clé du suivi est de détecter les erreurs et de les traiter au plus tôt. En fait, pas exactement. Le suivi c'est vérifier qu'on agit conformément au plan, cela ne garantit pas la qualité du plan. On ne détecte donc pas les erreurs mais les non-conformités

Le suivi n'est pas une tâche de créativité. On a défini des indicateurs, on a maintenant les yeux rivés sur le tableau de bord pour identifier toute dérive. Aux yeux





d'un chef de projet informatique, c'est assez peu valorisant. Voilà pourquoi le suivi est généralement bâclé et pourquoi les projets informatiques coûtent 3 fois plus cher que prévu.

Dans le suivi de projet, il n'y a pas de fautif : Une fois identifier la dérive, il faut construire ensemble la solution pour revenir dans les clous.

Si on commence à stigmatiser les fautifs, on va engendrer un état d'esprit où les acteurs vont chercher à camoufler leurs erreurs : lorsqu'on les découvrira, il sera trop tard. Ne l'oublions pas, le point clé est d'identifier au plus tôt.

Mais le plus beau plan du monde ne peut offrir que ce qu'il a. Si le plan est foireux, la solution le sera aussi.

Garantir la qualité du plan d'exécution est de la responsabilité de l'architecture.

Architecturer c'est modulariser pour isoler et limiter la propagation des problèmes  
C'est vrai

1. Pour la conception du produit
2. Pour la conception du process, du planning, du budget
3. Et pour la constitution des équipes

Dans ces 3 domaines, les règles de la C.O.O. (Conception Orientée Objet) demeurent valides (forte cohésion, faible couplage)

La modularisation va permettre en passant de un problème global à M problèmes locaux, de faire passer la complexité du projet de  $N^2$  à  $N \cdot \log(N)$

Dans ce cadre, l'approche systémique ne remet pas en cause la démarche cartésienne : il est toujours nécessaire de décomposer. Au contraire, en conduisant à concevoir des modules plus cohérent (tant vis-à-vis du produit que du process ou des ressources) l'approche systémique va permettre de pouvoir définir des indicateurs plus pertinents pour chacun des sous-systèmes, des ensembles et des composants du système final.

Pendant longtemps, les problèmes généraux étaient révélés lors de revue. Ces revues commençaient lors des tests d'intégration, lorsqu'on vérifiait que l'assemblage des différents composants fonctionnait.

La grande différence entre l'approche PLM et les anciens programmes est que la revue se fait sur un produit virtuel et peut être planifiée très tôt dans le programme : on peut vérifier l'assemblage des grands ensembles même si les pièces mécaniques qui vont les constituer n'ont pas encore été définies.

Ceci va permettre d'identifier au plus tôt les problèmes qu'on ne découvrait qu'en phase terminale du projet, avec toutes les conséquences que cela entraînait.



## Identifier les problèmes au plus tôt.

Quand j'avais 10 ans, avec mon cousin Yann,  
nous avons joué avec des allumettes en forêt



La situation a un peu dégénéré mais pas question d'appeler nos parents respectifs...

... Enfin, pendant 10 minutes



Un petit feu mal maîtrisé peut rapidement devenir un gigantesque incendie

Voire une catastrophe nationale (Californie, août 2007).



La bonne solution :

1. Pouvoir identifier au plus tôt les problèmes
2. Savoir comment réagir en cas de problème.

Dans un projet, les problèmes peuvent être des problèmes de Coût, de Délai, de Charge ou de Qualité

Savoir ce qu'on doit réaliser et obtenir, et donc identifier les divergences, est le rôle :

1. Du PQ (Plan Qualité)
2. Du Planning
3. Du Budget

Voir PSS9 - PQ (Plan Qualité) et PSS14 – Planning et Budget de la phase de réalisation

Savoir comment réagir en cas de problème est le rôle :

1. Du PGR (Plan de Gestion des Risques).
2. Du PM (Plan Management)

Voir PSS8 - PGR (Plan de Gestion des Risques) et PSS10 - PM (Plan Management)

Dans l'exemple qui nous intéresse (départ d'un incendie) on aura mis en place des systèmes d'alerte. Le plus basique étant un détecteur de fumée dans un appartement (c'est plus difficile à installer dans une forêt). .

Dans un projet, de façon à pouvoir identifier le problème qu'il va falloir adresser, on met en place des détecteurs de problèmes : ces détecteurs s'appellent des indicateurs.

Mais Antoine, il n'y aura peut-être pas de problème dans notre projet DELIRE ?

Un projet est une expérience unique, une prise de risques, qui se construit dans un monde flou, sans avoir toutes les données du problème et en faisant appel à l'innovation.

En d'autres termes, dans projet, en phase de réalisation, il y a toujours des problèmes à résoudre :

1. Certains projets, qui auront bien anticipé ces problèmes et mis en place un mécanisme de suivi rigoureux arriveront à contourner ces obstacles
2. D'autres projets, qui auront rêvé leur avenir sous un soleil radieux, et auront vécu dans l'insouciance du lendemain, se dirigeront irrémédiablement vers une débâcle sévère.



## Un indicateur.

Les deux difficultés sont

1. Que suivre ?
2. Comment prévoir ?

Comment choisir les bons indicateurs ?

1. Un projet est un subtil équilibre entre trois contraintes : le Coût, le Délai et la Qualité.
2. D'autre part, un projet est fait avec des humains, qui sont capables de produire un certain effort en fonction du temps.

On distingue donc deux types d'indicateurs

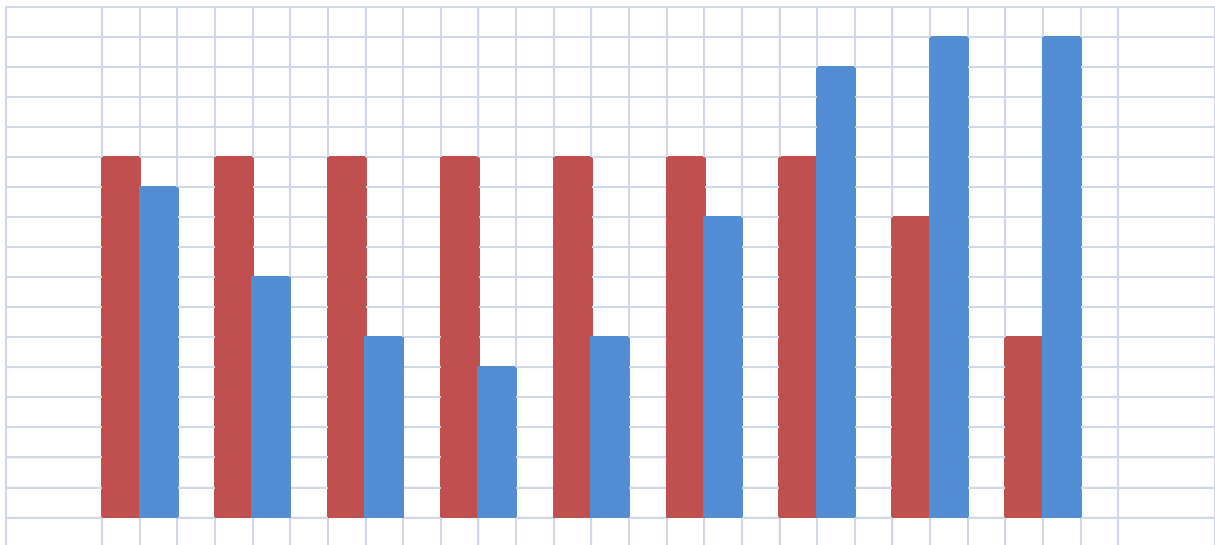
1. Des indicateurs de Coût, Délai et Charge d'une part
2. Des indicateurs de Qualité d'autre part

## Indicateurs de Coût et de Charges

Vos prévisions de couts sont, dans le projet GALERE, confondues avec vos prévisions de charges, puisque la seule dépense dans ce projet concerne des heures de travail. Il s'agit du Budget.

Je vous conseille de suivre vos coûts et vos charges

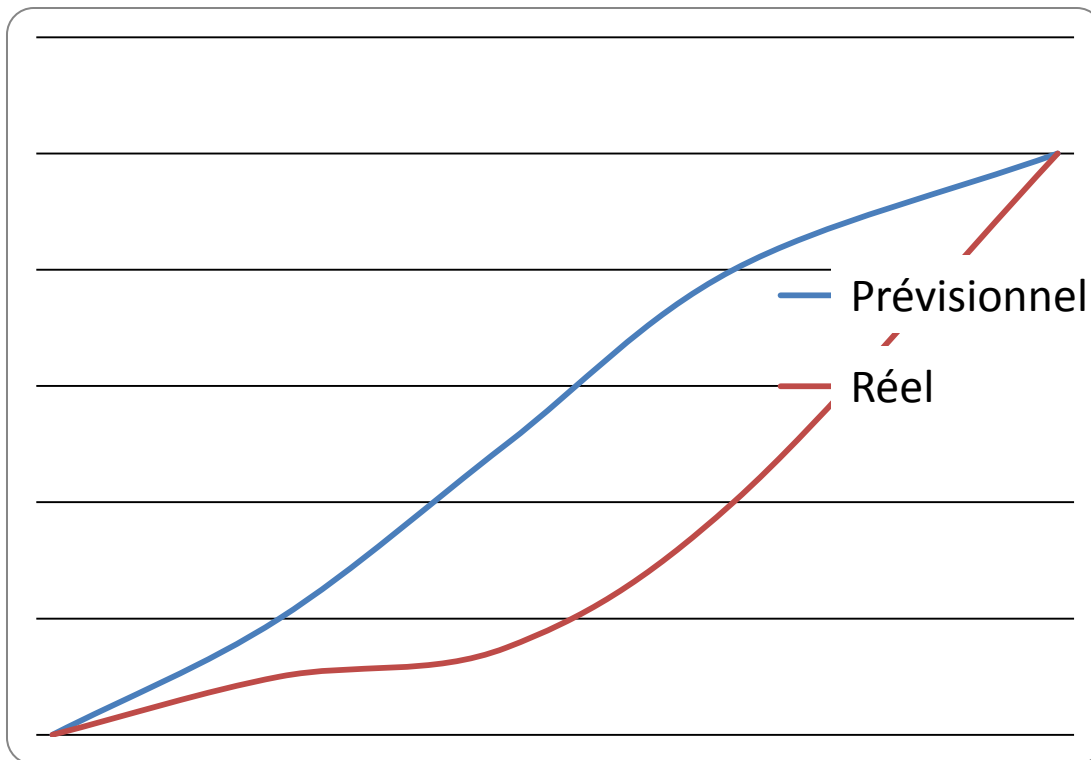
1. De façon unitaire, par exemple chaque semaine



2. De façon cumulative, depuis le début du projet ou de la phase. Cela permet de bien mettre en évidence des tendances







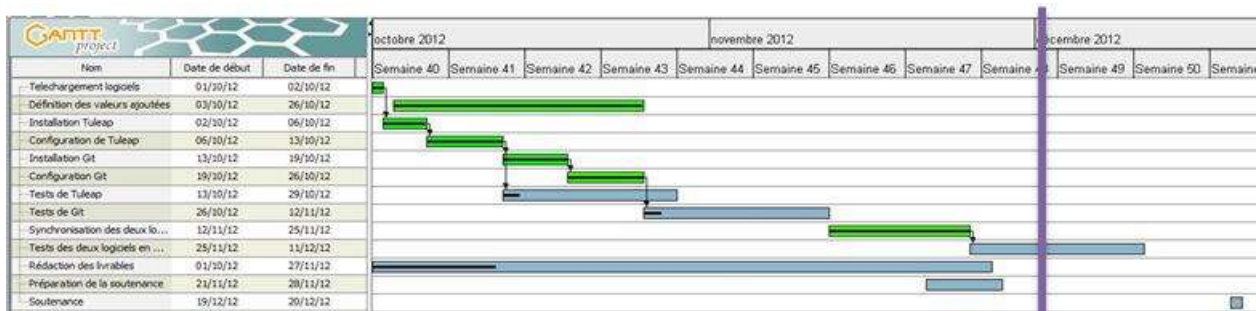
### **Indicateurs de Coût et Délai dans le projet DELIRE**

Dans le projet DELIRE, je vous propose de rester raisonnable dans ce que nous allons mettre en place.

Suivre sur le planning prévisionnel l'avancée de chaque tâche



en mettant en évidence la date du jour me semble largement suffisant.



Comme vous serez amenés à modifier régulièrement le planning, je vous recommande d'en garder toutes les versions. Plus spécifiquement je vous conseille de garder un planning par semaine. Ceci vous aidera, en phase de Post Mortem, pour refaire l'histoire du projet.





### Le délai, principal indicateur

Une dérive est si vite arrivée. Personne n'a envie d'avouer qu'il est en retard, qu'il ne tient pas son budget, ou qu'il n'atteint pas ses objectifs de qualité.

Le retard est en général le premier indicateur de dérive : il traduit souvent une augmentation de budget, et un problème technique qui se traduira par un problème de qualité.

Etre très ferme sur la tenue des dates de jalons est une aide à la tenue du budget et de la qualité. Toute annonce de dérive sur le planning doit être immédiatement traitée.

Le délai, premier indicateur de problème

1. Les problèmes Qualité ne peuvent être décelables au début que par un expert
2. Le fournisseur peut bouffer ses marges pour masquer un problème de budget
3. Par contre, un décalage de timing se voit immédiatement

Un problème de dérapage peut signifier:

1. Je n'ai pas ou je n'ai plus les ressources : Délai
2. Je me suis trompé sur le dimensionnement de l'effort : Coût => Délai
3. Les entrées qui m'ont été fournies ne sont pas de bonne qualité, je suis obligé de refaire : Qualité => Coût => Délai
4. Je ne maîtrise pas la technique : Qualité => Délai => Coût

Un dérapage peut dégénérer

1. Ressource non disponible ensuite
2. On rate une fenêtre d'opportunité (On liait la sortie à un événement à fort retentissement)

Un problème de dérapage peut néanmoins n'être qu'un problème de délai : problème temporaire de ressource (panne, maladie...)

Ne pas accepter un dérapage sans l'avoir compris





## Indicateurs Qualité

Définir des indicateurs (Prévision, mesure et comparaison) de Délai est relativement aisé

Définir des indicateurs de Coût est déjà plus complexe. Comment estimer le coût d'une tâche ?

C'est en forgeant qu'on devient forgeron. C'est en planifiant qu'on devient expert en planification. T'es bien gentil, Antoine, mais pour l'heure nous sommes des néophytes. On fait comment ?.

Lorsque vous dimensionnez une tâche, de coding ou de test par exemple, essayez d'intuiter le ou les paramètres dimensionnant. Puis, en vous appuyant sur votre (faible) expérience, définissez le coût probable de la tâche. Ce qui est surtout important est de mesurer, lors de son exécution, le coût réel de la tâche, pour vous permettre de vous construire une expérience.

Définir des indicateurs de qualité est franchement compliqué.

Le défaut classique est de faire du qualitatif, autrement dit de la langue de bois :

1. Notre produit sera performant
2. Notre produit sera convivial

Pour pouvoir avoir un indicateur, il est nécessaire de comparer, et donc d'avoir quelque chose de mesurable. Il va falloir transformer le qualitatif en quantitatif

Notre produit sera performant va devenir : l'exécution des principales commandes se fera en moins de 1 seconde.

Notre produit sera convivial va devenir : pour exécuter une commande, le nombre d'interactions devra être inférieur à 7, et plus de 80 % des interactions seront des clicks.

C'est pourquoi j'ai insisté et je continue à insister sur le fait que les objectifs de votre PQ doivent être chiffrés et mesurables

On distingue deux objectifs, et donc deux sources d'indicateurs, dans le PQ

1. Les objectifs Produit (ce qu'on réalise)
2. Les objectifs Process (comment on le réalise)

On peut très bien faire un produit exceptionnel en travaillant comme des gourgandins. Ou respecter scrupuleusement la méthodologie préconisée et sortir une grosse bouse. La vérité m'oblige à confesser que, dans la réalité, ceux qui travaillent comme des cochons ont tendance à générer une daube.

Les objectifs Produits sont souvent des objectifs chiffrés. Mais la difficulté sera de les mesurer avant qu'il ne soit trop tard

Ainsi, l'objectif : « l'exécution des principales commandes se fera en moins de 1 seconde » qui ne pourrait être mesuré qu'en tests d'intégration, va devenir :

1. Exécution de la couche 1 en moins de 0.3 secondes
2. Exécution de la couche 2 en moins de 0.5 secondes
3. Exécution de la couche 3 en moins de 0.2 secondes

Ce qui pourra être contrôlé en Unit tests



Les objectifs Process sont souvent des procédures à exécuter :

1. Standard de livrables
2. Standard de coding
3. Fiche de livraison de code...

On est donc plutôt dans un indicateur de type tout ou rien. L'important est de les suivre sans discontinuer tout au long du process.

### **Le choix des indicateurs**

Choisissez des indicateurs simples quand vous le pouvez

1. Simples à comprendre
  - a. Pas de bac + 8 pour comprendre ce qui est mesuré
  - b. Et celui qui va rapidement vous l'imposer est le grand chef de la boîte, qui veut tout de suite pouvoir comprendre et interpréter
2. Simples à mesurer
  - a. Ceci dépend des outils à votre disposition
  - b. Si vous avez des outils qui vous permettent de calculer automatiquement le nombre de lignes de code d'un composant et le nombre de lignes de commentaires, c'est un indicateur simple à déployer
  - c. Si vous devez calculer ces nombres de lignes à la main, cela devient tout de suite pénible.
3. Simples à interpréter
  - a. Il ne doit pas y avoir d'interprétation possible sur la façon de mesurer
  - b. Ni sur la façon de recevoir l'indicateur
  - c. Ainsi, pour un indicateur de performance, explicitez rapidement que la mesure passe à rouge si le temps mesuré est supérieur de plus de 10% à un temps de référence

A titre de comparaison, regarder ce qu'on mesure sur une voiture : la vitesse, la jauge d'essence, la température du moteur... Simples, faciles à comprendre, faciles à mesure, et à comparer avec des objectifs.





## La notion de tableau de bord.

La réussite de votre projet repose sur le choix de bon indicateurs.



ni trop



ni trop peu

Pour piloter efficacement, un nombre limité d'indicateur

1. Quelques valeurs que l'on suit, pour anticiper
2. Et des indicateurs tout-ou-rien d'alerte.



La présentation des indicateurs se fait au travers d'un tableau de bord, qui doit tenir sur une page.

1. Communication vers un public: simple et ciblé
2. Pas d'effort pour le lire: toujours identique

Des grands classiques

1. Utilisation des ressources (en %)
2. Tâches réalisées/tâches planifiées
3. Jalons
4. Date de fin initiale - Date de fin finale
5. Avancement en délai (%)
6. Nombre de tâches terminées/nombre de tâches prévues
7. Nombre de changements
8. Nombre de risques réalisés

Des indicateurs et des couleurs simples

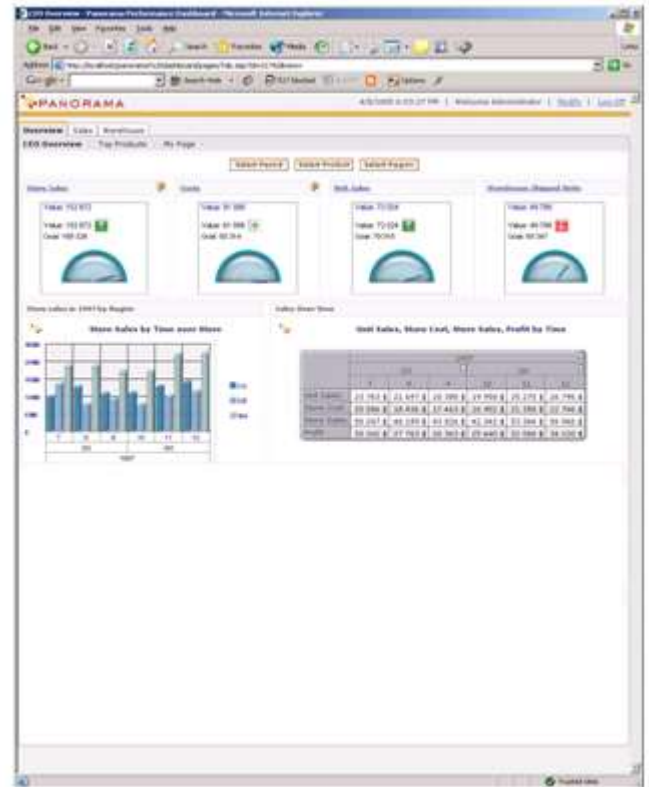
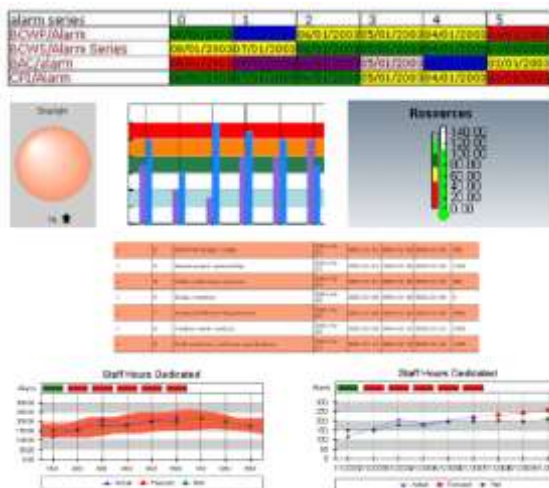
1. Blanc: S.O.
2. Vert : OK
3. Orange à surveiller



#### 4. Rouge : à traiter

Ayez des tableaux de bord différents à destination:

1. Du client
2. De la Direction Générale
3. Des équipes métiers
4. De l'équipe projet



#### Comment estimer une tâche en cours

Estimer l'avancement d'une tâche terminée est facile : 100%

Estimer l'avancement d'une tâche non commencée est facile : 0%

Mais que dire pour une tâche en cours.

Si une tâche est dimensionnée pour pouvoir durer 40 heures et produit 10 composants, que dire lorsqu'elle a déjà duré 30 heures et produit 2 composants : 75%, 20% ?

Il existe des méthodes, type l'avancement physique pour vous permettre de faire une estimation du délai et du coût prévisible.

Mais, dans le projet DELIRE, je vous conseille de laisser chacun estimer le % d'avancement de sa tâche. C'est le plus simple et le moins source de conflit.



### Définition du tableau de bord

L'idéal eut été de définir très tôt dans le projet un tableau de bord à suivre pendant tout le projet. Mais dans la réalité, il est difficile de figer un tableau de bord quand les spécifications et l'architecture physique ne sont pas encore définies.

Mon conseil est de définir un tableau de bord qui vous permette de faire le suivi du projet tant en phase de réalisation qu'en phase de convergence

Ceci signifie :

1. Qu'il y a des indicateurs qui resteront désespérément à 0% durant la phase de réalisation
2. Et des indicateurs qui seront inexorablement à 100% durant la phase de convergence

